CS4786/5786: Machine Learning for Data Science, Spring 2015
4/24/2015: Assignment 3: EM, graphical models

Due Tuesday May 5th at 11:59pm on CMS. Submit what you have at least once by an hour before that deadline, even if you havent quite added all the finishing touches CMS allows resubmissions up to, but not after, the deadline. If there is an emergency such that you need an extension, contact the professors.

You may work in groups of one up to four, for the same reasons as have been explained in previous assignments. Please ensure that each member of the group can individually defend or explain your groups submission equally well.

Students may discuss and exchange ideas with students not in their group, but only at the conceptual level. As discussed in previous assignments, we distinguish between "merely" violating the rules and violating academic integrity. The way to avoid violating academic integrity is to always document any portions of work you submit that are due to or influenced by other sources, even if those sources werent permitted by the rules.

1 EM Primer

The EM algorithm can be used whenever we have a maximum likelihood estimation problem with hidden variables. The idea is as follows: Imagine we have a graphical model parameterized by $\theta \in \Theta$. (Generally θ will be a vector.) Say O represents the set of observed variables and H the set of hidden variables. The goal of MLE is to find the parameter that best explains the observation. Specifically, we would like to obtain:

$$\theta^* = \underset{\theta \in \Theta}{\operatorname{argmax}} \log P_\theta(O)$$

However, in many cases, even if we are given θ, taking derivatives of or simply even writing down $P_\theta(O)$ is hard, but if the set of hidden variables H were only revealed, calculating (derivatives of) $P_\theta(O, H)$ is simple. The idea in the EM algorithm is to start with an initial guess of the best parameter θ^0, and based on this iteratively refine the parameter we obtain, as follows: At iteration i, based on the old parameter θ^{i-1} we calculate a distribution over the hidden variables given the observed variables O by setting

$$Q^i(H) = P(H|O, \theta^{i-1})$$

The above is called the E-step. Next, given this distribution over the hidden variables we find the subsequent θ^i as follows:

$$\theta^i = \underset{\theta \in \Theta}{\operatorname{argmax}} \sum_H Q^i(H) \log(P_\theta(O, H))$$

The above is the M-step. We iteratively repeat these two steps till convergence (or till we are bored). Now notice that the E-step is simply doing inference in a graphical model. For the M-step, the term inside the log is now a joint probability that factorizes over our graphical model; this should make the optimization in the M-step easy (or at least easier).

2 The apple-doesn't-fall-far-from-the-tree (ADFFFTT) model

In this question, we explore the possibility of making our own machine learning algorithm/model based on a generative story. We will specifically use a modified version of the apple-doesn't-fall-far-from-the-tree generative story we used as one of the examples in class for clustering.

Here is the generative story: in our orchard there are two types of apple trees, the green-apple trees and the red-apple trees. Starting from day one, each day a new tree sprouts as follows. The parameters of our model are: a mixture distribution π over the $K = 2$ types of trees, a single variance parameter $\sigma^2 > 0$ and locations $\mu_1, \mu_2 \in \mathbb{R}^2$ of the *first* tree locations for each of the tree types. The way trees are generated by nature is given by the following procedure:

1. On each round t, nature picks $c_t \in \{1, 2\}$ by drawing independently from the mixture distribution π. Red apples are picked with probability $\pi(1)$, so green apples are picked with probability $\pi(2) = 1 - \pi(1)$.

2. Next, given the type of apple tree c_t, the location of this new tree x_t is distributed according to the normal distribution with covariance $\sigma^2 I$ and mean given the location of the previous apple tree of type c_t. That is, if c_t was the red-apple type, then the location of this new red-apple tree is close to the location of the most recently sprouted red-apple tree, or specifically, normally distributed with a spherical covariance centered at the most recently sprouted red-apple tree.

3. We need to specify the deal for the first red-apple tree (RAT) and the first green-apple tree (GAT), because neither of them have previous tree of the same type, by definition. So: the location of the *first* RAT (on whichever day it first occured) is distributed according to the normal distribution with mean μ_1 and covariance $\sigma^2 I$, and similarly the location of the first GAT is distributed according to the normal distribution with mean μ_2 and covariance $\sigma^2 I$.

After N days you visit the orchard for the first time and only notice the sprouted trees. Can you guess which sprouted trees are of the same types?

3 Questions

*Q1 (***Graphical model).**

1. **Write down the parameters of the model we just specified.**

2. We want to write down a graphical model (Bayesian Network) that explains the ADFFFTT generative process. A direct approach would be to have observed variables $X_1, \ldots, X_N$ for the locations of sprouted trees and hidden variables $C_1, \ldots, C_N \in \{1, 2\}$ that specify the kind of each tree. **You don't need to write down this graphical model, but you do need to tell us in your writeup, explaining how you arrived at your answer: how many edges would such a model have?** We don't need exact number; big-O notation in terms of variable N will suffice.

3. Now let's try to simplify the graphical model by introduce some more carefully-chosen hidden variables. Specifically, let us introduce the superscripted and subscripted variables $X_1^1, \ldots, X_N^1$ and $X_1^2, \ldots, X_N^2$ where variable X_t^c indicates the location at time t (including the tree sprouted on day t) of the most recent tree of type $c \in \{1, 2\}$. Specifically,

   - Given $C_t = 1$, $X_t^1 \sim N(X_{t-1}^1; \sigma^2 I)$, $X_t^2 = X_{t-1}^2$ and $X_t = X_t^1$
   - Given $C_t = 2$, $X_t^2 \sim N(X_{t-1}^2; \sigma^2 I)$, $X_t^1 = X_{t-1}^1$ and $X_t = X_t^2$

   Notice that in the above, given C_t and X_t^1 and X_t^2, the observed variable X_t is *deterministically* chosen to be either X_t^1 or X_t^2 based on whether $C_t = 1$ or $C_t = 2$ respectively. Also notice that X_t^1's only depend on C_t and $X_{t-1}^1$and similarly for class 2.
   Draw the graphical model for this formulation, and write down how many edges it contains, explaining how you derived this.

4. **Give the conditional probability of each of the variables in the graphical model given its parents.**

5. We would like to perform inference on this graphical model given the parameters, specifically we want to compute for every t the probability[1] $p_\theta(X_t^1, X_t^2, C_t | X_1, \ldots, X_N)$. **Using either variable elimination or message-passing, write down how you would compute** $p_\theta(X_t^1, X_t^2, C_t | X_1, \ldots, X_N)$ **given the observations and some parameter** θ**.** We are not looking for the exact form/calculations, or in other words, your answers can be computations in terms of $p_\theta(\text{Variables}|\text{Parents})$ and need not involve the actual form of these distributions and the actual integrals or sums and the Gaussian distributions. (like we did for HMM's in lecture).

*Q2 (***EM Algorithm).**

Since in the previous question we asked you to give us the exact forms of the conditional probabilities for each node, for this question, you can just write these probabilities as $P_\theta(\text{some variable}|\text{some other variable})$ (i.e., don't expand them any further: so you're only talking about the graphical structure, not the specific distributions.) We would of course like to find/learn the parameters, based on the observations, via EM. We'll take the unconventional approach of starting with the M-step, for reasons explained later.

For the M-step what we would like to do is find, given the Q distribution,

$$\theta^i = \underset{\theta \in \Theta}{\text{argmax}} \sum_H Q^i(H) \log(P_\theta(O, H)) \tag{1}$$

- **Write down the set of hidden variables** H **and the set of observed variables** O **for this problem/graphical model.**

[1] We are using little-p here to indicate that this "probability" could be a density function. But for you, feel free to use integrals, summations, whichever — we aren't asking you to be particular about when dealing with a discrete vs. continuous distribution.

- We would want to simplify $\log(P_\theta(O, H))$ in Equation (1) before we attempt to perform the M-step. The nice thing about graphical models is that the joint distribution factors according to the graph. Use this fact and simplify $\log(P_\theta(O, H))$ to be of form:

$$\log(P_\theta(O, H)) = \sum_{t=1}^{N} \text{Some terms}_t$$

What are the terms in the above summation? Give them in your writeup. Again we don't care about the parameterization, so write your answers in terms of $p_\theta(\text{Variables}|\text{Parents})$ rather than expanding such terms out further.

- Using the above and swapping summations as in lecture, we can write

$$\sum_H Q^i(H) \log(P_\theta(O, H)) = \sum_{t=1}^{n} \sum_H Q^i(H) \text{ Some terms}_t$$

In the above, for every t, note that Some terms_t only involves a (small) subset of the terms of the graphical model. Hence, by marginalizing over terms not appearing in Some terms_t, we can write, for each t,

$$\sum_H Q^i(H) \text{ Some terms}_t = \sum_{H_t} Q^i(H_t) \text{ Some terms}_t$$

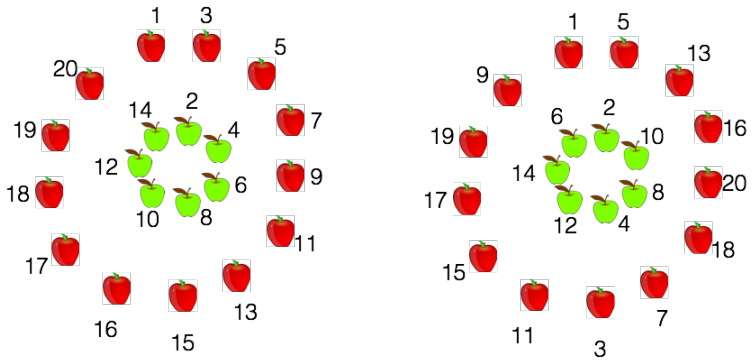
(In the above, the "Some terms" on the left- and right-hand sides of the equation are indeed the same.)

Write down what these variable sets H_t's are. Again, we don't require you to expand out $p_\theta(\text{Variables}|\text{Parents})$ terms.

The point of this question: The reason we did all this, including "starting (our analysis) with the M-step" is: based on the above simplification for the M-step, we see that in the E-step we don't need the entire $Q^i(H)$,but only terms $Q^i(H_t)$'s. Aren't you proud?

On the next page are *bonus* questions; solving them will provide you with "karma" points that we will factor into our final letter-grade determinations. If you wish to attempt them, your solutions should go in the same assignment writeup document as your answers to the questions above.

*Q3 (***Bonus Question).** Say we are given two sets of observations of 20 points each. The two sets are depicted in the figure below, with the following hidden information revealed: the sprout order, and which trees are RATs and which are GATS.



Both sets of data points have trees in the exact same locations. However, the order in which they occur in the left side figure is different from the ones on the right. Now if we were to run the developed EM algorithm for data set one and data set two, we would get very different answers. This is unlike k-means or single-link clustering algorithms where for both data sets we get exactly the same clustering. Why is this, and what would be the likely cluster assignment for the data in the left and in the right? Explain your answer

*Q4 (***Bonus Question [for the slightly brave ones!]).**

- Write down and derive the full inference algorithm/pseudocode for the E-step.
- Write down the full derivation and update form for the M-step.
- Implement the derived EM algorithm and submit your code (please include comments indicating the main steps).