IITM-CS6730 : Probabilistic Graphical Models | Release Date: Feb 28, 2020

Assignment 3 | Due Date : Mar 9, 23:59

Roll No: PGM20XYZ | Name: Markov

Collaborators (if any):

References (if any):

- Use LaTeX to write-up your solutions (in the solution blocks of the source LaTeX file of this assignment), and submit the resulting single pdf file at GradeScope by the due date. (Note: As always, **no late submissions** will be allowed, other than one-day late submission with 10% penalty! Within GradeScope, indicate the page number where your solution to each question starts, else we won't be able to grade it!).
- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes written for any programming assignments (i.e., write your own code; we will run plagiarism checks on codes).
- If you have referred a book or any other online material for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words.

1. (10 points) [IS HMM BP IN DISGUISE? Hmmm] Consider the following Hidden Markov Model (HMM): The hidden r.v.s are $Y_0, \ldots, Y_n$ taking values in $\{1, \ldots, a\}$ and the observed r.v.s are $X_0, \ldots, X_n$ taking values in $\{1, \ldots, b\}$. The parameters are given by the transition probability matrix $T \in \mathbb{R}_+^{a \times a}$, the emission probability matrix $V \in \mathbb{R}_+^{b \times a}$ and the initial probability vector $\pi \in \mathbb{R}_+^a$. Assume all parameters are known to you. Concretely,

$$\begin{aligned} T_{j,i} &= P(Y_{t+1} = j \mid Y_t = i) \\ V_{k,i} &= P(X_t = k \mid Y_t = i) \\ \pi_i &= P(Y_0 = i). \end{aligned}$$

The joint probability is given as:

$$P(X, Y) = P(Y_0)P(X_0 \mid Y_0) \prod_{t=1}^{n} P(Y_t \mid Y_{t-1})P(X_t \mid Y_t)$$

Consider the BP algo. from class with a two-phase message-passing schedule rooted at node Y_n.

(a) (3 points) [FORWARD ALGO. FOR HMM LIKELIHOOD] Derive the evidence probability $P(X = x)$ using the BP algorithm's first phase (bottom-up or forward phase). Simplify these first-phase BP messages by expressing them as recursive updates of $\alpha_t^i := P(X_0, X_1, \ldots, X_t, Y_t = i) = \sum_{Y_0, \ldots, Y_{t-1}} P(X_0, \ldots, X_t, Y_0, \ldots, Y_{t-1}, Y_t = i)$. What are the source and target nodes of the α_t^i message, and what message is sent from the observed X_t to the hidden Y_t?

(b) (2 points) [VITERBI ALGO. FOR HMM DECODING] What minimal changes will you need to make to the above algorithm so that it computes $\max_y P(X, Y = y)$ instead of $\sum_y P(X, Y = y)$. In particular, how will you change the definition of α_t^i and the corresponding recursive updates? Dont' forget to mention also how to initialize/terminate the recursive updates.

(c) (4 points) [BACKWARD ALGO. FOR HMM LEARNING] Simplify your second (top-down or backward) phase BP messages by expressing them as recursive updates of $\beta_t^i := P(X_{t+1}, X_{t+2}, \ldots, X_n \mid Y_t = i)$. What are the source and target nodes of the β_t^i message? How will you use these backward messages along with the forward messages to compute $P(Y_t = i \mid X)$ (a quantity useful for learning HMM parameters from data)?

(d) (1 point) How many JTs (junction trees) are possible for the HMM model above, and which of these JTs would you choose to get a JT algorithm whose messages are identical to the BP algorithm messages above?

2. (10 points) [JT IN THEORY] Let JT be any junction tree of a (chordal or non-chordal) MN H_Φ associated with a set of factors $\phi \in \Phi$ defined over n random variables (with mega-factors grouping these original factors denoted by $\psi_i(C_i)$).

(a) (3 points) Prove that the number of nodes in the JT is at most n. (Hint: For a given PEO of a chordal graph, assign each maximal clique to the first eliminated vertex in the clique; and count to show that the number of maximal cliques in the chordal graph is at most n.)

(b) (3 points) For any edge (C_i, C_j) in JT, let sepset $S_{ij} := C_i \cap C_j$, and let $X_{<(i,j)}$ be the set of all variables in the scope of clusters in the C_i side of the tree, and $X_{<(j,i)}$ be the set of all variables in the scope of clusters in the C_j side of the tree. Use separation criteria in H_Φ to prove that $(X_{<(i,j)} \perp X_{<(j,i)} \mid S_{ij})$ in distribution $P_\Phi(\mathcal{X})$.

(c) (2 points) At convergence (i.e., at the end of two phases of the message-passing schedule; aka (global) calibration) of the JT algorithm, prove that:

$$\mu_{ij}(S_{ij}) = m_{j \to i}(S_{ij}) m_{i \to j}(S_{ij})$$

where $\mu_{ij}(S_{ij}) := \sum_{C_i - S_{ij}} \beta_i(C_i)$ and βs are the unnormalized marginals as defined in class. Also, briefly note why $\mu_{ij}(S_{ij})$ is also equal to $\sum_{C_j - S_{ij}} \beta_j(C_j)$.

(d) (2 points) Use above result to prove that a converged $JT = (V_{JT}, E_{JT})$ can be used to reformulate the joint distribution as:

$$\tilde{P}_\Phi(\mathcal{X}) = \frac{\prod_{i \in V_{JT}} \beta_i(C_i)}{\prod_{(ij) \in E_{JT}} \mu_{ij}(S_{ij})}.$$

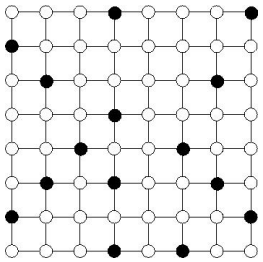
3. (10 points) [TOYING AROUND WITH MCMC::MH] Use normal distribution centered at the current state of the chain as a proposal distribution in the Metropolis-Hastings algorithm to sample from the $Gamma(\theta, 1)$ distribution when θ is a non-integer that is at least 2.

(a) (3 points) Write down the acceptance probability (after all simplifications). What properties do you need to verify to confirm your method reaches the right stationary distribution after running for a sufficiently long time? Show your verification.

(b) (4 points) Provide your code that simulates 1000 values from $\mathsf{Gamma}(5.5, 1)$, and show the trace plots, and the histogram of X_n (with the gamma density overlaid).

(c) (3 points) How did you choose your burn-in time? Report it along with your acceptance rate during the burn-in vs. sample collection periods. How did these values change as a function of the variance parameter of your normal distribution, and what do you think is the optimal variance for fast convergence?

4. (10 points) [HARD-CORE WITH MCMC::GIBBS]



Consider a (non-complete) connected graph *G = (V, E)* such as the one shown with $n = |V|$. Now each vertex in V gets either mapped to 0 or 1, where we only consider the following set $C \subset \{0,1\}^n$ of admissible configurations characterized by the property that pairs of adjacent vertices **cannot both** take the value 1 (see figure where black denotes 1).
Now, we want to pick one of the admissible configurations $\mathbf{x} \in C$ "at random". That is, we consider the (discrete) uniform distribution π on C, i.e. $\pi_{\mathbf{x}} = \frac{1}{|C|} \ \forall\, \mathbf{x} \in C$.

(a) (3 points) Write down the edge potentials of the undirected graphical model for this problem, and a Gibbs sampling algorithm for sampling from this model (including how you derived the associated conditional $P(X_i|X_{-i})$). Does your algorithm need to know the partition function $|C|$?

(b) (3 points) Is the Markov chain you set up irreducible and aperiodic? Does your chain admit a distribution that satisfies detailed balance? If it simplifies your proof, assume here that your Gibbs sampling routine employs "random scan" (pick a random i from $1, \ldots, n$ and then make a move based on $P(X_i|X_{-i})$ in each epoch) instead of "systematic scan" (cycle through all i from 1 to n in each epoch).

(c) (4 points) Provide your code and trace plots (of some functions that each map a configuration to a real value that helps visualize how well the chain is mixing). Plot the burn-in time you chose as a function of the size of the grid graphs you used. We didn't ask about the empirical acceptance rate here as it is always 100% for Gibbs sampling - prove that it is so under the same "random scan" epoch assumption above.