HOMEWORK 2: WRITTEN BELIEF PROPAGATION AND MCMC[1]

10-708 PROBABILISTIC GRAPHICAL MODELS (SPRING 2022)
https://andrejristeski.github.io/10708-22/

OUT: Feb 14th
DUE: Feb 28th at 11:59 PM
TAs: Kai-Ling Lo, Aaron Rumack

START HERE: Instructions

- **Collaboration policy:** The purpose of student collaboration is to facilitate learning, not to circumvent it. Studying the material in groups is strongly encouraged. It is also allowed to seek help from other students in understanding the material needed to solve a particular homework problem, provided no written notes (including code) are shared, or are taken at that time, and provided learning is facilitated, not circumvented. The actual solution must be done by each student alone. The presence or absence of any form of help or collaboration, whether given or received, must be explicitly stated and disclosed in full by all involved. See the Academic Integrity Section on the course site for more information: https://andrejristeski.github.io/10708-22/#:~:text=Academic%20Integrity%20Policies
- **Late Submission Policy:** See the late submission policy here: https://andrejristeski.github.io/10708-22/#:~:text=Grace%20Day/Late%20Homework%20Policy
- **Submitting your work to Gradescope:** We use Gradescope (https://www.gradescope.com/courses/349316/assignments) to collect PDF submissions of open-ended questions on the homework (e.g. mathematical derivations, plots, short answers). The course staff will manually grade your submission, and you'll receive personalized feedback explaining your final marks. The homework template must be used and can be completed in Latex or by hand. Handwritten submissions must be legible otherwise we will not be able to give credit to your solutions. No changes should be made to the template, boxes and choices **MUST** remain the same size and in the same locations between the template and your completed submission, the document has 16 pages so your submission must contain no more and no less than 16 pages.
- For **multiple choice** or **select all that apply questions**, shade in the box or circle in the template document corresponding to the correct answer(s) for each of the questions. For LaTeX users, replace `\choice` with `\CorrectChoice` to obtain a shaded box/circle, and don't change anything else.

[1]Compiled on Tuesday 15th February, 2022 at 03:42

A Written Questions [70 pts]

Answer the following questions in the template provided. Then upload your solutions to Gradescope. You may use LaTeX or print the template and hand-write your answers then scan it in. Failure to use the template will result in a penalty. There are 70 points and 11 questions.

A.1 Message Passing

a	$\psi_A(a)$
0	3
1	1

b	$\psi_B(b)$
0	1
1	1

c	$\psi_C(c)$
0	2
1	2

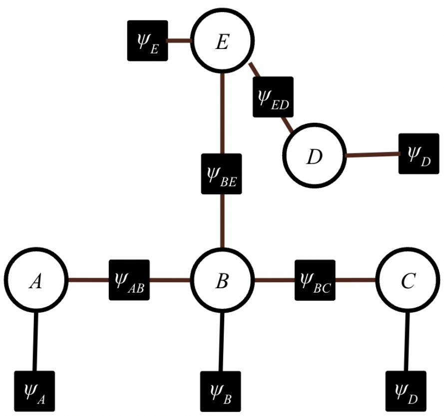

Figure A.1

d	$\psi_D(d)$
0	3
1	1

e	$\psi_E(e)$
0	1
1	4

a	b	$\psi_{AB}(a,b)$
0	0	2
0	1	1
1	0	1
1	1	1

b	e	$\psi_{BE}(b,e)$
0	0	1
0	1	1
1	0	1
1	1	1

b	c	$\psi_{BC}(b,c)$
0	0	1
0	1	2
1	0	1
1	1	3

e	d	$\psi_{ED}(e,d)$
0	0	1
0	1	2
1	0	1
1	1	2

1. Consider the factor graph in Figure A.1. On paper, carry out a run of belief propagation by sending messages first from the leaves, ψ_A, ψ_B, ψ_C and ψ_D, to the root ψ_E, and then from the root back to the leaves. Then answer the questions below. Assume all messages are un-normalized.

   (a) (1 point) **Numerical answer:** What is the message from A to ψ_{AB}?

   (b) (1 point) **Numerical answer:** What is the message from ψ_{ED} to D?

(c) (1 point) **Numerical answer:** What is the belief at variable A?

(d) (1 point) **Numerical answer:** What is the belief at variable B?

(e) (1 point) **Numerical answer:** What is the belief at factor ψ_{BE}?

(f) (1 point) **Numerical answer:** What is the value of the partition function?

A.2 Belief Propagation for a Pairwise MRF

Consider an *acyclic* undirected graphical model defined by $G = (V, E)$, with vertices V and edges E. We assume that each element $i \in V$ of the vertex set is an index with a corresponding variable Y_i for that node. Assume $Y_i \in \mathcal{Y}$ for all i. We denote the neighbors of a node i as $\mathcal{N}(i)$. Assume that G is a tree. We can define a pairwise MRF for G as:

$$p(\mathbf{y}) = \frac{1}{Z} \prod_{i \in V} \phi_i(y_i) \prod_{(i,j) \in E} \phi_{ij}(y_i, y_j)$$

where ϕ_i is the potential function for node i and ϕ_{ij} the potential for edge (i, j). Here we will design a belief propagation algorithm that is tailored to acyclic pairwise MRFs.

2. (4 points) Define a belief propagation algorithm that *only* stores messages sent from one variable y_i to another variable y_j denoted as $m_{i \to j}(y_j)$. Define both how to compute each message and an appropriate message passing order. Here, you should describe only the message passing portion of the algorithm (i.e. do not define the beliefs yet).

3. (2 points) After your algorithm from above terminates, how can you use the messages $m_{j \to i}(y_i)$ to compute the variable marginals $p(y_i)$ (i.e. normalized beliefs)?

4. (3 points) After your algorithm from above terminates, how can you use the messages $m_{i \to j}(y_j)$ to compute the edge marginals $p(y_i, y_j)$ for all $(i, j) \in E$?

A.3 Markov Chain Monte Carlo Methods

5. In class, we studied two Monte Carlo estimation methods: rejection sampling and importance sampling. Given a proposal distribution $Q(x)$, answer the following questions:

   (a) (1 point) If sampling from $Q(x)$ is computationally expensive, which of the following methods is likely to be more efficient?

   ◯ Rejection Sampling

   ◯ Importance Sampling

   ◯ Both are equally inefficient

   (b) (1 point) If $Q(x)$ is high-dimensional, which of the following methods is more efficient?

   ◯ Rejection Sampling

   ◯ Importance Sampling

   ◯ Both are inefficient.

   (c) (1 point) For high-dimensional distributions, MCMC methods such as Metropolis Hastings are generally more efficient than rejection sampling and importance sampling.

   ◯ True

   ◯ False

6. (10 points) Consider a 1-dimensional random variable X and a target density $P(x)$ from which we hope to sample using the rejection sampling algorithm. Let $Q(x)$ be the candidate/proposal density, which is easy to sample from, and satisfies the following properties:

   - The support of Q (denoted by $\mathcal{X}_Q$) includes the support of P (denoted by $\mathcal{X}_P$). That is, $\mathcal{X}_P \subseteq \mathcal{X}_Q$.
   - We know $k = \sup_{x \in \mathcal{X}_P} P(x)/Q(x) < \infty$.

   Recall that in every step, the rejection sampling algorithm takes a sample x from Q, as well as a sample u from the uniform distribution on the unit interval ($U[0,1]$). It then rejects the sample x if $u \leq \frac{P(x)}{kQ(x)}$ and accepts it otherwise.

   (a) (3 points) Show that the probability of a sample x getting accepted through the above process is $1/k$.

(b) (7 points) Show that the distribution of the accepted values from the rejection sampling algorithm above follows the target density P.

7. (2 points) Suppose you are using MCMC methods to sample from a distribution with multiple modes. Briefly explain what complications may arise while using MCMC.

8. (5 points) Consider the Markov Chain specified by the transition matrix

$$T = \begin{pmatrix} 1/4 & 1/4 & 1/2 \\ 0 & 1/3 & 2/3 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

Does it have a unique stationary distribution? If yes, why? If no, write down at least two different stationary distributions.

9. (5 points) Consider the Markov Chain specified by the transition graph in Figure A.2. Does it have a unique stationary distribution? If yes, why? If no, write down at least two different stationary distributions.

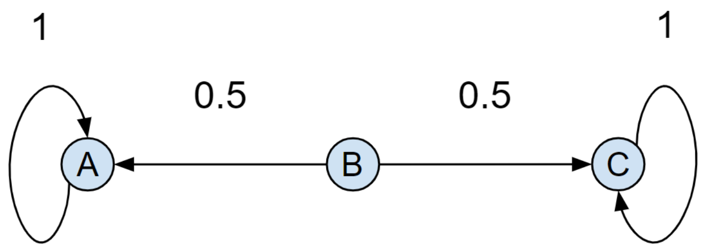

Figure A.2

10. Consider an Ising model

$$p(\mathbf{x}) = \frac{1}{Z_J} \exp\left(\sum_{i,j=1}^{d} J_{ij}\mathbf{x}_i\mathbf{x}_j + \sum_{i=1}^{d} J_i\mathbf{x}_i\right)$$

Through this exercise, we will see that the tasks of sampling and calculating the partition function are computationally equivalent in this model.

(a) (13 points) Suppose that you are given an oracle $\mathcal{O}(J)$, which given an Ising model with parameters $\{J_{ij}\}_{i,j\in[d]}, \{J_i\}_{i\in[d]}$ returns a random sample from $p(\mathbf{x})$.

Assuming $|J_{ij}| \leq C, \forall i,j \in [d]$ and $|J_i| \leq C, \forall i \in [d]$, show that given access to such an oracle, we can design a (randomized) algorithm which outputs, with probability $1 - 1/d$, an approximation of the partition function Z_J to within a multiplicative factor of $1 + \epsilon$ and runs in time poly$(d, 1/\epsilon, C)$, for every $\epsilon > 0$.
Hint: Recall annealed importance sampling, Chebyshev's inequality and the union bound.

(b) (7 points) Conversely, given an oracle $\mathcal{O}(J)$ that can calculate the partition function of an Ising model with parameters J, design a polynomial time algorithm to draw samples from a desired Ising model.

You may assume that for a real-valued $q \in [0, 1]$ that can be efficiently calculated from the parameters J in polynomial time, you can draw a sample from a Bernoulli variable with bias q in constant time.

11. (10 points) Consider an Ising model, described by an undirected graphical model $G(V, E)$ ($V = \{1, 2, \cdots, n\}$ and $\{i, j\} \in E$ means i, j are adjacent in G) and

$$p(\mathbf{x}) \propto \exp\left(\sum_{i,j:\{i,j\}\in E} J_{ij}\mathbf{x}_i\mathbf{x}_j + \sum_i J_i\mathbf{x}_i\right)$$

Note that $J_{ij} \neq 0$ only if i and j are adjacent in the graph G.

Show that if the graph G is an undirected tree (see Fig. A.3 for an example), there is a polynomial time algorithm that draws samples from p. You may assume that for any real-value $q \in [0, 1]$, drawing a sample from a Bernoulli variable with bias q can be done in constant time (i.e., $O(1)$).

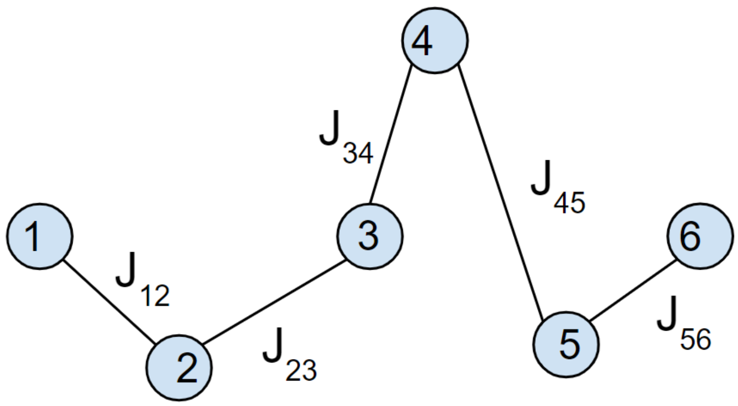

Figure A.3: An example of an undirected tree with $n = 6$. Note that the question addresses general trees, not this specific example.

A.4 Collaboration Policy

After you have completed all other components of this assignment, report your answers to the collaboration policy questions detailed in the Academic Integrity Policies for this course.

1. Did you receive any help whatsoever from anyone in solving this assignment? If so, include full details including names of people who helped you and the exact nature of help you received.

2. Did you give any help whatsoever to anyone in solving this assignment? If so, include full details including names of people you helped and the exact nature of help you offered.

3. Did you find or come across code that implements any part of this assignment? If so, include full details including the source of the code and how you used it in the assignment.