Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
6.438 Algorithms for Inference
Fall 2020

QUIZ 2

Thursday, December 3, 2020
9:00 am – 11:00 am / 7:00 pm – 9:00 pm

- This is a closed book exam, but four $8\frac{1}{2}$"$\times$11" sheets of notes (8 sides total) are allowed.
- Calculators are allowed, but probably won't be useful.
- There are **4** problems of unequal value as follows:

Problem	Points
1	20
2	10
3	15
4	25

  The problems add up to 70 **points but the maximum score that you can get is 60 points. If you get** x **points, then the score that contributes to your final grades is** $\min\{x, 60\}$. For example: if you receive 45/70, your score will be 45; if you receive 65/70, your score will be 60.
- The problems are not necessarily in order of difficulty. We recommend that you read through all the problems first, then do the problems in whatever order suits you best.
- Quite often later parts of a problem can be done independently of earlier parts, so if you get stuck in one part, proceed to the next and come back to the earlier part later.
- Record all your solutions either on a printed copy of the answer booklet, or blank sheets of paper with each problem part on a different page. You may want to first work things through on the scratch paper, and then carefully transfer to the solution that you would like us to look at.

- A correct answer does not guarantee full credit, and a wrong answer does not guarantee loss of credit. You should clearly but concisely indicate your reasoning and **show all relevant work**. Your grade on each problem will be based on our best assessment of your level of understanding as reflected by what you have written in the answer booklet. **No credit will be given for an answer without valid justification!**
- Please be neat—we can't grade what we can't decipher!

Problem 1 (20 points)

In this problem, we explore methods of parameter estimation on a modified hidden Markov model. Recall the homogeneous hidden Markov model given by the diagram below.

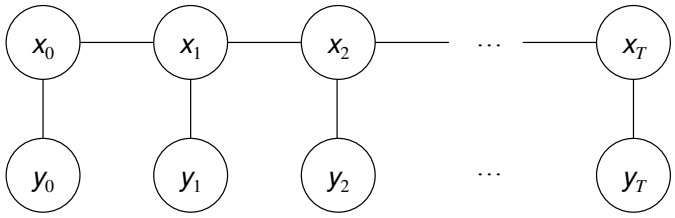


Figure 1: A simple homogeneous HMM.

We have latent variables $\mathbf{x} = \{x_0, \ldots, x_T\}$ and observed variables $\mathbf{y} = \{y_0, \ldots, y_T\}$, where x_t and y_t denote the latent and observed variables at timestep t, respectively. We'll assume that all variables take on values in $\{1, 2, \ldots, M\}$. We say that the model is time-homogeneous because the transition probability from one state to the next is invariant of time. The model parameters for this simple HMM can be written as $\theta = \{\pi, A, \eta\}$ where:

- π is the initial state distribution ($\pi_i = p_{x_0}(i)$ for $i \in \{1, \ldots, M\}$)
- A is the 2D transition matrix ($a_{ij} = p_{x_{t+1}|x_t}(j|i)$ for $i, j \in \{1, \ldots, M\}$)
- η is the emission distribution ($\eta_{ij} = p_{y_t|x_t}(j|i)$ for $i, j \in \{1, \ldots, M\}$)

In this problem, we will consider the modified HMM displayed below where each x_t now depends on the previous two states, x_{t-1} and x_{t-2}. The model parameters are

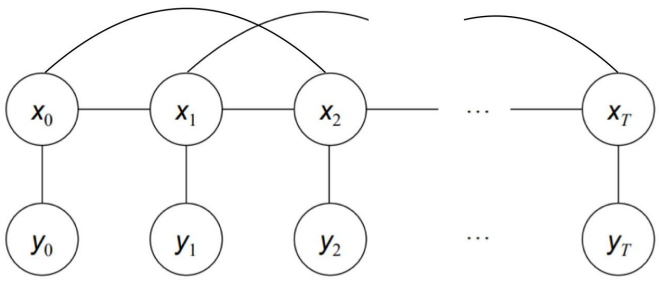


Figure 2: The modified homogeneous HMM.

$\theta_M = \{\tau, B_0, B, m\}$ where:

- τ is the initial state distribution ($\tau_i = p_{x_0}(i)$ for $i \in \{1, \ldots, M\}$)

- B_0 is the 2D transition matrix ($b_{ij} = p_{x_1|x_0}(j|i)$ for $i, j \in \{1, \ldots, M\}$)
- B is the 3D transition matrix ($b_{ijk} = p_{x_{t+1}|x_t,x_{t-1}}(k|j,i)$ for $i, j, k \in \{1, \ldots, M\}$)
- m is the emission distribution ($m_{ij} = p_{y_t|x_t}(j|i)$ for $i, j \in \{1, \ldots, M\}$)

For simplicity, we will focus on estimating parameters τ, B, m in this problem and not worry about B_0. For your solutions, you may use x_t^i, y_t^j to denote indicator variables $\mathbf{1}(x_t = i), \mathbf{1}(y_t = j)$, respectively.

(a) (4 points) First, consider the case where complete data is observed, and we have one observation for each x_t and y_t. Write the complete log-likelihood, $\ell(\theta; \mathbf{x}, \mathbf{y}) = \log \mathbb{P}_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}; \theta)$, in terms of the parameters τ, B_0, B, m of the modified HMM. Use this to find the maximum likelihood (ML) estimates for τ_i, b_{ijk}, m_{ij} for all $i, j, k \in \{1, \ldots, M\}$.

Now, we can consider when $\mathbf{x}$ is hidden and $\mathbf{y}$ is observed. Recall that the Baum-Welch algorithm can be used to find ML parameter estimates for the *original* HMM. At each iteration the algorithm performs the two steps:

E-step: Given current parameter estimates $\theta^{(\ell)} = \{\widehat{\pi}^{(\ell)}, \widehat{A}^{(\ell)}, \widehat{\eta}^{(\ell)}\}$, run the sum-product algorithm to compute all node and edge marginals, $\mathbb{P}(\mathsf{x}_t = i|\mathbf{y} = \mathbf{y}; \theta^{(\ell)})$ and $\mathbb{P}(\mathsf{x}_t = i, \mathsf{x}_{t+1} = j|\mathbf{y} = \mathbf{y}; \theta^{(\ell)})$.

M-step: Using node and edge marginals, compute $\theta^{(\ell+1)} = \{\widehat{\pi}^{(\ell+1)}, \widehat{A}^{(\ell+1)}, \widehat{\eta}^{(\ell+1)}\}$ using

$$\begin{aligned}
\widehat{\pi}_i^{(\ell+1)} &= \mathbb{P}(\mathsf{x}_0 = i|\mathbf{y} = \mathbf{y}; \theta^{(\ell)}) \\
\widehat{a}_{ij}^{(\ell+1)} &= \frac{\sum_{t=0}^{T-1} \mathbb{P}(\mathsf{x}_t = i, \mathsf{x}_{t+1} = j|\mathbf{y} = \mathbf{y}; \theta^{(\ell)})}{\sum_{t=0}^{T-1} \mathbb{P}(\mathsf{x}_t = i|\mathbf{y} = \mathbf{y}; \theta^{(\ell)})} \\
\widehat{\eta}_{ij}^{(\ell+1)} &= \frac{\sum_{t=0}^{T} \mathbb{P}(\mathsf{x}_t = i|\mathbf{y} = \mathbf{y}; \theta^{(\ell)}) \mathbf{1}(y_t = j)}{\sum_{t=0}^{T} \mathbb{P}(\mathsf{x}_t = i|\mathbf{y} = \mathbf{y}; \theta^{(\ell)})}
\end{aligned}$$

Notice that by computing certain marginals over the hidden variables, we can compute the ML estimate. In our *modified* HMM, we can use the same approach, but we have a new set of marginals which are needed to compute the ML estimate $\theta_M^{(\ell+1)}$. In parts (b)-(c) we will explore two methods of obtaining these marginal distributions.

(b) First, we can approximate marginal distributions by running loopy BP on the *modified* HMM. Suppose we define a new distribution over the subgraph without $\mathbf{y}$, factorizing into node potentials $\phi_t(x_t)$ and edge potentials $\psi_{t,t+1}(x_t, x_{t+1})$, $\psi_{t,t+2}(x_t, x_{t+2})$. This distribution can be used to approximate the conditional distribution $\mathbb{P}_{\mathbf{x},\mathbf{y}}(\mathbf{x}|\mathbf{y}; \theta_M^{(\ell)})$:

$$\mathbb{P}_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}; \theta_M^{(\ell)}) \cong \frac{1}{Z} \prod_{t=0}^{T} \phi_t(\mathsf{x}_t) \prod_{t=0}^{T-1} \psi_{t,t+1}(\mathsf{x}_t, \mathsf{x}_{t+1}) \prod_{t=0}^{T-2} \psi_{t,t+2}(\mathsf{x}_t, \mathsf{x}_{t+2})$$

where Z is the normalization constant. We iteratively update messages along edges through loopy BP, and can generate beliefs for the marginal distributions over nodes.

(i) (4 points) Express the belief of the marginal distribution $\mathbb{P}(\mathsf{x}_t = i, \mathsf{x}_{t+1} = j, \mathsf{x}_{t+2} = k|\mathbf{y} = \mathbf{y}; \theta_M^{(\ell)})$ for $2 \leq t \leq T-4$ in terms of node potentials, edge potentials, and messages $m_{r\to s}(x_s)$ where $0 \leq r, s \leq T$.

(ii) (4 points) Find $\theta_M^{(\ell+1)}$ in terms of the estimated marginal distributions $\mathbb{P}(\mathsf{x}_t = i, \mathsf{x}_{t+1} = j, \mathsf{x}_{t+2} = k|\mathbf{y} = \mathbf{y}; \theta_M^{(\ell)})$, $\mathbb{P}(\mathsf{x}_t = i, \mathsf{x}_{t+1} = j|\mathbf{y} = \mathbf{y}; \theta_M^{(\ell)})$, $\mathbb{P}(\mathsf{x}_t = i|\mathbf{y} = \mathbf{y}; \theta_M^{(\ell)})$.

(c) Instead of approximating the marginals with loopy BP, we can compute the exact marginals by converting our modified HMM into the original HMM structure.

(i) (4 points) Define a new set of variables $\mathbf{s}$, $\mathbf{v}$ in terms of $\mathbf{x}, \mathbf{y}$ that allows us to represent the modified HMM in Figure 2 in the form of the original HMM, with $\mathbf{s}$ as the hidden variables and $\mathbf{v}$ as the observed. With this reparameterization, we will be able to run the Baum-Welch algorithm on the redefined graph to determine the θ_M.
Note: You only need to define $\mathbf{s}$, $\mathbf{v}$ and briefly explain why they are represented by the original HMM graph. You do not need to determine any distributions on the new variables.

(ii) (4 points) Let $\theta_{s,v}$ be the parameters of the original HMM defined over $\mathbf{s}, \mathbf{v}$. We run Baum-Welch on this graph, which finds the ML estimates $\theta_{s,v}^{(\ell)}$ at each iteration. After each E-step, we obtain the marginals $\mathbb{P}(s_t, s_{t+1}|\mathbf{v}; \theta_{s,v}^{(\ell)})$ and $\mathbb{P}(s_t|\mathbf{v}; \theta_{s,v}^{(\ell)})$. Using these distributions, we can in fact compute the estimate $\theta_M^{(\ell+1)}$ of the parameters of the *modified* HMM over $\mathbf{x}$ and $\mathbf{y}$ as well. Determine $\widehat{\tau}_i^{(\ell+1)}$, $\widehat{b}_{ijk}^{(\ell+1)}$, $\widehat{m}_{ij}^{(\ell+1)}$ for all $i, j, k \in \{1, \ldots, M\}$ at each iteration of the M-step in terms of the node and edge marginals of newly defined graph, $\mathbb{P}(s_t, s_{t+1}|\mathbf{v}; \theta_{s,v}^{(\ell)})$ and $\mathbb{P}(s_t|\mathbf{v}; \theta_{s,v}^{(\ell)})$.

Problem 2 (10 points)

In this problem, we use do-calculus to determine causal effects from the causal DAG below. Consider the following 4 binary variables for an individual:

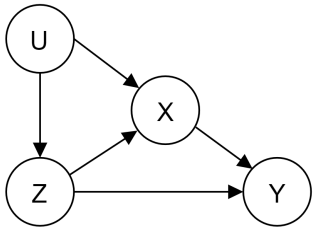


Figure 3: Causal DAG

x: full night's sleep (no=0, yes=1)

y: pass exam (no=0, yes=1)

z: hours of study (low=0, high=1)

u: number of other classes (low=0, high=1)

The DAG in Figure 3 represents the following causal theory: Whether a student passes their exam is affected by their hours of sleep and study, which are both affected by the number of other classes they're taking. The amount they study also directly affects how much sleep they get.

(a) (5 points) Suppose all four variables are observed. The table below provides the distributions $\mathbb{P}(u), \mathbb{P}(y|x,z)$, and $\mathbb{P}(x|z,u)$. Using the rules of do-calculus and the provided data, calculate $\mathbb{P}(y=1|do(z=0))$.

u	$\mathbb{P}(u)$
0	0.6
1	0.4

(x,z)	$\mathbb{P}(y=1\|x,z)$	$\mathbb{P}(y=0\|x,z)$
(1,1)	0.3	0.7
(1,0)	0.1	0.9
(0,1)	0.5	0.5
(0,0)	0.2	0.8

(z,u)	$\mathbb{P}(x=1\|z,u)$	$\mathbb{P}(x=0\|z,u)$
(1,1)	0.1	0.9
(1,0)	0.4	0.6
(0,1)	0.2	0.8
(0,0)	0.5	0.5

(b) (5 points) Suppose now that we don't have access to the number of other classes, u. We would like to determine whether the following causal effects can be determined from observing x, y, z only. State whether the following statements are true, and explain your reasoning.

(i) $\mathbb{P}(y|do(z))$ can be determined without observing u. (An equivalent statement is that $\{x\}$ is a valid adjustment set for (z, y)).

(ii) $\mathbb{P}(y|do(x))$ can be determined without observing u. (An equivalent statement is that $\{z\}$ is a valid adjustment set for (x, y)).

Problem 3 (15 points)

In this problem, we will focus on learning the structure of undirected Gaussian graphs using *correlation* (to be defined later) and *partial correlation* (to be defined later). We will focus on Gaussian trees for part (a) and part (b). We will focus on Gaussian graphs (that may have loops) for part (c).

Consider a *Gaussian random vector* $\mathbf{x} \in \mathbb{R}^p$ with a positive definite information matrix $\mathbf{J}$ and covariance matrix $\mathbf{\Lambda}$. Let the distribution of $\mathbf{x}$ factorize according to an undirected tree $\mathcal{G}_{\mathcal{T}} = (\mathcal{V}, \mathcal{E}_{\mathcal{T}})$ with vertices $\mathcal{V} = \{1, \cdots, p\}$ and edges $\mathcal{E}_{\mathcal{T}}$ such that $\mathbf{J}_{ij} = 0$ if and only if $(i,j) \notin \mathcal{E}_{\mathcal{T}}$. We want to learn the structure of $\mathcal{G}_{\mathcal{T}}$. Let $\mathcal{G}_c = (\mathcal{V}, \mathcal{E}_c)$ denote the complete graph i.e., $(i,j) \in \mathcal{E}_c$ for all $i \neq j$.

• Correlation: The *correlation* between variables x_i and x_j (denoted by $\rho_{i,j}$) is:

$$\rho_{i,j} \triangleq \frac{\mathrm{cov}(\mathsf{x}_i, \mathsf{x}_j)}{\sqrt{\mathrm{var}(\mathsf{x}_i)}\sqrt{\mathrm{var}(\mathsf{x}_j)}} \tag{1}$$

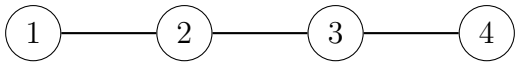
(a) (3 points) Suppose we know $\rho_{i,j}$ for all $i \neq j$. Let $\mathcal{G}_1 = (\mathcal{V}, \mathcal{E}_1)$ be such that $(i,j) \in \mathcal{E}_1$ if $\rho_{i,j} \neq 0$. Which of the following is true? (Please provide a proof for your answer.)

(i) $\mathcal{G}_1 = \mathcal{G}_{\mathcal{T}}$.

(ii) $\mathcal{G}_1 = \mathcal{G}_c$.

(iii) None of the above

Hint: You can use the following facts for a tree: (A) $\mathbf{\Lambda}_{ij} \neq 0$ for all $(i,j) \in \mathcal{E}_{\mathcal{T}}$. (B) For any $j \neq k$, ρ_{jk} is the product of the correlations along the edges of the path connecting nodes j and k. For example, $\rho_{14} = \rho_{12}\rho_{23}\rho_{34}$ for the tree below.



• Partial Correlation: The *partial correlation* between variables x_i and x_j (denoted by $r_{i,j}$) measures their conditional correlation given the values of the other variables $\mathsf{x}_{\mathcal{V}\setminus\{i,j\}} = \{x_k : k \in \mathcal{V} \setminus \{i,j\}\}$.

$$r_{i,j} \triangleq \frac{\mathrm{cov}(\mathsf{x}_i, \mathsf{x}_j | \mathsf{x}_{\mathcal{V}\setminus\{i,j\}})}{\sqrt{\mathrm{var}(\mathsf{x}_i | \mathsf{x}_{\mathcal{V}\setminus\{i,j\}})}\sqrt{\mathrm{var}(\mathsf{x}_j | \mathsf{x}_{\mathcal{V}\setminus\{i,j\}})}} \tag{2}$$

(b) (5 points) Suppose we know $r_{i,j}$ for all $i \neq j$. Let $\mathcal{G}_2 = (\mathcal{V}, \mathcal{E}_2)$ be such that $(i,j) \in \mathcal{E}_2$ if $r_{i,j} \neq 0$. Which of the following is true? (Please provide a proof for your answer.)

(i) $\mathcal{G}_2 = \mathcal{G}_{\mathcal{T}}$.

(ii) $\mathcal{G}_2 = \mathcal{G}_c$.

(iii) None of the above

• Graphs with potential loops: We will now shift our focus from Gaussian trees to Gaussian graphs (that may have loops). Consider a *Gaussian random vector* $\mathbf{x} \in \mathbb{R}^p$ with a positive definite information matrix $\mathbf{J}$ and covariance matrix $\mathbf{\Lambda}$. Let the distribution of $\mathbf{x}$ factorize according to an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \{1, \cdots, p\}$ and edges $\mathcal{E}$ such that $\mathbf{J}_{ij} = 0$ if and only if $(i, j) \notin \mathcal{E}$. We are interested in learning the structure of $\mathcal{G}$ from samples. As before, let $\mathcal{G}_c = (\mathcal{V}, \mathcal{E}_c)$ denote the complete graph i.e., $(i, j) \in \mathcal{E}_c$ for all $i \neq j$.

In part (b), we assumed access to the true *partial correlations* for all $i \neq j$. However, in practice, we do not have access to these. As a first step in learning the structure of $\mathcal{G}$, we can estimate the empirical partial correlations from the data i.e., $\hat{r}_{i,j}$ for all $i \neq j$. Next, we can set a specific threshold and remove all edges from the complete graph $\mathcal{G}_c$ corresponding to the empirical partial correlations that are less than the given threshold. Let the resulting graph be our estimate $\hat{\mathcal{G}}$.

Let $\hat{\mathbf{J}}^b$ denote the normalized empirical information matrix before thresholding i.e.,

$$\hat{\mathbf{J}}^b_{ij} = \begin{cases} 1 & \text{if } i = j \\ -\hat{r}_{i,j} & \text{for all } i \neq j \end{cases}. \tag{3}$$

We will assume $\hat{\mathbf{J}}^b$ is positive definite.

Let $\hat{\mathbf{J}}^a$ denote the normalized empirical information matrix $\hat{\mathbf{J}}$ after thresholding i.e.,

$$\hat{\mathbf{J}}^a_{ij} = \begin{cases} \hat{\mathbf{J}}^b_{ij} & \text{if } i = j \text{ or } (i, j) \in \hat{\mathcal{G}} \\ 0 & \text{otherwise} \end{cases}. \tag{4}$$

The thresholding algorithm described above often works well, but a disadvantage is that $\hat{\mathbf{J}}^a$ may no longer be positive definite.

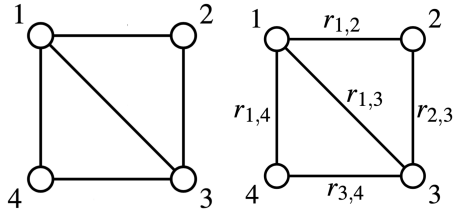
(c) (7 points) Show that $\hat{\mathbf{J}}^a$ is always positive definite if $\hat{\mathbf{J}}^b$ is MTP_2.
Hint: It is sufficient to consider the case when $\mathcal{G}_c$ and $\hat{\mathcal{G}}$ differ by only a single edge.

Problem 4 (25 points)

In this problem, we will look into a new interpretation of Gaussian belief propagation (BP) in trees and of the approximate method of loopy Gaussian BP in graphs with cycles. Throughout this problem, we will state certain properties that can you use without proof.

Consider a *Gaussian random vector* $\mathbf{x} \in \mathbb{R}^p$ with covariance matrix $\mathbf{\Lambda}$ and information matrix $\mathbf{J}$. Assume that $\mathbf{J}$ is positive definite ($\mathbf{J} \succ 0$) and $\mathbf{J}_{ii} = 1$ for all $i \in \{1, \cdots, p\}$. Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \{1, \cdots, p\}$ and edges $\mathcal{E}$ such that $\mathbf{J}_{ij} = 0$ for all $(i, j) \notin \mathcal{E}$. Let $\mathcal{N}(i)$ be the neighborhood of a node i. Our interest is in computing $\mathbf{\Lambda}_{ii}$ for all $i \in \{1, \cdots, p\}$.

• A new framework for Gaussian Models: We will now describe the framework of interest for Gaussian inference. Let $\mathbf{R} = \mathbf{I} - \mathbf{J}$ where $\mathbf{I}$ is the $p \times p$ identity matrix. Label each edge (i, j) of the graph $\mathcal{G}$ with costs $\mathbf{R}_{ij} = r_{i,j}$. See Figure 4 for an example.



(a) (b)

Figure 4: (a) An example of a Gaussian graphical model (say $\mathcal{G}_1$). (b) $\mathcal{G}_1$ with costs $r_{i,j}$.

A *path* of length $l \geq 0$ in $\mathcal{G}$ is a sequence $w = (w_0, w_1, ..., w_l)$ of nodes $w_k \in \mathcal{V}$ such that each step of the path (w_k, w_{k+1}) corresponds to an edge of the graph $\mathcal{G}$ i.e., $(w_k, w_{k+1}) \in \mathcal{E}$. Let $l(w)$ denote the length of path w. Paths may visit nodes and cross edges multiple times. For example, in $\mathcal{G}_1$, $w_1 = (1, 2, 3, 1, 2)$ is a path of length $l(w_1) = 4$. Define the *cost* of a path (denoted by $\phi(w)$) to be the product of cost of edges along the path:

$$\phi(w) = \prod_{k=1}^{l(w)} r_{w_{k-1}, w_k} \tag{1}$$

For example, $\phi(w_1) = r_{1,2} r_{2,3} r_{1,3} r_{1,2}$. We also allow zero-length "self" paths $w = (i)$ at each node i for which we define $\phi(w) = 1$. Given a set of paths $\mathcal{W}$, define the path-weight to be the sum of *costs* of all paths in $\mathcal{W}$:

$$\phi(\mathcal{W}) = \sum_{w \in \mathcal{W}} \phi(w) \tag{2}$$

Let $\mathcal{W}(\cdots)$ denote the set of paths having some property $\cdots$ and let $\phi(\cdots)$ denote the associated path-weights. For instance, $\mathcal{W}(i \xrightarrow{l} j)$ denotes the set of all paths from i to j of length l and $\phi(i \xrightarrow{l} j)$ is the corresponding path-weight. For example, in $\mathcal{G}_1$, $\mathcal{W}(1 \xrightarrow{2} 3) = \{(1,2,3),(1,4,3)\}$ and $\phi(1 \xrightarrow{2} 3) = r_{1,2}r_{2,3} + r_{1,4}r_{3,4}$. Similarly, $\mathcal{W}(i \to j)$ denotes the set of all paths from i to j and $\phi(i \to j)$ is the corresponding path-weight. For example, in $\mathcal{G}_1$, $\mathcal{W}(1 \to 3) = \{(1,3),(1,2,3),(1,4,3),(1,2,1,3),(1,4,1,3),(1,3,1,3),(1,3,2,3),(1,3,4,3),\cdots\}$ and $\phi(1 \to 3) = r_{1,3} + r_{1,2}r_{2,3} + r_{1,4}r_{4,3} + r_{1,2}r_{2,1}r_{1,3} + r_{1,4}r_{4,1}r_{1,3} + r_{1,3}r_{3,1}r_{1,3} + r_{1,3}r_{3,2}r_{2,3} + r_{1,3}r_{3,4}r_{4,3} + \cdots$.

In general, path-weights over countably many paths may or may not converge, and convergence may depend on the order of summation. A Gaussian distribution is *path-weightable* if for all $i, j \in \mathcal{V}$ the unordered sum over all paths w from i to j ($\phi(i \to j)$) is well-defined (i.e., converges to the same value for every possible summation order). Throughout this problem, we will only consider *path-weightable* distributions.

Property 1. Let $\mathbf{R}^l$ denote l-th power of the matrix $\mathbf{R}$. Then, $[\mathbf{R}^l]_{i,j} = \phi(i \xrightarrow{l} j)$.

Property 2. Path-weightability implies $\sum_k \mathbf{R}^k = (\mathbf{I} - \mathbf{R})^{-1}$.

(a) (2 points) We will now relate the covariances to path-weights. Show that: $\mathbf{\Lambda}_{ij} = \phi(i \to j)$.

• Product of paths: Given two paths $u = (u_0, \cdots, u_n)$ and $v = (v_0, \cdots, v_m)$ with $u_n = v_0$ (i.e., path v begins where path u ends), define the product of paths $uv = (u_0, ..., u_n, v_1, ..., v_m)$. For example, if $u = (1,2,3)$ and $v = (3,4,1)$, then $uv = (1,2,3,4,1)$. Let $\mathcal{U}$ and $\mathcal{V}$ be two countable sets of paths such that every path in $\mathcal{U}$ ends at a particular node and every path in $\mathcal{V}$ begin at this node. Define the product set $\mathcal{UV} = \{uv | u \in \mathcal{U}, v \in \mathcal{V}\}$.

Property 3. If for every $w \in \mathcal{UV}$ there is a unique pair $(u, v) \in \mathcal{U} \times \mathcal{V}$ such that $uv = w$, then $\phi(\mathcal{UV}) = \phi(\mathcal{U})\phi(\mathcal{V})$.

• Self-return paths: $\mathcal{W}(i \to i)$ is the set of *self-return paths at node i* (i.e., paths which begin and end at node i). This include paths which return to i many times. For example: $(1,2,1,2,1) \in \mathcal{W}(1 \to 1)$. Let $\mathcal{W}(i \xrightarrow{\setminus i} i)$ be the set of *single-revisit self-return paths at node i* i.e., the set of all paths with non-zero length that begin and end at i but do not visit i at any other point in between. For example: $(1,2,1,2,1) \notin \mathcal{W}(1 \xrightarrow{\setminus 1} 1)$ but $(1,2,1) \in \mathcal{W}(1 \xrightarrow{\setminus 1} 1)$. Let $\mathcal{W}^k(i \xrightarrow{\setminus i} i)$ be the set of *self-return paths that return exactly k times*. This is generated by taking the product of k copies of $\mathcal{W}(i \xrightarrow{\setminus i} i)$. Letting $\mathcal{W}^0(i \xrightarrow{\setminus i} i) \triangleq \{(i)\}$, we have

$$\mathcal{W}(i \to i) = \cup_{k \geq 0} \mathcal{W}^k(i \xrightarrow{\setminus i} i) \tag{3}$$

Property 4. Let $\mathcal{W} = \cup_{k=1}^{\infty} \mathcal{W}_k$ where the subsets $\mathcal{W}_k$ are disjoint. Then, $\phi(\mathcal{W}) = \sum_{k=1}^{\infty} \phi(\mathcal{W}_k)$.

(b) (5 points) We will now decompose variances in terms of *single-revisit self-return path-weights*. Let $\alpha_i = \phi(i \xrightarrow{\setminus i} i)$. Show that:

$$\mathbf{\Lambda}_{ii} = \frac{1}{1 - \alpha_i} \tag{4}$$

• Path-Weights and BP on Trees: Recall that in the Gaussian BP algorithm, the message sent from x_i to x_j can be written as

$$m_{i \to j}(\mathbf{x}_j) \propto \mathtt{N}^{-1}(\mathbf{x}_j; \mathbf{h}_{i \to j}, \mathbf{J}_{i \to j}), \tag{5a}$$

where

$$\mathbf{J}_{i \to j} = -\mathbf{J}_{ji} \Big(\mathbf{J}_{ii} + \sum_{k \in \mathcal{N}(i) \setminus \{j\}} \mathbf{J}_{k \to i} \Big)^{-1} \mathbf{J}_{ij}. \tag{5b}$$

We will now see that the exact path-weights over infinite sets of paths for variances in trees can be computed efficiently in a recursive fashion and these path-weight computations map exactly to BP updates.

Path-Weight Variance Calculation. The single-revisit self-return path-weight $\phi(j \xrightarrow{\setminus j} j)$ can be decomposed into weights over disjoint subsets of paths each of which corresponds to single-revisit self-return paths that exit node j via a specific one of its neighbors, say i. In particular, as illustrated in Figure 5, the single-revisit self-return paths that do this correspond to paths that live in the subtree $T_{i \to j}$ because the graph is a tree. Let $\mathcal{W}(j \xrightarrow{\setminus j} j | T_{i \to j})$ be the set of all single-revisit paths which are restricted to stay in subtree $T_{i \to j}$. We have,

$$\alpha_j = \phi(j \xrightarrow{\setminus j} j) = \sum_{i \in \mathcal{N}(j)} \phi(j \xrightarrow{\setminus j} j | T_{i \to j}) \triangleq \sum_{i \in \mathcal{N}(j)} \alpha_{i \to j} \tag{6}$$

Moreover, every single-revisit self-return path that lives in $T_{i \to j}$ must leave and return to node j through the single edge (i, j), and between these first and last steps must execute a (possibly multiple-revisit) self-return path at node i that is constrained not to pass through node j, that is, to live in the subtree $T_{i \setminus j}$ indicated in Figure 5. Thus

$$\alpha_{i \to j} = \phi(j \xrightarrow{\setminus j} j | T_{i \to j}) = r_{i,j}^2 \phi(i \to i | T_{i \setminus j}) \triangleq r_{i,j}^2 \gamma_{i \setminus j}. \tag{7}$$

(c) (8 points) We will now see that the path-weights $\alpha_{i \to j}$ (hence α_j and variances $\mathbf{\Lambda}_{jj}$) can be efficiently calculated by a path-weight analog of BP. First, derive a recursive relationship for $\alpha_{i \to j}$ in terms of $\alpha_{k \to i}$ where $k \in \mathcal{N}(i) \setminus \{j\}$ (This relationship can contain $r_{i,j}$ and numeric constants). Next, map this recursive relationship to (5b) and express $\alpha_{i \to j}$ and $\gamma_{i \setminus j}$ in terms of $\mathbf{J}_{i \to j}$ and $\mathbf{J}_{k \to i}$ where $k \in \mathcal{N}(i) \setminus \{j\}$ (This mapping can contain numeric constants).

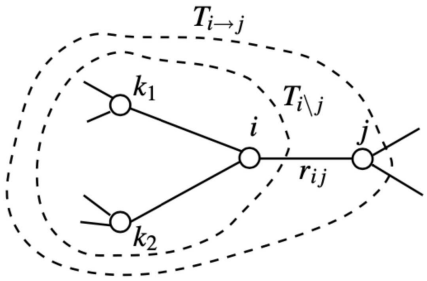


Figure 5: (a) Illustration of the subtree notation $T_{i\to j}$ and $T_{i\setminus j}$.

• Loopy BP: In Gaussian graphical models with cycles, loopy BP reduces to iterative application of (5) with information parameters of the messages denoted by $\mathbf{h}_{i\to j}^{(n)}$ and $\mathbf{J}_{i\to j}^{(n)}$. To each message $m_{i\to j}^{(n)}$ and marginal estimate $p_{\mathbf{x}_i}^{(n)}(\mathbf{x}_i)$ there are associated computation trees $T_{i\to j}^{(n)}$ and $T_i^{(n)}$ that summarize their pedigree. See Figure 6 for an example of a computation tree of $\mathcal{G}_1$.

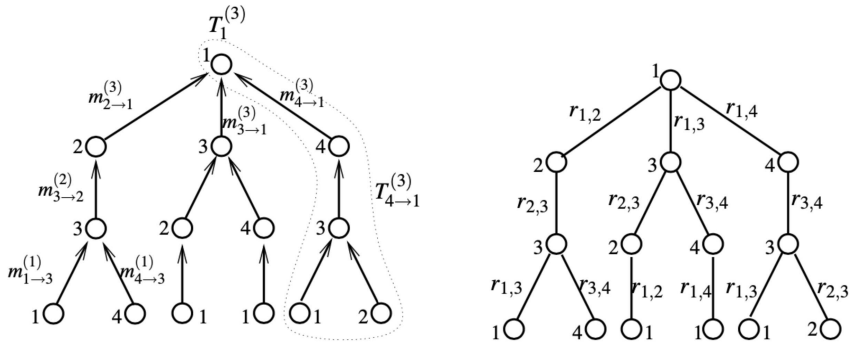


Figure 6: (Left:) The computation tree $T_1^{(3)}$ of node 1 in $\mathcal{G}_1$ (the subtree $T_{4\to1}^{(3)}$, associated with message $m_{4\to1}^{(3)}$, is also indicated within the dotted outline). (Right:) An equivalent tree model, with edge costs copied from Figure 4b, which has the same marginal at the root node as computed by loopy BP after 3 iterations.

• Loopy BP in Path-Weightable graphs: We will now see that for path-weightable graphs with cycles variances converge but not necessarily to the true values. Assume that the graph $\mathcal{G}$ has at least one cycle.

Let T_i denote the full computation tree (as $n \to \infty$). We assign the label 0 to the root node. Then, let $\mathbf{\Lambda}_{00}(T_i^{(n)})$ denote the variance at the root node of the n-th computation tree at node i in $\mathcal{G}$. The loopy BP variance estimate at node i after n iterations is equal to

$$\hat{\mathbf{\Lambda}}_{ii}^{(n)} = \mathbf{\Lambda}_{00}(T_i^{(n)}) = \phi(0 \to 0 | T_i^{(n)}) \tag{8}$$

Let $\hat{\mathbf{\Lambda}}_{ii}$ denote the loopy BP variance estimate at node i as $n \to \infty$.

Property 5. There is a one-to-one correspondence between finite-length paths in $\mathcal{G}$ that end at i, and paths in T_i that end at the root node. In particular, for each such path in $\mathcal{G}$, there is a corresponding path in $T_i^{(n)}$ for n large enough.

(d) (5 points) Show that the asymptotic loopy BP variance estimate i.e., $\hat{\mathbf{\Lambda}}_{ii}$ captures only a proper subset of the paths captured by the true variance.
Note: A proper subset of a set $\mathcal{S}$ is a subset of $\mathcal{S}$ that is not equal to $\mathcal{S}$.

We call the paths captured by both the asymptotic loopy BP variance estimate i.e., $\hat{\mathbf{\Lambda}}_{ii}$ and the true variance as *backtracking* paths.

Property 6. The asymptotic loopy BP variance estimate i.e., $\hat{\mathbf{\Lambda}}_{ii}$ converges to path-weights over the backtracking paths at each node.

(e) (5 points) Argue that with each loopy BP iteration, the variance estimate i.e., $\hat{\mathbf{\Lambda}}_{ii}^{(n)}$ grows monotonically.