10-708 PGM (Spring 2019): Homework 1 v1.1

Andrew ID: [no ID]
Name: [个人作答，仅供参考]

以下是我对这些题目的作答，仅供参考，如有错误疏漏，请批评指正。解答中所指“定理、定义 x.x”均出自 Koller & Friedman(2009) 教材。

1 Bayesian Networks [20 points] (Xun)

State True or False, and briefly justify your answer in a few sentences. You can cite theorems from Koller and Friedman (2009). Throughout the section, P is a distribution and $\mathcal{G}$ is a BN structure.

判断对错，并简要给出原因。可以引用 *Koller* 和 *Friedman*（*2009*）教材中的定理。在这一大题中，P 表示概率分布，$\mathcal{G}$ 表示贝叶斯网络结构。

1. **[2 points]** If $A \perp B \mid C$ and $A \perp C \mid B$, then $A \perp B$ and $A \perp C$. (Suppose the joint distribution of A, B, C is positive.)

Solution

错误。反例：$B = C$（无论 A 是否独立于 B 或 C）。
具体来说，从题目给出的两个条件独立出发，有：

$$P(A, B \mid C) = P(A \mid C)P(B \mid C)$$
$$P(A, C \mid B) = P(A \mid B)P(C \mid B)$$

而 A, B, C 的联合概率分布可以分别由 $(A, C \mid B)$ 和 $(A, B \mid C)$ 得出：

$$\begin{aligned} P(A, B, C) &= P(A, B \mid C)P(C) = P(A, C \mid B)P(B) \\ \text{(由条件独立性)} \quad &= P(A \mid C)P(B, C) = P(A \mid B)P(B, C) \end{aligned}$$

从而推得：

$$P(A \mid C) = P(A \mid B)$$

如果 $A \perp B$ 且 $A \perp C$，确实能导致上式成立，因为此时上式直接等价于 $P(A) = P(A)$。但这并不是唯一能使上式成立的情况：例如 $B = C$，此时无论 A 是否独立于 B, C，都能使上式成立。

$$
\begin{tikzcd}
A \arrow[r] & B \arrow[r] & C \\
D \arrow[ur] \arrow[r] & E
\end{tikzcd}
$$

Figure 1: A Bayesian network.

2. [**2 points**] In Figure 1, $E \perp C \mid B$.

> Solution
>
> **正确**。B 形成了对 E 和 C 的 d-分离。
> 条件概率形式的推导比较复杂，直接从 d-separation 的角度出发。C 与 E 之间唯一的通路是：
>
> $$E \leftarrow D \rightarrow B \rightarrow C$$
>
> 只有这条通路完全被打通，E 和 C 才相互（条件）**不**独立。而以 B 作为条件时，$D \rightarrow B \rightarrow C$ 的通路被阻断，此时根据 d-separation 原则（定理 3.4 的逆否命题），有 $E \perp C \mid B$ 成立。

3. [**2 points**] In Figure 1, $A \perp E \mid C$.

> Solution
>
> **错误**。在已知 C 的情况下，A 与 E 间不存在 d-分离。
> 尽管 B 作为 v-structure 的顶点并不存在于条件集里，但 C 是 B 的子节点（相当于也知道 B 的部分信息），从而使 $A \rightarrow B \leftarrow D$ 的路径开放。而由于并不控制 D 作为条件，$E \leftarrow D \rightarrow B$ 的路径也开放，因此从 A 到 E 的路径并未被阻断。因此，$A \not\perp E \mid C$。**【金标准，必然排除条件独立】**
> 这一命题可以通过将因果图亲缘化、无向化来近似地判断，去掉 C 后，能够看到 $A - D - E$ 这条无向边通路，并判断 A 与 E 并**不能保证条件独立**。参见附录中图 3开始的全过程。**【结果仅供参考，不能完全排除条件独立的可能性】**

$$
\begin{tikzcd}
P \text{ factorizes over } \mathcal{G} \arrow[r, Rightarrow, "(1)"] & \mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P) \arrow[r, Rightarrow, "(2)"] & \mathcal{I}_\ell(\mathcal{G}) \subseteq \mathcal{I}(P) \arrow[ll, Rightarrow, bend left, "(3)"]
\end{tikzcd}
$$

Figure 2: Some relations in Bayesian networks.

4. [**2 points**] In Figure 2, relation (1) is true.

Solution

正确。参照定理 3.1 和 3.2，命题”P factorizes over(according to) G” 是命题”$\mathcal{G}$ 是 P 的 I-map” 的充要条件；由定义 3.3 知，”$\mathcal{I}(\mathcal{G})$ 是 $\mathcal{I}(P)$ 的 I-map” 等价于 $\mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P)$。
顺便复习一下 $\mathcal{I}$ 的定义，$\mathcal{I}(\mathcal{G})$ 是贝叶斯网络 $\mathcal{G}$ 中含有的所有条件独立关系的集合。类似的，$\mathcal{I}(P)$ 是概率分布 P 中条件独立关系的集合。而如果 $\mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P)$，则称图 $\mathcal{G}$ 是概率分布 P 的 I-map。而 $\mathcal{I}_\ell(\mathcal{G})$ 则是贝叶斯网络 $\mathcal{G}$ 中的局部条件独立性假设的集合：

$$\mathcal{I}_\ell(\mathcal{G}) := \{X_i \perp NonDecendants_{X_i} \mid Pa_{X_i}, \forall i\}$$

5. [**2 points**] In Figure 2, relation (2) is true.

Solution

正确。$\mathcal{I}_\ell(\mathcal{G})$ 是 $\mathcal{G}$ 中的局部条件独立性，而贝叶斯网络中还可能存在其他条件独立性，即 $\mathcal{I}_\ell(\mathcal{G})$ 是 $\mathcal{I}(\mathcal{G})$ 的子集。由集合性质知该命题正确。
“其他的条件独立性” 例如：$A_3 \leftarrow A_2 \leftarrow A_1 \leftarrow P \rightarrow B_1 \rightarrow B_2 \rightarrow B_3$。在控制 A_1, P, B_1 中任意一点的条件下，A_3, B_3 条件独立，而 A_1, P, B_1 都不是 A_3, B_3 的父节点。

6. [**2 points**] In Figure 2, relation (3) is true.

Solution

正确。首先看 $\mathcal{I}_\ell(\mathcal{G})$ 与 $\mathcal{I}(\mathcal{G})$ 的关系，如果二者相等，则（3）必然成立。
如果 $\mathcal{I}_\ell(\mathcal{G})$ 是 $\mathcal{I}(\mathcal{G})$ 的真子集，如何？按照定义 3.2 和 3.3，$\mathcal{I}_\ell(\mathcal{G}) \subseteq \mathcal{I}(P)$ 时 $\mathcal{G}$ 即为 P 的 I-map，进而有（3）成立。

7. [**2 points**] If $\mathcal{G}$ is an I-map for P, then P may have extra conditional independencies than $\mathcal{G}$.

Solution

正确。$\mathcal{G}$ 是 P 的一个 I-map 的充要条件是：$\mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P)$，而等号并不总是成立，也就是 $\mathcal{I}(P)$ 中含有 $\mathcal{I}(\mathcal{G})$ 中不具有的某些条件独立性断言，这等价于题中命题。

8. [**2 points**] Two BN structures $\mathcal{G}_1$ and $\mathcal{G}_2$ are I-equivalent iff they have the same skeleton and the same set of v-structures.

Solution

错误。命题 "$\mathcal{G}_1$ 与 $\mathcal{G}_2$ 是 I-等价的" 的充要条件是 "$\mathcal{G}_1, \mathcal{G}_2$ 具有相同的框架 (定理 3.8)，与相同的无亲缘性集 (the same set of immorality)"。所谓的"immorality" 指的是对于 v-structure $X \to Z \leftarrow Y$，亲代节点 X, Y 之间没有有向边连接。

9. [**2 points**] The minimal I-map of a distribution is the I-map with fewest edges.

Solution

错误。最小 I-map 指的是不能再删除任何的边，而不是边最少。参考 Koller & Friedman(2009) 的图 3.8(b) 和 3.8(c)，这两个贝叶斯网络都是同一个分布的最小 I-map，但显然 (c) 图比 (b) 图多很多条边。

10. [**2 points**] The P-map of a distribution, if exists, is unique.

Solution

错误。对于同一个 P 来说，如果有 P-map，也不见得唯一。$X \to Z \to Y$, $Y \to Z \to X$ 代表了完全相同的条件独立性，即 $X \perp Y \mid Z$。对于这两张因果图完美对应的 P 来说，这两个因果图都是它的 P-map，二者具有 I-等价，但并不唯一。

2 Undirected Graphical Models [25 points] (Paul)

2.1 Local, Pairwise and Global Markov Properties [18 points]

1. Prove the following properties:
   - [**2 points**] If $A \perp (B, D) \mid C$ then $A \perp B \mid C$.
   - [**2 points**] If $A \perp (B, D) \mid C$ then $A \perp B \mid (C, D)$ and $A \perp D \mid (B, C)$.
   - [**2 points**] For strictly positive distributions, if $A \perp B \mid (C, D)$ and $A \perp C \mid (B, D)$ then $A \perp (B, C) \mid D$.

   Solution

   对上述三个性质的证明。
   性质 1:
   性质 2:
   性质 3:

2. [**6 points**] Show that for any undirected graph G and distribution P, if P factorizes according to G, then P will also satisfy the global Markov properties of G.

3. [**6 points**] Show that for any undirected graph G and distribution P, if P satisfies the local Markov property with respect to G, then P will also satisfy the pairwise Markov property of G.

2.2 Gaussian Graphical Models [7 points]

Now we consider a specific instance of undirected graphical models. Let $X = \{X_1, ..., X_d\}$ be a set of random variables and follow a joint Gaussian distribution $X \sim \mathcal{N}(\mu, \Lambda^{-1})$ where $\Lambda \in \mathbb{S}^{++}$ is the precision matrix. Let X_j, X_k be two nodes in X, and $Z = \{X_i \mid i \notin \{j, k\}\}$ denote the remaining nodes. Show that $X_j \perp X_k \mid Z$ if and only if $\Lambda_{jk} = 0$.

3 Exact Inference [40 points] (Xun)

3.1 Variable elimination on a grid [10 points]

Consider the following Markov network:

$$\begin{tikzcd}[arrows=-]
A \arrow[r] \arrow[d] & B \arrow[r] \arrow[d] & C \arrow[d] \\
D \arrow[r] \arrow[d] & E \arrow[r] \arrow[d] & F \arrow[d] \\
G \arrow[r] & H \arrow[r] & I
\end{tikzcd}$$

We are going to see how *tree-width*, a property of the graph, is related to the intrinsic complexity of variable elimination of a distribution.

1. [**2 points**] Write down largest clique(s) for the elimination order $E, D, H, F, B, A, G, I, C$.

   Solution

   The largest clique for order above is:

2. [**2 points**] Write down largest clique(s) for the elimination order $A, G, I, C, D, H, F, B, E$.
3. [**2 points**] Which of the above ordering is preferable? Explain briefly.
4. [**4 points**] Using this intuition, give a reasonable ($\ll n^2$) upper bound on the tree-width of the $n \times n$ grid.

3.2 Junction tree in action: part 1 [10 points]

Consider the following Bayesian network $\mathcal{G}$:

$$\begin{tikzcd}
A \arrow[r] \arrow[dr] & B \arrow[r] & C \arrow[d] \\
& E \arrow[r] & D
\end{tikzcd}$$

We are going to construct a junction tree $\mathcal{T}$ from $\mathcal{G}$. Please sketch the generated objects in each step.

1. [**1 pts**] Moralize $\mathcal{G}$ to construct an undirected graph $\mathcal{H}$.
2. [**3 pts**] Triangulate $\mathcal{H}$ to construct a chordal graph $\mathcal{H}^*$.

   (Although there are many ways to triangulate a graph, for the ease of grading, please use the triangulation that corresponds to the elimination order A, B, C, D, E.)

3. [**3 pts**] Construct a cluster graph $\mathcal{U}$ where each node is a maximal clique $\boldsymbol{C}_i$ from $\mathcal{H}^*$ and each edge is the sepset $\boldsymbol{S}_{i,j} = \boldsymbol{C}_i \cap \boldsymbol{C}_j$ between adjacent cliques $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$.
4. [**3 pts**] Run maximum spanning tree algorithm on $\mathcal{U}$ to construct a junction tree $\mathcal{T}$.

   (The cluster graph is small enough to calculate maximum spanning tree in one's head.)

3.3 Junction tree in action: part 2 [20 points]

Continuing from part 1, now assume all variables are binary and the CPDs are parameterized as follows:

A	$P(A)$
0	x_0

A	B	$P(B\|A)$
0	0	x_1
1	0	x_2

A	E	$P(E\|A)$
0	0	x_3
1	0	x_4

B	C	$P(C\|B)$
0	0	x_5
1	0	x_6

C	E	D	$P(D\|C,E)$
0	0	0	x_7
0	1	0	x_8
1	0	0	x_9
1	1	0	x_{10}

We are going to implement belief propagation on $\mathcal{T}$. The provided template `junction_tree.py` contains the following tasks:

- `initial_clique_potentials()`: Compute initial clique potentials $\psi_i(\boldsymbol{C}_i)$ from factors ϕ_i.
- `messages()`: Compute messages $\delta_{i\to j}$ from initial clique potentials $\psi_i(\boldsymbol{C}_i)$.
- `beliefs()`: Compute calibrated clique beliefs $\beta_i(\boldsymbol{C}_i)$ and sepset beliefs $\mu_{i,j}(\boldsymbol{S}_{i,j})$, using initial clique potentials $\psi_i(\boldsymbol{C}_i)$ and messages $\delta_{i\to j}$.
- Using the beliefs $\beta_i(\boldsymbol{C}_i), \mu_{i,j}(\boldsymbol{S}_{i,j})$, compute
  - `query1()`: $P(B)$
  - `query2()`: $P(A|C)$
  - `query3()`: $P(A, B, C, D, E)$

Please finish the unimplemented TODO blocks and submit completed `junction_tree.py` to Gradescope (`https://www.gradescope.com/courses/36025`).

In the implementation, please represent factors as `numpy.ndarray` and store different factors in a dictionary with its scope as the key. For example, as provided in the template, `phi['ab']` is a factor ϕ_{AB} represented as a 2×2 matrix, where `phi['ab'][0, 0]` $= \phi_{AB}(A = 0, B = 0) = P(B = 0|A = 0) = x_1$. For messages, one can use `delta['ab_cd']` to denote a message from AB to CD. Most functions can be written in 3 lines of code. You may find `np.einsum()` useful.

4 Parameter Learning [15 points] (Xun)

$$\begin{array}{ccccccc} Y_1 & \longrightarrow & Y_2 & \longrightarrow & \cdots & \longrightarrow & Y_T \\ \downarrow & & \downarrow & & & & \downarrow \\ X_1 & & X_2 & & & & X_T \end{array}$$

Consider an HMM with $Y_t \in [M]$, $X_t \in \mathbb{R}^K$ ($M, K \in \mathbb{N}$). Let $(\pi, A, \{\mu_i, \sigma_i^2\}_{i=1}^M)$ be its parameters, where $\pi \in \mathbb{R}^M$ is the initial state distribution, $A \in \mathbb{R}^{M \times M}$ is the transition matrix, $\mu_i \in \mathbb{R}^K$ and $\sigma_i^2 > 0$ are parameters of the emission distribution, which is defined to be an isotropic Gaussian. In other words,

$$P(Y_1 = i) = \pi_i \tag{1}$$

$$P(Y_{t+1} = j | Y_t = i) = A_{ij} \tag{2}$$

$$P(X_t | Y_t = i) = \mathcal{N}(X_t; \mu_i, \sigma_i^2 I). \tag{3}$$

We are going to implement the Baum-Welch (EM) algorithm that estimates parameters from data $\boldsymbol{X} \in \mathbb{R}^{N \times T \times K}$, which is a collection of N observed sequences of length T. Note that there are different forms of forward-backward algorithms, for instance the (α, γ)-recursion, which is slightly different from the (α, β)-recursion we saw in the class. For the ease of grading, however, please implement the (α, β) version, and remember to normalize the messages at each step for numerical stability.

Please complete the unimplemented TODO blocks in the template `baum_welch.py` and submit it to Gradescope (`https://www.gradescope.com/courses/36025`). The template has its own toy problem to verify the implementation. The test cases are ran on other randomly generated problem instances.

References

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

A 题目解答中提到的自制图表

A.1 问题 1.3：贝叶斯网络的亲缘化

判断 $A \perp E \mid C$ 的方法，比较方便的是改成亲缘图——当然，如果你记得住 do-calculus 里 v-structure 通路阻断的条件是“不知道顶点及其任何子节点”，是最好的。

亲缘图第一步是保留命题中出现的所有节点（待判断节点、条件节点）及其父节点的子图，删去无关的其他节点。但是这里待判断的恰好是 A, E，条件节点是 C，也就是整张图都要保留下来。

注：如果我们要判断 $A \perp E \mid B$，那么 C 以及 B, C 间的连线在这一步就可以去掉了。

$$\begin{tikzcd} A \arrow[r] & B \arrow[r] & C \\ D \arrow[ur] \arrow[r] & E \end{tikzcd}$$

Figure 3: 题目 1.3-原图（也是第一步处理后的图）

第二步：亲缘化，对于任意节点 X_i，如果它同时具有两个及以上的父节点（形成了以 X_i 为节点的 v-structure），那么父节点间两两连接为无向边。如果两个父节点间本就以有向边连接，则提前改为无向边吧——反正下一步所有的边都要改成无向的。在图 3中，需要连接的就是 A 和 D：

$$\begin{tikzcd} A \arrow[r] \arrow[d, dash] & B \arrow[r] & C \\ D \arrow[ur] \arrow[r] & E \end{tikzcd}$$

Figure 4: 题目 1.3-亲缘化

第三步，无向化。把所有的有向边都变成无向边，这个很简单。

第四步，删除条件节点及连接它们的路径，处理过程到此结束，参考下图。

$$\begin{tikzcd} A \arrow[r, dash] \arrow[d, dash] & B \\ D \arrow[ur, dash] \arrow[r, dash] & E \end{tikzcd}$$

Figure 5: 题目 1.3-无向化并删除条件节点

显然，A 与 E 之间仍然存在 $A - D - E$ 的无向链路，因此**无从保证二者在已知 C 时的条件独立性**。但**不能绝对的说二者一定不独立**，因为仍然存在数值独立的可能性，例如，$P(A \mid B)$ 与 $P(A)$ 对于 A, B 所有可能的取值均相等。

但反过来说，如果两个节点在亲缘化以后没有连在一起，那两个节点一定是条件独立的。