

# 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。

## 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:

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

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。

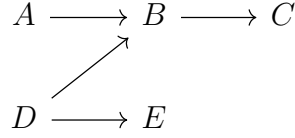


Figure 1: A Bayesian network.

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】**

$$P \text{ factorizes over } \mathcal{G} \xrightarrow{(1)} \mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P) \xrightarrow{(2)} \mathcal{I}_\ell(\mathcal{G}) \subseteq \mathcal{I}(P)$$

(3)

Figure 2: Some relations in Bayesian networks.

4. [2 points] In Figure 2, relation (1) is true.  
5. [2 points] In Figure 2, relation (2) is true.

6. [**2 points**] In Figure 2, relation (3) is true.
7. [**2 points**] If  $\mathcal{G}$  is an I-map for  $P$ , then  $P$  may have extra conditional independencies than  $\mathcal{G}$ .
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.
9. [**2 points**] The minimal I-map of a distribution is the I-map with fewest edges.
10. [**2 points**] The P-map of a distribution, if exists, is unique.

## 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  $A \perp (B, C) \mid D$ .

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  $G$ .

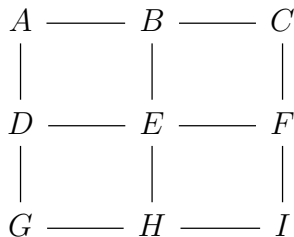
### 2.2 Gaussian Graphical Models [7 points]

Now we consider a specific instance of undirected graphical models. Let  $X = \{X_1, \dots, X_d\}$  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:

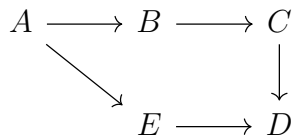


We are going to see how *tree-width*, a property of the graph, is related to the intrinsic complexity of variable elimination of a distribution.

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



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  $\mathbf{C}_i$  from  $\mathcal{H}^*$  and each edge is the sepset  $\mathbf{S}_{i,j} = \mathbf{C}_i \cap \mathbf{C}_j$  between adjacent cliques  $\mathbf{C}_i$  and  $\mathbf{C}_j$ .

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:

| $A$ | $P(A)$ | $A$ | $B$ | $P(B A)$ | $A$ | $E$ | $P(E A)$ | $B$ | $C$ | $P(C B)$ | $C$ | $E$ | $D$ | $P(D C, E)$ |
|-----|--------|-----|-----|----------|-----|-----|----------|-----|-----|----------|-----|-----|-----|-------------|
| 0   | $x_0$  | 0   | 0   | $x_1$    | 0   | 0   | $x_3$    | 0   | 0   | $x_5$    | 0   | 0   | 0   | $x_7$       |
|     |        | 1   | 0   | $x_2$    | 1   | 0   | $x_4$    | 1   | 0   | $x_6$    | 0   | 1   | 0   | $x_8$       |
|     |        |     |     |          |     |     |          |     |     |          | 1   | 0   | 0   | $x_9$       |
|     |        |     |     |          |     |     |          |     |     |          | 1   | 1   | 0   | $x_{10}$    |

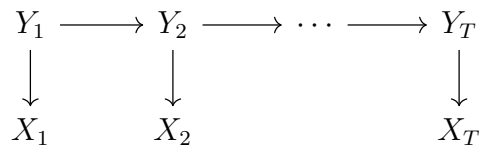
We are going to implement belief propagation on  $\mathcal{T}$ . The provided template `junction_tree.py` contains the following tasks:

- `initial_clique_potentials()`: Compute initial clique potentials  $\psi_i(\mathbf{C}_i)$  from factors  $\phi_i$ .
- `messages()`: Compute messages  $\delta_{i \rightarrow j}$  from initial clique potentials  $\psi_i(\mathbf{C}_i)$ .
- `beliefs()`: Compute calibrated clique beliefs  $\beta_i(\mathbf{C}_i)$  and sepset beliefs  $\mu_{i,j}(\mathbf{S}_{i,j})$ , using initial clique potentials  $\psi_i(\mathbf{C}_i)$  and messages  $\delta_{i \rightarrow j}$ .
- Using the beliefs  $\beta_i(\mathbf{C}_i), \mu_{i,j}(\mathbf{S}_{i,j})$ , 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  $\phi_{AB}$  represented as a  $2 \times 2$  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)



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  $\mathbf{X} \in \mathbb{R}^{N \times T \times K}$ , which is a collection of  $N$  observed sequences of length  $T$ . Note that there are different forms of forward-backward algorithms, for instance the  $(\alpha, \gamma)$ -recursion, which is slightly different from the  $(\alpha, \beta)$ -recursion we saw in the class. For the ease of grading, however, please implement the  $(\alpha, \beta)$  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。

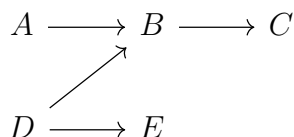


Figure 3: 题目 1.3-原图（也是第一步处理后的图）

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  $D$ ：

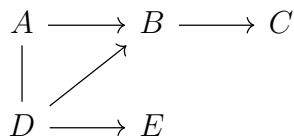


Figure 4: 题目 1.3-亲缘化

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。

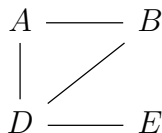


Figure 5: 题目 1.3-无向化并删除条件节点

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。