2024 秋季高级机器学习
习题三

221300079 王俊童

2024.12.12

一. (40 points) **概率图模型**

1. (20 points) 图 1是一个贝叶斯网络结构，请仿照教材 14.4.1 变量消去部分内容，推断图中边际概率 $P(x_5)$.

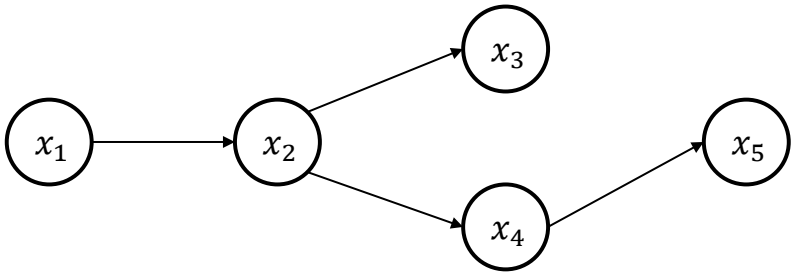

图 1: 贝叶斯网络结构

2. (20 points) 本题探究变分推断相关内容. 我们利用教材中相同的设定，假设当前有 N 个变量 $\{x_1, x_2, \ldots, x_N\}\}$ 均依赖于其他变量 z，所有能观察到的变量"的联合分布的概率密度函数是:

$$p(\mathbf{x} \mid \Theta) = \prod_{i=1}^{N} \sum_{\mathbf{z}} p(x_i, \mathbf{z} \mid \Theta), \tag{1}$$

而所对应的对数似然函数为:

$$\ln p(\mathbf{x} \mid \Theta) = \sum_{i=1}^{N} \ln \left\{ \sum_{\mathbf{z}} p(x_i, \mathbf{z} \mid \Theta) \right\}, \tag{2}$$

其中 $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}, \Theta$ 是 $\mathbf{x}$ 与 $\mathbf{z}$ 服从的分布参数.

我们的推断任务是求解 $p(\mathbf{z} \mid \mathbf{x}, \Theta)$ 和 Θ. 一种有效手段是基于最大化对数似然函数，对（2）式使用 EM 算法：在 E 步，根据 t 时刻的参数 Θ^t 对 $p(\mathbf{z} \mid \mathbf{x}, \Theta^t)$ 进行推断，并计算联合似然函数 $p(\mathbf{x}, \mathbf{z} \mid \Theta)$；在 M 步，基于 E 步的结果进行最大化寻优，即对关于变量 Θ 的函数 $Q(\Theta; \Theta^t)$

进行最大化从而求取：

$$\Theta^{t+1} = \arg\max_{\Theta} Q(\Theta; \Theta^t) \tag{3}$$

$$= \arg\max_{\Theta} \sum_{\mathbf{z}} p(\mathbf{z} \mid \mathbf{x}, \Theta^t) \ln p(\mathbf{x}, \mathbf{z} \mid \Theta). \tag{4}$$

（1）（10 points）$p(\mathbf{z} \mid \mathbf{x}, \Theta^t)$ 未必是隐变量 $\mathbf{z}$ 服从的真实分布，而只是一个近似分布. 现在将这个近似分布用 $q(\mathbf{z})$ 表示，请尝试验证

$$\ln p(\mathbf{x}) = \mathcal{L}(q) + KL(q \parallel p), \tag{5}$$

其中

$$\mathcal{L}(q) = \int q(\mathbf{z}) \ln \left\{ \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} \right\} d\mathbf{z}, \tag{6}$$

$$KL(q \parallel p) = -\int q(\mathbf{z}) \ln \left\{ \frac{p(\mathbf{z} \mid \mathbf{x})}{q(\mathbf{z})} \right\} d\mathbf{z}. \tag{7}$$

（2）（10 points）假设复杂的多变量 Z 可拆解为一系列相互独立的多变量 Z_i, 即 Z 服从分布：

$$q(\mathbf{z}) = \prod_{i=1}^{M} q_i(\mathbf{z}_i), \tag{8}$$

尝试从最大化 $\mathcal{L}(q)$ 的角度说明变量子集 $\mathbf{z}_j$ 所服从的最优分布 q_j^* 应满足

$$\ln q_j^*(\mathbf{z}_j) = \mathbb{E}_{i \neq j}[\ln p(\mathbf{x}, \mathbf{z})] + \text{const}. \tag{9}$$

解：

1. 对于这个题首先可以得到由联合概率分布求和的结果:

$$P(x_5) = \sum_{x_4} \sum_{x_3} \sum_{x_2} \sum_{x_1} P(x_1, x_2, x_3, x_4, x_5)$$

进一步推导得到:

$$P(x_5) = \sum_{x_4} \sum_{x_3} \sum_{x_2} \sum_{x_1} P(x_1)P(x_2|x_1)P(x_3|x_2)P(x_4|x_2)P(x_5|x_4)$$

所以根据书上说的，我们可以按照顺序计算加法，假设:m_{ij} 代表的是求加过程的中间结果，下标 i 代表此项对于 x_i 求加的结果，j 代表其他变量，则我们可以化简一些步骤例如：

$$P(x_5) = \sum_{x_4} P(x_5|x_4) \sum_{x_3} P(x_3|x_2) \sum_{x_2} P(x_4|x_2) \sum_{x_1} P(x_1)P(x_2|x_1)$$

其中比如：$\sum_{x_1} P(x_1)P(x_2|x_1) = m_{12}$, 所以我们可以得到最终结果为:

$$P(x_5) = m_{54}(x_5)$$

具体其中的化简步骤可以一步一步推导来，比如：$m_{23} = \sum_{x_3} P(x_3|x_2)m_{12}$

2.（1）. 如果要证明这个公式，我们首先从基于 $\boldsymbol{x}$ 的公式说起:

$$p(\boldsymbol{x}) = \frac{p(\boldsymbol{x}, \boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})}$$

所以两边同时变换再除以 $q(\boldsymbol{z})$:

$$p(\boldsymbol{x}) = \frac{p(\boldsymbol{x}, \boldsymbol{z})/q(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})/q(\boldsymbol{z})}$$

然后取对数:

$$\ln p(\boldsymbol{x}) = \ln \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} - \ln \frac{p(\boldsymbol{z}|\boldsymbol{x})}{q(\boldsymbol{z})}$$

两边同时乘上 $q(\boldsymbol{z})$, 并且积分:

$$\int q(\boldsymbol{z}) \ln p(\boldsymbol{x})\, d\boldsymbol{z} = \int q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})}\, d\boldsymbol{z} - \int q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{z}|\boldsymbol{x})}{q(\boldsymbol{z})}\, d\boldsymbol{z}$$

化简就得到了:

$$\ln p(\boldsymbol{x}) = \int q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})}\, d\boldsymbol{z} - \int q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{z}|\boldsymbol{x})}{q(\boldsymbol{z})}\, d\boldsymbol{z}$$

这个式子的形式就是上面所示:

$$\ln p(\boldsymbol{x}) = L(q) + KL(q||p)$$

（2）. 因为要从最大化 L(q) 的角度来说明这个问题，我们把 L(q) 拿出来:

$$L(q) = \int q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})}\, d\boldsymbol{z}$$

化简利用题目中条件可以得到:

$$L(q) = \int \prod_{i=1}^{M} q_i(\boldsymbol{z}_i) \{\ln p(\boldsymbol{x}, \boldsymbol{z}) - \ln \prod_{i=1}^{M} q_i(\boldsymbol{z}_i)\}\, d\boldsymbol{z}$$

$$L(q) = \int \prod_{i=1}^{M} q_i(\boldsymbol{z}_i) \ln p(\boldsymbol{x}, \boldsymbol{z}) - \prod_{i=1}^{M} q_i(\boldsymbol{z}_i) \ln \prod_{i=1}^{M} q_i(\boldsymbol{z}_i)\, d\boldsymbol{z} = L_1(q) + L_2(q)$$

先看第一部分:

$$L_1(q) = \int \prod_{i=1}^{M} q_i(\boldsymbol{z}_i) \ln p(\boldsymbol{x}, \boldsymbol{z}) = \int q_j \{\int \ln p(\boldsymbol{x}, \boldsymbol{z}) \prod_{i \neq j}^{M} (q_i(\boldsymbol{z}_i) d\boldsymbol{z_i})\} d\boldsymbol{z_j}$$

令第一部分的 $\ln \hat{p}(\boldsymbol{x}, \boldsymbol{z_j}) = \int \ln p(\boldsymbol{x}, \boldsymbol{z}) \prod_{i \neq j}^{M} (q_i(\boldsymbol{z}_i) d\boldsymbol{z_i})$, 则原式子等于:

$$L_1(q) = \int q_j \ln \hat{p}(\boldsymbol{x}, \boldsymbol{z_j})\, d\boldsymbol{z_j}$$

对于第二部分

$$L_2(q) = \int \prod_{i=1}^{M} q_i(\boldsymbol{z}_i) \ln \prod_{i=1}^{M} q_i(\boldsymbol{z}_i)\, d\boldsymbol{z} = \sum_{i1=1}^{M} \int \prod_{i2=1}^{M} q_{i2}(\boldsymbol{z_{i2}}) \ln q_{i1}(\boldsymbol{z_{i1}})\, d\boldsymbol{z}$$

令 i1 = j, 则 $\int \prod_{i2=1}^{M} q_{i2}(\boldsymbol{z_{i2}}) \ln q_{i1}(\boldsymbol{z_{i1}})\, d\boldsymbol{z} = \int q_j(\boldsymbol{z_j}) \ln q_j(\boldsymbol{z_j})\, d\boldsymbol{z_j}$, 则式子二可以化简为:

$$L_2(q) = \int q_j(\boldsymbol{z_j}) \ln q_j(\boldsymbol{z_j})\, d\boldsymbol{z_j} + const$$

所以 L(q) 可以表示为:

$$L(q) = -KL(q_j || \hat{p}(\boldsymbol{x}, \boldsymbol{z_j})) + const$$

当且仅当，$q_j = \hat{p}(\boldsymbol{x}, \boldsymbol{z_j})$ 时，KL 散度最小，此时估计值最近似。为了证明它满足这个分布，我们看之前得到的一个式子

$$\int q_j \{ \int \ln p(\boldsymbol{x}, \boldsymbol{z}) \prod_{i \neq j}^{M} (q_i(\boldsymbol{z}_i) d\boldsymbol{z_i}) \} d\boldsymbol{z_j} = \int q_j \mathbb{E}_{i \neq j}[\ln p(\mathbf{x}, \mathbf{z})]\, d\boldsymbol{z_j}$$

$$= \int q_j \hat{p}(\boldsymbol{x}, \boldsymbol{z_j})\, d\boldsymbol{z_j} - const$$

所以:

$$\ln q_j^*(\mathbf{z}_j) = \ln \hat{p}(\boldsymbol{x}, \boldsymbol{z_j}) = \mathbb{E}_{i \neq j}[\ln p(\mathbf{x}, \mathbf{z})] + \text{const.}$$

则证明了，从最大化 L(q) 的角度来说，子集 z_j 的最优分布满足上述条件。

二. (60 points) **强化学习**

1. (25 points) 价值迭代的更新公式为:

$$V^{k+1}(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s' \mid s, a)\, V^k(s') \right\}, \tag{10}$$

其中 s 表示 t 时刻的状态，s' 表示 $t+1$ 时刻的状态，a 表示 t 时刻的动作，γ 是折扣因子. 我们将其定义为一个贝尔曼最优算子 $\mathcal{T}$:

$$V^{k+1}(s) = \mathcal{T}V^k(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s' \mid s, a)\, V^k(s') \right\} \tag{11}$$

若 O 是一个算子，如果满足 $\|OV - OV'\|_q \leq \|V - V'\|_q$ 条件，则我们称 O 是一个压缩算子，其中 $\|x\|_q$ 表示 x 的 L_q 范数.

(1)（15 points）请证明，当 $\gamma < 1$ 时，贝尔曼最优算子 $\mathcal{T}$ 是一个 γ-压缩算子.（提示：证明 $\|\mathcal{T}V - \mathcal{T}V'\|_\infty \leq \gamma \|V - V'\|_\infty$ 即可）

(2)（10 points）在（1）的基础上，请说明价值迭代的收敛性.（提示：可以设最优价值函数为 V^*，考虑 $\left\|V^k - V^*\right\|_\infty$ 与迭代次数 k 的联系）

2. (15 points) 本题探究蒙特卡罗强化学习算法中的策略.

(1)（8 points）请你描述**重要性采样**的过程. 具体来说，我们希望估计某个函数 $f(x)$ 在概率分布 $p(x)$ 下的期望，但是 $p(x)$ 采样困难. 如何引入一个更容易采样的分布 $q(x)$ 来协助估计?

（2）（7 points）同策略蒙特卡罗强化学习算法和异策略蒙特卡罗强化学习算法有何差异？请你结合上一问中提到的方法进行讨论.

3. (20 points) 时序差分学习（TD 学习）是一种在强化学习中广泛应用的核心技术，结合了动态规划和蒙特卡洛方法的优点，用于估计策略的价值函数. 它通过直接从与环境的交互中学习，既不需要完整的模型，也无需等待整条轨迹结束即可更新估计. 这种特性使 TD 学习在在线学习和实时决策任务中非常高效. 教材中介绍了一种属于 TD 学习的经典算法-Sarsa 算法，下方为完整算法流程：

输入： 环境 E;
动作空间 A;
起始状态 x_0;
奖赏折扣 γ;
更新步长 α.

过程：

$Q(x,a)=0,\ \pi(x,a)=\frac{1}{|A(x)|}$;
$x=x_0,\ a=\pi(x)$;
for $t=1,2,\ldots$ **do**
  $r,x'=$ 在 E 中执行动作 a 产生的奖赏与转移的状态;
  $a'=\pi^{\epsilon}(x')$;
  $Q(x,a)=Q(x,a)+\alpha\big(r+\gamma Q(x',a')-Q(x,a)\big)$;
  $\pi(x)=\arg\max_{a''} Q(x,a'')$;
  $x=x',a=a'$
end for

输出： 策略 π

图 2: Sarsa 算法

结合状态值函数与状态-动作值函数的关系以及动态规划的特点，我们可以得到：

$$Q^{\pi}(x,a) = \sum_{x' \in X} P^{a}_{x \to x'} \left(R^{a}_{x \to x'} + \gamma V^{\pi}(x') \right) \tag{12}$$

$$= \sum_{x' \in X} P^{a}_{x \to x'} \left(R^{a}_{x \to x'} + \gamma \sum_{a' \in A} \pi(x',a') Q^{\pi}(x',a') \right). \tag{13}$$

请你根据式 (15),(16), 尝试推理出 Sarsa 算法的更新公式，即图二中的步骤 6.

解：

1.（1）. 首先给出我们需要的 TV 和 TV' 的定义式子：

$$TV = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P\left(s' \mid s,a\right) V\left(s'\right) \right\}$$

$$TV' = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P\left(s' \mid s,a\right) V'\left(s'\right) \right\}$$

则根据三角不等式:

$$|TV - TV'| \leq \max_{a \in \mathcal{A}} \gamma \left\{ \sum_{s' \in \mathcal{S}} P\left(s' \mid s,a\right) \left(V\left(s'\right) - V'\left(s'\right)\right) \right\}$$

对于其中 $\sum_{s' \in \mathcal{S}} P\left(s' \mid s,a\right) = 1$ 而且根据凸优化和无穷范数的定义可以得到:

$$||V - V'||_{\infty} = sup_{s \in S} |V(s) - V'(s)| = sup_{s \in S} \left\{ \sum_{s' \in \mathcal{S}} P\left(s' \mid s,a\right) \left(V\left(s'\right) - V'\left(s'\right)\right) \right\}$$

所以带入原式子可以得到:

$$|TV - TV'| \leq \gamma ||V - V'||_{\infty}$$

那么根据原式子可以得到:

$$||TV - TV'||_{\infty} = sup_{s \in S} |TV - TV'| \leq \gamma ||V - V'||_{\infty}$$

所以即证明了 bellman 算子是一个压缩算子。

（2）. 对于这个问题，由题目可以得到:

$$V^{*} = TV^{*}, V^{k+1} = TV^{k}$$

所以可以有:

$$||V^{k+1} - V^{*}||_{\infty} = ||TV^{k} - TV^{*}||_{\infty} \leq \gamma ||V^{k} - V^{*}||_{\infty}$$

所以根据压缩算子的迭代:

$$||V^{k} - V^{*}||_{\infty} \leq \gamma^{k} ||V^{0} - V^{*}||_{\infty}$$

因为 $\gamma < 1$，所以经过 k 轮次迭代之后，$\gamma^k \approx 0$, 则:

$$\gamma^k ||V^0 - V^*||_\infty \approx 0$$

所以原式子证明则有:

$$||V^k - V^*||_\infty \approx 0$$

所以收敛性证明完毕。

2. (1) . 对于如何引入一个 q(x) 来帮助我们估计，这种情况存在于异策略蒙特卡洛强化学习里面。对于异策略，行动策略和目标策略不是一个策略的时候，相当于我们在同策略的时候，我们用的是 $\epsilon -$ 来做的，所有的动作策略也基于这个贪心，而现在在异策略的情况下，只用贪心来做衡量，而选择不同的动作策略。假设当前有两个不同策略来产生轨迹的时候，被采样概率就会不一样。

根据题目假设，假如 p(x) 采样困难，我们可以引入一个 q(x) 策略，他的采样比较简单，来估计 E(q(x)) 的期望。这就是重要性采样:

$$E[f] = \int_x p(x) f(x)\, dx$$

那么此时的期望就是 (从 p 上采样估计的):

$$\hat{E}[f] = \frac{1}{m} \sum_{i=1}^{m} f(x)$$

但是如果 p(x) 很难做，我们引入一个 q(x):

$$E[f] = \int_x q(x) \frac{p(x)}{q(x)} f(x)\, dx$$

此时期望就是:

$$\hat{E}[f] = \frac{1}{m} \sum_{i=1}^{m} \frac{p(x_i)}{q(x_i)} f(x_i)$$

这样我们就通过了另一个分布采样来估计原不是很好估计的分布采样下的期望，这就是重要性采样。

(2) . 事实上，同策略蒙特卡洛强化学习并不会用到重要性采样这个方法，就跟我在上一问说的一样:

对于同策略蒙特卡洛算法，对采样轨迹中的每一对状态-动作，记录其后的奖赏值之和，作为该状态-动作的一次累积奖赏，通过多次采样后，使用累积奖赏的平均作为状态-动作值的估计，并引入 -贪心策略保证采样的多样性。这种算法保证了策略和动作是一样的。

而对于异策略蒙特卡洛算法，我们只是用贪心去衡量策略评估的时候，但是对于算法的策略改进，并不利用原始策略，即可以理解为目标策略和行动策略并不是同一个。

综合看下来就是同策略的其实并不需要调整权重之类的，如果我们把 $\frac{P^\pi}{P^{\pi'}} = \prod_{i=0}^{T-1} \frac{\pi(x_i, a_i)}{\pi'(x_i, a_i)}$ 看作一个权重的话，而异策略其实更考虑到新策略（更简单的）带来的影响并进行学习。这也可能导致，同策略对于简单环境更容易学习，而异策略对于复杂环境的适应性更高。

3.