

极客大学机器学习训练营

机器学习基本概念

王然

众微科技 AI Lab 负责人

二〇二一年一月十六日

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- ▶ AI 的语言 → 不理解数学，不可能理解模型
- ▶ 创新的根基 → 看起来创新不多，但是实际上有很多地方可以创新，而且创新没有那么难
- ▶ 数学锻炼思维

- ▶ 把数学当做语言：不管它的意思，严格按照要求 → 我们主要讲方法
- ▶ 数学真正的学法，是以证明为目的的

核心：

- ▶ Frame and Hypotheses
- ▶ Elements and Relationships
- ▶ Patterns
- ▶ Intuition
- ▶ Retrospect and Empathetic
- ▶ Bucket(In/Out/New)
- ▶ Strategic minds

- ▶ 机器学习的各种角度和建模流程
- ▶ 概率论和统计学基础概念复习
- ▶ 极大似然体系和 EM 算法
- ▶ 贝叶斯体系和 Variational Bayes 算法
- ▶ 矩阵代数：基本概念复习和 Tensor 求导

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- ▶ 最终目的：效果好，即准确性高
- ▶ 为了达到最终目的，必须从不同角度考虑

- ▶ 最简单的是视角
- ▶ 目标：给定 X 预测 y
- ▶ 假设：存在真实的 $y = f_0(X)$
- ▶ 如果知道 f_0 ，那么就不需要做任工作
- ▶ 但是我们不知道，所以需要逼近

- ▶ 观测 $\{X_i, y_i; i \in \mathcal{I}\}$
- ▶ 可以假设 $f \in \mathcal{F}$
- ▶ 目标：给定一个损失函数 c , 最小化 $\sum_i c(f(X_i), y_i)$
- ▶ 这个估计可以称之为 \hat{f}

什么样的 \hat{f} 是好的

- ▶ 最理想状况 $\hat{f} = f_0$; 事实上 (可能) 不可能
- ▶ 不可能原因 (一): 没有所有的 x 和 y 的组合
- ▶ 不可能原因 (二): $f_0 \notin \mathcal{F}$
- ▶ 不可能原因 (三): 求解 \hat{f} 时候有困难
- ▶ 但是基本启示是: 要找到一个足够大的 \mathcal{F} 使它包含 f_0 , 并且要求 \mathcal{F} 应该足够小使得求解比较容易 \rightarrow 自相矛盾

- ▶ 本质上来说，世界上是随机的
- ▶ 随机的来源：
 - ▶ 缺乏信息 → 最主要问题，在表格化数据中最为明显
 - ▶ 测量误差 → 大部分信息都有误差
 - ▶ 比如说年龄 800 岁，收入 400 万亿
 - ▶ 模型误差 → 假设模型形式和现实的差别
 - ▶ 估计误差 → 得到模型过程中造成的误差
 - ▶ 优化误差 → 求解过程中的误差
 - ▶ 评估误差 → 评估本身也存在误差

- ▶ 假设目标是用身高预测体重
- ▶ 为什么不可以进行插值？

请思考

- ▶ 缺乏信息：人有胖有瘦，仅仅给定身高，不可能判断
- ▶ 导致结果：如果要求身高必须解释体重，身高就承担了非理性的要求
- ▶ 相关结果：bias 较大
- ▶ 统计学根本区别于函数逼近的原因
 - ▶ 函数逼近： $y = f_0(X)$
 - ▶ 统计学 $y = f_0(X) + \epsilon$

- ▶ Bias: 话说得很详细, 但是很不准
 - ▶ 北京明天下午两点四十分会发生里氏 2.6 级地震
- ▶ Variance: 含糊其词, 但是很准
 - ▶ 在这个世界上有一天会发生地震
- ▶ 往往存在 Bias 和 Variance 的权衡 (但这不是全部, 它本身的数学理论只是针对回归的)
- ▶ Bias 大: 过拟合
- ▶ Variance 大: 欠拟合

- ▶ 往往难以处理
- ▶ 是数据预处理一个重要部分

- ▶ 假设背景：存在一个上帝知道的真实的模型，但他不知道部分误差，所以模型一定会有损失
 - ▶ 但就该损失函数而言，这个真实的模型一定是预测最好的
- ▶ 现实情况：因为我们不知道真实的模型，所以只能采用一些模型来逼近
 - ▶ 一般情况下不知道真实模型，只能选择一般的模型 → 估计方差大

- ▶ 即使对于同样的模型或问题，也有不同办法得到模型的参数
 - ▶ 极大似然估计和贝叶斯估计
 - ▶ 增强学习中的 Q-learning 和 Policy Gradient
- ▶ 好的方法可以减少其中误差

- ▶ 求解的过程，就是迭代的过程
- ▶ 迭代是否会收敛是一个很大的问题
- ▶ 在神经网络中尤其明显，但在传统模型中也存在

- ▶ 因为不知道真实的损失函数（除非有无限多的测试样本），所以必须评估
- ▶ 评估的越多，训练样本就越少 → 出现了交叉验证的概念
- ▶ 注意避免不公平的评估

- ▶ 只用训练集 → 不公平
- ▶ 无数次的测试训练集 → 不可以（否则猜就可以了）
- ▶ 建模数据和实际场景不同：在 2019 年建模预测 2020 年上半年旅游业情况

- ▶ **重要原则：**一定要看评估本身的误差多大，然后决定做法是否有提升
- ▶ **重要提示：**
- ▶ 越是误差小的领域，需要概率角度越多
- ▶ 误差大的领域，概率角度可能不能帮上太多忙，更应该找可以优化的地方

- ▶ 从概率理论上来说，预训练不应该有任何帮助：预训练和当前任务无关(?)，而且模型表达力没有变
- ▶ 预训练是深度学习最重要发明之一
 - ▶ 例子：从一个字预测出词语和预测情感没关系
 - ▶ 现实：预测词语表示了对语义的理解，所以对预测情感有帮助
 - ▶ 从优化的角度来说：有利于优化

- ▶ 很多问题要 case-by-case 分析
- ▶ 重点：从不同角度出发（数学思维）
- ▶ 从不同角度看同一个问题：其他角度的进展也可以帮助解决这个问题

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- ▶ 概率论是描述随机的语言
- ▶ 概率论分为朴素概率论和公理性概率论
- ▶ 主要讲朴素概率论

- ▶ 一维离散意味着可以直接讨论概率
- ▶ 一维离散意味着可以假设概率取值只是整数
- ▶ 例子：男 = 1, 女 = 2, 未知 = 3
 - ▶ $P(X < 3) = \dots$
 - ▶ $p(X = 1) = \dots$
 - ▶ $P(X \leq x) = \sum_{i \leq x} p(X = i)$, 或者用更标准的写法 $P(X \leq t) = \sum_{x \leq t} p(x)$

- ▶ 连续意味着可能性至少不是有限的
- ▶ 还是可以定义 $P(X \leq x)$
- ▶ 但是定义 $p(x)$ 的时候就有问题了

思考：为什么？

- ▶ 在给定一个连续变量时，只能定义 $P(X \leq m) = \int_{-\infty}^m p(x) dx$
- ▶ 虽然离散和连续的定义有所不同，但是积分本身就是一种非常复杂的加法
- ▶ $F_X(t) := P(X \leq t)$ 就是所谓的概率 Cumulative Distribution Function
- ▶ $p(x)$ 就是所谓的 Probability Density Function，不是概率值

- ▶ 以二维为例: $P(X \leq m, Y \leq n) = \int_{-\infty}^m \int_{-\infty}^n p(x, y) dx dy$
- ▶ 对于边际分布 $p(x) = \int p(x, y) dy$
- ▶ 条件概率 $p(x|y) = p(x, y) / p(y)$

给定一个概率密度函数 $p(x)$ ，再给定一个函数 $f(x)$ ，我们定义他的数学期望 (Expectation) 为

$$E_p[f(X)] := \int f(x)p(x)dx$$

给定一个条件概率密度函数 $p(x|y)$ ，再给定一个函数 $f(x)$ ，我们定义他的条件数学期望 (Conditional Expectation) 为

$$E_p[f(X)|Y] := \int f(x)p(x|y)dx$$

$$\begin{aligned}E_Y[E_p(f(X))|Y] &= E_Y\left[\int f(x)p(x|y)dy\right] \\&= \iint f(x)p(x|y)p(y)dxdy \\&= \iint f(x)p(x,y)dydx \\&= \int f(x) \int p(x,y)dydx \\&= \int f(x)p(x)dx \\&= E_p[f(X)]\end{aligned}$$

练习：手推贝叶斯公式

$$p(y|x) = \frac{p(y)p(x|y)}{\int p(x|y)p(y)dy}$$

- ▶ Multinomial: $P(X = x_i) = p_i$
- ▶ 正态分布: $p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$, 其中 μ 是 σ 是参数。这时候, 我们常常写成 $X \sim N(\mu, \sigma^2)$
- ▶ 其它常见的概率分布可以参见Shao (2003)

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
 - 极大似然估计基本思路 ■ （可选）EM 算法和 HMM
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
 - 极大似然估计基本思路 ■ （可选）EM 算法和 HMM
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- ▶ 考虑最简单的情况，即掷一个不公平的硬币
- ▶ 每一个硬币向上的概率为 $p(x_i)$ ，用 $y_i = 1$ 记载硬币向上
- ▶ 就此得到硬币向下的概率为 $1 - p(x_i)$ ，用 $y_i = 0$ 表示
- ▶ 整体观测到目前情况的概率为 $p(x_i)^{y_i} \times (1 - p(x_i))^{(1-y_i)}$ ，这就是所谓的**似然函数**
- ▶ 这个形式比较难看，所以不妨取个 \log ，这就是**对数似然函数**：
 $y_i \log(p(x_i)) + (1 - y_i) \log(1 - p(x_i))$

思考：什么是好的 p

- ▶ 如果我们知道 p ，那什么都不用做
- ▶ 但实际上我们既不知道 p ，还想知道什么是一个好的 p
- ▶ 假设只抛一次硬币，思考下面哪个概率更好？
 - ▶ 一个估计 p 的似然函数为 0.3
 - ▶ 另一个估计 p 的似然函数为 0.9

- ▶ 找到使目前似然函数最大的那个观测
- ▶ 或者由于对数变换是单调变化，找到负的对数似然函数最小的那个

- ▶ 只抛一次硬币，当然没有任何做推断的价值
- ▶ 现在假设我们抛 N 次硬币，得到观测 $\{x_i, y_i; i \leq N\}$
- ▶ 继续假定每次抛硬币的结果不影响下一次抛硬币的概率分布，即观测独立
- ▶ 则似然函数为 $\prod_i p(x_i)^{y_i} (1 - p(x_i))^{(1-y_i)}$
- ▶ 连乘带来的问题：因为如果连乘一个 0 到 1 之间的数，得到的乘积会越来越小，特别小的时候，电脑就会出现数值问题（比如说 10^{-10} 的负十万次方）

- ▶ 取个 \log 即可: $\log(xy) = \log(x) + \log(y)$ 。
- ▶ 则负的对数似然函数为: $-\sum_i (y_i \log(p(x_i)) + (1 - y_i) \log(1 - p(x_i)))$
- ▶ 眼熟吗? 这就是 Binary Cross Entropy

如何选择 $p(x_i)$ 的形式

- ▶ $p(x_i)$ 长什么样呢？
- ▶ 首先，要控制 $p(x_i)$ 取值在 0 到 1 之间
- ▶ 一个常见选择 $p(x_i) = \frac{1}{1+\exp(-f(x_i))}$
- ▶ 如果 $f(x_i) = \sum_k \beta_k x_{ik}$ ，其中 β_k 为未知参数（需要求解），则得到了逻辑回归的数学表达形式
- ▶ 注意：这种 f 的函数形式被称之为线性函数，近似于多个线性函数组合的函数是最重要的一类函数形式

- ▶ 现在假设有 y_i , 服从期望为 $f(x_i)$ 且方差为 1 的正态分布
- ▶ 也就是说 $p(y_i) = \frac{1}{\sqrt{2\pi}} \exp(-(y_i - f(x_i))^2/2)$
- ▶ 让我们来共同推导他的对数似然函数!

5 分钟尝试推导时间...

我们需要的负的对数似然函数等于

$$-\sum_i \log p(x_i) = -\sum_i (-(y_i - f(x_i))^2)/2 + K$$

其中 K 是一个跟 f 无关的常数，所以这里最小化的距离是 $\sum_i (y_i - f(x_i))^2$ ，
这就是**最小二乘法**

- ▶ 第一种情况，称之为二分类分类问题。对应多分类问题也可以进行对应推导
- ▶ 第二种情况，称之为回归问题
- ▶ 大部分机器学习工程师假设世界上只存在这两种问题，但是事实上，其他问题多的很（即使在监督学习框架下）

- ▶ 目标：小企业贷款额度确定
- ▶ 考虑方向：
 - ▶ 违规可能性：要把风险控制一定范围内
 - ▶ 需求：对贷款需求越高的企业应该给更多贷款
- ▶ 第一个问题可以作为分类问题解决
- ▶ 第二个问题不好解决

- ▶ 虽然观测不到企业的真实需求，但可以假设存在一个真实需求
- ▶ 我们知道实际放款额和实际使用金额，所以存在两种情况：
 - ▶ 放款额度大于实际使用金额，这时可以假定实际需求即为实际使用金额
 - ▶ 放款额度等于实际使用金额，这时虽然不知道实际需求，但是知道实际需求一定大于等于放款额度

- ▶ 假设真实需求为 y_i^*
- ▶ 进一步假设 $y_i^* = f(x_i) + \epsilon_i$, 且 ϵ_i 为正态分布
- ▶ 建行所给的真实的额度假设为 y_i
- ▶ 当发生截断时, 其似然函数为 $P(y_i^* \geq y_i)$
- ▶ 当不发生截断时, 其似然函数为 $p(y_i)$
- ▶ 两者结合, 即可以得到估计方式

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
 - 极大似然估计基本思路 ■ (可选) EM 算法和 HMM
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

- ▶ 通常情况下，在极大似然框架中，如果容易推导出对数似然函数的话，那么求解将会非常容易
- ▶ 但是如果存在隐变量，则推导变得非常困难
- ▶ 在一些情况下，EM 算法是解决隐变量问题的一个非常通用的框架（现实情况少见）
- ▶ 但部分大厂面试喜欢要求推导 HMM 的估计方式

- ▶ HMM 算法的估计方法称之为 Baum-Welch 算法
- ▶ 换句话说，这是两个数学家折腾好几年折腾出来的东西
- ▶ 即使知道 HMM 的推导思路，我个人在复现的时候也至少推导了三次，花了一天时间且中间还参考了各种讲义
- ▶ 如果我完全不知道推导思路，仅仅知道该算法是可以推导的，我最少也得花一个月时间才能搞清楚（保守估计）
- ▶ 相比之下，当我知道 Axiom Of Choice 等价于 Zorn's Lemma 时候，我只用了三天时间就推导出来

- ▶ 现场去“推导”该算法是不可能的
- ▶ 现场去“默写”该算法是有可能的
- ▶ 默写跟数学能力毫无关系

考虑以下关系：用 $l(\theta; X)$ 表示对数似然函数，则

$$\begin{aligned}l(\theta; X) &= \log p_{\theta}(X) \\&= \log \int p_{\theta}(X, y) dy \\&= \log \int \frac{p_{\theta}(X, y)}{p_{\tilde{\theta}}(y|X)} p_{\tilde{\theta}}(y|X) dy \\&\geq \int \log(p_{\theta}(X, y)) p_{\tilde{\theta}}(y|X) dy - \int \log(p_{\tilde{\theta}}(y|X)) p_{\tilde{\theta}}(y|X) dy \\&= E_{\tilde{\theta}}[\log p_{\theta}(X, y)|X] - E_{\tilde{\theta}}[\log p_{\tilde{\theta}}(y|X)|X]\end{aligned}$$

其中

注意在这里：

- ▶ y 是一个隐变量
- ▶ $\tilde{\theta}$ 是当前的估计，目标是通过迭代的方法找到下一步的估计 θ ，换句话说，因为 $E_{\tilde{\theta}}[\log P_{\tilde{\theta}}(y|X)|X]$ 跟 θ 没有关系，所以可以忽略
- ▶ 定义 $Q(\theta, \tilde{\theta}) = E_{\tilde{\theta}}[\log p_{\theta}(X, y)|X]$ ，则 EM 算法可以定义为
 - ▶ 计算 $Q(\theta, \tilde{\theta})$
 - ▶ 最大化 θ

- ▶ 假设对于每一个观测 d 可以观测到 $\{X_t^{(d)}, 1 \leq t \leq T\}$
- ▶ 它的概率分布取决于隐变量 $z_t^{(d)}$ 。并且该变量服从马尔可夫性质，因此，如果知道 $t-1$ 的信息，就不需要知道更早的信息，就可以得到 $z_t^{(d)}$ 的性质
- ▶ 假设 X 's 和 z 's 都只能取有限多个值

我们有

$$P(z, \mathcal{X}; \theta) = \prod_{d=1}^D \left(\pi_{z_1^{(d)}} B_{z_1^{(d)}} \left(x_1^{(d)} \right) \prod_{t=2}^T A_{z_{t-1}^{(d)} z_t^{(d)}} B_{z_t^{(d)}} \left(x_t^{(d)} \right) \right)$$

对上式取 \log 之后

$$\begin{aligned}\log P(z, \mathcal{X}; \theta) = & \sum_{d=1}^D [\log \pi_{z_1^{(d)}} + \sum_{t=2}^T \log A_{z_{t-1}^{(d)} z_t^{(d)}} \\ & + \sum_{t=1}^T \log B_{z_t^{(d)}}(x_t^{(d)})]\end{aligned}$$

放到 Q 函数中，假设目前的参数 θ^s ：

$$\begin{aligned} Q(\theta, \theta^s) = & \sum_{z \in \mathcal{Z}} \sum_{d=1}^D \log \pi_{z_1^{(d)}} P(z, \mathcal{X}; \theta^s) \\ & + \sum_{z \in \mathcal{Z}} \sum_{d=1}^D \sum_{t=2}^T \log A_{z_{t-1}^{(d)} z_t^{(d)}} P(z, \mathcal{X}; \theta^s) \\ & + \sum_{z \in \mathcal{Z}} \sum_{d=1}^D \sum_{t=1}^T \log B_{z_t^{(d)}} \left(x_t^{(d)} \right) P(z, \mathcal{X}; \theta^s) \end{aligned}$$

加上拉格朗日乘子：

$$\begin{aligned}\hat{L}(\theta, \theta^s) &:= Q(\theta, \theta^s) - \lambda_{\pi} \left(\sum_{i=1}^M \pi_i - 1 \right) \\ &\quad - \sum_{i=1}^M \lambda_{A_i} \left(\sum_{j=1}^M A_{ij} - 1 \right) \\ &\quad - \sum_{i=1}^M \lambda_{B_i} \left(\sum_{j=1}^N B_i(j) - 1 \right)\end{aligned}$$

下面让我们来首先求解 π_i 。

$$\begin{aligned}\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \pi_i} &= \frac{\partial}{\partial \pi_i} \left(\sum_{z \in \mathcal{Z}} \sum_{d=1}^D \log \pi_{z_1^{(d)}} P(z, \mathcal{X}; \theta^s) \right) - \lambda_\pi = 0 \\ &= \frac{\partial}{\partial \pi_i} \left(\sum_{j=1}^M \sum_{d=1}^D \log \pi_j P(z_1^{(d)} = j, \mathcal{X}; \theta^s) \right) - \lambda_\pi = 0 \\ &= \sum_{d=1}^D \frac{P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{\pi_i} - \lambda_\pi = 0\end{aligned}$$

$$\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \lambda_{\pi}} = - \left(\sum_{i=1}^M \pi_i - 1 \right) = 0$$

求解，我们可以得到

$$\begin{aligned}\pi_i &= \frac{\sum_{d=1}^D P\left(z_1^{(d)} = i, \mathcal{X}; \theta^s\right)}{\sum_{j=1}^M \sum_{d=1}^D P\left(z_1^{(d)} = j, \mathcal{X}; \theta^s\right)} = \frac{\sum_{d=1}^D P\left(z_1^{(d)} = i, \mathcal{X}; \theta^s\right)}{\sum_{d=1}^D \sum_{j=1}^M P\left(z_1^{(d)} = j, \mathcal{X}; \theta^s\right)} \\&= \frac{\sum_{d=1}^D P\left(z_1^{(d)} = i, \mathcal{X}; \theta^s\right)}{\sum_{d=1}^D P(\mathcal{X}; \theta^s)} = \frac{\sum_{d=1}^D P\left(z_1^{(d)} = i, \mathcal{X}; \theta^s\right)}{DP(\mathcal{X}; \theta^s)} \\&= \frac{\sum_{d=1}^D P(\mathcal{X}; \theta^s) P\left(z_1^{(d)} = i \mid \mathcal{X}; \theta^s\right)}{DP(\mathcal{X}; \theta^s)} = \frac{1}{D} \sum_{d=1}^D P\left(z_1^{(d)} = i \mid \mathcal{X}; \theta^s\right) \\&= \frac{1}{D} \sum_{d=1}^D P\left(z_1^{(d)} = i \mid \mathcal{X}^{(d)}; \theta^s\right)\end{aligned}$$

采用类似方法：

$$\begin{aligned} A_{ij} &= \frac{\sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s\right)}{\sum_{j=1}^M \sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s\right)} \\ &= \frac{\sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s\right)}{\sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i, \mathcal{X}; \theta^s\right)} \\ &= \frac{\sum_{d=1}^D \sum_{t=2}^T P(\mathcal{X}; \theta^s) P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j \mid \mathcal{X}; \theta^s\right)}{\sum_{d=1}^D \sum_{t=2}^T P(\mathcal{X}; \theta^s) P\left(z_{t-1}^{(d)} = i \mid \mathcal{X}; \theta^s\right)} \\ &= \frac{\sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j \mid \mathcal{X}^{(d)}; \theta^s\right)}{\sum_{d=1}^D \sum_{t=2}^T P\left(z_{t-1}^{(d)} = i \mid \mathcal{X}^{(d)}; \theta^s\right)} \end{aligned}$$

$$\begin{aligned} B_i(j) &= \frac{\sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i, \mathcal{X}; \theta^s\right) I\left(x_t^{(d)} = j\right)}{\sum_{j=1}^N \sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i, \mathcal{X}; \theta^s\right) I\left(x_t^{(d)} = j\right)} \\ &= \frac{\sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i, \mathcal{X}; \theta^s\right) I\left(x_t^{(d)} = j\right)}{\sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i, \mathcal{X}; \theta^s\right)} \\ &= \frac{\sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i \mid \mathcal{X}^{(d)}; \theta^s\right) I\left(x_t^{(d)} = j\right)}{\sum_{d=1}^D \sum_{t=1}^T P\left(z_t^{(d)} = i \mid \mathcal{X}^{(d)}; \theta^s\right)} \end{aligned}$$

- ▶ 为什么要推导 $P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j \mid X^{(d)}; \theta^s\right)$ 和 $P\left(z_t^{(d)} = i \mid X^{(d)}; \theta^s\right)$
- ▶ 这是因为这两者可以用动态规划很容易求解
- ▶ 细节作为（可选）练习题

- ▶ $P\left(z_{t-1}^{(d)} = i, z_t^{(d)} = j \mid X^{(d)}; \theta^s\right)$ 和 $P\left(z_t^{(d)} = i \mid X^{(d)}; \theta^s\right)$ 可以动态求解有效
动态求解这件事情不可能一眼看出来，甚至我们在开始推导的时候也不可能考虑到动态求解的问题
- ▶ 在这个推导过程中，如果不知道我们目标是推出这两个量的表达式，则可能会在几百个可能性当中折腾很久
- ▶ 如果知道，起码这道题在一天内还是有可能做出来的（也有可能很快做出来）
- ▶ 所以如果仅仅从推导过程来看，推导过程并不长。但是假如某个“业界大牛”告诉你他手推了一个小时就“推导”出来了，那他大概是把“推导”跟“默写”搞错了

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
 - 贝叶斯估计 ■ 变分贝叶斯法
- 6 矩阵和张量求导

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
 - 贝叶斯估计 ■ 变分贝叶斯法
- 6 矩阵和张量求导

- ▶ 在之前所有的模型中，我们均假设有所谓的真实参数或模型，目的只是推导这个真实的模型
- ▶ 贝叶斯学派的视角不同：
 - ▶ 假设参数是 θ ，我们将会对其有一个 prior，表示为 $p(\theta)$ ，换句话说 θ 本身就是随机
 - ▶ 现在得到了观测： X ，目标是得到 posterior： $p(\theta|X)$
- ▶ 根据贝叶斯公式，我们有

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{\int p(X|\theta)p(\theta)d\theta}$$

假设 $\mu \sim N(0, 1)$, $X|\mu \sim N(\mu, 1)$, 我们一起来推导 μ 的 posterior

$$\begin{aligned} p(\mu|X) &\propto \exp(-\mu^2/2)) \exp(-\sum_i (X_i - \mu)^2/2) \\ &\propto \exp(-(\frac{N+1}{2}\mu^2 - \mu \sum_i X_i)) \\ &\propto \exp\left(-(\mu^2 - \frac{2\sum_i X_i}{N+1}\mu)/(\frac{2}{N+1})\right) \\ &\propto \exp\left((\mu - \frac{\sum_i X_i}{N+1})^2 / \frac{2}{N+1}\right) \end{aligned}$$

- ▶ $\mu|X \sim N(\frac{\sum_i x_i}{N+1}, \frac{1}{(N+1)^2})$ 。
- ▶ 因此, posterior 也是正态分布
- ▶ 这称之为 Conjugate Priors

- ▶ 好处：
 - ▶ 很方便的处理隐变量
 - ▶ 可以对不确定性进行估计
- ▶ 坏处：计算麻烦 → 就目前深度学习应用来说，最方便的是变分法。我们将通过介绍 VAE 的方式介绍该方法

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
 - 贝叶斯估计 ■ 变分贝叶斯法
- 6 矩阵和张量求导

- ▶ 咱们现场来推导
- ▶ 你们先来十分钟，然后我们一起看

准备好了吗？

证明下式：

$$\log P(X) - \mathcal{D}[Q(z | X) \| P(z | X)] = E_{z \sim Q}[\log P(X | z)] - \mathcal{D}[Q(z | X) \| P(z)]$$

其中 \mathcal{D} 为 KL-divergence (请自己查是什么)

十分钟，然后我们一起玩

- 1 怎样学数学
- 2 机器学习的各种角度和建模流程
- 3 概率论和统计学复习
- 4 极大似然估计
- 5 贝叶斯估计和变分贝叶斯
- 6 矩阵和张量求导

见附件