re0: 从零开始的 VAE tutorial

黑山老妖

2021.02.09

Contents

前言

这是一个可能会 tj 的关于 VAE 的 tutorial。写这个的原因是因为一个搞工程的朋友问我 Auto-encoder（AE）和 VAE 的区别。因为 VAE 涉及的基础知识比较多，我没信心拿支笔就给讲清楚，所以就准备写一些草稿，列一些前置知识，也方便听完之后回看。

结果因为上班太忙，写了一半，一拖再拖，至今没给人家讲，乃至于这个笔记也可能要 tj 了，就挂上来分享一下。还有写东西没写，和 PG 的关系啥的，有空闲会来补完，不过大部分要点都写完了，看完这个再看原文应该不至于感到人生艰难，希望不会辜负打开这个文件的同学。

在我看来 AE 和 VAE 是两种不同研究风格的产物：

- AE 像是炼丹风格的产物，“加点这个和那个，然后这样应该能 work，果然可以，那我们来想点理论来解释一下“（我也不知道现在有没有理论来解释了）；
- VAE 只是隐变量模型的 (Amortized Variational Inference)AVI 的一个例子产物，后者才是文章的主要贡献，但是因为 VAE 广为人知，“A 推出 B，推出 C，应该能 work，来我们找个应用场景，生成模型挺适合的，就这么干”;

因为两者算法过程和名字上都很相近，容易被当作相似的东西，个人更喜欢第二种研究风格。这篇文章程度的数学，可能也会被归入“完全不懂数学”的类别里，但至少听起来更加 solid，有理有据。

当然，当前神经网络的研究领域，可能还是第一种风格占主导，也有了很多卓越的工作。记得某个名人说机器学习（还是编程）是技术与艺术的结合学科，第一种风格显然更属于艺术的范畴，当然这里就有艺术大师和车间工人的区别了。

1 相关基础

1. 贝叶斯公式
2. 决策理论
3. 采样方法

2 贝叶斯公式

贝叶斯是一种概率的解释 (Interpretation).

$$\underbrace{p(W|X)}_{posterior} = \frac{p(X,W)}{p(X)} = \frac{p(X|W)p(W)}{p(X)} \tag{1}$$

$$\propto \underbrace{p(X|W)}_{likelihood}\underbrace{p(W)}_{prior} \tag{2}$$

3 概率密度估计 $\rightarrow$ 分类器训练

有了概率密度函数，就能根据决策理论得到一个规则，作为分类器。PRML Sec 1.5 [1]

一般来说，概率密度是需要估计的。如果我们知道真实的概率密度函数 $p(x, y)$，得到的最优规则，称为贝叶斯最优。

如果得到了 $\hat{p}(y|x)$，一个分类器就可以直接通过规则得到，

$$\hat{y} = \arg\max_{y} \hat{p}(y|x)$$

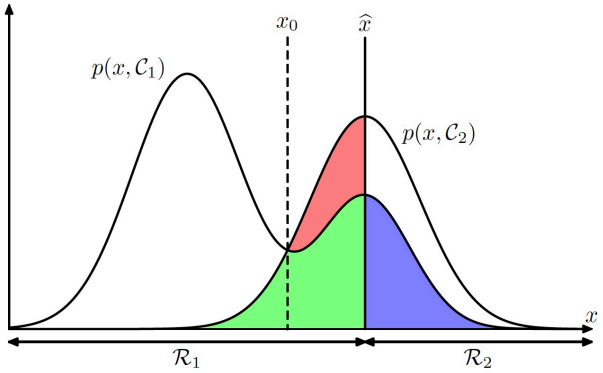

Figure 1.24 Schematic illustration of the joint probabilities $p(x, \mathcal{C}_k)$ for each of two classes plotted against x, together with the decision boundary $x = \widehat{x}$. Values of $x \geqslant \widehat{x}$ are classified as class $\mathcal{C}_2$ and hence belong to decision region $\mathcal{R}_2$, whereas points $x < \widehat{x}$ are classified as $\mathcal{C}_1$ and belong to $\mathcal{R}_1$. Errors arise from the blue, green, and red regions, so that for $x < \widehat{x}$ the errors are due to points from class $\mathcal{C}_2$ being misclassified as $\mathcal{C}_1$ (represented by the sum of the red and green regions), and conversely for points in the region $x \geqslant \widehat{x}$ the errors are due to points from class $\mathcal{C}_1$ being misclassified as $\mathcal{C}_2$ (represented by the blue region). As we vary the location $\widehat{x}$ of the decision boundary, the combined areas of the blue and green regions remains constant, whereas the size of the red region varies. The optimal choice for $\widehat{x}$ is where the curves for $p(x, \mathcal{C}_1)$ and $p(x, \mathcal{C}_2)$ cross, corresponding to $\widehat{x} = x_0$, because in this case the red region disappears. This is equivalent to the minimum misclassification rate decision rule, which assigns each value of x to the class having the higher posterior probability $p(\mathcal{C}_k|x)$.

图 1: linear attention illustration

3.1 交叉熵 loss 等价于 KL 散度

KL 散度描述两个分布的相似度，最小化 empirical risk 等价于最小化 “模型预测分布与样本分布” 的 KL 散度。AGSLT Sec 1.1.2 [2]

略。

4 采样方法

4.1 什么是采样

用样本均值来近似期望，用一个简单的式子来说明，

$$E_X[f(X)] \approx \frac{1}{L} \sum_{l} f(X^l), \tag{3}$$

X^l 是对 X 的采样样本

4.2 如何获取符合某个分布的样本

不同的采样方法都提现在如何获取符合分布的样本上。简单的采样方法只需要理解一个面积相等准则，基本上可以看图说话，

$$f(x)dx = g(z)dz$$

这里只需要知道 importance sampling, 参考 PRML Sec 11.1.4 [1]

(MCMC 里的 MH 采样 PRML 写的不好，可以翻 MLAPP [3])

5 混合模型 (Mixture model)

Mixture model 是指包含隐变量的模型，一度非常流行，很多叫得上名字的模型都属于混合模型：

GMM, HMM, pLSA, LDA, pPCA, hard attention

目标函数：

$$p(X;\theta) = \sum_Z p(X, Z;\theta) \tag{4}$$

略。

6 混合模型的优化：EM 算法，采样，variational, Amortized inference

6.1 Evidence lower bound

先放公式，稍后从采样的角度解释一下为什么要这么做。

这个是 VAE 和其他混合模型里用的最多的公式，写的比较详细。

（以下说明用 $\sum$ 来表达离散隐变量，连续隐变量换成 $\int$ 即可; 省略了 θ; 分类任务中 X 可以认为是

$[X, Y]$。）

$$\log p(X) = \log \sum_Z p(X, Z) \tag{5}$$

$$= \log \sum_Z p(X, Z) \frac{q(Z)}{q(Z)} \qquad \text{Sec 6.3.1会解释为什么引入} q \tag{6}$$

$$= \log E_q[\frac{p(X, Z)}{q(Z)}] \tag{7}$$

$$\geq E_q[\log \frac{p(X, Z)}{q(Z)}] \tag{8}$$

$$= \sum_Z q(Z) \log \frac{p(X, Z)}{q(Z)} \qquad \text{jensen 不等式} \tag{9}$$

实际上，可以推出多出的那一项 delta，接 eq 7:

$$= \sum_Z q(Z) \log \frac{p(X, Z)}{q(Z)} + S, \tag{10}$$

可以简单的推出 S,

$$S = \log p(X) - \sum_Z q(Z) \log \frac{p(X, Z)}{q(Z)} \tag{11}$$

$$= \sum_Z q(Z) \log p(X) - \sum_Z q(Z) \log \frac{p(X, Z)}{q(Z)} \tag{12}$$

$$= \sum_Z q(Z) \log p(X) \frac{q(Z)}{p(X, Z)} \tag{13}$$

$$= \sum_Z q(Z) \log \frac{q(Z)}{\frac{p(X,Z)}{p(X)}} \tag{14}$$

$$= \sum_Z q(Z) \log \frac{q(Z)}{p(Z|X)} \tag{15}$$

$$= -\sum_Z q(Z) \log \frac{p(Z|X)}{q(Z)} \tag{16}$$

$$= \mathrm{KL}(q(Z)||p(Z|X)) \tag{17}$$

最后，把 eq 16带入 eq 10, 顺便给所有 p 补上 θ, 可以得到，

$$\log p(X) = \log \sum_Z p(X, Z|\theta) \tag{18}$$

$$= \underbrace{\sum_Z q(Z) \log \frac{p(X, Z|\theta)}{q(Z)}}_{\text{ELBO}=\mathcal{L}(q,\theta)} \underbrace{- \sum_Z q(Z) \log \frac{p(Z|X, \theta)}{q(Z)}}_{KL(q(Z)||p(Z|X,\theta))} \tag{19}$$

$$\tag{20}$$

$KL(q||p) \geq 0$, 当 p, q 处处相等的时候取等号。所以 eq 19是 $p(X)$ 的一个下界，称为 evidence lower bound(ELBO).

6.1.1 EM

到这里，我们的目标变成了最大化下界 ELBO $= \mathcal{L}(q, \theta)$, 并且当 $KL(q||p) = 0$ 时，$p(X|\theta) = \mathcal{L}(q, \theta)$.

因为 eq 19是个恒等式，固定 θ 的情况下，减小 KL，就会增大 ELBO。这里有个方便的地方，KL 的最小值是已知的，即 $p = q$。于是就有了 EM 算法：

Algorithm 1: EM

while not convergent **do**
- E-step. $\max_q$ ELBO $\Rightarrow KL(q||p) = 0 \Rightarrow q = p$;
- M-step. $\max_\theta$ ELBO, (这一步 θ 变化，生了一个新的 $p(Z|X, \theta)$, 导致 $p \neq q$, $KL(q||p) \neq 0$, 又可以迭代上一步)

M-step 中，优化这个下界的好处是，可以通过采样近似期望，或者经常会有解析解，配合 KL 散度的优化，可以得到一个比较稳定的优化过程 [4]。

上述的 E-M 步骤可以理解为用坐标下降法 (coordinate descent) 的方式来优化 ELBO: (`https://en.wikipedia.org/wiki/Coordinate_descent`).

Algorithm 2: Coordinate descent for $\mathcal{L}(q, \theta)$

while not convergent **do**
- $\max_q \mathcal{L}(q, \theta) \Rightarrow KL(q||p) = 0 \Rightarrow q = p$;
- $\max_\theta \mathcal{L}(q, \theta)$

实际中, 很多时候无法得到 $p(Z|X)$ 的解析形式, 也就是说无法找到一个可计算的 q, 使 $KL(q||p) = 0$. 也就是说这里没法用坐标下降来直接到 q 的精确解了。

这时候我们会一般会使用近似的方法，给 q 做某种限定, 然后直接优化 $\mathcal{L}(q, \theta)$：

1. 分解 q(mean field 方法)，用泛函的方式直接优化函数 q;
2. 用参数化分布的 q 去近似 p, $q(Z) = q(Z|\phi)$, VAE 用神经网络来近似 q，并且针对连续型 Z, 用了一个变量替换的技巧

6.1.2 小结

总结一下，根据不同的模型设定，这里可以细分出 EM，VB，EP，以及 VAE 里用的 amortized 方法：

1. EM: 如果后验 $p(Z|X)$ 求解比较方便，或者有解析解，我们可以直接用 EM，或者 Generalized EM，或者配合采样来求解。

2. 如果后验不好求，可以用近似的方法，直接优化 ELBO，

$$\mathcal{L}(q,\theta)=\sum_Z q(Z)\log\frac{p(X,Z)}{q(Z)}$$

常用的近似方法有：

(a) mean field: 一般 Variational Bayesian(VB)/EP 都是指这类方法（可以避开积分问题）假设 $q(Z)=\prod_i q(Z_i)=\prod_i q_i$，然后使用变分法直接对 q_i 求导

(b) Amortized inference+ 采样：对 $q(Z)$ 采样，用样本均值近似期望

i. 变量替换 trick：针对连续型 Z，VAE 使用的方法，VB 需要为每个 x_i 设置一个参数，可以认为是一种 transductive 方法，无法推广到没见过的数据，amortized 方法学习一个全局的映射 $q(z|x)$，减少参数，并且属于 inductive 方法，可以应用到新数据，在表达力上会比 VB 弱。

ii. 策略梯度方法 (policy gradient): 针对离散型 Z

(这里补个脑图)

6.2 EM

当 $p(Z|X)$ 有解析解时，优化 KL 散度就等于使 $q(Z)=p(Z|X)$.

交替优化 ELBO 和 $KL(q||p)$. (eq 19)

EM 算法不详述了，这里列一下结果，方便后面讨论，细节可以参考 PRML Sec 9.4 [1].

Algorithm 3: EM: $\max p(X;\theta)$

Data: X

Result: θ

initialize θ^{old};

while $\underline{\text{dist}(p(X;\theta^{\text{old}}),p(X;\theta^{\text{new}}))>\epsilon}$ **do**

1. E step: 求 $p(Z|X,\theta^{\text{old}})$;

2. M step:

$$\theta^{\text{new}}=\arg\max_\theta \mathcal{Q}(\theta,\theta^{\text{old}}) \tag{21}$$

where

$$\mathcal{Q}(\theta,\theta^{\text{old}})=\sum_Z p(Z|X,\theta^{\text{old}})\log p(X,Z|\theta) \tag{22}$$

3. $\theta^{\text{old}}=\theta^{\text{new}}$

6.3 采样法

这节描述当后验 $p(Z|X)$ 不好求，或者后验的期望 $E_{p(Z|X)}$，怎么用采样的方式近似期望，顺便从采样的角度解释下 eq 6为什么引入 q

6.3.1 从零开始的 q

先忘记前面的 sec 6.1的公式，从最初的采样的角度一步步看为什么引入 q

首先，混合模型的假设中，隐变量 Z 需要积分掉，最简单的贝叶斯推导如下，

$$
\begin{aligned}
p(X) &= \sum_Z p(X, Z) && (23)\\
&= \sum_Z p(X|Z)p(Z) && (24)\\
&= E_Z[p(X|Z)] && (25)\\
&\approx \frac{1}{L}\sum_l p(X|Z^l) && (26)
\end{aligned}
$$

这里用样本均值代替期望 (Monte Carlo 方法)，采用 Z 的先验来采样，但是先验和 X 无关，容易采到不好的样本点，方差很大。（白板画图）

一种做法就是使用一个和 X 相关的分布来采样，然后用重要性系数来去偏，

$$
\begin{aligned}
p(X) &= \sum_Z p(X, Z) && (27)\\
&= \sum_Z p(X|Z)\frac{p(Z)}{Q(Z|X)}Q(Z|X) && (28)\\
&= E_Q[p(X|Z)\frac{p(Z)}{Q(Z|X)}] && (29)\\
&= E_Q[\frac{p(X,Z)}{Q(Z|X)}] && (30)\\
&\approx \frac{1}{L}\sum_l \frac{p(X,Z^l)}{Q(Z^l|X)} && (31)
\end{aligned}
$$

这里就回到了 eq 6

References

[1] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics). Berlin, Heidelberg: Springer-Verlag, 2006.

[2] S. Watanabe, Algebraic Geometry and Statistical Learning Theory. USA: Cambridge University Press, 2009.

[3] K. P. Murphy, Machine learning : a probabilistic perspective. Cambridge, Mass. [u.a.]: MIT Press, 2013.

[4] A. Gepperth and B. Pfülb, "Gradient-based training of Gaussian Mixture Models in High-Dimensional Spaces," arXiv:1912.09379 [cs, stat], June 2020. arXiv: 1912.09379.