生成模型介绍

Shwj

2023 年 2 月 20 日

1 目录与计划

- 介绍一下信息论的基础内容 KL 散度之类的
- 从 EM 算法开始讲起讲一下 EM 算法
- https://blog.csdn.net/junruitian/article/details/104406067

2 数学基础

2.1 KL 散度和信息论

KL 散度起源于信息论。信息论的主要目标是量化数据中有多少信息。信息论中最重要的指标称为熵，通常表示为 H。概率分布的熵的定义是

$$H = -\sum_{i=1}^{N} p(x_i) \log p(x_i)$$

我们可以把熵解释为“我们编码信息所需要的最小比特数”，熵的关键在于，只要知道所需位数的理论下限，我们就可以准确地量化数据中有多少信息。

KL 散度，是一个用来衡量两个概率分布的相似性的一个度量指标。我们知道，**现实世界里的任何观察都可以看成表示成信息和数据**，一般来说，我们无法获取数据的总体，我们只能拿到数据的部分样本，根据数据的部分样本，我们会对数据的整体做一个近似的估计，而数据整体本身有一个

真实的分布（我们可能永远无法知道）。那么**近似估计的概率分布和数据整体真实的概率分布的相似度**，或者说差异程度，可以用 KL 散度来表示。对于两个离散性随机变量 $x \sim p(x)$ 和 $x \sim q(x)$Kullback-Leibler divergence 定义为

$$D_{KL}(p||q) = \sum_{i=1}^{N} p(x_i)(\log p(x_i) - \log q(x_i))$$

而当 p 和 q 为连续随机变量时

$$D_{KL}(p||q) = \int_{-\infty}^{+\infty} p(x) \ln \frac{p(x)}{q(x)}$$

KL 散度不满足对称性，即 $D_{KL}(p||q) \neq D_{KL}(q||p)$ 所以它不能作为一种度量函数。KL 散度可以衡量使用基于 Q 的分布来编码服从 P 的分布的样本所需的额外的平均比特数。典型情况下，P 表示数据的真实分布，Q 表示数据的理论分布、估计的模型分布、或 P 的近似分布。[1]

KL 散度很重要的性质是 $D_{KL}(p||q) \geq 0$ 因为“使用一种编码方式编码另一套符号总会有损失”。也可以依靠 $\ln x \leq x - 1(x > 0)$, 有

$$\begin{aligned} -D_{KL}(p||q) &= \sum_{i=1}^{N} p(x_i) \ln \frac{q(x_i)}{p(x_i)} \leq \sum_{i=1}^{N} p(x_i)(\frac{q(x_i)}{p(x_i)} - 1) \\ &= \sum_{i=1}^{N} q(x_i) - p(x_i) = \sum_{i=1}^{N} q(x_i) - \sum_{i=1}^{N} p(x_i) = 0 \end{aligned}$$

2.2 先验概率分布

先验分布（prior distribution）一译“验前分布”“事前分布”。是概率分布的一种。与“后验分布”相对。与试验结果无关，或与随机抽样无关，反映在进行统计试验之前根据其他有关参数 的知识而得到的分布。贝叶斯学派认为，在进行观察以获得样本之前，人们对 也会有一些知识。因为是在试验观察之前，故称之为先验知识。因此，贝叶斯派认为，应该把 看作是随机变量。 的分布函数记为 H（ ）， 的密度函数记为 h( )，分别称为先验分布函数和先验密度函数，两者合称为先验分布。

[1]来自 Wikipedia

当参数 的先验分布已知时，称在给定样本 x 下 定条件分布为参数的后验分布 (posterior distribution)。后验分布可看成是在获得样本 x 后对参数先验知识的调整。

根据原因来估计结果的概率分布即似然估计。根据原因来统计各种可能结果的概率即似然函数。

$$后验概率 = \frac{似然概率 * 先验概率}{evidence}$$

2.3 隐变量的参数估计 (MLE、MAP) 问题

最大似然估计（ML，Maximum Likelihood）可以估计模型的参数。其目标是找出一组参数 θ 使得模型产生出观测数据 x 的概率最大：

$$\arg\max_{\theta} P(x|\theta)$$

假如这个参数有一个先验概率，那么参数该怎么估计呢？这就是 MAP 要考虑的问题。最大后验估计（MAP — Maxaposterior）。MAP 优化的是一个后验概率，即给定了观测值后使概率最大：

$$\arg\max_{\theta} P(x|\theta) = \arg\max_{\theta} \frac{P(x|\theta) * P(\theta)}{P(x)}$$

因为给定样本 x 后，p(x) 会在 空间上为一个定值，和 的大小没有关系，所以可以省略分母 p(x)。

$$\arg\max_{\theta} P(x|\theta) = \arg\max_{\theta} P(x|\theta) * P(\theta)$$

即为

$$posterior \propto Likelihood * prior$$

P(x) 相当于是一个归一化项，整个公式就表示为：后验概率正比于先验概率 * 似然函数。

3 自回归模型

考虑 markov 过程

PixelRNN 和 PixelCNN 通过链式法则计算似然函数 $p(x) = \prod_{i=1}^{n} p(x_i|x_1, x_2, ..., x_{i-1})$。然后最大化训练数据的似然概率。然而，生成的过程是序列处理的形式，尽管使用 PixelCNN 方法进行作为 PixeRNN 的改进，计算速度仍然很慢。

PixelRNN 和 PixelCNN 的优点是显式计算了似然概率 $p(x)$

4 EM 算法

EM 算法中文叫做：期望最大算法。主要解决的问题是：具有隐变量的混合模型的参数估计，即其极大似然估计。对于简单的模型 $x \sim p(x)$ 我们可以直接计算 p 的参数 θ

$$\theta_{MLE} = \arg\max_{\theta} \log p(x|\theta)$$

5 白板推导三十节

5.1 生成模型定义

对于给定的样本 X，生成模型关注的是 **样本 X 的分布 P(X)**，生成模型和他要解决的任务无关，既可以解决监督任务，也可以解决非监督任务

为什么生成模型是关注样本在样本空间的概率分布?考虑像素数 28*28=784 的 MNIST 的数据集，我们将其展开，则样本 $\boldsymbol{x} = (x_1, x_2, \cdots, x_7 84)$，其中 $x_i \in \{0, 1\}$，那么可以认为 $\{(x_1 = x_{29} = \cdots = x_{757} = 1), (x_2 = x_{30} = \cdots = x_{758} = 1), \cdots, (x_{28} = x_{56} = \cdots = x_{784} = 1)\}$ 的样本是更像阿拉伯数字“1”的，在“1”的数据集出现的概率比较高。而其他样本如 $(x_1 = x_2 = \cdots = x_{784} = 0)$，就在数据集出现的次数很小，概率也小。

现在若给出了样本 X 的分布 $P(X)$，我们从 $p(x)$ 比较大的地方采样 x，就可以生成模型

对于前者，我们对标签 Y 一起建模求 $P(X|Y)$

5.2 无监督与有监督

$$\text{有监督学习}\begin{cases}\text{数据：\{(样本, 标签)\} 的有序对集合}\\ \text{目标：学习 x->y 的映射}\\ \text{例子：分类，回归，标记}\end{cases}$$

$$\text{无监督学习}\begin{cases}\text{数据：\{(样本)\} 的数据集合}\\ \text{目标：学习样本中的模式与结构}\\ \text{例子：降维，聚类，特征学习，密度估计，生成数据}\end{cases}$$

我们认为：分类，回归，标记，为监督任务；而降维，聚类，特征学习，密度估计，生成数据为非监督任务

$Naive\ Bayes\ \ x \in \mathbb{R}^n\ P(X) = \prod_{i=1}^{n} P(X_i|y)$

5.3 表示 & 推断 & 学习

5.4 模型分类

5.5 重参数技巧

6 变分自编码模型 (Variational AutoEncoder) 和变分推断

6.1 背景介绍

在自编码器中图像 x 经过 encoder 压缩转变成 code 空间的 z，经过 decoder 输出 x'。然而 AE 只是学到了一个 x 到 x' 的映射，并且在码空间中，数据的分布是离散的，码空间没有学采样函数，所以不能在码空间中虽以采样一个 z' 生成得到 x'。我们希望在 code 空间中泛化

相对于 AE，VAE 预测了一个分布，再在分布中随机采样一个 z, 而噪声的分布 (均值和方差) 都是从样本中学习得到的，即从带噪声的 code 空间中采样

如果只像 AE 那样去训练重构图片和原始图片的差距 $\|x - x'\|^2$ 会怎么样？则网络会把方差置 0，模型退化为 AE

6.2 VAE 就是无数的 GMM

GMM 中，假设一共有 m 个高斯分布 $(g_1, g_2, ...g_m)$ 则 x 的分布 P(x)，可以用

$$P(x) = \sum_m p(g_i)q(x|g_i) \qquad \sum_m p(g_i) = 1$$

,$p(g_i)$ 代表了属于哪个高斯分布的可能,$q(x|g_i)$ 中 x $N(\mu_i, \sigma_i)$ 为高斯分布

VAE 可以认为是无数个 GMM 的叠加，是条件概率的积分

$$P(x) = \int_z p(z)q(x|z)$$

神经网络的输入是 z，输出是对均值和方差的预测 $\mu(z)$ 和 $\sigma(z)$**VAE 就是拿无限个高斯分布逼近真实的数据分布，隐变量 z 的每一维度代表了数据的一个属性**，假定 z 为正态分布主要为了数学性质

如果想极大似然 $L = \int_x \log P(x)$ 来学习得到 $P(x) = \int_z p(z)q(x|z)$ 的参数 $\mu(z), \sigma(z)$ 这过于困难，因为 $\mu(z), \sigma(z)$ 是网络的预测值，尽管我们能写出 $P(z)$ 的分布，但是无法显示表达神经网络。高斯分布的 $P(z)$ 中并不是都可以产生样本。我们需要另一个样本 $q(z|x)$ 表现了哪些 z 产生的高斯分布，是对 $P(x)$ 真正有效的，而其他的 z 就无需考虑。这实际上是一个编码器

6.3 重参数技巧

采样时，为了得到隐变量 z，我们不从

7 生成对抗网络 (2014)

讲故事.shwj 虽然有 100 元大钞, 但是他想靠画假钞来发财。刚开始有个新来的警察叔叔分不清假钞,shwj 拿着自己乱画的 100 元造假钞是可以骗过他的, 但是终究骗不过印钞机, 于是 shwj 把真钞和假钞给了警察, 让他做

一些判断并纠正了他哪张是真钞, 提升了判别假钞的能力. 然后 shwj 回家好好学习画 100 元, 企图让自己的画作骗过警察

就这样几轮极限拉扯之后, 警察叔叔的鉴别能力和 shwj 画假钞的能力一起提高了. 最后 shwj 拿着真钞和假钞给警察叔叔看, 但是此时警察叔叔没有了分辨能力, 认为这些钞票是真钞的概率都是 0.5, 于是 shwj 就有了画出真实 100 元大钞的能力,

设真实数据的采样为 $x \sim p_{data}$, 理想的数据满足分布 $x \sim p_g$，先验的随机噪声分布为 $z \sim p_z$, 数据的生成器为 $G(z;\theta_g)$, 这里 G 代表了一个 MLP 构成的可微函数，另一个 MLP 为 $D(X;\theta_d)$ 输出 0 或 1: 若 x 是从 p_g 中采样得到的，则输出 1，反之输出 0

我们先把真实的数据和噪声生成的数据送入 D 中，希望 D 能尽可能判断出数据 x 的真实性，这个过程就是

$$\max_D E_{x\sim P_{data}}[\log D(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

而对于判别器 G，希望它能尽可能骗过 G，让 D(G(z)) 的值尽可能接近于 1

$$\min_G E_{z\sim P_{p_z(z)}}[\log(1 - D(G(z)))]$$

总优化目标函数为

$$\min_G \max_D V(D,G) = E_{x\sim P_{data}}[\log D(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

8 Flow-based Generative Model

invertible neural network 简单的数据分布到复杂的数据分布的映射

生成模型的本质，就是希望用一个我们知道的概率模型来拟合所给的数据样本，也就是说，我们得写出一个带参数 θ 的分布 $q_\theta(x)$。然而，我们的神经网络只是“万能函数拟合器”，却不是“万能分布拟合器”，也就是它原则上能拟合任意函数，但不能随意拟合一个概率分布，因为概率分布有“非负”和“归一化”的要求。这样一来，我们能直接写出来的只有离散型的分布，或者是连续型的高斯分布。

记生成器为 G，网络定义了一个概率分布 p_G 对于 $z \sim \pi(z)$ 输出 $x = G(z)$ 是一个高维的向量，里面的每一个元素都是人脸图像的像素。

我们希望 G 生成的照片足够真实，设真实的人脸照片 x 服从于分布 $p_{data}(x)$，我们从其中采样 m 张图片，即选取 $\{x^1, x^2, ..., x^m\} from\ p_{data}(x)$

求取

$$G^* = \arg\max_g \sum_{i=1}^{m} \log P_G(x^i)$$

9 Diffusion Model

9.1 Denoising Diffusion Probabilistic Model(DDPM)

DDPM 有两个重要贡献。一个是 DM 想预测 x_t 直接预测 x_{t-1} 而 DDPM 认为其不便优化通过数学证明了可以以预测 x_{t-1} 到 x_{t-1} 所加的噪声达到同样的目的。第 t 步的时候 Unet 的输入除了有 x_t, 还有 t 的 embedding(位置编码和傅里叶特征)，这是因为采样过程中 Unet 的参数是每一步都共享的，希望开始的时候（t 较大时）多设工程一些图像的论文，而到了 t 接近 0 时，多生成一些图像的如边角高频信息。这样模型就变成了

$$P(x_{t-1}|x_t) = \|z_t - f_\epsilon(x_t, t_{embedding})\|$$

另一个贡献是 DDPM

前向过程中由于每个时刻 t 只与 $t-1$ 时刻有关，所以可以看做马尔科夫过程，在马尔科夫链的前向采样过程中，也就是扩散过程中可以将数据转换为高斯分布。即扩散过程通过 T 次累积对输入数据 x_i 添加高斯噪声，将这个跟马尔可夫假设相结合，于是可以对扩散过程表达成

$$q(x_{1:T}|x_0) := \prod_{t=1}^{T} q(x_t|x_{t-1}) := \prod_{t=1}^{T} N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \boldsymbol{I})$$

其中 $\beta_1, \beta_2, \cdots, \beta_T$, 是高斯分布方差的超参数。在扩散过程中，随着 t 的增大，x_t 越来越接近纯噪声。当 T 足够大的时候，x_t 可以收敛为标准高斯噪声 $N(0, \boldsymbol{I})$

逆向过程时模型学习逆扩散过程的概率分布，以从纯高斯噪声 $p(x_T) := N(x_T; 0, \boldsymbol{I})$ 开始生成新数据。根据马尔可夫规则表示，逆扩散过程当前时间步 t 只取决于上一个时间步 $t-1$，模型将逐步学习概率分布

$$p_\theta(x_{t-1}|x_t) := N(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

得到联合概率分布 $p_\theta(x_{T:0})$：

$$p_\theta(x_{T:0}) := p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) := p_\theta(x_{T:0}) := p(x_T)\prod_{t=1}^{T} N(x_{t-1};\mu_\theta(x_t,t),\Sigma_\theta(x_t,t))$$

就可以从 $p_\theta(x_0)$ 中采样得到 x_0 这样一来,DDPM 的每一步的推断可以总结为：

1. 每个时间步通过 x_t 和 t 来预测高斯噪声 $z_\theta(x_t,t)$，随后根据 $\mu_\theta(x_t,t) = \frac{1}{\sqrt{a_t}}(x_t - \frac{\beta_t}{\sqrt{1-\overline{a_t}}})z_\theta(x_t,t)$ 得到均值 $\mu_\theta(x_t,t)$.
2. 得到方差 $\Sigma(x_t,t) = \widetilde{\beta}_t = \frac{1-\overline{\alpha_{t-1}}}{1-\overline{\alpha_t}}\beta_t$
3. 利用 $p_\theta(x_{t-1}|x_t) = N(x_{t-1};\mu_\theta(x_t,t),\Sigma_\theta(x_t,t))$ 得到 x_{t-1}

   (https://zhuanlan.zhihu.com/p/572263021)

9.2 Improved DDPM

DDPM 认为噪声的方差是不用学习的，但是 DDPM+ 认为学习到方差后生成的效果更优

第二个改进是把添加噪声的 schedule 从 linear 变成了 cosEmbedding，更 work 了

第三个改进就是发现大模型对图像生成效果更优

9.3 Diffusion Beats GAN

classifier guidance 对采样有很大帮助 ,classifier guided Diffusion 是在训练模型的同时, 再去训练一个分类器，因为 DM 的图片都是加了噪声的，所以这个 classifier guided Diffusion 是在加了噪声的 ImageNet 上训练的

每次得到一个图片 x_t 后将图片输入到判别器中得到交叉熵损失 g,g 包含了图像是否有图像和图像是否真实到能被判别器判别的信息 (这类似于 GAN 的 Discriminator), 然后计算 ∇g, 得到下一步图像 x_{t-1} 生成的方向。牺牲了一部分图像的多样性换来图像的写实性

classifer guided Diffusion[2]

[2]Scale Matters: Money is All your need

9.4 Denoising Diffusion Implicit Models(DDIM)

9.5 Score-based Model

思想来自于 19 年的论文《Generative Modeling by Estimating Gradients of the Data Distribution》,Gradients of the Data Distribution 就是 model 的 score。问题是如何求 score，怎么用 score 作加强。作者提出的 *Langevin dynamics* 实际上和 DDPM 的采样思路类似

传统的生成模型都是学习数据分布的概率 $p_\theta(x)$, 其中 x 是样本。而 score 是对数概率的梯度，即 $\nabla_x \log p_\theta(x)$, 在样本空间中，梯度是指向样本密度最高的位置。如果我们获得了样本空间的每一点的 score，那么对于生成的每一步，我们只需要沿着当位置的 score 方向更新数据，那么就会逼近真实的样本。

具体地，如果我们我们通过了某种方法 (score matching) 得到了 score 的模型 $s_\theta(x) = \nabla_x \log p_\theta(x)$, 那么我们就可以用 Langevin dynamics 的迭代过程从**一个任意分布走到目标分布**

$$x_{i+1} \leftarrow x_i + \epsilon \nabla_x \log p_\theta(x) + \sqrt{2\epsilon} z_i, \qquad i = 0, 1, \cdots, K$$

在原始的数据上加噪声是为了更好地估计 score。训练地时候只是训练了一个去噪模型, 为什么可以做到图像生成? 这是由于:

$$\text{估计噪声} \Leftrightarrow \text{估计 score} \Leftrightarrow \text{估计数据分布梯度}$$

而获得了梯度后, 我们就可以像 gradient descent 一样去逐步逼近真实的数据分布, 达到图像生成的效果

9.6 背景介绍

啊啊啊啊