模式识别第二次作业

162050127 颜劭铭

2022 年 11 月 25 日

1 模型欠拟合、过拟合现象分别指什么？试配合图例说明。有哪些方式可以缓解欠拟合、过拟合问题？

欠拟合是指不能很好的从训练数据中，学习到数据背后的规律，从而针对训练数据和待预测的数据，均不能获得很好的预测效果。如果使用的训练样本过少，较容易获得欠拟合的训练模型。如图 1 所示，认为只要是绿色的就是树叶，因此判断树也是树叶。

过拟合是指过于精确地匹配了特定数据集，导致获得的模型不能良好地拟合其他数据或预测未来的观察结果的现象。也就是说模型不仅学习到了一般性的特征，还学习到了某些样本特别具有的特征并认为是所有样本都有的特征。如图 1 所示，认为树叶边缘必须有锯齿，因此判断该圆叶片不是树叶。

欠拟合的解决方式有增加网络复杂度或者在模型中增加特征等。

过拟合的解决方式有数据集增强、更换模型降低模型复杂度、正则化、drop out、早停等。

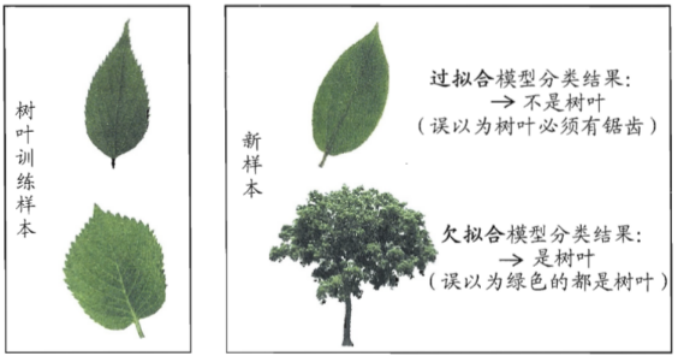


图 1: 欠拟合和过拟合

2 以 C 类分类问题为例，$\lambda(\hat{y}, y)$ 表示将真实类别 y 预测为类别 $\hat{y}$ 的损失函数，p(x) 表示样本分布，推导最小风险贝叶斯决策中的期望风险表达式。

已知样本分布 p(x)，由贝叶斯公式可以得到后验概率如下：

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

假设状态空间为 $\Omega = \{\omega_1, \omega_2, \ldots, \omega_c\}$ ，决策空间为 $A = \{\alpha_1, \alpha_2, \ldots, \alpha_c\}$

又已知 $\lambda(\hat{y}, y)$ 表示将真实类别 y 预测为类别 $\hat{y}$ 的损失函数，则对于某个样本 x，对它采取决策 α_i 的期望损失为：

$$\mathsf{R}(\alpha_i|x) = E[\lambda(\alpha_i,\omega_j)|x] = \sum_{j=1}^{c} \lambda\left(a_i, w_j\right) p\left(w_j \mid x\right) \quad i = 1,\ldots,c$$

因此假设有一个决策规则 $\alpha(x)$，则该决策规则对于所有样本 x 造成的期望损失如下所示：

$$R(a) = \int R(a(x) \mid x) p(x) dx$$

而最小化风险贝叶斯就是最小化该期望风险，如下所示：

$$\min_a R(a)$$

3 贝叶斯决策理论中，参数估计和非参数估计分别指什么？作为参数估计的两种实现方式，点估计和 Bayesian 估计的区别是什么？

参数估计：已知样本类别和函数模型（假设一个模型），根据样本估计模型中的未知参数。

非参数估计：已知样本类别，未知函数模型（不假设模型），直接利用样本信息学习估计模型。

点估计：构造一个统计量作为参数 θ 的估计 $\hat{\theta}$，点估计中常用的有最大似然估计，即把参数 θ 看成为确定的未知参数，然后求似然函数 $P(X^i|\theta)$，并将最大的 $\hat{theta}$ 作为最大的似然统计量。

Bayesian 估计：把参数 θ 看成为随机的未知参数，样本通过似然函数和贝叶斯公式得出后验分布，再求出估计量。

4 生成式模型和判别式模型的区别是什么？生成式模型相比判别式模型有什么好处？逻辑回归分类器、SVM、FLDA (Fisher 线性判别分析)、隐马尔科夫模型分别属于哪种类型？

生成式模型：主要学习的是联合概率密度分布 P(X,Y)，更关心数据的分布，反映出同类数据本身的相似度，而不关心划分不同类的边界。

判别式模型：直接学习决策函数 Y=f(X) 或条件概率 $P(Y|X)$，不关心训练数据本身的特点，即不关心数据的分布，而是寻找不同类别之前的边界，反映出不同类数据之间的差异。

生成式模型：隐马尔科夫模型。

判别式模型：逻辑回归分类器，SVM，FLDA。

生成式模型可以进行异常值检测，当样本数量较多时，收敛速度较快，能够应付存在隐变量的情况，并且过拟合的几率较小。

5 以二分类问题为例，推导 SVM 模型的损失函数、优化实现

首先在我们的样本空间中，我们需要寻找用于划分的鲁棒性最强的超平面可以定义为：

$$w^T x + b = 0$$

而由二分类问题我们可以得到：

$$\begin{cases} w^\top x_i + b > 0, y_i = 1 \\ w^\top x_i + b < 0, y_i = -1 \end{cases}$$

由我们希望离超平面最近的点到超平面的距离最远，利用点到平面的距离公式，可以得到：

$$\begin{cases} \min_{w,b} \frac{\|w\|^2}{2} \\ \text{s.t. } \mathrm{y}^{(i)} \cdot \left(w^T X^{(i)} + b\right) \geq 1, i = 1, 2, 3, \ldots, m \end{cases}$$

考虑使用拉格朗日乘子法，并引入 KKT 乘子 α_i:

$$L((w, b, \alpha)) = \frac{1}{2} + \sum_{i=1}^{m} \alpha_i \left(1 - y_i \left(w^T x_i + b\right)\right)$$

求偏导可得：

$$\begin{cases} \frac{\partial L}{\partial w} = 0 \\ \frac{\partial L}{\partial b} = 0 \end{cases}$$

计算后可得：

$$w = \sum_{i=1}^{m} \alpha_i y_i x_i$$
$$0 = \sum_{i=1}^{m} \alpha_i y_i$$

利用拉格朗日乘子法公式的对偶形式，并将上式计算结果代入可得：

$$\begin{aligned} \min \mathrm{L}_{W,b}(\ \mathrm{W},\ \mathrm{b}, \alpha) =& \frac{1}{2} \left(\sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} X^{(i)} \right)^2 + \sum_{i=1}^{m} \left(\alpha_i - \alpha_i \mathrm{y}^{(i)} \cdot \left(\sum_{j=1}^{m} \alpha_j \mathrm{y}^{(j)} {X^{(j)}}^T X^{(i)} + b \right) \right) \\ =& \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j \mathrm{y}^{(i)} \mathrm{y}^{(j)} {X^{(i)}}^T X^{(j)} + \sum_{i=1}^{m} \alpha_i - \sum_{i=1}^{m} \left(\alpha_i \mathrm{y}^{(i)} \cdot \left(\sum_{j=1}^{m} \alpha_j \mathrm{y}^{(j)} {X^{(j)}}^T X^{(i)} \right) \right) \\ & - \sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} b + \\ =& -\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j \mathrm{y}^{(i)} \mathrm{y}^{(j)} {X^{(i)}}^T X^{(j)} + \sum_{i=1}^{m} \alpha_i \end{aligned}$$

注意:

$$W^T = \left(\sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} X^{(i)} \right)^T = \sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} {X^{(i)}}^T$$
$$\frac{\|W\|^2}{2} = \frac{1}{2} W^T W = \frac{1}{2} \left(\sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} X^{(i)} \right) \left(\sum_{j=1}^{m} \alpha_j \mathrm{y}^{(j)} X^{(j)} \right)$$

因此最后的优化目标函数为:

$$\max_{\alpha, \alpha_i \geq 0} \left[\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j \mathrm{y}^{(i)} \mathrm{y}^{(j)} {X^{(i)}}^T X^{(j)} \right]$$

约束条件为:

$$\begin{cases} \alpha_i \geq = 0, i = 1, 2, 3, \ldots, m \\ \sum_{i=1}^{m} \alpha_i \mathrm{y}^{(i)} = 0 \end{cases}$$

6 变分自编码器优化中重参数技巧的作用是什么?

在变分自编码器中，变量具有随机性，而通过重参数化技巧，可以将变量的随机性转移到高斯分布的元变量中，此时模型需要学习的就是如何对于高斯分布的均值和方差进行调整。

具体来说，变分自编码器将对

$$\forall(y \mid x) \sim N\left(\mu(x), \sigma^2(x)\right)$$

的采样转换成从 $z \sim N(0, I)$ 采样，此时采样的随机性就转移到从标准高斯分布中采样，并且此时原本无法求导，无法梯度传播的中间节点可以求导。

7 推导出判别器的优化目标为真实分布与生成分布的 JS-divergence.

首先，假设存在两个分布 P,Q，并且 $M = \frac{1}{2}(P + Q)$ ，则

$$JSD(P\|Q) = \frac{1}{2}KL(P\|M) + \frac{1}{2}KL(Q||M)$$

把 KL 散度公式代入展开可得：

$$JSD(P \mid Q) = \frac{1}{2}\sum p(x)\log\left(\frac{p(x)}{\frac{p(x)+q(x)}{2}}\right) + \frac{1}{2}\sum q(x)\log\left(\frac{q(x)}{\frac{p(x)+q(x)}{2}}\right)$$

将 log 中的 $\frac{1}{2}$ 放到分母上，并将 2 提出，可得：

$$JSD(P\|Q) = \frac{1}{2}\sum p(x)\log\left(\frac{p(x)}{p(x)+q(x)}\right) + \frac{1}{2}\sum q(x)\log\left(\frac{q(x)}{p(x)+q(x)}\right) + \log 2$$

因此，利用 JS 散度和 KL 散度间的关系，并将 D^* 带入到 $\max_D V(G,D) = V\left(G, D^*\right)$ 可得：

$$\begin{aligned}
\max_D V(G,D) =& V\left(G, D^*\right) \\
=& E_{x\sim P_{\text{data}}}\left[\log\frac{P_{\text{data}}\left(x\right)}{P_{\text{data}}\left(x\right)+P_G(x)}\right] + E_{x\sim P_G}\left[\log\left(1-\frac{P_{\text{data}}\left(x\right)}{P_{\text{data}}\left(x\right)+P_G(x)}\right)\right] \\
=& \int_x P_{\text{data}}\left(x\right)\log\frac{P_{\text{data}}\left(x\right)}{P_{\text{data}}\left(x\right)+P_G(x)}dx + \int_x P_G(x)\log\left(1-\frac{P_{\text{data}}\left(x\right)}{P_{\text{data}}\left(x\right)+P_G(x)}\right)dx \\
=& \int_x P_{\text{data}}\left(x\right)\log\frac{P_{\text{data}}\left(x\right)}{\left(P_{\text{data}}\left(x\right)+P_G(x)\right)/2}dx + \int_x P_G(x)\log\left(1-\frac{P_{\text{data}}\left(x\right)}{\left(P_{\text{data}}\left(x\right)+P_G(x)\right)/2}\right)dx - 2\log 2 \\
=& -2\log 2 + KL\left(P_{\text{data}}\left(x\right)\|\frac{P_{\text{data}}\left(x\right)+P_G(x)}{2}\right) + KL\left(P_G(x)\|\frac{P_{\text{data}}\left(x\right)+P_G(x)}{2}\right) \\
=& -2\log 2 + 2JSD\left(P_{\text{data}}\left(x\right)\|P_G(x)\right)
\end{aligned}$$

8 分析 JS-divergence 为什么会导致生成对抗网络训练不稳定?

JK-divergence 的定义如下：

$$D_{JS}(p\|q) = \frac{1}{2}D_{KL}\left(p\|\frac{p+q}{2}\right) + \frac{1}{2}D_{KL}\left(q\|\frac{p+q}{2}\right)$$

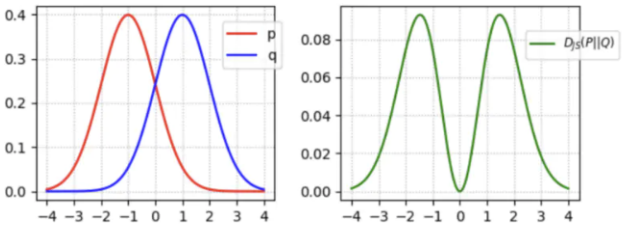


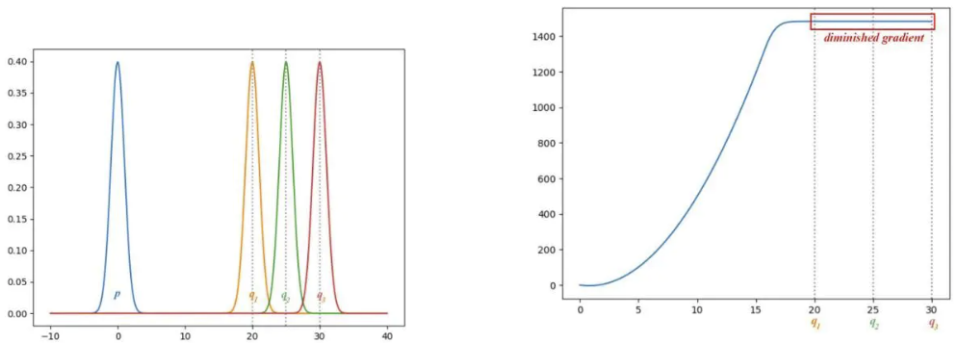
图 2: JS-divergence

如图 2 所示，可以发现 JS-divergence 是对称的，并且，在 GAN 中，若判别器是最优的，生成模型的目标函数将会变成;

$$\min_G V(D^*, G) = 2D_{JS}(p_r \| p_g) - 2\log 2$$

但是这是判别器最优的情况，网络训练的过程是一个不断优化的过程，可想而知，在更多的时候，判别器并没有达到最优的情况，那么，当生成的图像的数据分布 p 与真实图像的数据分布 q 不匹配的时候，JS-divergence 将会出现梯度消失的问题，而这会导致生成对抗网络训练不稳定。

由于 JS-divergence 是对称的，也就是一种非单调的，可以假设 p 和 q 服从高斯分布，且 p 的均值为 0，考虑 q 的均值在 0 到 30 之间时，如图 3.(a) 所示。



(a) 分布图　　(b) 梯度图

图 3: JS-divergence 服从高斯分布

如图 3.(b) 所示，可以看出，JS-divergence 在 q_1 到 q_3 的过程中梯度消失了，因此在这个过程中，GAN 的生成器学习会非常缓慢。

这个问题的出现就是因为当 p 和 q 两个分布完全不重叠的时候，也就是不连续的时候，即使两个分布的中心距离很近，但是 JS-divergence 都是一个常数，以至于梯度为 0，导致梯度消失，无法更新。以下式子给予证明：

当 p(x)=0 时:

$$\begin{aligned} JS(P\|Q) &= \frac{1}{2}\sum p(x)\log\left(\frac{p(x)}{p(x)+q(x)}\right) + \frac{1}{2}\sum q(x)\log\left(\frac{q(x)}{p(x)+q(x)}\right) + \log 2 \\ &= \frac{1}{2}\sum 0\times\log\left(\frac{0}{0+q(x)}\right) + \frac{1}{2}\sum q(x)\log\left(\frac{q(x)}{0+q(x)}\right) + \log 2 \\ &= \log 2 \end{aligned}$$

当 q(x)=0 时:

$$
\begin{aligned}
JS(P\|Q) &= \frac{1}{2}\sum p(x)\log\left(\frac{p(x)}{p(x)+q(x)}\right) + \frac{1}{2}\sum q(x)\log\left(\frac{q(x)}{p(x)+q(x)}\right) + \log 2 \\
&= \frac{1}{2}\sum p(x)\log\left(\frac{p(x)}{p(x)+0}\right) + \frac{1}{2}\sum 0\times\log\left(\frac{0}{p(x)+0}\right) + \log 2 \\
&= \log 2
\end{aligned}
$$

尤其是在早期训练过程中，p 和 q 的分布是非常不同的，因此这个时候很容易出现梯度消失的问题，所以生成对抗网络训练不稳定。

9 有哪些方法可以缓解这个问题?

1. 更换 JS-divergence 为 Wasserstein 距离，即：

$$
\mathbf{W}_p\left(q_1, q_2\right) = \left(\inf_{\gamma(x,y)\in\Gamma(q_1,q_2)} \mathrm{E}_{(x,y)\sim\gamma(x,y)}\left[d(x,y)^p\right]\right)^{\frac{1}{p}}
$$

   该距离可以在生成的图像的数据分布 p 与真实图像的数据分布 q 不匹配时，仍然能反应两个分布的远近。

2. 损失函数添加正则化项。

3. 在损失函数中添加梯度惩罚机制。

4. 对判别器 D 和生成器 G 使用不同的学习速度。低速更新规则用于生成器 G，判别器 D 使用高速更新规则，可以让生成器 G 和判别器 D 以 1:1 的速度更新，以使得生成的图像的数据分布 p 与真实图像的数据分布 q 更加匹配。