**第2章 模型评价与损失函数**

训练是指根据“训练集”寻找最优模型参数的过程。训练集是指从现实样本分布中采样的包含类别信息的样本集合。

本章首先介绍模型训练的一般概念和模型评价的若干指标，之后探讨分类问题的损失函数。经由损失函数，模型训练问题归约成了以模型参数为自变量，在自变量空间中寻找损失函数最小值的函数优化问题。本章主要介绍交叉熵损失函数，并从K-L散度和最大似然估计两种角度阐释交叉熵损失函数的原理。

阅读本章后，读者应当掌握了机器学习模型训练的一般概念和评价模型的方法，对交叉熵损失函数有了较深刻的理解。本章虽是在逻辑回归框架下进行讲解，但所有概念都可以直接用于神经网络和深度学习。

**2.1 训练集与测试集**

第1章已经介绍，给定权值向量和偏置值，对于样本，逻辑回归模型预测其为A类的概率是：

（2.1）

式（2.1）中的和就是逻辑回归模型的参数（parameters）。所谓“训练”（training）就是寻找参数和的值，使得模型可以很好地区分A类和B类样本。训练过程需要“训练集”（training set）。训练集由一批带类别信息的样本组成。这些样本是从现实中采样的属于A类或B类的样本。

训练样本的类别信息用一个实数标识。例如，用标识样本属于A类；用标识样本属于B类。值称为标签（label）。标签的1/0编码只是方法的一种，还可以采用其他编码。后文会看到不同编码的用途。

训练集是如式（2.2）描述的集合。

（2.2）

上标表示样本的编号。训练集中一共包含个样本。其中每一个是样本特征向量，是样本标签。

为了评价模型的表现，有必要取另一份带标签的样本集，称为测试集（test set）。只有在测试集上对模型进行评价，才能得到客观无偏的评价指标。第3章“正则化”会介绍模型自由度、过拟合、偏置-方差平衡等概念。届时会阐述必须在独立的测试集上评价模型的原因。

**2.2 分类模型的评价**

对于训练完成的逻辑回归模型，可以在测试集上评价它的表现。第1章曾提到：对于一个样本，逻辑回归给出的是它属于A类的概率。人有主动权选定一个阈值*t*，当时将判定为属于A类，否则判定属于B类：

（2.3）

^符号表示是模型预测的标签，与训练样本中的标签区分。对测试集中的所有样本计算。一旦选定了阈值*t*，根据式（2.3）就可以得出模型对每一个样本所判定的类别。和一样，用1/0编码A/B类别。

**2.2.1 混淆矩阵**

有了模型分类结果就可以绘制混淆矩阵（confusion matrix）：

|  |  |  |
| --- | --- | --- |
|  | 预测 B 类 | 预测 A 类 |
| 真实 B 类 | TN | FP |
| 真实 A 类 | FN | TP |

表2.1 二分类问题的混淆矩阵

二分类问题的混淆矩阵是一个矩阵。从左上到右下每一个元素分别是：

* TN（True Negative）：真实为B类，且模型判定为B类的样本个数；
* FP（False Positive）：真实为B类，但模型判定为A类的样本个数（被错误地判定为A类）；
* FN（False Negative）：真实为A类，但模型判定为B类的样本个数（被错误地判定为B类）；
* TP（True Positive）：真实为A类，且模型判定为A类的样本个数。

评价模型的指标都可以由混淆矩阵计算得出。下文介绍最常用的几个。

**2.2.2 正确率**

正确率（accuracy）的计算公式为：

（2.4）

正确率是混淆矩阵的对角线元素之和除以全体元素之和。它是模型正确分类的样本个数与全部样本个数之比。有时正确率并非一个良好的评价指标。假如测试集中A类样本和B类样本的数量比为99:1，那么模型将所有样本判定为A类就能够得到99%的正确率，但是该模型显然不是一个好模型。

**2.2.3 查准率**

查准率又称准确率（precision），其计算公式为：

（2.5）

A类查准率是混淆矩阵右下角元素除以第二列元素之和。它是模型正确判定为A类的样本数量与全部判定为A类的样本数量之比。评价模型判定为A类的准确程度。越高则模型的断言越可靠。同样也有B类查准率。

**2.2.4 查全率**

查全率又称召回率（recall），其计算公式为：

（2.6）

A类查全率是混淆矩阵右下角元素除以第二行元素之和。它是模型判断为A类的样本数量与全部A类样本数量之比。评价模型对A类的召回情况。越高则模型能把更多的A类样本识别出来。又称真阳率（，True Positive Rate）。与之对应还有假阳率（，False Positive Rate）：

（2.7）

是所有B类样本中被模型错判成A类的比例。它越高则模型表现越差。查全率，真阳率和假阳率也都可以对B类计算。

上述指标都基于分类结果，而分类结果依赖于可人为调节的概率阈值*t*。假如*t*设得较低，可以想象低门槛将导致更多的样本被判定为A类，会升高。但同时也会把更多B类样本错判为A类，从而抬高，降低。反之，若*t*设得较高，会降低，但将升高。所以选择阈值是对模型两种相反的倾向做权衡。依据是具体问题的需要。

**2.2.5 ROC曲线**

这对指标随着*t*值变化同升同降。对于识别A类的问题来说，高是我们愿意看到的，而高是希望避免的。希望在提高的同时不要大幅度地提高。随着*t*的变化行为可由ROC（receiver operating characteristic）曲线刻画。如图2-1所示。

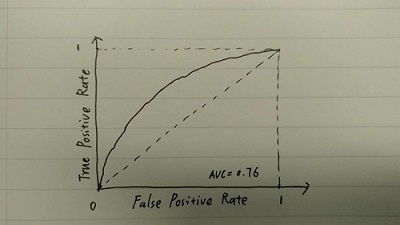


图2-1 ROC 曲线

ROC 曲线以为横轴，以为纵轴，将不同*t*值对应的值对以散点的形式绘出。得到的图形是一条拱起的曲线。ROC曲线上拱得高说明在较低的水平能够得到较高的。ROC曲线下的面积（Area Under Curve，AUC）可以衡量模型的质量。高AUC意味着ROC曲线上拱，模型的表现更优。AUC不依赖阈值*t*，是一个全面衡量模型质量的指标。

我们希望模型在测试集上有较优的表现，但我们无法用测试集上的指标来指导模型参数的选择。因为评价指标不是模型参数的连续函数。参数在空间中的极小位移会导致模型输出概率的极小变化。当这个变化不足以使概率跨越阈值时，模型对样本的分类不改变，各种评价指标也就不变。而一旦某个微小位移导致了概率跨越了阈值，各种评价指标将发生跳跃式变化。模型参数和评价指标之间缺乏一个显式的连续的映射，使我们无法利用评价指标来调优模型参数。

**2.3 损失函数**

我们需要采用一种“代理”评价指标。它应该是一个关于模型参数的显式连续函数。这种“代理”评价指标称为损失函数（loss function）。损失函数以某种方式衡量模型的质量。于是模型训练就变成了在参数空间中寻找损失函数最小值的问题。

损失函数有很多种，本书只介绍分类问题中最常用的交叉熵（cross entropy）损失函数。我们将从信息论和贝叶斯两种视角阐释交叉熵损失函数的含义。

**2.3.1 K-L散度与交叉熵**

随机变量有种不同的取值：。令取的概率为，简写作。将看作一个信号源。观察到就相当于收到了一条信息。克劳德·香农为一条信息的信息量做了定量定义：

（2.8）

式（2.8）中的对数可以以2为底，也可以取其他底，比如自然对数的底。不同的底得到的信息量相差一个常系数。如果以2为底，信息量的单位是比特（bit）。 称为这条信息的自信息量（self-information）。随着变化的图像如图2-2所示。趋向于1时，趋向于0；趋向于0时，趋向于正无穷。

自信息量定义背后的洞见是：信息所告知的事件的概率越小，则信息的信息量越大。假如有人告诉你：“即将开奖的彩票中奖号码是31415926”。这条信息非常有用，你愿意花大价钱购买它。假如有人告诉你：“明天太阳照常升起”。这条信息几乎是无用的。你不用别人告诉也知道明天太阳几乎肯定照常升起。前一条信息所告知的事件的概率极小，所以信息量很大；后一条信息所告知的事件的概率极大，所以信息量很小。

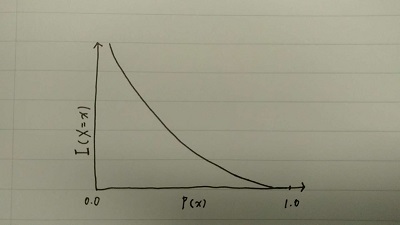


图2-2 自信息量的图像

令信息源取不同值的概率为。定义信息源的熵（entropy）为：

（2.9）

信息源由分布描述，故将熵视为的函数。熵的概念来自热力学。又被称作平均自信息。因为是对的所有取值以概率为权重取平均。换句话说是在分布上的期望（exception）。式（2.9）是针对离散型随机变量。对连续型随机变量应以积分取代求和。

有两个分布和，定义与的K-L散度（Kullback-Leibler Divergence）是：

（2.10）

K-L散度是在分布上的期望。注意。如果对于所有有，即，则。两个相同分布和的K-L散度为0。K-L散度用来衡量两个分布之间的差异程度。

注意式（2.10）第二个等号后。将第一项定义为分布和的交叉熵（cross entropy）：

（2.11）

是在分布上的期望。根据式（2.10）和（2.11），有：

（2.12）

分布和的交叉熵等于和的K-L散度加上的熵。如果分布不变，则与之间相差一个常数。于是也可用来衡量分布和的差异程度：越小则和越相似。

对于一个训练样本，可以认为标签给出了一个类别的伯努利分布：

（2.13）

当属于A类时，该分布就是，；当属于B类时，该分布就是，。这是一个“确定”的分布。

逻辑回归模型的输出也是一个伯努利分布：

（2.14）

我们希望模型给出的分布与训练标签给出的分布越相似越好。于是可以将训练标签给出的分布和模型给出的分布的交叉熵作为在样本上的损失：

（2.15）

较大表示模型分布与训练标签分布之间的差异较大。反之亦然。

式（2.15）是在一个训练样本上的损失。在整个训练集上的损失就是在所有样本上的损失的平均：

（2.16）

式（2.16）就是交叉熵损失函数。它与训练集有关，以模型参数和为自变量。逻辑回归模型的训练就是寻找使尽可能小的和。这就将模型训练问题转化为一个函数优化问题。

交叉熵作为“代理”评价指标，与2.2节介绍的各种评价指标没有直接的、显式的关系。但通过最小化交叉熵，我们拉近了模型的预测类别分布与训练样本的真实类别分布之间的“距离”。经过训练，可以期待模型抓住数据背后的分布规律，从而在测试集上获得较好的效果。

**2.3.2 最大似然估计**

本节从最大似然估计的视角阐释交叉熵损失的含义。和是两个离散型随机变量。贝叶斯公式（Bayes Rule）是：

（2.17）

式（2.17）左边的称为后验概率（posterior probability）。它是观察到事件的前提下，事件发生的概率。右边分子上的称为先验概率（prior probability）。它是事件发生的概率。分子上的称为似然概率（likelihood）。它是事件发生的前提下，事件发生的概率。分母是事件发生的边缘概率（Marginal Probability）：

（2.18）

由于是对的所有可能取值求和，所以与的取值无关。式（2.17）的证明很简单：将右边的分母乘到左边，根据条件概率的定义，等号两边都是——事件和同时发生的概率。

回到逻辑回归的语境下，事件是模型参数是特定值和，事件是观察到训练集。代入贝叶斯公式：

（2.19）

观察到训练集前提下参数值是和的后验概率，等于参数值是和的先验概率乘以参数值是和前提下观察到训练集的似然概率，再除以观察到训练集的概率。

训练的目标是寻找观察到训练集前提下最有可能的参数值，也就是使后验概率最大的参数值：

（2.20）

假设先验分布是均匀的，与参数取值无关，那么问题转化为寻找似然概率最大的参数值：

（2.21）

称为最大似然估计（Maximum Likelihood Estimate，MLE）。

对于一个训练样本，用1或0标识样本属于A类或B类。逻辑回归模型预测属于所标识的类别的概率是：

（2.22）

式（2.22）的技巧是利用任何数的0次方都等于1的事实，根据是1还是0选择或。假设训练样本是独立的，可以得到：

（2.23）

因为是单调递增的。式（2.21）等价于寻找：

（2.24）

根据公式（2.22）、（2.23）和（2.24），最大似然估计是寻找和使式（2.25）最大化：

（2.25）

最大化等价于最小化它的相反数。于是最大似然估计就是寻找和使式（2.26）最小化：

（2.26）

除了一个常系数，式（2.26）和式（2.16）相同。所以最小化交叉熵（2.16）的和正是最大似然估计。

最大似然估计使似然概率最大化。但其实我们想要的是最大化后验概率。在假设模型参数的先验分布是均匀的前提下，此二者等价。在第3章“正则化”中我们将看到，为损失函数加上“正则化项”相当于取一个参数先验分布，然后最大化后验概率。正则化的强度与先验分布的方差有关。

**2.3.3 从几何角度理解逻辑回归的交叉熵损失**

第1章介绍过，逻辑回归只能形成超平面分界面（以下把2维直线、3维平面以及更高维的超平面统称超平面）。如果以为阈值，则分界超平面的方程是：

（2.27）

经过简单的计算，可知式（2.27）等价于：

（2.28）

满足式（2.28）的所有点构成一个以为法向量的超平面。上的点与的内积是常数，即上的向量向方向的投影长度都相同：

(2.29)

若空间某一点向方向的投影长度大于，则它位于的一侧，称之为正侧；反之若某一点向方向的投影长度小于，则它位于的另一侧，称之为负侧。的投影长度与之差的绝对值是与超平面之间的距离。如图2-3所示。

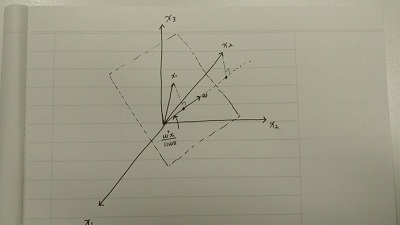


图2-3 超平面将空间分为两侧以及点到超平面的距离

现在用另一种编码标识训练样本的类别：

（2.30）

假设一个逻辑回归模型能够完美地将A/B两类训练样本分开。所有的样本有。于是，即位于的正侧。此时；所有的样本有。于是，即位于的负侧。此时仍有。

当，即和符号相反时，模型分类错误——A类样本位于的负侧，；B类样本位于的正侧，。损失函数应该对分错的情况施以惩罚。

交叉熵损失（2.16）中的用1/0标识A/B类别。对于所有，与的关系是：

（2.31）

将式（2.31）代入交叉熵损失（2.16），得到：

（2.32）

每一个训练样本对交叉熵损失的贡献是：

（2.33）

A类样本的，公式（2.33）成为：

（2.34）

B类样本的，公式（2.33）成为：

（2.35）

结合公式（2.34）和（2.35），得到：

（2.36）

将视作的函数，其图像如图2-4所示。

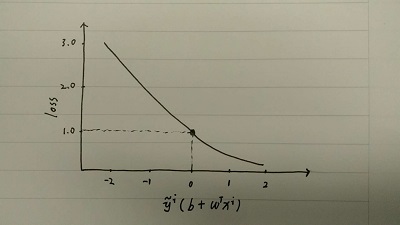


图2-4 交叉熵损失作为的函数的图像

以为阈值。如果对某一个训练样本有，则分类正确。越大距离分界面越远。如果则分类错误。越小，即越大则在错误的一侧距离分界面越远。后一种情况下损失函数的值应该更大。如图2-4所示，交叉熵损失函数恰当地惩罚了分类错误。