机器学习中的数学（2）：信息熵与损失函数，致敬Shannon神

AI有点可ai AI有点可ai 昨天

> “经常用均方误差和交叉熵都作为损失函数？想要深入了解一下这两者的联系与区别？拒绝简单调包，这里是机器学习中的数学（2），带你走进划时代的信息论❤️。”

❤️ 00 内容概览

机器学习中的数学（2）：信息熵与损失函数，致敬Shannon神

在众多的机器学习和深度算法中，我们见到许多度量模型效果的损失函数，在回归任务中常见的是均方误差函数，在分类任务中，交叉信息熵则使用很频繁，为什么呢？本次文章将带你领略香农信息论的魔力。

本期导读：

- 香农与信息论
- 信息熵
- 相对熵与交叉熵
- 均方误差与交叉熵对比
- 多目标分类
- 最小化交叉熵与最大化似然函数

申明

本文原理解释及公式推导部分均由LSayhi完成，允许部分或全部转载，但请注明出处；详细数据及代码可在github查阅。

GitHub：https://github.com/LSayhi/book-paper-note

CSDN博客：https://blog.csdn.net/LSayhi

微信公众号：AI有点可ai（文末附二维码，感谢您的关注）

❤️ 1 香农与信息论

信息论是研究信息及其传输的一般规律的学科，运用数学和其他相关方法研究信息的性质、计量以及获得、传输、存储、处理和交换等。香农被称为是“信息论之父”，通常将香农于1948年10月发表于《贝尔系统技术学报》上的论文《A Mathematical Theory of Communication》作为现代信息论研究的开端,在该文中，香农给出了信息熵的定义,从此信息量的度量有了更精确的数学描述，而不再是以”多”或”少”来衡量，信息论中的很多概念都有跨学科的应用，不只在通信领域，在编码学、密码学、数据压缩、检测与估计理论中就广泛地运用了信息论的相关概念，机器学习和深度学习也涉及到许多信息论的知识，下图是香农半神。

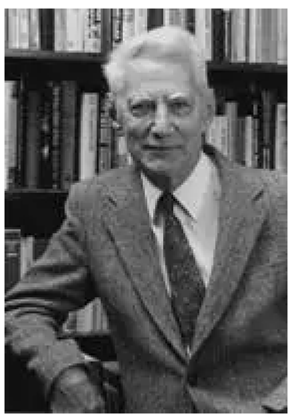


Figure1:Shannon

2 信息熵

物质、能量、信息是世界的三大要素。在信息论诞生之前，物质和能量众多物理学家和数学家已经给出了基本的定义和度量，但是关于信息的度量还没有一个明确的方式，香农在1948年提出了度量信息的法则，并称之为信息熵。
由于信息论首先是应用在通信领域的，我们沿用通信系统的假设，在定义信息熵之前，先给出“自信息”的度量，对于一个分布为p(x)随机变量X，自信息表示为：

- 为了理解这个公式的含义，举个例子，宅男A说他昨晚约了校花出去玩,我们的第一反应是很吃惊，随口一句，“卧槽，怎么可能，信息量太大了”，而当男神B说他十一准备约女朋友的云南旅游，我们的反应除了“卧槽，禽兽，好好玩【滑稽】”就没了。为什么呢?因为按照常理，宅男A约到校花的概率很小，基本上不可能发生的，而男神B约他女票已经大家习以为常的事情，这种事情发生的概率本来就很大，恭喜你已经有了直观的对信息量多少的概念了，这时我们再看自信息的公式，其很完美的契合了我们对信息量的直观感受，就是概率越小的事情，信息量越大，打雷快要下雨蕴含的信息量不多，但是学渣考了满分信息量很大。
- 说完了自信息的概念，我们来引入信息熵，对于分布为p(x)的随机变量X，其信息熵定义为：

  对于离散的情况，

  信息熵总是大于0的，从定义公式来看，信息熵可以理解为自信息的数学期望。那些接近确定性的分布，信息熵比较低，而越是接近均匀分布的，信息熵比较高，这可以对信息熵求最值推导出。这个和越不容易发生的事情信息越大这个基本思想是一致的。从这个角度看，信息可以看做是不确定性的衡量，而信息熵就是对这种不确定性的数学描述（换句话说，就是消除系统不确定性所需的信息量，而不是系统的信息量）。

3 相对熵与交叉熵

信息熵使用来衡量一个分布p其自身的不确定性，相对熵则用来衡量两个分布p和q之间的差异，在信息论中也称为KL散度，其定义为：

$$D_{KL}(p||q)=\int p(x)log\frac{p(x)}{q(x)}=-\int p(x)logq(x)dx-(-\int p(x)logp(x)dx)$$

- 相对熵公式的前半部分就是交叉熵。相对熵只有在p(x)=q(x)时,值才为0。若p和q不同分布，其值就会大于0，证明如下：

$$D_{KL}(p||q)=\int p(x)log\frac{p(x)}{q(x)}=E(log\frac{p(x)}{q(x)})\geq logE\frac{p(x)}{q(x)}=-log\int p(x)\frac{q(x)}{p(x)}dx=-log\int q(x)dx=0$$

- 上式中不等号利用的是Jensen不等式，当p=q时，等号成立。在机器学习中，比如分类问题，如果把结果当作是概率分布来看，标签表示的就是数据真实的概率分布，由logistic函数和softmax函数产生的结果其实是对于数据的预测分布，预测分布和真实分布差值叫做KL散度或者是相对熵。若p(x)是数据的真实概率分布，q(x)是由数据计算假设的概率分布，我们目的就是让q（x）尽可能地逼近甚至等于p(x)，从而使得相对熵接近最小值0。在统计学中，概率学派认为真实的概率分布是固定的（例如抛硬币正反面都是概率是0.5），相对熵公式的后半部分就成了一个常数，最小化相对熵就等效于最小化交叉熵。值得一提的是，对交叉熵求最小值，也“等效”于最大化似然函数（见第五部分）。

4 均方误差与交叉熵

对于损失函数，最直观的是采用均方误差函数，所以先讨论均方误差函数作为损失函数的情况。常系数1/2是为了计算方便美观，m是样本数据量大小，x为样本，y为样本标签，激活函数取常见的sigmoid函数：

$$lossfunction=L=\frac{1}{2m}\sum||a(x)-y||^2$$

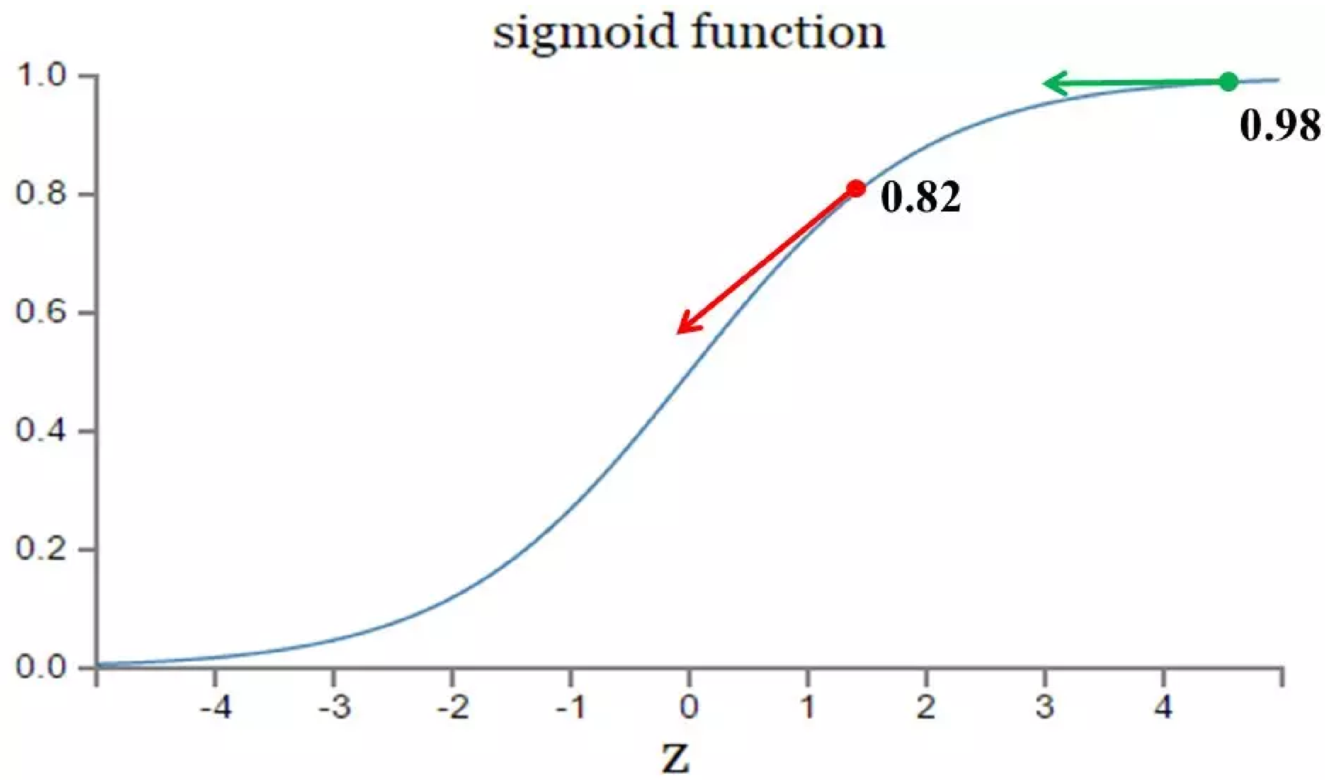
- 为了方便讨论，我们取一个样本x来推导说明，对于minibatch梯度下降只需将x替换为向量X即可。此时

$$L=\frac{||a(x)-y||^2}{2}=\frac{||\sigma(z)-y||^2}{2}=\frac{||\sigma(wx+b)-y||^2}{2}$$

于是可得，偏导数

$$\frac{\partial L}{\partial w}=(a-y)\sigma'(z)x$$

$$\frac{\partial L}{\partial b}=(a-y)\sigma'(z)$$



- 我们知道sigmoid函数图像如上所示，它的函数值在z很小或者很大时变化很慢，即对z的导数很小，结合以上的两个偏导公式，我们可以发现，当|z|较大时,sigmoid的导数那项较小，导致两个偏导数较小（即梯度较小），于是梯度下降时w和b的更新速度较慢，所以代价函数收敛的较慢，可以会导致梯度消失问题，为了解决这个问题，人们提出了relu等其它激活函数来使得梯度下降的速度保持较高的水平，但同时也带了其它问题，还有一种解决方案是，重新定义损失函数，使得损失函数与sigmoid函数的导数无关。

人们发现既然相对熵（等价于交叉熵+常数）可以衡量两个分布之间的差异，在二分类或多分类的机器学习任务中，输出值在0到1之间，实际上也可以认为是一种概率空间，那么其也应该可以用来作为损失函数，更让人兴奋的是，应用交叉信息熵作为损失函数时，其梯度与sigmoid的导数项无关。以下是用交叉熵作为损失度量时的梯度推导。

- 对于二分类情况(比如，判断一张图片中是否有猫)，交叉熵损失函数为：

$$L=-\frac{1}{2m}\sum_{i=1}^{m}[y_i loga_i+(1-y_i)log(1-a_i)]$$

对于多目标分类（例如一张图片中有猫有狗有老虎有狮子），n为类别数，交叉熵损失函数为：

$$L=-\frac{1}{2m}\sum_{i=1}^{m}\sum_{j=1}^{n}[y_{ij} loga_{ij}+(1-y_{ij})log(1-a_{ij})]$$

同样地，为了表达式更简洁，推导采用一个样本来说明，对于minibatch只需向量化X即可，此时

$$L=-\sum_{j=1}^{n}[y_j loga_j+(1-y_j)log(1-a_j)]$$

假设底数为2，可得偏导数

$$\frac{\partial L}{\partial w}=-\sum_{j=1}^{n}[y_j\frac{\sigma'(z_j)}{\sigma(z_j)}x_j ln2+(1-y_j)\frac{\sigma'(z_j)}{1-\sigma(z_j)}(-x_j)ln2]=-ln2\sum_{j=1}^{n}(a_j-y_j)x_j$$

$$\frac{\partial L}{\partial b}=-\sum_{j=1}^{n}[y_j\frac{\sigma'(z_j)}{\sigma(z_j)}ln2+(1-y_j)\frac{\sigma'(z_j)}{1-\sigma(z_j)}ln2]=-ln2\sum_{j=1}^{n}(a_j-y_j)$$

对比损失函数是均方误差情况下的偏导数，可以看出，交叉熵损失函数的两个偏导数的值均与sigmoid函数的导数无关，所以更容易避免梯度消失的问题，能够提高训练的速率。

5 最小化交叉熵与最大化似然函数

首先，在讲解这点之前，纠正一个广泛的错误表达，在大量的博客中，充斥着一句话"最小化交叉熵相当于求最大似然估计"，这句话是有一些问题的，在于最大似然估计是求参数，最小化交叉熵不仅要求参数，还要给出损失大小。

- 最大似然估计是基于统计方法去估计模型参数从而重建模型的方法。最大似然估计的基本过程是对已知的分布中独立同分布地抽取出n个样本，然后利用这n个样本去估计该分布的未知参数。例如我们知道高考成绩服从正态分布，但我们不知道这个正态分布的均值和方差，于是我们可以从考生成绩样本空间中独立同分布的抽取足够多的n个样本，然后利用统计的方法估计出均值和方差，这就是最大似然估计的过程。
- 那么似然函数又是什么?似然函数是在求最大似然估计过程中用到的一个概率，这个概率是抽样的n个样本的联合概率分布，可以写作

  $$P = f_{model}(x1, x2, \ldots xn|\theta)$$

  而x1,x2,xn是独立同分布的，这个时候

  $$P = f_{model}(x1, x2, \ldots xn|\theta) = \prod_{i=1}^{n} f(x_i|\theta)$$

  参数θ是需要估计的，参数θ需要使P最大，我们称P为似然函数，把求使P最大的参数θ的过程叫最大似然估计，为了简化计算，我们定义下式对数似然函数

  $$L = \sum_{i=1}^{n} log f_{model}(x_i|\theta)$$

  由于log函数为单调函数，所以对数似然函数最大时，似然函数也最大，此时对应的最大似然估计为

  $$\hat{\theta} = argmax \sum_{i=1}^{n} log f_{model}(x_i|\theta)$$

  对此式除以n，不改变最大似然估计的值，于是

  注意到式

  $$E_{f_{data}} log f_{model}(x_i|\theta)$$

  恰巧就是交叉信息熵的相反数，于是我们得知，我们可以知道最大化似然函数（也即最大化上式），就相当于是最小化交叉信息熵。

6 总结与预告

本期主要介绍了信息论中的相关概念，深入浅出的带大家推导和理解了信息熵交叉熵等概念，并将均方误差损失和交叉熵损失进行对比，给出了多目标分类下的证明，随后对最小化交叉熵与最大化似然函数的等价关系进行了证明，下期我们将对最大似然估计的"亲朋好友"进行介绍。

7 本期资料

7.1 本文相应的代码及资料已经以.ipynb文件和.pdf形式在github中给出。

- .ipynb文件在链接https://github.com/LSayhi/book-paper-note
- .pdf文件在链接https://github.com/LSayhi/book-paper-note

欢迎star,fork,pull

点击【阅读原文】，本期资料github链接

往期精彩推荐

机器学习系列（1）：十分钟掌握深度学习的原理、推导与实现
机器学习系列（2）：初始化的一小步，网络性能的一大步

机器学习系列（3）：几分钟了解正则化及其python实现

机器学习系列（4）：除了双路泰坦，你还有优化算法

机器学习中的数学（1）：MIT大牛的机器学习数学综述

没遇到你之前，我的世界是混沌，和你在一起后，我们的路将明确。

-by LSayhi的信息论

▼ 更多**原创干货**和最新资讯，请**关注**我吧 ▼

AI有点可ai

阅读原文