贝叶斯理论

侠之大者

2016 年 3 月 17 日

在概率与统计领域中，贝叶斯理论可谓是重中之重。而在概率与统计广泛应用于机器学习，数据挖掘，图像处理等学科的今天，贝叶斯理论就显得尤为重要。

现代的概率论是由苏联数学家柯尔莫哥洛夫于 1933 年在实分析的基础上建立的，这使概率论变得严谨而抽象。而我们这里说的概率论是传统的，经典的，也是在科研和实践中被大家广泛使用的。

1 随机变量

在概率论中，最基础的概念莫过于随机变量。那什么是随机变量，为什么需要随机变量？大家可能从来没有想过这样的问题。这里有两段对话，大家可以感受一下。

1.1 两段小对话

对话一：

A：桌子上有一枚硬币，它是正面朝上，还是反面朝上呀？

B (看了一眼)：正面朝上。

A：如果我把这枚硬币抛起落到桌面之后，那它是正面朝上，还是反面朝上？

B：都有可能吧。

那现在问题来了，如何描述正面和反面都有可能？说是正面吧，它可能反面朝上；说是反面吧，它有可能是正面朝上，纠结。

对话二：

A：你今年多大了？

B（不假思索）：25 岁。

A：那你们班的同学今年都多大了?

B：◦◦◦ (我经常在 QQ 聊天时遇到，这次终于派上用场了)

如何刻画一个班甚至一个学校的学生的年龄? 可能 24 岁，可能 25 岁，麻烦。

1.2 概率化描述

人们思考和解答上述的问题的时候，就产生了随机变量的概念。

我们先来说说确定变量吧。比如在第一段对话中，一开始躺在桌子上的硬币，它的正面朝上状态是确定的；在第二段对话中，同学 B 的年龄是 25 岁，这也是确定的。

而抛弃后的硬币的状态是不确定的，一个班同学的年龄也是不确定的。随机变量就是不确定的变量，有好几种甚至无数种情况可能出现的变量。那么如何描述和刻画这种变量?

我们的做法是: 量化每一种可能性，并给出对应的概率。

对于一枚抛起来的硬币，设其落地后的状态为 X，如果正面朝上，记 $X=1$, 如果反面朝上，记 $X=-1$，则随机变量 X 为:

$$P(X=1)=1/2 \quad P(X=-1)=1/2$$

就这么简单，这就完美地刻画了硬币落地后的状态。

我们把一个班同学的年龄记为 X, 则随机变量 X 的分布可能如下:

$$P(X=24)=1/3 \quad P(X=25)=1/2 \quad P(X=26)=1/6$$

有人不禁要问，这就行了吗? 对，这就行了，不缺不漏地描述了每一种情况。

当然，这里给出的都是离散型随机变量的例子。如果是连续型随机变量，它的概率密度函数就非常完美地描述了这个随机变量。

有了随机变量的概念，我们回过头来看看确定变量。可不可以这样理解: 确定变量只是随机变量的特例，它只有某一个值，出现的概率为 1.

可以这样理解，只不过有点不太严谨。虽然确定性事件的出现的概率为 1, 但概率为 1 的事件却不一定总是出现。

1.3 更多的概念

有了随机变量的概念之后，其他的概念就纷至沓来了。

1.3.1 期望与方差

我们说描述一个随机变量，只要给出它的概率分布或者密度分布即可。但是，大部分情况下，随机变量的分布情况都比较复杂，比如一个班学生的成绩，这就需要我们创造更简单的概念去更直观地描述它。其中，期望和方差就是最常用的。

期望的定义如下：

$$E(X) = \sum_x xP(X = x) \quad or \quad E(X) = \int_x xf(x)dx$$

这里的 $f(x)$ 是连续型随机变量 X 的概率密度函数。离散和连续的区别在于求和与积分。

方差的定义如下：

$$D(X) = E((X - E(X))^2) = E(X^2) - (E(X))^2$$

这个公式相信大家都会背吧，有时间可以简单推导一下，像我这样的就不用了，因为我一眼就能看出来，嘿嘿。

1.3.2 随机事件

随机事件，就是随机变量取某个值或某些值时定义的事件。

比如说，硬币抛起后正面朝上就是一个随机事件，我们这里用 $X = 1$ 来描述，其对应的概率为 $P(X = 1) = 1/2$.

再比如说，随机从一个班抽出一个学生，他的年龄为 x, 这也是一个随机事件，这里用 $X = x$ 表示，其对应的概率为 $P(X = x)$，或者简写为 $P(x)$.

在这里，我们不厌其烦地这样去写，主要是说明一些数学符号的表示问题。一般来说，随机变量用大写字母表示，其对应的数值用小写字母表示。

顺便说一句，有时候不得不吐槽有些教材和论文，滥用、混用数学符号，给读者增加了理解上的负担。

1.3.3 样本

这是一个非常非常重要的概念。在现在的“大数据时代”，有可能穷尽随机变量的每一种可能性。但是在以前，我们想要了解一个比较复杂的随机变量，获得它的分布，只能通过抽样的方法。

我们这里说的抽样指的是简单随机抽样，它可以保证抽取的样本是独立同分布的。简单随机抽样是指抽出一个样本后，再放回去，搅一搅，再抽下一次。这样上一次抽取的结果就不会影响下一次。

而我们日常生活中，往往是不放回的抽样。这样的话，如果总体的数量比较多的话，也可以近似认为是简单随机抽样。

在一次简单随机抽样中，我们抽取了样本 $D = \{x_1, ..., x_i, ..., x_n\}$，那么，我们抽取这个样本的概率为：

$$P(D) = P(x_1, ..., x_i, ..., x_n) = \prod_i^n P(x_i)$$

这里之所以可以写出连乘的形式，是因为样本之间相互独立。

1.3.4 条件概率

这里介绍条件概率，主要是为了后面介绍贝叶斯定理做准备。

对于一个随机事件 x，如果我们已经知道与之相关的另一个随机事件 y 已经发生了，那么在这种情况下，随机事件 x 发生的概率多大呢?

$$P(x|y) = \frac{P(x,y)}{P(y)} \tag{1}$$

其中，$P(x,y)$ 表示随机事件 x, y 同时发生的概率。

这里用一张图来表示就非常清楚。

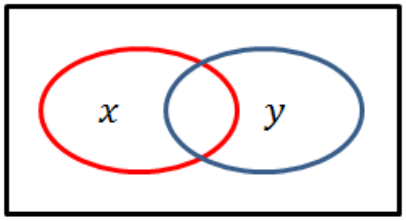

图 1: 条件概率示意图

2 贝叶斯定理

我们长舒一口气，终于迎来了最重要的内容：贝叶斯定理。废话不说，直接上公式:

$$P(x|y) = \frac{P(x)P(y|x)}{P(y)}$$

很多人疑惑不解，为什么贝叶斯定理会这么重要，这不就是把条件概率的公式 (1) 中的分母变换了一下，$P(x,y) = P(x)P(y|x)$。

不急，慢慢来，我们将用本文剩下的所有篇幅来讲述贝叶斯定理。

2.1 简单的介绍

首先，我们给出贝叶斯定理各部分的称呼。

$P(x)$ 为先验概率，英文为 prior;

$P(y|x)$ 为似然，英文为 likelihood;

$P(x|y)$ 为后验概率，英文为 posterior;

$P(y)$ 为证据，英文为 evidence, 主要是为了概率归一化，不知道为啥取这个名字，有点丈二和尚。

这里有 4 个部分，其中“证据”$P(y)$ 不用太多关心，因为在讨论后验概率 $P(x|y)$ 的时候，随机事件 y 已经发生了，$P(y)$ 是一个确定的值。

下面分别用中文和英文描述一下贝叶斯定理;

$$\text{后验概率} = \frac{\text{先验概率} \times \text{似然}}{\text{证据}} \qquad posterior = \frac{prior \times likelihood}{evidence}$$

2.2 通俗的解释

几乎所有的教材都会说：贝叶斯定理把先验概率转化成了后验概率。

人们一开始对某一件事有自己的认识，这就是先验。但在过程中，发生了其他的事，改变了人们原先的想法，产生了新的理解，这就是后验。

在举世瞩目的人机大战中，起初你根据自己对围棋和人工智能的了解，预测李世石大胜 AlphaGo，这是你的先验判断。可是当第二盘结束的时候，AlphaGo 以 2:0 领先李世石，你就会有了新的认识，这是你的后验。

可能你现在也没有看懂，我也不知道怎么把它说明白，毕竟能力有限嘛。这样吧，大家把贝叶斯定理的公式，包括中英文，多看几遍，赶紧进入下一节吧。

3 参数估计

我们的生活中处处充满着随机性，如何去描述这些随机现象呢？前面讲到，使用随机变量即可。只要我们掌握了随机变量的概率分布或密度分布，就能透彻地理解随机现象的规律。就这么简单，真的是这样。

那问题来了，如何获得某一个随机变量的分布呢？列举每一种可能性，并逐一计算概率，实践中这种做法不可行。

实践中，我们获取一个随机变量的分布，就像买衣服一样，先选款式，再试大小。

首先，选款式。教材上不是介绍了各种各样的分布吗？常见的离散型分布有贝努利分布，二项分布，泊松分布，几何分布等，连续型分布有均匀分布，高斯分布，指数分布等。在这些分布之中，我们拍一下脑袋，找到一个合适的分布先给随机变量套上。

接着，试大小。这些分布一般都预留了一两个参数，提供了相当的灵活性。比如高斯分布 $N(\mu, \sigma^2)$, 我们可以通过参数 μ, σ 去控制分布的形状。因此，我们需要为所选分布选择合适的参数。那如何估计分布的参数，自然是通过样本呀。

我们重新来梳理一下这个过程。

有一个随机变量 X, 我们假设它服从某一分布，即 $X \sim F(\theta)$, 不过这个参数 θ 未知。我们从这个随机变量 X 中抽取 n 个样本 $D = \{x_1, ..., x_i, ..., x_n\}$, 利用这些样本，我们来估计参数 θ 的值。

关于参数估计的问题，这里就涉及了派别之争。

3.1 派别之争

说到派别之争，这往往比杀父之仇，夺妻之恨更可怕。

在近代物理学上，光的微粒说和波动说，持续争论了四百多年，到现在也没有弄清楚。相信大家都做过托马斯 · 杨的双缝干涉实验吧。

在 « 数学之美 » 中, 吴军老师介绍了自然语言处理领域中的基于规则和基于统计的理念之争。不过现在，基于规则的理论已经被淘汰在历史长河中了。

在我们机器学习领域，也存在着神经网络 (Neural Network) 与支持向量机 (SVM) 之争。上世纪 80 年代，那时候流行神经网络。从 90 年代初开始，SVM 被 Vapnik 等人提出来 (to kill Neural Network)。SVM 凭借其简

明的算法和优越的性能，风头迅速盖过了神经网络。那时，神经网络无人问津。但是自从 10 年前，Hinton 大师提出了深度学习 (call SVM shallow learning)，即深层神经网络，现在无人不知，无人不晓。

为什么神经网络会再次崛起呢? 因为以前的神经网络只有 3 层，很难有效地学习许多知识。而且那时候用来训练的数据量太小，很容易过拟合，计算能力也跟不上。现在据说，余凯在百度训练深度网络时，使用了一千亿个训练样本。在以前，这不可想象，估计连数据存储都成问题，那时候还没有 Hadoop,Spark 这样的工具。

不管怎样，学术界的派别之争大大促进了学术的繁荣!

在参数估计这个问题上，有频率学派和贝叶斯学派，他们对这个问题有不同的看法。

频率学派认为参数就是参数，是一个固定的值，只是未知而已，可以使用最大似然法去估计参数。

贝叶斯学派认为，参数 θ 不是一个固定不变的数，而是服从一定分布的随机变量。在被估计之前，参数 θ 有一个先验分布 $P(\theta)$。

3.2 最大似然估计 (Maximum Likelihood)

最大似然估计就是求出使样本出现概率最大的那个参数。样本出现的概率为:

$$P(D;\theta) = P(x_1, ..., x_i, ...x_n; \theta) = \prod_i P(x_i; \theta)$$

一般的，我们将似然函数记为:

$$L(\theta) = \prod_i P(x_i; \theta)$$

为了求解方便，一般同时两边取对数，得到对数似然函数

$$lnL(\theta) = \sum_i lnP(x_i; \theta)$$

最大似然估计公式为:

$$\hat{\theta}_{ML} = arg \max_{\theta} L(\theta) = arg \max_{\theta} lnL(\theta)$$

接着对其求导，令导数为 0, 便求出了要估计的参数值。这个过程相信大家都比较熟悉。

现在我们来探讨一些细节。

先来看看最大似然估计名称的由来。“似然”存在于贝叶斯学派的话语体系之中。在贝叶斯学派看来，参数 θ 有其先验分布 $P(\theta)$，样本出现的概率为 $P(D|\theta)$，即为似然。当我们抽样后，参数 θ 的后验分布为 $P(\theta|D)$。这样看来，频率学派这样求参数的过程的确就是最大似然估计。

还有一个小细节：在写最大似然估计的时候，我用分号”;” 来分割样本和参数 θ，而在贝叶斯表述中，使用的是”|”。这样因为频率学派认为参数 θ 是一个确定的数，不是一个随机变量。其实，这两者的数学表达是一致的。

3.3 贝叶斯估计

贝叶斯估计方法估计参数时，完全根据贝叶斯定理的框架来的。贝叶斯估计认为待估计的参数 θ 是符合某个先验分布 $P(\theta)$ 的随机变量，通过对样本 D 的观测，修正了对参数的初始估计，得到了参数的后验分布 $P(\theta|D)$。

这里给出贝叶斯方法估计参数 θ 的公式:

$$P(\theta|D) = \frac{P(\theta)P(D|\theta)}{P(D)} = \frac{P(\theta)P(D|\theta)}{\sum_\theta P(\theta)P(D|\theta)}$$

3.3.1 简单的例子

这里给出一个简单的例子，稍微复杂的例子可能计算就比较麻烦。

有一枚质地不均匀的硬币，抛起后，正反面朝上的概率不一样。当正面朝上时，我们记随机变量 $X = 1$；当反面朝上时，记 $X = 0$。这样，随机变量 X 服从贝努利分布 (即通常说的两点分布)$X \sim B(\theta)$，其中

$$P(x) = \begin{cases} \theta & X = 1 \\ 1 - \theta & X = 0 \end{cases}$$

在抛这枚硬币之前，我们把它放在手上观察一番，觉得正面朝上的概率比较大。于是，一拍脑袋，大手一挥，给出了参数 θ 的先验分布 $\theta \sim U(1/2, 2/3)$，这里 $U(a, b)$ 指的是 $[a, b]$ 上的均匀分布。参数 θ 的概率密度函数为

$$f(\theta) = \begin{cases} 6 & 1/2 \le \theta \le 2/3 \\ 0 & else \end{cases}$$

现在有了随机变量 X 的分布函数，和参数 θ 的先验分布。接下来我们需要抽样，来得到参数 θ 的后验分布。

于是我们开始抛硬币，一抛，嘣的一声，落在桌面上，我擦，反面朝上。好吧，就这样吧，我们来计算此时 θ 的后验概率分布吧。

样本 $D = \{x = 0\}$，样本的似然为

$$P(D|\theta) = 1 - \theta$$

$$P(D) = \int_{\theta} f(\theta)P(D|\theta)d\theta = \int_{1/2}^{2/3} 6(1-\theta)d\theta = 5/12$$

于是参数 θ 的后验分布为

$$f(\theta|D) = \frac{f(\theta)P(D|\theta)}{P(D)} = \begin{cases} \frac{72}{5}(1-\theta) & 1/2 \le \theta \le 2/3 \\ 0 & else \end{cases}$$

3.3.2 图形化表示

为了更清楚地显示这次实验对后验概率分布的影响，我们把参数 θ 的先验分布 (蓝色) 和后验分布 (绿色) 画了出来，如图 (2) 所示。我们可以很明显地看出，后验分布曲线往 1/2 这一端倾斜，这是符合我们的预期的。

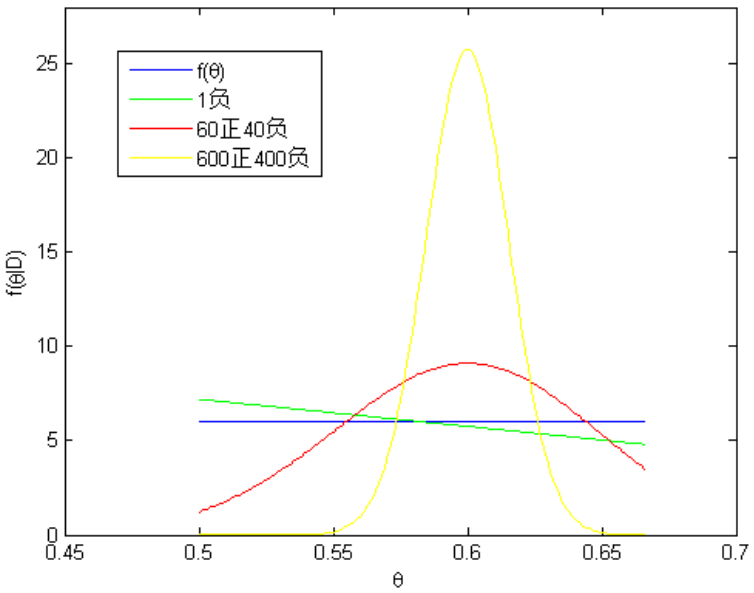

图 2: 后验概率密度分布

这里，我们来对比一下最大似然估计和贝叶斯估计的结果。对于这次实验，如果我们采用最大似然估计，只有一个样本，反面朝上，显然 $\hat{\theta}_{ML} = 0$。

有人肯定说，这样本数量太少。假如我们做了 100 次实验，其中 60 次正面朝上，40 次反面朝上。这样的话，我们可以口算出，最大似然估计的结果为 $\hat{\theta}_{ML} = 0.6$。但对于贝叶斯估计，我们只能用 Matlab 去计算，计算后的后验概率密度分布是图 (2) 的红色曲线，胖胖的，似乎也在 0.6 处附近达到最高点。

这时，如果我们再增加样本的个数，比如做了 1000 次实验，其中 600 次正面朝上，400 次反面朝上。最大似然估计的结果还是 $\hat{\theta}_{ML} = 0.6$, 我们看此时的贝叶斯估计结果，后验概率密度分布是图 (2) 的黄色曲线，在 0.6 处变得非常尖锐。

3.3.3 简单的对比

首先，我们说贝叶斯估计提高了模型的稳定性，特别是在样本数量比较少的时候。可以这样理解：最大似然估计只利用了样本的信息去估计参数，而贝叶斯估计是通过样本修正了人们对参数的先验认识。

然后，贝叶斯估计在计算上比较繁琐，不如最大似然估计简单明了，而且，算来算去，还都是参数的密度分布。

那有没有一种参数估计的方法，既可以利用先验知识，又比较容易计算呢？有，那就是下一节的最大后验概率估计。

3.4 最大后验概率估计 (MAP)

最大后验概率估计的思想比较简单。

首先，它仍然假设参数 θ 是一个随机变量，有一个先验分布 $P(\theta)$。然后，它并不是求参数 θ 的后验分布，而是求使后验分布值最大时的参数值。在图 (2) 中，最大后验概率估计的结果就是后验概率密度曲线最高点处对应的参数值 θ。

最大后验概率估计的公式为：

$$\hat{\theta}_{MAP} = arg\max_{\theta} P(\theta|D) = arg\max_{\theta} P(\theta)P(D|\theta)$$

因为 $P(D)$ 是一个定值，这里可以略去。

再来看看最大似然估计：

$$\hat{\theta}_{ML} = arg\max_{\theta} P(D;\theta) = arg\max_{\theta} P(D|\theta)$$

对比一下，我们可以发现，两者的区别只是在优化的表达式中多了参数 θ 的先验分布 $P(\theta)$。

3.4.1 硬币实验结果

我们回顾一下前面关于质地不均匀硬币的 3 次实验。

第一次，负面朝上。最大似然估计的结果 $\hat{\theta}_{ML} = 0$，而最大后验概率估计的结果是 $\hat{\theta}_{MAP} = 0.5$。(注意 θ 的定义域)

第二次和第三次实验中, 由于我们假设参数 θ 的先验分布 $f(\theta) = 6$ 为常数，所以，最大似然估计和最大后验概率估计的结果是相同的，即 $\hat{\theta}_{ML} = \hat{\theta}_{MAP} = 0.6$。

3.4.2 简单的变换

下面三小节的内容是我自己的思考，可能不太严谨。

我们可以简单变换一下最大后验概率的公式;

$$\begin{aligned}\hat{\theta}_{MAP} =& arg\max_{\theta} P(\theta)P(D|\theta) = arg\max_{\theta} \quad lnP(\theta) + lnP(D|\theta) \\ =& arg\max_{\theta} E(\theta) + E(\theta, D)\end{aligned}$$

通过取对数，将“乘”变成了“加”，形式上有点类似于能量函数。它还是有两项，先验项和似然项。

先验项，我们认为我们的模型应该是个什么样子。

似然项，尽可能使我们的模型贴近我们已有的样本数据。

3.4.3 机器学习与 MAP

在我们机器学习领域中，有一个非常重要的理论，偏差方差均衡。它的意思是说，通过机器学习得到的模型，既要在训练样本上的误差非常小，又要模型的复杂度尽可能地低。因为，一个复杂的模型很可能把样本数据中的噪声都学习了，这样它的测试样本上的准确率就会很差，这就是通常所说的机器学习中的过拟合现象。

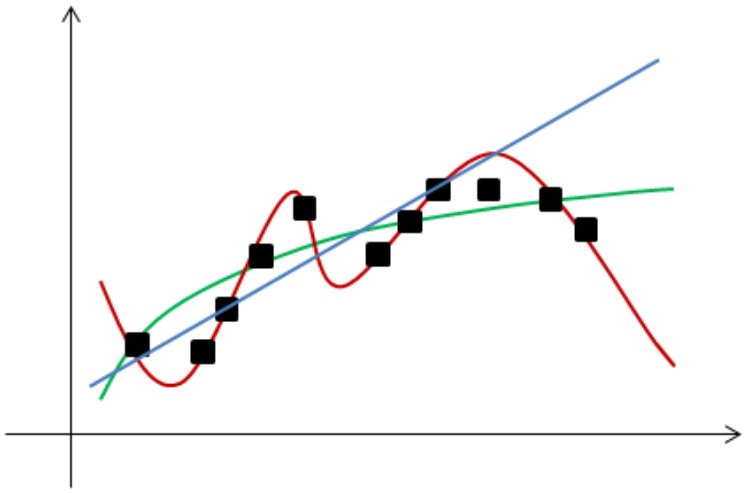

图 3: 数据拟合

如图 (3) 所示，有一组数据点 $(x_i, y_i), i = 1, ..., n$，我们通过这组数据点预测它们的走势，比如我们采用多项式函数 $h(x)$，则

$$\min_h \sum_i^n (y_i - h(x_i))^2 + \alpha d(h)$$

其中，$d(h)$ 表示多项式函数 $h(x)$ 的次数。第一项，我们称为误差项。一般来说，约束模型参数的那一项，比如这里的 $d(h)$，我们称为正则化项 (regularization, 规则化)。

在图 (3) 中，红色曲线虽然训练误差较小，但模型过于复杂，把部分噪声也当成了有用的信息。而蓝色的直线虽然模型简单，但拟合误差太大。所以，这里比较好的模型是绿色曲线。

这里，我们再来对比一下。

贝叶斯理论中的先验项与这里的正则化项，它们都是用来约束模型的先验信息。贝叶斯理论中的似然项与这里的误差项，它们都是使模型尽可能满足样本数据。

3.4.4 图像处理与 MAP

在图像处理领域，如果没有利用先验知识，可以说束手无策，寸步难行。

有一幅加了高斯噪声的图像 Y，如果不利用先验信息，根据最大似然估计，它的去噪后的图像仍然是噪声图像 Y 本身。为什么?

因为对于原图像的某个像素点 x 来说，它加高斯噪声之后，变成了 y。根据最大似然估计，这噪声值最大可能性为 0。这样的话，就有 $\hat{x} = y$。

假设有一个噪声图像块 y, 我们利用稀疏表达对其进行去噪，

$$\min_{\alpha} ||y - D\alpha||^2 + \beta||\alpha||_0$$

其中，D 为字典，α 为表达系数。这里的第一项，一般称为保真项，第二项，为稀疏项。

同样，这里的保真项与前面的似然项，误差项异曲同工，稀疏项与前面的先验项，正则化项也不谋而合。

再进一步去想，邻域中像素点的颜色值比较相似，这也是先验知识。

从以前的傅里叶变换，小波分析，到后来的稀疏，低秩，再到现在的深层卷积神经网络，都是一步一步地在寻找图像更好的先验约束，这就是图像底层处理的本质（这也是我两年来学术研究的最大心得）。

当然，现在的深层卷积神经网络可以自己学习到解决某个问题所需要的先验知识。

4 混合模型 (Mixture Model)

到这里，本来应该结束了，但是好像还差一点内容，就是这里所介绍的混合模型。

前面讲到描述一个随机变量，先假设它的分布类型，再求出分布的参数。但是在数学教材上，我们学习到的各种各样的分布都是单模态 (unimodal) 的，即从分布的几何形状来看，只有一个波峰或波谷，所以它们的实际描述能力是有限的。

比如，一个学校的学生的身高分布可能如下图 (4) 中的绿色曲线所示，有两个波峰，因为男生普遍高于女生。这样，我们很难找到一个合适分布去刻画它。怎么办?

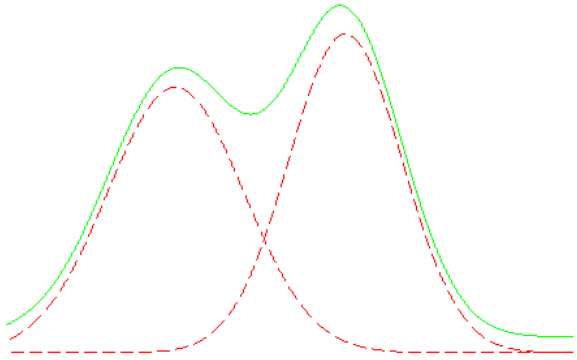

图 4: 混合密度分布曲线

在工程上，我们一般采用多个不同分布按一定比例线性叠加的方法来表示复杂的分布，这就是混合模型。在上图 (4) 中，我们可以用两个高斯分布之和去描述这个复杂的分布：

$$f(x) = w_1 N(\mu_1, \sigma_1^2) + w_2 N(\mu_2, \sigma_2^2), \quad w_1 + w_2 = 1$$

假设现在有一堆样本 $D = \{x_1, x_2, ..., x_n\}$, 并且知道每一个样本的类别 (男或女)。这样的话，上面的参数估计就比较简单，首先利用每一个类别的样本分别去估计对应类别的高斯分布的均值和标准差，再根据两个类别人数来决定 w_1, w_2 的比例。

但是，通常情况下，我们并不知道每一个样本的类别。这样的话，我们就又多了一个变量，即每一个样本属于哪一个类别，这个变量一般称为隐变量。

带有隐变量的参数估计可以期望最大化算法 (EM) 求解，它位列机器学习十大算法之中，号称上帝的算法。一般的机器学习算法是用来做分类或者回归预测的，而期望最大化算法是专门用来求解带有隐变量的模型参数的。EM 算法在各个领域被广泛使用，为每一个科研人员所必备技能。

首先，随机地给每一个样本初始化一个类别，这样我们估计出当前的每个类别的参数和对应的比例。

接着，利用上面求出来的参数，我们再来更新每一样本的类别。

这样周而复始，交替迭代，直至收敛。

这里只是简单介绍了 EM 算法，详细的内容可以参考李航老师的《统计学习方法》教材的第九章，满满的干货。

就到这里吧，估计大家都没有耐心再看下去了。♡