核密度估计的概念与实验

June 4, 2014

1 问题

本文讲述核密度估计的概念与原理，同时给出一个实验。文章是参考 wikipedia 上的定义[1]及 python 中的 scipy 包[2][3]的源码写就的。

2 解答

2.1 概念

以函数估计打比方。现在手上有几个函数值 (x_i, y_i)，即平面上的几个点。现在要从这个几个点，推测是从哪个函数中抽样的。如 $y = kx + b$,$y = x^2$ 等。如果得到函数的表达式，就可以得到在指定区间，如 $[-5, 5]$ 内，任意位置处的函数值了。

核密度估计是一样的概念。估计的是概率密度函数。

从几个样本，估计其服从的分布，即求出其概率密度函数。有了概率密度函数以后，就可以得到在任意区间（值）处的概率了。

所以，核密度是一个从具体 (样本) 到普遍（概率密度函数）的过程。然后再用普遍指导具体。

[1] http://en.wikipedia.org/wiki/Kernel_density_estimation

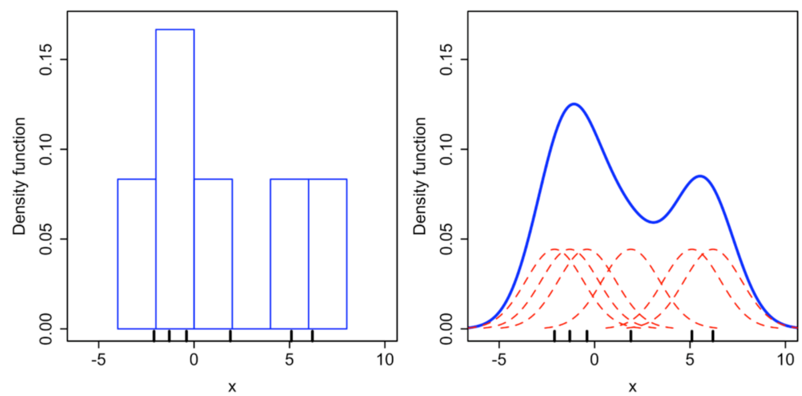
[2] http://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gaussian_kde.html

[3] https://github.com/scipy/scipy/blob/master/scipy/stats/kde.py

2.2 如何估计

2.2.1 放草帽求平均

样本到底是从哪个分布中抽样的，我们不知道。不如假设每个样本都服从高斯分布，在该点处做一个高斯分布的图形。有N个样本，就能得到N个高斯分布。如 wiki 页面给出的图：



其中，6 条短黑线，即 6 个样本；6 个红草帽，即样本所在点的高斯分布；一条蓝线，即密度估计的结果。有了这条蓝线，我们可以知道任意处的函数值。

在 [-5,10] 的范围内，取任一点 x_0 为例，其所在处的值，认为等于N个高斯分布在此处的平均值。即，

$$F(x_0) = \frac{g_1(x_0) + g_2(x_0) + g_3(x_0) + g_4(x_0) + g_5(x_0) + g_6(x_0)}{6} \tag{1}$$

更简洁（唬人）的写法是：

$$F(x_0) = \frac{1}{N}\sum_{i=0}^{N} G_i(x_0) \tag{2}$$

如果说一个高斯分布是一个草帽的形状。上面这个过程就是一个放草帽的过程。在所有的样本点处放一个草帽，然后所有草帽相加求平均。

2.2.2 为什么假定服从高斯分布？

选高斯分布的合理性在哪里？这跟中心极限定理[4]有关系。即，任何分布，经过多次独立实验，最终都服从高斯分布。高斯分布的特殊性可见一斑。

[4]http://en.wikipedia.org/wiki/Central_limit_theorem

说穿了很简单，一群人中，（成绩 / 品德 / 身高 / 财富）特别好的特别坏的都是少数，大部分是普通人。这与我们的生活经验相符。不符合这个规律？那是因为你取的样本不够多。

高斯分布又叫正态分布，就是正常状态的意思。

高斯函数在这里就叫“核函数”。当然有别的核函数可选。但那都不是正常状态。

2.2.3 草帽的形状

放草帽求平均就是核密度估计。那么，如何确定草帽的形状？草帽的形状与高峰期分布的参数有关，以 1 维为例：

$$G(x) = \frac{1}{\sqrt{2\pi} * \sigma} e^{-\frac{(x-\mu)^2}{2*\sigma^2}} \tag{3}$$

μ，即期望，决定了草帽的中轴在哪里。

σ^2，即方差，决定了草帽的宽度 (与高度)

在核密度估计的过程中，即放草帽的过程中：

期望，是各个样本点的位置，

方差，是所有样本点的方差。

2.2.4 引入带宽的概念

既然是估计，肯定就会有校正。比如，估计的概率密度函数新鲜出炉了。结果发现用它求出的值总是偏大，怎么办？那就缩放一下吧：

$$G(x) = \frac{1}{\sqrt{2\pi} * \sigma * h} e^{-\frac{(x-\mu)^2}{2*\sigma^2*h^2}} \tag{4}$$

合并一下的话，可以看到，本质是将 σ 变成了 $\sigma * h$。这个缩放因子 h，通常叫做带宽。

如何计算带宽？那就是靠经验了。各家有各家的方法，就是”rule of thumb”。一般跟样本的数量及样本的维度有关系。

2.3 实验

真正的应用场景，不知道分布是概率密度函数是什么样的。做实验就不一样了：

- 从标准高斯分布 (μ=0,σ=1) 里抽取 100 个点, 区间为 [-5,5]。
- 按照上面的方法，估计出一个概率密度函数 f(x)
- 依据密度函数，画出在 [-5,-5] 的分布

观察画出的分布，与标准高斯分布的曲线越接近，说明估计的越准确。

2.3.1 准备数据

从标准高斯分布里，生成 50 个随机数。

2.3.2 求方差

```
mean = sum(dataset)/N
variance = (dataset − mean)*(dataset − mean)/(N−1)
```

2.3.3 求带宽

这里选用的是 scipy 库的默认的 scotts_factor。其计算与两个因素有关：

- 样本的个数，N
- 样本的维度，d

在这个实验里，显然，$N = 50$,$d = 1$

$$h = N^{-\frac{1}{d+4}} \tag{5}$$

2.3.4 计算过程

以 [-5,5] 中的一点 x_{pos} 的值为例，计算该处的函数值：

$$F(x_0) = \frac{1}{N*h}\sum_{i=0}^{N} e^{-\frac{(x_{pos}-x_i)^2}{2*\sigma^2*h^2}} \tag{6}$$

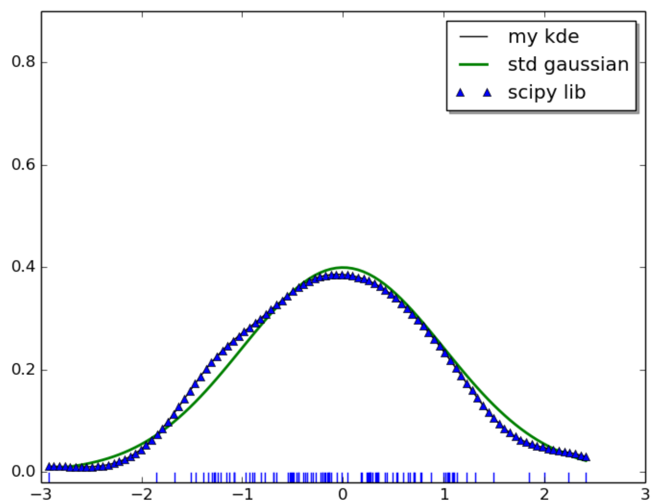
x_i 为所有的样本点，可以看到:
当 $i = 0$ 时，表示以 x_0 为中轴的高斯分布，在 $x = x_{pos}$ 时，取得的值
当 $i = 1$ 时，表示以 x_1 为中轴的高斯分布，在 $x = x_{pos}$ 时，取得的值

当 $i=2$ 时，表示以 x_2 为中轴的高斯分布，在 $x=x_{pos}$ 处，取得的值……

对所有的 x_i，放置一项草帽，再将草帽在 x_{pos} 处的值累加，求平均。就认为是 x_{pos} 的值。

x_{pos} 属于 [-5,5] 的任意点，所以求整个分布，就是将这个公式计算 M 次，M 是 [-5,-5] 被分成了多少个间隔。间隔越小，画出的曲线越平滑。

2.3.5 实验结果



可以看到，本文的结果与 scipy lib 的结果是一致的，重合。因此准确性没问题。

2.3.6 源码

完整的源码在：https://github.com/xueyayang/v4l2_demo/blob/master/kernel-density-estimation/gaussian_1d_kde.py

2.3.7 复杂度分析

样本数 N，估计位置数 M。O(n)=M*N

3 总结

- 计算方差很重要。
- 计算带宽很重要。
- 熟悉高斯公式很重要。
- 囿于个人所学，文章重应用不重数学原理推导。