Chapter 4 非参数技术

Author： 中山大学 17数据科学与计算机学院 YSY

https://github.com/ysyisyourbrother

参数估计存在以下问题：

- 给出的概率密度的形式不符合实际情况
- 实际问题是多模的密度函数
- 高维密度函数表示成一些一维密度函数的乘积的假设通常不成立

非参数技术能处理任意的规律分布，而不必假设密度的参数形式已知。

局限性：

- 实际需要的训练样本的个数极大
- 维数灾难

概率密度的估计

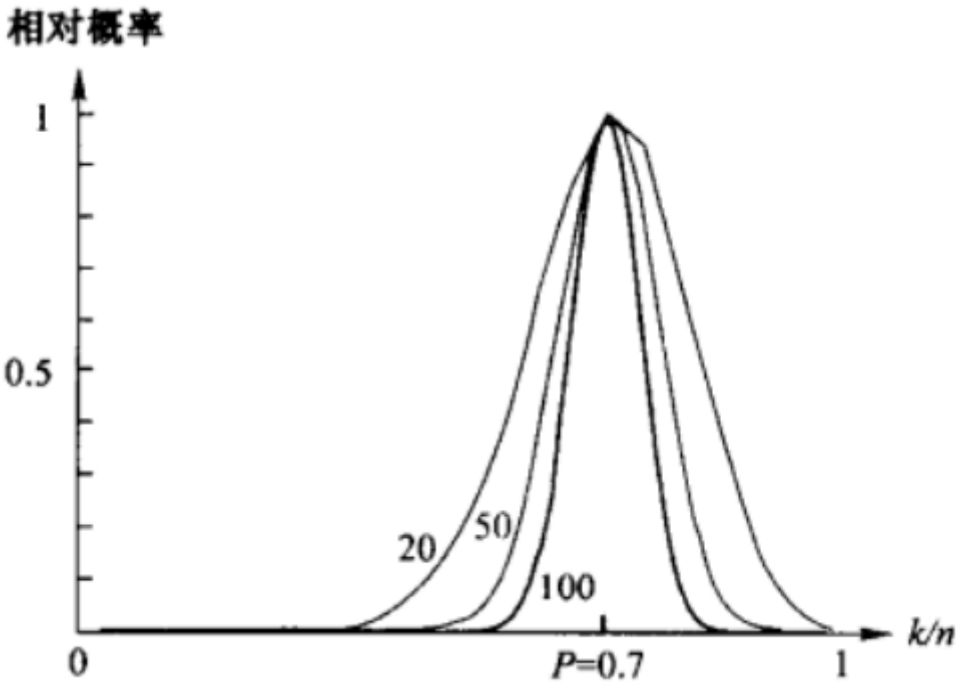
一个基本事实是：一个向量x落在区域R中的概率为：

$$P = \int_R p(x')dx'$$

其实就相当于落在区域R中的所有x出现的概率和。因此P是概率密度函数p(x)的平滑后的（求和也就相当于考虑了所有点也就相当于平滑）版本。因此，我们可以通过估计P来估计概率密度函数p。

假设n个样本$x_1, x_2 \dots x_n$是从p(x)中独立同分布抽取的。服从二项式：

$$P_k = \binom{n}{k} P^k (1-P)^{n-k}$$



假设落在区域R中的概率是0.7，那我们随机采样n个，落在区域R中的点和总的采样点的比值就会有比较大的概率在0.7左右，但因为是随机采样n个可能有偏差。如果采样的数量n增大，此时就会在真正的概率0.7处形成显著的波峰，当n趋于正无穷时，曲线逼近一个冲激函数。

k的二项式形式的分布在**均值附近有非常显著的波峰**，因此我们可以想象到比值k/n就是概率P的一个很好的估计。**我们假设p(x)是连续的，并且区域R足够小**，以**致于在这个区间中p几乎没有变化**（p是连续的概率密度函数，只有区间足够小可以看做概率密度没有改变）。那么有

$$P = \int_R p(x')dx' \approx p(x)V$$

x代表R中一个点，而V是体积，也可以看做有多少个点。因为假设了区间足够小，概率密度不变，因此每个点的概率密度都是p(x)。综上，我们可以得到p(x)的估计为：

$$p(x) \approx \frac{k/n}{V}$$

从表面上看可以简单理解为：k/n是x落在区域R上的概率，但因为是连续函数，单一个点上我们的概率为0，因此我们需要考虑一个区域R上的所有点，计算他们总的概率密度然后除以体积（这就是所谓平滑）也就是，取个平均值得到的就是一个x落到这个区间上的概率。因此，上式的估计实际是空间平滑后的结果。

$$p(x) \approx \frac{P}{V} = \frac{\int_R p(x')dx'}{\int_R dx'}$$

如果想要不平滑，我们就要让区域R的体积无限小，不过这样也存在问题：

1. 在固定样本个数的条件下，$V \to 0$，若R不含有任何样本，导致$p(x) \approx 0$此时估计就没有了意义。
2. 若R含有多个样本，导致$p(x) \approx \infty$。所以平滑效果必然存在。

Parzen窗引入：

考虑样本**无穷**的情况。构造一系列包含x的区域：$R_1, R_2, \ldots, R_n$，第一个区域使用一个样本，第二个区域使用两个样本，记V_n为区域R_n的体积，$p_n(x)$表示对p(x)的第n次估计：

$$p_n(x) = \frac{k_n/n}{V_n} \tag{1}$$

随着使用的样本数越来越多还不够，还需要区域R的体积越来越小， 要不会因为平滑被影响。

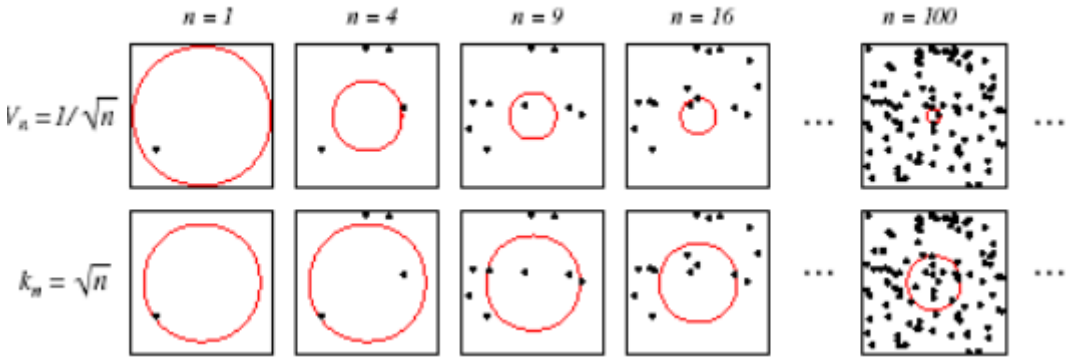
若要求 $\lim_{n\to\infty} p_n(x) = p(x)$**，变量需要满足**

$$\lim_{n\to\infty} V_n = 0$$
$$\lim_{n\to\infty} k_n = \infty$$
$$\lim_{n\to\infty} k_n/n = 0$$

- 第一个条件保证点在x连续的情况下，空间平滑了的P/V能收敛到$p(x)$。
- 第二个条件保证频率之比能收敛到概率，也就是让区域中的点充满V。
- 第三个条件对于式(1)显然是必要的。否则$p_n(x)$不存在。也是在说虽然R中的样本点非常的多，但比起全体样本是一个很小的比例。**就像在一个连续概率密度函数中，某一个点的概率为0**

获得满足上述三个条件的区域序列的方法：

- Parzen窗方法： $V_n = \frac{1}{\sqrt{n}}$，在区域内统计点数。
- k nearest neigbhor: 确定k_n为n的某个函数，如$k_n = \sqrt{n}$，进而得到体积。



Parzen 窗方法(kernel density estimation)

假设区间R_n是一个d维的超立方体。

h_n为超立方体一条边的长度

通过定义窗函数可以解析得到落在窗中的样本个数 k_n

定义$\varphi(\mathbf{u})$是一个中心在原点的**单位超立方体**

$$
\begin{aligned}
V_n &= h_n^d \\
\varphi(\mathbf{u}) &= \begin{cases} 1, & |u_j| \leq 1/2, j = 1, 2, \ldots, d \\ 0, & otherwise \end{cases} \\
k_n &= \sum_{i=1}^{n} \varphi(\frac{x - x_i}{h_n}) \\
p_n(x) &= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \varphi(\frac{x - x_i}{h_n})
\end{aligned}
$$

这个方程说明了一种更一般的估计概率密度函数的方法，**即不必规定区间是超立方体**，而可以是一个更一般的形式。上面最后一条式子告诉我们，**对p(x)的估计是每一个样本依据它离x的远近不同而对结果做出不同的贡献。**

对新的$\varphi(x)$，我们要求：

$$
\begin{aligned}
\varphi(x) &\geq 0 \\
p_n(x) &\geq 0 \\
\int p_n(x) dx &= \int \varphi(\boldsymbol{u}) d\boldsymbol{u} = 1
\end{aligned}
$$

满足上述条件的$p_n(x)$是一个合理的概率密度函数。

接下来讨论窗的宽度对$p_n(x)$的影响。重新定义核函数：

$$
\delta_n(x) = \frac{1}{V_n} \varphi(\frac{x}{h_n})
$$

核函数反应了一个观测样本x_i对在x处的概率密度估计的贡献，与样本x_i和x的距离有关。而概率密度估计就是在这一点上把所有观测样本的贡献进行平均：

$$
p_n(x) = \frac{1}{n} \sum_{i=1}^{n} \delta_n(x - x_i)
$$

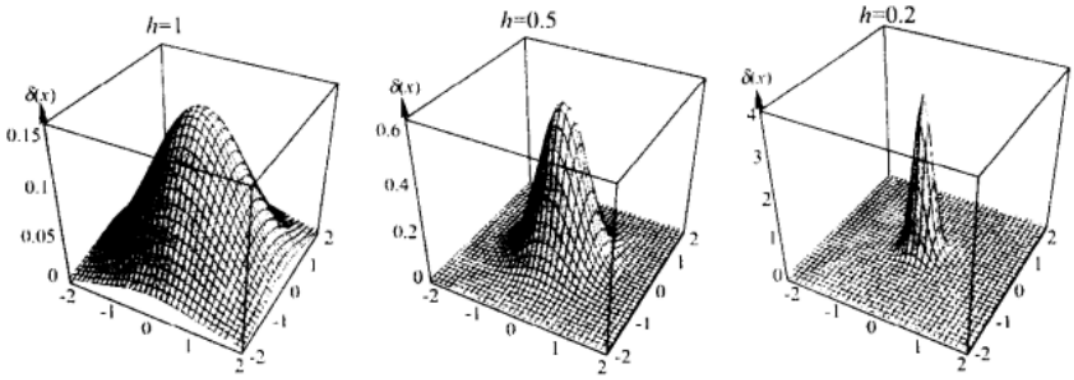
分析：若h_n很大，即使$x-x_i$很大，$\delta_n(x-x_i)$和$\delta_n(0)$的差距不大，这个时候$p(x)$是n个宽的、慢变的函数的叠加，$p_n(x)$较为平滑。若h_n很小，那么$\delta_n(x-x_i)$的峰值很大，这种情况下$p_n(x)$，**是n个以样本点为中心的尖脉冲的叠加**，是一个充满噪声的估计。当$h_n \to 0$，$\delta_n(x-x_i)$为中心在样本点x_i的狄拉克函数。当样本个数无限的时候，无论h_n多大，都能让$p_n(x)$收敛到某个概率密度函数$p(x)$。

新的核函数组成的概率密度函数，仍然需要归一化积分和为1，直接推导：

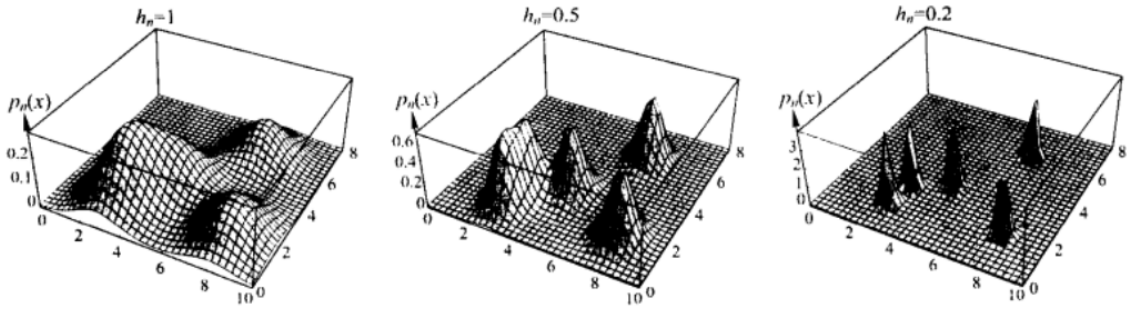
$$\begin{aligned}\int p_n(x)dx &= \int \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi(\frac{x-x_i}{h_n})dx \\ &= \frac{1}{n}\sum_{i=1}^{n}\int \frac{1}{V_n}\varphi(\frac{x-x_i}{h_n})dx \\ &= \frac{1}{n}\sum_{i=1}^{n}\int \varphi(\boldsymbol{u})du \\ &= \frac{1}{n}\times n = 1\end{aligned}$$

看上式，原本每个点的概率密度的公式是先计算每个点周围的n个点的核函数的和，然后除以总体积。但移动了积分号的位置后有了全新的理解：**每个点都对任意一个点为中心有一个概率的贡献，先将这个点对所有点的概率贡献加起来，再计算每个点的贡献和再加起来平均。**

这样就可以用高斯函数来作为核函数，每个样本对自己当前位置的贡献最大，然后向四周递减，因为要使每个点的高斯（下方体积）的和的均值为1，就要保证每个点的高斯为1。h_n影响这个高斯函数宽度。



> **Q：上图为什么是一个分布**？
>
> **A：用的核函数可能是高斯或者其他，它是一个连续的函数，保证在中间是一个波峰，并且下面的总体积为1，当h趋近于0的时候，接近于一个狄拉克函数。**



经过上面的分析，显然h_n的选取会影响估计的概率，如果h_n很大，结果是非常平滑的，是**散焦估计**，而如果h_n很小，那么峰值就会非常大，最后$p_n(x)$相当于n个以样本点为中心的尖脉冲叠加，但估计结果的**统计稳定性不够。**

在真实情况样本数量有限的情况下，我们只能做可接受的折中。不过如果样本无限，我们可以在n增加的时候，让V_n缓慢接近于0（满足上面的三个条件），最后$p_n(x)$就会收敛到某个概率密度函数$p(x)$。下面证明是可以收敛的。

收敛性证明

我们需要先知道要证明什么：因为$p_n(x)$本身就带有均值和方差的属性，如上面的高斯窗图举例。于是我们证明的目标就是：

$$\lim_{n\to\infty}\hat{p}_n(x)=p(x)$$
$$\lim_{n\to\infty}\hat{\sigma}_n^2(x)=0$$

证明收敛性用到以下条件（习题1）

$$\sup_u\varphi(u)<\infty$$
$$\lim_{||u||\to\infty}\varphi(u)\prod_{i=1}^{d}u_i=0$$
$$\lim_{n\to\infty}V_n=0$$
$$\lim_{n\to\infty}nV_n=\infty$$

均值收敛性

$$\begin{aligned}\bar{p}_n(x)&=\mathbb{E}[p_n(x)]\\&=\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}(\frac{1}{V_n}\varphi(\frac{x-x_i}{h_n}))\\&=\int\frac{1}{V_n}\varphi(\frac{x-v}{h_n})p(v)dv\\&=\int\delta_n(x-v)p(v)dv\\&=p(x)\end{aligned}$$

这里的期望是对x_i作用的，x是窗口的中心。因为每个样本都是独立同分布，所以，他们的期望都相同。

注意： 使$\bar{p}_n(x)$趋近于$p(x)$，并没有必要获得无限多的训练样本，也可以直接让V趋于0。

方差收敛性

由于$p_n(x)$是一些关于关于独立的随机变量的函数的和，方差是分开的项的和。

$$\begin{aligned}\sigma_n^2(x)&=\sum_{i=1}^{n}\mathbb{E}[(\frac{1}{nV_n}\varphi(\frac{x-x_i}{h_n})-\frac{1}{n}\bar{p}_n(x))^2]\\&=n\mathbb{E}[\frac{1}{n^2V_n^2}\varphi^2(\frac{x-x_i}{h_n})]-\frac{1}{n}\bar{p}_n^2(x)\\&=\frac{1}{nV_n}\int\frac{1}{V_n}\varphi^2(\frac{x-v}{h_n})p(v)dv-\frac{1}{n}\bar{p}_n^2(x)\\&\le\frac{\sup(\varphi(u))\bar{p}_n(x)}{nV_n}\end{aligned}$$

举例说明

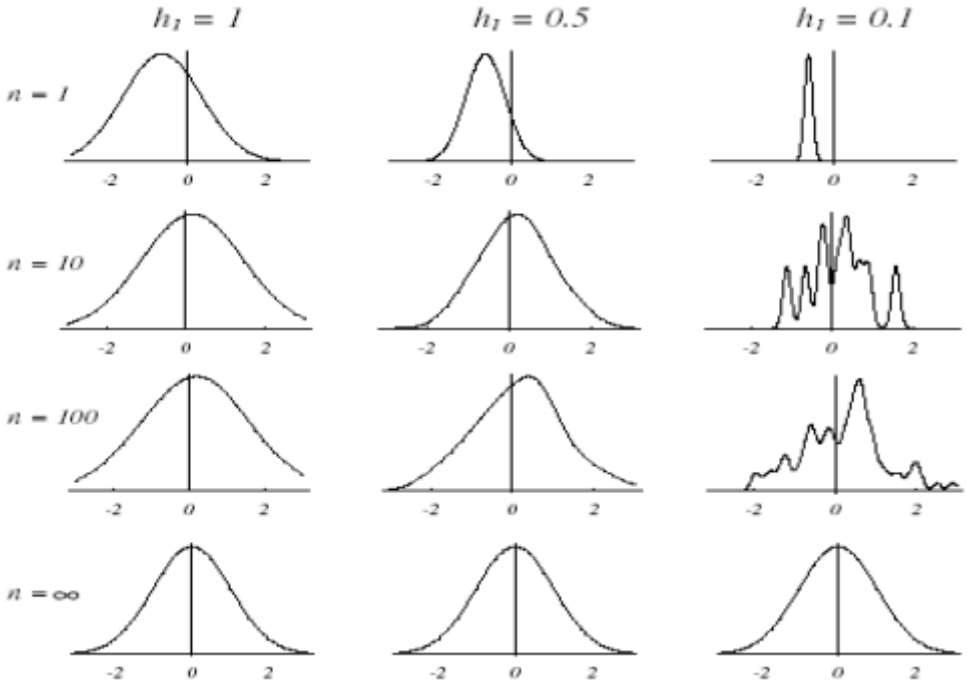
一维高斯分布

定义窗函数为

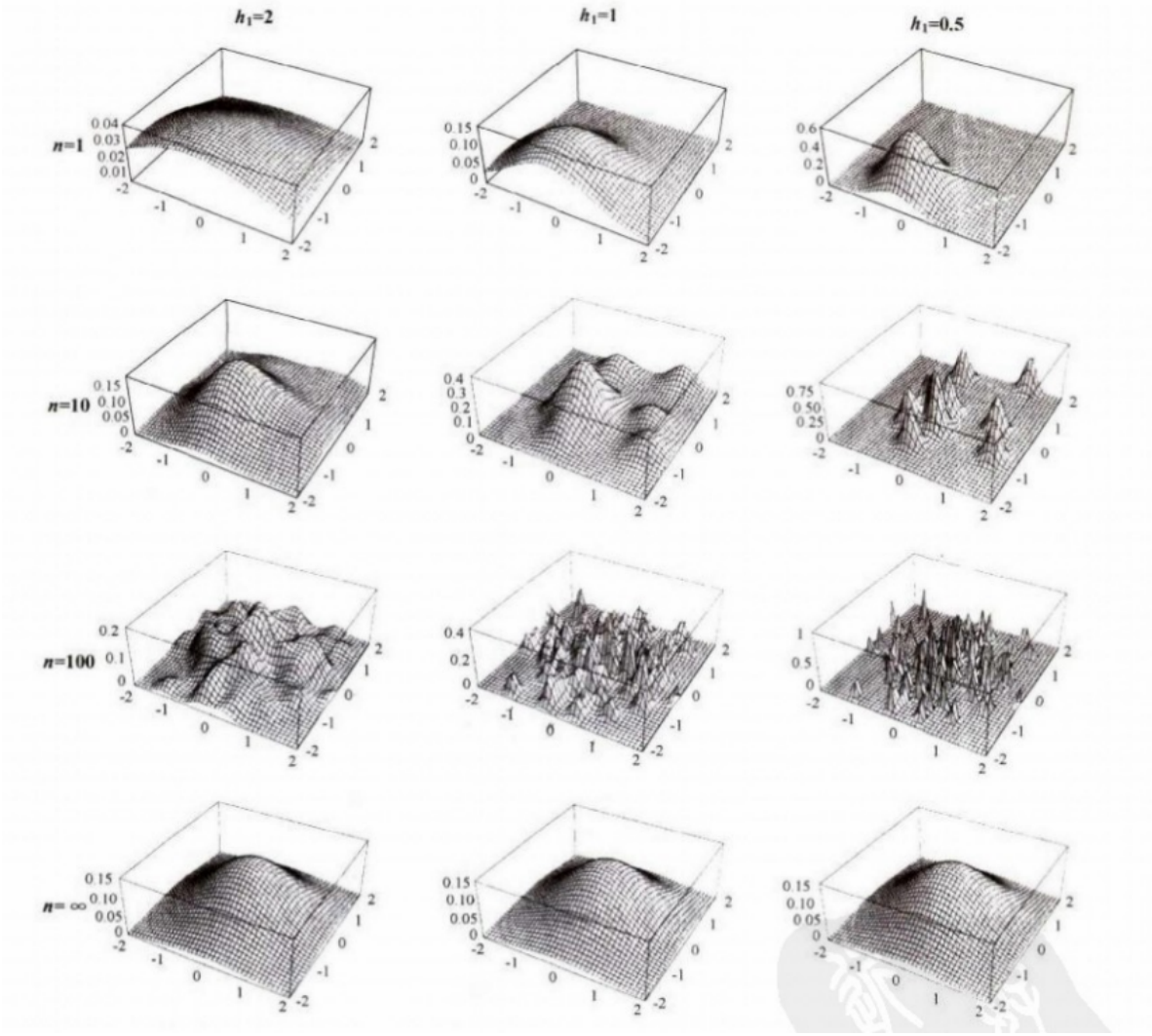
$$\varphi(u)=\frac{1}{\sqrt{2\pi}}e^{-\mu^2/2}$$

然后，令$h_n=h_1/\sqrt{n}$其中h_1是可以随意选取的一个参数。$p_n(x)$就是各个以样本点为中心的正态概率密度函数的叠加：

$$p_n(x)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_n}\varphi(\frac{x-x_i}{h_n})$$



$$\lim_{n\to\infty}\bar{p}_n(x)=p(x)$$
$$\lim_{n\to\infty}\sigma_n^2(x)=0$$



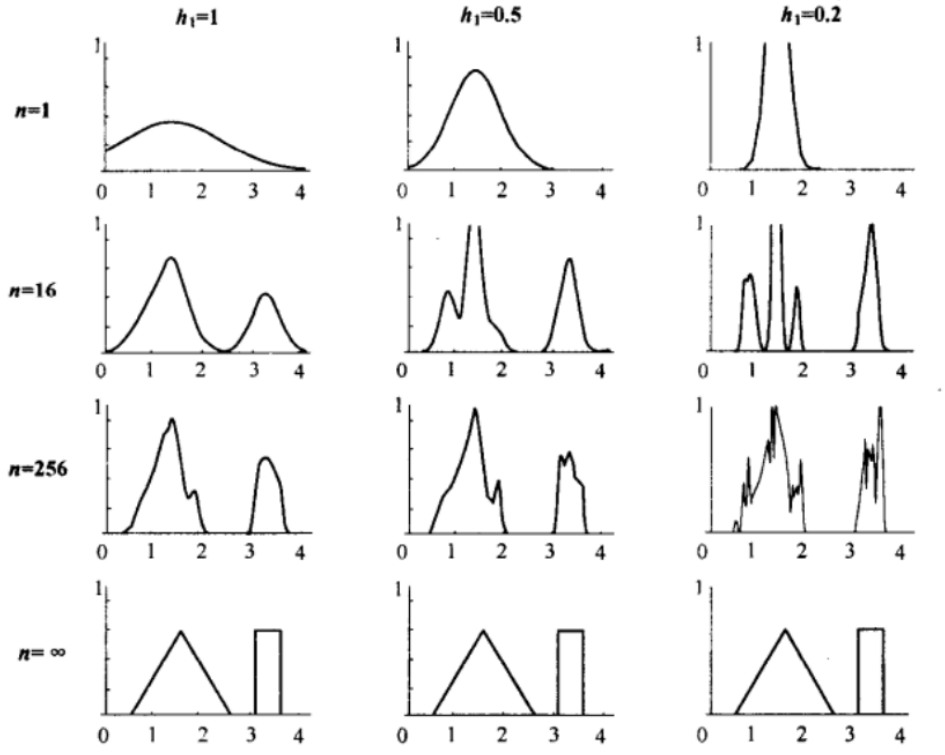


图 4-7　使用不同的窗宽度和样本数量对一个混合概率密度函数进行 Parzen 窗估计的结果。特别注意，当 $n=\infty$时，各种估计的结果都是相同的(等于真实的概率密度函数)，虽然窗宽度不同

Parzen用于分类问题

假如每一个分类都包含一组样本，我们可以利用parzen的公式求出每一个分类对应的概率密度函数

然后将测试数据代入到三个概率密度函数中，概率密度函数最大的那个类便是该数据所在的分类。

概率神经网络（PNN）

假设我们要实现Parzen估计，共有n个d维样本，都是随机的从c个类别中选取的。

输入层由d个输入单元组成，每一个输入单元都与n个模式单元相连。而每一个模式单元都与c个类别中的其中之一相连（不是全连接）。

模式层相当于每个点估计概率密度，然后对应的类别和类别层结点连接，同一类的样本点的概率密度函数求和可以得到该类别对应的概率密度函数。

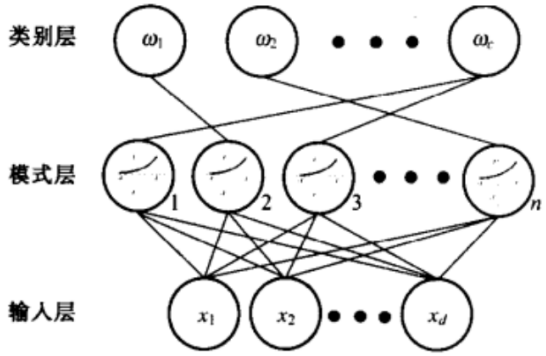


图 4-9　一个概率神经网络(PNN 网络)的结构。其中有 d 个输入层单元，n 个模式层单元，c 个类别层单元。每一个模式层单元能够对它的权重向量和归一化的样本向量 $\mathbf{x}$ 作内积，得到 $z=\mathbf{w}^t\mathbf{x}$，然后映射为 $\exp[(z-1)/\sigma^2]$。每一个类别单元把与它相连的模式层单元的输出结果相加。这样的结果，就保证了类别单元处得到的就是使用协方差为 $\sigma^2\mathbf{I}$ 的圆周对称高斯窗函数的 Parzen 窗的估计结果。其中 $\mathbf{I}$ 为 $d\times d$ 的单位矩阵

- 样本需要归一化（统一为单位长度），即$\sum_{i=1}^{d} x_i^2 = 1$
- 输入层与模式层完全连通
- 模式层的神经元个数是训练样本总数，每个训练样本和一个类别关联
- 模式层与类别层的连通与**训练的样本类别有关**
- w_k就是模式层每个单元对应的中心$w_k = x_k$，也就是一个样本。它和输入层输入的x做差，输出一个标量，这个标量可以带入高斯核函数计算输入x给这个中心点带来的概率密度

  第一层为输入层，用于接收来自训练样本的值，将数据传递给隐含层，神经元个数与输入向量长度相等。第二层隐含层是径向基层，每一个隐含层的神经元节点拥有一个中心，该层接收输入层的样本输入，计算输入向量与中心的距离，最后返回一个标量值，神经元个数与输入训练样本个数相同。向量 $\boldsymbol{x}$ 输入到隐含层，隐含层中第 i 类模式的第 j 神经元所确定的输入/输出关系由下式定义：

$$\Phi_{ij}(x)=\frac{1}{(2\pi)^{\frac{1}{2}}\sigma^d}\mathrm{e}^{-\frac{(x-x_{ij})(x-x_{ij})^{\mathrm{T}}}{\sigma^2}}$$

- 训练完后对数据进行分类，设模式层与类别层的激活函数为：

$$net_k = w_k^t x$$

考虑中心在某一个样本w_k处**未经归一化**的高斯窗函数。

$$\begin{aligned}\varphi(\frac{x-w_k}{h_n}) &\propto e^{-(x-w_k)^T(x-w_k)/2\sigma^2} \\ &= e^{-(x^Tx+w_k^tw_k-2x^tw_k)/2\sigma^2} \\ &= e^{(net_k-1)/\sigma^2}\end{aligned}$$

这个式子和上面的式子是一样的，不过它进行了一个推理，使用了归一化条件$x^tx=w_k^tw_k=1$

σ是用户设置的一个参数，它决定了高斯分布的宽度，可看作高斯窗的宽度。

PNN训练

initialize $j \leftarrow 0, n, a_{ji} \leftarrow 0$ for $j=1,\cdots,n; i=1,\cdots,c$

do $j \leftarrow j+1$

$x_{jk} \leftarrow x_{jk} / \left(\sum_{i=1}^{d} x_{ji}^2\right)^{1/2}$

$w_{jk} \leftarrow x_{jk}$

if $\mathbf{x}_j \in \omega_i$ then $a_{ji} \leftarrow 1$

until $j=n$

end

PNN分类算法

initialize $k \leftarrow 0, \mathbf{x} \leftarrow$ test pattern, $g_i \leftarrow 0$

do $k \leftarrow k+1$

$net_k \leftarrow \mathbf{w}_k^t\mathbf{x}$

if $a_{ki}=1$ then $g_i \leftarrow g_i + \exp[(net_k-1)/\sigma^2]$

until $k=n$

return $class \leftarrow \arg\max_i g_i(\mathbf{x})$

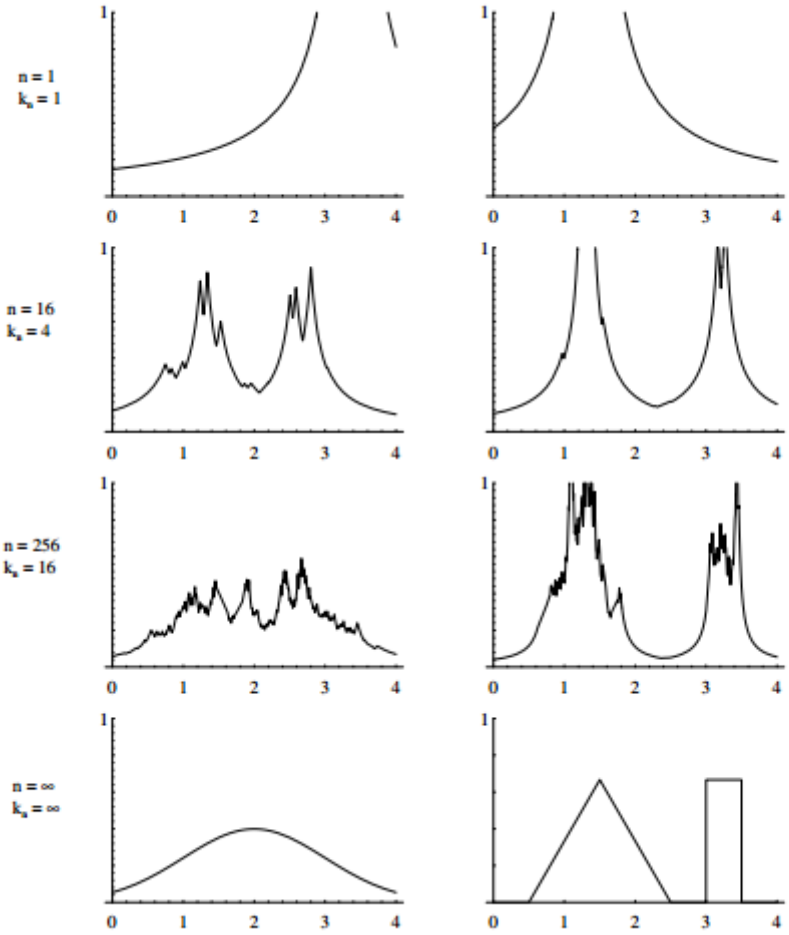
end

PNN算法适用于计算速度要求高且存储器资源比较容易满足的情况。新的训练样本很容易加入以前训练好的分类器中。其实就相当于并行运算不同点的概率分布。

K nearest neighbor

收敛的充要条件

- $p(x)$ 在所有点连续
- $\lim\limits_{n\to\infty} k_n = \infty$
- $\lim\limits_{n\to\infty} k_n/n = 0$



最邻近规则

最近邻规则的分类方法就是把点x分为x'所属的类别。

在无限训练样本的情况下，这个误差率至多不会超过贝叶斯误差率的两倍。

感性认识

最近邻标记θ'是一个随机变量，$\theta = \omega_i$的概率就是后验概率$P(\omega_i|x')$。不过这个后验概率一开始不一定准确，不过在样本个数非常多的时候，我们有理由认为x'和x足够近，使得$p(\omega_i|x) \approx p(\omega_i|x')$。

个人理解：对点X'的不同类别的概率可以离x'最近的某一个类别的点的距离来衡量。所以会有一个概率分布$P(\omega_i|x)$我们想要找到让这个概率最大化的类别$P(\omega_m|x) = (P(\omega_i|x)$，当$P(\omega_m|x) = 1$的时候，最近邻规则的误差率很低，和贝叶斯估计的结果相同。

证明

定义n个样本时的平均误差率

$$P_n(e) = \int P_n(e|x)p(x)dx \qquad (1)$$

利用贝叶斯决策论，记最小误差值为

$$P^*(e|x) = 1 - P(w_m|x)$$

则最小平均误差率为

$$P^* = \int P^*(e|x)p(x)dx$$

考虑

$$P_n(e|x) = \int P_n(e|x,x')p_n(x'|x)dx' \tag{2}$$

得到概率密度函数$p_n(x'|x)$是十分困难的，但是若

$$p_n(x'|x) \to \delta(x'-x),\ \ n \to \infty \tag{3}$$

那么问题就能得到简化.

假设给定的x点，$p(\cdot)$是连续的且值非零

任何样本落在x为中心的超球体S的概率为

$$P_S = \int_{x' \in S} p(x')dx'$$

所有n个样本都落在球外的概率为$(1-P_S)^n \to 0, n \to \infty$。故（3）式成立。

现在考虑（2）式中的$P(e|x,x')$，由于选取x'_n和选取x是独立的，有

$$P(\theta, \theta'_n|x,x'_n) = P(\theta|x)P(\theta'_n|x'_n)$$

其中θ_j是c种类别$w_1, w_2, \dots, w_c$中的一种。

$$\begin{aligned} P_n(e|x,x'_n) &= 1 - \sum_{i=1}^{c} P(\theta = w_i, \theta'_n = w_i|x,x'_n) \\ &= 1 - \sum_{i=1}^{c} P(w_i|x)P(w_i|x'_n) \end{aligned}$$

（2）式可写为

$$\begin{aligned} \lim_{n\to\infty} P_n(e|x) &= \lim_{n\to\infty} \int [1 - \sum_{i=1}^{c} P(w_i|x)P(w_i|x'_n)]\delta(x'_n - x)dx'_n \\ &= 1 - \sum_{i=1}^{c} P^2(w_i|x)x \end{aligned}$$

（1）式可写为

$$\begin{aligned} P &= \lim_{n\to\infty} P_n(e) \\ &= \lim_{n\to\infty} \int P_n(e|x)p(x)dx \\ &= \int [1 - \sum_{i=1}^{c} P^2(w_i|x)]p(x)dx \end{aligned} \tag{4}$$

由于

$$\sum_{i=1}^{c} P^2(w_i|x) = P^2(w_m|x) + \sum_{i \neq m} P^2(w_i|x)$$

根据贝叶斯决策论

$$P(w_i|x) = \begin{cases} \frac{P^*(e|x)}{c-1}, & i \neq m \\ 1 - P^*(e|x), & i = m \end{cases}$$

取得$P(w_i|x)$的最小值。所以有

$$\sum_{i=1}^{c} P^2(w_i|x) \geq (1 - P^*(e|x))^2 + \frac{P^*(e|x)}{c-1}$$

$$1 - \sum_{i=1}^{c} P^2(w_i|x) \leq 2P^*(e|x) - \frac{c}{c-1} P^{*2}(e|x)$$

代入（4）式，得证。

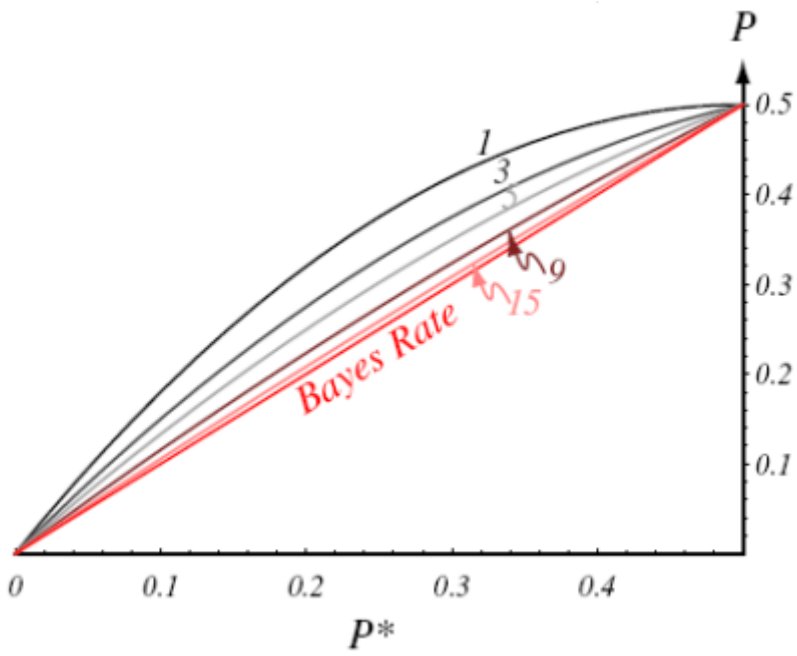
利用方差，我们可以得到一个更加紧致的上界

$$\begin{aligned} Var[P^*(e|x)] &= \int [P^*(e|x) - P^*]^2 p(x) dx \\ &= \int P^{*2}(e|x) p(x) dx - P^{*2} \geq 0 \end{aligned}$$

当且仅当$P^*(e|x) = 0$时，等式成立。代入（4）式，有

$$P \leq 2 \int P^*(e|x) p(x) dx - \frac{c}{c-1} P^{*2} = P^*(2 - \frac{c}{c-1} P^*)$$

K 近邻



注意，只有当n趋近于无穷大时，我们才能保证K近邻规则几乎是最优的分类规则。

降低时间复杂度

- 计算部分距离
- 构造kd树*（不保证找到最近邻）

- 剪枝（删去无用的点，无法再增加训练样本）

距离度量和最近邻分类

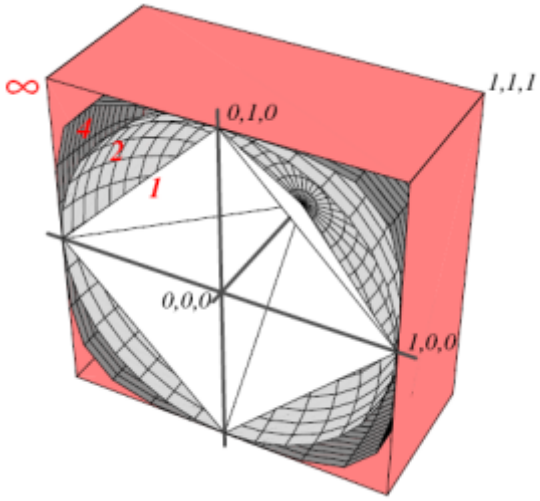
在设计最近邻分类器的时候，需要衡量模式之间距离的度量函数。目前只使用了d维空间中的欧几里得距离。但距离的定义很广泛，下面就会讨论各种可能距离。

- 非负性：$D(a,b) \geq 0$
- 自反性：$D(a,b) = 0$ iff $a = b$
- 对称性：$D(a,b) = D(b,a)$
- 三角不等式：$D(a,b) + D(b,c) \geq D(a,c)$

Minkowski距离度量

$$L_k(a,b) = (\sum_{i=1}^{d} |a_i - b_i|^k)^{1/k}$$

k越大，表达能力越强。



集合度量

$$D(S_1, S_2) = \frac{n_1 + n_2 - 2n_{12}}{n_1 + n_2 - n_{12}}$$

切空间

欧几里得距离没有平移不变性，如下面例子：

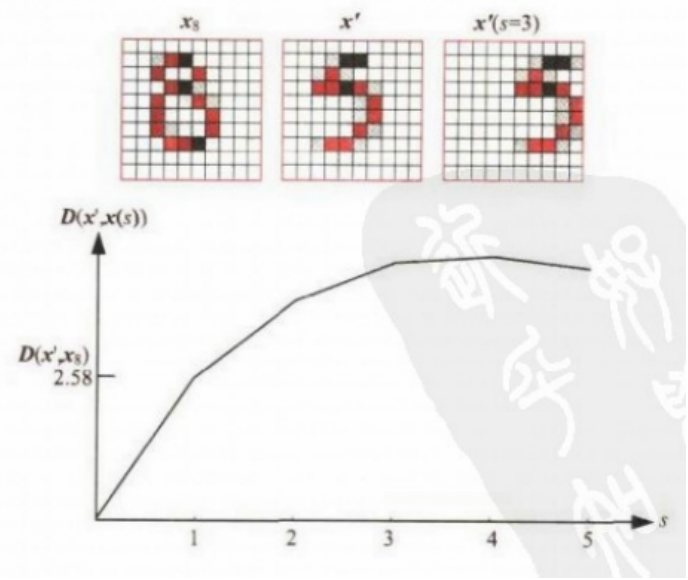


图 4-20　因为忽略平移不变性问题而不加分辨地使用欧几里德距离有时候会带来严重的误差。上图中的模式 $\mathbf{x}'$ 代表一个手写体字符“5”，而 $\mathbf{x}'(s=3)$ 代表同一个形状，但是经过了向右的 3 个像素的平移。这样，欧几里德距离度量的结果 $D(\mathbf{x}', \mathbf{x}'(s=3))$ 要比 $D(\mathbf{x}', \mathbf{x}_8)$ 大得多，其中的 $\mathbf{x}_8$ 表示一个手写体字符“8”。这样，使用欧几里德距离度量的最近邻规则分类器就会导致很大的分类误差。所以，为了解决这个问题，我们必须寻找一个对一些已知的变换（比如平移，旋转，尺度变换等）不敏感的距离度量

对每一个原型样本点x'，进行每一种变换操作。