Chapter 4

Nonparametric Techniques

Bayes Theorem for Classification

$$P(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)\cdot P(\omega_j)}{p(\mathbf{x})} \quad (1 \leq j \leq c)$$ (Bayes Formula)

To compute posterior probability $P(\omega_j|\mathbf{x})$, we need to know:

Prior probability: $P(\omega_j)$ Likelihood: $p(\mathbf{x}|\omega_j)$

- **Case I:** $p(\mathbf{x}|\omega_j)$ has certain **parametric form** $p(\mathbf{x}|\omega_j, \boldsymbol{\theta}_j)$

Maximum-Likelihood (ML) Estimation

Bayesian Parameter Estimation

Bayes Theorem for Classification (Cont.)

Potential problems for Case I

The assumed parametric form may not fit the ground-truth density encountered in practice, e.g.:

Assumed parametric form: Unimodal (单峰, such as Gaussian pdf)

Ground-truth form: Multimodal (多峰)

- **Case II:** $p(\mathbf{x}|\omega_j)$ doesn't have **parametric form**

Let the data speak for themselves! → **Parzen Windows**

k_n-nearest-neighbor

Density Estimation

General settings

Feature space: $\mathcal{F} = \mathbf{R}^d$

Feature vector: $\mathbf{x} \in \mathcal{F}$

pdf function: $\mathbf{x} \sim p(\cdot)$

How to estimate $p(\mathbf{x})$ from the training examples?

Fundamental fact

The probability of a vector **x** **falling into a region** $\mathcal{R} \subset \mathcal{F}$:

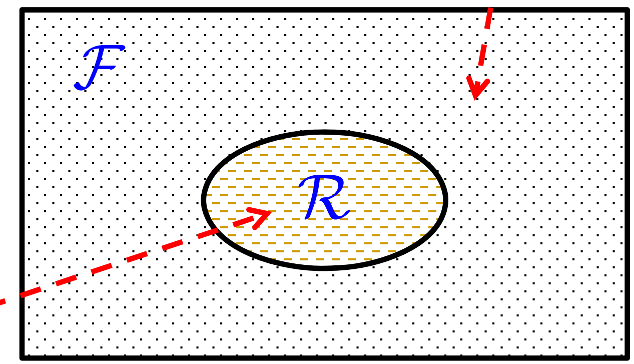
$$P = \Pr[\mathbf{x} \in \mathcal{R}] = \int_{\mathcal{R}} p(\mathbf{x}')\, d\mathbf{x}'$$

A smoothed/averaged version of $p(\mathbf{x})$

Density Estimation (Cont.)

$$P = \Pr[\mathbf{x} \in \mathcal{R}] = \int_{\mathcal{R}} p(\mathbf{x}')\, d\mathbf{x}'$$

$\Pr[\mathbf{x} \notin \mathcal{R}] = 1 - P$



$\Pr[\mathbf{x} \in \mathcal{R}] = P$

Given n examples (*i.i.d.*) $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ with $\mathbf{x}_i \sim p(\cdot)\ (1 \leq i \leq n)$

Let X be the (discrete) **random variable** representing the number of examples falling into $\mathcal{R}$

X will take Binomial distribution (二项分布): $X \sim \mathcal{B}(n, P)$

Density Estimation (Cont.)

$$\Pr[X = r] = \binom{n}{r} P^r (1-P)^{n-r} \quad (0 \le r \le n)$$

$X \sim \mathcal{B}(n, P)$

$\mathcal{E}[X] = nP$ **Table 3.1** [pp.109]

$$\boxed{P = \frac{\mathcal{E}[X]}{n}}$$

Assume $\mathcal{R}$ is small

$p(\cdot)$ hardly varies within $\mathcal{R}$

$$P = \Pr[\mathbf{x} \in \mathcal{R}] = \int_{\mathcal{R}} p(\mathbf{x}')\, d\mathbf{x}'$$

$$\simeq p(\mathbf{x}) \int_{\mathcal{R}} 1\, d\mathbf{x}'$$ ($\mathbf{x}$ is a point within $\mathcal{R}$)

$$\boxed{P \simeq p(\mathbf{x})\, V}$$ (V is the volume enclosed by $\mathcal{R}$)

Density Estimation (Cont.)

$$P = \frac{\mathcal{E}[X]}{n}$$

$$P \simeq p(\mathbf{x})\,V$$

$$p(\mathbf{x}) = \frac{\mathcal{E}[X]/n}{V}$$

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

$X \sim \mathcal{B}(n, P)$

X **peaks sharply** about $\mathcal{E}[X]$ when n is large enough

Let k be the **actual value of X** after observing the *i.i.d.* training examples $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$

$$k \simeq \mathcal{E}[X]$$

Density Estimation (Cont.)

To show the explicit relationships with n:

$\mathcal{R}$ ⇨ $\mathcal{R}_n$ (containing $\mathbf{x}$)

$$p(\mathbf{x}) = \frac{k/n}{V} \quad \Rightarrow \quad p_n(\mathbf{x}) = \frac{k_n/n}{V_n}$$

Quantities:

V_n: volume of $\mathcal{R}_n$

n: # training examples

k_n: # training examples falling within $\mathcal{R}_n$

Fix V_n and determine k_n ⇨ Parzen Windows

Fix k_n and determine V_n ⇨ k_n-nearest-neighbor

Parzen Windows

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

Fix V_n, and then determine k_n

Assume $\mathcal{R}_n$ is a d-dimensional hypercube (超立方体)

The length of each edge is h_n

$$V_n = h_n^d$$

Determine k_n with window function (窗口函数), a.k.a. kernel function (核函数), potential function (势函数), etc.

Emanuel Parzen (1929-)

Parzen Windows (Cont.)

Window function: $\varphi(\mathbf{u}) = \begin{cases} 1 & |u_j| \leq 1/2; \quad j = 1, \ldots, d \\ 0 & \text{otherwise} \end{cases}$

$\varphi(\mathbf{u})$ defines a **unit hypercube** centered at the origin

$\varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right) = 1 \iff$ $\mathbf{x}_i$ falls within the hypercube of volume V_n centered at $\mathbf{x}$

$$k_n = \sum_{i=1}^{n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right)$$

Parzen Windows (Cont.)

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

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right)$$

$$k_n = \sum_{i=1}^{n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right)$$

An average of functions of $\mathbf{x}$ and $\mathbf{x}_i$

$\varphi(\cdot)$ is not limited to be the hypercube window function of Eq.9 [pp.164]

$\varphi(\cdot)$ could be any pdf function:

$$\varphi(\mathbf{u}) \geq 0$$

$$\int \varphi(\mathbf{u})\, d\mathbf{u} = 1$$

Parzen Windows (Cont.)

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right) \qquad (V_n = h_n^d)$$

$\varphi(\cdot)$ being a pdf function $\Rightarrow$ $p_n(\cdot)$ being a pdf function

$$\int p_n(\mathbf{x})d\mathbf{x} = \frac{1}{nV_n}\sum_{i=1}^{n}\int \varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right)d\mathbf{x}$$

Integration by substitution (换元积分)

Let $\mathbf{u} = (\mathbf{x}-\mathbf{x}_i)/h_n$

$$= \frac{1}{nV_n}\sum_{i=1}^{n}\int h_n^d\,\varphi(\mathbf{u})\,d(\mathbf{u}) = \frac{1}{n}\sum_{i=1}^{n}\int \varphi(\mathbf{u})\,d(\mathbf{u}) = 1$$

window function (being pdf) $\varphi(\cdot)$ + window width h_n + training data $\mathbf{x}_i$ $\Rightarrow$ Parzen pdf $p_n(\cdot)$

Parzen Windows (Cont.)

Parzen pdf: $$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right) \quad (V_n = h_n^d)$$

$\varphi(\cdot)$ being a pdf function $\Rightarrow$ $p_n(\cdot)$ being a pdf function

$$\delta_n(\mathbf{x}) = \frac{1}{V_n}\varphi\left(\frac{\mathbf{x}}{h_n}\right) \Rightarrow p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\delta_n(\mathbf{x}-\mathbf{x}_i)$$

What is the effect of h_n ("window width") on the Parzen pdf?

- $p_n(\mathbf{x})$: **superposition** (叠加) of n interpolations (插值)
- $\mathbf{x}_i$: contributes to $p_n(\mathbf{x})$ based on its "**distance**" from $\mathbf{x}$ (i.e. "$\mathbf{x}$-$\mathbf{x}_i$")

Parzen Windows (Cont.)

The effect of h_n ("window width")

$$\delta_n(\mathbf{x}) = \frac{1}{V_n}\varphi\left(\frac{\mathbf{x}}{h_n}\right) = \underbrace{\frac{1}{h_n^d}}_{\text{Affects the amplitude}}\varphi\underbrace{\left(\frac{\mathbf{x}}{h_n}\right)}_{\text{Affects the width}}$$

Affects the *amplitude* (vertical scale, 幅度)

Affects the *width* (horizontal scale, 宽度)

What do "amplitude" and "width" mean for a function?

For $\varphi(\mathbf{u})$:

$|\varphi(\mathbf{u})| \leq a$ *(amplitude)* ⇒ $|\delta_n(\mathbf{x})| \leq \left(1/h_n^d\right) \cdot a$

$|u_j| \leq b_j$ *(width)* $(j = 1, \ldots, d)$ ⇒ $|x_j| \leq h_n \cdot b_j \;\; (j = 1, \ldots, d)$

For $\delta_n(\mathbf{x})$:

Parzen Windows (Cont.)

$$\delta_n(\mathbf{x}) = \frac{1}{V_n}\varphi\left(\frac{\mathbf{x}}{h_n}\right) = \frac{1}{h_n^d}\varphi\left(\frac{\mathbf{x}}{h_n}\right)$$

$\delta_n(\cdot)$ being a pdf function

Integration by substitution

Let $\mathbf{u} = \mathbf{x}/h_n$

$$\int \delta_n(\mathbf{x})\,d\mathbf{x} = \int \frac{1}{h_n^d}\varphi\left(\frac{\mathbf{x}}{h_n}\right)\,d\mathbf{x}$$

$$= \int \frac{1}{h_n^d}\cdot\varphi\left(\mathbf{u}\right)\cdot h_n^d\;d\mathbf{u} = \int \varphi\left(\mathbf{u}\right)\,d\mathbf{u} = 1$$

Case I

$h_n \uparrow$

Function amplitude (vertical scale) $\downarrow$

Function width (horizontal scale) $\uparrow$

Case II

$h_n \downarrow$

Function amplitude (vertical scale) $\uparrow$

Function width (horizontal scale) $\downarrow$

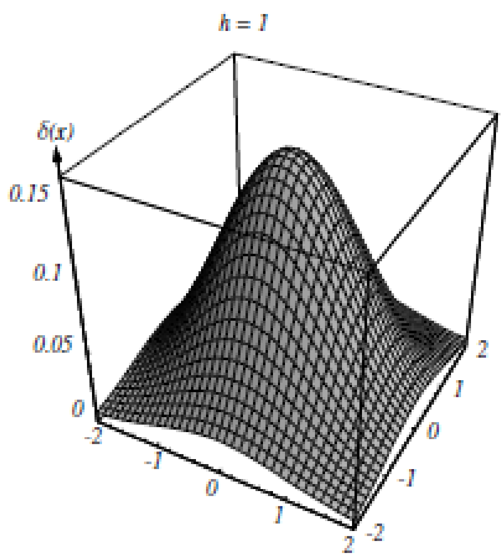
Parzen Windows (Cont.)

$$\delta_n(\mathbf{x}) = \frac{1}{h_n^d}\varphi\left(\frac{\mathbf{x}}{h_n}\right)$$

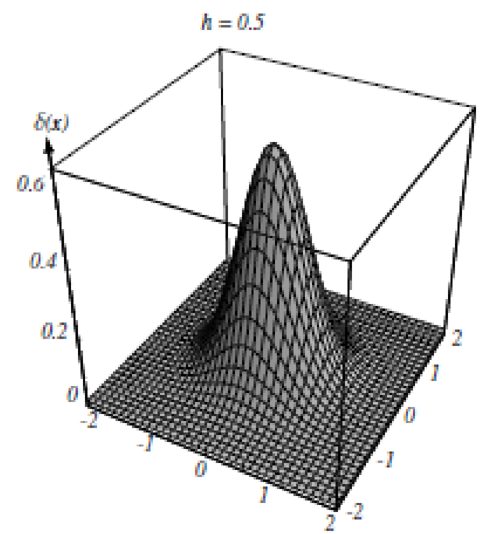
Suppose $\varphi(\cdot)$ being a 2-d

Gaussian pdf

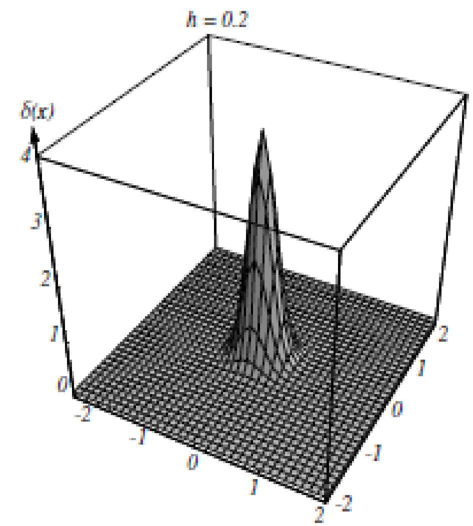
The shape of $\delta_n(\mathbf{x})$ with decreasing values of h_n



h=1.0



h=0.5



h=0.2

Parzen Windows (Cont.)

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\delta_n(\mathbf{x}-\mathbf{x}_i), \text{ where } \delta_n(\mathbf{x}) = \frac{1}{h_n^d}\varphi\left(\frac{\mathbf{x}}{h_n}\right)$$

- h_n very large ➔ $\delta_n(\mathbf{x})$ being *broad* with *small amplitude*

  $p_n(\mathbf{x})$ will be the superposition of n broad, slowly changing (慢变) functions, i.e. being *smooth* (平滑) with *low resolution* (低分辨率)

- h_n very small ➔ $\delta_n(\mathbf{x})$ being *sharp* with *large amplitude*

  $p_n(\mathbf{x})$ will be the superposition of n sharp pulses (尖脉冲), i.e. being *variable/unstable* (易变) with *high resolution* (高分辨率)

A compromised value (折衷值) of h_n should be sought for limited number of training examples

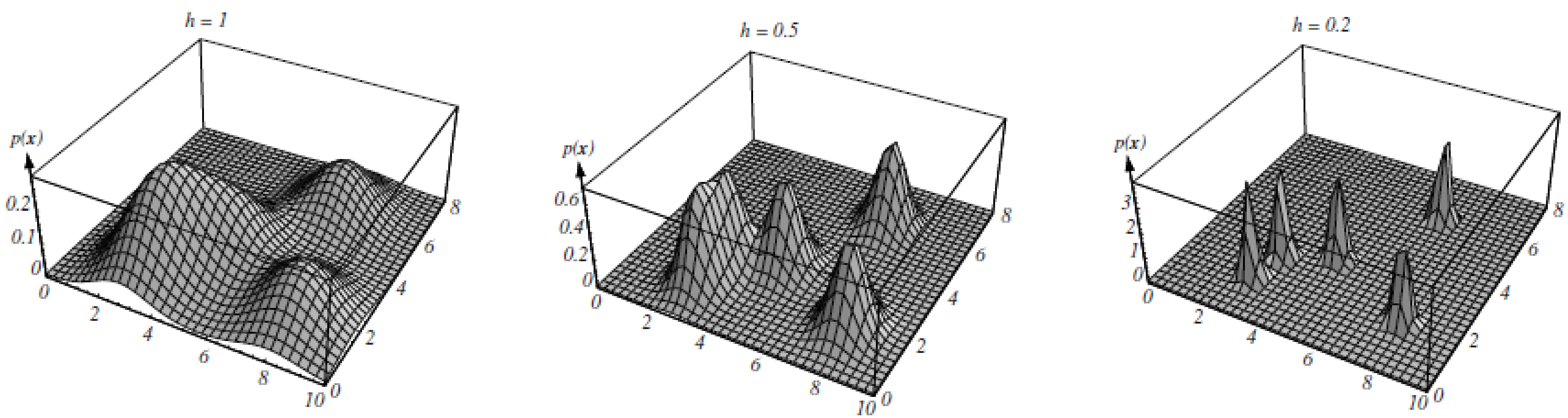
Parzen Windows (Cont.)

More illustrations:
Subsection 4.3.3 [pp.168]

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\delta_n(\mathbf{x}-\mathbf{x}_i), \text{ where } \delta_n(\mathbf{x}) = \frac{1}{h_n^d}\varphi\left(\frac{\mathbf{x}}{h_n}\right)$$

Suppose $\varphi(\cdot)$ being a 2-d *Gaussian pdf* and *n*=5

The shape of $p_n(\mathbf{x})$ with decreasing values of h_n



h=1.0 *h*=0.5 *h*=0.2

k_n-Neareast-Neighbor

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

Fix k_n, and then determine V_n

specify k_n ➔ center a cell about **x** ➔ grow the cell until capturing k_n *nearest examples* ➔ return cell volume as V_n

The principled rule to specify k_n [pp.175]

$$\lim_{n\to\infty} k_n = \infty$$

$$\lim_{n\to\infty} \frac{k_n}{n} = 0$$

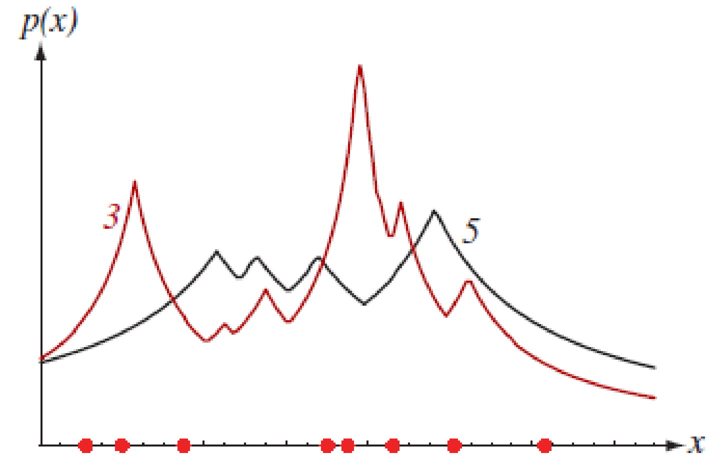
➔ A rule-of-thumb choice for k_n: $k_n = \sqrt{n}$

k_n-Neareast-Neighbor (Cont.)

Eight points in one dimension (n=8, d=1)

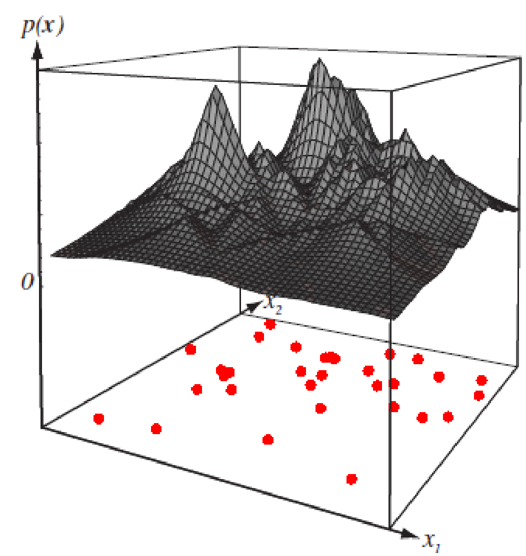
red curve: k_n=3

black curve: k_n=5



Thirty-one points in two dimensions (n=31, d=2)

black surface: k_n=5



Summary

- Basic settings for nonparametric techniques
  - Let the data speak for themselves
    - Parametric form not assumed for class-conditional pdf
  - Estimate class-conditional pdf from training examples

    ➔ Make predictions based on Bayes Formula
- Fundamental result in density estimation

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

n: # training examples

V_n: volume of region $\mathcal{R}_n$ containing $\mathbf{x}$

k_n: # training examples falling within $\mathcal{R}_n$

Summary (Cont.)

- Parzen Windows: Fix V_n ➔ Determine k_n
  - Effect of h_n (window width): A compromised value for a fixed number of training examples should be chosen

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right) \qquad (V_n = h_n^d)$$

$\varphi(\cdot)$ being a pdf function ➔ $p_n(\cdot)$ being a pdf function

window function (being pdf) $\varphi(\cdot)$ + window width h_n + training data $\mathbf{x}_i$ ➔ Parzen pdf $p_n(\cdot)$

Summary (Cont.)

- k_n-nearest-neighbor: Fix k_n ➔ Determine V_n

specify k_n ➔ center a cell about **x** ➔ grow the cell until capturing k_n *nearest examples* ➔ return cell volume as V_n

The principled rule to specify k_n [pp.175]

$$\lim_{n\to\infty} k_n = \infty$$

$$\lim_{n\to\infty} \frac{k_n}{n} = 0$$

A rule-of-thumb choice for k_n: $k_n = \sqrt{n}$