Supervised Learning:

Statistical Pattern Recognition

Classification: Approaches

Design of Classifier:

1- The simplest and the most intuitive approach to classifier design is based on the ***concept of similarity***: patterns that are similar should be assigned to the same class[one-nearest neighbor decision rule (1-NN)].

2- The second main is based on the ***probabilistic approach***. The optimal Bayes decision rule (with the 0/1 loss function) assigns a pattern to the class with the maximum posterior probability[k-nearest neighbor (k-NN) rule and the Parzen classifier].

3- The third category of classifiers is to ***construct decision boundaries directly by optimizing certain error criterion***. While this approach depends on the chosen metric, sometimes classifiers of this type may approximate the Bayes classifier asymptotically. The driving force of the training procedure is, however, the minimization of a criterion such as the apparent classification error or the mean squared error (MSE) between the classifier output and some preset target value [feed-forward neural networks also called Multi-Layer Perceptrons(MLPs)].

Classification Methods Based on Similarity

- **Once a good metric to define similarity, patterns can be classified by template matching or minimum distance classifier using a few prototypes per class.**

- **The choice of the metric and prototypes is crucial to the success of this approach.**

Classification Methods Based on Similarity

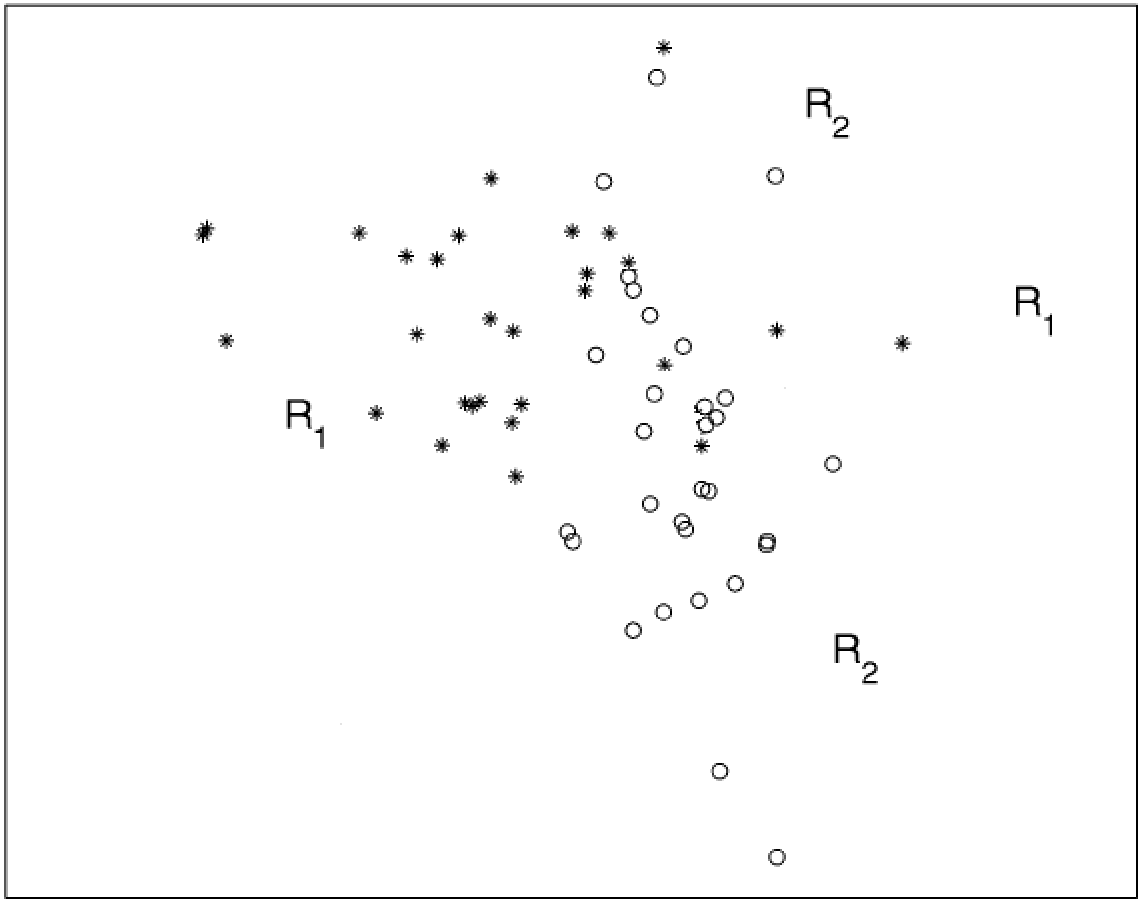
- Template Matching
  - Assign Pattern to the most similar template
- Nearest Mean Classifier
  - Assign Pattern to the nearest class mean
- Subspace Method
  - Assign Pattern to the nearest subspace (invariance)
- 1-Nearest Neighbor Rule
  - Assign Pattern to the class of the nearest training pattern

Homework #2, p.1

One-nearest neighbor decision rule (1-NN)

•The most straightforward 1-NN rule can be *conveniently used as a benchmark* for all the other classifiers since it appears to always provide a *reasonable classification performance in most applications*.

•Further, as the 1-NN classifier does not require any user-specified parameters (except perhaps the distance metric used to find the nearest neighbor, but *Euclidean distance* is commonly used), its classification results are implementation independent.



Bayes classification rule

- Statistical nature of feature vectors

$$\underline{x} = \left[x_1, x_2, \ldots, x_l \right]^T$$

- Assign the pattern represented by feature vector $\underline{x}$
  to the most probable of the available classes

$$\omega_1, \omega_2, \ldots, \omega_M$$

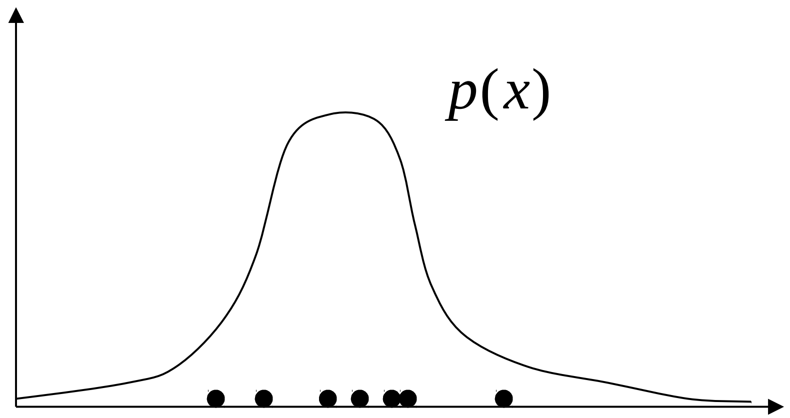
That is $\underline{x} \rightarrow \omega_i : P(\omega_i | \underline{x})$

maximum

Bayes classification rule: Probability Density Function(PDF)

- Treat patterns (feature vectors) as observations of random variable (vector).

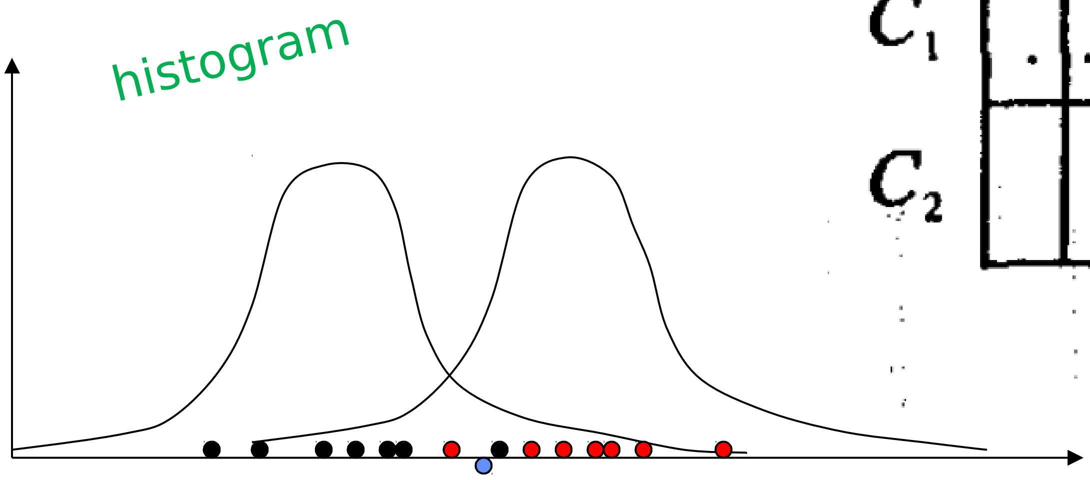
- Random variable is defined by the probability density function.

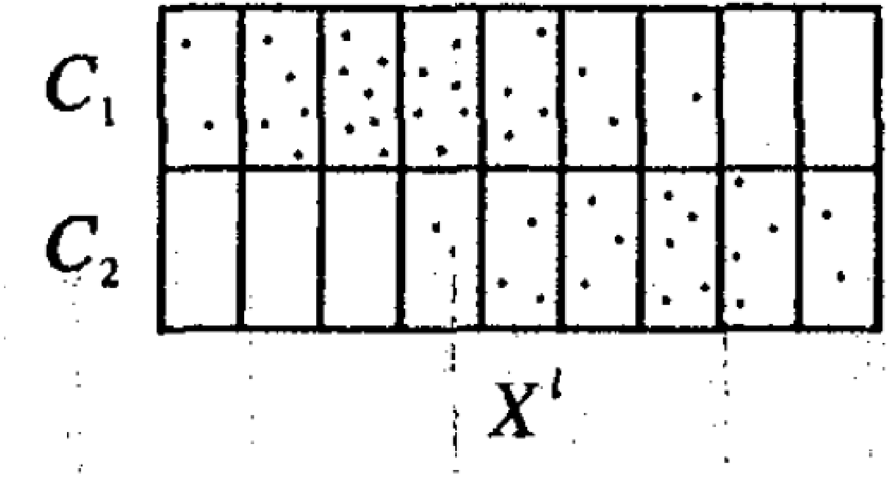


Probability density function of random variable and few observations.

Bayes classification rule

- **Suppose we have 2 classes and we know probability density functions of their feature vectors. How some new pattern should be classified?**





Bayes classification rule

- Bayes formula:

$$P(w_i \mid x) = \frac{p(x \mid w_i)P(w_i)}{p(x)}$$

Posterior = (Likelihood. Prior) / Evidence

There are four main concepts in this formula:

Prior Probability that is independent of the observations and our measurements. It is one of the characteristics of the problem and forces us before utilizing the available information.

Examples: The probability that a person is a woman is 50%.

The probability that a person has an academic degree is 30%.

Liklihood that is described by considering the distribution of the samples belong to one of the classes. In other word, it determines that the probability distribution of members of each class in different regions of the feature spaces.

Bayes classification rule

- Bayes formula: $$P(w_i \mid x) = \frac{p(x \mid w_i) P(w_i)}{p(x)}$$

Posterior = (Likelihood. Prior) / Evidence

Evidence that points out to the proof itself or the marginal probability that an observation is seen(which is almost 100% in our cases).

In a 2-class problem this evidence can be calculated using the below equation:

$$p(x) = \sum_{i=1}^{2} p(x \mid w_i) P(w_i)$$

Posterior Probability gives us the membership probability of a sample in of the classes after considering the evidence and employing them in the Bayes formula.

Above formula is a consequent of following probability theory equations:

$$P(A, B) = P(A \mid B) P(B) = P(B \mid A) P(A)$$

$$P(C) = P(C, A) + P(C, B), \text{ if } A \cap B = 0, A \cup B = 1$$

Bayes classification rule

- Bayes classification rule: classify x to the class w_i which has biggest posterior probability $P(w_i \mid x)$

$$P(w_1 \mid x) > P(w_2 \mid x) \ ? \quad w_1 \quad : \quad w_2$$

Using Bayes formula, we can rewrite classification rule:

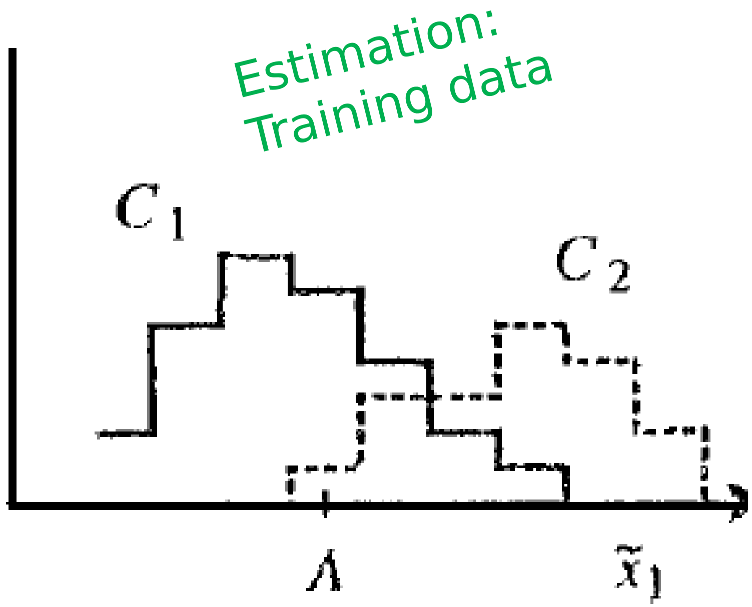
$$\frac{p(x \mid w_1)P(w_1)}{p(x)} > \frac{p(x \mid w_2)P(w_2)}{p(x)}$$

$$p(x \mid w_1)P(w_1) > p(x \mid w_2)P(w_2) \ ? \quad w_1 \quad : \quad w_2$$

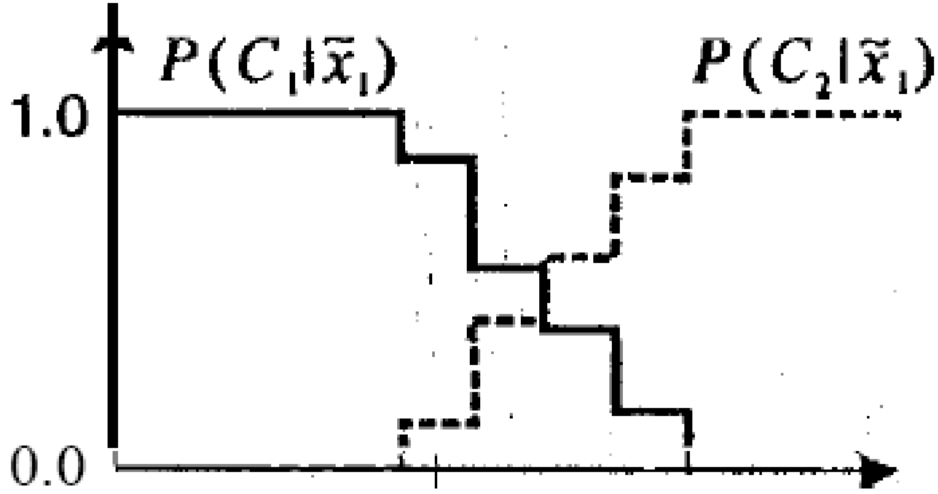
if $P(\omega_1 \mid x) > P(\omega_2 \mid x)$ $\Rightarrow$ True state of nature = ω_1

if $P(\omega_1 \mid x) < P(\omega_2 \mid x)$ $\Rightarrow$ True state of nature = ω_2

Bayes classification rule

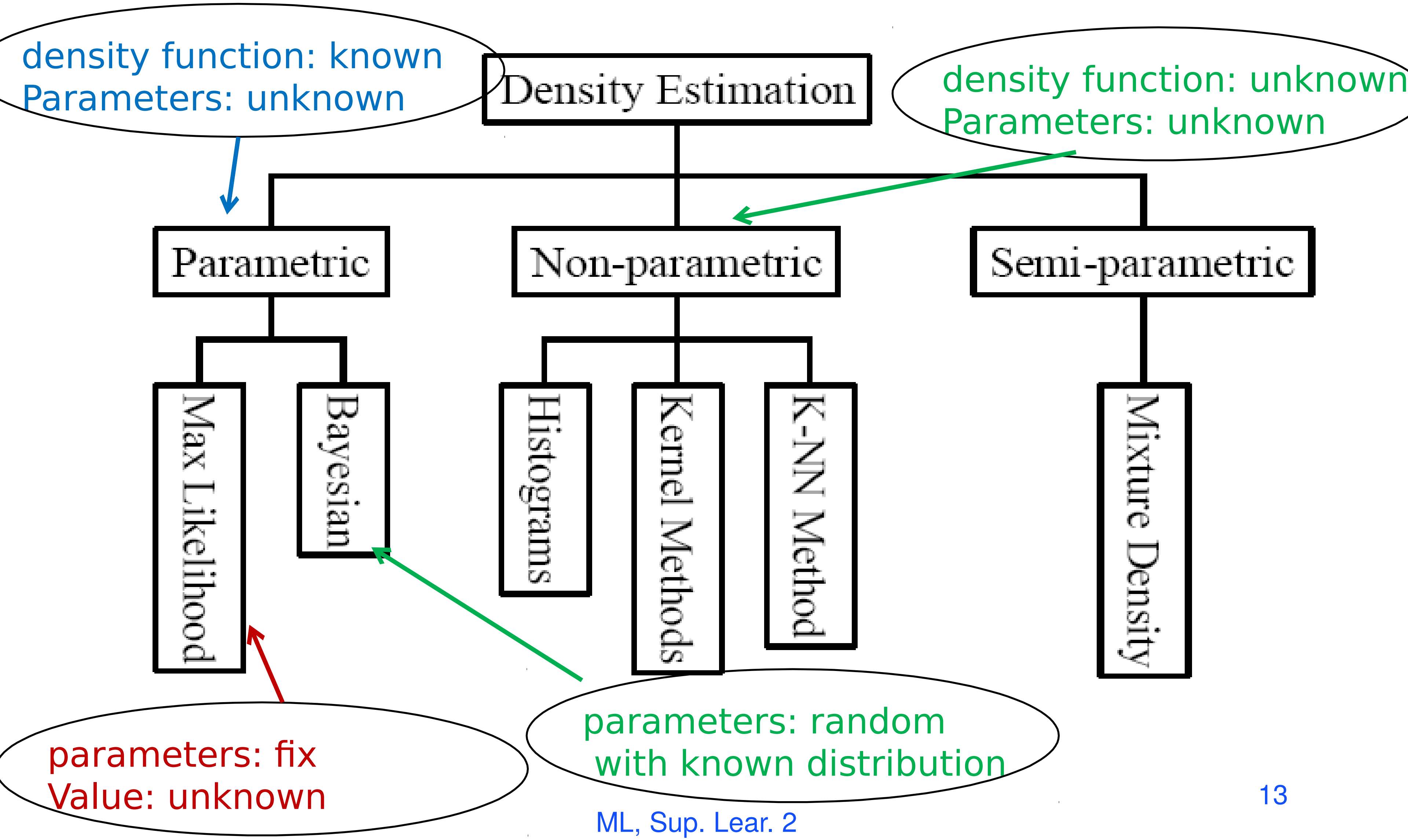


Histogram plot of feature variables



Histogram plot of posterior probabilities

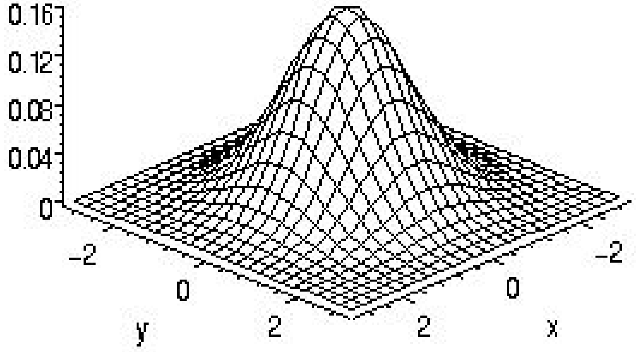
Probability Density Function(PDF)



Estimating probability density function.

- In applications, probability density function of class features is unknown.
- Solution: model unknown probability density function $p(x \mid w_i)$ of class w_i by some parametric function $p_i(x;\theta)$ and determine parameters based on training samples.

Example: model pdf as a Gaussian function with unitary covariance matrix and unknown mean



$$p(x;\mu) = \frac{1}{(2\pi)^{l/2}} e^{-\frac{1}{2}(x-\mu)^2}$$

Discriminate function(parametric classification)

Homework #2, p.2

$$g_i(x) = P(\omega_i \mid x)$$

$$= \mathrm{P}(\mathrm{x} \mid \omega_i)\, \mathrm{P}(\omega_i)$$

... or equivalently

$$= \log P(x \mid \omega_i) + \log P(\omega_i)$$

if we can assume that $P(x \mid \omega_i)$ *are Gaussian*

$$P(x \mid \omega_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[- \frac{(x - \mu_i)^2}{2\sigma_i^{\ 2}} \right]$$

$$g_i(x) = -\frac{1}{2}\log 2\pi - \log \sigma_i - \frac{(x - \mu_i)^2}{2\sigma_i^{\ 2}} + \log P(\omega_i)$$

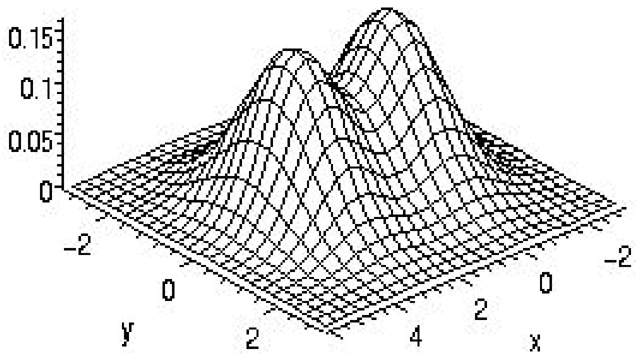
for multivariate:

$$P(x \mid \omega_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp\left[- \frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right]$$

d: input dimension

$$g_i(x) = -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i) + \log P(\omega_i)$$

Mixture of Gaussian functions



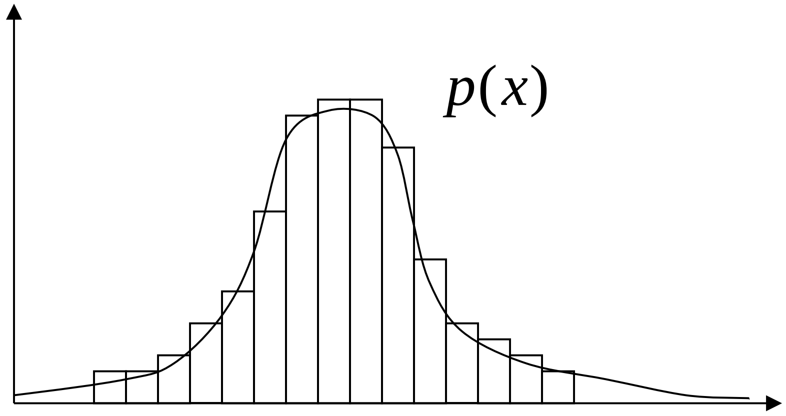
$$p(x;\mu) = \sum_{i=1}^{N} \frac{P_i}{(2\pi\sigma_i^2)^{l/2}} e^{-\frac{1}{2}\frac{(x-\mu_i)^2}{\sigma_i^2}}$$

- No direct computation of optimal values of parameters P_i, μ_i, σ_i is possible.
- Generic methods for finding extreme points of non-linear functions can be used: gradient descent, Newton's algorithm, Lagrange multipliers.
- Usually used: expectation-maximization (EM) algorithm.

Mixture of Experts(Ensemble learning)

Nonparametric pdf estimation

Histogram method:

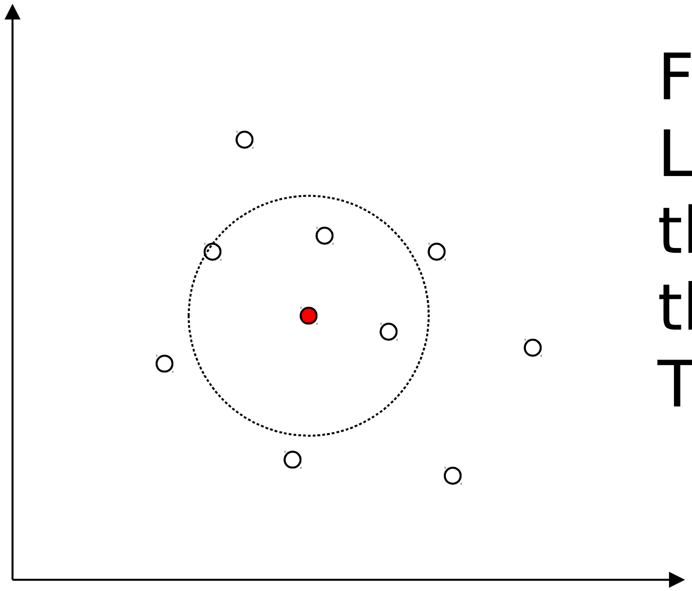


Split feature space into bins of width h.
Approximate p(x) by:

$$\hat{p}(x) = \frac{1}{h} \frac{\text{Number of training samples inside bin}}{\text{Total number of training samples}}$$

Nonparametric pdf estimation

K-Nearest Neighbor PDF Estimation:

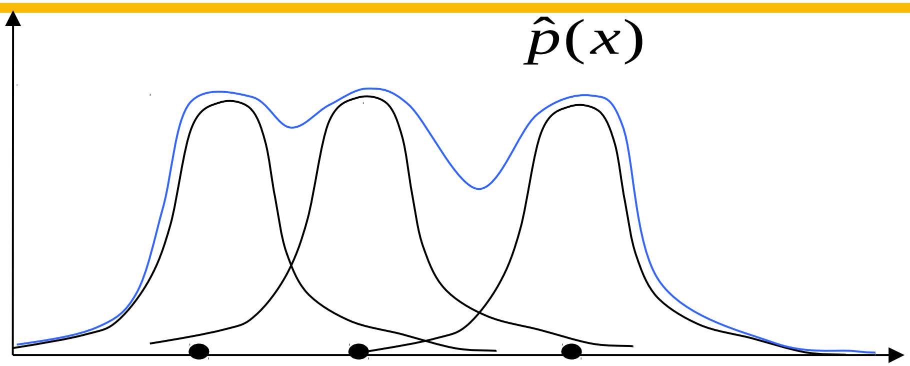


Find k nearest neighbors.
Let V be the volume of the sphere containing these k training samples.
Then approximate pdf by:

$$\hat{p}(x) = \frac{n}{K}$$

Nonparametric pdf estimation

Parzen windows:



Each training point contributes one Parzen kernel function to pdf construction:

$$\hat{p}(x) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{h}\varphi\left(\frac{x_i - x}{h}\right)\right)$$

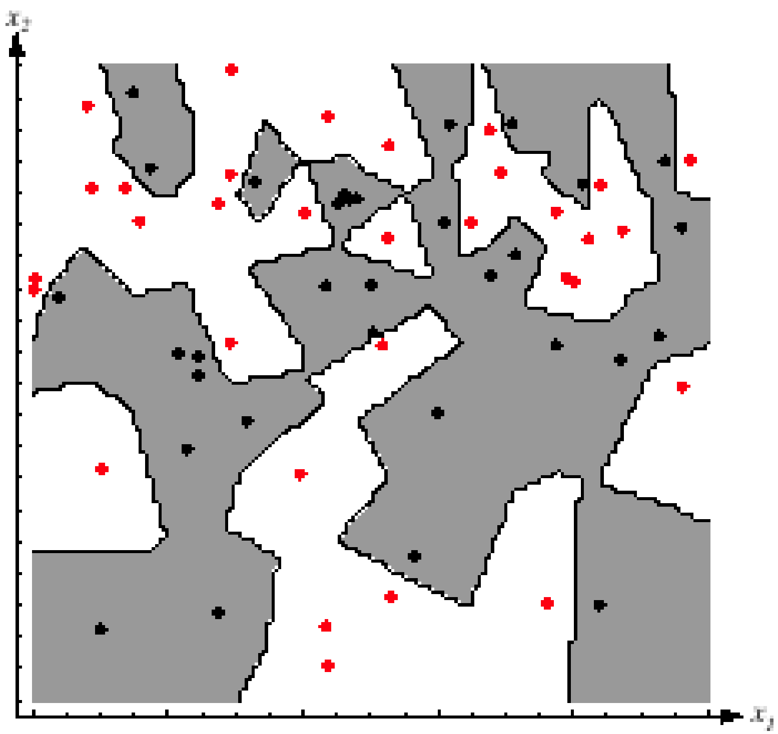
- *Important to choose proper h.*

- Take cluster centers as centers for Parzen kernel functions.

- Make contribution of the cluster proportional to the number of training samples cluster has.

$$\hat{p}(x) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{N_i}{h}\varphi\left(\frac{c_i - x}{h}\right)\right)$$

Nonparametric pdf estimation _ Parzen Window



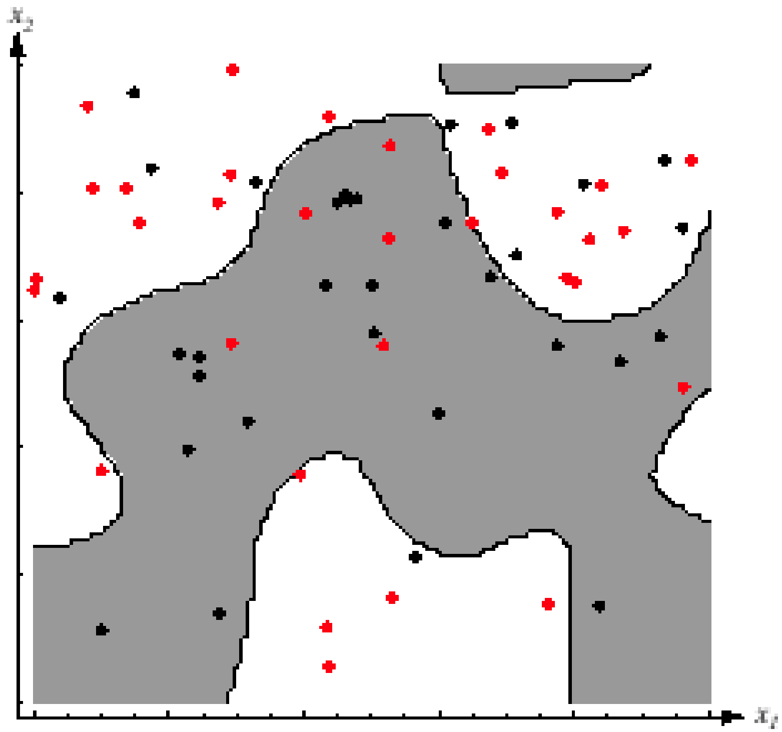


FIGURE 4.8. The decision boundaries in a two-dimensional Parzen-window dichotomizer depend on the window width h. At the left a small h leads to boundaries that are more complicated than for large h on same data set, shown at the right. Apparently, for these data a small h would be appropriate for the upper region, while a large h would be appropriate for the lower region; no single window width is ideal overall. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*.