UCA
Universidad de Cádiz

CHAPTER 4: CLASSIFICATION

Grado en Ingeniería Informática

Curso 2014 / 15

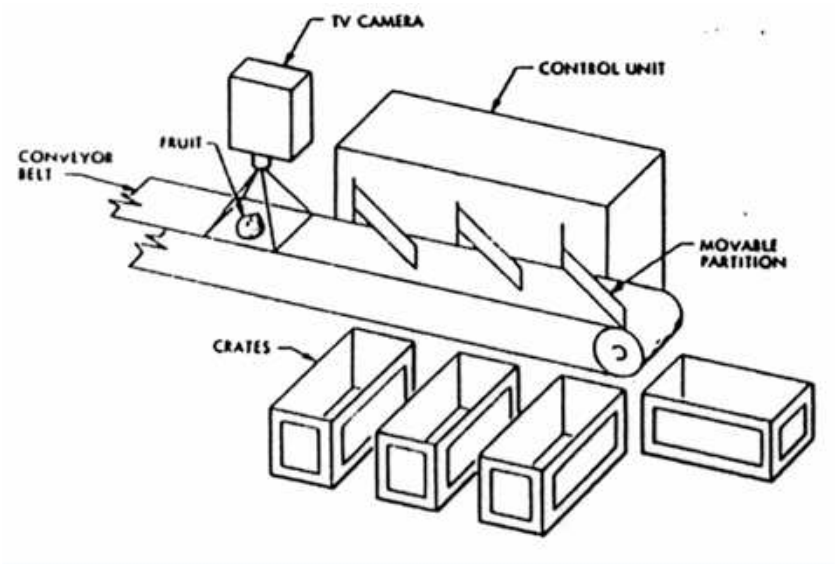
Topics

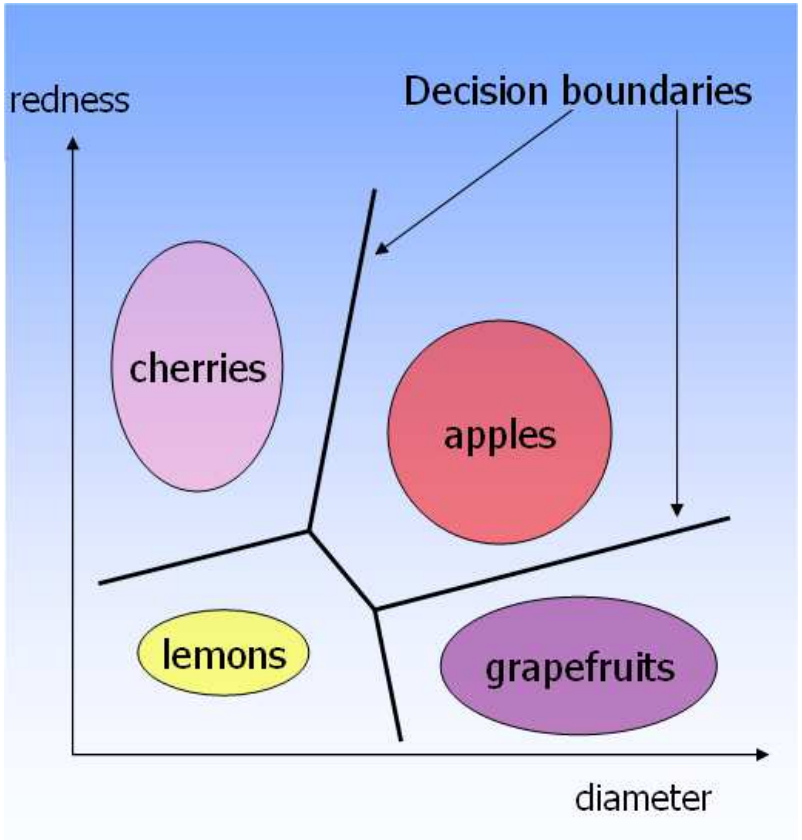
1. Introduction
2. Features
3. Classification
   a) Methods
   b) Prototype selection
   c) Distance selection
   d) Non parametric classification
4. Clustering

INTRODUCTION

Introduction

- Fruit recognition



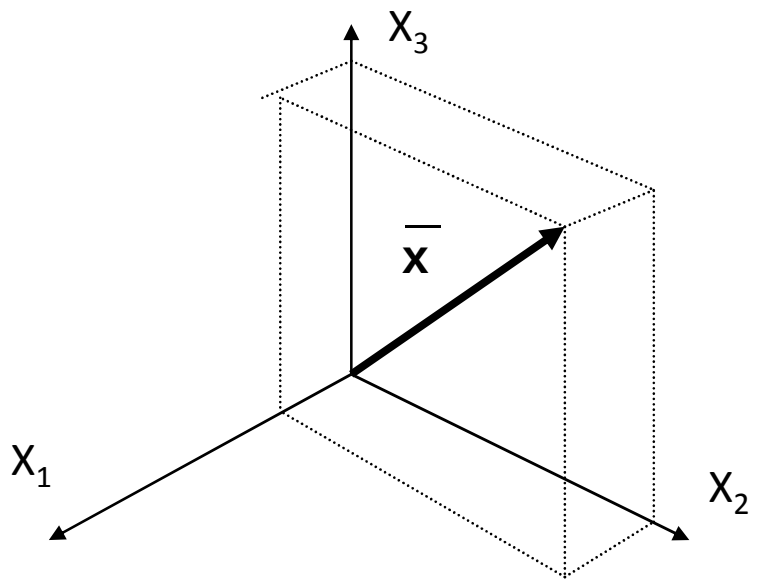


FEATURES

Feature vector

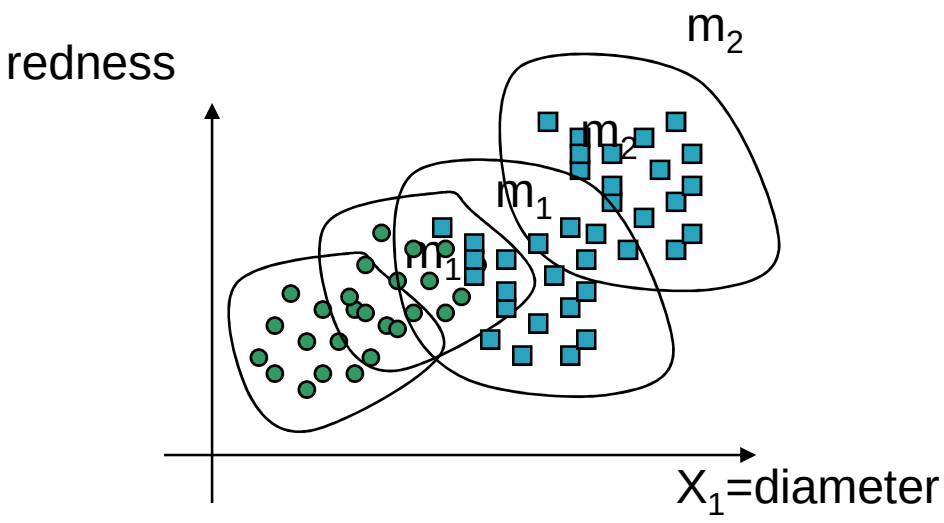
- The composition of various features in a vector is called feature vector
- A feature vector defines a point in an n-dimensional space

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$



Feature space

- The set of patterns belonging to the same class are grouped in some region of space
- Example fruit recognition:



$\mathbf{m_1}$ = lemons
$\mathbf{m_2}$ = apples

In this case, the separation is perfect, but it will not always be so, since classes are often overlapping

Difficult but very important

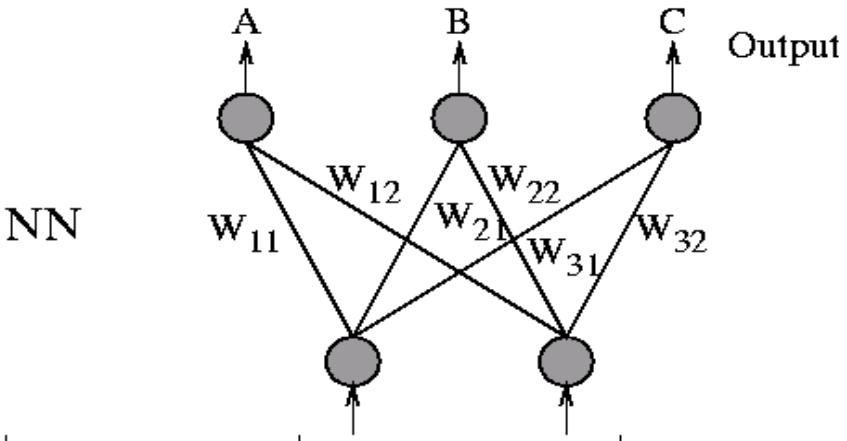
- Cats / dogs recognition

CLASSIFICATION

Intuitive approach

- **Intuitive approach**: comparing the unknown pattern with a standard pattern of each class, and choose the kind that comes closest.
  - How to compare?
  - How to select the standard pattern?
  - How to measure the degree of closeness?

Example



Symbols	Instantiations	Feature Representation		Network Output
A	a	2.1	3.1	A
A	a	1.9	2.7	A
B	b	1.1	2.2	B
C	c	1.2	1.2	C
C	c	1.1	1.6	-
B	b	1.1	1.9	B
B	b	1.1	1.4	C
C	c	0.9	1.1	C
B	b	1.0	2.2	B
A	a	2.2	3.0	A

Problem description

- To assign the input pattern to a single from N categories

$$y = f : R^n \Rightarrow \{Y_0, Y_1, \ldots Y_N\}$$

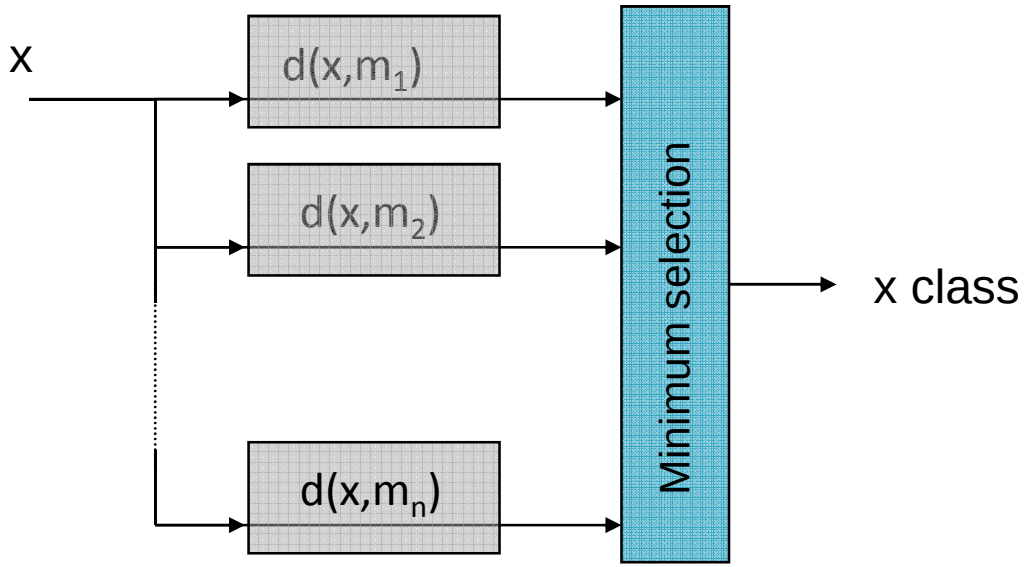
- Examples:
  - Speech Recognition
  - OCR
  - Expert Systems

Methods

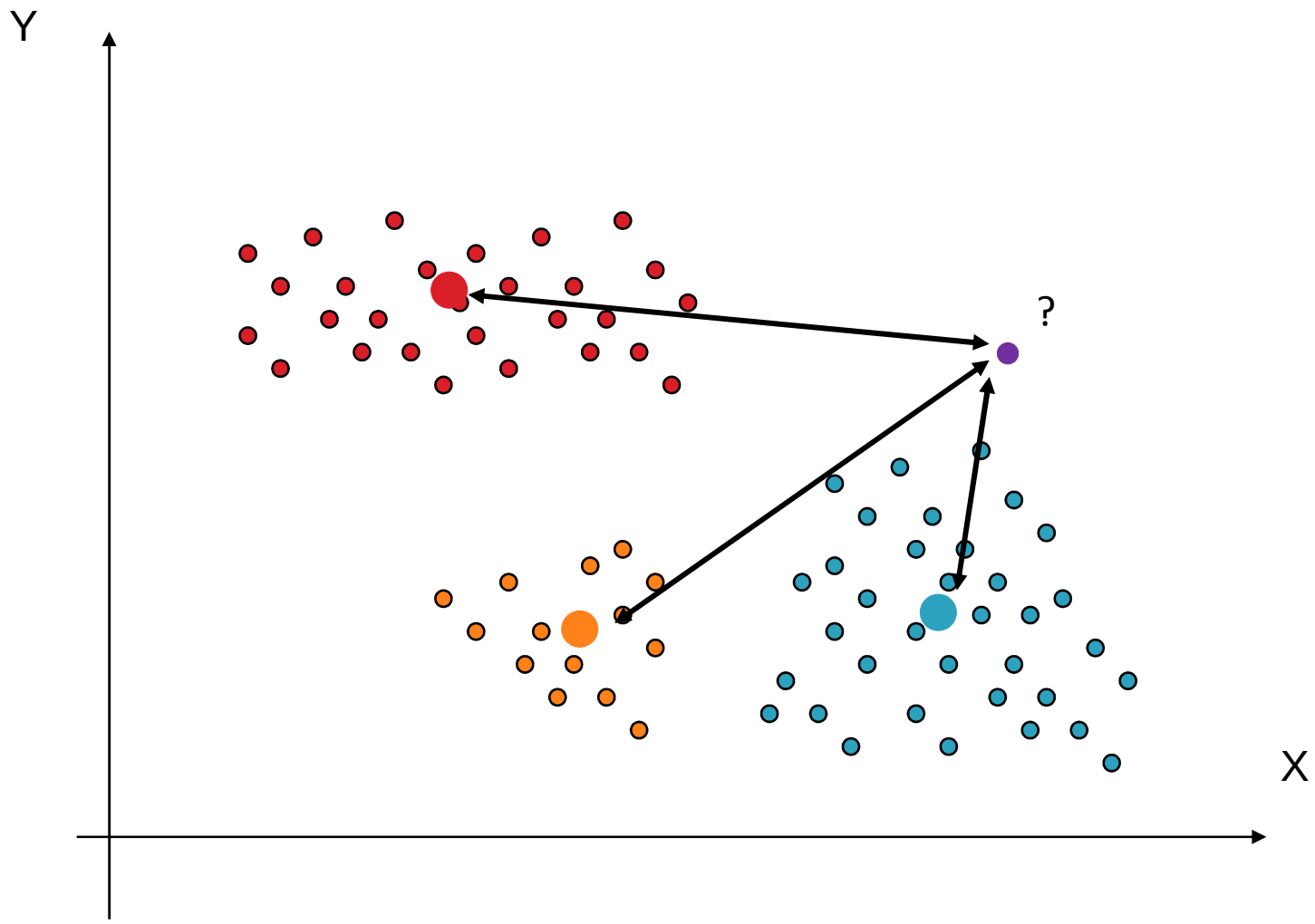
- Parametric techniques
  - To characterize each class
    - Minimum Distance method
    - Calculation of Discriminant Functions
  - To Characterize each border between classes
    - Borders decision
- Nonparametric techniques
  - Nearest neighbor
  - K-nearest neighbors

Minimum Distance method

- Define a prototype for each class m_k
- Find the distance from each pattern x to m_k
- Select the class whose prototype is closest to x

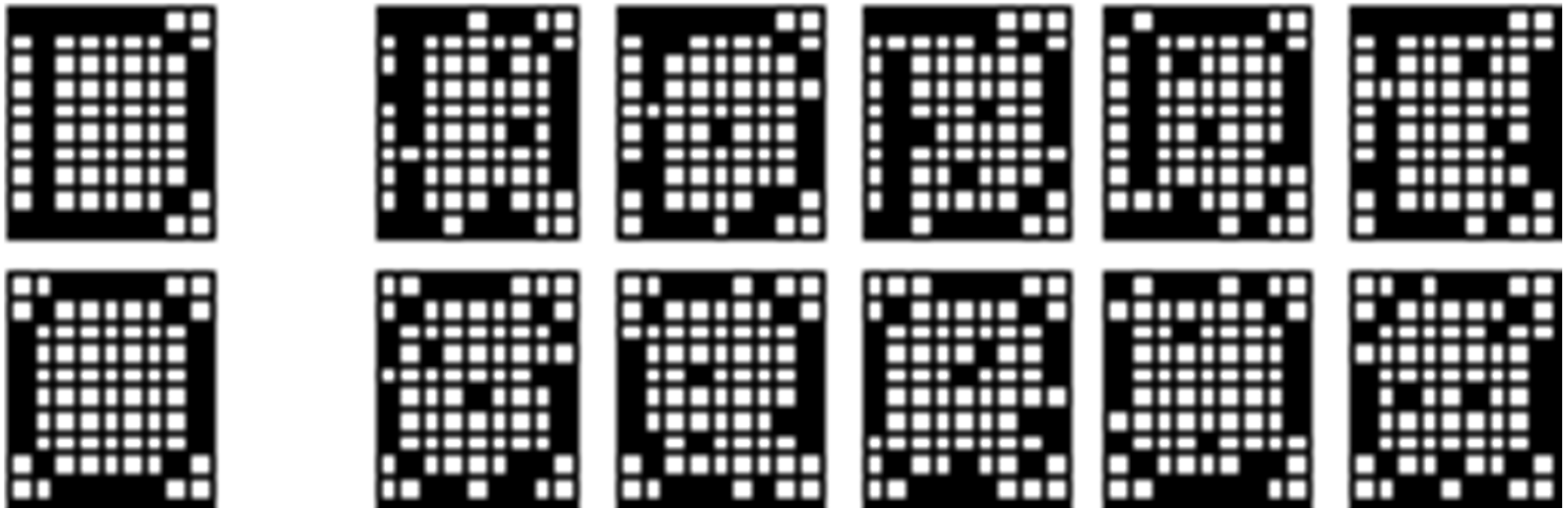


Geometric interpretation of the minimum distance

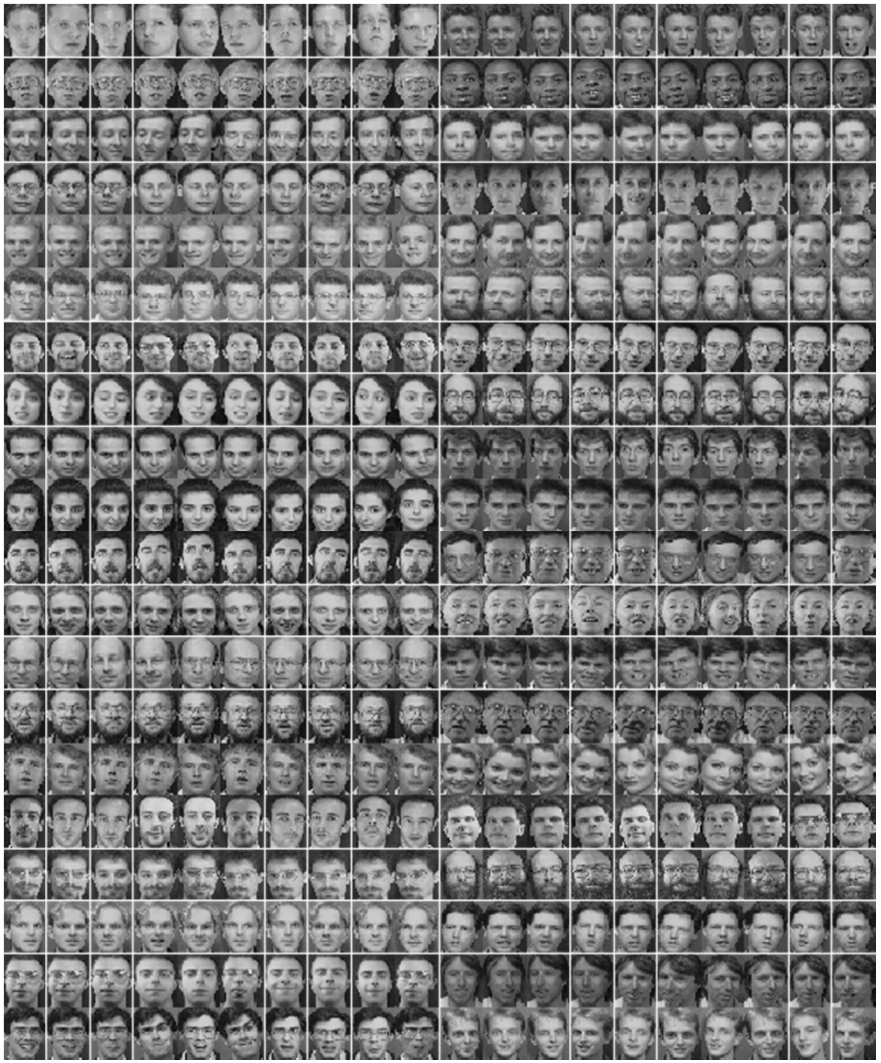


Prototype selection

- If possible, we will choose the not noisy prototype from which the data were obtained
- Otherwise, the prototype is usually approximated as the average of the available patterns

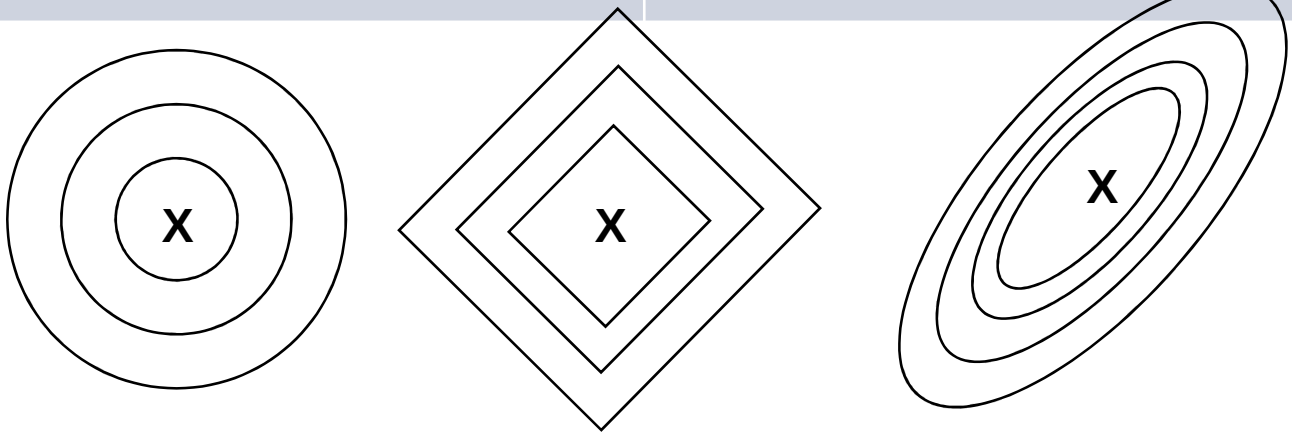


Prototype selection



Distance selection

Distance	Expresion
Eucliedan	$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
Manhattan	$d(x,y) = \sum_i \lvert x_i - y_i \rvert$
Mahalanobis	$d(x,y) = x' * C^{-1} * y$



Euclidean Manhattan Mahalanobis

The covariance

- It is defined as:

$$c(i,j) = \frac{1}{N-1}\sum_{k=1}^{N}(x_k - \mu_k^i)\cdot(x_k - \mu_k^j)$$

- The c(i, j) possible values are:
  - If c (i, j) > 0, both features tend to increase or decrease together
  - If c (i, j) < 0, a feature increases, the other decreases (and vice versa)
  - If c (i, j) = 0, both features are said to be independent

The Mahalanobis distance

- Mahalanobis distance is defined as

$$d(x, y) = x' * C^{-1} * y$$

and where C is the covariance matrix

- The minimum distance classifier Mahalanobis used for each class a mean and covariance matrix
- Properties:
  - Scaled
  - Correlation
  - Nonlinear class boundaries

Distance measures

Name	Formula
Euclidean metric	$d_E(\mathbf{x}_i, \mathbf{x}_j) = \{\sum_g w_g (x_{gi} - x_{gj})^2\}^{1/2}$
Unstandardized	$w_g = 1$
Standardized by s.d.	$w_g = 1/s_g^2$.
(Karl Pearson distance)	
Standardized by range	$w_g = 1/R_g^2$.
Mahalanobis metric	$d_{Ml}(\mathbf{x}_i, \mathbf{x}_j) = \{(\mathbf{x}_i - \mathbf{x}_j) S^{-1} (\mathbf{x}_i - \mathbf{x}_j)'\}^{1/2}$ $= \{\sum_g \sum_{g'} s^{-1}_{gg'} (x_{gi} - x_{gj})(x_{g'i} - x_{g'j})\}^{1/2}$ where $S = (s_{gg'})$ is any $G \times G$ positive definite matrix, usually the sample covariance matrix of the variables. When the matrix is the identity, this reduces to the unstandardized Euclidean distance.
Manhattan metric	$d_{Mn}(\mathbf{x}_i, \mathbf{x}_j) = \sum_g w_g \lvert x_{gi} - x_{gj} \rvert$
Minkowski metric	$d_{Mk}(\mathbf{x}_i, \mathbf{x}_j) = \{\sum_g w_g \lvert x_{gi} - x_{gj} \rvert^{\lambda}\}^{1/\lambda}$, $\lambda \geq 1$. $\lambda = 1$: Manhattan distance $\lambda = 2$: Euclidean distance
Canberra metric	$d_C(\mathbf{x}_i, \mathbf{x}_j) = \sum_g \frac{\lvert x_{gi} - x_{gj} \rvert}{(x_{gi} + x_{gj})}$
One minus Pearson correlation	$d_{corr}(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{\sum_g (x_{gi} - \bar{x}_{.i})(x_{gj} - \bar{x}_{.j})}{\{\sum_g (x_{gi} - \bar{x}_{.i})^2\}^{1/2} \{\sum_g (x_{gj} - \bar{x}_{.j})^2\}^{1/2}}$

The formulae refer to distances between observations (arrays).

The confusion matrix

- It indicates in matrix form the successes and failures committed in the classification process

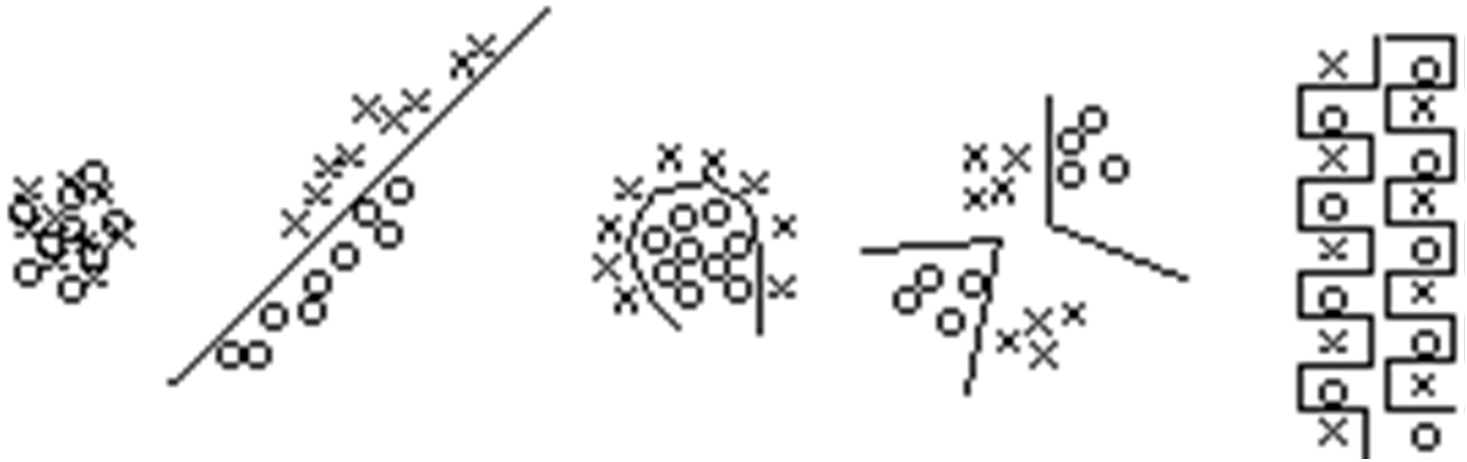
		Estimated classes	
		ClassA	Class B
Real classes	Class A	94.63	5.37
	Class B	13.95	86.05

Confusion matrix

Minimum distance method problems

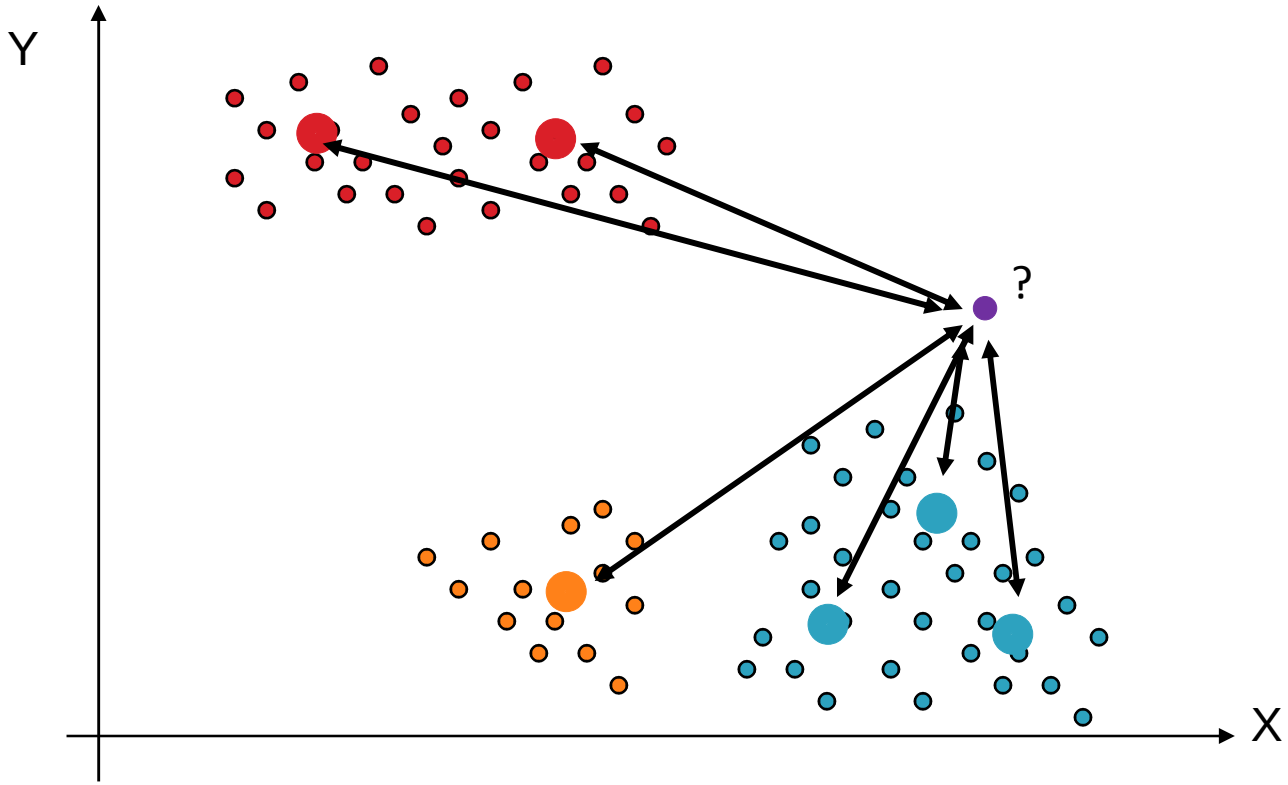
- Inadequate features
- High correlation
- Nonlinear decision boundaries
- Subcategories existence
- Complex classes separations

Solution: feature extraction



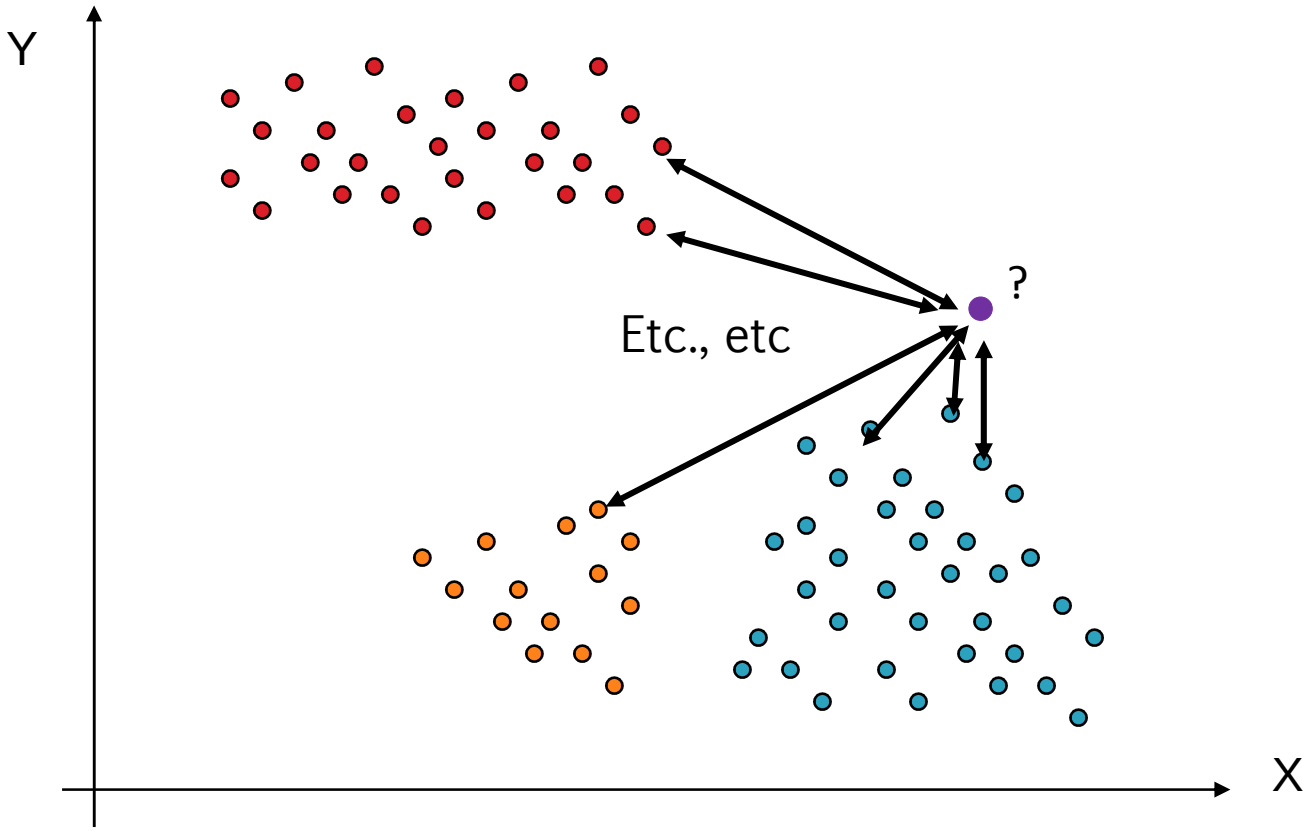
Multiple prototypes -> Clustering

- What if instead of a single prototype for each class, we selected several prototypes?



Pattern = prototype -> nonparametric methods

- What if each pattern is considered a valid prototype?



Non parametric classification

- Consider each pattern as a valid prototype
- The most common nonparametric techniques are:
  - Nearest Neighbor
  - K-nearest neighbors

Nearest Neighbor technique

- We start with a set of patterns that we know their class
- An unknown pattern is assigned to the class which the closest pattern belongs
- You must define a distance function, which satisfies:

$$d(x, y) = 0 \Leftrightarrow x = y$$

$$d(x, y) = d(y, x)$$

$$d(x, z) \leq d(x, y) + d(y, z)$$

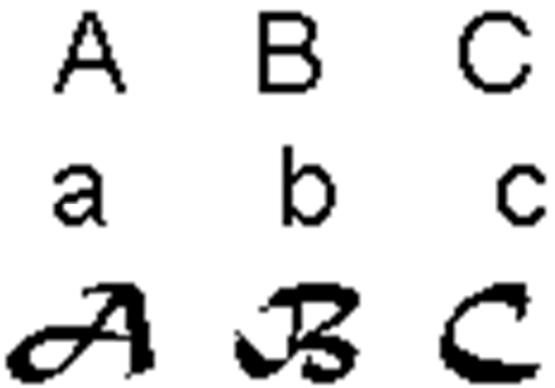
K-nearest neighbors technique (KNN)

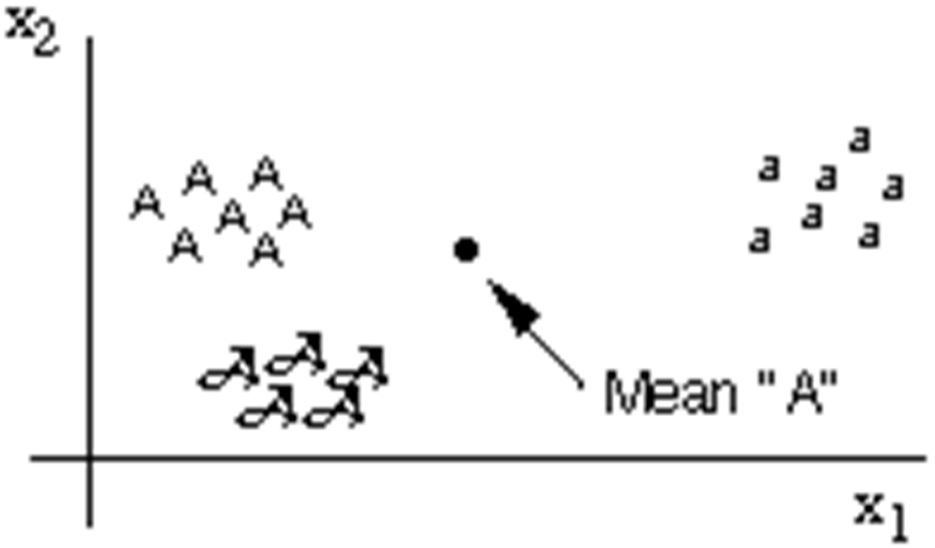
- K nearest neighbors are searched
- The pattern to the majority class (K-nearest neighbor) is assigned by a vote
- In case of tie, is normally assigned to the class of the pattern closest
- It offers excellent results, at the expense of sluggishness classification

CLUSTERING

Clustering

- Can detect the existence of subclasses data
- Used unsupervised learning



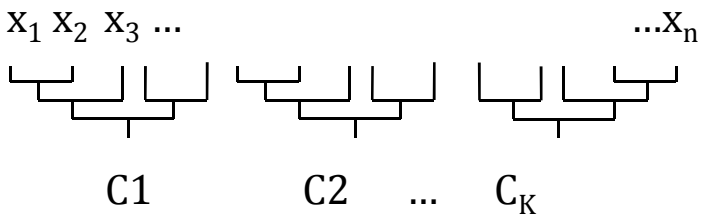


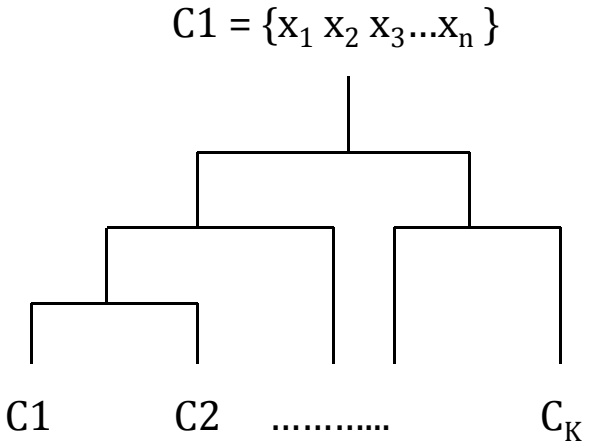
Clustering methods

- The most popular methods are:
  - Hierarchical methods
    - Agglomerative
    - Divisive
  - Method based on K-means
    - Method of K-means (also called LBG algorithm, or Generalized Lloyd algorithm)
    - Method of Fuzzy K-means
  - EM algorithm
  - Kohonen self-organizing maps

Hierarchical methods

- Agglomerative method:
  - Initially, each data is a valid prototype
  - Mix the more similar two prototypes
  - Repeat until the number of clusters is the desired
- Divisive method:
  - Initially considered a single cluster
  - According to a predefined criterion, a cluster is chosen, and is divided into two (or more)
  - Repeat until the number of clusters is the desired





K – Means method

- Decide the value of K
- Selecting initial K vectors, m_1, m_2, ... m_k, for example, between the input patterns
- Repeat,
  - a pattern x belongs to cluster j-th if its distance to m_j is the lowest
  - Recalculate the m_1, m_2, ... m_k as the mean vectors of each cluster
- Until it converges or No> Maxit
- remain unchanged until the m_1, m_2, ... m_k

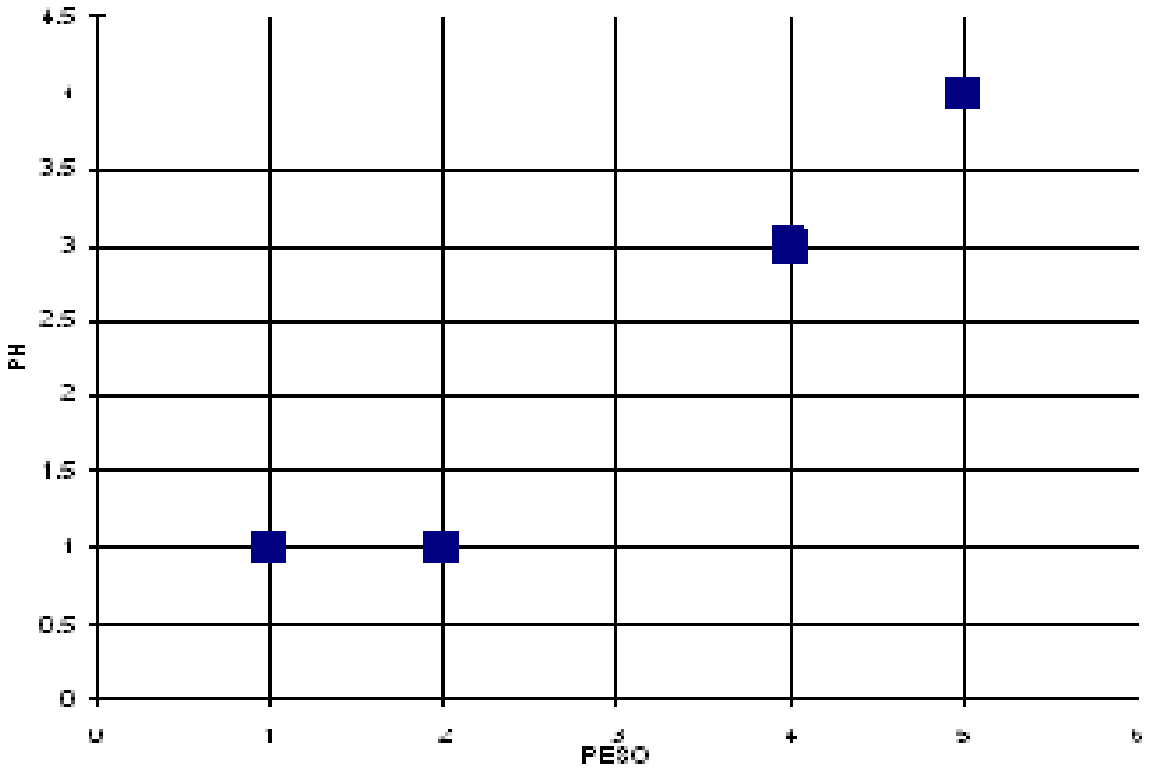
K – Means example

- The weight and PH index of four medicine types are known:

Medicines	Weight	PH index
A	1	1
B	2	1
C	4	3
D	5	4

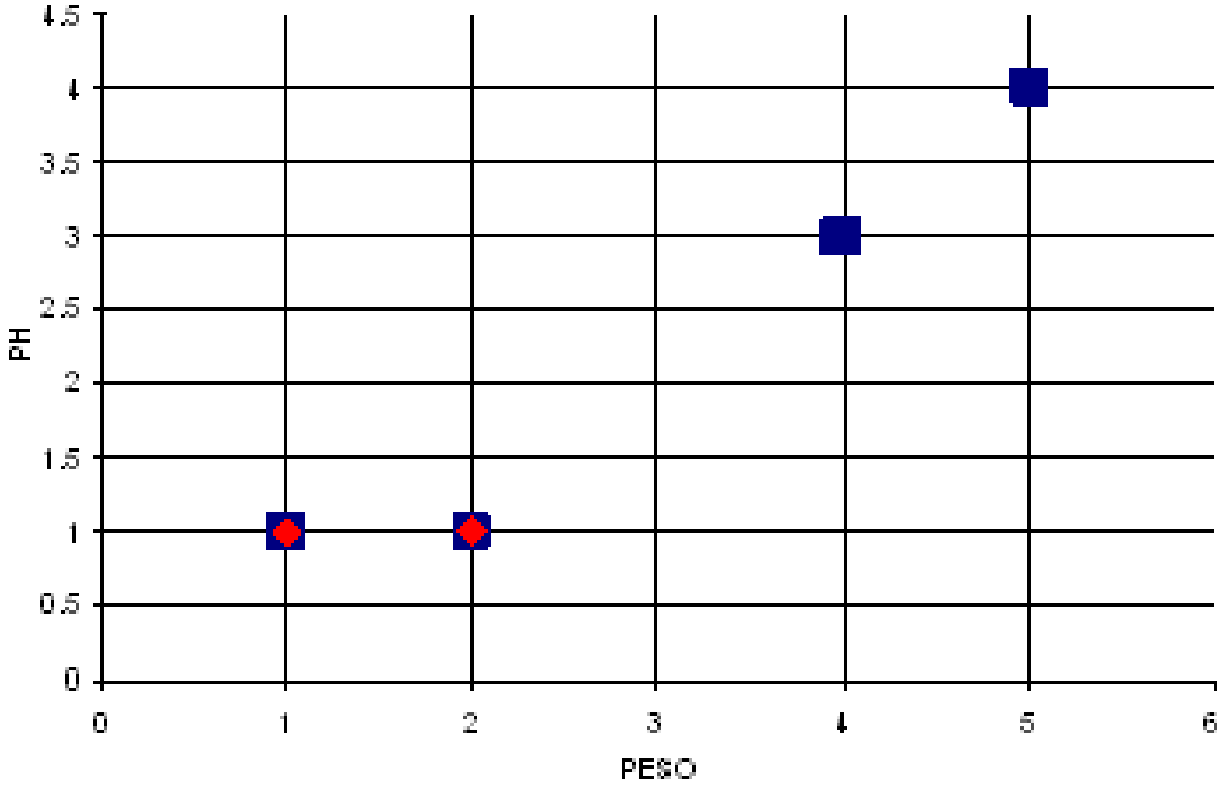
K – Means example

- Each type of medicine can be represented as a point in space based on two attributes:



K – Means example

- Initially chose two centroids of the k - groups that match the types of medicines A (1,1) and B (2,1)



K – Means example

- Distance calculation from each object to centroids (Euclidean distance):
  - Distance between type of medicine C (4,3) and the first centroid (1,1):

$$\sqrt{(4-1)^2+(3-1)^2}=3.61$$

  - Distance between type of medicine C (4, 3) and the second centroid (2 1):

$$\sqrt{(4-2)^2+(3-1)^2}=2.83$$

- Distance matrix:

$$D^0=\begin{bmatrix}0 & 1 & 3.61 & 5.00\\ 1 & 0 & 2.83 & 4.24\end{bmatrix}$$

K – Means example

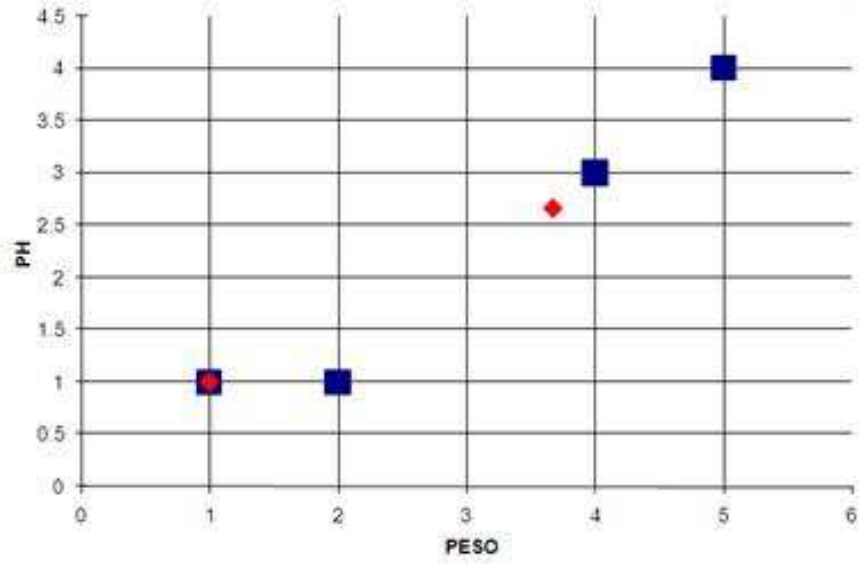
- Clustering: each object is assigned to a group taking into account the minimum error.
  - The type of medicine A is assigned to group 1 (centroid (1,1))
  - Types B, C and D are assigned to group 2 (centroid (2,1))

$$G^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

K – Means example

- Iteration 1, centroids are determined
  - As group 1 has only one member (type A) is left as is.
  - Group 2 has three members, therefore, its centroid is:

$$c_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3}\right) = \left(\frac{11}{3}, \frac{8}{3}\right)$$



K – Means example

- Iteration 1, distance matrix:

$$D^1 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \end{bmatrix}$$

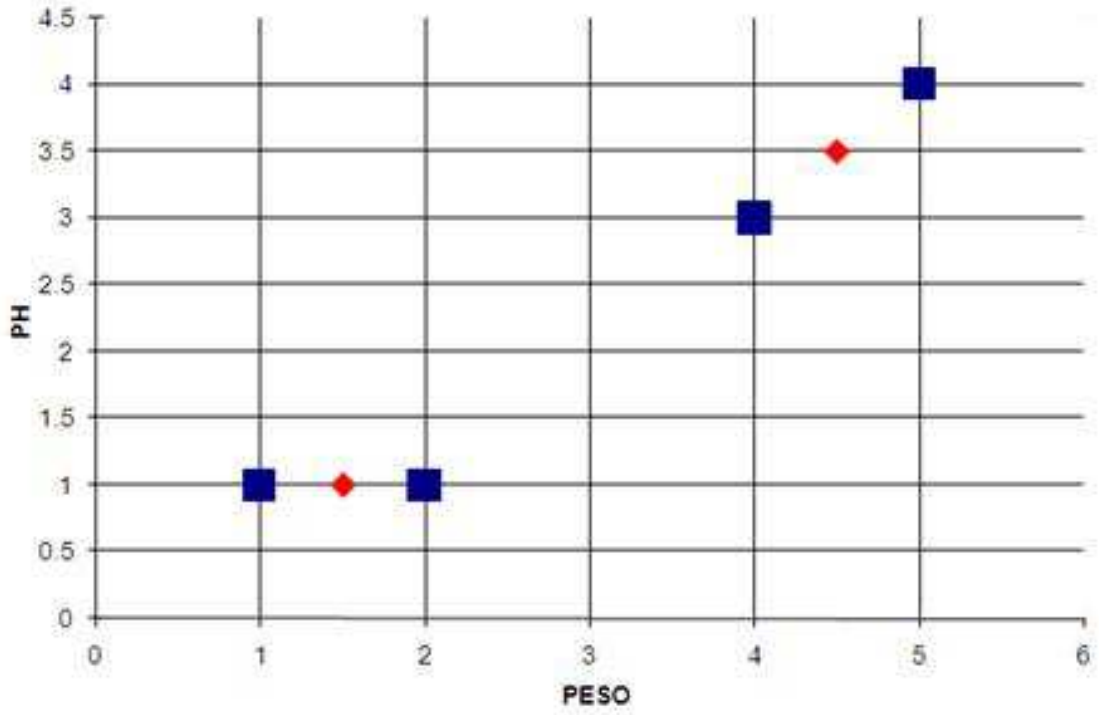
- Iteration 1, clustering:
  - A and B are assigned to group 1 (centroid in (1,1))
  - C and D are assigned to group 2 (centroid (11/3, 8/3))

$$G^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

K – Means example

- Iteration 2, centroids calculation

$$c_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = \left(\frac{3}{2}, 1\right) \quad c_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = \left(\frac{9}{2}, \frac{7}{2}\right)$$



K – Means example

- Iteration 2, distance matrix:

$$D^2 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.30 & 3.54 & 0.71 & 0.71 \end{bmatrix}$$

- Iteration 2, clustering:
  - A and B are assigned to group 1 (centroid in (1,1))
  - C and D are assigned to group 2 (centroid (11/3, 8/3))

$$G^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Fuzzy K-Means method

- It allows each point belongs partly to several (all?) clusters
- Define a degree of membership in each cluster, depending on their distance to the cluster
- Must be defined
  - A distance measure of a vector to a cluster (or prototype), such as:

$$a_{ij} = \frac{1}{\| x_j - m_i \|^2} \qquad a_{ij} = e^{-\| x_j - m_i \|^2}$$

  - Membership degree of a vector to a cluster. It is usual to use:

$$u_{ij} = \frac{a_{ij}}{\sum_{j=1}^{K} a_{ij}}$$

Fuzzy K-Means method

- Decide the value of K
- Let i = 1,2, ... K the number of classes; $j = 1,2, \ldots N$ the number of points, the steps are:
  - Determine the initial values of the centers of the K classes my (random initialization, using K-means, ...)
  - Repeat
    - Determine the values a_{ij} using the distance function chosen
    - Determine the values u_{ij} $$u_{ij} = \frac{a_{ij}}{\sum_{j=1}^{K} a_{ij}}$$
    - Recalculate centers me as: $$m_i = \frac{\sum_{j=1}^{N} u_{ij} \cdot x_j}{\sum_{j=1}^{N} u_{ij}}$$

Until it converges or No> Maxit

EM algorithm

- The EM algorithm is a general method to find the estimate of maximum likelihood parameters of an underlying distribution to a data set
- This is an iterative algorithm, named after the two steps in each iteration is divided:
  - Expectation
  - Maximization

EM, clustering based on probabilities

- Instances have some probability of belonging to a cluster, we look for the clusters group most likely given the data.
- The basis of this type of clustering is a statistical model called mixture of distributions (Mixtures finite):
  - Each distribution shows the probability that an object has a particular set of attribute-value pairs if you know who is a member of that cluster.
  - They have k probability distributions representing k clusters.

EM – Simple case

- Numeric attributes with Gaussian distributions where we know what each data cluster belongs.

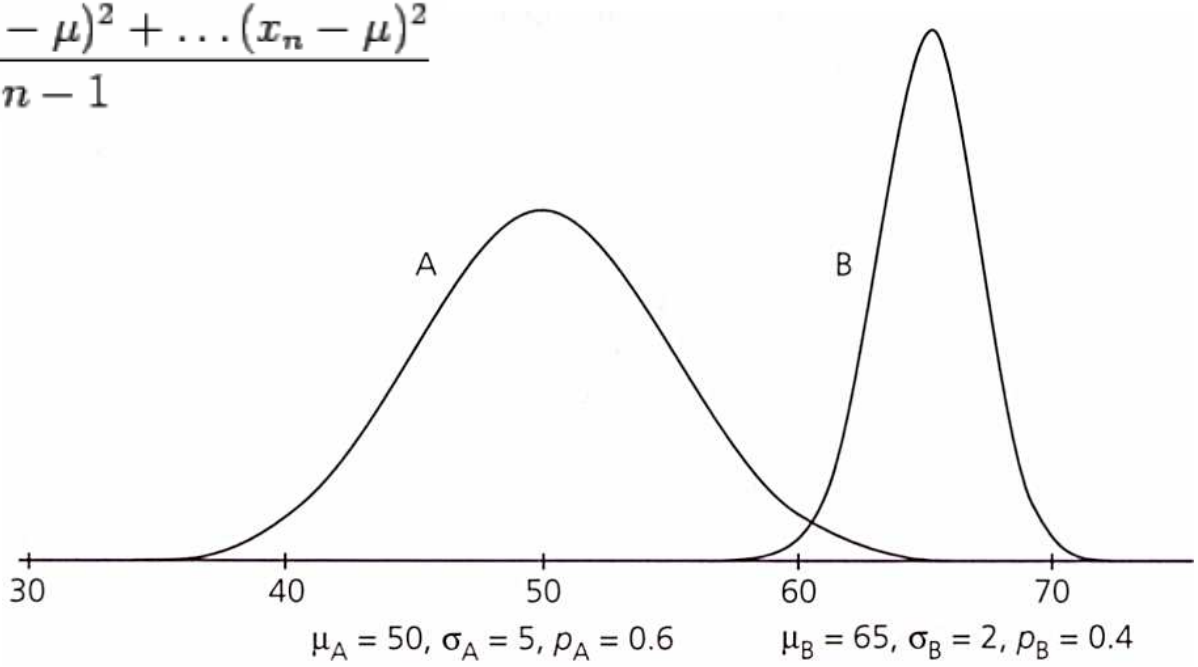
A	51	B	62	B	64	A	48	A	39	A	51
A	43	A	47	A	51	B	64	B	62	A	48
B	62	A	52	A	52	A	51	B	64	B	64
B	64	B	64	B	62	B	63	A	52	A	42
A	45	A	51	A	49	A	43	B	63	A	48
A	42	B	65	A	48	B	65	B	64	A	41
A	46	A	48	B	62	B	66	A	48		
A	45	A	49	A	43	B	65	B	64		
A	45	A	46	A	40	A	46	A	48		

EM – Simple case

$$\mu = \frac{x_1 + x_2 + \ldots x_n}{n}$$

$$P(A|x) = \frac{P(x|A)P(A)}{P(x)} = \frac{f(x;\mu_A,\sigma_A)P_A}{P(x)}$$

$$\sigma^2 = \frac{(x_1 - \mu)^2 + (x_2 - \mu)^2 + \ldots (x_n - \mu)^2}{n-1}$$



EM - What to do when the case is not ideal?

- **Problem**: we do not know what each data distribution is and we do not know the parameters of the distributions.
- **Solution:** EM algorithm

EM – Algorithm steps

1. The distributions parameters will "guess"
2. The parameters values of the distributions are used to calculate the likelihood that each object belongs to a cluster (expectation):

$$P(A|x) = \frac{P(x|A)P(A)}{P(x)} = \frac{f(x;\mu_A,\sigma_A)P_A}{P(x)}$$

3. The parameters of the distributions (maximization) are recalculated and return to step 2.

$$\mu_A = \frac{w_1x_1 + w_2x_2 + \ldots w_nx_n}{w_1 + w_2 + \ldots w_n} \qquad \sigma_A^2 = \frac{w_1(x_1-\mu)^2 + w_2(x_2-\mu)^2 + \ldots w_n(x_n-\mu)^2}{w_1 + w_2 + \ldots w_n}$$

EM - Concluding remarks

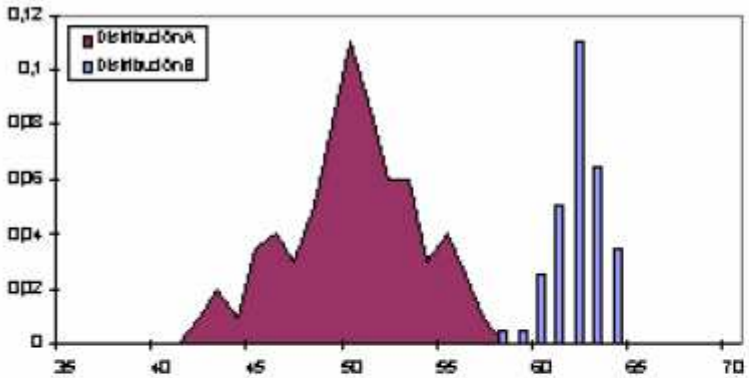
- The algorithm tends to converge, but never reaches a fixed point.
- Objective: Maximize (maximization) the likelihood of distributions given the data:

$$\prod_i (P_A P(x_i|A) + P_B P(x_i|B))$$

- The algorithm iterates until growth is negligible.
- Convergence can be a local maximum, repeat the process several times.

EM - Example

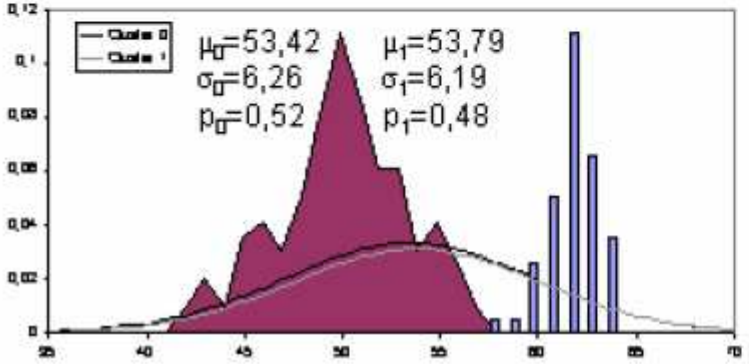
Distribuciones Originales



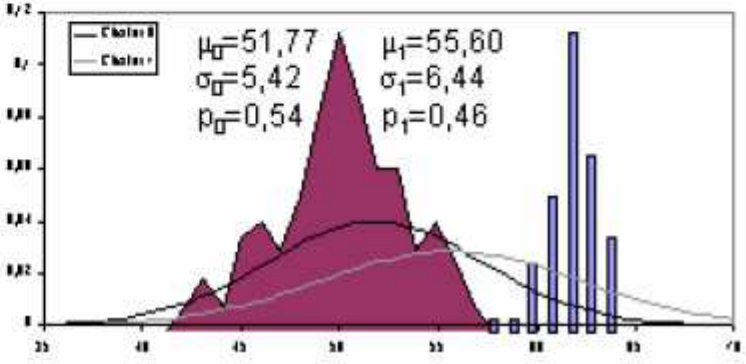
200 ejemplos que forman dos distribuciones desconocidas con parámetros:

μ_A=50 μ_B=62
σ_A=3,4 σ_B=1,26
p_A=0,7 p_B=0,3

Iteración 1

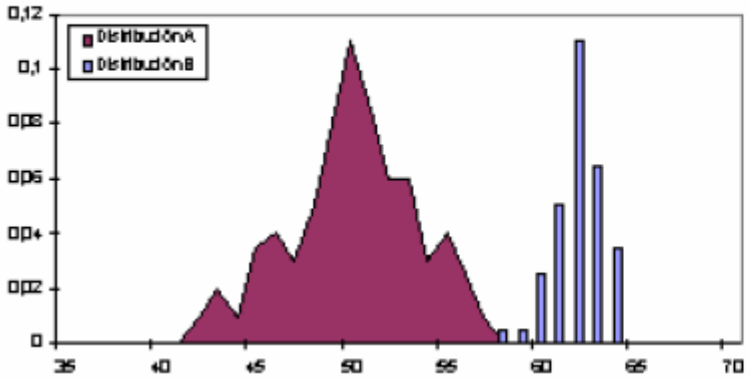


Iteración 5



EM - Example

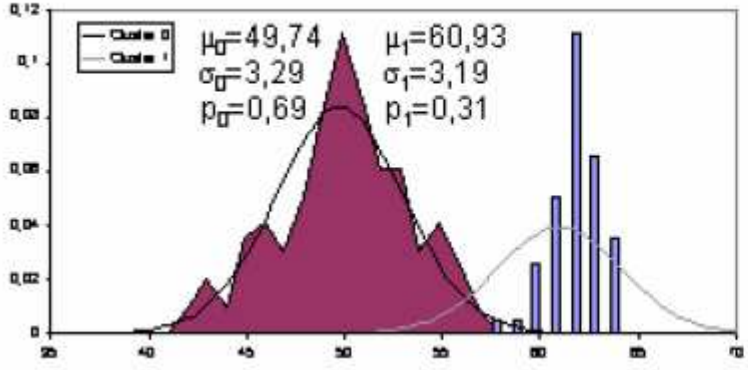
Distribuciones Originales



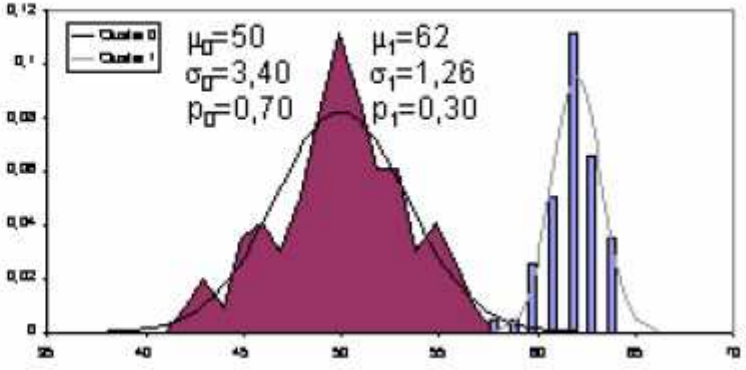
200 ejemplos que forman dos distribuciones desconocidas con parámetros:

$\mu_A=50$ $\mu_B=62$
$\sigma_A=3,4$ $\sigma_B=1,26$
$p_A=0,7$ $p_B=0,3$

Iteración 8



Iteración 11



EM algorithm applied to multivariate Gaussian

- Extension to instances with multiple attributes:
  - If the attributes independence is assumed, it can be made by multiplying the probabilities of each attribute together to get a joint probability distribution.
  - If there are correlated attributes, it may be modeled with a bivariate normal distribution, wherein a covariance matrix is used. The number of parameters grows, it can be a overfitting problem
- For nominal attributes with *m* possible values, is characterized by *m* numerical values representing the probability of each value.