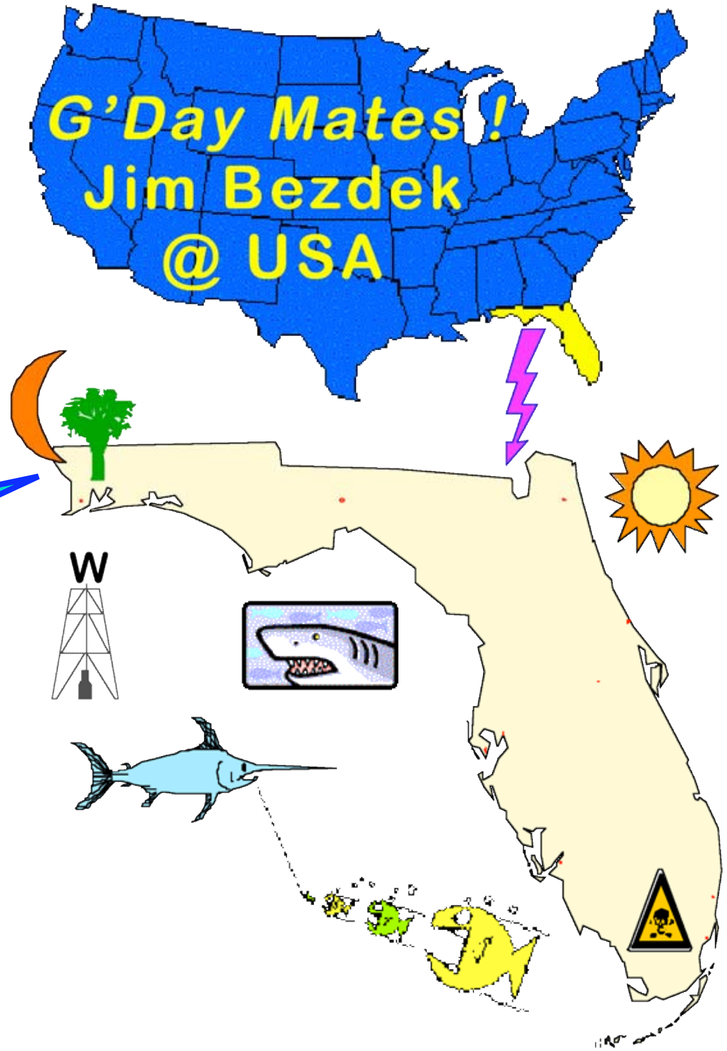
I'm from ...

University of West Florida

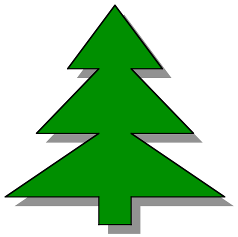


Pensacola

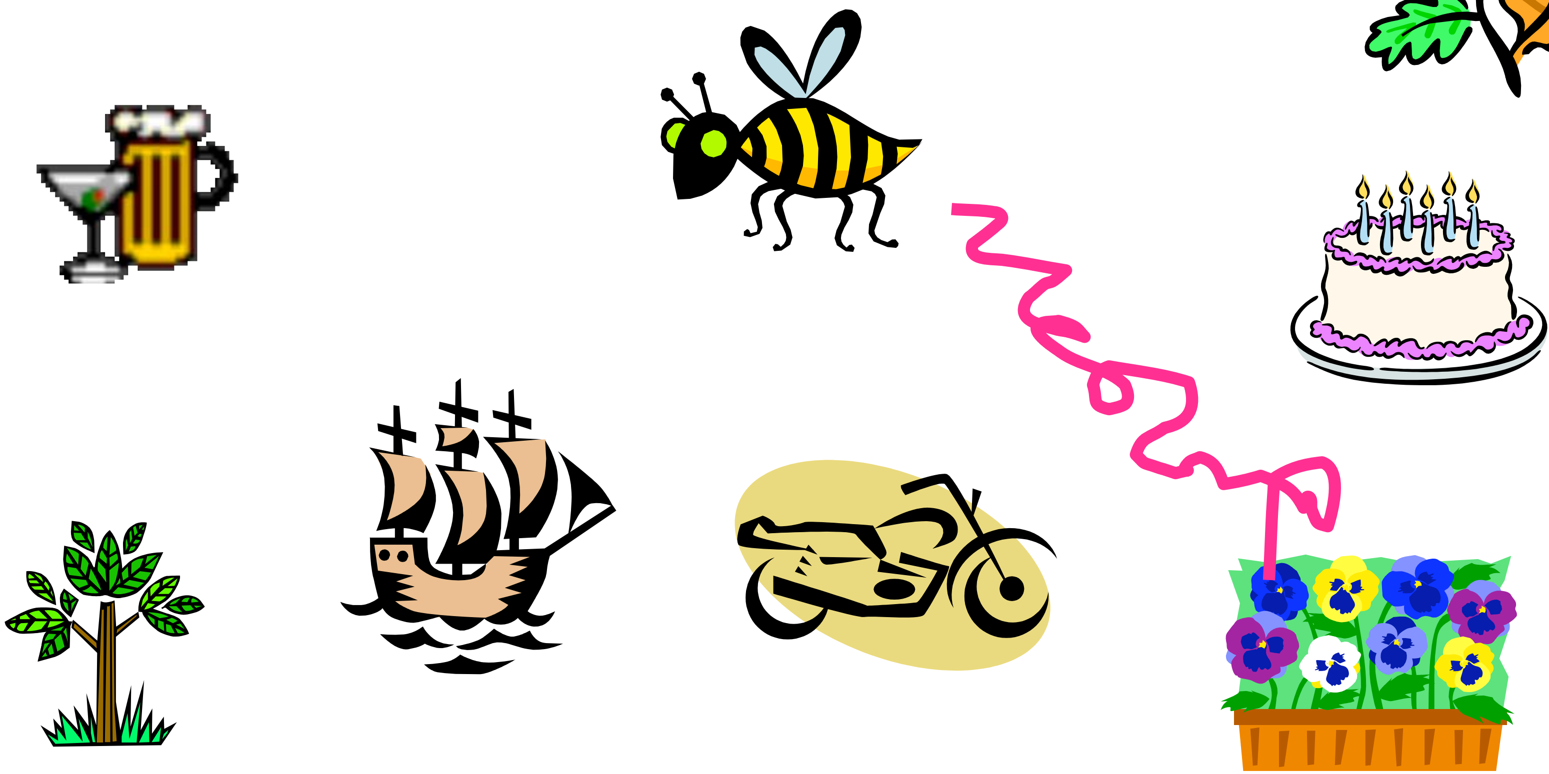
Pensacola before Hurricane IVAN


Pensacola
after 9/16/04

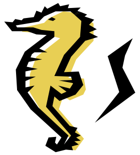
What IS Pattern Recognition ?



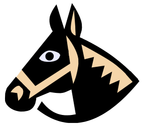
Where will this bee go?



Pattern Recognition

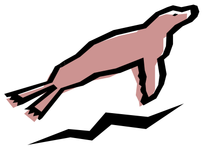


Numerical Pattern Recognition

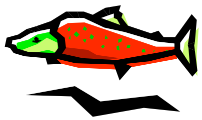


Feature Analysis

Similarity : Distances and Norms



Label Vectors and Partitions



Cluster Analysis

Classifier Design

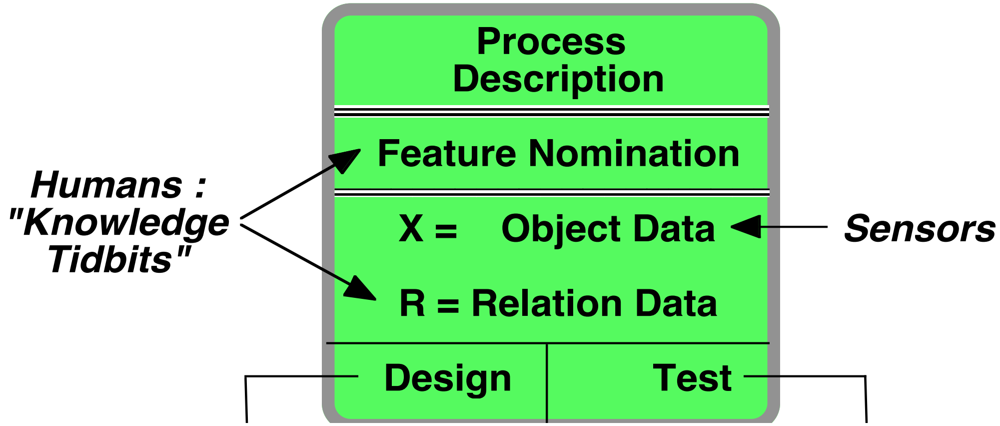
Numerical Pattern Recognition

Process Description

Feature Analysis

Cluster Analysis

Classifier Design



2 kinds of data for Pattern Recognition

Objects $O = \{o_1, \ldots, o_n\}$: o_i = i-th *physical* object

Object Data $X = \{x_1, \ldots, x_n\}$: x_i = *feature vector* for o_i

x_{ji} = j-th (*measured*) feature of x_i : $1 \leq j \leq p$

Relational data $R = [r_{ij}]$ = *relationship*(o_i, o_j)) or (x_i, x_j)

s_{ij} = *similarity* (o_i, o_j) or (x_i, x_j)

d_{ij} = *dissimilarity* (o_i, o_j) or (x_i, x_j)

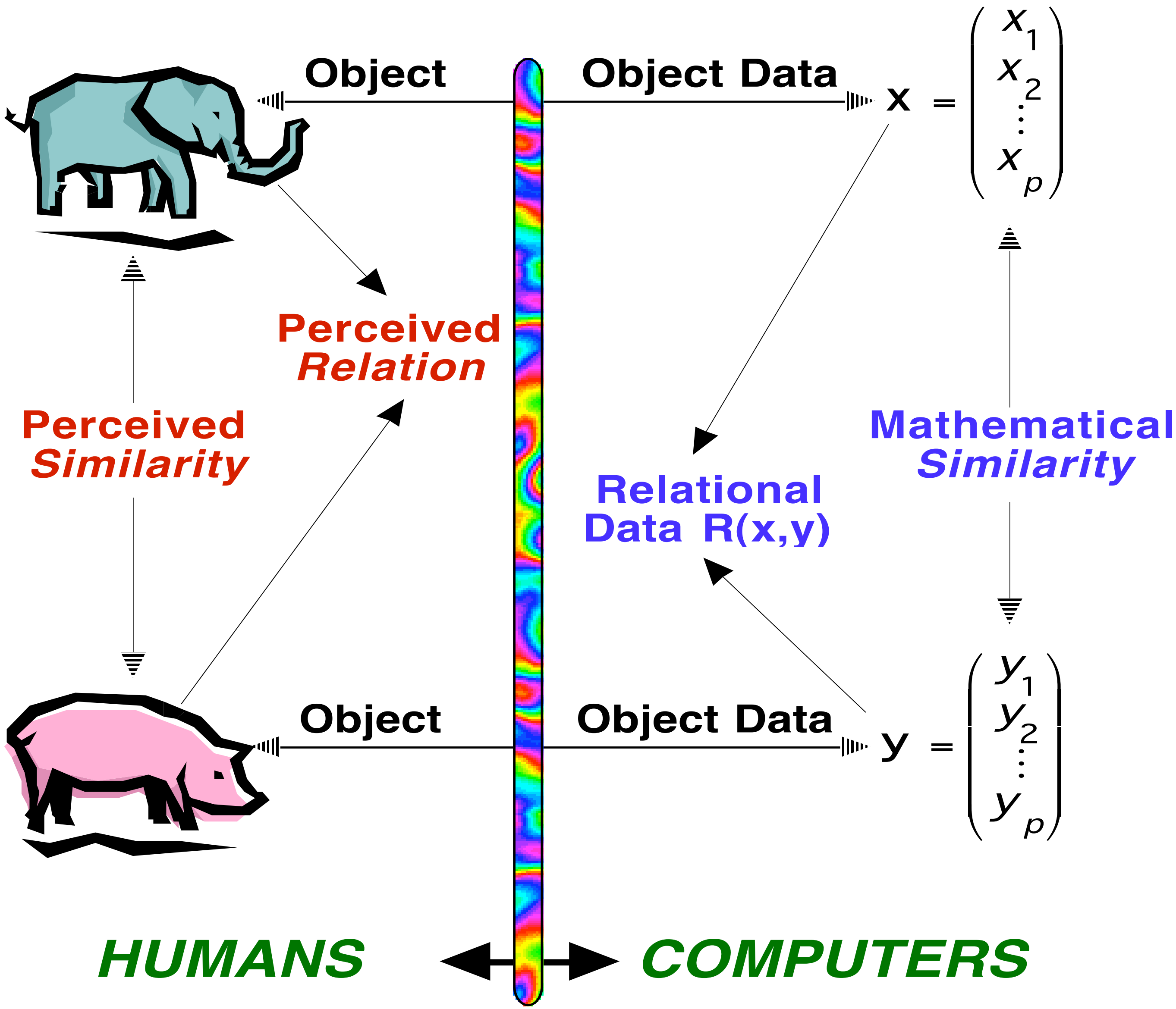
Tyically *(R = D)*

(i) $d_{ij} \geq 0$ $1 \leq i \neq j \leq n$

(ii) $d_{ii} = 0$ $1 \leq i \leq n$

(iii) $d_{ij} = d_{ji}$ $1 \leq i \neq j \leq n$

When $X \rightarrow D$, $d_{ij} = \|x_i - x_j\|_E$, D is "Euclidean"


Object
Object Data
x =
x_1
x_2
x_p
Perceived Relation
Perceived Similarity
Relational Data R(x,y)
Mathematical Similarity
Object
Object Data
y =
y_1
y_2
y_p
HUMANS
COMPUTERS

Measuring (Dis)Similarity : The 5 Good Norms

Mahalonobis

< , > Norms

Diagonal

$$\delta_{M^{-1}} = \|x - v\|_{M^{-1}} = \sqrt{(x - v)^T M^{-1} (x - v)}$$

$$\delta_{D^{-1}} = \|x - v\|_{D^{-1}} = \sqrt{(x - v)^T D^{-1} (x - v)}$$

Euclidean

$$\delta_2 = \|x - v\|_{I^{-1}} = \sqrt{(x - v)^T I^{-1} (x - v)}$$

City Block

Minkowski Norms

Sup or Max

$$\delta_1 = \|x - v\|_1 = \sum_{j=1}^{p} |x_j - v_j|$$

$$\delta_\infty = \|x - v\|_\infty = \max_{j=1}^{p} \{|x_j - v_j|\}$$

Parameters of the Statistical Norms

Mean Vector

$$\bar{v} = \sum_{k=1}^{n} x_k / n$$

Covariance Matrix

$$M = [m_{ij}] = \frac{\sum_{k=1}^{n} (x_k - \bar{v})(x_k - \bar{v})^T}{n}$$

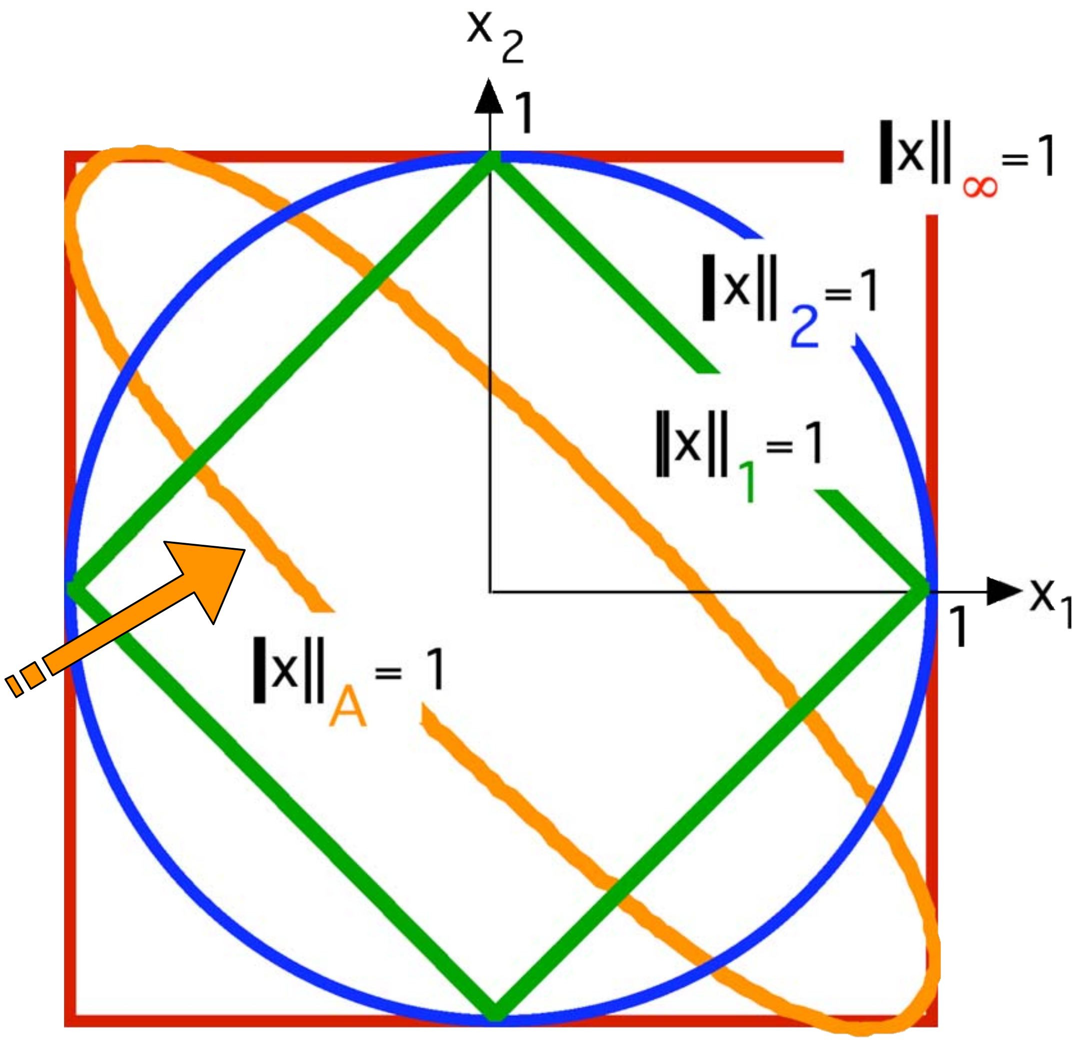
Variance (Matrix)

$$D = \begin{bmatrix} m_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & m_{pp} \end{bmatrix} = \begin{bmatrix} s_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_p^2 \end{bmatrix}$$

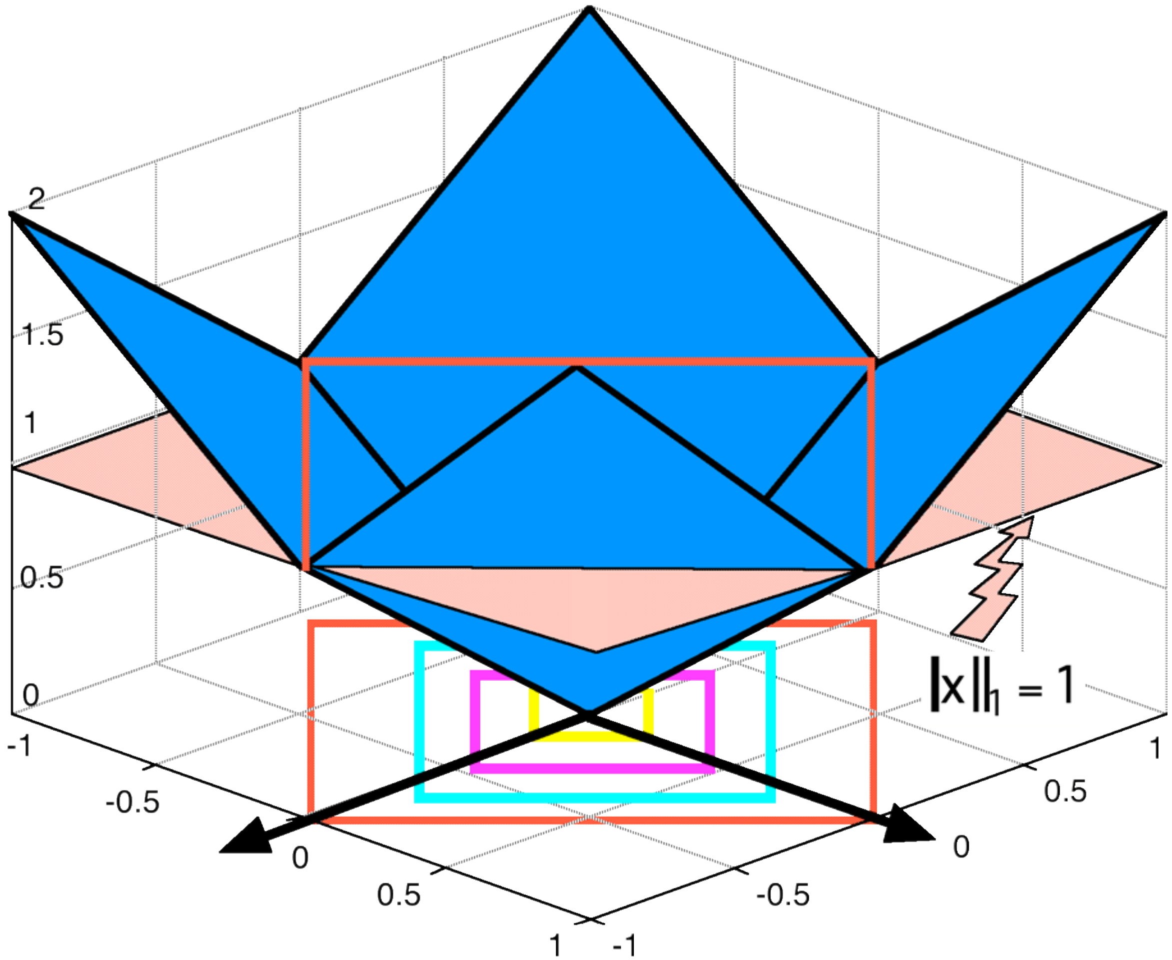
Level Sets and Open balls of Norms in R^2

Inner Product Norms
induce elliptical balls
via eigenstructure of *A*

$$\langle x, x \rangle_A = \|x\|_A^2 = x^T A x$$

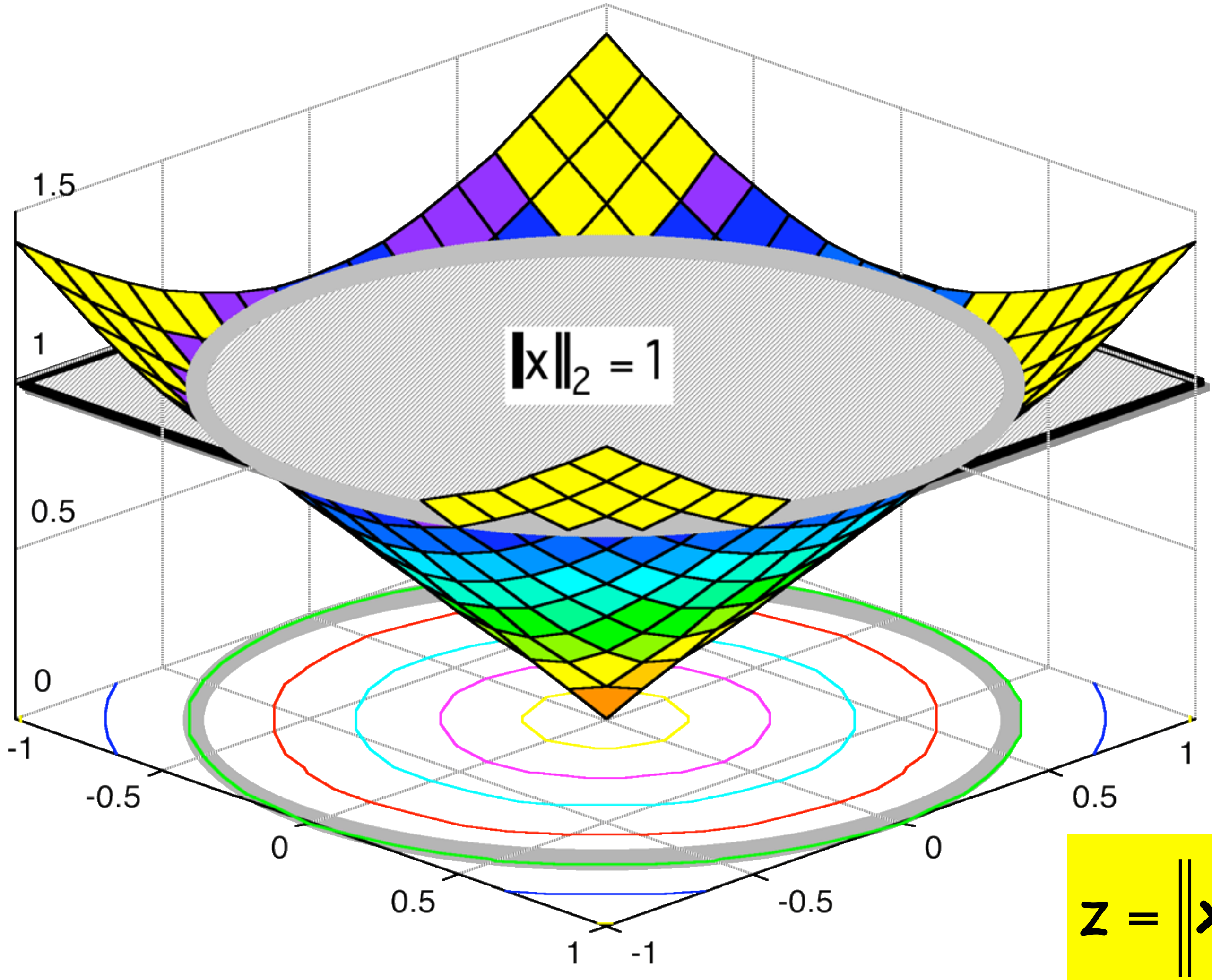


Level Sets and Open Balls of 1-norm



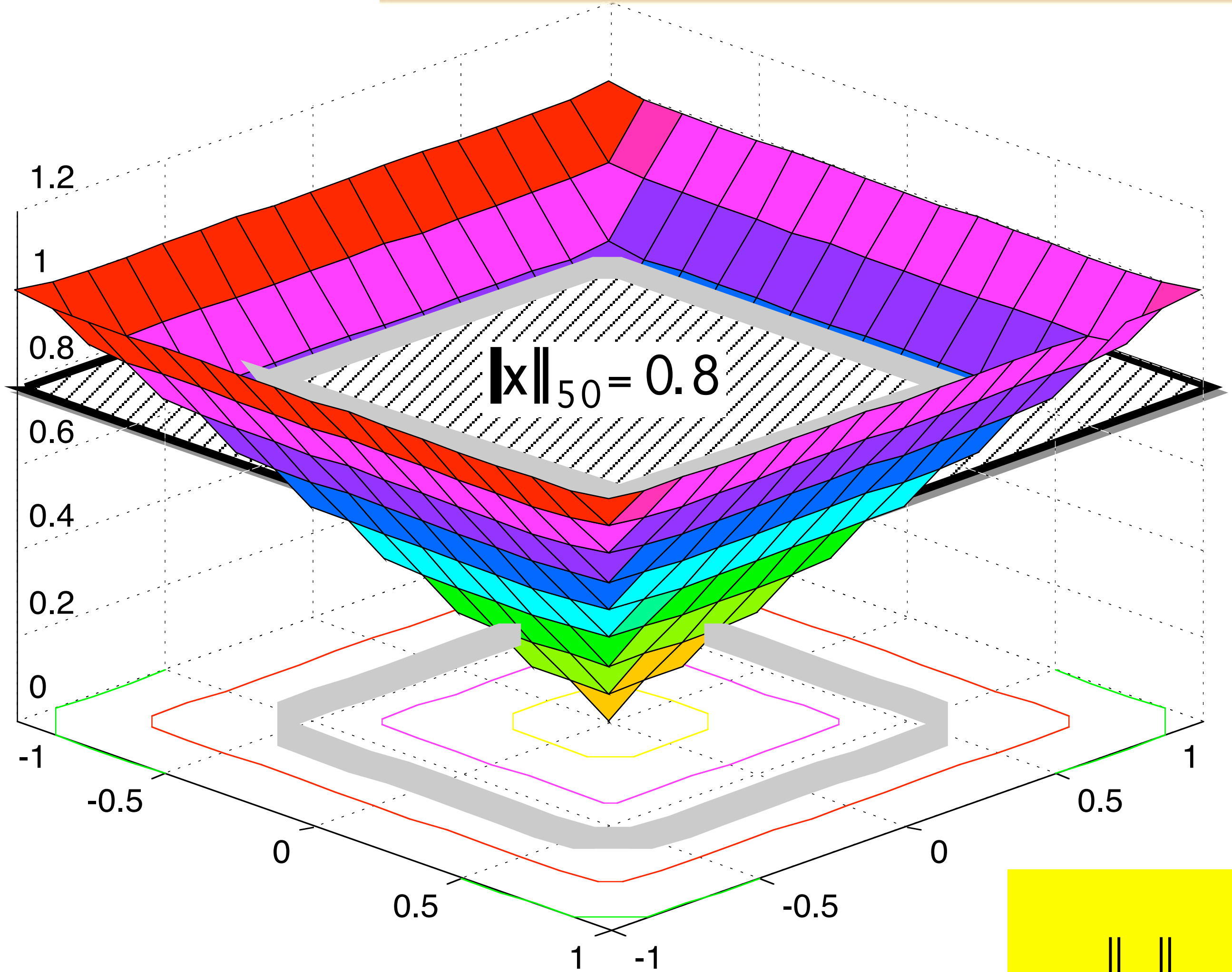
$$z = \|x\|_1 = |x| + |y|$$

Level Sets and Open Balls of 2-norm



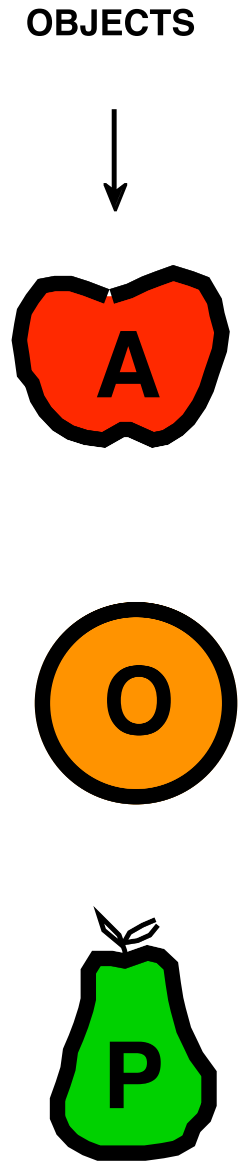
$$z = \|\mathbf{x}\|_2 = \sqrt{x^2 + y^2}$$

Level Sets and Open Balls of 50-norm



$$z = \|x\|_{50} = \left(x^{50} + y^{50}\right)^{\frac{1}{50}}$$

Feature Analysis



Feature Nomination : Issues

? What are the problems to solve ?

? What variables are important ?

? Which variables *can be* measured ?

? What data analysis will be done ?

? Are *more* features better than *less* ?

⇓ ⇓ ⇓

Propose and collect raw (object) data

Feature Extraction : Why?

Find $\phi : \underbrace{P(\Re^p)}_{X} \mapsto \underbrace{P(\Re^q)}_{Y=\phi[X]}$ to:

1. Increase dimension : p < q

Functional link networks
Support Vector Machines

2. Decrease dimension : p > q

Decrease time and space
Eliminate “redundant” features
Assess tendencies visually (q=2)

3. Y should have the “same information” as X for :

Clustering and Classification
Prediction and Control

Feature Extraction : How?

1. Functions

$$f : \underbrace{\Re^p}_{X} \mapsto \underbrace{\Re^q}_{Y=f(X)}$$

Linear

Axial ON Projection
Derived ON Projection (PCA)
Derived non-OG Projection

Non-Linear

3. Use your 

2. Algorithms

$$A : \underbrace{P(\Re^p)}_{X} \mapsto \underbrace{P(\Re^q)}_{Y=A[X]}$$

Sammon's Algorithm
FFBP Neural Nets
SVMs (in theory !)

Feature Analysis Example : Iris Data

Each object vector represents an Iris flower

n = 150 vectors

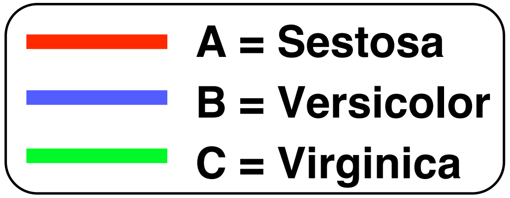
p = 4 features

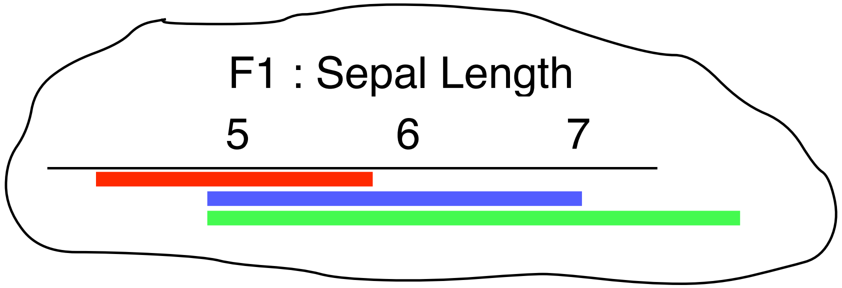
c = 3 *physical* subspecies

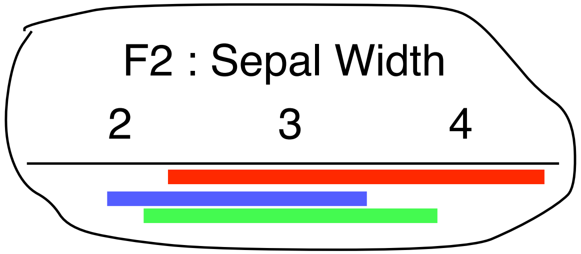
(n_i = 50 vectors each)

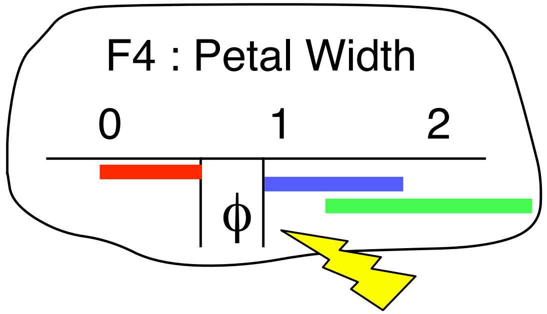
$$\mathbf{x}_k = \begin{pmatrix} x_{1k} \leftarrow \text{sepal length} \\ x_{2k} \leftarrow \text{sepal width} \\ x_{3k} \leftarrow \text{petal length} \\ x_{4k} \leftarrow \text{petal width} \end{pmatrix}$$

Visual Examination of Feature Ranges

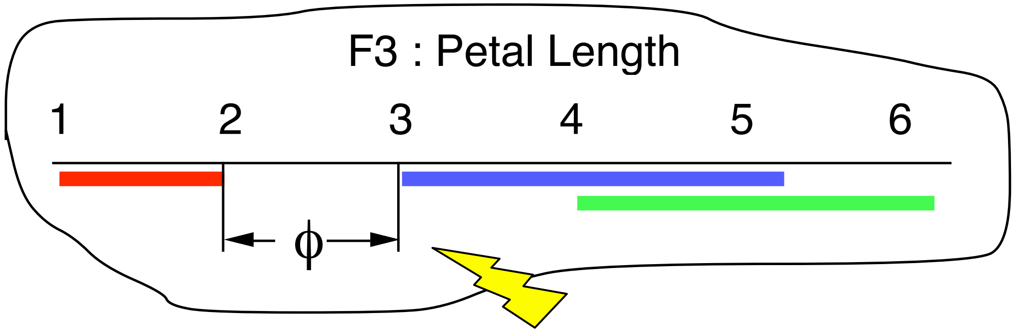








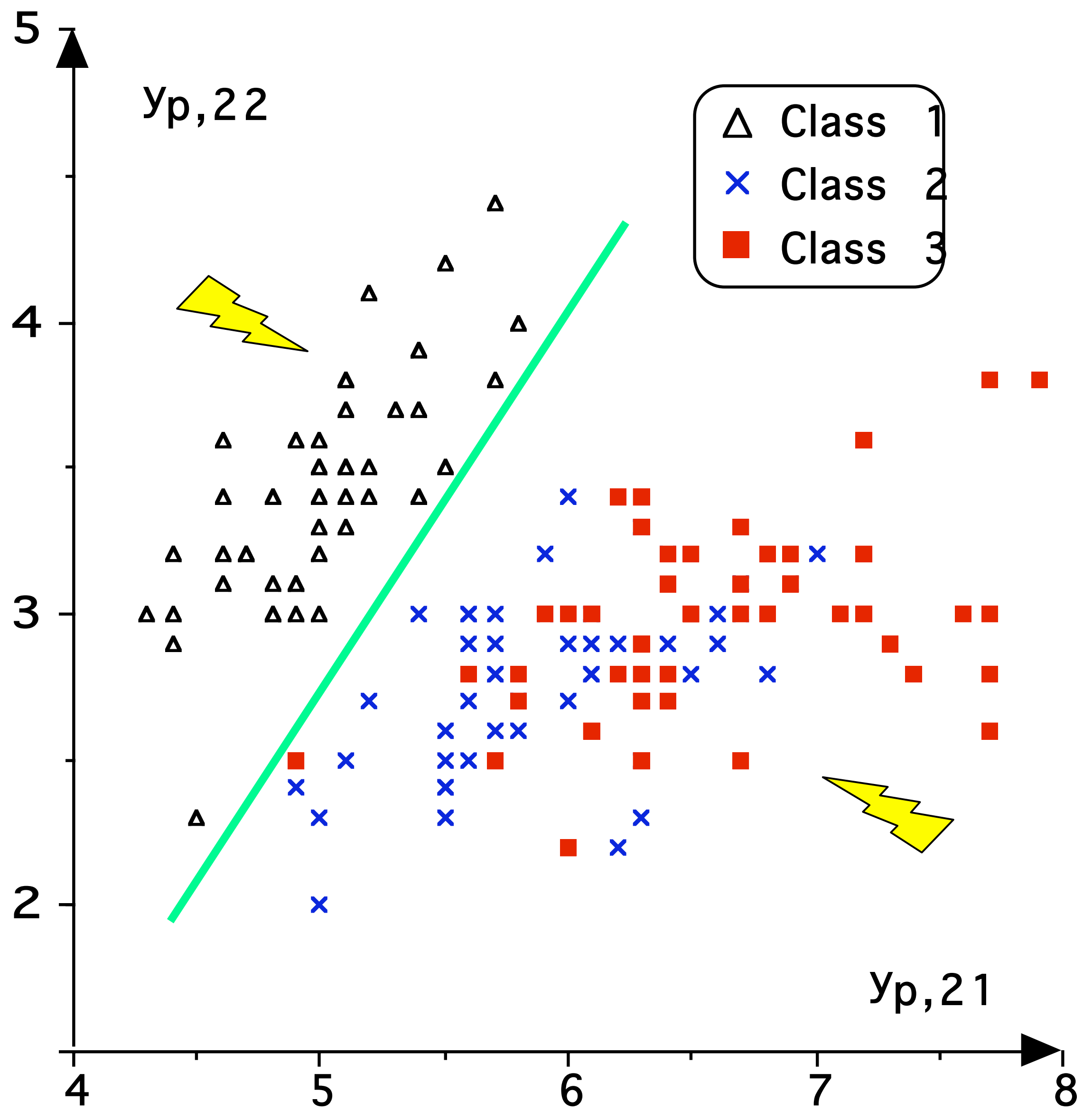
F3 or F4 separates A from BUC



Linear Selection (Projection)

F1 vs F2

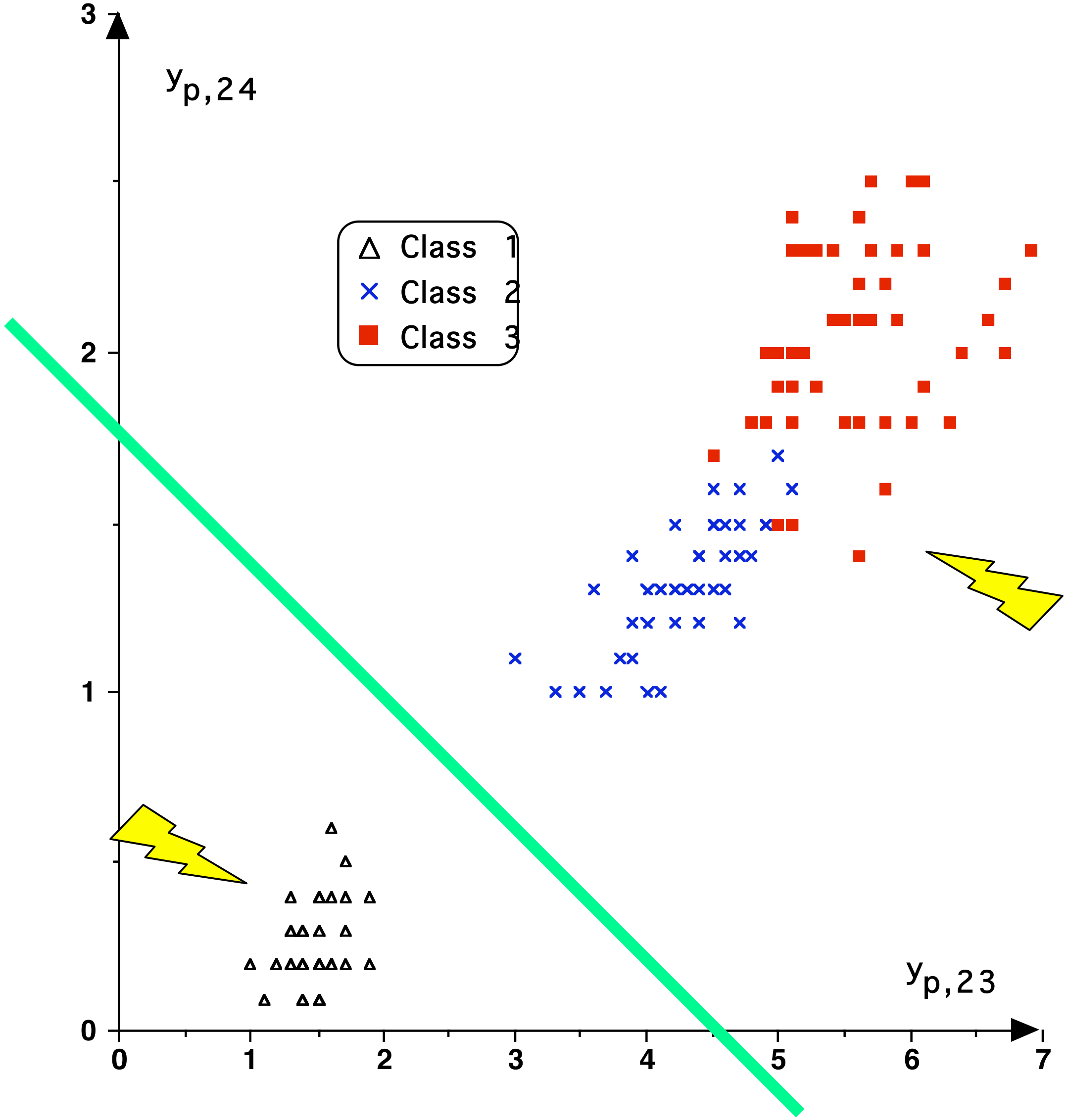
(F1, F2) *linearly* separates A from B∪C
------and------
suggests that Iris has only c = 2 *geometric* clusters



Linear Selection (Projection)

F3 vs F4

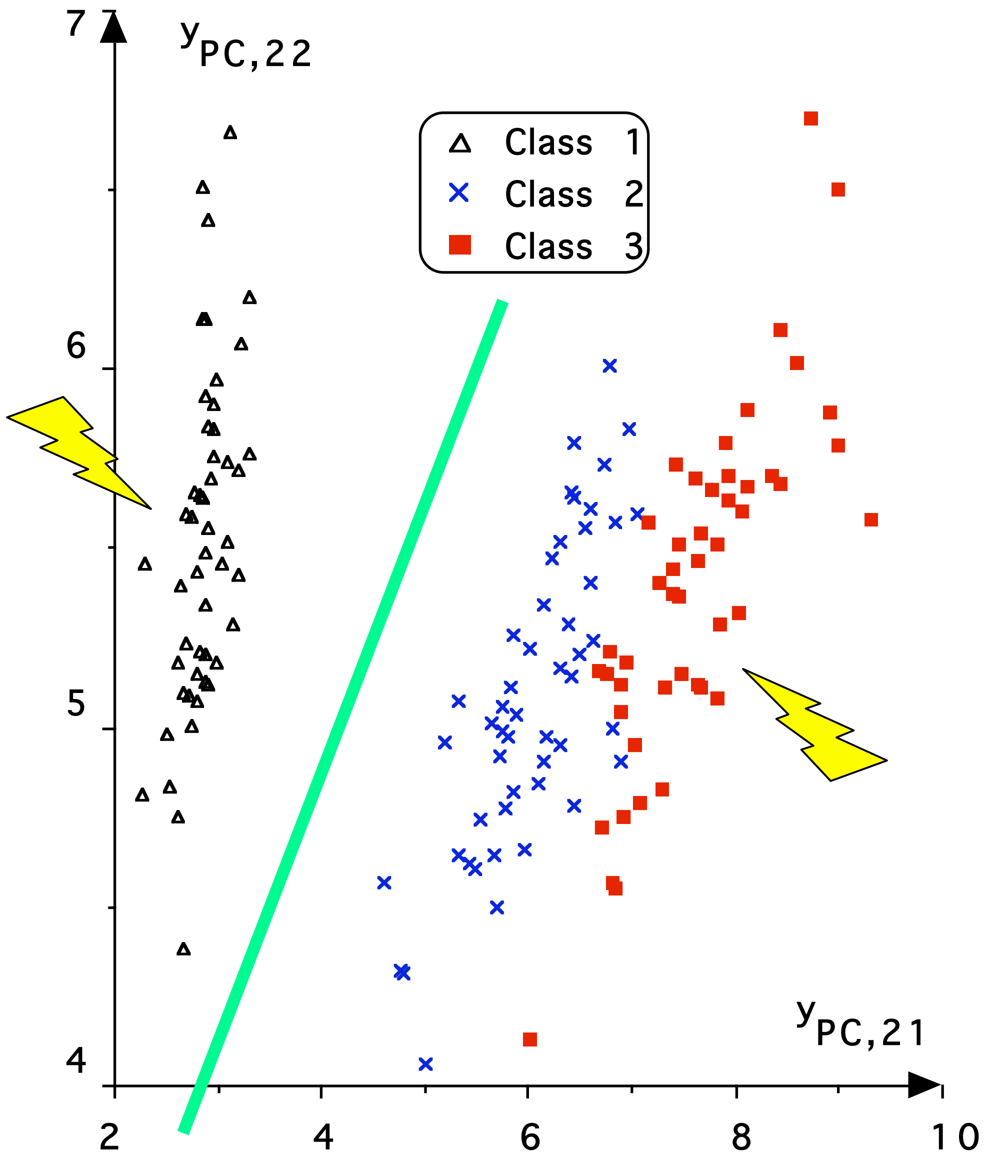
(F1, F2) *linearly* separates A from B∪C
------and------
suggests that Iris has only c = 2
geometric clusters



Linear Extraction (PCA Projection)

First 2 Principle Components (y_1, y_2)

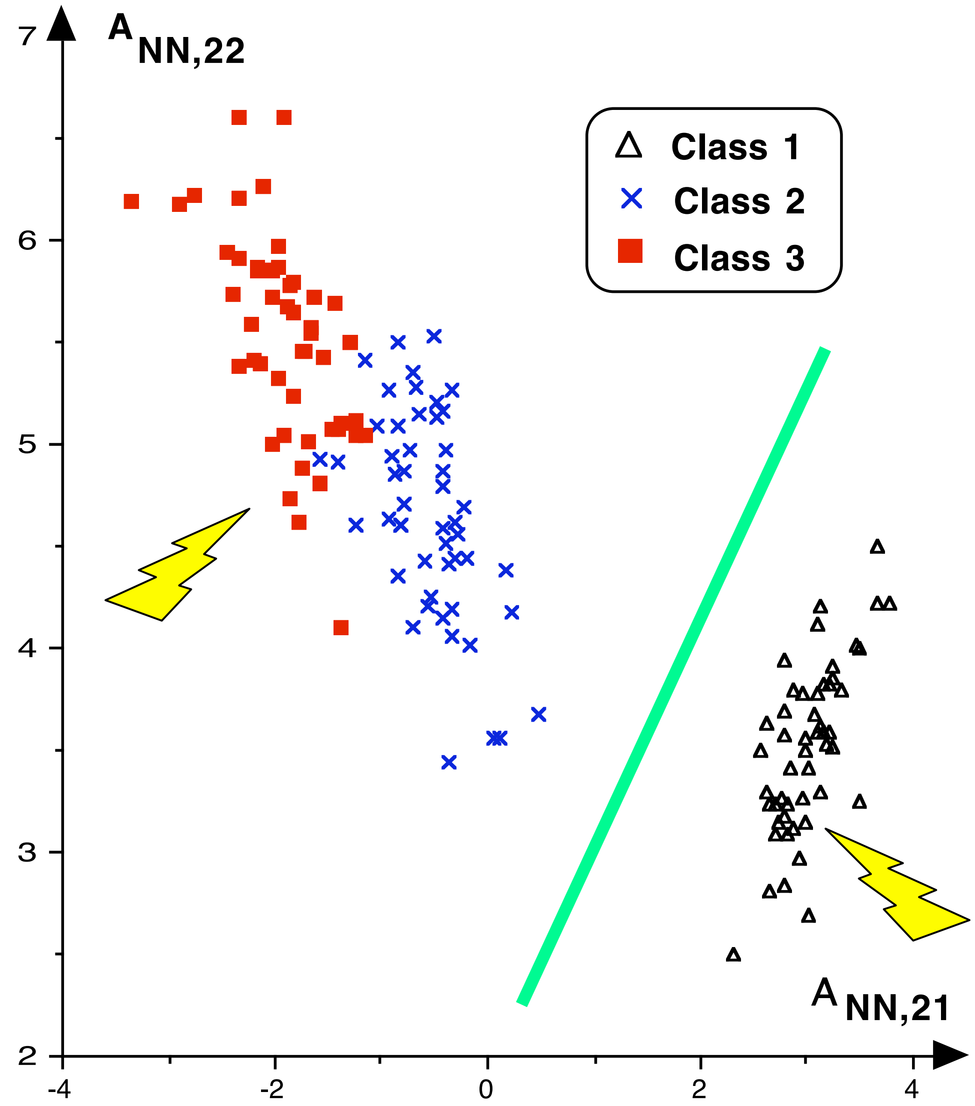
(y1, y2) *linearly* separates A from B∪C
------and------
suggests that Iris has only c = 2 *geometric* clusters



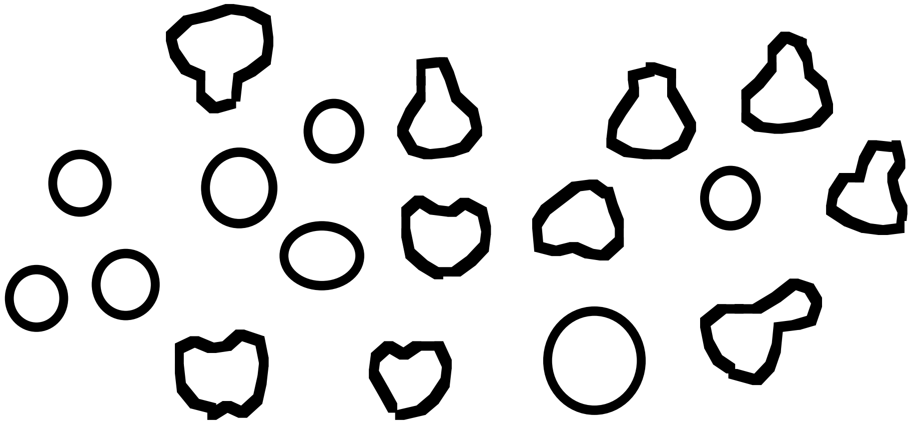
NL Extraction by 4:2:4 FFBP NN

Extracted feature pair (NN_{21}, NN_{22})

(NN_{21}, NN_{22})
linearly
separates A
from B∪C
------and------
suggests that Iris
has only c = 2
geometric clusters



Label the similar objects

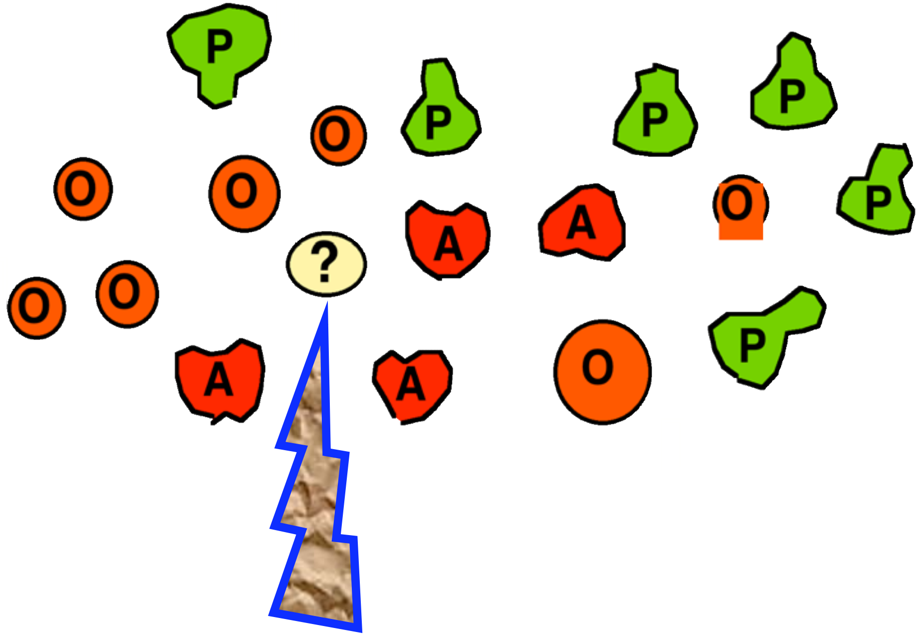


What is Clustering ?

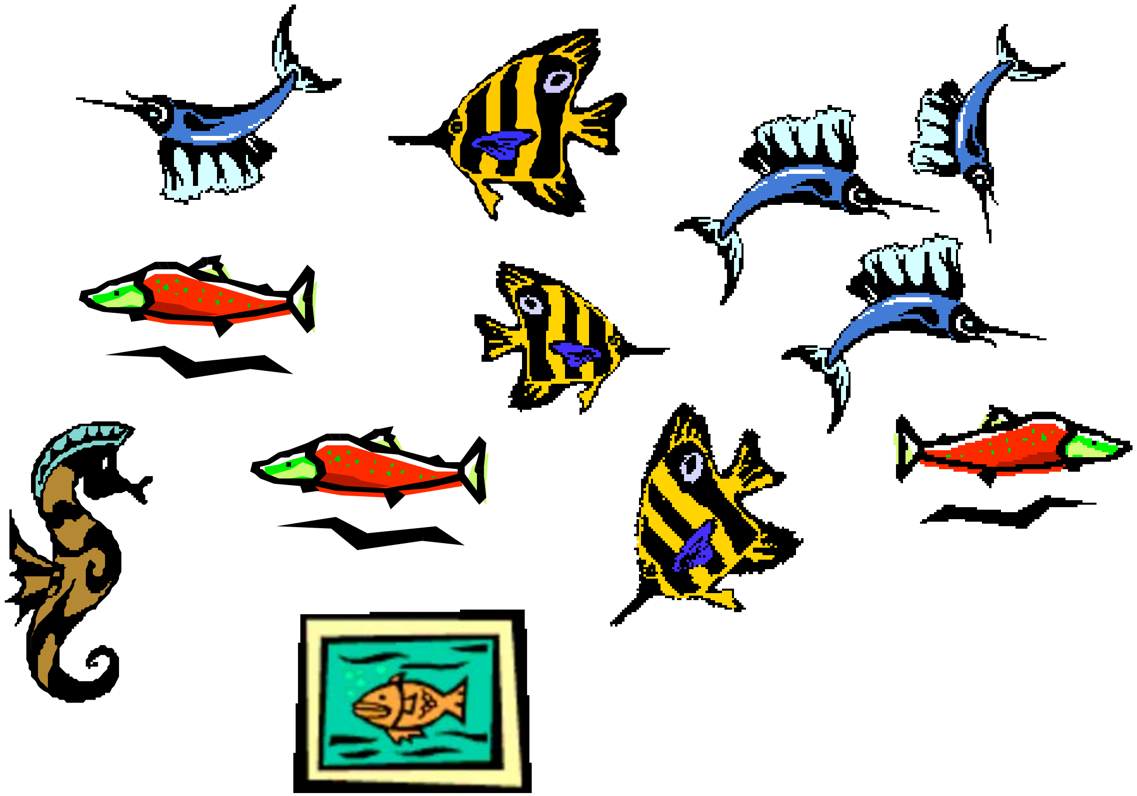
(Unsupervised Learning}

What is "similar"?

How many groups (c) ?



Data sets usually contain *mixtures* of objects

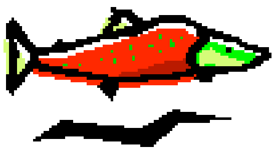


Clustering groups and labels objects

Easy for Humans !

Hard for computers !

Similarity Mistakes



What is "similar"?

Confusing Objects

Wrong features?

Mystery Data (Noise)

Incomplete data

Non-linear boundaries

How many groups (c)?

What kind of labels?

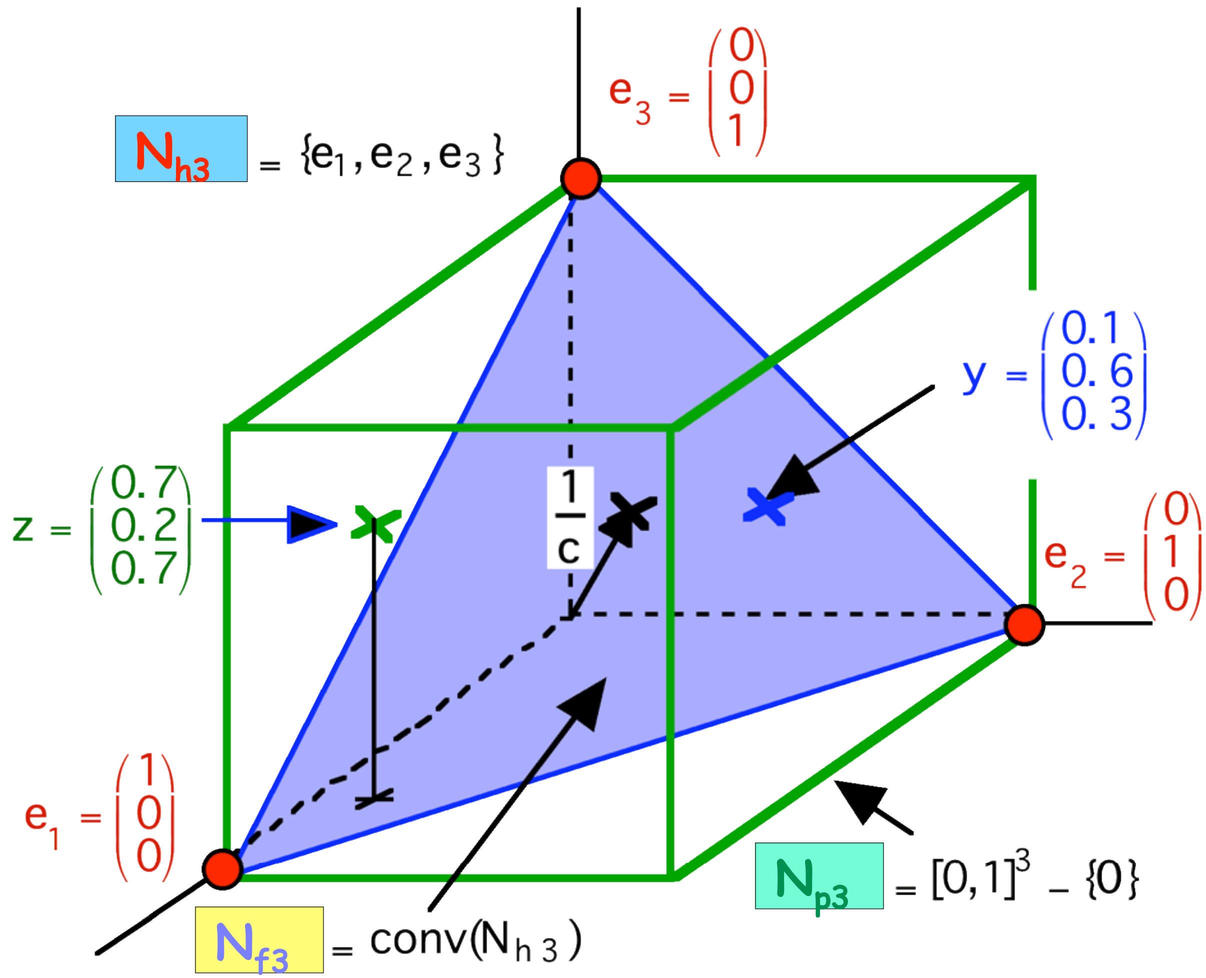
Yipes !

Unipolar Label Vectors @ c = 3

Crisp
= Vertices

Fuzzy or
Probabilistic
= Triangle

Possibilistic
= Cube-{θ}



Partition Matrices

row i ➠ M'ship of all o_k's in cluster i

o_1 o_k o_n *Objects*

$$U = \begin{bmatrix} u_{11} & \cdots & u_{1k} & \cdots & u_{1n} \\ \vdots & & \vdots & & \vdots \\ u_{i1} & \cdots & u_{ik} & \cdots & u_{in} \\ \vdots & & \vdots & & \vdots \\ u_{c1} & \cdots & u_{ck} & \cdots & u_{cn} \end{bmatrix}$$

clusters

col j ➠ M'ship of o_k in each cluster

Membership Functions

$u_i : O \rightarrow [0,1]$

$u_i(o_k) = u_{ik}$ = M'ship of o_k in cluster i

	Crisp	Fuzzy/Prob	Possibilistic
Row sums	$\sum_{k} u_{ik} > 0$	same	same
Col sums	$\sum_{i} u_{ik} = 1$	same	$\sum_{i=1}^{c} u_{ik} \leq c$
M'ships	$u_{ik} \in \{0,1\}$	$u_{ik} \in [0,1]$	same
Set Name	M_{hcn} $\subset$	M_{fcn} $\subset$	M_{pcn}
Example	$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{matrix}$	$\begin{matrix} 1 & .07 & 0 & .44 \\ 0 & .91 & 0 & .06 \\ 0 & .02 & 1 & .50 \end{matrix}$	$\begin{matrix} 1 & .07 & 1 & .44 \\ 0 & .91 & 0 & .52 \\ 0 & .02 & 1 & .38 \end{matrix}$

Take a 2nd 

Partition Set Theory

Non Degenerate — Degenerate (0 rows)

$$\underbrace{\left(0 < \sum_{k=1}^{n} u_{ik}\right)}_{M_{hcn} \subset M_{fcn}} \rightarrow \underbrace{\left(0 \le \sum_{k=1}^{n} u_{ik}\right)}_{M_{hcn0} \subset M_{fcn0}}$$

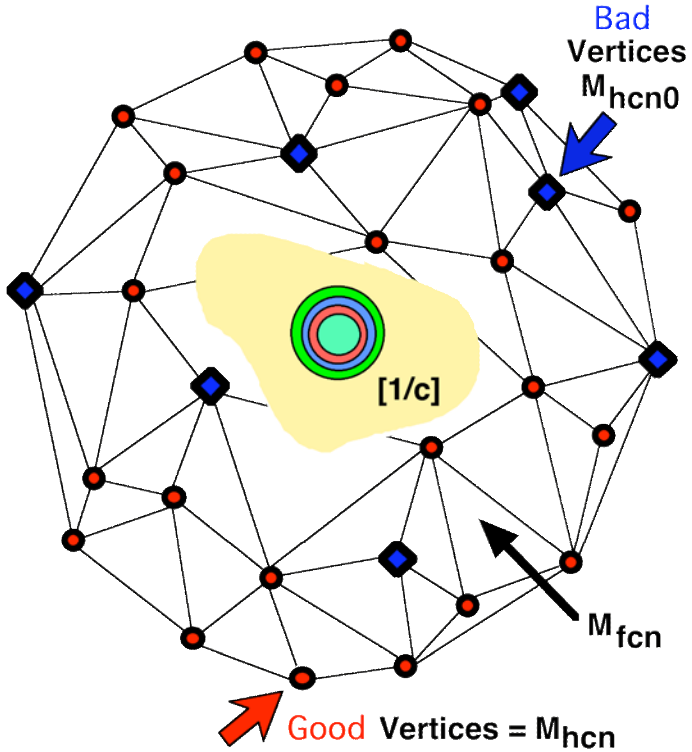
M_{fcn}=conv(M_{fcn0})=convex polytope

Each face of M_{fcn0} looks like N_{fc}

U =[1/c] = centroid of M_{fcn}

$$\left|M_{hcn}\right| = \left(\frac{1}{c!}\right)\sum_{j=1}^{c}\binom{c}{j}(-1)^{c-j} j^{n} \approx \left(\frac{c^{n}}{c!}\right)$$

dim(M_{fcn})=n(c-1)



Hardening Partitions

Defuzzification (Deprobabilization = Bayes Rule for $U \in M_{fcn}$!)

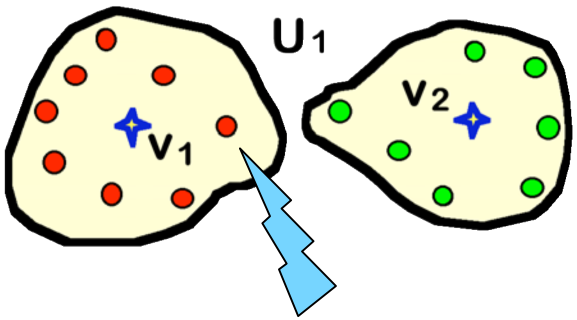
$$U = \begin{bmatrix} 1 & .07 & .6 & .5 \\ 0 & .81 & .2 & 0 \\ 0 & .12 & .7 & .5 \end{bmatrix} \in M_{pcn} \qquad H_1(U) = U_{MM} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in M_{hcn}$$

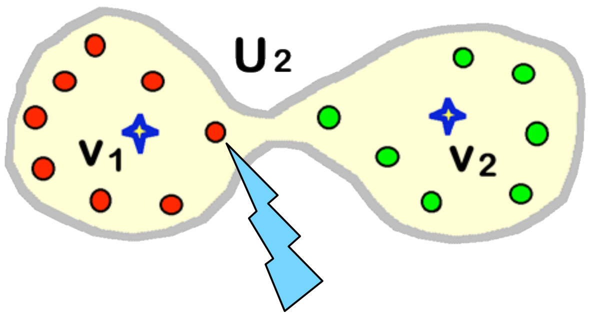
resolve ties randomly

α -cut thresholding (0< α <1) filters chosen levels of m'ship

$$H_{.9}(U) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in M_{hcn} \qquad H_{.8}(U) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in M_{hcn}$$

Clusters represented by partition matrix (U) *or* prototypes $V = \{v_k\}$





$$U_c = \begin{bmatrix} 1 & 1 & \ldots & 1 & 0 & 0 & \ldots 0 \\ 0 & 0 & \ldots & 0 & 1 & 1 & \ldots 1 \end{bmatrix}$$

$$U_f = \begin{bmatrix} .9 & 1 & \ldots & .6 & .3 & 0 & .2 & \ldots & 0 \\ .1 & 0 & \ldots & .4 & .7 & 1 & .8 & \ldots & 1 \end{bmatrix}$$

Crisp : hard boundaries

Fuzzy : soft boundaries

4 types of clustering models

U only models

V only models

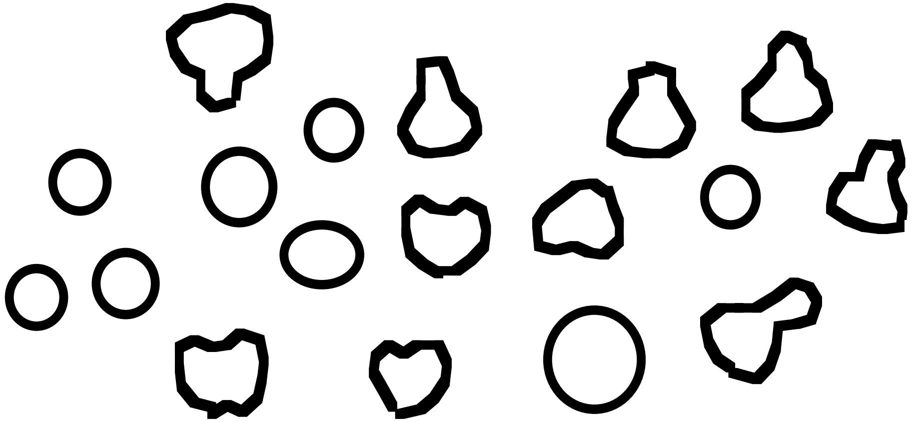
(U, V, +) models

(U, V) models

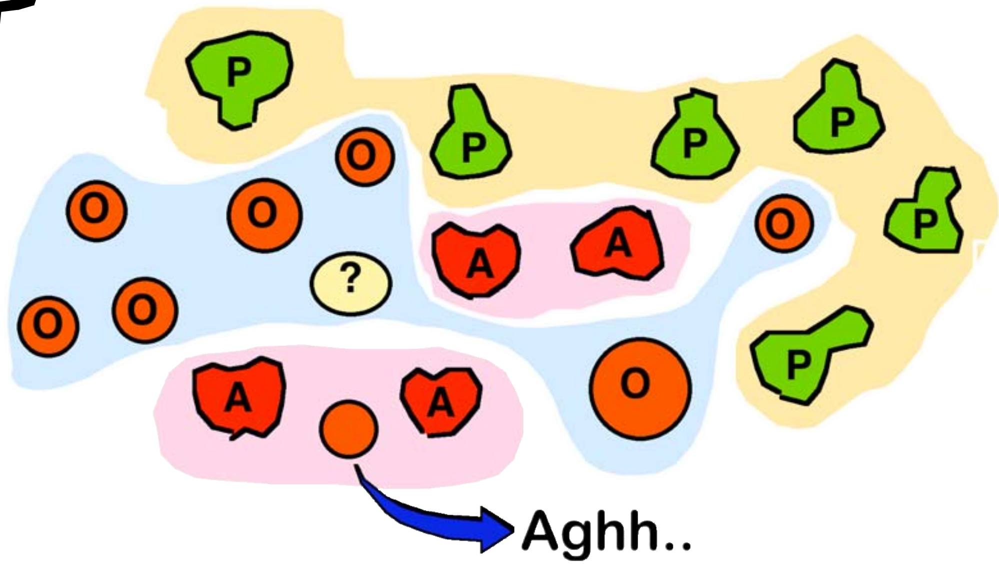
What is Classification ?

(Supervised Learning)

Find regions that contain similar objects

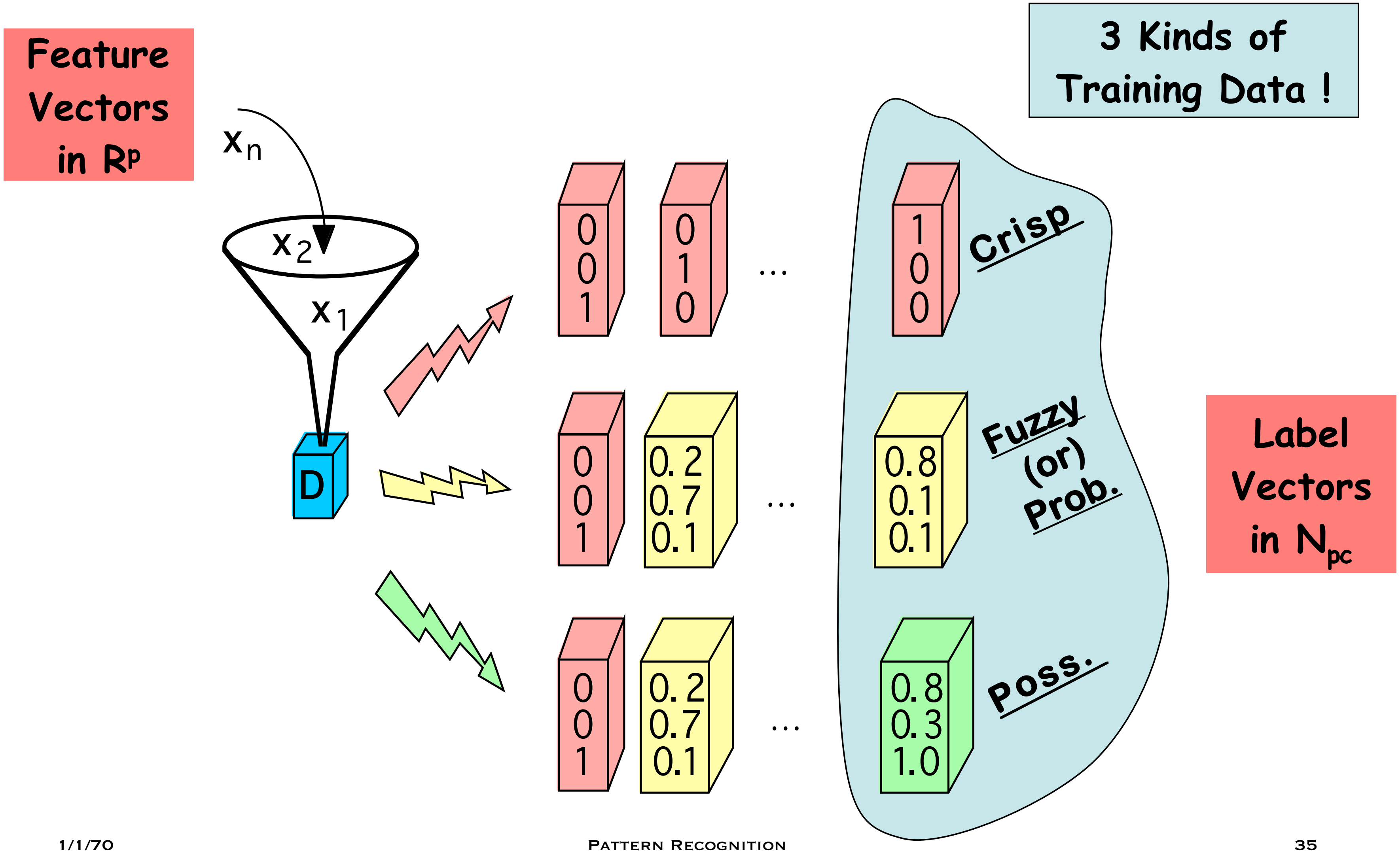


What are " (decision) regions"?



Classifier Functions

$D : \mathfrak{R}^p \mapsto N_{pc} = [0, 1]^c$



Classifier Training

1 Choose a family of models $\mathcal{F}$ = { D_1 (θ) … D_i (θ) … }

2 Training Data X_t = { $x_{t,1}$ … $x_{t,k}$ … $x_{t,n}$ }

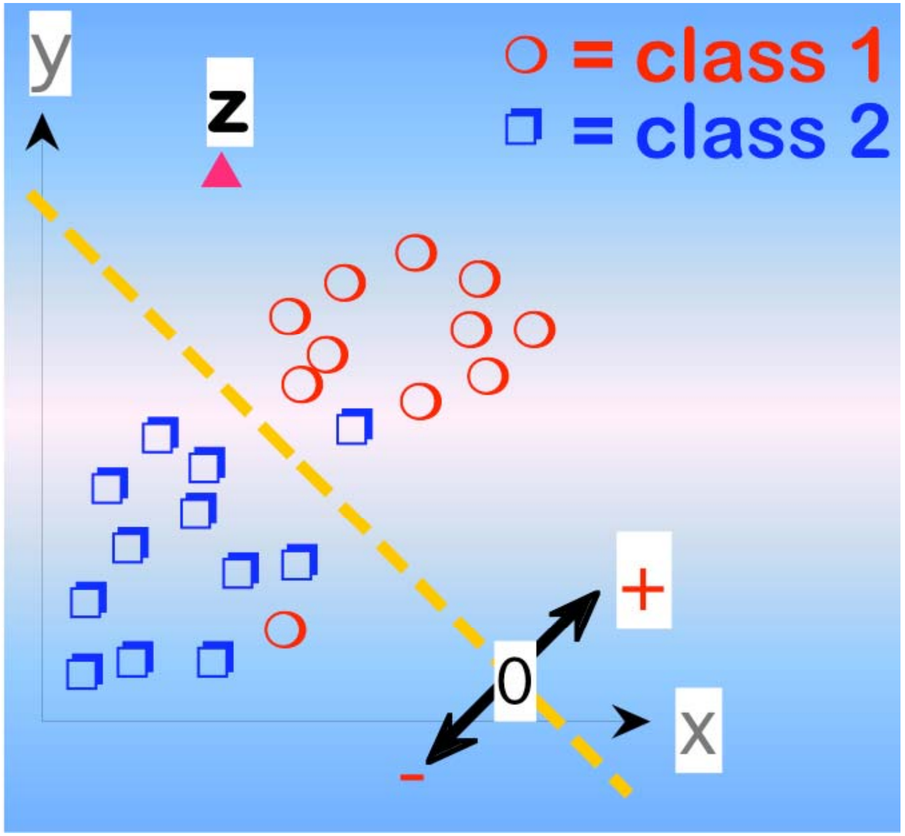
3 Training Labels U_t =

1	.4	.7
0	.6	.5

4 Algorithm A "looks for" an optimal (θ*) of D

5 Use (X, U_X, A) to find (θ*) of D ("learning")

$$\underbrace{d_{HP}(x) = \left\langle x, \vec{w} \right\rangle + \alpha}_{\text{(Linear) HP Model}} + \underbrace{X_t}_{\text{Data}} + \underbrace{U_t}_{\text{Labels}} + \underbrace{\text{Hyperplane}}_{\text{Algorithm}}$$



$$\theta = \left[\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}, \alpha \right]$$

$$\underbrace{d_{HP}(x) = x + y - 1}_{\text{(e.g. SVM) Classifier}}$$

$$\theta^* = \left[\begin{pmatrix} 1 \\ 1 \end{pmatrix}, -1 \right]$$

D^*_{HP} *crisply* labels *every* point in R^p

Classifier Function

$$D^*_{HP}(x) = \begin{cases} e_1 = (1 \quad 0)^T & ; d_{HP}(x) > \alpha^* \quad \Rightarrow x \in 1 \\ e_1 \text{ or } e_2 & ; d_{HP}(x) = \alpha^* \quad \Rightarrow \text{Tie} \\ e_2 = (0 \quad 1)^T & ; d_{HP}(x) < \alpha^* \quad \Rightarrow x \in 2 \end{cases}$$

Decision (or Discriminant) function

Mistakes can be costly : a doctor treats patient x with A or B

Use $(A|x \in 1) \Rightarrow$ cures x

Use $(B|x \in 1) \Rightarrow$ no cure, no harm

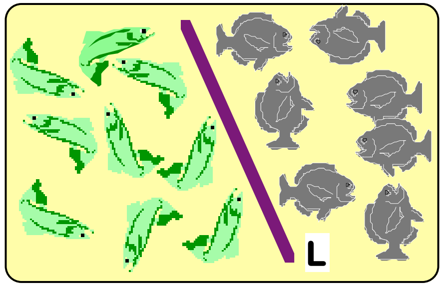
Use $(B|x \in 2) \Rightarrow$ cures x

Use $(A|x \in 2) \Rightarrow$  - a cure?

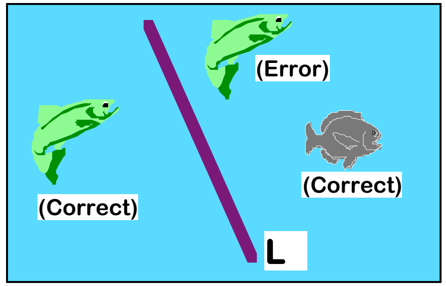
Patient x *wants* D^*_{HP} to be *soft*

$$D^*_{HP}(x) = \begin{cases} \begin{pmatrix} 0.95 \\ 0.05 \end{pmatrix} \Rightarrow \begin{pmatrix} \text{use} \\ A \end{pmatrix} \\ \begin{pmatrix} 0.53 \\ 0.47 \end{pmatrix} \Rightarrow \begin{pmatrix} \text{more} \\ \text{tests} \end{pmatrix} \end{cases}$$

Non-Adaptive
Off Line Training



Non-Adaptive
On Line Operation



3 Kinds of Training and Operation

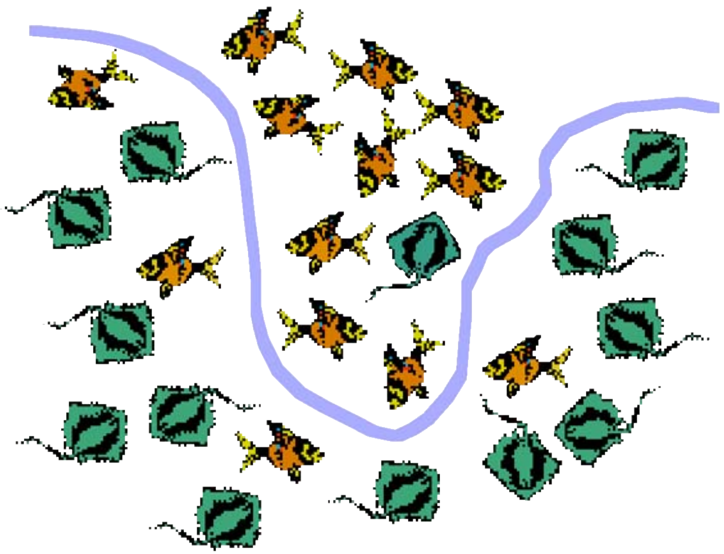
3 Data Sets needed to Train and Test a Classifier

Training $X_{tr} = \{x_{tr,1} \dots x_{tr,n}\}$ with labels $u_{tr} \in N_{pc}$

Validation $X_v = \{x_{va,1} \dots x_{va,s}\}$ with labels $u_{va} \in N_{hc}$

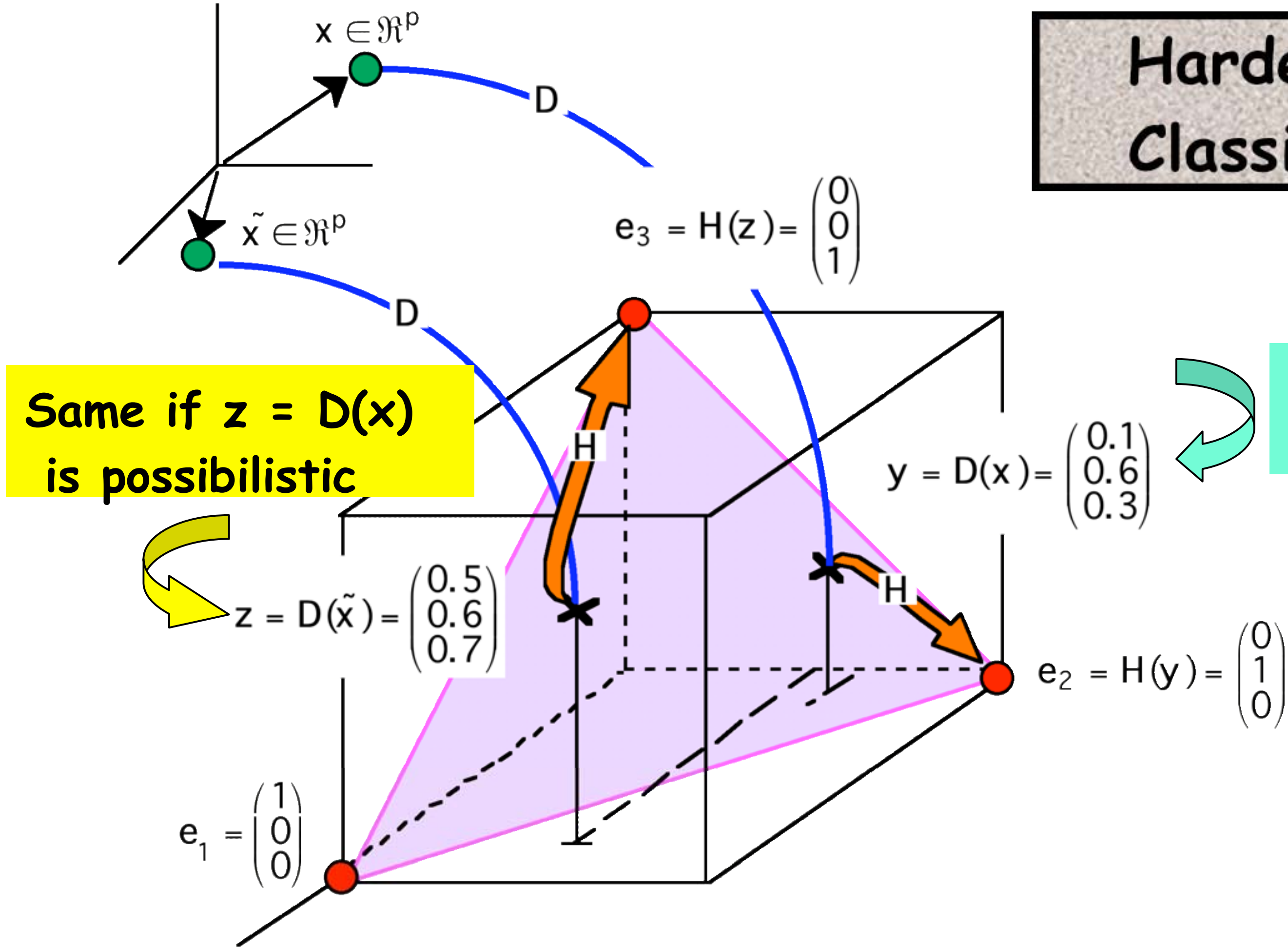
Test $X_t = \{x_{te,1} \dots x_{te,q}\}$ with labels $u_{te} \in N_{hc}$

Data can be uncooperative !



- Non-Linear Clusters
- Noise & Outliers
- Mixtures of Points
- High Dimensional
- Wrong Features

Hardening SOFT Classifier Labels



y = D(x) = fuzzy
(or probabilistic)

$y \in i$
$\Updownarrow$
$y_i > y_k$

(or)

$$H(y) = e_i \Leftrightarrow \left\| y - e_i \right\| = \min_k \left\{ \left\| y - e_k \right\| \right\}$$

Error rate estimation

$D(X)$ = classifier learned by applying algorithm A to X

$U_{D(X)}$ = (possibly) soft labels given to X by classifier D

$H(U_{D(X)})$ = crisp labels given to X by hardening $U_{D(X)}$

U_X = (known) crisp labels of data set X (X_{va} or X_{te})

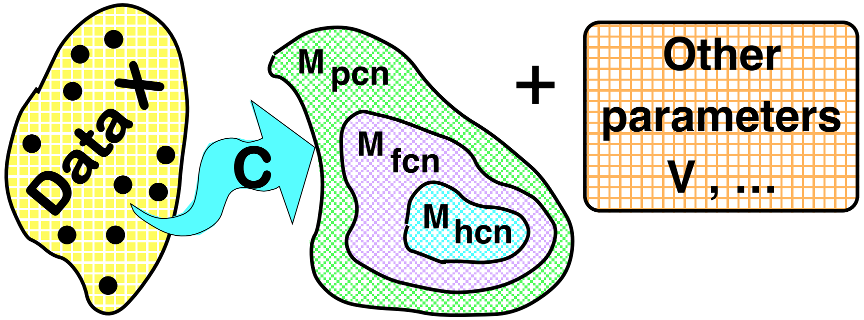
Empirical Error Rate (generalization error)

$$E_D(X) = \left(\frac{\#\ \text{wrong}}{\#\ \text{tried}} \right) = \frac{\left\| U_X - H(U_{D(X)}) \right\|}{2\,|X|}$$

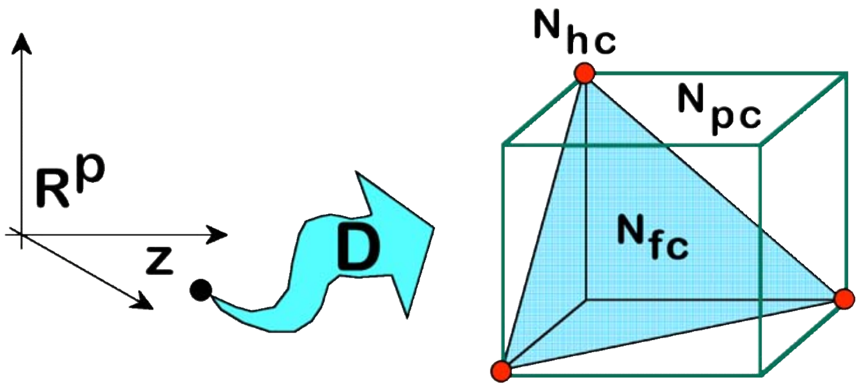
Minimize E_D iteratively by learning with X_{tr} and then computing $E_D(X_{va})$

Estimate E_D ONCE by computing $E_D(X_{te})$

Clustering vs Classification : Summary



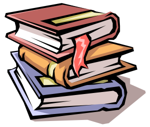
Clustering labels *only* the submitted data



Classifiers can label *all* the data in their domain space

¡ Building a PR system - the *big*

Process Description	Feature Analysis	Cluster Analysis	Classifier Design
Model Type	**Operation**	**Criterion**	**Model Basis**
Numerical Rule-Based ⋮ Syntactic	Extraction Filtering ⋮ 2D-Displays	Dissimilarity Connectivity ⋮ Frequency	Deterministic Fuzzy ⋮ Statistical
Data Needed	**Model + Algorithm**	**Model + Algorithm**	**Model + Algorithm**
Type Structure Nomination Collection ⋮ Sensors	Projection PCA/LDA Sammon SOMs ⋮ VAT-coVAT	c Means SAHN Entropy GMD ⋮ ART NN	FFBP NN 1-np rules K-nn rules SVMs ⋮ Bayes Quad.



Numerical Pattern Recognition

Duda, R. and Hart, P. (1973). Pattern Classification and Scene Analysis, Wiley Interscience, NY, 1973.

Devijver, P. and Kittler, J., Pattern Recognition: A Statistical Approach, Prentice-Hall, Englewood Cliffs, NJ, 1982.

Theodoridis, S. and Koutroumbas, K. (2003). Pattern Recognition, 2nd ed., Academic Press, NY.

Bezdek, J. C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, NY.

Bezdek, J. C. and Pal, S. K. (1992). Fuzzy Models for Pattern Recognition, IEEE Press, Piscataway, NJ.

Bezdek, J. C., Keller, J. M., Krishnapuram, R. and Pal, N. R. (1999). Fuzzy models and algorithms for Pattern Recognition and Image Processing, Springer, NY.

Machine Learning & SVMs

Scholkopf, B. and Smola,A.J. (2002). Learning with Kernels, MIT Press, Boston

Vidyasagar, M. (2003). Learning and Generalization, Springer, London.

Devroye, L., Györfi, L. and Lugosi, G. (1996). A Probabilistic Theory of Pattern Recognition, Springer, NY,

Syntactic Pattern Recognition

Pavlidis, T.(1980) , Structural pattern recognition, Springer-Verlag, New York.

Fu, K. S. (1982). Syntactic Pattern Recognition and Applications, Prentice Hall, Inc, New Jersey.

Bunke, H., (ed.) (1992) Advances in Structural and Syntactic Pattern Recognition, World Scientific Press, Singapore.