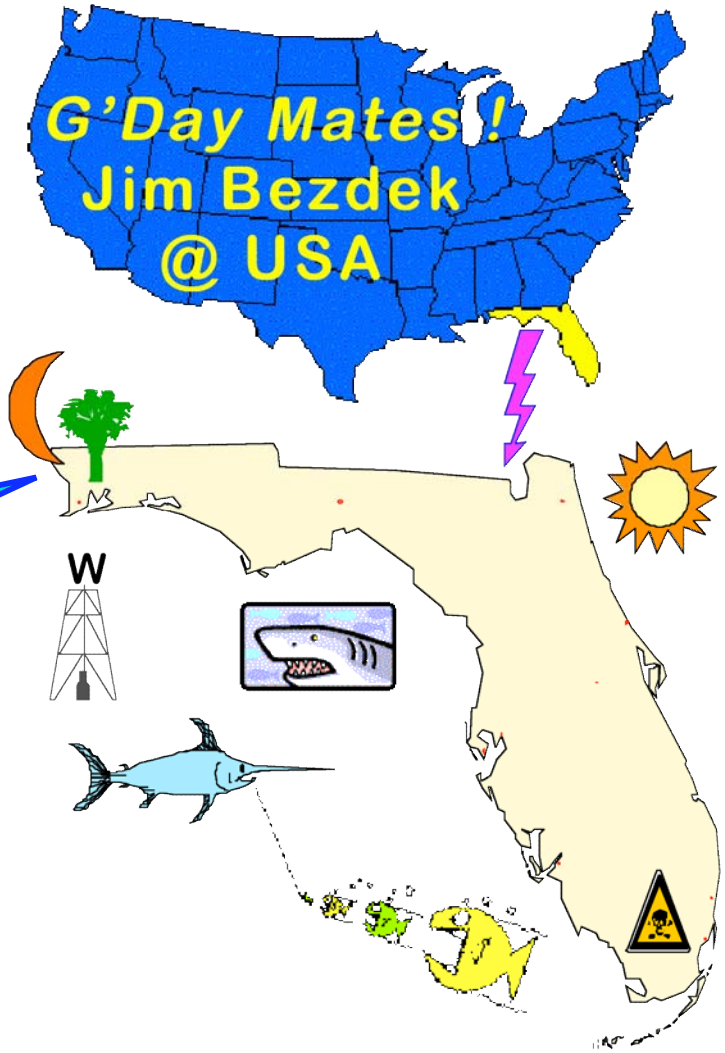
I'm from ...

University of West Florida

Pensacola



Pensacola before Hurricane IVAN

Pensacola
after 9/16/04

Approximate Clustering in Very Large Data Sets

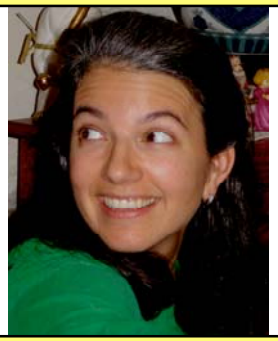
eFFCM - VL Image Data

Jim Bezdek Nik Pal

geFFCM/geFEM - VL Feature Vector Data

Jim Bezdek Rick Hathaway

eNERF - VL Relational Data



Chris Leckie Rao Kotagiri Jackie Huband Jim Bezdek Rick Hathaway

Today's Talk

1. Cluster Analysis
2. Very Large Data
3. Progressive Sampling and Extension
4. eFFCM --------> VL Image Data
5. geFFCM -------> VL Object Data
6. eNERF ----> VL Relational Data

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 : $1 \le j \le p$

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)

We assume *(R = D)*

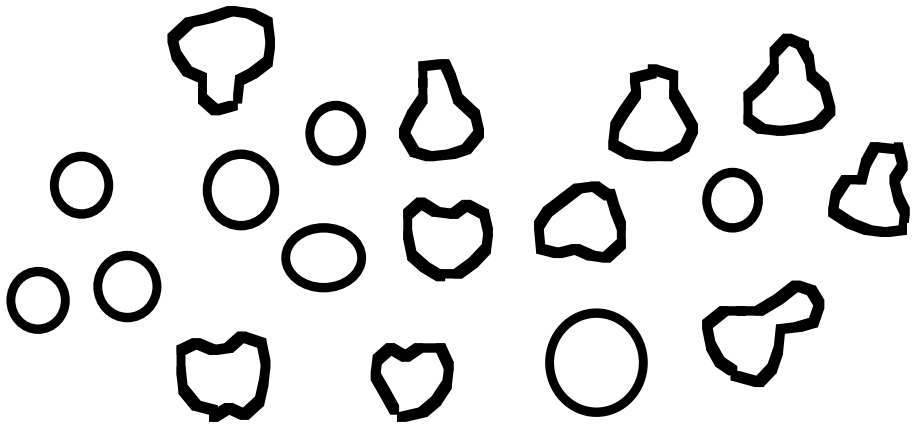
(i) $d_{ij} \ge 0$ $\quad 1 \le i \ne j \le n$

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

(iii) $d_{ij} = d_{ji}$ $\quad 1 \le i \ne j \le n$

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

Label the similar objects

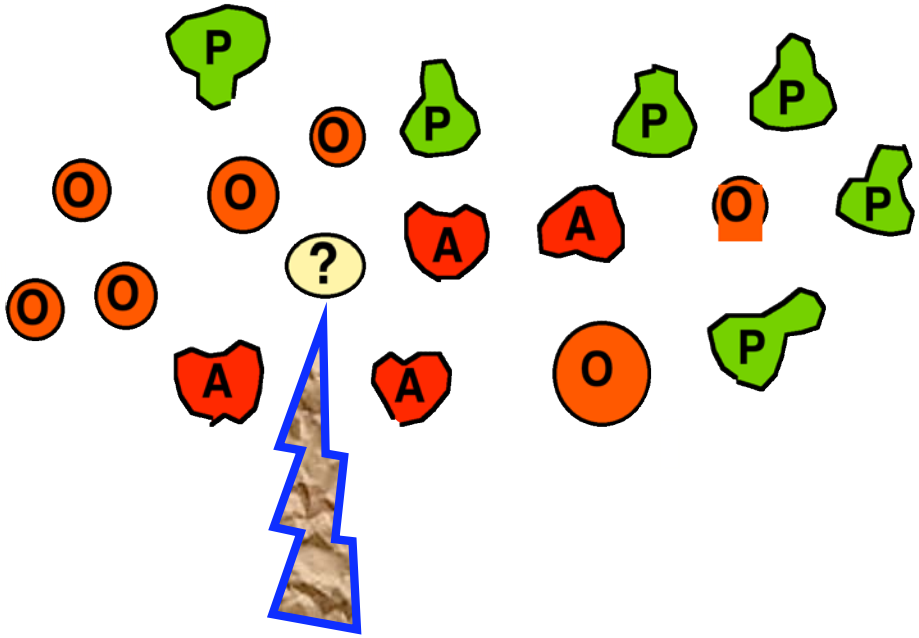


What is Clustering ?

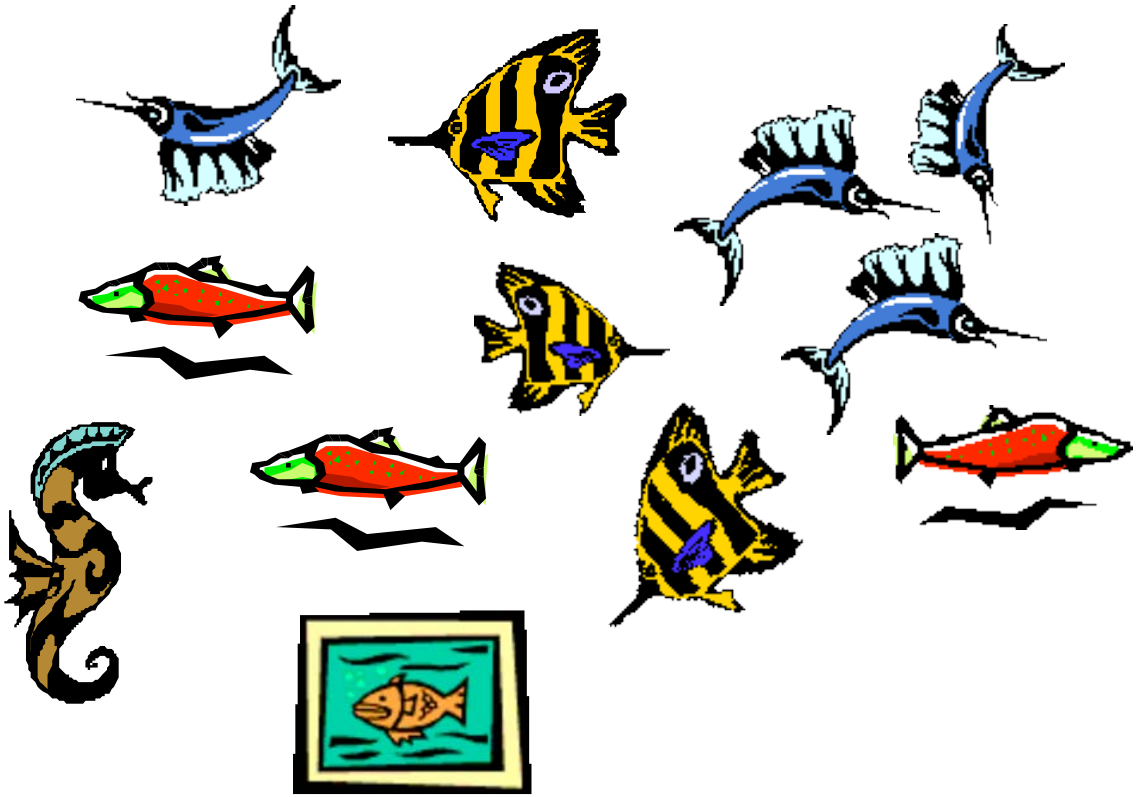
(Unsupervised Learning)

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?

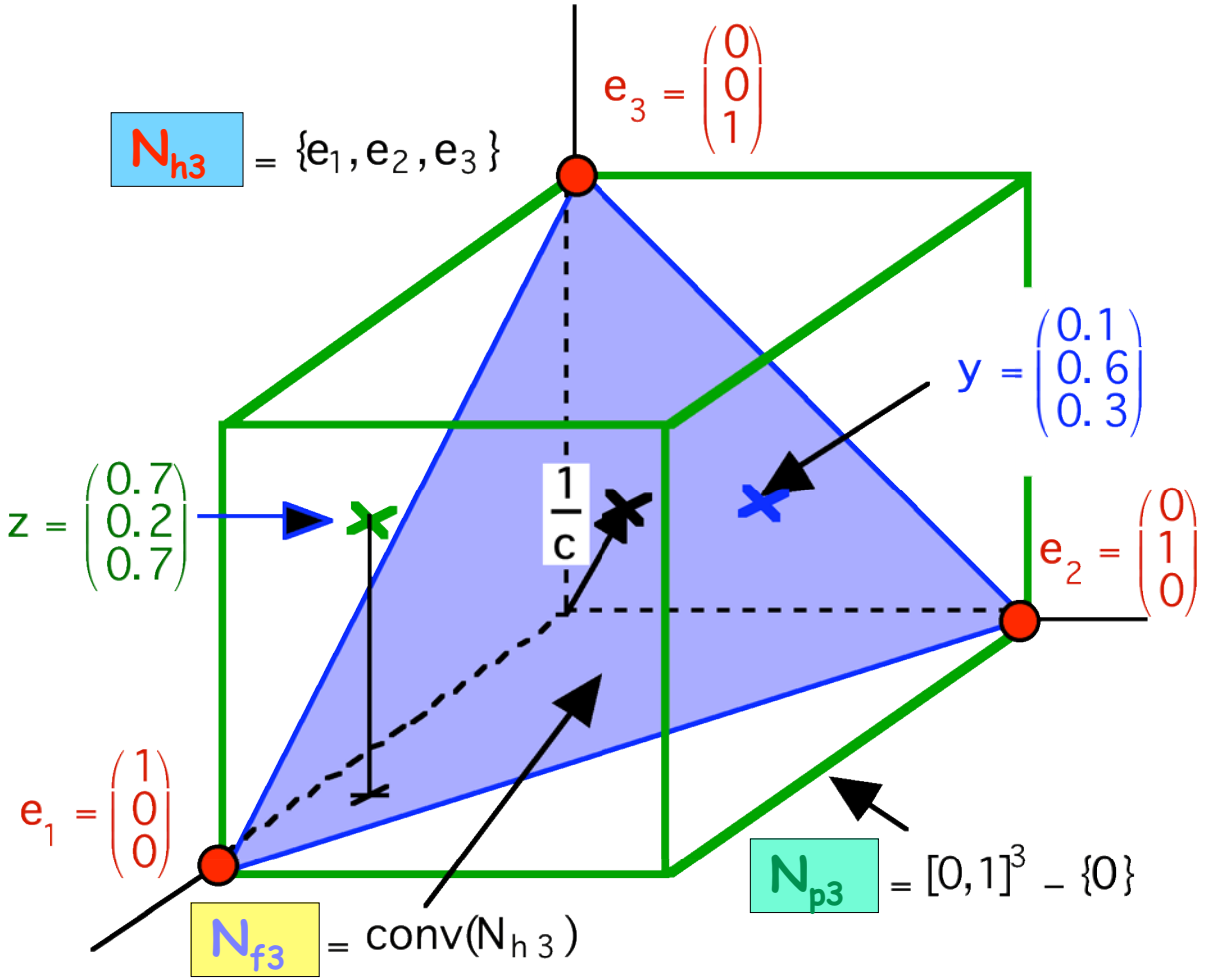
Yipes !

Unipolar Label Vectors @ c = 3

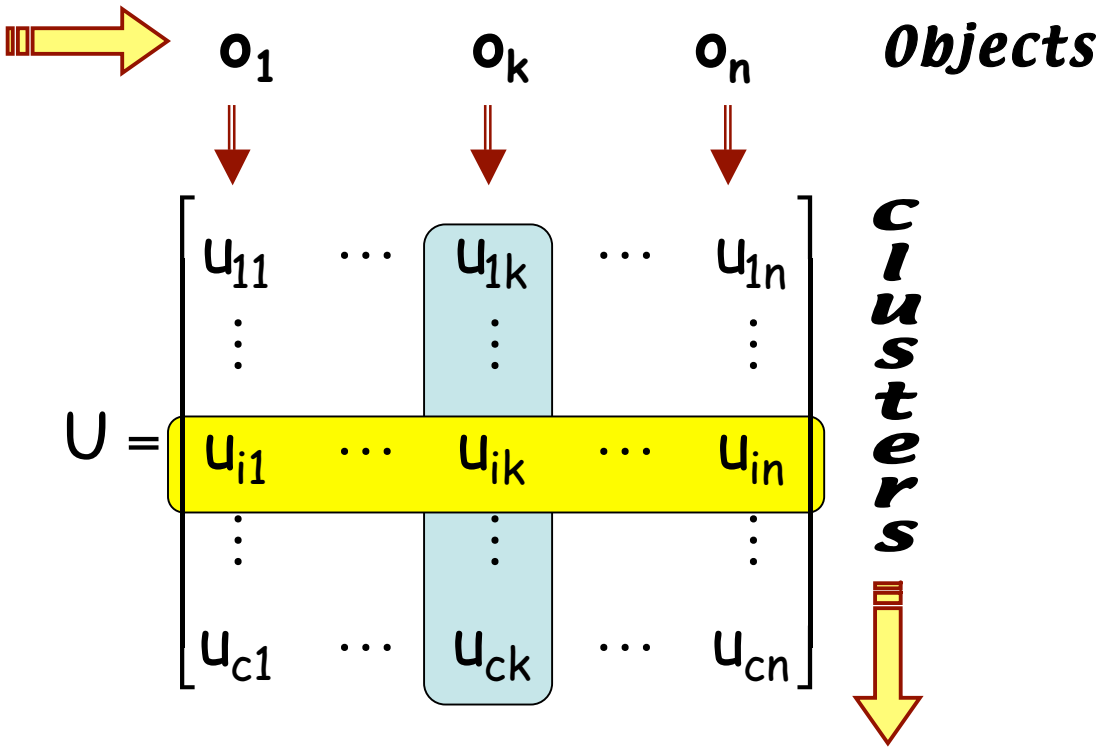
Crisp = Vertices

Fuzzy or Probabilistic = Triangle

Possibilistic = Cube-{θ}



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



col j ➠ M'ship of o_k in each cluster

Partition Matrices

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	$\sum_{i=1}^{c} u_{ik} \le c$
M'ships	$u_{ik} \in \{0,1\}$	$u_{ik} \in [0,1]$	same
Set Name	M_{hcn}	$\subset$ M_{fcn} $\subset$	M_{pcn}
Example	1 0 0 0 0 1 0 0 0 0 1 1	1 .07 0 .44 0 .91 0 .06 0 .02 1 .50	1 .07 1 .44 0 .91 0 .52 0 .02 1 .38

Take a 2nd

Hardening Partitions

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

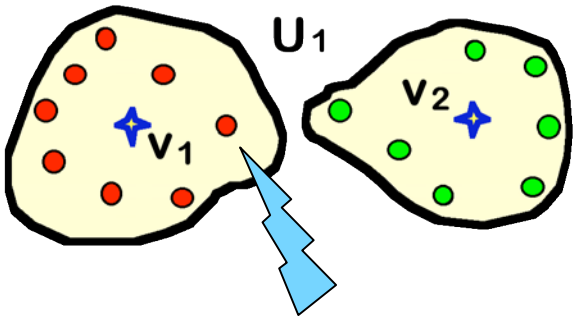
$$U = \begin{bmatrix} 1 & .07 & .6 & .5 \\ 0 & .81 & .2 & 0 \\ 0 & .12 & .7 & .5 \end{bmatrix} \in M_{pcn} \qquad H_1(U) = U_{MM} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in M_{hcn}$$

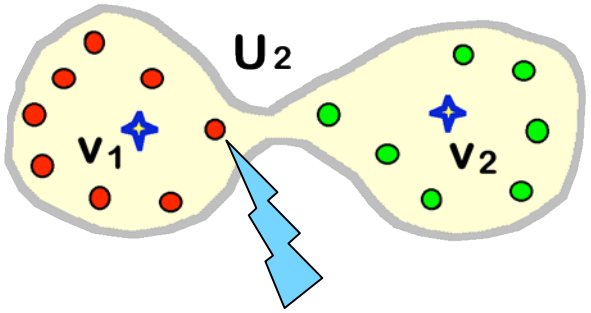
resolve ties randomly

α -cut thresholding ($0 < \alpha < 1$) 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 = $\{v_k\}$





$$U_c = \begin{matrix} 1 & 1 & \dots & 1 & 0 & 0 & \dots 0 \\ 0 & 0 & \dots & 0 & 1 & 1 & \dots 1 \end{matrix}$$

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

Crisp : hard boundaries

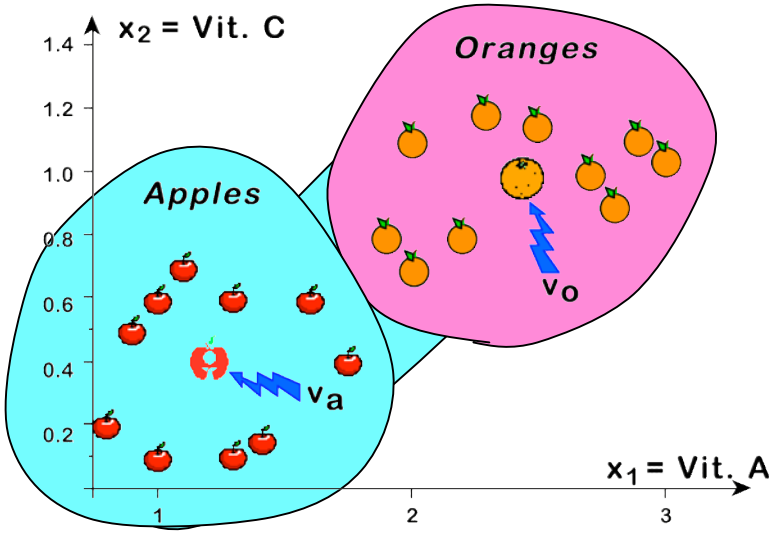
Fuzzy : soft boundaries

The c-Means models : one way to find U's and v's (for object data)

$$\underset{\substack{U \in M_{fcn} \\ V \in \Re^{cp}}}{\text{minimize}} \left\{ J_m(U,V) = \sum \sum u_{ik}^m \left\| x_k - v_i \right\|^2 \right\}$$

			Init	Init	HCM	HCM	FCM	FCM
x	x_1	x_2	U_{10}	U_{20}	U_1	U_2	U_1	U_2
1	1.00	0.60	1	0	1	0	0.97	0.03
2	1.75	0.40	1	0	1	0	0.77	0.23
3	1.30	0.10	1	0	1	0	0.96	0.04
4	0.80	0.20	1	0	1	0	0.94	0.06
5	1.10	0.70	1	0	1	0	0.95	0.05
6	1.30	0.60	1	0	1	0	0.97	0.03
7	0.90	0.50	1	0	1	0	0.96	0.04
8	1.60	0.60	1	0	1	0	0.84	0.16
9	1.40	0.15	1	0	1	0	0.95	0.05
10	1.00	0.10	1	0	1	0	0.95	0.05
11	2.00	0.70	1	0	0	1	0.33	0.67
12	2.00	1.10	1	0	0	1	0.19	0.81
13	1.90	0.80	1	0	0	1	0.39	0.59
14	2.20	0.80	1	0	0	1	0.10	0.90
15	2.30	1.20	1	0	0	1	0.04	0.96
16	2.50	1.15	1	0	0	1	0.01	0.99
17	2.70	1.00	0	1	0	1	0.01	0.99
18	2.90	1.10	0	1	0	1	0.05	0.95
19	2.80	0.90	0	1	0	1	0.03	0.97
20	3.00	1.05	0	1	0	1	0.06	0.94
	$\bar{v}_1$	$\bar{v}_2$	$v_{1,0}$	$v_{2,0}$	$v_{1,f}$	$v_{2,f}$	$v_{1,f}$	$v_{2,f}$
	1.21	2.43	1.57	2.85	1.21	2.43	1.21	2.50
	0.39	0.98	0.61	1.01	0.39	0.98	0.41	1.00

X = 20 (unlabeled) points in R^2



$H(U_{FCM}) = H_{MM} = U_{HCM}$

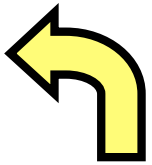
Not always true !

$V_{AVG} = V_{HCM} \sim V_{FCM}$

Very Large (VL) Data

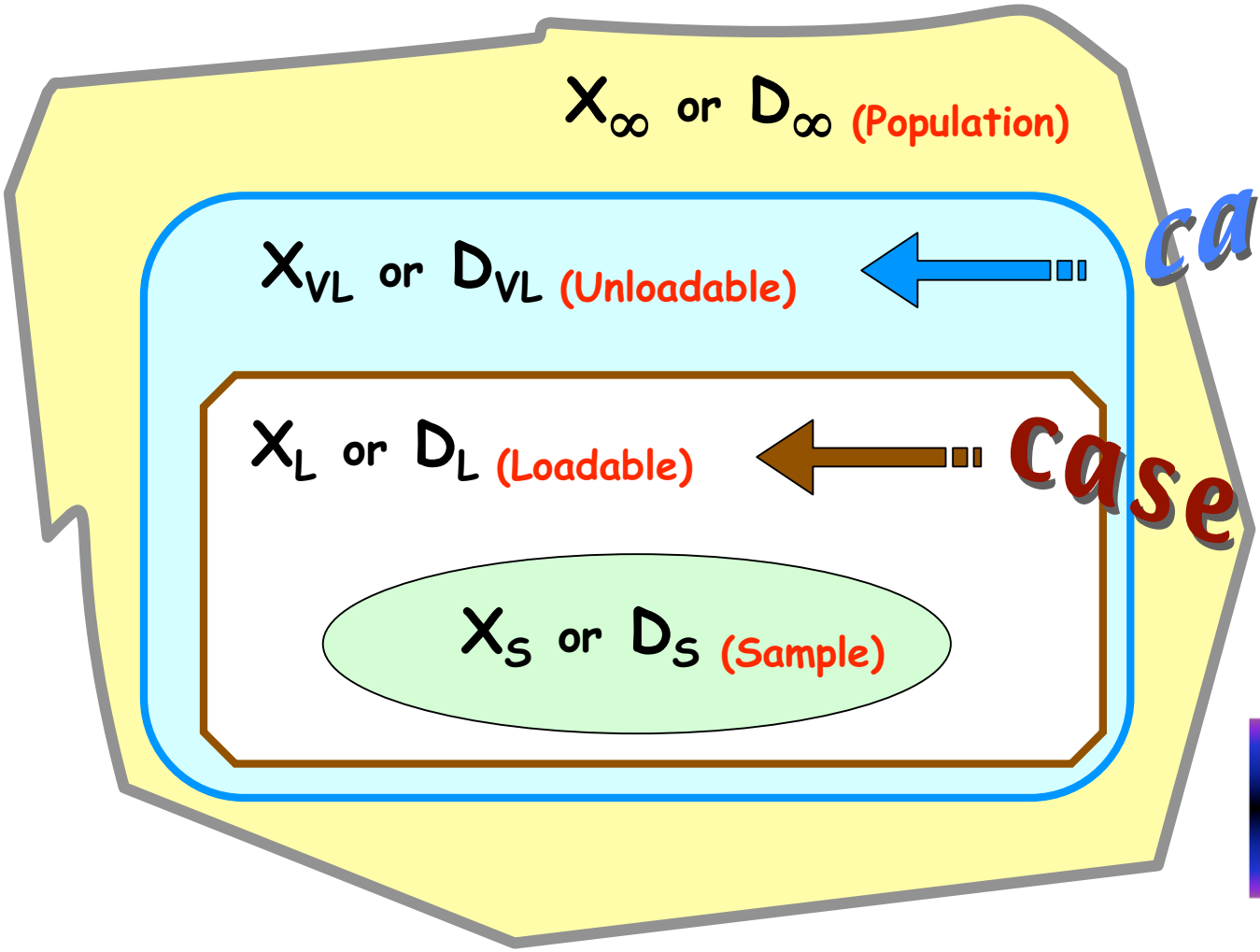
How Big is (VL) Data ?

Bytes	"Data Size"
10^2	Tiny
10^4	Small
10^6	Mega - medium
10^9	Giga - large
10^{10}	Huge
10^{12}	Tera - Monster
$10^{>12}$	Very Large (VL)



We can't cluster data this big (in a single computer) : VL is "unloadable"… so …

… we build a "cluster-friendly" (loadable) sample



2 Sampling Sub-Cases

case 1

If X_L or D_L is *Loadable* we can *Quantitatively* Compare

Lit-Clus[$*_L$] ↔ Approx-Clus[$*_L$]

case 2

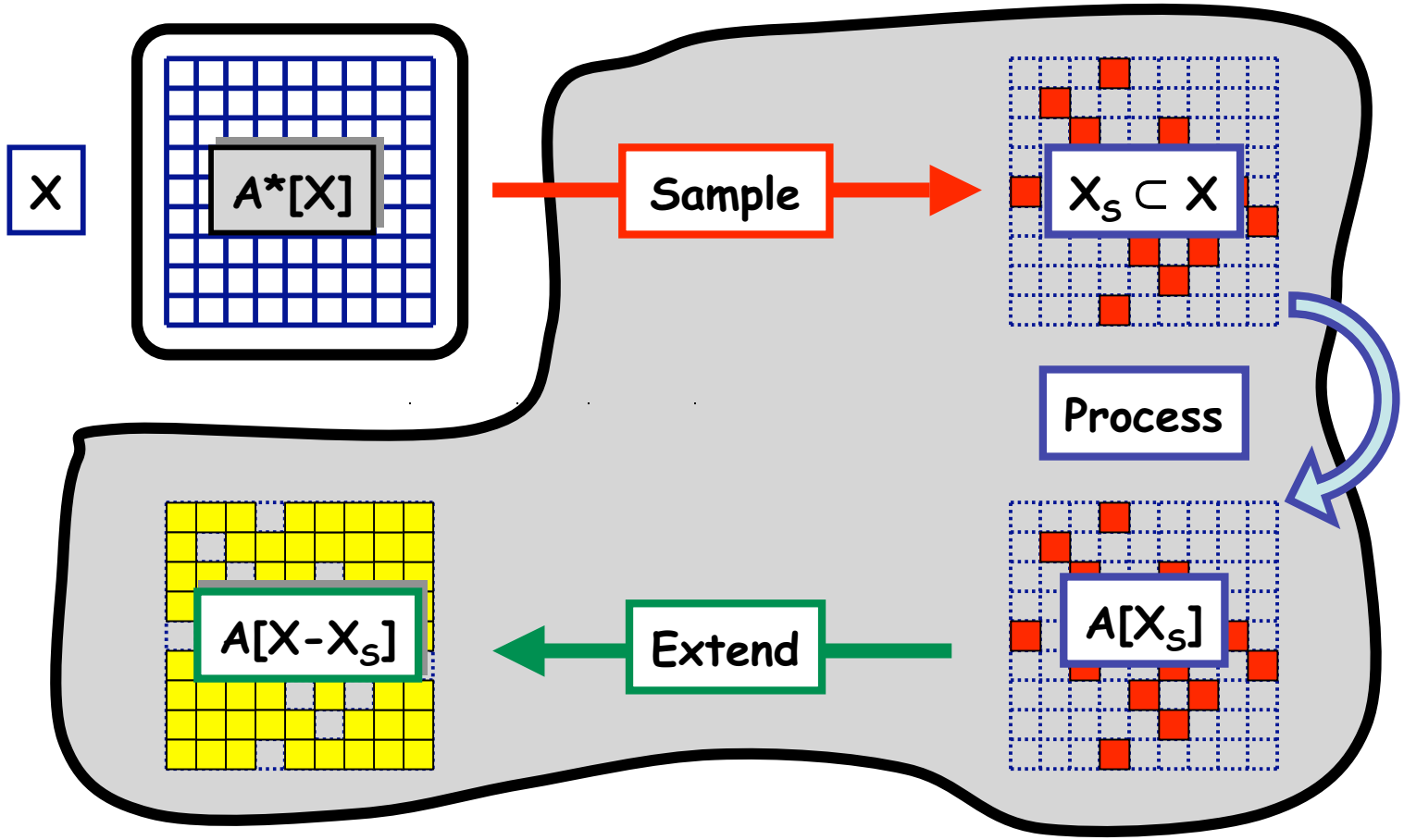
If X_{VL} or D_{VL} is *Unloadable* Comparison is *impossible*

so …

case 2 validity rests with "good" case 1 examples

What is *Extensibility* ?

Algorithm $A : X \subset \Re^p \mapsto A[X] \subset \Re^q$



Literal $A^*[X]$ $\cong$ **Approx.** $A[X_S] || A[X-X_S]$

Sample $X_s \subset X$

Process FCM[X_s]

Extend FCM[X_s] → FCM[$X - X_s$]

(Non-Iterative) Extension of *Fuzzy c-Means* [FCM → eFFCM/geFFCM]

with prototypes V_s and $x_k \in X - X_s$

$(X - X_s) \ni x_k$

$\{\cdots v_{i,s} \cdots v_{j,s} \cdots\} = V_s$

FONC for U to min J_m

$$u_{ik} = \left[\sum_{j=1}^{c} \left(\|x_k - v_{i,s}\|_A \Big/ \|x_k - v_{j,s}\|_A \right)^{\frac{2}{m-1}} \right]^{-1}$$

Works as a *classifier*, but trained w° labels !

Sample	$X_S \subset X$
Process	$EM[X_S]$
Extend	$EM[X_S] \rightarrow EM[X-X_S]$

(Non-Iterative) Extension of EM (*Gaussian Mixture Decomposition*)

with priors $\{p_{iS}\}$, means $\{\mu_{iS}\}$, covariances $\{\Sigma_{iS}\}$ and $x_k \in X-X_S$

$(X - X_S) \ni x_k$ $\{\mu_{iS}\}$ $\{\Sigma_{iS}\}$

$$g_{ik} = \exp\left(-0.5 * \left\| x_k - \mu_{iS} \right\|^2_{\Sigma^{-1}_{iS}}\right) \Big/ \sqrt{(2\pi)^s \left|\Sigma_{iS}\right|}$$

FONC for U to min ln(L)

$$u_{ik} = p_{iS}\, g_{ik} \Big/ \sum_{j=1}^{c} p_{jS}\, g_{jk}$$

$\{p_{iS}\}$

Works as a *classifier*, but trained w° labels !

Case 1 (loadable data) : Measure of Accuracy 1

Hardened Literal U

$$U^* = H(U^*_{LFCM}) \text{ (or) } H(U^*_{LNERF})$$

Hardened Approx U

$$U_{app} = H(U_S \,||\, U_{X-X_S})$$

$$0 \le E_{app}(U^*, U_{app}) = \frac{1}{2n}\left(\sum_{k=1}^{n} \left\| U^*_k - U_{app,k} \right\|_2^2\right) \le 1$$

Perfect Match

Approximation error

Complete Mismatch

Case 1 (loadable data) : Measure of Accuracy 2

Hardened Approx U

$U_{app} = H(U_S \,||\, U_{X-X_S})$

Crisp training labels

U_{tr}

$$0 \le E_{tr}(U_{app}, U_{tr}) = \frac{1}{2n}\left(\sum_{k=1}^{n} \left\| U_{app,k} - U_{tr,k} \right\|_2^2\right) \le 1$$

No Errors

This is training or resubstitution error

All Errors

Case 1 (loadable data) : Measure of Accuracy 3

Literal Cluster Centers V^*

Approx. Cluster Centers V_{app}

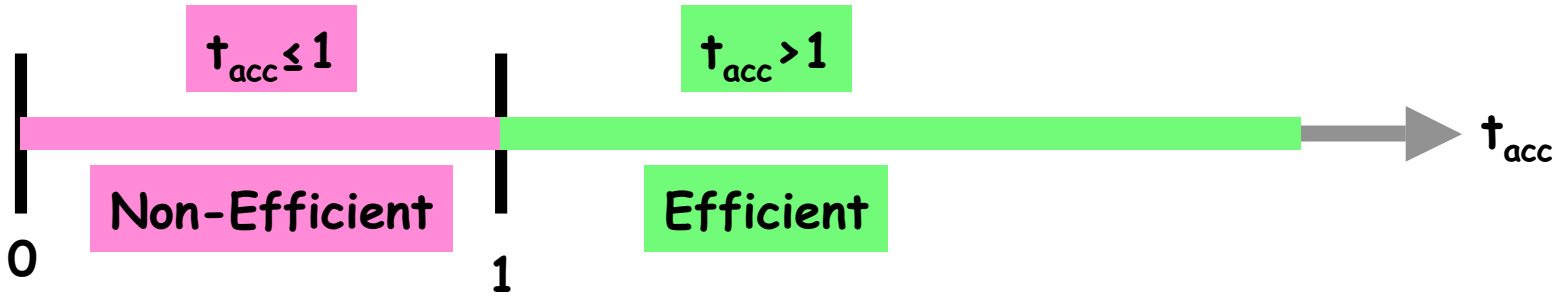
$$0 \leq E_V(V^*, V_{app}) = \frac{\left\| V^* - V_{app} \right\|}{\left\| V^* \right\|}$$

Overall error of c sample prototypes as an *approximation* to c literal prototypes

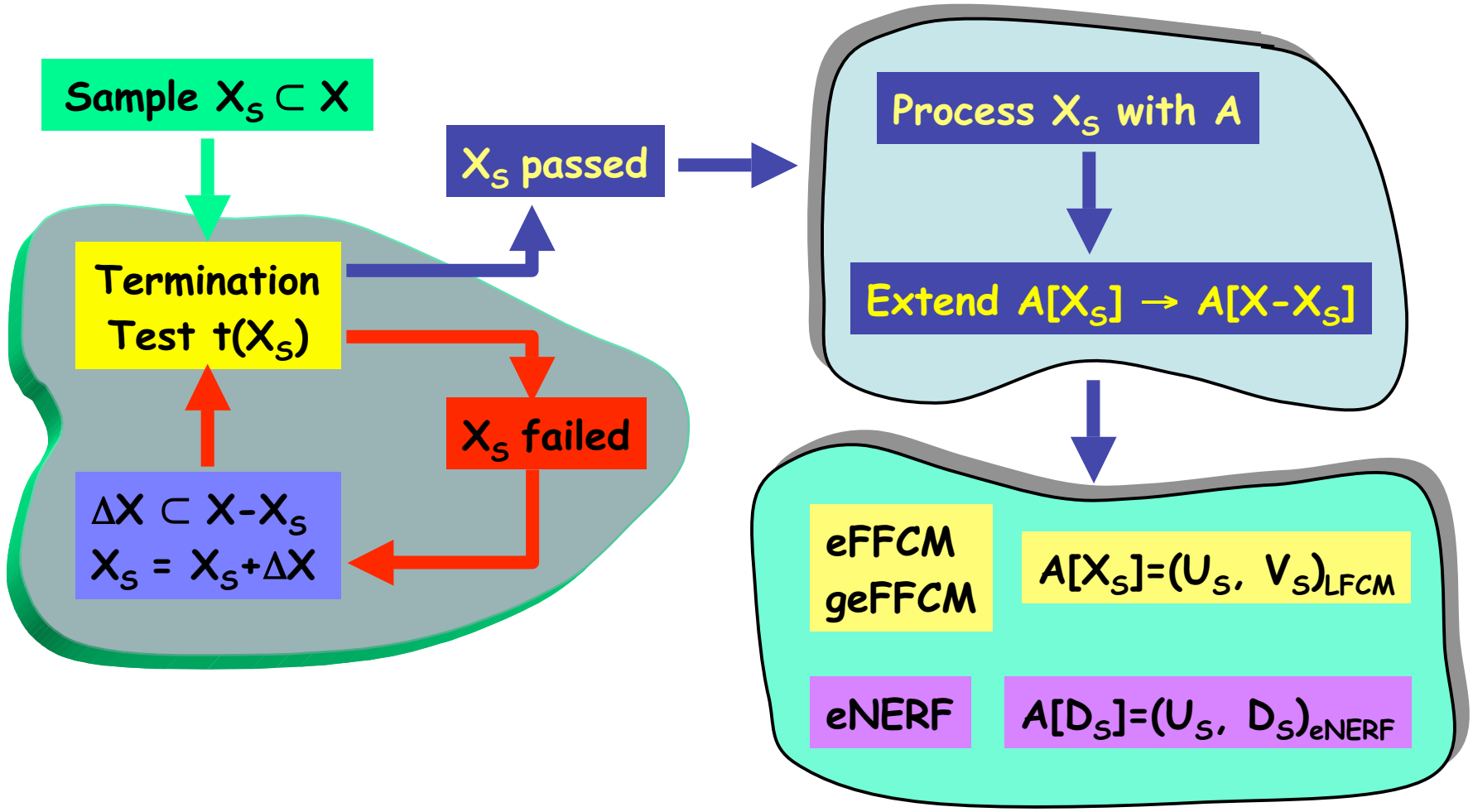
Case 1 (loadable data) : Measure of Acceleration

$$t_{acc}(U^*, U_{app}) = \left(\frac{\text{time to get } U^*}{\text{time to get } U_{app}} \right) = \left(\frac{t^*}{t_{app}} \right)$$

Algorithm A is *efficiently extensible* if approximation by extension is faster than literal implementation



What is *Progressive Sampling* ?



Sampling differs for the 3 cases

eFFCM — Images must be sampled to account for *BOTH intensity and spatial distributions*

geFFCM — Feature vectors have *only spatial distribution*

eNERF — (The objects underlying) relational data have *unknown distributional properties*

Guiding Principle for all 3 cases → **Attempt to capture samples from *each cluster* in the VL Data**

Que es proximo?

eFFCM - VL Image Data

geFFCM/geFEM - VL Feature Vector Data

eNERF - VL Relational Data