# Couplage de clustering et d’analyses de sensibilité pour les modèles à sorties multivariées

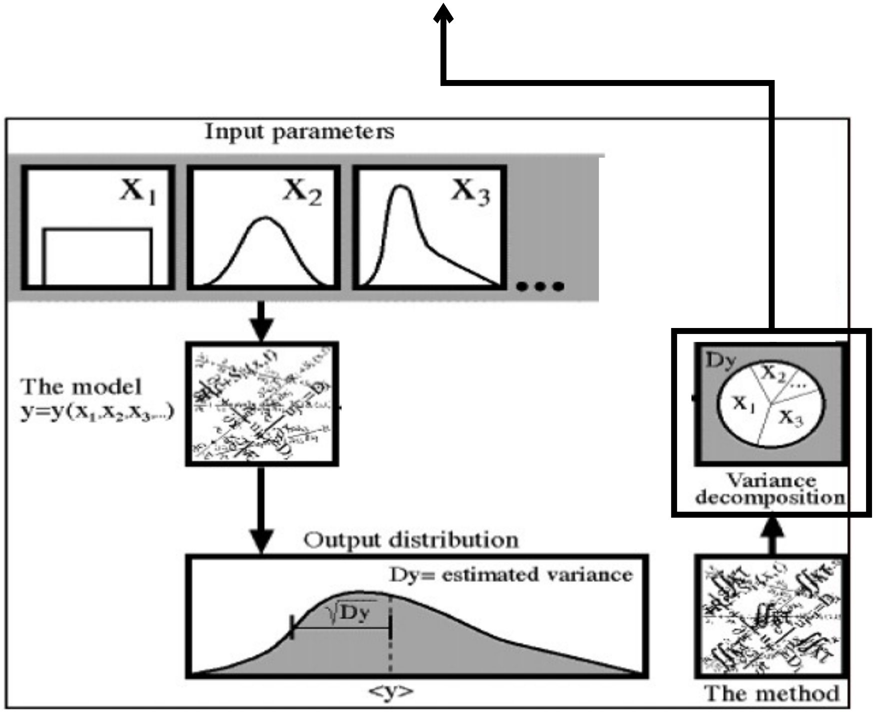
Sébastien Roux[1],Patrice Loisel[1], Samuel Buis[2]

[1] INRAE, UMR MISTEA, Montpellier, France
[2] INRAE, UMR EMMAH, Avignon, France,

*Journées du réseau Mexico, Toulouse, 29-30 novembre 2021*

# Global Sensitivity Analysis: Variance-Based methods

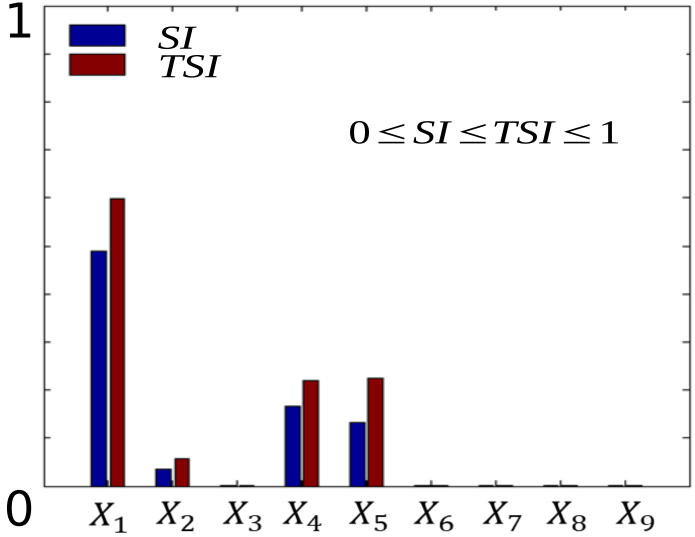
**Parts of variance explained by the different parameters**




From Saltelli et al (2001)

**Sobol' Indices:**

| $X_i$ alone | $X_i$ and its interactions |
| --- | --- |
| $SI(X_i) = \frac{V(E(Y \mid X_i))}{V(Y)}$ | $TSI(X_i) = 1 - \frac{V(E(Y \mid X_{-i}))}{V(Y)}$ |

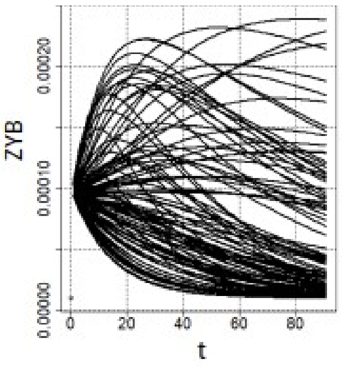


# Motivation

⇒ Which parameters drive the model ouputs toward different behaviors / regions in the model output space?

# Motivation

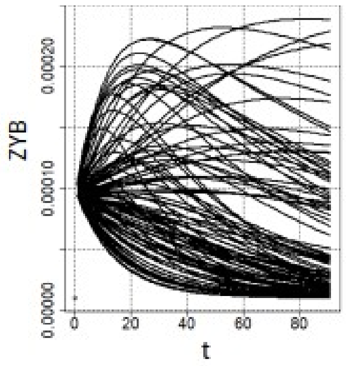
⇒ Which parameters drive the model ouputs toward different behaviors / regions in the model output space?



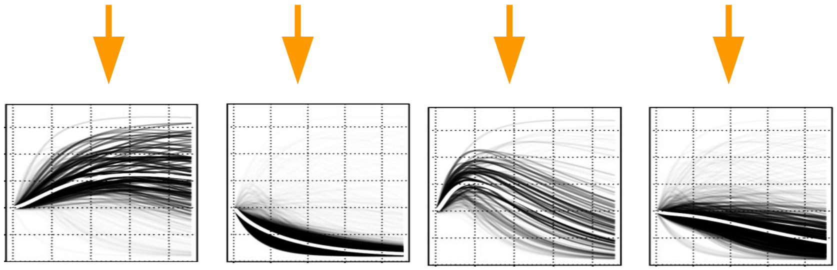

nd
multivariate

# Motivation

⇒ Which parameters drive the model ouputs toward different behaviors / regions in the model output space?

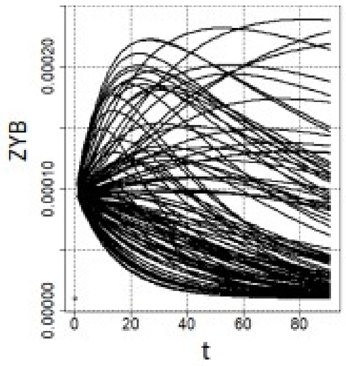


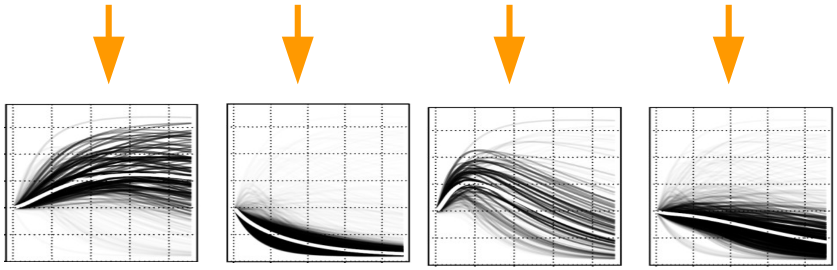

nd
multivariate



# Motivation

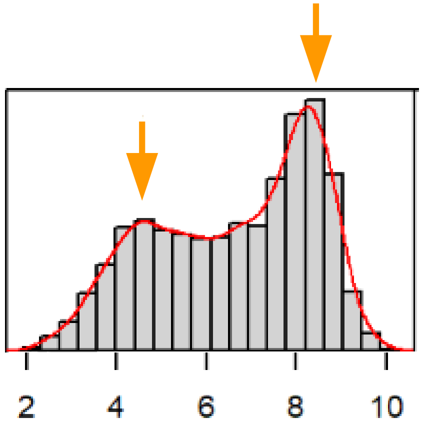
⇒ Which parameters drive the model ouputs toward different behaviors / regions in the model output space?



nd
multivariate

1d
bimodal



# Cluster-based GSA : Principle

**Combines clustering methods**

**=> reveal and characterize multiple distinct behaviors of the model outputs**

**and variance-based methods**

**=> identify in a robust way parameters and interactions that drive these different behaviors**

*Roux, S., Buis, S., Lafolie, F., & Lamboni, M. (2021).*
*Cluster-based GSA: Global sensitivity analysis of models with temporal or spatial outputs using clustering. Environmental Modelling & Software,*

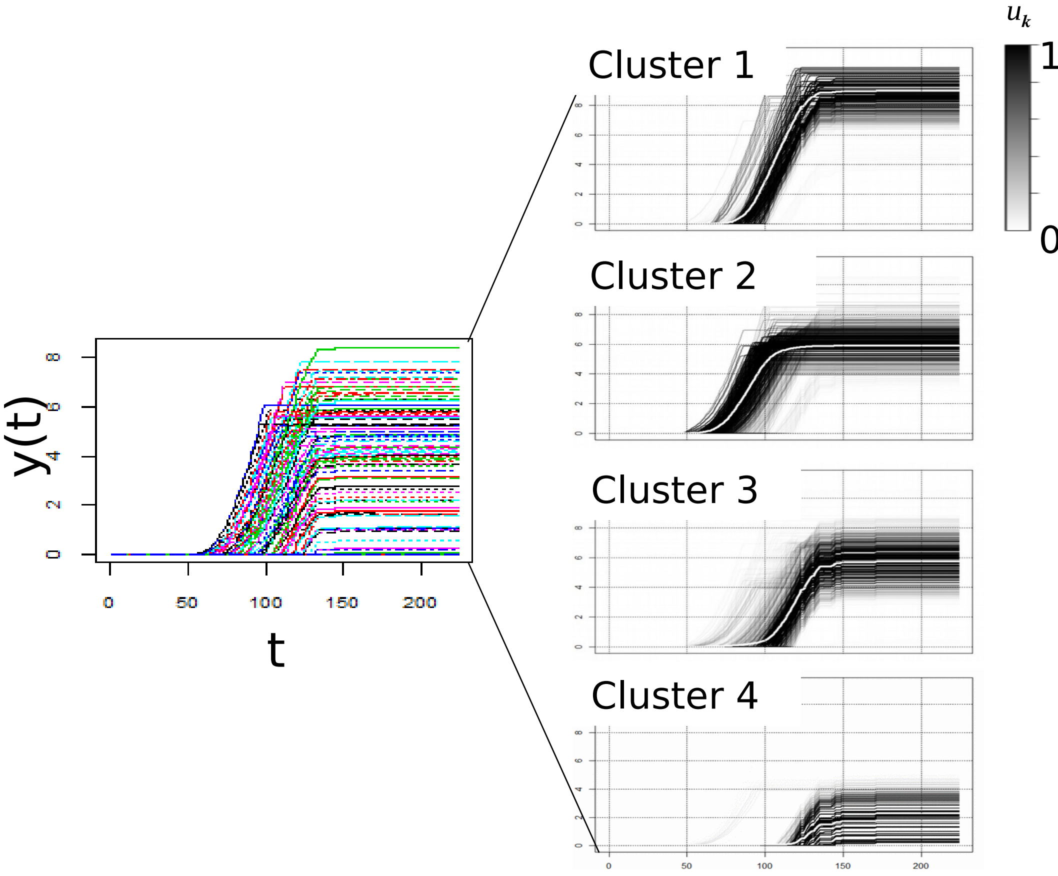
# Cluster-based GSA : scalar membership functions

**output = Membership Functions (MF)**

$y^i(t) \rightarrow [u^i_1, .., u^i_K]$

$u^i_k$: level of membership of object i to cluster k

$\sum_{k=1}^{K} u^i_k = 1$




*Roux, S., Buis, S., Lafolie, F., & Lamboni, M. (2021).*
*Cluster-based GSA: Global sensitivity analysis of models with temporal or spatial outputs using clustering. Environmental Modelling & Software,*

# Cluster based GSA indices

$y^i(t) \rightarrow [u_1^i, .., u_K^i]$

- Sensitivity indices on membership functions
  => Which parameters (or interactions) drive the model outputs toward a targeted cluster?

$$SI_k(X_j) = \frac{V(E(u_k|X_j))}{V(u_k)}$$

$$TSI_k(X_j) = 1 - \frac{V(E(u_k|X_{-j}))}{V(u_k)}$$

- Sensitivity indices on membership function differences
  => Which parameters (or interactions) drive the model outputs from one cluster to another?

$$SI_{kl}(X_j) = \frac{V(E((u_k - u_l)|X_j))}{V(u_k - u_l)}$$

$$TSI_{kl}(X_j) = 1 - \frac{V(E((u_k - u_l)|X_{-j}))}{V(u_k - u_l)}$$

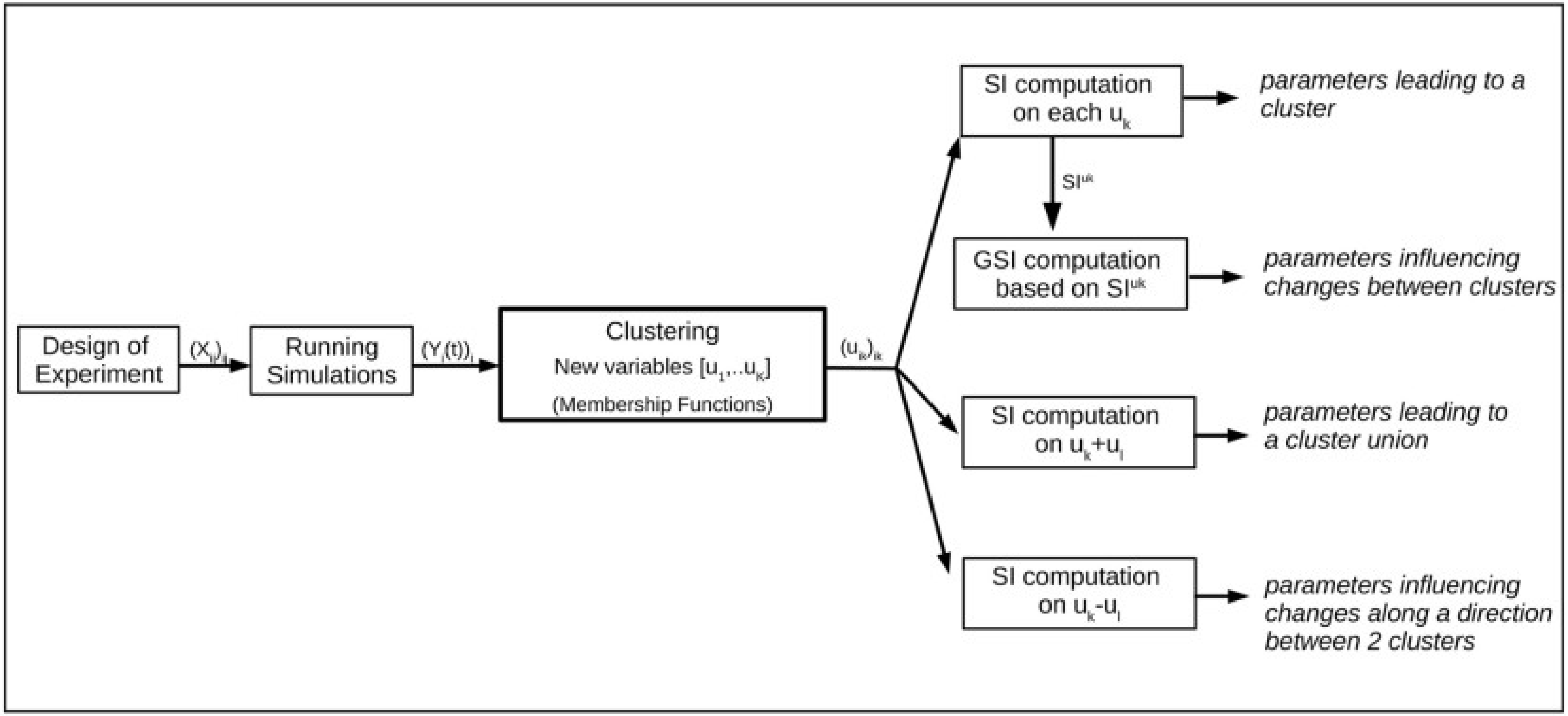
- Aggregated indices on the vector of membership functions
  => Which parameters (or interactions) globally impact changes between clusters

$$SI(X_j) = \frac{\sum_{k=1}^{K} V(u_k) SI_k(X_j)}{\sum_{k=1}^{K} V(u_k)}$$

$$TSI(X_j) = \frac{\sum_{k=1}^{K} V(u_k) TSI_k(X_j)}{\sum_{k=1}^{K} V(u_k)}$$

*Roux, S., Buis, S., Lafolie, F., & Lamboni, M. (2021).*
*Cluster-based GSA: Global sensitivity analysis of models with temporal or spatial outputs using clustering. Environmental Modelling & Software,*

# Cluster-based GSA : workflow




*Roux, S., Buis, S., Lafolie, F., & Lamboni, M. (2021).*
*Cluster-based GSA: Global sensitivity analysis of models with temporal or spatial outputs using clustering. Environmental Modelling & Software,*

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions
- There are 3 general ways of using cluster-based indices

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions

- There are 3 general ways of using ClustSIs

  - **Prior partitions**

  (expertise-driven clustering)

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions

- There are 3 general ways of using ClustSIs

- **Prior partitions**
  (expertise-driven clustering)

- **Optimized Partitions**

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions

- There are 3 general ways of using ClustSIs

  - **Prior partitions**
    (expertise-driven clustering)

  - **Optimized Partitions**

**-> Data-driven clustering**
ClusterBased GSA
Optimization based on Y

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions

- There are 3 general ways of using ClustSIs

- **Prior partitions**
(expertise-driven clustering)

- **Optimized Partitions**

**-> Data-driven clustering**
ClusterBased GSA
Optimization based on Y

**->Sensitivity-driven clustering**
Optimization based on Y and X
(typically using $SI_1(Xi)$)

# Possible use of cluster-based sensitivity indices

- Using ClustSIs, we can associate sensitivity indices to output space partitions
- There are 3 general ways of using ClustSIs

- **Prior partitions** (expertise-driven clustering)

- **Optimized Partitions**

*CHARACTERIZATION*

*EXPLORATION*

**-> Data-driven clustering**
ClusterBased GSA
Optimization based on Y

**->Sensitivity-driven clustering**
Optimization based on Y and X
(typically using $SI_1(Xi)$)

# Sensitivity-driven clustering

- Objectives :

  - revealing behaviors (ie regions of the output space) most (or very much) impacted by variations of a parameter (or a group or an interaction,..) using an optimization procedure

  - expressing graphically the sensitivity of the input factors (or a group or an interaction,..) on the output, including in the case of MV outputs

# Sensitivity-driven clustering

- 1D « analytical example »
- 2D (numerical with different approaches )
- ND (numerical)

# Optimized sensitive partioning in 1D

- we restrict to the study to binary partitions A,B of [0,1]

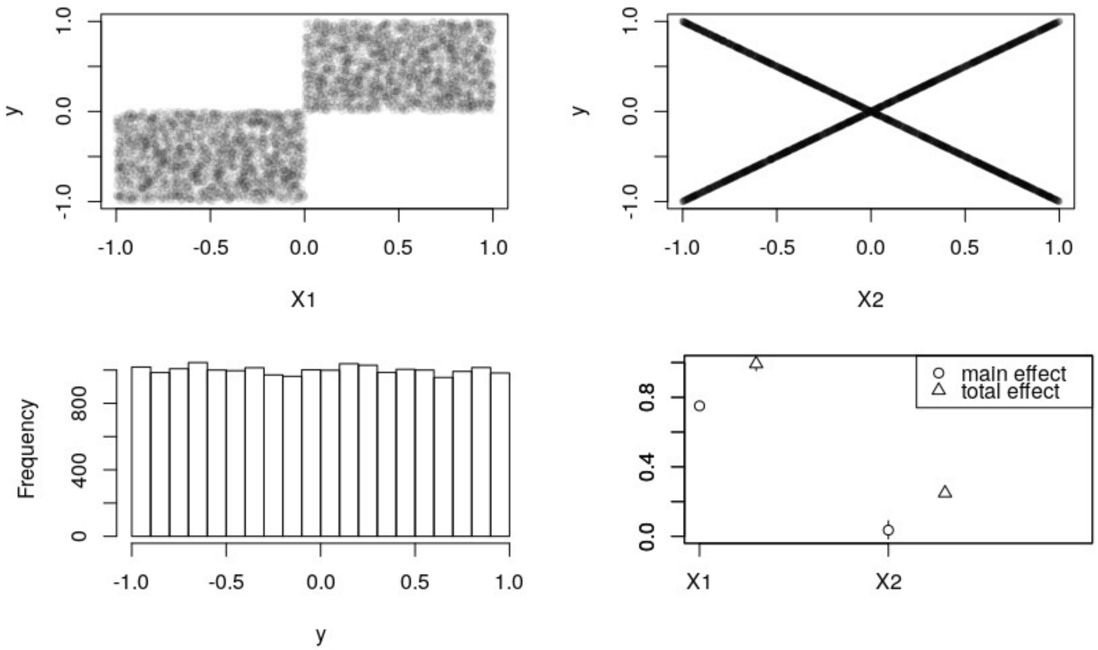
- many possible situations depending on connexity


- binarization with 2 connected components => parameterization of by a single cutting value yc

- binarization with 3 connected components => parameterized by a two cutting values yc1 and yc2

# 1D « analytical example »

$$Y(x_1, x_2) = sign(X_1) \cdot |X_2|$$
$$x_1 \sim U[-1,1]$$
$$x_2 \sim U[-1,1]$$

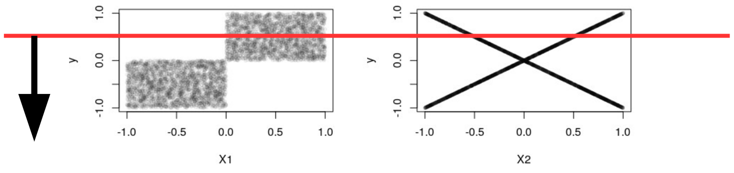


# 1D « analytical example »

$$Y(x_1, x_2) = sign(X_1) \cdot |X_2|$$
$$x_1 \sim U[-1,1]$$
$$x_2 \sim U[-1,1]$$

Binarization with 2
Connected components:

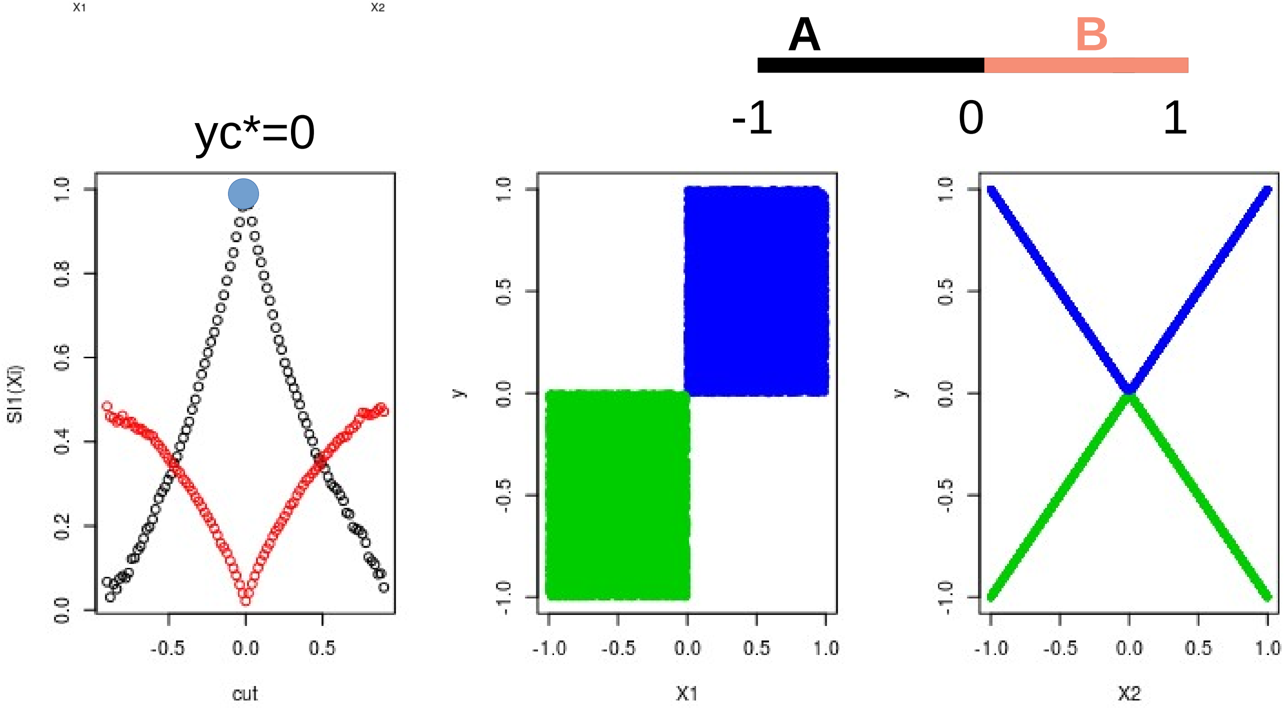
$$\tilde{Y}^{y_c}(X_1, X_2) = \mathbb{1}_{Y(X_1,X_2) \leq y_c}$$




$SI_1$(X1) for every cutting values yc

$SI_1$(X2) for every cutting values yc

$$SI_k(X_j) = \frac{V(E(u_k|X_j))}{V(u_k)}$$

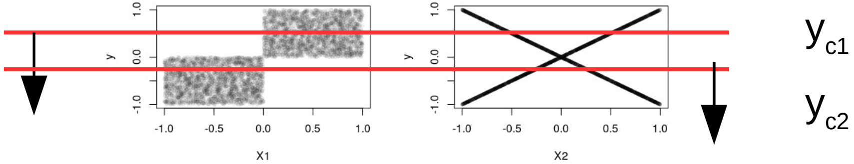


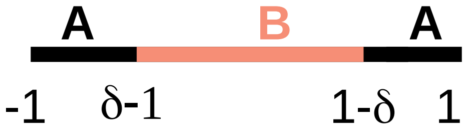
# 1D « analytical example »

$$Y(x_1, x_2) = sign(X_1) \cdot |X_2|$$
$$x_1 \sim U[-1,1]$$
$$x_2 \sim U[-1,1]$$

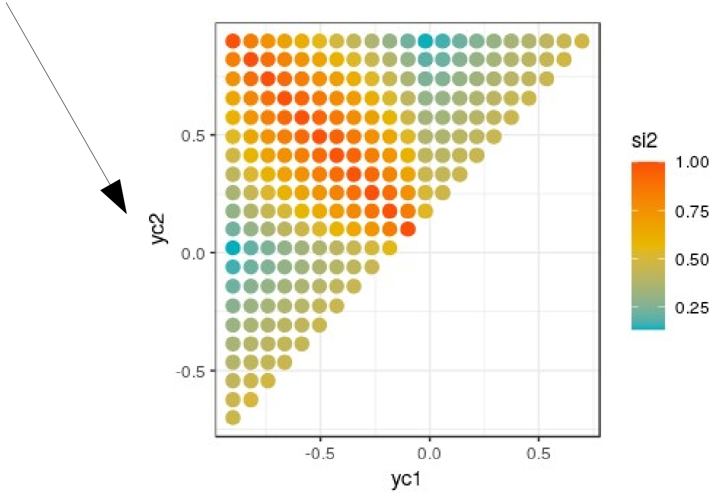
Binarization with 3 Connected components:

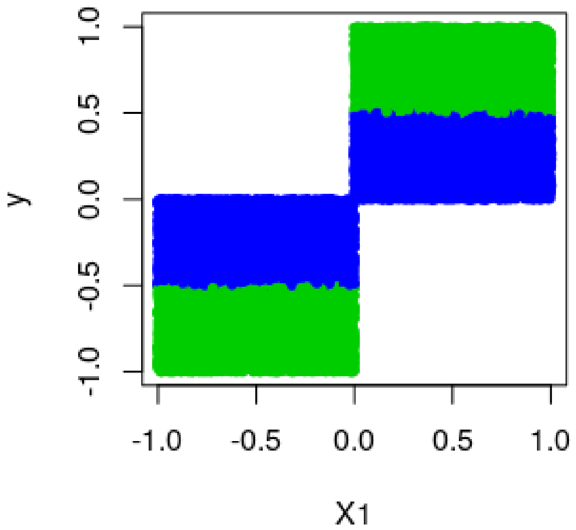
$$\tilde{Y}^{y_{c1},y_{c2}}(X_1, X_2) = \mathbb{1}_{Y(X_1,X_2)\in[y_{c1},y_{c2}]}$$

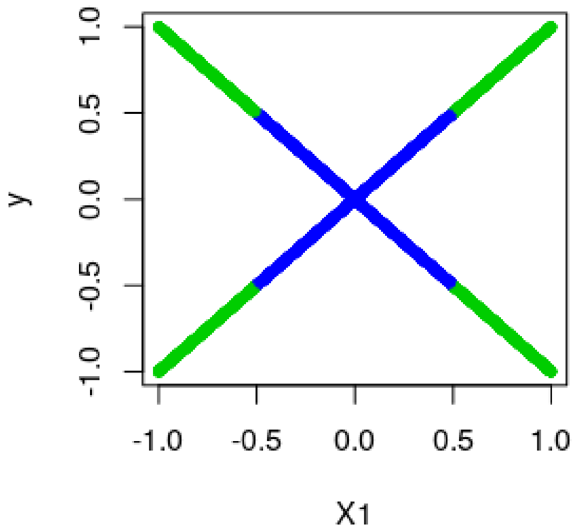



$SI_1$(X2) for every cutting values yc1,yc2

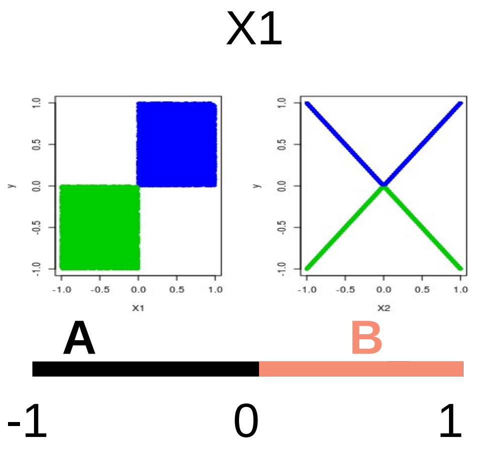


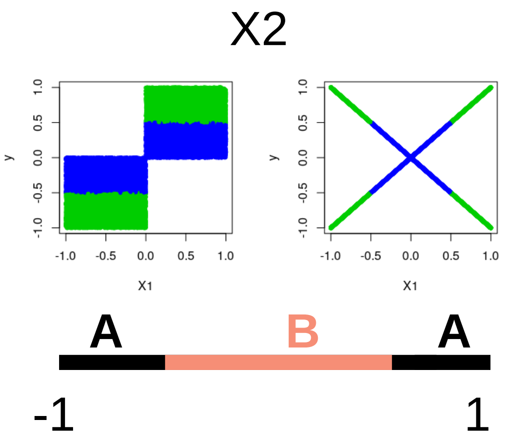



# 1D « analytical example »

$$Y(x_1, x_2) = sign(X_1) \cdot |X_2|$$
$$x_1 \sim U[-1,1]$$
$$x_2 \sim U[-1,1]$$



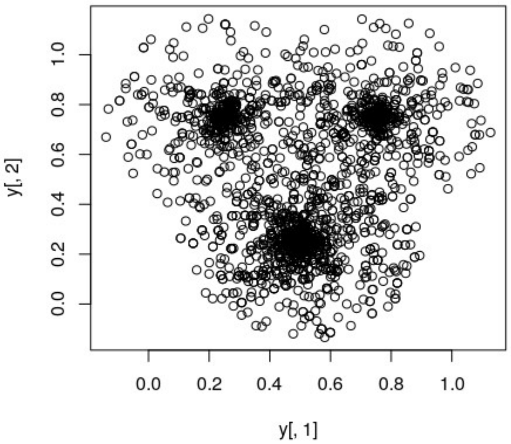

- Possibility to solve analytically
- Optimal partition depends on the parameter X1 or X2
- Optimum found even if the the space of partitions has not been completely explored (as SI=1 is optimal)
- We can have SI_C*>SI  (clustSI2*=1, SI2=0)
- Optimal partitions can have more than 2 connected components
- The optimal partition in not unique

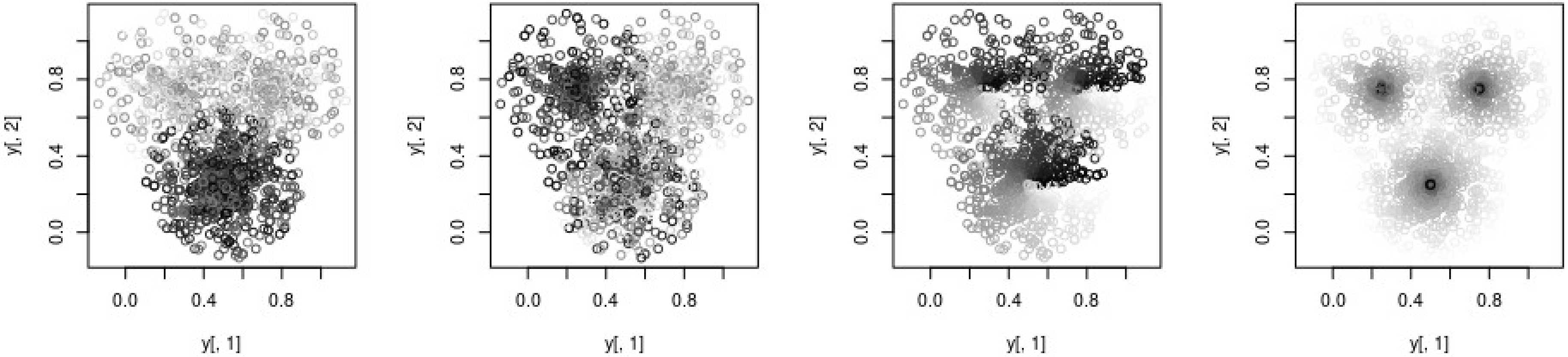
# 2D numerical example

```
test2d_1 = function(x)
{
cy = cent_y[1+as.numeric(x[1]>0.5)]
if (cy==0.25)
        cx=0.5
if (cy==0.75)
        cx=cent_x[1+ as.numeric(x[2]>0.5)]

y1 = cx + 0.4* cos(2*pi*x[3])*x[4]^3
y2 = cy + 0.4* sin(2*pi*x[3])*x[4]^3
return(c(y1,y2))
}
```

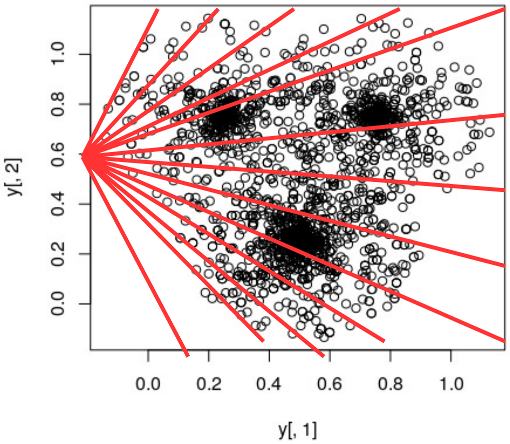
**Y=(Y1,Y2) = f(X1,X2,x3,X4)**
**3 centers**
- **X1,X2 : choice of center**
- **X3 : angle**
- **X4 : distance from center**



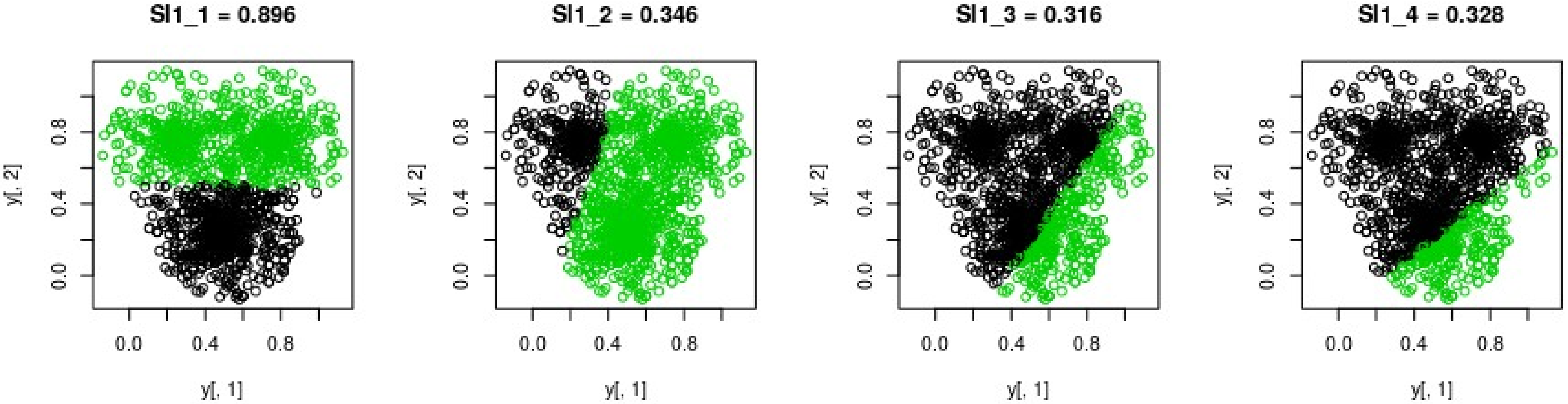




color=X1 color=X2 color=X3 color=X4

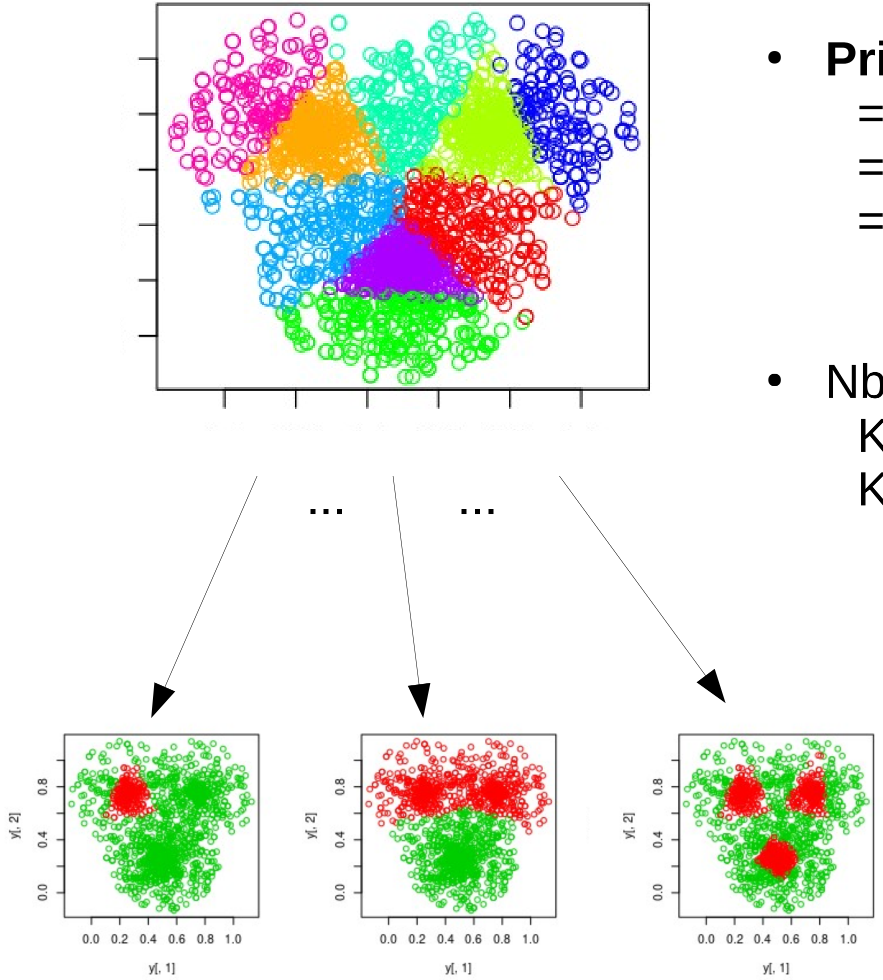
# 2D partitioning: connected binarization with straights lines




- Boundary discretization : n pts per border
- complexity: $6.n^2$
- N=7 => 294 splitting



# 2D (and nD) partitioning: non-connected partitioning
# SI1 criterion




- **Principle of the algorithm for SI/TSI criteria :**
  => Clustering of the outputs into K clusters
  => Generate all partitions [1..K] into 2 sets
  => Compute SI/TSI criteria for each binarization

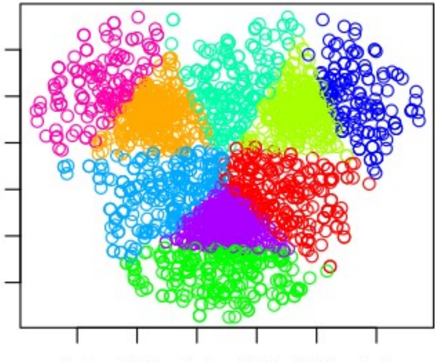
- Nb : 2^(K-1)
  K=10 : Nb= 512
  K=20 : Nb= 524288

⇒ solve the issue of getting non connected set
⇒ very flexible (various SI-based criteria, adding constraints)
⇒ limited spatial resolution (K..)

⇒ can handle MV output (providing the clustering does)

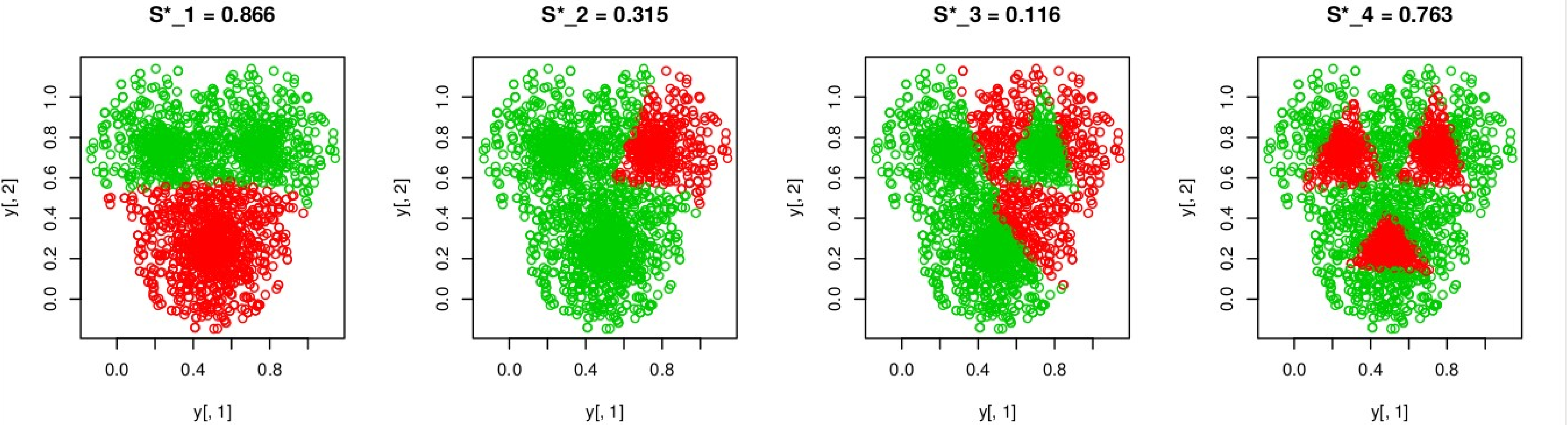
# 2D (and nD) partitioning: non-connected partitioning
# SI1 criterion

**K=9**



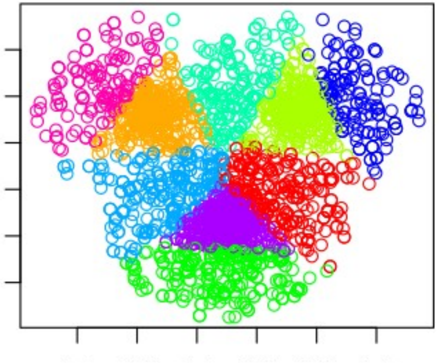
- **Principle of the algorithm for SI/TSI criteria: (sensitivity indices on Membership functions)**
  => Clustering of the outputs into K clusters
  => Generate all partitions [1..K] into **2 sets**
  => Compute SI/TSI criteria for each binarization

$$SI_k(X_j) = \frac{V(E(u_k|X_j))}{V(u_k)}$$



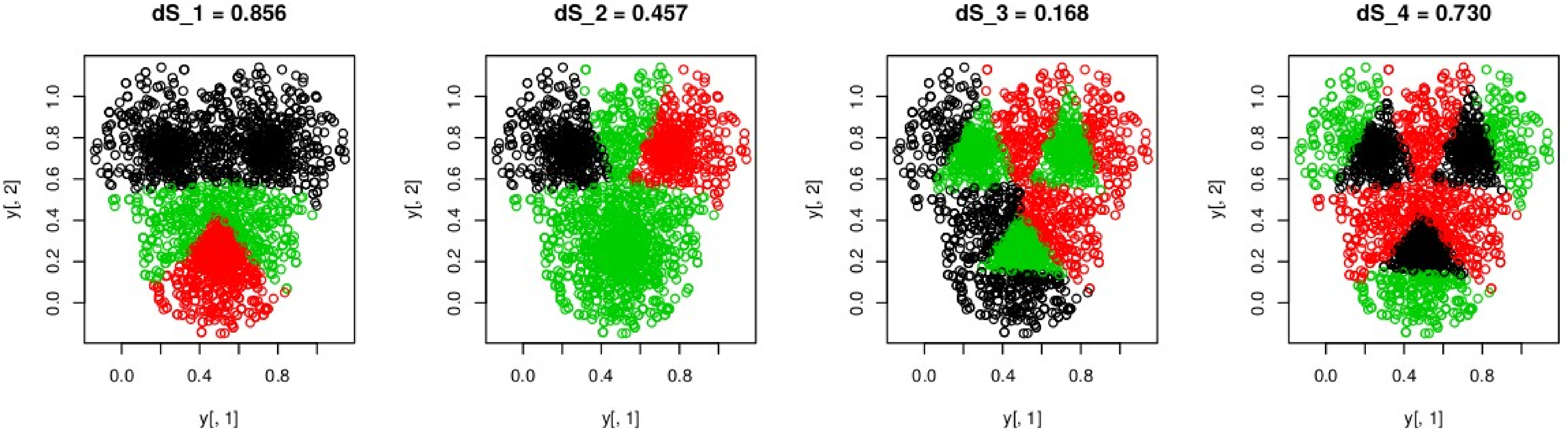
# 2D (and nD) partitioning: non-connected partitioning
# Other criterion : “neutral class”

**K=9**



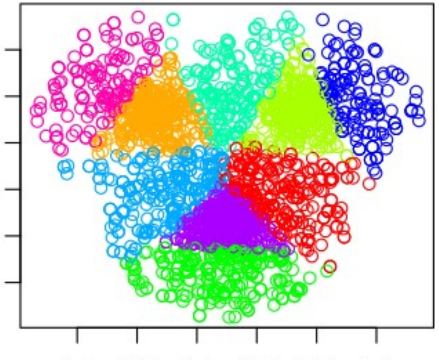
- **Principle of the algorithm using ‘neutral class’ (SI/TSI criteria on MF differences)**
- => Clustering of the outputs into K clusters
- => Generate all partitions [1..K] into 3 sets
- => Compute SI/TSI criteria for each set using u1-u2

$$SI_{kl}(X_j)=\frac{V(E((u_k-u_l)|X_j))}{V(u_k-u_l)}$$



# 2D (and nD) partitioning: non-connected partitioning
# Other criterion : “GSI”

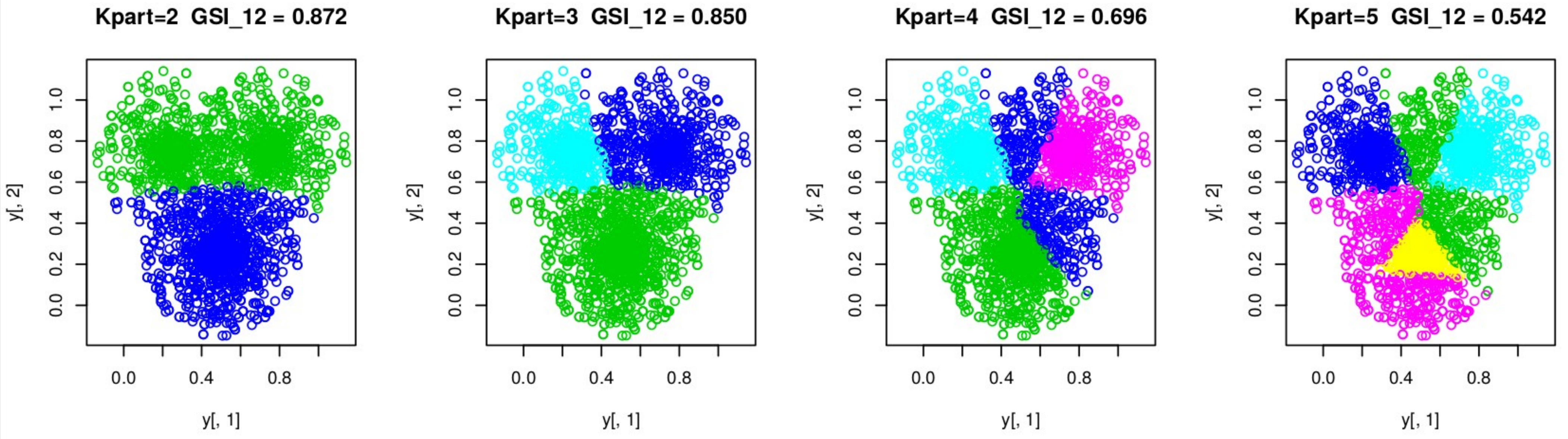
**K=9**



- **Principle of the algorithm**
- => Clustering of the outputs into K clusters
- => Generate all partitions [1..K] into 2-3-4-5 sets
- => Compute GSI criteria for each set

$$SI(X_j)=\frac{\sum_{k=1}^{K} V(u_k) SI_k(X_j)}{\sum_{k=1}^{K} V(u_k)}$$

***Studying GSI for (X1,X2) and 2-3-4-5 sets***



# 2D (and nD) partitioning: non-connected partitioning
# Algorithm improvement for SI1 criterion

*⇒ **Keeping the exhaustive step (seems necessary to handle non connectivity)***
*⇒ **Improving the pre-processing step by using properties of the criterion***

We consider a set of elementary patches Ck (region of the output space)
We consider the conditional distribution of Xi for Y in Ck
We compute the associated histograms for a given discretization of Xi into bins

The SI1 based clustering criterion writes

$$S_C = \frac{n_x}{\sum_{j=1}^{n_x} h_j^C (N - \sum_{j=1}^{n_x} h_j^C)} \sum_{i=1}^{n_x} (h_i^C - \frac{1}{n_x} \sum_{j=1}^{n_x} h_j^C)^2$$

# 2D (and nD) partitioning: non-connected partitioning
# Algorithm improvement for SI1 criterion

⇒ ***Keeping the exhaustive step (seems necessary to handle non connectivity)***
⇒ ***Improving the pre-processing step by using properties of the criterion***

Let’s consider two patches Ck, Ck’ and their associated histograms Hk,Hk’
Let’s suppose that Hk,Hk’ are highly correlated

Let’s denote (C*,-C*) the optimal partition built from the (Ck)

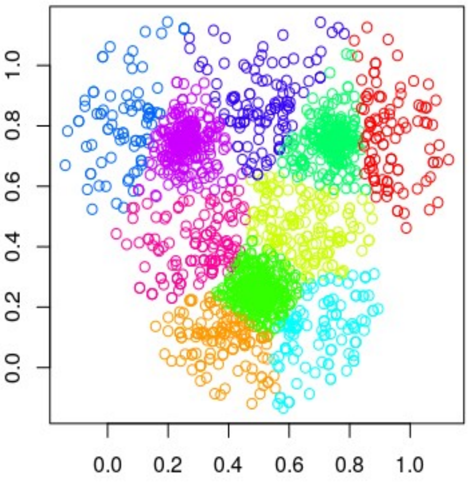
Then
Either Ck and Ck’ belongs to C*
Ck and Ck’ belongs to -C*

# 2D (and nD) partitioning: non-connected partitioning
# Algorithm improvement for SI1 criterion

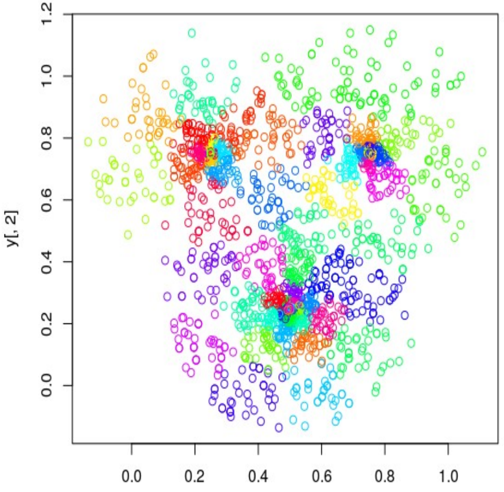
*⇒ **Keeping the exhaustive step (seems necessary to handle non connectivity)***
*⇒ **Improving the pre-processing step by using properties of the criterion***

- High resolution clustering of the output space into K cluster (e.g. 150)
- New : Aggregation of clusters based on histogram correlation
  → K' metapatches
- Generate all partitions [1..K'] into 2 sets
- Compute SI/TSI criteria for each binarization

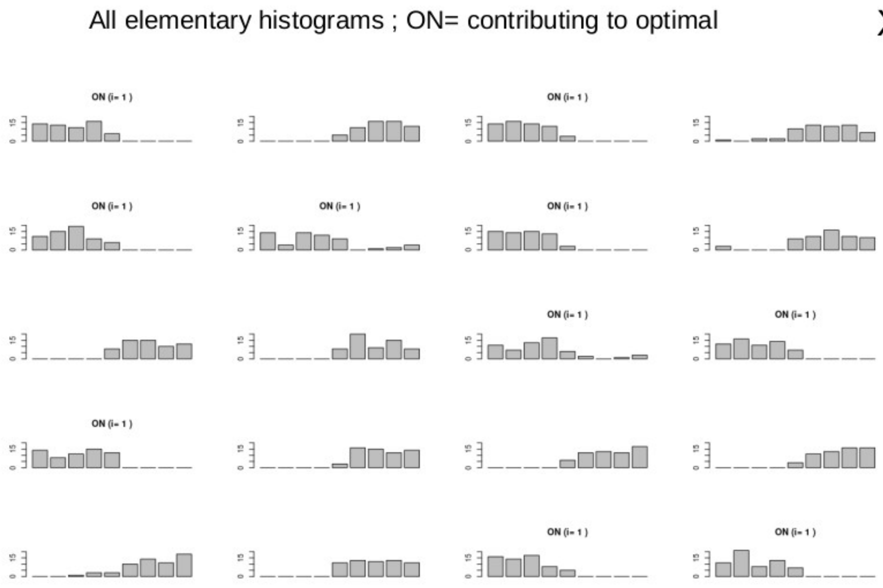
# *Hierarchical clustering of elementary histograms*



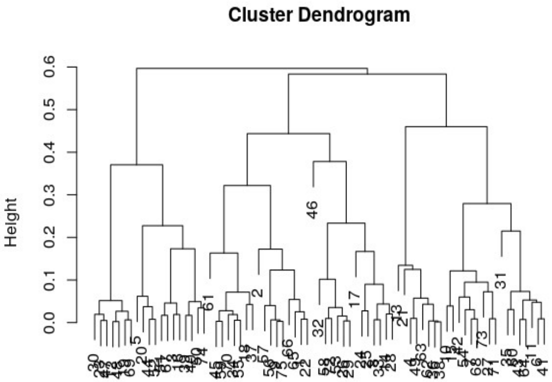

**K=10**

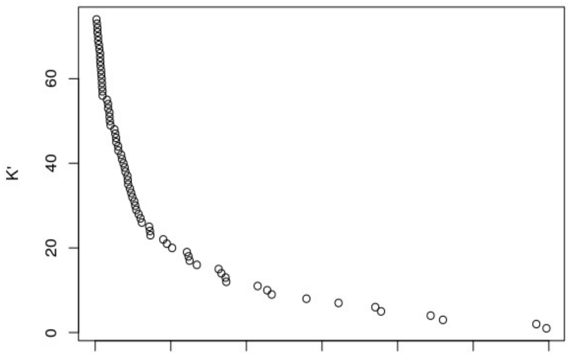



**X3**

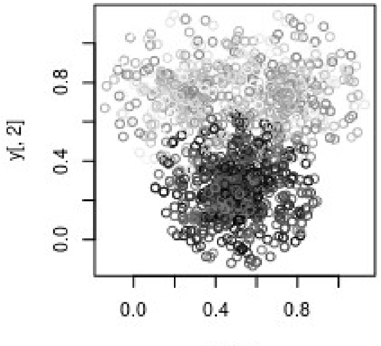



X1

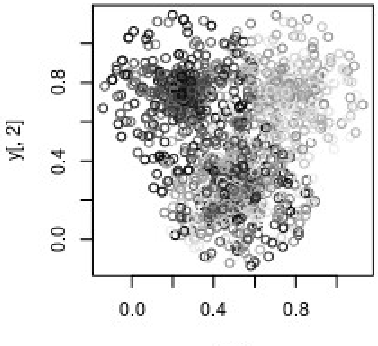


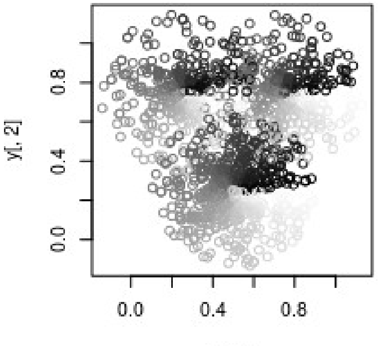
*Results*



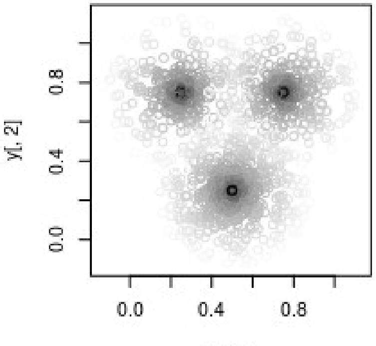

color=X1




color=X2




color=X3



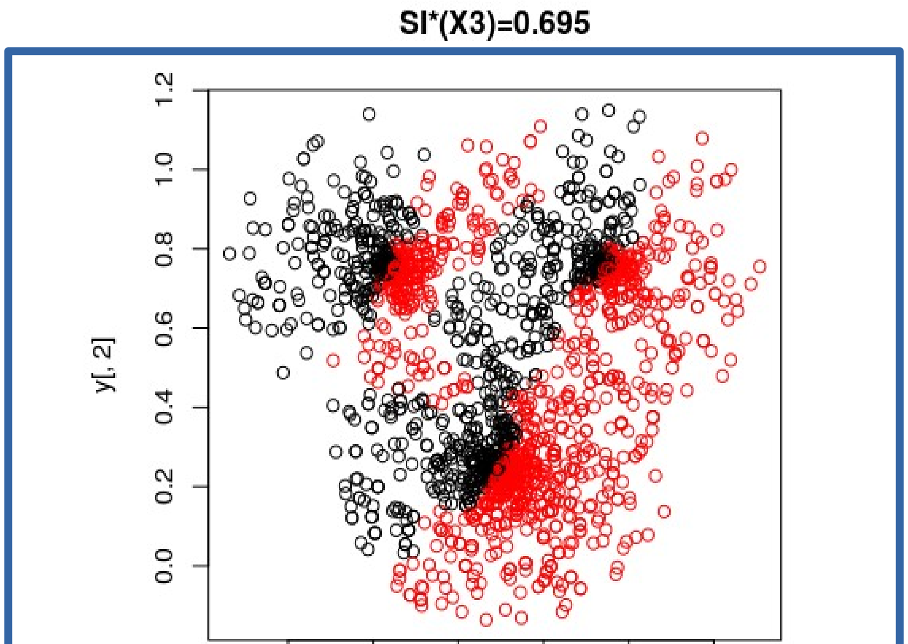

color=X4

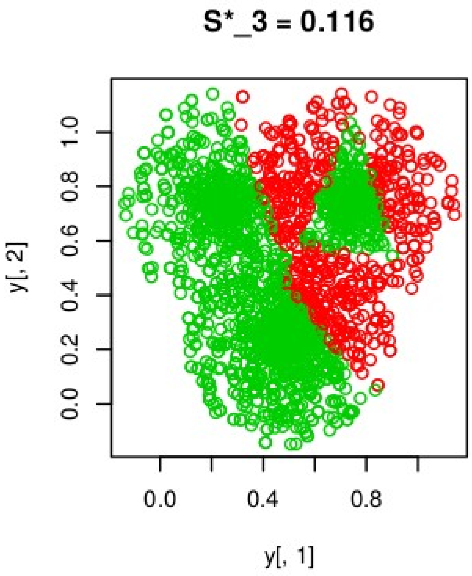
**X3**

**K=150**

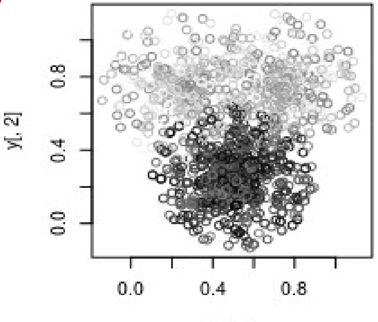
**Cut=0.2**

**K'=16**

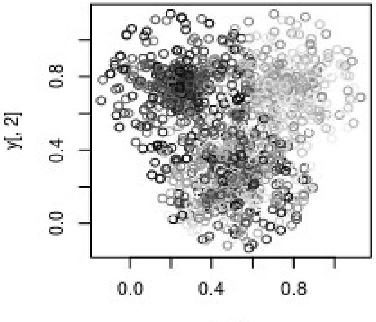


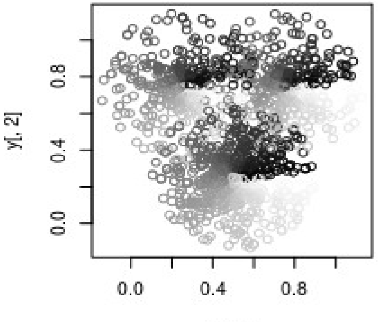
*Resultats*



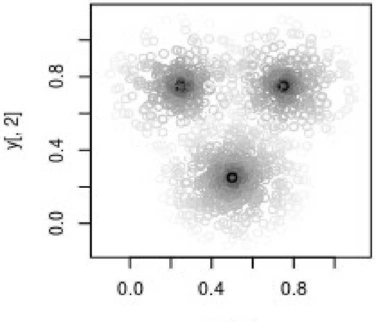

color=X1




color=X2

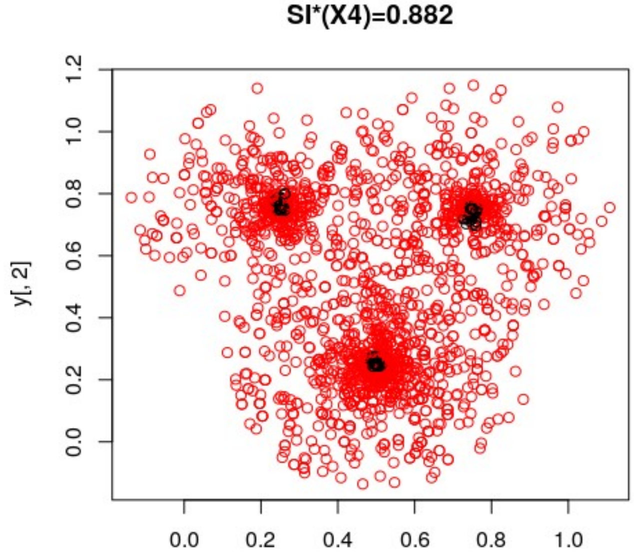


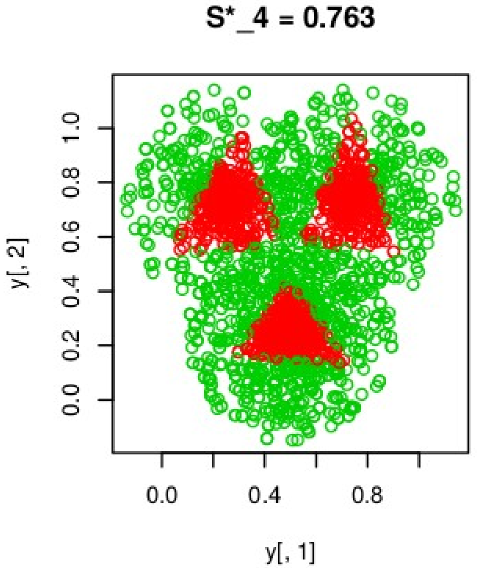

color=X3




color=X4

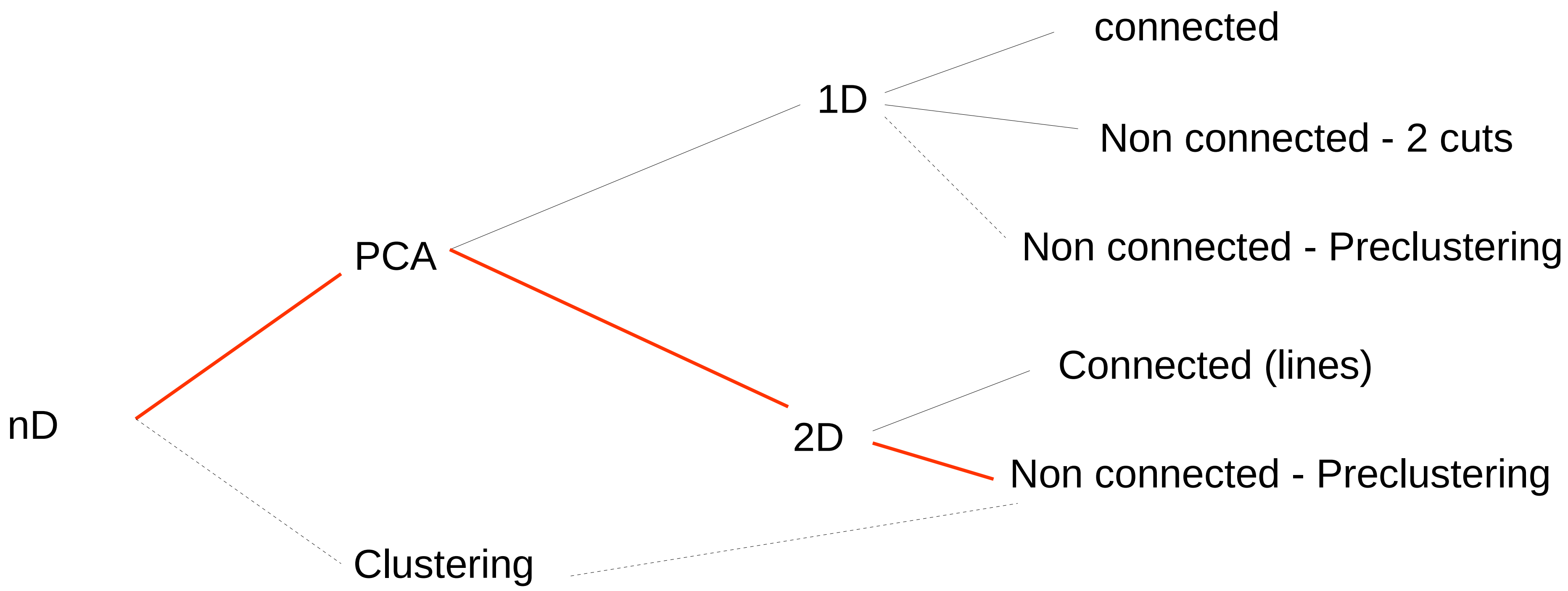
**X4**
**Cut=0.1**
**K'=13**



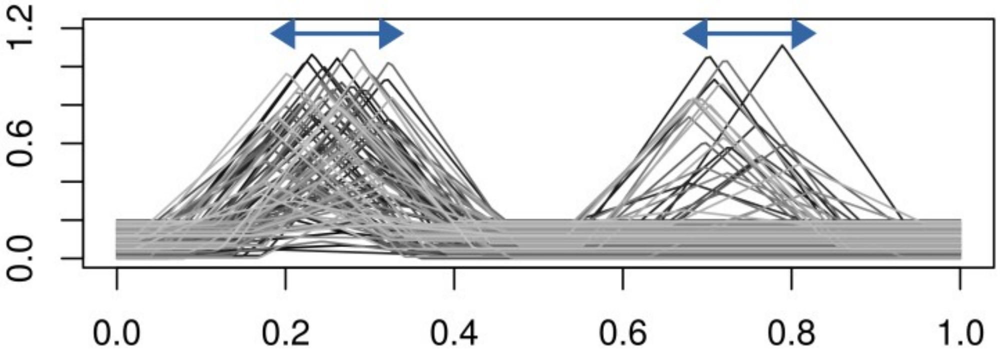



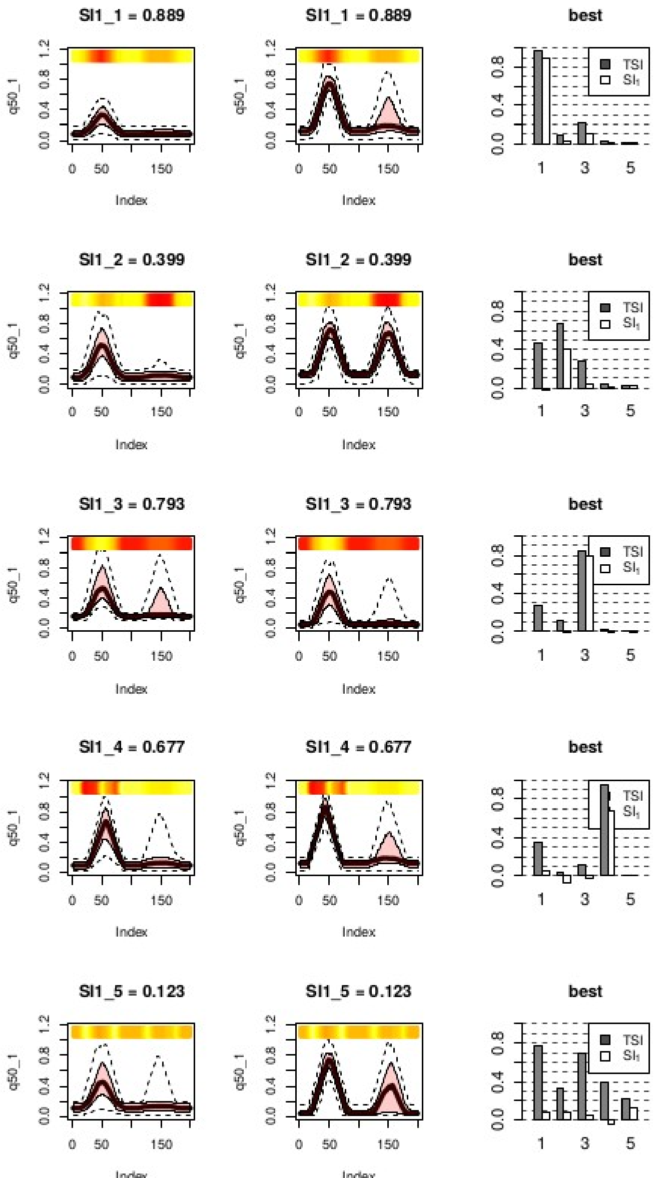

(ancienne approche)

# nD numerical example: ToyCurve



# nD numerical ToyCurve



# Perspectives

- CB-GSA : applications on environmental models (crop mixtures or hydrologic models)
  => Key issues
    - DOE
    - Finding appropriate clustering (a priori or automatic)

- CB-GSA :Complementary analysis : intra-cluster and pure cluster transitions (which amounts to AS with dependent inputs)

- Sensitivity-driven clustering : test on a realistic model with MV outputs
- Sensitivity driven clustering : more efficient algorithms ??