The background of the slide is a classic marbled paper pattern, featuring a complex, organic design of swirling, interlocking shapes in shades of light beige, cream, and pale grey. The pattern is dense and covers the entire area.

RECONNAISSANCE DE FORMES

Une Courte Introduction

Introduction

Codage

Analyse

Apprentissage & Décision

Qu 'est-ce qu ' une forme ?

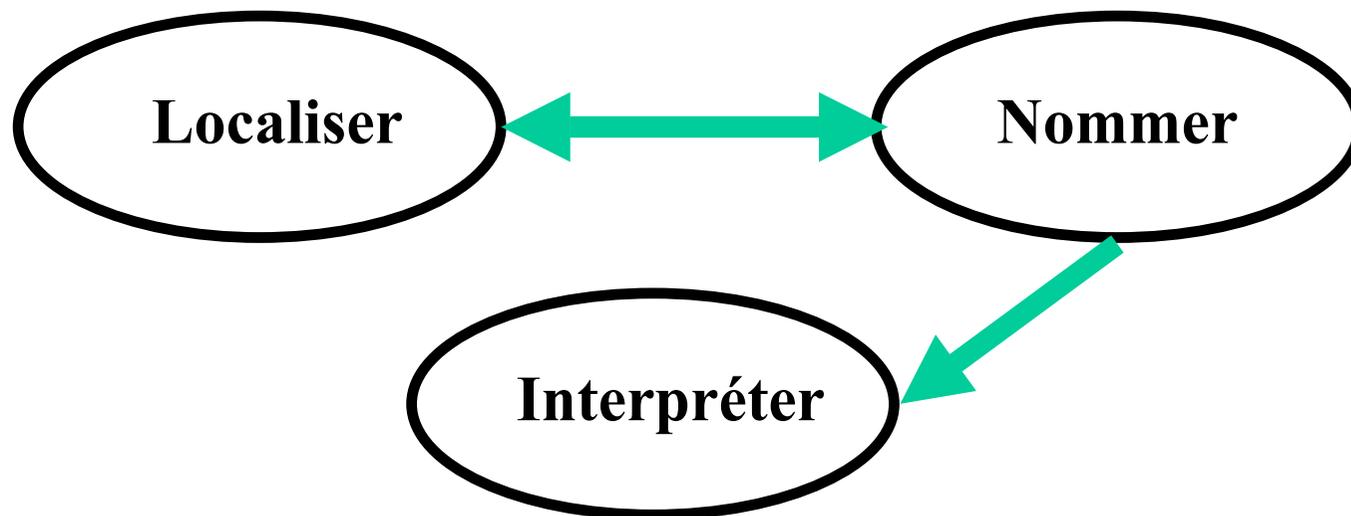
Introduction

Codage

Analyse

Apprentissage & Décision

Qu 'est-ce que reconnaître une forme ?



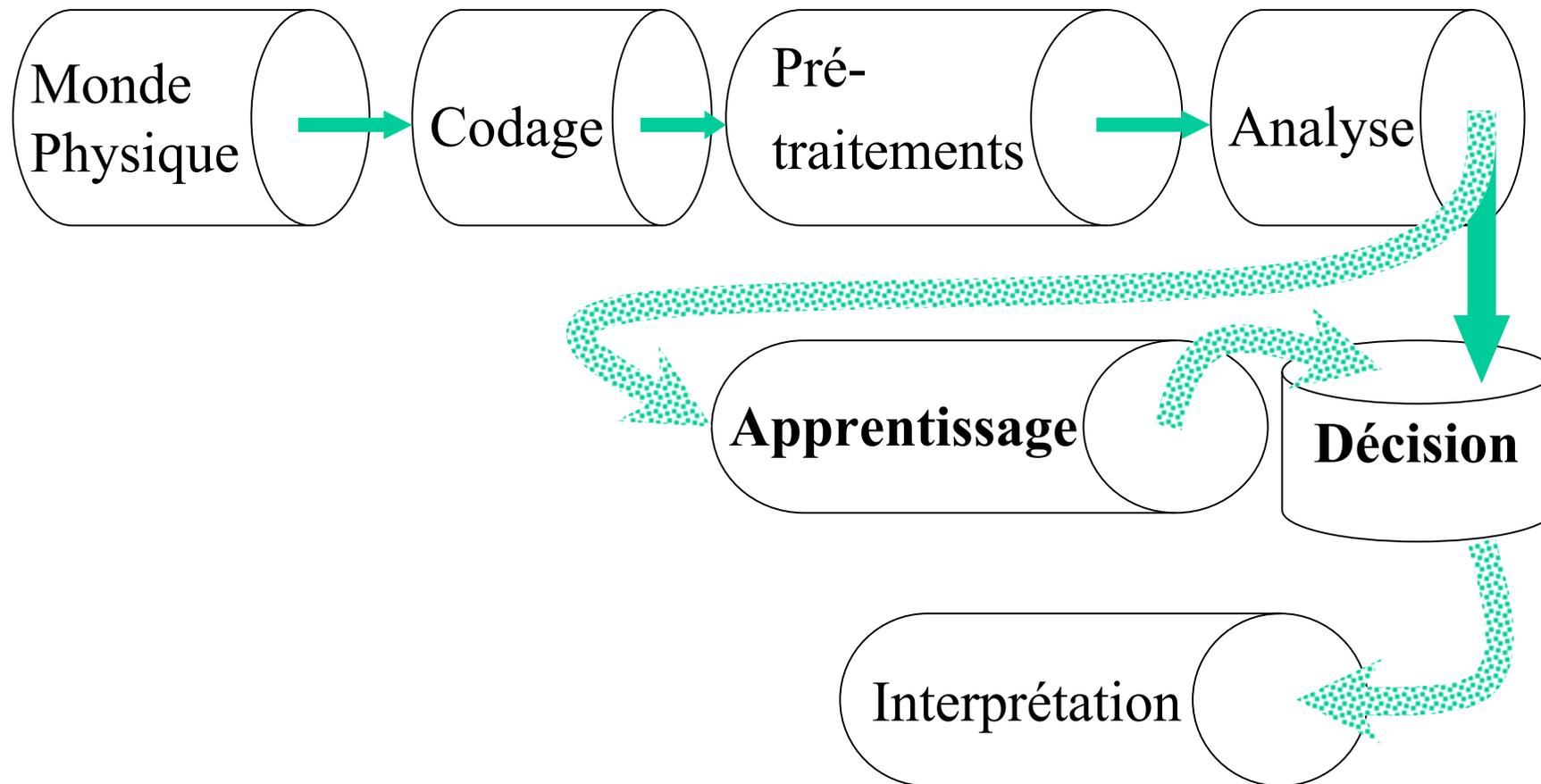
Introduction

Codage

Analyse

Apprentissage & Décision

Systeme de Reconnaissance de Formes :



Introduction

Codage

Analyse

Apprentissage & Décision

Espace
des
Formes

Analogique



Codage



Espace
des
Représentations

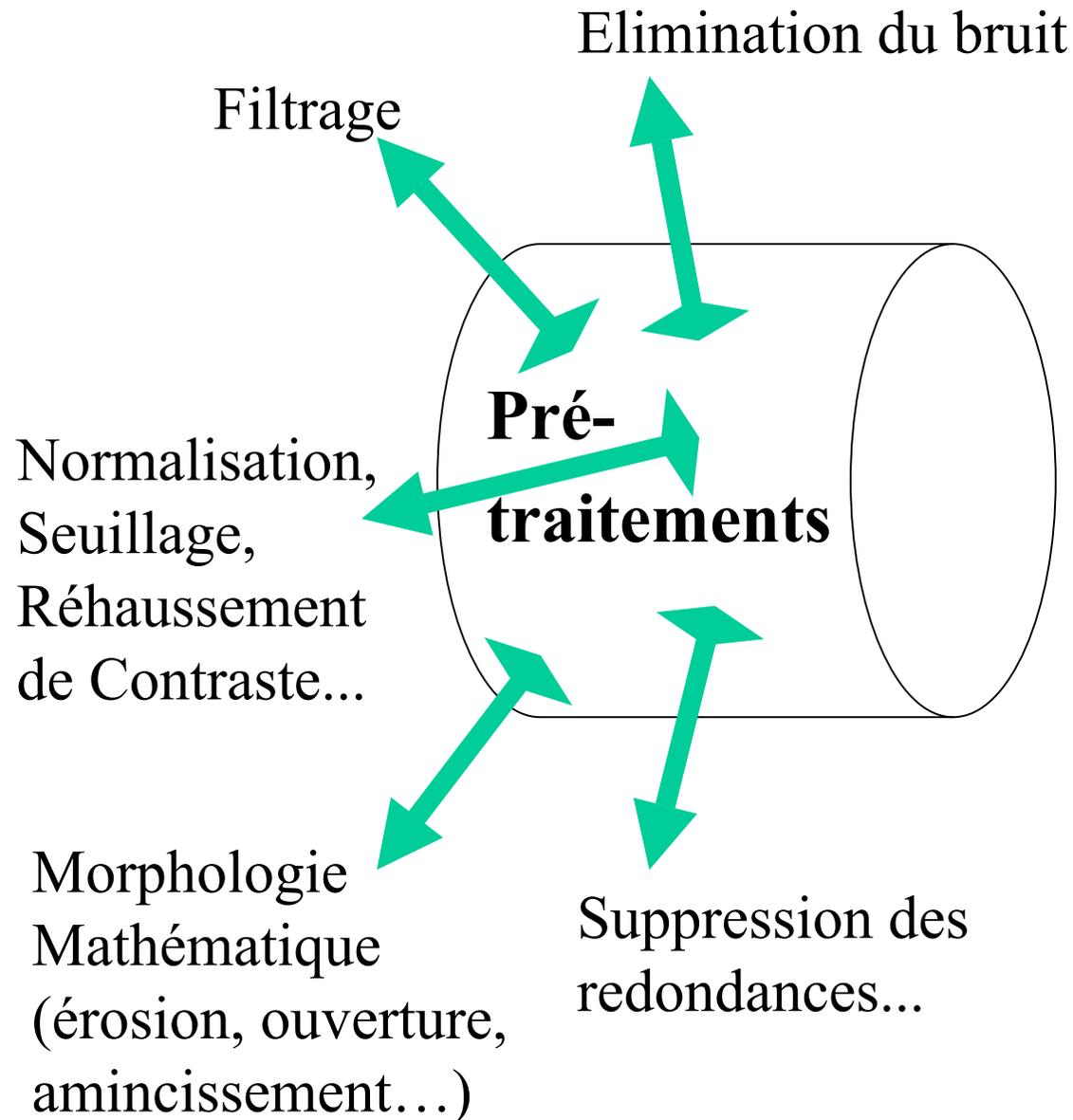
Discret

Introduction

Codage

Analyse

Apprentissage & Décision

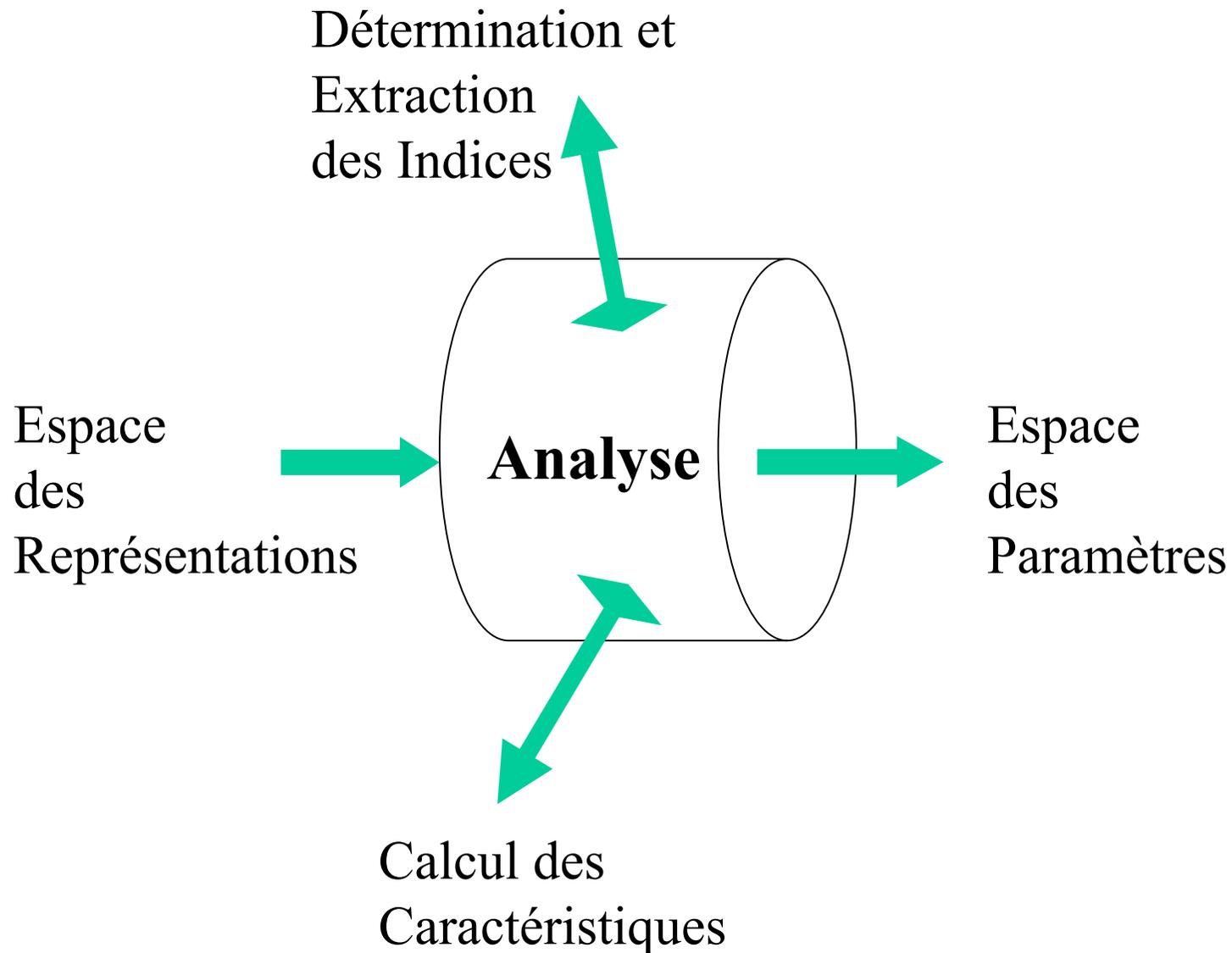


Introduction

Codage

Analyse

Apprentissage & Décision



Introduction

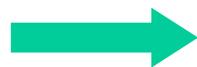
Codage

Analyse

Apprentissage & Décision

Exploitation de la
Connaissance *a priori*

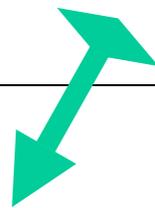
Espace
des
Paramètres



Apprentissage



Espace
des
Noms



Extraction de
Prototypes de Classe
ou Modèles

Introduction

Codage

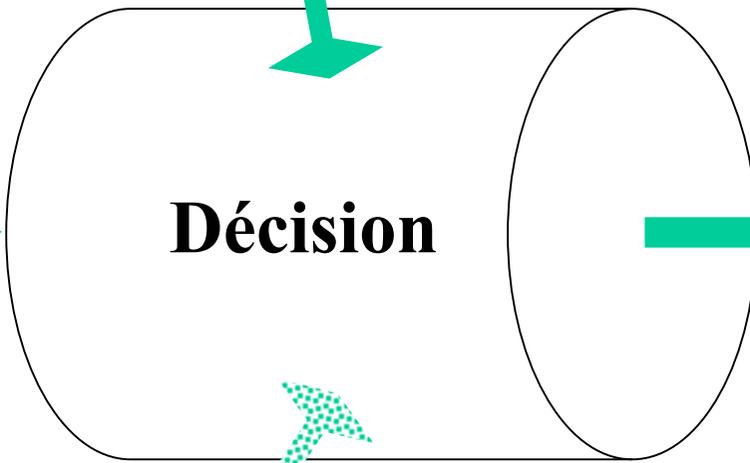
Analyse

Apprentissage & Décision

Classement :

- définitif,
- ambigu avec score
- ou rejet

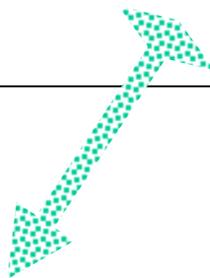
Espace
des
Paramètres



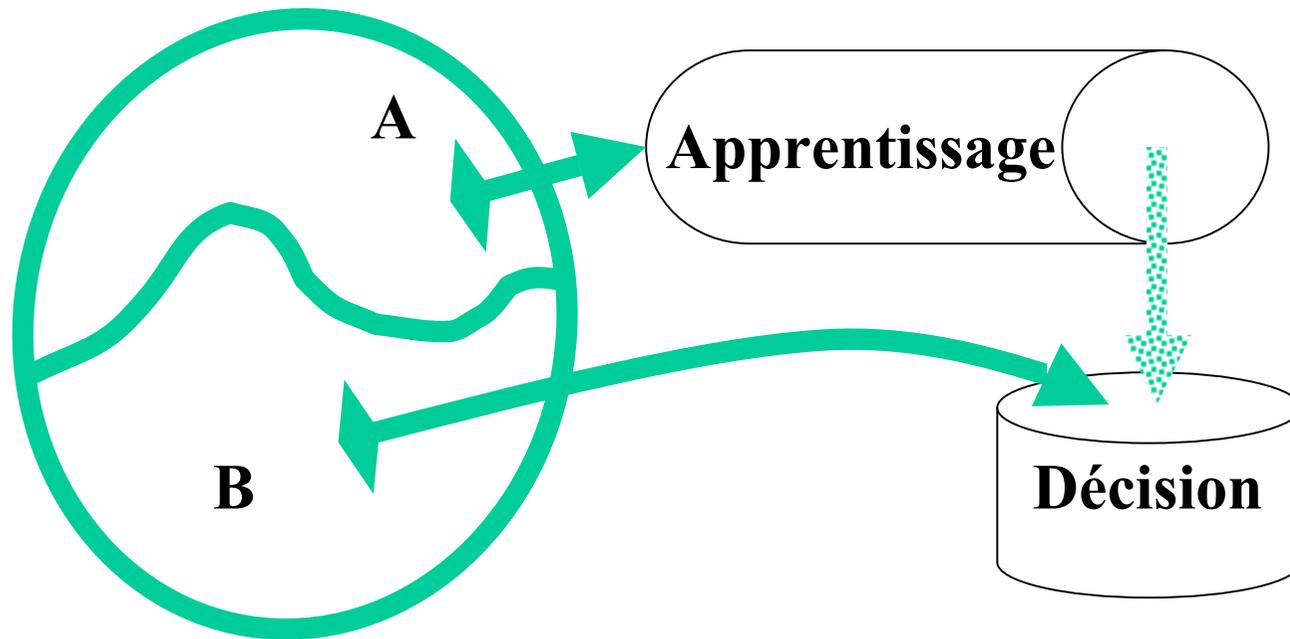
Espace
des
Noms



Interprétation en
fonction du contexte



Ensemble des Formes



A = Ensemble d'échantillons pour chaque classe

$A = \emptyset \Rightarrow$ Apprentissage Non Supervisé

$A \neq \emptyset \Rightarrow$ Apprentissage Supervisé

- Principe de Réduction de dimension des espaces successifs :
 - Importante lors de la phase d 'Analyse
 - Finale lors de la phase de Décision
- Une Histoire de Projections Successives
- Choix Déterminant des Caractéristiques lors de l 'Analyse

Introduction

Codage

Analyse

Apprentissage & Décision

Applications

PAROLE

ECRITURE

VISION

**NEZ
ELECTRONIQUE**

Introduction

Codage

Analyse

Apprentissage & Décision

Applications

- Commande Vocale
- Dictée Automatique
- Traduction Temps Réel de Langues Etrangères
- Rééducation de Mal-Entendants
- Indexation de films

Discipline :

- Décodage Acoustico-phonétique
- Reconnaissance de Mots, de Phrases, de Locuteurs
- Compréhension du Dialogue Oral Homme-Machine

PAROLE

Introduction

Codage

Analyse

Apprentissage & Décision

Applications

- Tri Automatique de Courrier par Lecture et Reconnaissance des Adresses
- Authentification de chèques bancaires
- Saisie et Archivage de Documents
- Reconnaissance de Signatures
- OCR

ECRITURE

Introduction

Codage

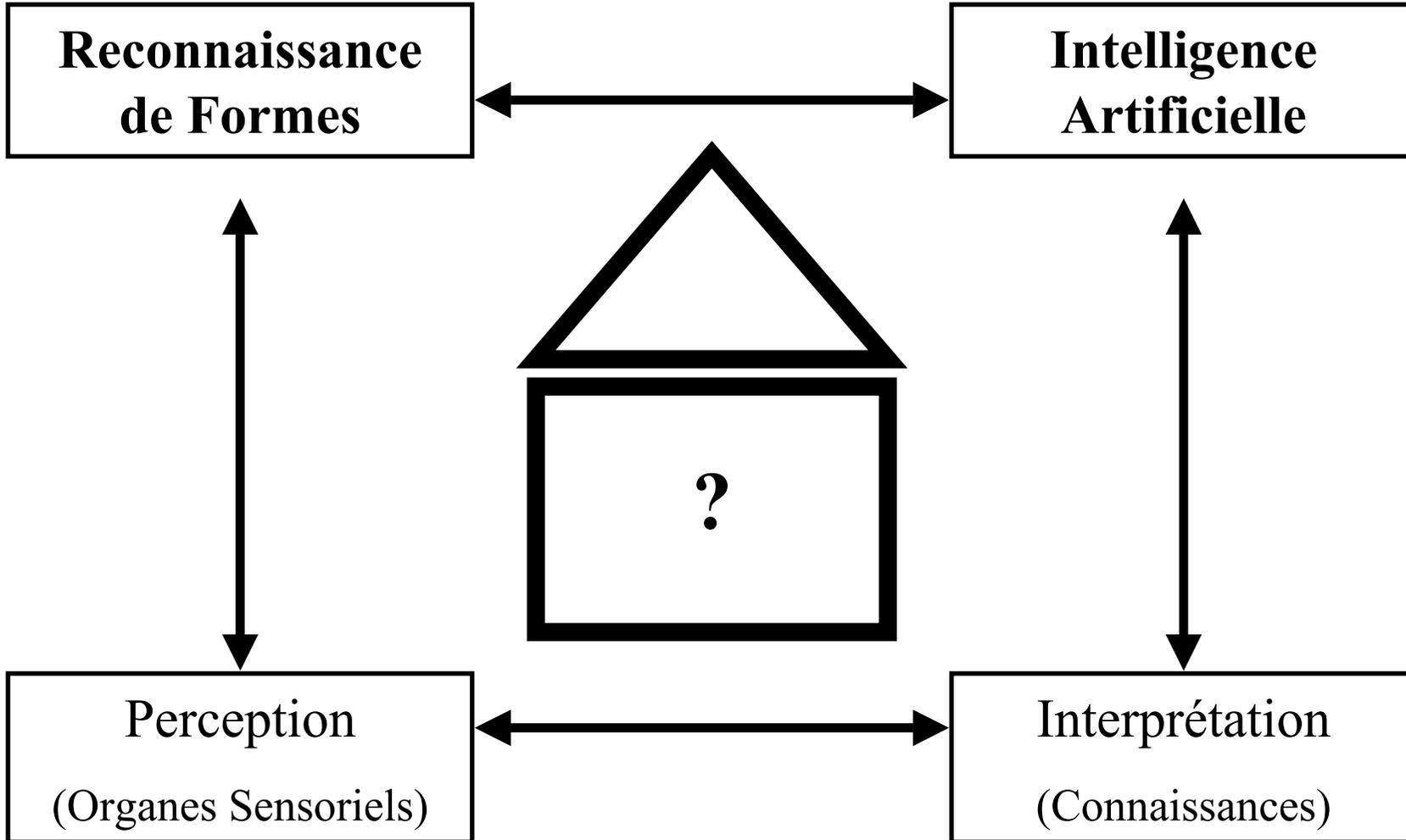
Analyse

Apprentissage & Décision

Applications

- Traitement, Analyse et Interprétation des Images
- Reconnaissance des Empreintes Digitales
- Analyse d'Images de Radiographie ou d'Echographie
- Analyse de Défauts de Pièces d'Usinages
- Surveillance de Processus en Robotique
- Géophysique : Analyse d'Images de Satellites pour les Prévisions Météorologiques
- Analyse du Sol en Pétrographie
- Indexation

VISION

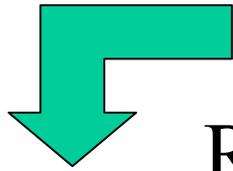


Introduction

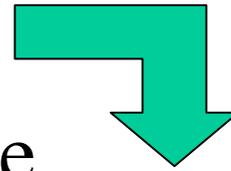
Codage

Analyse

Apprentissage & Décision



2 Voies
pour la
Reconnaissance



Statistique

Théorie de la décision

Approche holistique

Extraction de mesures
caractéristiques globales
ou « *features* »

Partitionnement d'espaces
de paramètres

Structurelle

Théorie des langages

Approche par décomposition

Décomposition en un
ensemble de composants
ou « *pattern primitives* »

Analyse grammaticale selon
des règles syntaxiques

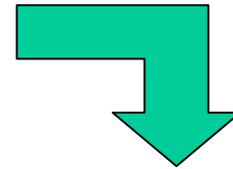
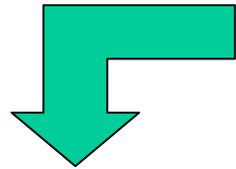
Introduction

Codage

Analyse

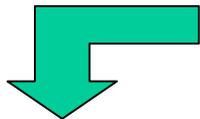
Apprentissage & Décision

2 Voies
pour
1 'Analyse

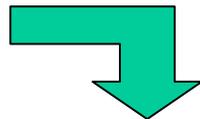


Globale

Formes Simples



Numérique
ou
Quantitative



Logique
ou
Qualitative

Structurelle

Formes Riches en
Informations Structurelles

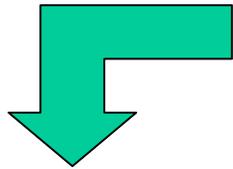
Description en Formes Primitives
et
Agencement Relatif de ces Sous-
Formes

Introduction

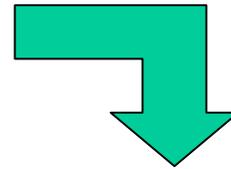
Codage

Analyse

Apprentissage & Décision



2 Voies
pour
la Décision



Statistique

Classification Automatique

Discrimination Fonctionnelle

Méthodes Connexionnistes

Méthodes Statistiques Bayésiennes

Les k Plus Proches Voisins

Structurelle

Structures de Graphes

Structures Syntaxiques

Programmation Dynamique

Méthodes Stochastiques

Reconnaissance de chiffres manuscrites :

1. Réfléchir sur la forme des chiffres et en faire l'analyse : orientations statistique, structurelle, statistico-structurelle.
2. Définition des caractéristiques discriminantes.
3. Extraire l'ensemble de ces attributs caractéristiques sur un ensemble d'apprentissage.
4. Déterminer les séparatrices des classes sur l'ensemble d'apprentissage ou entraîner un réseau de neurones...
5. Lancer la décision (affectation à une des 10 classes) sur les formes présentées à l'aide du processus de décision choisi pour l'analyse : traitement de l'image pour segmenter les chiffres, les prétraiter, extraire les caractéristiques...

Introduction

Codage

Analyse

Apprentissage & Décision

Exemple

Codage



Prétraitement



- Filtrage du bruit
- Normalisation
- Squelettisation
- Segmentation

Analyse



Décision

« 3 »

avec un score de
0.6

Apprentissage

```
0000000000333333
0000000000033333
0000000000043333
0000000000033333
0000000000083555
0000000000088555
9990000000888555
999966666888555
99996666677755
99996666677775
222226666777715
222222664777111
222222444777111
222224444777111
222224444777111
222224444777111
```

Exemple

- Problème de la segmentation de mots en caractères ou graphèmes

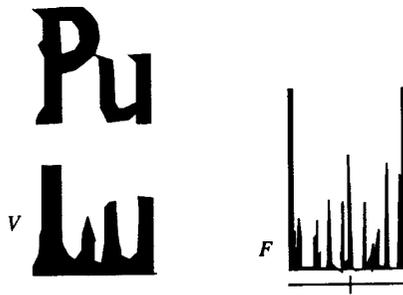


Figure 6.45. *Emplacement de la ligature par projection verticale, d'après [Kah87].*

- ⇒ Approche holistique de la reconnaissance de mots
- ⇒ Programmation Dynamique

Introduction

Codage

Analyse

Apprentissage & Décision

Image

Un format portable : *pgm* ou *ppm* ou *pbm*

P2

#feep.pgm

10 5

4

0 0 0 1 1 4 4 3 0 0

0 1 4 1 1 4 4 3 0 0

0 0 3 1 1 4 4 3 0 0

0 0 0 1 1 3 3 3 0 0

0 0 0 1 1 2 2 2 2 2

P3

example from the man
page

4 4

15

0 0 0 0 0 0 0 0 15 0 15

0 0 0 0 15 7 0 0 0 0 0 0

0 0 0 0 0 0 0 15 7 0 0 0

15 0 15 0 0 0 0 0 0 0 0 0

Logiciels libres de manipulation: *gimp* et *imageJ*

Exercice : <http://sip-crip5.org/lomn/ti.html>

Création des biomorphes par fractales

Noir et blanc

Pour chaque point du plan imaginaire ($z=x+iy$)

On lance une boucle de 10 itérations :

On calcule $z=f(z)$

Si ($|x|>10$ ou $|y|>10$ ou $|z|>10$),
on quitte la boucle.

En fin de boucle :

Si ($|x|>10$ ou $|y|>10$),
on marque un pixel noir sur fond blanc.

Couleur

La couleur dépend du nombre d'itérations et de la valeur de $|x|$, $|y|$, $|z|$.

Fonction $f(z)$

$$f(z) = z^{\text{Exposant}} + C$$

Ensembles de Julia

$c=0.3+0.6i$	$c=-0,4+0,7i,$
$c=-0.0519+0.688i$	$c=-0,7+0,3i$
$c=0.32+0.43i$	$c=-1,77+0,01i$
$c=-0.0986-0.65186i$	$c=-0,15+0,45i$
$c=0+i$	



Qu'est-ce que la parole :

- Parole = Onde
- Intensité : débit du flot d'air +
ouverture des cordes vocales
- Hauteur : fréquence de vibration des cordes vocales
pour les sons voisés
- Timbre : modulation de l'onde
par la forme du conduit vocal



Travail du phonéticien :

- Isoler l 'onde continue en « segments » phonétiques
- Classifier les sons d 'une langue
- Utilisation d 'une unité sonore minimale : le phonème

Introduction

Codage

Analyse

Apprentissage & Décision

Parole

Caractéristiques Générales des Voyelles du Français :

- Vibrations des Cordes Vocales
- Lieu de l'articulation
- Stabilité des Articulateurs
- Deux classes : voyelles orales et voyelles nasales

Introduction

Codage

Analyse

Apprentissage & Décision

Parole

Caractéristiques Articulatoires des Voyelles du Français :

- Aperture Vocalique (petite, moyenne et grande)
- Lieu d 'Articulation (antérieure, postérieure)
- Intervention des Lèvres (labiales et non labiales)

Introduction

Codage

Analyse

Apprentissage & Décision

Parole

Classement Articulateur des Voyelles Orales du Français :

Voyelles	Fermées	Semi-Fermées	Semi-Ouvertes	Ouvertes
Antérieures non labiales	i	e	ɛ	a
Antérieures Labiales	y	ø	ɔ œ	
Postérieures Labiales	u	o	ɔ	

Aigu
↑
↓
Grave

Diffus ← → Compact

Introduction

Codage

Analyse

Apprentissage & Décision

Parole

Caractéristiques Acoustiques des Voyelles du Français :

- Sons Voisés
- Résonance de l'onde glottique dans la cavité orale
- Formants F_i = Maxima du spectre d'énergie par ordre décroissant

Introduction

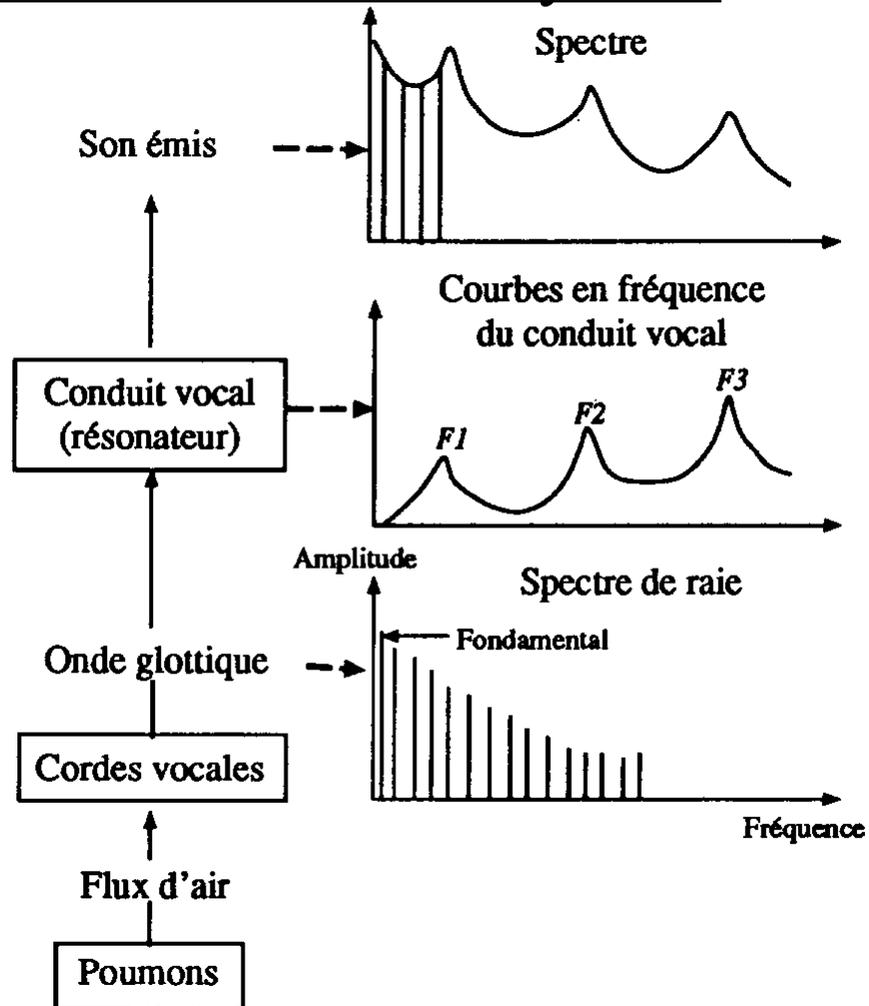
Codage

Analyse

Apprentissage & Décision

Parole

Schéma de Production d'une Voyelle :



Introduction

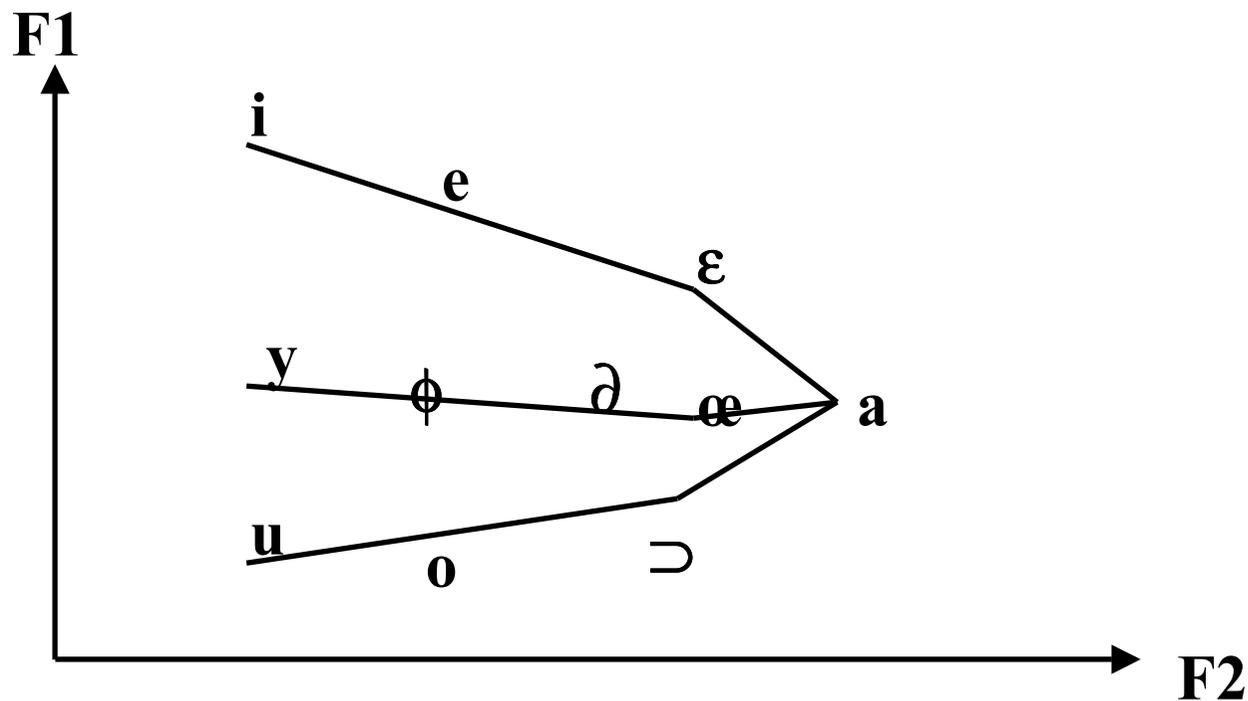
Codage

Analyse

Apprentissage & Décision

Parole

Classement Acoustique des Voyelles Orales du Français
selon les deux premiers formants F1 et F2 :



Introduction

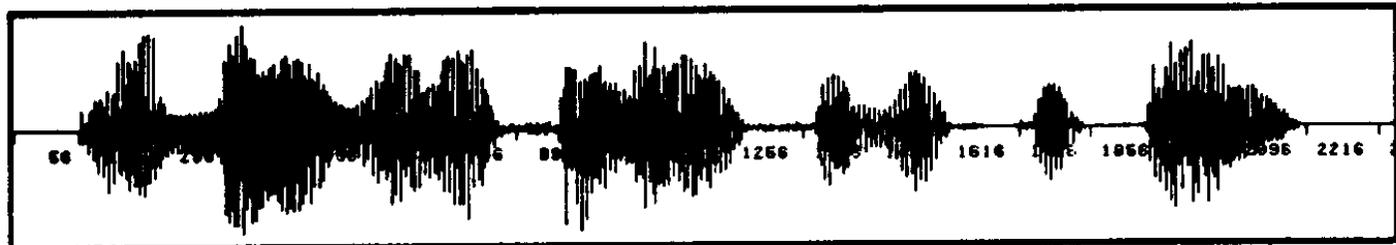
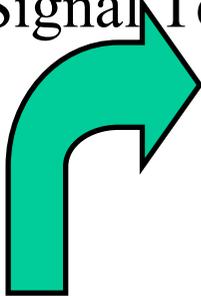
Codage

Analyse

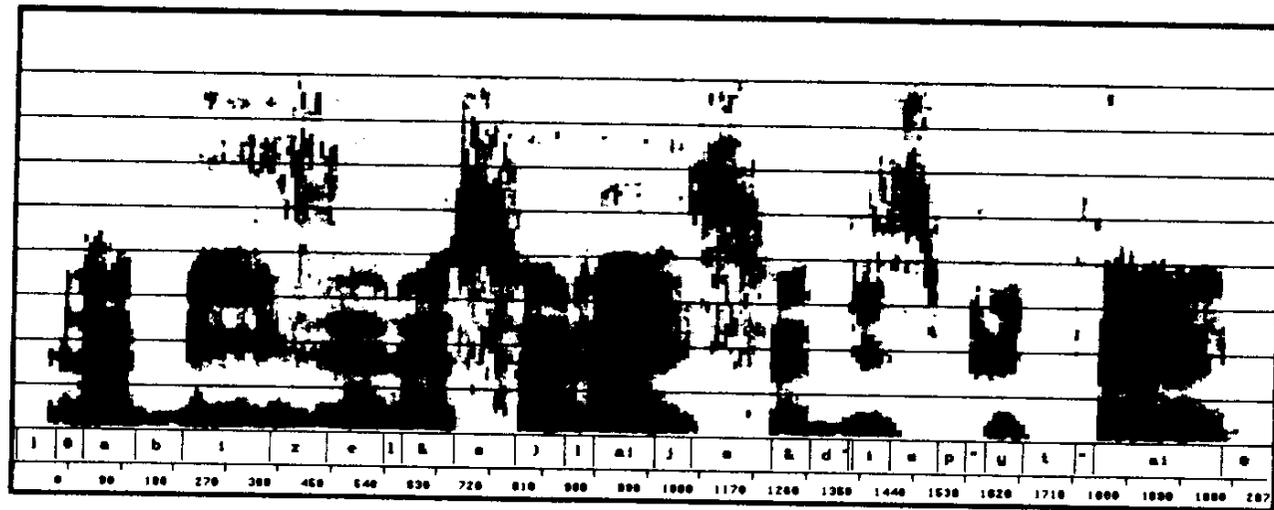
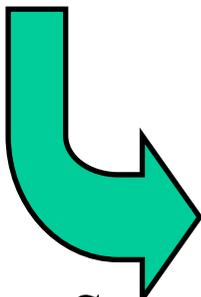
Apprentissage & Décision

Parole

Signal Temporel Brut



Modèles de
Représentation



Spectrogramme



Une image peut être modélisée de plusieurs façons :

- une fonction continue de deux variables $f(x,y)$
- une matrice 2D, discrétisant la fonction $f(x,y)$
- la réalisation d'une Variable Aléatoire

-> modèle de traitement mathématique, algorithmique, physique



- Suppression du Bruit (hautes fréquences ou bruit blanc)
- Correction des Erreurs
- Homogénéisation des Données
- Réduction des Données

Introduction

Codage

Analyse

Apprentissage & Décision

Prétraitements

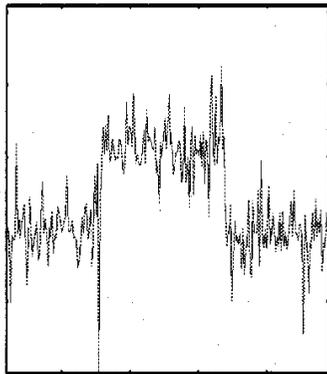
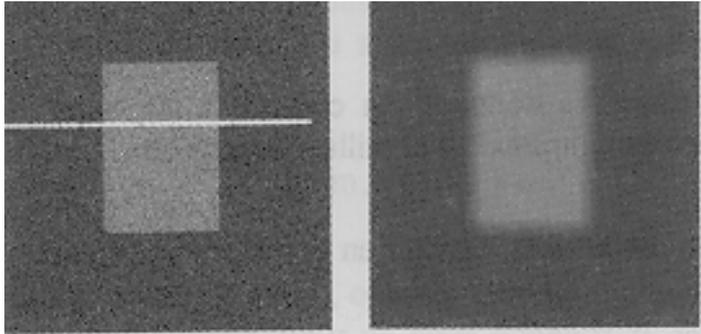
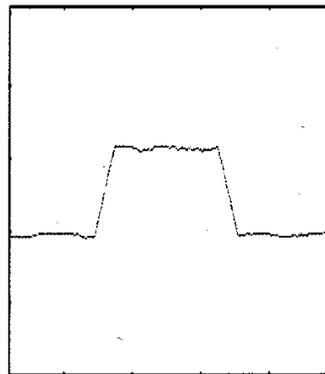


image initiale



filtrage moyennneur 15x15

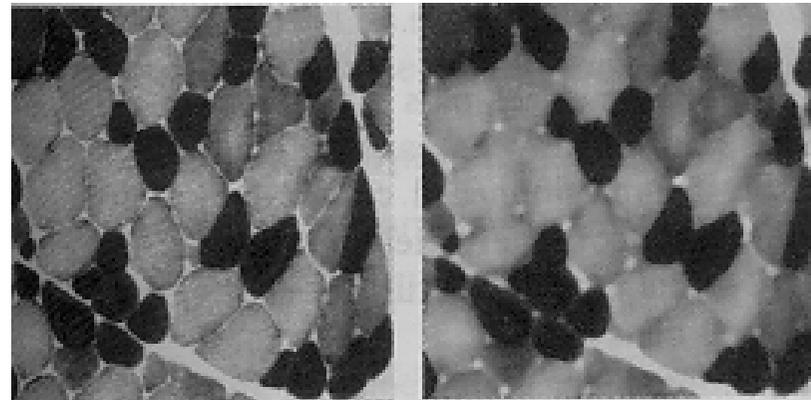


image initiale

Filtre médian

Introduction

Codage

Analyse

Apprentissage & Décision

Prétraitements

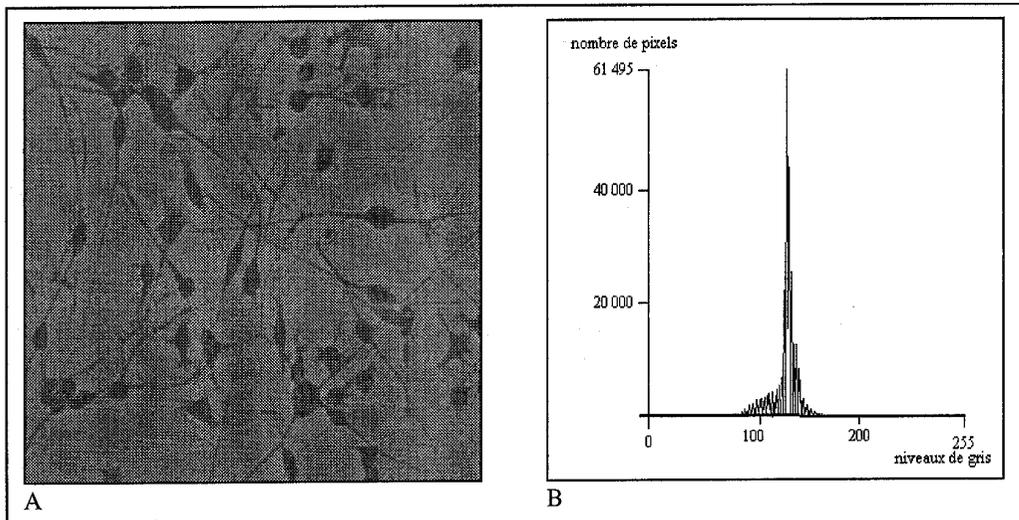


Figure IV.2 : Image de Neuroblastomes de souris différenciés et son histogramme

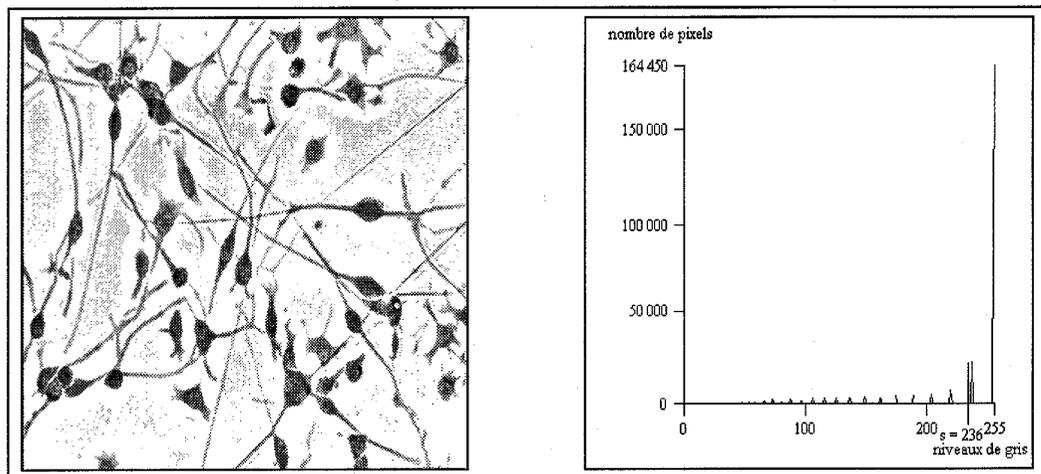


Figure IV.40 : Image de neuroblastomes de souris différenciés pré-traitée et son histogramme sur lequel on a placé la valeur du seuil déterminée automatiquement



Analyse d'images

Indices Visuels ou Primitives :

- Régions
 - Contours
- } Primitives
Duales

Introduction

Codage

Analyse

Apprentissage & Décision

Images

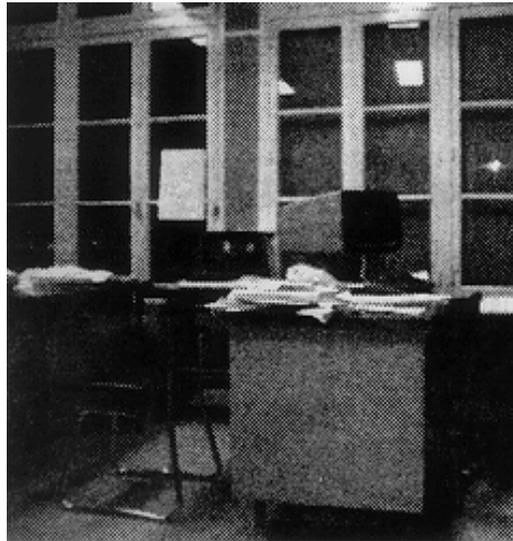
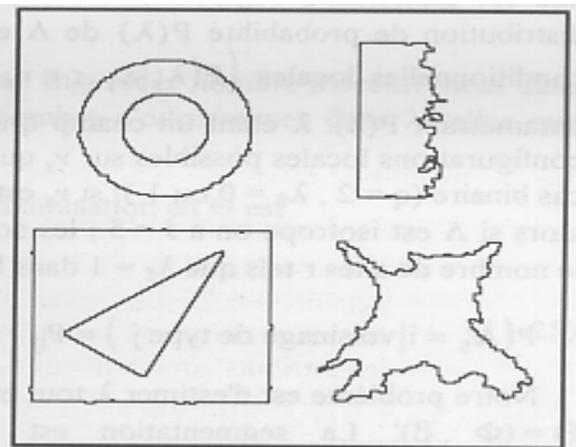
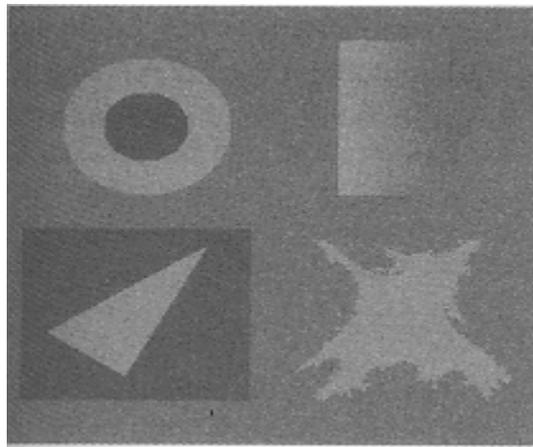


Image originale



Images segmentées en régions



Introduction

Codage

Analyse

Apprentissage & Décision

Images

Paramètres calculés sur les Indices Visuels:

- Topologiques : *Surfaces, Périmètres, Concavités, Trous*
- d' Orientations : *Moments d' Inertie du Second Ordre*
- de Formes : *Compacité, Allongement*
- Statistiques : *Histogrammes, Moyennes, Variances, Entropie, Dissymétrie, Aplatissement*
- Relationnelles : $f(R_i, R_j) : f = \text{Plus-Grand, Au-dessus, Adjacent} \dots$

Indices Visuels :

- Frontières Homogènes

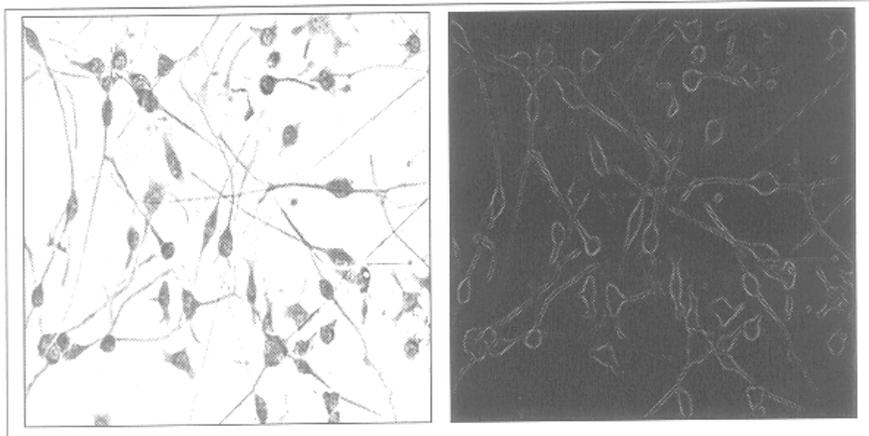


Image de contour par
Deriche

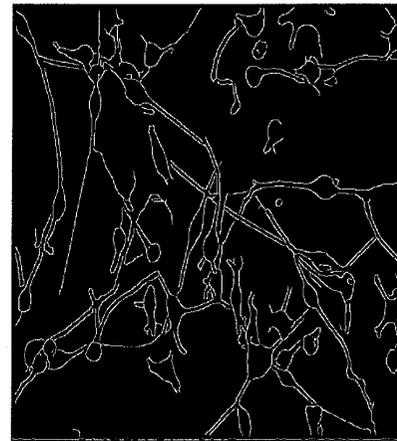


Figure IV.22 : Deriche ($\alpha = 1$) sur l'image
originale suivi d'un seuillage par hystérésis
($S_h = 25$, $S_b = 5$)

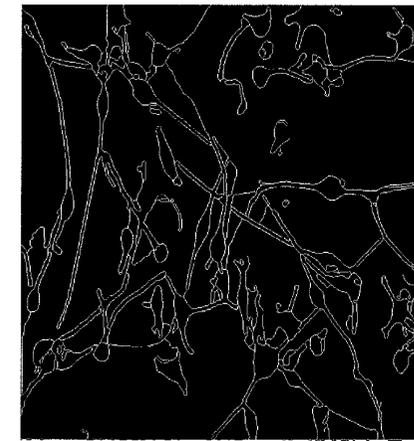


Figure IV.23 : Fermeture de contours sur
l'image de la figure IV.22

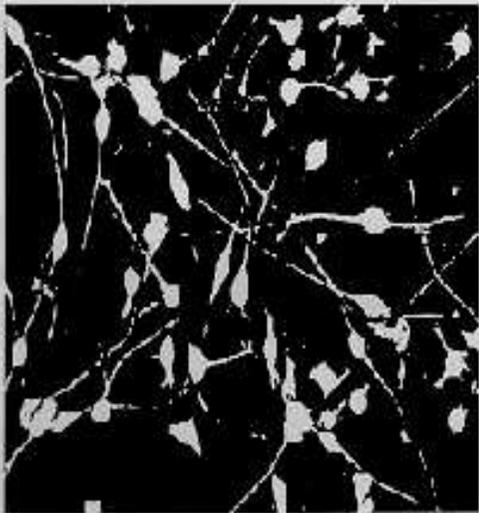


Figure IV.41 : Image binaire obtenue par le seuillage automatique

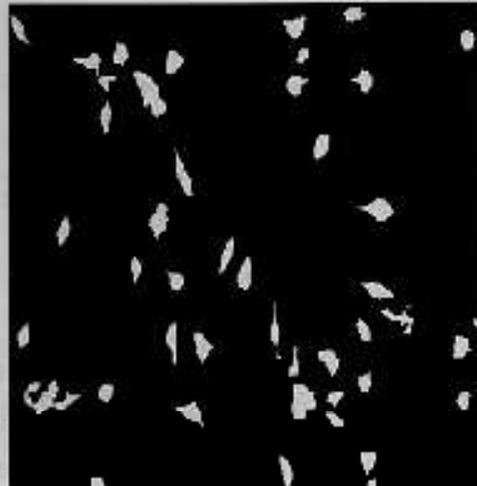


Figure IV.43 : Image des germes

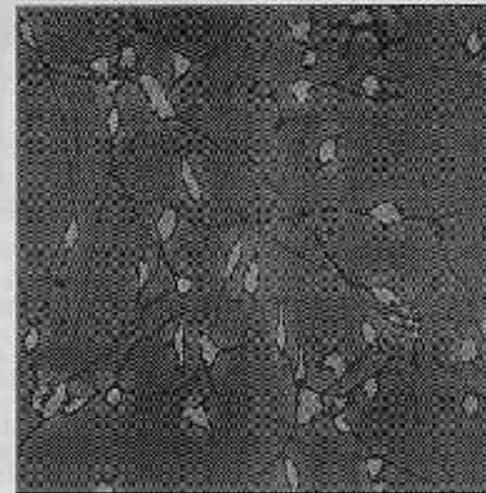


Figure IV.44 : Superposition de l'image des germes et de l'image originale

Introduction

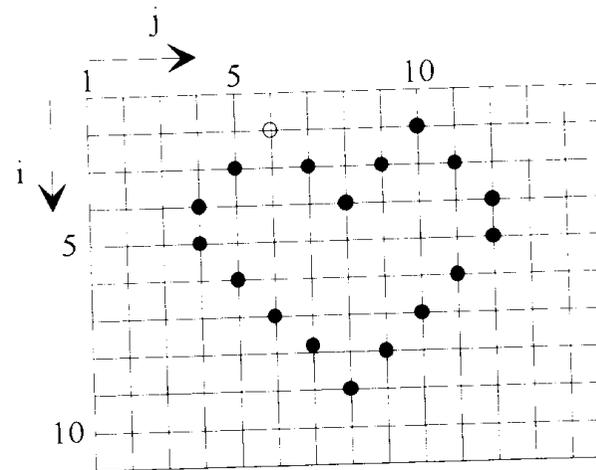
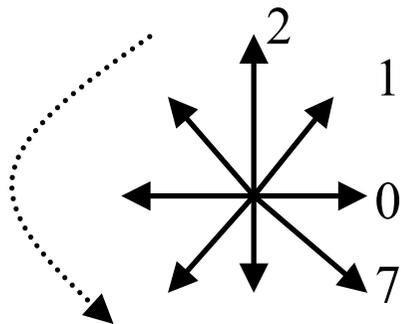
Codage

Analyse

Apprentissage & Décision

Images de
Contour

- Approximation polygonale
- Codage de Freeman ...



$P_0 = [1,6]$ et suite des directions : $\{7,7,1,1,7,7,6,5,5,5,5,3,3,3,3,2,1,1\}$

Figure 3.17 : Exemple de code de Freeman

Introduction

Codage

Analyse

Apprentissage & Décision

Images de
Contour

Paramètres :

- Convexité
- Abscisse Curviligne
- Courbure

Introduction

Codage

Analyse

Apprentissage & Décision

Parole

Indices phonétique ou linguistique ou

Primitives :

- Mots
- Syllabes
- Phonèmes

Paramètres d 'analyse :

- Temporelle, Fréquentielle
- Cepstrale

Introduction

Codage

Analyse

Apprentissage & Décision

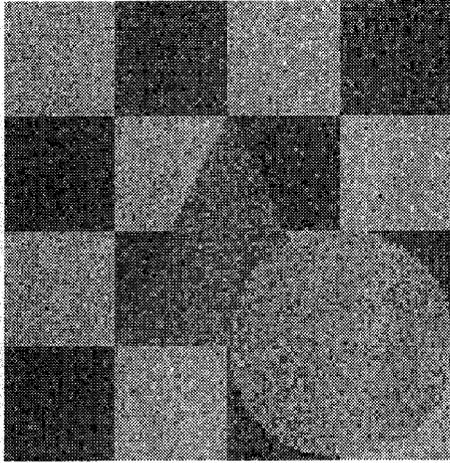
- Segmentation : clé de voûte de l 'Analyse
d 'Image ou de Parole
- Segmenter - Reconnaître : Paradoxe et Confusion

Introduction

Codage

Analyse

Apprentissage & Décision



a) Image 1.a bruitée : SNB = 5dB

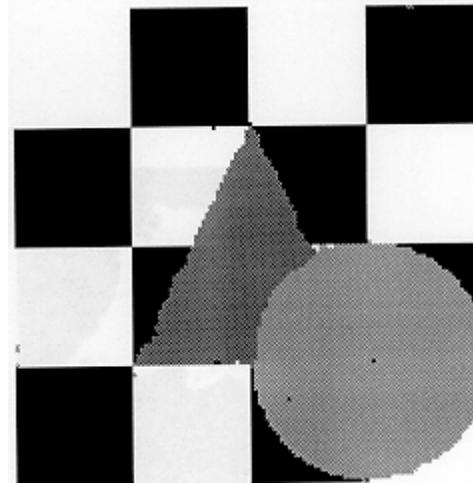
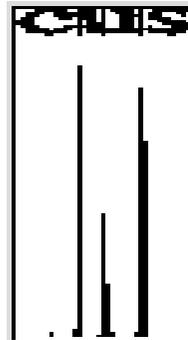


Image « segmentée » = abus de langage ?

According to Nita, there is no way to go through customs at Atlanta without changing



classification

segmentation

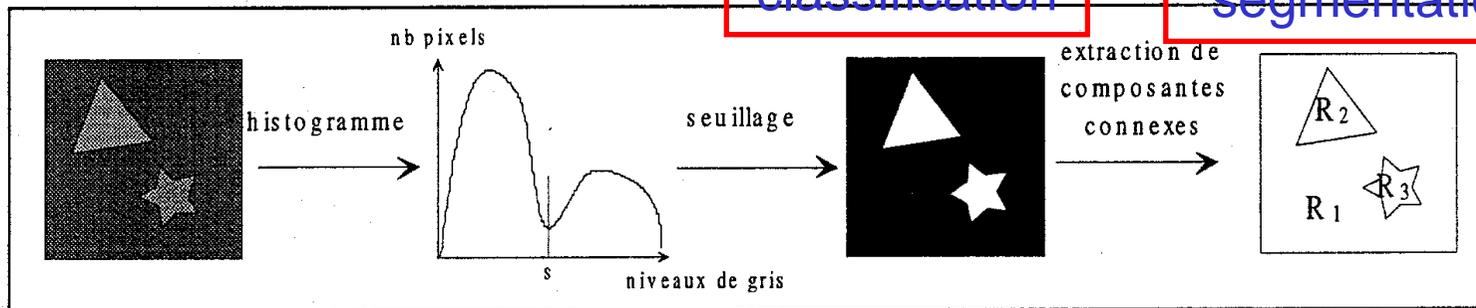
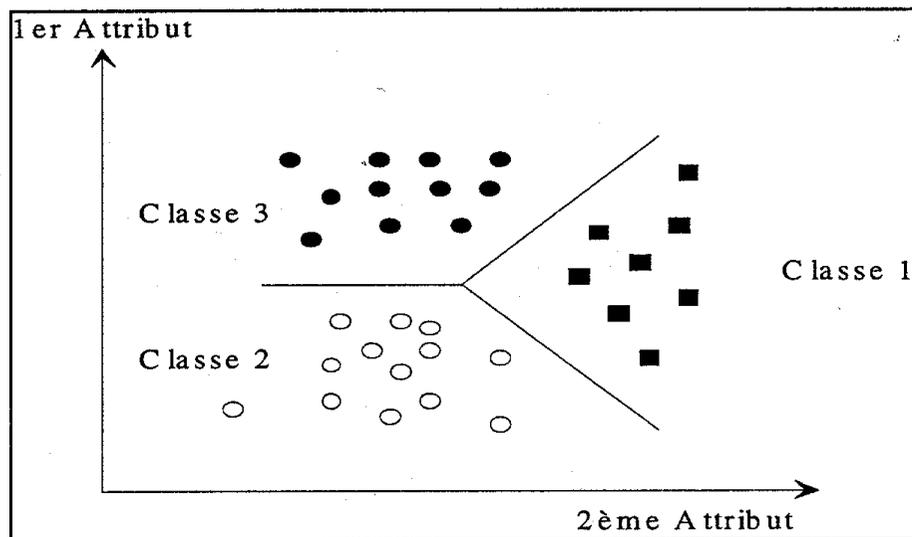
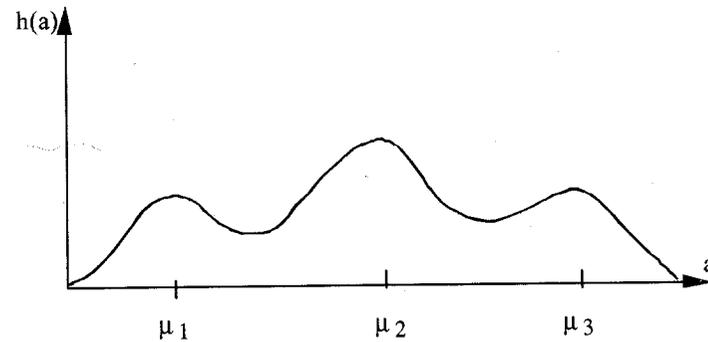
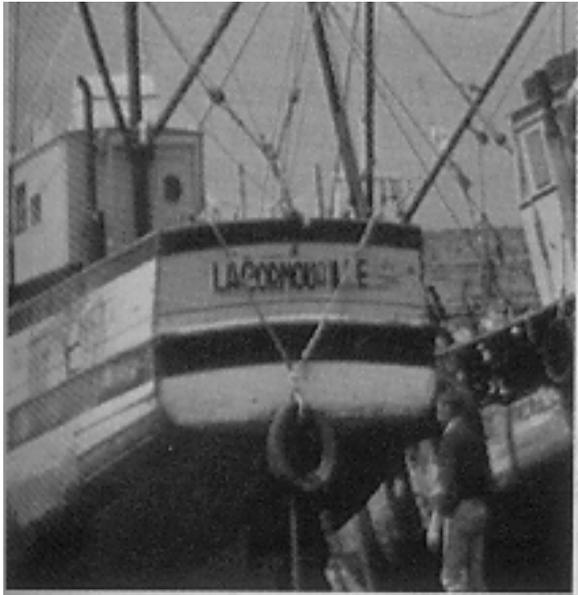


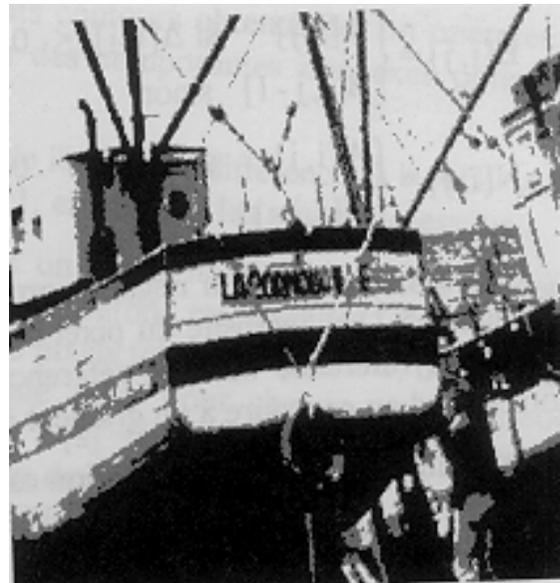
Figure III.7 : Segmentation en régions par seuillage

Figure III.8 : Classification des points dans l'espace des attributs
(3 classes, dimension 2 de l'espace)

Segmentation, souvent égale à de la **classification automatique de pixels** dans un espace \mathfrak{R}^n

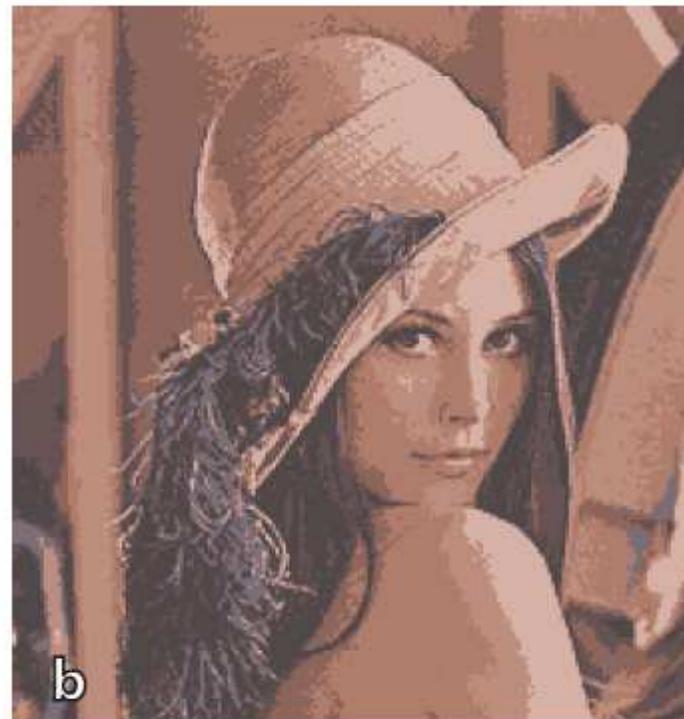


Quantification en niveau de gris dans \mathfrak{R}^1 ou en couleur dans \mathfrak{R}^3



Exemples :

a - image initiale

b - quantification couleur par la méthode
des nuées dynamiques. ((7 classes)**Applications :**

Cosmétique, Diagnostic, quantification couleur.

Introduction

Codage

Analyse

Apprentissage & Décision

Classification Automatique

- Cadre typique de la classification **non supervisée**, cad sans exemples.
- Notion de Proximité dans un espace métrique
- Principe : Classification et Décision par la Distance
- Propriétés d'une Classification Idéale :
 - Compacité
 - Séparabilité
- Dans la pratique, Défaillance de Formes (« floues »)

- Distance Point-Point:

$$d(p,m) = \textit{Hamming, Euclidienne}$$

- Distance Point-Classe :

$$d(p,C) = \inf \{d(p,m); m \in C\}$$

- Distance Classe-Classe :

$$d(C1,C2) = \inf \{d(p,m); p \in C1 \text{ et } m \in C2\}$$

Introduction

Codage

Analyse

Apprentissage & Décision

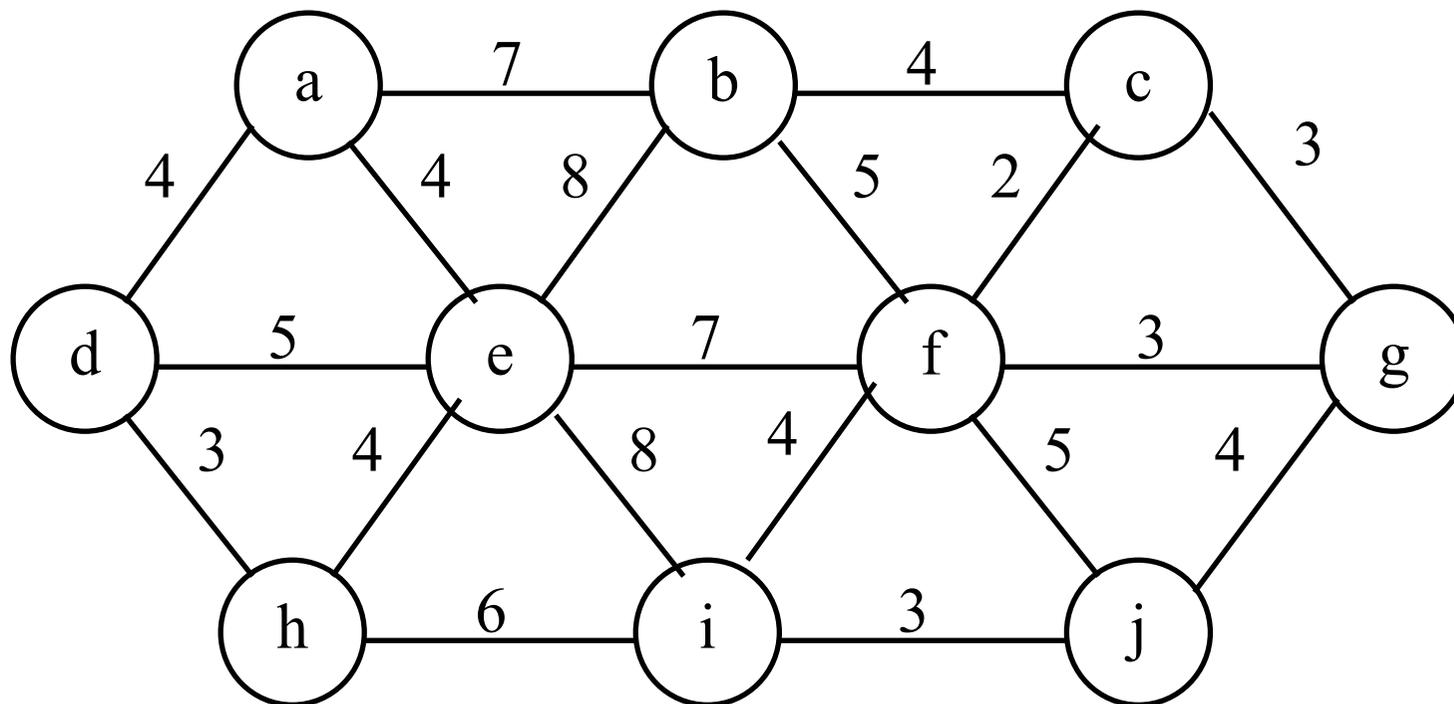
Classification Automatique

- RF \Leftrightarrow Définition de Distances Adaptées

- Attribution de x à $C_k \Leftrightarrow C_k = \mathit{Argmin}_{C_i} d(x, C_i)$

Arbre de Longueur Minimale

Le Graphe des Distances :



Arbre de Longueur Minimale

Algorithme de Prim :

Initialiser le graphe résultat à un point quelconque

pour i de 2 à n (nombre de points du graphe) faire

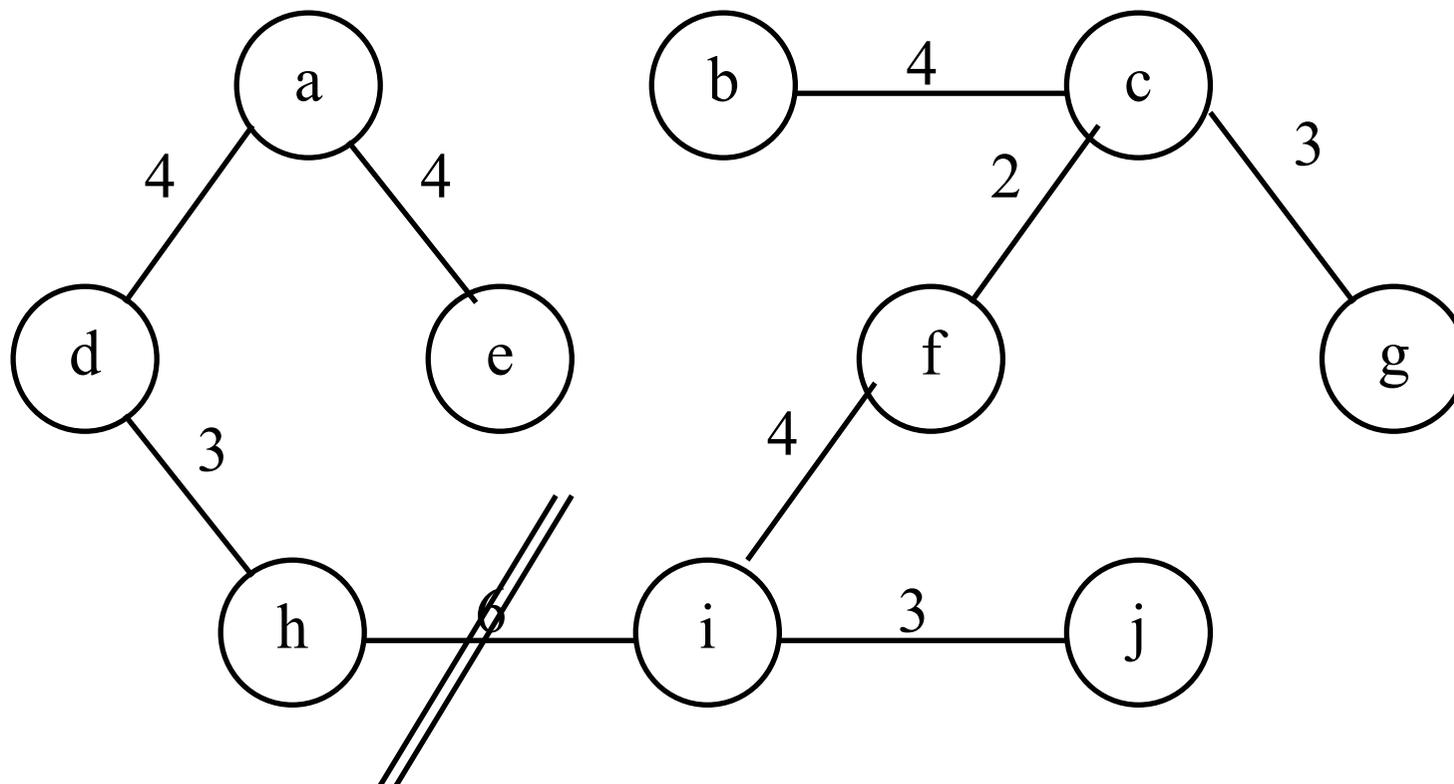
Chercher une arête (x,y) de coût minimal joignant un point du graphe résultat à un point extérieur au graphe résultat

L'ajouter au graphe résultat : $g \leftarrow g \cup (x,y)$

fin pour

Arbre de Longueur Minimale

*Arbre de Longueur Minimale et
Application à la Classification :*



Introduction

Codage

Analyse

Apprentissage & Décision

Classification Automatique

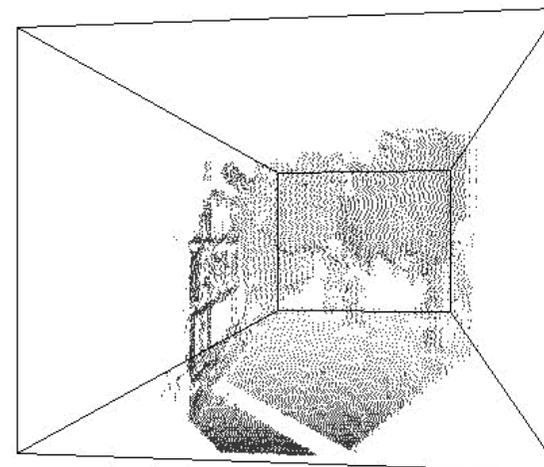
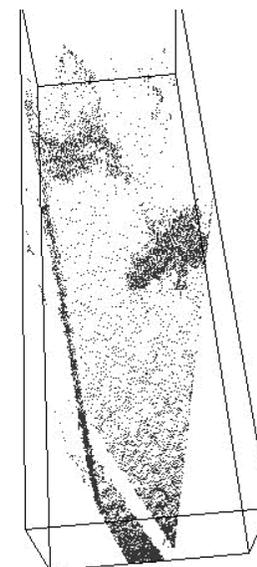
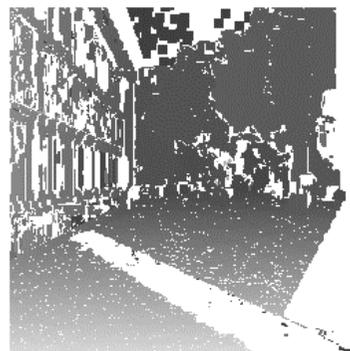
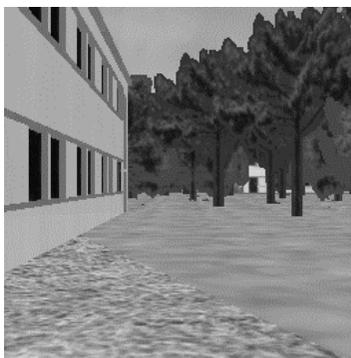
Arbre de Longueur Minimale

- Lié au groupement hiérarchique et à la notion de distance ultramétrique
- Sensibilité au choix de la distance
- Instabilité Structurelle ou Effet de Chaîne

Nuées Dynamiques ou K-Means

- Partition en K classes d'un ensemble E :
 - Minimisant un critère global J :
 - A partir d'éléments suffisamment représentatifs de chaque classe ou noyau
- Algorithme à **convergence rapide** vers des **minima locaux** pour classer **plusieurs milliers d'objets**

« Clustering »



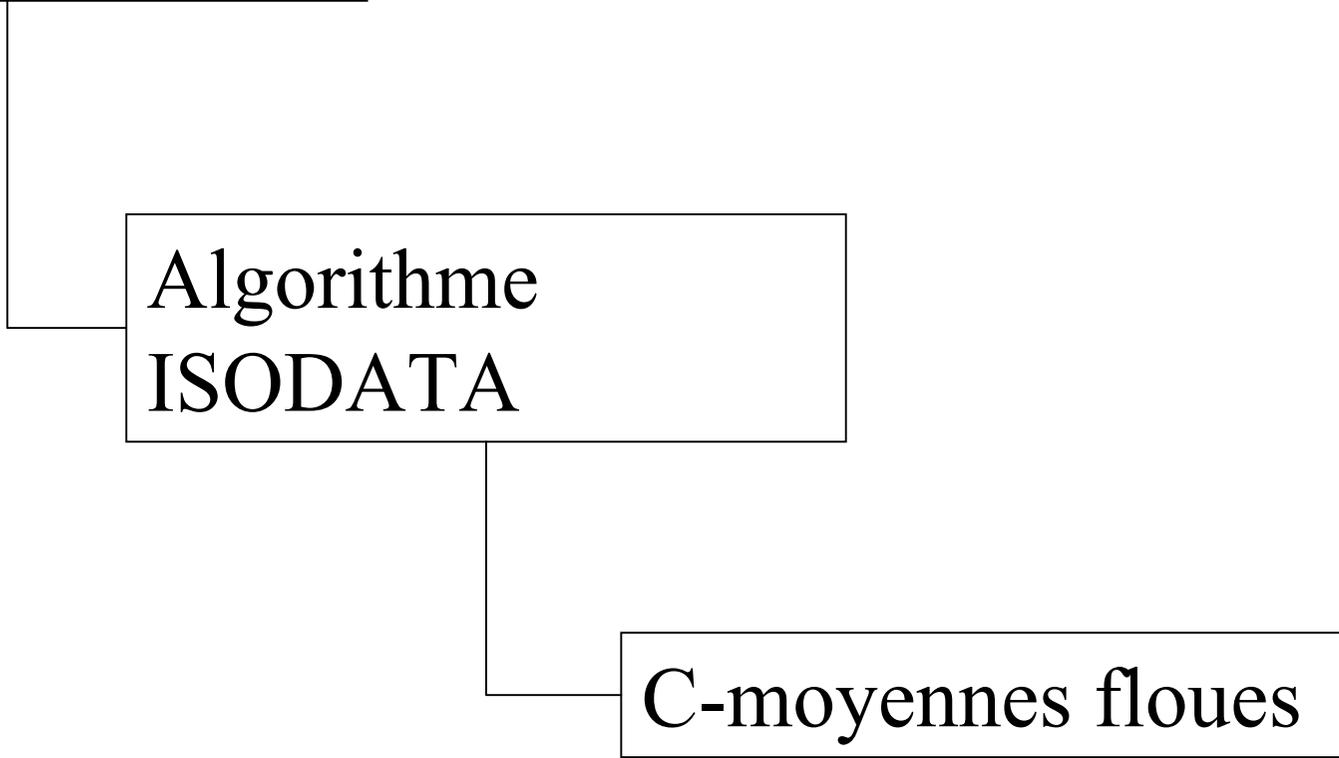
« Clustering »



Nuées Dynamiques

Algorithme
ISODATA

C-moyennes floues

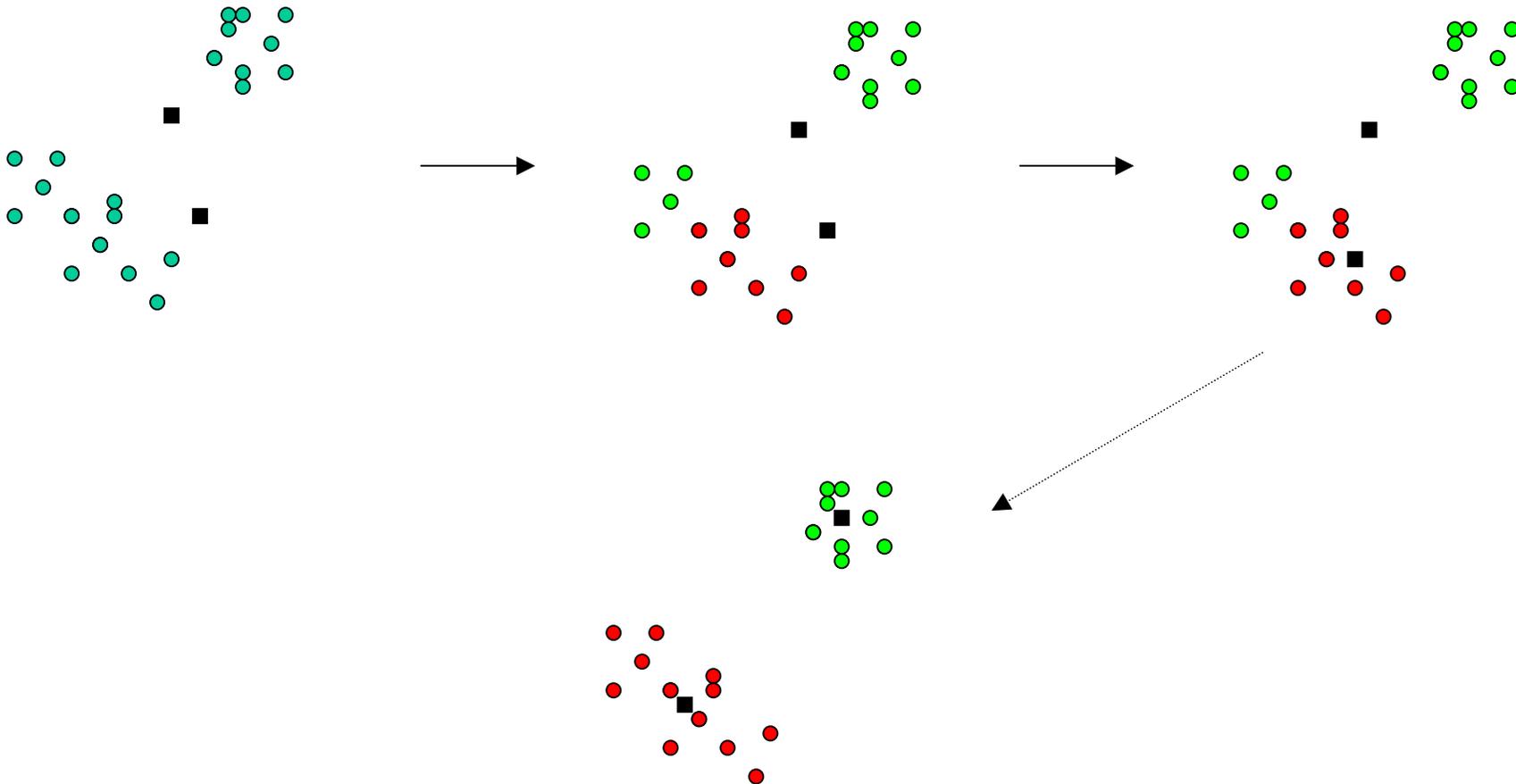


« Clustering »



Principe des Nuées Dynamiques

DIDAY



« Clustering »



Algorithme itératif de type ISODATA

- ⇒ Nombre de groupements C connu
- ⇒ Minimisation itérative d'une fonctionnelle $J(V)$
- ⇒ V est un vecteur de paramètres de forme : on peut prendre par exemple le barycentre m_c d'un nuage de points C .

« Clustering »

Cas « Général »



Principe des C-moyennes

⇒ Minimisation de la fonction objective :

$$J_m(U;V) = \sum_{j=1}^C \sum_{i=1}^N u_{ij}^m d(x_i, V_j)$$

« Clustering » Cas « Crisp » : $u_{ij} \in \{0, 1\}$

Nuées Dynamiques ou K-Means ou C-moyennes

Algorithme :

1. Choisir au hasard C Noyaux $\{Y_1^0, Y_2^0, \dots, Y_K^0\}$
2. $n \leftarrow 0$
3. $n \leftarrow n+1$
4. **Pour chaque** valeur de k (de 1 à C), calculer
$$\omega_k^n \leftarrow \left\{ x \in E \mid \forall j \neq k, f(x, Y_k^{n-1}) \leq f(x, Y_j^{n-1}) \right\}$$

Calcul de Y_k^n à partir de ω_k^n

Fin pour

5. **Si** $\omega_k^n \neq \omega_k^{n+1}$ et $n \leq n_0$, retourner en 3

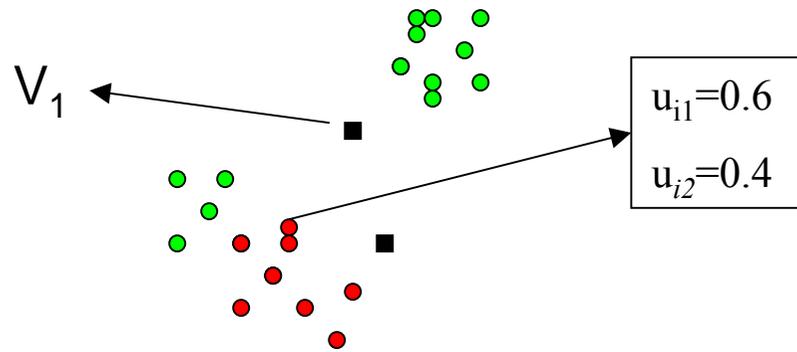
Sinon arrêt

Fin Si

« Clustering »

Cas « Fuzzy » : $u_{ij} \in [0, 1]$

⇒ Coefficient d'appartenance u_{ij} dans $[0, 1]$



⇒ Dans le cas où $m=2$, minimisation de la fonctionnelle suivante :

$$J(U; V) = \sum_{j=1}^C \sum_{i=1}^N u_{ij}^2 d(x_i, V_j)$$

« Clustering »

Cas « Fuzzy » : $u_{ij} \in [0, 1]$

En utilisant la formulation lagrangienne, on montre que minimiser J revient à résoudre ce système couplé :

$$\left\{ \begin{array}{l} \sum_{i=1}^N u_{ij}^2 \frac{\partial d(x_i, V_j)}{\partial V_j} = 0 \\ u_{rs} = \frac{1}{\sum_{j=1}^C \frac{d(x_i, V_s)}{d(x_i, V_j)}} \end{array} \right.$$

« Clustering »

Cas « Fuzzy » : $u_{ij} \in [0, 1]$

Système résolu par le schéma algorithmique itératif suivant :

1. Choisir au hasard C Noyaux V_j

2. $t \leftarrow 0$ /* Iteration*/

3. $t \leftarrow t+1$

4. **Répéter**

• **Pour chaque** valeur de i (de 1 à N),

Pour chaque valeur de j (de 1 à C), calculer $u_{ij}(t) = \frac{1}{\sum_{k=1}^C \frac{d(x_i, V_j)}{d(x_i, V_k)}}$

Fin pour

Fin pour

• $t = t+1$

• **Pour chaque** valeur de j (de 1 à C),

Résoudre $\sum_{i=1}^N u_{ij}^2(t-1) \frac{\partial d(x_i, V_j)}{\partial V_j} = 0$

Fin pour

Tant que un critère d'arrêt n'est pas atteint

« Clustering »

Cas « Fuzzy » : $u_{ij} \in [0, 1]$

Dans le cas classique où on utilise la distance classique

$$d(x_i, V_j) = (x_i - V_j)^T A (x_i - V_j)$$

On met à jour les prototypes par l'équation suivante :

$$V_j(t) = \frac{\sum_{i=1}^N u_{ij}^2(t-1) x_i}{\sum_{j=1}^N u_{ij}^2(t-1)}$$

Il s'agit de l'algorithme du Fuzzy C-Means classique

« Clustering »

Cas « Fuzzy » : $u_{ij} \in [0, 1]$

Ajoutons que pour $m=1$, il n'y a pas de clustering flou meilleur que le meilleur des « crisp clustering ».

Mais pour $m=2$ (le cas étudié), il y a des cas pour lesquels le clustering flou a de plus petites valeurs pour $J_2(U;V)$

« Clustering »



Algorithme des C-Moyennes Floues Exponentielles (CMFE)

⇒ Prototypes : centroïdes V_j et matrices de covariance floue F_j

$$F_j = \frac{\sum_{i=1}^N u_{ij}^2 (x_i - V_j)(x_i - V_j)^T}{\sum_{i=1}^N u_{ij}^2}$$

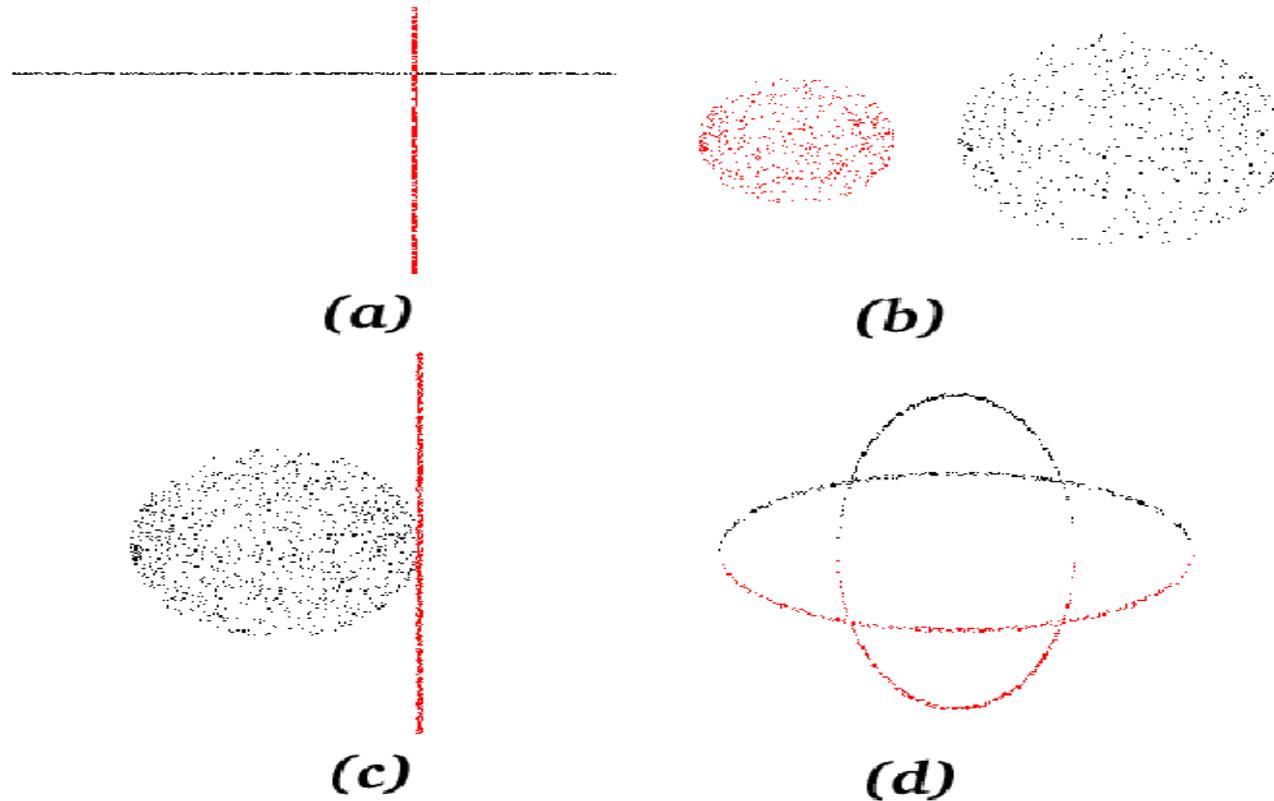
⇒ Distance : *exponentielle* d_e

$$d_e^2(x_i, v_j) = \frac{[\det(F_j)]^{1/2}}{P_j} \exp\left[\frac{1}{2} (x_i - v_j)^T F_j^{-1} (x_i - v_j)\right]$$

« Clustering »



Algorithme des C-Moyennes Floues Exponentielles (CMFE)

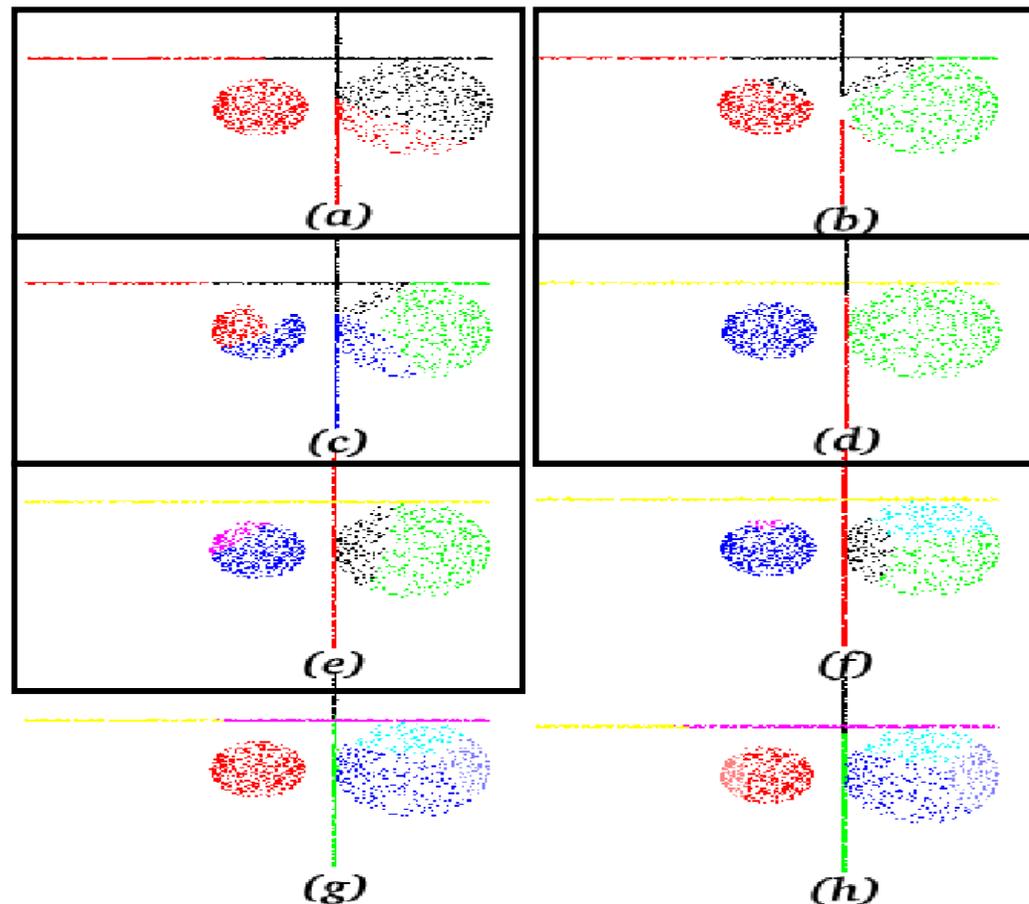


« Clustering »



Comment déterminer
le nombre de groupements C optimal ?

Critère numérique :
la Densité Moyenne
de Partition
(*DMP*) ?



« Clustering »



Critère numérique : la Densité Moyenne de Partition

$$DPM(C) = \frac{1}{C} \sum_{j=1}^C \frac{S_j}{V_j}$$

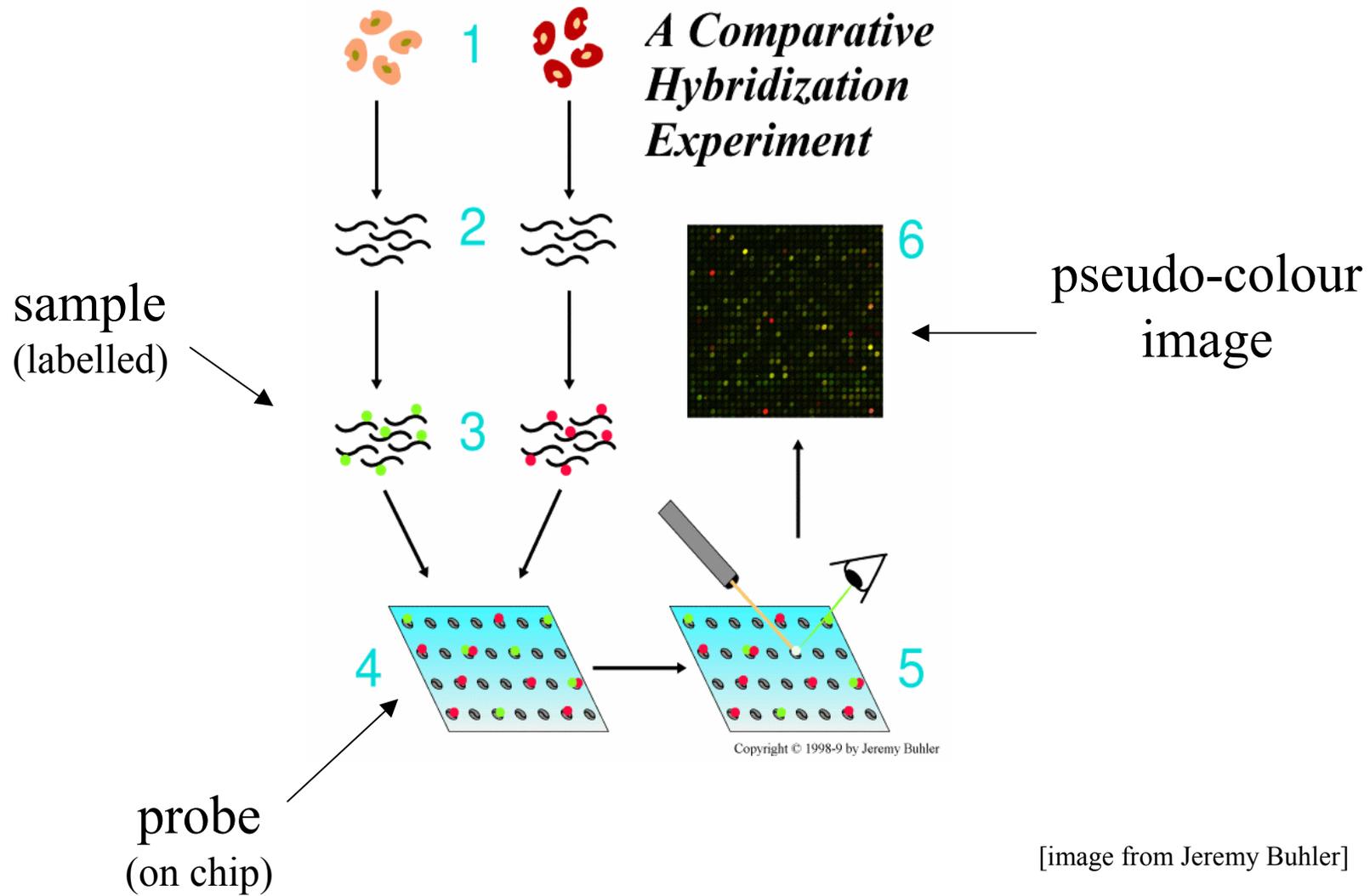
Avec $X_j = \{x \in X : (x - V_j) F_j^{-1}(x - V_j) < 1\}$

$$S_j = \sum_{x_i \in X_j} u_{ij}$$

et l'hypervolume flou de chaque cluster

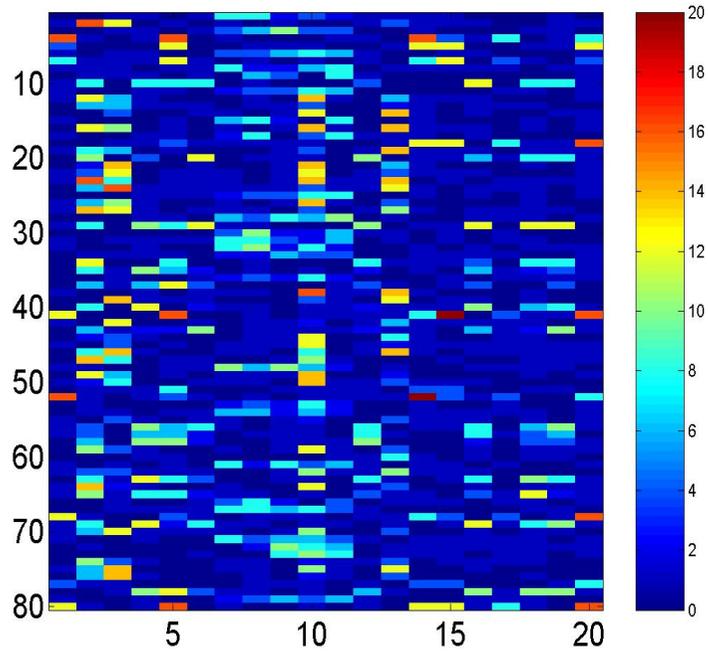
$$V_j = |F_j|^{1/2}$$

« Clustering »

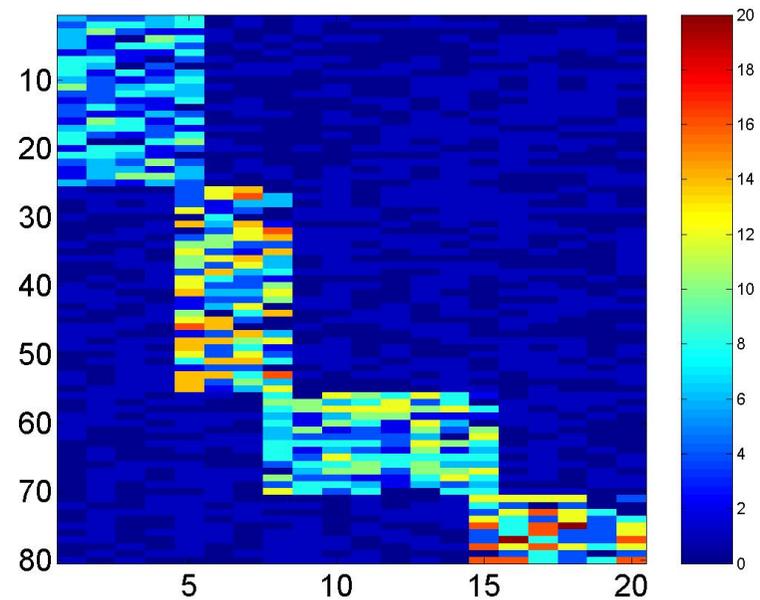
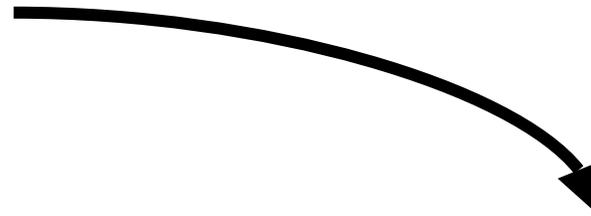


[image from Jeremy Buhler]

« Clustering »



Regroupement en
famille de gènes



Nuées Dynamiques ou K-Means

- Fonction de dissemblance f :

$$\forall x \in E, \forall \omega \in \wp(E) \Rightarrow f(x, \omega) \in \mathfrak{R}^+$$

- Critère à Minimiser Théorique :

$$J(\wp) = \sum_{\omega \in \wp} \sum_{x \in \omega} f(x, \omega)$$

- Critère à Minimiser Pratique à l'aide d'un noyau Y_ω par classe :

$$J(\wp) = \sum_{\omega \in \wp} \sum_{x \in \omega} f(x, Y_\omega)$$

Nuées Dynamiques ou K-Means

Algorithme :

1. Choisir au hasard K Noyaux $\{Y_1^0, Y_2^0, \dots, Y_K^0\}$
2. $n \leftarrow 0$
3. $n \leftarrow n+1$
4. **Pour chaque** valeur de k (de 1 à K), calculer
$$\omega_k^n \leftarrow \left\{ x \in E \mid \forall j \neq k, f(x, Y_k^{n-1}) \leq f(x, Y_j^{n-1}) \right\}$$

Calcul de Y_k^n à partir de ω_k^n

Fin pour

5. **Si** $\omega_k^n \neq \omega_k^{n+1}$ et $n \leq n_0$, retourner en 3

Sinon arrêt

Fin Si

Nuées Dynamiques ou K-Means

Exemples :

- $Y_\omega = \text{Centre de Gravité de la classe } \omega = \frac{1}{\text{card}(\omega)} \sum_{x \in \omega} x$
- $f(x, Y) = d(x, Y_\omega)$

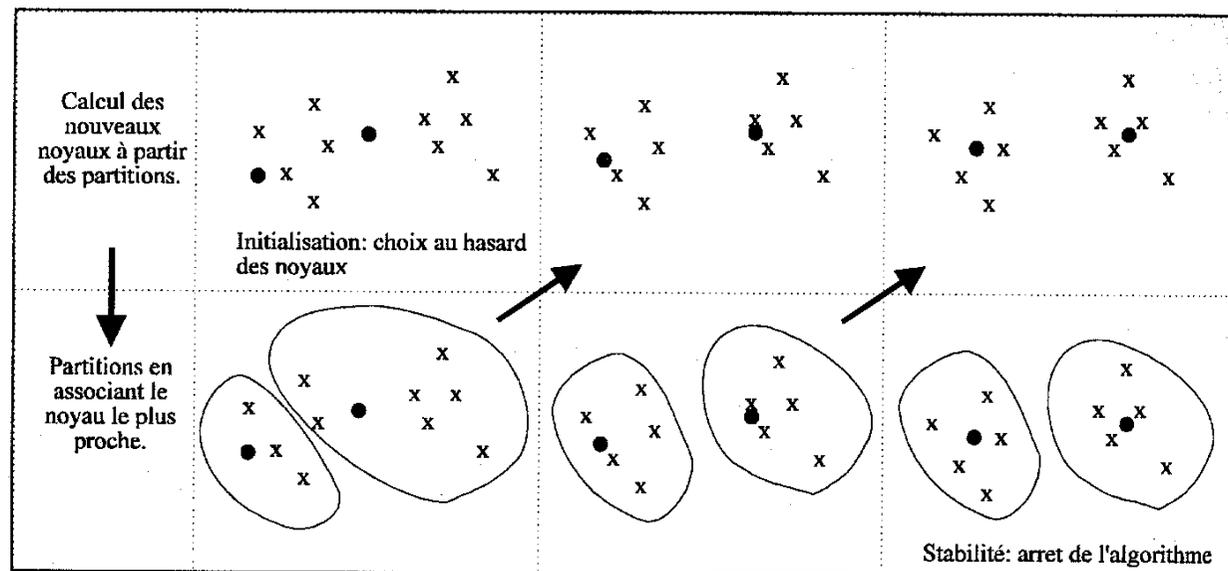


Figure 10.8 : Construction de 2 classes par l'algorithme des nuées dynamiques

Introduction

Codage

Analyse

Apprentissage & Décision

Classification Automatique

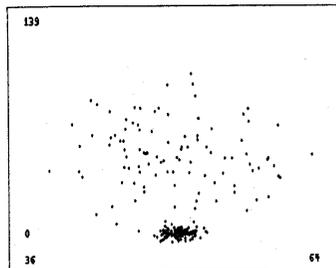
Nuées Dynamiques ou K-Means

Amélioration :

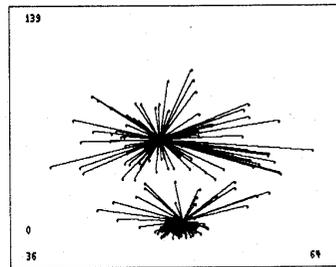
- Prendre en Compte la forme des Classes en Création :
 - ➔ Distance à l'axe principal d'inertie par exemple
- Gérer le recouvrement des Classes
 - ➔ Attribution floue par exemple

Nuées Dynamiques ou K-Means

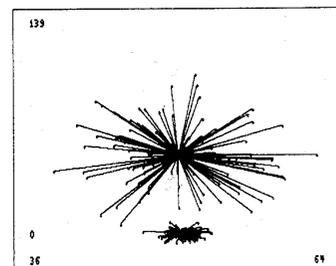
Exemple du Fuzzy K-Means :



(a)



(b)



(c)

Apprentissage et Décision

Apprentissage et Décision

Objectifs

1. Classification automatique
2. Programmation Dynamique
3. Discrimination Fonctionnelle
4. Connexionisme
5. Statistiques bayésiennes
6. K-ppv
7. Méthodes stochastiques
8. Approches syntaxiques et structurelles

Apprentissage (automatique ou artificiel) :

Machine Learning

Cette notion englobe toute méthode permettant de construire un modèle de la réalité à partir de données.

Il existe deux tendances principales :

- *Celle issue de l'IA qualifiée de symbolique*
- *Celle issue des statistiques qualifiée de numérique*

Fouille de données (Extraction de connaissances à partir des données) :

Data Mining (Knowledge discovery in data)

La fouille de données prend en charge le processus complet d'extraction de connaissances : stockage dans une BD, sélection des données à étudier, nettoyage de ces données, puis utilisation des apprentissages symboliques et numériques afin de proposer des modèles à l'utilisateur, enfin validation des modèles proposés.

Précision vs. Généralisation

Le grand dilemme de l'apprentissage.

La précision est définie par un écart entre une valeur mesurée ou prédite et une valeur réelle. Apprendre avec trop de précision conduit à un “sur-apprentissage”, comme l'apprentissage par coeur, pour lequel des détails insignifiants (dûs au bruit) sont appris.

Apprendre avec trop peu de précision, conduit à une surgénéralisation telle que le modèle s'applique même quand l'utilisateur ne le désire pas.

Il existe des mesures de généralisation, et l'utilisateur peut fixer le seuil de généralisation qu'IL juge optimal.

Classification :

La classification (voire Analyse de données) consiste à regrouper des ensembles d'exemples non supervisés en classes.

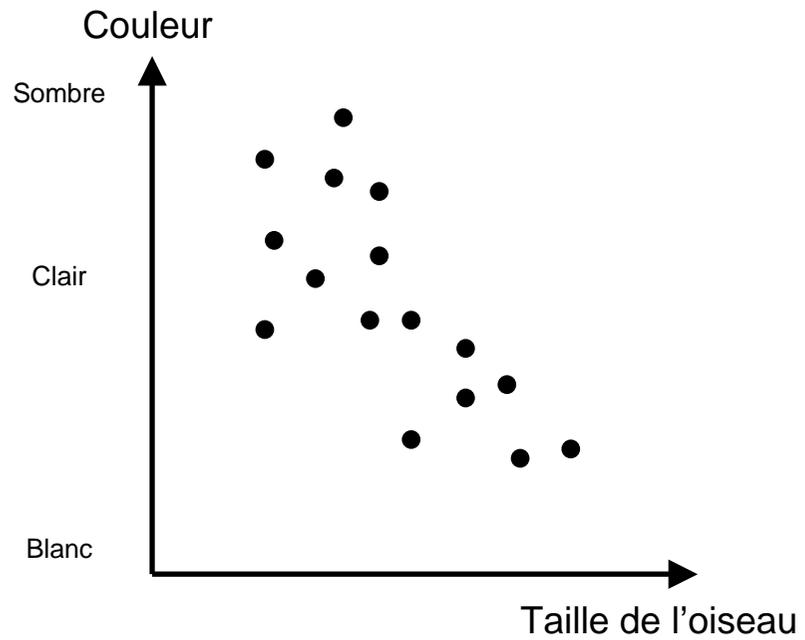
Ces classes sont souvent organisées en une structure (clustering). Si cette structure est un arbre, alors on parle de taxonomie ou de taxinomie (taxonomy).

Il s'agit par exemple de prévoir l'appartenance d'un oiseau observé à la classe "canard" ou "oie".

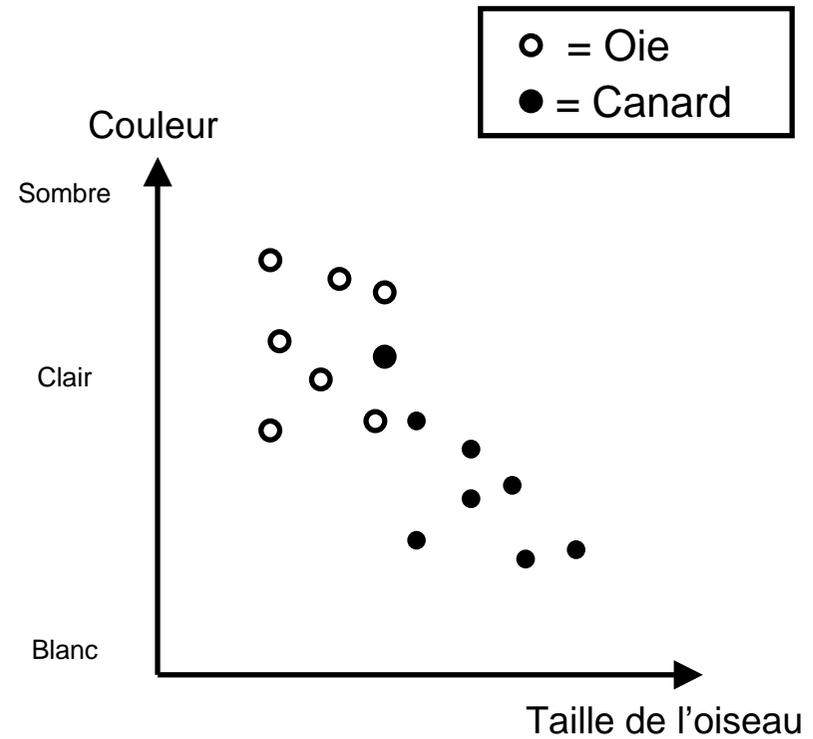
Deux champs industriels de l'Apprentissage :

1. La Reconnaissance des Formes (image, parole, signaux bio-médicaux)
2. La Fouille de Données

Exemples d'apprentissage :



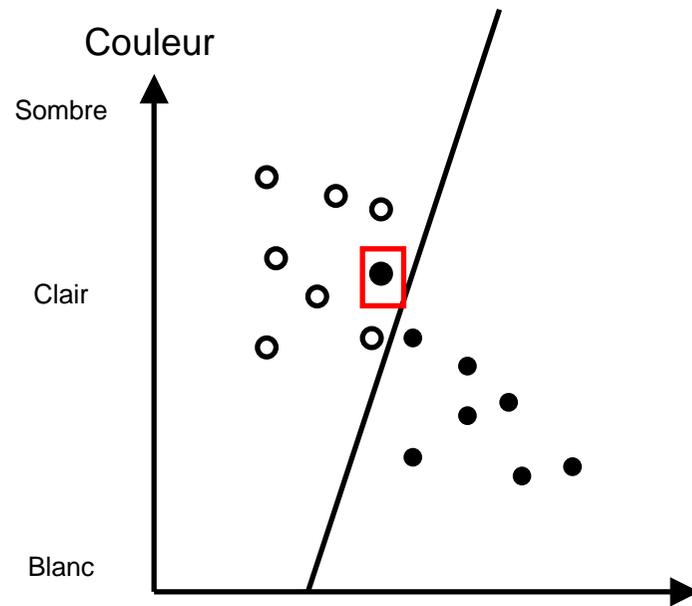
Oiseaux Observés par un débutant



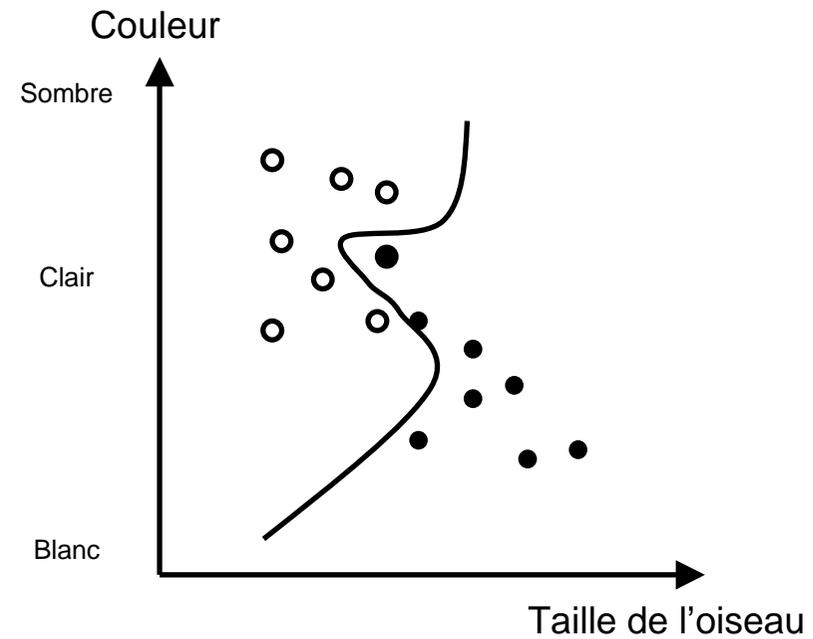
Données Etiquetées par un expert (superviseur)

Exemples d'apprentissage :

○ = Oie
● = Canard



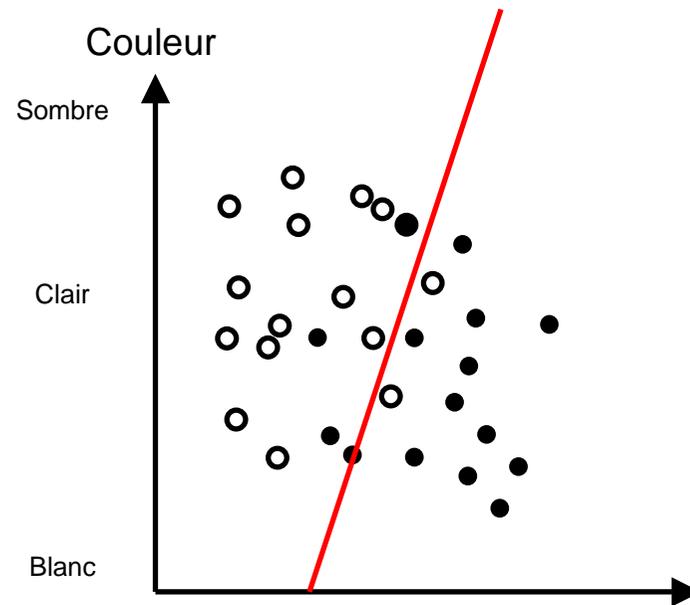
Apprentissage d'une règle de décision simple



Apprentissage d'une règle de décision complexe

Exemples d'apprentissage :

○ = Oie
● = Canard



Test en Décision sur d'autres
oiseaux à partir de la règle de
décision simple

Introduction

Codage

Analyse

Apprentissage & Décision

*Discrimination
Fonctionnelle*

Principe :

définir les **fonctions de discrimination f** permettant de séparer **partiellement ou totalement** les classes représentées par les vecteurs paramètres x de leurs échantillons. L'ensemble des x exemples représente l'**ensemble d'apprentissage S** .

Introduction

Codage

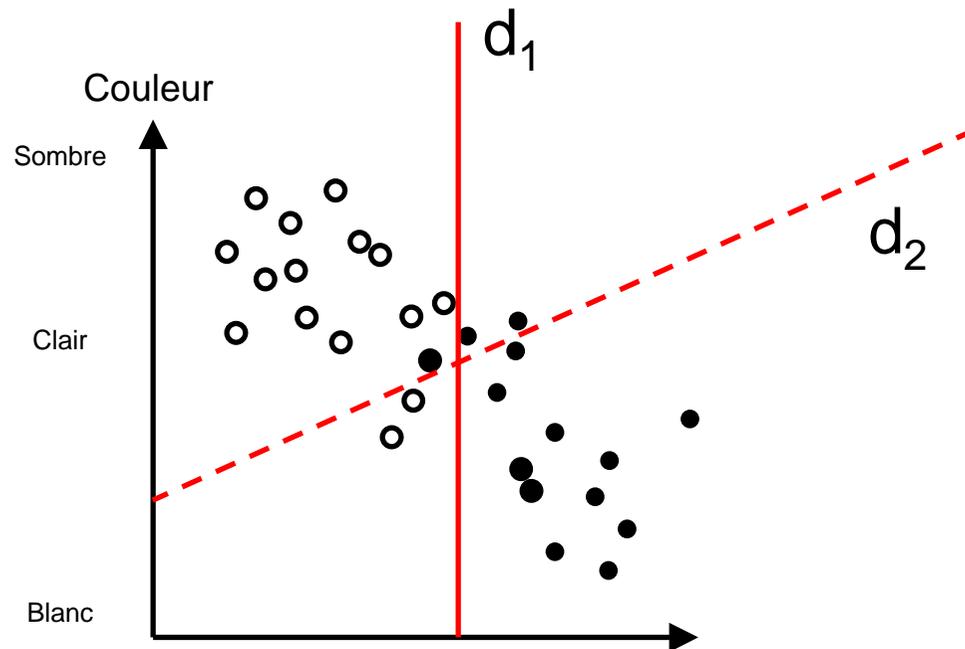
Analyse

Apprentissage & Décision

*Discrimination
Fonctionnelle
Linéaire*

Objectifs :

- Il s'agit d'apprendre un concept sous la forme géométrique la plus simple : celle d'un hyperplan.
- Apprentissage de surfaces séparatrices linéaires dans un espace de représentation nécessairement numérique.
- Lien avec les Réseaux de Neurones et les SVM



- (d_1) minimise le nombre d'erreurs dans l'ensemble d'apprentissage
- (d_2) minimise le risque bayésien d'erreurs par rapport à l'ensemble d'apprentissage

Hypothèses :

Les classes sont linéairement séparables.

Remarque : hypothèse pas plus injustifiée que la classique hypothèse statistique *a priori* gaussienne

Dans \mathfrak{R}^n , une surface linéaire est un hyperplan f :

$$f(x) = \omega_0 + \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_n x_n = W^t \cdot X = 0$$

$$\text{avec } X = (1, x_1, x_2, \dots, x_n)^t = (1, x)^t$$

$$\text{et } W = (w_0, w_1, w_2, \dots, w_n)^t = (w_0, w)^t$$

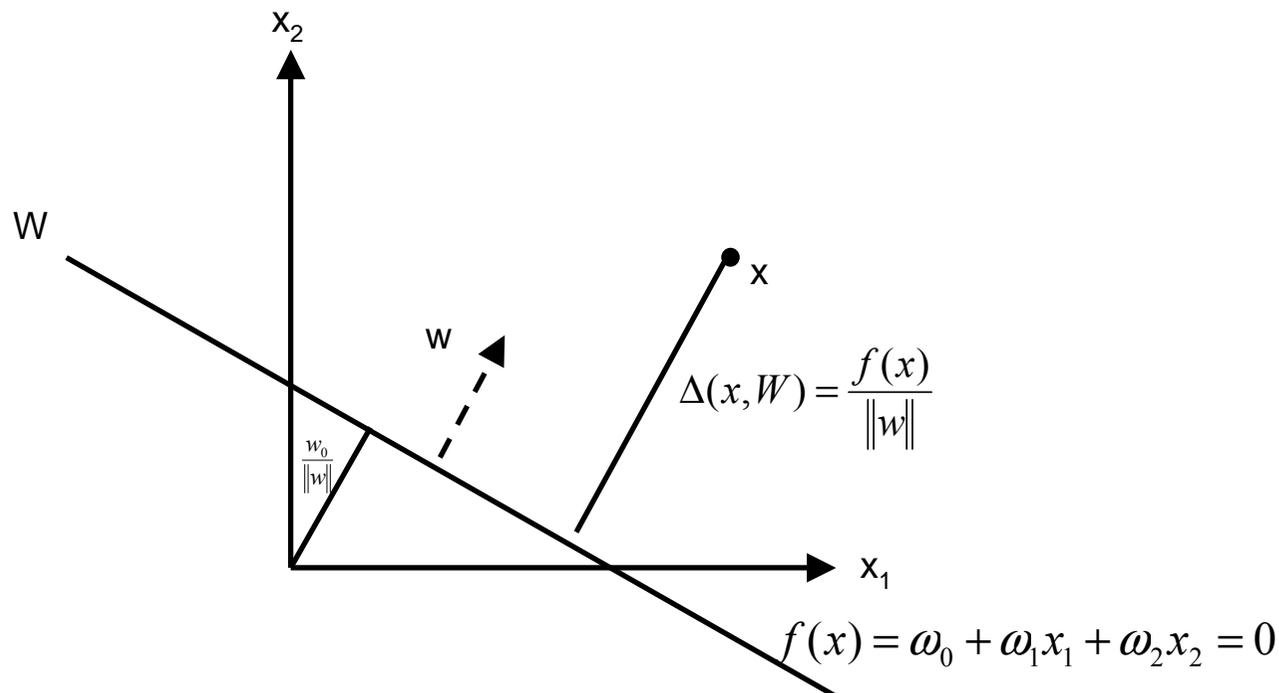
Dans \mathfrak{R}^2 , une droite est définie par :

$$f(x) = \omega_0 + \omega_1 x_1 + \omega_2 x_2 = 0$$

que l'on va écrire avec $x = (x_1, x_2)^t$

$$f(x) = \omega_0 + w^t x = 0 \quad \text{et} \quad w = (w_1, w_2)^t$$

Remarque : Δ est une distance signée



Introduction

Codage

Analyse

Apprentissage & Décision

DFL

Apprentissage :

- Déterminer les coefficients W de ces fonctions discriminantes à partir de formes connues

- Algorithme du **Perceptron** : le plus ancien et le plus simple

Cas de 2 classes :

$$f(X) = W^t \cdot X \begin{cases} > 0 & \text{si } X \in C_1 \\ < 0 & \text{si } X \in C_2 \end{cases}$$

On pose
$$\begin{cases} \forall X \in C_1, & Y = X \\ \forall X \in C_2, & Y = -X \end{cases}$$

Alors

$$W^t \cdot Y > 0 \Leftrightarrow \begin{cases} W^t \cdot X > 0 & \text{si } X \in C_1 \\ W^t \cdot X < 0 & \text{si } X \in C_2 \end{cases}$$

Cas de 2 classes :

Si on range maintenant les m vecteurs X comme colonnes d'une matrice M , le problème de séparation linéaire revient à la recherche d'un vecteur W dans \mathfrak{R}^{n+1} , tel que : $W^t M = B^t$

où B est un vecteur positif inconnu de \mathfrak{R}^{m+1}

Comme en général M n'est pas inversible et que les données ne sont pas réellement complètement linéairement séparables, il faut trouver l'hyperplan W le meilleur possible selon par exemple le critère :

$$J(W, B) = \frac{1}{2} \|W^t M - B^t\|^2$$

Cas de 2 classes :

Or, si W et B minimisent bien $J(W,B)$: $J(W,B) = \sum_{x \in S \text{ mal classés par } W} \Delta(x,W)^2$

Ce qui revient à se placer dans le cadre du critère aux moindres carrés.

Donc dans l'hypothèse que les distances entre W et les points mal classés par W sont répartis selon une distribution gaussienne, et que ceux-ci sont les seuls à compter dans le positionnement de W , la minimisation de $J(W,B)$ est la recherche du meilleur hyperplan *a posteriori* au sens bayésien, cad l'eplus probable connaissant les données.

Cas de 2 classes :

$$J(W, B) = \frac{1}{2} \|W^t M - B^t\|^2 \longrightarrow \nabla_A J(W, B) = (W^t M - B^t) M^t$$

qui atteint son minimum pour $(W^t M - B^t) = 0$

$$\text{soit } W^t = B^t M^+$$

où M^+ est la pseudo-inverse de $M = M^T(MM^T)^{-1}$ (user de la SVD si nécessaire)

Comme on ne connaît pas B , l'algorithme doit réaliser une minimisation de $J(W, B)$ sous la contrainte B positif ou nul

Cas de 2 classes :Méthode globale basée sur une descente de gradient :

Algorithme de Ho et Kashyap

INPUT : B_0 et α positifs quelconques

- $t \leftarrow 0$
- Tant que *critère d'arrêt non satisfait* faire
 - $W_{(t)}^T \leftarrow B_{(t)}^T M^+$
 - $B_{(t+1)}^T \leftarrow B_{(t)}^T + \alpha \lfloor W_{(t)}^T M - B_{(t)}^T \rfloor$
 - $t \leftarrow t+1$
- Fin tant que

OUTPUT : l'hyperplan optimal W

$W_{(t)}^T = B_{(t)}^T M^+$ puis trouver $B_{(t+1)}$ tel que $J(W(t), B(t+1)) \leq J(W(t), B(t))$

or $\nabla_{B_{(t)}} J(W_{(t)}, B_{(t)}) = -2(W_{(t)}^T M - B_{(t)}^T)$ d'où $B_{(t+1)}^T = B_{(t)}^T + \alpha(W_{(t)}^T M - B_{(t)}^T)$

or $B \geq 0$, d'où $\lfloor W_{(t)}^T M - B_{(t)}^T \rfloor = 0$ si $W_{(t)}^T M - B_{(t)}^T < 0$. De plus on peut avoir $\alpha_{(t)}$

Cas de 2 classes :

DFL

Méthode itérative historique :

Algorithme du Perceptron

INPUT : $w_{(0)}$ et α positifs quelconques

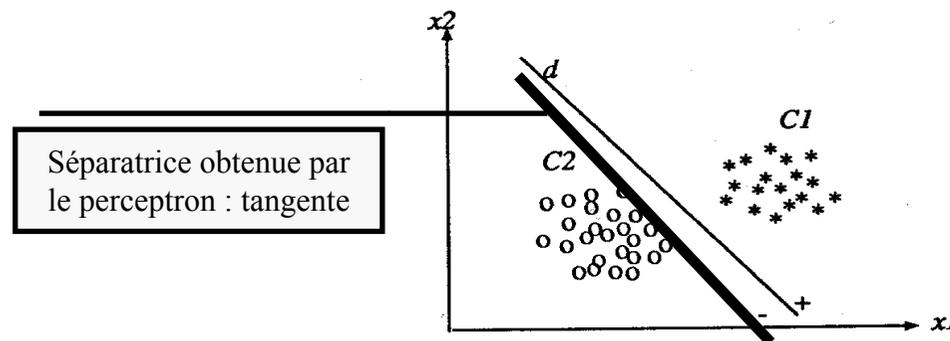
- $t \leftarrow 0$
- Tant que $t \leq t_{max}$ faire
 - $modif \leftarrow 0$
 - Pour chaque donnée d'apprentissage x faire
 - Si x est mal classé alors
 - Si $x \in C_1$ alors
 - $modif \leftarrow modif + \alpha x$
 - Sinon
 - $modif \leftarrow modif - \alpha x$
 - Fin si
 - Fin si
 - Fin pour
 - $w_{(t+1)} \leftarrow w_{(t)} + modif$
 - $t \leftarrow t + 1$
- Fin tant que

une époque d'apprentissage :
cumul des contributions de
chaque exemple de S

OUTPUT : l'hyperplan optimal W

- Convergence, mais lente et loin de l'optimum

$$W_{t+1} Y_{t+1}^T = W_t Y_{t+1}^T + \|Y_{t+1}\|^2$$

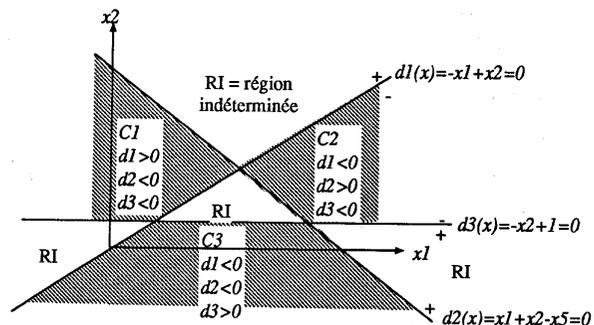
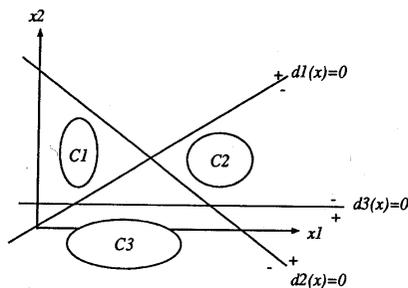


- Minimisation de $J(W) = - \sum_{x \in S} W^T x$ mal classés par W

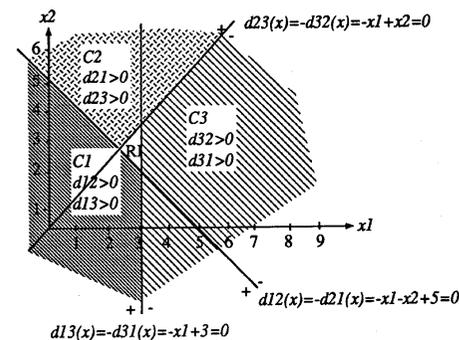
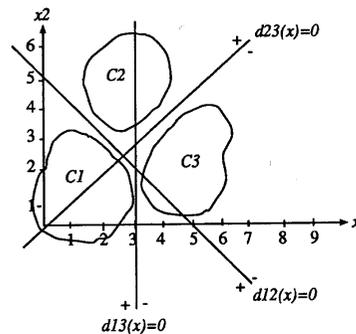
- La base S peut être parcourue plusieurs fois
- Version stochastique mais meilleure généralisation de Ho et Kashyap

Cas général : M classes C_1, C_2, \dots, C_M

Chaque classe est séparée des autres classes par un hyperplan : au total M hyperplans



Les classes sont séparées deux à deux : au total $M(M-1)/2$ hyperplans



Introduction

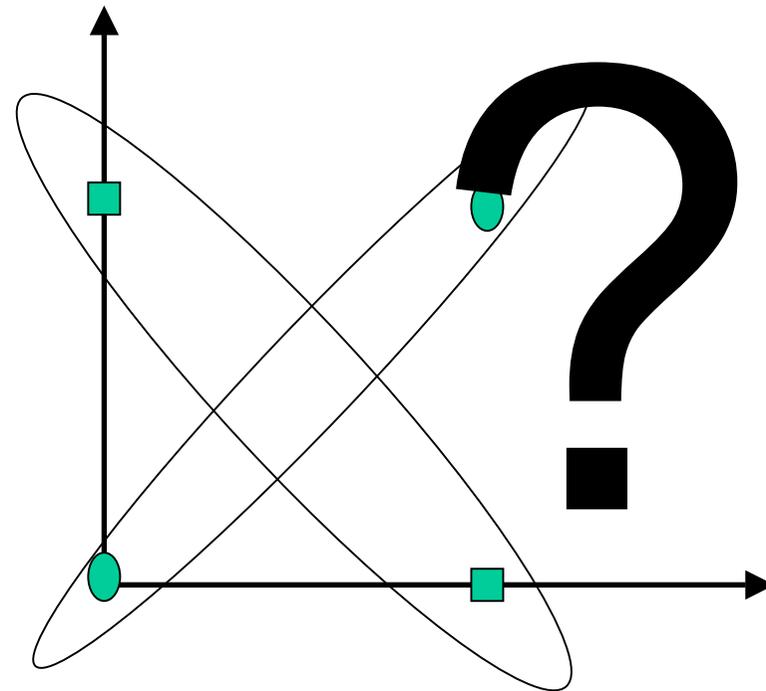
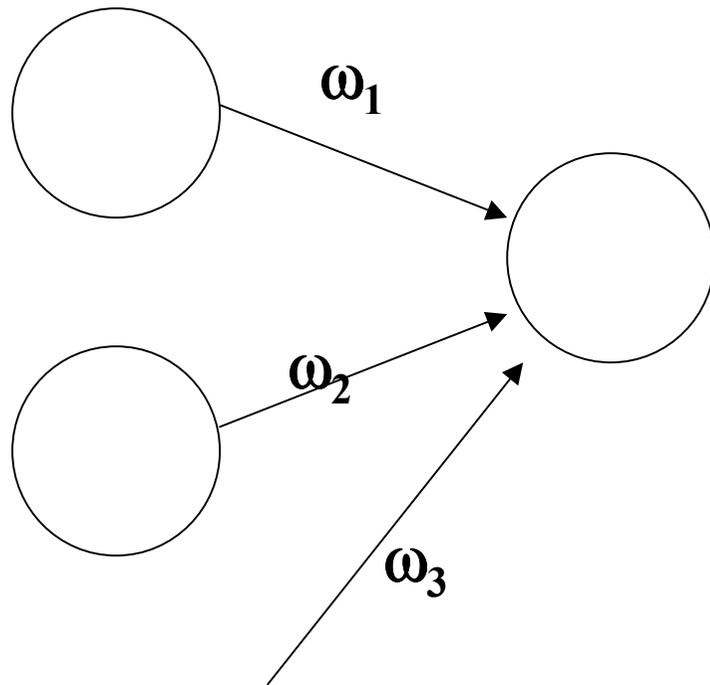
Codage

Analyse

Apprentissage & Décision

DFL

Lien avec les réseaux de Neurones



Echec Sur le Problème du XOR

- Neurones Formels et Réseaux de Neurones :

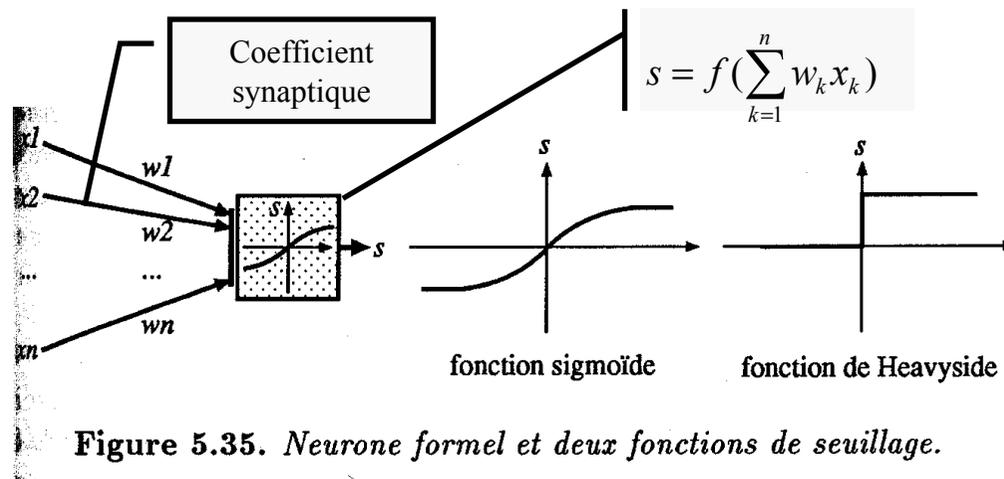
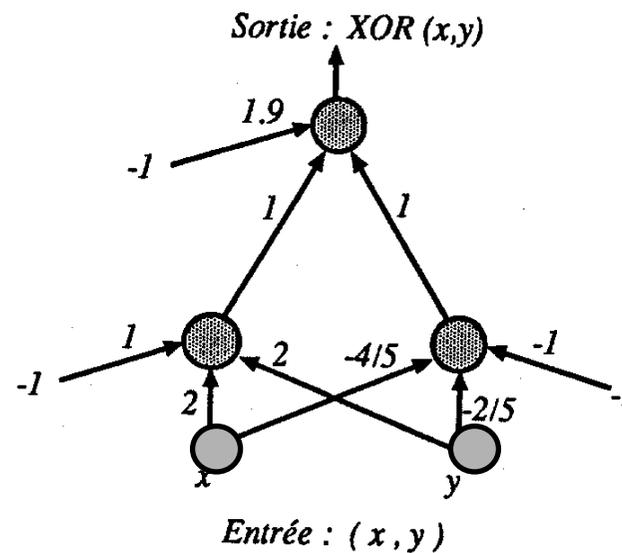
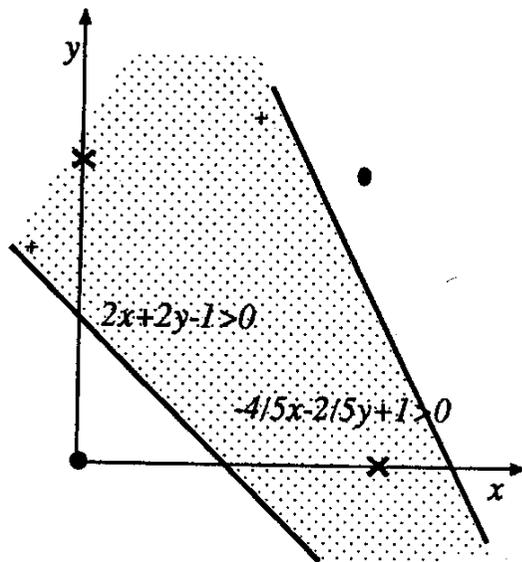


Figure 5.35. Neurone formel et deux fonctions de seuillage.

- Un Neurone Formel effectue une discrimination linéaire
- Un réseau de Neurones effectue **une discrimination linéaire par morceaux**

Succès sur le Problème du XOR avec une couche cachée :



Introduction

Codage

Analyse

Apprentissage & Décision

Connexionnisme

Divers Points de Vue sur les Réseaux de Neurones :

- Point de Vue Informatique :

Parallélisme à grains très fins;

- Point de Vue de la Classification :

Discriminateur Linéaire Complexe;

Introduction

Codage

Analyse

Apprentissage & Décision

Connexionnisme

Avantages Principaux :

- Inclus dans son traitement Analyse et Prétraitement
- Grande Capacité Autonome d 'Apprentissage

➡ *Algorithme de Rétropropagation*

- Problème de la Généralisation
- Extension non linéaire et bien fondée théoriquement
donnée par les SVM : Séparateurs à Vastes Marges ou
Support Vector Machines -> *kernel-based machine learning*

Introduction

Codage

Analyse

Apprentissage & Décision

Analyse bayésienne

- Modèle Probabiliste de Représentation des Formes ?
- Assure en théorie le Minimum d'Erreur en Classification

Propriété :

La règle de décision bayésienne est la règle de décision optimale au sens où elle minimise le risque réel en utilisant l'information disponible de manière optimale.

Etant donné que l'on m'a fourni un échantillon de données, comment cela doit-il modifier mes croyances antérieures sur le monde ?

Règle de Bayes de révision des probabilités :

- x : représentation de la forme, ω : classe

$$P(x / \omega) = \frac{P(x, \omega)}{p(\omega)}$$

$$p(\omega / x) = \frac{P(x, \omega)}{P(x)}$$

Théorème de Bayes

mesurable estimable

$p(\omega / x) = \frac{P(x / \omega) p(\omega)}{P(x)}$

Difficile à calculer

Probabilité *a posteriori*

Probabilités *a priori*

Cas de 2 classes :

$\omega_1 = oiseau$ et $\omega_2 = canard$ après observation de la variable $x = couleur_aile$

$$p(oiseau = canard / couleur_aile = noire) = \frac{P(couleur_aile = noire / oiseau = canard) p(oiseau = canard)}{P(couleur_aile = noire)}$$

Risque bayésien :

Le but de l'agent est de prendre une décision (*faire un diagnostic vital par exemple*) **minimisant l'espérance de risque**.

On définit une fonction de décision $S : X \rightarrow H$, où H est vu comme un ensemble de décisions à prendre.

Dans le cas de N classes, H peut comprendre les N classes possibles Ω en décision de classification plus une classe de **rejet** quand l'incertitude est trop forte et ne permet pas de prendre de décision.

Avant toute observation sur le monde, et en prenant seulement en compte les connaissances *a priori*, l'espérance de risque associée à une décision h peut s'écrire :
$$R(h) = \sum_{\omega \in \Omega} l(h|\omega) p(\omega)$$

où $p(\omega)$ dénote la probabilité *a priori* que le monde soit dans l'état ω , tandis que $l(h|\omega)$ est le **coût ou perte** encouru lorsque la décision h est prise alors que l'état du monde est ω .

Introduction

Codage

Analyse

Apprentissage & Décision

Analyse bayésienne

En général,

$$l(h|\omega) = \begin{cases} 0 & \text{si } h = \omega \text{ (décision correcte)} \\ 1 & \text{si } h \neq \omega \text{ (décision incorrecte)} \\ r & \text{si } h = \text{rejet (doute trop important)} \end{cases}$$

Mais , le coût de ne pas diagnostiquer à tort une tumeur est bien plus élevé que de faire un faux diagnostic. Le coût de la décision incorrecte est ajustable.

Principe de la Décision Bayésienne :

⇒ Choisir pour l'observation x la classe ω qui minimise l'espérance de risque.

$$h^* = \underset{h \in H}{\text{ArgMin}} \sum_{\omega \in \Omega} l(h|\omega) p(\omega) P(x|\omega)$$

Principe de la Décision Bayésienne : cas particuliers

Règle du Maximum A Posteriori (MAP) :

Lorsque les coûts de mauvaise classification sont égaux, la règle de décision bayésienne de risque minimal devient la règle du MAP :

$$h^* = \underset{h \in \Omega}{\text{ArgMax}} p(\omega)P(x|\omega)$$

Remarque : cette règle minimise le **nombre d'erreurs** en classification

Principe de la Décision Bayésienne : cas particuliers

Règle du Maximum de Vraisemblance

Maximum Likelihood (ML) :

Si de plus toutes les hypothèses ont la même probabilité a priori, alors la règle du MAP devient la règle du Maximum de Vraisemblance :

$$h^* = \underset{h \in \Omega}{\text{ArgMax}} P(x|\omega)$$

Introduction

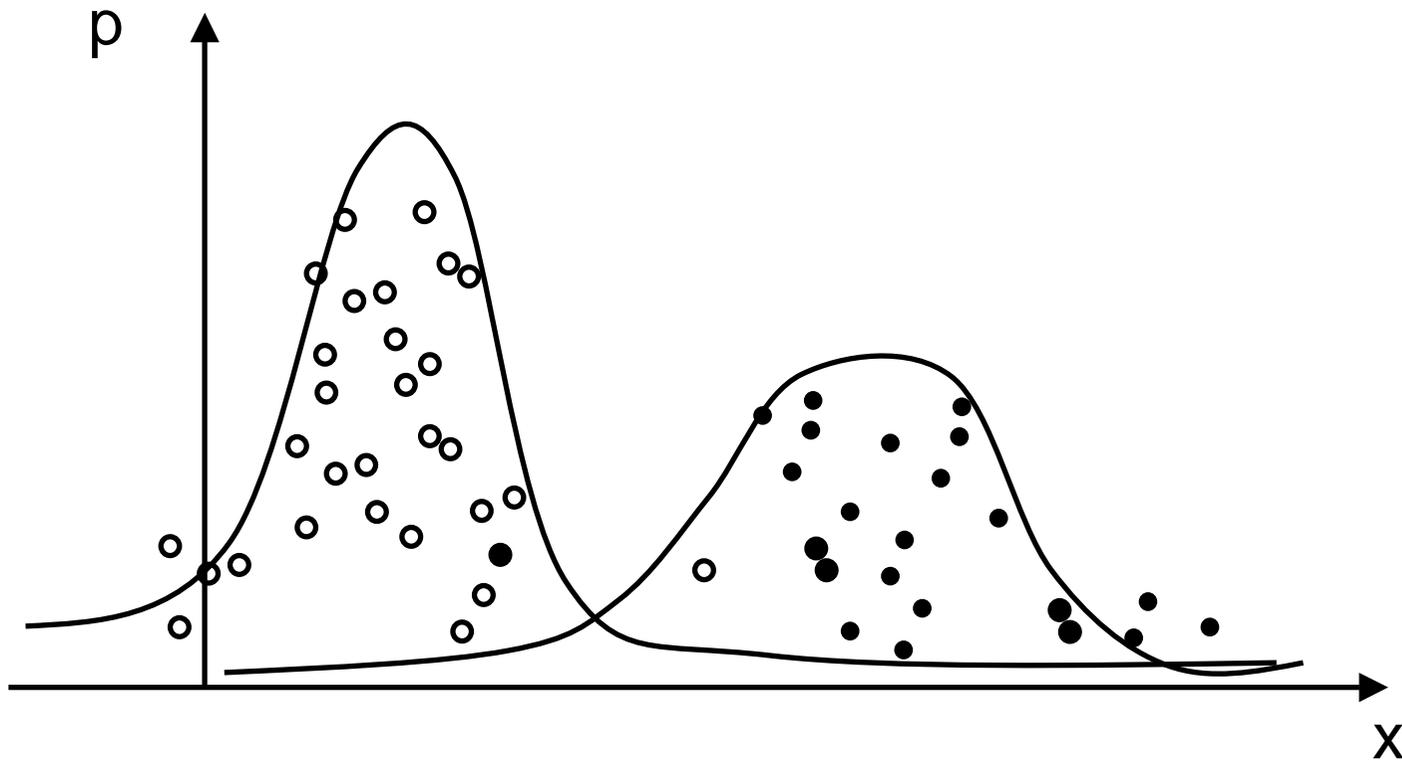
Codage

Analyse

Apprentissage & Décision

Analyse bayésienne

Principe de la Décision Bayésienne :
cas particuliers



Principe de la Décision Bayésienne : cas particuliers

Nous supposons maintenant que la tâche d'apprentissage consiste à discriminer les formes observées x en 2 classes : $H = \Omega = \{\omega_1, \omega_2\}$
Etant donnée l'observation x , les espérances de risque associés à chaque décision sont respectivement en notant $l(\omega_i|\omega_j)=l_{ij}$:

$$R(\omega_1)=l_{11} p(\omega_1 |x) + l_{12} p(\omega_2 |x)$$

$$R(\omega_2)=l_{21} p(\omega_1 |x) + l_{22} p(\omega_2 |x)$$

La règle de Bayes stipule de choisir l'hypothèse d'espérance de risque minimal. Il faut donc attribuer la forme x à la classe ω_1 ssi :

$$(l_{21}-l_{11}) p(\omega_1 |x) \geq (l_{12}-l_{22}) p(\omega_2 |x),$$

Que l'on peut écrire en appliquant la formule de Bayes de révision des probabilités :

$$(l_{21}-l_{11}) p(x|\omega_1) p(\omega_1) \geq (l_{12}-l_{22}) p(x|\omega_2) p(\omega_2), \text{ soit :}$$

$$d(x) = \log \frac{p(x|\omega_1)}{p(x|\omega_2)} + \log \frac{(l_{21}-l_{11})p(\omega_1)}{(l_{12}-l_{22})p(\omega_2)} \geq 0$$

Principe de la Décision Bayésienne : cas particuliers

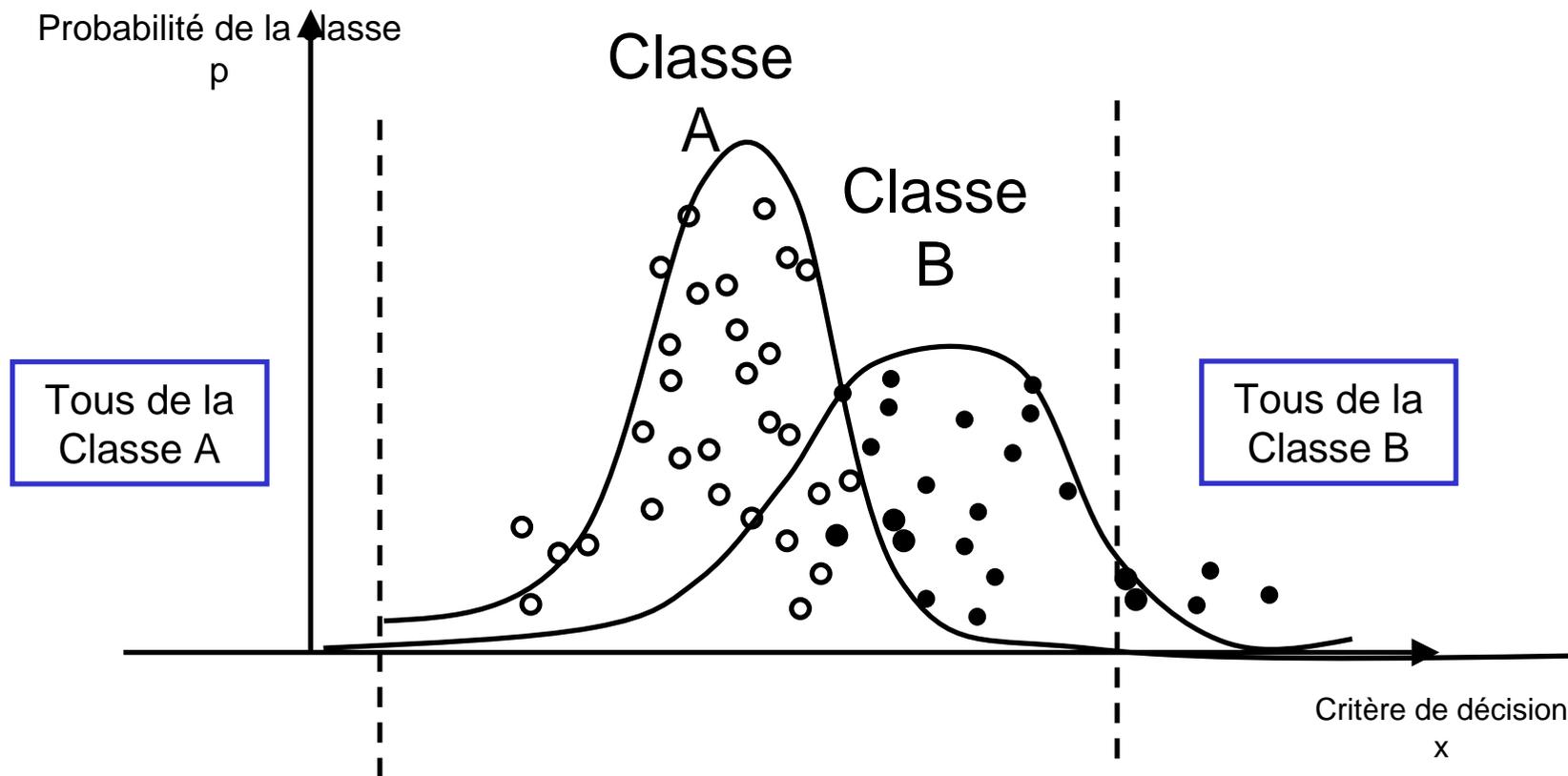
La règle de décision bayésienne se traduit ainsi par une fonction de discrimination ou de décision d décrivant une frontière ou surface de décision dans l'espace X .

On peut donc essayer d'apprendre directement cette frontière de décision plutôt que les probabilités, plus difficiles à estimer -> voire les méthodes de classifications fonctionnelles et le connexionnisme.

Dans le cas particulier de la discrimination entre deux classes, de distribution normale gaussienne de moyennes μ_1 et μ_2 avec des matrices de covariances égales Σ , la fonction de décision $d(x)$ est une fonction linéaire :

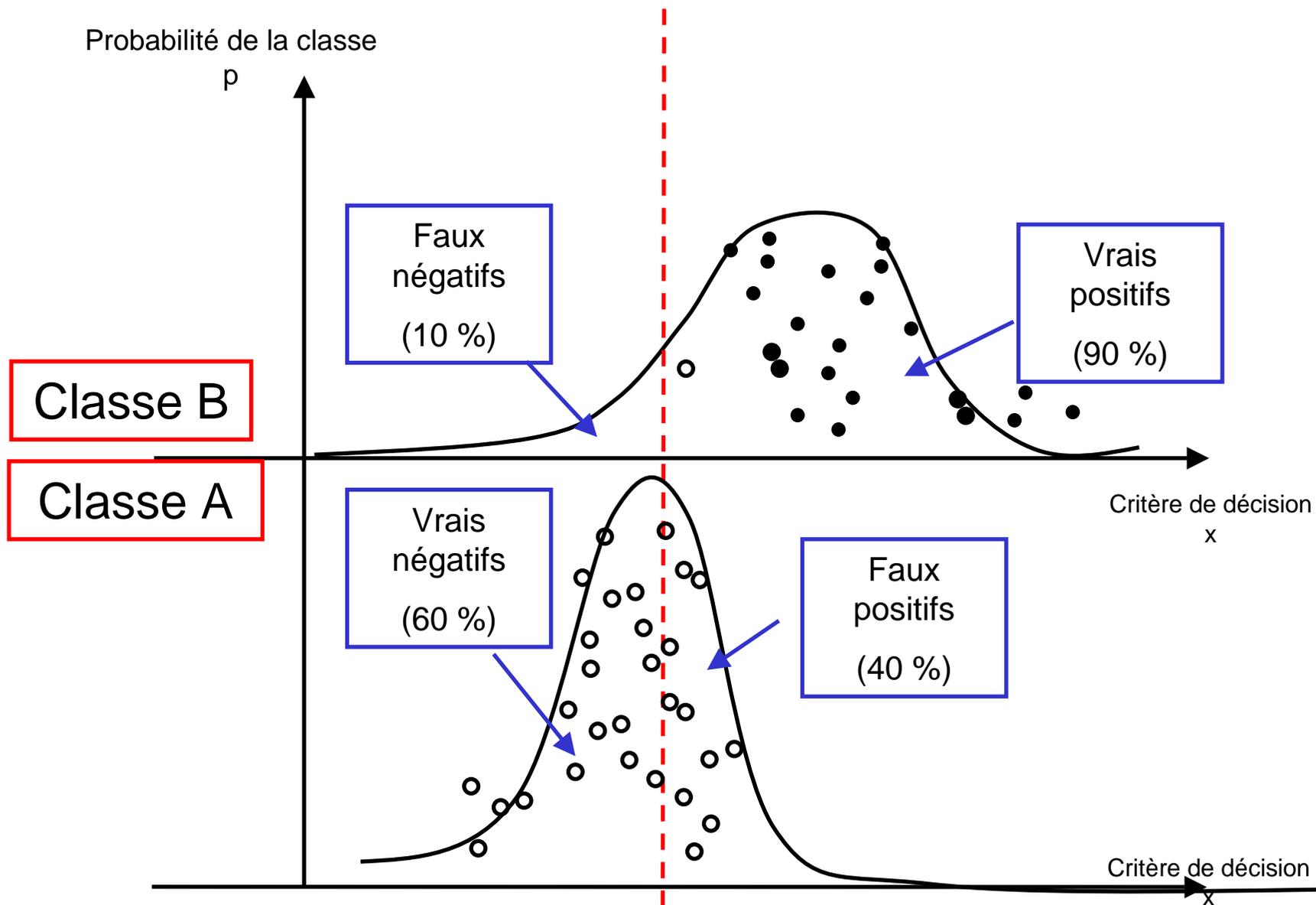
$$d(x) = \left(x - \frac{1}{2(\mu_1 - \mu_2)}\right)^T \Sigma^{-1} (\mu_1 - \mu_2) + \log \frac{(l_{21} - l_{11})p(\omega_1)}{(l_{12} - l_{22})p(\omega_2)}$$

Validation de l'apprentissage

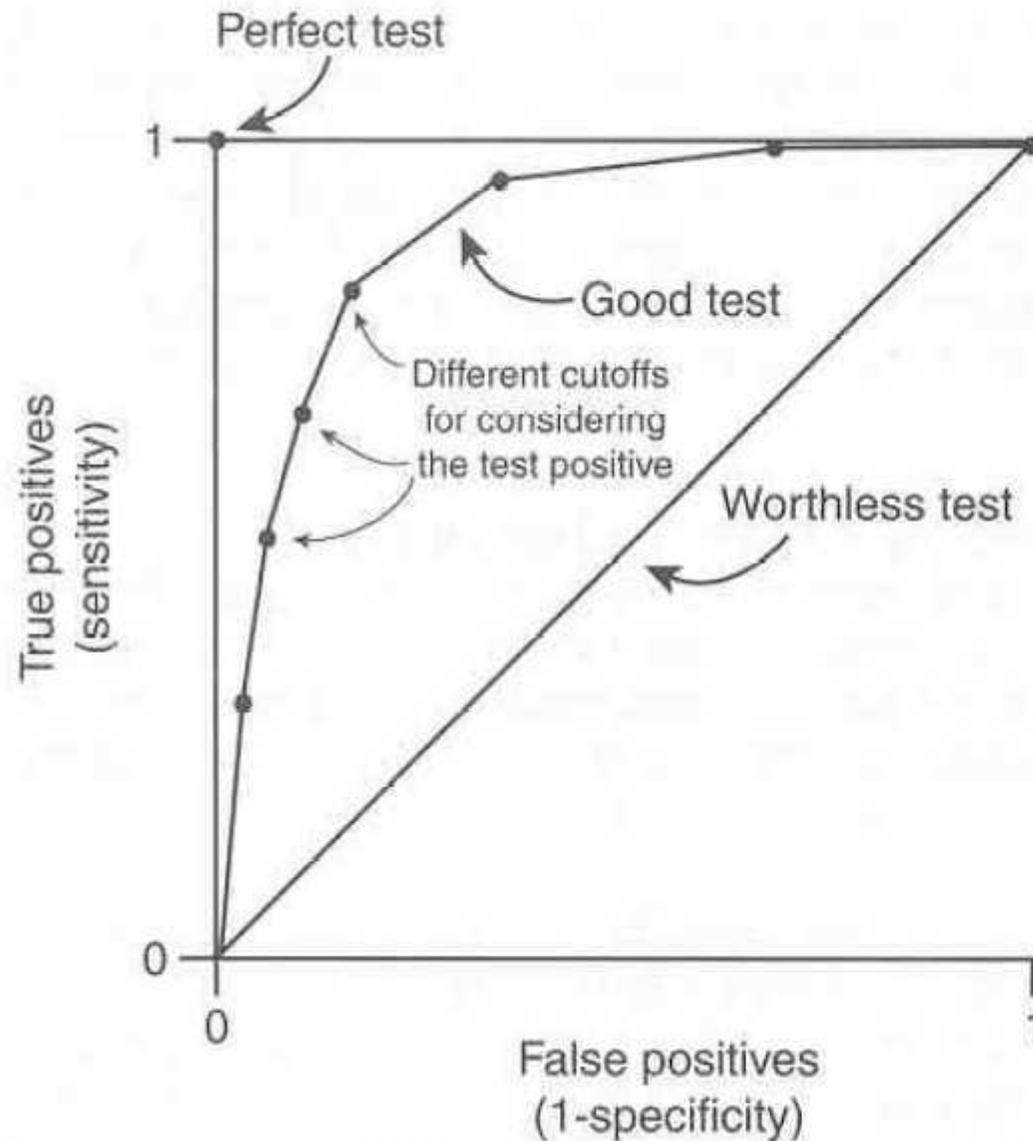


Validation de l'apprentissage

Analyse bayésienne



Validation de l'apprentissage

Analyse bayésienne

**Construction et
signification de
la courbe ROC**

Apprentissage :

- $X_n = \{x_1, x_2, \dots, x_n\}$: échantillons d'une même classe
- $f(x/\omega) = f(x, \Theta)$ où $\Theta = \{\Theta_1, \Theta_2, \dots, \Theta_m\}$ paramètres

- Choisir Θ Tel que

$$f(X_n, \Theta) = \prod_{k=1}^n f(x_k, \Theta) = \sum_{k=1}^n \log(f(x_k, \Theta)) \text{ Maximum}$$

- f est supposée Gaussienne $\rightarrow \Theta = \{\mu, \sigma\}$ et $\left\{ \begin{array}{l} \bar{\mu} = \frac{1}{n} \sum_{k=1}^n x_k \\ \bar{\sigma} = \frac{1}{n} \sum_{k=1}^n (x_k - \bar{\mu})^2 \end{array} \right.$

Introduction

Codage

Analyse

Apprentissage & Décision

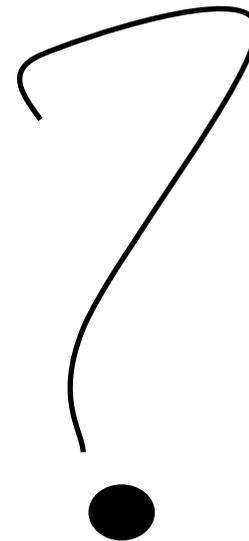
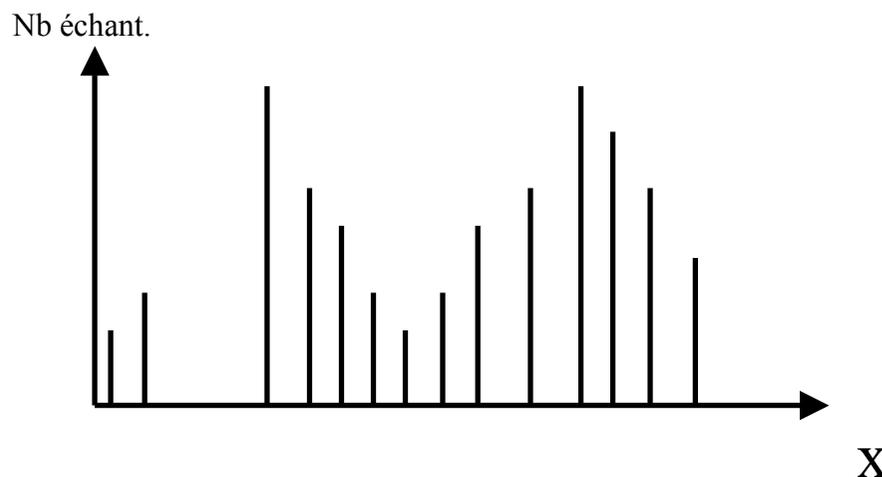
*Statistiques
bayésiennes*

Inconvénients :

⇒ Difficulté d 'Estimation

⇒ Hypothèse Gaussienne trop Simple

⇒ Nécessité de Beaucoup d 'Echantillons



- « k plus proches voisins » à rapprocher :
 - ⇒ de la Classification Automatique (notion de « Proximité »)
 - ⇒ des Techniques Bayésiennes (hypothèses sur la forme des classes en moins)
- Convergence « remarquable » quand nb échant. $\rightarrow \infty$
- Simplicité et Qualité \Rightarrow Méthode de référence pour l'évaluation des méthodes connexionnistes

Introduction

Codage

Analyse

Apprentissage & Décision

K-ppv

Principe :

Déterminer la classe de chacun des k points les plus proches de x dans \mathcal{R}^n parmi les formes d'apprentissage et affecter x à la classe la plus représentée

Introduction

Codage

Analyse

Apprentissage & Décision

K-ppv

Inconvénients :

- Temps de calcul en Décision
 - ⇒ Calcul de N distances dans un espace à m dimensions
- Organiser l'espace de représentation des formes (pavage, tri, hiérarchie...)

Introduction

Codage

Analyse

Apprentissage & Décision

*Méthodes
Stochastiques*

- Prise en Compte du Contexte
- Chaînes de Markov
- Modèles de Markov Caché
- En Parole et Ecriture Cursive

- Réaliser un appariement élastique
- Trouver le prototype nécessitant le minimum de déformation
- Vocabulaire limité
- Utilisées en reconnaissance de la parole (recalage temporelle) et en reconnaissance de l'écriture cursive (recalage spatiale)

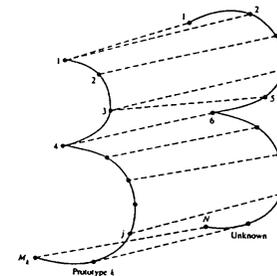


Figure 3.2 : Appariement élastique

Programmation Dynamique

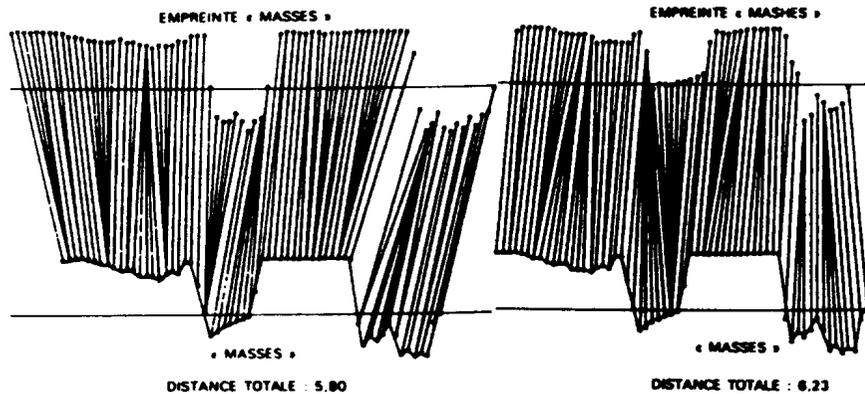


Figure 5.2. Comparaison entre les mots “masses” et “mashes” donnée dans [Lev84].

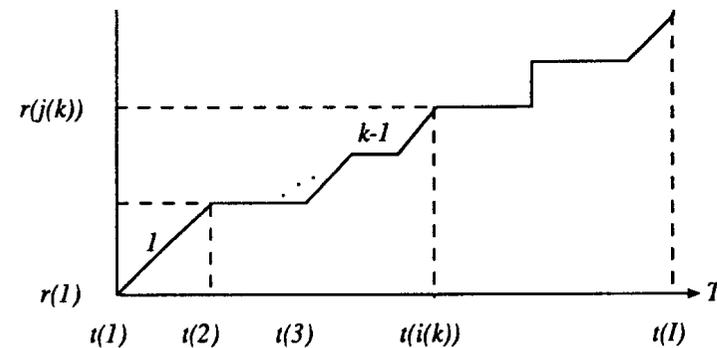


Figure 5.3. Exemple de chemin de recalage.

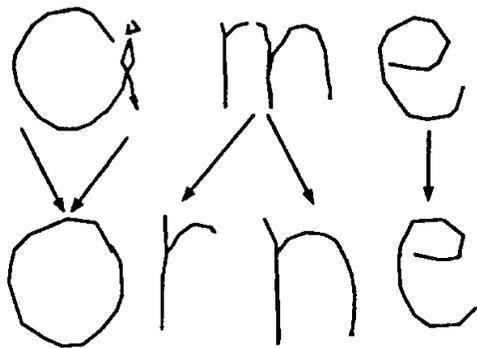


Figure 6.53. Appariement entre les mots “cime” et “orné”.

e	4			•	
n	3		•		
r	2		•		
o	1	•	•		
		1	2	3	4
		c	i	m	e

Introduction

Codage

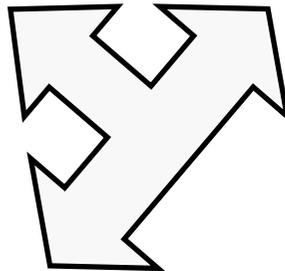
Analyse

Apprentissage & Décision

*Approches
Syntaxique et
Structurelle*

Comparaisons de Chaînes

- Programmation Dynamique
- Distances de Chaînes



Isomorphismes de Graphe

Grammaires de Langages

- Noam Chomsky

Introduction

Codage

Analyse

Apprentissage & Décision

*Choix des Attributs
et des Primitives*

- Taille de l'ensemble d'apprentissage /
Dimension de l'espace de représentation ?
- Choix des Traits Caractéristiques ou des
Primitives ?

Introduction

Codage

Analyse

Apprentissage & Décision

*Evaluation des
Classifieurs*

- Tests et Validations :

- ⇒ Resubstitution

- ⇒ « Hold-out »

- ⇒ « Leave-one-out »

- ⇒ « Bootstrap »

- Type d 'Erreurs :

- ⇒ Faux Rejet

- ⇒ Fausse Reconnaissance

- ⇒ Confusion

Introduction

Codage

Analyse

Apprentissage & Décision

*Evaluation des
Classifieurs*

- Taux d 'erreur = taux de rejet (faux rejet) + taux de confusion (fausses reconnaissances et confusions)
- Taux de reconnaissance = 100% - taux d 'erreur
- Bornes d 'erreurs (avec un nombre de formes infini)

$$err_B \ll \dots E_k \ll E_{k-1} \dots \ll E_2 \ll E_1 \ll 2err_B$$