Outils Modernes de Conception

Ingénierie de la Fiabilité

(Extraction de Connaissances & Fiabilité)

Ecole Centrale de Lyon - Métiers 3A

2009-2010

Alexander Saidi

0.1 Propos

- Cycle de Vie de proudits inustriels

- Prévision de la Fiabilité et la défaillance par des **méthodes informatiques modernes**

  ➥ Phase design optimisé par l'analyse de la fiabilité des systèmes

Plan

1- Apreçu des méthodes générales de l'extraction de connaissances

  ➥ A travers des exemples simples

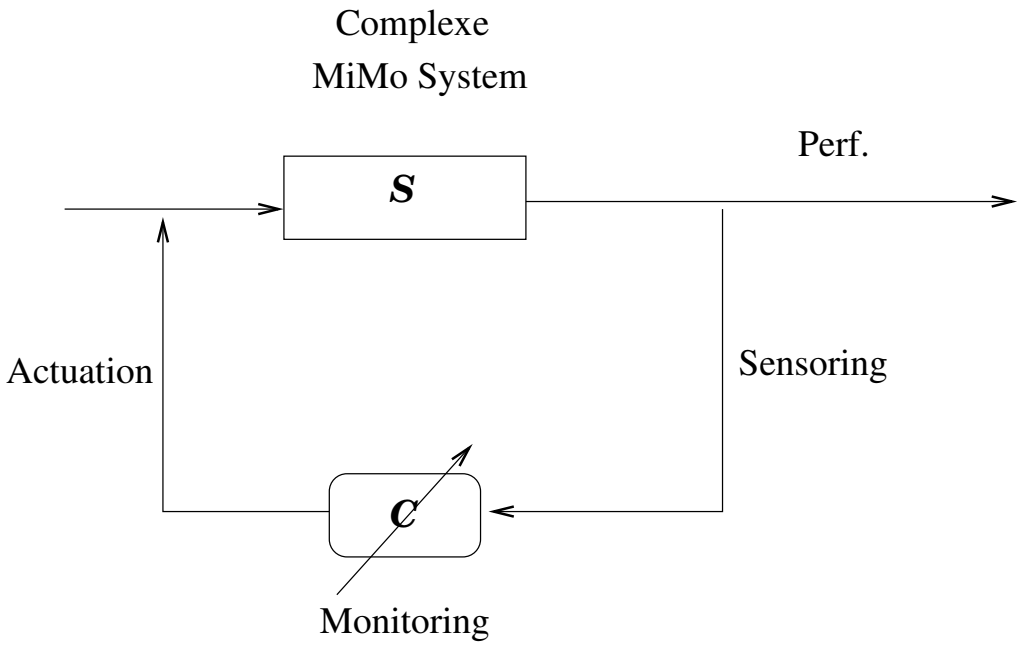
2- Lois de défaillance et de fiabilité des processus complexes

  ➥ Fiabilité et Cycle de vie de produits

  ➥ Métriques et Distributions

  ➥ Utilisation du cardre (Réseau) Bayésien, Réseaux de Neurones, etc.

0.1.1 Une vue Syntétique de la fiabilité



- Analyse Fiabiliste : principalement **S** (cf. Chapitre / séance 2)

- Synthèse Fiabiliste : Fiabilité de la fonction *Perf.* cf. *la semaine de 4 jeudis !*

➥ Exemples, ABS, Suspension Hydro-active, etc.

0.2 Un exemple d'analyse

Analyse des données de fiabilité du type succès/échec.

➙ Succès/échec des composants d'un système plus complexe.

• L'objectif : données + hypothèse (*a priori*) ➙ prédiction (*a posteriori*)

• Dans cet exemple : données disponibles à propos de générateurs d'urgence (diesel) en dépannage dans une centrale nucléaire.

➙ On a observé leur comportement (démarrage ou non) lors d'une sollicitation (en urgence).

☞ N.B. : un autre exemple similaire (succès/échec) peut être les résultats à l'allumage de missiles (militaires) construits par différents fabricants. Voir plus loin.

• Ce type de données est traitée avec une distribution **Binomiale**.

☞ La *Binomiale* est appropriée dans le cas de test d'un nombre fixe n de composants testés, où les tests sont supposés indépendants étant donné une proba de succès π .

• La fonction de densité de probabilité dans ce cas : $f(x|n,\pi) = \begin{pmatrix} n \\ x \end{pmatrix} \pi^x \, (1-\pi)^{n-x}$

Où $0 \leq \pi \leq 1$ = la proba de succès.

☞ NB. 1 : la loi binomiale n'est pas adaptée si les tests ne sont pas indép ; De plus, elle ne s'applique que si tous les événements ont la même probabilité de succès.

☞ NB. 2 : la distrib de **Bernoulli** est un cas spécial de la Binomial pour $n = 1$.

☞ NB. 3 : Parfois, on utilise le temps d'échec (défaillance) comme une donnée de type succès/échec par rapport à un temps spécifié t.

➙ C'est à dire : x sera le nbr de défaillances parmi n items testés avant le délai t .

➙ La raison principale pour cela est d'éviter un modèle de temps défaillance (continue temporel) ;

Mais dans ce cas, il y a beaucoup de perte d'information

➥ on ne l'utilise pas comme cas général.

Précisions :

• Pour le model Binomial, la probabilité π est le paramètre inconnu que l'on veut estimer.

• Dans un model succès/échec, l'équation $f(x|n,\pi) = \begin{pmatrix} n \\ x \end{pmatrix} \pi^x \, (1-\pi)^{n-x}$

est comme une fonction de π pour x succès observés,

➙ Elle est la fonction de **vraissemblance** (au sens Bayesian) pour les données binomiales.

• S'il y a m BDs binomiales (soit $x_1, ..., x_m$ succès sur $n_1, ..., n_m$ tests), alors sous l'hypothèse d'indépendance et de taux π constante, la fonction de vraissemblance sera le **produit** des m vraissemblances individuelles (chacune donnée par l'équation ci-dessus).

On a donc la vraissemblance. ../..

• Pour compléter le modèle et faire une prédiction, on doit spécifier une distribution a priori pour π (cf. Bayes).

• La distribution adaptée pour π est celle qui est conjugée.

➙ Une loi a priori conjuguée est celle que a **la même forme** que la distribution a posteriori.

• La *distribution a priori conjuguée* pour une donnée binomiale est une distribution **Beta** :

$$p(\pi|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\,\pi^{\alpha-1}(1-\pi)^{\beta-1} \qquad 0 \leq x \leq 1,\ \alpha > 0,\ \beta > 0$$

➙ Avec α interprété comme le nombre *a priori* de succès du composant testé et

➙ β comme le nombre *a priori* des échecs et

➙ Donc $\alpha + \beta$ est comme la taille *a priori* des données (la BD de référence).

• La distribution *Beta* est un choix naturel pour la loi a priori de π d'autant que son support est l'intervalle $(0, 1)$.

• On a donc notre vraisemblance + a priori. ../..

La prédiction (ajustement de l'*a priori*) :

• Après calculs, la distribution *a posteriori* de π (conditionnée par x succès observés parmi n tests) est de la forme : $p(\pi|x) \propto \pi^{\alpha+x-1}\,(1-\pi)^{\beta+n-x-1}$

➙ Cela veut dire que la distribution a posteriori de π sachant x est :

$$\pi|x \sim Beta(\alpha+x, \beta+n-x)$$

Un exemple : les générateurs de secours

• Les données (*USA nuclear power plants*) sur une centrale nucléaire (no 63) sont :

– $x = 212$ succès au démarrage sur 212 sollicitaions de génrétauers (tests).

– Cependant, les données de la centrale no 62 (référence) donnent 273 démarrages avec succès sur 278 (donc 5 échecs)

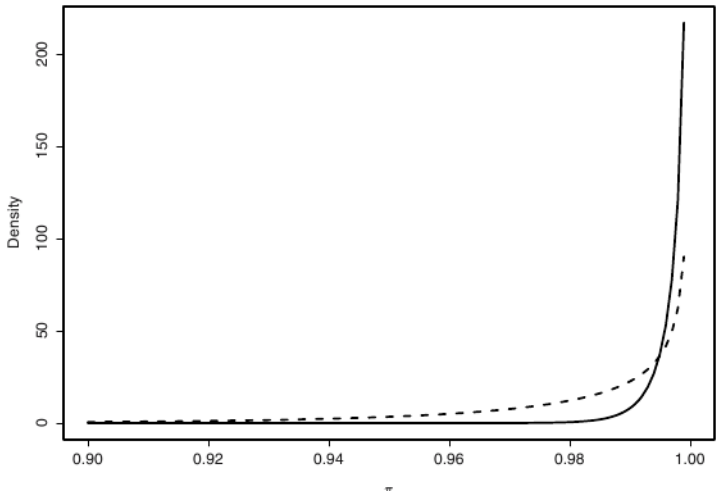
– mais on dispose seulement des 10% des données de la 62.

• Dans ce cas, la 62 (la référence) génère une distribution *a priori* pour la probabilité de succès π :

$$Beta(\alpha = (273/278)/27.8,\ \beta = (1 - 273/278)/27.8)$$

• De la distribution a posteriori précédente, la probabilité π pour la centrale 63 sera :

$$Beta(\alpha + x, \beta + n - x) = Beta(239.3 = (273/278)/28.3 + 212,\ 0.5 = (1 - 5/278)/28.3 + 0)$$



La figure montre les distributions *a priori* et *a posteriori* pour la probabilité de succès π

La distribution a posteriori (lignes pleines) est plus piquée que l'a priori (pointillée).

Les données démontrent davantage d'évidence en faveur d'un taux de succès élevé.

1 Introduction à l'EC

- Quantité et disponibilité d'informations (Data) dans tous domaines
  - ➥ En général, les données brutes (+ bruits !) sont accessibles à tous
  - ➥ Des données de qualité sont plus rares
  - ➥ L'exploitation des données (cf. motifs pertinents) est un atout
    - ➤ Données → Connaissances → Décisions

- Tous domaines : économie, éducation, finances, commerces, Industrie, santé, ·...

1.0.1 L'apprentissage

• **L'apprentissage** = méthode pratique de définition de **concepts**.

⇒ L'humain (enfant) utilise des instances de concepts pour se représenter le monde :

➥ animaux, plantes, homme, femme, jouet, ...

➥ On apprend des instances particulières → On choisit des attributs

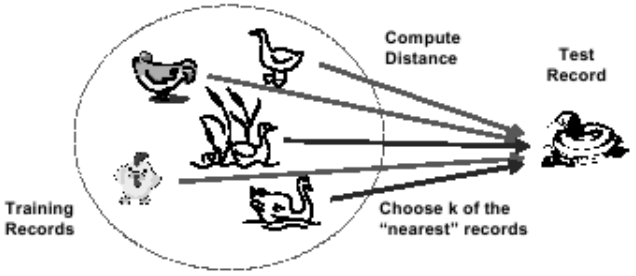
➥ On forme des modèles de classification

➥ On utilise ces modèles pour identifier des objets similaires.

• Deviner le suivant :

1,2,3,4,5... 1,2,3,5,8,13,... 1,2,3,5,7,11,13,... 5 ? 5 ? 5 =6

➥ Les données (chiffres) contiennent des connaissances



1.0.2 Fouillez ! Vous saurez

De la qualité des données, bruits et l'information (connaissance) extraite

• Crimes et chatiments en Floride-USA (1973-78)

Meurtrier	**Jugement**	
	peine de mort	**autre**
Black	59	2547
Blanc	72	2185

➥ 3.2% des meurtriers blancs ont été condamnés à la peine de mort

➥ 2.3% des meurtriers blacks ont été condamnés à la peine de mort

➤ On ne peut franchement pas accuser ! ../..

• **Données détaillées** : le paradoxe de Yule-Simpson

Victime	Meurtrier	Jugement	
		peine de mort	**autre**
Black	Black	11	2309
	Blanc	**0**	111
Blanc	Black	48	238
	Blanc	72	2074

➥ Pour les victimes Blacks, 4,7% des meurtriers Blacks sont condamnés à la peine de mort vs. **0%** pour les meurtriers Blancs

➥ Pour les victimes Blancs, **16,8%** des meurtriers Black sont condamnés à la peine de mort vs. 3,4% pour les meurtriers Blancs

1.1 Des données brutes à la Connaissance (information)

• **Pourquoi :** traiter et analyser de grandes quantités de données ➙ connaissances

➥ Données (BDs) économiques, médicales, industrielles, scientifiques, vie, . . .

• **Techniques** : *Management des connaissances* (KM)

➥ Statistiques (*observation* → *loi*) & Algorithmiques (modèles),

+ Intelligence Artificielle pour manipuler les connaissances ➙ Décisions

➥ KM = le résultat de la convergence de ces deux domaines techniques.

• **Exemples d'application** : contrôle de véhicules autonomes, modélisation des risques opérationnels, analyse et localisation des gènes, Basket Analysis, Prédiction, météo, finaces, jeux . . .

Un exemple (domaine difficile) :

```
AAB24882        TYHMCQFHCRYVNNHSGEKLYECNERSKAFSCPSHLQCHKRRQIGEKTHEHNQCGKA
AAB24881        --------------------YECNQCGKAFAQHSSLKCHYRTHIGEKPYECNQCGKA
                                     ****: .***:  * *:** * :****.:* ******

AAB24882        PSHLQYHERTHTGEKPYECHQCGQAFKKCSLLQRHKRTHTGEKPYE-CNQCGKAFAQ
AAB24881        HSHLQCHKRTHTGEKPYECNQCGKAFSQHGLLQRHKRTHTGEKPYMNVINMVKPLHN
                 **** *:***********:***:**.:  .****************     :   *.: :
```

FIGURE 1.1 – Un alignement de séquence entre deux protéines humaines.

1.2 Raisons d'émergence de l'EC

- **Constat :** les domaines sont complexes, et les données abondantes

  ➥ Les données sont souvent incomplètes ou erronées

  ➥ Elles peuvent décrire plusieurs situations (générales)

    ➤ Par exemple : en médecine, la même combinaison de sympthômes peut être observée dans différentes pathologies.

  ➥ Une solution classique : Statistique / analyse de données (2 experts requis)

  ➥ Une autre solution (IA) : Système Expert + incertitude dans le raisonnement.

  ➥ Unifier et enrichir ces techniques (surtout si beaucoup de données) :

    ➔ DM = une approche quantitative et qualitative.

../..

Un scénario courant (où le lien = la connaissance) :

1 • L'expert sait : `A a une influence sur B, si B est observé, il y a de forte chance que C se produise,` ....

➥ Mais il a des incertitudes (conviction $\pm$ précise sur les liens entre ces faits).

2 • Il dispose aussi d'un ensemble de données contenant des connaissances qui ne sont pas directement accessibles à un analyste (car noyée dans les chiffres).

➥ Il transforme ces données en modèle (e.g. de causalité) qui met en lumière les liens.

➥ Pour rendre cette connaissance opérationnelle $\rightarrow$ il quantifie les **incertitudes**

Les techniques EC permettent de transformer une connaissance subjective en chiffres, + transformer les connaissances contenues dans les chiffres en modèle interprétable (e.g. Loi de proba.)

1.3 Aperçu des méthodes Bayesiennes

- Peut être supervisée ou non (expert en amont/aval ?)
- Directement issues du domaine des probabilités/statistiques (conditionnelle)
- Utilisation des données historiques disponibles pour calculer
  - ➥ la probabilité pour qu'un certain événement (ou propriété) ait lieu.

- Exemple (thèse EADS) : dans le cas de fiabilité des pièces mécanique (aide d'expert)
  - ➥ mesurer la probabilité de la *rupture* de la pièce
  - ➥ étant donné un certain environnement (complexe) d'utilisation.

1.3.1 BNs : un exemple introudctif

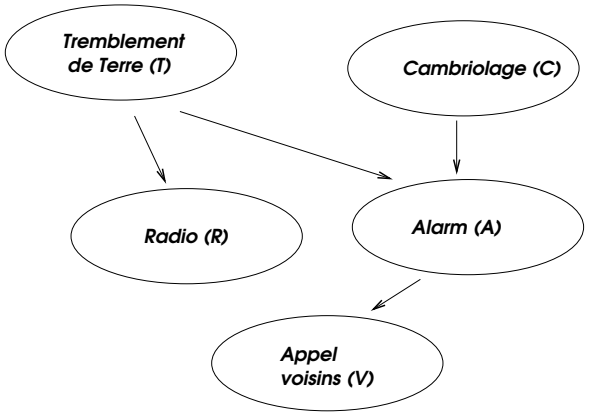
• Un tableau d'observations (variables) : *cambriolage* (C), *Tremblement de terre* (T), *Alarme* (A), *Annonce à la radio* (R) (et accessoirement *Appel des voisins* (V) si A).

• Par des calculs sur le tableau de données, on constate quelques indépendances :

➥ (C) et (T) sont indépendants,

➥ C et R indépendantes p/r à T : pas d'événement qui cause à la fois C et R.

➥ C et R indépendantes p/r à T : T peut causer un appel Radio, pas suite à un C.



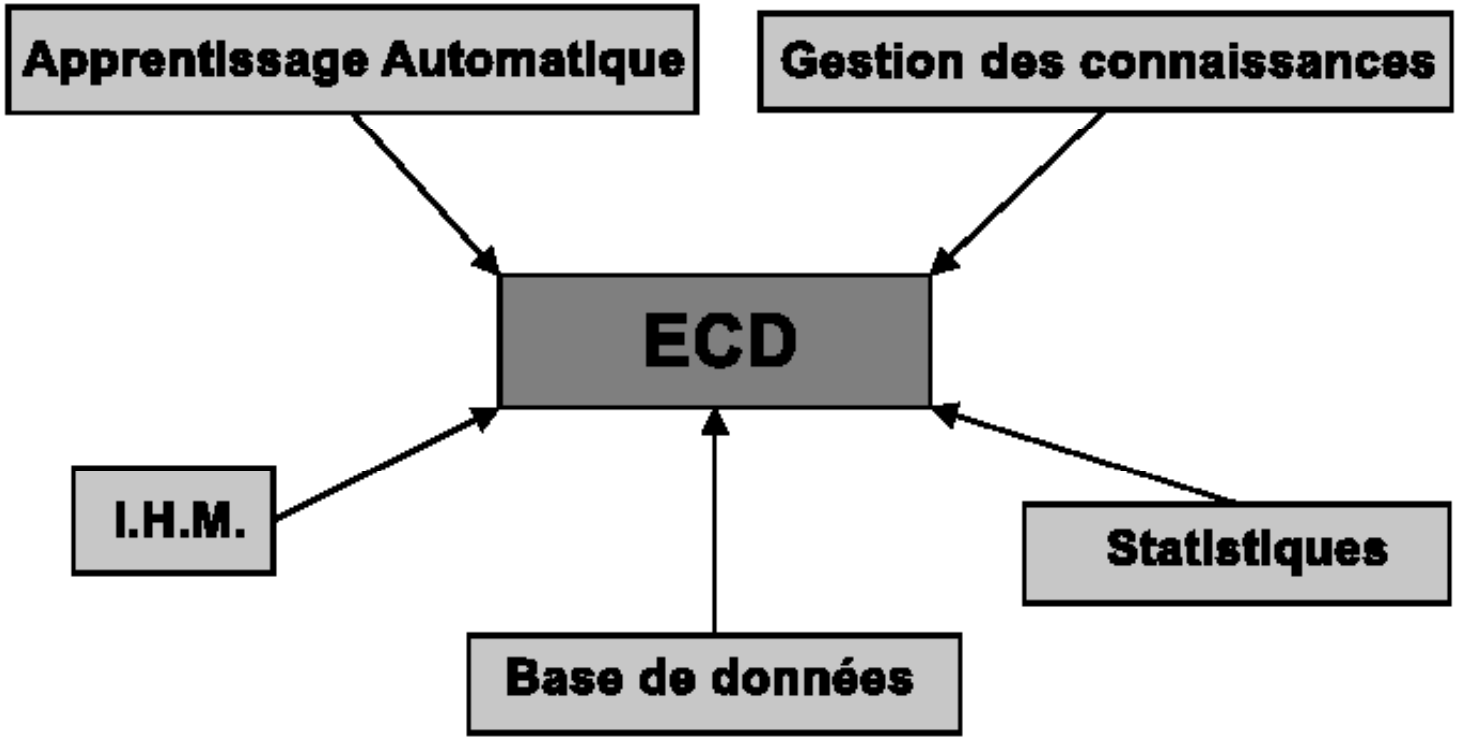
T	C	P[A|T,C]	P[~A | T,C]
T	T	0.90	0.10
T	F	0.20	0.80
F	T	0.90	0.10
F	F	0.001	0.999

• Etant donné ces indépendances, la probabilité **jointe** des 4

$$Pr[A, R, T, C] = Pr[A, |R, T, C] * Pr[R|T, C] * Pr[T|C]Pr[C]$$

➤ Les indépendances réduisent grandement le nombre de probas jointes.

1.4 Extraction de Connaissances : Domaine multi disciplinaire



1.5 Objectifs de l'Extraction de Connaissances

• La tâche de l'Extraction de Connaissances est divisée en 2 catégories :

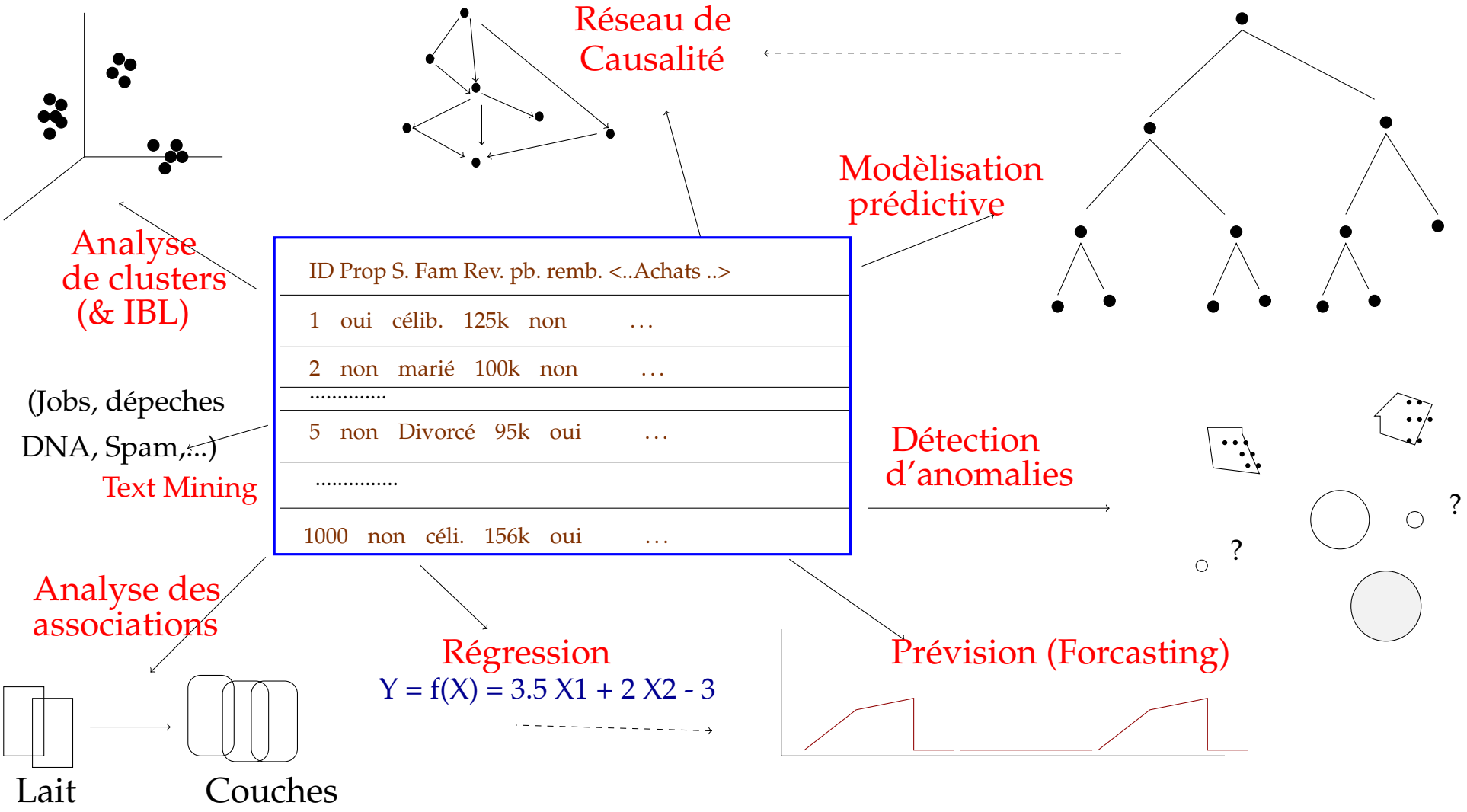
➤ **Prédictive :** prédiction des valeurs des attributs (dépendantes) en fonction des variables d'explication (indépendantes)

➤ **Descriptive :** Extraction de motifs (corrélations, clusters, tendances ou anomalies, ...) qui résument les relations entre les données.

➤ Apprentissage **supervisé** = création de modèles en formant des *définitions de concepts* à partir de des données contenant des classes prédéfinies.

➤ Apprentissage **non supervisé** = création de modèles à partir de de données sans l'aide de classes prédéfinies

1.5.1 *Principales tâches de l'EC*



1.6 Fouille de données : quelques exemples indicatifs

Exemple 1-1 : *Quel rapport ?*

Pourquoi beaucoup de tournois de Golf télévisés sont sponsorisés par des courtiers (brokers) en ligne ?

➔ Découvert dans les fichiers :

1 ⇒ plus de 70% des investisseurs en ligne sont des hommes d'age environ 40 ans et qui jouent au Golf.

2 ⇒ 60% de tous les investisseurs en stock option sont des Golfeurs.

Exemple 1-2 : *Let's shake it !*

Est il utile, pour une compagnie de musique, de faire de la pub pour la musique ***Rap*** *dans des magazines pour les* ***seniors*** *?*

➔ Oui : les seniors offrent souvent de la musique Rap à leur petits enfants (adolescents).

Exemple 1-3 : *The big brother !*

Comment les banques (service CB) peuvent-elles suspecter une carte volée, même si le propriétaire n'est pas conscient du vol ?

➔ Beaucoup de ces compagnies (de crédits) stockent un modèle général de nos habitudes d'achat avec la carte.

➔ Ce modèle alerte la compagnie d'un vol possible lors d'une transaction qui ne rentre pas dans le profil général de nos habitudes.

Exemple 1-4 : *La fertilisation in vitro :*

– Collecter des oeufs + fertilisation + transfert vers l'utérus de la "mère" porteuse

– Comment choisir les "meilleurs" oeufs avec un maximum de chance de survie.

– Le nombre de caractéristiques (**attributs ou variables**) assez grand (env. 60)

➥ la morphologie, oocyte, follicule (taille oeuf), qualité sperme, etc.

➥ difficile de trouver une corrélation pour prédire la survie de l'embryon.

• Utilisation des techniques **Apprentissage Automatique** dans un projet EC en Angleterre pour aider à la sélection.

1.7 L'émergence de l'EC (forme récente)

□ Grand volume de données disponibles (volume doublé tous les 20 mois).

□ Omni présence des ordinateurs → on sauvegarde des données (en N versions !)

□ Le rapport $\frac{taille}{prix}$ du support de stockage en hausse constante

➥ Sauvegarde de nos choix / décisions / habitudes financières, etc

⇒ Dans ces *données*, il y a potentiellement de l'*information* utile, rarement explicite

⇒ Opportunités économiques, médicales, Industrielles, Bio-xx ....

⇒ *Modélisation* et *Généralisation* vs. vérification d'hypothèses

⇒ Traitement de tous types de données par des méthodes algorithmiques

➥ *algorithmes Génétiques*, *Réseaux de Neurones*, les méthodes algorithmiques pures (*ID3, C45,* ... → arbres / règles) , *inférence et réseaux Bayesiennes nouveaux*

1.7.1 Recherche de motifs (pattern) dans les données

➤ **Pas une nouvelle science/discipline**

➥ L'homme a toujours procédé à la recherche de motifs (*patterns*) :

- les chasseurs, les fermiers, les politiciens : recherche de patterns

- les scientifiques : découvert de motifs qui expliquent les phénomènes physiques

→ proposition de théories pour **prédire** une nouvelle situation.

- les entrepreneurs (bourse, opportunités)

- même les amoureux : patterns dans les réponses des partenaires.

➥ Les économistes, statisticiens, prévisionnistes (e.g. météo), etc ont déjà travaillé sur cette idée (e.g. *régression linéaire*)

➤ Ce qui est nouveau : **Techniques et moyens Informatiques**

Des domaines très demandeurs :

• **Industrie** : Fiabilité et robustesse, gestion des risques, prévisions ...

• **Santé :**

• **Economie/Marketing :**

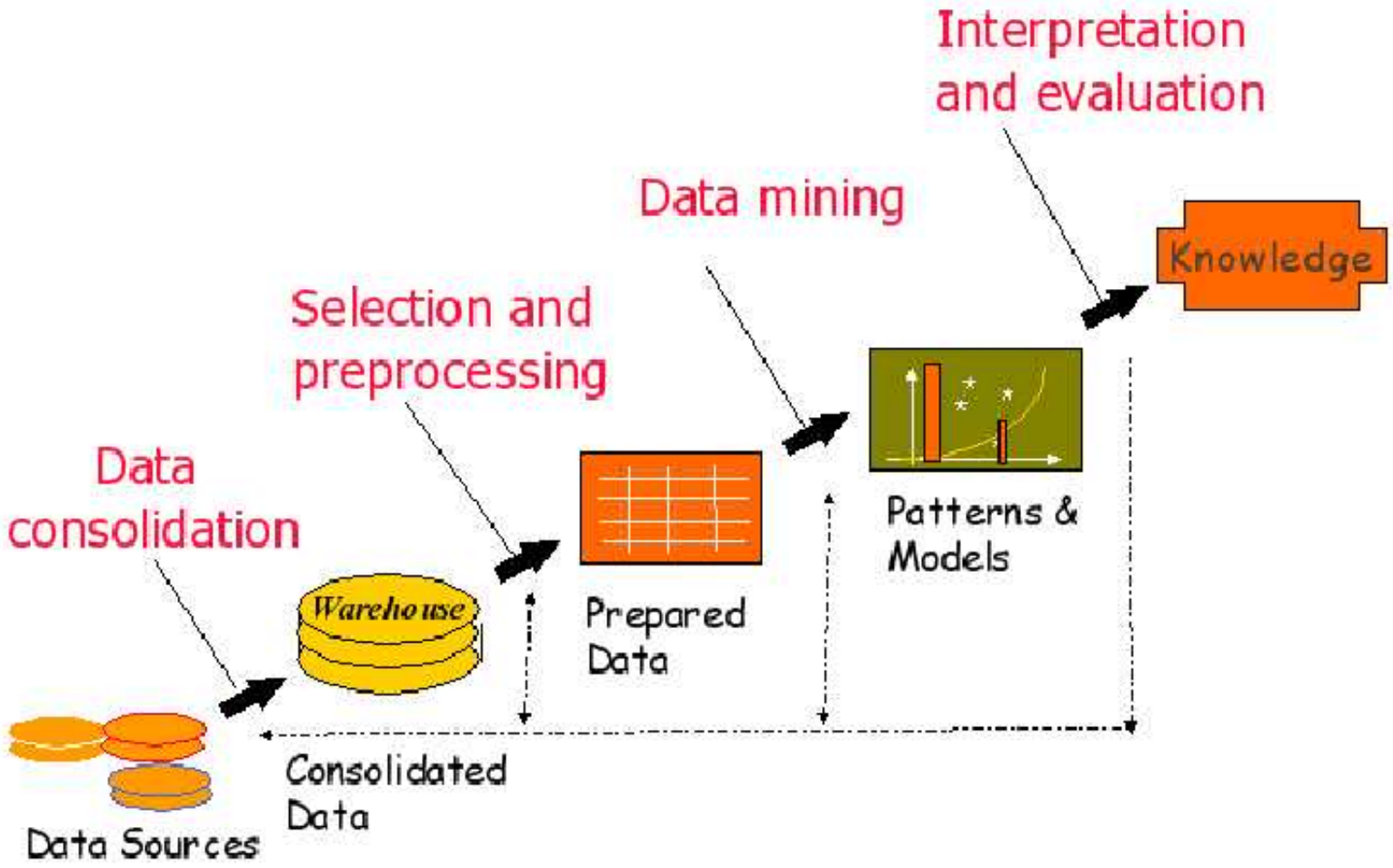
⇒ Orientation récente de l'économie : très compétitive et orientée service.

⇒ Les données = le **fuel de base** de la prospérité économique, si bien fouillées !

➥ Ex : Gestion de clients (le problème de fidélité) sur un marché compétitif.

- On peut analyser le comportement des clients,
- Dégager des caractéristiques des clients fidèles (et ceux qui risquent de partir !).
- Traitement particulier des clients fidèles (coûterait cher si appliqué à tous)
- Identifier les clients intéressés par d'autres services de l'entreprise (avec des promotions ciblées pour les attirer).

1.7.2 KDD : le Processus d'Extraction de Connaissances dans les BDs



1.8 Qu'est-ce que les ordinateurs peuvent apprendre

• Le processus d'apprentissage est complexe.

Fait : une simple expression de la "vérité" (fait avéré) $\rightarrow 2 \times 2 = 4$

Concepts : un ensemble d'objets, symboles ou événements groupés ensemble qui ont quelques caractéristiques en commun → un oiseau

Procédure : une séquence pas à pas d'actions pour résoudre des problèmes

→ scenario de construction de ... X

Principes : niveau supérieur d'apprentissage .

➥ des vérités générales / des lois basiques utiles à d'autres faits/vérités.

→ la règle de 3

Les ordinateurs apprennent bien les concepts :

• **Concept** = la sortie d'un processus de Extraction de Connaissances .

→ Exemples : maison, homme, automobile, bon client, comestible, panne ...

• La forme des concepts appris est imposé par l'outil DM.

• Ils sont représentés par : ***arbres, règles, réseaux, équations/fonction, ...***

⇒ Les arbres (de décision) et les règles (compréhensibles pour l'humain)

⇒ Idem pour les réseaux Bayesiens (graphes)

⇒ Les réseaux (de neurones) et les équations, ... (moins évidents !)

➔ le "comment / pourquoi" : black-Box vs. explication

1.9 Recherche de Concepts par Généralisation (Induction)

Enumération de l'espace de concepts : Espace de Versions (théorique)

• L'espace de description de concepts consistents

➥ **L** (least) et **G** (greatest) general descriptions

L : les descriptions *les plus spécifiques* qui couvrent tous les exemples positifs et aucun exemple négatif (de manière spécifique).

G : les descriptions *les plus générales* qui ne couvrent aucun exemple négatif mais tous les exemples positifs (de manière générale).

• On a juste besoin de maintenir et mettre à jour **L** et **G**

• **Inconvénients :** C'est encore coûteux en temps de calcul et..

Permet une définition théorique et claire mais ne résout pas de problème pratique !

□ **Exemple :**

Soit le vocabulaire : couleurs ∈ {rouge, vert}, animaux ∈ {vache, poule}

Ex. positifs	Ex. négatifs	L	G
<>		{ }	{< * , * >}
<vache, verte>		{<vache, verte>}	{< * , * >}
	<poule, rouge>	{<vache, verte>}	{<*, verte>, <vache, *>}
<poule, verte>		{<*, verte>}	{<*, verte>, <vache, *>}

• Si l'on ajoute <vache, rouge> ? :

→ Si ex. positif : ajouter <vache, *> à L

→ Si ex. négatif : retirer <vache, *> de G


Inconsistance si les exemples positifs et négatifs se contredisent.

1.9.1 Exemple d'expressions de concepts (3 vues)

■ 1- Une **vue classique** : définition claire et déterministe du concept

⇒ Tous les exemples d'un concept particulier sont, de manière égale, représentatifs de ce concept.

⇒ On peut voir (tester) si un individu est un exemple d'un concept particulier.

• Un exemple :

⇒ Définir un risque acceptable pour un prêt (un bon risque = un bon client) :

Si les revenus annuel $\geq$ *30,000*

& *L'ancienneté dans l'entreprise* $\geq$ *5 ans*

& *Possède sa maison = vrai*

Alors Risque de prêt acceptable = vrai

■ 2- Une **vue probabiliste** : les concepts représentés par des propriétés probables.

⇒ Ces concepts = des généralisations des **instances**.

• Exemple : une vue probabiliste d'un risque acceptable (pour un prêt) :

> • *La moyenne annuelle des revenus des individus qui paient à temps leurs mensualités est de 30000.*
>
> • *La plupart des individus qui sont des bons risques de crédit ont travaillé au moins 5 ans pour la même entreprise.*
>
> • *La majorité des bons clients (risques) de crédit possèdent leur propre maison.*

⇒ Des grandes lignes des caractéristiques représentatives d'un bon risque :

> *Un client possédant sa maison avec un revenu annuel de 27000, embauché dans la même entreprise depuis 4 ans peut être classé à 0.85 comme un bon risque.*

■ 3- Une **vue à base d'instance** :

⇒ un nouvel exemple (**instance**) est une instance d'un concept si elle est *assez proche* d'un ensemble d'un ou plusieurs exemples connus du concept

→ cf. apprentissage à base d'instance (IBL).

⇒ Dans l'exemple de prêt, un demandeur = *un bon risque s'il est assez proche d'un ou plusieurs instances stockées présentant un bon risque.*

• L'homme utilise beaucoup cette vue : on stocke des exemples de concept qui seront utilisés pour classifier de nouvelles instances . ../..

⇒ Exemple : une liste possible de bons risques :

Instance 1 *: revenus annuel = 32000*

& dans la même boîte=6

& Possède sa maison

Instance 2 *: revenus annuel = 52000*

& dans la même boîte=16

& Locataire

Instance 3 *: revenus annuel = 28000*

& dans la même boîte=12

& Possède sa maison

⇒ On peut associer une probabilité (d'appartenance au concept) à chaque cas.

1.10 Manque de données et Simulation : Monte-Carlo

• Méthode générale utilisée dans les calculs probabilistes basés sur l'échantillonnage à répétition

➙ distributions diverses de variables aléatoires.

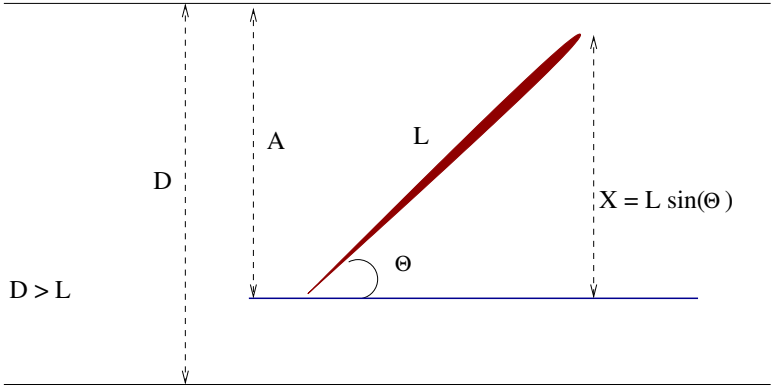
• Le nom date de la 2e guerre mondiale ; vient de *Monte Carlo* (Roulettes, Casions !)

• La méthode a été utilisée dans le développement de la Bombe atomique ; appliquée aux problèmes de diffusuion aléatoire des neutrons dans le matériel fissible.

• Un exemple (simple des origines) : le problème des aiguilles de Buffon (1773) :

On lance plusieurs fois une aiguille sur une feuille cadriée et l'on observe si elle croise une des lignes horizontales.

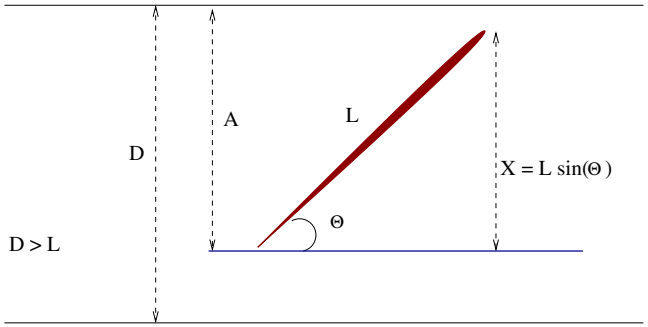
➥ Quelle est la probabilité que l'aiguille croise une des lignes verticales ?



- La position de l'aiguille relative à la ligne la plus proche est donnée par un vecteur aléatoire $V_{(A\theta)}$

$$A \in [0, D], \quad \theta \in [0, \pi]$$

- L'aiguille croise une des lignes si $A < L\ sin(\theta)$ [1]



- A = dist. de la pointe et la ligne la plus proche (zone contenant l'aiguille)
- Le vecteur $V_{(A\theta)}$ est distribué uniformément sur la zone $[0, D] \times [0, \pi]$
- Sa fonction de densité de probabilité (PDF) est $\frac{1}{D\ .\ \pi}$ $(\frac{1}{D} \times \frac{1}{\pi})$

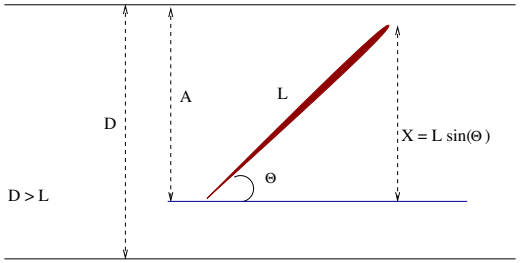
  Car on a : $f_{A\Theta}(a, \theta) = f_A(a).f_\Theta(\theta)$ et A et θ sont indépendants (d'où la multiplication).

- La probabilité pour que l'aiguille croise une des lignes est (CDF) :

$$p = Pr[A < L\ sin(\theta)] = \int_0^{\pi} \int_0^{L.sin(\theta)} \frac{1}{D\ .\ \pi}\ d(A)\ d(\theta) = \frac{2L}{D\ .\ \pi} \qquad [2]$$

• Rappel : la probabilité de croiser une des lignes :

$$\boxed{p = Pr[A < L\ sin(\theta)] = \frac{2L}{D\ .\ \pi}} \qquad [2]$$



• *Laplace* a proposé (pour s'amuser !) de calculer une approximation de la valeur de π par ce résultat : il a lancé l'aiguille n fois .

• Soit la variable aléatoire **M** représentant le nbr. de fois où l'aiguille a croisé une ligne.

➥ Si $E[M]$ = l'espérence de M, la probabilité de croiser une ligne : $\boxed{p = \frac{E[M]}{n}}$ [3]

• Les expressions [2] et [3] représentent la même probabilités ➥ $\frac{E[M]}{n} = \frac{2L}{D\ .\ \pi}$

➥ Un estimateur statistique de la valeur de π : $\boxed{\pi = \frac{n}{E[M]} \cdot \frac{2L}{D}}$

Estimation de π : on lance l'aiguille n fois, elle touche une ligne **m** fois.

➤ Dans $\pi = \frac{2.L.n}{E[M].D}$ on peut remplacer la variable aléatoire M par m et le nombre que l'on obtient une valeur (estimation chiffrée) de π.

• En 1864, pendant sa convalescence, le *Capitaine Fox* a fait des tests :

n	**m**	**L** (cm)	**D** (cm)	**plateau (des tests)**	**estimation** (π)
500	236	7.5	10	stationnaire	3.1780
530	253	7.5	10	tournant	3.1423
590	939	12.5	5*	tournant	3.1416

• Deux résultats importants (pour MCL) à en tirer :

(1) La première ligne du tableau : résultats pauvres

➥ *Fox* a fait tourner le plateau entre les essais

➥ Cette action (confirmée par les résultats) élimine le **biais** de *sa position*

➥ Il est important d'éliminer le biais dans l'implantation de la méthode MCL.

➥ Le biais vient souvent des générateurs de nombres aléatoires utilisés (équivalent à la position dans ces lancers d'aiguille).

(2) Dans ses expériences, *Fox* a utilisé le cas $D < L$ (dernière ligne du tableau).

➥ L'aiguille a pu croiser plusieurs lignes (le cas * : n=590, m=939).

➥ Cette technique est appelée la tech. de **réduction de variance** (de l'estimateur).

Bilan MCL :

• Le processus MCL est une technique générale d'Intégration (i.e. pour formule [2])

➥ peut être utilisé pour simuler tout processus influencé par des facteurs aléatoires.

• On l'utilisera dans l'estimation probabiliste des paramètres de fiabilité des systèmes complexes (cf. Chap 2).

➥ Elle est également utilisée dans des problèmes non probabilistes.

1.11 Exemples d'extraction de Motifs (Patterns) structurels

- Qu'est-ce ?
- La forme des entrées ?
- Comment les décrire ?

... Des exemples ....

L'expertise est indispensable en apprentissage (supervisé ou non supervisé)

1.12 Exemple d'un jeu in-door

Num	Temps(S)	Temperature(T)	Humidité(H)	Vent(V)	Jouer
1	ensoleillé	Elevée	Elevée	non	Non
2	ensoleillé	Elevée	Elevée	oui	Non
3	nuageux	Elevée	Elevée	non	Oui
4	pluvieux	Moyenne	Elevée	non	Oui
5	pluvieux	Faible	Normale	non	Oui
6	pluvieux	Faible	Normale	oui	Non
7	nuageux	Faible	Normale	oui	Oui
8	ensoleillé	Moyenne	Elevée	non	Non
9	ensoleillé	Faible	Normale	non	Oui
10	pluvieux	Moyenne	Normale	non	Oui
11	ensoleillé	Moyenne	Normale	oui	Oui
12	nuageux	Moyenne	Elevée	oui	Oui
13	nuageux	Elevée	Normale	non	Oui
14	pluvieux	Moyenne	Elevée	oui	Non

Résultats d'une première tentative d'apprentissage : (règles de classification)

Un 1er ensemble de règles apprises (indicatif) : une liste de décision

- *Si aspect=ensoleillé & humidité=forte alors jouer=non*	(1)
- *Si aspect = pluvieux & vent = vrai alors jouer=non*	(2)
- *Si aspect = nuageux alors jouer=oui*	(3)
- *Si humidité = normale alors jouer=oui*	(4)
- *Si* ***aucun ci-dessus*** *alors jouer=oui*	(5)

La même base d'instance avec deux attributs **numériques** :

Num	Temps(S)	Temperature(T)	Humidité(H)	Vent(V)	Jouer
1	ensoleillé	81	78	non	Non
2	ensoleillé	80	90	oui	Non
3	nuageux	83	80	non	Oui
4	pluvieux	75	96	non	Oui
5	pluvieux	69	75	non	Oui
6	pluvieux	64	70	oui	Non
7	nuageux	65	65	oui	Oui
8	ensoleillé	72	83	non	Non
9	ensoleillé	68	72	non	Oui
10	pluvieux	71	74	non	Oui
11	ensoleillé	75	69	oui	Oui
12	nuageux	70	77	oui	Oui
13	nuageux	85	70	non	Oui
14	pluvieux	73	82	oui	Non

- Attributs mixtes dans la BD ➙ **discrétisation**

Si aspect = ensoleillé & **humidité** $>$ **83** *Alors jouer=non*

1.12.1 Arbre de décision

- Stratégie *Diviser et Régner* (Divide & Conquer)

Outlook	Temp.	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

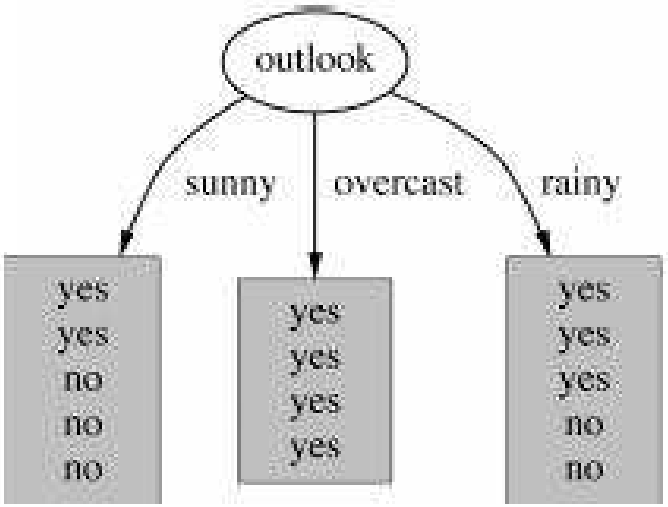
TABLE 1.1 – Données météo (jeu)

- Problème : quel attribut choisir ? ➙ notion d'information / pureté

info([2,3]) = 0.971 bits

info([4,0]) = 0.0 bits

info([3,2]) = 0.971 bits



⇒ On calcule la moyenne de ces valeurs en tenant compte du nombre d'instances de chaque branche : **5**, **4** et **5**

*info([2,3], [4,0], [3,2]) = (5/14) * 0.971 + (4/14) * 0 + (5/14) * 0.971 = 0.693 bits.*

➔ Cette moyenne : l'incertitude pour spécifier la classe d'une nouvelle instance (la part pour "outlook" à la racine).

• Notion de **gain d'information**

⇒ On aura :

gain(outlook) = 0.247
gain(Température) = 0.029
gain(Humidité) = 0.152
gain(Windy) = 0.048

⇒ On prend donc "outlook" pour scinder l'arbre à la racine.

→ = le seul choix pour lequel la fille est la plus "pure" (max de gain d'information)

⇒ On continue récursivement sur chaque noeud créé.

⇒ Les possibilités de branches sachant "Outlook= sunny" :

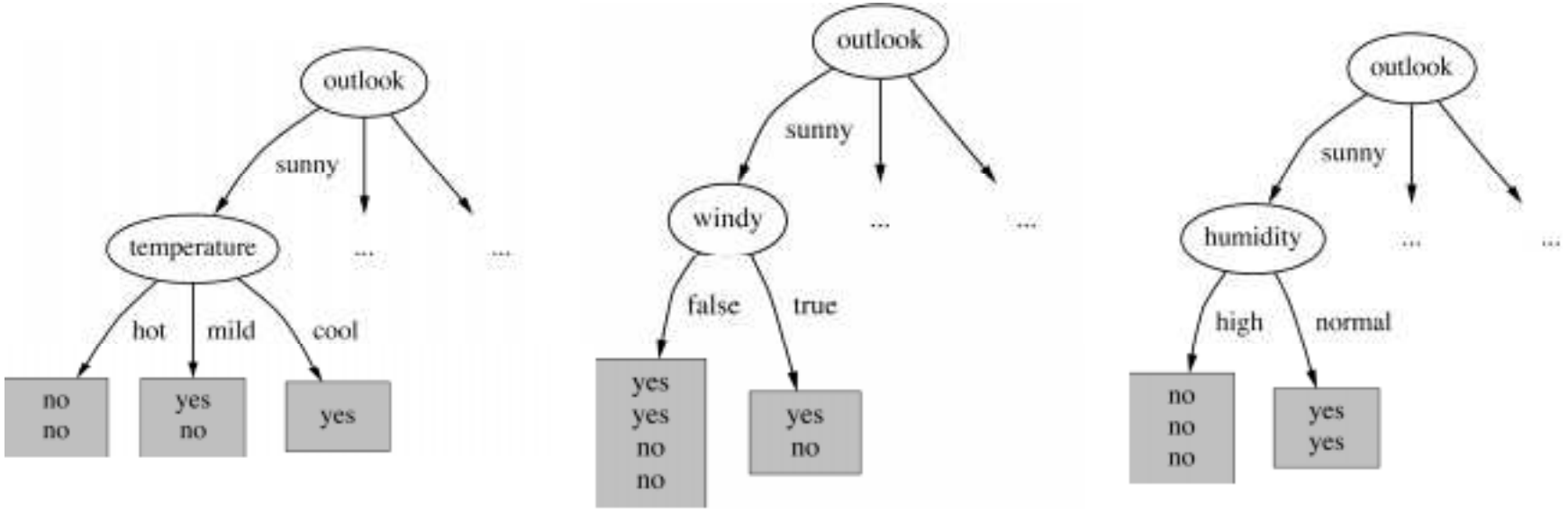


FIGURE 1.2 – suite création arbres de décision sur "sunny" pour "météo"

⇒ Remarque évidente : une nouvelle décision sur "outlook" (!) n'apporte rien.

⇒ Les gains des 3 attributs restants :

gain(Température) = 0.571

gain(Humidité) = 0.971

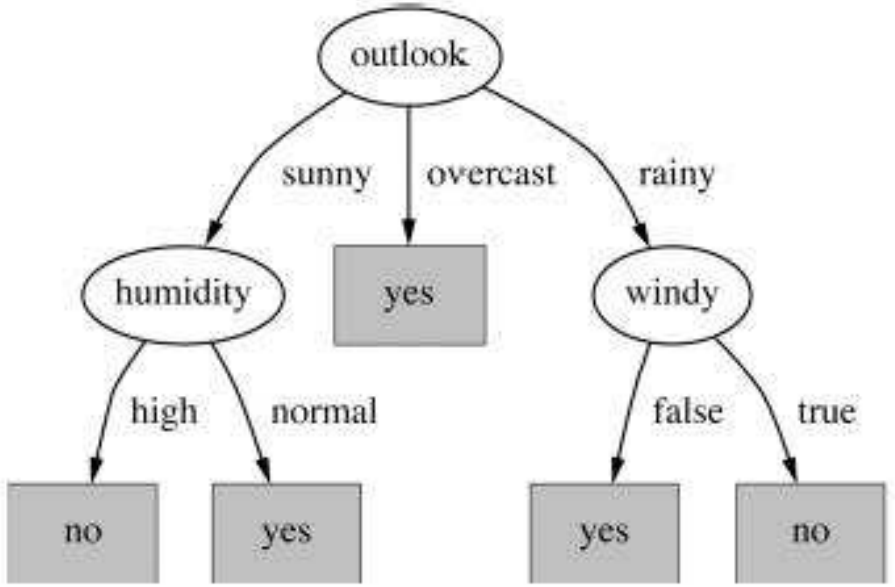
gain(Windy) = 0.020

⇒ On choisit "Humidity" pour scinder la branche → terminé pour cette partie.

../..

L'arbre de décision final :

Outlook	Temp.	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No



1.12.2 Le calcul de l'information

• La fonction utilisée : l'**entropie** (la *valeur d'information*)

$$entropie(p_1, p_2, ..., p_n) = -p_1 \times log\ p_1 \quad - p_2 \times log\ p_2 \quad ... \quad - p_n \times log\ p_n$$

➥ "log" des fractions p_i est négatif, l'entropie est positive (en nombre de bits)

➥ Les arguments p_i sont des fractions et la somme des p_i = 1.

Étant donné une distribution de probabilités (ici fréquences), **l'information nécessaire pour prédire un événement** est *l'entropie de la distribution*

➥ Par exemple : **info([2,3,4]) = entropie(2/9, 3/9, 4/9)**.

➔ L'entropie donne cette information nécessaire en nombre de bits

➔ Les "log" sont en générale **en base 2** → l'unité de l'entropie est **"bit"**.

• La propriété de la décision multi-niveaux ($p + q + r = 1$) :

$$entropie(p, q, r) = entropie(p, q + r) + (q + r) \times entropie(\frac{q}{q+r}, \frac{r}{q+r})$$

• N.B. : On peut calculer l'information sans avoir à manipuler les fractions individuelles.

➥ Par ex :

info([2,3,4]) = entropie(2/9, 3/9, 4/9)

= -2/9 . log 2/9 - 3/9 log 3/9 - 4/9 log 4/9

= [-2 log2 - 3 log3 - 4 log4 + 9 log9] / 9 .

⇒ C'est une manière habituelle de calcul d'information.

⇒ Ne pas simplifier *log9* (en *2log3*), on laisse tous les termes 2,3,4 et 9.

Remarques sur l'entropy :

L'**entropie** mesure *l'incertitude* et permet de quantifier le caractère aléatoire d'une distribution de probabilités.

➥ Elle permet de mesurer l'incertitude relative à l'appartenance des objets aux différentes classes.

➥ Lorsque tous les objets appartiennent à une seule classe, l'incertitude est nulle.

• Étant données une position p dans l'arbre et c classes que l'on cherche à prédire, l'entropie associée à p est donnée par

$$H(p) = -\sum_{k=1}^{c} Pr(k|p) log_2(Pr(k|p))$$

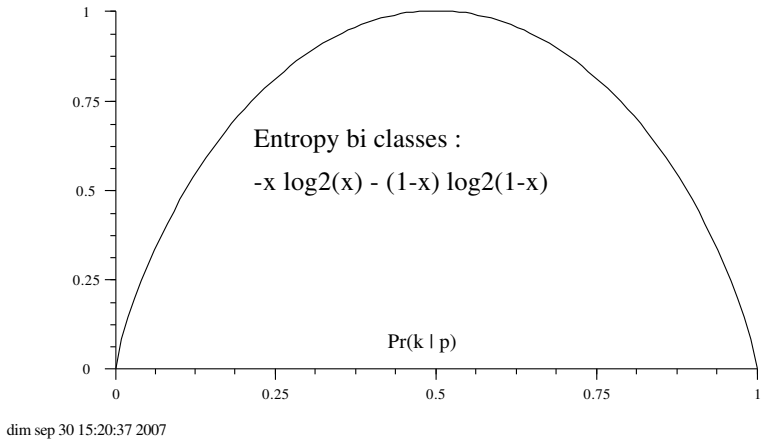


TABLE 1.2 – La fonction entropie et la courbe d'entropie pour 2 classes (log base2 , axe x=Pr(k | p))

1.12.3 Approches similaires

• Méthodes de choix d'attribut avec des résultats équivalents (cf. Chap. 6)

➥ Ex : **CART** (Classification and Regression Trees = arbres de décision **binaires**)

1.12.3.1 La méthode CART

• Étant données une position p dans l'arbre et c classes que l'on cherche à prédire, l'entropie associée à p est donnée par

$$Gini(p) = 1 - \sum_{k=1}^{c} Pr(k|p)^2$$

$$= 2 \sum_{k<k'} Pr(k|p) Pr(k'|p)$$

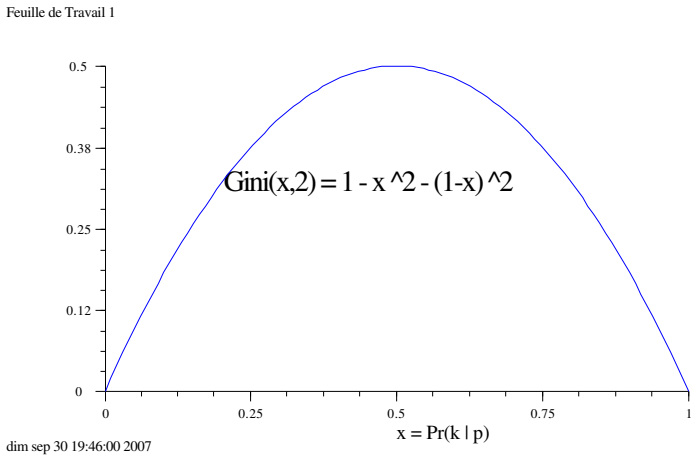


TABLE 1.3 – La fonction Gini et sa courbe pour 2 classes (axe x=Pr(k | p))

2 Modélisation Statistique

Quelques rappels sur les probabilités

• S : espace d'exemples, d'instances, d'observations, d'événements

• La somme des probabilités pour un événement = 1

• E.g., pour les lancés de dès : $S = \{(1,1),(1,2),....,(6,6)\}$

➥ Pour un dès non pipé (indépendant) : $P(1,1) = P(1,2) = ... = P(6,6) = \frac{1}{36}$

• L'ensemble de toutes ces probabilités = la **distribution de probabilités** .

2.1 Indépendance des variables

• Deux variables A et B sont indépendantes ssi $\boxed{P(A,B) = P(A)P(B)}$

pour toute valeur x de A et toute valeur y de B

➥ C-à-d. : **On peut calculer la jointe à partir des marginales**

• Cette'indépendance est une propriété de la distribution de probabilité sous-jacente

➥ On peut avoir, pour 2 distributions différentes P et P' :

$$P(A,B) = P(A)P(B) \qquad MAIS \qquad P'(A,B) \neq P'(A)P'(B)$$

2.2 Probabilité Conditionnelle & Indépendance : un exemple simple

• 3 variables aléatoires booléennes : **Se_Lever_a_8h , Prendre_Train, Etre_A_l_Heure**

• Supposons que :

$$P(Etre\_A\_Heure|Prendre\_Train, Se\_Lever\_a\_8h) = P(Etre\_A\_Heure|Prendre\_Train)$$

➤ Et $P(Etre\_A\_Heure|Se\_Lever\_a\_8h) = P(Etre\_A\_Heure)$

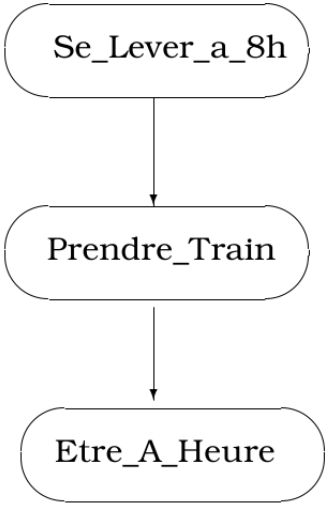
Le BN et l'indépendance :

• Même si *Se_Lever_a_8h* et *Etre_A_Heure* sont dépendantes,

elles sont ici rendues indépendantes étant donné **Prendre_Train**

➥ I.e. la connaissance sur l'une n'apporte rien (ne préjuge pas) sur la valeur de l'autre (= hyp. retenue).

➥ Se traduit par l'absence de causalité directe entre les deux (causalité indirecte possible)



2.2.1 Probabilité Conditionnelle et règles

- La probabilité conditionnelle peut représenter une relation **cause-effet** dans les 2 directions :

  ➥ De la cause à un effet (probable) :

  démarrage_voiture = false ← Batterie_a_plat

  P(démarrage_voiture = false | Batterie_a_plat = true) = 0.9

  ➥ De l'effet à une cause (probable) :

  Batterie_a_plat ←démarrage_voiture = false

  P(Batterie_a_plat = true | démarrage_voiture = false) = 0.7

2.3 Règles de produit et règle de Bayes

• **Règle de produit :** $P(A, B|C) = P(A|B, C)\ P(B|C)$

$$= P(B|A, C)\ P(A|C)$$

➥ Justification : avec $P(X, Y) = P(X|Y).P(Y)$ $\rightarrow$ poser $Y = (B, C)$ puis réappliquer

$$P(A, B|C) = \frac{P(A, B, C)}{P(C)} = \frac{P(A|B, C).P(B, C)}{P(C)} = P(A|B, C)\ P(B|C)$$

• **Règle de Bayes** : $P(A|B, C) = \dfrac{P(B|A, C)\ P(A|C)}{P(B|C)}$

➥ Justification : appliquer la 1e égalité de la règle produit puis la 2e.

• Plus général : la formule de Bayes dans un context (background) c :

$$\boxed{Pr[H|E, c] = \frac{Pr[E|H, c] \times Pr[H|c]}{Pr[E|c]}} \quad \rightarrow \quad \boxed{P(Modèle|Data) = \frac{P(Modèle).P(Data|Modèle)}{P(Data)}}$$

2.4 Exemple introductif : Station services

• Dans une station-service, on connaît les différentes probabilités d'avoir de $0..k$ clients dans un délai de 15 minutes (loi binomiale) :

x_i	0	1	2	3	4
$Pr[X = x_i]$	0.1	0.2	0.4	0.2	0.1

➙ On sait également que la probabilité qu'un client entré demande du Diesel = 0.4.

• Q1- **Quelle est** la *vraisemblance* (la loi conditionnelle) de Y pour $X = x_i$?

➥ i.e. la probabilité pour k demandes de diesel schant x_i entrés (en 15 min) ?

• Q2- **Quelle est** la loi du couple (X,Y), celle de Y ?

• Q3- **Combien** d'entrés (loi de X) sachant k demandes de Diesel ?... ../..

2.4.1 Solution

- On fixe $X = x_i$ dont chacun (entré) a une proba de 0.4 de demander du diesel.

- Le nombre de ces personnes est donné par la variable aléatoire binomiale $\beta(x_i, 0.4)$.

Q1 - La vraisemblance de k demandes de diesel si x_i clients sont entrés :

$$Pr[Y = k \; demandes \;\; Diesel \mid X = x_i] = C_{x_i}^k \;\; 0.4^k \;\; 0.6^{x_i - k} \quad si \; k \leq x_i \quad (et = 0 \; sinon)$$

➥ Par exemple, $Pr(Y = 1|X = 1) == 0.4$ Et $Pr(Y = 1|X = 2) = 0.48$

Q2 - La loi du couple (X, Y) sera :

$$Pr(Y = k, X = x_i) = Pr(X = x_i, Y = k) = Pr[Y = k|X = x_i] \times Pr(X = x_i)$$

Q3- Et enfin la loi de X (combien d'entrées) sachant k demandes de Diesel ?

$$Pr(X = x_i \mid Y = k) = \frac{Pr(X = x_i, Y = k)}{Pr(Y = k)} = \frac{Pr[Y = k|X = x_i] \times Pr(X = x_i)}{Pr(Y = k)}$$

☞ Par rapport à l'exemple en préembule, on s'appuie ici seulement sur la vraissemblance (les données), il nous manque la loi *a priori* pour faire une analyse complète.

2.5 Prédiction Bayésienne sur l'exemple Météo

Outlook	Temp.	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

TABLE 2.1 – BD exemple "météo"

- Deux hypothèses : les attributs sont d'importance égale,

  ➙ elles statistiquement indépendantes (vis à vis de la classe)

  ➥ Indépendance : les connaissances sur la valeur d'un attribut particulier ne disent rien sur la valeur d'un autre attribut (pour une classe connue)

2.5.1 La probabilité conditionnelle de Bayes

• L'hypothèse H et l'évidence E basée sur H : $\boxed{\Pr[\mathrm{H} \mid \mathrm{E}] = \frac{Pr[E|H] \times Pr[H]}{Pr[E]}}$

➥ **Hypothèse** (naïve) de Bayes (indépendance) : l'évidence E se décompose ici en ses composantes indépendantes p/r à la classe (les attributs de l'instance) :

$$\boxed{\Pr[\mathbf{H} \mid \mathrm{E}] = \frac{Pr[E|H] \times Pr[H]}{Pr[E]} = \frac{Pr[E_1|H] \times Pr[E_2|H] \times \cdots \times Pr[E_n|H] \times Pr[\mathbf{H}]}{Pr[E]}}$$

➥ Pour classer (e.g. play=yes) un nouvel exemple dépendant des 4 attributs :

$$\boxed{\Pr[\mathbf{yes} \mid \mathrm{E}] = \frac{Pr[Outlook|yes] \times Pr[Temp|yes] \times Pr[Hum|yes] \times Pr[Windy|yes] \times Pr[\mathbf{yes}]}{Pr[E]}}$$


On ne peut multiplier les probabilités que sous l'hypothèse de l'indépendance.

• Un exemple : météo

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

■ A new day:

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Likelihood of the two classes

For "yes" = 2/9 × 3/9 × 3/9 × 3/9 × 9/14 = 0.0053

For "no" = 3/5 × 1/5 × 4/5 × 3/5 × 5/14 = 0.0206

Conversion into a probability by normalization:

P("yes") = 0.0053 / (0.0053 + 0.0206) = 0.205

P("no") = 0.0206 / (0.0053 + 0.0206) = 0.795

FIGURE 2.1 – Probabilités conditionnelles pour les données "météo"

• Pour une nouvelle instance à classer, **l'évidence E** :

Outlook	Temperature	Humidity	Windy	Play
Sunny	Cool	High	True	? ?

⇒ Probabilité de "yes" pour la nouvelle instance :

$$\begin{aligned}\textbf{Pr[yes | E] = } & \textbf{Pr[Outlook = Sunny | yes]} \times \\ & \textbf{Pr[Temperature = Cool | yes]} \times \\ & \textbf{Pr[Humidity = High | yes]} \times \\ & \textbf{Pr[Windy = True | yes]} \times \frac{Pr[Yes]}{Pr[E]} \\ & = \frac{2/9 \times 3/9 \times 3/9 \times 3/9 \times 9/14}{Pr[E]}\end{aligned}$$

⇒ N.B. Ici, $Pr[E]$ (= $Pr[yes|E] + Pr[no|E]$) disparaîtra avec la normalisation.

2.5.2 *Remarques sur la méthode Bayesienne*

• **Avantages de Bayes** : Méthode simple, résultats intéressants,

➥ S'améliore si suppression d'attributs redondants → suppression des dépendances

➥ Techniques statistiques de calcul de la dépendance.

• **Inconvénients** : fonctionne mal si les valeurs d'un attribut particulier ne sont pas associées à toutes les valeurs de classe finale (dans la BD).

➥ E.g. si sunny toujours associé avec no

➔ Pr[yes | sunny]=0 qui multiplie ... = 0

➔ probabilité finale = 0 ➙ "sunny" a un droit de veto.

➥ Solution : ajustement (calcul des probabilités à partir des fréquences).

Ajustement et Correction du droit de veto : estimateur de **Laplace**

• Exemple : dans "météo", pour "yes", on a *sunny* 2/9 fois, *overcast* 4/9, *rainy* 3/9

⇒ On ajoute 1 à chaque numérateur puis 3 au dénominateur :

➔ On aura les valeurs ajustées équivalentes : 3/12, 5/12 et 4/12.

➥ Permet de changer la valeur d'un attribut de 0/x en 1/x' → évite le pb. !

⇒ Cette méthode est une technique standard appelée → **Estimateur de Laplace**

➔ On considère l'ajout d'une petite constante α (par défaut, $\alpha = 0$)

➔ Pour l'exemple (ici $\alpha = 3$) : $\frac{2+\alpha/3}{9+\alpha}$, $\frac{4+\alpha/3}{9+\alpha}$, $\frac{3+\alpha/3}{9+\alpha}$

⇒ Une grande valeur de α signale l'importance des poids p/r aux nouvelles évidences,

⇒ un petit α dénote une moindre influence.

• Généralisation : au lieu de $\frac{1}{3}$ pour chaque, prendre p_i avec $p_1 + p_2 + p_3 = 1$.

2.5.3 Valeurs Numériques dans Bayes

• Hypothèse : elles ont une distribution **normale** ou **Gaussienne** de probabilités

• La *fonction de densité de probabilité* pour une distribution normale avec :

μ la moyenne : $\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$ σ l'écart type : $\sigma^2 = \frac{\sum_{i=1}^{n} (x_i - \mu)^2}{\mathbf{n-1}}$

➔ la fonction de densité (loi normale/gaussienne) : $\boxed{\mathbf{f(x)} = \frac{\mathbf{1}}{\sqrt{\mathbf{2}\pi}\,\sigma}\, \mathbf{e}^{-\frac{(\mathbf{x}-\mu)^{\mathbf{2}}}{\mathbf{2}\sigma^{\mathbf{2}}}}}$

⇨ Le "-1" sur "n" concerne **le degré de liberté** dans les instances

⇨ Les valeurs manquantes (dans une instance) n'interviennent pas dans μ et σ

• Pour calculer la probabilité (e.g. pour "yes") pour une nouvelle instance , on utilisera la fonction de densité pour les numériques et les fréquences pour les nominaux

• **La fonction de densité :** (pour x centrée)

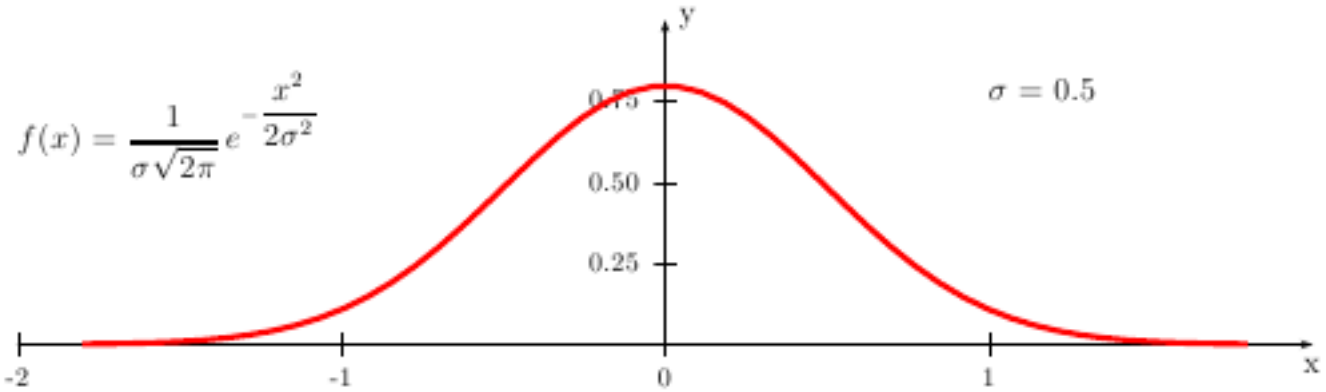


FIGURE 2.2 – La fonction de densité pour une variable centrée ($\mu = 0$) avec $\sigma = 0.5$

Cette intégrale vaut 1.

• Exemple avec la "météo" (avec attributs numériques et nominaux) :

1	outlook	temperatur	humidity	windy	play
2					
3	sunny	85	85	FALSE	no
4	sunny	80	90	TRUE	no
5	overcast	83	86	FALSE	yes
6	rainy	70	96	FALSE	yes
7	rainy	68	80	FALSE	yes
8	rainy	65	70	TRUE	no
9	overcast	64	65	TRUE	yes

⇒ Application : pour une nouvelle instance à classer :

Outlook	Temperature	Humidity	Windy	Play
Sunny	66	90	True	? ?

Sachant par ailleurs $\mu = 73$ et $\sigma = 6.2$

../..

• proba de *play="yes"* si la température = 66 ($\rightarrow x = 66$, $\mu = 73$ et $\sigma = 6.2$)

• La table des calculs ("météo") :

Outlook			Temperature			Humidity			Windy		Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3		83	85		86	85	False	6	2	9	5
Overcast	4	0		70	80		96	90	True	3	3		
Rainy	3	2		68	65		80	70					
				...	...		...	...					
Sunny	2/9	3/5	*mean*	73	74.6	*mean*	79.1	86.2	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	*std dev*	6.2	7.9	*std dev*	10.2	9.7	True	3/9	3/5		
Rainy	3/9	2/5											

FIGURE 2.3 – Probabilités pour les données "météo" (numériques et nominaux)

➥ La fonction de densité pour la "Température" (play="yes" si Temp.=66) :

$$\mathbf{f(Temperature = 66|yes)} = \frac{\mathbf{1}}{\mathbf{\sqrt{2\pi}\ .\ 6.2}}\,\mathbf{e}^{-\frac{\mathbf{(66-73)^2}}{\mathbf{2\ .\ 6.2^2}}} = \mathbf{0.0340}$$

$\Rightarrow$ De manière analogue, la densité de proba de play="yes" si Humidité=90 :

$$\mathbf{f(Humidity = 90|yes) = 0.0221}$$

• Donc, pour la nouvelle instance :

outlook	temperature	humidity	windy	**play**
sunny	66	90	true	?

Vraisemblance de "yes" = $2/9 \times 0.0340 \times 0.0221 \times 3/9 \times 9/14 = 0.000036$

Vraisemblance de "no" = $3/5 \times 0.0291 \times 0.0380 \times 3/5 \times 5/14 = 0.000136$

$\Rightarrow$ Et les probabilités :

$$\text{Pr("yes")} = \frac{0.000036}{0.000036 + 0.000136} = 20.9\%$$

$$\text{Pr("no")} = \frac{0.000136}{0.000036 + 0.000136} = 79.1\%$$

➥ Valeurs très proches des calculs précédents :

➥ la température 66 est proche de "cool" et l'humidité 90 proche de "high".

2.6 Le réseau Bayesien de l'exemple météo

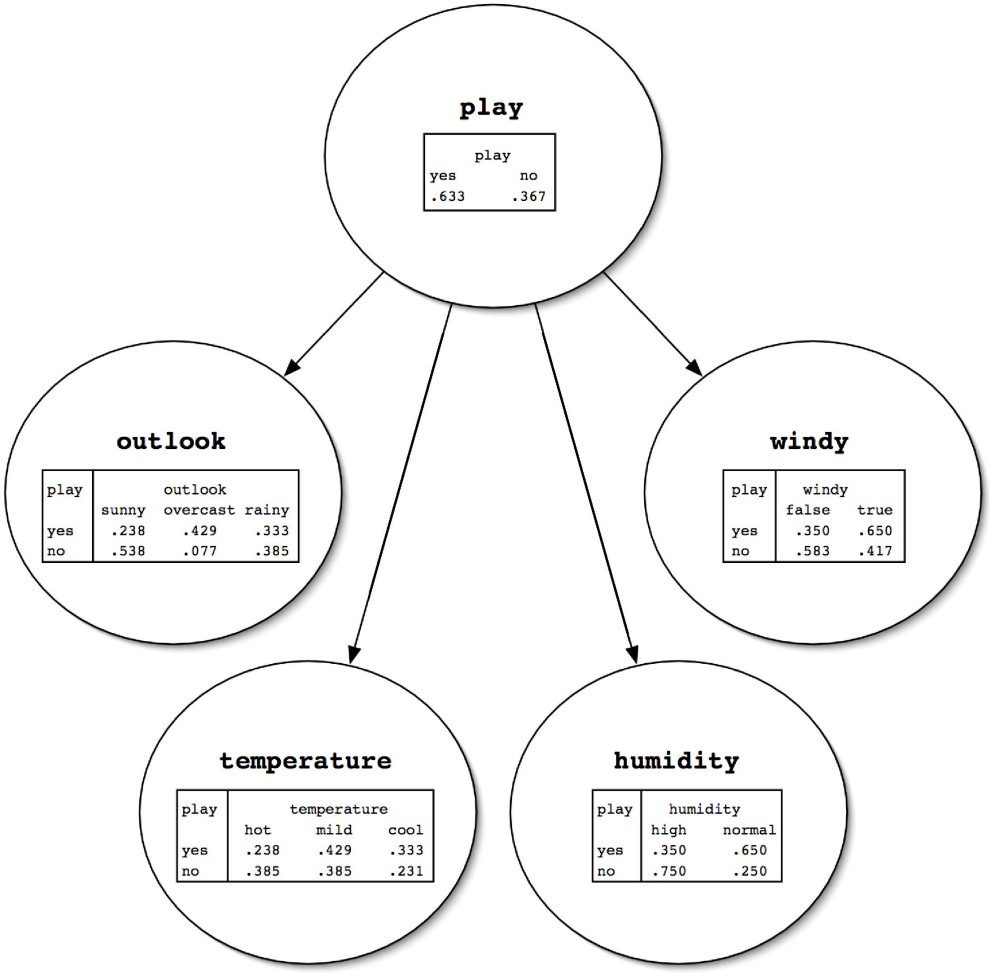


FIGURE 2.4 – BN pour les données météo avec un seul parent

Et un BN plus complexe :

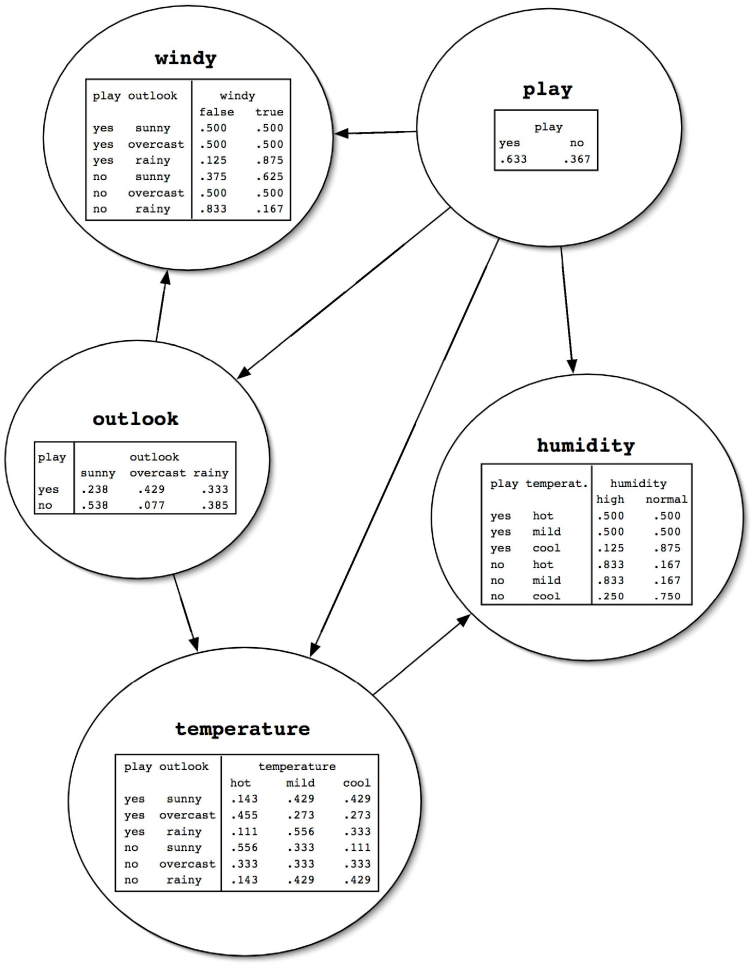


FIGURE 2.5 – BN pour les données météo avec plus d'un parent

Exemple de prédiction à l'aide du BN (météo) :

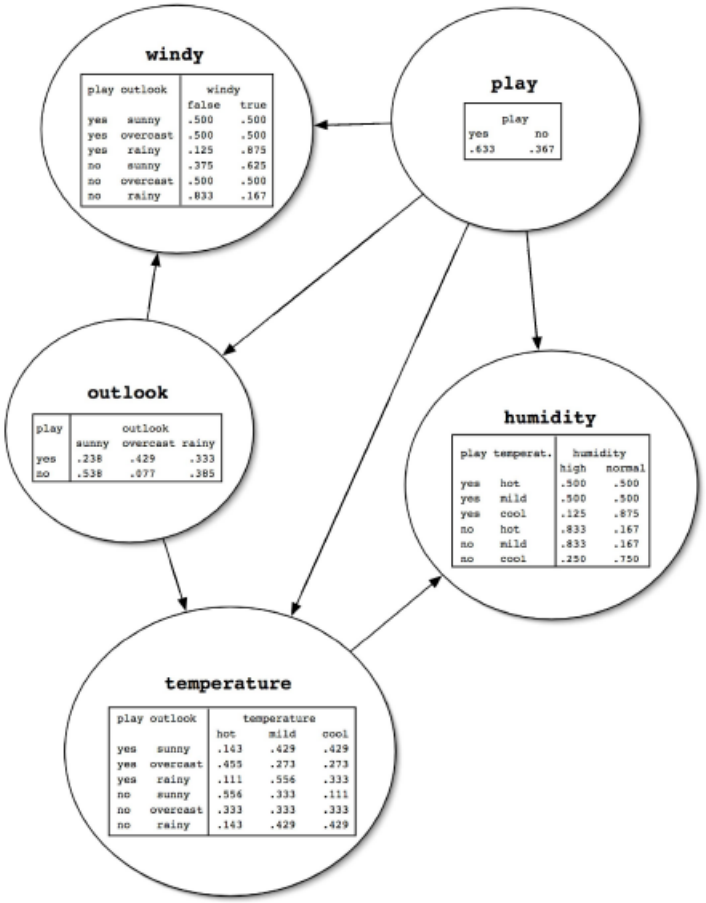
$Pr[play = no \mid E] =?$ pour la nouvelle instance E :

E : $outlook = rainy, temp = cool, humidity = hight, windy = true$

- $Pr[play = no] = 0.367$ (proba a priori)
- $Pr[outlook = rainy | play = no] = 0.385$
- $Pr[Temp = cool | play = no, outlook = rainy] = 0.429$
- $Pr[humidity = high | play = no, Temp = cool] = 0.250$
- $Pr[windy = true | play = no, outlook = rainy] = 0.167$

➥ $= 0.367 * 0.385 * 0.429 * 0.250 * 0.167 = 0.0025$

➥ A normaliser



2.7 Prédiction numérique : exemple des Performances

• Performances relatives des PC selon certains attributs.

	Cycle time (ns)	Main memory (Kb)		Cache (Kb)	Channels		Performance
	MYCT	MMIN	MMAX	CACH	CHMIN	CHMAX	PRP
1	125	256	6000	256	16	128	198
2	29	8000	32000	32	8	32	269
...							
208	480	512	8000	32	0	0	67
209	480	1000	4000	0	0	0	45

TABLE 2.2 – Données Performances CPU

• 209 configurations différentes (une ligne = une configuration)

⇒ Les attributs et la sortie sont tous **numériques** (vs. cas général : mixtes).

• Solution avec la prédiction d'une valeur **continue** (*performances*) :

⇒ Utilisation d'une somme linéaire pondérée des attributs (avec des poids)

⇒ La sortie (*performances*) = une fonction de (*Bus* x *Ram* x *Cache* x *Canaux*)

PRP = -55.9 + 0.0489 MYCY + 0.0153 MMIN + 0.0056 MMAX + 0.6410 CACH - 0.2700 CHMIN + 1.480 CHMAX.

• La **régression** = le processus de détermination des poids dans l'équation.

⇒ N.B. : une variante de la régression produit une équation non linéaire.

2.8 L'apprentissage : Supervisé ou non supervisé

• **L'apprentissage** : méthode pratique de définition de **concepts**.

⇒ L'humain (enfant) utilise des instances de concepts pour se représenter le monde :

➥ animaux, plantes, homme, femme, jouet, ...

➥ On apprend des instances particulières → On choisit des attributs

➥ On forme des *modèles de classification*

➥ On utilise ces modèles pour identifier des objets similaires.

La construction du modèle est inductive, son application est déductive.

• Création de modèles de classification à partir de d'**instances positifs** et **négatifs**.

• L'apprentissage : supervisé ou non supervisé (recherche de groupes s)

2.9 Clustering : un apprentissage Non Supervisé

• Construction d'un modèle sans classes prédéfines.

• Groupage des données sur la base des **similarités**

• A l'aide de techniques d'évaluation, on définit/interprète le sens des classes formées.

ID Client	Type Cpte	Cpte Bourse	Méth. Transac°	Op°/mois	Sexe	Age	Sport favori	Rev./an
1005	joint	non	en ligne	12.5	F	30-39	Tennis	40-59k
1013	ép-enf	non	courtier	0.5	F	50-59	Ski	80-99k
1245	joint	non	en ligne	3.6	M	20-29	Golf	20-39k
2110	indiv	oui	courtier	22.3	M	30-39	Pêche	40-59k
1001	indiv	oui	en ligne	5.0	M	40-49	Golf	60-79k

TABLE 2.3 – Données de l'Investisseur de Stock ACME

• Exemple : table 2.3 sur 5 clients :

• Pour un compte de type "Bourse", la banque prête de l'argent liquide au client.

• On veut trouver des "motifs" dans ces données.

• **On se pose les questions suivantes** ($\rightarrow$ pub / clients actuels) :

1. *Profil général d'un investisseur (client) en ligne ?*

   $\Rightarrow$ *Quelles caractéristiques ? Comment les distinguer d'un courtier ?*

2. *Peut-on savoir si un client sans "compte bourse" est susceptible d'en ouvrir un ?*

3. *Modèle de prédiction du nombre moyen de transactions (stocks achetés) / mois pour un nouveau client ?*

4. *Quelles différences entre les clients hommes et femmes ?, etc...*

• **Choix 1 :** Méthode **d'apprentissage supervisé** à base d'un des attributs :

$\Rightarrow$ Pour la question 1, la classe (attribut de sortie) = la méthode de transaction,

$\Rightarrow$ Pour la question 2, c'est le compte bourse,

$\Rightarrow$ Pour 3, Op/mois et pour la question 4, le "sexe".

• **Choix 2 : Choix d'un apprentissage non supervisé :**

⇒ On appliquera une méthode non supervisée si l'on veut savoir :

1- Quelles similitudes d'attributs peut grouper les clients de cette entreprise ?

2- Quelles différences d'attributs peuvent segmenter ces données ?

➥ Création de **clusters**.

• Certaines méthodes non supervisées réclament une *estimation du nombre total de clusters ;*

d'autres essaient de trouver ce nombre.

➥ Dans les 2 cas, on scinde les données en groupes (clusters).

• Supposons qu'un algorithme de classification non supervisé a produit 3 clusters.

⇒ Les règles représentatives de ces clusters : ../..

Si Compte-bourse=oui & Age=20-29 & Revenus-annuels=40-59k

Alors Cluster=1 *{justesse=0.80, couverture=0.50}* *(1)*

Si Type-compte=épargne-enfant & Sport-favori=Ski & Revenus-annuels=80-99k

Alors Cluster=2 *{justesse=0.95, couverture=0.35}* *(2)*

Si Type-compte=joint & Nb-transactions > 5 & Méthode-Transaction=en-ligne

Alors Cluster=3 *{justesse=0.82, couverture=0.65}* *(3)*

→ Couverture = **support**, justesse = **confiance**

- Cluster 1 : logique : clients plus jeunes, revenus raisonnables, approche moins conservative.

- Cluster 3 : ce n'est pas une découverte ! Mais

- Cluster 2 : peut être intéressant.

La compagnie ACME (American Company Making Everything) peut consacrer une partie de son budget **pub dans les magazines de Ski** avec promo sur les comptes épargne-enfant.

2.10 Extraction de règles d'association

• Comme les Règles de Classification :

▶ La même manière d'induction de règles

▶ Avec toute expression d'attributs possible en partie droite (RHS)

▶ Une seule règle d'association peut prédire les valeurs de plusieurs attributs.

• La procédure d'induction de règles pour **toute combinaison possible** d'attributs,

➥ avec toute **combinaison possible de valeurs**

Mais cette approche est tout à fait infaisable (complexité) !

➔ Un nombre important de Règle d'Association obtenues

➥ Élaguer sur la base de deux mesures : **support** et **confiance**

▶ **Support** : nombre d'instances correctement couvertes par une règle

• Les règles porteront sur des sous-ensemble d'attributs

➔ Pour le moment, on ignore la distinction entre RHS et LHS

➥ on cherche des combinaisons d'attribut-valeur avec un support min prédéfini.

▶ **item** : une paire attribut-valeur

▶ **itemset** : tous les items figurant dans une règle

➔ Terminologie de l'analyse du panier/caddie (*market basket analyse*) :

→ Les items sont des articles dans le caddie

→ Le manager du supermarché cherche des associations dans les achats.

⇒ **But** : Trouver les itemsets avec un support maxi → **itemsets fréquents**.

➔ Générer les règles qui s'appliquent aux plus grands nombres d'instances

2.10.1 Exemple

• Soit $support \geq 2$

BD trans. ➙ **Ens.** C_1

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Exemple (suite) : avec support ≥ 2

BD trans. → **Ens.** C_1 → **Ens.** L_1

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Itemset	Supp.
{2}	2
{3}	3
{4}	3
{5}	4

BD trans. ➙ **Ens.** C_1 ➙ **Ens.** L_1 **Ens.** C_2

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Itemset	Supp.
{2}	2
{3}	3
{4}	3
{5}	4

Itemset	Supp.
{2,3}	2
{2, 4}	0
{2, 5}	2
{3,4}	1
{3, 5}	3
{4,5}	2

➥ N.B. : C_2 construit en croisant L_1 avec lui même.

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Itemset	Supp.
{2}	2
{3}	3
{4}	3
{5}	4

Itemset	Supp.
{2,3}	2
{2, 4}	0
{2, 5}	2
{3,4}	1
{3, 5}	3
{4,5}	2

$\longleftarrow$ Ens. L_2

Itemset	Supp.
{2,3}	2
{2, 5}	2
{3, 5}	3
{4,5}	2

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Itemset	Supp.
{2}	2
{3}	3
{4}	3
{5}	4

Itemset	Supp.
{2,3}	2
{2, 4}	0
{2, 5}	2
{3,4}	1
{3, 5}	3
{4,5}	2

$\longleftarrow$ Ens. L_2

$\longleftarrow$ Ens. C_3

Itemset	Supp.
{2,3}	2
{2, 5}	2
{3, 5}	3
{4,5}	2

Itemset	Supp.
{2,3, 5}	2

Id Trans.	Les items
50	4, 6, 7
100	4, 5
120	2, 3, 5
150	3, 4, 5
200	1, 2, 3, 5

Itemset	Supp.
{1}	1
{2}	2
{3}	3
{4}	3
{5}	4
{6}	1
{7}	1

Itemset	Supp.
{2}	2
{3}	3
{4}	3
{5}	4

Itemset	Supp.
{2,3}	2
{2, 4}	0
{2, 5}	2
{3,4}	1
{3, 5}	3
{4,5}	2

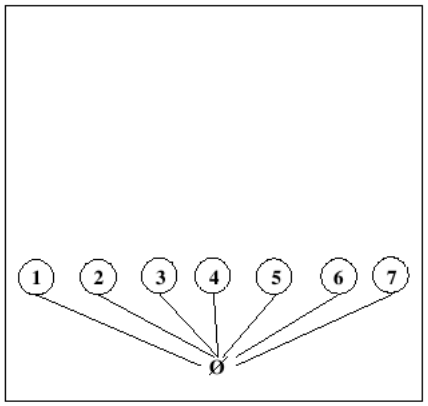
← Ens. L_2

← Ens. $C_3 = Ens.$ $\mathbf{L_3}$

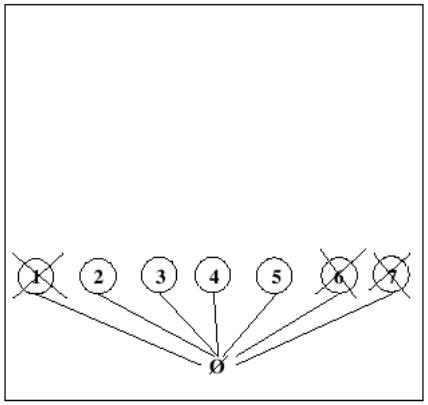
Itemset	Supp.
{2,3}	2
{2, 5}	2
{3, 5}	3
{4,5}	2

Itemset	Supp.
{2,3, 5}	2

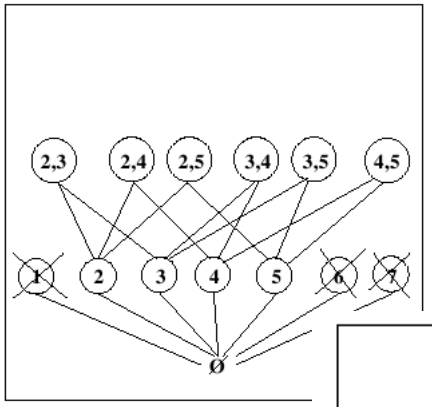
2.10.2 *Itemsets : visions par Treillis*

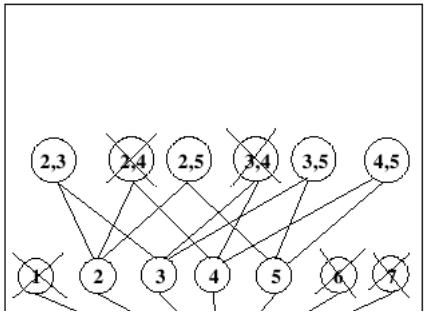


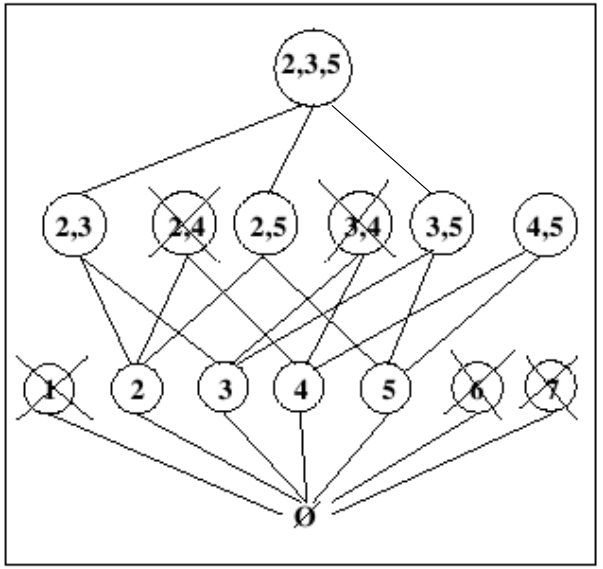
$C_1 = \{1, 2, 3, 4, 5, 6, 7\}$

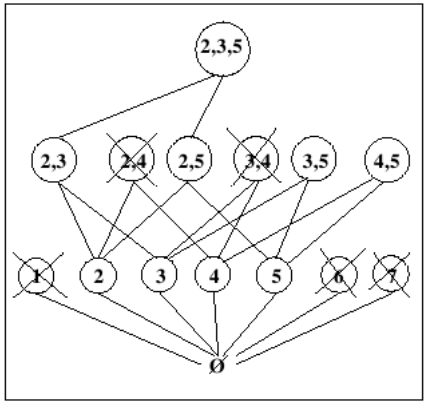


$L_1 = \{2, 3, 4, 5\}$









→

$$L = L1 \cup L2 \cup L3 = \{\{2\}, \{3\}, \{4\}, \{5\}, \{2, 3\}, \{2, 5\}, \{3, 5\}, \{4, 5\}, \{2, 3, 5\}\}$$

2.10.3 *Itemsets l'exemple météo*

Outlook	**Temp.**	**Humidity**	**Windy**	**Play**
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

TABLE 2.4 – Rappel BD exemple "météo"

2.10.4 *Item sets (météo)*

	UN-Itemsets	DEUX-Itemsets	TROIS-Itemsets	QUATRE-Itemsets
1	Outlook=Sunny (5)	Outlook=Sunny Temperature = Mild (2)	Outlook=Sunny Temperature = Hot Humidity = High (2)	Outlook=Sunny Temperature = Hot Humidity = High Play = no (2)
2	Outlook=Overcast (4)	Outlook=Sunny Temperature = Hot (2)	Outlook=Sunny Temperature = Hot Play=no (2)	Outlook=Sunny Windy = False Humidity = High Play = no (2)
4	Temperature = Cool (4)	Outlook=Sunny Humidity = High (3)	Outlook=Sunny Humidity = High Windy=false (2)	Outlook=Rainy Temperature = Mild Windy=false (2) Play = yes (2)
7	Humidity=Normale (7)	Outlook=Sunny Play = yes (2)	Outlook=Overcast Temperature = Hot Play = no (2)	
12	Play=no (5)	Outlook=Overcast Windy=true (2)	Outlook=Overcast Windy = false Play = yes (2)	
...		...	...	
47		Windy=false Play=no (2)		

TABLE 2.5 – Les Item-sets pour l'exemple "météo" avec un $support \geq 2$

2.10.5 Mesures et contraintes : Support, Fréquence et confiance

- Base de données D, Itemset I,
- Support d'un itemset I : ensemble des transactions de D qui contiennent I.
- Fréquence d'un itemset I : $Freq(I, D) = \frac{|Support(I, D)|}{|D|}$
- Confiance d'une règle d'association (pour L = k-fréquent itemsets) :
  - ➥ Pour un itemset $I \in L$ et le sous-itemset $X \subset I, X \neq \oslash$:

$$Conf(X \Rightarrow I \setminus X) = \frac{Freq(I, D)}{Freq(X, D)}$$

  - ➥ Si $Conf(X \Rightarrow I \setminus X) \geq min\_conf$, la règle est retenue.

- Recherche de règles de la forme {Outlook=Sunny, Temp=Hot} ➙ {Play=Yes} à 80 %

2.10.6 Règles d'association pour météo (A PRIORI)

• On transforme chaque item-set en (un ensemble éventuellement vide de) règles avec la confiance min spécifiée.

⇒ Par exemple, le Trois-itemset **Humidity=normal, windy=false, play=yes (4)**

donne 7 ($2^N - 1$) règles potentielles :

if humidité=normale and windy=false then play=yes	(4/4)
if humidité=normale and play=yes then windy=false	(4/6)
if windy=false and play=yes then humidité=normale	(4/6)
if humidité=normale then windy=false and play=yes	(4/7)
if windy=false then humidité=normale and play=yes	(4/8)
if play=yes then humidité=normale and windy=false	(4/9)
if True then humidité=normale and windy=false and play=yes	(4/14)

2.11 Apprentissage à base d'instances (IBL)

• Stockage des instances

• Fonction de **distance** pour déterminer la classe d'une nouvelle instance

⇒ La fonction de *distance définit ce qui est appris*

⇒ La fonction de distance simple si les attributs numériques.

⇒ La plupart des schémas *IBL* utilisent une **distance Euclidienne**.

⇒ La distance entre les instances avec des valeurs de k attributs :

1^e instance $a^{(1)} : a_1^{(1)}, a_2^{(1)}, ...a_k^{(1)}$ $\quad$ 2^e instance $a^{(2)} : a_1^{(2)}, a_2^{(2)}, ...a_k^{(2)}$

La **distance Euclidienne** = $\boxed{\sqrt{\left(a_1^{(1)} - a_1^{(2)}\right)^2 + \left(a_2^{(1)} - a_2^{(2)}\right)^2 + \ldots + \left(a_k^{(1)} - a_k^{(2)}\right)^2}}$

➥ Quand on compare les distances, la racine carré est inutile

⇒ Une alternative à la distance Euclidienne

➥ La métrique **"Manhattan"** ou "city-block" :

➔ Simple addition des différences (pas au carré)

$$\boxed{\left(a_1^{(1)} - a_1^{(2)}\right) + \left(a_2^{(1)} - a_2^{(2)}\right) + \ldots + \left(a_k^{(1)} - a_k^{(2)}\right)}$$

➥ Augmentent l'influence des grandes différences au détriment des petites.

⇒ En générale, la **distance Euclidienne** représente **un bon compromis.**

• **Choix de la métrique** : qu'est-ce ?

➥ Le sens de la distance qui sépare 2 instances,

➥ Que veut dire le double de cette distance, etc. ?

➥ Différents attributs sont mesurés à différentes échelles.

➔ Si distance Euclidienne → l'effet de certains attributs peut être complètement inhibé par d'autres qui ont une échelle de mesure plus large.

➔ Solution : **normaliser** tous les attributs (ramenés entre 0 et 1) en calculant :

$$a_i = \frac{v_i - \ min\ v}{max\ v - \ min\ v}$$

avec v_i et la vraie valeur de l'attribut i, min et max de l'ensemble des instances.

Remarque sur le plus proche voisin (apprentissage à base d'instances) :

• L'apprentissage à base d'instances avec le **1-NN** (*Un-plus proche voisin*)

➥ (cf. distance ci-dessus) : calcul simple et juste mais lente.

➔ On doit examiner **tout l'ensemble d'apprentissage** pour une prédiction

➔ Tous les attributs sont supposés d'égale importance (comme pour Bayes)

➔ Une solution : sélection d'attribut prépondérant ou affectation de poids

• Pour les données erronées (bruités) :

➔ Enlever les instances "parasitées" = choix de "bons" exemples (difficile !)

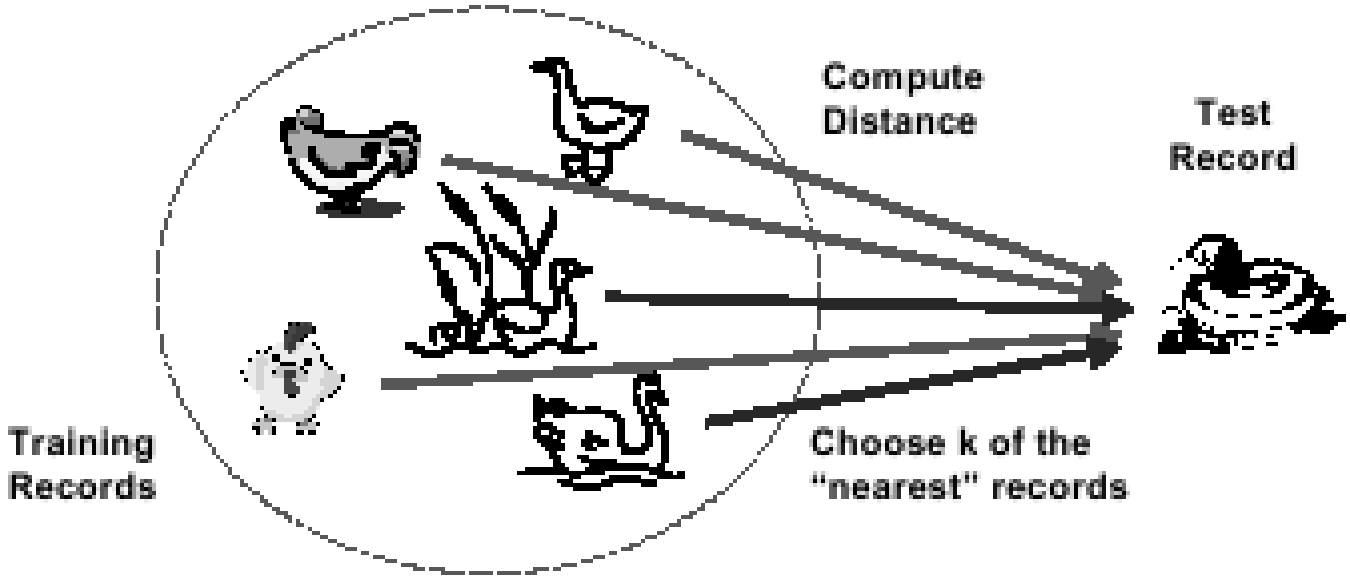
➔ Utiliser un vote majoritaire sur les K-plus proches voisins (e.g. K=5)

• **K-plus proches voisins** utilisée depuis 1950s . . . (et Stat et en PR)

Une solution : K-plus proches voisins

⇒ On prend la classe la plus commune (majoritaire) des K voisins.

➥ Empêche le mauvais classement dû à une instance atypique de la table.



2.11.1 Un exemple : la cueillette de champignons

• Un groupe d'amis se propose d'aller ramasser des champignons le week-end prochain.

Mais les champignons sont capricieux et seules de bonnes conditions météorologiques dans la journée qui précède favorisent leur croissance.

Aussi, ils préfèrent analyser leur cueillette des années précédentes pour savoir si le ramassage permettra ou non une fricassée au dîner.

Ils disposent de l'agenda suivant :

		Temps	Humidité	Vent	Bonne cueillette
E1	WE 1, année-1	nuageux	haute	oui	oui
E2	WE 2, année-1	nuageux	basse	non	non
E3	WE 3, année-1	nuageux	basse	oui	non
E4	WE 1, année-2	soleil	haute	oui	oui
E5	WE 2, année-2	pluvieux	basse	oui	non
E6	WE 3, année-2	nuageux	haute	non	oui
E7	WE 4, année-2	soleil	basse	non	non

TABLE 2.6 – Anals champignons

Ce week-end : **ensoleillé, humide et sans vent**. ➙ Ira-t-on cueillir des champignons ?

Processus de décision :

1 : Choisir une distance sur chacun des attributs.

2 : Utiliser la distance Euclidienne pour calculer les voisins de notre dimanche.

3 : Varier K : ira-t-on cueillir des champignons si k = 3, k = 4, k = 5 ?

Réponse :

- Si k = 3, on retient E4, E6 et E7, donc la cueillette sera bonne.

- Si k = 4, on retient E4, E6, E7 et E1 ➙ pas de problème, la cueillette sera bonne.

➥ Mais on aurait pu retenir E2 plutôt que E1 (même distance à E8).

➥ Si l'on retient E4, E6, E7 et E2, pas de conclusion (car deux oui et deux non).

- Si k = 5, on retient E4, E6, E7, E1 et E2, donc la cueillette sera bonne.

Remarques :

- Pour mieux conclure, mieux vaut prendre un nombre de voisins dont *le modulo du nombre de classes* n'est pas égal à zéro.

2.12 Cas et Domaines réels d'application

2.12.1 Décision nécessitant un jugement : cas de prêt bancaire

• La décision de la banque en 2 temps :

$\Rightarrow$ 1) méthodes statistiques $\rightarrow$ accept/refus franc.

$\Rightarrow$ 2) puis le jugement (humain) pour le reste.

• Méthodes statistiques $\rightarrow$ une note à la demande de prêt (90% des cas) :

$\Rightarrow$ Si la note $>$ *bsup* ($<$ *binf*) $\rightarrow$ accord (refus) $\rightarrow$ Décision directe ;

$\Rightarrow$ Sinon (le reste) : on prend une décision.

• Données historiques importantes : $\frac{1}{2}$ des clients marginaux (ni acceptés ni refusés directement) approuvés ont fait défaut.

$\Rightarrow$ On peut ne pas leur prêter.

$\Rightarrow$ Mais (constatation) : ce sont des clients intéressants

➥ ce sont des clients actifs de l'institution de crédits (leur finance volatile $\rightarrow$ ils empruntent souvent, finissent par payer – même avec retard – ...).

➥ important : bien déterminer la tendance de la situation d'un client marginal

• **Echelle typique d'Apprentissage Automatique dans ce cas réel** :

$\Rightarrow$ 1000 exemples d'apprentissage de cas marginaux + s'ils ont "finalement" payé

$\Rightarrow$ 20 attributs étudiés : age, nbr. années dans la même boîte, nbr. années à la même adresse, années avec la banque, autres cartes de crédits, ...

• Un algorithme simple de classification $\rightarrow$ 2/3 des cas MARGINAUX correctement prédits (testé avec un ensemble de cas aléatoires).

$\Rightarrow$ Aide à la décision des banques + l'explication (le **pourquoi**) de la décision.

2.12.2 Détection/nettoyage (screening) d'images

Satellites → images → recherche de tâches de pétrole → catastrophe écologique.

• Les satellites radars → surveillance des côtes jour et nuit (par tout temps).

⇒ Ces tâches changent d'aspect et de taille avec le temps / les conditions de la mer.

• Mais il y a des tâches (semblables) dûes au climat local/vent très fort.

➥ Il faut des très spécialistes entrâinés pour diagnostiquer

• Un système de détection a été développé.

• Utilisation par gouvernements/individus/géographes/...

• Système d'Apprentissage Automatique **supervisé** entraîné sur des exemples positifs / négatifs (fausses alertes).

• Pas de classification mais un schéma d'apprentissage directement déployé.

⇒ Input = un ensemble de pixels.

⇒ Output = un ensemble (bien + petit) de pixels de **trace de pétrole entourée**.

➥ Normalisation des images + régions (noires) suspectes détectées.

• Plusieurs douzaines d'attributs : taille, forme, zone, intensité, contours, proximité d'autres régions, historique des catastrophes de la région,... → vecteurs d'attributs.

• Cette étude soulève plusieurs problèmes intéressants :

⇒ Rareté : manque de données (incomplètes) → classification manuelle difficile.

⇒ Justesse : toute tâche noire n'est pas du pétrole → coût d'erreur

➥ Surveiller la détection, donner des alertes de niveaux différents.

⇒ Les images (groupées) en régions avec des backgrounds complètement différents.

2.12.3 Prévision de charge

• Prévision de Charge électrique (demande à venir) aussi loin que possible.

• Prévoir le max/min de la charge par heure/jour/mois/saison/année

➥ Electricité non stockable → Economies sur la réserve opérationnelle,

➥ planification de la maintenance, gestion de fuel...

• Comment a-t-on fait auparavant ? → un modèle sophistiqué (à la main).

• Création d'un outil de prévision de charge (load forecasting) automatique pour les prévisions de charges par heure avec 2 jours en avance.

⇒ 3 composantes : charge de base de l'année, périodicité sur l'année, effet des vacances.

• 1e étape : récupération de données sur les 15 dernières années

Charge de base (normalisée) : les données de chaque année passée standardisées par ***(charge heure - moyenne de charge de cette année-là) / écart type sur l'année***

Périodicité : 3 fréquences

- *Jour* : matin (min), midi, soir (max).

- *Semaine* : consommation en baisse pendant les WE

- *Saison* : hausse de la demande en hiver (chauffage), l'été (clim) → cycle.

Vacances :

- *Importantes* : e.g. Noël, Jour de l'an, etc. avec trop de variations (la moyenne d'heure sur les 15 dernières années a été retenue)

- *Moins importantes* : vacances scolaires avec un décalage p/r normal.

⇒ But : prévision par jour (vacances apart), climat supposé normal toute l'année

• 2e étape (DM) : pris en compte du changement du temps

⇒ Trouver dans l'historique la veille la plus proche (distance pondérée) ;

➥ Pour éviter les erreurs importantes, utiliser la moyenne de 8 jours semblables.

⇒ + BD de température / humidité / vitesse de vent / couverture nuageux sur les centres climatiques et sur les 15 dernières années (par heure)

⇒ + la différence entre la prédiction par le modèle statique et la vraie charge.

⇒ Une régression linéaire pour déterminer les effets relatifs des paramètres sur la charge.


Le système créé donne les mêmes résultats qu'un expert entraîné mais bien plus rapidement (quelques secondes vs. des heures) pour générer une prévision.

2.12.4 Diagnostic et prévision de pannes

• Une autre application principale des systèmes experts et de DM.

• Prévention de pannes et maintenance dans l'industrie sur les machines électromécaniques : → Importace des coûts et le fonctionnement.

• Problèmes classiques : pièces mal alignés, perte mécaniques, comportement erroné, pompes mal balancées, ...

⇒ Habituellement : techniciens expérimentés inspectent les machines et (pré)voient les pannes

⇒ Mesure des vibrations + analyse par la transformée de Fourier.

⇒ Tâche et conditions difficiles (noisy informations)

→ il faut un expert pour diagnostiquer

→ il faut répéter tout pour chaque machine.

• Dans ce domaine :

- *Systèmes Experts* : les règles de production fonctionnent bien

- *Apprentissage Automatique* quand la création de règles de production est difficile.

• Un système ML proposé :

✓ 600 points de défaut

✓ chacun avec un ensemble de mesures

✓ + diagnostic de l'expert (20 ans d'expérience dans le domaine).

☞ Le but du système produit : déterminer le type d'un défaut avéré (→ diagnostic).

⇒ 1/2 des cas non satisfaisants par les méthodes classiques.

⇒ 1/2 restante → données pour l'Apprentissage Automatique .

• L'expert a ajouté des connaissances (fonctions sur les attributs)

• Un algorithme d'induction → un ensemble de règles (complexes) de production.

• Les règles acceptées par l'expert + ses connaissances en mécanique.

⇒ 1/3 des règles produites = ce qu'il utilisait

➥ Il les a utilisé pour en trouver d'autres et clarifier son expertise.


Les règles générées ont été "meilleures" que celles trouvées manuellement

➥ Elles ont été insérées dans un système d'expert .

• Utilisation dans le domaine de la chimie industrielle (processus très complexe).

2.12.5 Domaine de Marketing & Ventes

• Un domaine très actif de Extraction de Connaissances , beaucoup de succès.

• Volume très important de données.

➥ On a réalisé seulement récemment l'importance de ces données.

• L'objectif recherché : **la prédiction**,

➥ Important : la découverte de la structure pour prendre des décisions.

• Les applications bancaires :

- Fidélisation des clients (par traitement/promotion particulier).

- Les banques et les demandes de crédits.

- comportement/insatisfaction de clients → changement de banque.

- changement dans la vie/ville, ... → peut entraîner un changement de banque.

• Exemples d'éléments intéressant les banques :

- insatisfaction des clients qui font leur opérations par téléphone avec un service téléphonique lent.

- des clients qui retirent peu de cash sur leur carte sauf vers noël ou pour les frais de fin d'année.

- des clients de retour des vacances → découvert !

- des clients que l'on peut charger !

• Analogues aux compagnies de téléphonie cellulaire qui détectent de changement de téléphone ou envie de nouveaux services.

• Pour les banques : une opération générale de promo coûte très cher

➥ Détecter des motifs dans les comportements des clients et cibler les opérations pour maximiser les résultats (bénéfices) à l'aide des outils Extraction de Connaissances

Market Basket analyse :

→ techniques pour trouver des motifs/groupes de produits qui vont ensemble.

• Ces données sont souvent la seule source d'info sur les clients.

• Une analyse classique ne fait pas sortir la relation *bière* → *chips*.

• On a trouvé que des clients achetaient (le jeudi) des déguisements, figures autocollantes, couches bébé + bière.

➥ Bizarre au départ, mais des jeunes couples commencent à préparer le WE ! !.

• Informations utiles : rangement des rayons, limiter des promos sur un seul des articles du groupe, offrir des coupons pour l'autre quand l'un est acheté seul, ...

• Beaucoup d'intérêt à connaître les habitudes des clients.

⇒ Prolifération de discount/coupon/carte fidélité

⇒ Identification individuelle des clients (pour les cibler).

⇒ Informations beaucoup plus importantes que le prix du discount.

Direct Marketing : un autre domaine

⇒ Les promos sont chères et ne sont intéressantes que si elles "marchent".

⇒ On utilise des courriers/méls directs ou l'on demande à appeler un N° gratuit.

⇒ Il y a peu de réponse de la part des clients.

➥ Important : réduire au maximum le nombre des clients visés pour obtenir les bonnes réponses (éviter ceux qui ne seront pas intéressés).

➥ On peut travailler sur les données démographiques (basées sur le code zip) pour trouver des corrélations avec des clients futurs.

➥ Les habitants d'un même quartier risquent d'avoir les mêmes comportements !

3 Clustering

- **Technique non supervisée**

  Versus la classification où l'on connaît les classes par avance

  ➥ Dans Clustering : les clusters (classes) sont inconnus a priori.

- Clustering = Regrouper les données en plusieurs groupes (=clusters)

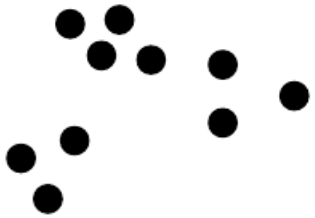
- Chaque groupe doit être **homogène** et se distinguer des autres groupes.

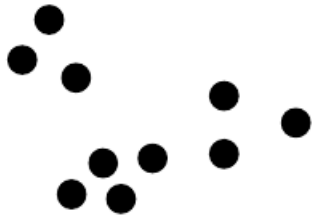
  ➥ Minimiser les intra-distances, maximiser les inter-distances

  ➥ Il arrive d'avoir des groupes qui se recouvrent ➙ probabilités

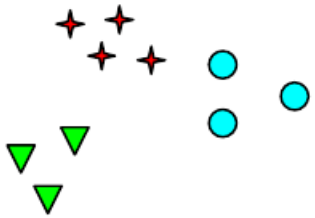
- Autre variante : Techniques Hiérarchiques ➙ Raffinement par niveau

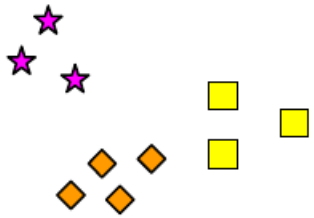
- **La notion de Cluster est ambigue :**



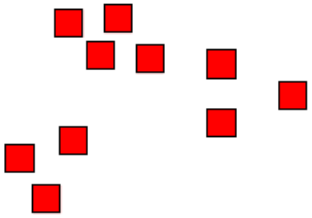


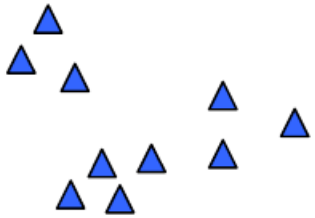
Combien de clusters ?



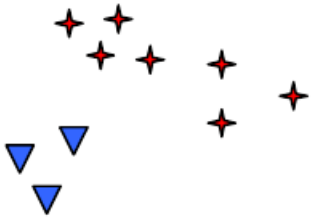


Six clusters





Deux clusters



Quatre clusters

3.1 Mesures de similarité et distance

• Soit O un ensemble d'instances

• Une fonction $d : O \times O \rightarrow \mathbb{R}$ définit une **distance** si pour tout $x, y, z \in O$:

- $d(x, y) \geq 0$
- $d(x, y) = 0 \; ssi \; x = y$ réflexive
- $d(x, y) = d(y, x)$ commutative
- $d(x, z) \leq d(x, y) + d(y, z)$ ligne droite

3.2 Distance dans $\mathbb{R}^n$

Exemple en $\mathbb{R}^n$

• Soient $\vec{x} = (x_1, x_2, ..., x_n)$ et $\vec{y} = (y_1, y_2, ..., y_n)$ deux points dans $\mathbb{R}^n$

✘ Distance **Euclidienne** : $d_{Eucl}(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

✘ Distance **Manhattan** (ou "city block") : $d_{Manh}(\vec{x}, \vec{y}) = \sum_{i=1}^{n}|x_i - y_i|$

✘ Distance de **Minkowski** : Soit $q \in \mathbb{N}, q > 0$.

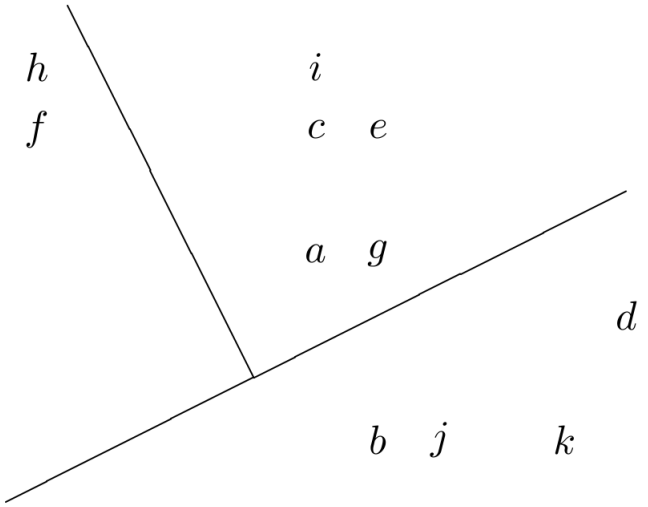
$$d_{Mink(q)}(\vec{x}, \vec{y}) = \sqrt[q]{\sum_{i=1}^{n}|x_i - y_i|^q}$$

• N.B. : $d_{Eucl}(\vec{x}, \vec{y}) = d_{Mink(2)}(\vec{x}, \vec{y})$ et

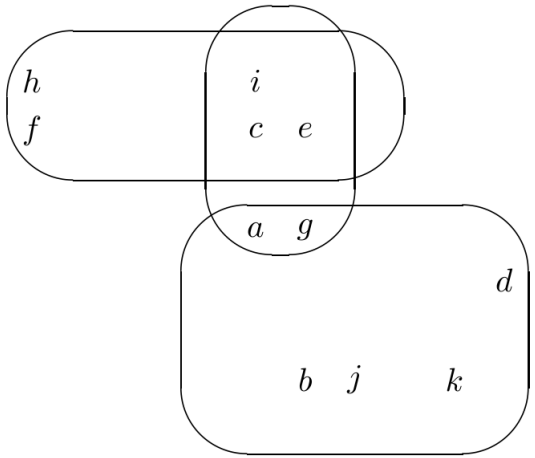
$d_{Manh}(\vec{x}, \vec{y}) = d_{Mink(1)}(\vec{x}, \vec{y})$

3.3 Aperçu des différents types de Clustering

clusters disjoints



clusters pas forcément disjoints



◗ Voir aussi la version Hiérarchique en section 3.3

../..

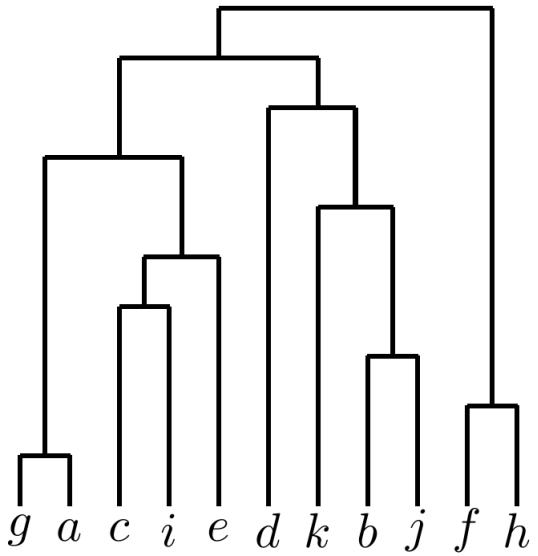
Différents types de clustering (suite)

Clusters Probabilistes

L'instance a *répartie en 3 clusters*

	1	2	3
a	0.3	0.4	0.3
b	0.1	0.2	0.7
c	0.4	0.5	0.1
$\vdots$		$\vdots$	

Hiérarchie de clusters (dendrogram)



3.4 Kmeans

Principe général : clustering Itératif à base d'instances

• On souhaite partitionner un ensemble S en $k \geq 2$ clusters.

• Soient $C_1, C_2, ..., C_k$ des clusters avec les centres $c_1, c_2, ..., c_k$ respectivement.

• On définit la **dispersion intra-cluster** par $\sum_{i=1}^{k} \sum_{x \in C_k} d(c_k, x)$

• Le but est de trouver un clustering avec une dispersion intra-cluster minimale.

3.4.1 *Algorithme de Kmeans*

Partitionner un ensemble S en k clusters.

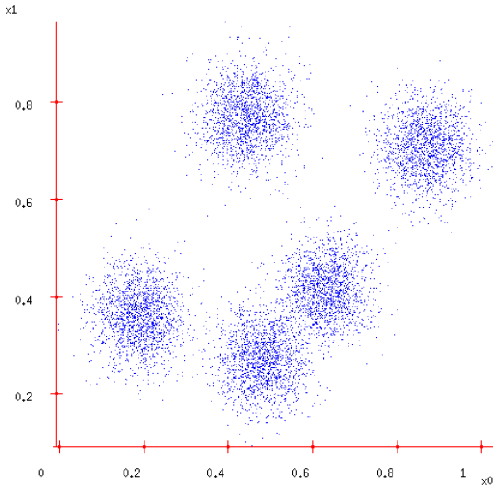
Données : D l'ensemble d'apprentissage
Résultat : Un ensemble de k clusters
début
  Choisir les moyennes $m_1, m_2, ..., m_k$ (e.g. : distance Euclidienne);
  répéter
    Attribuer tout objet de S à la moyenne la plus proche;

    Soient $C_1, C_2, ..., C_k$ les ensembles d'objets attribués respectivement à $m_1, m_2, ..., m_k$;
    Ajuster les moyennes par : ;
      $m1 \leftarrow$ la moyenne de C_1;
      $m2 \leftarrow$ la moyenne de C_1;
        ...;
      $mk \leftarrow$ la moyenne de C_1;
  jusqu'à m_i ne changent pas (stabilisation des clusters) ;
fin

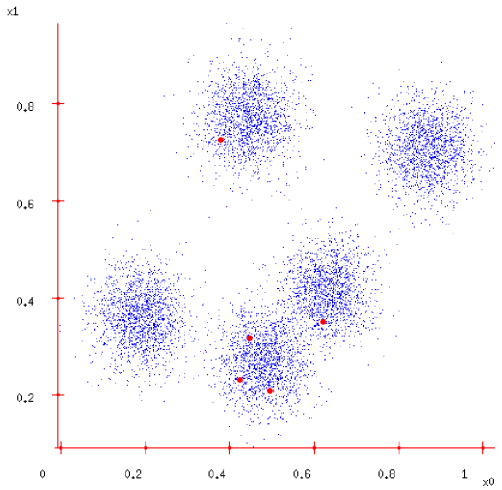
Algorithme 1 : K-Means basique

Rappel de la distance **Euclidienne** = $\sqrt{\left(a_1^{(1)} - a_1^{(2)}\right)^2 + \left(a_2^{(1)} - a_2^{(2)}\right)^2 + \ldots + \left(a_k^{(1)} - a_k^{(2)}\right)^2}$

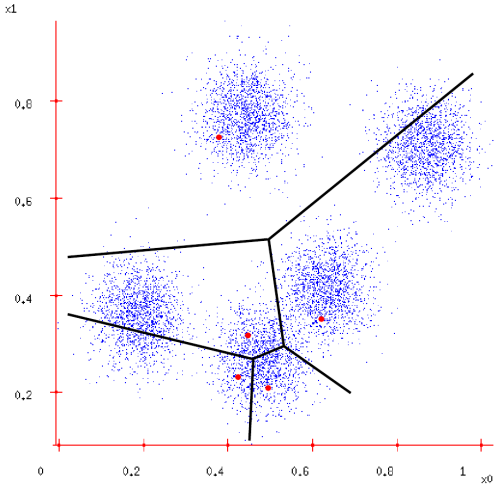
3.4.2 *Illustration de K-means*



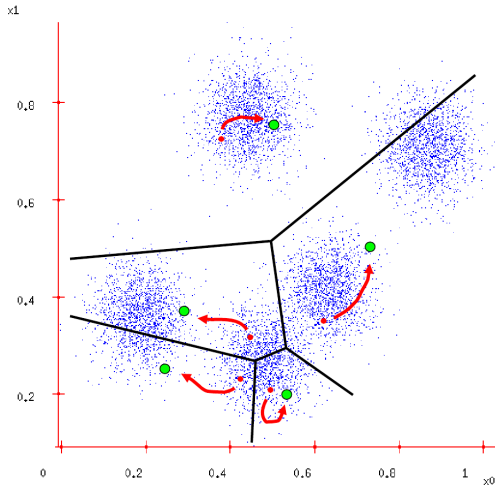
Points d'origine



Initialisation

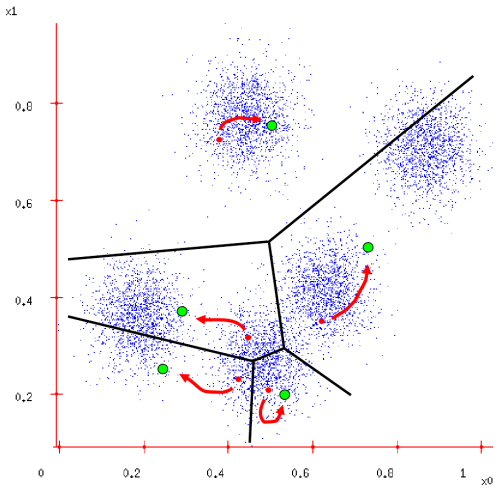


Etape 1

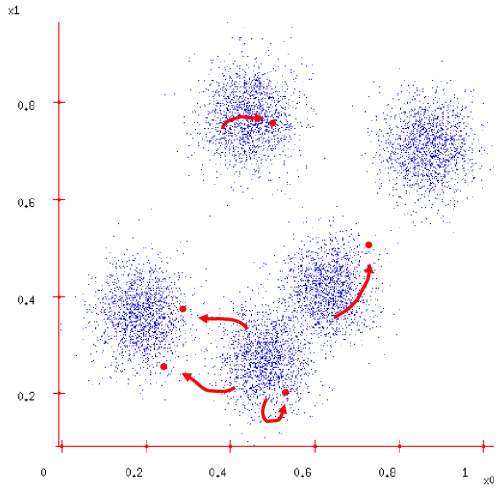


Etape 2

Illustration de K-means (suite) :

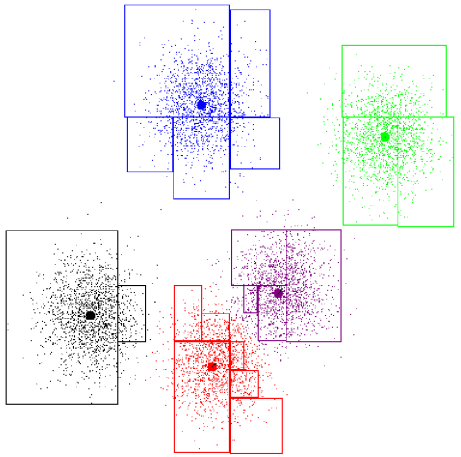


Etape 3



Etape 4 .....

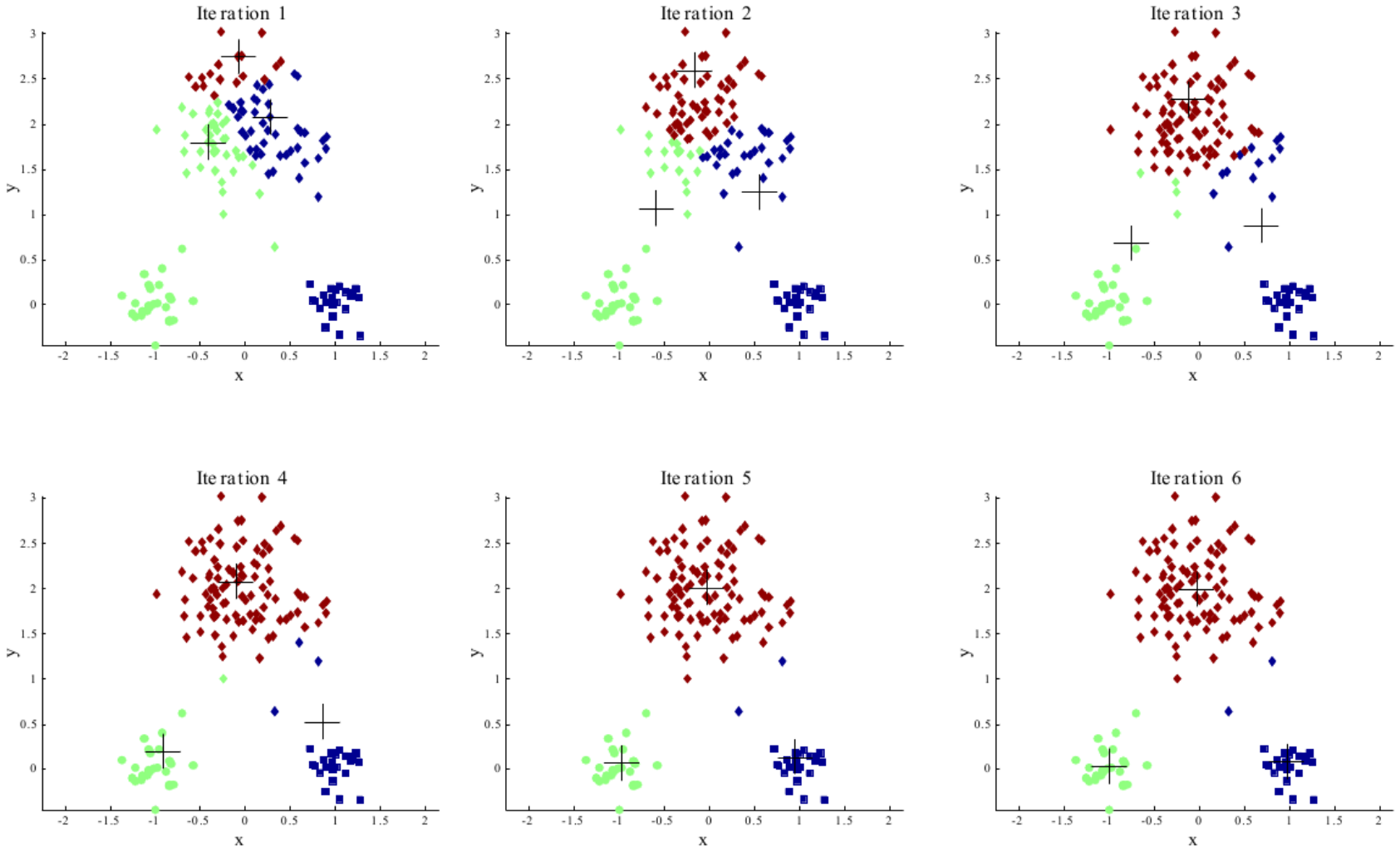
- On fixe $k = 5$ et les centroïdes aléatoires initiaux ;
- Chaque instance "trouve" le Centre le plus "proche" ;
- Chaque Centre "trouve" le Centroïde des points qui lui sont devenus "proches"
- Et s'y déplace !
- .... Jusqu'à ne plus se déplacer !



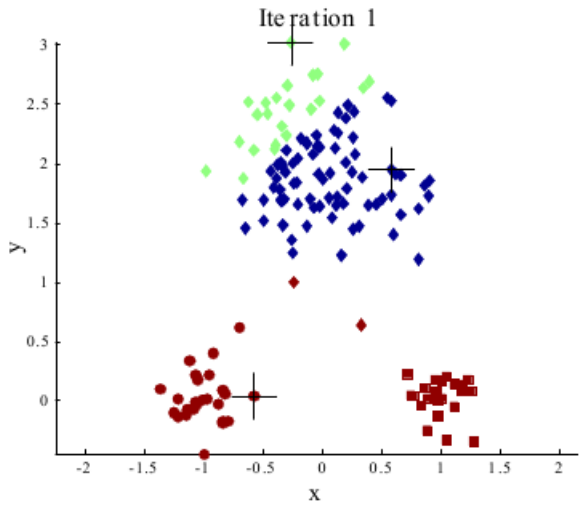
Etape finale

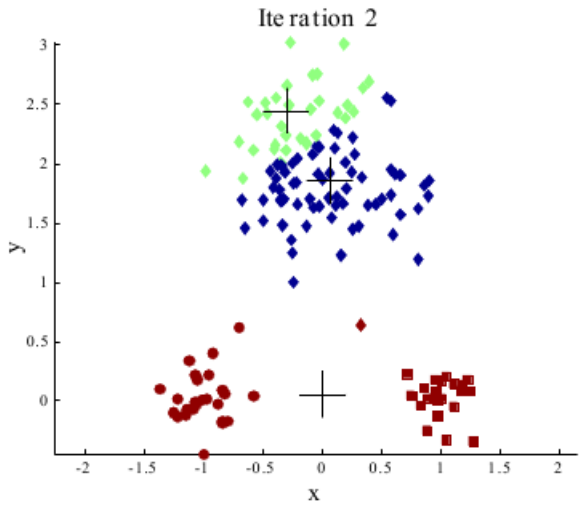
3.4.3 *Kmeans : importance des choix initiaux*

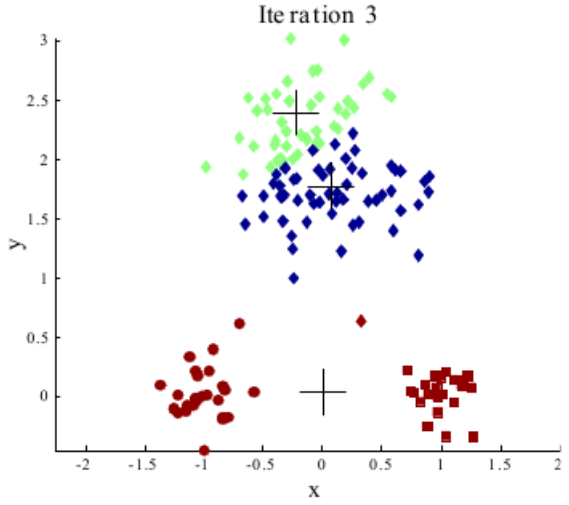
Exemple : le choix des centres initiaux est important (ici, bon choix)

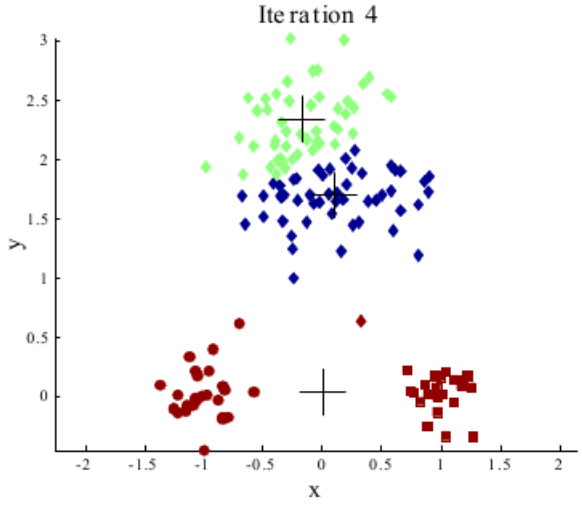


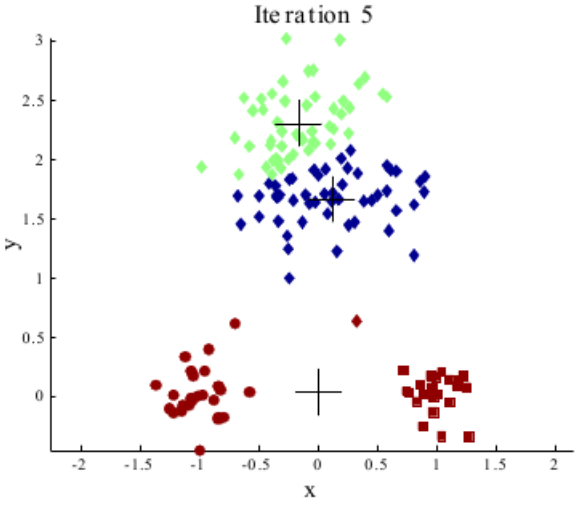
Exemple (suite) : ici, mauvais choix initiaux











3.5 Evaluation de K-means

Notion de **distorsion** :

- Soit les instances $\overrightarrow{x_i}, i = 1..n$ dans un espace $\mathbb{R}^d$ (d attributs)
- Soient une fonction d'encodage $encoder$ $\mathbb{R}^d \rightarrow [1..K]$ (cluster d'une instance)

  et une fonction de décodage $decoder$ $[1..K] \rightarrow \mathbb{R}^d$ (le centre d'un cluster)
- On définit Distorsion : $$Distorsion = \sum_{i=1}^{n} [x_i - decoder(encoder(x_i))]^2$$
- Ceci est la base de calcul de l'erreur de Clustering.
- Ainsi, si $decoder[j] = m_j$ le centre du cluster C_j, on obtient l'erreur SSE

➧ Chaque Centroïde de C_j doit être au centre (moyenne) de son domaine ($dom(m_j)$)

Evaluation de K-means (suite) :

• La mesure la plus commune : **la somme des erreurs au carré** (SSE) :

✘ Pour chaque point, l'erreur est la distance au cluster le plus proche

✘ Pour calculer SSE, on lève ces écarts au carré et on somme :

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} dist^2(m_i, x)$$

x : une instance dans le clauster C_i, m_i : le point représentatif du claster C_i

➥ On peut montrer que m_i correspond au centre (moyenne) du cluster C_i

✘ Etant donné deux *clustering*, on peut choisir celui avec le minimum d'erreur

✘ Une manière simple de réduire SSE est d' augmenter k (le nombre de clusters)

Néanmoins : un bon clustering avec un plus petit k peut avoir une moindre SSE qu'un mauvais clustering avec un k plus élevé.

3.5.1 *K-means Dichotomique (Bisecting K-means)*

- Un variante de Kmeans qui peut produire un clustering par partition ou Hiérarchique
- Le principe : partition Dichotomique des clusters jusqu'à l'obtention de k clusters.
- **Algorithme :**

Données : D l'ensemble d'apprentissage ; M : nombre d'itérations
Résultat : Un ensemble de k clusters
début
  Initialiser la liste LC des clusters par un seul contenant tous les points;
  répéter
    Choisir un cluster C de la liste des clusters;
    pour tous les i de 1 à M : $nbr.d'itérations$ **faire**
      % M : nombre d'itérations fixé en entrée;
      Découper C en deux clusters en utilisant Kmeans basique (et minimiser SSE);
    fin
    Le cluster C a produit 2 meilleurs clusters (après M tentatives);
    Ajouer à LC les deux clusters de la bisection avec un minimum de SSE;
  jusqu'à avoir k clusters ;
  LC contient k clusters ;
fin

Algorithme 2 : K-Means Dichotomique : Bisecting K-means

3.6 Clustering Probabiliste

Justification du Clustering Statistique (Probabiliste) :

• D'un point de vu statistique, la tâche de Clustering est de trouver un ensemble de clusters les plus **vraissemblables**, étant donné une BD.

• En l'absence d'assez d'évidence (sur la décision de placer une instance définitivement dans tel ou tel cluster), on peut plutôt considérer la probabilité pour une instance d'être dans tel ou tel cluster .

3.6.1 Modèle de Mélange (Mixture) Finie

• La base de la méthode de Clustering Statistique est le modèle "mixture finie".

• Une **mixture** est un ensemble de K distributions de probabilités qui représenteront les K clusters. Chaque distribution est celle des attributs d'un cluster .

➥ Plus exactement, chaque distribution Θ_i donne la probabilité pour une instance particulière d'avoir un certain ensemble de valeurs d'attributs *"S'il devait appratenir réellement au cluster C_i"*.

• Les clusters ne sont pas semblables.

➥ Chaque cluster a une distribution qui représente sa population.

➥ Chaque instance appartient à un seul cluster mais l'on ne sait pas lequel.

3.6.2 Un cas simple de Mélange

- Un seul attribut numérique.

- Deux clusters A et B.

- Les clusters ont chacun une distribution **Normale** (gaussienne) avec moyenne et variance différentes.

- Une BD, un seul attribut numérique ➙ on a un ensemble de valeurs numériques.

• La tâche est de claculer la *moyenne* et la *variance* pour chaque cluster afin de déterminer les population pour ces deux clusters.

• Le modèle Mixture combine plusiers distributions **Normales** (ici, 2) dont les fonctions de densité de probabilité ressemblent à une *chaîne de montagnes* (ici 2 sommets).

Exemple : $P_A + P_B = 1$

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

FIGURE 3.1 – Données de l'exemple de Mixture finie des clusters A et B (connus)

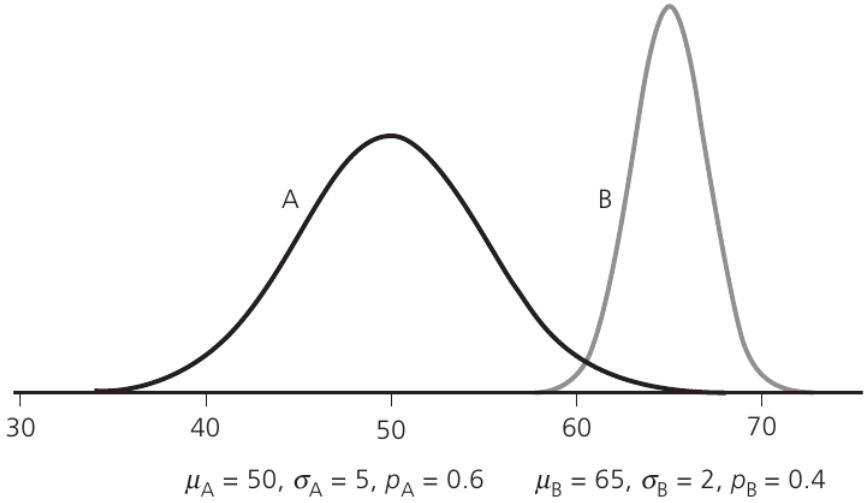


FIGURE 3.2 – Distributions Normales de l'exemple de Mixture finie

Exemple Mixture Finie : les 5 paramètres $\mu_A, \sigma_A, P_A, \mu_B, \sigma_B$ ($P_B = 1 - P_A$).

• Si l'on connaît de quelle distribution vient chaque instance, **trouver les 5 params** serait facile (les x_i sont des instances de la distribution A ou B) :

- Soit $A = \{x_1, x_2, \cdots, x_{n_A}\}$ $B = \{y_1, y_2, \cdots, y_{n_B}\}$
- Estimer $\mu_A, \sigma_A, \mu_B, \sigma_B$ séparément par (pour A puis pour B) :
  - Moyenne de l'échantillon $\mu_A = \frac{1}{n_A} \sum_1^{n_A} x_i$
  - Variance de l'échantillon $\sigma_A^2 = \frac{1}{n_A - 1} \sum_1^{n_A} (x_i - \mu_A)^2$
- Si parmi n instances, n_A sont dans A et n_B dans B, alors $P_A = \frac{n_A}{n}$
- de même pour B

• NB : $n_A - 1$ est une technique d'échantillonnage ; même effet que n si n grand

Suite : on trouve le cluster de chaque instance (à l'aide des 5 paramètres) :

➥ Pour une instance x, sa probabilité $Pr[A|x]$ d'appartenir au cluser A :

$$Pr[A|x] = \frac{Pr[x|A] \,.\, Pr(A)}{Pr[x]} = \frac{f(x; \mu_A, \sigma_A) \,.\, P_A}{Pr[x]}$$

Où $f(x; \mu_A, \sigma_A)$ est la densité de probabilité pour la distribution *Normale* de A :

$$f(x; \mu_A, \sigma_A) = \frac{1}{\sqrt{2\pi}\ \sigma_A} \,.\, e^{-\frac{(x-\mu_A)^2}{2\sigma_A^2}}$$

- Le traitement est le même que pour **Bayésien naïf** :

  ➥ traitement des attributs numériques et mixtes.

  ➥ le dénominateur $Pr[x]$ disparaîtra (normalisation).

  ➥ $f(x)$ n'est pas exactement la probabilité $Pr[x|A]$ (nulle pour une valeur particulière)

  mais la normalisation permet d'avoir un résultat final correct.

Remarque : Mélange (Mixture) Fini et la génération d'une BD

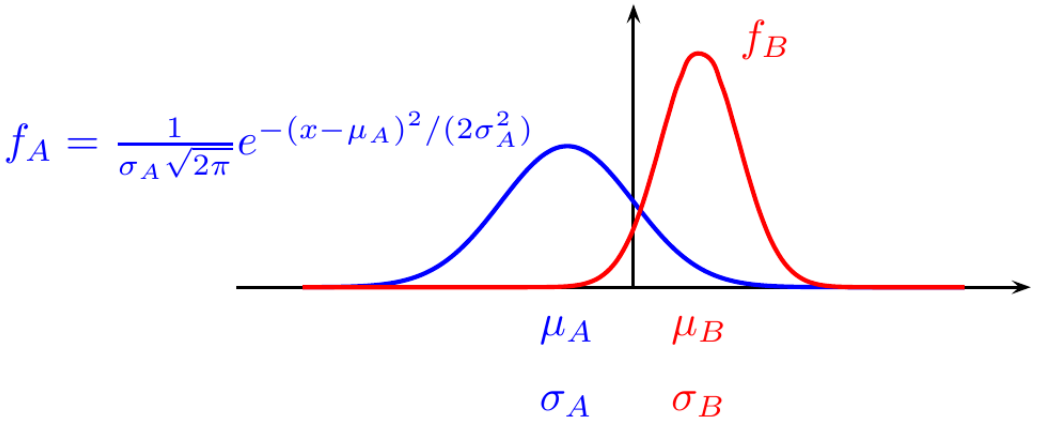
• Lorsque les 5 params sont connus, on peut générer n valeurs aléatoires de la manière suivante :

Soit $P_A + P_B = 1$.

1- Générer $P_A \times n$ valeurs selon la distribution f_A

2- Générer $P_B \times n$ valeurs selon la distribution f_B

Cette génération dépend de cinq paramètres : $p_A, \mu_A, \sigma_A, \mu_B, \sigma_B$.



3.6.3 La Méthode EM

Hypothèse et intérêt du Clustering :

On ne connait pas les classes (clusters) des instances de l'exemple 3.1.

➥ On ne connaît ni la distribution d'origine de chaque instance ;

➥ ni les 5 params du modèle Mixture finie.

Principe de la méthode Expectation Maximization (EM) :

➤ On adopte le cadre général de K-means en itérant.

➤ On commence par une valeur estimée pour chacun des 5 params $p_A, \mu_A, \sigma_A, \mu_B, \sigma_B$.

➤ On les utilise pour calculer la proba de cluster de chaque instance (cf. ci-dessus),

➤ On utilise ces probabilités pour ré estimer les 5 params ...

➤ Répéter jusqu'à la maximisation de la vraisemblance (voir plus loin)

Expectation Maximization (EM) (suite) :

• Variante : On peut commncer par une estimation du cluster des instances.

• Les étapes ci-dessus décrivent la méthode *Expectation Maximization* :

➥ étape Expectation : calcul des probabilités des clusters (expected class values)

➥ étape Maximization : calcul des distributions et la maximization de la vraissemblance des distributions étant donnée la BD.

Expectation Maximization (EM) (suite) : les pondérations

Un ajustement mineure :

➥ Les valeurs sont des probabilités de clusters et non les clusters eux mêmes .

➥ Ces probabilités se comportent comme des pondérations :

Si w_i est la probabilité que l'instance x_i appartienne au cluster A, on aura :

$$\mu_A = \frac{w_1 x_1 + w_2 x_2 + ... w_n x_n}{w_1 + w_2 + ... + w_n} = \frac{\sum_i w_i \,.\, x_i}{\sum_i w_i} \qquad et \qquad P_A = \frac{\sum_i w_i}{n}$$

$$\sigma_A^2 = \frac{w_1 (x_1 - \mu_A)^2 + w_2 (x_2 - \mu_A)^2 + ... + w_n (x_n - \mu_A)^2}{w_1 + w_2 + ... + w_n} = \frac{\sum_i w_i \,.\, (x_i - \mu_A)^2}{\sum_i w_i}$$

Cette fois, on prend en compte **TOUS les** x_i et pas seulement ceux du clauster A.

3.6.4 La terminaison de l'algorithme EM

• Rappel : K-means s'arrête sur une sorte de *point fixe*

➥ quand les classes (clusters) des instances ne changent plus.

Dans EM, c'est moins simple :

➥ L'algorithme converge vers son point fixe mais ne l'atteint jamais.

➥ Mais on peut savoir combien on y est proche en calculant la vraissemblance globale que les données **viennnent effectivement de la BD** dont on a claculé les 5 params.

• La vraissemblance globale est le produit des probabilités des instances indépendantes x_i (pour un cas bi-clusters). ../..

La terminaison de l'algorithme EM (suite) :

• Rappel : on a vu que $Pr[A|x_i] = \dfrac{Pr[x_i|A] \,.\, Pr(A)}{Pr[x_i]}$ (idem pour le cluster B).

Dans cette expression (cf. Bayésien) que le dénominateur $Pr[x]$ utilisé pour la normalisation représente $Pr[x_i|A] \,.\, Pr(A) + Pr[x_i|B] \,.\, Pr(B) = Pr[x_i]$

➥ Avec $Pr[x_i|A] = f(x_i; \mu_A, \sigma_A)$ (idem pour B) et $f(x_i; \mu_A, \sigma_A) = \dfrac{1}{\sqrt{2\pi}\,\sigma_A} \,.\, e^{-\dfrac{(x_i - \mu_A)^2}{2\sigma_A^2}}$

➥ Cette somme ($Pr[x_i]$) correspond à la **probabilité de générer** x_i, étant donné les 5 paramètres.

• En supposant l'indépendance des instances x_i, la probabilité de générer la BD, étant donné les 5 paramètres sera la vraissemblance (cas bi-cluster) :

$$L = \prod_i^n P_A \times Pr[x_i|A] + P_B \times Pr[x_i|B] = \prod_i^n P_A \times f_A(x_i) + P_B \times f_B(x_i)$$

➥ $P_A \times f_A(x) + P_B \times f_B(x)$ ci-dessus $\sim$ probabilité de générer x

• L est une mesure de qualité du clustering ; elle augmente à chaque itération de l'EM .

3.6.5 *Bilan de la méthode EM bi-classes*

- On demande de diviser l'ensemble S de n valeurs en deux clusters A et B.
- On ne connaît aucun des cinq paramètres.

1. Choisir A et $B = S \setminus A$ deux clusters de départ.
2. Calculer $p_A, \mu_A, \sigma_A, \mu_B, \sigma_B$ (utiliser $f(x; \mu, \sigma)$)
3. Expectation : Calculer $Pr(A|x)$ et $Pr(B|x)$ pour tout $x \in S$ (utilise $f(x; \mu_A, \sigma_A)$ et P_A)
4. Maximization :

$$\mu_A := \frac{\sum_{x \in S} Pr(A|x) \,.\, x}{\sum_{x \in S} Pr(A|x)} \qquad \sigma_A^2 := \frac{\sum_{x \in S} Pr(A|x) \,.\, (x - \mu_A)^2}{\sum_{x \in S} Pr(A|x)} \qquad p_A := \frac{\sum_{x \in S} Pr(A|x)}{n}$$

   Idem pour B.

5. Aller à 3 (Jusqu'à stabilisation)

Bilan de la méthode EM bi-classes (suite)

• Une fois les 5 paramètres connus, on peut calculer le cluster de chaque x_i

$$Pr[A|x_i] = \frac{Pr[x_i|A] \times Pr[A]}{Pr[x_i]} \sim \frac{f_A(x_i) \times P_A}{Pr[x_i]} \qquad \text{De même pour } B.$$

• On sait aussi calculer $Pr[A|x_i]$ via :

$$\frac{Pr[A|x_i]}{Pr[B|x_i]} = \frac{f_A(x_i) \times P_A}{f_B(x_i) \times P_B} \qquad \text{Avec } Pr[A|x_i] + Pr[B|x_i] = 1$$

➥ On a alors : $Pr[A|x_i] = \dfrac{f_A(x_i) \times P_A}{f_A(x_i) \times P_A + f_B(x_i) \times P_B}$

➥ $$Pr[B|x_i] = \frac{f_B(x_i) \times P_B}{f_A(x_i) \times P_A + f_B(x_i) \times P_B}$$

Rappel : comme dans Bayes, le dénominateur peuvent être ignorés.

Dans ce cas, ce sont les valeurs relatives des probabilités qui détermineent les clusters.

3.6.6 Une application d'EM : données manquantes

• Retrouver des valeurs manquantes d'une BD (avec 4 valeurs x_1, x_2, x_3, x_4).

• On connait $\frac{1}{2}$ des valeurs d'une BD (uni-variable, attribut continue, loi Normale).

➥ L'autre moitié (x_3, x_4) manquante ou corrompue.

• Calculer les valeurs manquantes par EM (meilleures estimations)

• Hypothèse : la variance est unitaire (=1).

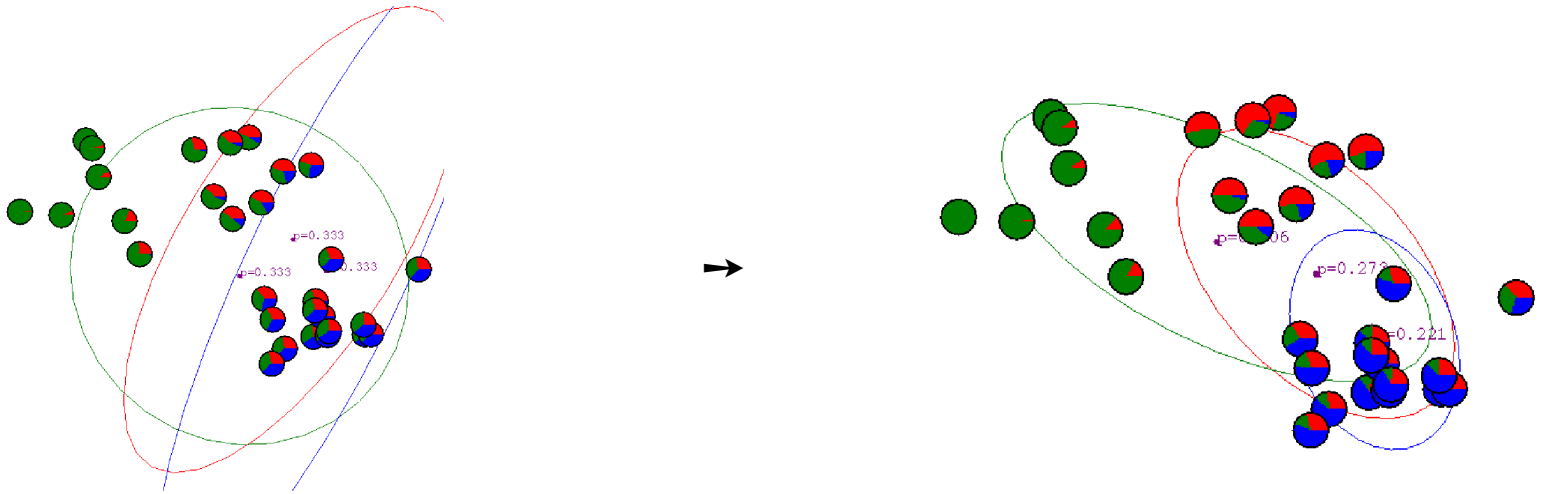
• **Etape E** : estimer les variables manquantes (μ puis =x_3 et x_4).

• **Etape M** : ré estimer les paramètres de la distribution pour maximiser la vraisemblance des données, sachant la BD (complétée)

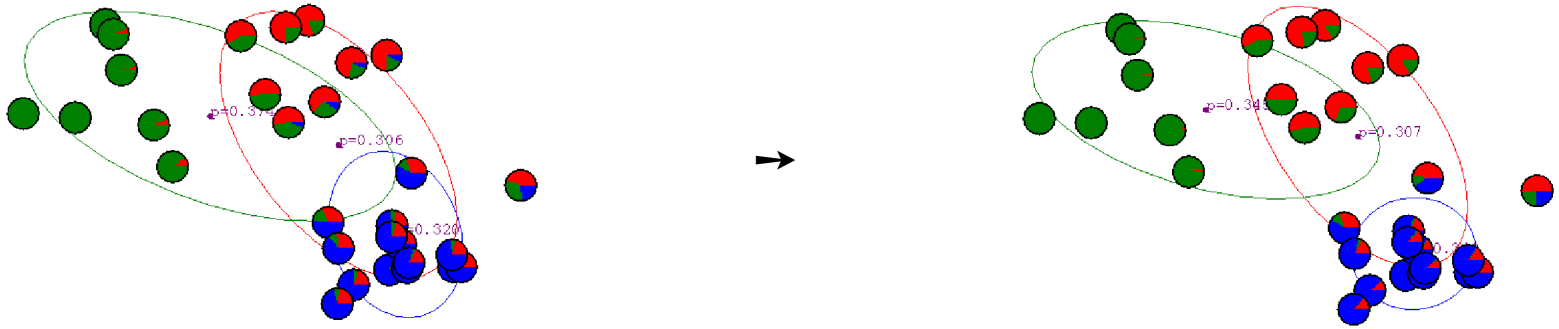
Déroulement d'un exemple :

- Initialisation : BD = [4, 10, ?, ?], $\sigma = 1$,
- On choisit $\mu = 0$, ➙ Nouvelle BD : [4, 10, 0, 0]
- Nouvelle $\mu = 3.5$, ➙ Nouvelle BD : [4, 10, 3.5, 3.5]
- Nouvelle $\mu = 5.25$, ➙ Nouvelle BD : [4, 10, 5.25, 5.25]
- Nouvelle $\mu = 6.125$, ➙ Nouvelle BD : [4, 10, 6.125, 6.125]
- Nouvelle $\mu = 6.5625$, ➙ Nouvelle BD : [4, 10, 6.5625, 6.5625]
- Nouvelle $\mu = 6.7825$, ➙ Nouvelle BD : [4, 10, 6.7825, 6.7825]
- Nouvelle $\mu = 6.890625$, ...
- .....
- Nouvelle $\mu = 7$, Nouvelle BD : [4, 10, 7,7] ➙ Ne change plus.

3.6.7 *Exemple EM*



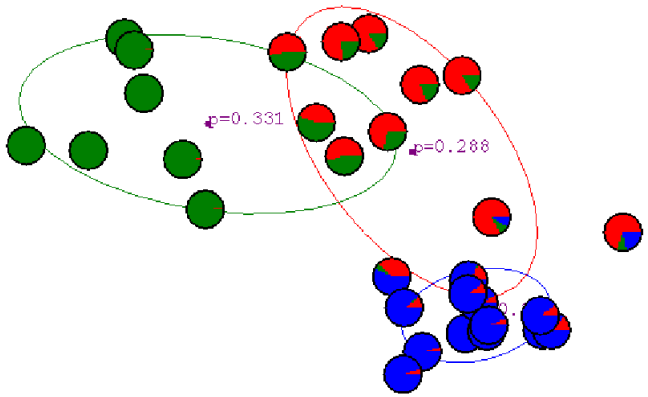
(a) Initialisation (b) 1e itération



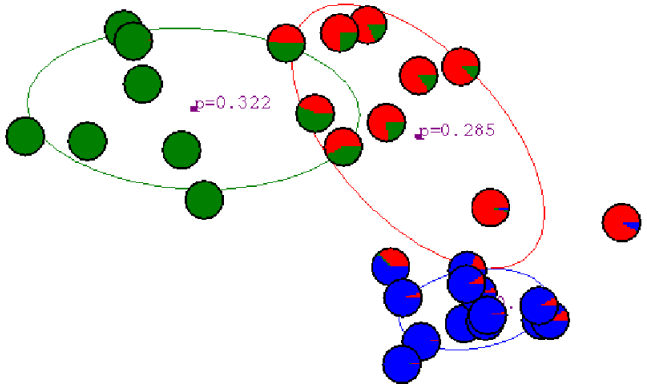
(c) 2e itération (d) 3e itération

../..

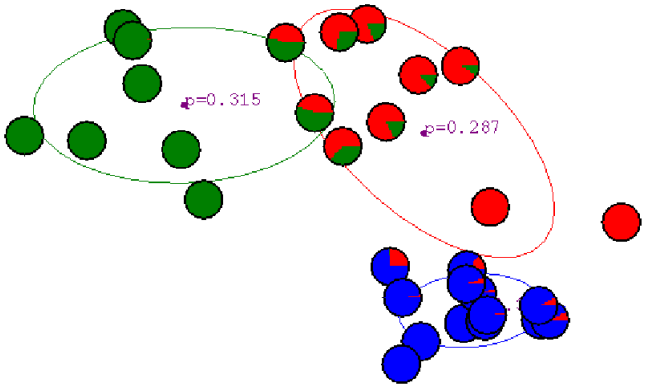
Exemple (suite) :



(e) 4e itération

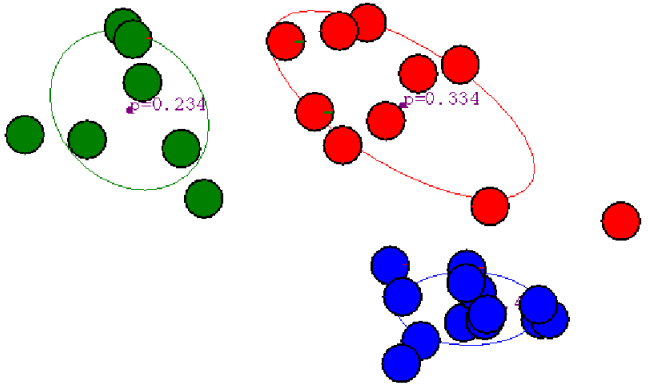


(f) 5e itération



(g) 6e itération ...

...



Après la 20e itération

3.7 Classification Hiérarchique

• Produit un ensemble de clusters imbriqués organisé comme un arbre Hiérarchique

• Peut être visualisé comme un **dendrogram**

Exemple : un diagramme arborescent qui représente une séquence de fusion-partitions

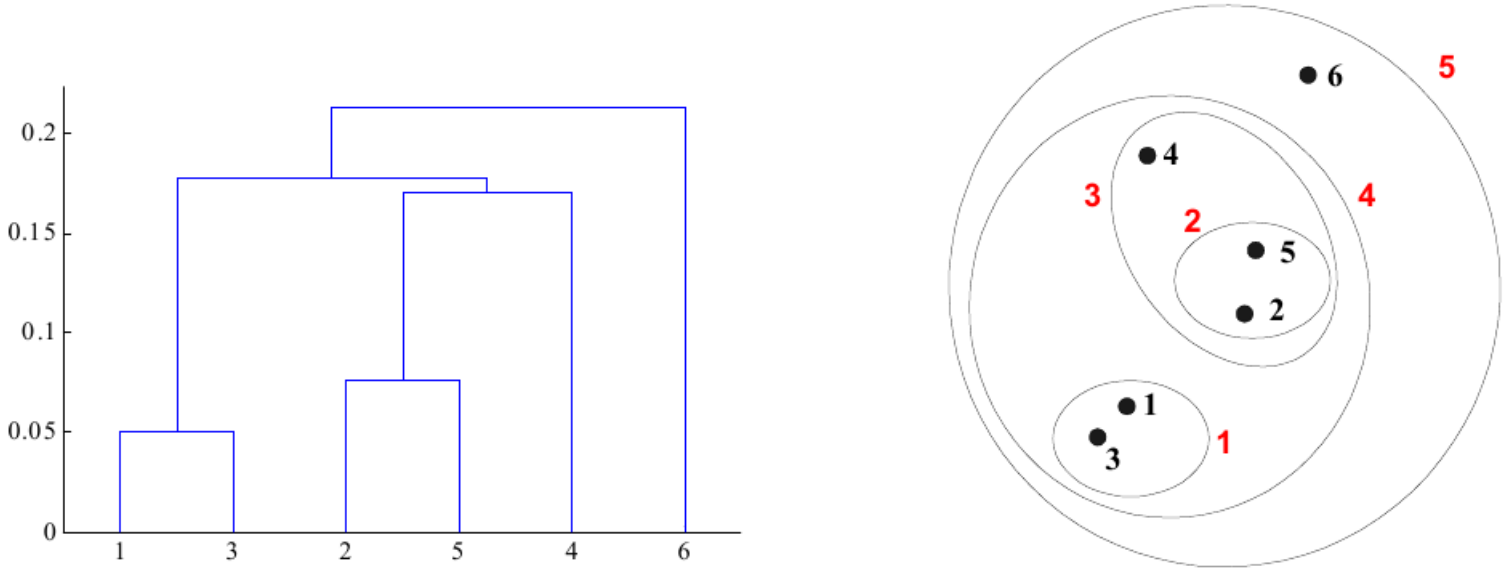


FIGURE 3.3 – exemple de dendrogram

Classification Hiérarchique (suite) :

Caractéristiques :

• On ne connaît pas (par avance) le nombre de clusters

➥ Le nombre de clusters désirés s'obtient en "coupant" le dendrogram.

• La structure hiérarchique (dendrogramme) peut correspondre à une taxonomie / ontologie réelle ➙ E.g. : dans le cas des animaux / objets structurés, etc.

Dendogram et performances du Clustering

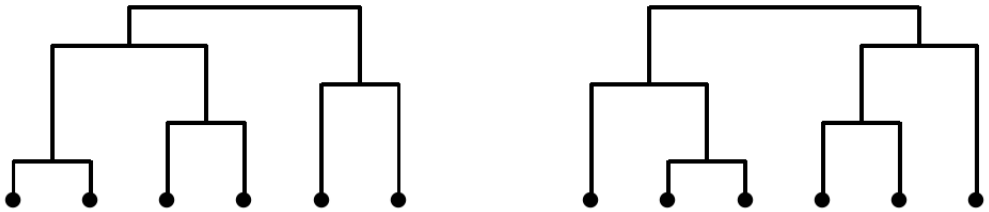


FIGURE 3.4 – Exemples de dendrogram

3.7.1 Deux Types de Clustering Hiérarchiques

➤ Classification Agglomérative (ascendant) :

➥ Débute avec un point (instance) par cluster

➥ A chaque étape, on peut fusionner une paire de clusters "proches" jusqu'à l'obtention de k (désiré) clusters ; voire un seul !

➤ Classification Divisive (descendant) :

➥ Commence par un seul cluster contenant tous les éléments

➥ A chaque étape, on divise un cluster en deux jusqu'à l'obtention de k (désiré) clusters ; voire un par point par cluster !

• La plupart des algorithmes hiérarchiques utilise une **matrice de similarité** (ou distance) pour fusionner deux clusters ou partitionner un cluster à la fois.

3.7.2 *Exemple Clustering Agglomérative*

Déroulement : on commence par des clusters singletons + matrice de similarité

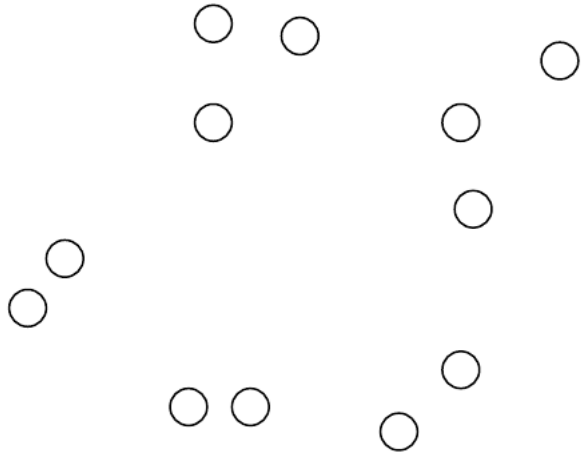


Fig : Les points à classer

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						

Matrice de proximité des points (= clusters)

Clusters initiaux (singletons)

Clustering Agglomérative (suite déroulement) :

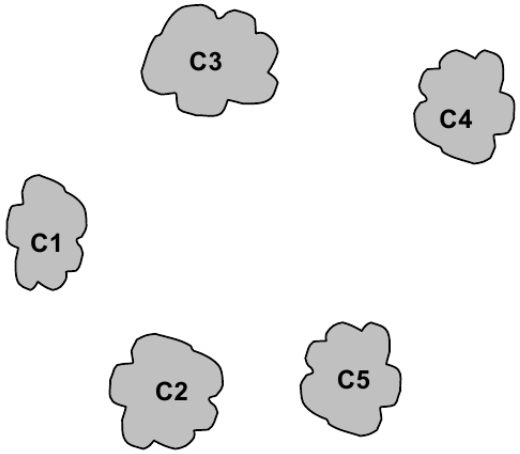
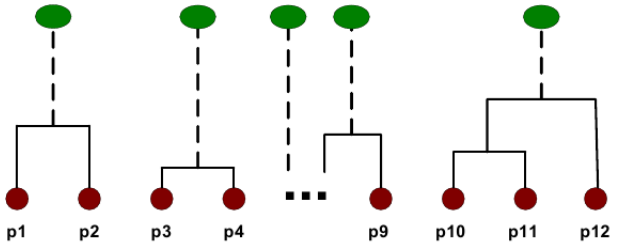


Fig : Les clusters de l'étape courante

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Matrice de proximité (des clusters)



Représentation Hiérarchique des clusters

Clustering Agglomérative (suite déroulement) :

Préparation de la fusion de deux clusters et la MAJ de la matrice :

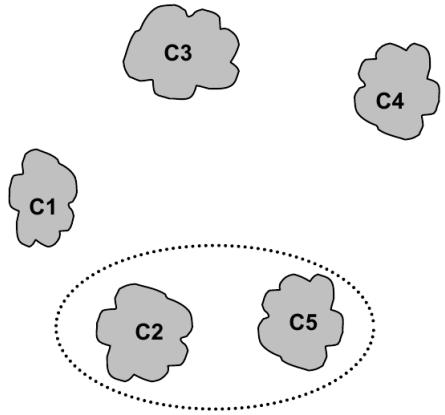
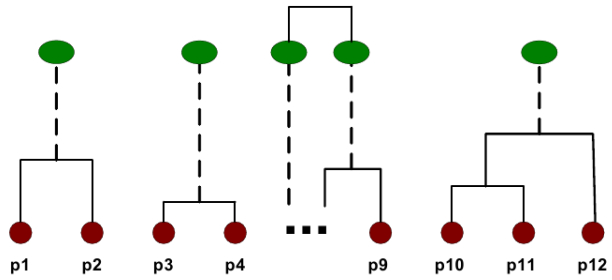


Fig : Fusion de deux clusters



Représentation Hiérarchique de la fusion

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

⇓ *Fusion de la matrice de proximité* ⇓

	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Représentation Hiérarchique de la fusion

3.7.3 La similarité inter-classes

- Comment définir la mesure de similarité entre deux clusters (pour la fusion) ?

  ➥ **Minimum** des distances entre tout couple de points (un par cluster)

  ➥ **Maximum** des distances entre tout couple de points (un par cluster)

  ➥ **Moyenne** de distance de tout couple (un par cluster)

  ➥ Entre les **centroïdes** de chaque groupe

- Ces mesures doivent satisfaire une fonction objective :

  ➥ Ex : carré des erreurs (Méthode de Ward, v. plus loin section **??**)

- N.B. : *Min*, *Max* et *centroïdes* s'appliquent entre deux singletons

  ➥ Dit du type "Lien Simple" (Single Link)

  La *Moyenne* est appelé "Lien Complet" (Complete Link) : entre deux groupes.

La similarité inter-classes (suite) :

- Min des distances entre tout couple de points (un par cluster) ;
- Max des distances entre tout couple de points (un par cluster) .

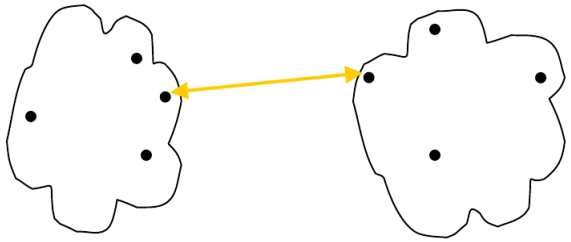


Fig : Distance Minimum

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						

La matrice de proximité des points

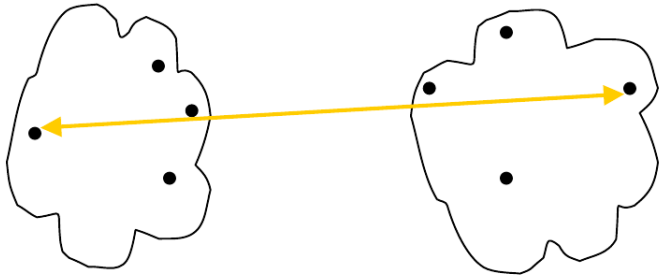
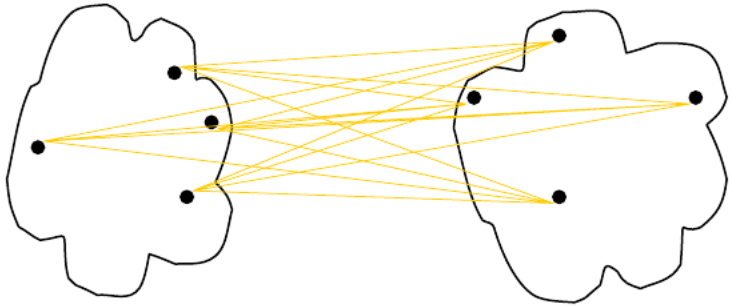


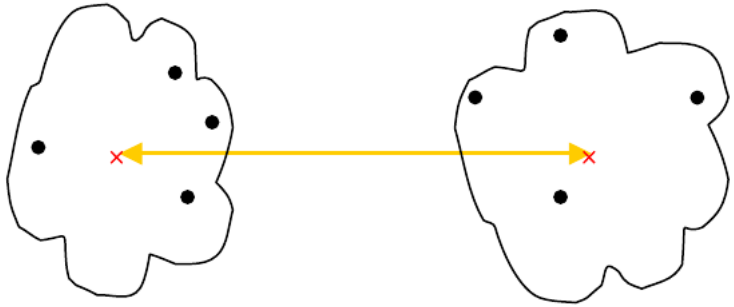
Fig : Distance Maximum

La similarité inter-classes (suite) :

• Moyenne des distances entre tous couples (un par cluster)



• Entre les centroïdes de chaque groupe



• Ces deux mesures utilisent la matrice de proximité des points.

Comportement des similarités inter-classes (suite) :

Une comparaison des distances :

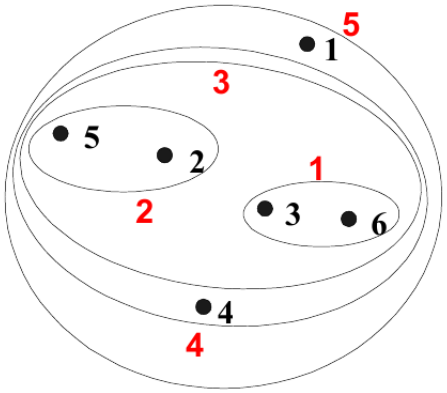
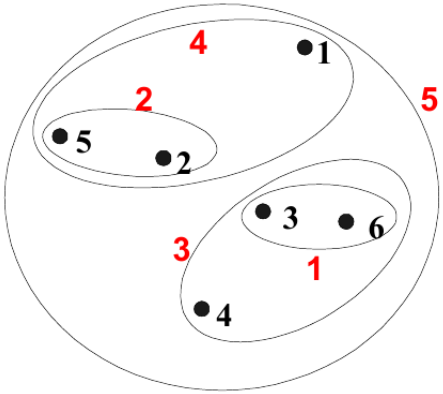
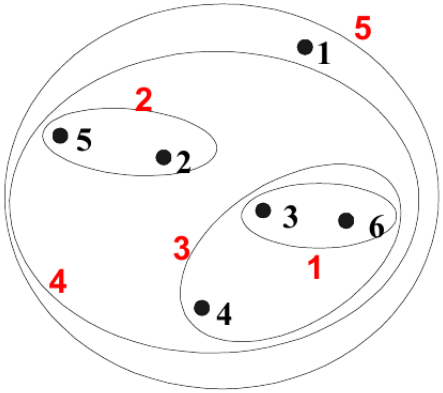


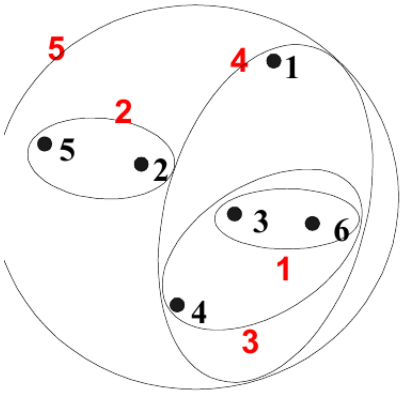
Fig : Distance Min



Distance Max



Moyenne du groupe



Méthode Ward

3.7.4 Remarques générales (Clustering Hiérarchique)

- **Complexité** $O(N^2)$ en **espace** : matrice de proximité avec N = nbr de points
- $O(N^3)$ en temps : N étapes et chaque étape $O(N^2)$

  ➥ Peut être ramené à $O(N^2\ log(N))$ dans certaines approches (optimisées).

Inconvénients du Clustering Hiérarchique :

- Une fois une décision (fusion, division) prise, elle ne peut pas être révoquée.
- Pas de véritable fonction objective à minimiser
- Les différentes approches ont des problèmes avec :

  ➥ Sensibilité aux bruits et outliers

  ➥ Difficulté avec des nuages de forme/taille différentes

  ➥ Tendance à réduire les grands clusters

3.7.5 Classification Hiérarchique Divisive

En 2 étapes : (1) construction MST (un seul cluster) puis (2) division.

Etape 1 : Construction MST (Arbre de recouvrent de poids min) :

• Commencer le MST A avec n'importe quel point

• Chercher les paires de points (p, q) les plus proches tels que $p \in A$ et $q \notin A$ (sinon : circuit) ; Ajouter q à A et établir l'arête qui l'y relie.

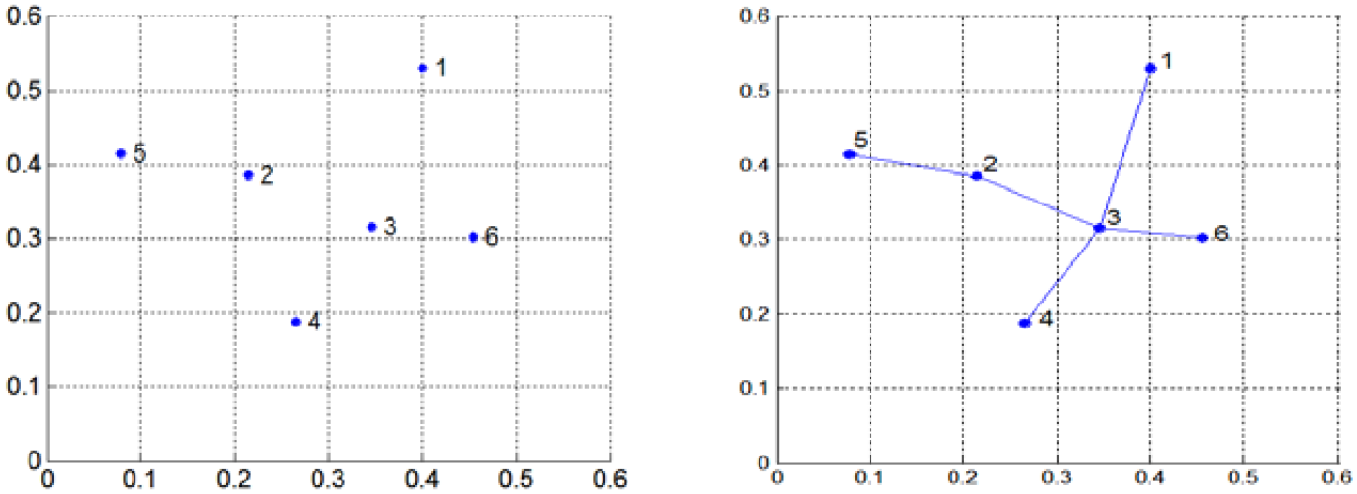


FIGURE 3.5 – MST

Classification Hiérarchique Divisive (suite) :

Etape 2 : Division (après avoir construit le MST) :

1. Construire MST (étape 1)
2. Répéter

   Créer un nouveau cluster en coupant le lien correspondant à la distance la plus grande (la plus petite similarité)
3. jusqu'à l'obtention de k cluster ($k \geq 1$)

Manque une étiquette algo

3.8 Quelques références Bibliographiques

• Ce cours s'est inspiré de plusieurs documents sur l'Extraction de Connaissances ainsi que les outils employés (outils Mathématiques, Probabilité et Statistiques, etc.).

• Aussi, un nombre important d'articles et des rapports de recherche peuvent être consultés (*http ://citeseer.ist.psu.edu/cs*).

• A titre d'indication, les ouvrages suivants peuvent être approfondis :

Data Mining : Introductory and Advanced Topics : Margaret H. Dunham. Prentice Hall 2002.

Data Mining : Concepts, Models, Methods and Algorithms. M. Kantardzic. IEEE 2001

Data Mining : A tutorial based primer. R.J. Roiger , M.W.Geatz

Principles of Data Mining : D. Hand, H. Manila, P. Smyth. MIT Press, Cambridge, 2001

ET bien d'autres !

Table des matières