Cours d'aide à la décision : Du datawarehouse au datamining

L. Jourdan

Laetitia.jourdan@lifl.fr

Plan

Partie 1 : Introduction aux SI et Besoins liés aux SI décisionnels

Partie 2 : Focus sur les entrepôts de données

Partie 3 : Focus sur la fouille de données (data mining)

Le contexte

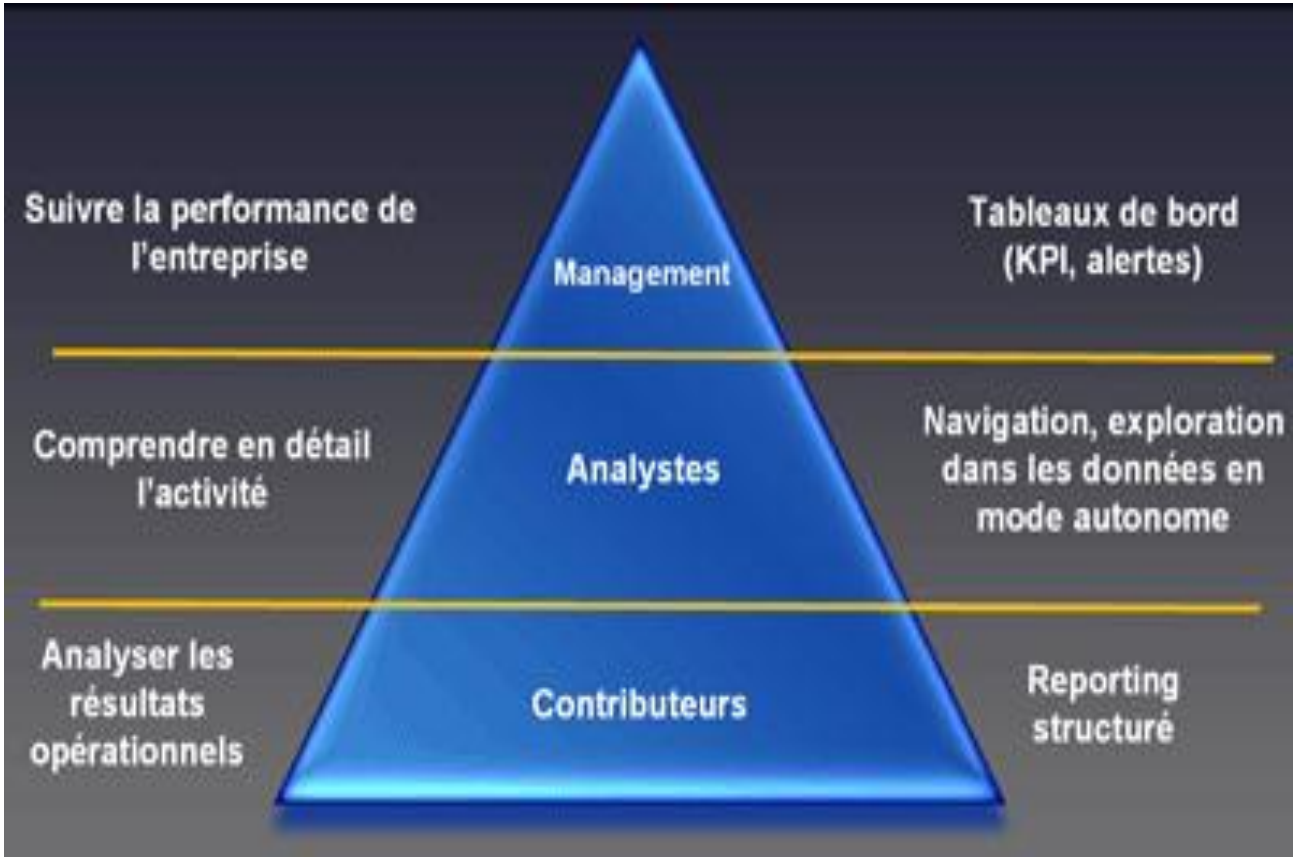
Besoin: prise de décisions stratégiques et tactiques

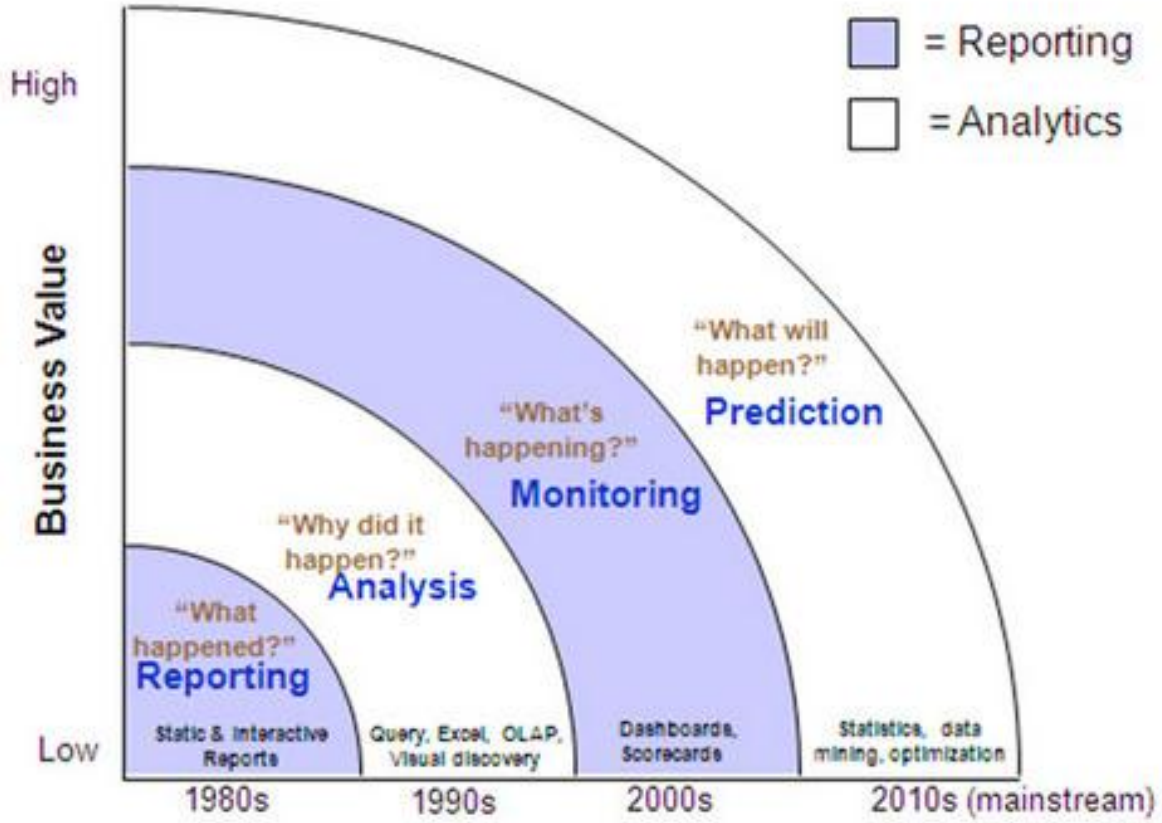
Pourquoi: besoin de réactivité

Qui: les décideurs (non informaticiens)

Comment: répondre aux demandes d'analyse des données, dégager des informations qualitatives nouvelles

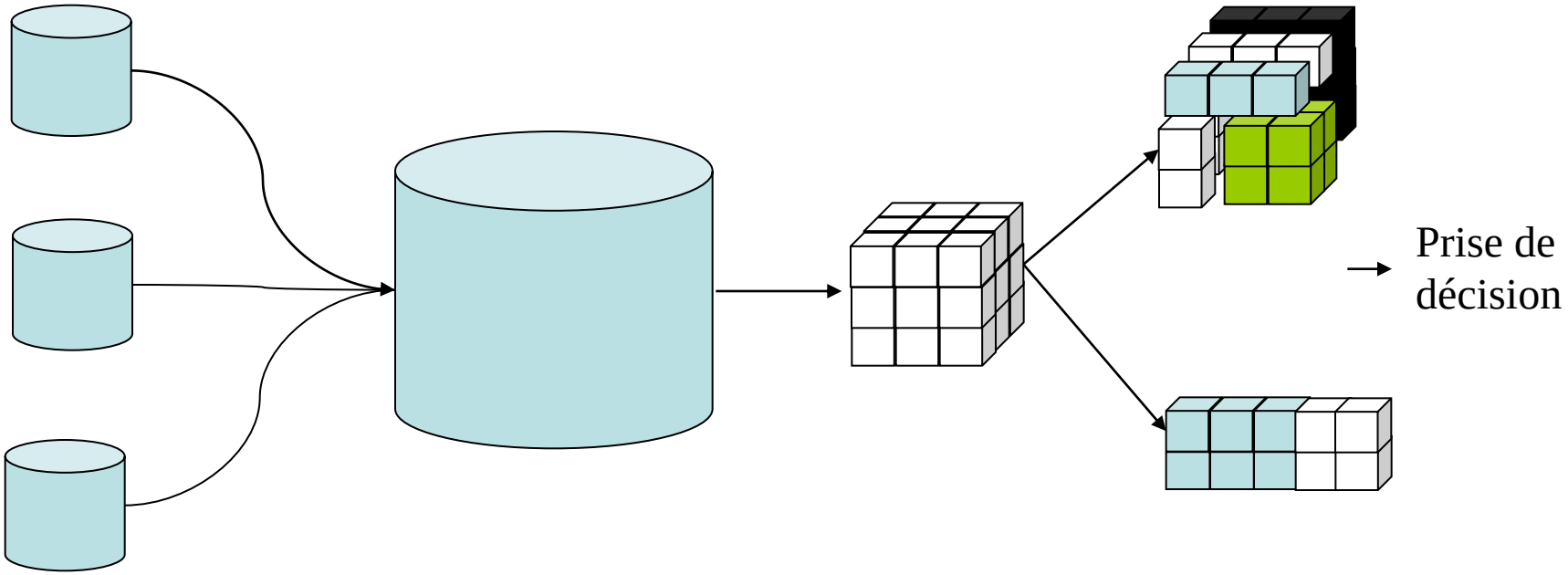
Le contexte


Suivre la performance de l'entreprise
Management
Tableaux de bord (KPI, alertes)
Comprendre en détail l'activité
Analystes
Navigation, exploration dans les données en mode autonome
Analyser les résultats opérationnels
Contributeurs
Reporting structuré


= Reporting
= Analytics
High
Business Value
Low
"What will happen?"
Prediction
"What's happening?"
Monitoring
"Why did it happen?"
Analysis
"What happened?"
Reporting
Static & interactive Reports
Query, Excel, OLAP, Visual discovery
Dashboards, Scorecards
Statistics, data mining, optimization
1980s
1990s
2000s
2010s (mainstream)

Introduction au SI

Le processus de prise de décision



Utilisation

Mailing

- amélioration du taux de réponse

Banque, Assurance

- déterminer les profils client
  - Risque d'un Prêt, Prime plus précise

Commerce

- ciblage de clientèle
- déterminer les promotions
- aménagement des rayons (2 produits en corrélation)

Utilisation

Logistique

- adéquation demande / production

Santé

- épidémiologie (VIH, Amiante, ...)

Econométrie

- prédiction de trafic autoroutier

Ressources Humaines

- adéquation activité / personnel

Déclinaisons métiers du Décisionnel

SPM (Strategic Performance Management)

- Déterminer et contrôler les indicateurs clé de la performance de l'entreprise

FI (Finance Intelligence)

- Planifier, analyse et diffuser l'information financière. Mesurer et gérer les risques.

HCM (Human Capital Management)

- Aligner les stratégies RH, les processus et les technologies. Modéliser la carte des RH (Ressources Humaines)

CRM (Customer Relationship Management)

- Améliorer la connaissance client, Identifier et prévoir la rentabilité client. Accroître l'efficacité du marketing client.

SRM (Supplier Relationship Management)

- Classifier et évaluer l'ensemble des fournisseurs. Planifier et piloter la stratégie Achat.

Rentabilisation

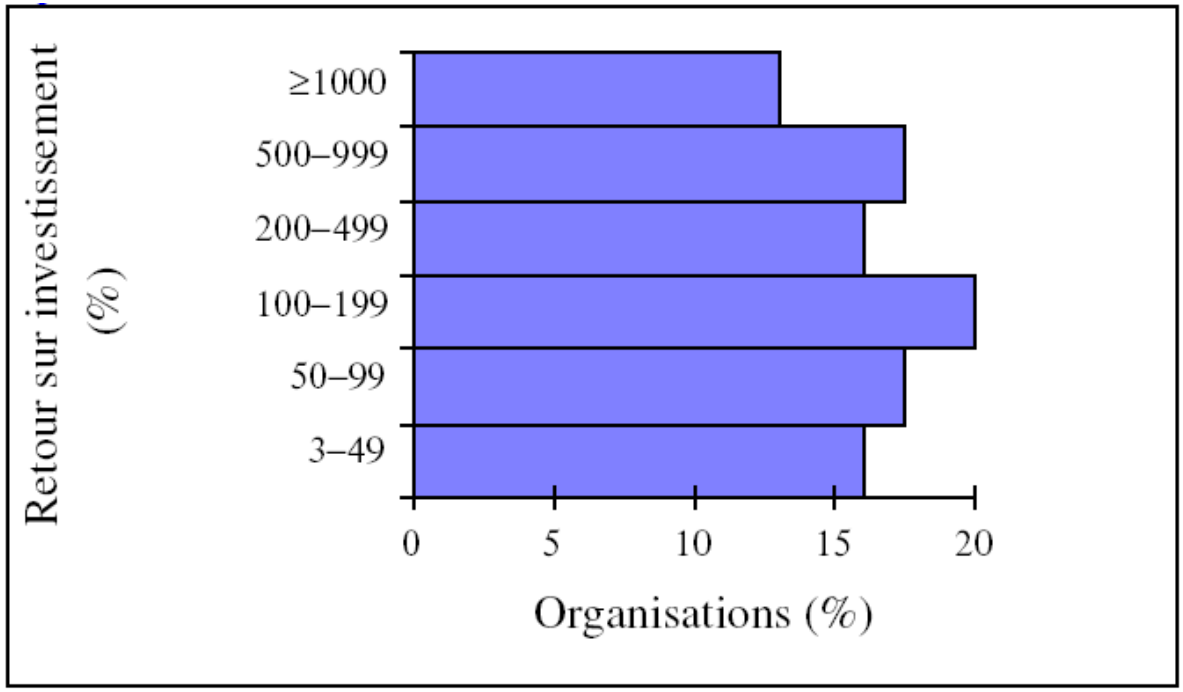
Constat: orientation marché (client, techno,produit)

- Stratégies proactives meilleures que des stratégies réactives

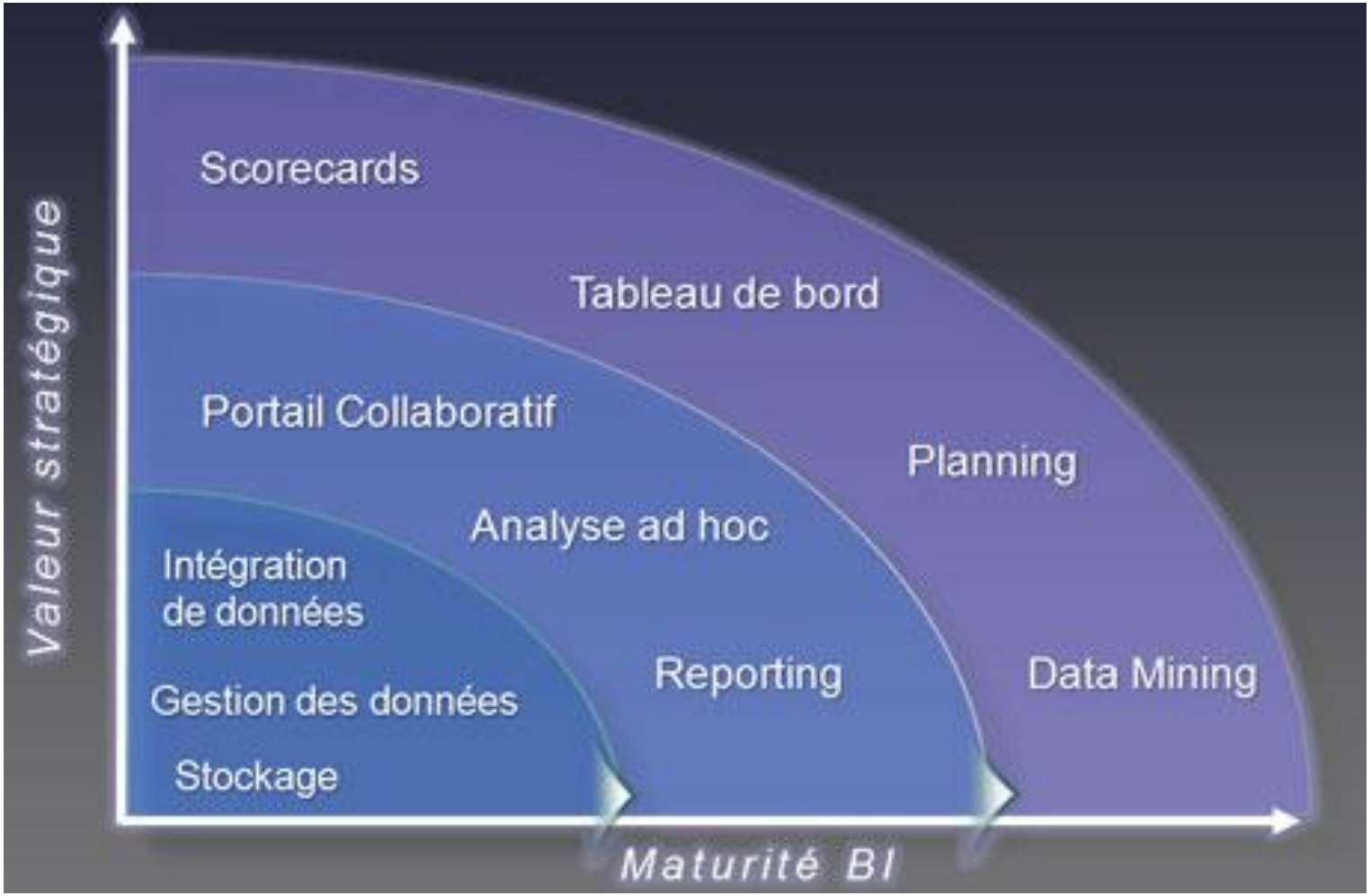
D'après une enquête de l'IDC auprès de 45 organisations ayant un Data Warehouse en fonctionnement *:*

- 90% des entreprises ont un RSI au moins égal à 40%
- 50% ont un RSI supérieur à 160%
- 25% ont un RSI supérieur à 600%

RSI : Retour sur Investissement



Valorisation



Entrepôt de données (Datawarehouse)

L'Entrepôt de Données (Data Warehouse)

Définition de Bill Inmon (1996)

«Le Data Warehouse est une collection de données orientées sujet, intégrées, non volatiles et historisées,organisées pour le support d'un processus d'aide à la décision.»

Principe

- Base de Données utilisée à des fins d'analyse.
- Caractéristiques :
  - orientation sujets («métiers»)
  - données intégrées
  - données non volatiles
  - données datées

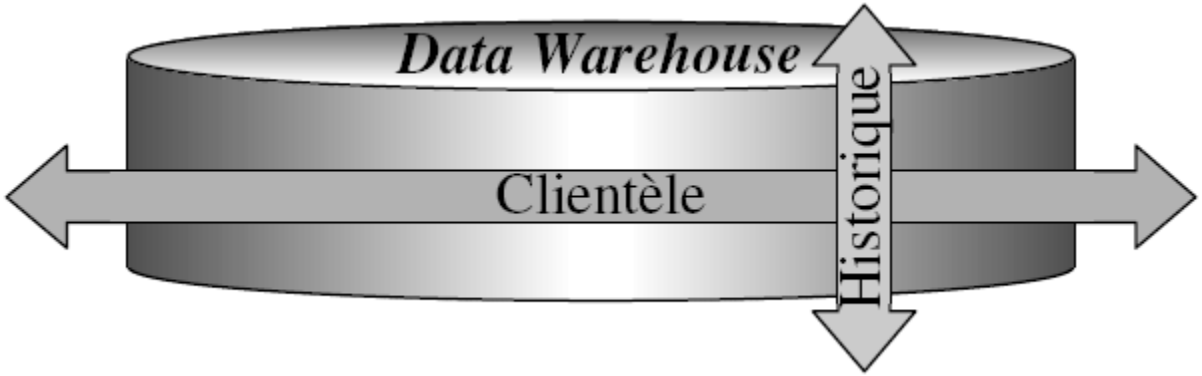
L'Entrepôt de Données

Objectif

- Retrouver une information historique et transversale à l'entreprise

Comment

- Fédérer/Regrouper l'ensemble des données de l'entreprise



Définition

Système de base de données séparé des systèmes (transactionnels) basés sur les données opérationnelles

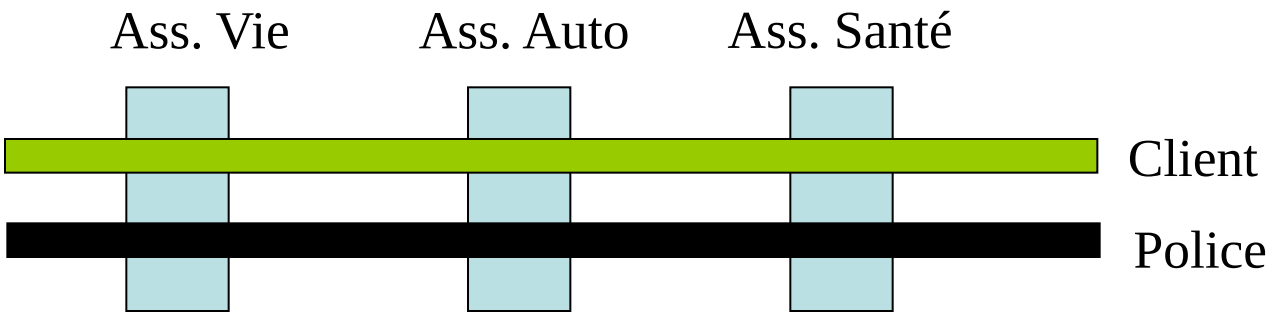
- couvrant un horizon temporel plus grand
- contenant des données plus uniformisées
- optimisés pour répondre à des questions complexes (des gestionnaires et analystes)
- séparé pour des raisons de performances, d ’accès, de format et de qualité

3 variantes:

- **data warehouse** : entrepôt pour toute l’entreprise
- **data mart :** un entrepôt miniature pour une unité de gestion ou un département
- **organizational data store :** techniques d’entrepôt appliquées aux systèmes transactionnels
- + 1 : **virtual data warehouse :** non séparé des bases de données opérationnelles

Les 4 caractéristiques des data warehouse

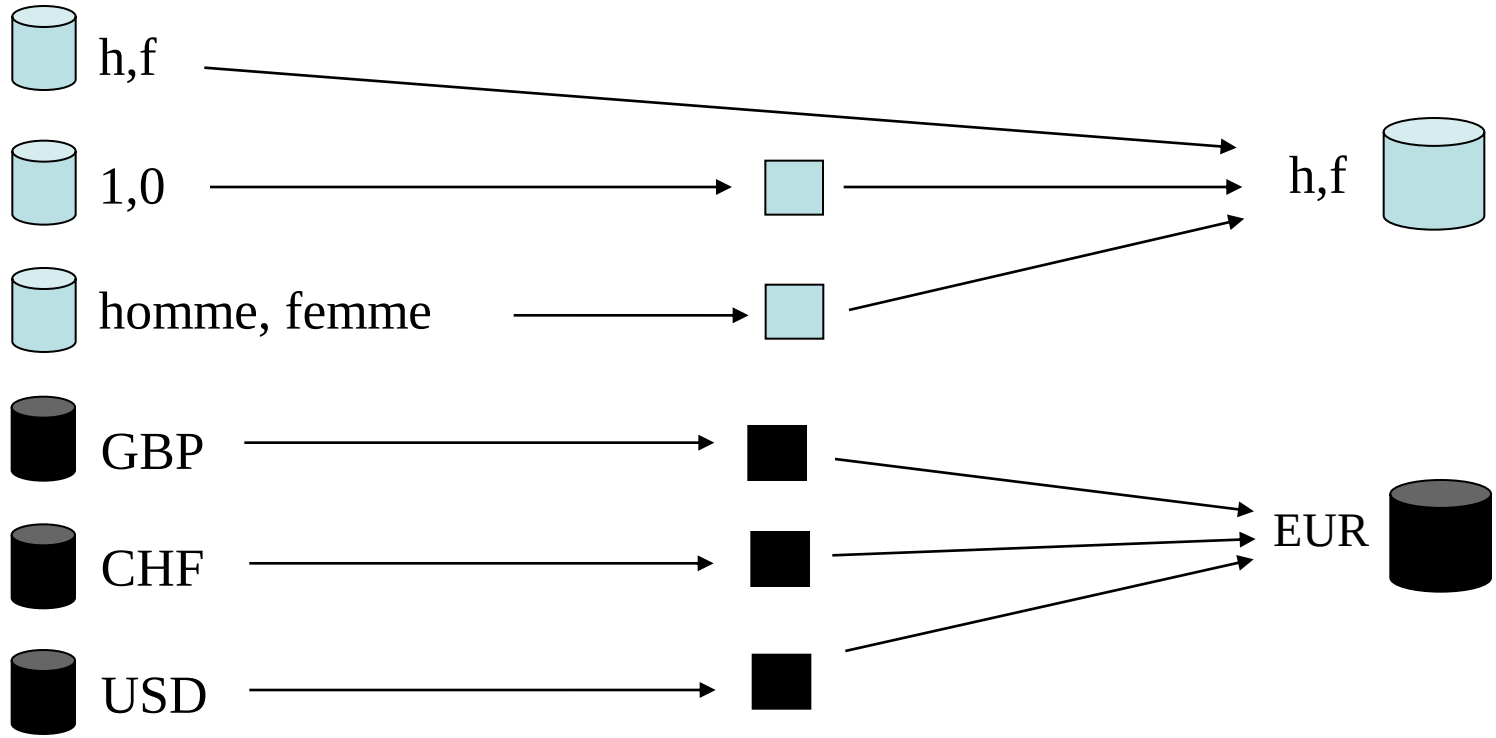
1. Données orientées sujet:
   - Regroupe les informations des différents métiers
   - Ne tiens pas compte de l'organisation fonctionnelle des données



Les 4 caractéristiques des data warehouse

2. Données intégrées:

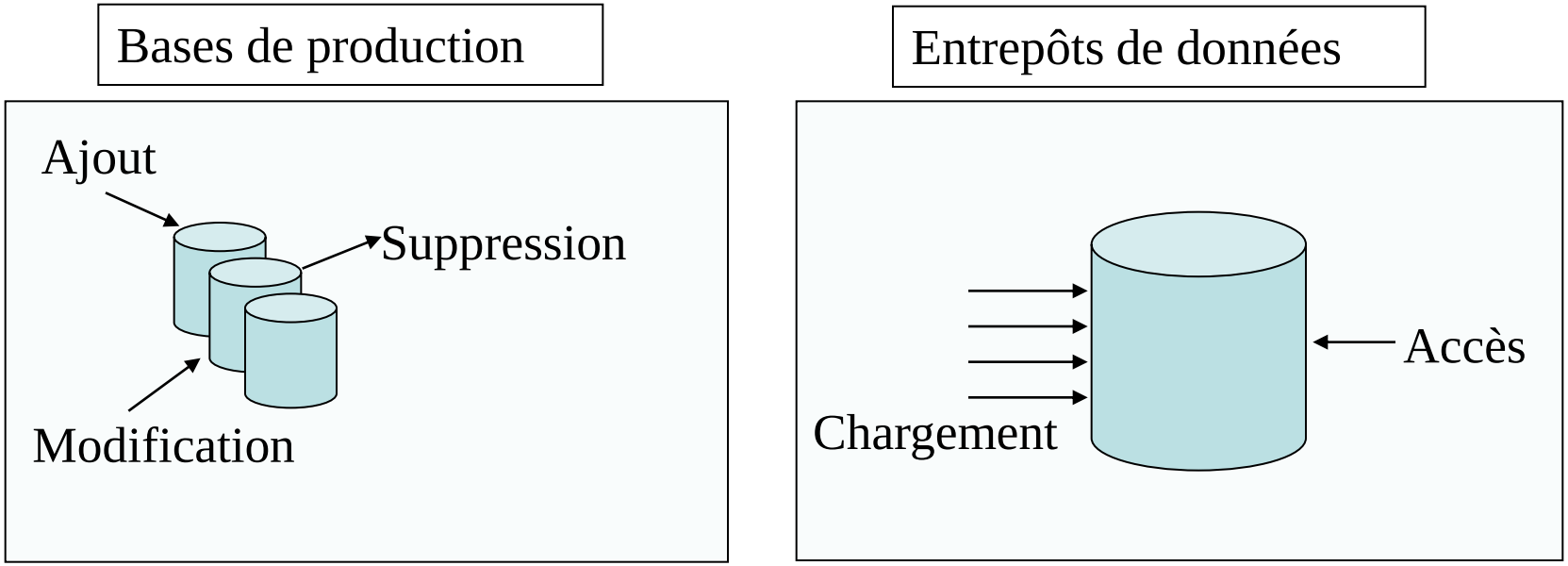
- Normalisation des données
- Définition d'un référentiel unique



Les 4 caractéristiques des data warehouse

3. Données non volatiles

- Traçabilité des informations et des décisions prises
- Copie des données de production



Les 4 caractéristiques des data warehouse

4. Données datées

- Les données persistent dans le temps
- Mise en place d'un référentiel temps

Base de production

Image de la base en Mai 2005

Répertoire

Nom	Ville
Dupont	Paris
Durand	Lyon

Image de la base en Juillet 2006

Répertoire

Nom	Ville
Dupont	Marseille
Durand	Lyon

Entrepôt de données

Calendrier

Code	Année	Mois
1	2005	Mai
2	2006	Juillet

Répertoire

Code	Année	Mois
1	Dupont	Paris
1	Durand	Lyon
2	Dupont	Marseille

Entrepôts vs BDR

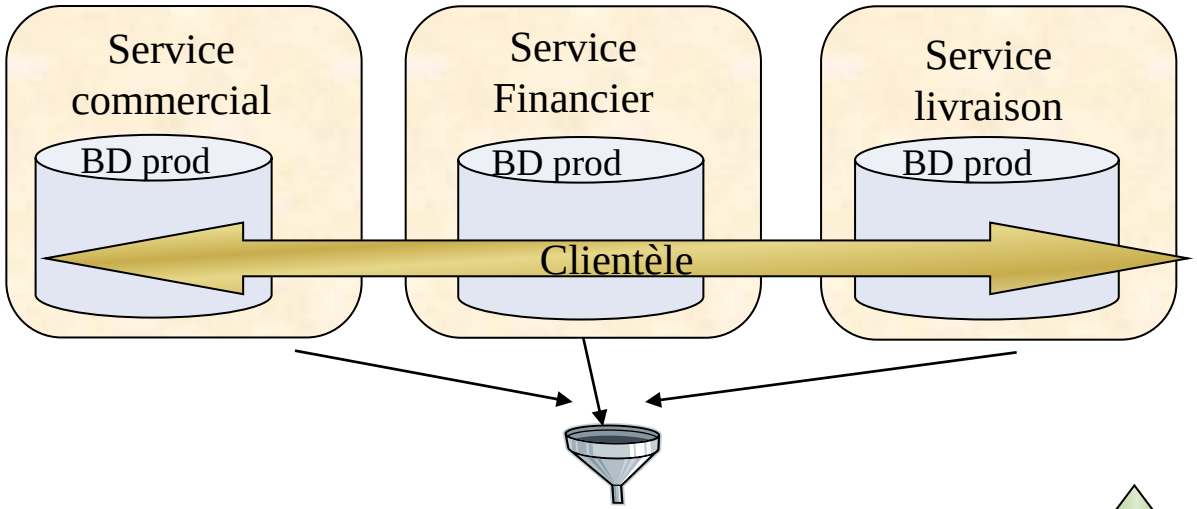
BD opérationelle :

- Flux, Temps réel . . .
- cohérence, requêtes rapides

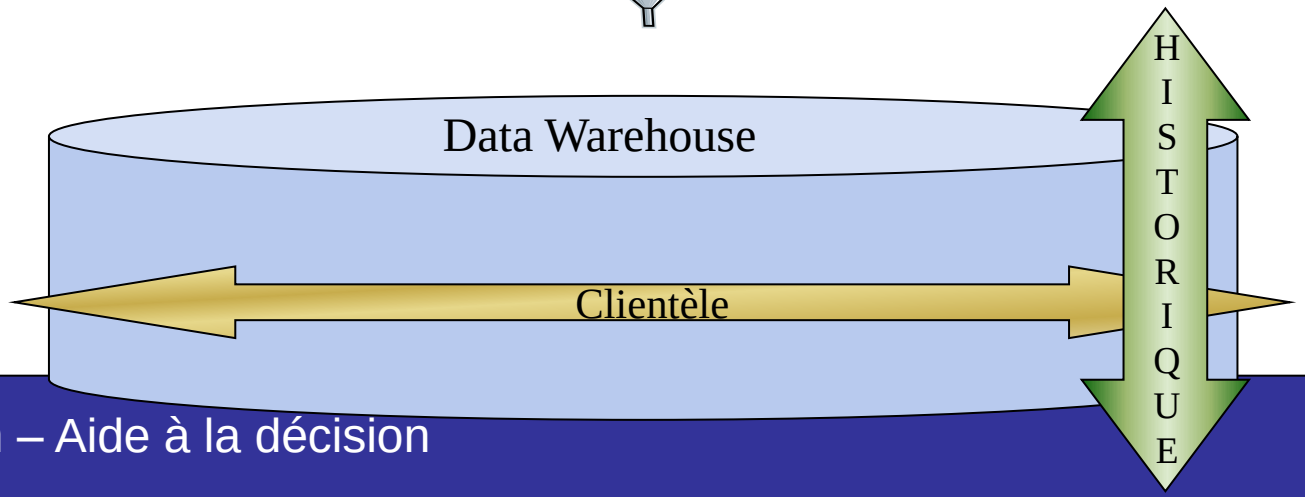
BD décisionnelle :

- Vue historique.
- Sources multiples.
- Données standardisées.

OLTP: On-Line Transactional Processing



OLAP: On-Line Analitical Processing



Données

Données opérationnelles	Données décisionnelles
Orientées application, détaillées, précises au moment de l'accès	Orientée activité (thème, sujet), condensées, représentes des données historiques
Mise à jour interactive possible de la part des utilisateurs	Pas de mise à jour interactive de la part des utilisateurs
Accédées de façon unitaires par une personne à la fois	Utilisées par l'ensemble des analystes, gérées par sous-ensemble
Cohérence atomique	Cohérence globale
Haute disponibilité en continu	Exigence différente, haute disponibilité ponctuelle
Uniques (pas de redondance en théorie)	Peuvent être redondantes
Structure statique, contenu variable	Structure flexible
Petite quantité de données utilisées par un traitement	Grande quantité de données utilisée par les traitements
Réalisation des opérations au jour le jour	Cycle de vie différent
Forte probabilité d'accès	Faible probabilité d'accès
Utilisées de façon répétitive	Utilisée de façon aléatoire

Ethique

Protection de la vie privée

les 8 principes de l ’OCDE

- pour mettre en place une législation dans un pays

Limitation des données privées
Qualité des données
Objectifs clairs
Limitation de l'usage
Sécurité assurée
Ouverture (commission)
Participation des fichés
Responsabilité

- *données médicales, religieuses …*
- *garantie par l ’entreprise*
- *l’objectif de la collecte est précisé*
- *les données ne servent qu ’à cet objectif*
- *garantie par l ’entreprise*
- *vérification possible*
- *les fichés ont accès et peuvent réagir*
- *des entreprises*

Objectifs

I Rendre les données facilement accessibles

- Lecture facile.
- Manipulation aisée, outils conviviaux.
- Rapidité . . .? ? ?

II Présentation cohérente

- Données nettoyées, vérifiées, crédibles.
- Codage et représentation documentés.
- Assurer la qualité des données.

III Architecture évolutive

- Compatible avec les nouvelles requêtes.
- Résistance aux changements.
- Compatibilité ascendante.
- Modifications documentées.

IV L'entrepôt de données doit servir à la prise de décision Il doit être accepté par les utilisateurs . . . et utilisé !

Les utilisateurs peuvent se passer de l'entrepôt de données. . .pas de l'information opérationnelle. . .

Travail du concepteur

- Ecouter les utilisateurs : besoins, décisions . . .
- Cibler les « meilleurs » utilisateurs.
- Sélectionner les données pertinentes.
- Concevoir des outils de visualisation/interrogation simples.
- Contrôler la validité des données présentées.
- Enrichir constamment la base.
- Faire de la pub !

Recueil des besoins

OBJECTIF PRINCIPAL

- Qu'attendez-vous principalement du Data Warehouse ?

DECISIONS

- Quelles décisions avez-vous à prendre ? (Quoi ?)
- Quels sont les critères qui influencent la prise de décision ? (Comment ?)
- Dans quel(s) but(s) les décisions sont-elles prises ? (Pourquoi ?)

DIFFICULTES ACTUELLES

- Quelles sont les difficultés actuellement rencontrées dans la prise de décision, difficultés en rapport avec les données ?
  - précision des données (détails, actualisation, vérification)
  -
  - synthèse des données (regroupements)
  - évolution (temps)
  - autres...

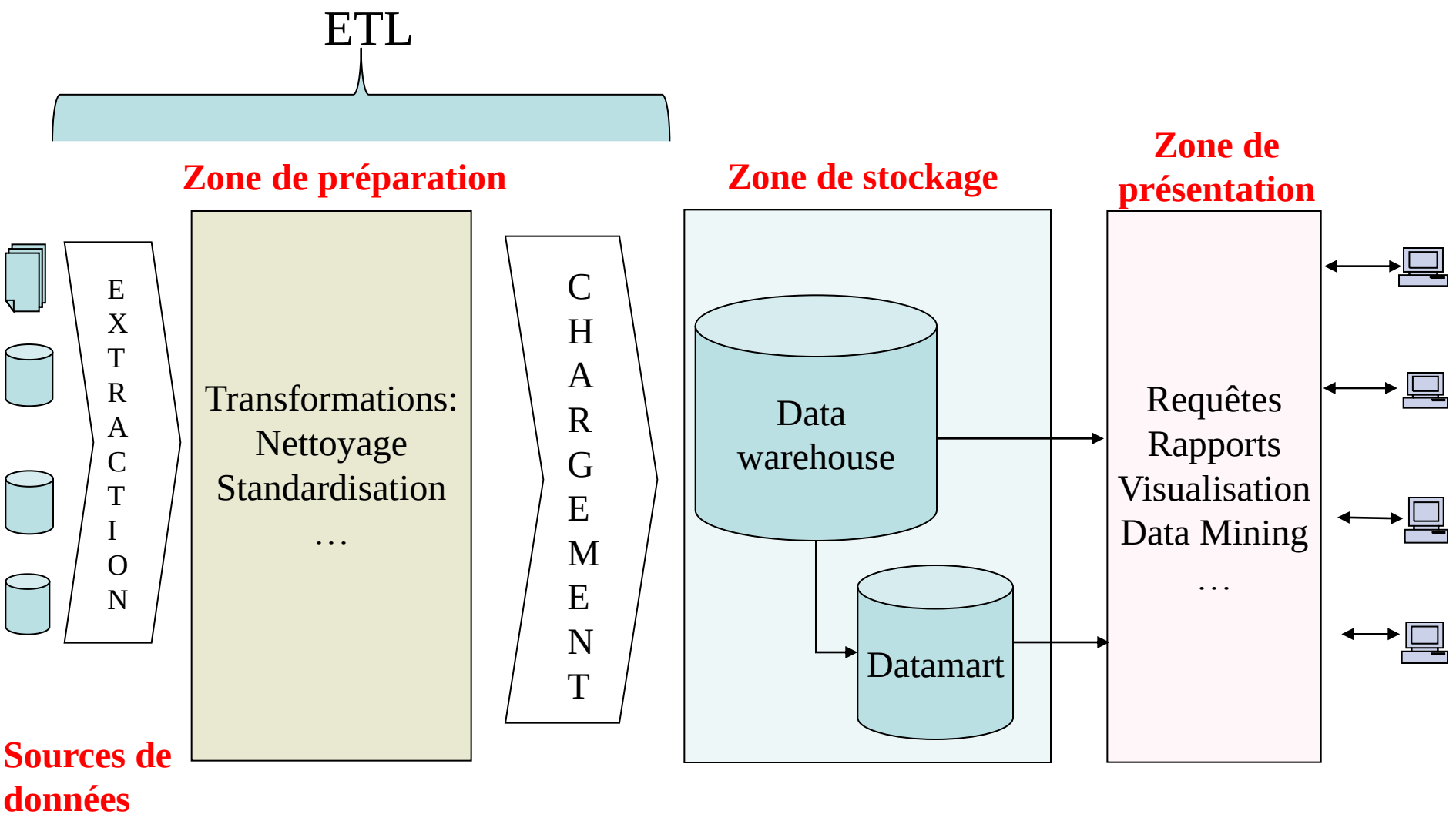
ACTUALISATION DES INFORMATIONS

- Quels sont les besoins concernant la fréquence de mise à jour des informations proposées par le Data Warehouse ?

PRESENTATION DES INFORMATIONS

- » Quelles sont vos préférences dans la présentation des informations
- tableaux, graphiques, ?
- » Type de graphiques : barres-graphes, "camemberts", nuages de points ... ?
- » Existe-t-il une présentation actuelle ou habituelle à conserver ?

Architecture générale



4 niveaux de construction

1. préparation des données
2. intégration des données
3. agrégation des données
4. personnalisation des données

Alimentation/ mise à jour de l'entrepôt

Entrepôt mis à jour régulièrement

Besoin d'un outil permettant d'automatiser les chargements dans l'entrepôt

➡ Utilisation d'outils ETL (Extract, Transform, Load)

Définition d'un ETL

- Offre un environnement de développement
- Offre des outils de gestion des opérations et de maintenance
- Permet de découvrir, analyser et extraire les données à partir de sources hétérogènes
- Permet de nettoyer et standardiser les données
- Permet de charger les données dans un entrepôt

ETL

support et/ou automatisation des tâches suivantes :

- Extraction : méthode d'accès aux différentes sources
- Nettoyage : recherche et résolution des inconsistances dans les sources
- Analyse : e.g., détection de valeurs non valides ou inattendues
- Transformation : entre différents formats, langages, etc.
- Chargement : alimentation de l'ODS

Applications sources

Extraire les informations (pertinentes, utiles) du flux de données de l'entreprise.

- Données temps réel → c'est l'entrepôt qui se charge de l'historique.
- Sources multiples, disjointes, redondantes . . .

Préparation

jusqu’`a 80 % du temps de développement d’un entrepôt

I Les données arrivent à l' état brut, elles sont :

- Extraites :
  - Sélectionner les données, les champs
  - Définir un protocole (fréquence, heure . . .)

-II- Les données sont transformées :

- Nettoyées
  - Cohérence
  - Valeurs manquantes
  - Conversion d‘échelle ou de types.
  - Doublons.

III Les données sont ensuite chargées dans les entrepôts proprement dits.

- Intégrées à la base dimensionnelle.
- Mises à la disposition des utilisateurs (les prévenir).
- Architecture de la base très simple : les seuls traitements sont des tris et des traitements séquentiels (lecture de la base . . .)

Extraction

Extraire des données des systèmes de production

Dialoguer avec différentes sources:

- Base de données,
- Fichiers,
- Bases propriétaires

Utilise divers connecteurs :

- ODBC,
- SQL natif,
- Fichiers plats

Nettoyage et Transformation

5 à 30 % des données des BD commerciales sont erronées

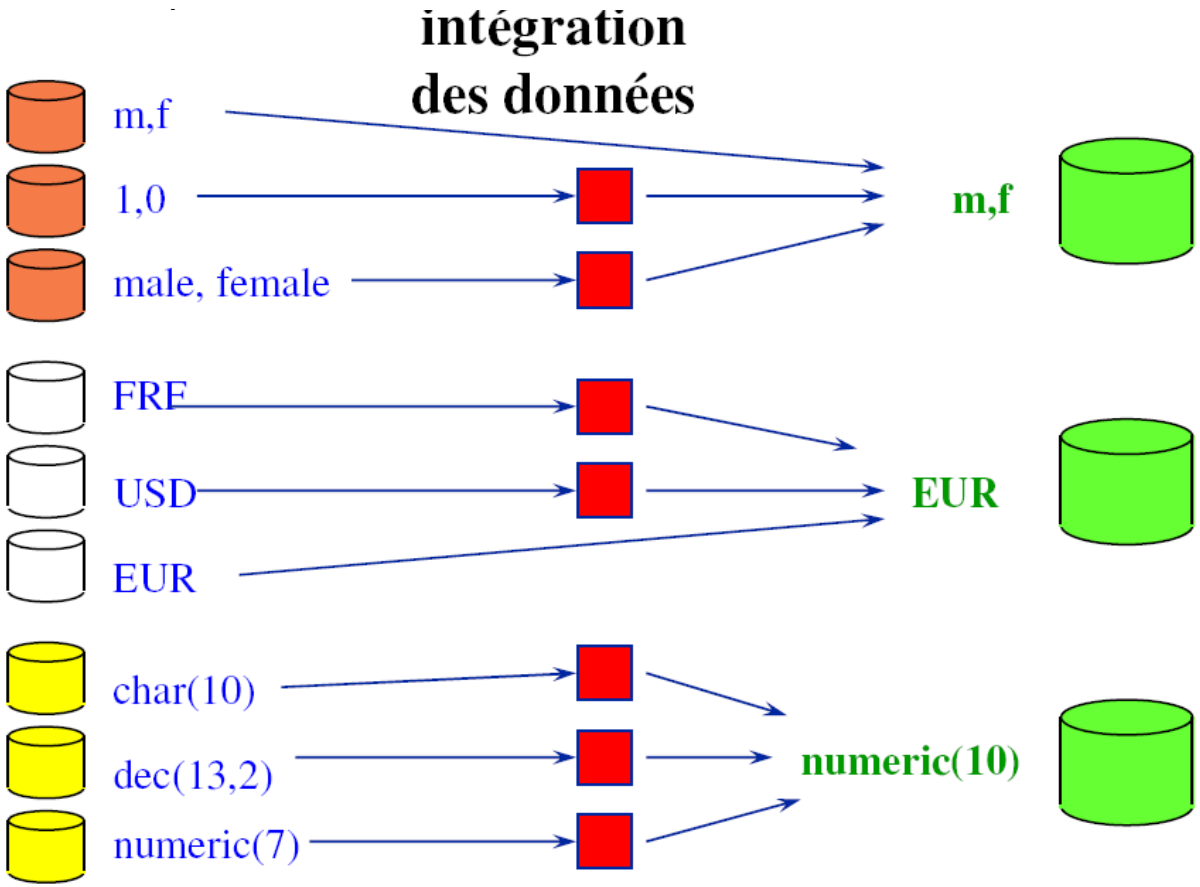
une centaine de type d'inconsistances ont été répertoriées

but : résoudre le problème de consistance des données au sein de chaque source

Nettoyage et Transformation

Définition de table de règles :

valeur source	*remplacé par*	Valeur cible
Mr		M
monsieur		M
Masculin		M
M		M
Msieur		M

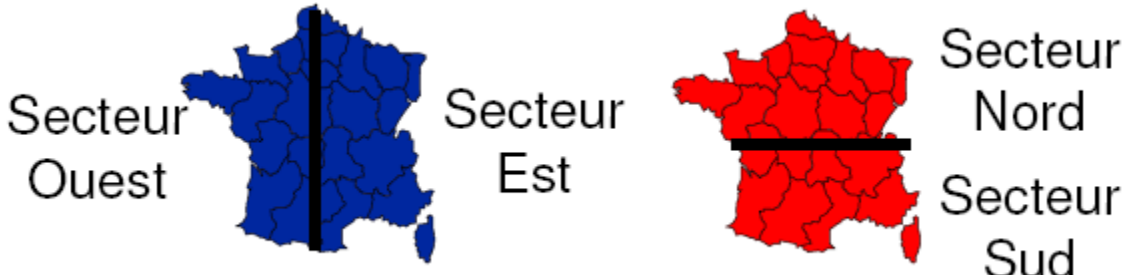


Transformation

Existence de plusieurs sources

non conformité des représentations

découpages géographiques différents



codage des couleurs

identification des produits différents

¾ produits en vrac

difficulté de comparaison des sources de données

Mise en conformité nécessaire

Nettoyage

un outil de nettoyage comprend :

- des fonctions d’analyse
- des fonctions de normalisation
- des fonctions de conversion
- des dictionnaires de synonymes ou d’abréviations

Normalisation, conversion, dictionnaires, …

Définition de table de règles

remplacer valeur	→par
Mr	→M
Monsieur	→ M
Mnsieur	→ M
masculin	→ M
M	→ M
Msieur	→ M
M.	→ M
Monseur	→ M

utilisation d'expression régulière, suppression de doublons, de valeur nulle, …

Transformation des données

Objectifs :

Suppression des incohérences sémantiques entre les sources pouvant survenir lors de l'intégration :

des schémas :

- problème de modélisation : différents modèles de données sont utilisés
- problèmes de terminologie : un objet est désigné par 2 noms différents, un même nom désigne 2 objets différents
- incompatibilités de contraintes : 2 concepts équivalents ont des contraintes incompatibles
- conflit sémantique : choix de différents niveaux d'abstraction pour un même concept
- conflits de structures : choix de différentes propriétés pour un même

Concept

- conflits de représentation : 2 représentations différentes choisies pour les mêmes propriétés d'un même objet
- des données :
  - Equivalence de champs

Chargement

définitions de vues relationnelles sur les données sources

matérialisation des vues dans l'entrepôt

Plus

- Tris
- consolidations (pré-agrégation)
- Indexation
- partitionnement des données
- enregistrement de méta-données
- ...

Meta-données (Metadata)

informations présentes dans l'entrepôt

- données source
- données dérivées, dimensions, hiérarchies
- contraintes d'intégrités
- schéma de l'entrepôt
- indexes, partitions
- requêtes prédéfinies

...

Meta-données (Metadata)

- informations d'administration
- règles de nettoyage, transformation, extraction
- politique de rafraîchissement
- sécurité
- monitoring, statistiques
- traçage des données
- ...

Meta-données (Metadata

chaque composant de l'entrepôt

- fournit des méta-données
- doit connaitre celles des autres composants
- doit savoir où ces méta-données sont situées

une BD est dédiée aux méta-données

Métadonnées

C'est la documentation de l'architecture de l'entrepôt :

- Format des entrées.
- Règles de sélection et de transformation des données entrantes.
- Protocole de chargement.
- Format des tables.
- Statistiques d'usage, d'accès . . .

L'entrepôt de données devient un nouveau secteur d'activité de l'entreprise

Type d'information	Signification
Sémantique	Que signifie la donnée
Origine	D'où vient-elle, où, par qui est-elle créée ou mise à jour
Règle de calcul	Règle de calcul, de gestion
Règle d'agrégation	Périmètre de consolidation
Stockage, format	Où, comment est-elle stockée, sous quel format
Utilisation	Programmes informatiques qui l'utilisent, Machines : comment et sur lesquelles, à disposition, Temps de conservation

Méta-données

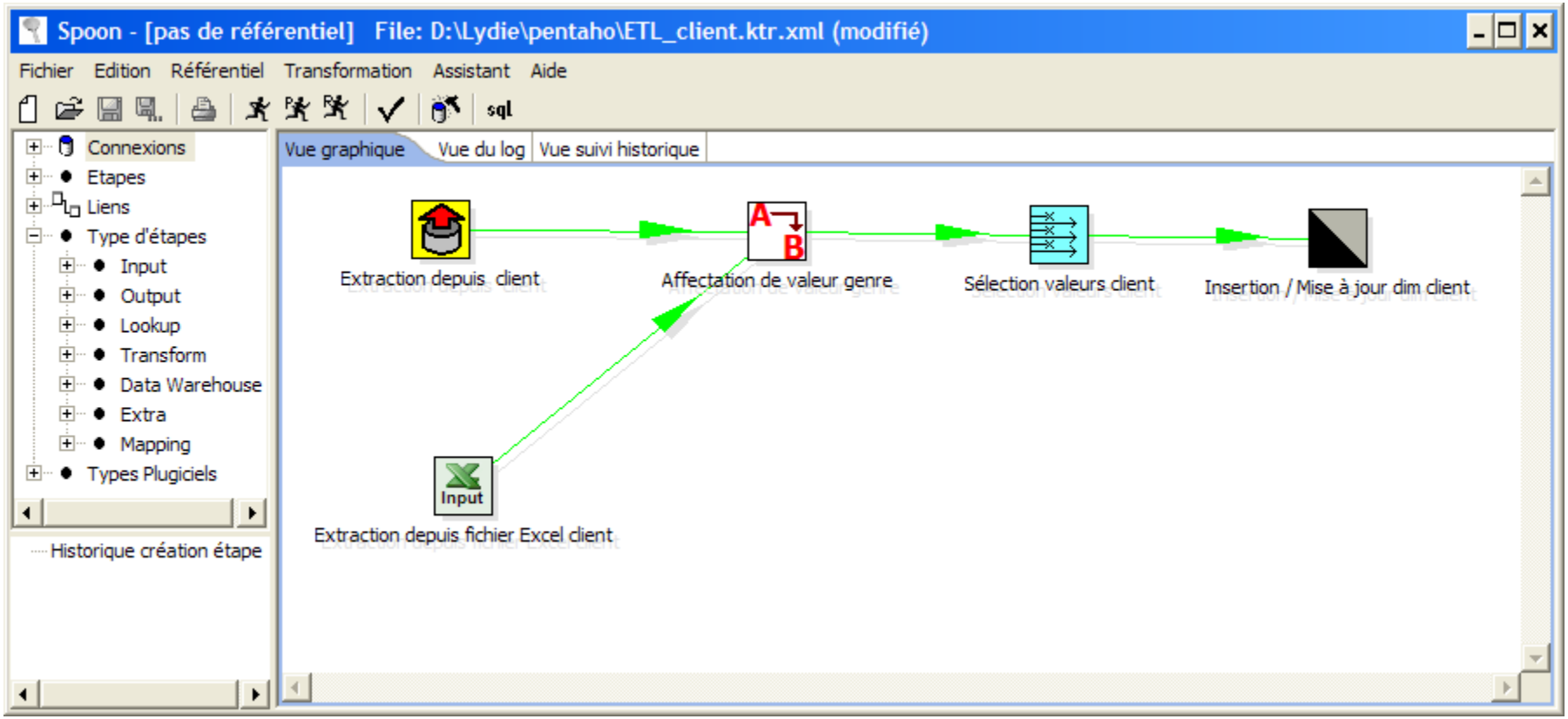
Types de lien	Signification
Domaines, sujets	Chaque donnée va être indexée par sujet ou domaine
Structure organisationnelle, structure géographique	Une donnée peut avoir des sens légèrement différents selon la personne qui la manipule
Concepts génériques	Exemple : notion de produit se déclinant en lignes de produits, services,...
Applications, programmes	Donnée manipulée par une ou plusieurs applications ou programmes
Tables, colonnes	Donnée située dans une ou plusieurs colonnes, tables et bases de données
Sites, machines	Localisation physique de la donnée

Méta-données

Exemple de standard : Common Warehouse Metamodel

- proposé par l'OMG
- basé notamment sur UML, XML
- conçu par IBM, Unisys, NCR, Oracle, Hyperion, ...

Aperçu d'un ETL



Présentation

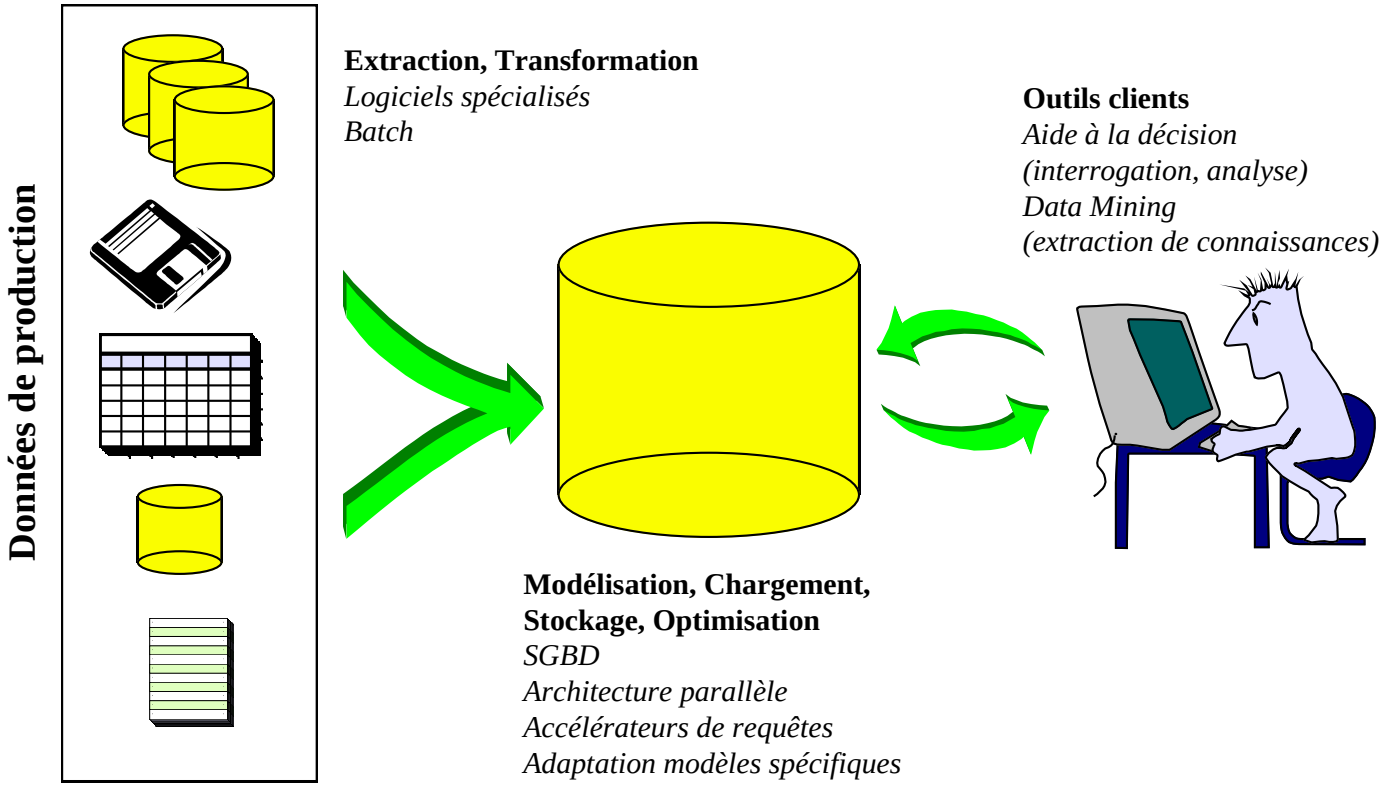
Zone où sont stockées les données.

Organisée en marché d'information (datamart).

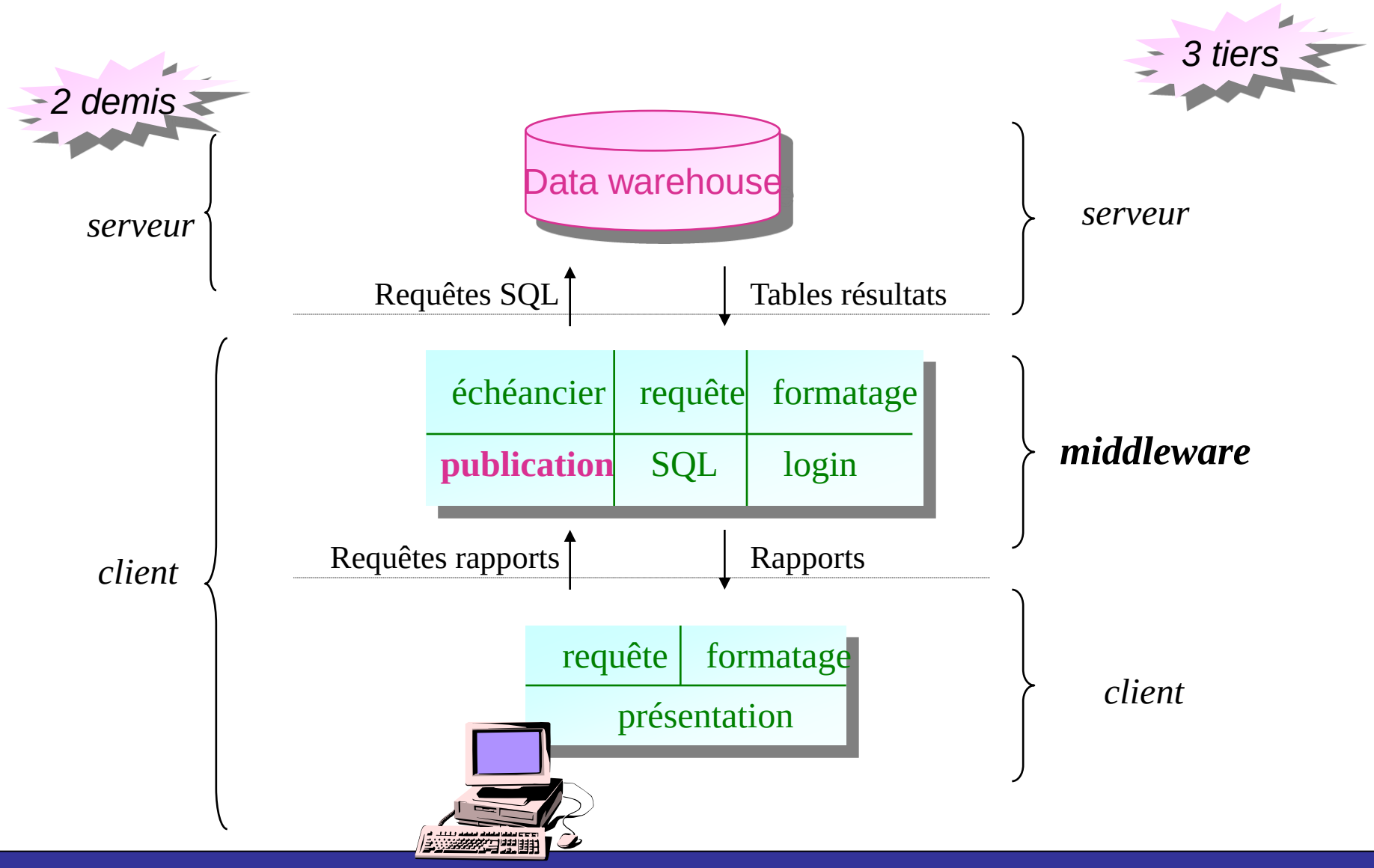
Un datamart = un secteur de l'entreprise

Datamarts interconnectés → Standardisation des données

Réalisation plus simple, incrémentale.


Données de production
Extraction, Transformation
Logiciels spécialisés
Batch
Outils clients
Aide à la décision
(interrogation, analyse)
Data Mining
(extraction de connaissances)
Modélisation, Chargement,
Stockage, Optimisation
SGBD
Architecture parallèle
Accélérateurs de requêtes
Adaptation modèles spécifiques

Architecture



Datamart

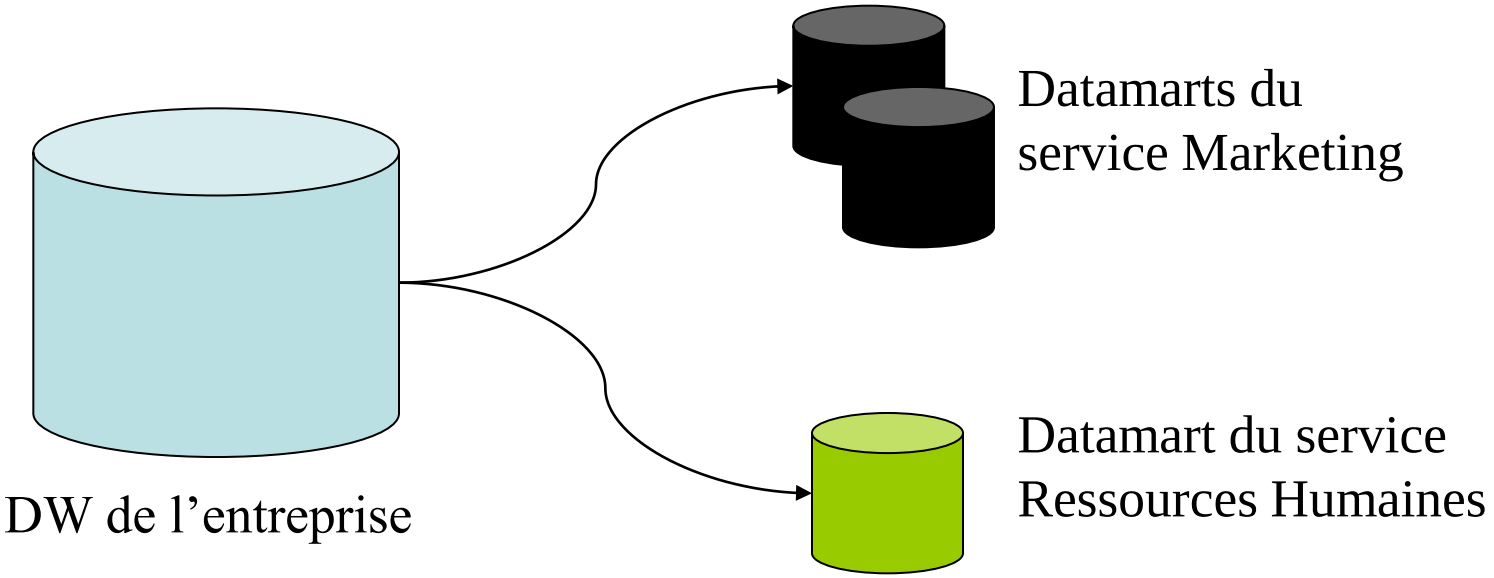
	Data Warehouse	Data Mart
Cible utilisateur	Toute l'entreprise	Département
Implication du service informatique	Elevée	Faible ou moyen
Base de données d'entreprise	SQL type serveur	SQL milieu de gamme, bases multidimensionnelles
Modèles de données	A l'échelle de l'entreprise	Département
Champ applicatif	Multi sujets, neutre	Quelques sujets, spécifique
Sources de données	Multiples	Quelques unes
Stockage	Base de données	Plusieurs bases distribuées
Taille	Centaine de GO et plus	Une à 2 dizaines de GO
Temps de mise en place	9 à 18 mois pour les 3 étapes	6 à 12 mois (installation en plusieurs étapes)
Coût	> 1millions €	100.000 à 0.5 million d'€
Matériel	Unix	NT, petit serveur Unix

Datamart

Sous-ensemble d'un entrepôt de données

Destiné à répondre aux besoins d'un secteur ou d'une fonction particulière de l'entreprise

Point de vue spécifique selon des critères métiers



Intérêt des datamart

Nouvel environnement structuré et formaté en fonction des besoins d'un métier ou d'un usage particulier

Moins de données que DW

- Plus facile à comprendre, à manipuler
- Amélioration des temps de réponse

Utilisateurs plus ciblés: DM plus facile à définir

Organisation d'un datamart

Principes généraux

- Système de fichiers plats :
  - Table des faits.
  - Tables des dimensions
- Cube de données :
  - développer selon toutes les dimensions sur lesquelles on veut pouvoir étudier les données.
  - Fixer le niveau de détail le plus fin possible (choix irréversible !)
  - Éventuellement prévoir des résumés.

Accès aux données

Outils permettant de consulter les données des datamarts :

- Requêtes (interface, SQL . . .)
- Rapports (standards, ad hoc . . .)
- Modélisation : catégorisation (clustering) Classification.
- Prévision.

Métadonnées

C'est la documentation de l'architecture de l'entrepôt :

- Format des entrées.
- Règles de sélection et de transformation des données entrantes.
- Protocole de chargement.
- Format des tables.
- Statistiques d'usage, d'accès . . .

L'entrepôt de données devient un nouveau secteur d'activité de l'entreprise

Type d'information	Signification
Sémantique	Que signifie la donnée
Origine	D'où vient-elle, où, par qui est-elle créée ou mise à jour
Règle de calcul	Règle de calcul, de gestion
Règle d'agrégation	Périmètre de consolidation
Stockage, format	Où, comment est-elle stockée, sous quel format
Utilisation	Programmes informatiques qui l'utilisent, Machines : comment et sur lesquelles, à disposition, Temps de conservation

Méta-données

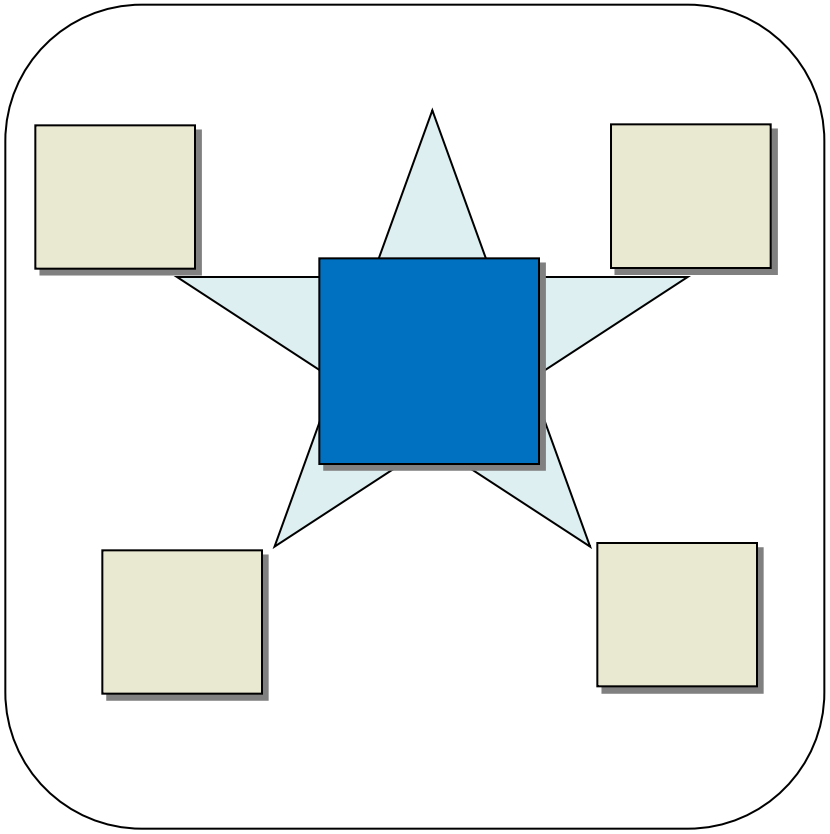
Types de lien	Signification
Domaines, sujets	Chaque donnée va être indexée par sujet ou domaine
Structure organisationnelle, structure géographique	Une donnée peut avoir des sens légèrement différents selon la personne qui la manipule
Concepts génériques	Exemple : notion de produit se déclinant en lignes de produits, services,...
Applications, programmes	Donnée manipulée par une ou plusieurs applications ou programmes
Tables, colonnes	Donnée située dans une ou plusieurs colonnes, tables et bases de données
Sites, machines	Localisation physique de la donnée

Niveau conceptuel

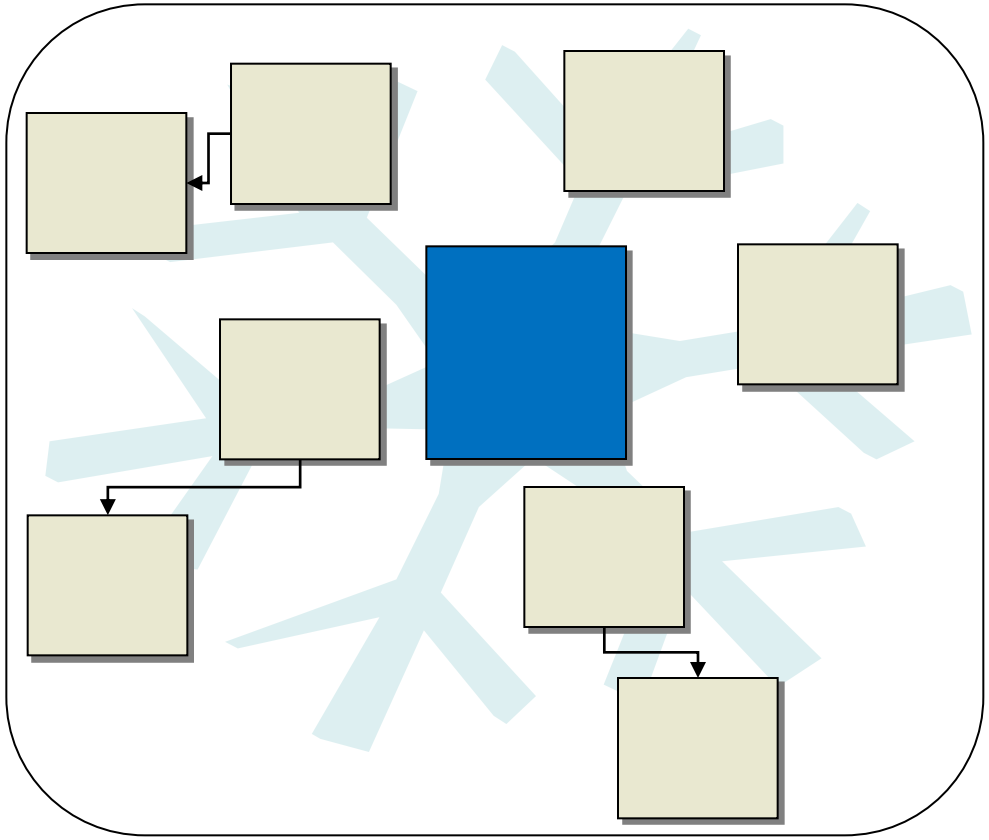
- schéma en étoile (star schema)
- schéma en flocon (snowflake schema)
- constellation de faits (fact constellation)

le schéma en étoile est souvent utilisé pour l'implantation physique

Les types de modèles

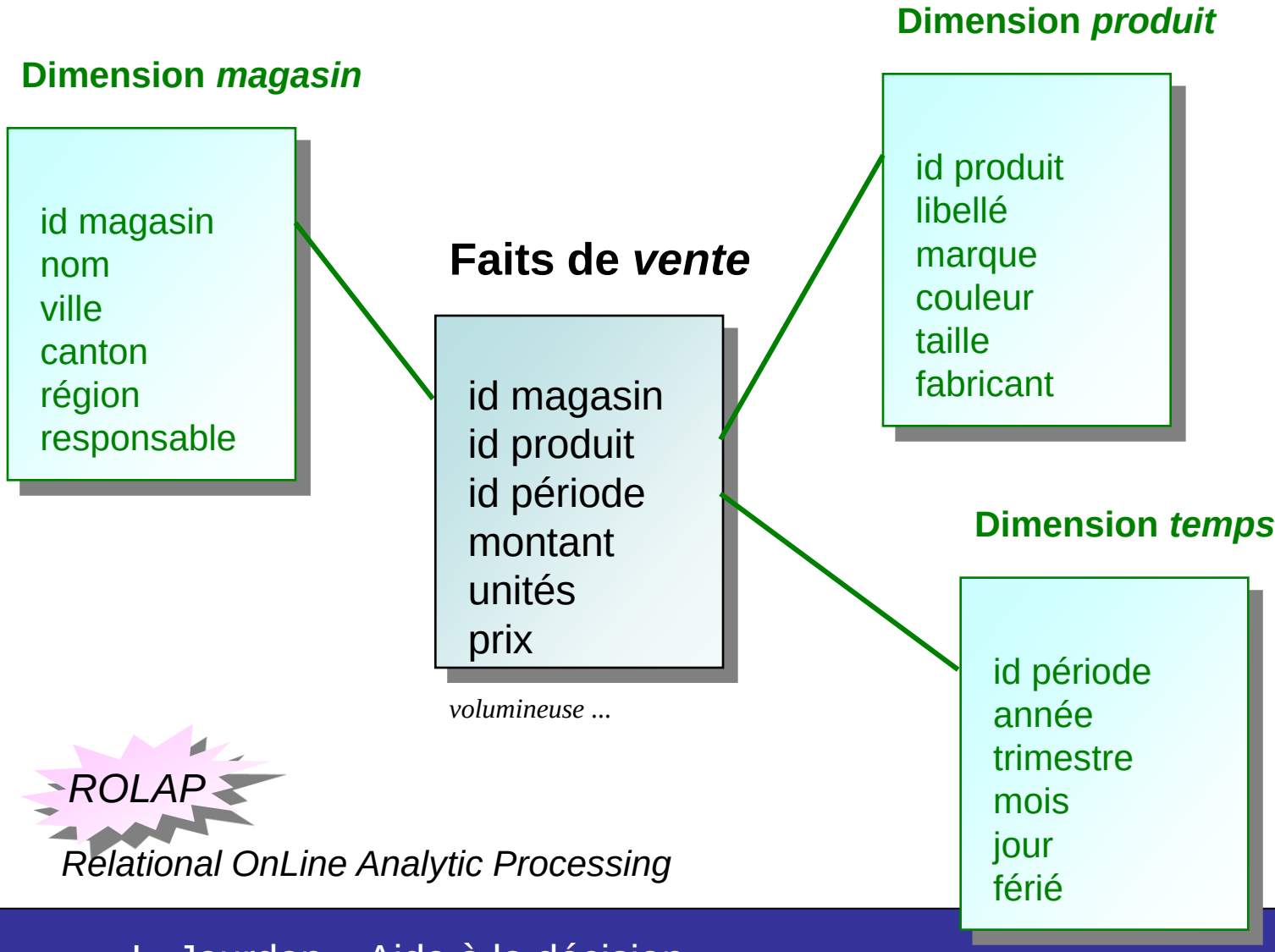


Modèle en étoile



Modèle en flocon

Multi-dimensionnel : *Structure en étoile*



ROLAP

Relational OnLine Analytic Processing

Le modèle en étoile :

Une (ou plusieurs) table(s) de faits : identifiants des tables de dimension ; une ou plusieurs mesures .

Plusieurs tables de dimension : descripteurs des dimensions.

Une granularité définie par les identifiants dans la table des faits.

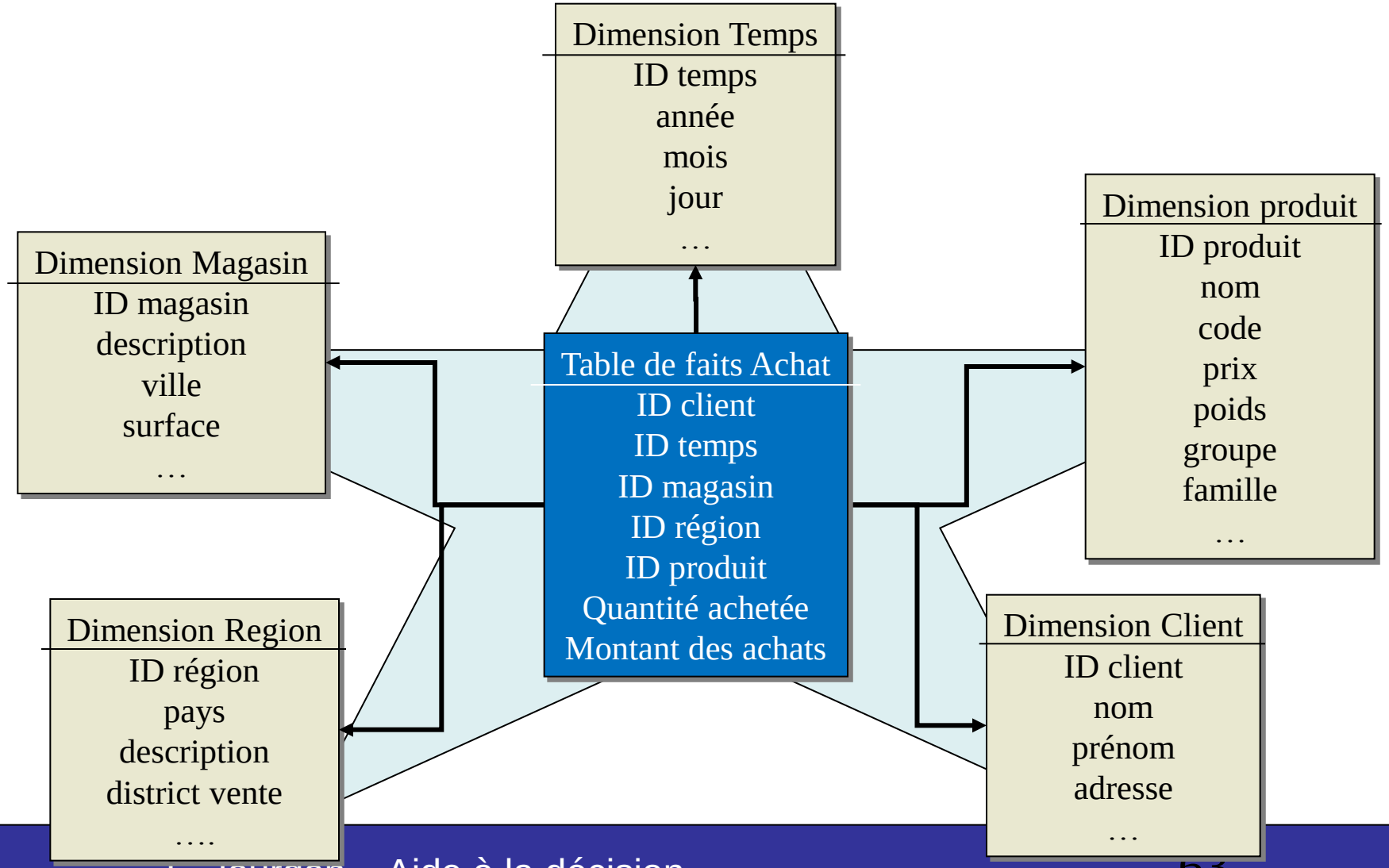
Avantages :

- Facilite de navigation
- Performances : nombre de jointures limite ; gestion des données creuses.
- Gestion des agrégats
- Fiabilité des résultats

Inconvénients :

- Toutes les dimensions ne concernent pas les mesures
- Redondances dans les dimensions
- Alimentation complexe.

Modèle en étoile



structure simple utilisant le modèle entité-relation

- une entité centrale (table des faits)
  - objets de l'analyse
  - taille très importante
  - beaucoup de champs
- des entités périphériques (tables de dimensions)
  - critères de l'analyse
  - taille peu importante
  - peu de champs

Modèle en flocon

Le modèle du DW doit être simple à comprendre.

On peut augmenter sa lisibilité en regroupant certaines dimensions.

On définit ainsi des hiérarchies.

Celles-ci peuvent être géographiques ou organisationnelles.

Lorsque les tables sont trop volumineuses

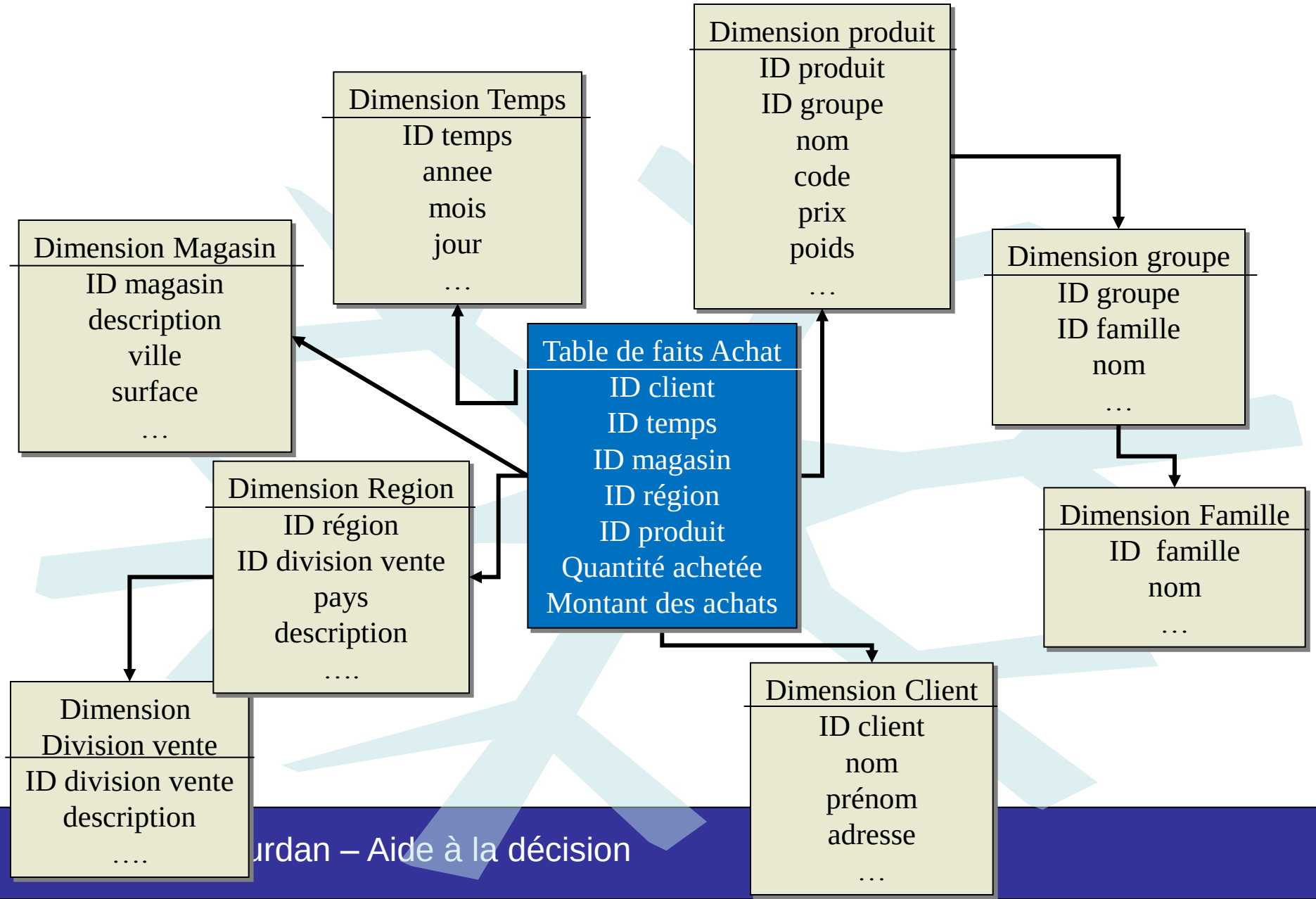
Avantages :

- réduction du volume,
- permettre des analyses par pallier (drill down) sur la dimension hiérarchisée.

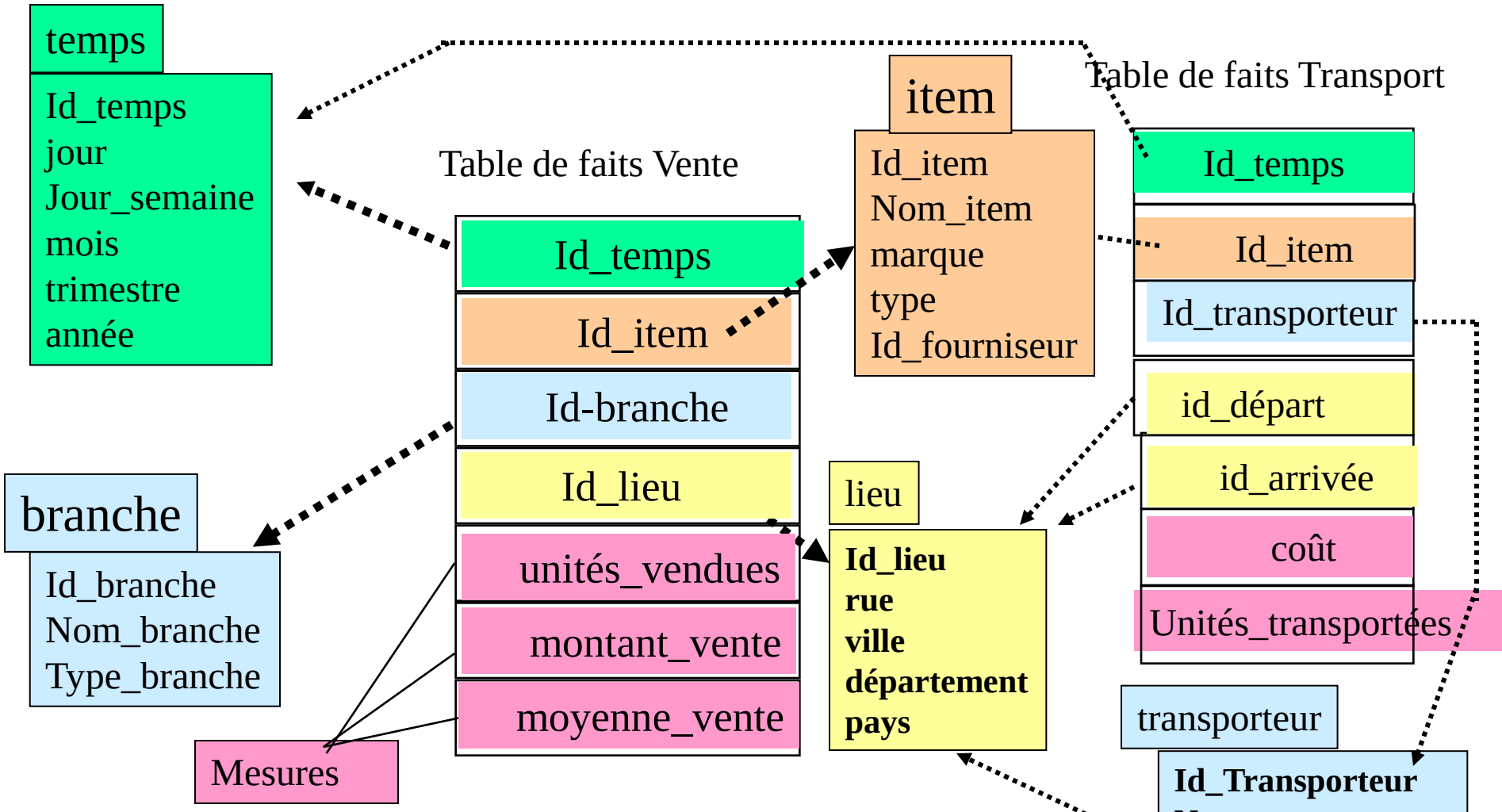
Inconvénients :

- navigation difficile ;
- nombreuses jointures.

Modèle en flocon



Modèle en constellation



La table de faits

En volume : la partie la plus importante de la base.

Un fait = une ligne de la table = une mesure

Une mesure = une valeur selon n dimensions

D'où la notion de cube de données

Un fait= un cube atomique.

Une requête : couper une tranche dans ce cube.

Quels faits représenter?

Les faits numériques additifs.

Total des ventes par magasin et par mois.

Prix d'un article vendu dans un magasin dans une tranche horaire.

Nombre d'articles en stock

Mais pas de pourcentage, de numéro de facture . . .

Additivité des Attributs de Fait

Plusieurs millions de faits à résumer

- compter les faits
- additionner les mesures

Propriété d ’additivité

- Fait additif
  - additionnable suivant toutes les dimensions
- Fait semi additif
  - additionnable seulement suivant certaines dimensions
- Fait non additif
  - non additionnable quelque soit la dimension
    - comptage des faits ou affichage 1 par 1

Additivité des Attributs de Fait

Exemple

- quantité vendue, chiffre d ’affaire, coût, nombre de clients, nombre d’appel ...

Fait additif

- quantité vendue, chiffre d ’affaire, coût

Fait semi additif

- niveau de stock, de solde (valeurs instantanées)
  - excepté sur la dimension temps
- nombre de transaction, de client
  - excepté sur la dimension produit

Fait non additif

- ex: un attribut ratio
- ex: marge brute = 1 - Coût/CA

Granularité / Finesse des Faits

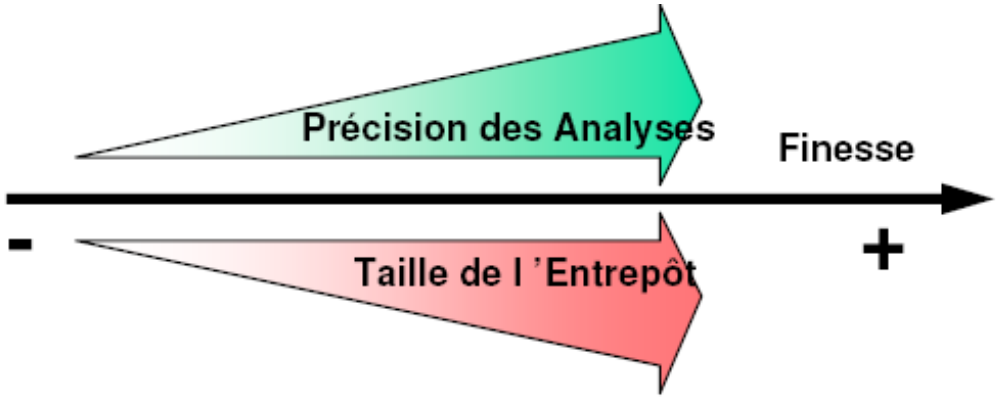
Tables éparses

- hypothèse d ’un monde fermé
  - s ’il y a pas de fait (vente = 0$), on ne le représente pas

Niveau de détail de représentation

- journée > heure du jour
- magasin > rayonnage

Choix de la granularité



Tables de Dimension

- Membre d'une dimension
  - membre spécifique munie de caractéristiques propres
- Description
  - en général textuelle
  - parfois discrète (ensemble limité de valeurs) : parfum de glace, couleur d'habit, …
- Utilisation
  - contrainte applicative
  - entête de ligne (dans des tableaux)
- Remarque importante et Rappel
  - Tables de dimension << Table de fait

Dimension Temps

Commune à tout entrepôt

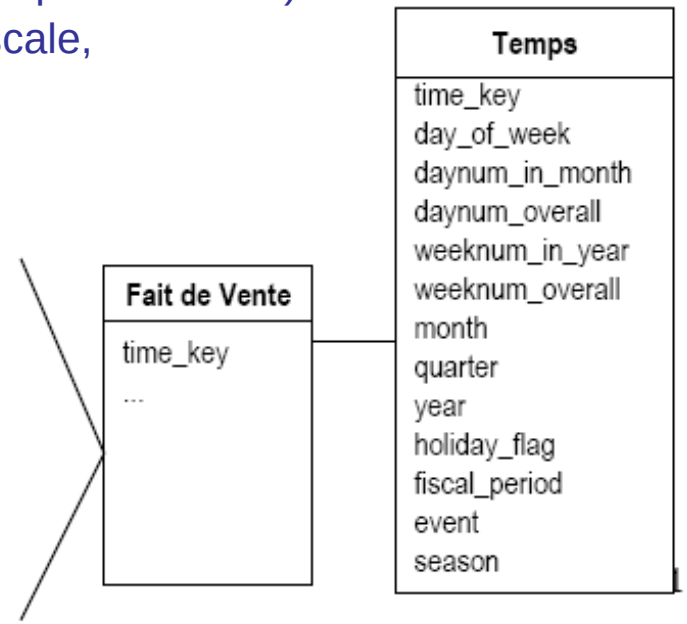
Relié à toute table de fait

2 choix d ’implantation

- Type SQL DATE
- Calendrier + Table Temps
  - informations supplémentaires
    - événement (match de finale de coupe du monde)
    - jours fériés, vacances, période fiscale,
    - saison haute ou basse, …

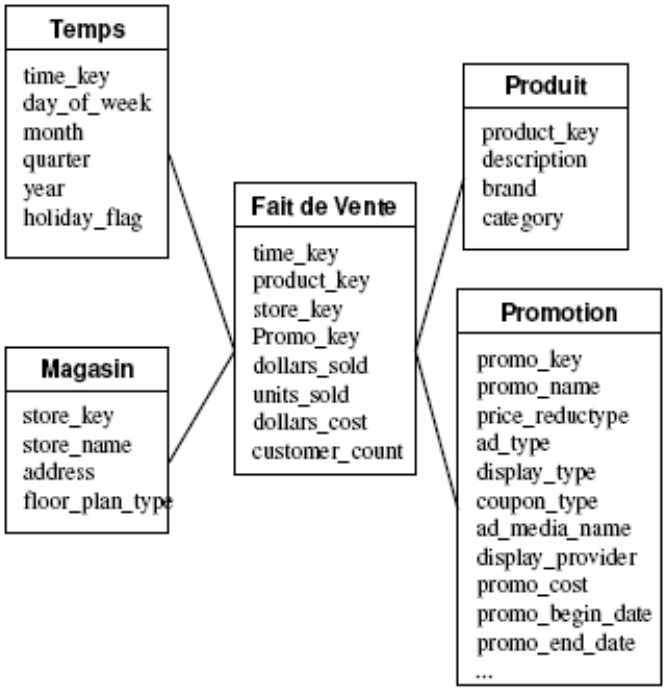
Sémantique du temps

- Validation
  - occurrence du fait
- Transaction
  - prise en compte dans l ’entrepôt



Dimension Causale

- dimension qui provoque le fait
  - ex: la dimension Promotion est supposée avoir provoqué le Fait de Vente



Clés dans l ’entrepôt

Tables de dimension

- clé primaire

Tables de fait

- clé composite ou concaténée
  - clés étrangères des tables de dimension
  - utilisée dans les contraintes de jointure naturelle

Choix des clés d ’une table de dimension

- Taille d ’un fait et Coût des comparaisons de jointures
  - valeurs entières anonymes (4 octets)
- Clés étendues
  - 2 mêmes produits de couleurs différentes = 2 membres
  - Dimension à évolution lente

Grandes Dimensions

Nombreux membres

réduire la taille des tables

- dimension Produits (300.000)
- dimension Clients (10.000.000)

Solutions

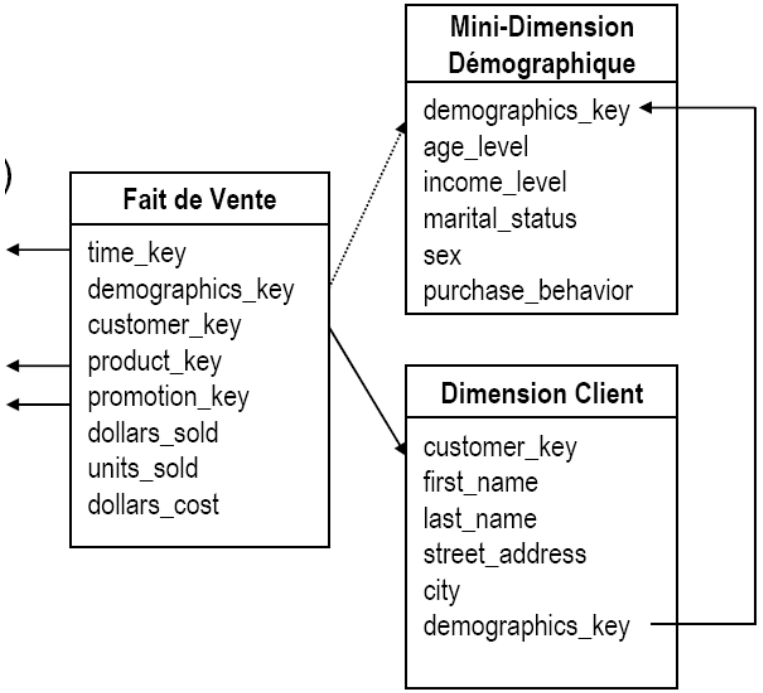
- L ’appel du Flocon de Neige
  - tables de dimension secondaires (déportées)
  - associée à une table de dimension
  - Faible gain de place et Navigation compromise
- ☺ Mini Dimensions
  - Mini dimensions démographiques pour les clients

Dimension client

- nombreux enregistrements, nombreux attributs

Solutions

- Flocons
- ☺ Mini-Dimension
  - Combinaisons (<100000) d'intervalles de valeurs démographiques



Évolution des dimensions

Dimensions à évolution lente

Dimensions à évolution rapide

Dimensions à évolution lente

Changement de description des membres dans les dimensions

- un client peut changer d ’adresse, se marier, ...
- un produit peut changer de noms, de formulations
- « Tree ’s » en « M&M », « Raider » en « Twix », « Yaourt à la vanille en Yaourt » en « saveur Vanille », « bio » en « Activa »

Choix entre 3 solutions

- écrasement de l ’ancienne valeur
- versionnement
- valeur d ’origine / valeur courante

Remarque

- quand la transition n ’est pas immédiate : il reste pendant un certain temps des anciens produits en rayon

Solution : 2 membres différents

Dimensions à évolution lente (1/3)

Écrasement de l'ancienne valeur :

- Correction des informations erronées

Avantage:

- Facile à mettre en œuvre

Inconvénients:

- Perte de la trace des valeurs antérieures des attributs
- Perte de la cause de l'évolution dans les faits mesurés

Clé produit	Description du produit	Groupe de produits
12345	Intelli-Kids	~~Logiciel~~

Jeux éducatifs

Dimensions à évolution lente (2/3)

Ajout d'un nouvel enregistrement:

- Utilisation d'une clé de substitution

Avantages:

- Permet de suivre l'évolution des attributs
- Permet de segmenter la table de faits en fonction de l'historique

Inconvénient:

- Accroit le volume de la table

Clé produit	Description du produit	Groupe de produits
12345	Intelli-Kids	Logiciel
25963	Intelli-Kids	Jeux éducatifs

Dimensions à évolution lente (3/3)

Ajout d'un nouvel attribut:

- Valeur origine/valeur courante

Avantages:

- Avoir deux visions simultanées des données :
  - Voir les données récentes avec l'ancien attribut
  - Voir les données anciennes avec le nouvel attribut
- Voir les données comme si le changement n'avait pas eu lieu

Inconvénient:

- Inadapté pour suivre plusieurs valeurs d'attributs intermédiaires

Clé produit	Description du produit	Groupe de produits	Nouveau groupe de produits
12345	Intelli-Kids	Logiciel	Jeux éducatifs

Évolution des dimensions

Dimensions à évolution lente

Dimensions à évolution rapide

- Subit des changements très fréquents (tous les mois) dont on veut préserver l'historique
- Solution: isoler les attributs qui changent rapidement

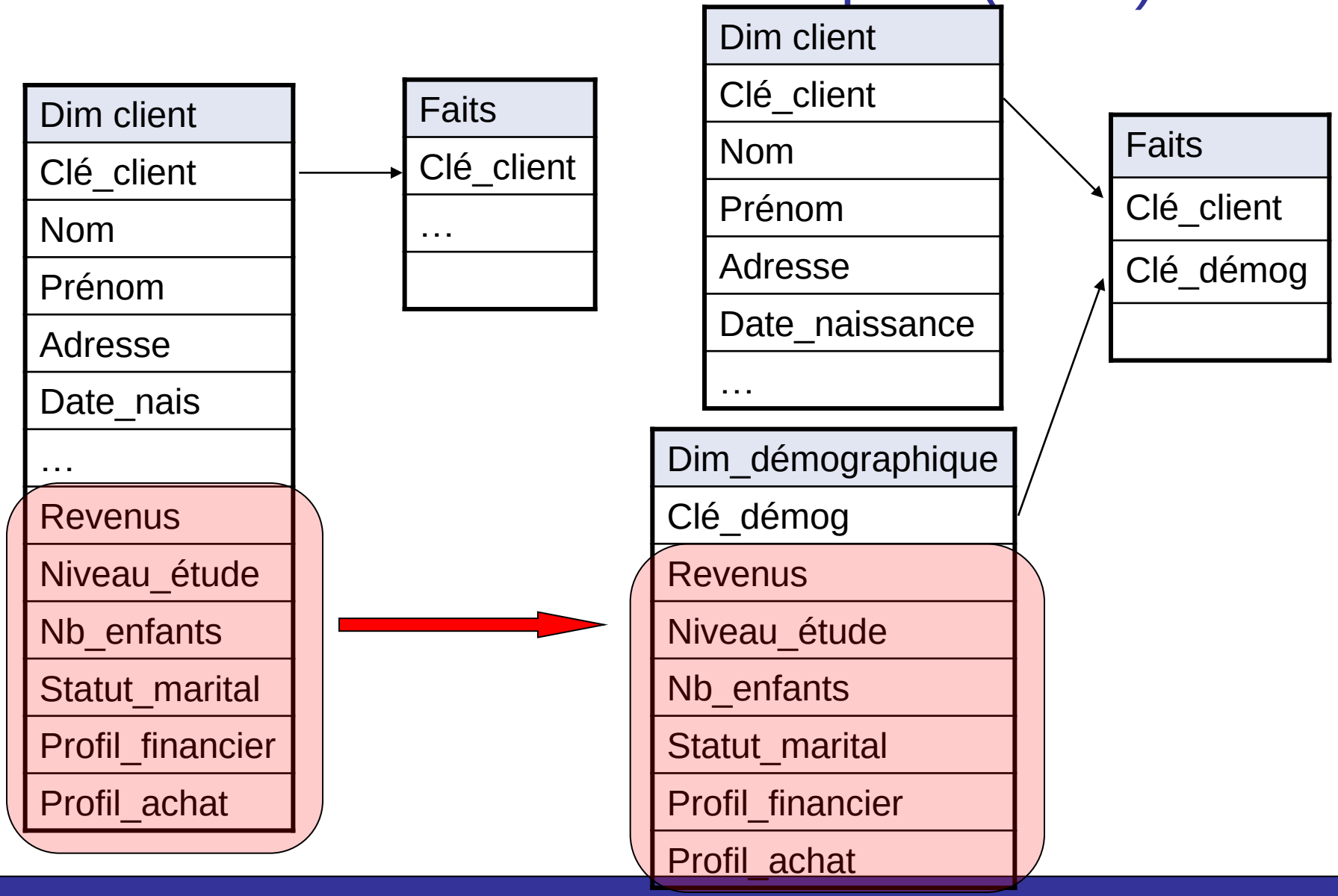
Dimensions à évolution rapide

Changements fréquents des attributs dont on veut garder l'historique

- Clients pour une compagnie d'assurance

Isoler les attributs qui évoluent vite

Dimensions à évolution rapide (suite)



Dimension Douteuse

Exemple :

Dimension Client dans laquelle la même personne peut apparaître de nombreuses fois

- orthographes légèrement différentes
- attributs différents

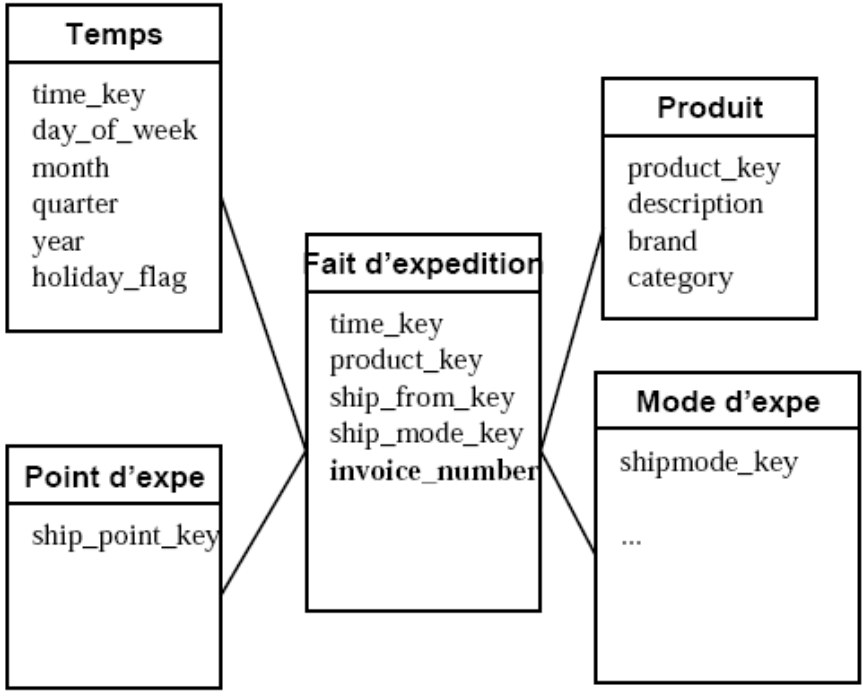
Dimension Dégénérée

Dimension sans attribut

- Pas de table
- Mais la clé de dimension est dans la table de fait

Exemple

- numéro de facture (invoice number),
- numéro de ticket
- …



Codage des Clés et des Mesures

Mesure de fait

- valeurs entières (4 octets)
- parfois plus
- ex: PNB des USA au cent près

Clés

- valeurs entières anonymes (4 octets)
  - réduit la taille de l ’enregistrement de fait
  - réduit le coût CPU des comparaison de jointure
- la correspondance clé opérationnelle et clé entrepôt est faite à l’extraction

Estimation de la taille de l ’entrepôt

Dimensionner l ’entrepôt

Choix des granularités

Choix d ’une machine/SGBD cible (benchmark)

Exemple : Supermarché

Dimensions

- Temps : 4 ans * 365 jours = 1460 jours
- Magasin : 300
- Produit : 200000 références GENCOD (10% vendus chaque jour)
- Promotion : un article est dans une seule condition de promotion par jour et par magasin

Fait

- 1460 * 300 * 20000 * 1 = 8,76 milliards d ’enregistrements
- Nb de champs de clé = 4
- Nb de champs de fait = 4

Table des Faits = 8,76.10^9 * 8 champs * 4 octets = 280 Go

Estimation de la taille de l ’entrepôt

Exemple : Ligne d ’article en Grande Distribution

- Temps : 3 ans * 365 jours = 1095 jours
- CA annuel = 80 000 000 000 $
- Montant moyen d ’un article = 5 $
- Nb de champs de clé = 4
- Nb de champs de fait = 4
- Nombre de Faits = 3*(80.10^9/5) = 48.10^9
- Table de Faits= 48.10^9 * 8 champs * 4 octets = 1,59 To

Exemple : Suivi d ’appels téléphoniques

- Temps : 3 ans * 365 jours = 1095 jours
- Nombre d ’appel par jour = 100 000 000
- Nb de champs de clé = 5
- Nb de champs de fait = 3
- Table des Faits = 1095.10^8 * 8 champs * 4 octets = 3,49 To

Estimation de la taille de l ’entrepôt

Exemple : Suivi d ’achats par carte de crédit

- Temps : 3 ans * 12 mois = 36 mois
- Nombre de compte carte = 50 000 000
- Nombre moyen d ’achat par mois par carte = 50
- Nb de champs de clé = 5
- Nb de champs de fait = 3
- Table des Faits = 90.10^9 * 8 champs * 4 octets = 2,6 To

Exemples de DW

Analyse en 4 étapes

I Sélectionner le processus d'entreprise à modéliser :

- commande client (vente ou marketing ?)
- Stock (en entrée ou en sortie ?)
- Communication téléphonique (bénéfice ou bande passante ?)

-II- Définition d'un fait : dimensions, granularité

III Description des dimensions.

IV Définition d'une mesure.

Exemple 1

La grande distribution Chaîne de magasins d'alimentation :

- 100 magasins.
- Répartis en 5 régions.
- 8 rayons principaux.

Chaque magasin contient 60.000 articles.

55000 articles proviennent de fournisseurs. 5000 sans code barre (pain, fleurs ...)

Que demande l'entreprise

- Commandes, fournisseurs?
- Etat des stocks?
- Logistique ?
- Maximiser le profit

Maximiser le profit

Suivi des prix.

Analyse des promotions :

- Vecteur le plus efficace?
- Efficacité d'une action?
- …

Collecte de l'information

Bilan journalier par magasin.

Entrée et sortie des stocks

Caisses enregistreuses.

Etape 1

Sélectionner le processus à modéliser

Comprendre le comportement des clients :

- Produits qui se vendent bien.
- Jours de meilleures ventes.
- Promos les plus efficaces.
- Panier de la ménagère
- . . .

Etape 2 : Définition du fait

Le grain le plus fin possible : une ligne sur le ticket de caisse.

Autres choix :

- Un ticket de caisse.
- Récapitulatif horaire par caisse (par produit . . .)
- Un produit par jour par magasin . . .

Taille mémoire : ok.

Etape 3 : Choix des dimensions

- Temps.
- Produit.
- Magasin.
- Promotion.
- Responsable magasin.

Etape 4 : Choix des faits (mesures)

- Le nombre d'articles identiques.
- Coût pour le client.
- Prix de revient.
- Bénéfice sur cette vente.
- Numéro du ticket de caisse

La dimension date

Fixe : peu se construire à l'avance.

Pas très grande (unité : jour, 10ans = 3650 lignes)

Contenu :

- Date.
- Jour de la semaine.
- Jour férié ou ouvrable.
- Année.
- Trimestre.
- . . .

Informations redondantes : évitent les calculs, ne prennent pas trop de place.

Introduire les heures : plutôt créer une autre dimension.

Dimension produit

- Numéro d'identification (code barre)
- Marque
- Catégorie, sous-catégorie …
- Rayon
- Emballage
- Dimension L*H*l

Dimension produit

60 000 produits : la table est grande mais de taille négligeable par rapport à la table de faits

Ne pas hésiter à mettre des informations (facilement disponibles)

Notion de hiérarchie : forage vers le haut ou vers le bas

Toutes les infos sont disponibles

Dimension magasin

Infos à construire soi-même

Peuvent s'enrichir de données extérieures (géographiques …)

- Nom, adresse
- Région, ville
- Taille
- Responsable (si assez constant)
- …

La dimension promotion

- Identifiant
- Type de promotion
- Vecteur publicitaire
- Date
- Durée
- Coût
- …

Problème : il n'y a rien dans la table des faits sur les articles en promotion qui ne sont pas vendus …

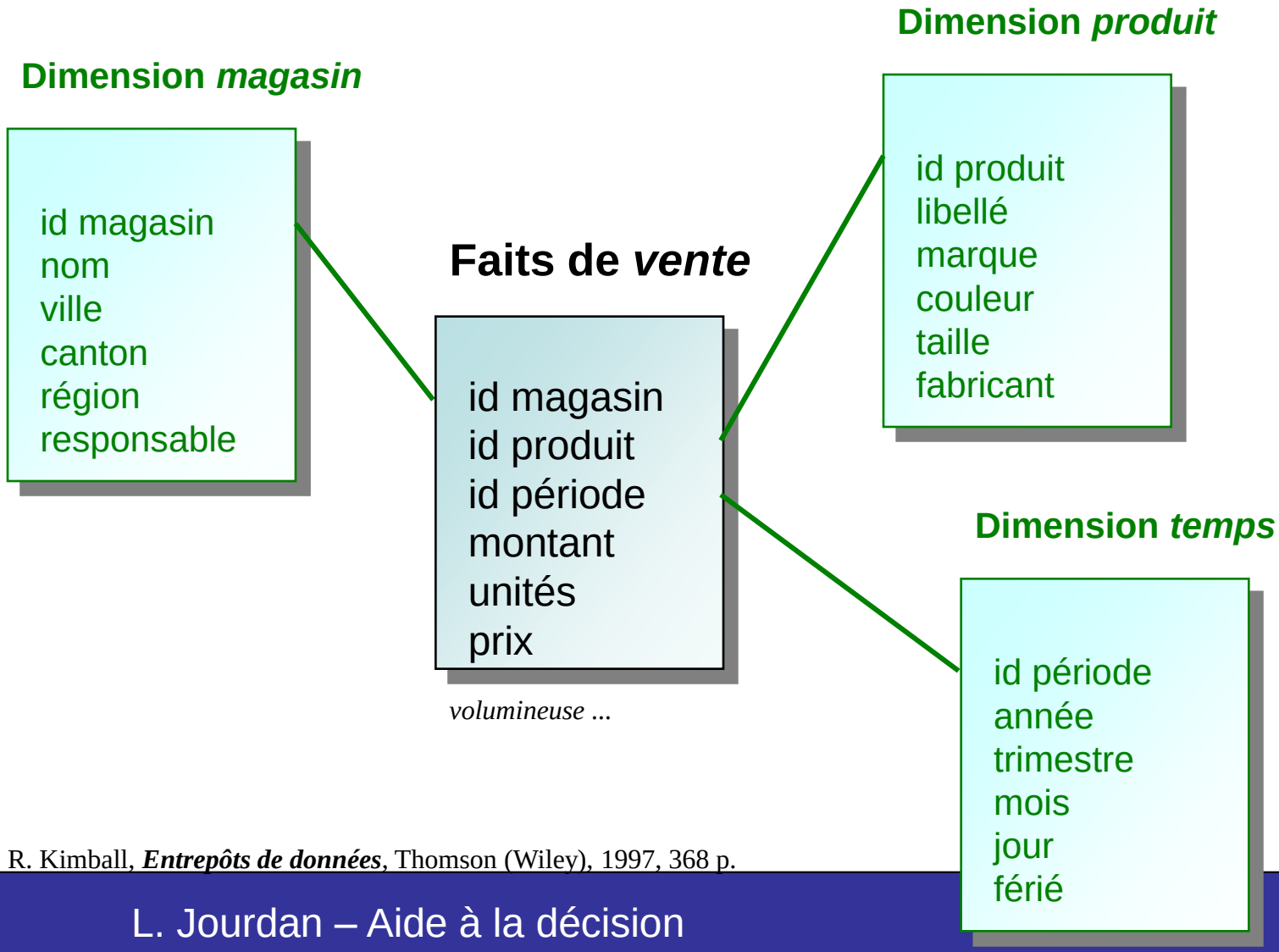
Extension de l’entrepôt

Ajout de dimension

- Carte de fidélité
- Caissier/Caissière
- Heure de passage en caisse

Ajout de champ à des dimensions

Exemple de construction



R. Kimball, ***Entrepôts de données***, Thomson (Wiley), 1997, 368 p.

Faits

Numériques

valorisés de façon continue (*continuously valued*)

- prenant une valeur à l ’intérieur d ’une grande fourchette
- contrairement aux dimensions

additifs

- pour synthétiser (additionner) de grandes masses de chiffres

tables des faits éparses (*sparse*)

- évitant les zéros signifiant « rien à signaler »

Dimensions

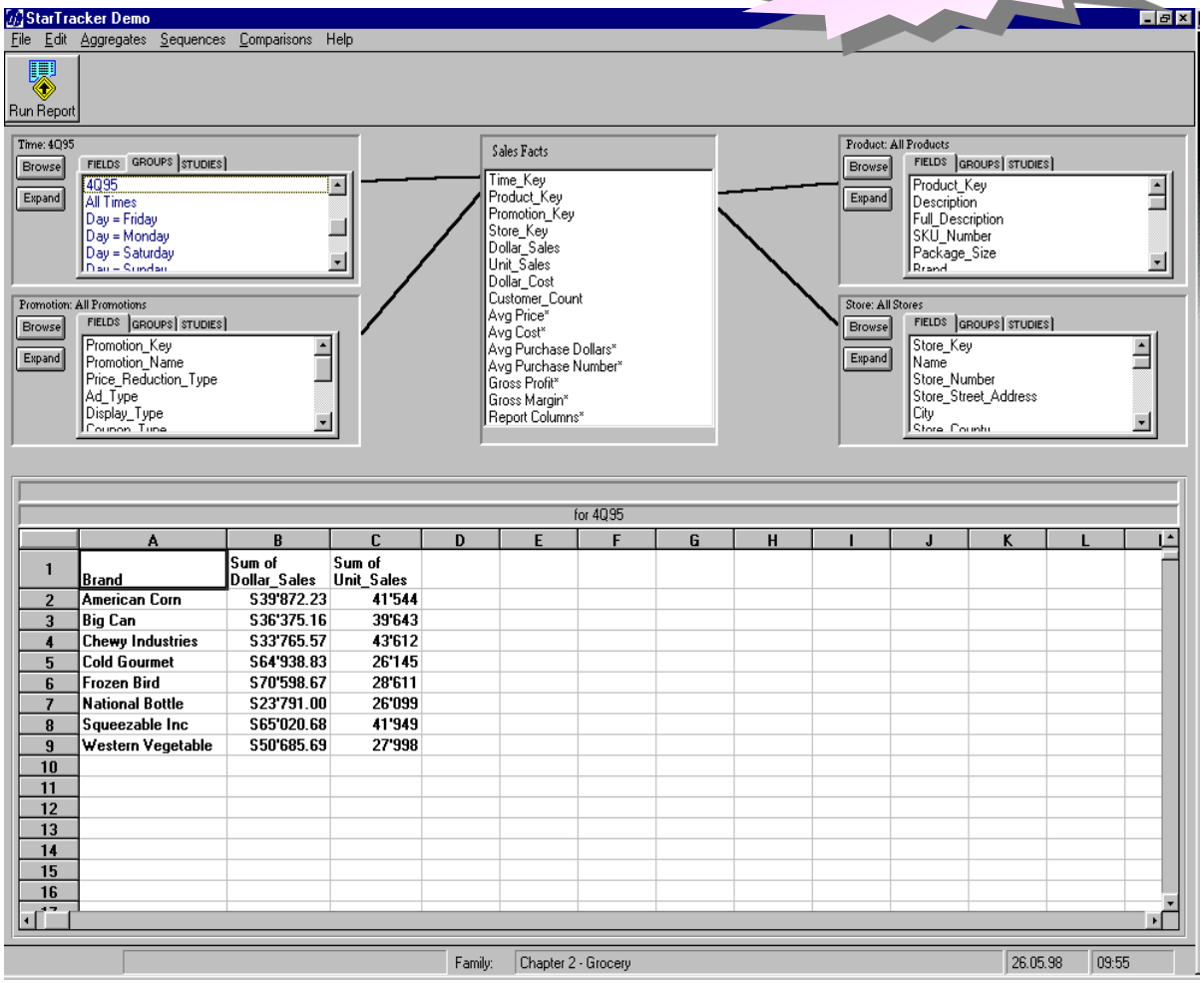
Caractérisées par des attributs

- textuels
- discrets
  - propriétés constantes (parfum de glace, couleur d'habit, …)
  - contrairement aux faits
- utilisés comme contraintes (filtres) ou en-têtes dans les rapports

enjeux majeurs de la modélisation ...

Multi-dimension

Star Tracker



SQL

```
SELECT [Time_SQL], [Time_KEY], [Time_TYPE] FROM [Time_GP] WHERE
[Time_GRP_NAME] = '4Q95' ORDER BY [Time_KEY]
```

```
SELECT [Product].[Brand],
                Sum([Sales Fact].[Dollar_Sales]) as
Col1,
                Sum([Sales Fact].[Unit_Sales]) as Col2
 FROM   [Sales Fact], [Time], [Product]
 WHERE  [Sales Fact].[time_key]=[Time].[time_key]
 AND    [Sales
Fact].[product_key]=[Product].[product_key]
 AND    [Time].[Fiscal_Period] IN ('4Q95')
```

colonnes

Jointures
fait & dimensions

Contrainte
filtre sur dimensions

rupture

Modélisation

choix du **processus d 'activité** à modéliser

- opérationnel, pour lequel les données sont disponibles

choix du **grain** du processus d 'activité

- ou niveau de détail (atomique) des données de la table de faits
  - transactions individuelles
  - récapitulations individuelles (quotidiennes, hebdomadaires ou mensuelles)

choix des **dimensions**

- temps (& magasins, produits, client, promotion …)
- et de leurs *attributs*

choix des **faits**

- quantités additives
- et de leurs *mesures*

Grande surface
contexte

plusieurs magasins

quelques milliers de produits

- unités de stock (***SKU*** ou *stock keeping units*)
- code barre (***UPC*** ou *universal product code*)
  - pour les produits livrés par les fournisseurs (2/3 des produits)

points de vente (***POS*** ou *point of sale*)

- avec scanning des codes à barres

influence sur les ventes: ajustement des prix et promotion

- réduction de prix temporaires (***TPR*** ou *temporary price reduction*)
  - action, ...
- présentation des gondoles (shelf display)
- présentation des têtes de gondoles (end aisle display)

Activité

sur bases des processus de gestion

- des systèmes transactionnels
- des données disponibles (enregistrées)

mouvements journaliers des articles vendus

- quels articles sont vendus
- dans quels magasins,
- à quels prix et
- quels jours

ou des articles remis en stock, en rayon, ...

Grains

TEMPS: mouvement journalier des articles

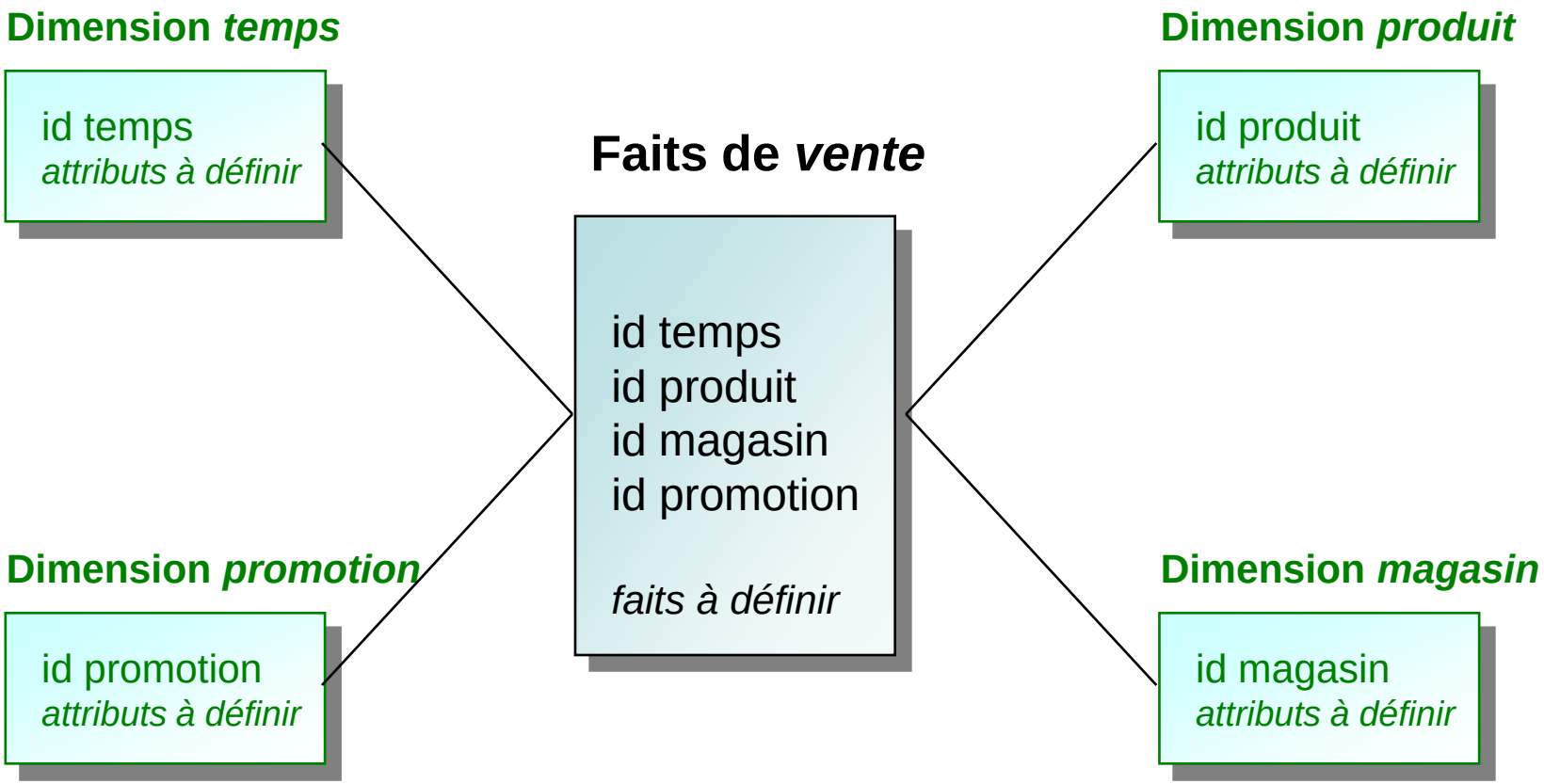
- ou pour chaque transaction
  - chaque passage au point de vente (un ticket client)
  - grand volume
  - si identification du client (carte de fidélité, …), analyse plus fine du comportement
- ou enregistrement hebdomadaire (ou mensuel)
  - mais ignorance de nombreux phénomènes
    - mesure des actions quotidiennes
    - analyse par jour (lundi ou week-end, …

PRODUIT: au niveau des *SKU*

- ou par marque, unité d ’emballage, ...
  - mais ignorance de nombreux phénomènes

les informations exprimées dans le grain le plus bas possible

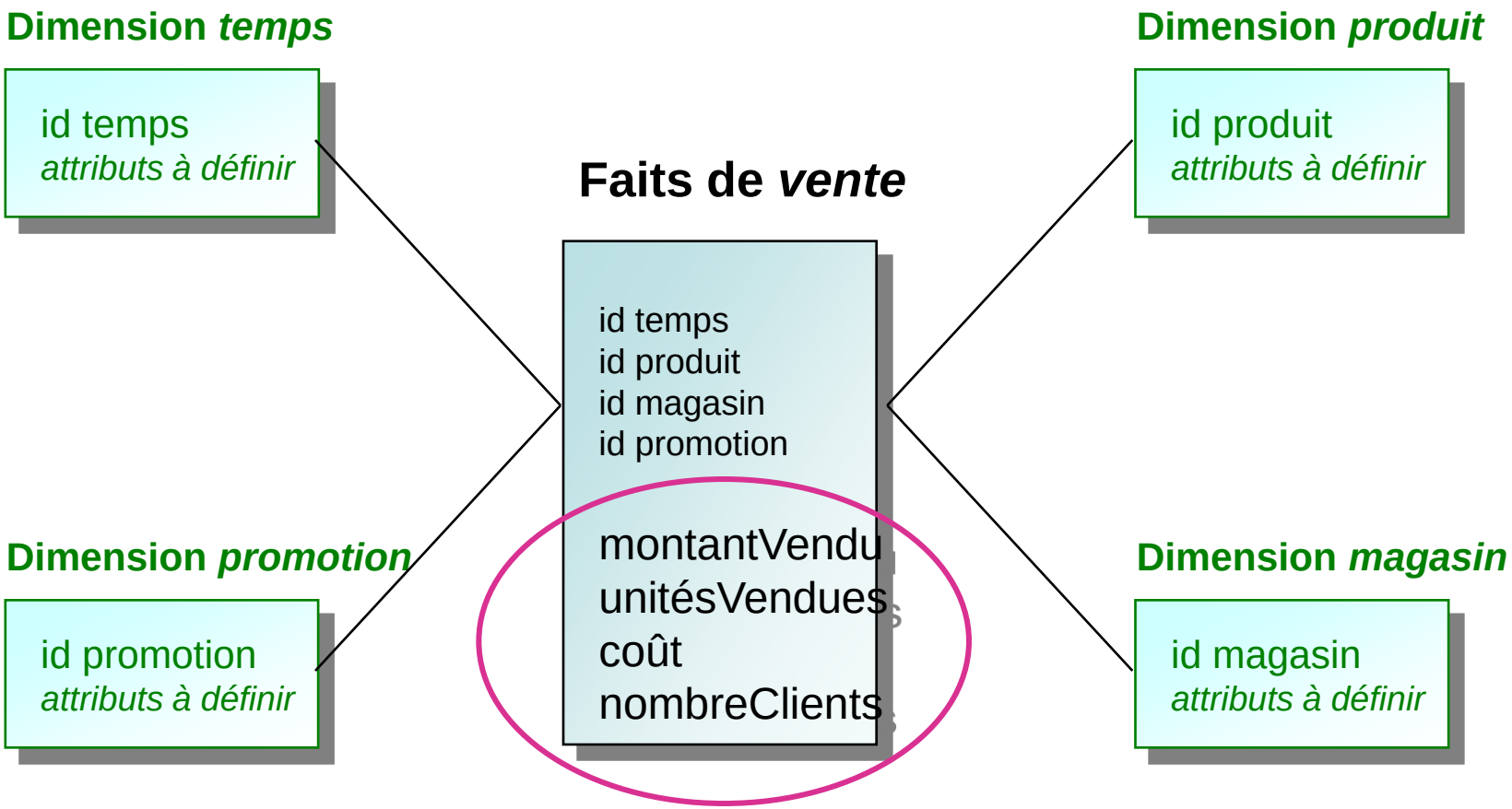
Dimensions



Résister à la normalisation

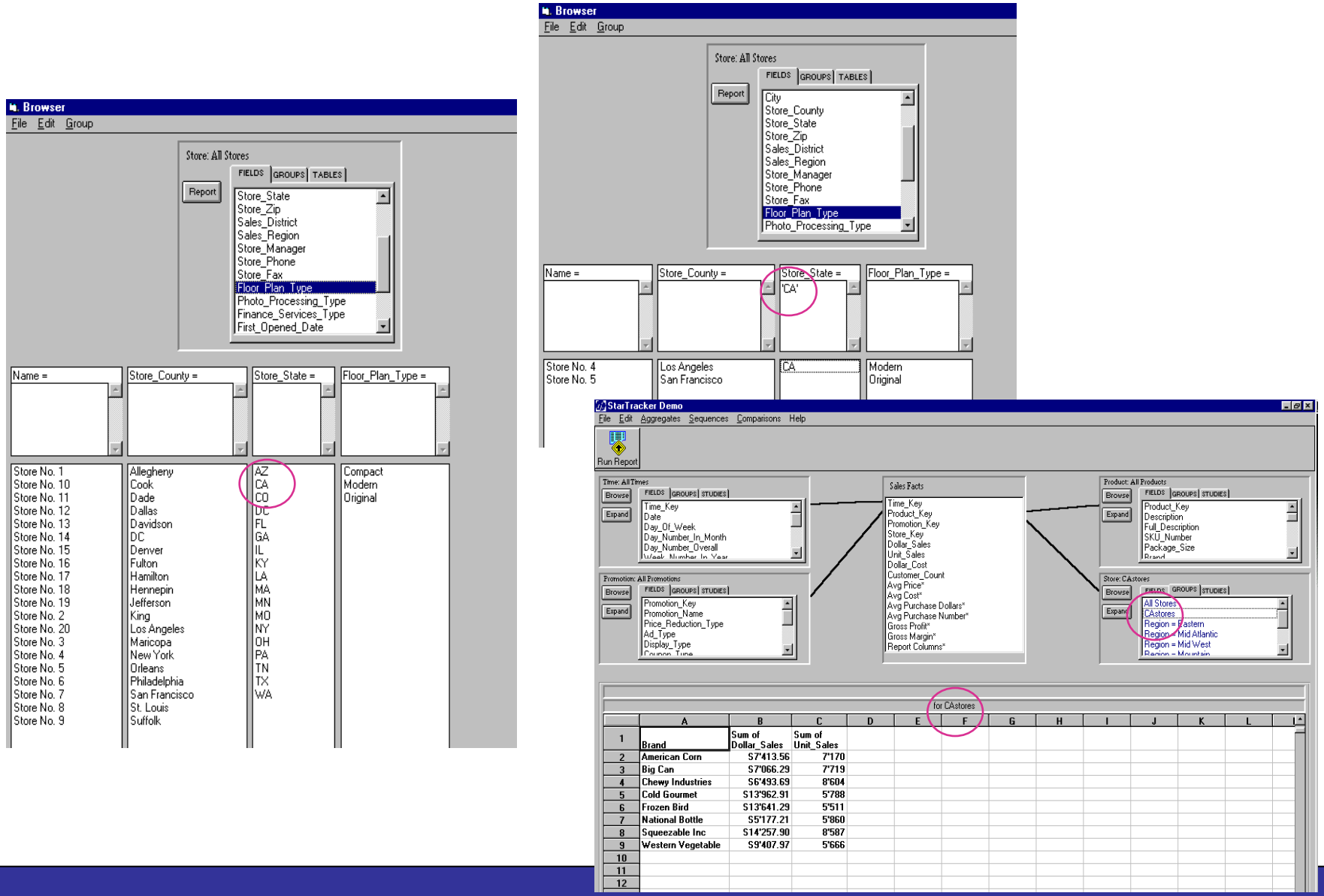
pour préserver une navigation efficace (au détriment d'une certaine redondance)

Faits
mesures



La normalisation est naturellement élevée

Navigation


Browser
File Edit Group
Store: All Stores
FIELDS GROUPS TABLES
Report
Store_State
Store_Zip
Sales_District
Sales_Region
Store_Manager
Store_Phone
Store_Fax
Floor_Plan_Type
Photo_Processing_Type
Finance_Services_Type
First_Opened_Date
Name =
Store_County =
Store_State =
Floor_Plan_Type =
Store No. 1
Store No. 10
Store No. 11
Store No. 12
Store No. 13
Store No. 14
Store No. 15
Store No. 16
Store No. 17
Store No. 18
Store No. 19
Store No. 2
Store No. 20
Store No. 3
Store No. 4
Store No. 5
Store No. 6
Store No. 7
Store No. 8
Store No. 9
Allegheny
Cook
Dade
Dallas
Davidson
DC
Denver
Fulton
Hamilton
Hennepin
Jefferson
King
Los Angeles
Maricopa
New York
Orleans
Philadelphia
San Francisco
St. Louis
Suffolk
AZ
CA
CO
DC
FL
GA
IL
KY
LA
MA
MN
MO
NY
OH
PA
TN
TX
WA
Compact
Modern
Original
Browser
File Edit Group
Store: All Stores
FIELDS GROUPS TABLES
Report
City
Store_County
Store_State
Store_Zip
Sales_District
Sales_Region
Store_Manager
Store_Phone
Store_Fax
Floor_Plan_Type
Photo_Processing_Type
Name =
Store_County =
Store_State =
'CA'
Floor_Plan_Type =
Store No. 4
Store No. 5
Los Angeles
San Francisco
CA
Modern
Original
StarTracker Demo
File Edit Aggregates Sequences Comparisons Help
Run Report
Time: All Times
Browse
Expand
FIELDS GROUPS STUDIES
Time_Key
Date
Day_Of_Week
Day_Number_In_Month
Day_Number_Overall
Sales Facts
Time_Key
Product_Key
Promotion_Key
Store_Key
Dollar_Sales
Unit_Sales
Dollar_Cost
Customer_Count
Avg Price*
Avg Cost*
Avg Purchase Dollars*
Avg Purchase Number*
Gross Profit*
Gross Margin*
Report Columns*
Product: All Products
Product_Key
Description
Full_Description
SKU_Number
Package_Size
Promotion: All Promotions
Promotion_Key
Promotion_Name
Price_Reduction_Type
Ad_Type
Display_Type
Store: CAstores
All Stores
CAstores
Region = Eastern
Region = Mid Atlantic
Region = Mid West
for CAstores
Brand
Sum of Dollar_Sales
Sum of Unit_Sales
American Corn $7'413.56 7'170
Big Can $7'066.29 7'719
Chewy Industries $6'493.69 8'604
Cold Gourmet $13'962.91 5'788
Frozen Bird $13'641.29 5'511
National Bottle $5'177.21 5'860
Squeezable Inc $14'257.90 8'587
Western Vegetable $9'407.97 5'666

Dimension *Temps*

le type SQL « date » standard est insuffisant

- *peut éventuellement être la clé de la table (et la clé étrangère dans la table des faits)*

chaque jour peut être caractérisé par

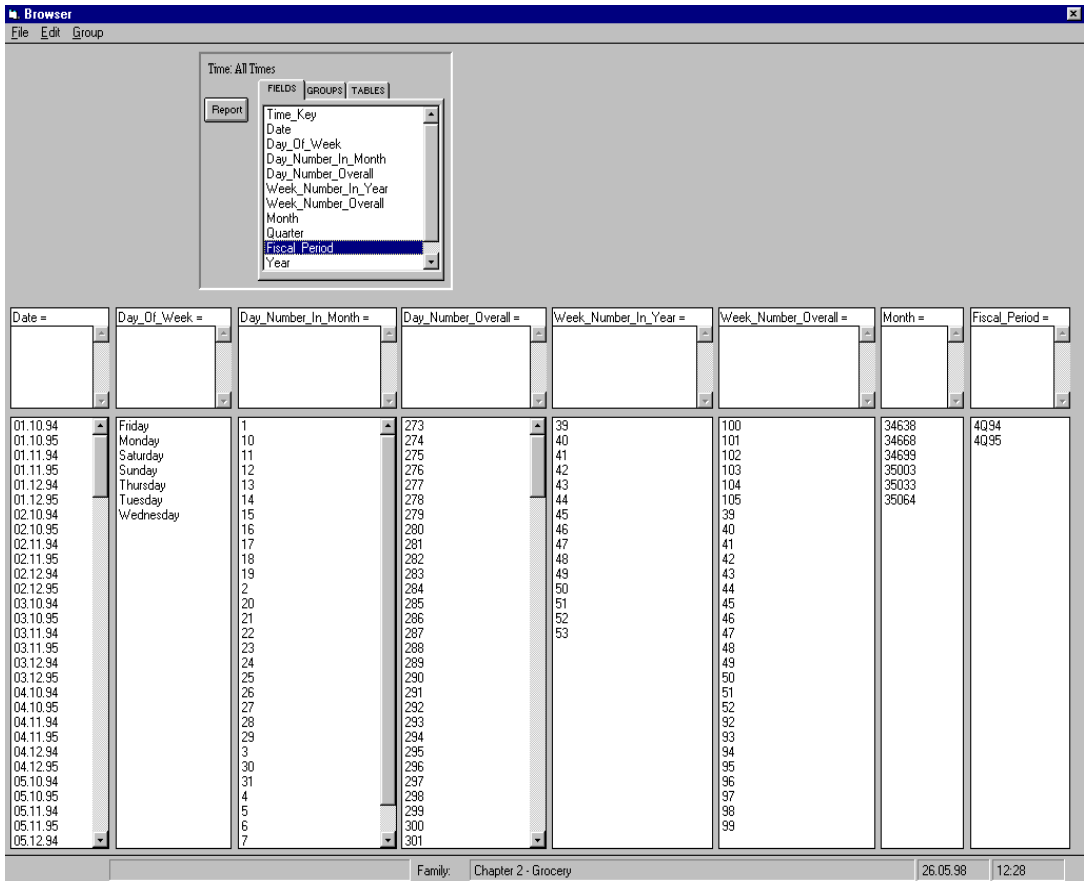
- le jour de la semaine (lundi, …)
- le numéro du jour du mois (1, 2, …)
- s 'il est le dernier jour du mois (O/N)
- le numéro du jour (calendrier julien à partir d 'une date donnée)
- le numéro de semaine dans l 'année (1, 2, … 52)
- le numéro du mois (1, 2, …12)
- le mois (janvier, février, …)
- le trimestre (1er, 2ème, …)
- la période fiscale (1Q98, 2Q98, …)
- s 'il est férié ou non
- la saison (printemps, été, …)
- un événement (final de foot, …)

Dimension *temps*

id temps

jourSemaine
noJourMois
dernJour
noJour
noSemaine
noMois
mois
trimestre
périodeFisc
férié
saison
événement

Temps



Peut être générée et garnie à l 'avance

Dimension *Produit*

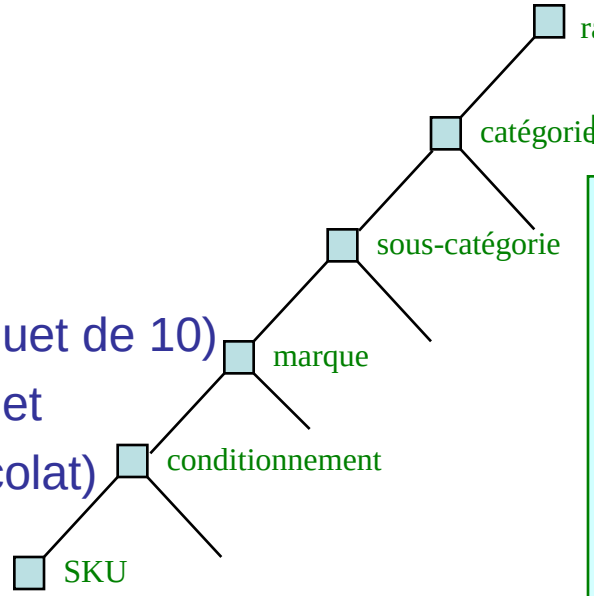
Au niveau des unités de stockage (SKU)

gestion de la hiérarchie des attributs

- pour chaque *SKU*:
  - le rayon (alimentation)
  - la catégorie (surgelé)
  - la sous-catégorie (glace)
  - sa marque (Nestlé)
  - son conditionnement (paquet de 10)
  - le numéro du *SKU* (*UPC*) et
  - son libellé (Magnum Chocolat)

et d'autres attributs

  - emballage
  - longueur du rayon
  - largeur du rayon
  - profondeur du rayon



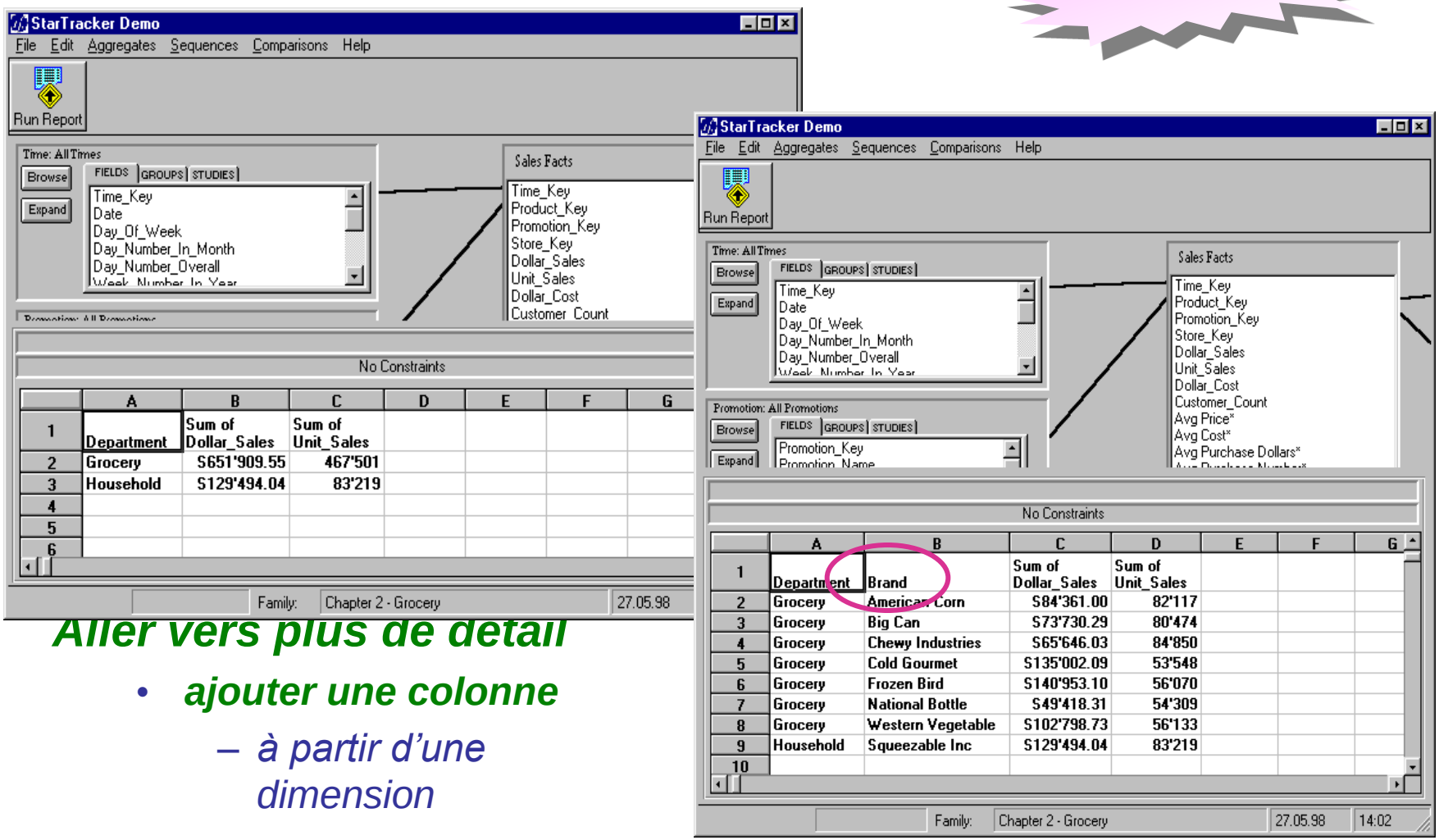
Dimension *produit*

id produit

rayon
catégorie
sous-catégorie
marque
conditionnement
numéroSKU
libelléSKU
emballage
diététique
poids
unitéMesure
...

Forage vers le bas

Drill down



Aller vers plus de detail

- ajouter une colonne
  - *à partir d'une dimension*

Dimension *magasin*

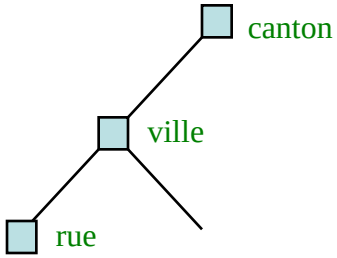
Point dans l'espace

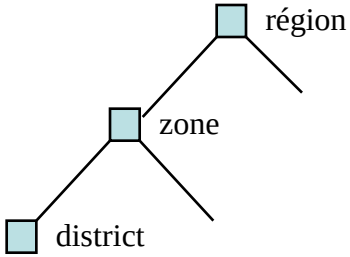
par rapport à des hiérarchies

- *géographique (NPA, …)*
  - *ville*
  - *canton*
- *de régions de vente*
  - région
  - zone
  - district

et d'autres attributs

- type (M, MM, MMM …)
- surface (500, 1500, 5000, …)
- nom
- responsable
- date d 'ouverture





Dimension *magasin*

id magasin

canton
ville
NPA
région
zone
district
type
surface
nom
responsable
dateOuverture
...

Dimension *promotion*

modalités de promotion ou facteurs

dimensions?

- réductions de prix
- annonces publicitaires
- présentations en tête de gondole
- coupons

1 ou n **causale**

supposés provoquer des modifications dans les ventes

- *gain lors d'une promotion*
- *décalage temporel (time shifting)*
  - *perte de gain après la promotion*
- *cannibalisation*
  - *perte de gains pour d 'autres produits*
- *croissance du marché*
  - *avant, pendant et après la promotion*

corrélation sont fréquentes

- *réduction, annonce et tête de gondole*, par exemple
- création d 'un enregistrement pour chaque combinaison

Dimension *promotion*

id promotion

nom
réductionPrix
annonce
têteGondole
coupon
affichage
agencePublicité
coût
dateDébut
dateFin
...

Additivité

À partir des faits (ventes)

- quantité ou unités vendues
- vente en Euros (chiffre d ’affaire)
- coût

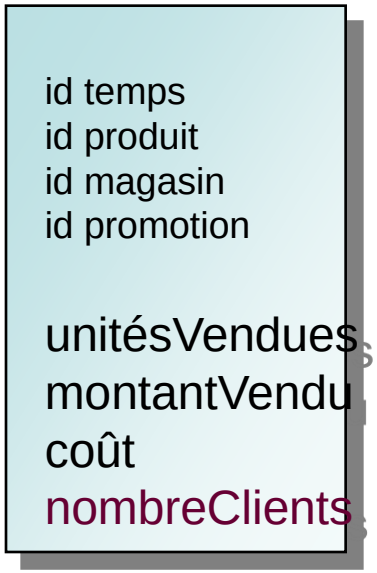
on peut notamment calculer

- le *profit brut* (vente - coût)
- la *marge brute* (profit brut / vente)

mais il faut remarquer:

- qu’un ratio, comme la *marge brute*, n ’est pas additif
- pour une tranche de l’entrepôt, il faut calculer
  - le ratio des sommes
  - et non la somme des ratios

Faits de *vente*



Semi-additivité

l'attribut *nombreClients* n 'est pas additif

- sur la dimension *Produit*
- car le grain concerne des récapitulations (périodiques)
  - et non les transactions (tickets)
- donc on compte plusieurs fois un même événement *clien*
  - sur différents produits
  - exemple:
    - si le produit x a été acheté par 20 clients, le produit y par 30 clients
    - on ne peut pas savoir le nombre de clients ayant acheté x ou y
      - compris entre 30 et 50
- solutions:
  - changer le grain (transaction),
  - calcul plus agrégé (par catégorie, par ex. et non par article) ou
  - ***Cf. agrégat mémorisé***

Faits de *vente*

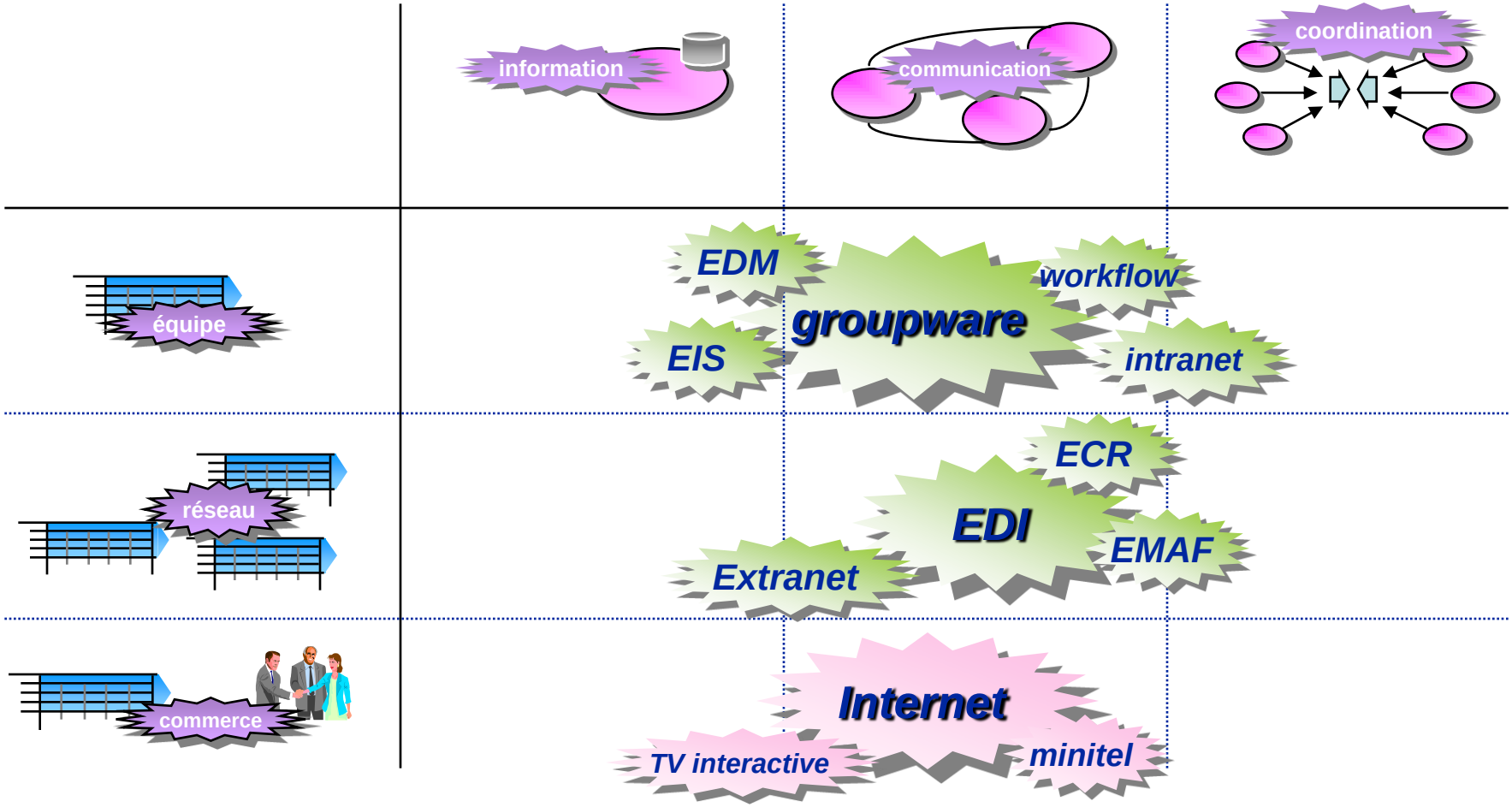
id temps
id produit
id magasin
id promotion

unitésVendues
montantVendu
coût
nombreClients

Volume

Dimension TEMPS	**2 ans x 365 jours, soit 730**
Dimension MAGASIN	**300 magasins**
Dimension PRODUIT	**30'000 produits, don't 3'000 vendus chaque jour**
Dimension PROMOTION	**un article n'apparaît que dans une promo. Par jour par magasin**
FAITS élémentaires	**730 x 300 x 3'000 x 1, soit 657 millions**
champs clés & mesures	**4 dimensions + 4 mesures, soit 8 champs**
TAILLE des FAITS	**657 millions x 8 champs x 4 octets, soit 21 gigaOctets**

Autres technologies à suivre ...



OLAP, MOLAP, ROLAP

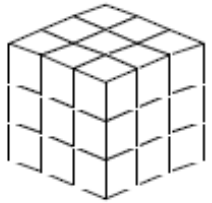
L'Analyse MultiDimensionnelle

Objectif

- obtenir des informations déjà agrégées selon les besoins de l'utilisateur : simplicité et rapidité d'accès

HyperCube OLAP

- représentation de l'information dans un hypercube à N dimensions



OLAP (On-Line Analytical Processing)

- fonctionnalités qui servent à faciliter l'analyse multidimensionnelle : opérations réalisables sur l'hypercube

OLAP / OLTP

OLTP

- Les applications conçues pour des opérations quotidiennes dans les BDs.
- Ces transactions nécessitent des données détaillées et actualisés.
- Les BD vont de quelques milliers de Mo à des Go.

OLAP

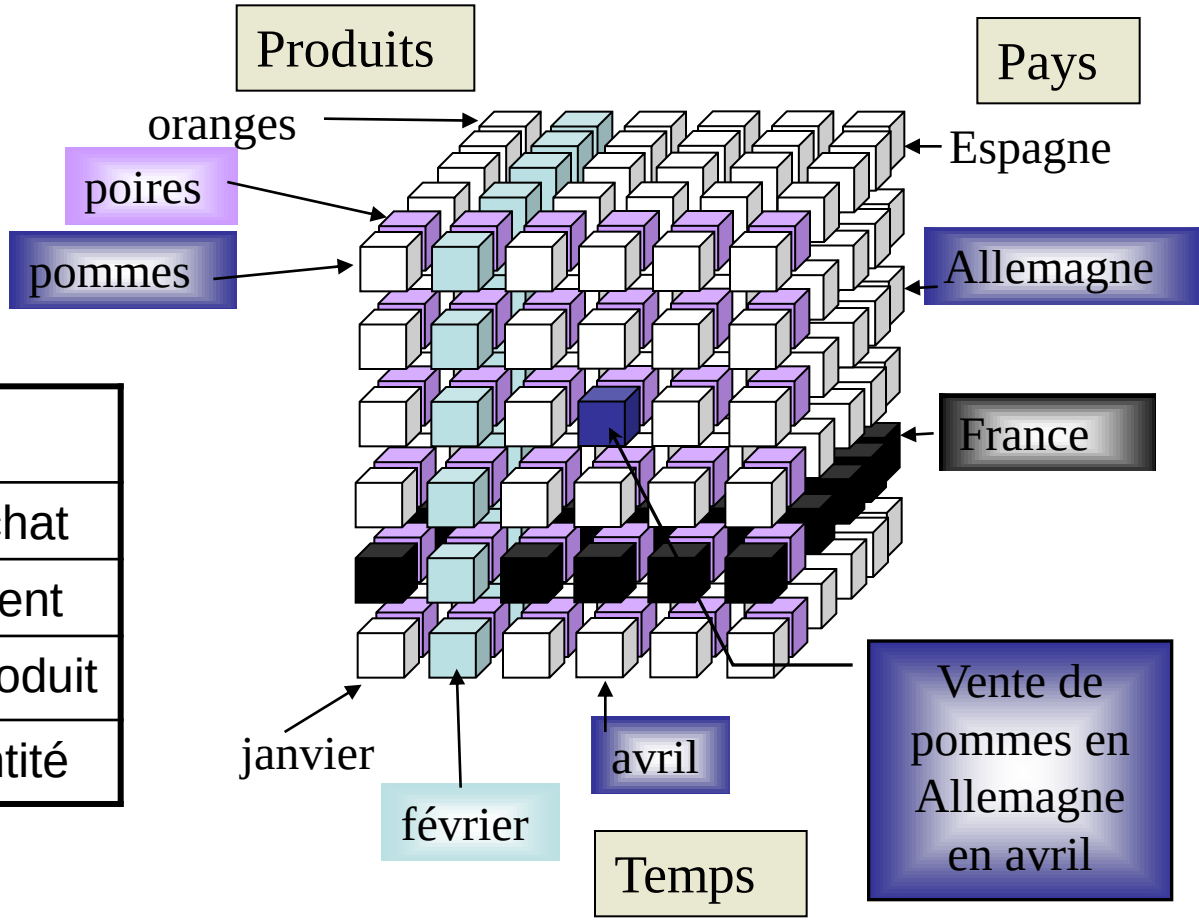
- Les données sont historisées, résumées, consolidées.
- Les DW contiennent des données sur une longue période de temps.
- Les DW vont de centaines de Go à des To (Téra octets).

OLTP VS OLAP

Produit	
PK	id_produit
	Libellé
	Famille

Achat	
PK	id_achat
FK	id_client
	id_produit
	Quantité

client	
PK	id_client
	Nom
	adresse



Glossaire OLAP

Dimension

- Temps, Produit, Géographie, ...

Niveau : hiérarchisation des dimensions

- Temps :
  - Année, Semestre, Trimestre, Mois, Semaine, ...
- Produit :
  - Rayon, Catégorie, Nature, ...
- Géographie :
  - Région, Département, Ville, Magasin

Membre d'un Niveau

- Produit :Rayon
  - Frais, Surgelé, ..., Liquide
- Produit::Rayon.Catégorie
  - Frais.Laitage, ..., Liquide.Vin
- Produit::Rayon.Catégorie.Nature
  - Frais.Laitage.Yaourt, ... , Liquide.Vin.Champagne

Glossaire OLAP

Cellule

- Intersection des membres des différentes dim.

Formule

- calcul, expression, règle, croisement des dim.
  - Somme(Qte), Somme(Qte*PrixVente),
  - Moyenne(Qte*(PrixVente-PrixAchat)), ...

Les 12 règles OLAP

1°) une vue multidimensionnelle des données.

2°) La transparence vis à vis de l'utilisateur qui doit accéder à la BD par l'intermédiaire d'outils simples (tableur, par ex).

3°) La BD doit disposer d'un modèle et d'outils permettant d'accéder à de multiples sources, d'effectuer les conversions et extractions nécessaire pour alimenter la Base OLAP.

4°) Le modèle de données, le nombre de dimensions ou le nombre de niveaux d'agrégation doivent pouvoir changer, sans remettre en cause le fonctionnement de la base.

5°) Architecture Client/Serveur.

6°) Toutes les dimensions définies dans le modèle de données doivent être accessibles pour chacune des données.

Les 12 règles OLAP

7°) Gestion des matrices creuses. Les parties vides du cube multidimensionnel doivent être stockées de manière à ne pas détériorer les temps d'accès.

8°) Accessibilité simultanément par plusieurs utilisateurs.

9°) Toutes les données stockées ou calculées dans le cube doivent être accessibles et les règles de gestion doivent toujours s'y appliquer. Toutes les tranches de cube doivent être visualisées.

10°) Navigation aisée dans les données pour les utilisateurs, de manière intuitive.

11°) Outil de présentation des données.

12°) Nombre illimité de dimensions et de niveaux d'agrégation.

Opérations OLAP

But

- Visualisation/Utilisation d'un fragment de l'Hypercube

Opérations OLAP

- Drill Up / Drill Down
- Rotate
- Slicing
- Scoping

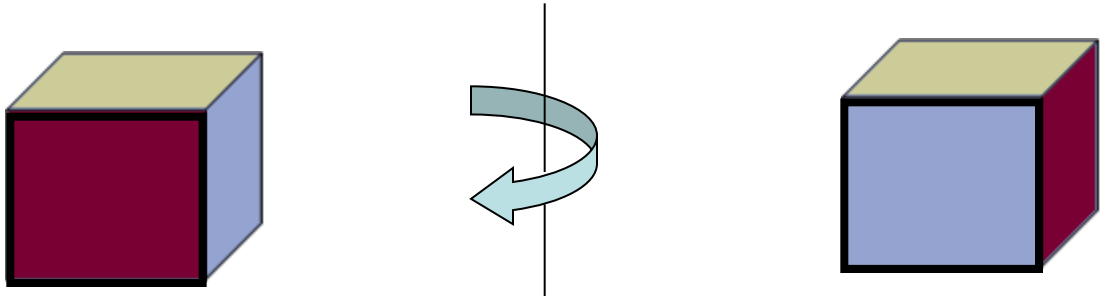
Manipulation des données multidimensionnelles

Opération agissant sur la structure

- Rotation (rotate): présenter une autre face du cube

	09	10	11
Œuf	221	263	139
Viande	275	257	116

	09	10	11
Idf	101	120	52
Nord	395	400	203



Manipulation des données multidimensionnelles

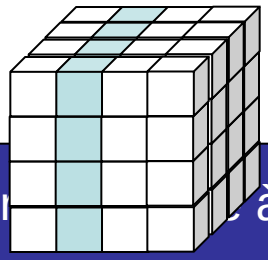
Opération agissant sur la structure

- Tranchage (slicing): consiste à ne travailler que sur une tranche du cube. Une des dimensions est alors réduite à une seule valeur

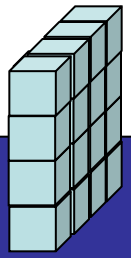
		09	10	11
Œuf	Idf	220	265	284
	59	225	245	240
Viande	Idf	163	152	145
	59	187	174	184

→

		09
Œuf	Idf	265
	59	245
Viande	Idf	152
	59	174



→



Manipulation des données multidimensionnelles

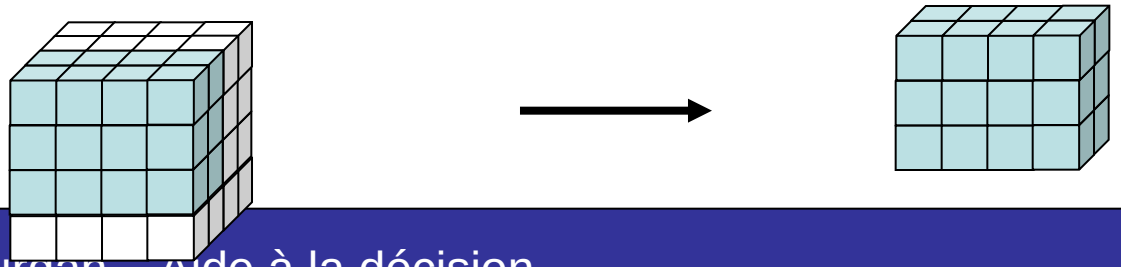
Opération agissant sur la structure

- Extraction d'un bloc de données (dicing): ne travailler que sous un sous-cube

		09	10	11
Œuf	Idf	220	265	284
	59	225	245	240
Viande	Idf	163	152	145
	59	187	174	184

→

		09	10	11
Œuf	Idf	220	265	284
	59	225	245	240

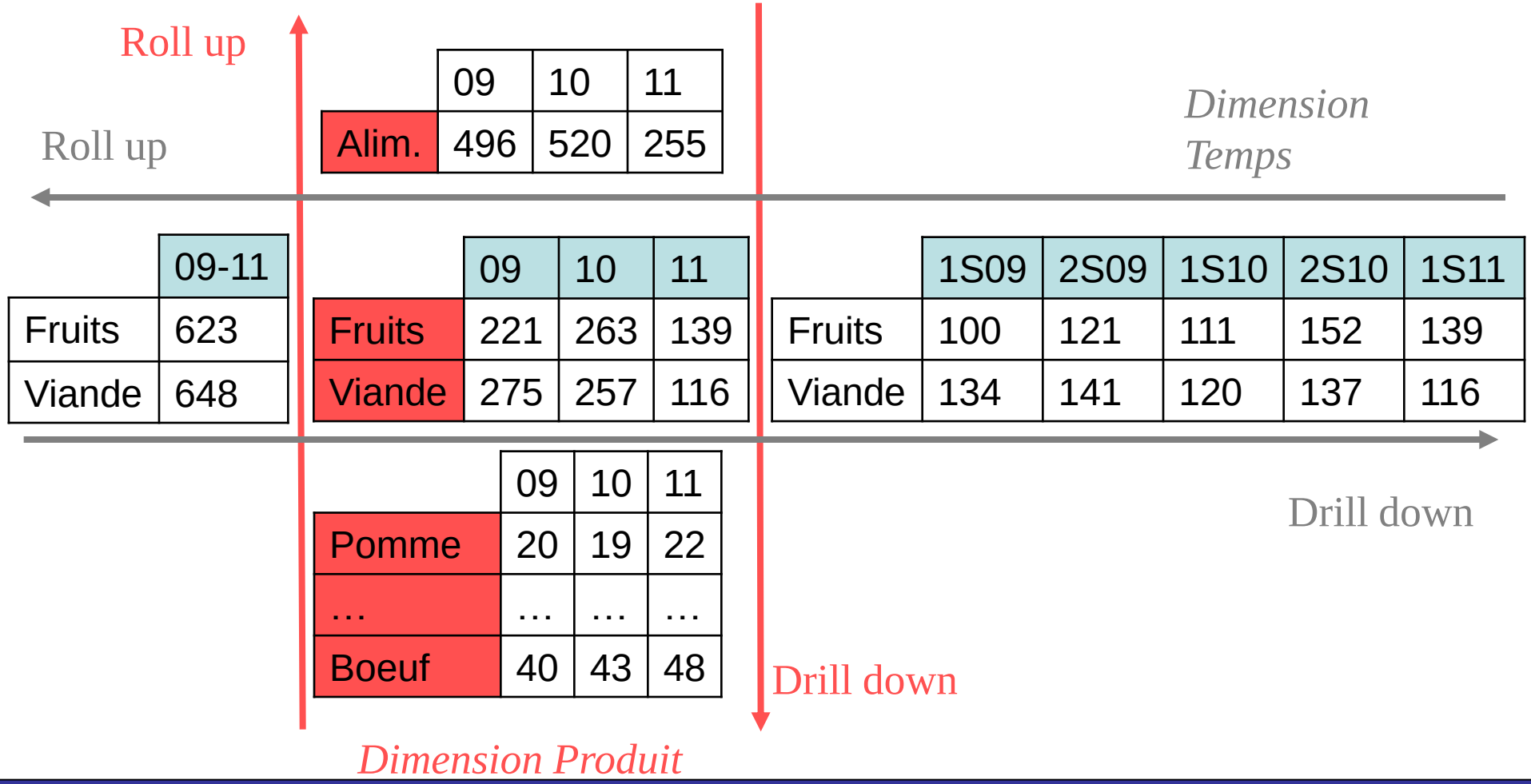


Manipulation des données multidimensionnelles

Opération agissant sur la granularité

- Forage vers le haut (roll-up): « dézoomer »
  - Obtenir un niveau de granularité supérieur
  - Utilisation de fonctions d'agrégation
- Forage vers le bas (drill-down): « zoomer »
  - Obtenir un niveau de granularité inférieur
  - Données plus détaillées

Drill-up, drill-down



	09	10	11
Alim.	496	520	255

	09-11
Fruits	623
Viande	648

	09	10	11
Fruits	221	263	139
Viande	275	257	116

	1S09	2S09	1S10	2S10	1S11
Fruits	100	121	111	152	139
Viande	134	141	120	137	116

	09	10	11
Pomme	20	19	22
…	…	…	…
Boeuf	40	43	48

Drill

L’opération du Drill peut se décliner en plusieurs autres opérations :

• **Drill accross :** Drill latéral, comparaison sur des mesures dans plusieurs tables

de faits ;

• **Drill through :** voir l’information à travers plusieurs dimensions

• **Reach through :** voir l’information en profondeur, jusqu’aux données de base.

OLAP

Constitution de l'Hypercube

- Administration
- Définition des Dimensions / Niveaux / Membres
  - Automatique, Manuel, Configuration Métier

Serveurs OLAP / Clients OLAP

- Le client utilise une partie de l'hypercube qu'il cache
- Le serveur calcule, stocke l'hypercube et permet son partage.

Stockage

- M-OLAP : accède à une base multidimensionnelle
  - + rapidité
- R-OLAP : accède à une base relationnelle
  - + mise à jour
- H-OLAP : hybride, multidimensionnel avec accès au niveau le
  - + bas à une base relationnelle

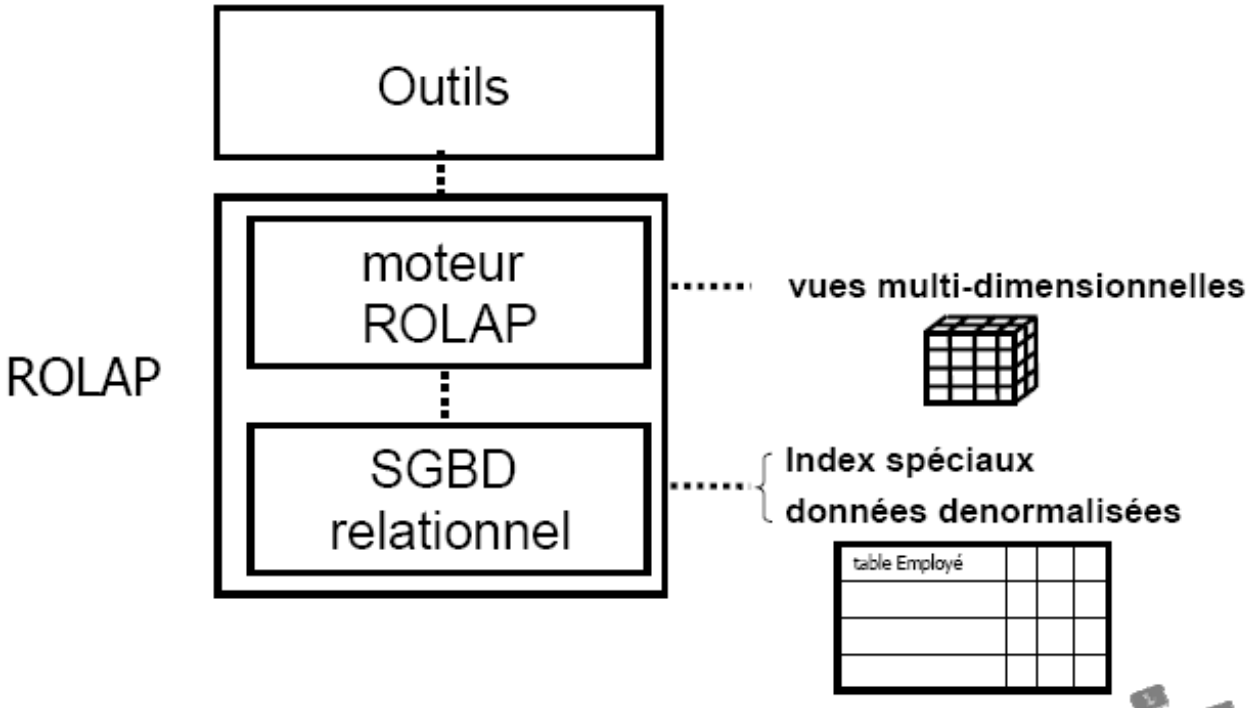
ROLAP

Idée:

- Données stockées en relationnel.
- La conception du schéma est particulière: schéma en étoile, schéma en flocon
- Des vues (matérialisées) sont utilisées pour la représentation multidimensionnelle
- Les requêtes OLAP (slice, rollup…) sont traduites en SQL.
- Utilisation d'index spéciaux: bitmap
- Administration (tuning) particulier de la base

Avantages/inconvénients

- Souplesse, évolution facile, permet de stocker de gros volumes.
- Mais peu efficace pour les calculs complexes


Outils
moteur
ROLAP
SGBD
relationnel
ROLAP
vues multi-dimensionnelles
Index spéciaux
données denormalisées
table Employé

MOLAP

Idée:

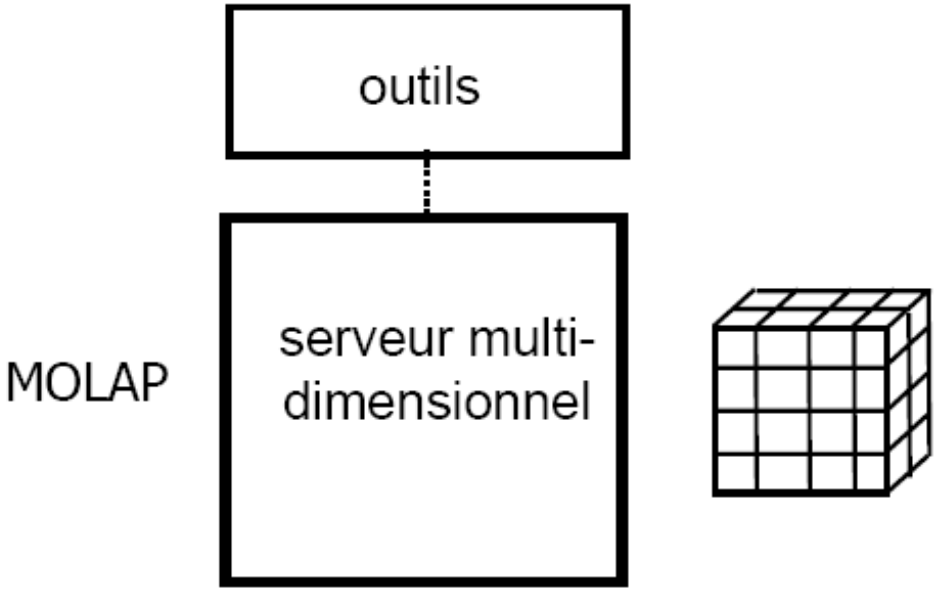
- Modélisation directe du cube
- Ces cubes sont implémentés comme des matrices à plusieurs dimensions
- CUBE [1:m, 1:n, 1:p…] (mesure)
- Le cube est indexé sur ses dimensions

Avantages/inconvénients:

- rapide
- formats propriétaires
- ne supporte pas de très gros volumes de données

MOLAP

- Multi-Dimensional OLAP



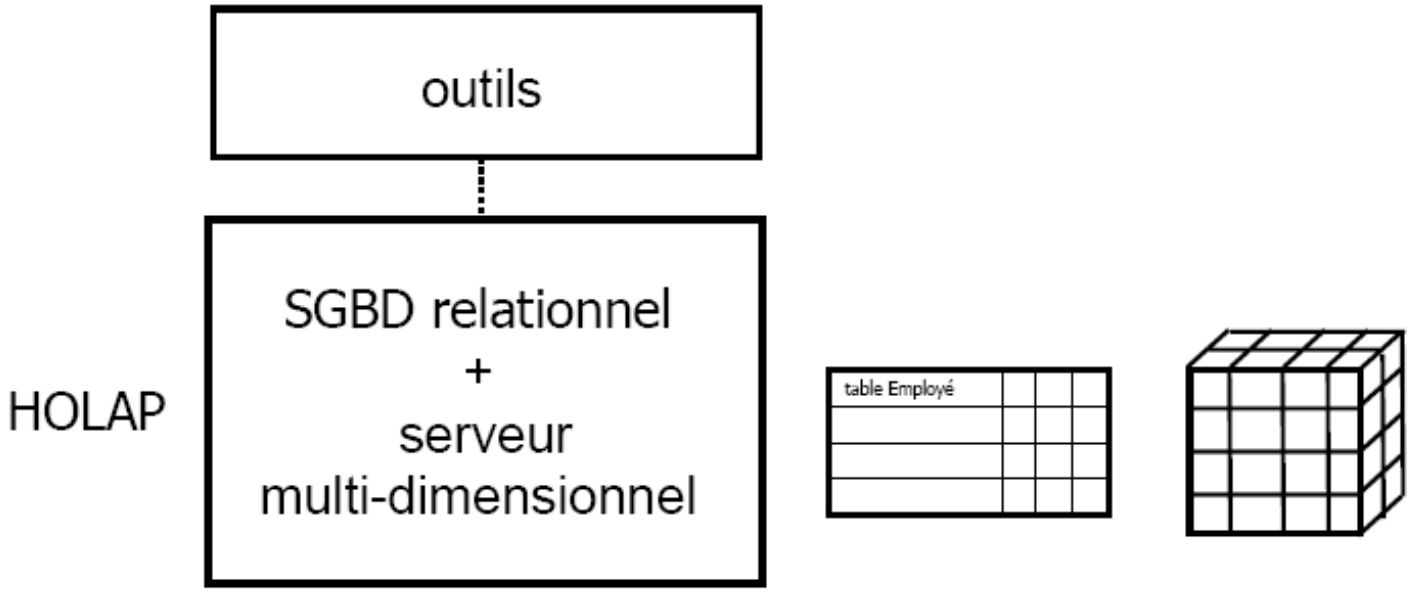
HOLAP

Idée:

- MOLAP + ROLAP
- Données stockées dans des tables relationnelles
- Données agrégées stockées dans des cubes.
- Les requêtes vont chercher les données dans les tables et les cubes

HOLAP

- Hybrid OLAP



Restitution des informations

Requêteurs

- donne une réponse à une question plus ou moins complexe (type SQL)

EIS (Executive Information Systems)

- outils de visualisation et de navigation dans les données

Statistiques + interfaçage graphique

Applications spécialisées (ad-hoc)

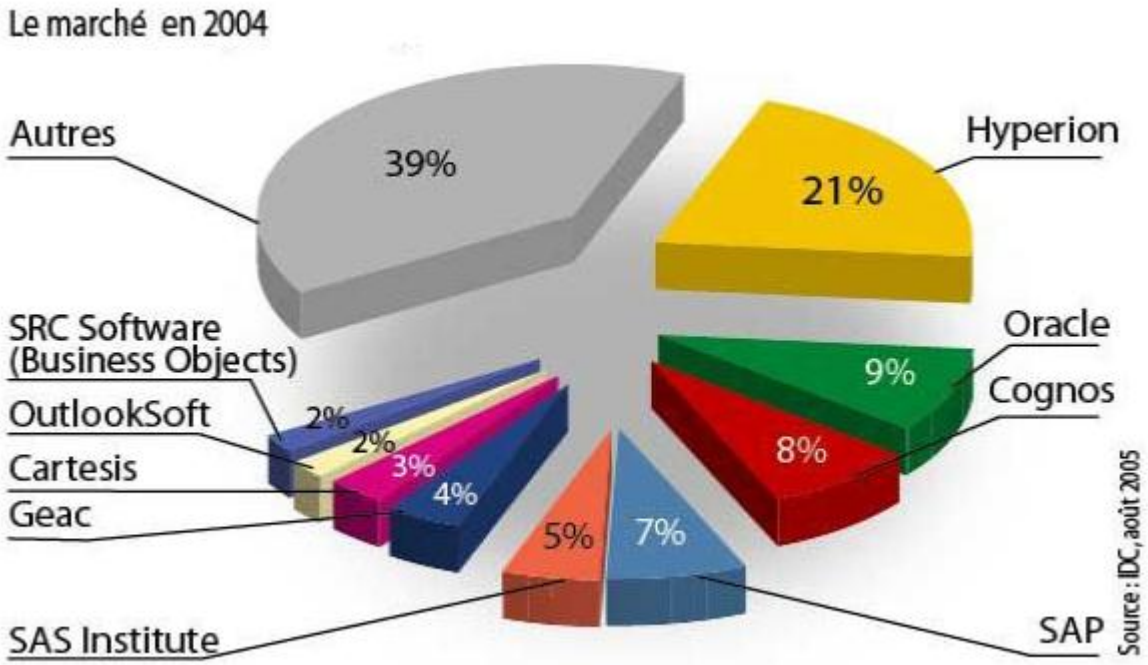
- applications développées spécialement pour les besoins de l'entreprise

Data Mining

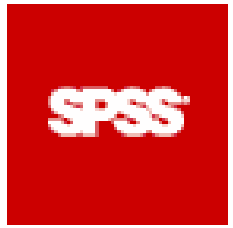
- outils évolués de prédiction, simulation, ...

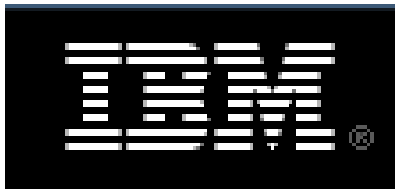
Outils

Le marché du décisionnel



Quelques solutions commerciales





Constitution du DataWarehouse

Administration

- SourcePoint (Software AG), ISM/OpenMaster (Bull), CA-UniCenter, DataHub (IBM), CPE (SAS), Warehouse Administrator (SAS)

Extraction et Purification

- Warehouse Manager (Prism), Integrity Data Reengineering (Vality), Access (SAS), DataStage (VMark), Génio (Léonard's Logic), InfoRefiner (Platinum), PASSPORT et NATURAL (Software AG), Gentia (Planning Sciences)

Stockage

DataWarehouse

- Oracle, Sybase, Informix, Ingres (CA), DB2 (IBM), Tandem, Teradata, ...

Serveur OLAP

- Express (Oracle), Business Objects, Powerplay / Impromptu (Cognos), Adabas (Software AG), Opera (CFI), ALEA (MIS AG), Harry Cube (Adviseurs), Gentia (Planning Sciences), Essbase (Arbor Software), Informix, Pilot, ...

Extraction d'Information

Rétro-ingénierie (Reverse-Engineering)

- Business Object, DB-Main

Browser OLAP

- Discoverer (Oracle), ESPERANT (Software AG), InfoBeacon (Platinum), Explorer (Business Objects), le VCL DecisionCube de Delphi Cl/Sv

Quelques solutions open source

ETL	Entrepôt de données	OLAP	Reporting	Data Mining
■Octopus ■Kettle ■CloverETL ■Talend	■MySql ■Postgresql ■Greenplum/Bizgres	■Mondrian ■Palo	■Birt ■Open Report ■Jasper Report ■JFreeReport	■Weka ■R-Project ■Orange ■Xelopes

Intégré
■Pentaho (Kettle, Mondrian, JFreeReport, Weka)
■SpagoBI

Préparation de données

Type de données

- Continues
- Nominales
- Type discret ordonné (ordinales)
- Dates
- Texte
- Formule sur plusieurs attributs

Nettoyage

- Données manquantes
- Bruitées
- Inconsistances

Nettoyage : données manquantes

Fiches mal remplies.

Donnée non disponible.

Pas de sens pour cet exemple.

Stratégies :

- Ignorer l'exemple.
- Ajouter une valeur.
- Remplacer par la valeur majoritaire (moyenne/médiane).
- Remplacer par la valeur majoritaire (moyenne/médiane) chez les exemples de la même classe.
- Laisser.
- Trouver la 'meilleure' valeur par apprentissage.

Valeurs manquantes

Certains sont naturellement robustes aux valeurs manquantes (VM)

- Bayes Naive : VM ignorées dans le calcul des probas
- K-NN : VM ignorées dans le calcul de distances

D'autres intègrent des mécanismes spéciaux pour traiter les VM

- CART : dichotomies de substitution (surrogate splits)
- C5.0 : fractionnement des cas + modif de la mesure de gain

D'autres algorithmes : pas de stratégie générique vis-à-vis des VM. Ex. réseaux neuronaux. Dépend de chaque implémentation

- SNNS : demande des données complètes. A l'utilisateur de traiter les VM
- Clementine : mécanisme simple d'imputation des VM

Traitement des valeurs manquantes

Supprimer

- Les cas ayant des valeurs manquantes
- Les variables ayant les valeurs manquantes

Remplir les valeurs manquantes

- Imputation par le mode (variables nominales) / la moyenne (variables continues). Globalement ou par classe (mieux). Biaise la distribution de la variable (sous-estime la variance).
- Imputation par régression : régression linéaire sur les données complètes (biais introduit si relation non linéaire)
- Imputation par apprentissage automatique :
  - Prendre un algorithme robuste aux VM (C5, Naive Bayes, CART, …)
  - Transformer la variable ayant des VM en variable cible

Discrétisation de variables continues

Certains algorithmes nécessitent des variables catégoriques.

- Ex : A-priori

D'autres algorithmes donnent de meilleurs résultats sur certaines données lorsque toutes les variables sont catégoriques.

- Ex : C5, Bayes Naive, …

Discrétisation de variables continues

Discrétisation supervisée / non supervisée

- Non supervisée : segmente les valeurs en K intervalles sans tenir compte d'une variable cible (K prédéfini)
  - Binning : répartissent les valeurs en K intervalles de longueur ou de fréquence égale.
  - Clustering : ex. K-means. On peut déterminer K automatiquement
- Supervisée : exploite la variable de classe pour segmenter les valeurs

Discrétisation de variables continues

Discrétisation par scission / fusion

- Scission (splitting) : commencer par un intervalle et diviser itérativement. Ex. : Algorithme Disc_EntMDL : Maximiser le gain d'information (Entropie, MDL : minimum description length).
- Fusion (merging) : commencer par autant d'intervalles que de valeurs distinctes, puis fusionner itérativement. Ex. : Algorithme Disc_ChiMerge (test de khi2)
- En mode non supervisée, cette distinction = clustering hiérarchique (divisif/agglomératif)

Données bruitées

Erreurs de mesure.

Fautes de frappe.

Incertitude des mesures :

bruit 'invisible' → lisser

Intrus (outliers) → détecter

Comment s'en rendre compte ?

- Sondage
- Comparaison de la distribution avec une population similaire
- Outils statistiques et visualisation

Il est important de se familiariser avec les données :

- Outils statistiques integrés (OLAP , weka . . .)
- Boîte à outils autonome (R, ggobi,SAS)
- Statisticiens.

Lissage

Diviser l'intervalle de valeurs en casiers (bins, buckets)

Dans chaque paquet, modifier les valeurs :

- centrer les valeurs (médiane).
- Prendre les valeurs les extrémités.

Exemple :

1,1,2,3,4,6,8,9,12,57

- En trois paquets de même largeur :
  118 : 1,1,2,3,4,6,8,9 → 4
  1936 : 12 → 12
  3757 : 57 → 57

- En trois paquets de même cardinal :
  1,1,2
  3,4,6,8
  9,12,57

Inconsistence

Exemples :

- un poids de 100kg pour une taille d'un mètre.
- Un pic montagneux au milieu d'une plaine.
- Un nombre de grossesses non nul pour un homme
- Médicament déconseillé pour une pathologie

Fixer des règles de cohérence :

- dès le départ (expert du domaine).
- Quand on a remarqué des valeurs 'louches'.

Enrichissement

Ajouter des informations :

- Propres à chaque exemple.
- Résultat de statistiques ou de fouilles précédentes.

Problèmes :

- Formats des données.
- Cohérence avec les données connues.
- Unités.
- Actualité des données.

Transformation

Lissage.

Regroupement d'attributs (boutons à cocher).

Généralisation, hiérarchisation : Ville → région → pays

Normalisation.

Calcul de nouveaux attributs.

Réduction

Réduire la complexité des données → réduire la complexité du calcul (temps, espace)

- Regrouper plusieurs attributs.
- Supprimer les attributs sans intérêt.
- Discrétiser les valeurs continues.
- Echantillonner l'ensemble d'apprentissage.

Sélection d'attributs

Les méthodes de DM face aux variables non pertinentes

- Très résistantes aux variables non pertinentes. Ex : Bayes Naive
- Assez résistentes aux variables non pertinentes. Ex : C5
- Peu résistentes aux variables non pertinentes. Ex : Réseaux neuronaux

Sélection d'attributs : Pourquoi ?

Eliminer les variables non pertinentes

- Augmentent l'erreur de prédiction

Eliminer les variables redondantes

- Non utiles à la prédiction, augmentent le temps de calcul

Trop de variables nuisent à la compréhensibilité

Sélection d'attributs

Articulation sur le processus de DM : imbrication, filtrage, emballage

Critère de sélection de variables : mesures de pertinence, de performance

Stratégie de sélection

- Approche univariée : évaluer chaque variable suivant un critère choisi, puis sélectionner les p meilleures. Approche simpliste : ignore toute interaction de variables
- Approche multivariée : à chaque étape, on choisit un sous-ensemble des variables candidates. Pose le problème de stratégie de recherche.

Sélection d'attributs : critères de sélection

Sélection par filtrage : mesure de pertinence à la variable cible

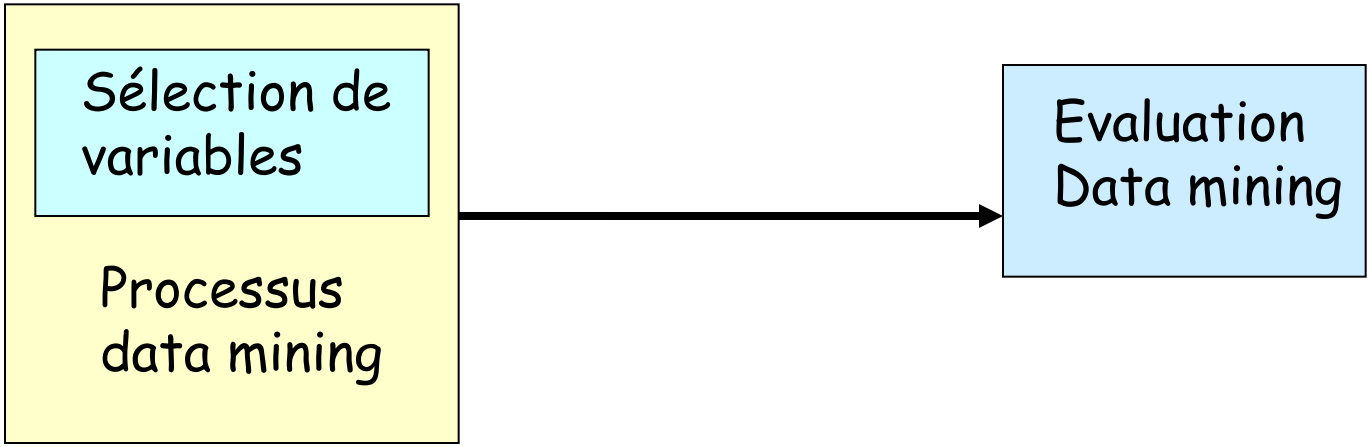
- Basées sur Khi2
- Basées sur l'entropie
- Basées sur l'index Gini, …

Sélection par emballage : mesure des performances du modèle appris

- Taux de bonnes réponses
- Erreur quadratique moyenne, …

Articulation sur le processus de DM

Sélection par imbrication : le processus de sélection est incorporée dans l'algorithme de DM

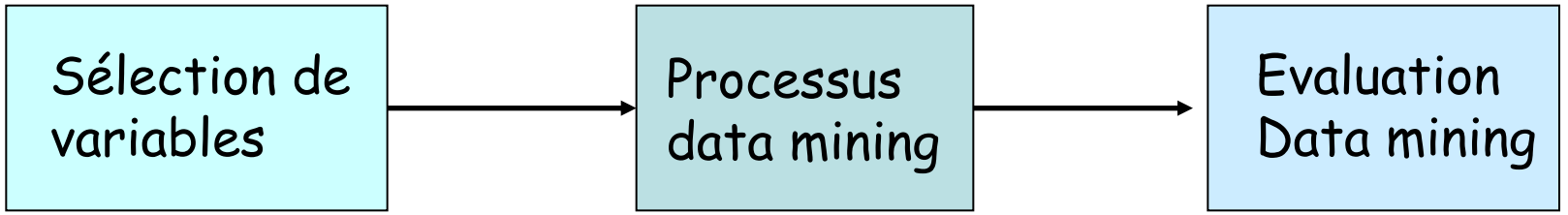


Ex : C5 : choisit le variable de test à chaque niveau de l'arbre

- Critère de sélection : information mutuelle entre variable et classe
- Inconvénient : évalue les variables individuellement sans tenir compte de leur interaction

Articulation sur le processus de DM

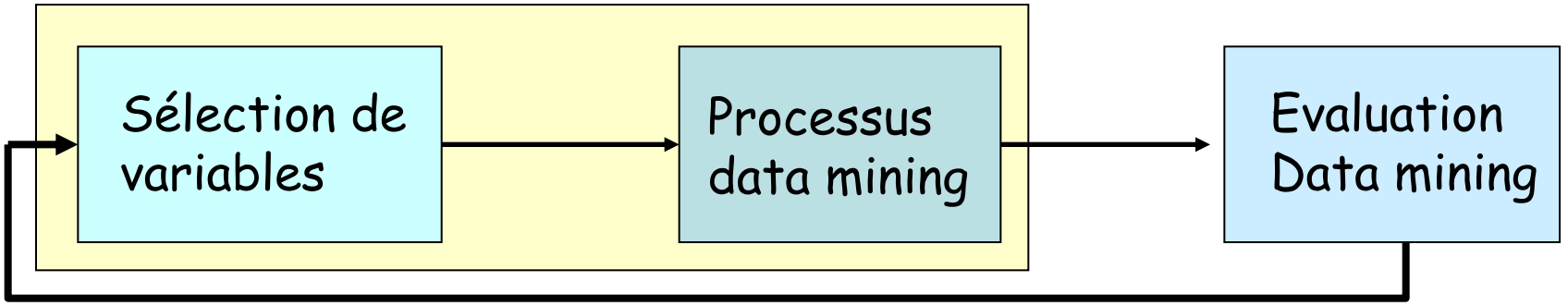
Sélection par filtrage : la sélection se fait avant et indépendamment du processus de DM



Articulation sur le processus de DM

Sélection par emballage : la sélection de variables englobe le processus de DM

- Critère de sélection : performances du modèle appris
- Coût de calcul très élevé car boucle sur le processus d'apprentissage



Sélection d'attributs : critères de sélection

Sélection par filtrage : mesure de pertinence à la variable cible

- Basées sur Khi2
- Basées sur l'entropie
- Basées sur l'index Gini, …

Sélection par emballage : mesure des performances du modèle appris

- Taux de bonnes réponses
- Erreur quadratique moyenne, …

Validation

Résultats non triviaux (validation par l'utilisateur).

Évaluation correcte de l'erreur.

Trouve la classe d'un exemple avec une bonne efficacité.

Regroupe les exemples en 'paquets' correspondant à une réalité.

Réferences

- “Entrepôts de données : guide pratique du concepteur de data
- warehouse”, R.Kimball, International Thomson Publishing, 1997.
- “Concevoir et déployer un data warehouse”, R.Kimball, L.Reeves,
- M.Ross, W.Thornthwaite, Eyrolles, 2000.
- “Bases de données : objet et relationnel”, G.Gardarin, Eyrolles, 1999.
- “ The Data Warehousing Information Center ”, L.Greenfield,
- www.dwinfocenter.org

Fouille de données

Introduction au Data Mining

Définition du Data Mining : fouille de données, extraction de connaissances, ...

Pourquoi le Data Mining ?

Description du processus KDD (Knowledge Data Discovery)

Applications

Tâches et Techniques du Data Mining

Qu'est-ce que le DM ?

Processus inductif, *itératif* et *interactif* de découverte dans les BD larges de modèles de données *valides*, *nouveaux*, *utiles* et *compréhensibles*.

- Itératif : nécessite plusieurs passes
- Interactif : l'utilisateur est dans la boucle du processus
- Valides : valables dans le futur
- Nouveaux : non prévisibles
- Utiles : permettent à l'utilisateur de prendre des décisions
- Compréhensibles : présentation simple

Notion d'induction [Peirce 1903]

Induction : Généralisation d'une observation ou d'un raisonnement établis à partir de cas singuliers.

Utilisée en Data mining (tirer une conclusion à partir d'une série de faits, pas sûre à 100%)

- La clio a 4 roues, La Peugeot 106 a 4 roues, La BMW M3 a 4 roues, La Mercedes 190 a 4 roues
- ==> Toutes les voitures ont 4 roues

Motivations (1)

Explosion des données

- Masse importante de données (millions de milliards d'instances) : elle double tous les 20 mois.
  - BD très larges - Very Large Databases (VLDB)
- Données multi-dimensionnelles (milliers d'attributs)
  - BD denses
- Inexploitables par les méthodes d'analyse classiques
- Collecte de masses importantes de données (Gbytes/heure)
  - Données satellitaires, génomiques (micro-arrays, …), simulations scientifiques, etc.
- Besoin de traitement en temps réel de ces données

Motivations (2)

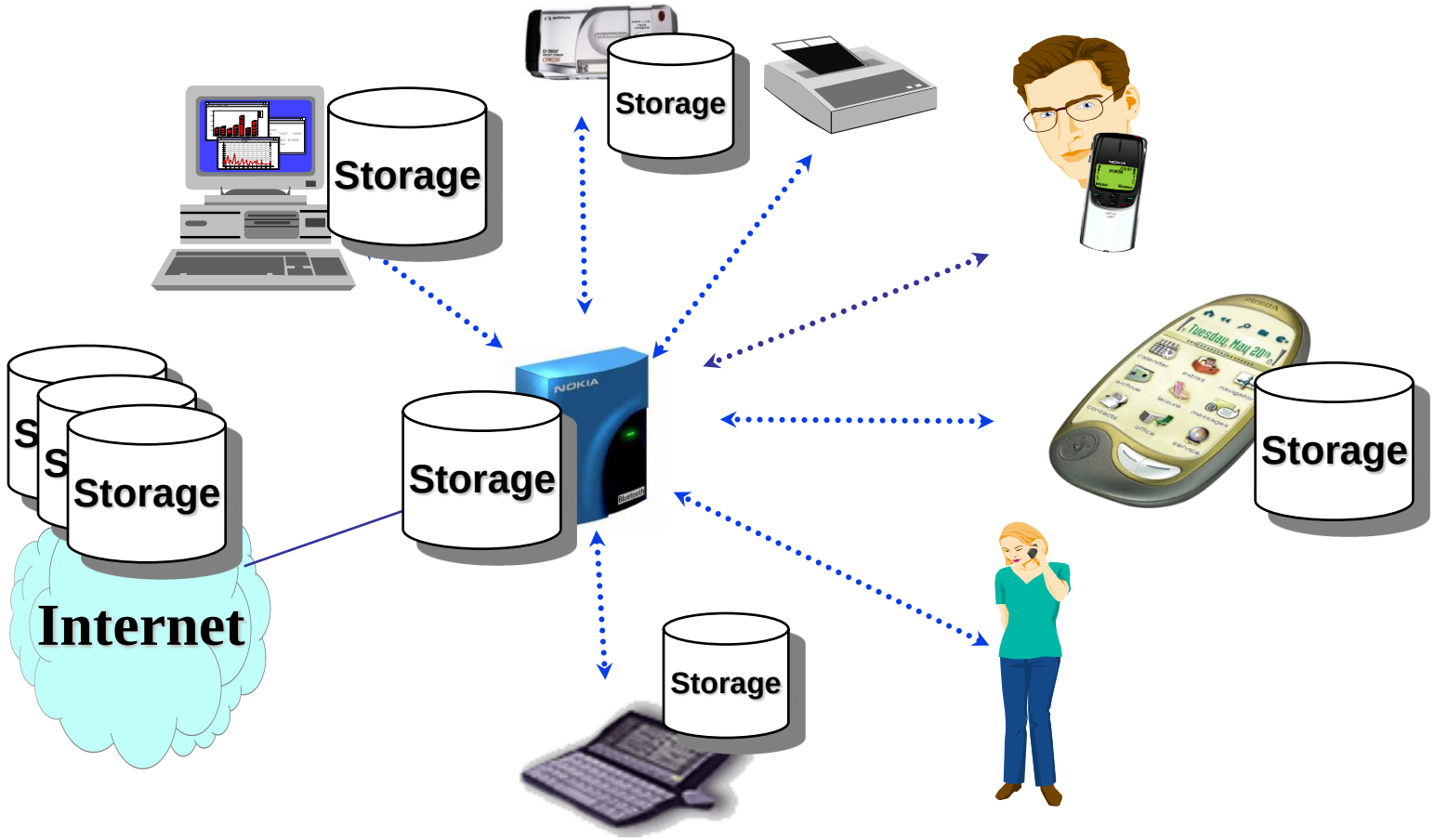
Améliorer la productivité

- Forte pression due à la concurrence du marché
- Brièveté du cycle de vie des produits
- Besoin de prendre des décisions stratégiques efficaces
  - Exploiter le vécu (données historiques) pour prédire le futur et anticiper le marché
  - individualisation des consommateurs (dé-massification).

Croissance en puissance/coût des machines capables

- de supporter de gros volumes de données
- d'exécuter le processus intensif d'exploration
- hétérogénéité des supports de stockage

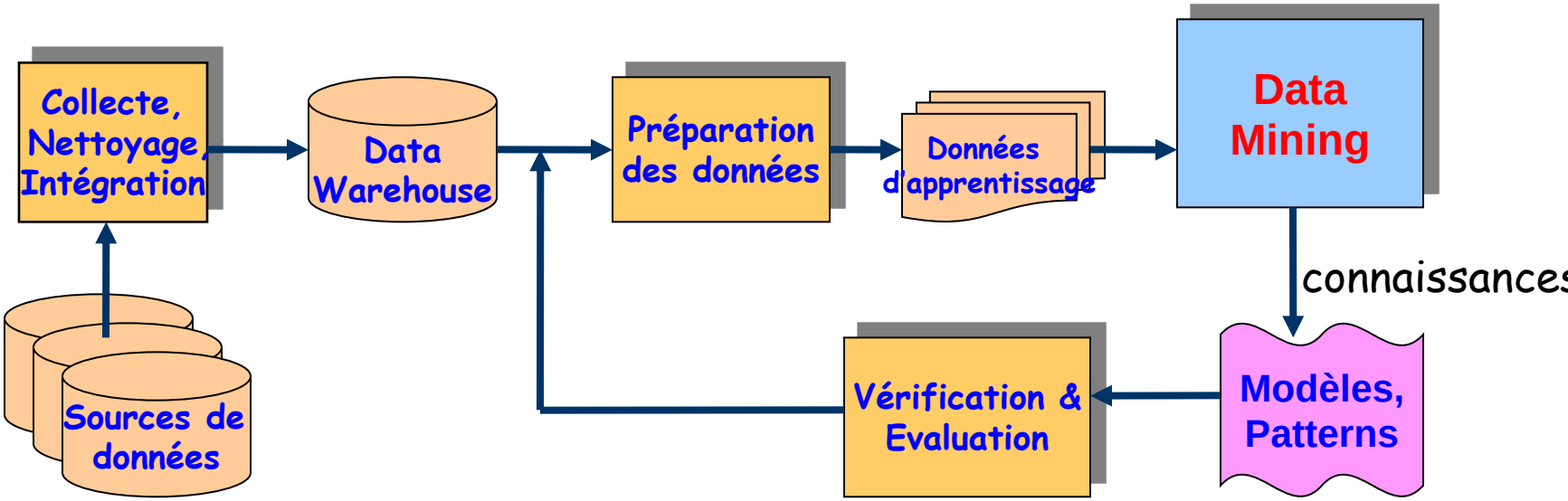
Motivations (3)



Masse importante de données – supports hétérogènes

Le processus de découverte de connaissances

- Data mining : coeur de KDD (Knowledge Data Discovery).



Modèle : résumé global de l'ensemble des données
Motif (pattern) : résumé local d'une région de l'espace des données (ex : règle a→b)

Démarche méthodologique (1)

Comprendre l'application

- Connaissances *a priori*, objectifs, etc.

Sélectionner un échantillon de données

- Choisir une méthode d'échantillonnage

Nettoyage et transformation des données

- Supprimer le «bruit» : données superflues, marginales, données manquantes, etc.
- Effectuer une sélection d'attributs, réduire la dimension du problème, discrétisation des variables continues, etc.

Appliquer les techniques de fouille de données (DM)

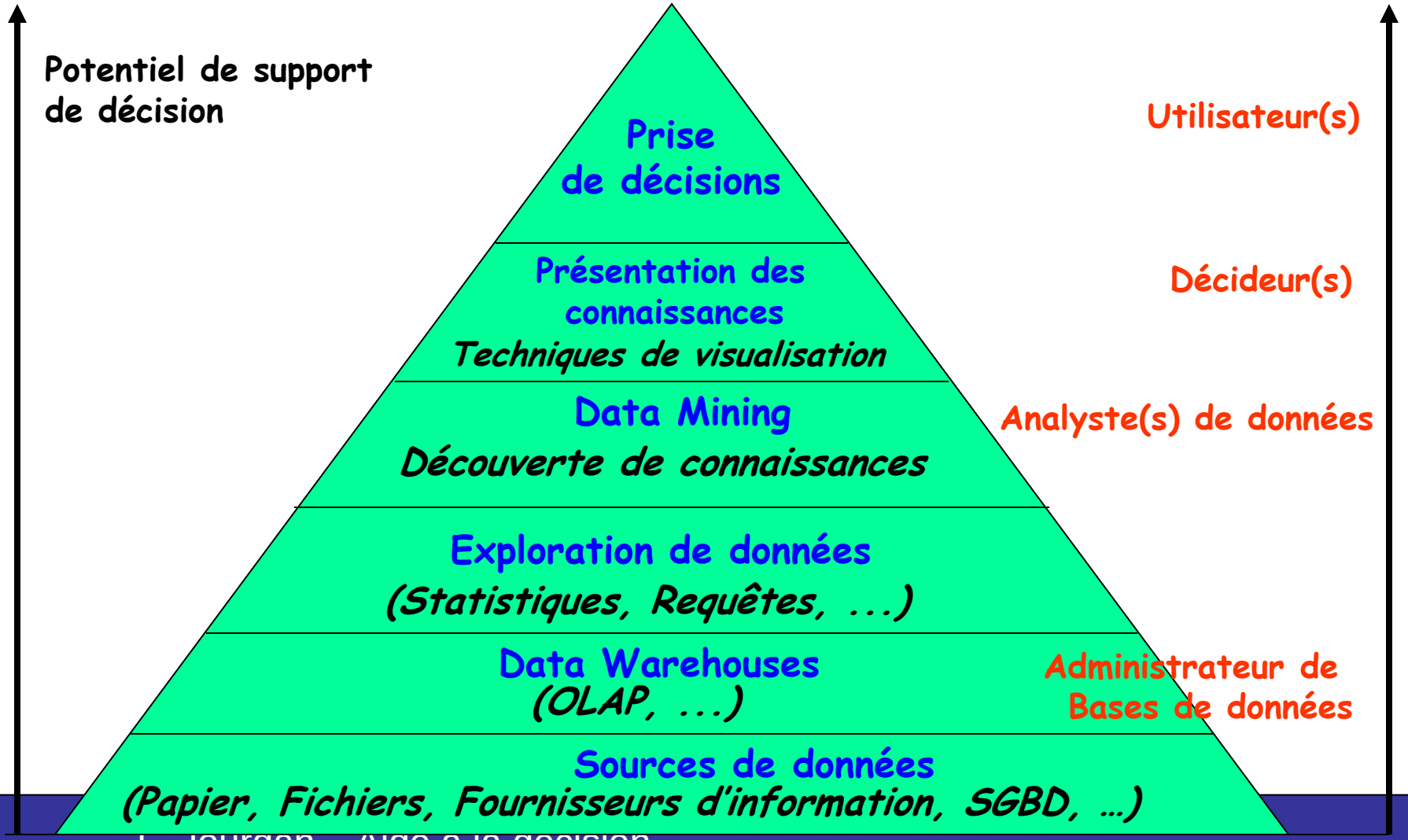
- le cœur du KDD
- Choisir le bon modèle et le bon algorithme

Démarche méthodologique (2)

- Visualiser, évaluer et interpréter les modèles découverts
  - Analyser la connaissance (intérêt, critères d'évaluation)
  - Compréhensibilité souvent capitale
  - Vérifier sa validité (sur le reste de la base de données)
  - Réitérer le processus si nécessaire

- Gérer/déployer la connaissance découverte
  - La mettre à la disposition des décideurs
  - L'échanger avec d'autres applications (système expert, …)
  - etc.

Data Mining et aide à la décision



Objectifs

- Développer des techniques et systèmes *efficaces* et *extensibles* pour l'exploration de :
  - BD larges et multi-dimensionnelles
  - Données distribuées

- Faciliter l'utilisation des systèmes de DM
  - Limiter l'intervention de l'utilisateur
  - Représentation simple de la connaissance
  - Visualisation sous forme exploitable

Scalability : Extensibilité

Problème d'extensibilité dû au volume sans cesse croissant des données (exemples, attributs)

Quelques ordres de grandeur

- AT& T : 350 M de numéros de téléphone, 250-350 M appels / jours
- Auchan : qqs M Tickets de caisse / mois
- SKICAT : classification de qqs M d'étoiles et de galaxies à partir d'images
- Génome : plusieurs milliers de gènes.
- Web mining : 1-2 M sessions Internet / jour
- …

Scalability : Solutions

Côté données

- Réduire le nombre d'exemples (échantillonage)
- Réduire le nombre d'attributs (sélection d'attributs, transformation de variables)

Côté Algorithmes

- Algorithmes efficaces (complexité)
- Algorithmes incrémentaux
- Algorithmes parallèles, out-of-core (pouvant traiter des données non résidentes)

Sources du Data Mining

Intelligence artificielle : apprentissage symbolique, reconnaissance de formes, réseaux de neurones, …

Bases de données : VLDB, BD relationnelles, OLAP, Entrepôts de données, …

Statistiques : Analyse exploratoire, Modélisation paramétrique / non paramétrique

Communautés impliquées

- Intelligence artificielle et apprentissage automatique
- Bases de données
- Analyse de données (statistiques)
- Visualisation
- Recherche opérationnelle et optimisation
- Informatique parallèle et distribuée (HPC)
- Analyse d'images (image mining)
- Recherche d'infos, traitement langage naturel (web mining, text mining)

Data Mining et Statistiques

Data mining # Statistiques

Data mining : Exploratoire, Data-driven modeling – Découverte de nouvelles connaissances

Statistiques : Confirmatoire, User-driven modeling – Vérification d'hypothèses

Distribution d 'une seule variable : moyenne, médiane, variance, écart-type, …

Explorer les relation entre variables : coefficient de corrélation, …

Découverte de la cause des relations entre de nombreuses variables est assez complexe.

test du X2, ...

Réseaux bayésiens (probabilités conditionnelles)

Data Mining et autres

Data mining # apprentissage automatique

Apprentissage automatique : Données pas forcément prêtes, pas forcément massives

Data mining : suppose la pré-existence de très grands volumes de données

Data mining # Entrepôts de données

Entrepôt de données : base de données résumant diverses BD transactionnelles pour servir de support à la prise de décision

Data mining : effectué sur des entrepôts de données aussi bien que sur des BD transactionnelles.

Data mining # OLAP (online analytical processing)

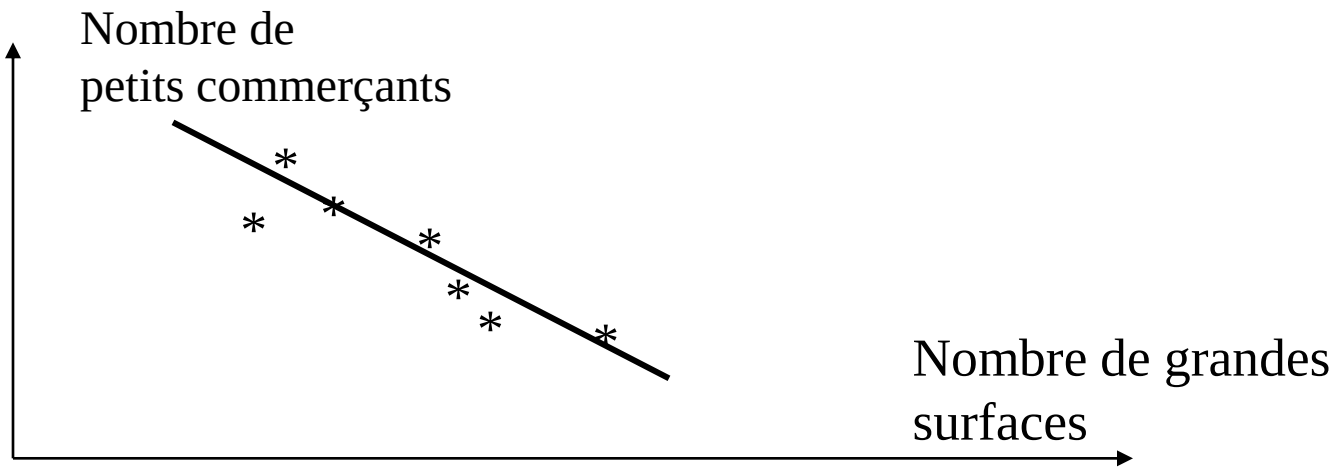
OLAP : interactif, piloté par l'utilisateur (data mining manuel)

Data mining : largement automatisé

Découverte de modèles fonctionnels

Méthodes de régression :

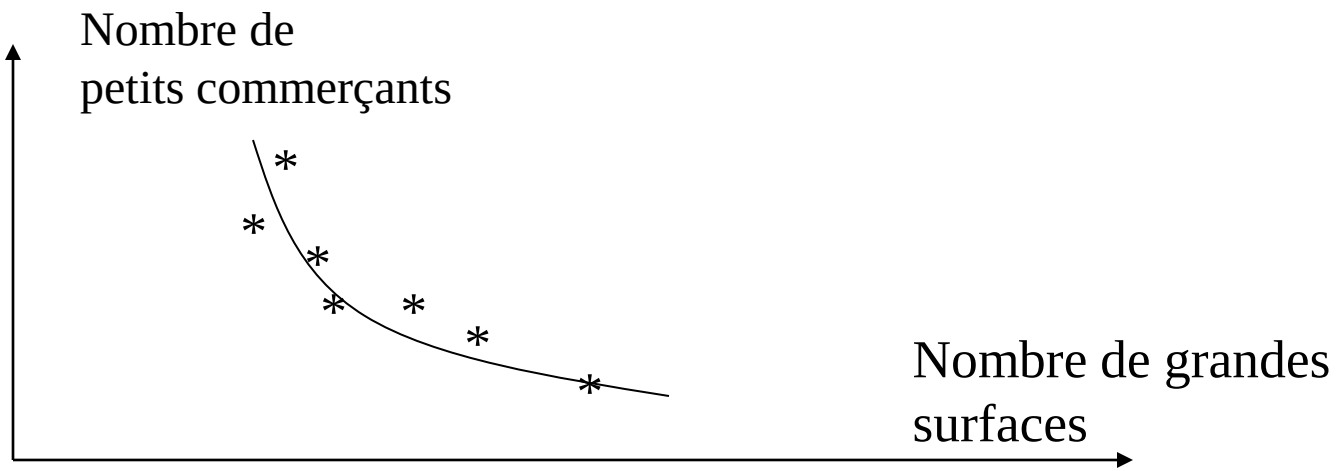
- régression linéaire : Y = aX+ b (a, b : valeurs réelles)



  - Rapide et efficace (valeurs réelles)
  - Insuffisante pour l ’analyse d’espace multidimentionnel

Découverte de modèles fonctionnels

Kernel regression : découvrir graphiquement la fonction à utiliser, peut être une courbe



Techniques statistiques inadéquates : nombre de facteurs important, modèles non linéaires.

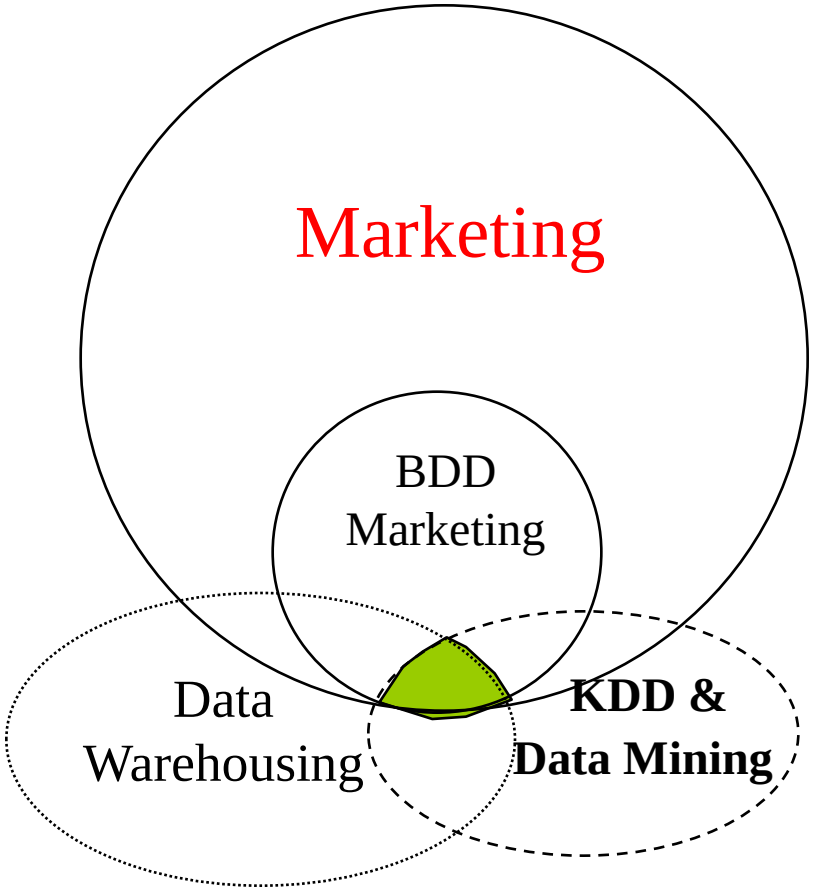
Domaines d'application

Prise de décision basée sur de nouvelles connaissances

Ex., impact sur le marketing

Le rôle et l'importance du KDD et DM est de plus en plus important

Mais le DM n'est pas seulement dans le marketing...



Domaines d'application

- Marketing direct : population à cibler (âge, sexe, profession, habitation, région, …) pour un publipostage.
- Gestion et analyse des marchés : Ex. Grande distribution : profils des consommateurs, modèle d 'achat, effet des périodes de solde ou de publicité, « panier de la ménagère »
- Détection de fraudes : Télécommunications, ...
- Gestion de stocks : quand commander un produit, quelle quantité demander, …
- Analyse financière : maximiser l 'investissement de portefeuilles d 'actions.

Domaines d'application

Gestion et analyse de risque : Assurances, Banques (crédit accordé ou non)

Compagnies aériennes

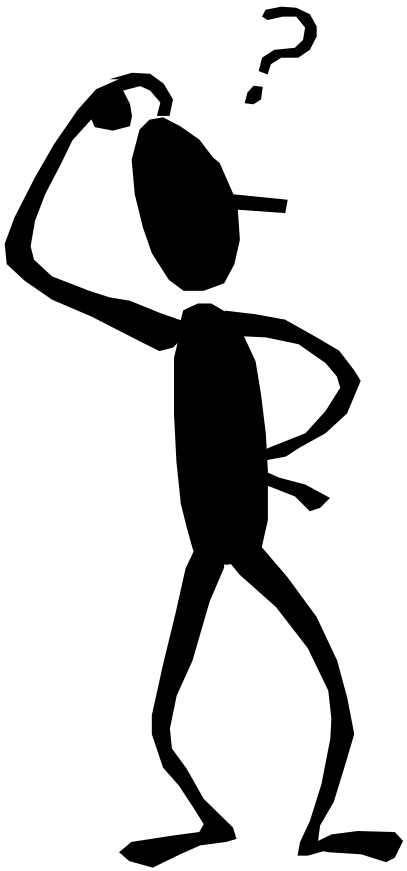
Bioinformatique et Génome : ADN mining, …

Médecine et pharmacie :

- Diagnostic : découvrir d 'après les symptomes du patient sa maladie
- Choix du médicament le plus approprié pour guérir une maladie donné

Web mining, text mining, etc.

Exemple 1 - Marketing



Vous êtes gestionnaire marketing d'un opérateur de télécommunications mobiles :

- Les clients reçoivent un téléphone gratuit (valeur 150€) avec un contrat d'un an ; vous payer une commission de vente de 250€ par contrat
- Problème : Taux de renouvellement (à la fin du contrat) est de 25%
- Donner un nouveau téléphone à toute personne ayant expirer son contrat coûte cher.
- Faire revenir un client après avoir quitter est difficile et coûteux.

Exemple 1 - Marketing

Trois mois avant l'expiration du contrat, prédire les clients qui vont quitter :

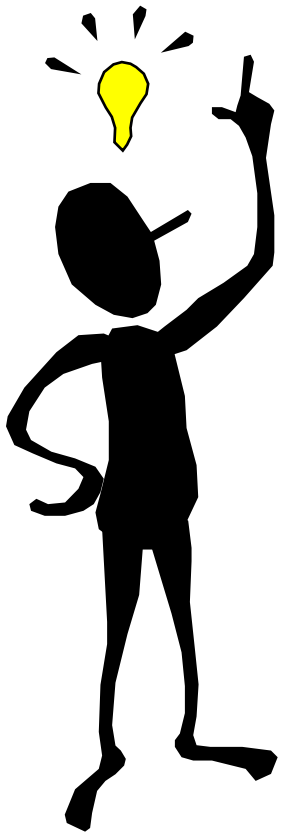
- Si vous voulez les garder, offrir un nouveau téléphone.

Exemple 2 - Assurances

Vous êtes un agent d'assurance et vous devez définir un paiement mensuel adapté à un jeune de 18 ans qui a acheté une Ferrari.

Qu'est ce qu'il faut faire ?

Exemple 2 - Assurances



Analyser les données de tous les clients de la compagnie.

La probabilité d'avoir un accident est basée sur … ?

- Sexe du client (M/F) et l'âge
- Modèle de la voiture, âge, adresse, ...
- etc.

Si la probabilité d'avoir un accident est supérieure à la moyenne, initialiser la mensualité suivant les risques.

Exemple 3 – Banque - Télécom

Vous êtes à l'étranger et quelqu'un a volé votre carte de crédit ou votre mobile …

compagnies bancaires …

- Utiliser les données historiques pour construire un modèle de comportement frauduleux et utiliser le data mining pour identifier des instances similaires.

compagnies téléphoniques …

- Analyser les "patterns" qui dérivent du comportement attendu (destinataire, durée, etc.)

Exemple 4 - Web

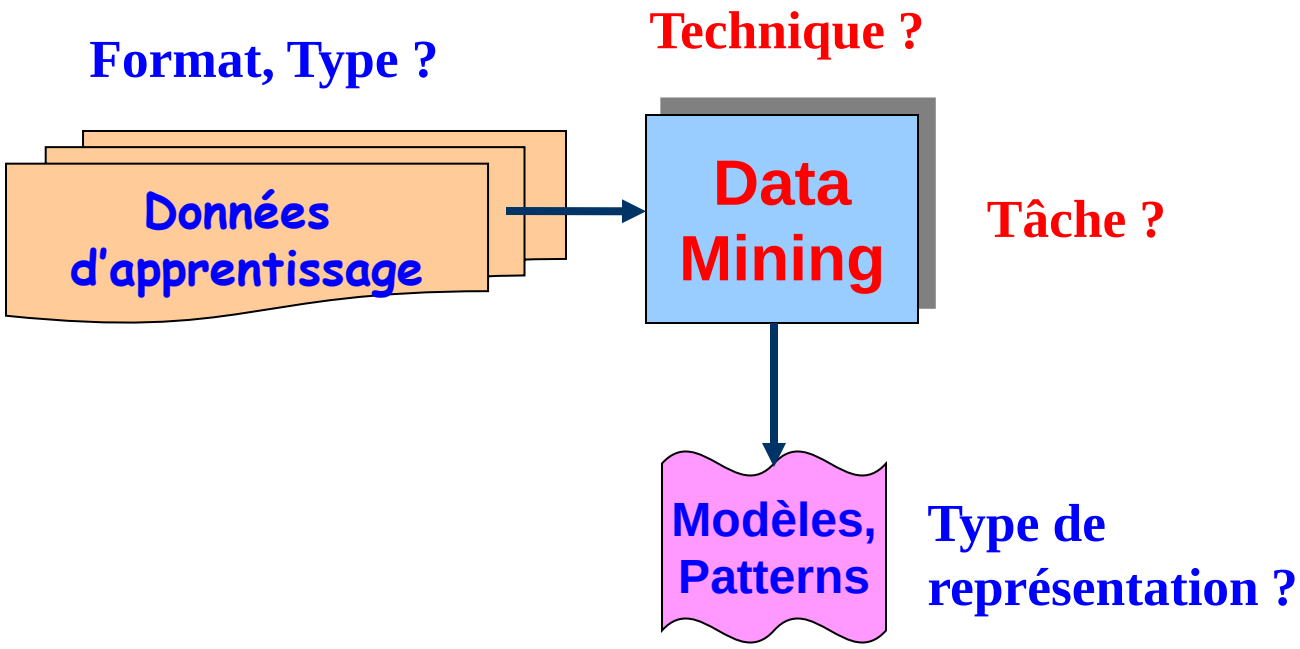
Les logs des accès Web sont analysés pour …

- Découvrir les préférences des utilisateurs
- Améliorer l'organisation du site Web

De manière similaire …

- L'analyse de tous les types d'informations sur les logs
- Adaptation de l'interface utilisateur/service

Paramètres d'un processus KDD



Les données : matière première

Valeurs des champs (p attributs ou variables) des enregistrements (n lignes ou cas) des tables de l'entrepôt (base de données, matrice n*p)

Types :

- Données discrètes : données binaires (sexe, …), données énumératives (couleur, …), énumératives ordonnées (réponses 1:très satisfait, 2:satisfait, …).
- Données continues : données entières ou réelles (âge, salaire, …)
- Dates
- Données textuelles
- Pages/liens web, Multimédia, …

Tâches du Data Mining

- Classification
- Clustering (Segmentation)
- Recherche d'associations
- Recherche de séquences
- Détection de déviation

Classification

Elle permet de prédire si une instance de donnée est membre d'un groupe ou d'une classe prédéfinie.

Chaque instance = p variables prédictives + 1 variable cible (à prédire)

Classification : variable cible discrète

Régression : variable cible continue

Classes

Groupes d'instances avec des profils particuliers

Apprentissage supervisé : classes connues à l'avance

Classification : Applications

- Banques-Assurances : évaluer les demandes de crédit
- Marketing : cibler les clients qui répondront à un mailing
- Finances : Détecter les tendances boursières
- Médecine : Diagnostic d'une maladie
- Grande distribution : classement des clients.

Etc …

Régression :

- prédire le salaire qu'une personne peut espérer,
- prédire la durée d'hospitalisation d'un patient,
- estimer la valeur d'un bien immobilier,
- estimer le retour sur investissement d'une campagne publicitaire

Clustering (Segmentation)

- Partitionnement logique de la base de données en clusters
  - Clusters : groupes d'instances ayant les mêmes caractéristiques
  - Apprentissage non supervisé (classes inconnues)
  - Pb : interprétation des clusters identifiés
  - Applications : Economie (segmentation de marchés), médecine (localisation de tumeurs dans le cerveau), etc.

Règles d'association

• Corrélations (ou relations) entre attributs (méthode non supervisée)

• Applications : grande distribution, gestion des stocks, web (pages visitées), etc.

Exemple

- BD commerciale : panier de la ménagère
- Articles figurant dans le même ticket de caisse
- Ex : achat de riz + vin blanc ==> achat de poisson
- Achats bières et couches-culottes (USA, Week-end)

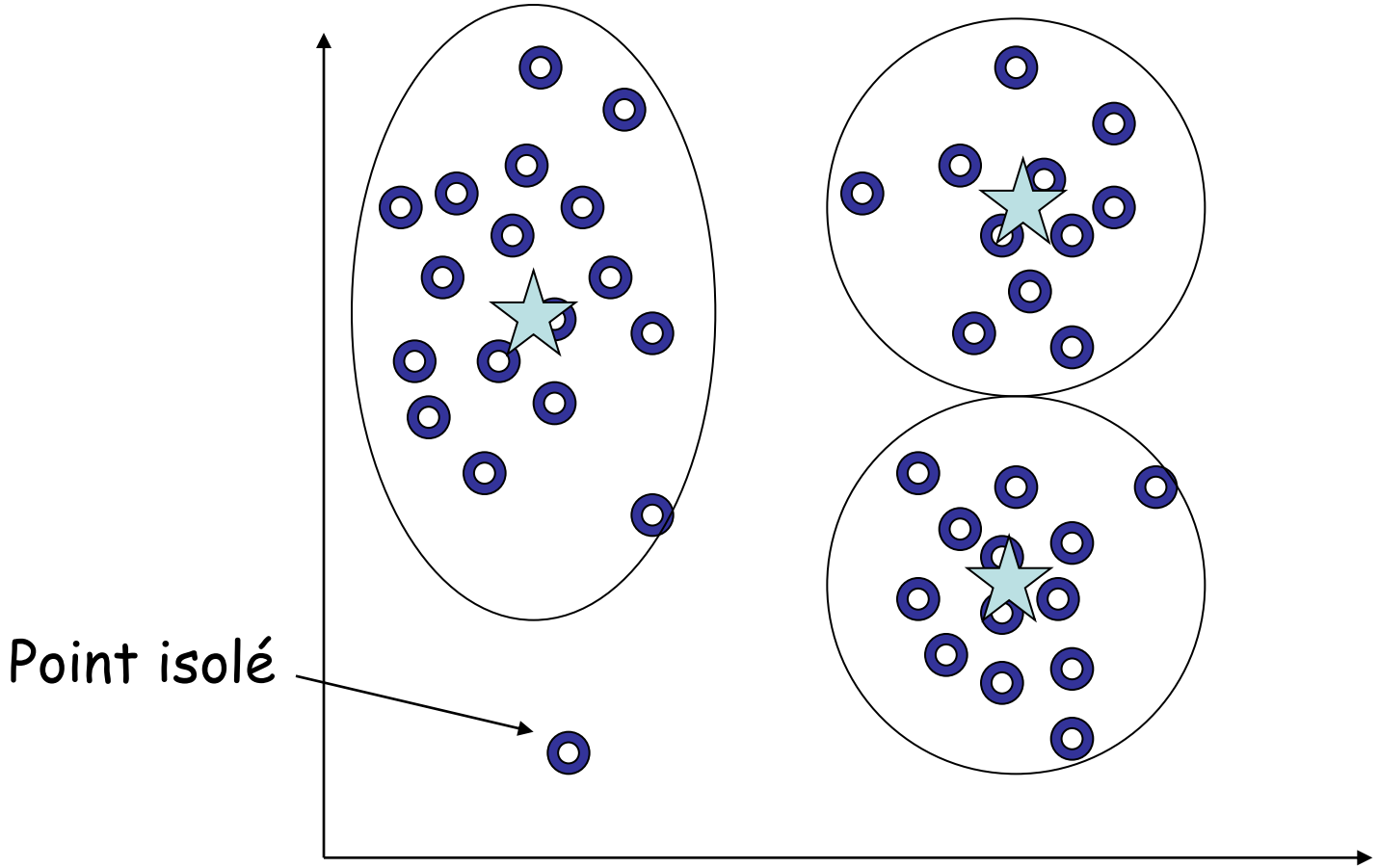
Recherche de séquences

- Recherche de séquences
  - Liaisons entre événements sur une période de temps
  - Extension des règles d'association
    - Prise en compte du temps (série temporelle)
    - Achat Télévision ==> Achat Magnétoscope d'ici 5 ans
  - Applications : marketing direct (anticipation des commandes), bioinformatique (séquences d'ADN), bourse (prédiction des valeurs des actions)
- Exemple
  - BD commerciale (ventes par correspondance)
  - Commandes de clients
  - Ex : 60% des consommateurs qui commandent la bière «Mort subite» commandent de l'aspro juste après
  - Séquences d'AND : ACGTC est suivie par GTCA après un gap de 9, avec une probabilité de 30%

Détection de déviation

- Instances ayant des caractéristiques les plus différentes des autres
  - Basée sur la notion de distance entre instances
  - Expression du problème
    - Temporelle : évolution des instances ?
    - Spatiale : caractéristique d'un cluster d'instances ?
- Applications
  - Détection de fraudes (transactions avec une carte bancaire inhabituelle en telemarketing)
  - Détection d'intrusions illicites (sécurité réseau, …)
- Caractéristiques
  - Problème d'interprétation : bruit ou exception (donc connaissance intéressante)

Illustration



Techniques utilisées

- K-moyennes, A-priori, K-NN
- Réseaux de neurones
- Algorithmes génétiques
- Chaînes de Markov cachées
- Arbres de décision
- Réseaux bayesiens
- Soft computing : ensembles flous
- …

Algorithmes de DM : Composantes

• Structure de modèle/motif : structure sous-jacente ou forme fonctionnelle des connaissances que l'on cherche à extraire (arbre de décision, réseaux de neurones, ...)

•Fonction d'évaluation : mesurer la qualité d'un modèle ou d'une hypothèse

•Méthode de recherche/d'optimisation : stratégie utilisée pour parcourir l'espace de recherche et trouver le meilleur modèle

•Stratégie de gestion des données : la façon de stocker, d'indexer et d'accéder aux données pendant le processus de recherche

Résumé - Introduction

Data mining : découverte automatique de modèles intéressants à partir d'ensemble de données de grande taille

KDD (knowledge data discovery) est un processus :

- Pré-traitement (Pre-processing)
- Data mining
- Post-traitement (Post-processing)

Pour le data mining, utilisation de différents …

- Base de données (relationelle, orientée objet, spatiale, WWW, …)
- Connaissances (classification, clustering, association, …)
- Techniques (apprentissage, statistiques, optimisation, …)
- Applications (génomique, télécom, banque, assurance, distribution, …)

Clustering (Segmentation)

Clustering - Plan

Problèmatique du clustering

Applications

Similarité et types de données

Méthodes de clustering

- Méthodes de partitionnement
- Méthodes hiérarchiques
- Méthodes par voisinage dense

Application réelle en génomique

Résumé

Problèmatique

Soient N instances de données à k attributs,

Trouver un partitionnement en c clusters (groupes) ayant un sens (Similitude)

Affectation automatique de “labels” aux clusters

c peut être donné, ou “découvert”

Plus difficile que la classification car les classes ne sont pas connues à l’avance (non supervisé)

Attributs

- Numériques (distance bien définie)
- Enumératifs ou mixtes (distance difficile à définir)

Qualité d'un clustering

Une bonne méthode de clustering produira des clusters d'excellente qualité avec :

- Similarité intra-classe importante
- Similarité inter-classe faible

La qualité d'un clustering dépend de :

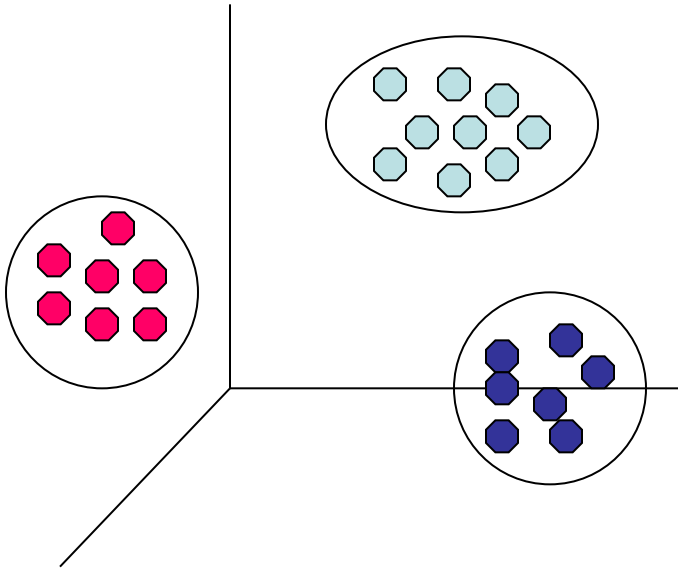
- La mesure de similarité utilisée
- L'implémentation de la mesure de similarité

La qualité d'une méthode de clustering est évaluée par son habilité à découvrir certains ou tous les "patterns" cachés.

Objectifs du clustering

Minimiser les distances intra-cluster

Maximiser les distances inter-clusters



Exemples d'applications

Marketing : segmentation du marché en découvrant des groupes de clients distincts à partir de bases de données d'achats.

Environnement : identification des zones terrestres similaires (en termes d'utilisation) dans une base de données d'observation de la terre.

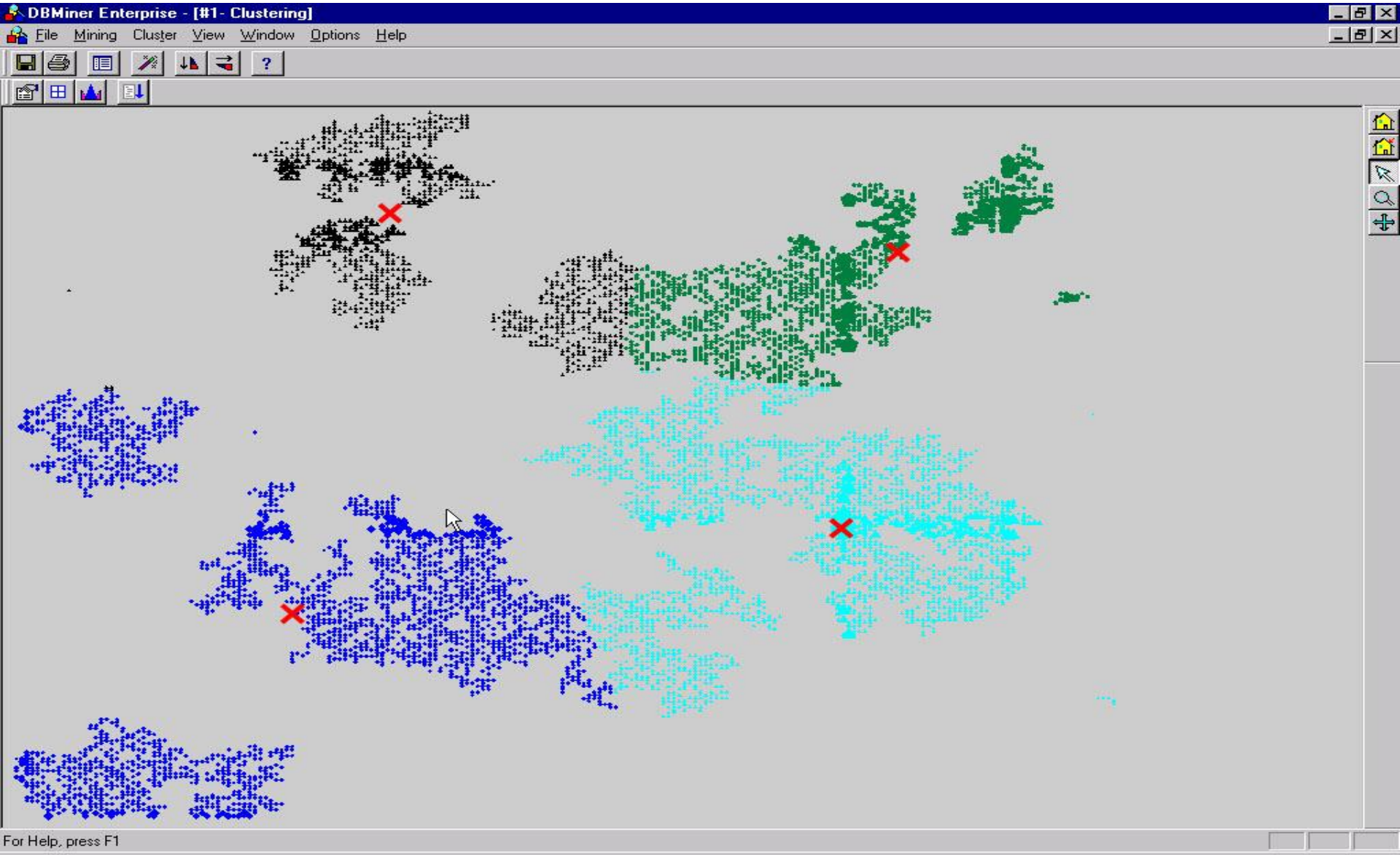
Assurance: identification de groupes d'assurés distincts associés à un nombre important de déclarations.

Planification de villes : identification de groupes d'habitations suivant le type d'habitation, valeur, localisation géographique, …

Médecine : Localisation de tumeurs dans le cerveau

- Nuage de points du cerveau fournis par le neurologue
- Identification des points définissant une tumeur

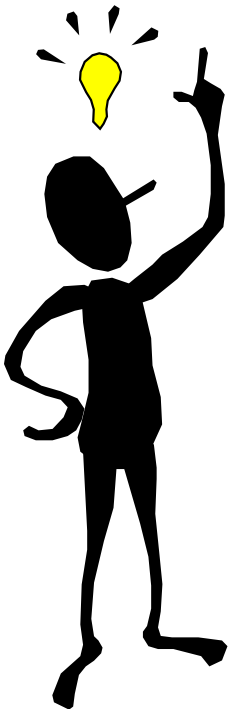
Exemple: segmentation de marchés



Mesure de la similarité

Il n'y a pas de **définition unique** de la similarité entre objets

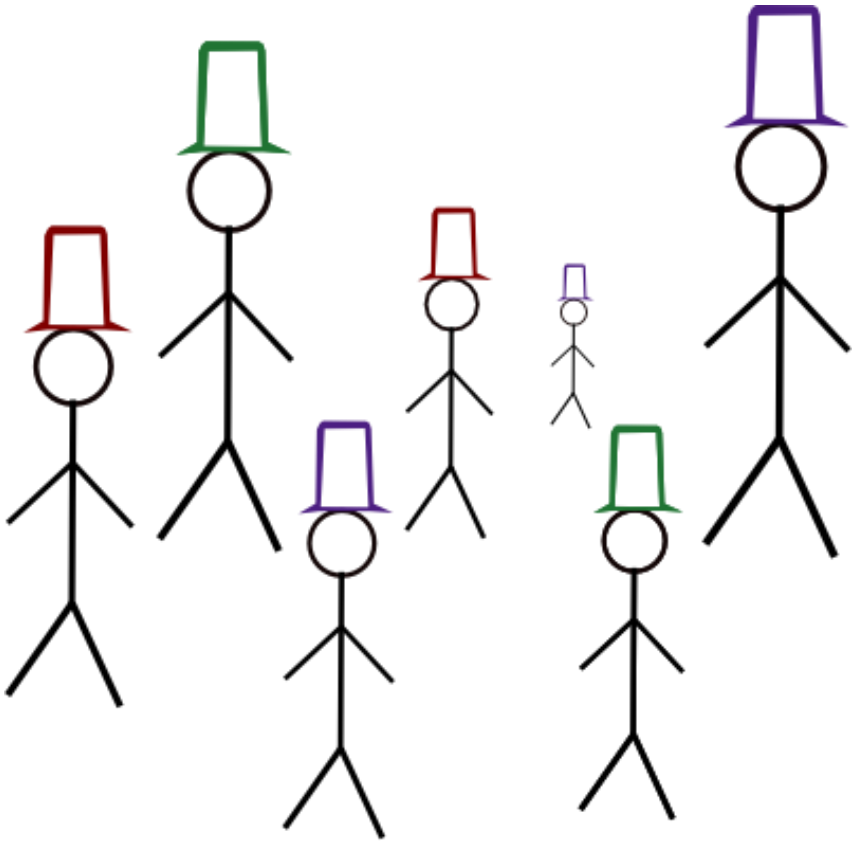
- Différentes mesures de distances d(x,y)



La définition de la similarité entre objets dépend de :

- Le type des données considérées
- Le type de similarité recherchée

Mesure de la similarité



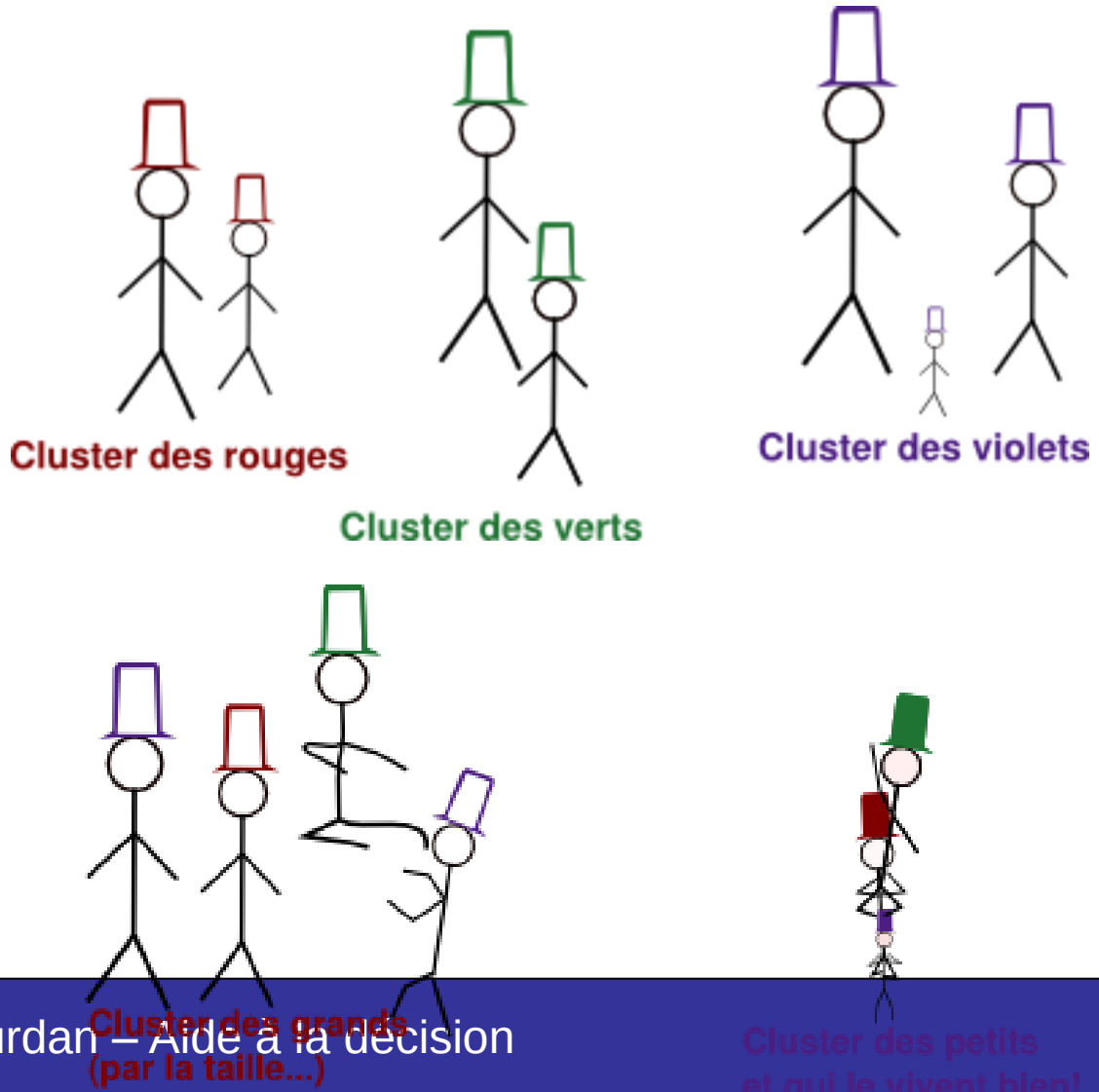
Il n'y a pas de définition unique de la similarité entre objets

- Différentes mesures de distances d(x,y)

La définition de la similarité entre objets dépend de :

- Le type des données considérées
- Le type de similarité recherchée

Mesure de similarité



Choix de la distance

Propriétés d'une distance :

1. $d(x, y) \geq 0$
2. $d(x, y) = 0 \text{ iff } x = y$
3. $d(x, y) = d(y, x)$
4. $d(x, z) \leq d(x, y) + d(y, z)$

Définir une distance sur chacun des champs

Champs numériques : d(x,y) = |x-y|, d(x,y)= |x-y|/dmax (distance normalisée).

Exemple : Age, taille, poids, …

Distance – Données numériques

Combiner les distances : Soient $x=(x_1,\ldots,x_n)$ et $y=(y_1, \ldots,y_n)$

Exemples numériques :

Distance euclidienne : $$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

Distance de Manhattan : $$d(x,y)=\sum_{i=1}^{n}|x_i-y_i|$$

Distance de Minkowski : $$d(x,y)=\sqrt[q]{\sum_{i=1}^{n}|x_i-y_i|^q}$$

q=1 : distance de Manhattan.

q=2 : distance euclidienne

Choix de la distance

Champs discrets :

- Données binaires : d(0,0)=d(1,1)=0, d(0,1)=d(1,0)=1
- Donnée énumératives : distance nulle si les valeurs sont égales et 1 sinon.
- Donnée énumératives ordonnées : idem. On peut définir une distance utilisant la relation d'ordre.

Données de types complexes : textes, images, données génétiques, ...

Distance – Données binaires

Table de contingence (dissimilarité)

		Object j		
		1	0	*sum*
Object i	1	a	b	$a+b$
	0	c	d	$c+d$
	sum	$a+c$	$b+d$	p

- **Coefficient de correspondance simple** (similarité invariante, si la variable binaire est **symétrique**) : $$d(i,j) = \frac{b+c}{a+b+c+d}$$
- **Coefficient de Jaccard** (similarité non invariante, si la variable binaire est **asymétrique**): $$d(i,j) = \frac{b+c}{a+b+c}$$

Distance – Données binaires

Exemple : dissimilarité entre variables binaires

- Table de patients

Nom	Sexe	Fièvre	Toux	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- 8 attributs, avec
  - Sexe un attribut symétrique, et
  - Les attributs restants sont asymétriques
  - (test VIH, …)

Distance – Données binaires

Les valeurs Y et P sont initialisées à 1, et la valeur N à 0.

Calculer la distance entre patients, basée sur le coefficient de Jaccard.

Jack	M	1	0	1	0	0	0
Mary	F	1	0	1	0	1	0
Jim	M	1	1	0	0	0	0

Jack/Mary	1	0
1	A=2	B=0
0	C=1	D=3

Jack/Jim	1	0
1	A=1	B=1
0	C=1	D=3

Jim, Mary	1	0
1	A=1	B=1
0	C=2	D=2

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

Distance – Données énumératives

Généralisation des variables binaires, avec plus de 2 états, e.g., rouge, jaune, bleu, vert

Méthode 1: correpondance simple

- m: # de correspondances, p: # total de variables

$$d(i,j) = \frac{p-m}{p}$$

Distance – Données mixtes

Exemple : (Age, Propriétaire résidence principale, montant des mensualités en cours)

x=(30,1,1000), y=(40,0,2200), z=(45,1,4000)

$d(x,y)=sqrt((10/15)^2 + 1^2 + (1200/3000)^2) = 1.27$

$d(x,z)= sqrt((15/15)^2 + 0^2 + (3000/3000)^2) = 1.41$

$d(y,z)= sqrt((5/15)^2 + 1^2 + (1800/3000)^2) = 1.21$

plus proche voisin de x = y

Distances normalisées.

Sommation : $d(x,y)=d1(x1,y1) + \ldots + dn(xn,yn)$

Données mixtes – Exemple 1

Base de données « Cancer du sein »

http://www1.ics.uci.edu/~mlearn/MLSummary.html

#instances = 286 (Institut Oncologie, Yugoslavie)

attributs = 10

- Classe : no-recurence-events, recurrence-events
- Age : 10-19, 20-29, 30-39, 40-49, …, 90-99
- Menopause : Lt40, Ge40, premeno
- Taille de la tumeur : 0-4, 5-9, 10-14, …, 55-59
- Inv-nodes : 0-2, 3-5, 6-8, …, 36-39 (ganglions lymphatiques)
- Node-caps : Oui, Non
- Deg-malig : 1, 2, 3 (Dégré de malignité)
- Sein : Gauche, Droit
- Breast-quad : left-up, left-low, right-up, right-low, central
- Irradiation : Oui, Non

Données mixtes – Exemple 2

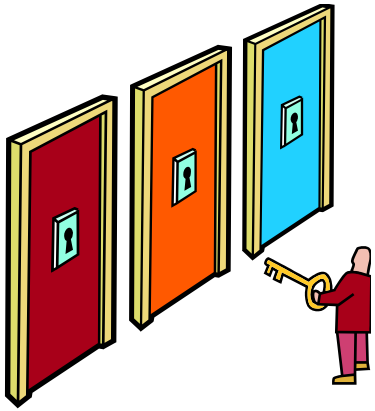
Base de données « Diabète » : Diagnostic (OMS)
http://www1.ics.uci.edu/~mlearn/MLSummary.html

#instances = 768 (Arizona, USA)

attributs = 8

- Nombre de grossesses
- Concentration du taux de glucose dans le plasma
- Pression sanguine diastolique (mm Hg)
- Epaisseur de la graisse du triceps (mm)
- Taux d'insuline après 2 heures (repas) (mu U/ml)
- Indice de masse corporelle (poids en kg / (taille en m)^2)
- Fonction « Diabete pedigree »
- Age (ans)
- Classe (Positif ou Négatif)

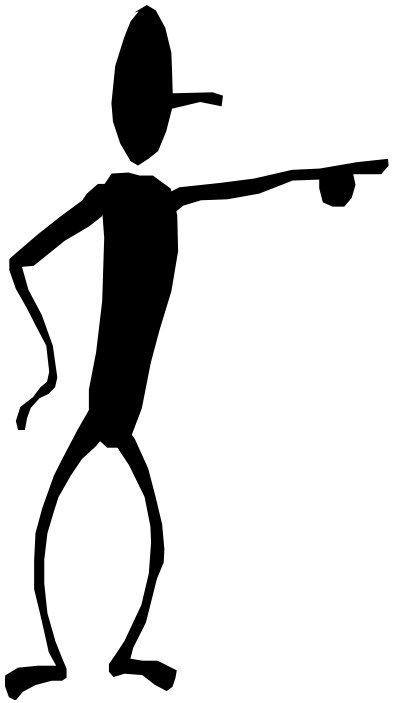
Méthodes de Clustering



- Méthode de partitionnement (K-moyennes)
- Méthodes hiérarchiques (par agglomération)
- Méthode par voisinage dense

- Caractéristiques
  - Apprentissage non supervisé (classes inconnues)
  - Pb : interprétation des clusters identifiés

Méthodes de clustering - Caractéristiques



Extensibilité

Habilité à traiter différents types de données

Découverte de clusters de différents formes

Connaissances requises (paramètres de l'algorithme)

Habilité à traiter les données bruitées et isolées.

Algorithme des k-moyennes (K-means)

Entrée : un échantillon de m enregistrements x_1, …, x_m

1. Choisir k centres initiaux c_1, …, c_k

2. Répartir chacun des m enregistrements dans le groupe i dont le centre ci est le plus proche.

3. Si aucun élément ne change de groupe alors arrêt et sortir les groupes

4. Calculer les nouveaux centres : pour tout i, c_i est la moyenne des éléments du groupe i.

Aller en 2.

Illustration (1)

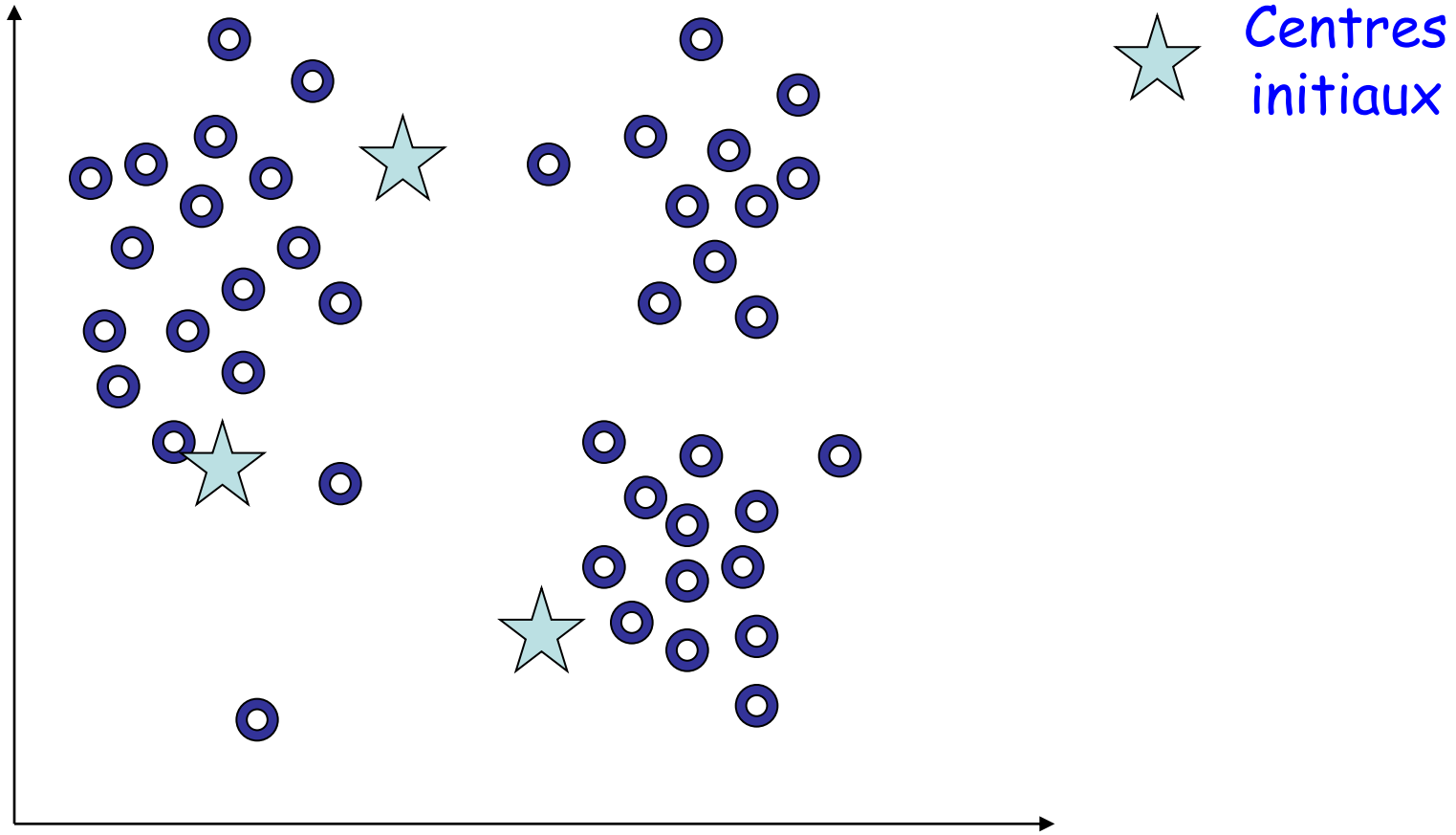


Illustration (1)

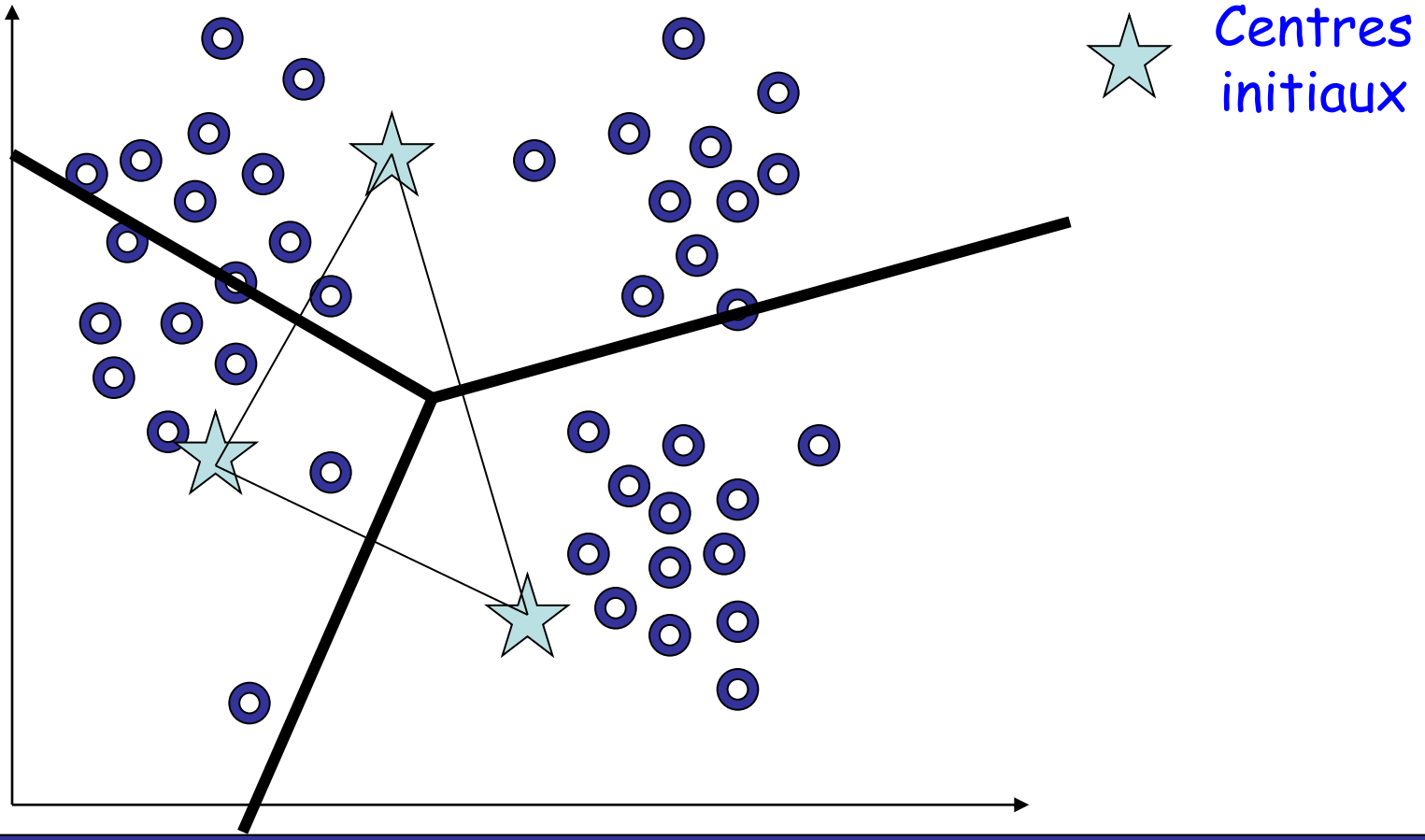


Illustration (1)

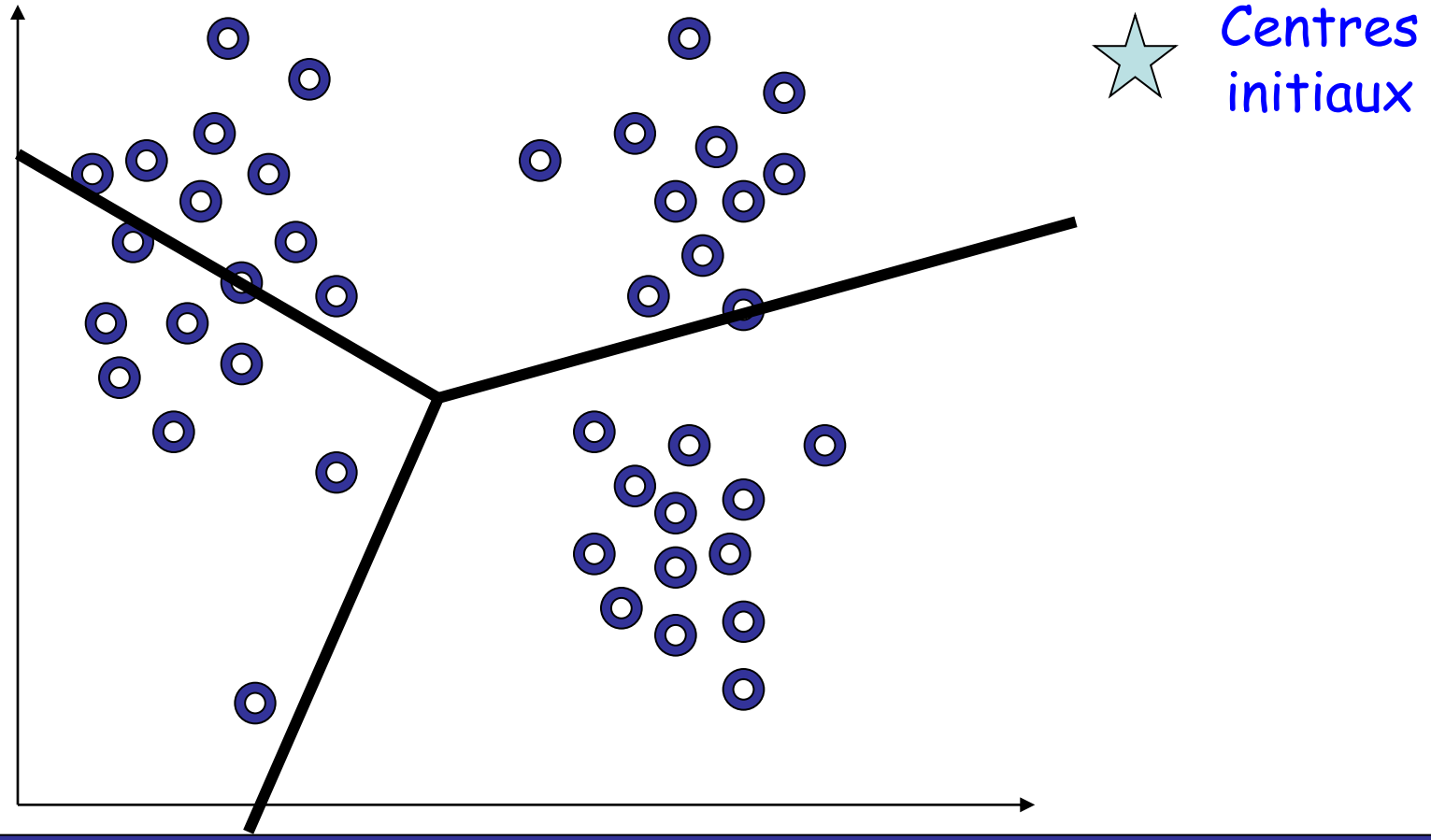


Illustration (1)

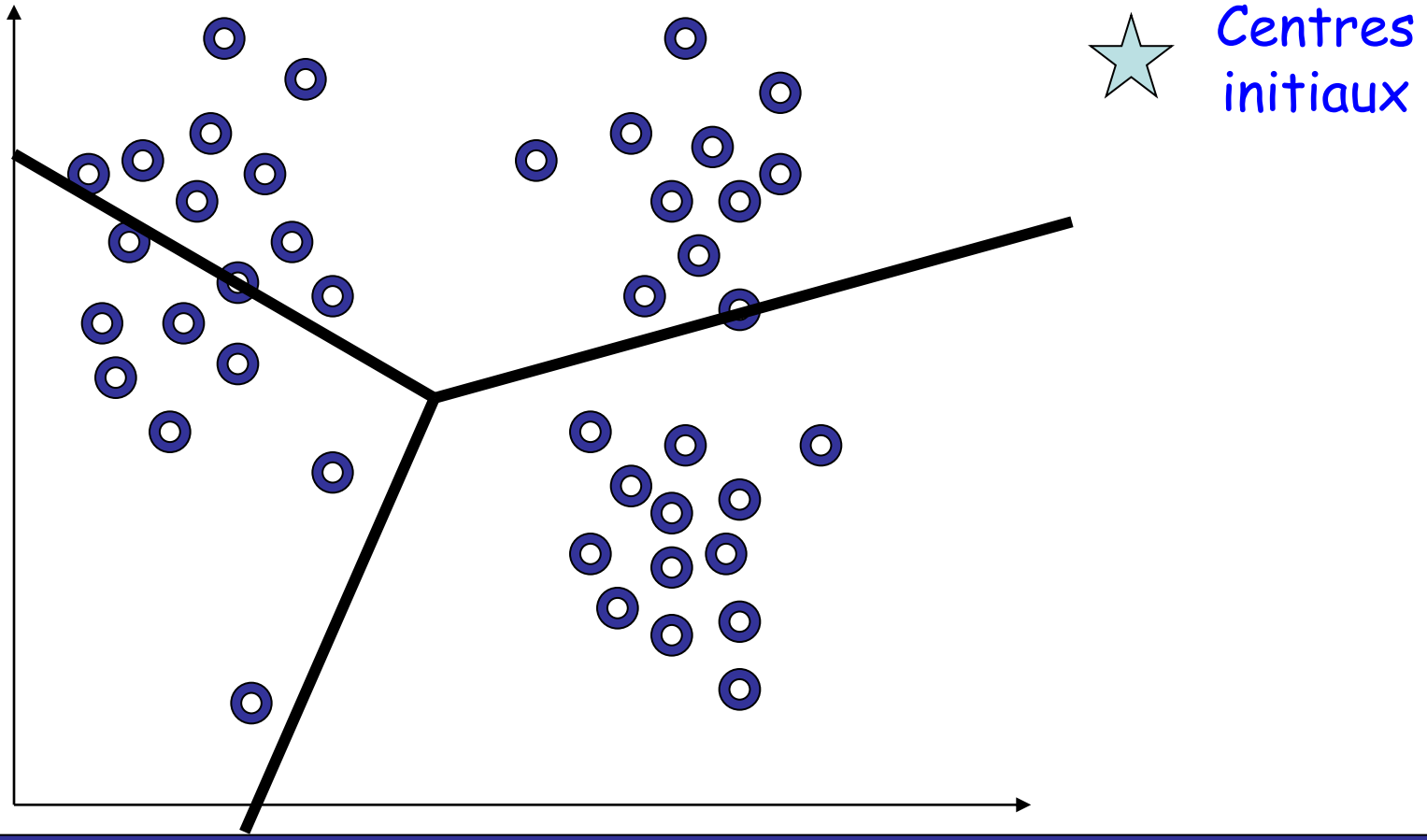
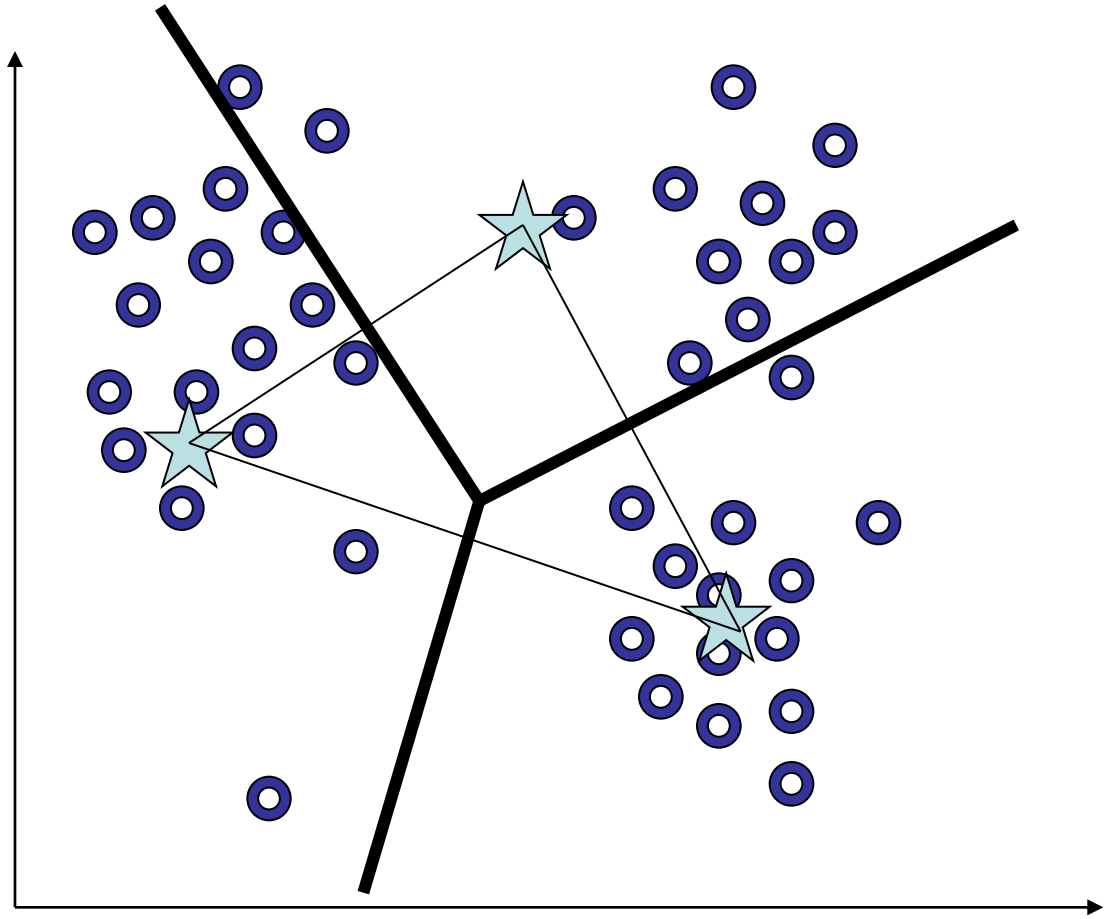
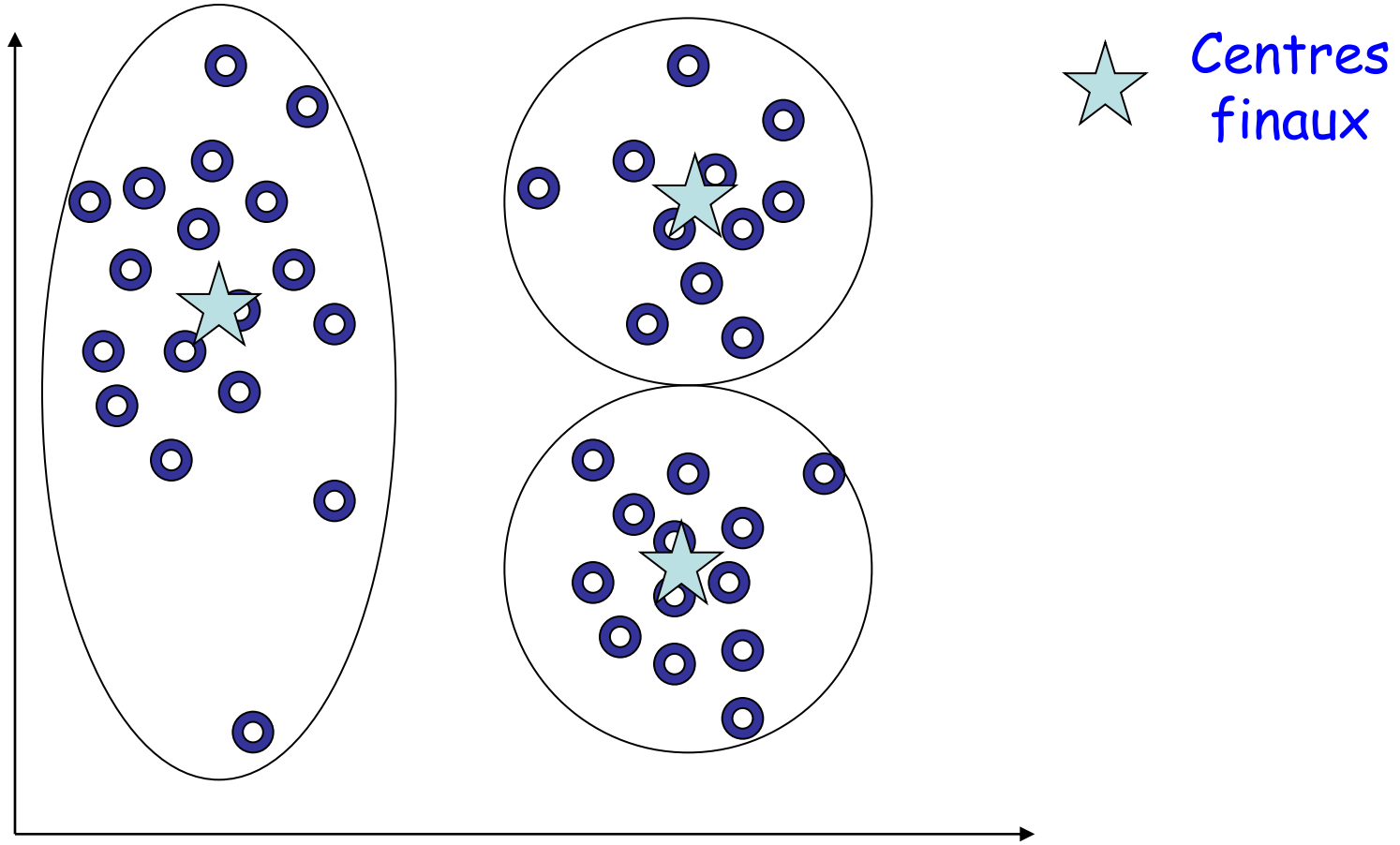


Illustration (2)



Nouveaux centres

Illustration (3)

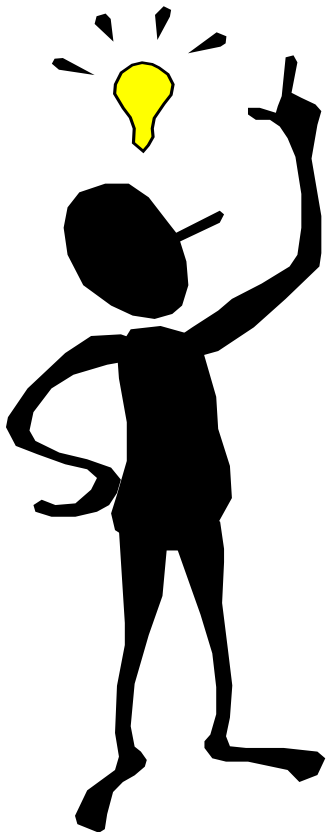


Algorithme des k-moyennes : Exemple

- 8 points A, …, H de l 'espace euclidéen 2D. k=2 (2 groupes)
- Tire aléatoirement 2 centres : B et D choisis.

points	Centre D(2,4), B(2,2)	Centre D(2,4), I(27/7,17/7)	Centre J(5/3,10/3), K(24/5,11/5)
A(1,3)	B	D	J
B(2,2)	B	I	J
C(2,3)	B	D	J
D(2,4)	D	D	J
E(4,2)	B	I	K
F(5,2)	B	I	K
G(6,2)	B	I	K
H(7,3)	B	I	K

K-moyennes : Avantages



Relativement extensible dans le traitement d'ensembles de taille importante

Relativement efficace : $O(t.k.n)$, où n représente # objets, k # clusters, et t # iterations. Normalement, $k, t << n$.

Produit généralement un **optimum local** ; un optimum global peut être obtenu en utilisant d'autres techniques telles que : algorithmes génétiques, …

K-moyennes : Inconvénients

Applicable seulement dans le cas où la moyenne des objets est définie

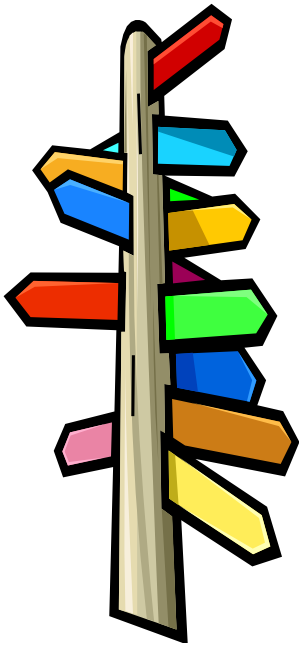
Besoin de spécifier *k, le nombre de clusters,* a priori

Incapable de traiter les données bruitées (noisy).

Non adapté pour découvrir des clusters avec structures non-convexes, et des clusters de tailles différentes

Les **points isolés** sont mal gérés (doivent-ils appartenir obligatoirement à un cluster ?) - probabiliste

K-moyennes : Variantes



Sélection des centres initiaux

Calcul des similarités

Calcul des centres (K-medoids : [Kaufman & Rousseeuw'87])

GMM : Variantes de *K-moyennes* basées sur les probabilités

K-modes : données catégorielles [Huang'98]

K-prototype : données mixtes (numériques et catégorielles)

Méthodes hiérarchiques

Une méthode hiérarchique : construit une hiérarchie de clusters, non seulement une partition unique des objets.

Le nombre de clusters **k** n'est pas exigé comme donnée

Utilise une **matrice de distances** comme critère de clustering

Une **condition de terminaison** peut être utilisée (ex. Nombre de clusters)

Méthodes hiérarchiques

Entrée : un échantillon de m enregistrements x_1, …, x_m

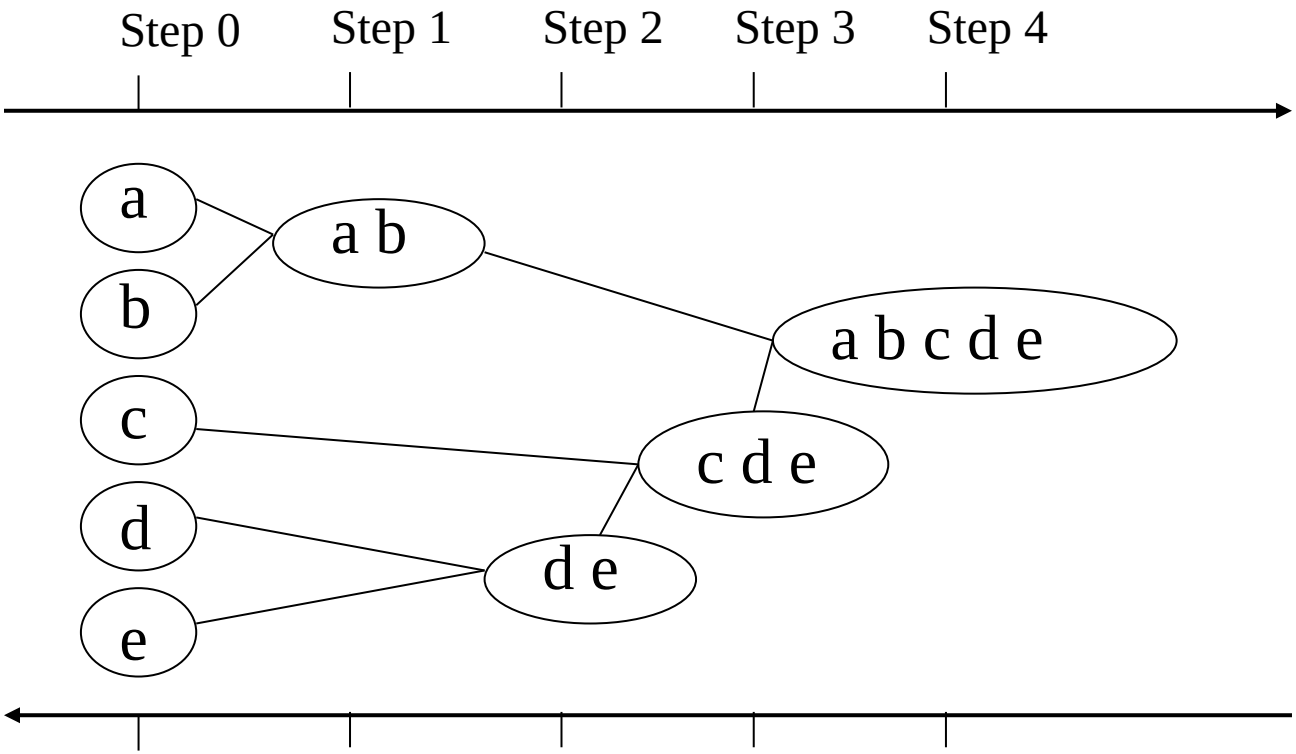
1. On commence avec m clusters (cluster = 1 enregistrement)

2. Grouper les deux clusters les plus « proches ».

3. S'arrêter lorsque tous les enregistrements sont membres d'un seul groupe

4. Aller en 2.

Arbre de clusters : Exemple



Arbre de clusters

Résultat : Graphe hiérarchique qui peut être coupé à un niveau de dissimilarité pour former une partition.

La hiérarchie de clusters est représentée comme un arbre de clusters, appelé dendrogramme

- Les feuilles de l'arbre représentent les objets
- Les nœuds intermédiaires de l'arbre représentent les clusters

Distance entre clusters

Distance entre les centres des clusters (Centroid Method)

Distance minimale entre toutes les paires de données des 2 clusters (Single Link Method)

$$d(i, j) = \min_{x \in C_i, y \in C_j} \{ d(x, y) \}$$

Distance maximale entre toutes les paires de données des 2 clusters (Complete Link Method)

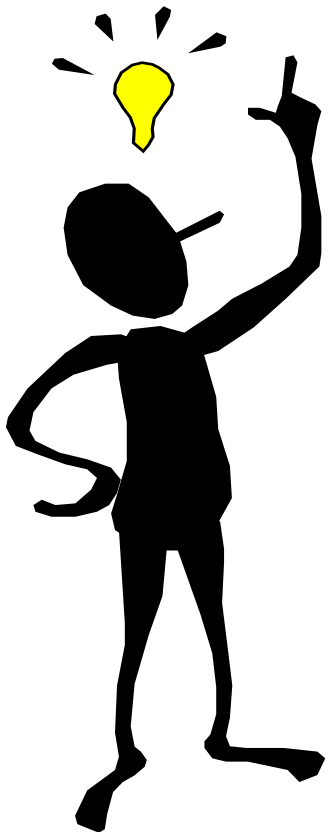
$$d(i, j) = \max_{x \in C_i, y \in C_j} \{ d(x, y) \}$$

Distance moyenne entre toutes la paires d'enregistrements (Average Linkage)

$$d(i, j) = avg_{x \in C_i, y \in C_j} \{ d(x, y) \}$$

Méthodes hiérarchiques : Avantages

Conceptuellement simple

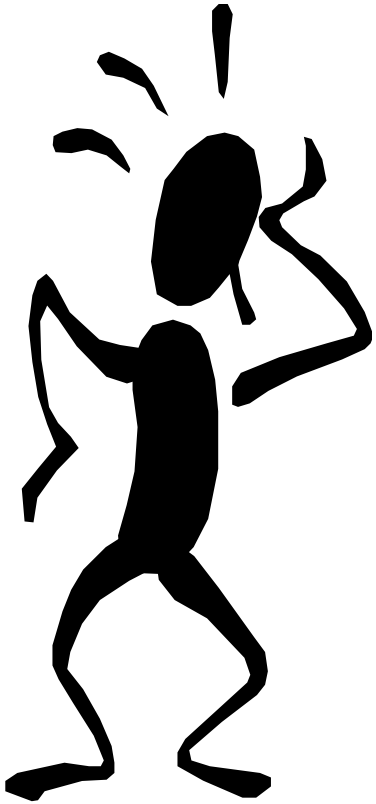


Propriétés théoriques sont bien connues

Quand les clusters sont groupés, la décision est définitive => le nombre d'alternatives différentes à à examiner est réduit

Méthodes hiérarchiques : Inconvénients

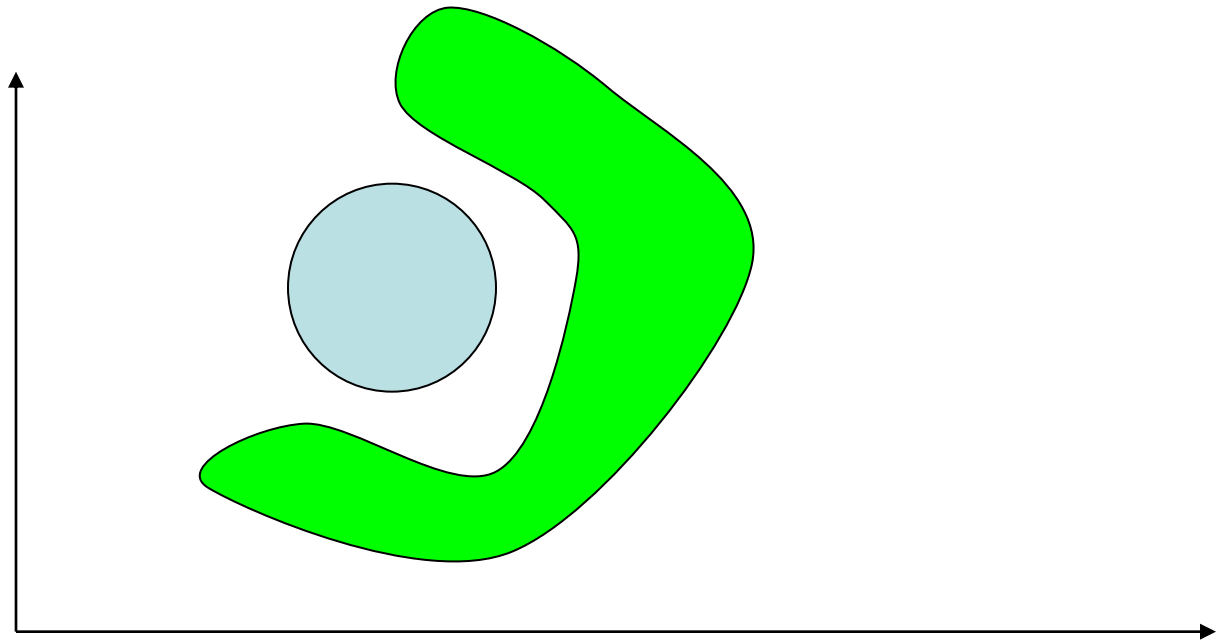
Groupement de clusters est définitif => décisions erronées sont impossibles à modifier ultérieurement (méthode gloutonne)



Méthodes non extensibles pour des ensembles de données de grandes tailles

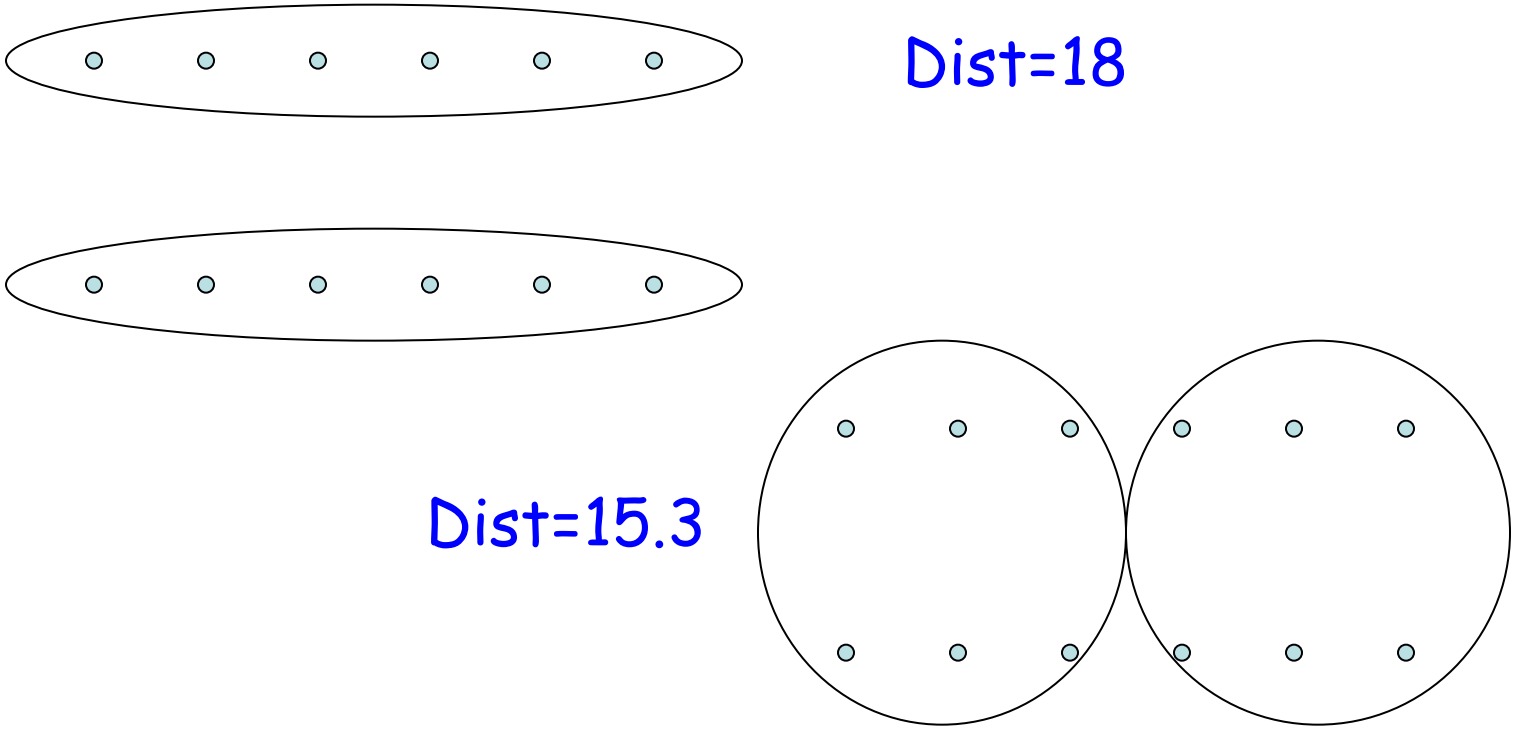
Méthodes basées sur la densité

Pour ce types de problèmes, l'utilisation de mesures de similarité (distance) est moins efficace que l'utilisation de densité de voisinage.



Méthodes basées sur la densité

Minimiser la distance inter-clusters n'est pas toujours un bon critère pour reconnaître des «formes » (applications géographiques, reconnaissance de formes – tumeurs, …).



Méthodes basées sur la densité (1)

Soit d* un nombre réel positif

Si d(P,Q)<=d*, Alors P et Q appartiennent au même cluster

Si P et Q appartiennent au même cluster, et d(Q,R)<=d*, Alors P et R appartiennent au même cluster

Méthodes basées sur la densité (2)

Soit e* un nombre réel positif

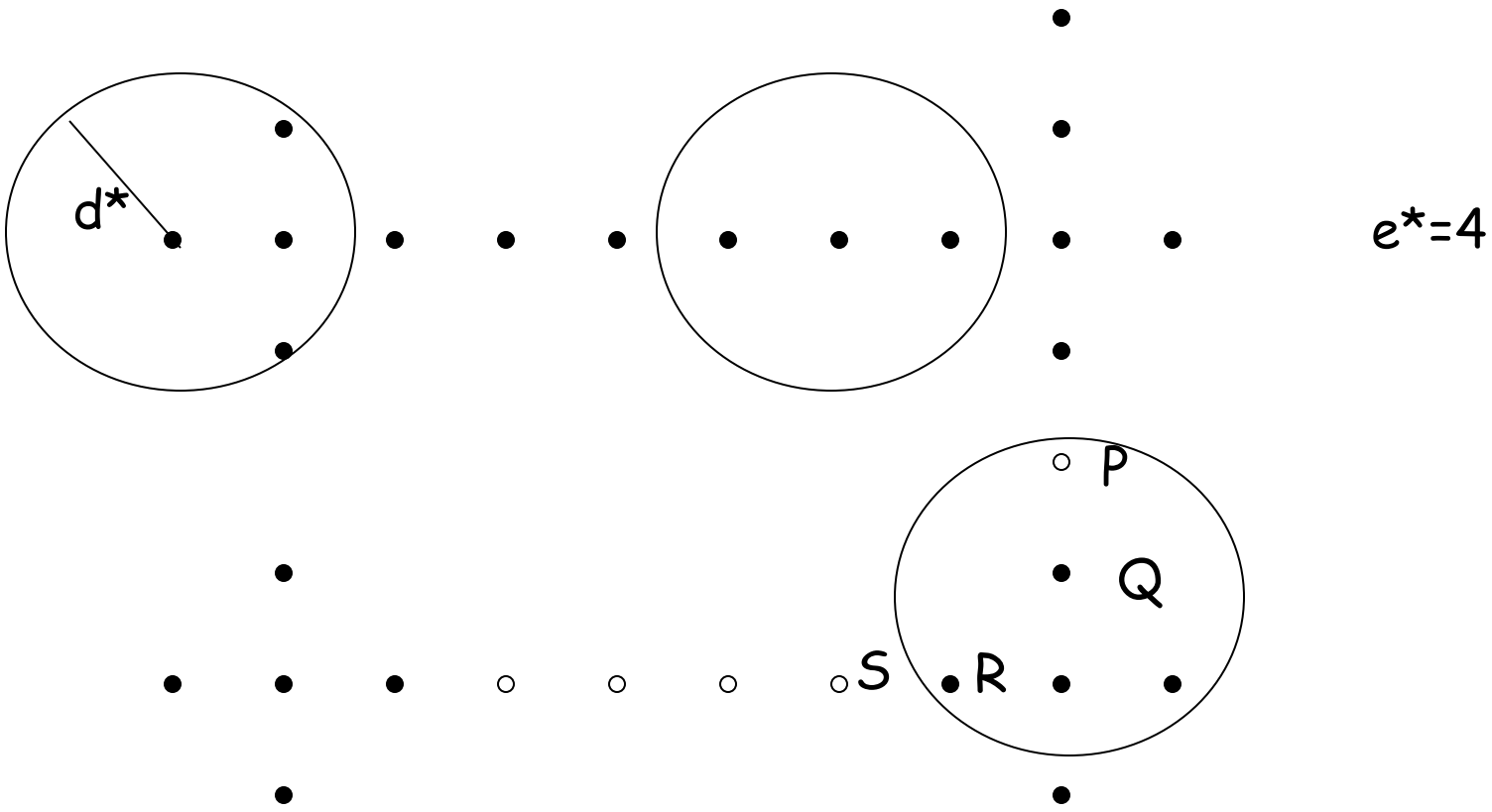
Un point P est dense ssi |{Q/d(P,Q)<=d*}|>=e*

Si P et Q appartiennent au même cluster, et d(Q,R)<=d* et Q est dense, Alors P et R appartiennent au même cluster

Les points non-denses sont appelés points de « bordure ».

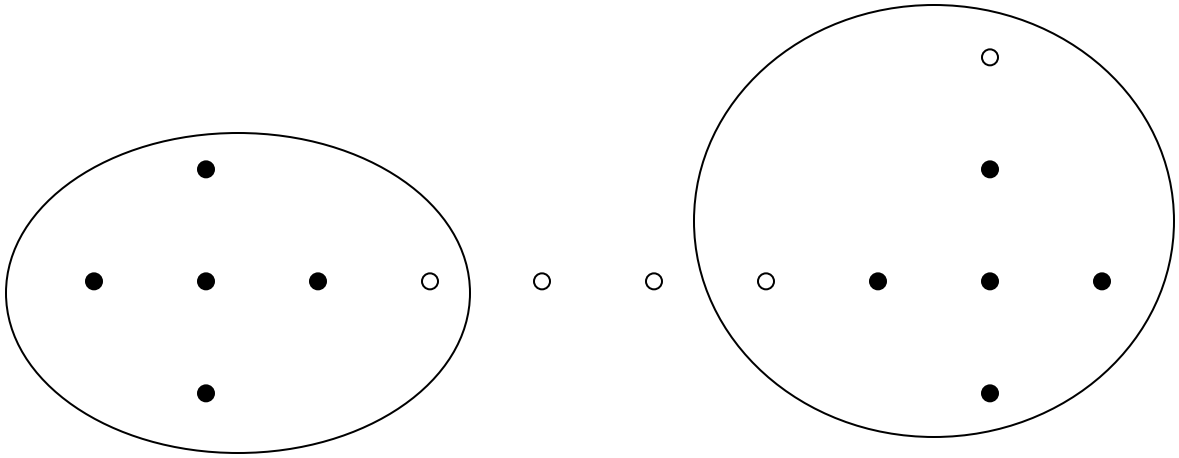
Les points en dehors des clusters sont appelés « bruits ».

Méthodes basées sur la densité



- Points noirs sont denses ; les autres ne sont pas denses
- Pour montrer que P et S appartiennent au même cluster, il suffit de montrer que P et R appartiennent au même cluster. Pour le montrer pour P et R, il suffit de le montrer pour P et Q …

Méthodes basées sur la densité



- Deux clusters sont trouvés
- Deux points sont des « bruits »
- Trois points sont des « bordures »

Clustering : Validation

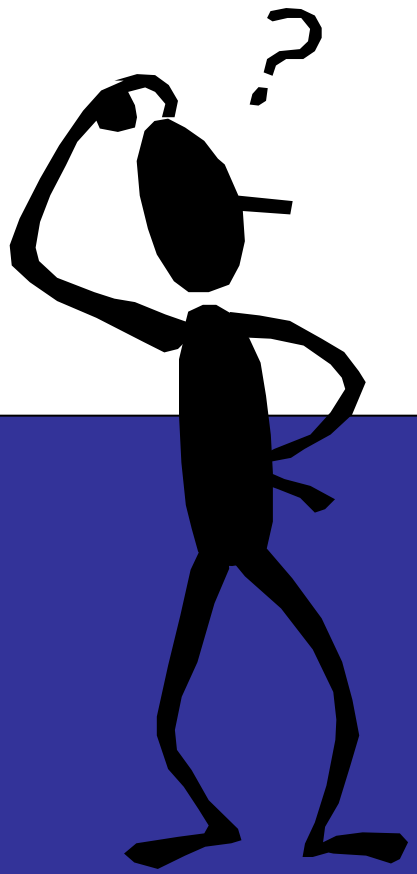
Solution optimale connue (table de contingence) :

$$\frac{n_{11}+n_{00}}{n_{11}+n_{10}+n_{01}}$$

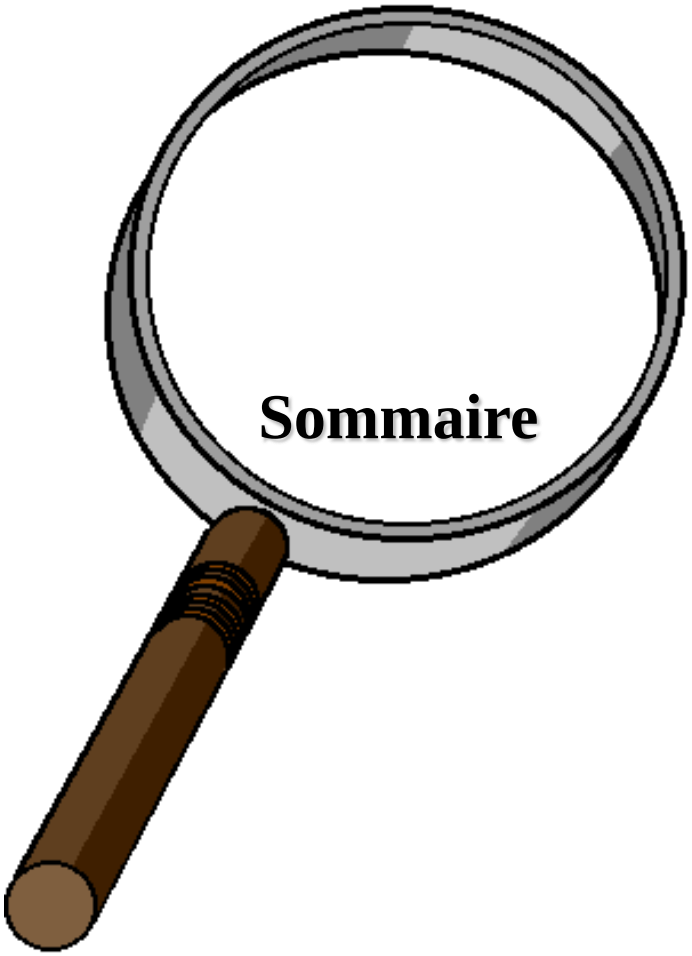
Solution optimale inconnue :

- Homogénéité (intra-cluster)
- Séparation (inter-cluster)

Règles d’association



Sommaire



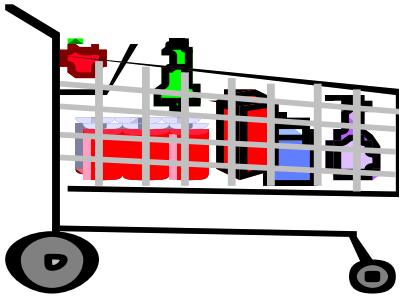
Exemple : Analyse du panier de la ménagère

- Découverte d'**associations** et de **corrélations** entre les articles achetés par les clients en analysant les achats effectués (panier)

Lait, Oeufs, Céréale, Lait

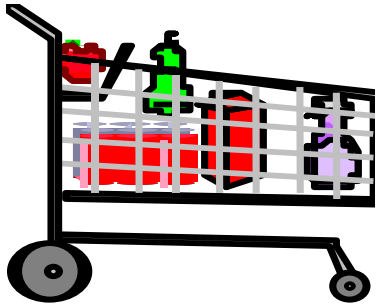
Client 2

Lait, Oeufs, Sucre, Pain



Client 1

Oeufs, Sucre



Client 3

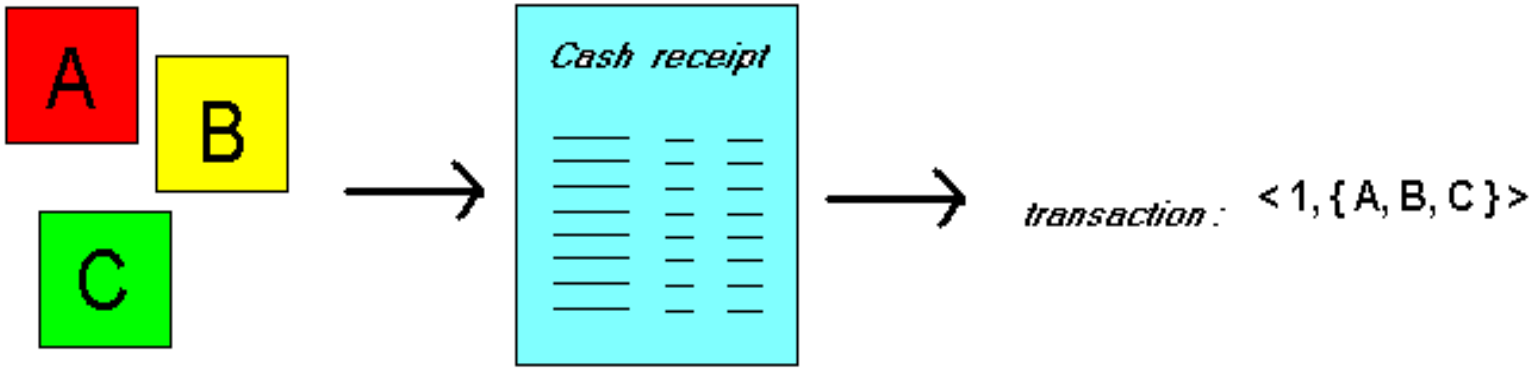
Exemple : Analyse du panier de la ménagère

Etant donnée :

- Une base de données de transactions de clients, où chaque transaction est représentée par un ensemble d'articles -**set of items-** (ex., produits)

Trouver :

- Groupes d'articles (itemset) achetés fréquemment (ensemble)



Exemple : Analyse du panier de la ménagère

Extraction d'informations sur le comportement de clients

- SI achat de riz + vin blanc ALORS achat de poisson (avec une grande probabilité)

Intérêt de l'information : peut suggérer ...

- Disposition des produits dans le magasin
- Quels produits mettre en promotion, gestion de stock, …

Approche applicable dans d'autres domaines

- Cartes de crédit, e-commerce, …
- Services des compagnies de télécommunication
- Services bancaires
- Traitements médicaux, …

Règles d'associations

Recherche de règles d'association :

- Découvrir des patterns, corrélations, associations fréquentes, à partir d'ensembles d'items contenus dans des base de données.

Compréhensibles : Facile à comprendre

Utiles : Aide à la décision

Efficaces : Algorithmes de recherche

Applications :

- Analyse des achats de clients, Marketing, Accés Web, Design de catalogue, Génomique, etc.

Règles d'associations

Formats de représentation des règles d'association :

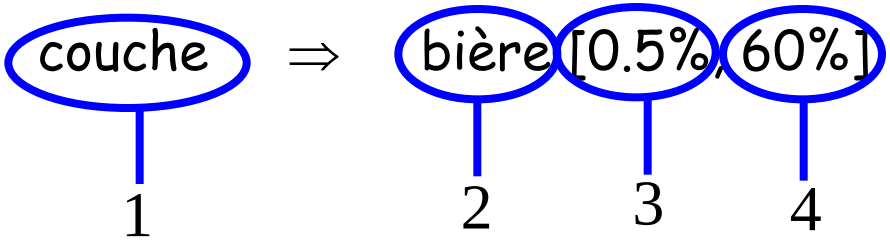
- couches $\Rightarrow$ bière [0.5%, 60%]
- achète:couches $\Rightarrow$ achète:bière [0.5%, 60%]

- "SI achète couches ALORS achète bière dans 60% de cas. Les couches et la bière sont tous deux achetés dans 0.5% des transactions de la base de données."

Autres représentations (utilisée dans l'ouvrage de Han) :

- achète(x, "couches") $\Rightarrow$ achète(x, "bière") [0.5%, 60%]

Règles d'associations



"**SI** achète couche, **ALORS** achète bière, dans 60% de cas, dans 0.5% de la base"

Condition, partie gauche de la règle

Conséquence, partie droite de la règle

Support, fréquence ("partie gauche **et** droite sont présentes ensemble dans la base")

Confiance ("si partie gauche de la règle est vérifiée, probabilité que la partie droite de la règle soit vérifiée")

Règles d'associations

- **Support :** % d'instances de la base vérifiant la règle.

support(A ⇒ B [s, c]) = p(A et B) = support ({A,B})

- **Confiance :** % d'instances de la base vérifiant l'implication

confiance(A ⇒ B [s, c]) = p(B|A) = p(A et B) / p(A) = support({A,B}) / support({A})

Exemple

TID	Items
1	Pain, Lait
2	Bière, Couches, Pain, Oeufs
3	Bière, Coca, Couches, Lait
4	Bière, Pain, Couches, Lait
5	Coca, Pain, Couches, Lait

Règle : $X \Rightarrow_{s,\alpha} y$

Support : $s = \frac{\sigma(X \cup y)}{|T|} (s = P(X, y))$

Confiance : $\alpha = \frac{\sigma(X \cup y)}{\sigma(X)|} (\alpha = P(y \mid X))$

$$\{Couches, Lait\} \Rightarrow_{s,\alpha} Bière$$

$$s = \frac{\sigma(Couches, Lait, Bière)}{\text{Nombre total d'instances}} = \frac{2}{5} = 0.4$$

$$\alpha = \frac{\sigma(Couches, Lait, Bière)}{\sigma(Couches, Lait)|} = 0.66$$

Règles d'associations

Support minimum σ :

- Elevé $\Rightarrow$ peu d'itemsets fréquents
- $\Rightarrow$ peu de règles valides qui ont été souvent vérifiées

- Réduit $\Rightarrow$ plusieurs règles valides qui ont été rarement vérifiées

Confiance minimum γ :

- Elevée $\Rightarrow$ peu de règles, mais toutes "pratiquement" correctes
- Réduite $\Rightarrow$ plusieurs règles, plusieurs d'entre elles sont "incertaines"

Valeurs utilisées : σ = 2 - 10 %, γ = 70 - 90 %

Recherche de règles d'association

Données d 'entrée : liste d 'achats

Achat = liste d 'articles (longueur variable)

	Produit A	Produit B	Produit C	Produit D	Produit E
Achat 1	*			*	
Achat 2	*	*	*		
Achat 3	*				*
Achat 4	*			*	*
Achat 5		*		*	

Recherche de règles d'association

Tableau de co-occurrence : combien de fois deux produits ont été achetés ensemble ?

	Produit A	Produit B	Produit C	Produit D	Produit E
Produit A	4	1	1	2	2
Produit B	1	2	1	1	0
Produit C	1	1	1	0	0
Produit D	2	1	0	3	1
Produit E	2	0	0	1	2

Illustration / Exemple

- Règle d ’association :
  - Si A alors B (règle 1)
  - Si A alors D (règle 2)
  - Si D alors A (règle 3)
- Supports :
  - Support(1)=20% ; Support(2)=Support(3)=40%
- Confiances :
  - Confiance(2) = 50% ; Confiance(3) = 67%

- On préfère la règle 3 à la règle 2.

Description de la méthode

- Support et confiance ne sont pas toujours suffisants
- Ex : Soient les 3 articles A, B et C

article	A	B	C	A et B	A et C	B et C	A, B et C
fréquence	45%	42,5%	40%	25%	20%	15%	5%

- Règles à 3 articles : même support 5%
- Confiance
  - Règle : Si A et B alors C = 0.20
  - Règle : Si A et C alors B = 0.25
  - Règle : Si B et C alors A = 0.33

Description de la méthode

- Amélioration = confiance / fréq(résultat)
- Comparer le résultat de la prédiction en utilisant la règle avec la prédiction sans la règle
- Règle intéressante si Amélioration > 1

Règle	Confiance	F(résultat)	Amélioration
Si A et B alors C	0.20	40%	0.50
Si A et C alors B	0.25	42.5%	0.59
Si B et C alors A	0.33	45%	0.74

- Règle : Si A alors B ; support=25% ; confiance=55% ; Amélioration = 1.31 → Meilleure règle

Recherche de règles

- Soient une liste de n articles et de m achats.
- 1. Calculer le nombre d’occurrences de chaque article.
- 2. Calculer le tableau des co-occurrences pour les paires d ’articles.
- 3. Déterminer les règles de niveau 2 en utilisant les valeurs de support, confiance et amélioration.
- 4. Calculer le tableau des co-occurrences pour les triplets d ’articles.
- 5. Déterminer les règles de niveau 3 en utilisant les valeurs de support, confiance et amélioration
- ...

Complexité

- Soient :
  - n : nombre de transactions dans la BD
  - m : Nombre d'attributs (items) différents

- Complexité
  - Nombre de règles d'association : $O(\mathbf{m}.2^{\mathbf{m}-1})$
  - Complexité de calcul : $O(\mathbf{n}.\mathbf{m}.2^{\mathbf{m}})$

Réduction de la complexité

- n de l'ordre du million (parcours de la liste nécessaire)
- Taille des tableaux en fonction de m et du nombre d 'articles présents dans la règle

	2	3	4
n	n(n-1)/2	n(n-1)(n-2)/6	n(n-1)(n-2)(n-3)/24
100	4950	161 700	3 921 225
10000	5.10^7	$1.7\ 10^{11}$	$4.2\ 10^{14}$

- Conclusion de la règle restreinte à un sous-ensemble de l 'ensemble des articles vendus.
  - Exemple : articles nouvellement vendues.
- Création de groupes d 'articles (différents niveaux d'abstraction).
- Elagage par support minimum.

Illustration sur une BD commerciale

Attribut	Compteur
Pain	**4**
Coca	**2**
Lait	**4**
Bière	**3**
Couches	**4**
Oeufs	**1**

Attributs (1-itemsets)

Itemset	Compteur
{Pain,Lait}	**3**
{Pain,Bière}	**2**
{Pain,Couches}	**3**
{Lait,Bière}	**2**
{Lait,Couches}	**3**
{Bière,Couches}	**3**

paires (2-itemsets)

Support Minimum = 3

Triplets (3-itemsets)

Itemset	Compteur
{Pain,Lait,Couches}	**3**
{Lait,Couches,Bière}	**2**

Si tout sous-ensemble est considéré,

$C^6_1 + C^6_2 + C^6_3 = 41$

En considérant un seuil support min,

$6 + 6 + 2 = 14$

L'algorithme Apriori [Agrawal93]

- Deux étapes
  - Recherche des k-itemsets fréquents (support≥MINSUP)
    - (Pain, Fromage, Vin) = 3-itemset
    - Principe : Les sous-itemsets d'un k-itemset fréquent sont obligatoirement fréquents
  - Construction des règles à partir des k-itemsets trouvés
    - Une règle fréquente est retenue si et seulement si sa confiance c≥ MINCONF
    - Exemple : ABCD fréquent
    - AB → CD est retenue si sa confiance ≥ MINCONF

Recherche des k-itemsets fréquents (1)

Exemple

- I = {A, B, C, D, E, F}
- T = {AB, ABCD, ABD, ABDF, ACDE, BCDF}
- MINSUP = 1/2

Calcul de L1 (ensemble des 1-itemsets)

- C_1 = I = {A,B,C,D,E,F} // C1 : ensemble de 1-itemsets candidats
- s(A) = s(B) = 5/6, s(C) = 3/6, s(D) = 5/6, s(E) = 1/6, s(F) = 2/6
- L_1 = {A, B, C, D}

Calcul de L2 (ensemble des 2-itemsets)

- C_2 = L1xL1 = {AB,AC, AD, BC, BD, CD}
- s(AB) = 4/6, s(AC) = 2/6, s(AD) = 4/6, s(BC) = 2/6, s(BD) = 4/6, s(CD) = 3/6
- L_2 = {AB,AD, BD, CD}

Recherche des k-itemsets fréquents (2)

- Calcul de L_3 (ensemble des 3-itemsets)
  - C_3 = {ABD} (ABC $\notin C_3$ car AC $\notin L_2$)
  - s(ABD) = 3/6
  - L_3 = {ABD}
- Calcul de L_4 (ensemble des 4-itemsets)
  - $C_4 = \phi$
  - $L_4 = \phi$
- Calcul de L (ensembles des itemsets fréquents)
  - $L = \cup L_i$ = {A, B, C, D, AB, AD, BD, CD, ABD}

L'algorithme Apriori

L1 = {1-itemsets fréquents};
for (k=2; Lk-1≠ ϕ; k++) **do**
    Ck = apriori_gen(Lk-1);
    forall instances t∈T **do**
        Ct = subset(Ck,t);
        forall candidats c ∈ Ct **do**
            c.count++;
    Lk = { c∈ Ck / c.count ≥ MINSUP }
L = ∪iLi;

La procédure Apriori_gen

{ Jointure L_{k-1} * L_{k-1} ; k-2 éléments communs}

insert into C_k;

select $p.item_1$, $p.item_2$, …, $p.item_{k-1}$, $q.item_{k-1}$

from L_{k-1p}, L_{k-1q}

where $p.item_1=q.item_1$, …, $p.item_{k-2}=q.item_{k-2}$

, $p.item_{k-1} < q.item_{k-1}$

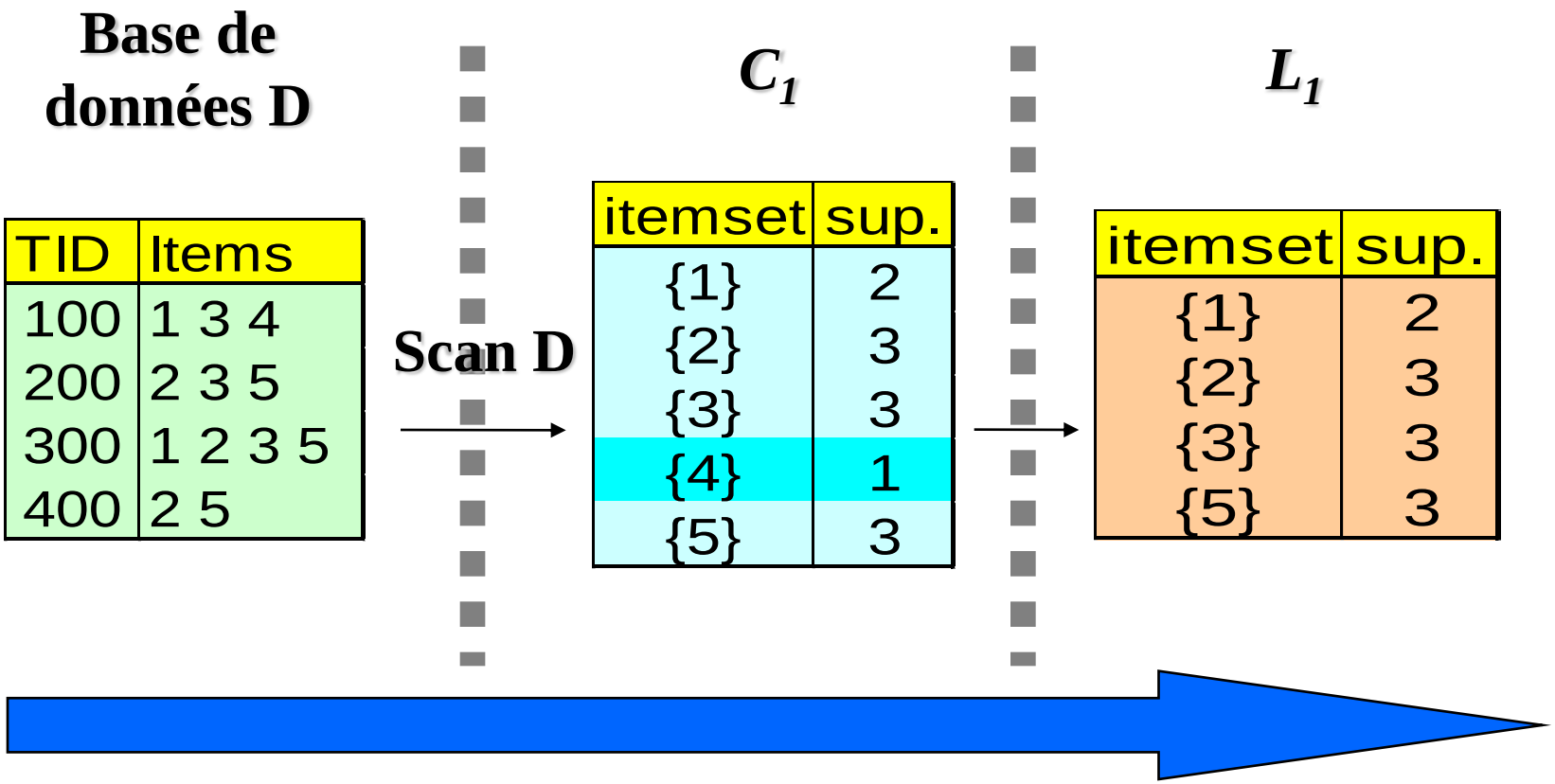
forall itemsets $c \in C_k$ **do**

forall (k-1)-itemsets $s \subset c$ **do**

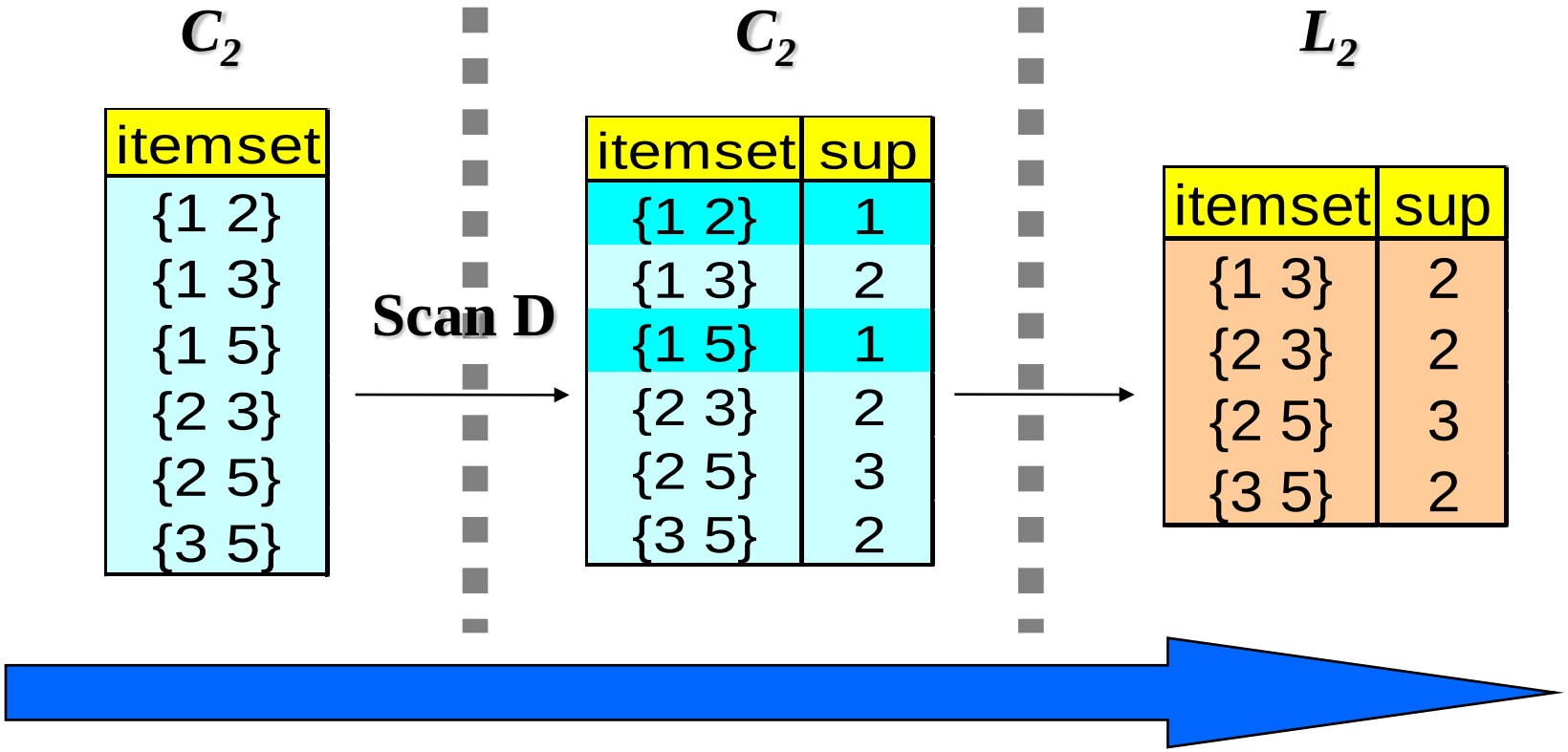
if $s \notin L_{k-1}$ **then**

delete c **from** C_k;

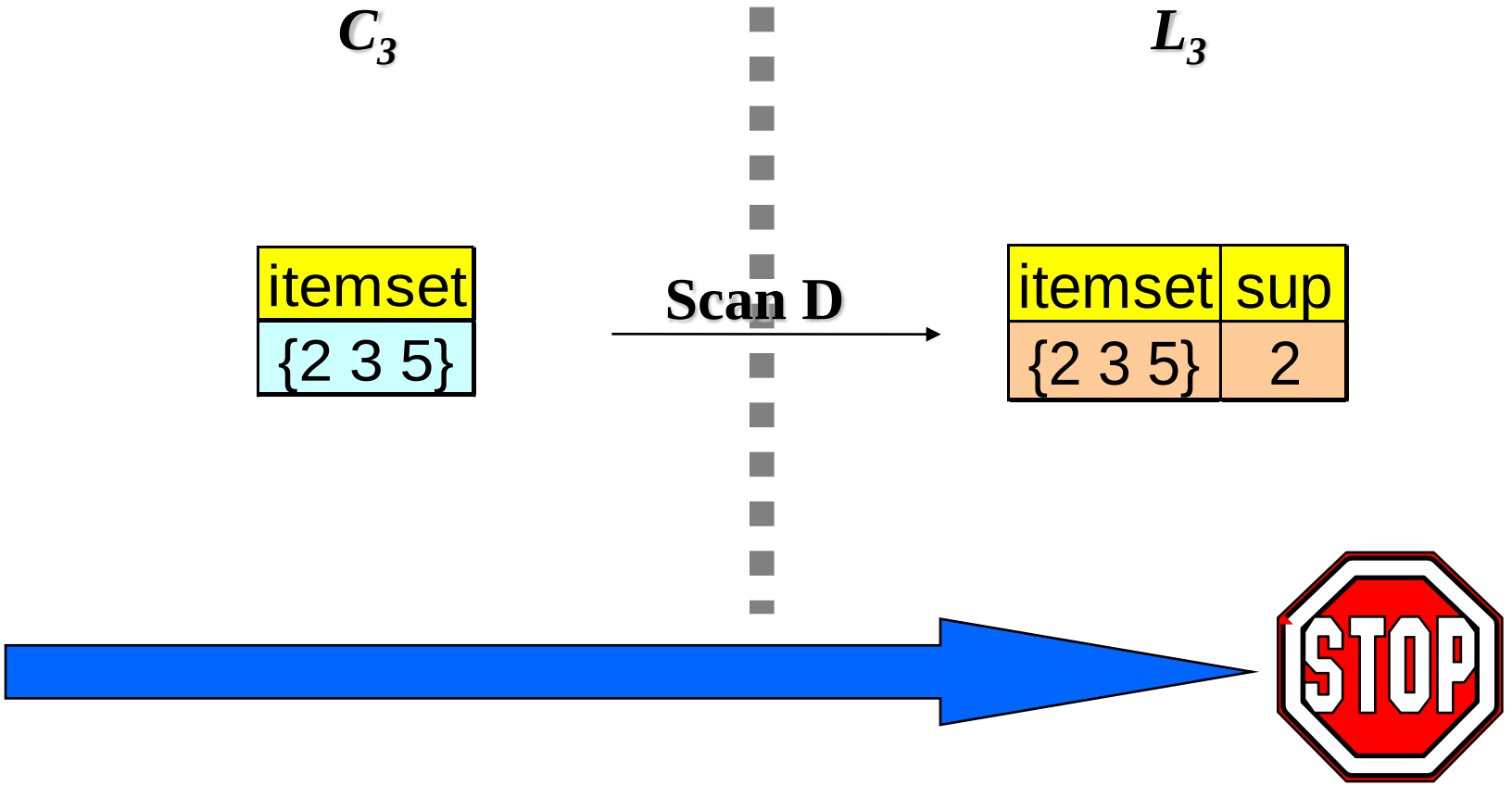
Apriori - Exemple



Apriori - Exemple

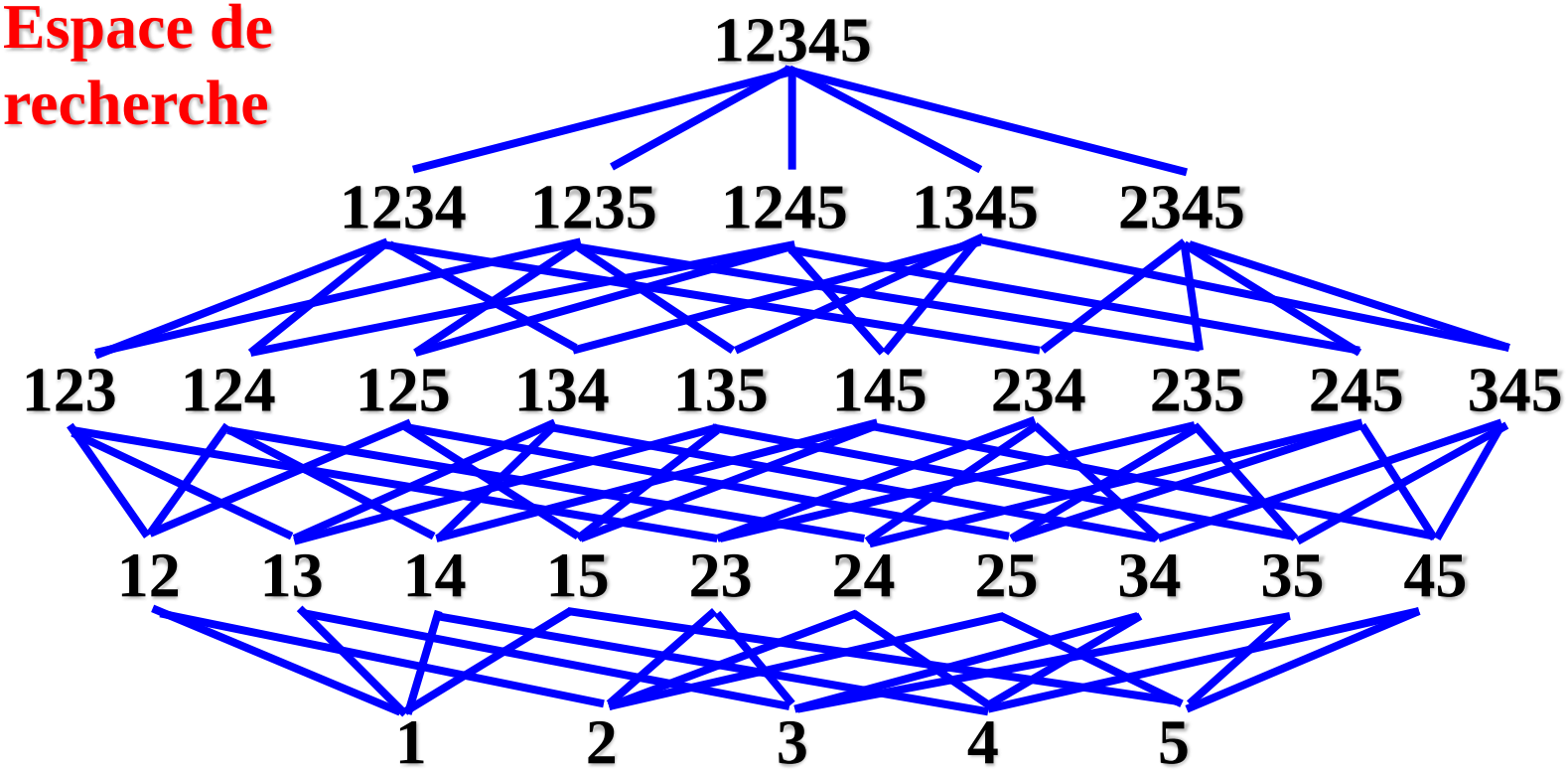


Apriori - Exemple



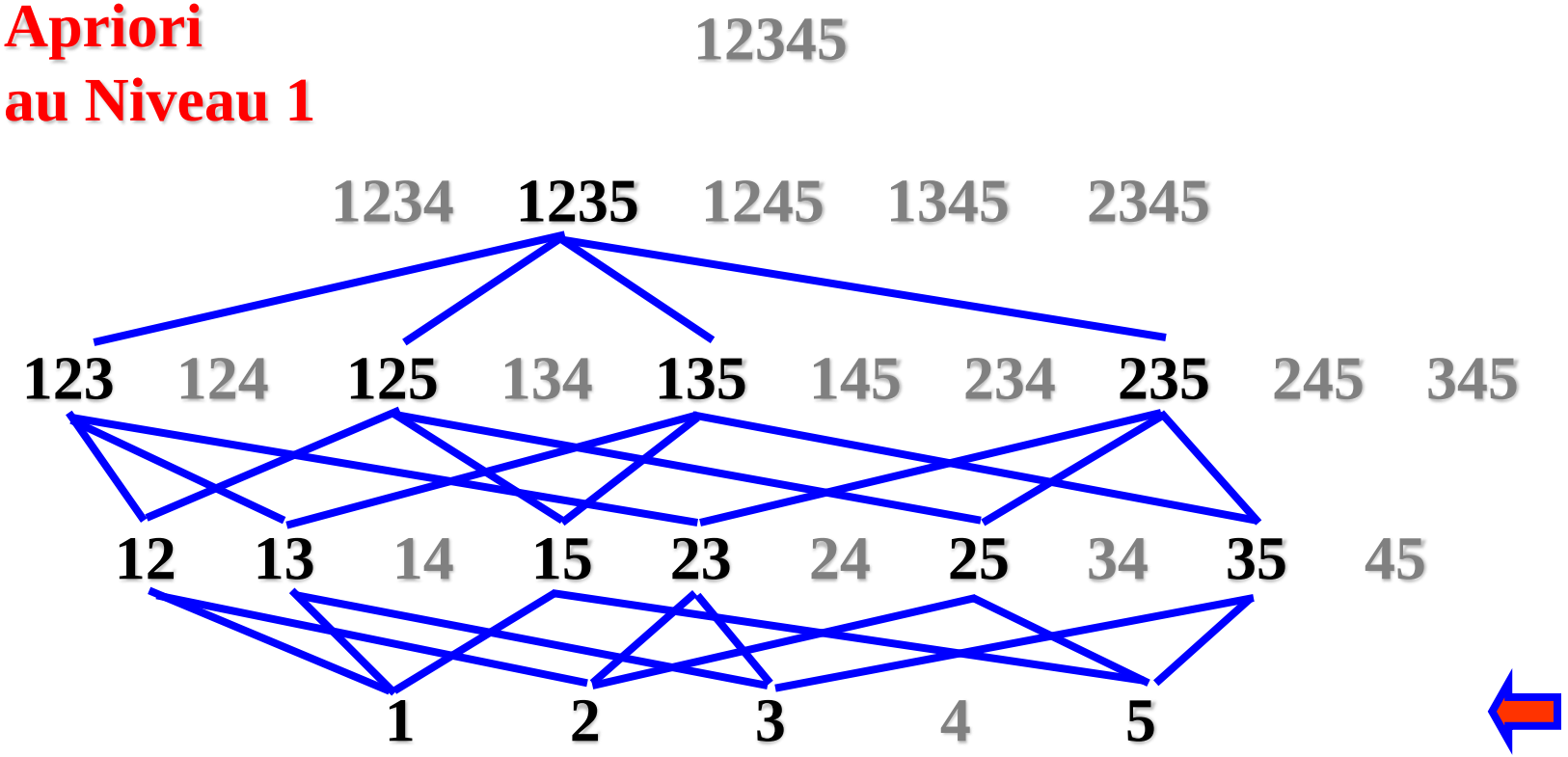
Apriori - Exemple

Espace de recherche



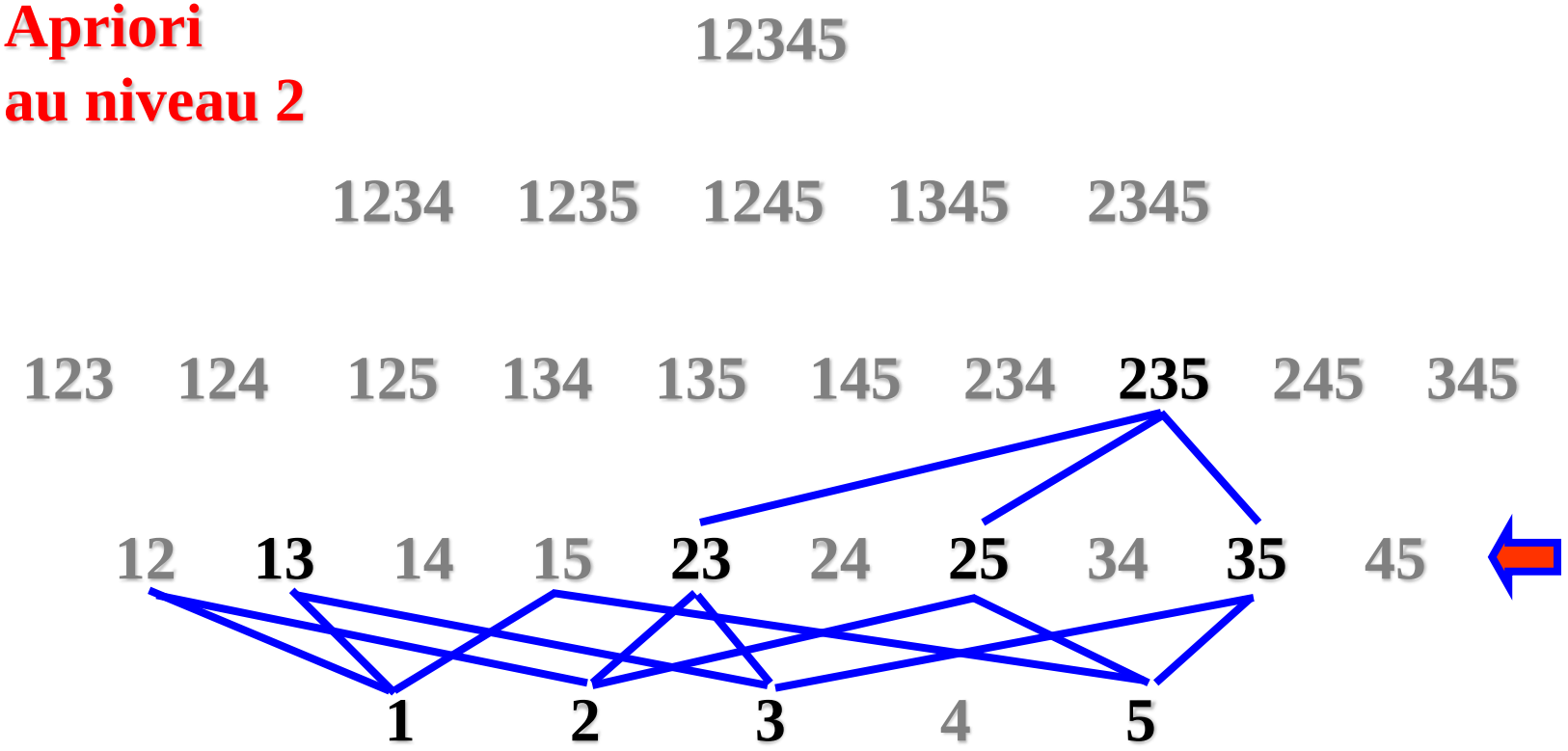
Apriori - Exemple

Apriori
au Niveau 1



Apriori - Exemple

Apriori
au niveau 2



Génération des règles à partir des itemsets

Pseudo-code :

- **pour** chaque itemset fréquent *l*
- générer tous les sous-itemsets non vides *s* de *l*
- **pour** chaque sous-itemset non vide *s* de *l*
- produire la règle "*s* ⇒ (*l*-*s*)" si *support(l)/support(s)* ≥ *min_conf*", où *min_conf* est la confiance minimale

- **Exemple : itemset fréquent l = {abc},**
- **Sous-itemsets s = {a, b, c, ab, ac, bc)**

  - a ⇒ bc, b ⇒ ac, c ⇒ ab
  - ab ⇒ c, ac ⇒ b, bc ⇒ a

Génération des règles à partir des itemsets

Règle 1 à mémoriser :

- La génération des itemsets fréquents est une opération coûteuse
- La génération des règles d'association à partir des itemsets fréquents est rapide

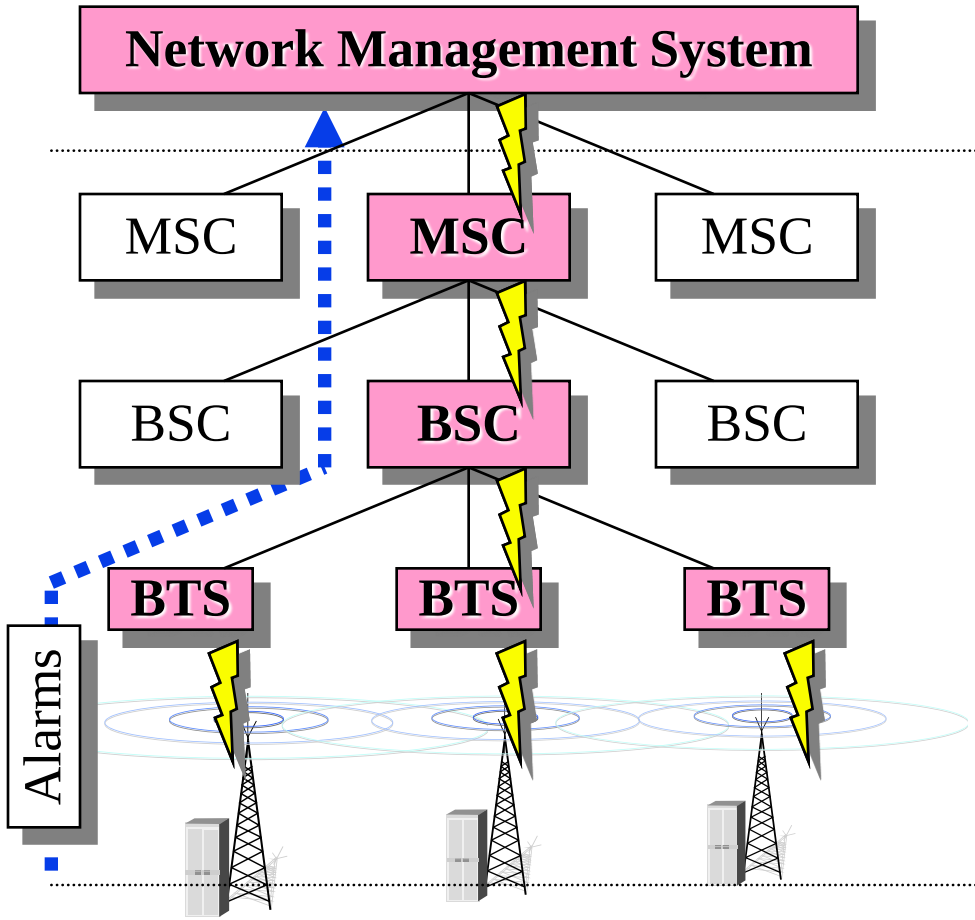
Règle 2 à mémoriser :

- Pour la génération des itemsets, le seuil support est utilisé.
- Pour la génération des règles d'association, le seuil confiance est utilisé.

Complexité en pratique ?

- A partir d'un exemple réel (petite taille) …
- Expériences réalisées sur un serveur Alpha Citum 4/275 avec 512 MB de RAM & Red Hat Linux release 5.0 (kernel 2.0.30)

Exemple de performances



Réseau switché

NOKIA

Réseau d'accés

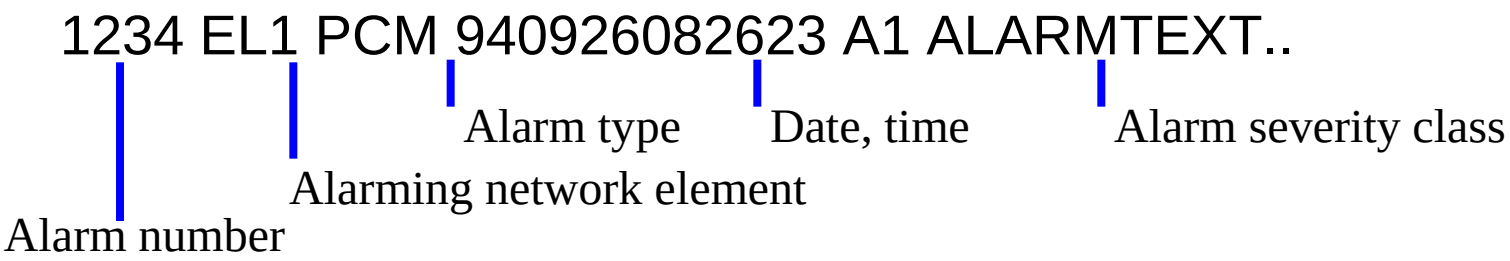
MSC Mobile station controller

BSC Base station controller

BTS Base station transceiver

Exemple de performances

Données télécom contenant des alarmes :



Exemple de données 1 :

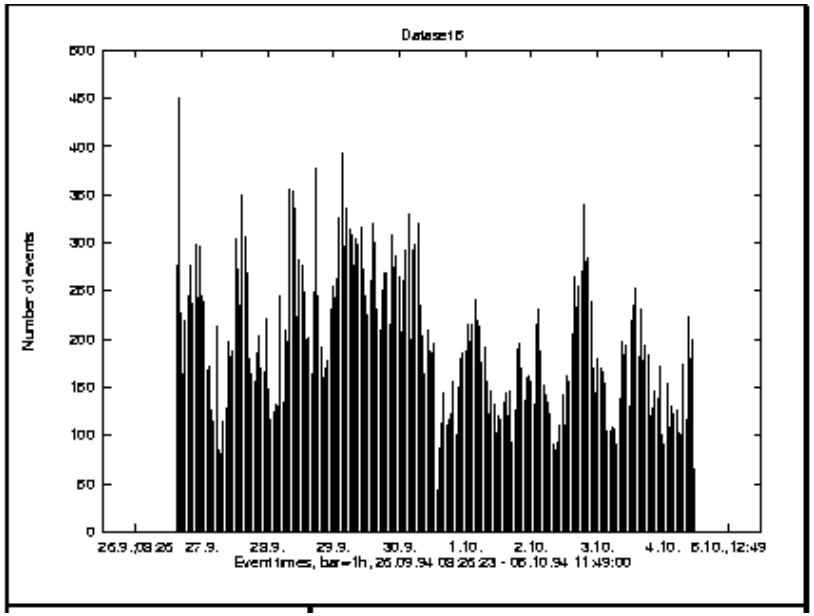
- 43 478 alarmes (26.9.94 - 5.10.94; ~ 10 jours)
- 2 234 différent types d'alarmes, 23 attributs, 5503 différentes valeurs

Exemple de données 2 :

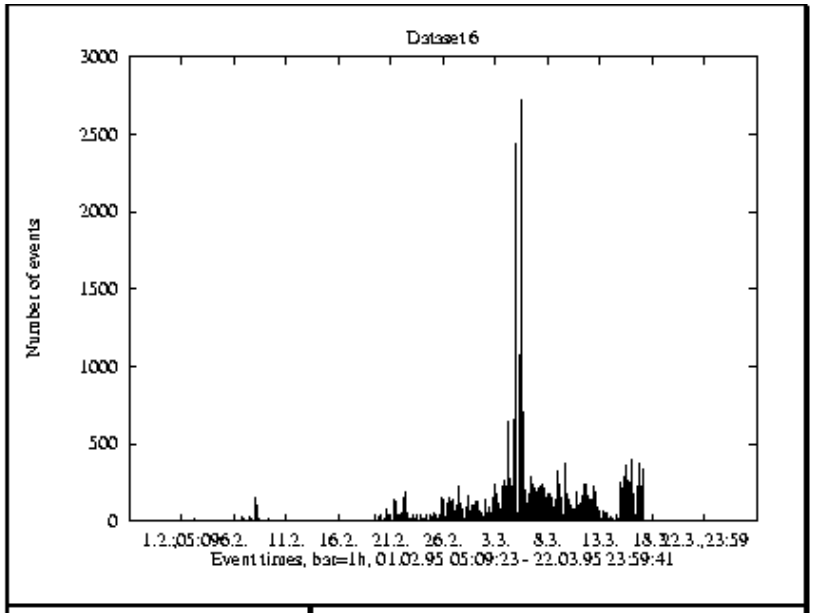
- 73 679 alarmes (1.2.95 - 22.3.95; ~ 7 semaines)
- 287 différent types d'alarmes, 19 attributs, 3411 différentes valeurs

Exemple de performances

Ensemble données 1 (~10 jours)



Ensemble données 2 (~7 semaines)



Exemple de règles :

alarm_number=1234, alarm_type=PCM $\Rightarrow$ alarm_severity=A1 [2%,45%]

Exemple de performances

Exemple de résultats pour les données 1 :

- Seuil de fréquence : 0.1
- Itemsets candidats : 109 719 Temps: 12.02 s
- Itemsets fréquents : 79 311 Temps: 64 855.73 s
- Règles : 3 750 000 Temps: 860.60 s

Exemple de résultats pour les données 2 :

- Seuil de fréquence : 0.1
- Itemsets candidats : 43 600 Temps: 1.70 s
- Itemsets fréquents : 13 321 Temps: 10 478.93 s
- Règles : 509 075 Temps: 143.35 s

Apriori - Complexité

Phase coûteuse : Génération des candidats

- Ensemble des candidats de grande taille :
  - 10^4 1-itemset fréquents génèrent 10^7 candidats pour les 2-itemsets
  - Pour trouver un itemset de taille 100, e.x., {a1, a2, …, a100}, on doit générer $2^{100} \approx 10^{30}$ candidats.

- Multiple scans de la base de données :
  - Besoin de (n +1) scans, n est la longueur de l'itemset le plus long

Apriori - Complexité

En pratique :

- Pour l'algorithme Apriori basique, le nombre d'attributs est généralement plus critique que le nombre de transactions
- Par exemple :
  - 50 attributs chacun possédant 1-3 valeurs, 100.000 transactions (not very bad)
  - 50 attributs chacun possédant 10-100 valeurs, 100.000 transactions (quite bad)
  - 10.000 attributs chacun possédant 5-10 valeurs, 100 transactions (very bad...)
- Notons :
  - Un attribut peut avoir plusieurs valeurs différentes
  - Les algorithmes traitent chaque paire attribut-valeur comme un attribut (2 attributs avec 5 valeurs → "10 attributs"

Quelques pistes pour résoudre le problème …

Apriori – Réduction de la complexité

Suppression de transactions :

- Une transaction qui ne contient pas de k-itemsets fréquents est inutile à traiter dans les parcours (scan) suivants.

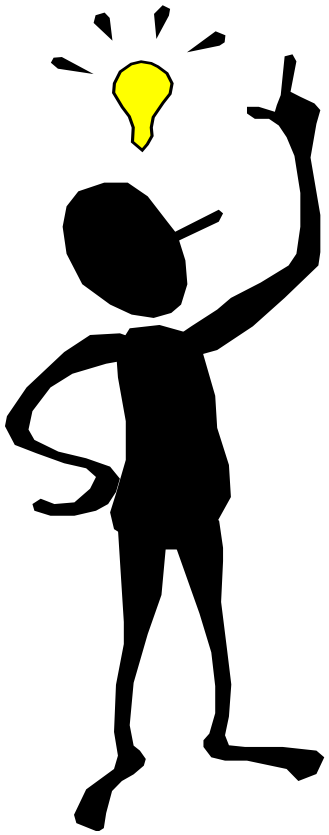
Partitionnement :

- Tout itemset qui est potentiellement fréquent dans une BD doit être potentiellement fréquent dans au moins une des partitions de la BD.

Echantillonage :

- Extraction à partir d'un sous-ensemble de données, décroitre le seuil support

Apriori - Avantages



- Résultats clairs : règles faciles à interpréter.
- Simplicité de la méthode
- Aucune hypothèse préalable (Apprentissage non supervisé)
- Introduction du temps : méthode facile à adapter aux séries temporelles. Ex : Un client ayant acheté le produit A est susceptible d ’acheter le produit B dans deux ans.

Apriori - Inconvénients

- Coût de la méthode : méthode coûteuse en temps
- Qualité des règles : production d'un nombre important de règles triviales ou inutiles.
- Articles rares : méthode non efficace pour les articles rares.
- Adapté aux règles binaires
- Apriori amélioré
  - Variantes de Apriori : DHP, DIC, etc.
  - Partition [Savasere et al. 1995]
  - Eclat et Clique [Zaki et al. 1997]
  - …

Typologie des règles

- Règles d'association binaires
  - Forme : *if* C *then* P. C,P : ensembles d'objets
- Règles d'association quantitatives
  - Forme : *if* C *then* P
    - C = terme1 & terme2 & … & termen
    - P = termen+1
    - termei = <attributj, *op*, valeur> ou <attributj, *op*, valeur_de, valeur_a>
  - Classes : valeurs de P
  - Exemple : if ((Age>30) & (situation=marié)) then prêt=prioritaire
- Règles de classification généralisée
  - Forme : *if* C *then* P, P=p1, p2, …, pm P: attribut but
- etc.

Règles d'association – Résumé

- Probablement la contribution la plus significative de la communauté KDD
- Méthodes de recherche de règles :
  - A-priori
  - Algorithmes génétiques
- Plusieurs travaux ont été publiés dans ce domaine

Règles d'association – Résumé

Plusieurs issues ont été explorées : intérêt d'une règle, optimisation des algorithmes, parallélisme et distribution, …

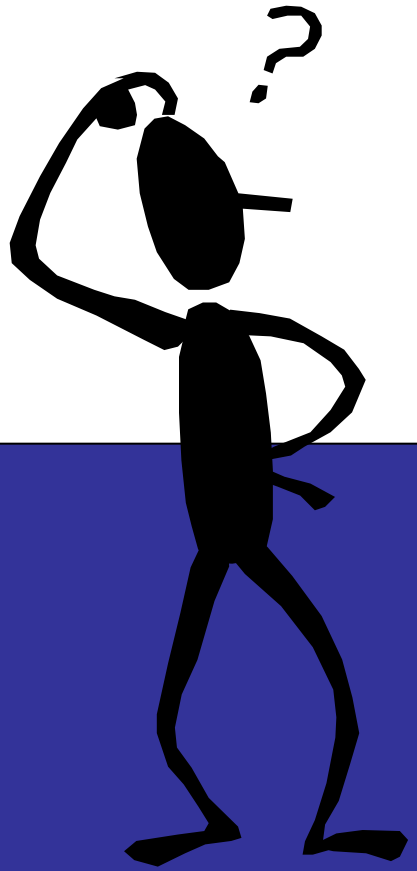
Directions de recherche :

- Règles d'associations pour d'autres types de données : données spatiales, multimedia, séries temporelles, …

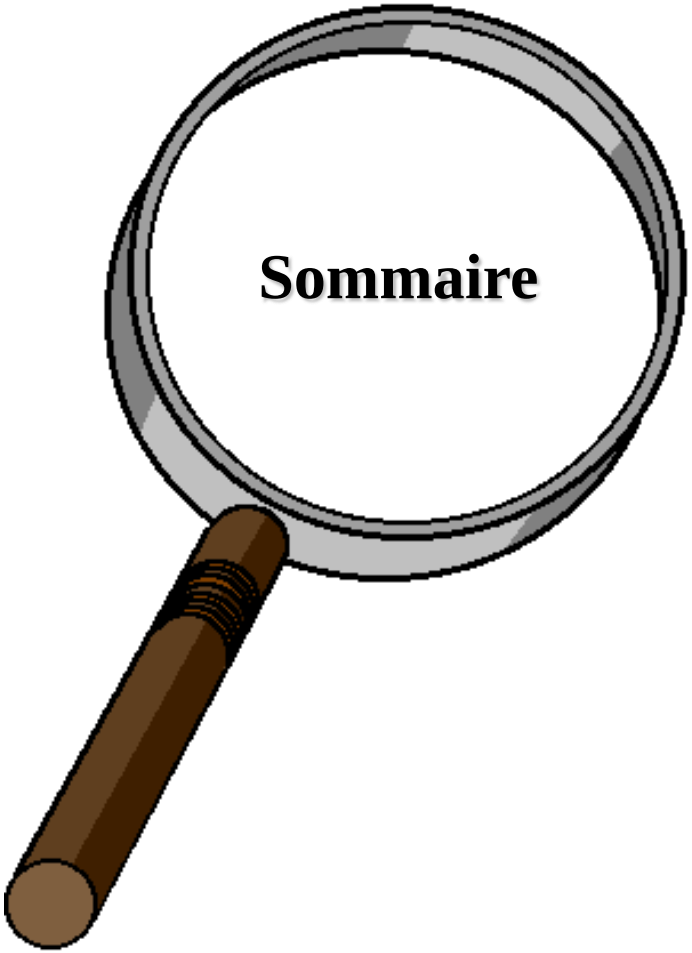
Critères pour les règles

Mesure	Formule	Effet
Support S	$\frac{C\ et\ P}{N}$	% transactions qui contiennent C et P
Confiance C	$\frac{C\ et\ P}{C}$	Probabilité conditionnelle
Intérêt I	$\frac{C\ et\ P}{C \times P}$	Privilégie les motifs rares (ayant un support faible)
Conviction V	$\frac{C \times \overline{P}}{C\ et\ \overline{P}}$	Mesure la faiblesse de (C, not P) V >> :: P se passe avec C
Piatetsky-Shapiro's	$C\ et\ P - C \times P$	Mesure la dépendance
Surprise R	$\frac{\left(C\ et\ P - C\ et\ \overline{P}\right)}{P}$	Cherche des règles étonnantes Mesure l'infirmation(C, NOT P)

Classification



Sommaire



Définition

Validation d'une classification (accuracy)

K-NN (plus proches voisins)

Arbres de décision

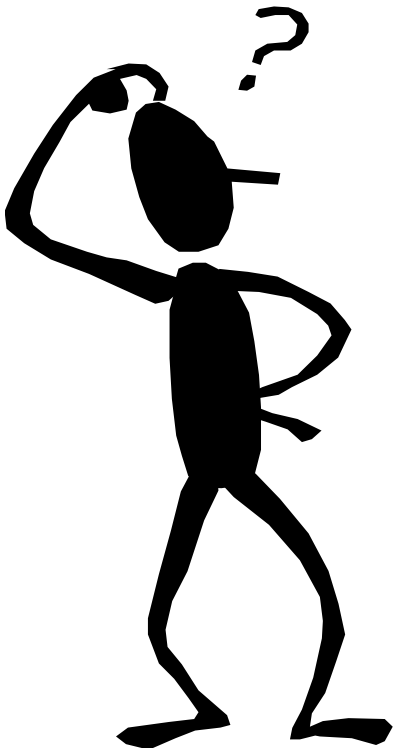
Réseaux de neurones

Autres méthodes de classification

Etude de cas réel : Protéomique

Résumé

Classification



- Elle permet de **prédire** si un élément est membre d'un groupe ou d 'une catégorie donné.
- Classes
  - Identification de groupes avec des profils particuliers
  - Possibilité de décider de l'appartenance d'une entité à une classe
- Caractéristiques
  - Apprentissage supervisé : classes connues à l'avance
  - Pb : qualité de la classification (taux d'erreur)
    - Ex : établir un diagnostic (si erreur !!!)

Classification - Applications

Accord de crédit

Marketing ciblé

Diagnostic médical

Analyse de l'effet d'un traitement

Détection de fraudes fiscales

etc.

Processus à deux étapes

Etape 1 :

Construction du modèle à partir de l'ensemble d'apprentissage (training set)

Etape 2 :

Utilisation du modèle : tester la précision du modèle et l'utiliser dans la classification de nouvelles données

Construction du modèle

Chaque instance est supposée appartenir à une classe prédéfinie

La classe d'une instance est déterminée par l'attribut "classe"

L'ensemble des instances d'apprentissage est utilisé dans la construction du modèle

Le modèle est représenté par des règles de classification, arbres de décision, formules mathématiques, ...

Utilisation du modèle

Classification de nouvelles instances ou instances inconnues

Estimer le taux d'erreur du modèle

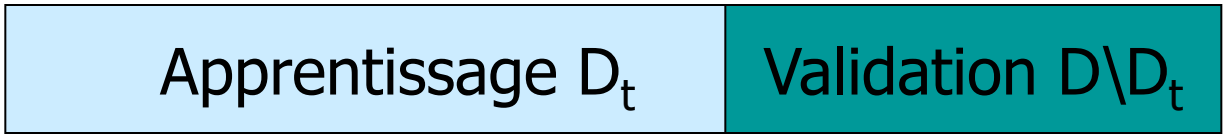
- la classe connue d'une instance test est comparée avec le résultat du modèle
- Taux d'erreur = pourcentage de tests incorrectement classés par le modèle

Validation de la Classification (accuracy)

Estimation des taux d'erreurs :

Partitionnement : apprentissage et test (ensemble de données important)

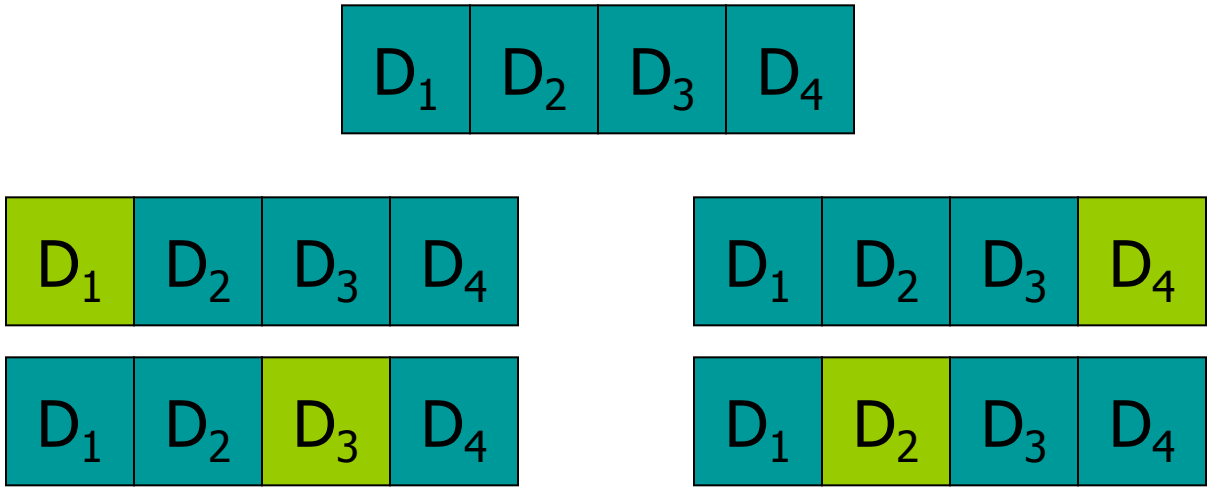
- Utiliser 2 ensembles indépendents, e.g., ensemble d'apprentissage (2/3), ensemble test (1/3)



Validation de la Classification (accuracy)

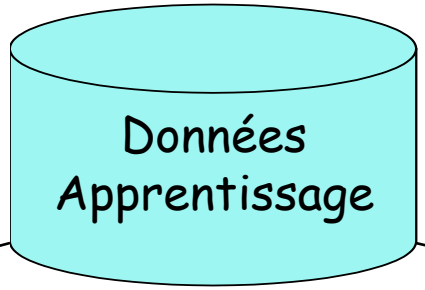
Validation croisée (ensemble de données modéré)

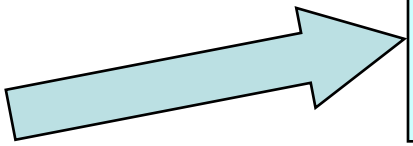
- Diviser les données en k sous-ensembles
- Utiliser k-1 sous-ensembles comme données d'apprentissage et un sous-ensemble comme données test

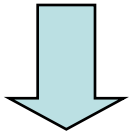


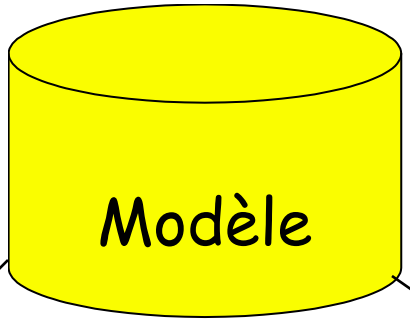
Bootstrapping : n instances test aléatoires (ensemble de données réduit)

Exemple : Construction du modèle





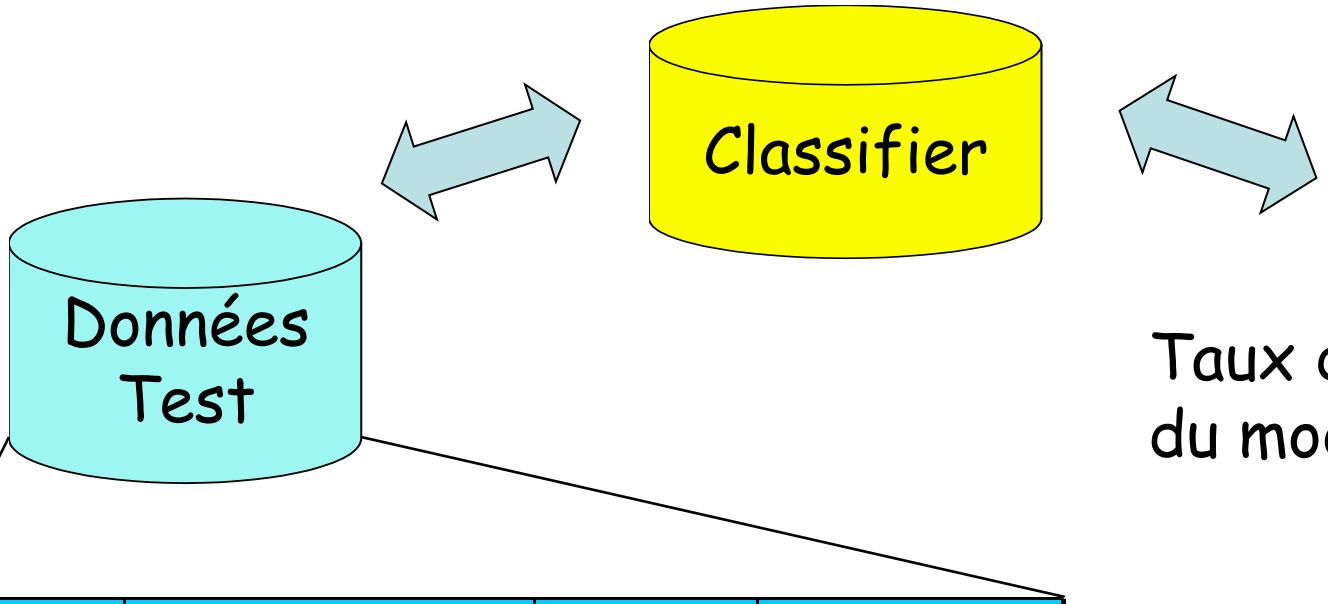




Nom	Rang	Année	Titulaire
Mary	Assistant Prof	3	non
James	Assistant Prof	7	oui
Bill	Professor	2	oui
John	Associate Prof	7	oui
Mark	Assistant Prof	6	non
Annie	Associate Prof	3	non

Si Rang = 'Professor'
Ou Année > 6
Alors Titulaire = Oui

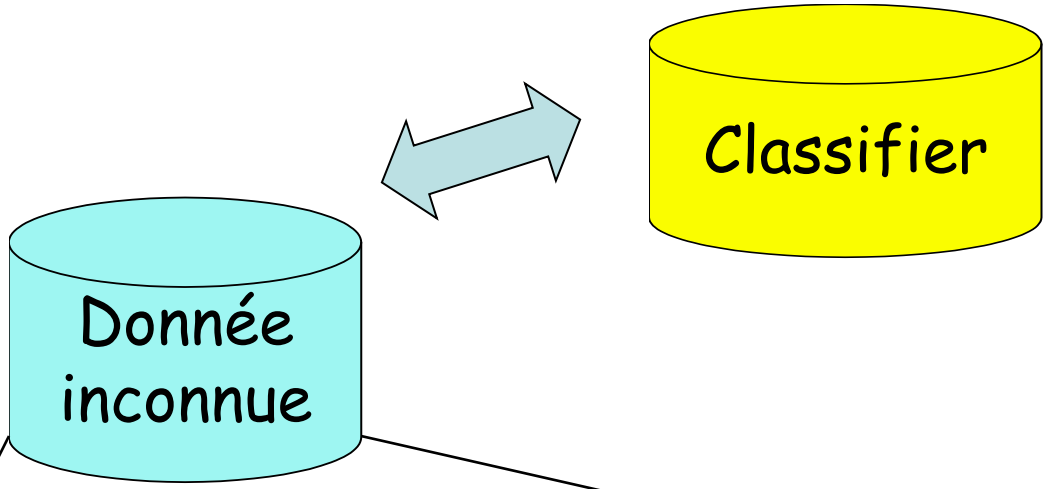
Exemple : Utilisation du modèle



Taux d'erreur
du modèle ?

Nom	Rang	Année	Titulaire
Tom	Assistant Prof	2	non
Lisa	Associate Prof	7	non
Jack	Professor	5	oui
Ann	Assistant Prof	7	oui

Exemple : Utilisation du modèle

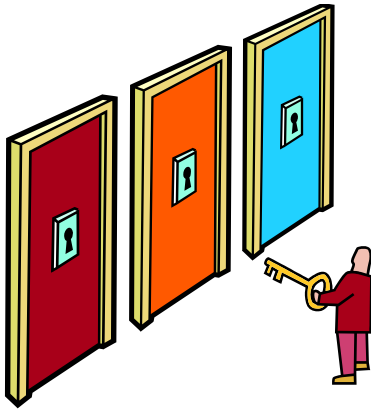


Titulaire ?

Nom	Rang	Année	Titulaire
Jeff	Professor	4	?
Paul	Associate Prof	7	?

Oui
Oui

Evaluation des méthodes de classification



Taux d'erreur (Accuracy)

Temps d'exécution (construction, utilisation)

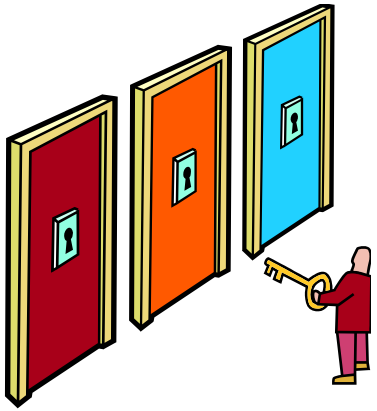
Robustesse (bruit, données manquantes,...)

Extensibilité

Interprétabilité

Simplicité

Méthodes de Classification



- Méthode K-NN (plus proche voisin)
- Arbres de décision
- Réseaux de neurones
- Classification bayésienne

- Caractéristiques
  - Apprentissage supervisé (classes connues)

Méthode des plus proches voisins

Méthode dédiée à la classification (k-NN : nearest Neighbors).

Méthode de raisonnement à partir de cas : prendre des décisions en recherchant un ou des cas similaires déjà résolus.

Pas d'étape d 'apprentissage : construction d 'un modèle à partir d'un échantillon d 'apprentissage (réseaux de neurones, arbres de décision, …).

Modèle = échantillon d'apprentissage + fonction de distance + fonction de choix de la classe en fonction des classes des voisins les plus proches.

Algorithme kNN (K-nearest neighbors)

Objectif : affecter une classe à une nouvelle instance

donnée : un échantillon de m enregistrements classés (x, c(x))

entrée : un enregistrement y

- 1. Déterminer les k plus proches enregistrements de y
- 2. combiner les classes de ces k exemples en une classe c

sortie : la classe de y est c(y)=c

Algorithme kNN : sélection de la classe

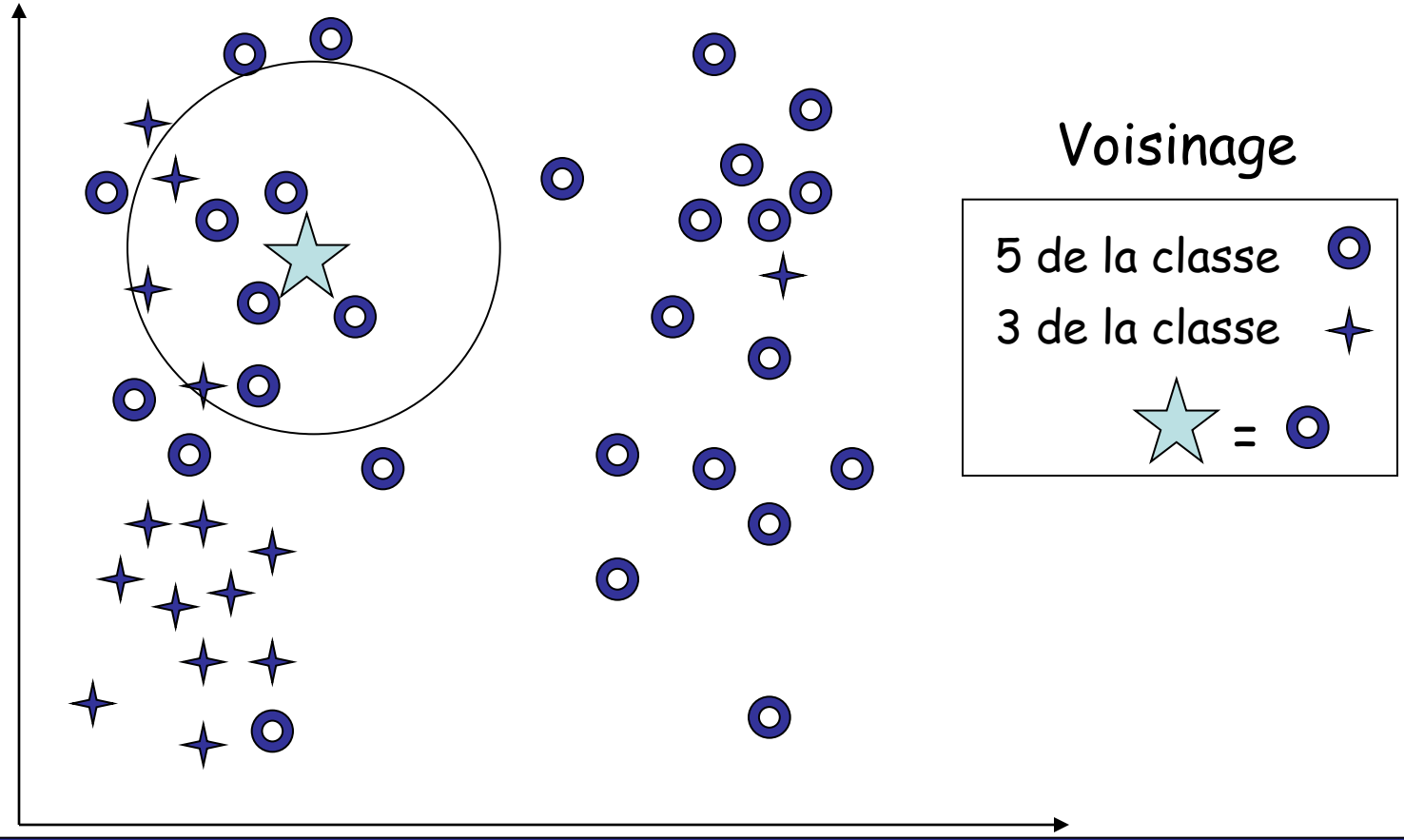
Solution simple : rechercher le cas le plus proche et prendre la même décision (Méthode 1-NN).

Combinaison des k classes :

- Heuristique : k = nombre d 'attributs + 1
- Vote majoritaire : prendre la classe majoritaire.
- Vote majoritaire pondéré : chaque classe est pondérée. Le poids de c(xi) est inversement proportionnel à la distance d(y,xi).

Confiance : Définir une confiance dans la classe attribuée = rapport entre les votes gagnants et le total des votes.

Illustration



Retour sur KNN : Exemple (1)

Customer	Age	Income	No. credit cards	Loyal
John 	35	35K	3	No
Rachel 	22	50K	2	Yes
Hannah 	63	200K	1	No
Tom 	59	170K	1	No
Nellie 	25	40K	4	Yes
David 	37	50K	2	?

Retour sur KNN : Exemple (2)

K = 3

Customer	Age	Income	No. credit cards	Loyal
John 	35	35K	3	No
Rachel 	22	50K	2	Yes
Hannah 	63	200K	1	No
Tom 	59	170K	1	No
Nellie 	25	40K	4	Yes
David 	37	50K	2	**Yes**

Distance from David
sqrt [(35-37)²+(35-50)² +(3-2)²]=**15.16**
sqrt [(22-37)²+(50-50)² +(2-2)²]=**15**
sqrt [(63-37)²+(200-50)² +(1-2)²]=**152.23**
sqrt [(59-37)²+(170-50)² +(1-2)²]=**122**
sqrt [(25-37)²+(40-50)² +(4-2)²]=**15.74**

Algorithme kNN : critique

Pas d'apprentissage : introduction de nouvelles données ne nécessite pas la reconstruction du modèle.

Clarté des résultats

Tout type de données

Nombre d'attributs

Temps de classification : -

Stocker le modèle : -

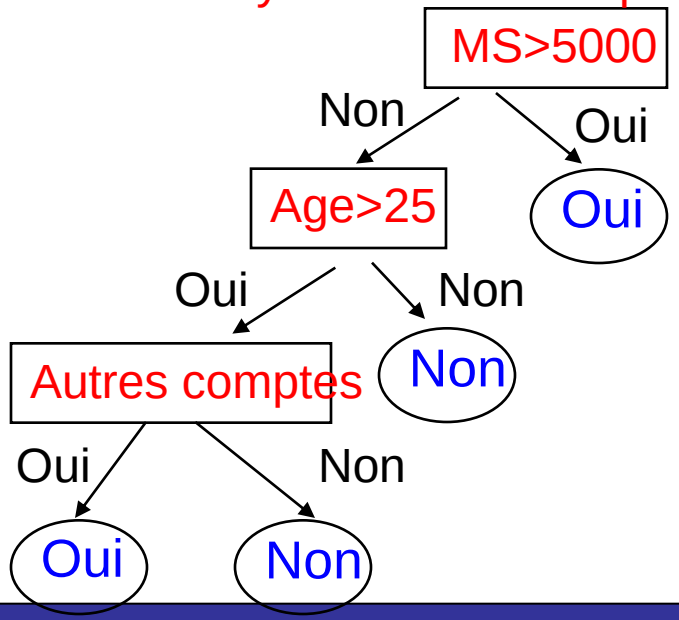
Distance et nombre de voisins : dépend de la distance, du nombre de voisins et du mode de combinaison.

Arbres de décision

- **Génération d'arbres de décision à partir des données**
- **Arbre = Représentation graphique d'une procédure de classification**

Accord d'un prêt bancaire

MS : moyenne solde compte courant



Un arbre de décision est un arbre où :

Noeud interne = un attribut

Branche d'un noeud = un test sur un attribut

Feuilles = classe donnée

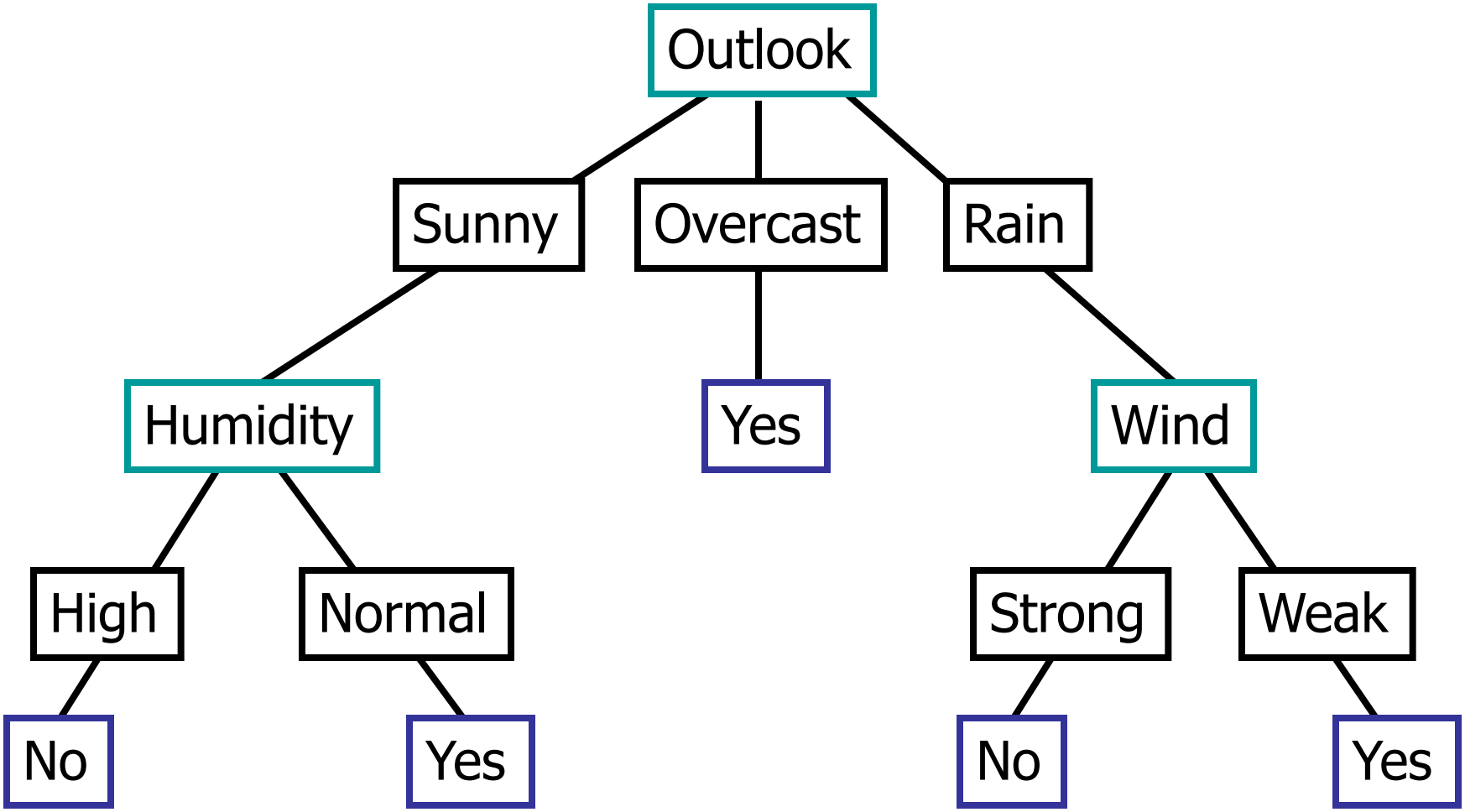
Arbre de décision - Exemple

Ensemble d'apprentissage

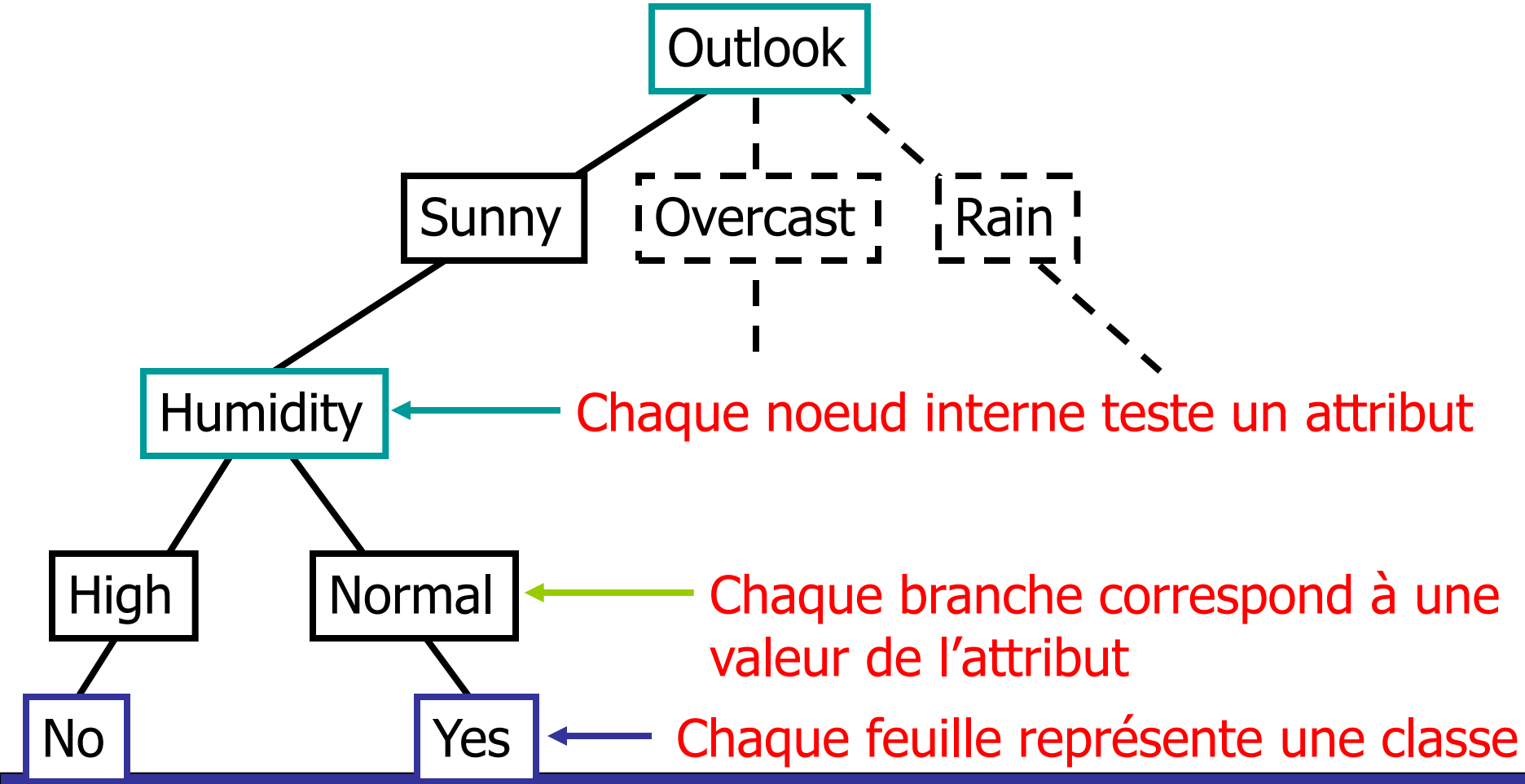
Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Jouer au tennis ?

Arbre de décision - Exemple



Exemple – Jouer au tennis ?

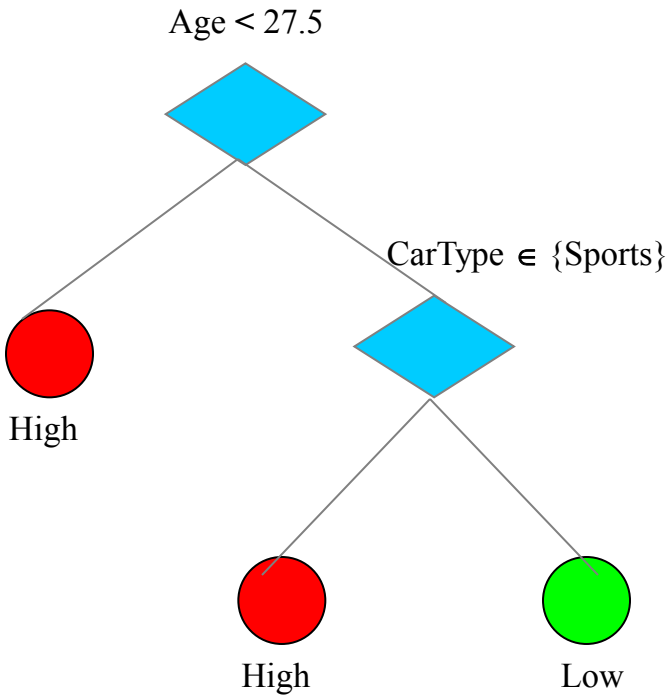


Arbres de décision – Exemple

Risque - Assurances

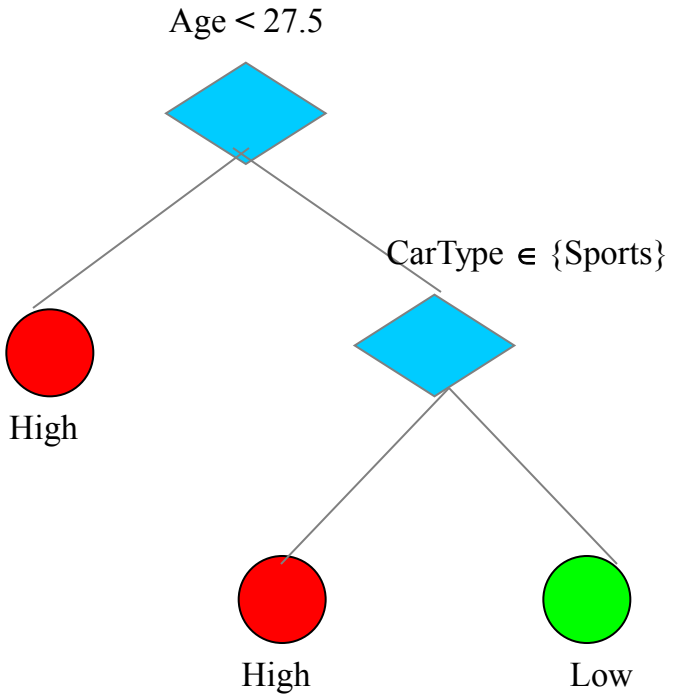
Tid	Age	Car Type	Class
0	23	Family	High
1	17	Sports	High
2	43	Sports	High
3	68	Family	Low
4	32	Truck	Low
5	20	Family	High

Numérique Enumératif



Age=40, CarType=Family ⇒ Class=Low

Des arbres de décision aux règles



1) Age < 27.5 ⇒ High

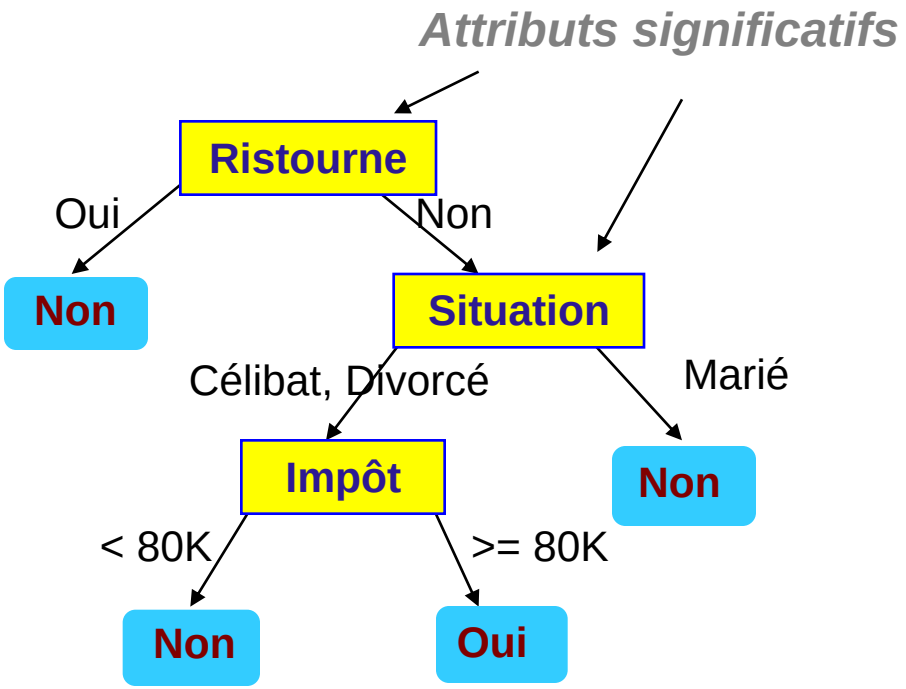
2) Age >= 27.5 and
CarType = Sports ⇒ High

3) Age >= 27.5 and
CarType ≠ Sports ⇒ Low

Arbres de décision – Exemple
Détection de fraudes fiscales

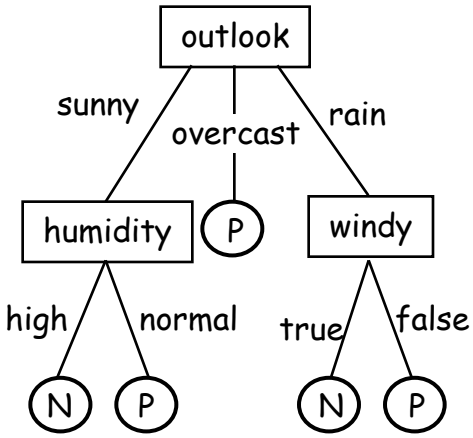
énumératif énumératif numérique classe

Id	Ristourne	Situation famille	Impôt revenu	Fraude
1	Oui	Célibat.	125K	Non
2	Non	Marié	100K	Non
3	Non	Célibat.	70K	Non
4	Oui	Marié	120K	Non
5	Non	Divorcé	95K	Oui
6	Non	Marié	60K	Non
7	Oui	Divorcé	220K	Non
8	Non	Célibat.	85K	Oui
9	Non	Marié	75K	Non
10	Non	Célibat.	90K	Oui



- **L'attribut significatif à un noeud est déterminé en se basant sur l'indice Gini.**
- **Pour classer une instance : descendre dans l'arbre selon les réponses aux différents tests. Ex = (Ristourne=Non, Situation=Divorcé, Impôt=100K) ➔ Oui**

De l'arbre de décision aux règles de classification

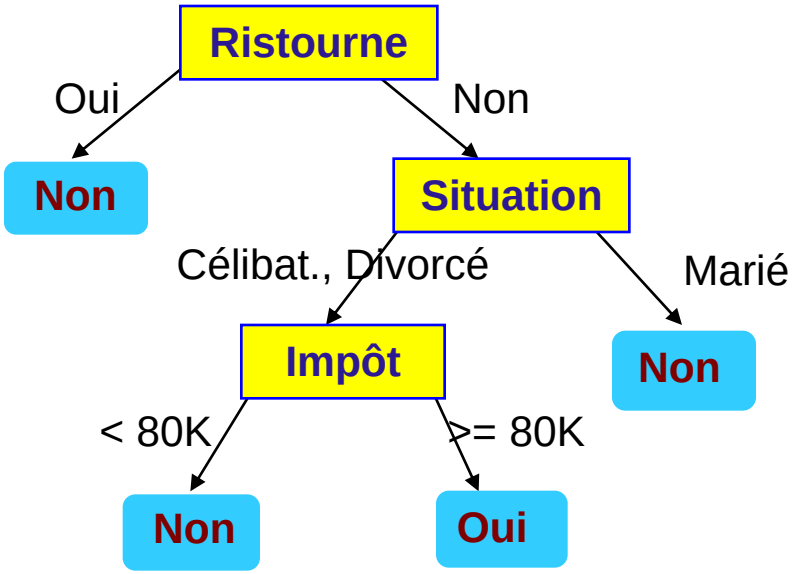


Si outlook=sunny
Et humidity=normal
Alors play tennis

- Une règle est générée pour chaque chemin de l'arbre (de la racine à une feuille)
- Les paires attribut-valeur d'un chemin forment une conjonction
- Le nœud terminal représente la classe prédite
- Les règles sont généralement plus faciles à comprendre que les arbres

Des arbres de décision aux règles

Arbre de décision = Système de règles exhaustives et mutuellement exclusives



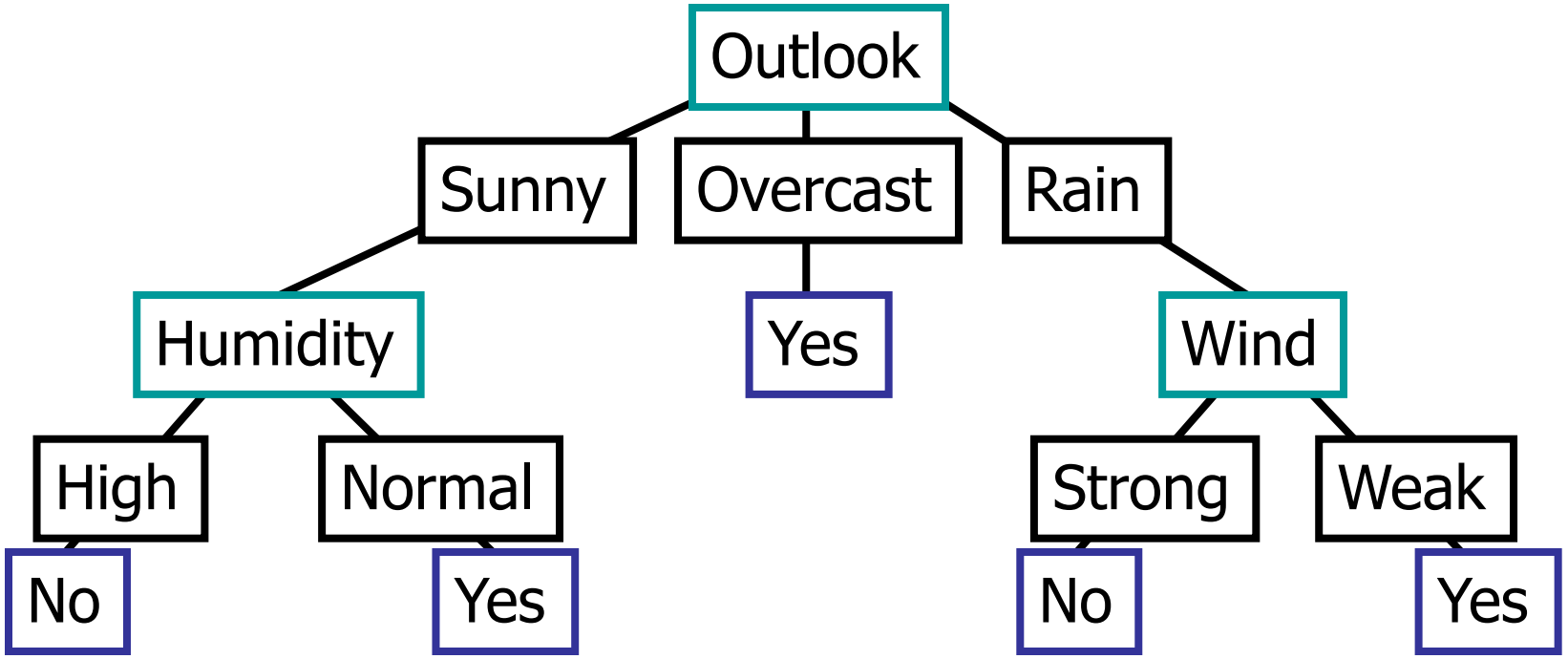
1) Ristourne = Oui $\Rightarrow$ Non

2) Ristourne = Non et Situation in {Célibat., Divorcé} et Impôt < 80K $\Rightarrow$ Non

3) Ristourne = Non et Situation in {Célibat., Divorcé} et Impôt >= 80K $\Rightarrow$ Oui

4) Ristourne = Non et Situation in {Marié} $\Rightarrow$ Non

Des arbres de décision aux règles



R_1: If (Outlook=Sunny) $\wedge$ (Humidity=High) Then PlayTennis=No
R_2: If (Outlook=Sunny) $\wedge$ (Humidity=Normal) Then PlayTennis=Yes
R_3: If (Outlook=Overcast) Then PlayTennis=Yes
R_4: If (Outlook=Rain) $\wedge$ (Wind=Strong) Then PlayTennis=No
R_5: If (Outlook=Rain) $\wedge$ (Wind=Weak) Then PlayTennis=Yes

Génération de l'arbre de décision

Deux phases dans la génération de l'arbre :

1. Construction de l'arbre
   - Arbre peut atteindre une taille élevée
2. Elaguer l'arbre (Pruning)
   - Identifier et supprimer les branches qui représentent du "bruit" → Améliorer le taux d'erreur

Algorithmes de classification

Construction de l'arbre

- Au départ, toutes les instances d'apprentissage sont à la racine de l'arbre
- Sélectionner un attribut et choisir un test de séparation (split) sur l'attribut, qui sépare le "mieux" les instances.
- La sélection des attributs est basée sur une heuristique ou une mesure statistique.
- Partitionner les instances entre les noeuds fils suivant la satisfaction des tests logiques

Algorithmes de classification

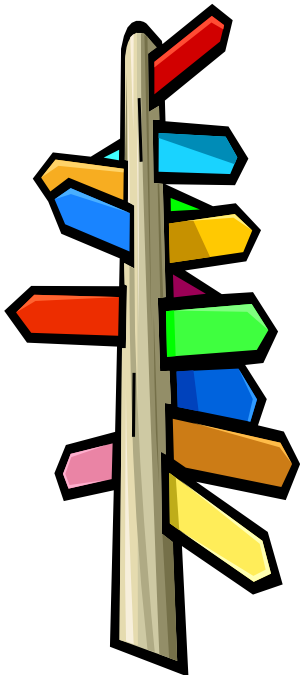
- Traiter chaque nœud fils de façon récursive
- Répéter jusqu'à ce que tous les nœuds soient des terminaux. Un nœud courant est terminal si :
  - Il n'y a plus d'attributs disponibles
  - Le nœud est "pur", i.e. toutes les instances appartiennent à une seule classe,
  - Le nœud est "presque pur", i.e. la majorité des instances appartiennent à une seule classe (Ex : 95%)
  - Nombre minimum d'instances par branche (Ex : algorithme C5 évite la croissance de l'arbre, k=2 par défaut)
- Etiqueter le nœud terminal par la classe majoritaire

Algorithmes de classification

Elaguer l'arbre obtenu (pruning)

- Supprimer les sous-arbres qui n'améliorent pas l'erreur de la classification (accuracy) ➔ arbre ayant un meilleur pouvoir de généralisation, même si on augmente l'erreur sur l'ensemble d'apprentissage
- Eviter le problème de sur-spécialisation (over-fitting), i.e., on a appris "par cœur" l'ensemble d'apprentissage, mais on n'est pas capable de généraliser

Sur-spécialisation - arbre de décision



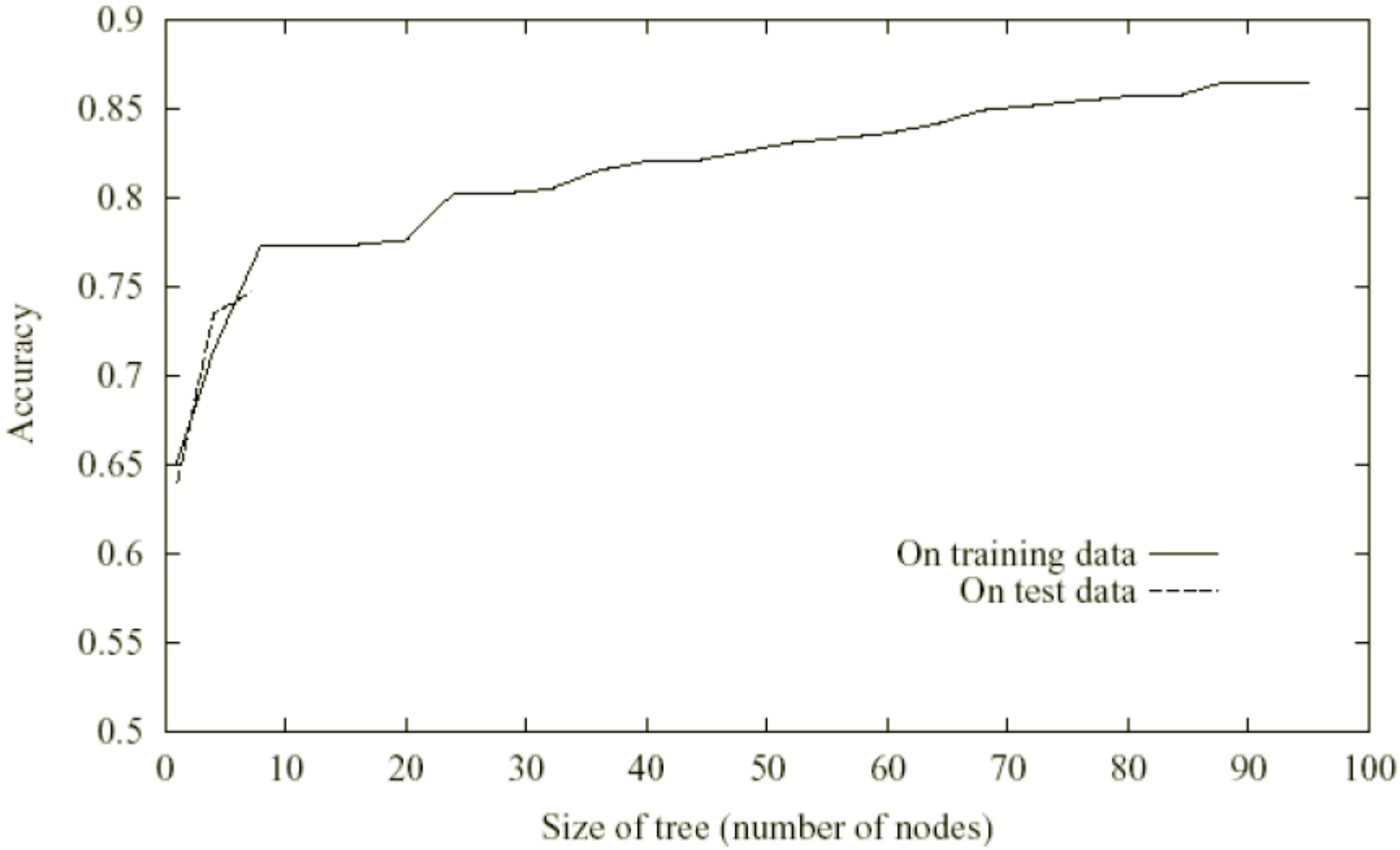
L'arbre généré peut sur-spécialiser l'ensemble d'apprentissage

- Plusieurs branches
- Taux d'erreur important pour les instances inconnues

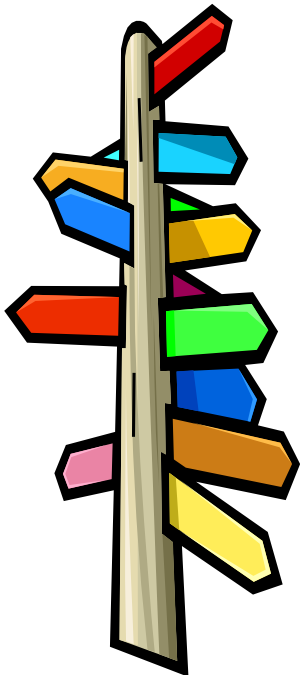
Raisons de la sur-spécialisation

- bruits et exceptions
- Peu de donnée d'apprentissage
- Maxima locaux dans la recherche gloutonne

Overfitting dans les arbres de décision



Comment éviter l'overfitting ?



Deux approches :

Pré-élagage : Arrêter de façon prématurée la construction de l'arbre

Post-élagage : Supprimer des branches de l'arbre complet ("fully grown")

Convertir l'arbre en règles ; élaguer les règles de façon indépendante (C4.5)

Construction de l'arbre - Synthèse

Evaluation des différents branchements pour tous les attributs

Sélection du "meilleur" branchement "et de l'attribut "gagnant"

Partitionner les données entre les fils

Construction en largeur (C4.5) ou en profondeur (SPLIT)

Questions critiques :

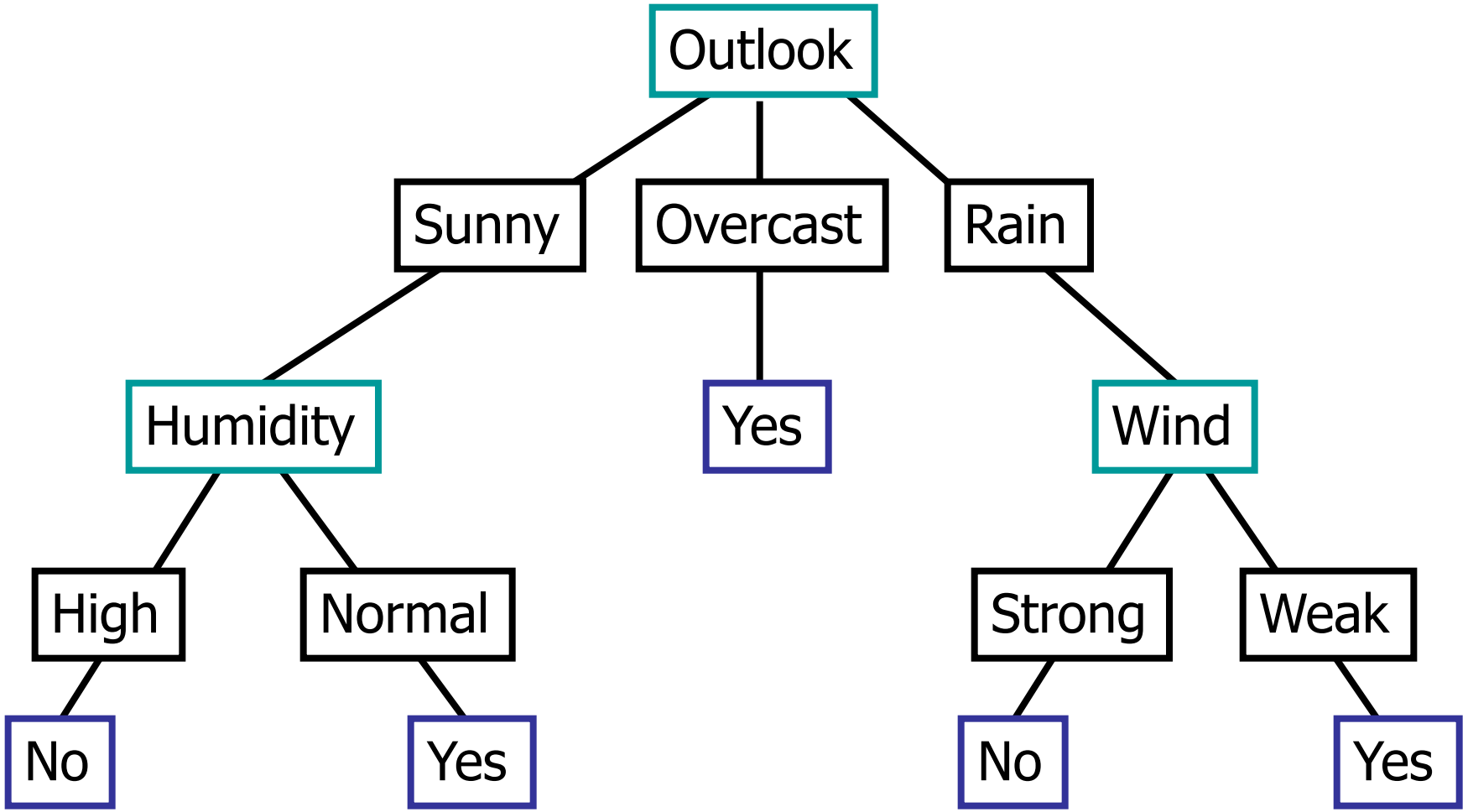
- Formulation des tests de branchement
- Mesure de sélection des attributs

Exemple : Jouer au tennis ?

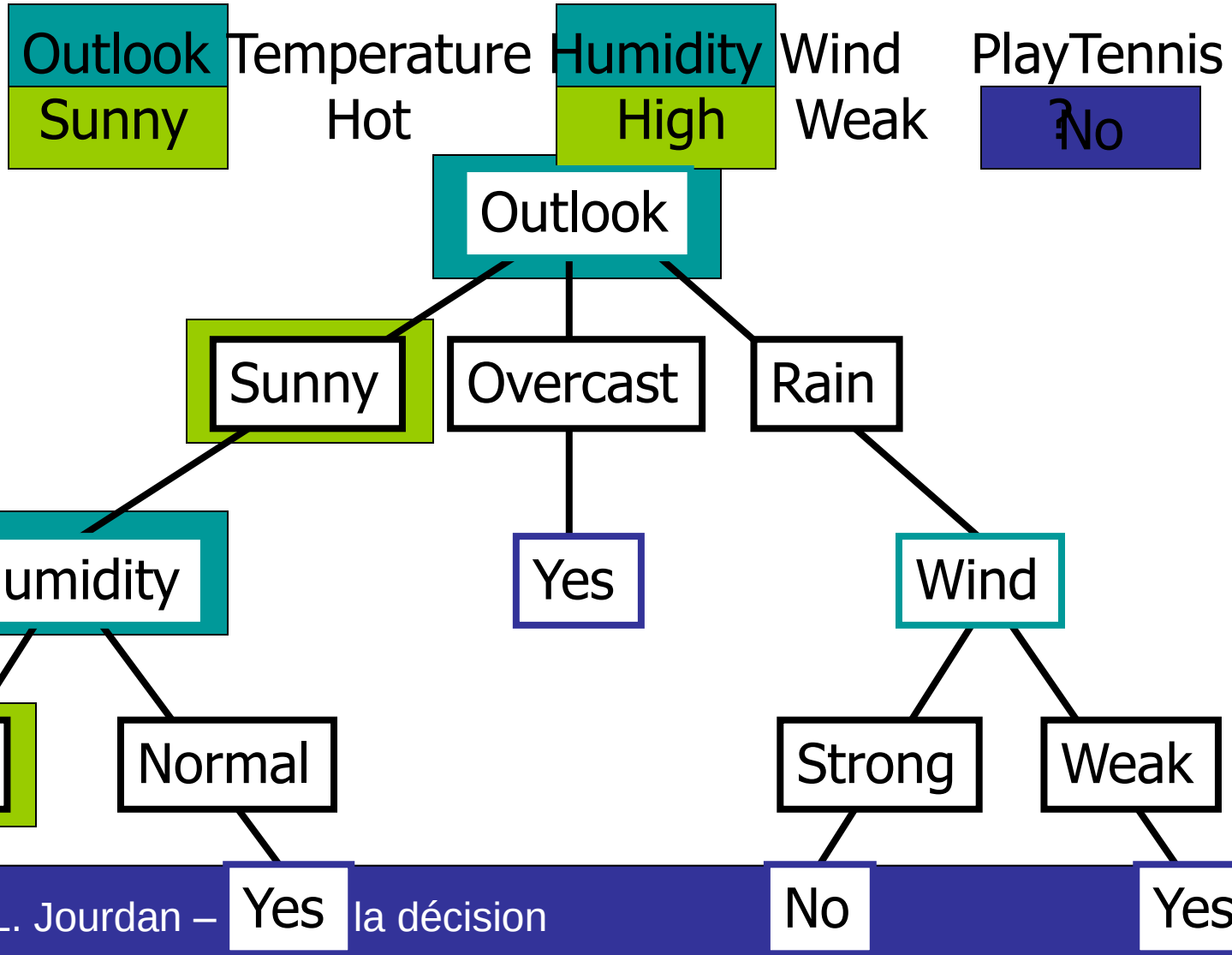
Ensemble d'apprentissage

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Arbre de décision obtenu avec ID3 (Quinlan 86)

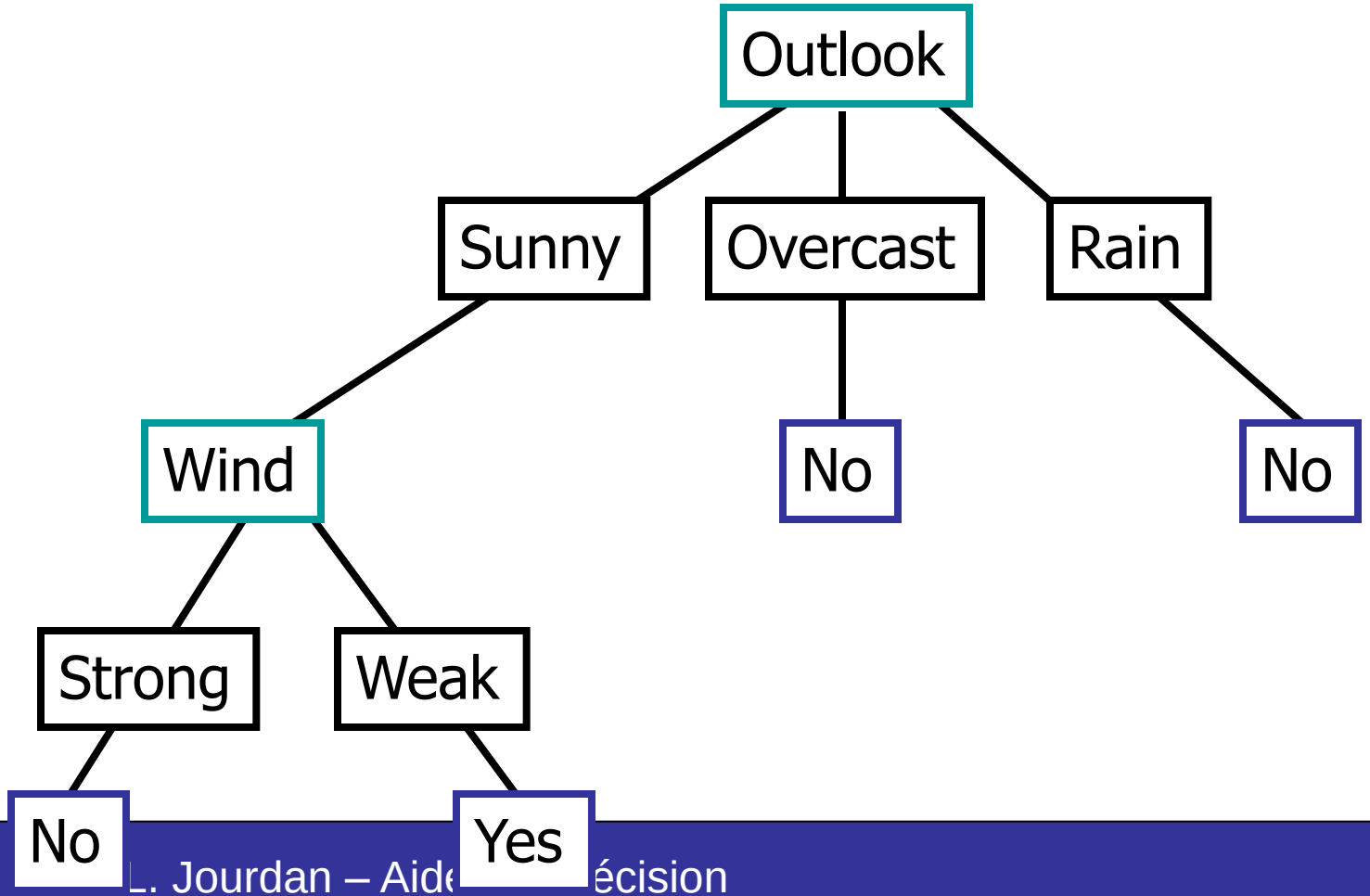


Arbre de décision obtenu avec ID3 (Quinlan 86)

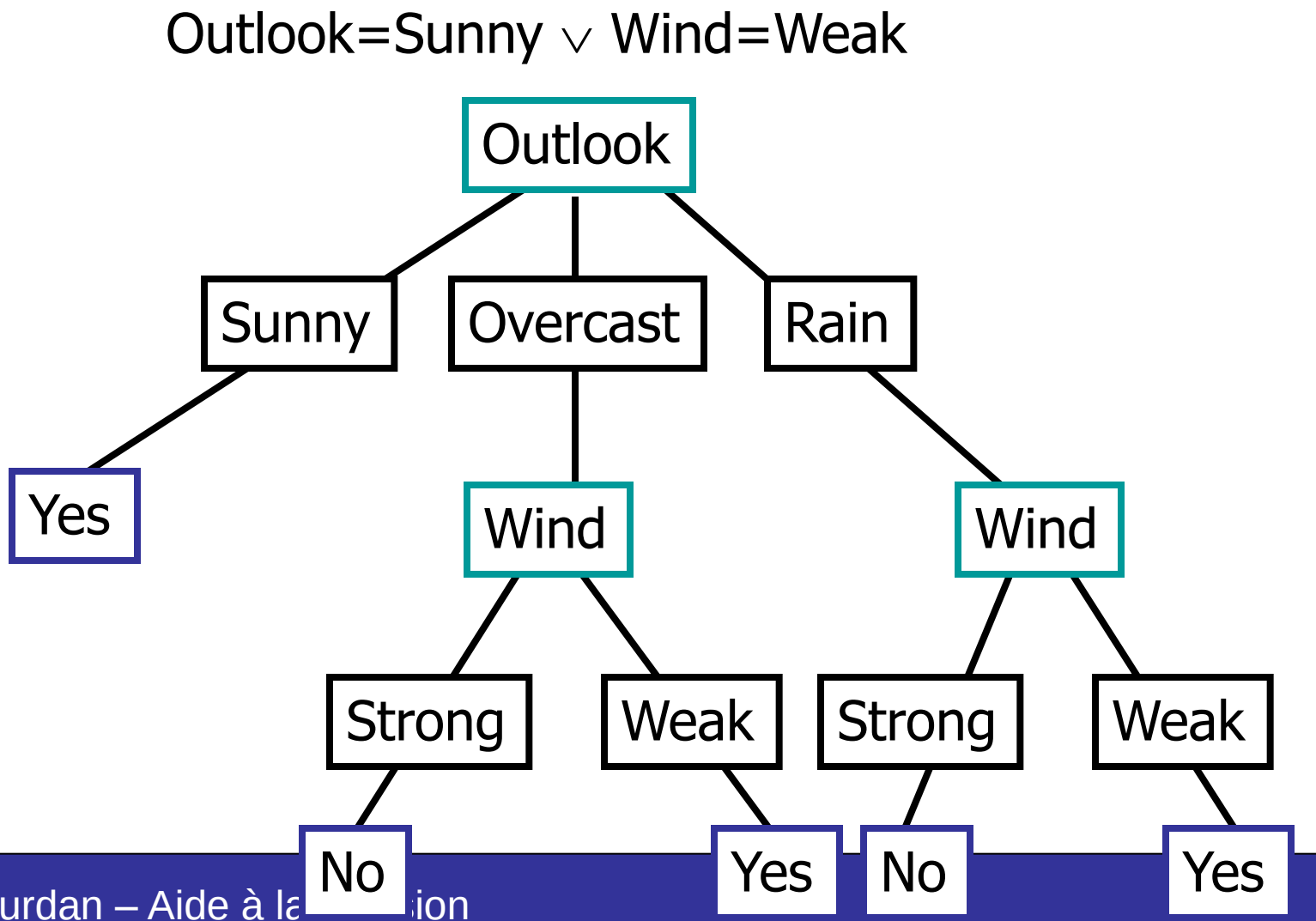


Arbre de décision et conjonction

Outlook=Sunny $\wedge$ Wind=Weak

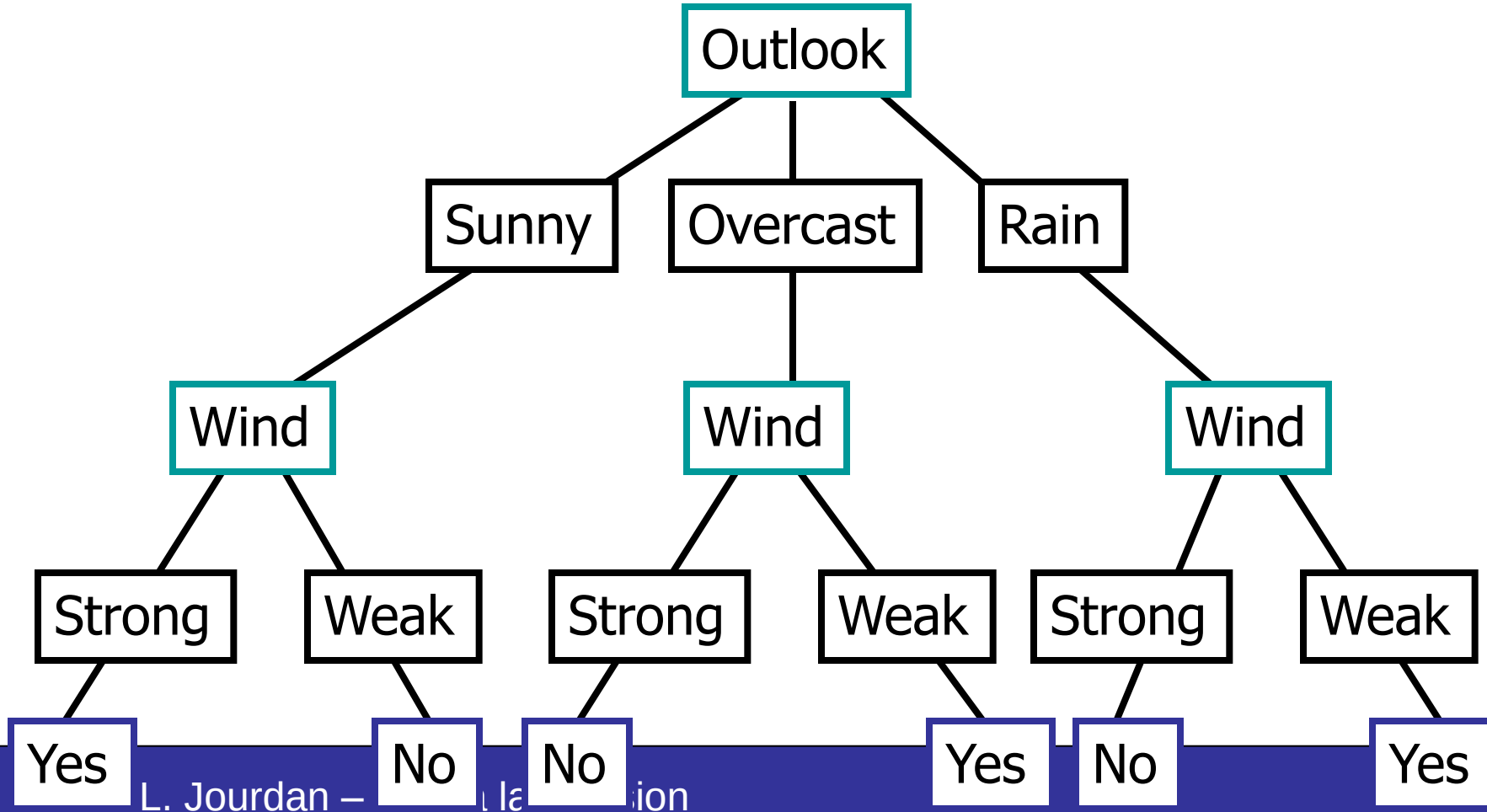


Arbre de décision et disjonction



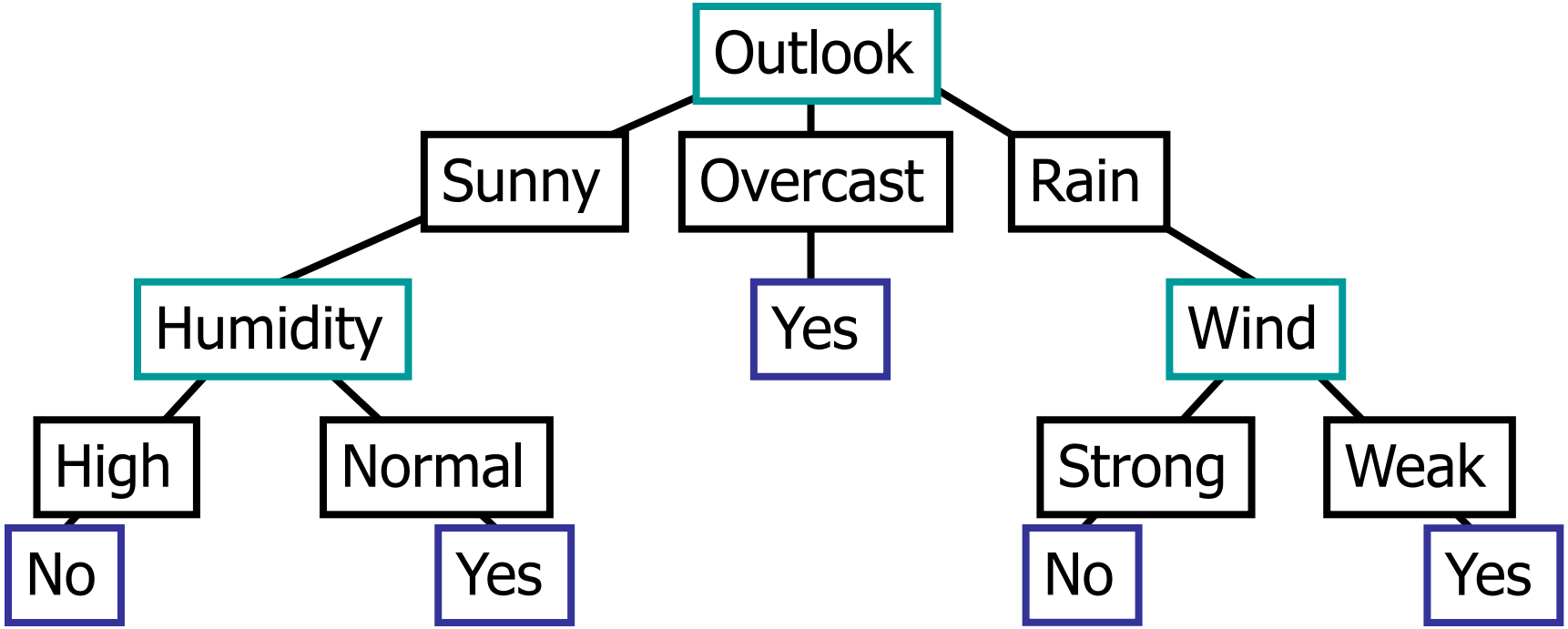
Arbre de décision et XOR

Outlook=Sunny XOR Wind=Weak



Arbre de décision et conjonction

- arbre de décision représente des disjonctions de conjonctions



(Outlook=Sunny $\wedge$ Humidity=Normal)
$\vee$ (Outlook=Overcast)
$\vee$ (Outlook=Rain $\wedge$ Wind=Weak)

Algorithmes pour les arbres de décision

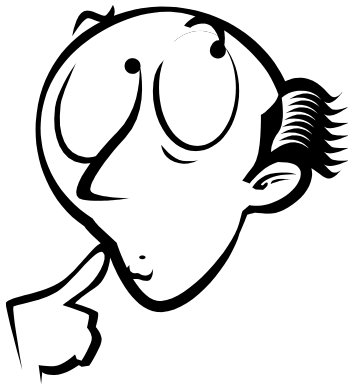
Algorithme de base

- Construction récursive d'un arbre de manière "diviser-pour-régner" descendante
- Attributs considérés énumératifs
- Glouton (piégé par les optima locaux)

Plusieurs variantes : ID3, C4.5, CART, CHAID

- Différence principale : mesure de sélection d'un attribut – critère de branchement (split)
- Ex : CART : 2 partitions par nœuds

Mesures de sélection d'attributs



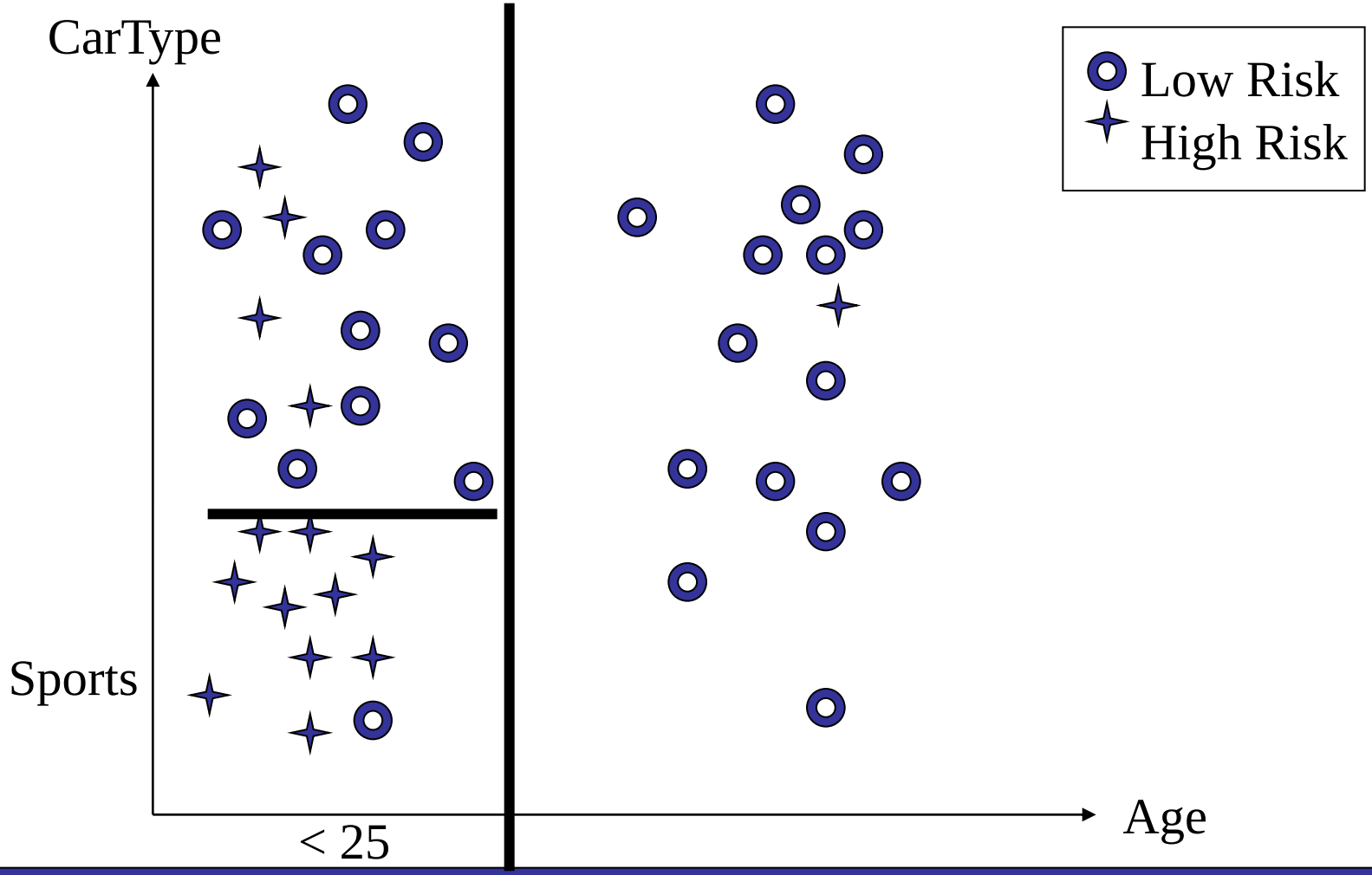
Gain d'Information (ID3, C4.5, C5)

Indice Gini (CART)

Table de contingence statistique $\chi 2$ (CHAID)

G-statistic

Bonne sélection et branchement ?



Gain d'information

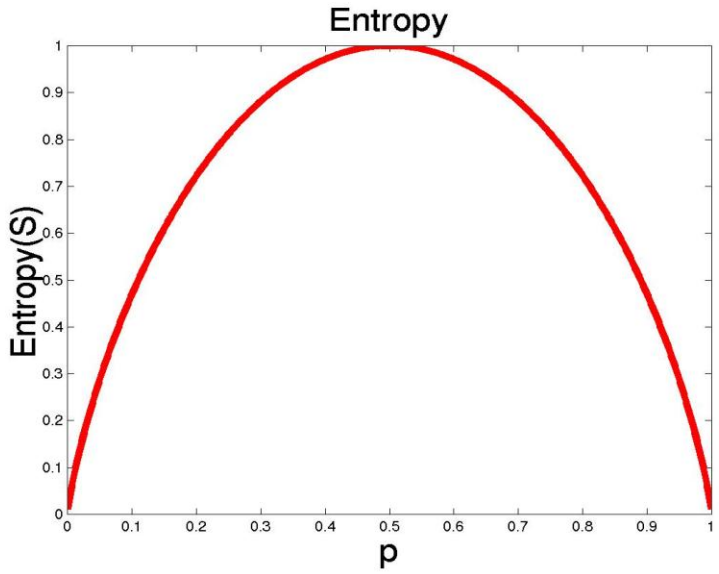
Sélectionner l'attribut avec le plus grand gain d'information

Soient P et N deux classes et S un ensemble d'instances avec p éléments de P et n éléments de N

L'information nécessaire pour déterminer si une instance prise au hasard fait partie de P ou N est (entropie) :

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

Entropie



Ex : var. booléenne X=1
Avec probabilité p

S est l'ensemble d'apprentissage

p+ est la proportion d'exemples positifs (P)

p- est la proportion d'exemples négatifs (N)

Entropie mesure l'impureté de S

- Entropie(S) = -p+ log2 p+ - p- log2 p-

Gain d'information

Soient les ensembles $\{S_1, S_2, \ldots, S_v\}$ formant une partition de l'ensemble S, en utilisant l'attribut A

Toute partition Si contient pi instances de P et ni instances de N

L'entropie, ou l'information nécessaire pour classifier les instances dans les sous-arbres Si est (entropie conditionnelle classe/attribut A):

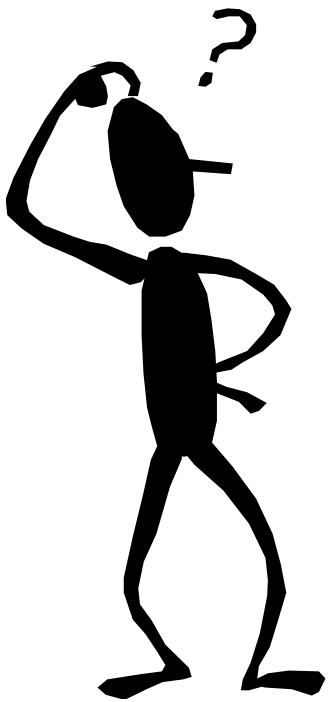
$$E(A) = \sum_{i=1}^{v} \frac{p_i + n_i}{p + n} I(p_i, n_i)$$

Le gain d'information par rapport au branchement sur A est

$$Gain(A) = I(p, n) - E(A)$$

Choisir l'attribut qui maximise le gain → besoin d'information minimal (recherche "greedy" – gloutonne)

Gain d'information - Exemple



Hypothèses :

Classe P : jouer_tennis = "oui"

Classe N : jouer_tennis = "non"

Information nécessaire pour classer un exemple donné est :

$$I(p,n) = I(9,5) = 0.940$$

Gain d'information - Exemple

Calculer l'entropie pour

l'attribut outlook :

outlook	p_i	n_i	$I(p_i, n_i)$
sunny	2	3	0,971
overcast	4	0	0
rain	3	2	0,971

On a $E(outlook) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$

Alors $Gain(outlook) = I(9,5) - E(outlook) = 0.246$

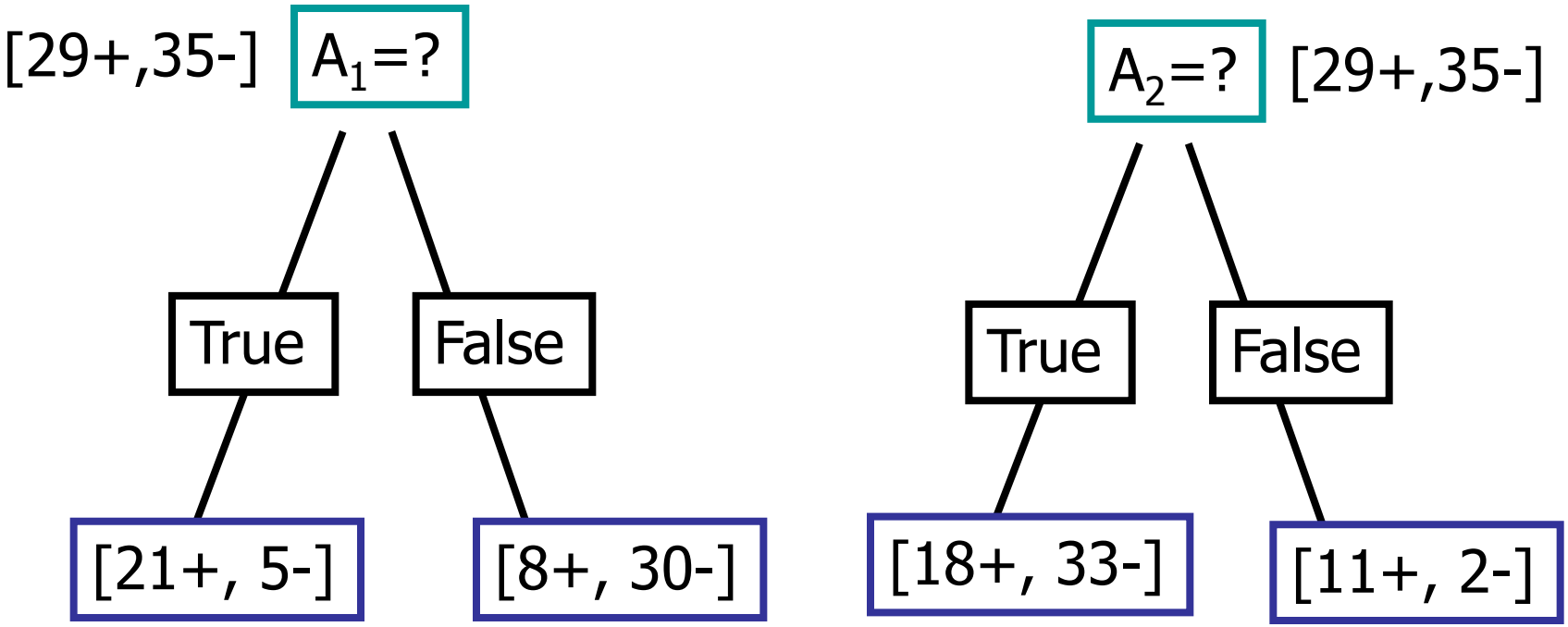
De manière similaire

$Gain(temperature) = 0.029$

$Gain(humidity) = 0.151$

$Gain(windy) = 0.048$

Quel Attribut est ”meilleur” ?

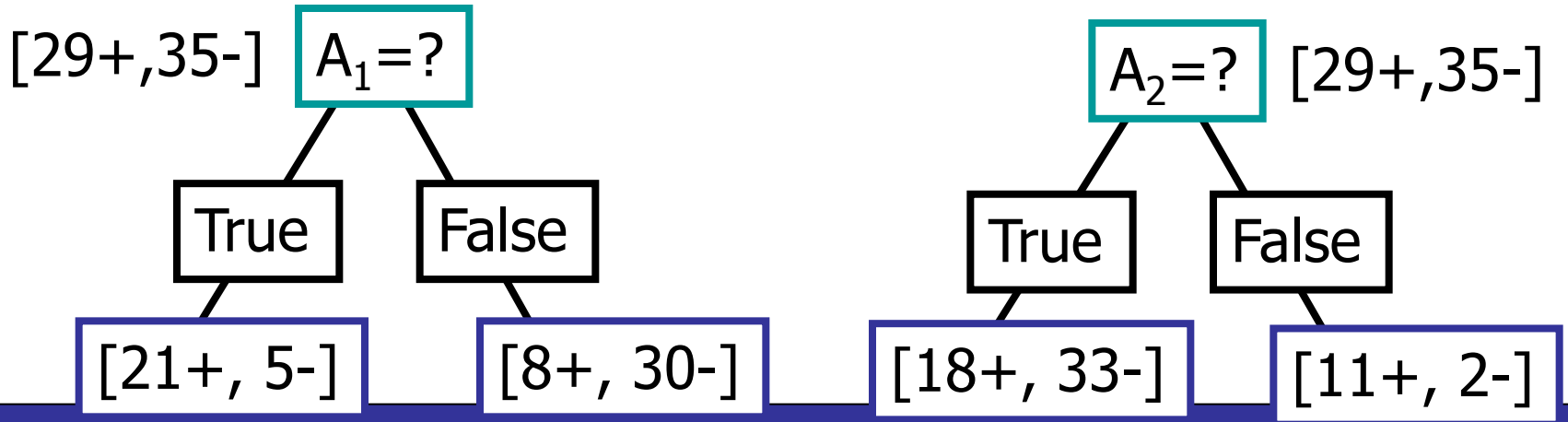


Gain d'information - Exemple

Gain(S,A) : réduction attendue de l'entropie dûe au branchement de S sur l'attribut A

$$Gain(S,A)=Entropie(S) - \sum_{v \in values(A)} |S_v|/|S| \; Entropie(S_v)$$

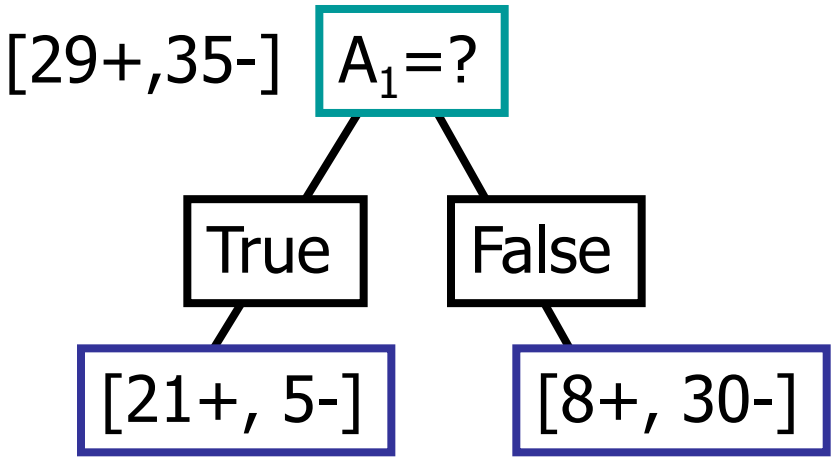
$$Entropie([29+,35-]) = -29/64 \log_2 29/64 - 35/64 \log_2 35/64 = 0.99$$

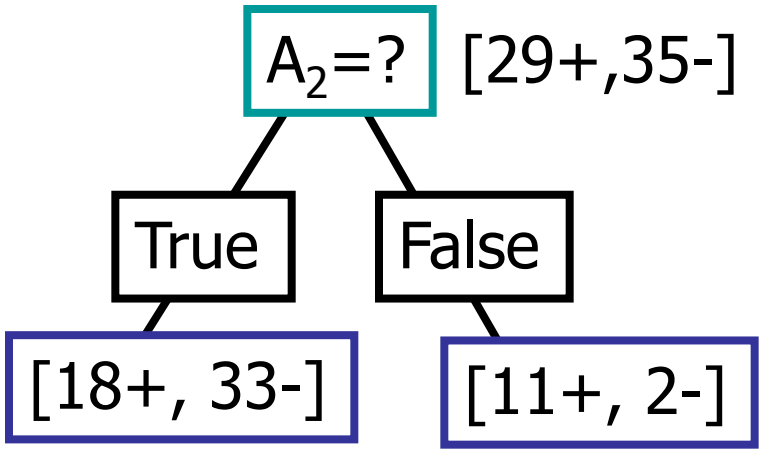


Gain d'information - Exemple

Entropie([21+,5-]) = 0.71
Entropie([8+,30-]) = 0.74
Gain(S,A_1)=Entropie(S)
-26/64*Entropie([21+,5-])
-38/64*Entropie([8+,30-])
=0.27

Entropie([18+,33-]) = 0.94
Entropie([11+,2-]) = 0.62
Gain(S,A_2)=Entropie(S)
-51/64*Entropie([18+,33-])
-13/64*Entropie([11+,2-])
=0.12

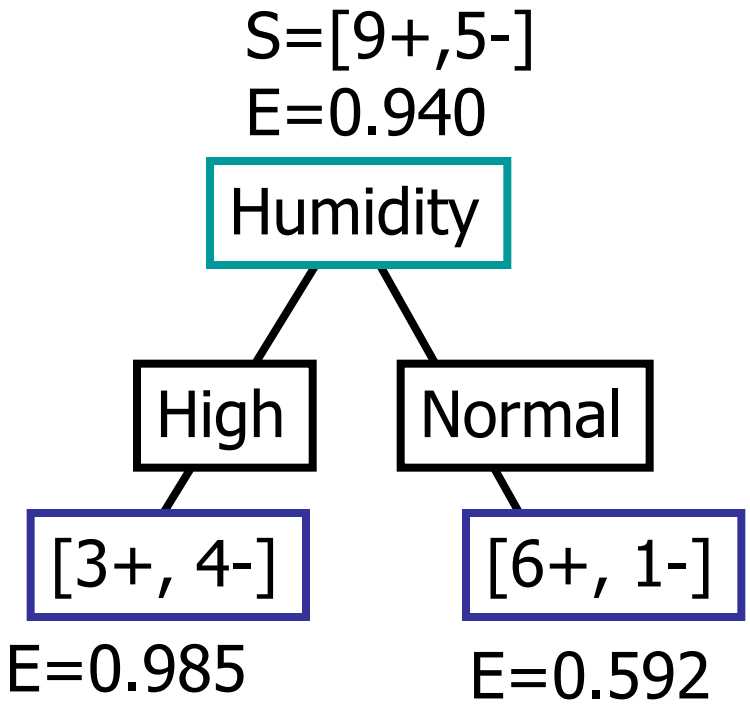


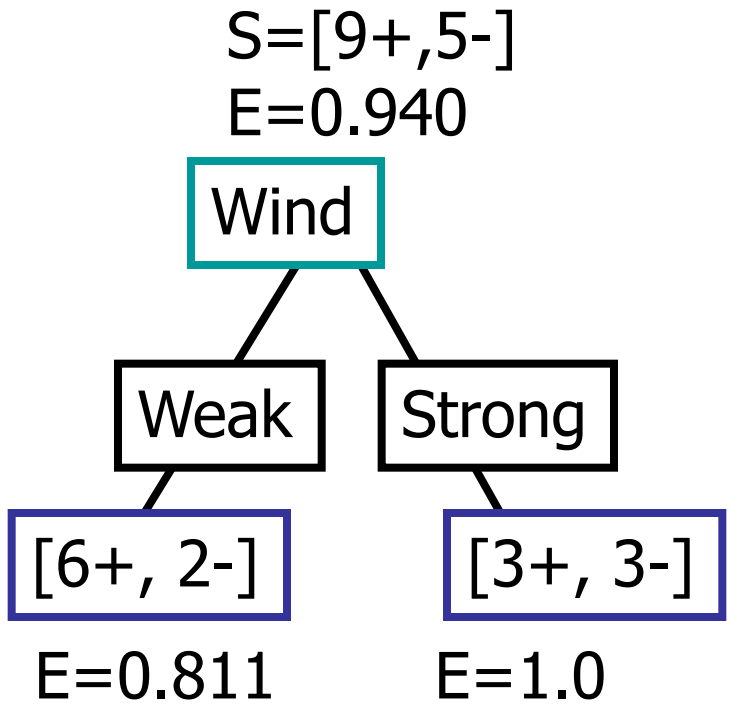


Exemple d'apprentissage

Day	Outlook	Temp.	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Weak	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D10	Rain	Mild	Normal	Strong	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Sélection de l'attribut suivant

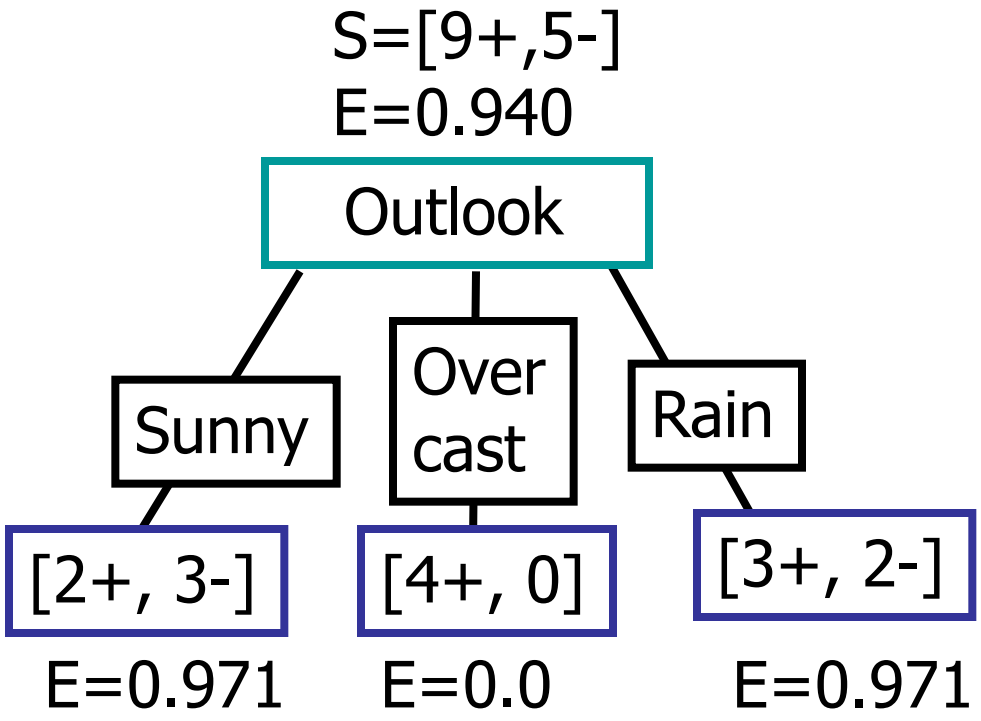




Gain(S,Humidity)
=0.940-(7/14)*0.985
– (7/14)*0.592
=0.151

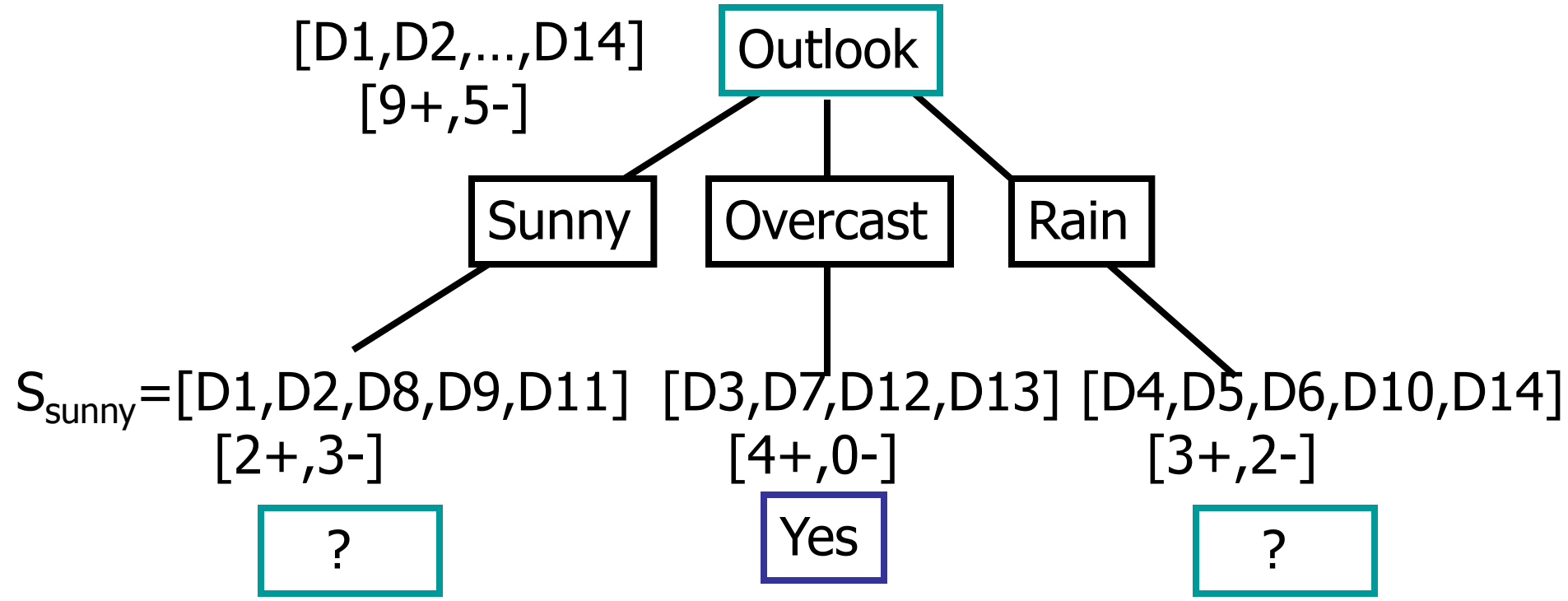
Gain(S,Wind)
=0.940-(8/14)*0.811
– (6/14)*1.0
=0.048

Sélection de l'attribut suivant



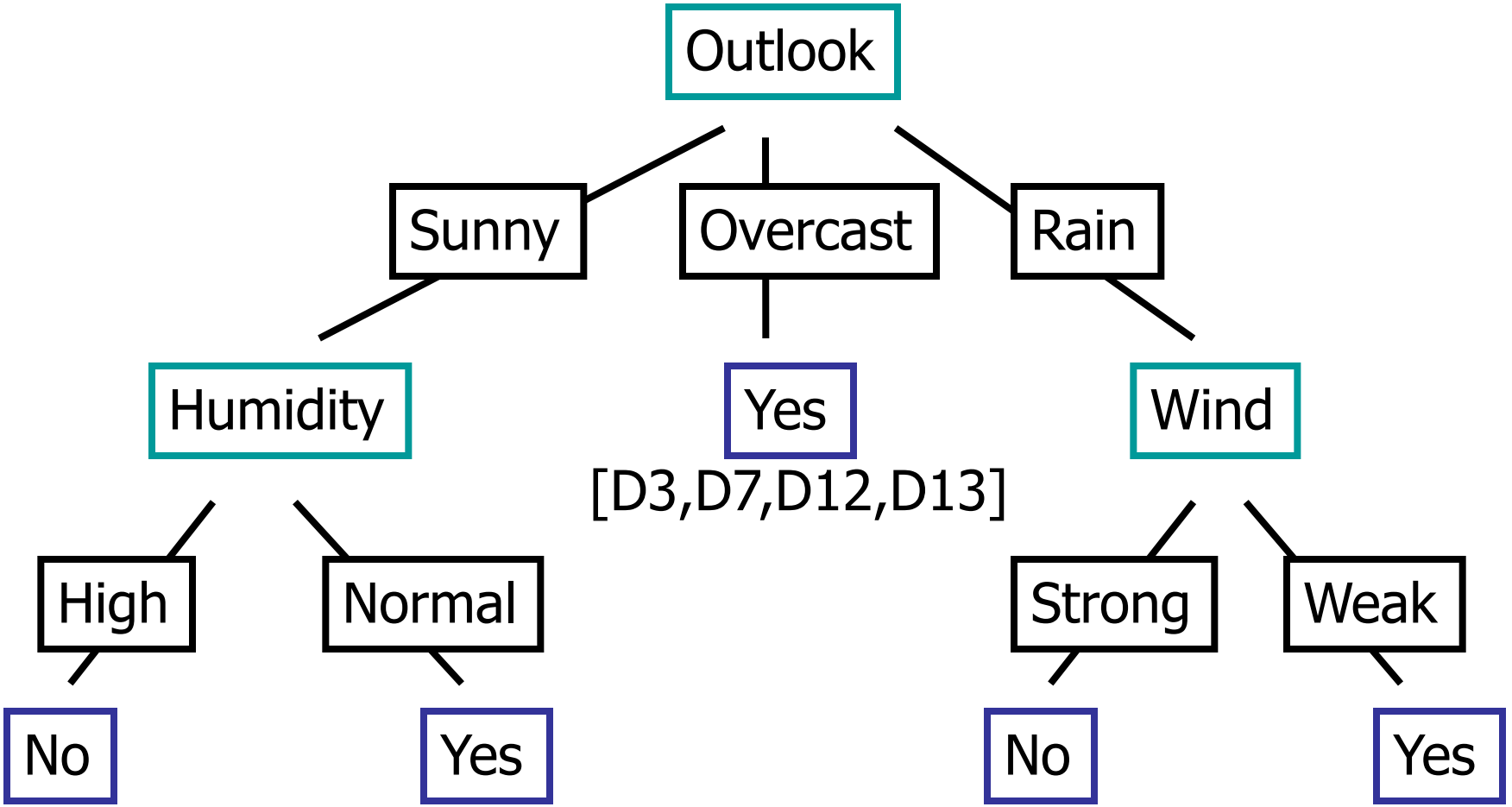
Gain(S,Outlook)
=0.940-(5/14)*0.971
-(4/14)*0.0 – (5/14)*0.0971
=0.247

Algorithme ID3



$Gain(S_{sunny}, Humidity)=0.970-(3/5)0.0 – 2/5(0.0) = 0.970$
$Gain(S_{sunny}, Temp.)=0.970-(2/5)0.0 –2/5(1.0)-(1/5)0.0 = 0.570$
$Gain(S_{sunny}, Wind)=0.970= -(2/5)1.0 – 3/5(0.918) = 0.019$

Algorithme ID3



Indice Gini

Utiliser l'indice Gini pour un partitionnement pur

$$Gini(S) = 1 - \sum_{i=1}^{c} p_i^2$$

$$Gini(S_1, S_2) = \frac{n_1}{n} Gini(S_1) + \frac{n_2}{n} Gini(S_2)$$

pi est la fréquence relative de la classe c dans S

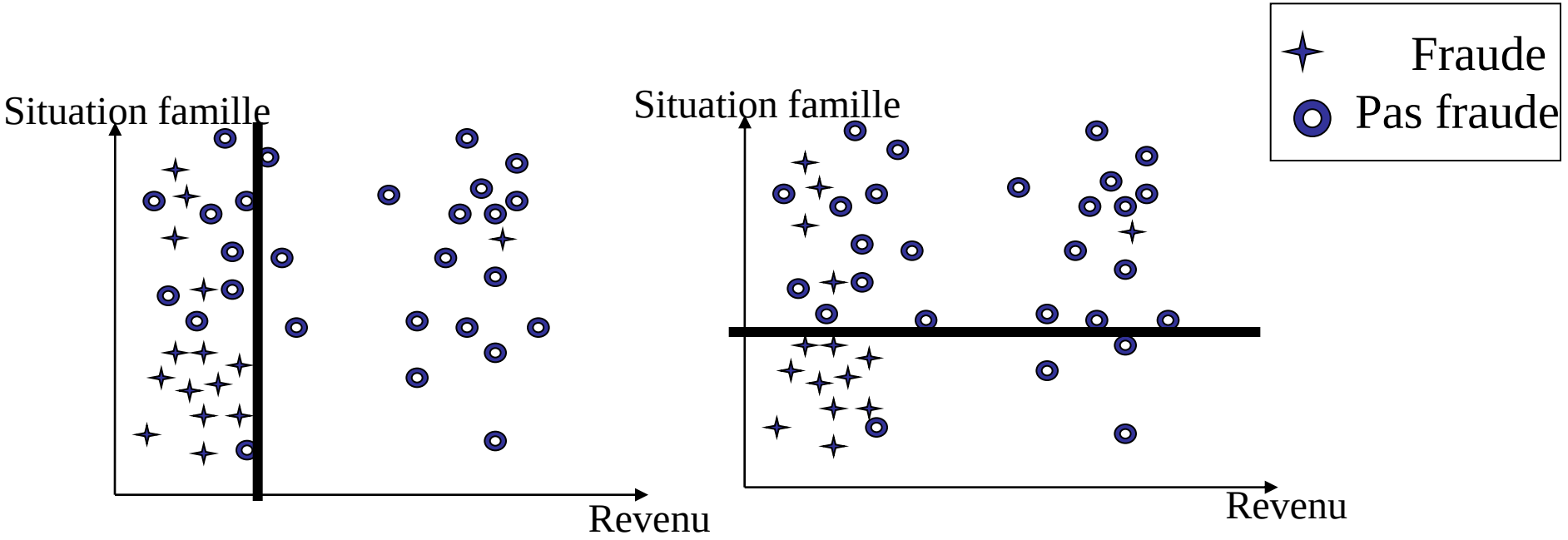
Si S est pur (classe unique), Gini(S) = 0

Gini(S1,S2) = Gini pour une partition de S en deux sous-ensembles S1 et S2 selon un test donné.

Trouver le branchement (split-point) qui minimise l'indice Gini

Nécessite seulement les distributions de classes

Indice Gini - Exemple



Calcul de Gini nécessite une Matrice de dénombrement

	Non	Oui
<80K	14	9
>80K	1	18

Gini(split) = **0.31**

	Non	Oui
M	5	23
F	10	4

Gini(split) = 0.34

Attributs énumératifs – indice GINI

	CarType	
	{Sports, Luxury}	{Family}
C1	3	1
C2	2	4
Gini	0.400	

- Pour chaque valeur distincte, calculer le nombre d'instances de chaque classe
- Utiliser la matrice de dénombrement pour la prise de décision

Partage en plusieurs classes

	CarType		
	Family	Sports	Luxury
C1	1	2	1
C2	4	1	1
Gini	0.393		

Partage en deux "classes"
(trouver la meilleure partition de valeurs)

	CarType	
	{Sports}	{Family, Luxury}
C1	2	2
C2	1	5
Gini	0.419	

Attributs numériques – indice GINI

calcul efficace : pour chaque attribut,

- Trier les instances selon la valeur de l'attribut
- Entre chaque valeur de cette liste : un test possible (split)
- Evaluation de Gini pour chacun des test
- Choisir le split qui minimise l'indice gini

Fraude	No	No	No	Yes	Yes	Yes	No	No	No	No
	Revenu imposable									
Valeurs triées →	60	70	75	85	90	95	100	120	125	220
Positions Split →	55	65	72	80	87	92	97	110	122	172	230
	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >
Yes	0 3	0 3	0 3	0 3	1 2	2 1	3 0	3 0	3 0	3 0	3 0
No	0 7	1 6	2 5	3 4	3 4	3 4	3 4	4 3	5 2	6 1	7 0
Gini	0.420	0.400	0.375	0.343	0.417	0.400	***0.300***	0.343	0.375	0.400	0.420

Algorithme CART

Indice de Gini

$$I = 1 - \sum_{i}^{n} f_i^{\,2}$$

- N = nombre de classes à prédire
- Fi = fréquence de la classe i dans le nœud

Plus l'indice de Gini est bas, plus le nœud est pure

Algorithme CART

Problèmes des arbres trop étoffés

- Complexité de l'arbre, trop de règles
- Trop spécifique aux données d'apprentissage
  - Règles non reproductibles (« surapprentissage »)
- Trop peu d'individus dans les feuilles (aucune signification réelle)
  - minimum conseillé : 20-30 individus

Solution ➔ Élagage

Algorithme CART

Processus d'élagage de CART

- Création de l'arbre maximum
  - Toutes les feuilles des extrémités sont pures
- Élagages successifs de l'arbre
- Retient l'arbre élagué pour lequel le taux d'erreur estimé mesuré sur un échantillon test est le plus bas possible

Avantages

Résultats explicites

- Arbre
- Règles de décisions simples
- Modèle facilement programmable pour affecter de nouveaux individus

Peu de perturbation des individus extrêmes

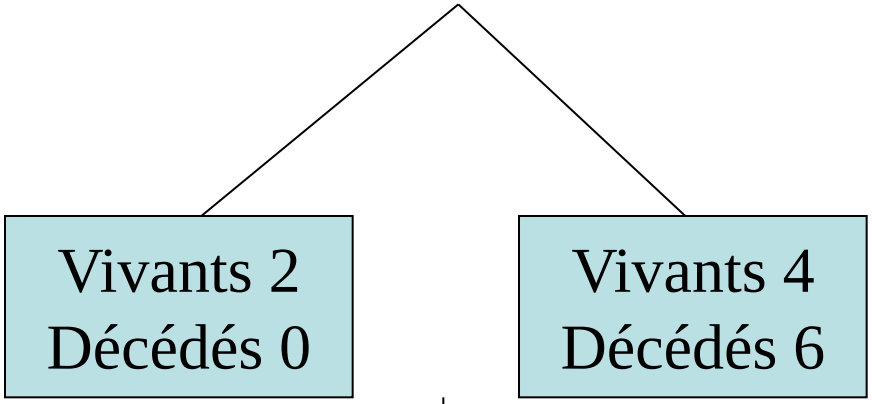
- Isolés dans des petites feuilles

Peu sensible au bruit des variables non discriminantes

- Non introduites dans le modèle

Algorithme CART

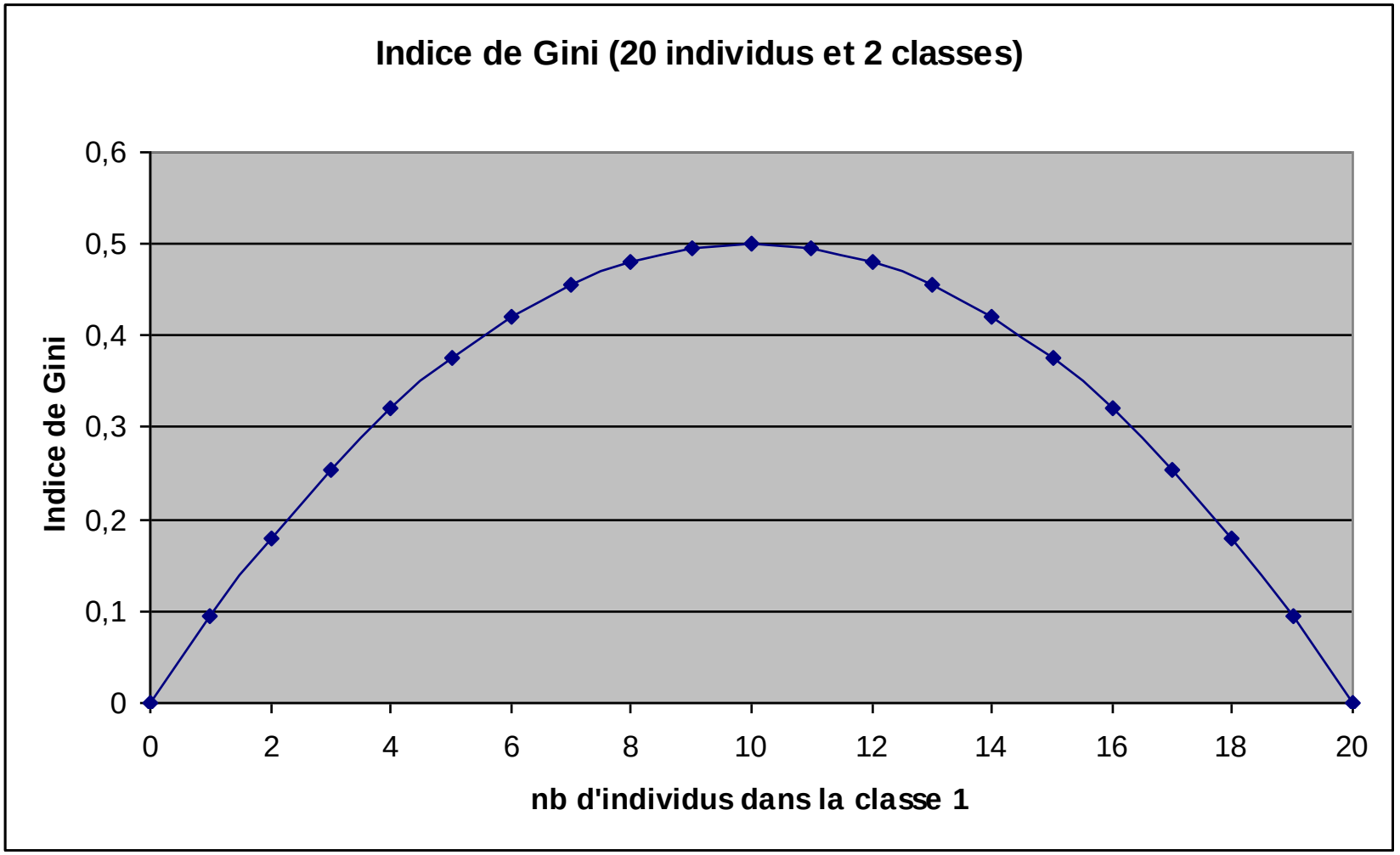
Exemple :



$$I = 1 - \left[\left(\frac{2}{2} \right)^2 + \left(\frac{0}{2} \right)^2 \right] = 0$$

$$I = 1 - \left[\left(\frac{4}{10} \right)^2 + \left(\frac{6}{10} \right)^2 \right] = 1 - 0{,}52 = 0.48$$

Algorithme CART



Algorithme CART

Ainsi,

- En séparant 1 nœud en 2 nœuds fils on cherche la plus grande hausse de la pureté
- La variable la plus discriminante doit maximiser :

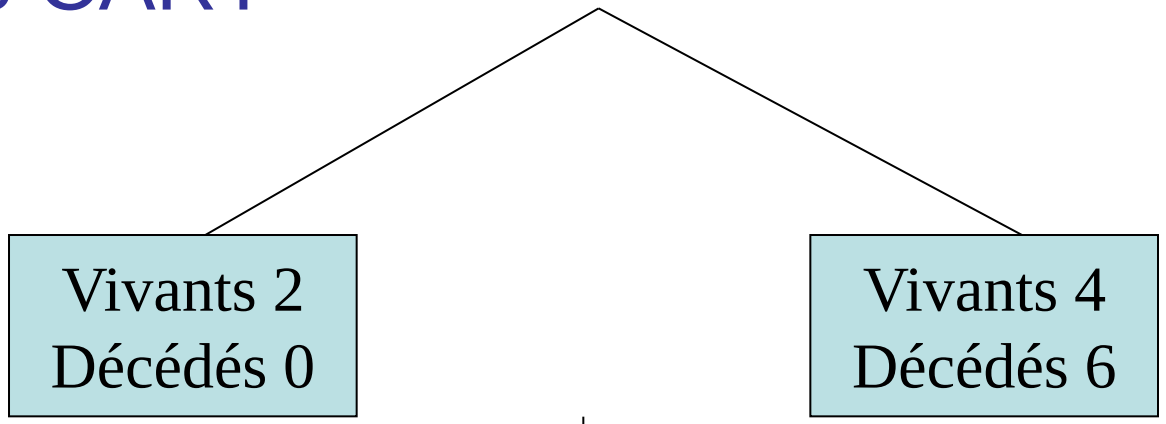
IG(avant sep.)-[IG(fils1)+IG(fils2)]

Algorithme CART

Répartition des individus dans les nœuds

- Quand l'arbre est construit : critères de division connus
- On affecte chaque individu selon les règles obtenues ➔ remplissage des feuilles
  - Pour chaque feuille : plusieurs classes C
    - Pc = Proportion d'individus de la feuille appartenant à la classe c
  - On affecte à la feuille la classe pour laquelle Pc est la plus grande

Algorithme CART



P(vivants)=1

P(décédés)=0

➔ Feuille « vivants »

Taux d'erreur feuille= 0

P(vivants)=0,4

P(décédés)=0,6

➔ Feuille « Décédés »

Taux d'erreur feuille= 0,4

Taux d'erreur global de l'arbre = somme pondérée des taux d'erreur des feuilles

Pondération = proba qu'un individu soit dans la feuille (= taille de la feuille)

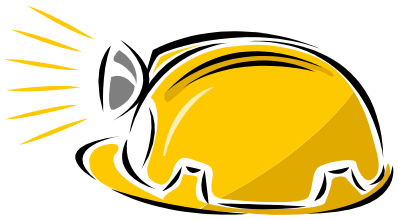
Algorithme CART

Problèmes des arbres trop étoffés

- Complexité de l'arbre, trop de règles
- Trop spécifique aux données d'apprentissage
  - Règles non reproductibles (« surapprentissage »)
- Trop peu d'individus dans les feuilles (aucune signification réelle)
  - minimum conseillé : 20-30 individus

Solution ➔ Élagage

Méthodes à base d'arbres de décision



CART (BFO'80 - Classification and regression trees, variables numériques, Gini, Elagage ascendant)

C5 (Quinlan'93 - dernière version ID3 et C4.5, attributs d'arité quelconque, entropie et gain d'information)

SLIQ (EDBT'96 — Mehta et al. IBM)

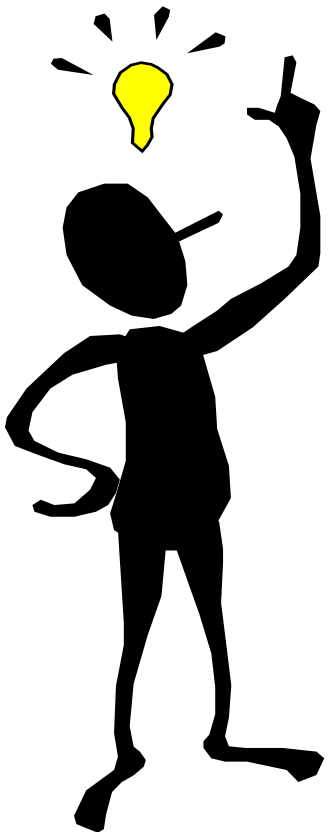
SPRINT (VLDB'96—J. Shafer et al. IBM)

PUBLIC (VLDB'98 — Rastogi & Shim)

RainForest (VLDB'98 — Gehrke, Ramakrishnan & Ganti)

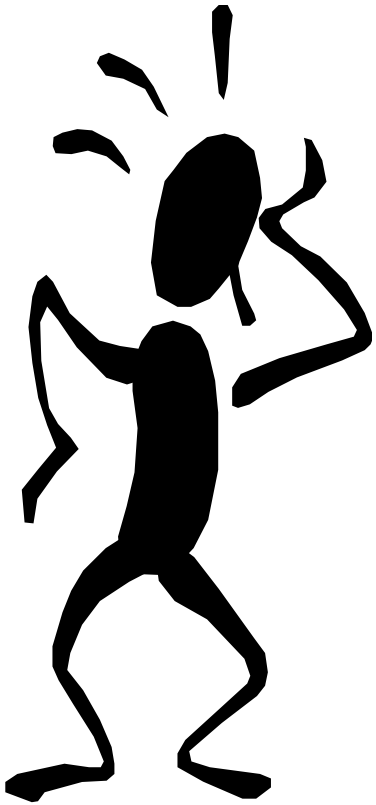
CHAID (Chi-square Automation Interaction Detection – variables discrètes)

Arbres de décision - Avantages



- Compréhensible pour tout utilisateur (lisibilité du résultat – règles - arbre)
- Justification de la classification d'une instance (racine → feuille)
- Tout type de données
- Robuste au bruit et aux valeurs manquantes
- Attributs apparaissent dans l'ordre de pertinence → tâche de pré-traitement (sélection d'attributs)
- Classification rapide (parcours d'un chemin dans un arbre)
- Outils disponibles dans la plupart des environnements de data mining

Arbres de décision - Inconvénients



Sensibles au nombre de classes : performances se dégradent

Evolutivité dans le temps : si les données évoluent dans le temps, il est nécessaire de relancer la phase d'apprentissage

Construction du modèle plus ou moins coûteuse

Classification bayésienne : Pourquoi ? (1)

Apprentissage probabiliste :

- calcul explicite de probabilités sur des hypothèses
- Approche pratique pour certains types de problèmes d'apprentissage

Incrémental :

- Chaque instance d'apprentissage peut de façon incrémentale augmenter/diminuer la probabilité qu'une hypothèse est correcte
- Des connaissances a priori peuvent être combinées avec les données observées.

Classification bayésienne : Pourquoi ? (2)

Prédiction Probabiliste :

- Prédit des hypothèses multiples, pondérées par leurs probabilités.

Référence en terme d'évaluation :

- Même si les méthodes bayésiennes sont coûteuses en temps d'exécution, elles peuvent fournir des solutions optimales à partir desquelles les autres méthodes peuvent être évaluées.

Classification bayésienne

Le problème de classification peut être formulé en utilisant les probabilités a-posteriori :

- P(C|X) = probabilité que le tuple (instance)
- X=<x1,…,xk> est dans la classe C

Par exemple

- P(classe=N | outlook=sunny,windy=true,…)

Idée : affecter à une instance X la classe C telle que P(C|X) est maximale

Estimation des probabilités a-posteriori

Théorème de Bayes :

- $P(C|X) = P(X|C) \cdot P(C) / P(X)$

$P(X)$ est une constante pour toutes les classes

$P(C)$ = fréquence relative des instances de la classe C

C tel que $P(C|X)$ est maximal =
C tel que $P(X|C) \cdot P(C)$ est maximal

Problème : calculer $P(X|C)$ est non faisable !

Classification bayésienne naive

Hypothèse Naïve : indépendance des attributs

$P(x_1,\ldots,x_k|C) = P(x_1|C)\cdot\ldots\cdot P(x_k|C)$

$P(x_i|C)$ est estimée comme la fréquence relative des instances possédant la valeur xi (i-ème attribut) dans la classe C

Non coûteux à calculer dans les deux cas

Classification bayésienne – Exemple (1)

Estimation de $P(x_i|C)$

P(*p*) = 9/14
P(*n*) = 5/14

Outlook	
P(sunny | p) = 2/9	P(sunny | n) = 3/5
P(overcast | p) = 4/9	P(overcast | n) = 0
P(rain | p) = 3/9	P(rain | n) = 2/5
Temperature	
P(hot | p) = 2/9	P(hot | n) = 2/5
P(mild | p) = 4/9	P(mild | n) = 2/5
P(cool | p) = 3/9	P(cool | n) = 1/5

Humidity	
P(high | p) = 3/9	P(high | n) = 4/5
P(normal | p) = 6/9	P(normal | n) = 1/5
Windy	
P(true | p) = 3/9	P(true | n) = 3/5
P(false | p) = 6/9	P(false | n) = 2/5

Classification bayésienne – Exemple (1)

Classification de X :

- Une instance inconnue X = <rain, hot, high, false>
- P(X|p)·P(p) =
  P(rain|p)·P(hot|p)·P(high|p)·P(false|p)·P(p) = 3/9· 2/9· 3/9· 6/9· 9/14 = 0.010582
- P(X|n)·P(n) =
  P(rain|n)·P(hot|n)·P(high|n)·P(false|n)·P(n) = 2/5· 2/5· 4/5· 2/5· 5/14 = 0.018286
- Instance X est classifiée dans la classe n (ne pas jouer)

Classification bayésienne – l'hypothèse d'indépendance

… fait que le calcul est possible

… trouve un modèle de classification optimal si hypothèse satisfaite

… mais est rarement satisfaite en pratique, étant donné que les attributs (variables) sont souvent corrélés.

Pour éliminer cette limitation :

- Réseaux bayésiens, qui combinent le raisonnement bayésien et la relation causale entre attributs
- Arbres de décision, qui traitent un attribut à la fois, considérant les attributs les plus importants en premier

Autres méthodes de classification

Réseaux bayésiens

Algorithmes génétiques

Case-based reasoning

Ensembles flous

Rough set

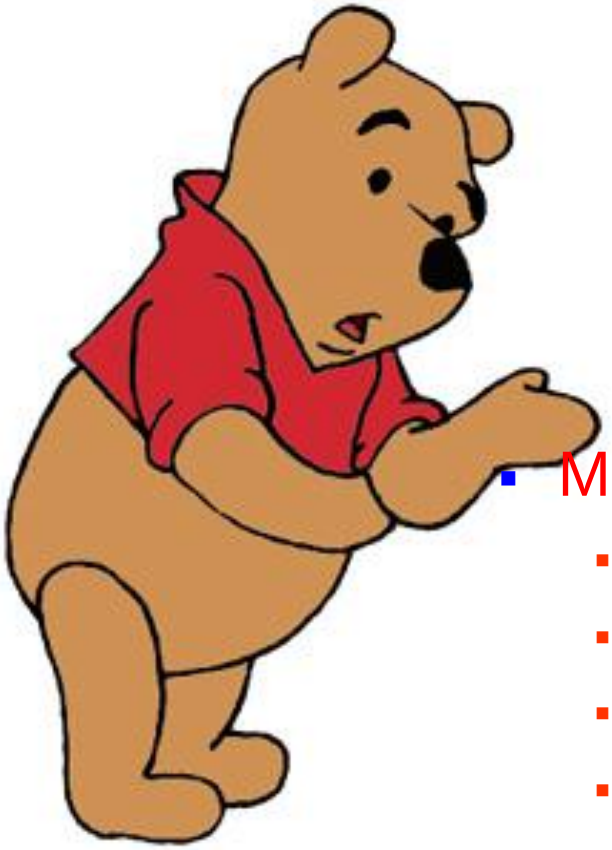
Analyse discriminante (Discriminant linéaire de Fisher, Algorithme Closest Class Mean -CCM-)

Chaînes de Markov cachées

Classification - Résumé

La classification est un problème largement étudié

La classification, avec ses nombreuses extensions, est probablement la technique la plus répandue



- Modèles
  - Arbres de décision
  - Règles d'induction
  - Modèles de régression
  - Réseaux de neurones

Facile à comprendre

Difficile à comprendre

Classification - Résumé

L'extensibilité reste une issue importante pour les applications

Directions de recherche :
classification de données non relationnels, e.x., texte, spatiales et données multimédia

Classification - Références

J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufman, 1993.

J. R. Quinlan. Induction of decision trees. Machine Learning, 1:81-106, 1986.

L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth International Group, 1984.

S. M. Weiss and C. A. Kulikowski. Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Morgan Kaufman, 1991.

D. E. Rumelhart, G. E. Hinton and R. J. Williams. Learning internal representation by error propagation. In D. E. Rumelhart and J. L. McClelland (eds.) Parallel Distributed Processing. The MIT Press, 1986

Outils pour le Data Mining

Comment Choisir un outil ?

Systèmes commerciaux de data mining possèdent peu de propriétés communes :

- Différentes méthodologies et fonctionalités de data mining
- Différents types d'ensembles de données

Pour la sélection d'un outil, on a besoin d'une analyse multi-critère des systèmes existants

Comment Choisir un outil ?

Types de donnés : relationnel, transactionnel, texte, séquences temporelles, spaciales ?

Issues systèmes

- Support systèmes d'exploitation ?
- Architecture client/serveur ?
- Fournit une interface Web et permet des données XML en entrée et/ou en sortie ?

Sources des données :

- Fichiers texte ASCII, sources de données relationnels multiples, …
- Support ODBC (OLE DB, JDBC) ?

Comment choisir un bon outil ?

Fonctionalités et méthodologies

- une vs. plusieurs fonctions de data mining
- une vs. plusieurs méthodes par fonction

Couplage avec les systèmes de gestion de base de données et les entropots de données

Outils de visualization : visualisation des données, visualisation des résultats obtenus, visualisation du processus, visualisation interactive (split attribut, …), etc.

Comment Choisir un outil ?

Extensibilité (Scalability)

- instances (Taille de la base de données)
- attributs (dimension de la base)
- Extensibilité en terme d'attributs est plus difficile à assurer que l'extensibilité en terme d'instances

Langage de requête et interface graphique (IHM)

- easy-to-use et qualité de l'interface
- data mining interactif

Exemple d'outils (1)

- Intelligent Miner d'IBM
  - Intelligent Miner for Data (IMA)
  - Intelligent Miner for Text (IMT)
  - Tâches : groupage de données, classification, recherche d'associations, etc.
- Entreprise Miner de SAS
  - SAS : longue expérience en statistiques
  - Outil «complet» pour le DM
- Darwin de Thinking Machines
  - Trois techniques : réseaux de neurones, arbres de décision et régression.
  - Client-Serveur

Exemples d'outils (2)

- MineSet de Silicon Graphics
  - Fonctionnalités interactives et graphiques
  - Techniques sous-jacentes : classification, segmentation, recherche de règles d'association.
- Outils/librairies libres
  - SIPINA
  - WEKA
- Data-Miner Software Kit (DMSK)
  - Kit de programmes : méthodes statistiques, segmentation, groupage, réseaux de neurones, etc.
  - Il existe une version en java
- etc.

SAS Entreprise Miner (1)

Société : SAS Institute Inc.

Création : Mai 1998

Plate-formes : Windows NT & Unix

Utilisation

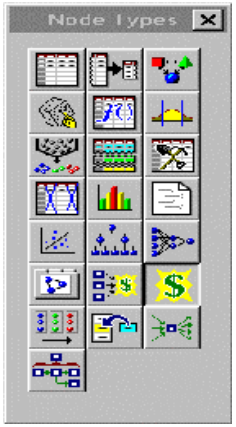
- Réduction des coûts
- Maîtrise des risques
- Fidélisation
- Prospection

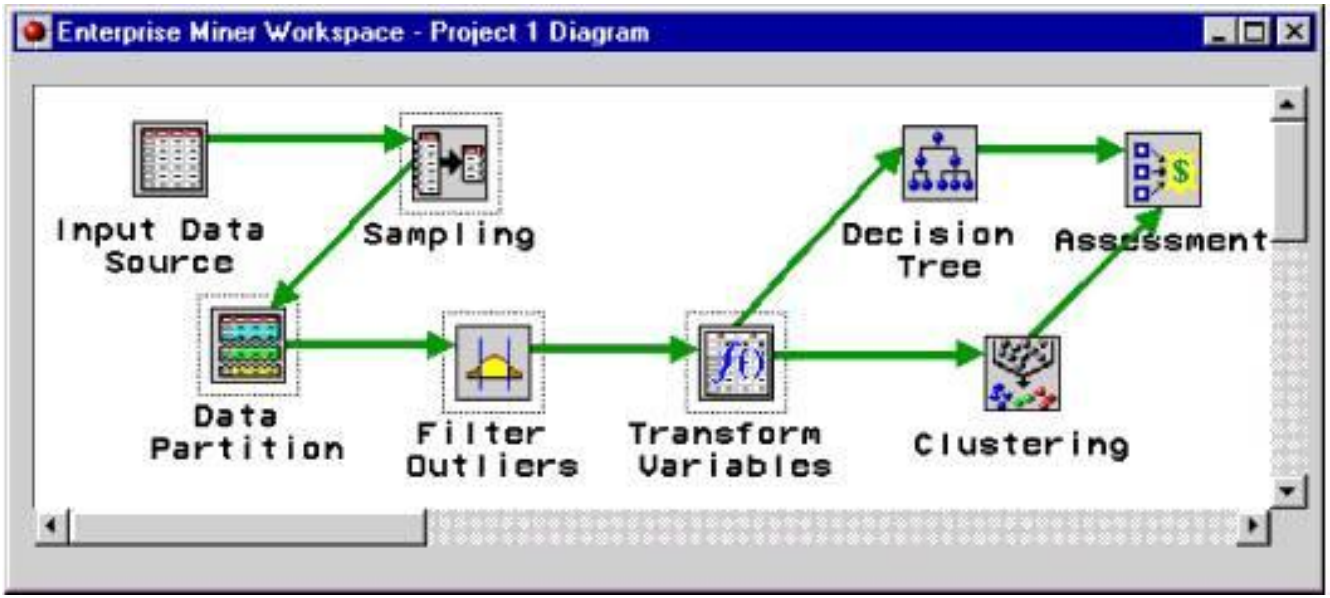
Outils de data warehouse

SAS Entreprise Miner (2)

Interface graphique (icônes)

Construction d'un diagramme





SAS Entreprise Miner (3)

Deux types d'utilisateurs

- Spécialistes en statistiques
- Spécialistes métiers (chef de projet, études…)

Techniques implémentées

- Arbres de décision
- Régression
- Réseaux de neurones

Alice (1)

Société : ISoft

Création : 1988

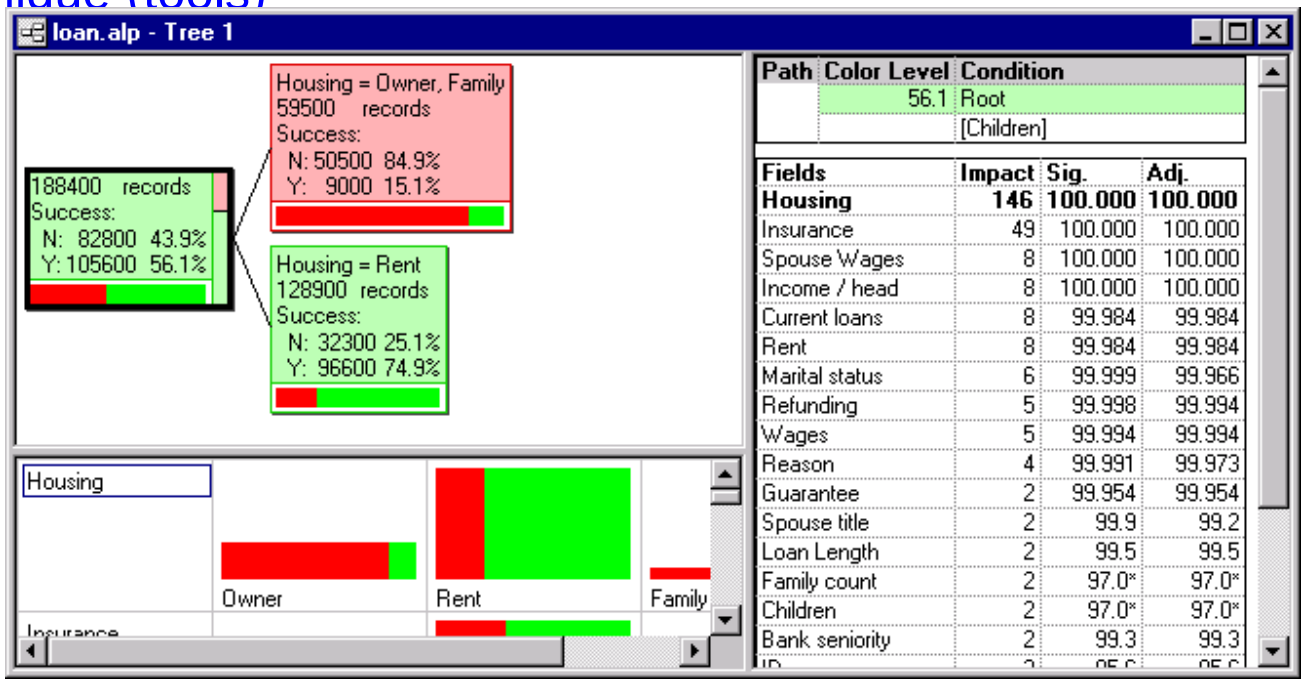
Plate-formes : Windows 95/98/NT/2000, TSE, Metaframe

Utilisation

- Marketing : études de marché, segmentation …
- Banque, Assurance : scoring, analyse de risques, détection de fraudes
- Industrie : contrôle qualité, diagnostic, segmentation, classification, construction de modèles, prédiction et simulation

Alice (2)

Interface graphique (tools)



Type d'utilisateur : responsables opérationnels

Clementine (1)

Société : ISL (*Integral Solutions Limited*)

Création : 1994

Plate-formes : Windows NT, Unix

Utilisation

- Prévision de parts de marché
- Détection de fraudes
- Segmentation de marché
- Implantation de points de vente …

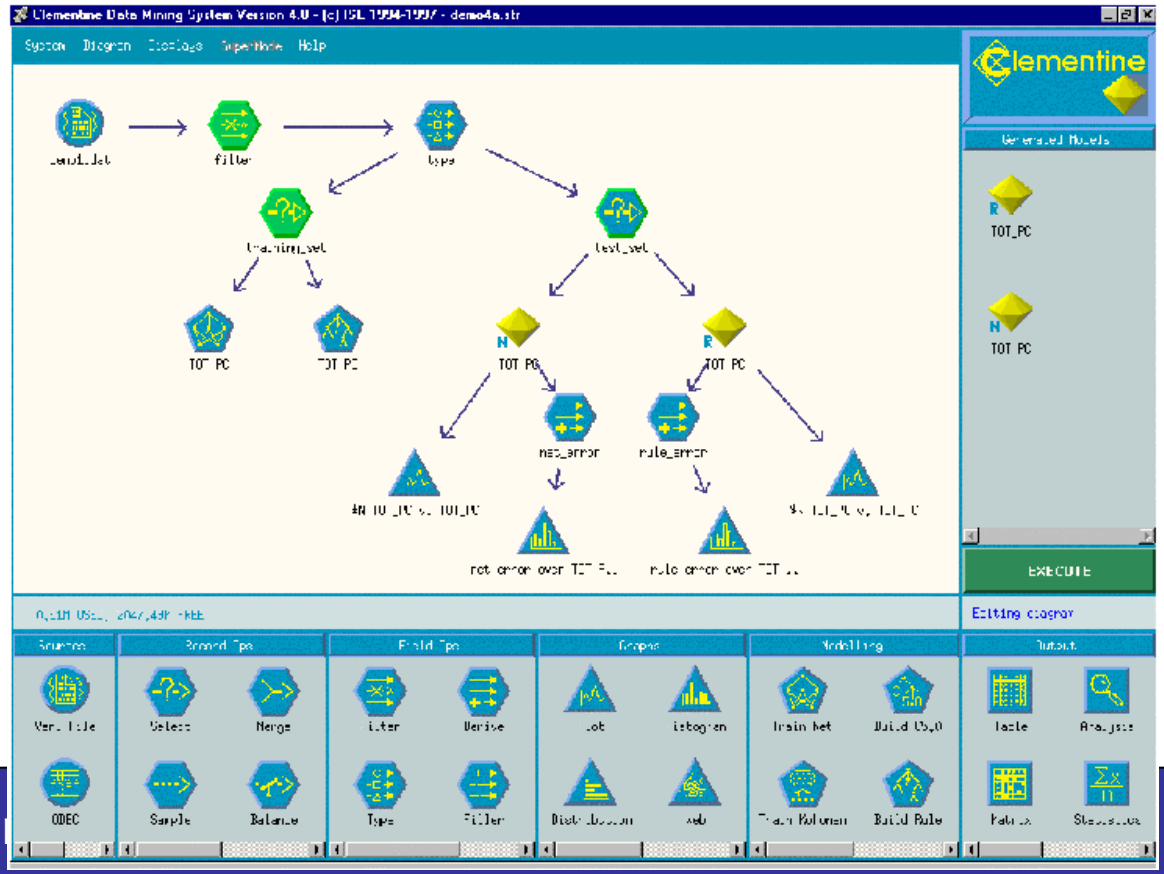
Environnement intégré : #Types d'utilisateurs

- Gens du métier (pas forcement des informaticiens)
- Développeurs / End users

Clementine (2)

Interface simple, puissante et complète
interface conviviale

⇨



Clementine (3)

Techniques :

- Arbres de décision
- Induction de règles
- Réseaux de neurones
- Méthodes statistiques

Forecast Pro (1)

Société : Business Forecast Systems

Création : 1997

Plate-formes : Windows 95, NT

Utilisation

- Tous domaines activités et secteurs
- Notamment la prévision (5 types différents)

Outil d'analyse incomparable

Le plus utilisé dans le monde

Forecast Pro (2)

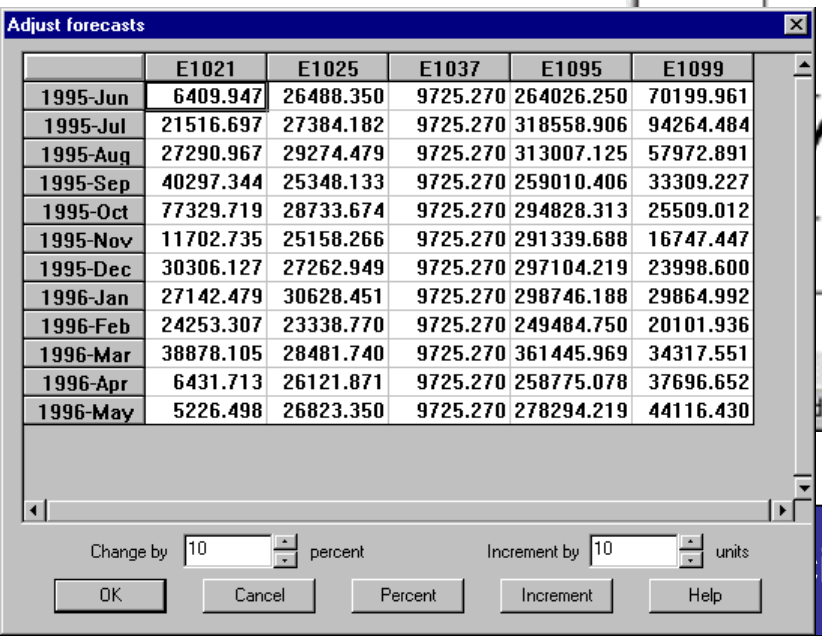
Types d'utilisateurs : PME/PMI, administrations, consultants, universitaires, chefs de projets,…

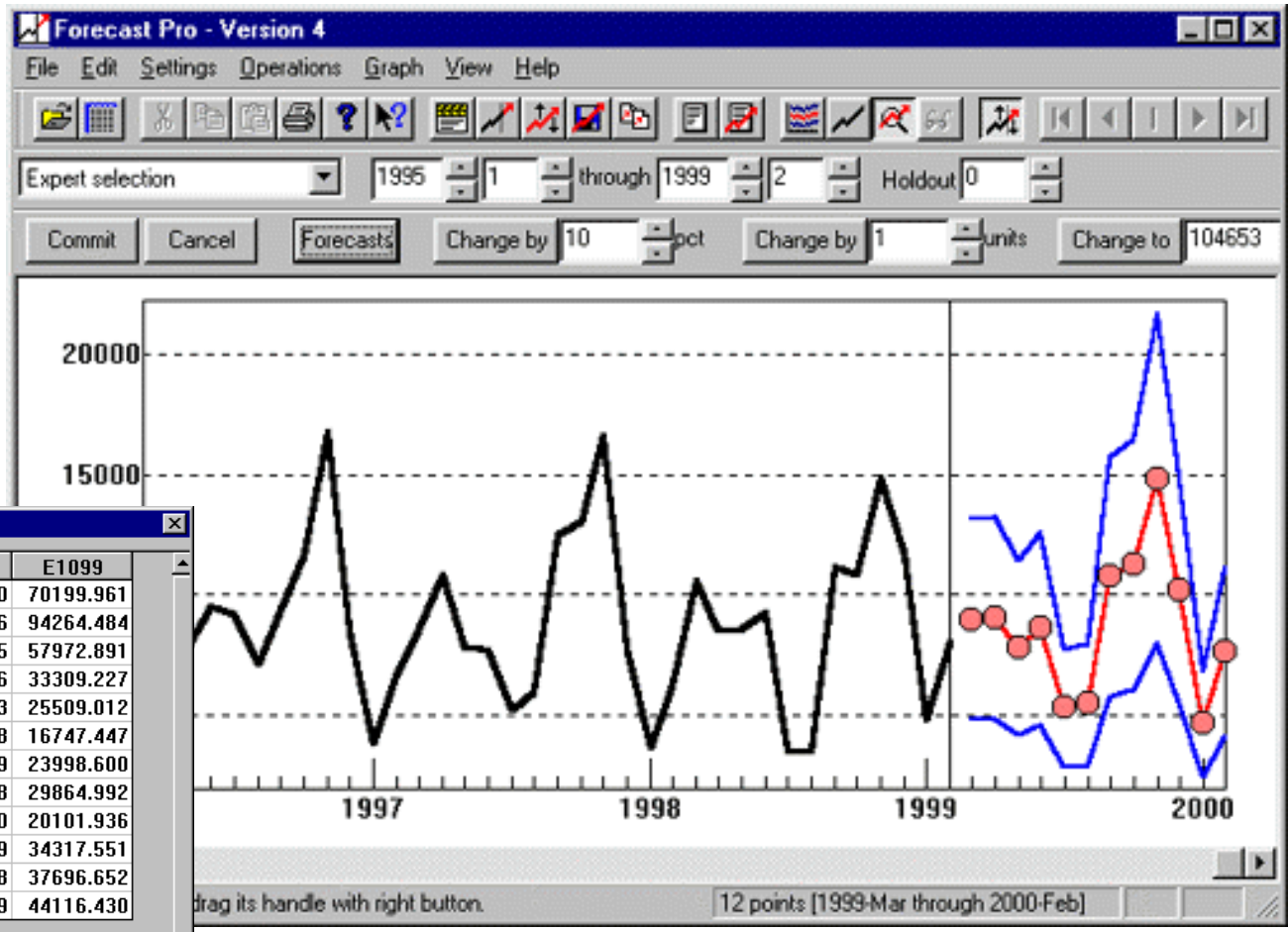
Facilité d'utilisation (connaissances en statistiques non requises)

Vaste palette de choix graphiques

- Valeurs observées, prévisions, valeurs calculées sur l'historique, intervalles de confiance, diagnostics (erreurs)

Forecast Pro (3)





ision

Intelligent Miner (1)

Société : IBM

Création : 1998

Plate-formes : AIX, OS/390, OS/400, Solaris, Windows 2000 & NT

Utilisation

- Domaines où l'aide à la décision est très importante (exemple : domaine médical)
- Analyse de textes

Fortement couplé avec DB2 (BD relationnel)

Intelligent Miner (2)

Deux versions

- Intelligent Miner for Data (IMD)
- Intelligent Miner for Text (IMT)

Types d'utilisateurs : spécialistes ou professionnels expérimentés

Parallel Intelligent Miner

Intelligent Miner (3)

L'IMD

- Sélection et codage des données à explorer
- Détermination des valeurs manquantes
- Agrégation de valeurs
- Diverses techniques pour la fouille de données
  - Règles d'association (*Apriori*), classification (*Arbres de décision, réseaux de neurones*), clustering, détection de déviation (analyse statistique & visualisation)
- Visualisation des résultats
- Algorithmes extensibles (scalability)

Intelligent Miner (4)

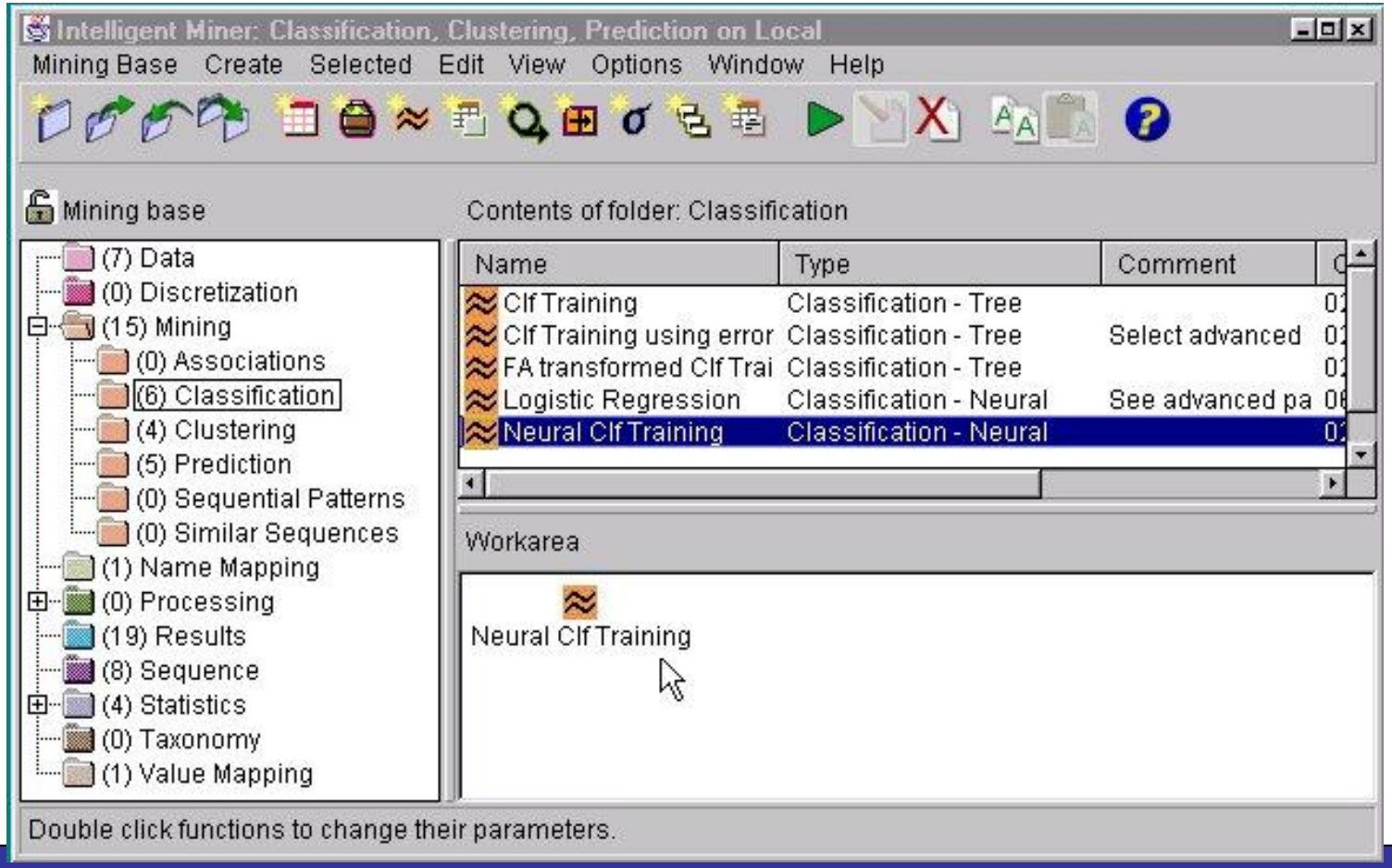
IMT = analyse de textes libres

Trois composants

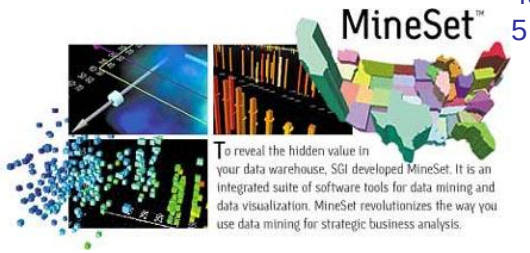
- Moteur de recherche textuel avancé *(TextMiner)*
- Outil d'accès au Web (moteur de recherche NetQuestion et un méta-moteur)
- Outil d'analyse de textes *(Text Analysis)*

L'objectif général est de faciliter la compréhension des textes

Intelligent Miner (5)



MineSet (1)



Société : SGI (*Silicon Graphics Inc*.)

Création : 1996

Plate-forme : *Silicon Graphics*

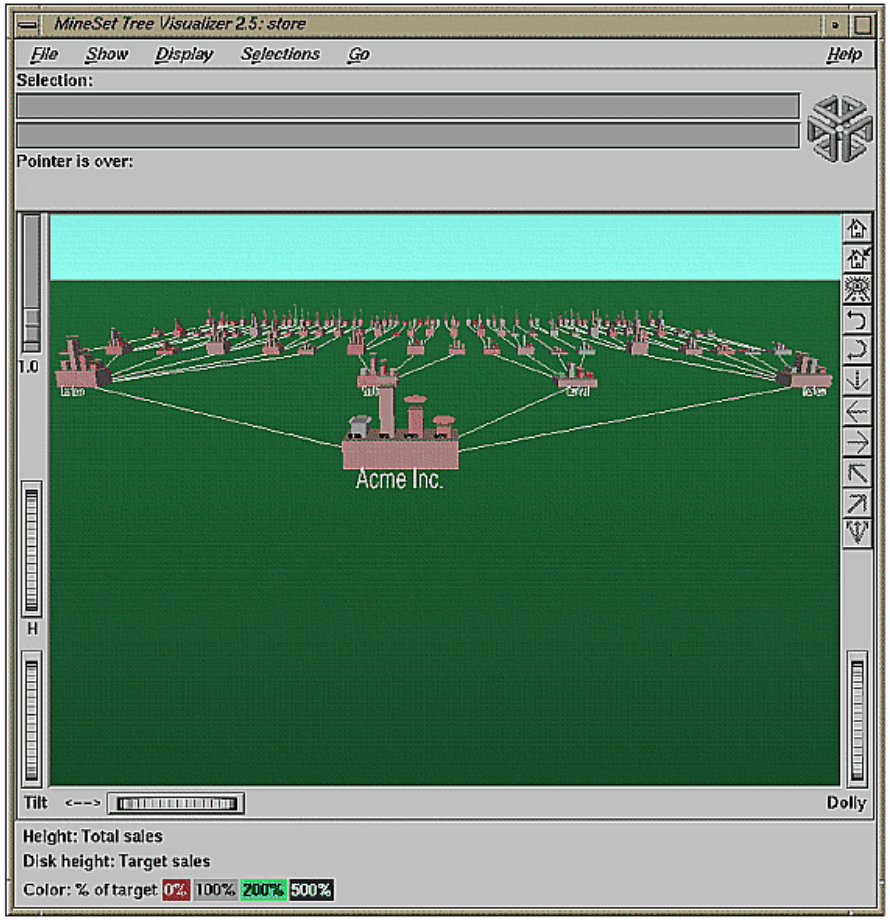
Utilisation

- Services financiers
- Prise de décisions

Algorithmes de visualisation avancés

MineSet (2)

Interface visuelle 3D



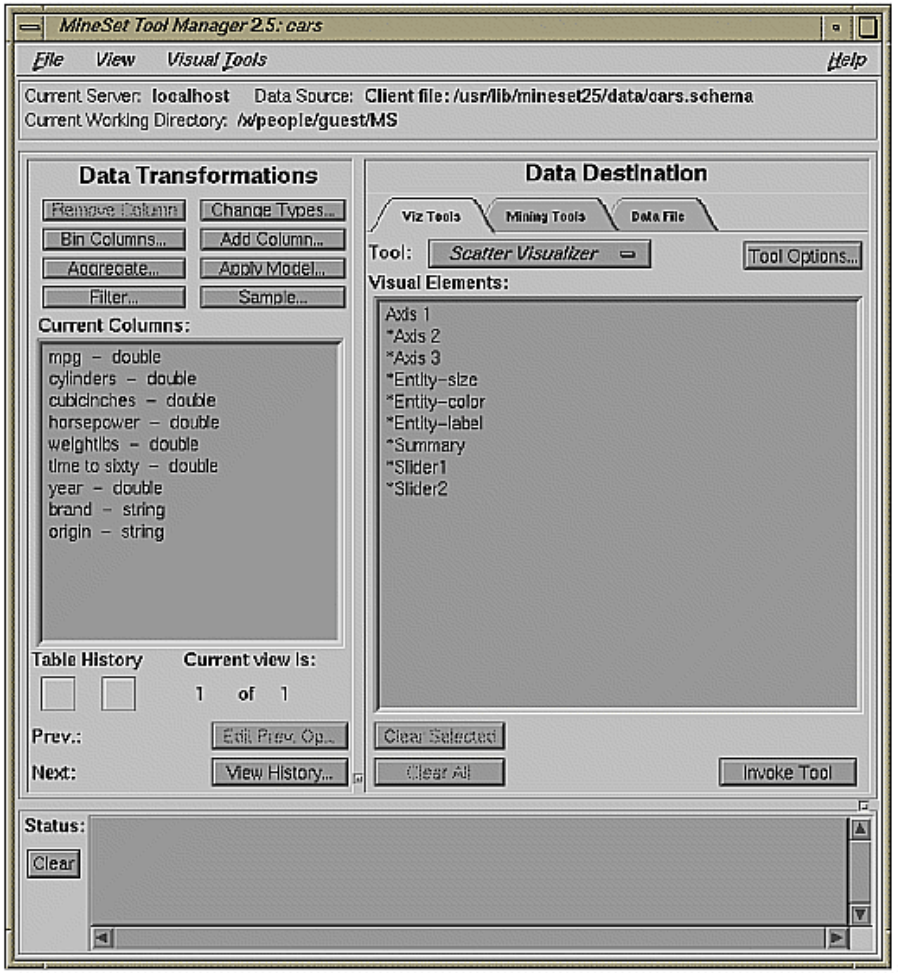
MineSet (3)

Interface graphique client/serveur

- Tool Manager (Client)
- DataMover (Server)

Utilisateurs

- Managers
- Analystes



MineSet (4)

Tâches

- Règles d'association
- Classification

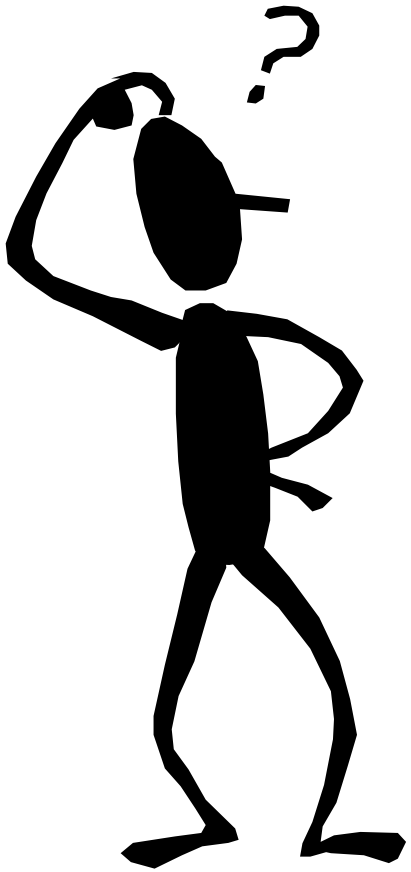
Présentation de la connaissance

- Arbre
- Statistiques
- Clusters (nuages de points)

Synthèse

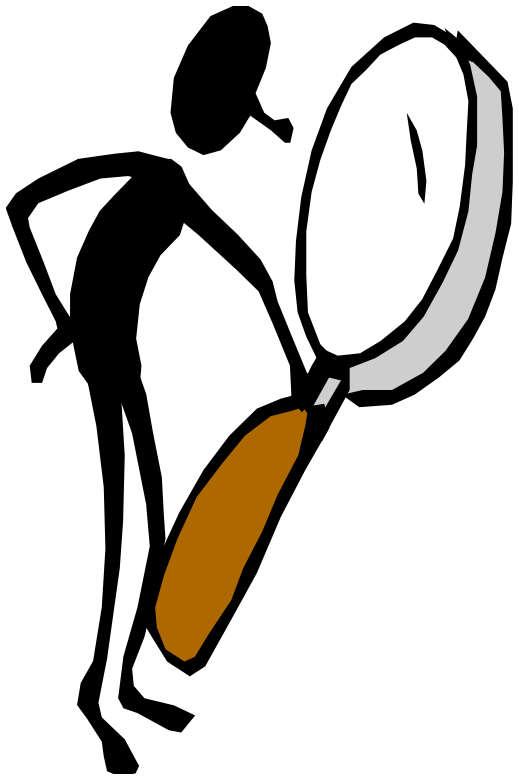
Company	Product	Link Analysis	Classification	Clustering	Statistics	Prediction	OS	Others
IBM	Intelligent Miner	3/4	3/7	2/3	6/6	4/5	1/3	3/5
ISoft	Alice / AC2	0/4	1/7	0/3	3/6	0/5	3/3	1/5
SAS Institute Inc.	SAS Enterprise Miner	0/4	3/7	2/3	6/6	3/5	2/3	0/5
Silicon Graphics Inc.	MineSet	0/4	2/7	1/3	0/6	1/5	2/3	4/5
SPSS Inc.	Clementine	3/4	3/7	1/3	3/6	2/5	2/3	2/5

Autres techniques de Data Mining



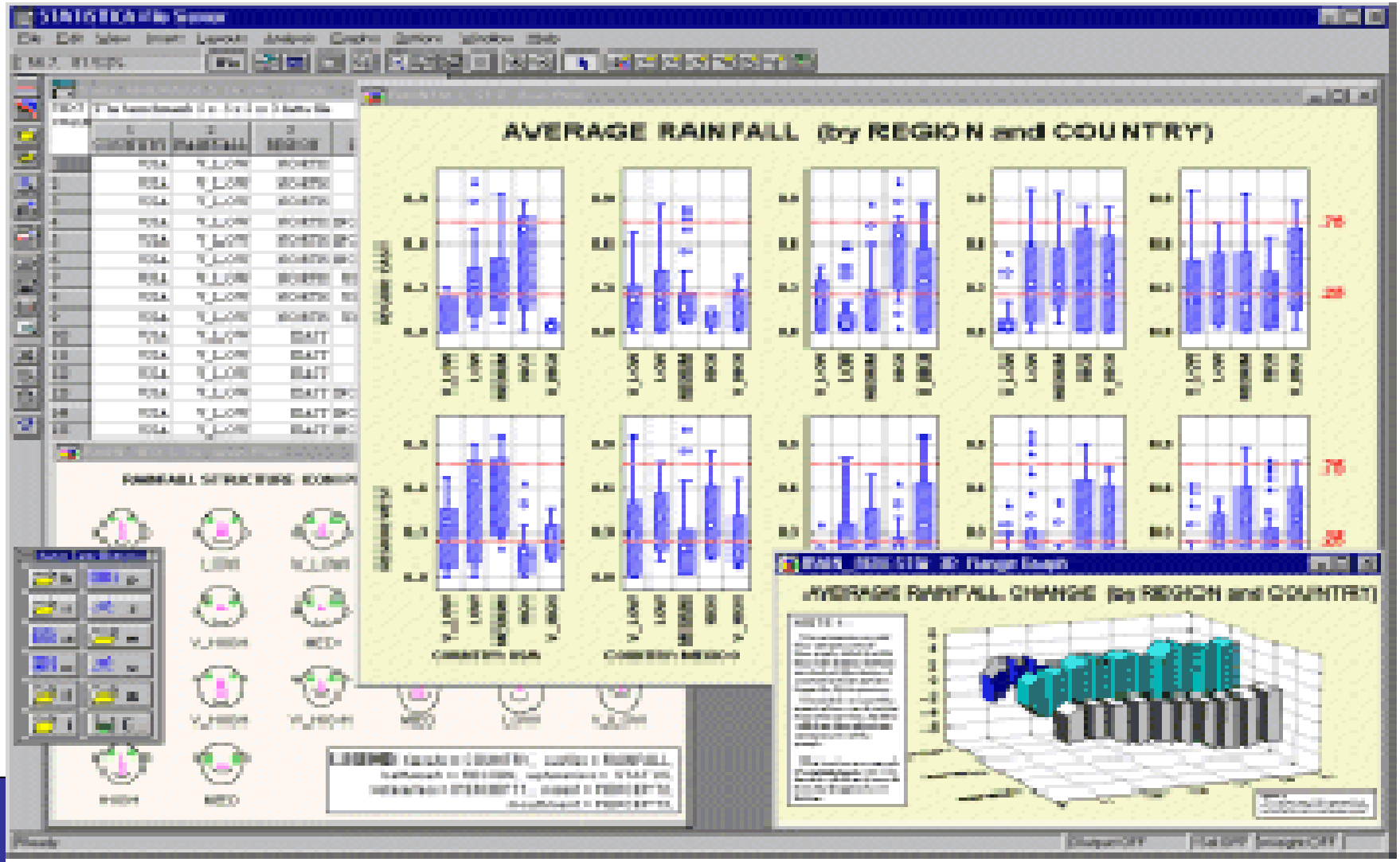
- Web mining (contenu, usage, …)
- Visual data mining (images)
- Audio data mining (son, musique)
- Data mining et requêtes d'interrogation "intelligentes"

Visualisation de données

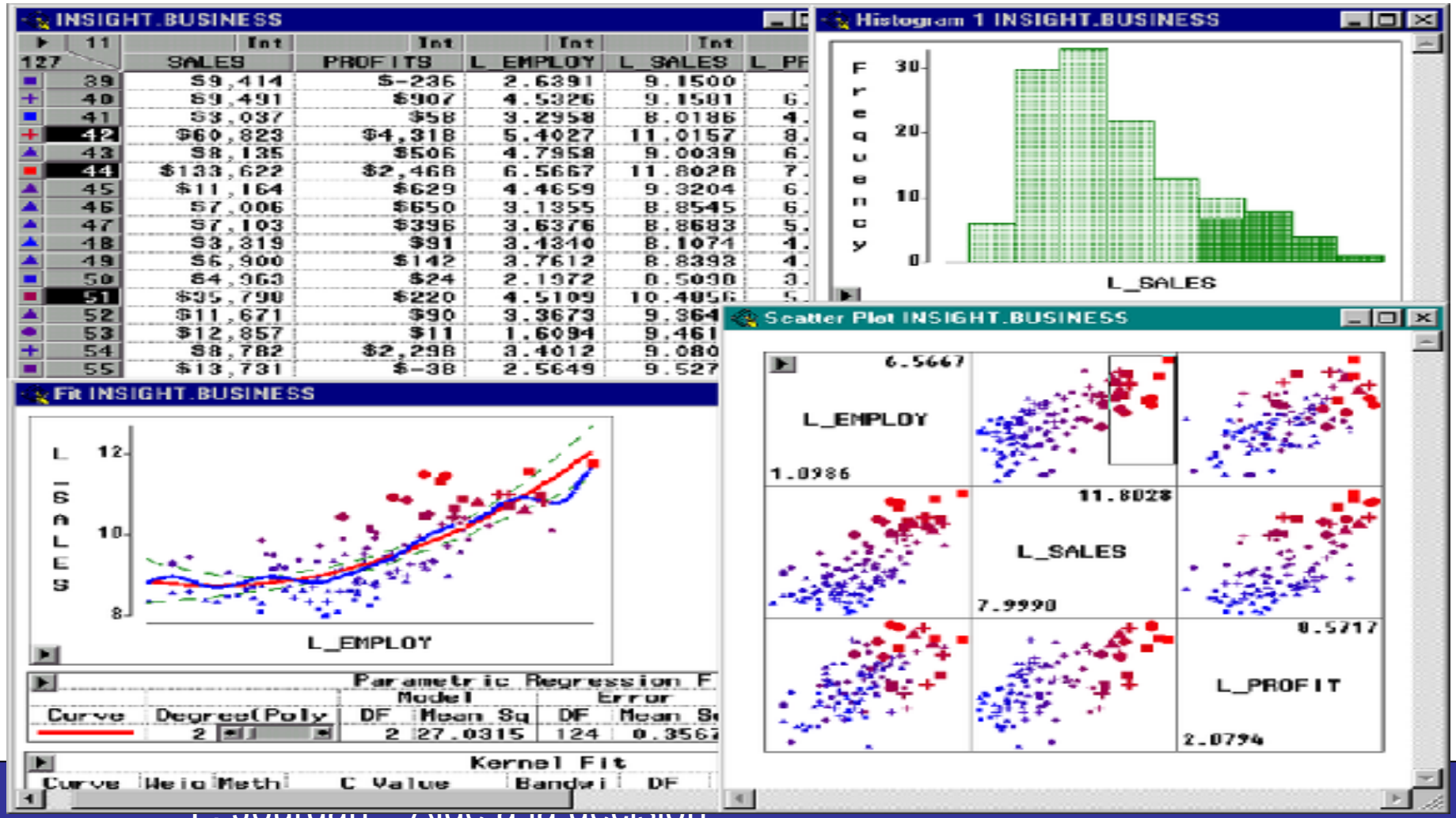


- **Données dans un base de données ou un entropot de données peuvent être visualisées :**
  - À différents niveaux de granularité ou d'abstraction
  - A l'aide de différentes combinaisons d'attributs ou dimensions
- **Résultats des outils de Data Mining peuvent être présentées sous diverses formes visuelles**

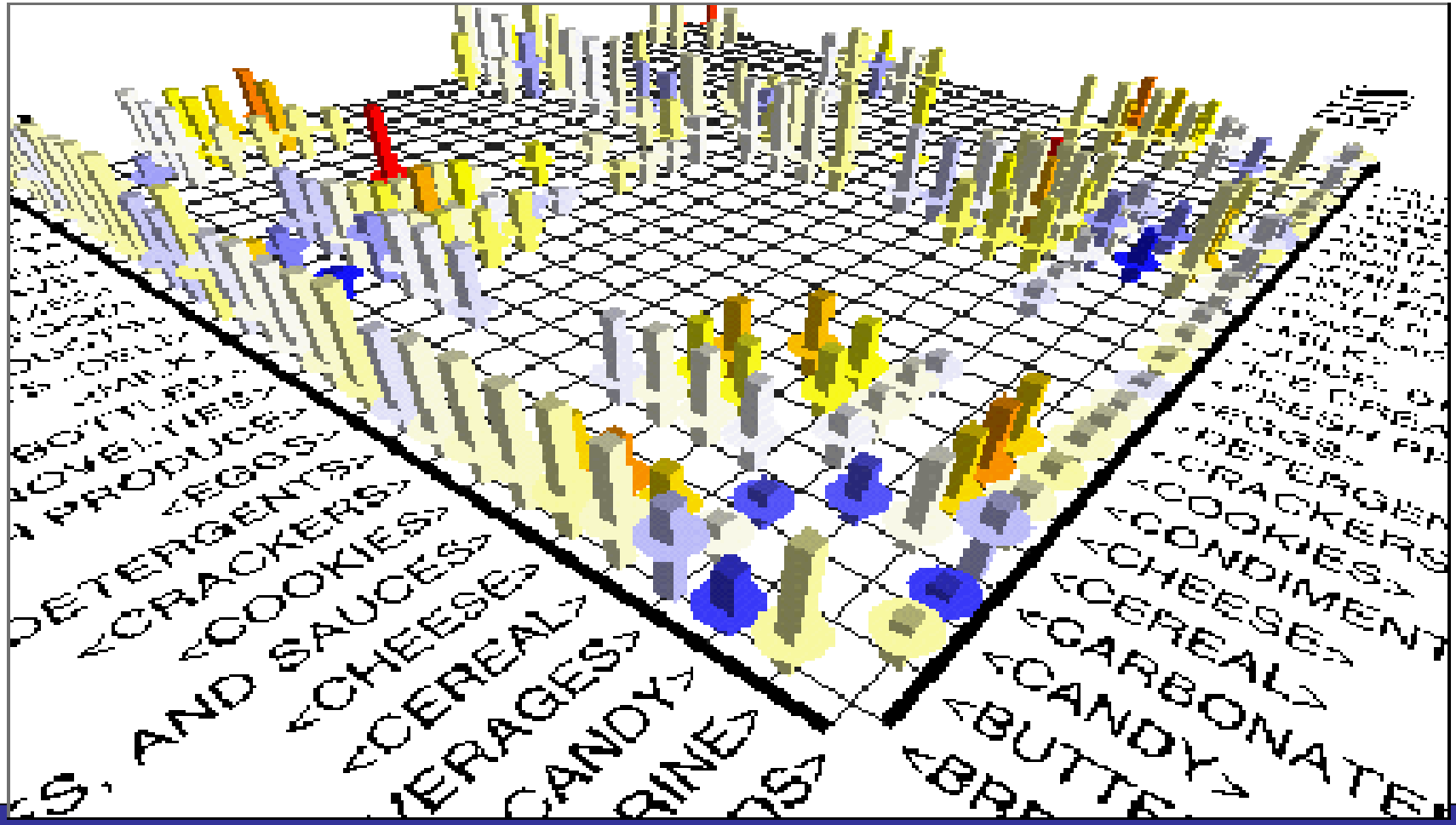
Box-plots dans StatSoft



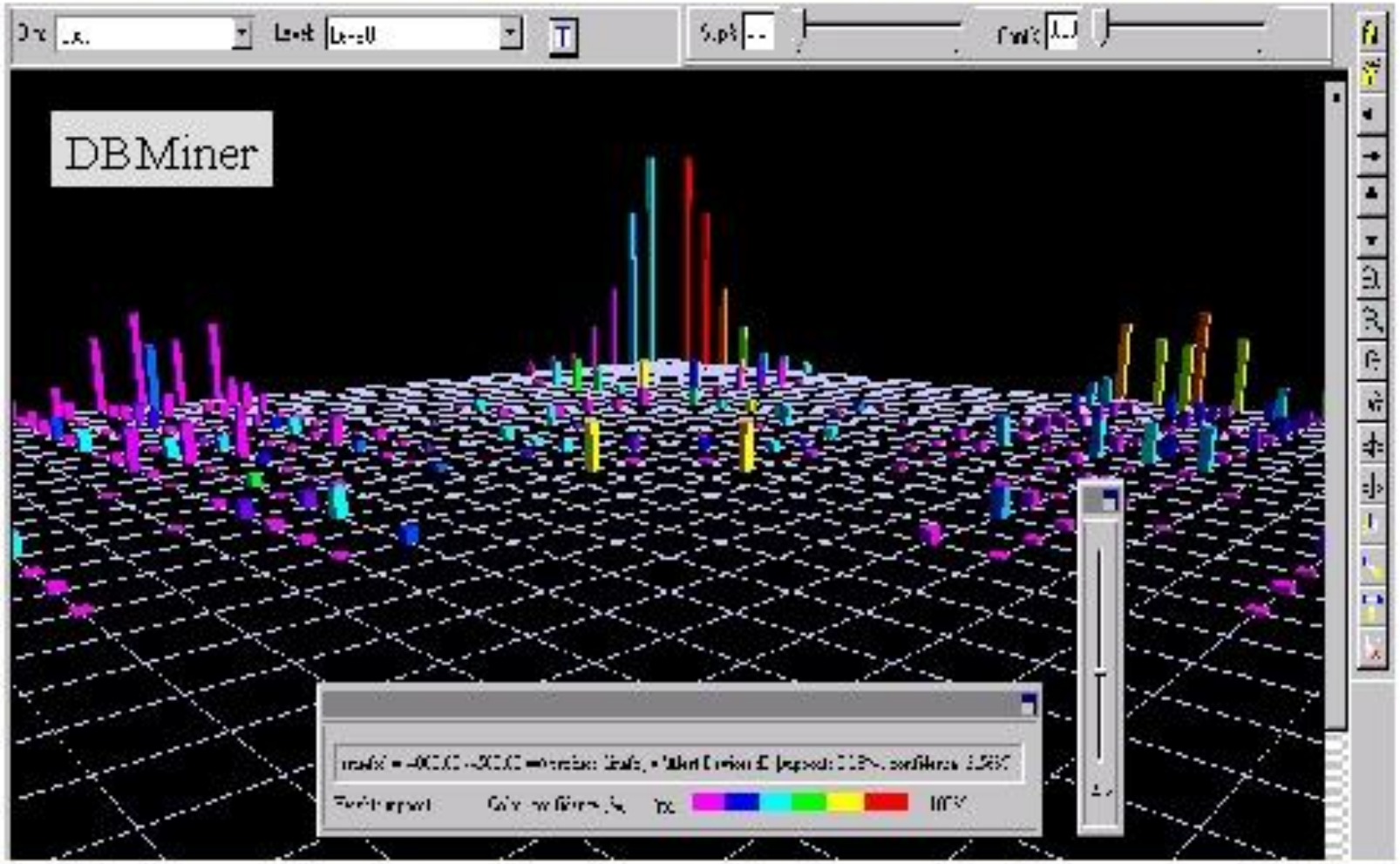
Scatter-plots dans SAS Enterprise Miner



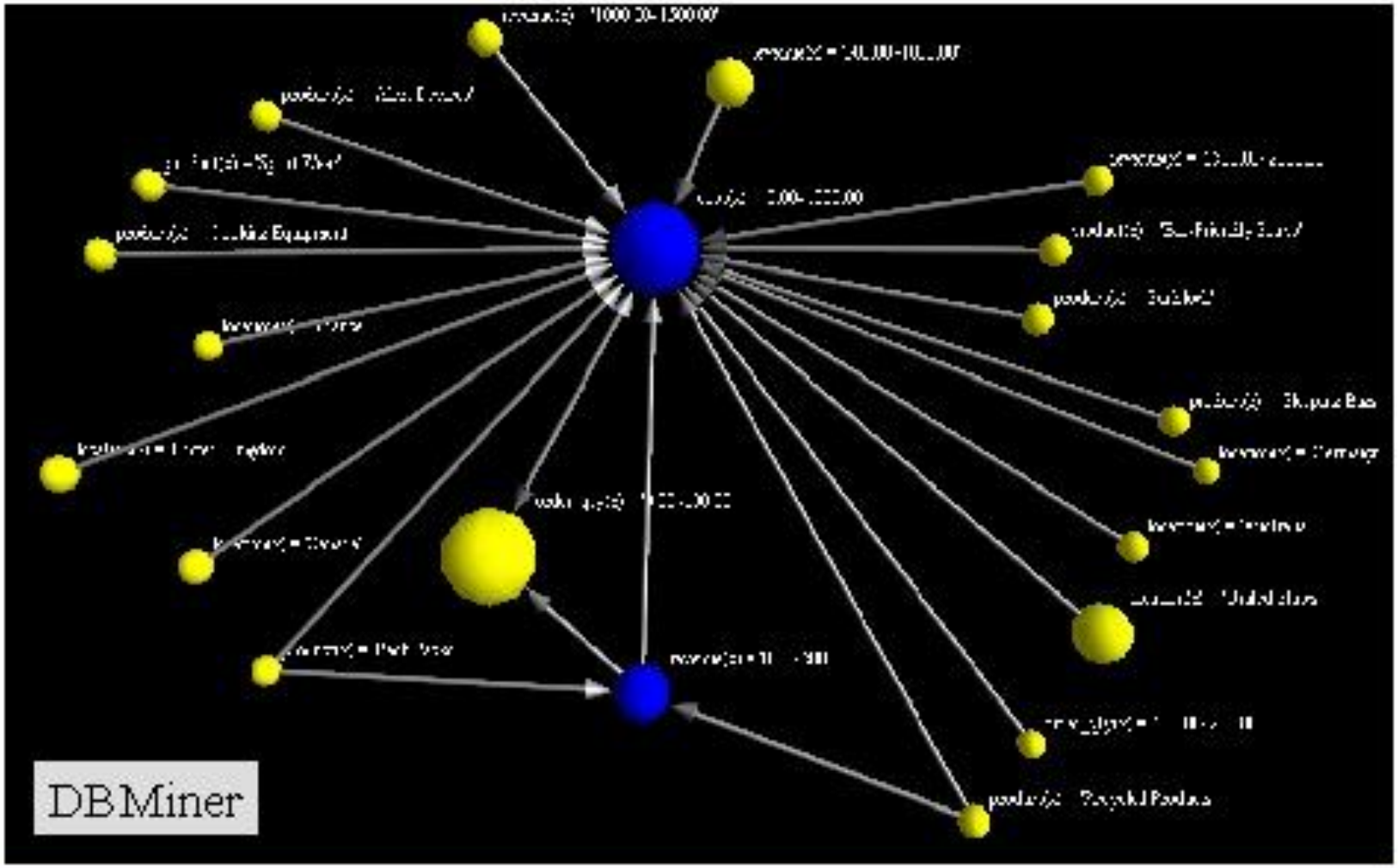
Règles d'association dans MineSet 3.0



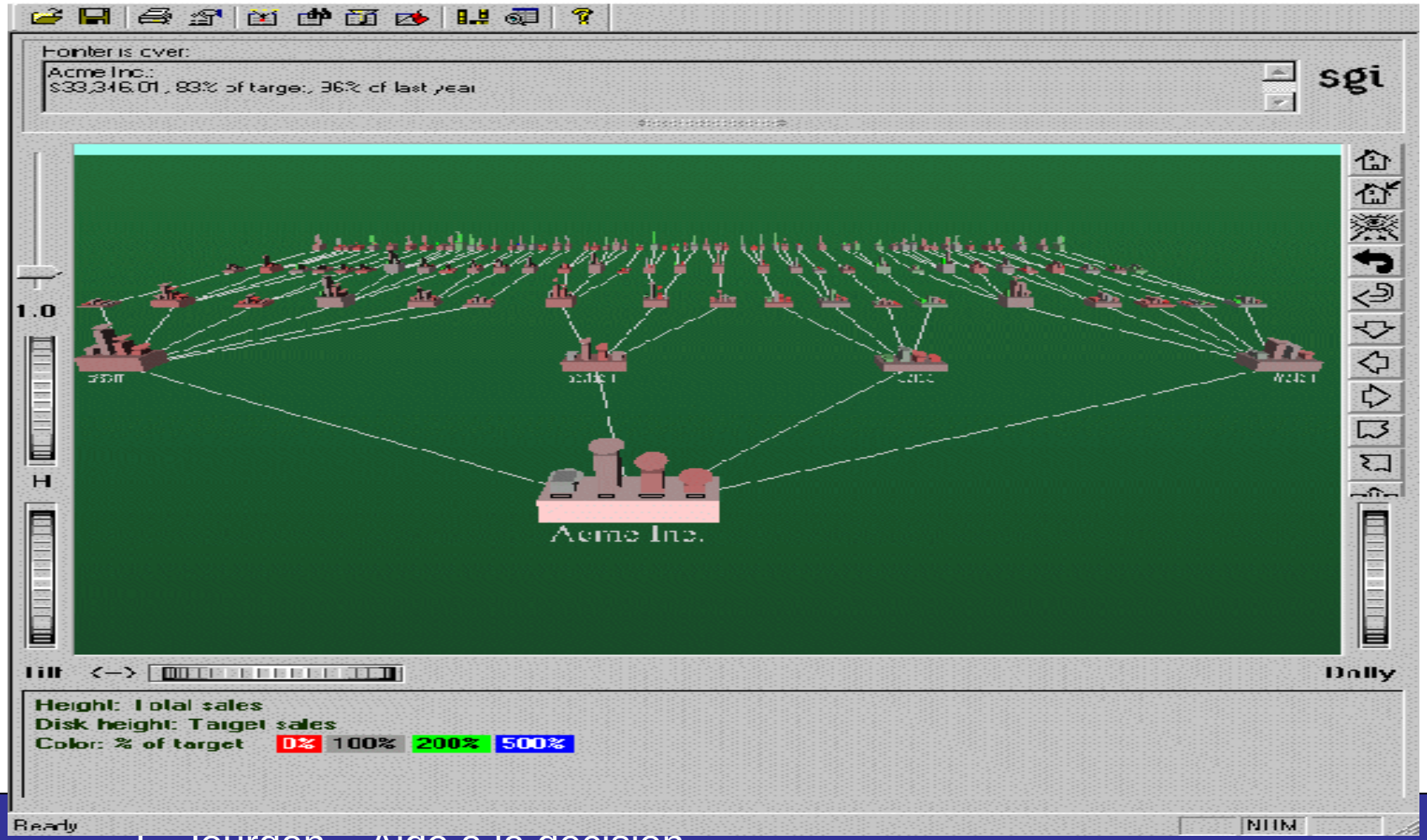
Visualization of Association Rule in Plane Form



Visualization of Association Rule Using Rule Graph

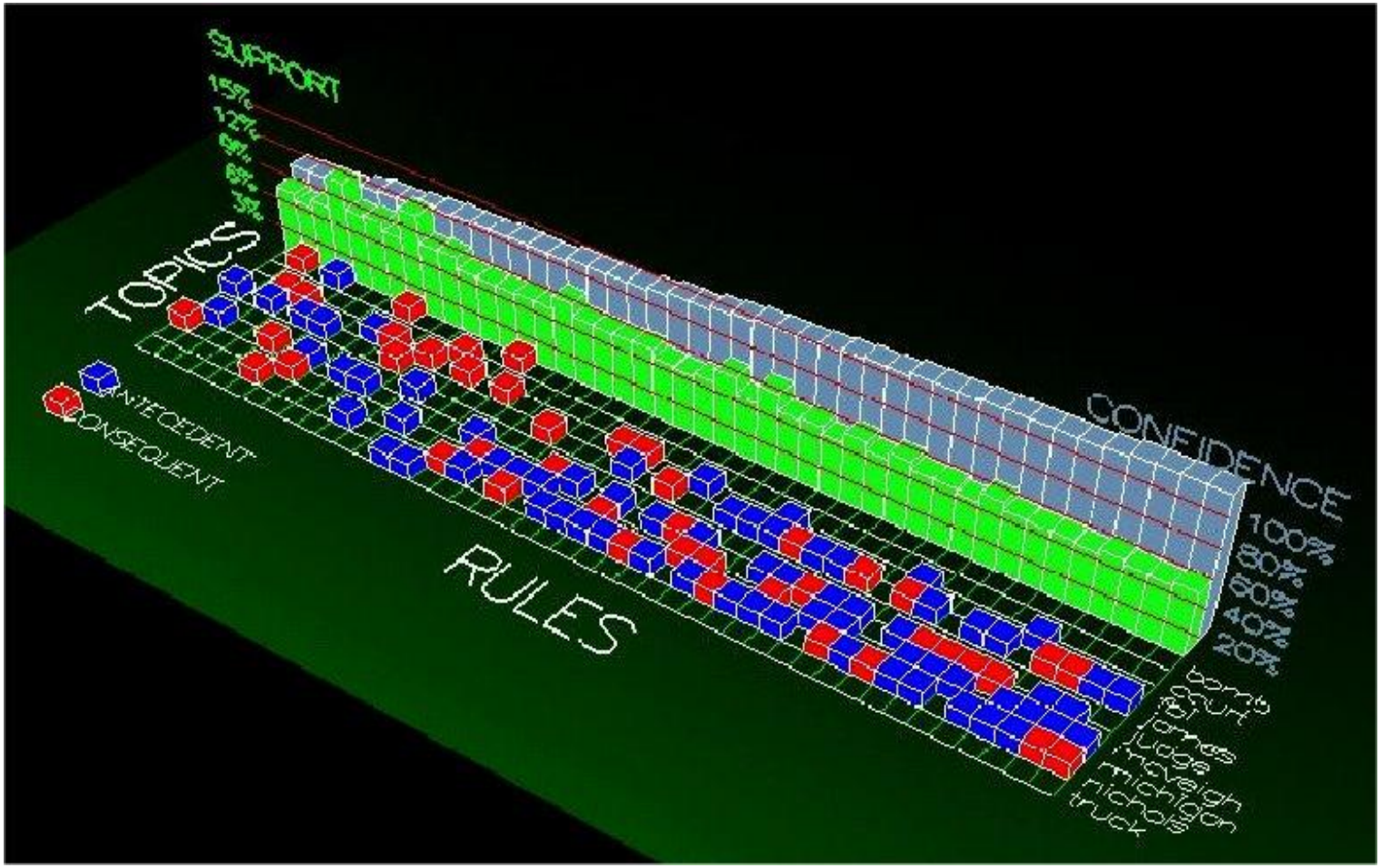


Arbres de décision dans MineSet 3.0

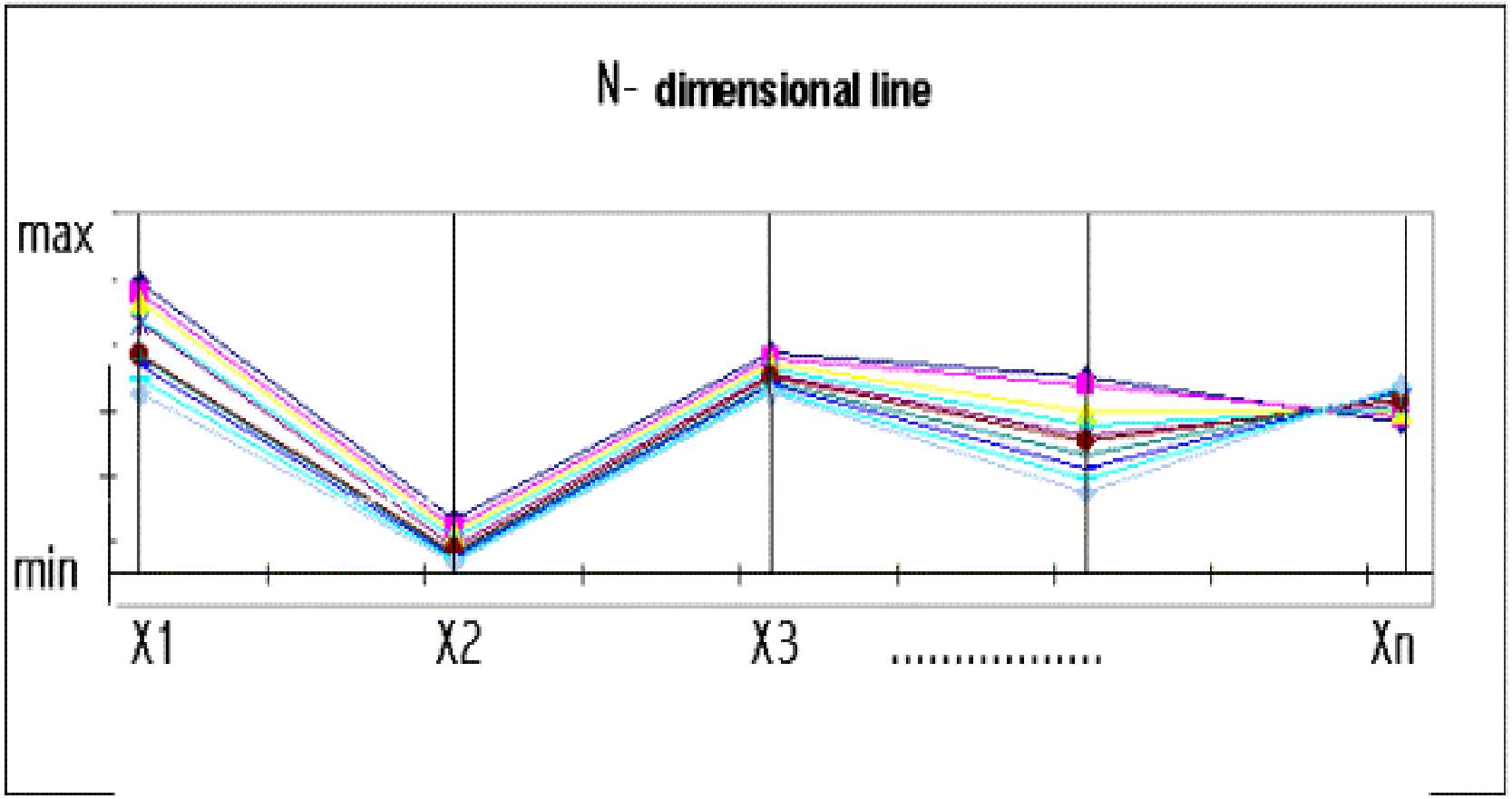


Clusters dans IBM Intelligent Miner

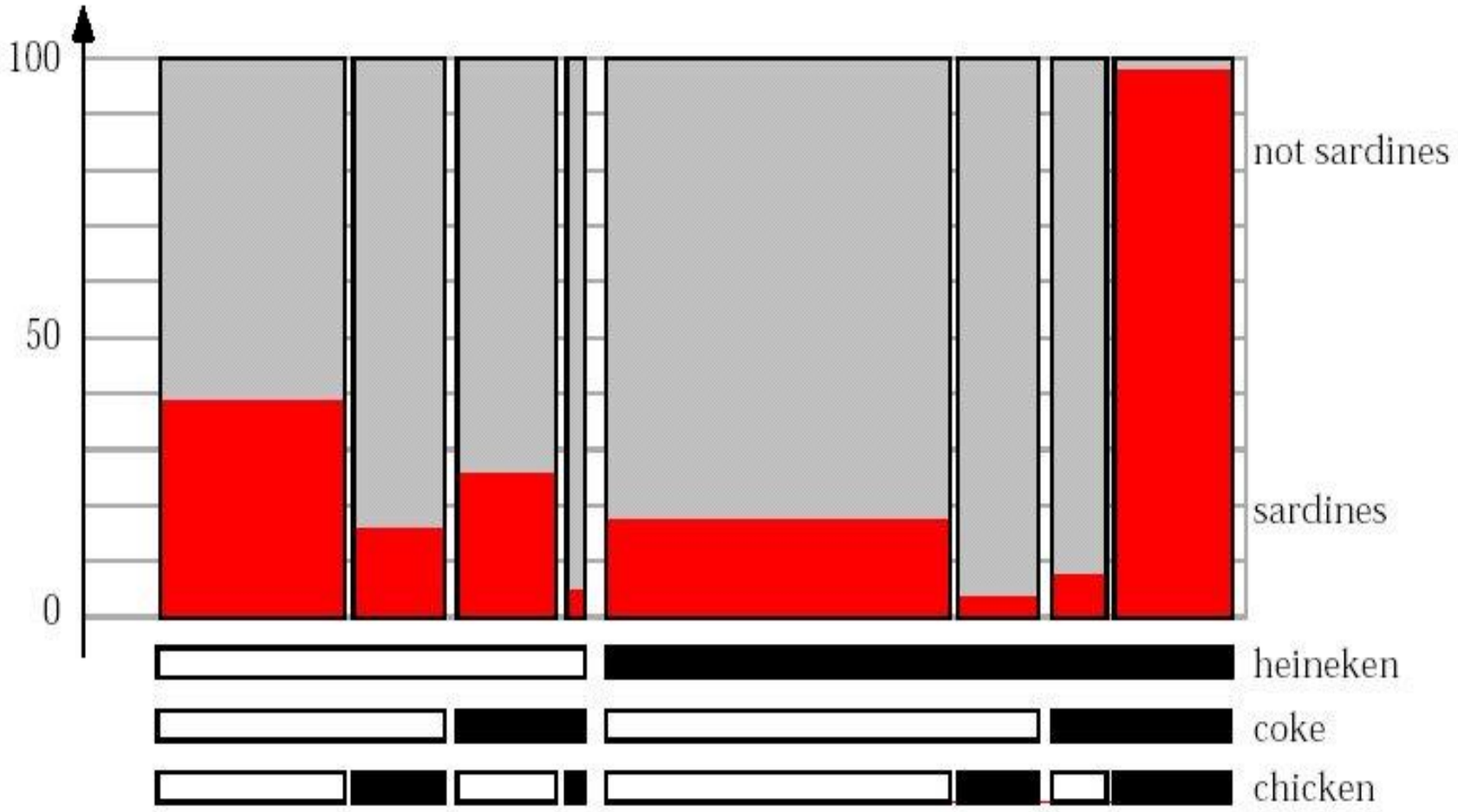
Règles d'association : La visualisation 3D



Règles d'association : Le N-Dimensional Line



Règles d'association : Le Double Decker Plot



Résumé

- **Data mining :** découverte automatique de "patterns" intéressants à partir d'ensembles de données de grande taille
- **KDD (Knowledge discovery) est un processus :**
  - pré-traitement
  - data mining
  - post-traitement
- **Domaines d'application :** distribution, finances, biologie, médecine, télécommunications, assurances, banques, ...

Résumé

- **L'information peut être extraite à partir de différentes types de bases de données** (relationnel, orienté objet, spatial, WWW, ...)
- **Plusieurs fonctions de data mining (différents modèles) :** clustering, classification, règles d'association, ...
- **Plusieurs techniques dans différents domaines :** apprentissage, statistiques, IA, optimisation, ...

Résumé

- **Plusieurs problèmes ouverts :**
  - Visualisation
  - Parallélisme et distribution
  - Issues de sécurité et confidentialité
- **Futur prometteur …**

Références bibliographiques (0)

- Statistiques/analyse de données : R. Duda, P. Hart, « Pattern classification », Wiley, 2000.
- Apprentissage automatique/Intelligence artificielle : T. Mitchell, « Machine learning », McGraw-Hill, 1997
- Bases de données : J. Han, M. Kamber, « Data mining: concepts and techniques », Morgan Kaufmann, 2000
- Synthèse Stats/BD : D.J. Hand, H. Mannila,P. Smyth, « Principles of data mining », MIT Press, 2001
- Pré-traitement des données : D. Pyle, « Data preparation for data mining », Morgan Kaufmann, 1999.

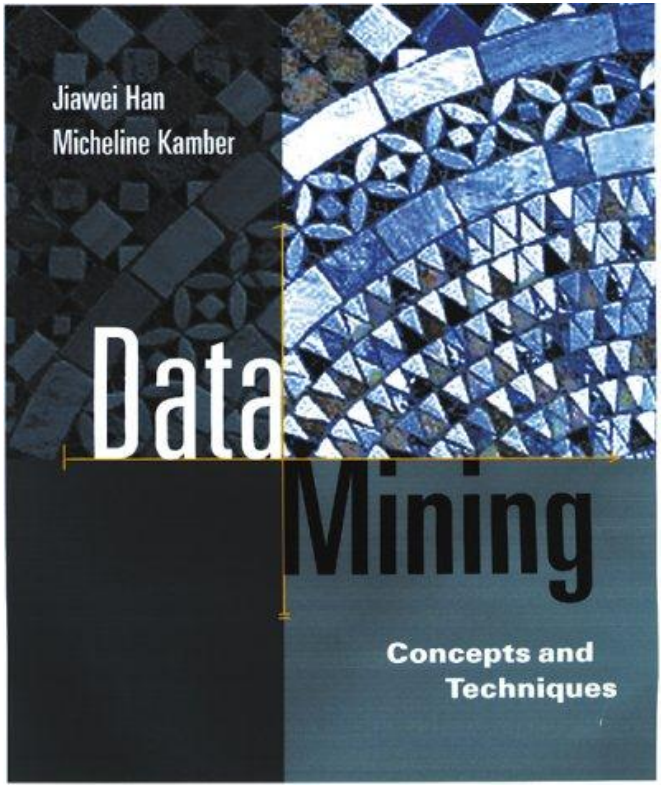
Références bibliographiques (1)

- Georges Gardarin
  - Université de Versailles (laboratoire PRISM)
  - Internet/intranet et bases de données – Data Web, Data Warehouse, Data Mining, Ed. Eyrolles
  - http://torquenada.prism.uvsq.fr/~gardarin/home.html
- Rakesh Agrawal (IBM)
  - IBM Almaden Research Center
  - http://www.almaden.ibm.com/cs/people/ragrawal/
- Mohammed Zaki
  - Rensselaer Polytechnic Institute, New York
  - http://www.cs.rpi.edu/~zaki/

Références bibliographiques (2)

- Vipin Kumar
  - Army High Performance Computing Research Center
  - http://www-users.cs.umn.edu/~kumar
- Rémi Gilleron
  - Découverte de connaissances à partir de données, polycopié (Université de Lille 3)
  - http://www.univ-lille3.fr/grappa
- *The Data Mine*
  - http://www.cs.bham.ac.uk/~anp/TheDataMine.html
- *Knowledge Discovery Nuggets* (*Kdnuggets*)
  - www.kdnuggets.com

Références bibliographiques (3)



•"Data Mining: Concepts and Techniques“
by Jiawei Han and Micheline Kamber,
Morgan Kaufmann Publishers,
August 2000. 550 pages. ISBN 1-55860-489-8