- Pourquoi prétraiter les données ?
- Nettoyage des données
- Intégration et transformation
- Réduction des données
- Discrétisation et génération de hiérarchies de concepts

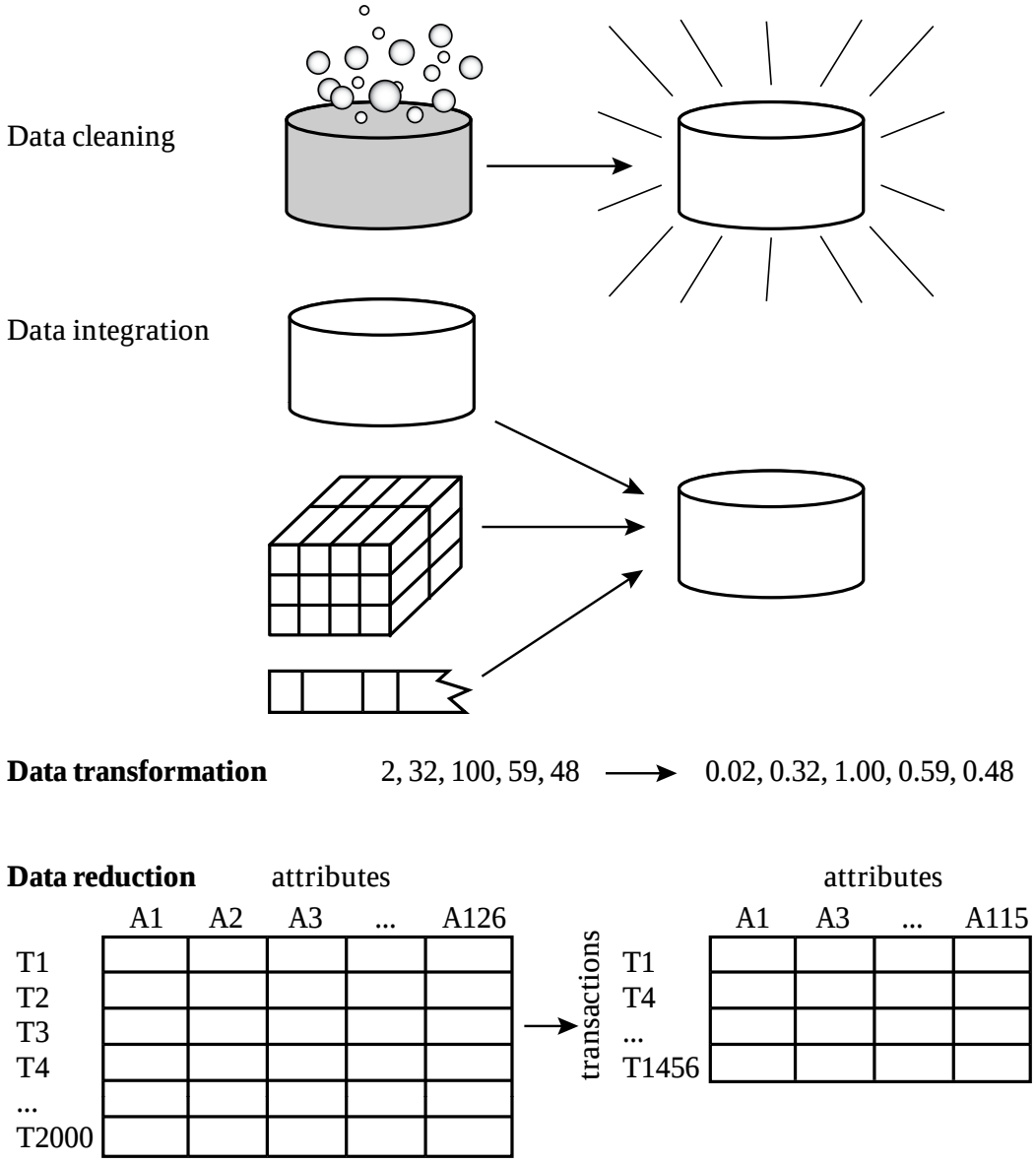
# Pourquoi prétraiter les données ?

- Données réelles souvent
  - incomplètes : valeurs manquantes, données simplifiées
  - bruitées : erreurs et exceptions
  - incohérentes : nommage, codage

- Résultats de la fouille dépendent de la qualité des données

# Principales étapes dans le prétraitement des données

- Nettoyage
- Intégration
- Transformation
- Réduction
- Discrétisation



# Données manquantes

- Données non disponibles
  - certains attributs n'ont pas de valeur
- Causes :
  - mauvais fonctionnement de l'équipement
  - incohérences avec d'autres données et donc supprimées
  - non saisies car non ou mal comprises
  - considérées peu importantes au moment de la saisie
- Ces données doivent être inférées

## Comment remplir les trous ?

- Ignorer le tuple
  - peu efficace quand le pourcentage de valeurs manquantes est élevé
- Compléter manuellement les données
  - Laborieux ou infaisable
- Utiliser une constante globale
  - ex : « inconnue », une nouvelle catégorie ?
- Utiliser la moyenne de l'attribut
- Utiliser la moyenne de l'attribut pour la même classe
  - mieux
- Utiliser la valeur la plus probable
  - formule Bayésienne ou arbre de décision

# Données bruitées

- Bruit : erreur ou variance aléatoire d'une variable mesurée
- Causes :
  - Instrument de mesure défectueux
  - Problème de saisie
  - Problème de transmission
  - Limitation technologique
  - Incohérence dans les conventions de nommage
- Autres problèmes :
  - enregistrement dupliqués
  - données incomplètes
  - données incohérentes

## Correction du bruit

- Par partitionnement (binning)
  - trier et partitionner les données
  - lisser les partitions par la moyenne, la médiane, les bornes, …
- Clustering
  - détecter et supprimer les exceptions
- Inspection humaine et informatique combinée
  - détection des valeurs suspectes et vérification humaine
- Régression
  - lisser les données par des fonctions de régression

# Partitionnement simple : lissage

- équi-largeur (distance) : *n* intervalles de même taille
- équi-profondeur : *n* intervalles contenant le même nombre de valeurs

* données triées : 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* équi-profondeur :

- partition 1 : 4, 8, 9, 15
- partition 2 : 21, 21, 24, 25
- partition 3 : 26, 28, 29, 34

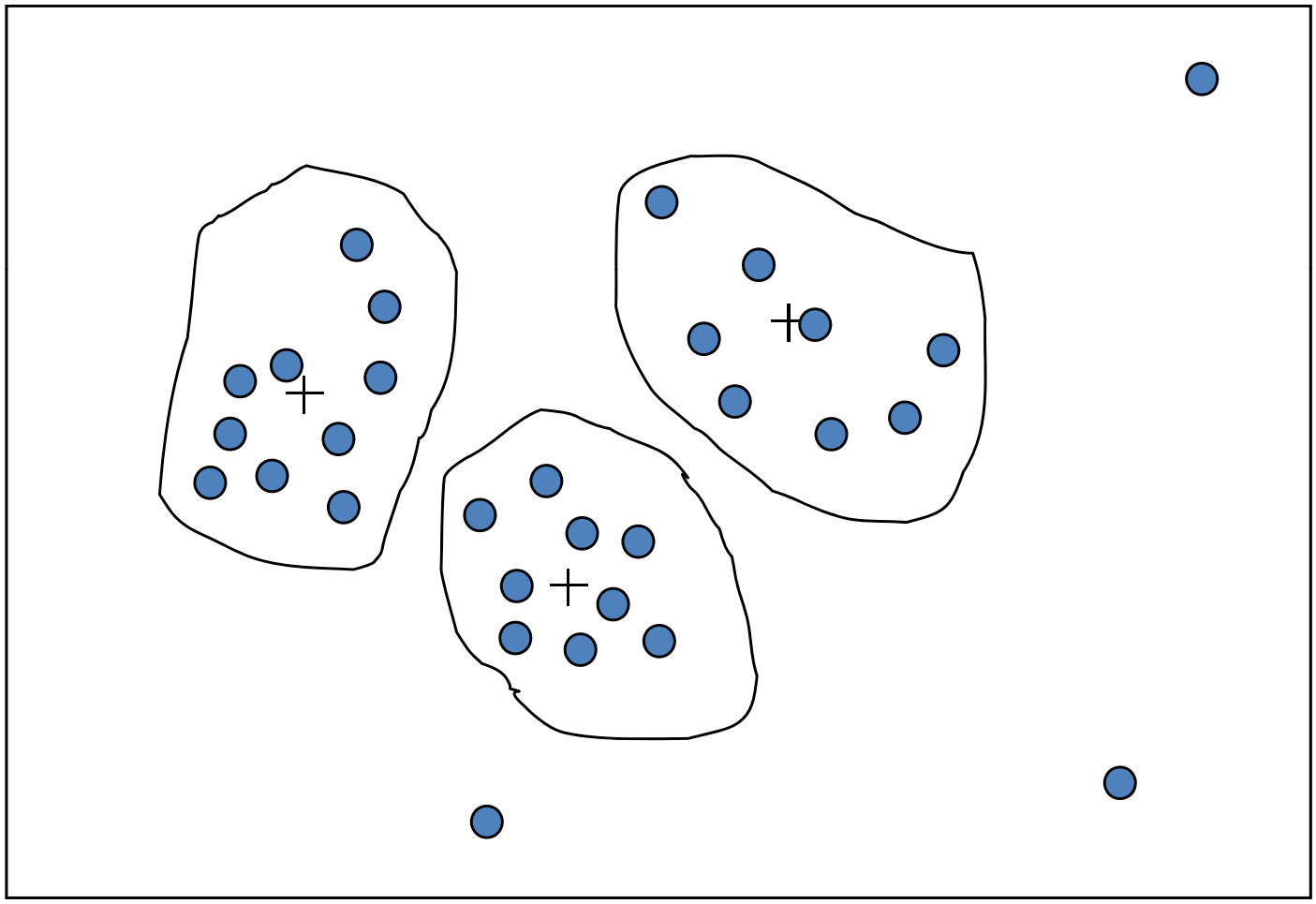
* lissage par la moyenne :

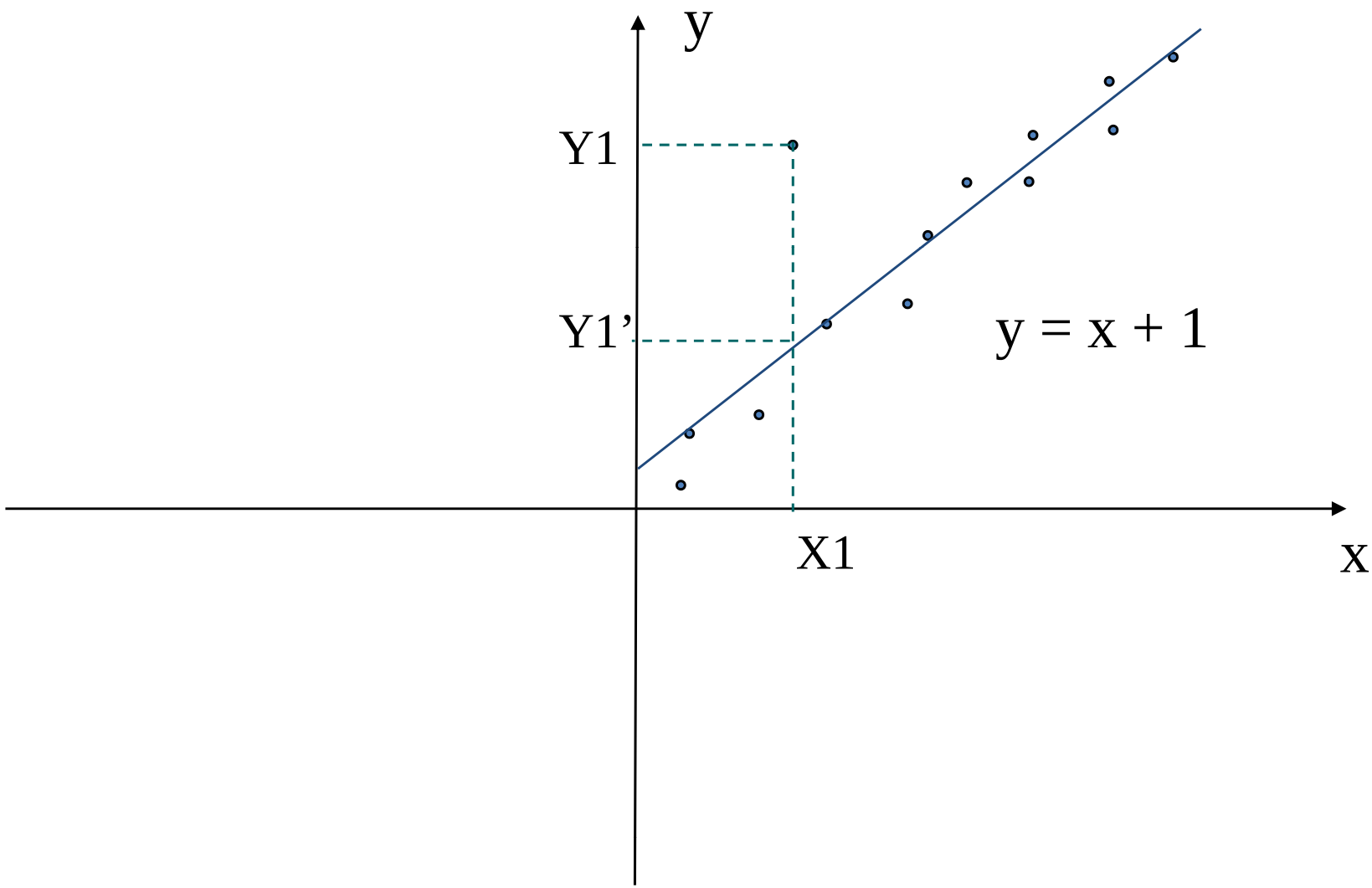
- partition 1: 9, 9, 9, 9
- partition 2: 23, 23, 23, 23
- partition 3: 29, 29, 29, 29

* lissage par les bornes :

- partition 1: 4, 4, 4, 15
- partition 2: 21, 21, 25, 25
- partition 3: 26, 26, 26, 34

# Clustering : suppression des exceptions




y
Y1
Y1'
y = x + 1
X1
x

- Intégration des données :
  - combinaison de différentes sources en une seule
- Intégration des schémas :
  - intégrer les méta-données de différentes sources
  - problème de nommage : identifier les différents noms des mêmes données réelles, ex : `num_client` ≡ `client_id`
- Détecter et résoudre les conflits de valeurs
  - pour les mêmes entités réelles, les valeurs des attributs provenant de sources différentes sont différentes
  - causes : représentation différentes, échelles différentes, ex : cm et pouces

# Gestion de la redondance

- fréquente lors de l'intégration de plusieurs sources de données
  - le même attribut peut avoir des noms différents
  - un attribut peut être déduit d'un autre
- peut être détectée par des analyses de corrélation

## Transformation des données

- Lissage : réduire le bruit dans les données
- Agrégation : simplification, construction de cubes de données
- Généralisation : hiérarchie de concepts
- Normalisation : mise à l'échelle pour avoir un petit intervalle spécifié
  - min-max
  - z-score
  - mise à l'échelle décimale

- min-max

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

- z-score

$$v' = \frac{v - mean_A}{stand\_dev_A}$$

- mise à l'échelle décimale

$$v' = \frac{v}{10^j}$$ avec $j$ le plus petit entier tq $\max(|v'|)<1$

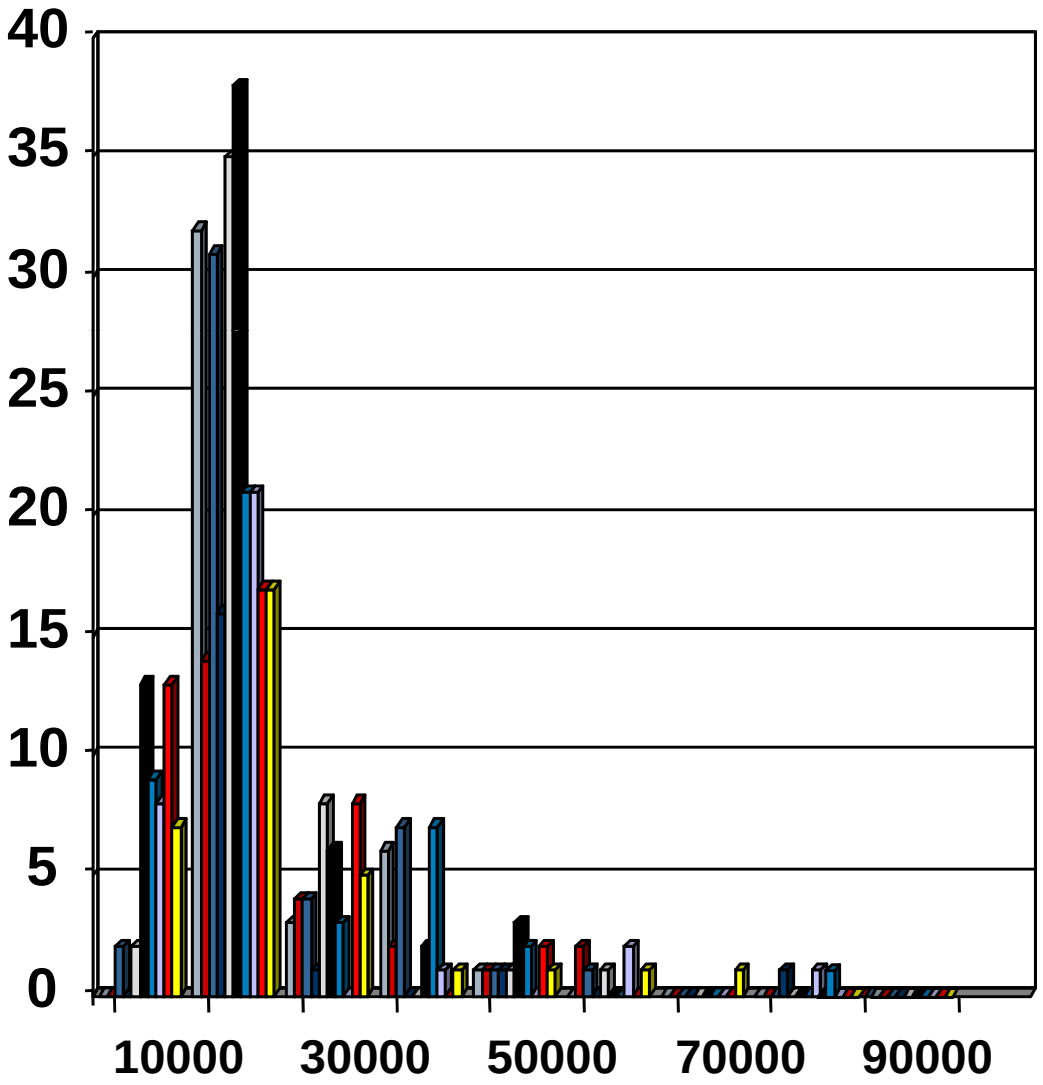
# Réduction des données

- La fouille de données peut être très longue sur les données complètes
- Réduction des données
  - obtenir une représentation réduite du jeu de données, plus petite en volume, mais qui produit les mêmes (ou presque) résultats analytiques
- Stratégies
  - Agrégation par cubes de données
  - Réduction de dimension
  - Réduction de numérosité
  - Discrétisation et génération de hiérarchies de concepts

- Méthodes paramétriques
  - suppose que les données suivent un modèle. Estimer et stocker seulement les paramètres du modèle
  - modèle log linéaire : approximation de la distribution des valeurs dans un espace multi-dimensionnel
- Méthodes non paramétriques
  - les données ne suivent pas un modèle
  - principales : histogrammes, clustering, échantillonnage

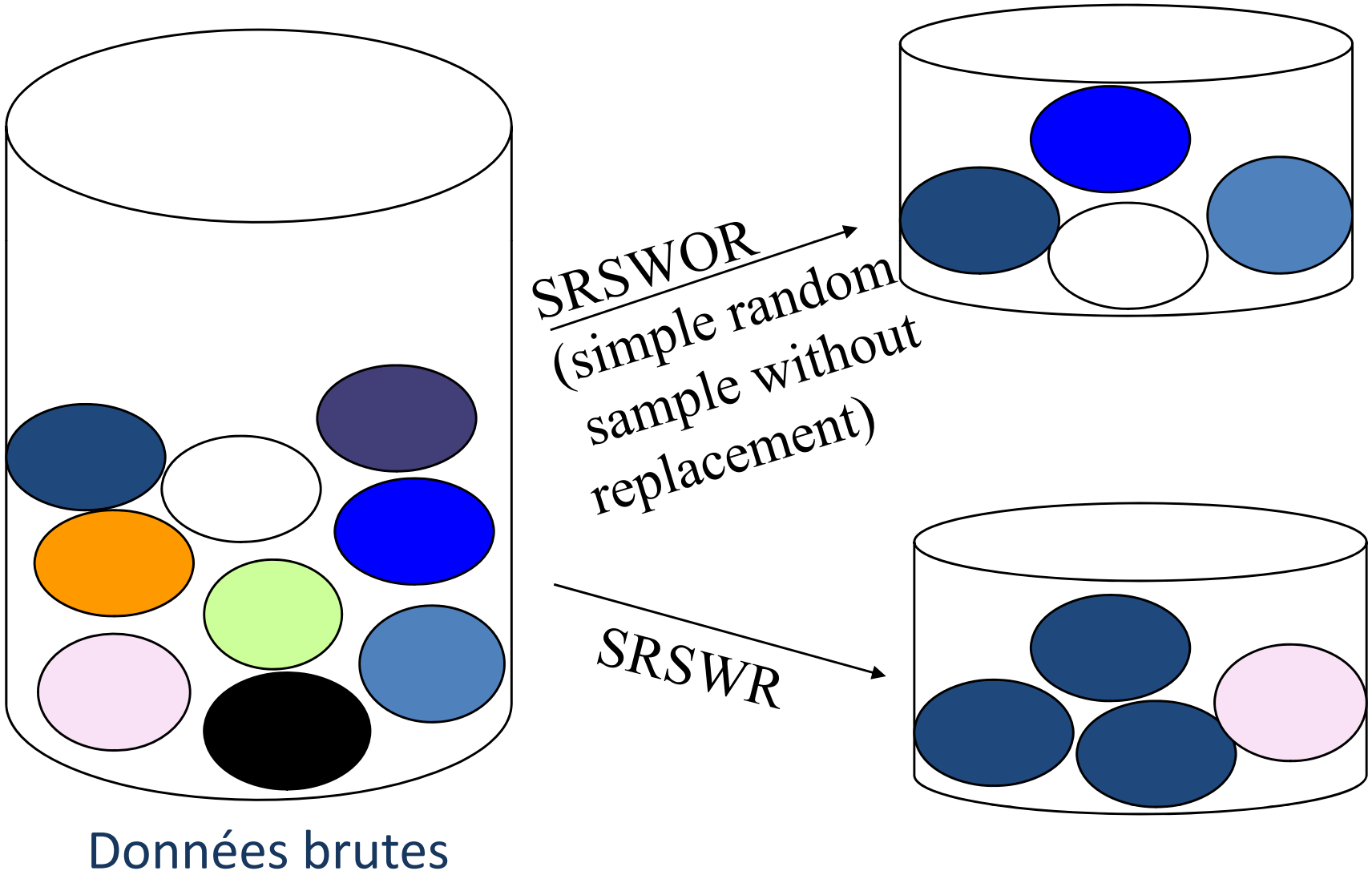
# Histogrammes

- populaire
- diviser en intervalles et stocker la moyenne (somme)
- mise en œuvre optimale sur une dimension par programmation dynamique

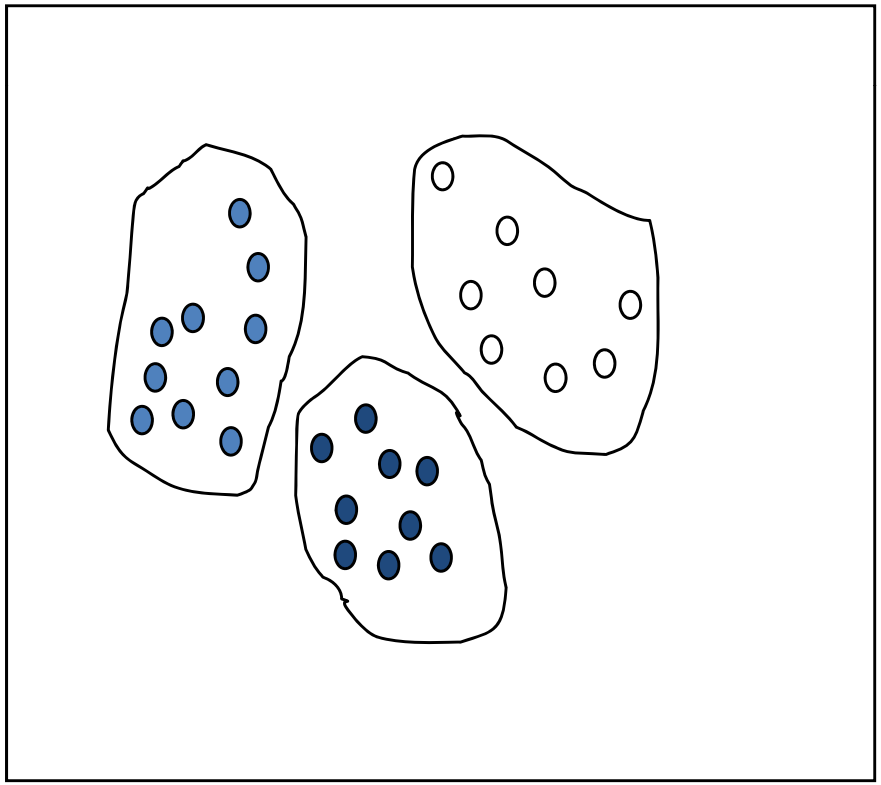


# Echantillonnage

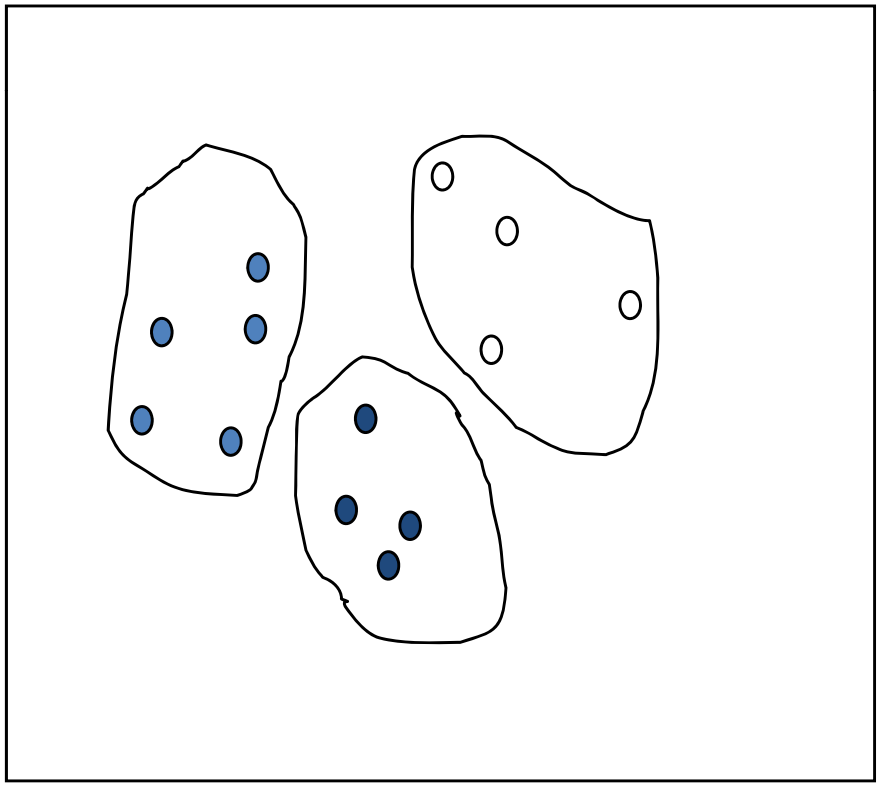
- Permet à un algorithme de s'exécuter en un temps sous-linéaire de la taille des données
- Choix d'un sous-ensemble représentatif des données
  - potentiellement mauvais dans le cas de biais dans les données
- Méthodes d'échantillonnage adaptatives
  - échantillonnage stratifié
    - approximer le pourcentage de chaque classe (ou sous population d'intérêt) dans le jeu de données complet
    - utilisé dans le cas de données biaisées
- L'échantillonnage peut ne pas réduire le nombre d'entrées/sorties


SRSWOR
(simple random
sample without
replacement)
SRSWR
Données brutes

## Données brutes



## Echantillon stratifié



# Discrétisation

- Trois types d'attributs
  - Nominal ou catégorique : valeurs d'un ensemble
  - Ordinal : valeurs d'un ensemble ordonné
  - Continu : réels
- Discrétisation

  → diviser l'intervalle de valeurs possibles en sous intervalles
  - certains algorithmes acceptent seulement des attributs catégoriques
  - réduit le volume des données
  - préparation pour de futures analyses

- Discrétisation
  - réduit le nombre de valeurs d'un attribut (continu) donné

- Hiérarchie de concepts
  - réduit les données en collectant et remplaçant les concepts de bas niveau (âge) par des concepts de niveau d'abstraction plus élevé (jeune, sénior)

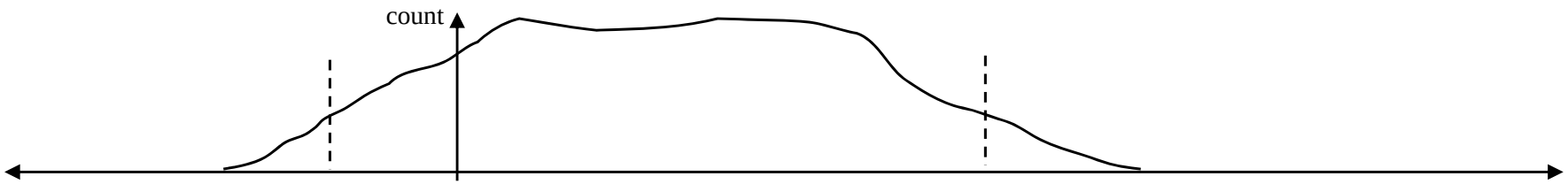
- Partitionnement (binning)
- Histogramme
- Clustering
- Basée entropie
- Segmentation par partitionnement naturel

# Segmentation par partitionnement naturel

La règle 3-4-5 peut être utilisée pour segmenter des données numériques en intervalles relativement uniformes

- Si un intervalle couvre 3, 6, 7 ou 9 valeurs distinctes au chiffre le plus significatif alors partitionner l'intervalle en 3 intervalles de même largeur
- Si un intervalle couvre 2, 4, ou 8 valeurs distinctes alors partitionner en 4 intervalles
- Si un intervalle couvre 1, 5, ou 10 valeurs distinctes alors partitionner en 5 intervalles

# Règle 3-4-5 : exemple

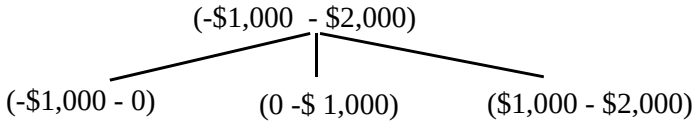



Step 1: -$351 -$159 profit $1,838 $4,700

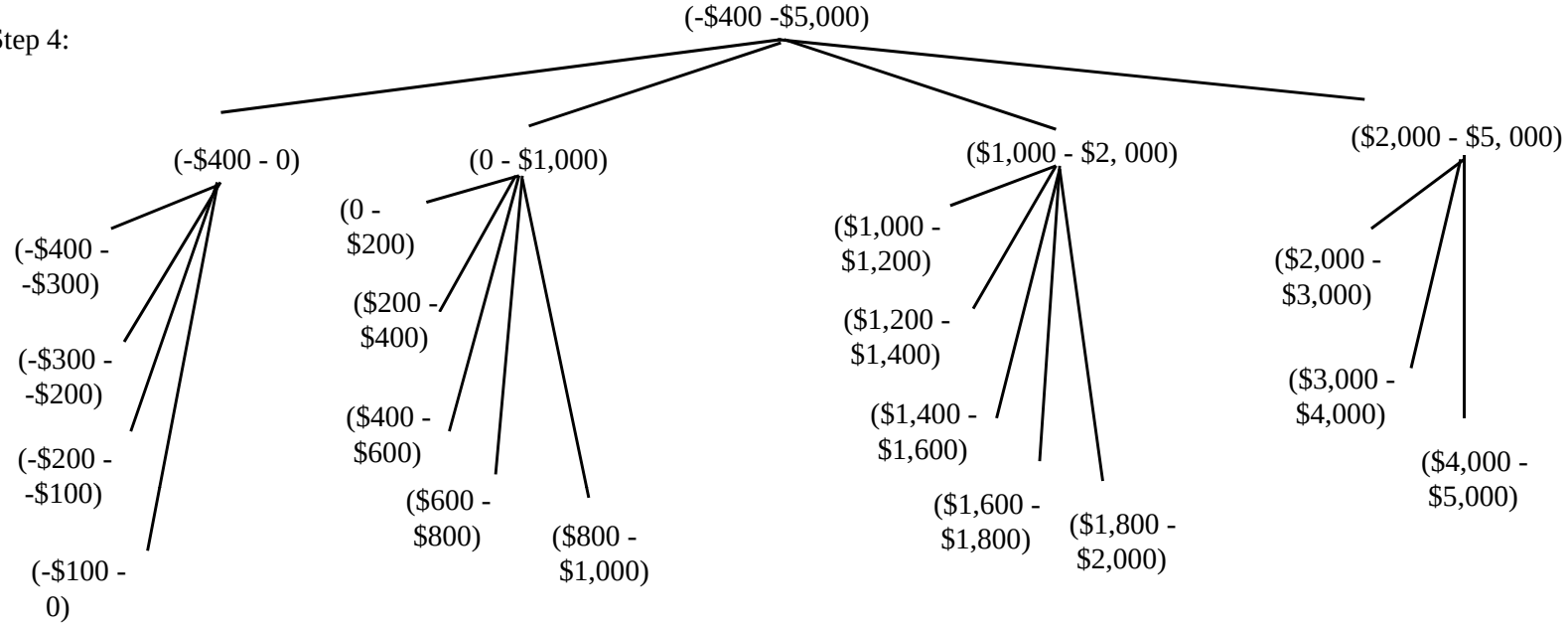
Min Low (i.e, 5%-tile) High(i.e, 95%-0 tile) Max

Step 2: msd=1,000 Low=-$1,000 High=$2,000

Step 3:



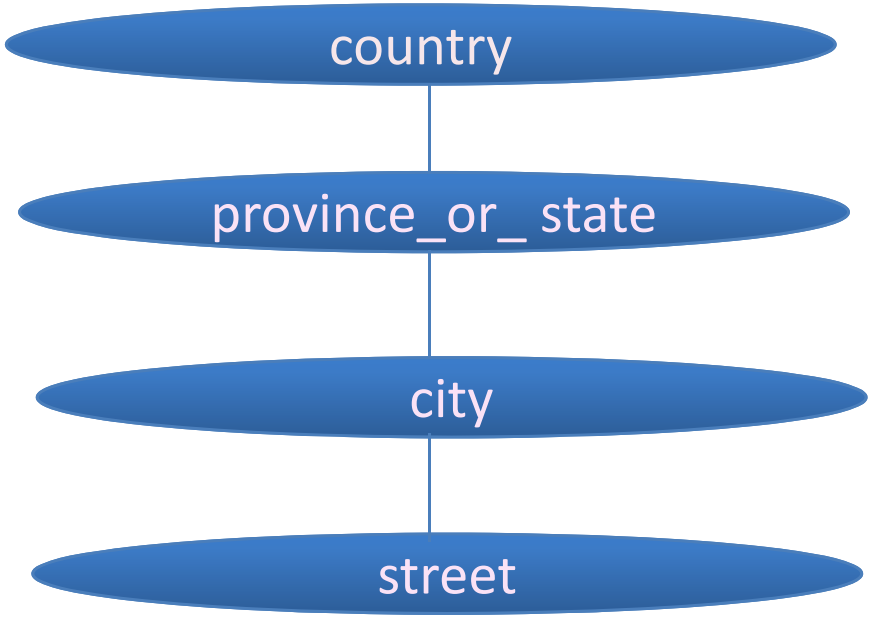

Step 4:



## Génération de hiérarchie de concepts pour des données nominales

- Spécification d'un ordre partiel par des utilisateurs ou des experts
  - ex : Gene Ontology
- Spécification d'une portion de hiérarchie par le groupage explicite des données
- Spécification d'un ensemble d'attributs sans ordre partiel
- Spécification partielle d'un ensemble

# La hiérarchie de concepts peut être générée automatiquement en se basant sur le nombre de valeurs distinctes d'un attribut.




15 valeurs distinctes

65 valeurs distinctes

3 567 valeurs distinctes

674 339 valeurs distinctes