Data Mining

2.6 Data Reduction

Fall 2008

Instructor: Dr. Masoud Yaghini

Data Reduction

- Why data reduction?
  - A database/data warehouse may store terabytes of data
  - Complex data analysis/mining may take a very long time to run on the complete data set
- Data reduction
  - Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results

Data Reduction

- Data reduction strategies:
  - Data aggregation
  - Attribute subset selection
  - Numerosity reduction
  - Discretization and concept hierarchy generation

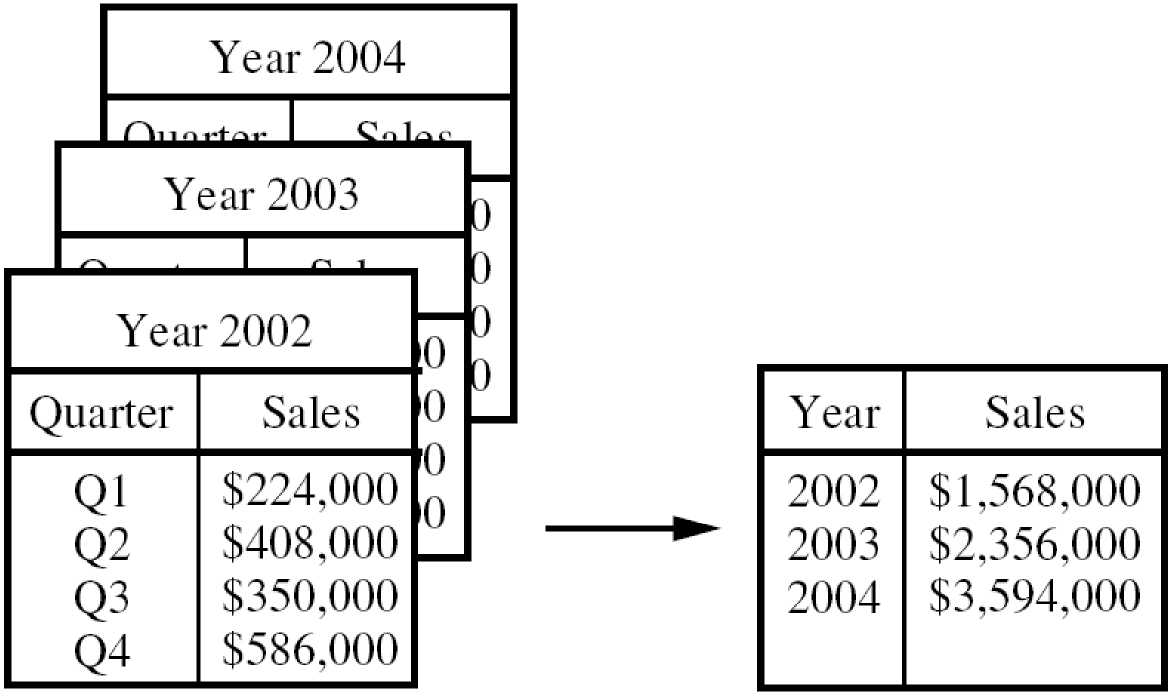
Outline

- Data Aggregation
- Attribute Subset Selection
- Numerosity Reduction
- References

Data Aggregation

Data Aggregation

- On the left, the sales are shown per quarter. On the right, the data are aggregated to provide the annual sales
- Sales data for a given branch of *AllElectronics* for the years 2002 to 2004.



Year 2004

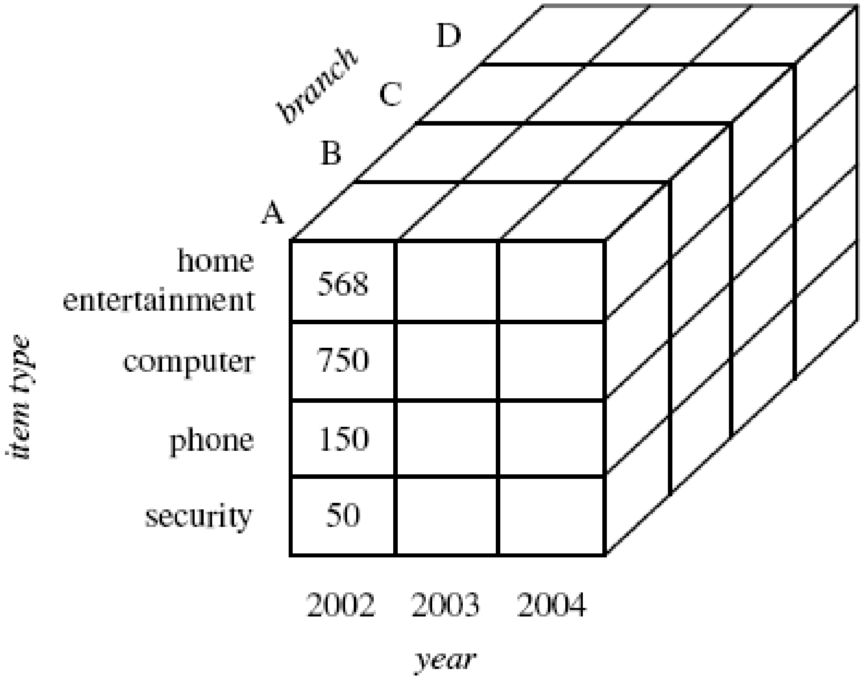
Year 2003

Year 2002	
Quarter	Sales
Q1	$224,000
Q2	$408,000
Q3	$350,000
Q4	$586,000

Year	Sales
2002	$1,568,000
2003	$2,356,000
2004	$3,594,000

Data Aggregation

- Data cubes store multidimensional aggregated information.
- Data cubes provide fast access to precomputed, summarized data, thereby benefiting on-line analytical processing as well as data mining.
- A data cube for sales at *AllElectronics.*



Data Cube Aggregation

- Base cuboid:
  - The cube created at the lowest level of abstraction is referred to as the *base cuboid.*
  - The base cuboid should correspond to an individual entity of interest, such as sales or customer.
- Apex cuboid:
  - A cube at the highest level of abstraction is the apex cuboid*.*
  - For the sales data, the apex cuboid would give one total—the total *sales.*

Attribute Subset Selection

Attribute Subset Selection

- Why attribute subset selection
  - Data sets for analysis may contain hundreds of attributes, many of which may be irrelevant to the mining task or redundant.
  - For example,
    - if the task is to classify customers as to whether or not they are likely to purchase a popular new CD at *AllElectronics* when notified of a sale, attributes such as the **customer's telephone number** are likely to be irrelevant, unlike attributes such as ***age*** or ***music taste****.*

Attribute Subset Selection

- Using domain expert to pick out some of the useful attributes
  - Sometimes this can be a difficult and time-consuming task, especially when the behavior of the data is not well known.
- Leaving out relevant attributes or keeping irrelevant attributes result in discovered patterns of poor quality.
- In addition, the added volume of irrelevant or redundant attributes can slow down the mining process.

Attribute Subset Selection

- Attribute subset selection (feature selection):
  - Reduce the data set size by removing irrelevant or redundant attributes
  - **Goal**: select a minimum set of features (attributes) such that the probability distribution of different classes given the values for those features is as close as possible to the original distribution given the values of all features
  - It reduces the number of attributes appearing in the discovered patterns, helping to make the patterns easier to understand.

Attribute Subset Selection

- How can we find a ‘good’ subset of the original attributes?
  - For n attributes, there are 2^n possible subsets.
  - An exhaustive search for the optimal subset of attributes can be prohibitively expensive, especially as n increase.
  - Heuristic methods that explore a reduced search space are commonly used for attribute subset selection.
  - These methods are typically greedy in that, while searching through attribute space, they always make what looks to be the best choice at the time.
  - Such greedy methods are effective in practice and may come close to estimating an optimal solution.

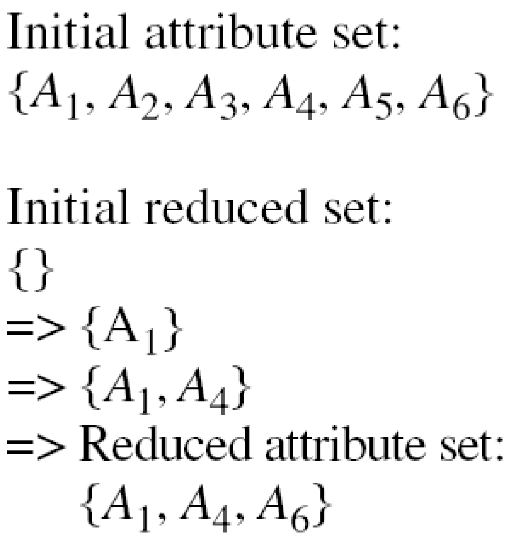
Attribute Subset Selection

- Heuristic methods:
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

- The “best” (and “worst”) attributes are typically determined using:
  - the tests of ***statistical significance***, which assume that the attributes are independent of one another.
  - the ***information gain*** measure used in building decision trees for classification.

Attribute Subset Selection

- Stepwise forward selection:
  - The procedure starts with an empty set of attributes as the reduced set.
  - First: The best single-feature is picked.
  - Next: At each subsequent iteration or step, the best of the remaining original attributes is added to the set.

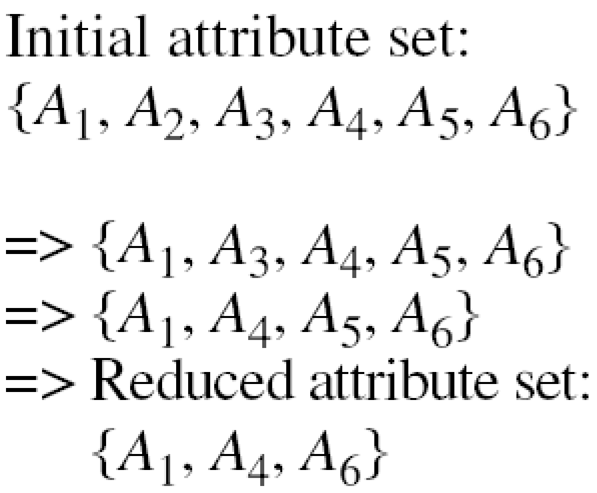


Initial attribute set:
$\{A_1, A_2, A_3, A_4, A_5, A_6\}$

Initial reduced set:
{}
=> $\{A_1\}$
=> $\{A_1, A_4\}$
=> Reduced attribute set:
$\{A_1, A_4, A_6\}$

Attribute Subset Selection

- Stepwise backward elimination:
  - The procedure starts with the full set of attributes.
  - At each step, it removes the worst attribute remaining in the set.



Initial attribute set:
$\{A_1, A_2, A_3, A_4, A_5, A_6\}$

=> $\{A_1, A_3, A_4, A_5, A_6\}$
=> $\{A_1, A_4, A_5, A_6\}$
=> Reduced attribute set:
$\{A_1, A_4, A_6\}$

Attribute Subset Selection

- Combining forward selection and backward elimination:
  - The stepwise forward selection and backward elimination methods can be combined
  - At each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.

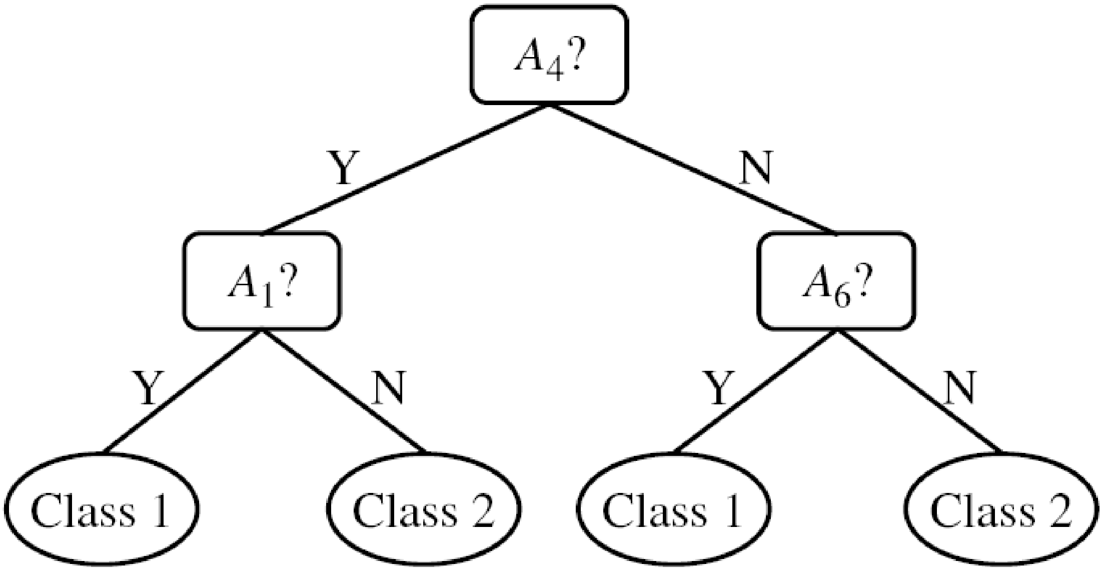
Attribute Subset Selection

- Decision tree induction:
  - Decision tree algorithms, such as ID3, C4.5, and CART, were originally intended for classification.
  - Decision tree induction constructs a flowchart-like structure where each internal (nonleaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction.
  - At each node, the algorithm chooses the "best" attribute to partition the data into individual classes.
  - When decision tree induction is used for attribute subset selection, a tree is constructed from the given data.
  - All attributes that do not appear in the tree are assumed to be irrelevant.

Attribute Subset Selection

- Decision tree induction

Initial attribute set:
$\{A_1, A_2, A_3, A_4, A_5, A_6\}$



=> Reduced attribute set:
$\{A_1, A_4, A_6\}$

Numerosity Reduction

Numerosity Reduction

- Reduce data volume by choosing alternative, smaller forms of data representation
- There are several methods for storing reduced representations of the data include histograms, clustering, and sampling.

Data Reduction: Sampling

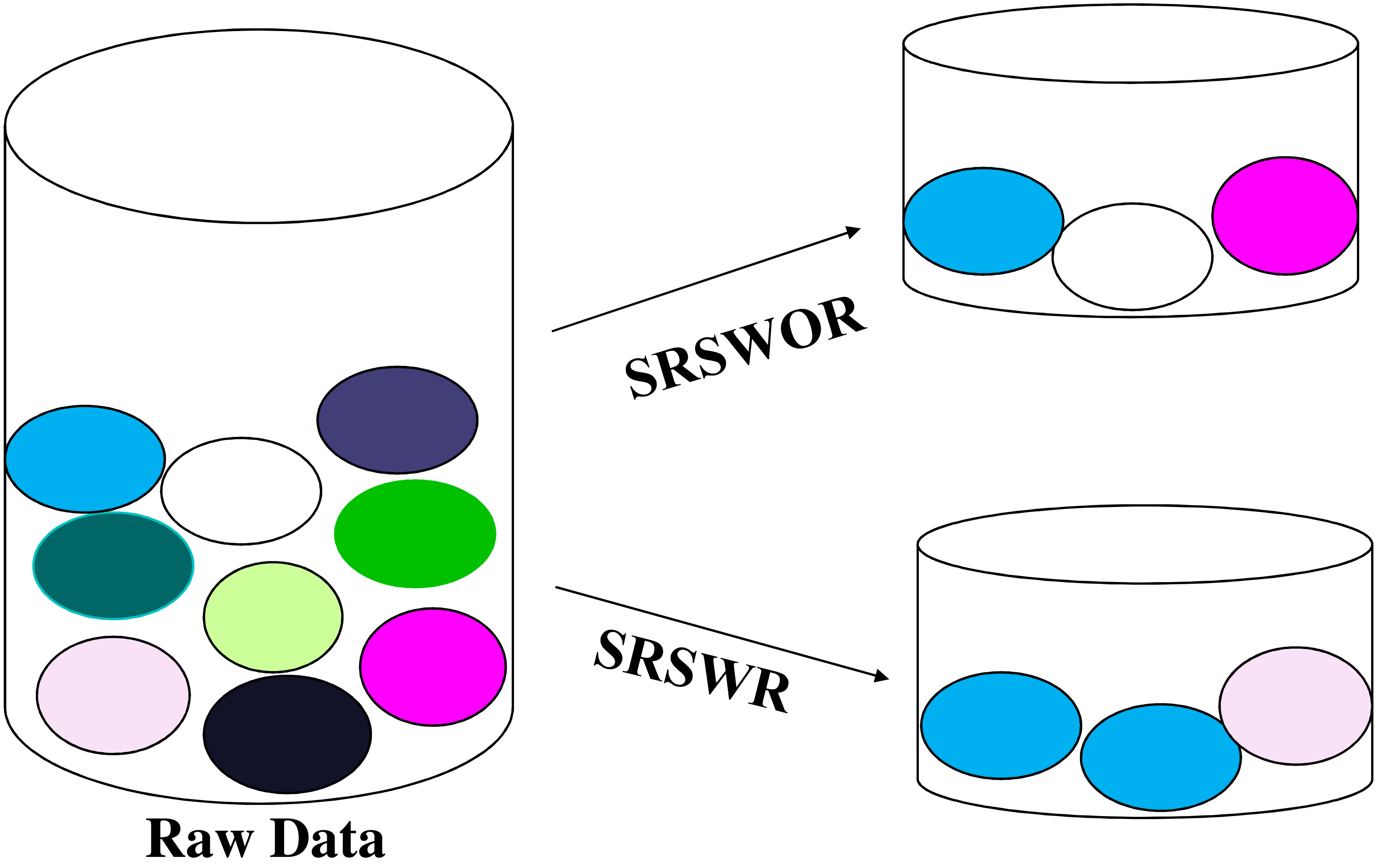
- Sampling: obtaining a small sample s to represent the whole data set N
- Suppose that a large data set, D, contains N instances.
- The most common ways that we could sample D for data reduction:
  - Simple random sample without replacement (SRSWOR)
  - Simple random sample with replacement (SRSWR)
  - Cluster sample
  - Stratified sample

Data Reduction: Sampling

- Simple random sample without replacement (SRSWOR) of size s:
  - This is created by drawing s of the N instances from D ($s < N$), where the probability of drawing any tuple in D is 1=N, that is, all instances are equally likely to be sampled.

- Simple random sample with replacement (SRSWR) of size s:
  - This is similar to SRSWOR, except that each time a tuple is drawn from D, it is recorded and then replaced.
  - That is, after a instance is drawn, it is placed back in D so that it may be drawn again.

Data Reduction: Sampling

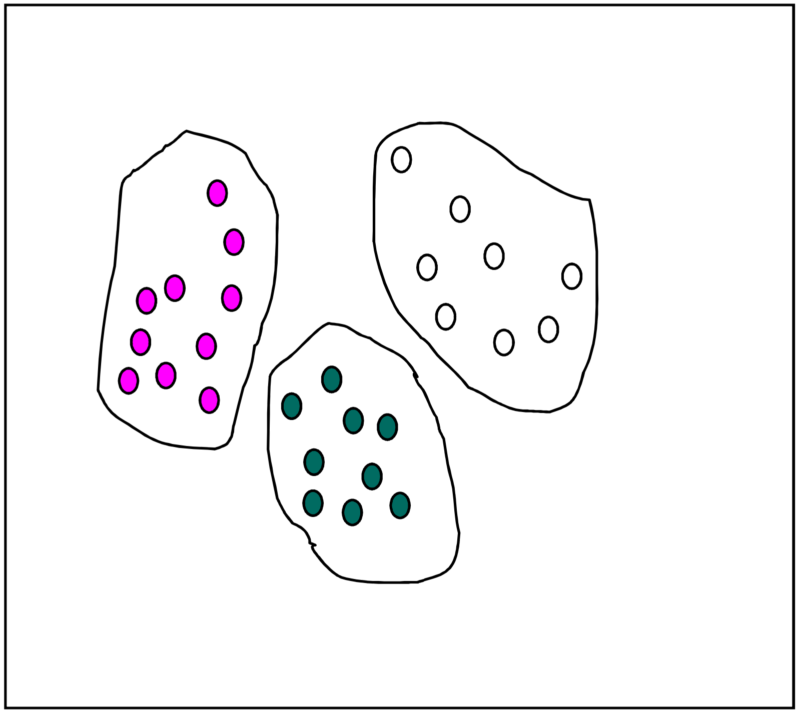


Data Reduction: Sampling

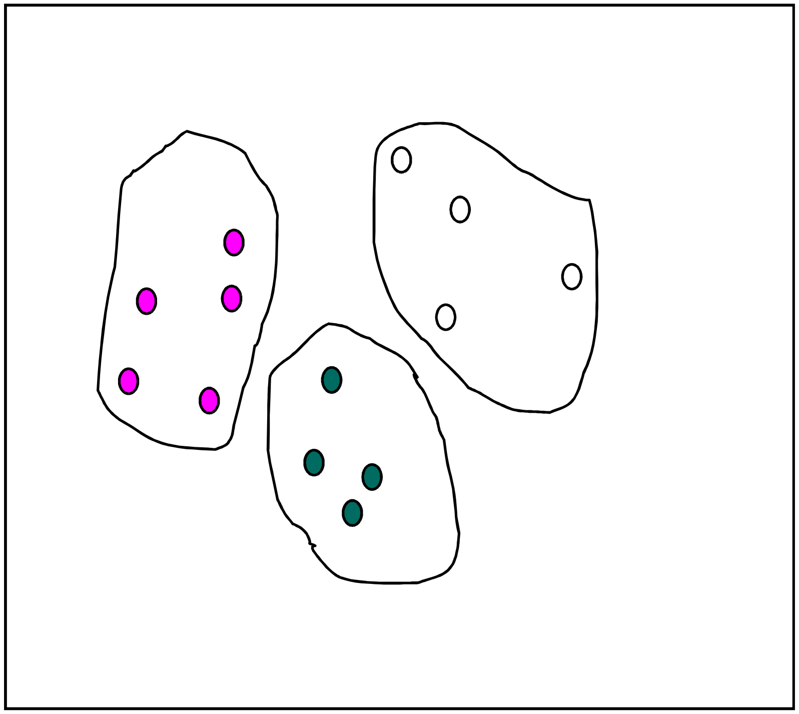
- Stratified Sample:
  - Simple random sampling may have very poor performance in the presence of skew
  - Approximate the percentage of each class (or subpopulation of interest) in the overall database
  - Used in conjunction with skewed data

Data Reduction: Sampling

Raw Data



Stratified Sample



References

References

- J. Han, M. Kamber, **Data Mining: Concepts and Techniques**, Elsevier Inc. (2006). (Chapter 2)

The end