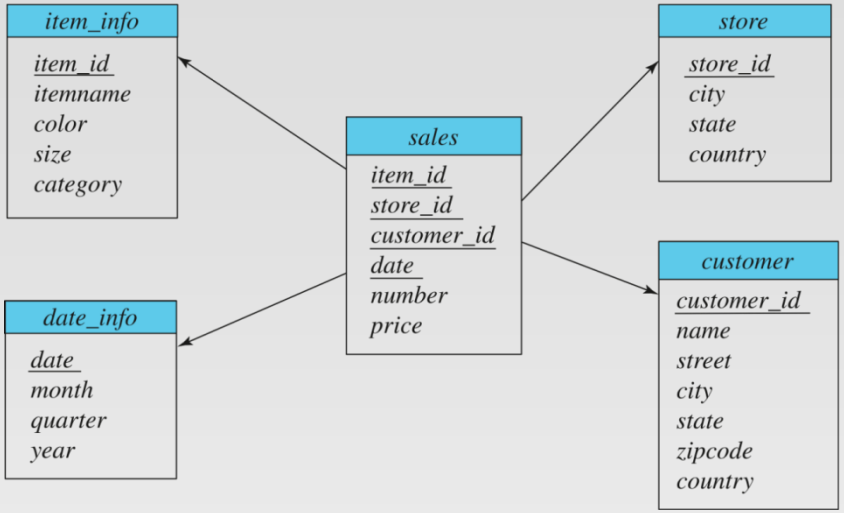
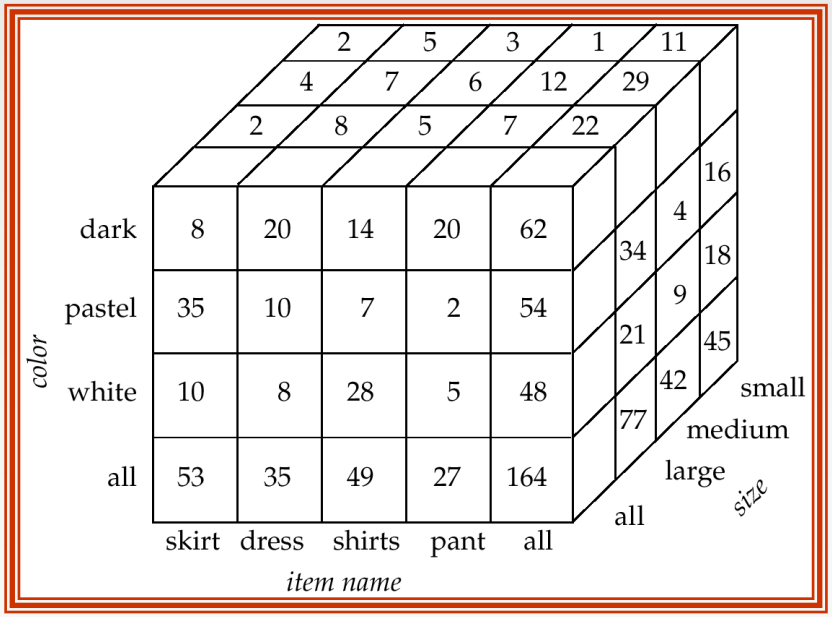
Data Mining

These slides are a modified version of the slides of the book "Database System Concepts" (Chapter 18), 5th Ed., McGraw-Hill, by Silberschatz, Korth and Sudarshan.
Original slides are available at www.db-book.com

Multidimensional Data

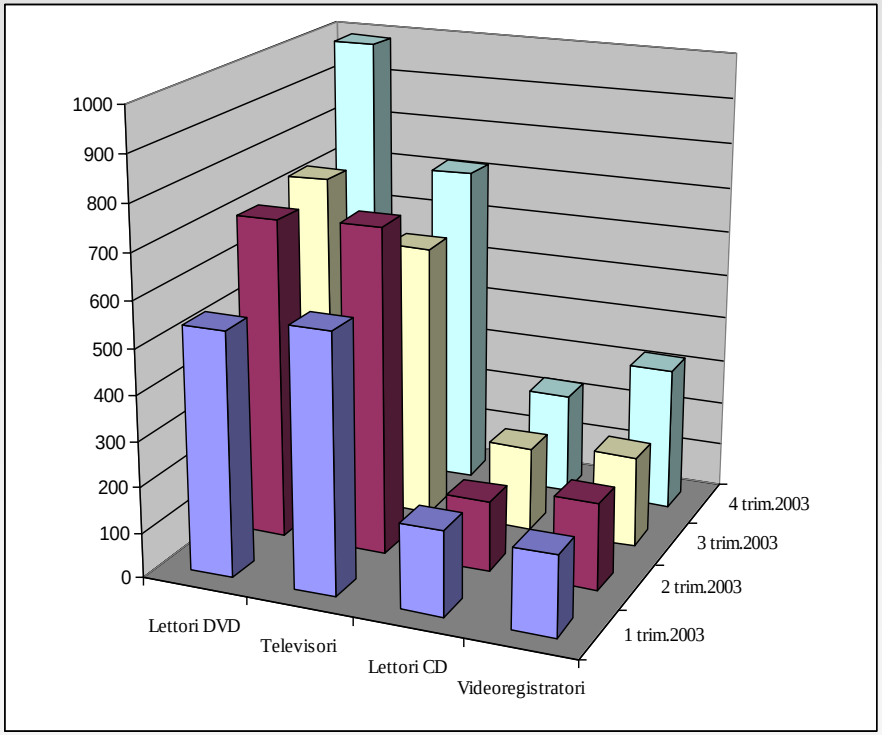




Data anlysis visualization

OLAP systems enable a user to view data from different points of view

Graphical representation



OLAP can be used for data mining

Data Mining

- Data mining is the process of semi-automatically analyzing large set of data to find useful informatin
  - **extracted information are called “pattern”**
- Like knowledge discovery in artificial intelligence (machine learning) and statistical analysis; differ in that it deals with large volume of data stored primarily on databases
  - **Knowledge discovery in data-bases**
- Types of knowledge discovered from a database:
  - can be represented by a set of rules
  - represented by equations relating different variables each other
  - predicting outcomes when the value of some variable is known

Example: “Young women with annual income greated than $50,000, are the most likely people to buy small sports cars”

- Not universally true
- Degree of support and confidence

Data Mining

- **Manual component to data mining**
  - preprocessing data to a form acceptable to the algorithms
  - postprocessing of discovered patterns to find novel ones that could be useful
  - Manual interaction to pick different useful types of patterns

  → data mining is a semi-automatic process in real life

- We concentrate on the automatic aspect of mining.
  We will study a few examples of patterns and we will see how they may be automatically derived from a database

Applications of Data Mining

- **Prediction**: applications that require some form of prediction

  Example: Prediction based on past history

  - Credit card company: Predict if a credit card applicant poses a good credit risk, based on some attributes (income, job type, age, ..) and past history
  - Predict what types of phone calling card usage are likely to be fraudulent (pattern of phone calling card usage)

- **Association**: applications look for associations (Descriptive Patterns)

  For instance: books that tend to be bought together
  If a customer buys a book, an on-line book store may suggest other associated books.
  If a person buys a camera, the system may suggest accessories.

  - Associations may be used as a first step in detecting **causation**
    - E.g. association between exposure to chemical X and cancer

Prediction mechanisms

- Some examples of prediction mechanisms:
  - **Classification**
    - Assume that items belong to one of several classes
    - Given past instances of items that along with the classes to which they belong (training instances)
    - Given a new item whose class is unknown, predict to which class it belongs (attributes of the instance must be used to predict the class)
  - **Regression**
    - Prediction of a value, rather than a class
      Given a set of variables X1, …, Xn, we wish to predict the value of a variable Y.
    - One way is to infer coefficients a_0, a_1, a_1, …, a_n such that

      $$Y = a_0 + a_1 * X_1 + a_2 * X_2 + \ldots + a_n * X_n$$

      Finding such a linear polynomial is called **linear regression**.

      The process of finding a curve that fits the data is also called **curve fitting**.

Classification Rules

- Classification rules help assign new objects to classes.

E.g., given a new automobile insurance applicant, should he or she be classified as low risk, medium risk or high risk?

- Classification rules for above example could use a variety of data, such as educational level, salary, age, etc., other than the actual payment history (which is unavailable for new customers)

- The company assigns a credit level to customers, excellent, average, good, average and bad, to each of a sample set of current customers according to each customer ‘s payment history.

Then the company finds rules that classify its current customers into excellent, average, good, average and bad, on the basis of information about the person, not the actual payment history.

Classification Rules

The rules may be of the following form:

- $\forall$ person P, P.degree = masters **and** P.income > 75,000

  $\Rightarrow$ P.credit = excellent

- $\forall$ person P, P.degree = bachelors **and**
  (P.income $\geq$ 25,000 and P.income $\leq$ 75,000)

  $\Rightarrow$ P.credit = good

Similar rules are present for the other credit levels (average and bad)

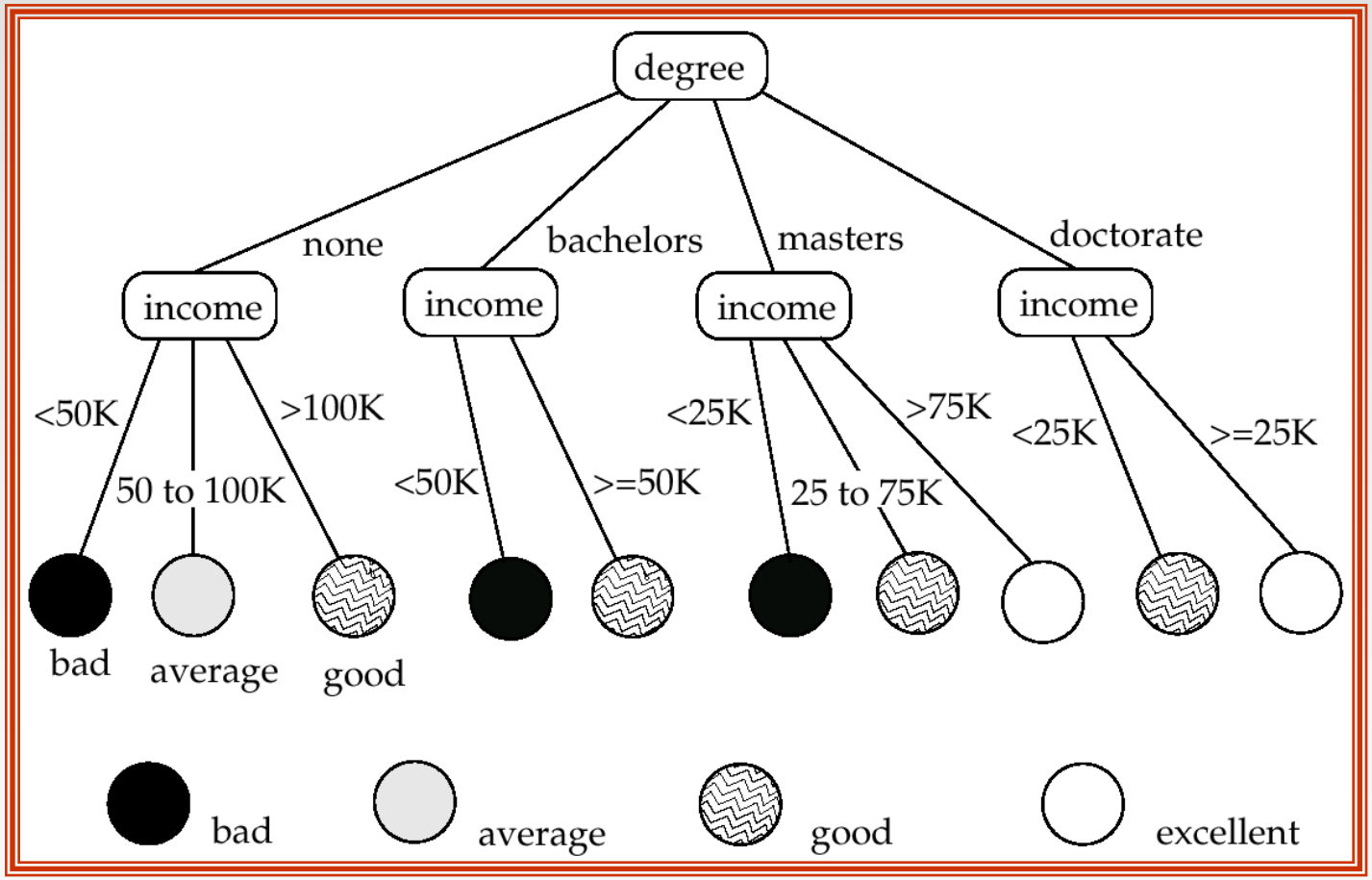
The process of building a classifier starts from a sample of data, called the training set.

For each tuple in the training set, the class to which the tuple belongs is already known.

- Rules are not necessarily exact: there may be some misclassifications

Decision Tree

Classification rules can be shown compactly as a decision tree.
Each node has an associated class
Each internal node has a predicate



Construction of Decision Trees

- **Training set**: a data sample in which the classification is already known.
- **Greedy** algorithm which works recursively starting at the root and building the tree downward (top down generation of decision trees).

  - Initially there is only the **root**. All training instances are associated with the root.
  - A each node if all or almost all instances associated with the node belongs to the same class, the node becomes a **leaf node** associated with that class.
  - Otherwise a **partitioning attribute** and a **partitioning condition** must be selected to create child nodes.
  - The data associated to each child node is the set of training instances that satisfy the partitioning condition for that child node.

Construction of Decision Trees

- Each internal node of the tree partitions the data into groups based on a **partitioning attribute**, and a **partitioning condition** for the node
- **Leaf** node:
  - all (or most) of the items at the node belong to the same class (leaves are “pure”) , or
  - all attributes have been considered, and no further partitioning is possible.

- Different algorithms to choose the sequence of partitioning attributes.

  Best Splits algorithm measures the purity (all training instances belong to only one class) of the data at the children resulting by partitioning using by that attribute using the selected conditions. The attribute and condition that results in the maximum purity is chosen.
- To classify a new instance, we start at the root and traverse the tree to reach a leaf. At an intermediate node we evaluate the predicate to find which child to go to.

Decision-Tree Construction Algorithm

Procedure *GrowTree* (*S*)
  Partition (*S*);

Procedure Partition (*S*)
  if (*purity* (*S*) > δ_p or |*S*| < δ_s) **then**
    return;
  for each attribute *A*
    evaluate splits on attribute *A*;
  Use best split found (across all attributes) to partition
    S into $S_1, S_2, \ldots, S_r$,
  for *i* = 1, 2, ….., *r*
    Partition (S_i);

Association Rules

- Retail shops are often interested in associations between different items that people buy.
  - Someone who buys bread is quite likely also to buy milk
  - A person who bought the book *Database System Concepts* is quite likely also to buy the book *Operating System Concepts*.

- **Association rules:**

  bread $\Rightarrow$ *milk* *DB-Concepts, OS-Concepts* $\Rightarrow$ Networks

  - Left hand side: antecedent, right hand side: consequent

- An association rule must have an associated population; the population consists of a set of instances
  - E.g. each transaction (sale) at a shop is an instance, and the set of all transactions is the population

Association Rules (Cont.)

Rules have an associated support, as well as an associated confidence.

- **Support** is a measure of what fraction of the population satisfies both the antecedent and the consequent of the rule.
  - E.g. suppose only 0.001 percent of all purchases include milk and screwdrivers. The support for the rule is *milk* $\Rightarrow$ *screwdrivers* is low.
  - Businesses are usually not interested in rules that have low support

  If 50 percent of all purchases involve milk and bread then support for the rule *bread* $\Rightarrow$ *milk* is high and may be worth attention

  Minimum degree of attention depends on the application.

- **Confidence** is a measure of how often the consequent is true when the antecedent is true.
  - E.g. the rule *bread* $\Rightarrow$ *milk* has a confidence of 80 percent if 80 percent of the purchases that include bread also include milk.

  The confidence *bread* $\Rightarrow$ *milk* may be different from the confidence *bread* $\Rightarrow$ *milk* although both have the same support

Finding Association Rules

- We are generally only interested in association rules with reasonably high support (e.g. support of 2% or greater)
- To discover association rules of the form

  $a1, a2, \ldots, an \Rightarrow a$

  1. Consider all possible sets of relevant items
  2. For each set find its support (i.e. count how many transactions purchase all items in the set).

     Large itemsets: sets with sufficiently high support
  3. Use large itemsets to generate association rules.
     For each itemset , output all rules with sufficient confidence that involve all and only the elements in the set

     For instance, from itemset *A* generate the rule

     $A - \{b\} \Rightarrow b$ for each $b \in A$.

     - Support of rule = support (*A*).

     - Confidence of rule = support (*A*) / support (*A* - {*b* })

Finding Support

How to generate all large itemsets.

- The **a priori** technique to find large itemsets:
  - Pass 1: consider sets with only 1 item.
    Count support of all sets with just 1 item.
    Eliminate those items with low support.

    ……..

  - Pass *i*: candidates: every set of *i* items such that all its *i-1* item subsets are large itemsets

    Count support of all candidates

    Eliminate those items with low support.

    Stop if there are no candidates

- Once a set is eliminated, none of its supersets needs to be considered.
  In pass i, it suffices to tests all subsets of size i-1.

Other Types of Associations

- Basic association rules have several limitations
- Deviations from the expected probability are more interesting
  - E.g. if many people purchase bread, and many people purchase cereal, quite a few would be expected to purchase both
  - We are interested in positive as well as negative correlations between sets of items
    - Positive correlation: co-occurrence is higher than predicted
    - Negative correlation: co-occurrence is lower than predicted
- Sequence associations / correlations
  - E.g. whenever bonds go up, stock prices go down in 2 days
- Deviations from temporal patterns
  - E.g. deviation from a steady growth
  - E.g. sales of winter wear go down in summer
    - Not surprising, part of a known pattern.
    - Look for deviation from value predicted using past patterns