Unit 3 :
Data Preprocessing
(3 Hrs.)

Compiled By: Madan Nath
BSc. CSIT 7th Semester

Contents:

- Data cleaning,
- Data integration and transformation,
- Data reduction,
- Data discretization and Concept Hierarchy Generation,
- Data mining primitives

Data Preprocessing:

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.

Major Tasks in Data Preprocessing !

- **Data cleaning:** Fill in missing values, smooth noisy data, identify or remove outliers and noisy data, and resolve inconsistencies
- **Data integration:** Integration of multiple databases, or files
- **Data transformation:** Normalization and aggregation
- **Data reduction:** Obtains reduced representation in volume but produces the same or similar analytical results
- **Data discretization** (for numerical data)

Data Cleaning !

Importance:

- Data cleaning is the number one problem in data warehousing.

Data cleaning tasks:

- Fill in missing values
- Identify outliers and smooth out noisy data
- Correct inconsistent data
- Resolve redundancy caused by data integration

Data Cleaning: 1) Missing Data

- **Data is not always available**:
  - E.g., many tuples have no recorded values for several attributes, such as customer income in sales data.
- **Missing data may be due to**
  - equipment malfunction
  - inconsistent with other recorded data and thus deleted
  - data not entered due to misunderstanding
  - certain data may not be considered important at the time of entry
  - no register history or changes of the data
  - expansion of data schema

Data Cleaning: 1) Missing Data

How to Handle Missing Data?

- Ignore the tuple (loss of information)
- **Fill in missing values manually:**
  - tedious, infeasible?
- Fill in it automatically with
  - **a global constant :** e.g., “unknown”, a new class
  - **the attribute mean**
  - **the most probable value:** inference-based such as Bayesian formula, decision tree, or EM algorithm

Data Cleaning: 2) Noisy Data

- **Noise:** random error or variance in a measured variable.
- **Incorrect attribute values may due to**
  - faulty data collection instruments
  - data entry problems
  - data transmission problems etc.
- **Other data problems which requires data cleaning**
  - duplicate records, incomplete data, inconsistent data

Data Cleaning: 2) Noisy Data

How to Handle Noisy Data?

- **Binning method:**
  - first sort data and partition into (equi-size) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- **Clustering**
  - detect and remove outliers
- **Combined computer and human inspection**
  - detect suspicious values and check by human (e.g., deal with possible outliers)

Data Cleaning: 2) Noisy Data

Binning Methods for Data Smoothing

1. **Sorted data for price (in dollars):**
    1. Eg. 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
2. **Partition into (equi-size) bins:**
    1. **Bin 1:** 4, 8, 9, 15
    2. **Bin 2:** 21, 21, 24, 25
    3. **Bin 3:** 26, 28, 29, 34
3. **Smoothing by bin means:** finding individual bin mean
    1. Bin 1: 9, 9, 9, 9
    2. Bin 2: 23, 23, 23, 23
    3. Bin 3: 29, 29, 29, 29
4. **Smoothing by bin boundaries:** near to the boundary
    1. Bin 1: 4, 4, 4, 15
    2. Bin 2: 21, 21, 21, 25
    3. Bin 3: 26, 26, 26, 34

Data Cleaning: 3) Outlier Removal

- Data points inconsistent with the majority of data.
- Different outliers
  - **Valid:** CEO's salary,
  - **Noisy:** One's age = 200, widely deviated points
- Removal methods
  - Clustering
  - Curve-fitting
  - Hypothesis-testing with a given model

Data Integration

- **Data integration:** combines data from multiple sources
- **Schema integration:**
  - integrate metadata from different sources
  - **Entity identification problem:** identify real world entities from multiple data sources, e.g., A.cust_id ≡ B.cust_#
- **Detecting and resolving data value conflicts**
  - for the same real world entity, attribute values from different sources are different, e.g., different scales, metric vs. British units
- **Removing duplicates and redundant data**

Data Transformation

Data transformation involves following steps:

- **Smoothing:** remove noise from data
- **Normalization:** scaled to fall within a small, specified range
- **Attribute/feature construction**
  - New attributes constructed from the given ones !
- **Aggregation:** summarization
  - Integrate data from different sources (tables)
- **Generalization:** concept hierarchy climbing

Data Transformation: 1) Smoothing

- It is a process that is used to remove noise from the dataset using some algorithms.
- It allows for highlighting important features present in the dataset.
- It helps in predicting the patterns. When collecting data, it can be manipulated to eliminate or reduce any variance or any other noise form.
- The concept behind data smoothing is that it will be able to identify simple changes to help predict different trends and patterns.
- This serves as a help to analyst's or traders who need to look at a lot of data which can often be difficult to digest for finding patterns that they wouldn't see otherwise.

Data Transformation: 2) Aggregation

- Data collection or aggregation is the method of storing and presenting data in a summary format.
- The data may be obtained from multiple data sources to integrate these data sources into a data analysis description.
- This is a crucial step since the accuracy of data analysis insights is highly dependent on the quantity and quality of the data used. Gathering accurate data of high quality and a large enough quantity is necessary to produce relevant results.
- The collection of data is useful for everything from decisions concerning financing or business strategy of the product, pricing, operations, and marketing strategies.
- For example: Sales, data may be aggregated to compute monthly& annual total amounts.

Data Transformation: 3) Discretization:

- It is a process of transforming continuous data into set of small intervals.
- Most Data Mining activities in the real world require continuous attributes. Yet many of the existing data mining frameworks are unable to handle these attributes.
- Also, even if a data mining task can manage a continuous attribute, it can significantly improve its efficiency by replacing a constant quality attribute with its discrete values.
- **For example:** (1-10, 11-20) (age:- young, middle age, senior).

Data Transformation: 4) Attribute Construction

- Where new attributes are created & applied to assist the mining process from the given set of attributes.
- This simplifies the original data & makes the mining more efficient.

Data Transformation: 5) Generalization:

- It converts low-level data attributes to high-level data attributes using concept hierarchy.
- **For Example:** Age initially in Numerical form (22, 25) is converted into categorical value (young, old).
- **For example:** Categorical attributes, such as house addresses, may be generalized to higher-level definitions, such as town or country.

Data Transformation: 6) Normalization:

- Data normalization involves converting all data variable into a given range.
- Techniques that are used for normalization are:

Min-Max Normalization:

- This transforms the original data linearly.
- Suppose that: min_A is the minima and max_A is the maxima of an attribute, P
- **For Example:**

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

  - Where v is the value you want to plot in the new range.
  - v' is the new value you get after normalizing the old value.

Data Transformation: 6) Normalization:

Min-Max Normalization:

- For example: Suppose the minimum and maximum value for an attribute profit(P) are Rs. 10, 000 and Rs. 100, 000. We want to plot the profit in the range [0, 1]. Using min-max normalization the value of Rs. 20, 000 for attribute profit can be plotted to:

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

$$\frac{20000 - 10000}{100000 - 10000}(1 - 0) + 0 = 0.11$$

- And hence, we get the value of v' as 0.11

Data Transformation: 6) Normalization:

z-score Normalization:

- In z-score normalization (or zero-mean normalization) the values of an attribute (A), are normalized based on the mean of A and its standard deviation.
- A value v, of attribute A is normalized to v' by computing:

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

Data Transformation: 6) Normalization:

z-score Normalization:

- **For example:** Let mean of an attribute P = 60, 000, Standard Deviation = 10, 000, for the attribute P. Using z-score normalization, a value of 85,000 for P can be transformed to:

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

$$\frac{85000 - 60000}{10000} = 2.50$$

- And hence we get the value of v' to be 2.5

Data Transformation: 6) Normalization:

Decimal Scaling:

- It normalizes the values of an attribute by changing the position of their decimal points
- The number of points by which the decimal point is moved can be determined by the absolute maximum value of attribute A.
- A value, v, of attribute A is normalized to v' by computing

$$v' = \frac{v}{10^j}$$

- where j is the smallest integer such that Max(|v'|) < 1.

Data Transformation: 6) Normalization:

Decimal Scaling:

- For example: S uppose: Values of an attribute P varies from -99 to 99.
- The maximum absolute value of P is 99.
- For normalizing the values we divide the numbers by 100 (i.e., j = 2) or (number of integers in the largest number) so that values come out to be as 0.98, 0.97 and so on.

Data Reduction Strategies

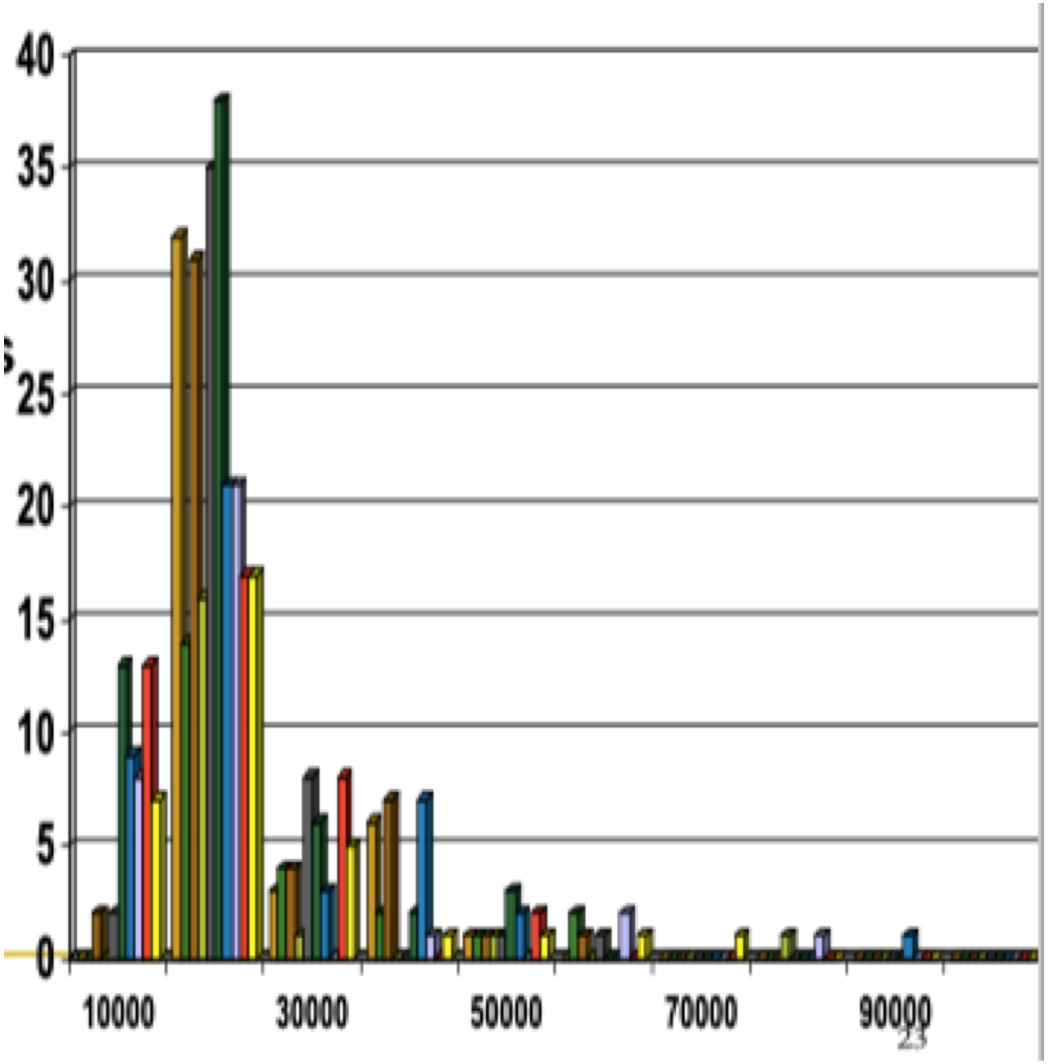
- **Data is too big to work with**
  - Too many instances
  - too many features (attributes) – curse of dimensionality
- **Data reduction**
- Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results (easily said but difficult to do)
- **Data reduction strategies**
  - **Dimensionality reduction** — remove unimportant attributes
  - **Aggregation and clustering –**
    - Remove redundant or close associated ones
  - **Sampling**

Dimensionality Reduction

- **Feature selection** (i.e., attribute subset selection):
- **Direct methods –**
  - Select a minimum set of attributes (features) that is sufficient for the data mining task.
- **Indirect methods –**
  - Principal component analysis (PCA)
  - Singular value decomposition (SVD)
  - Independent component analysis (ICA)
  - Various spectral and/or manifold embedding (active topics)
- **Heuristic methods (due to exponential # of choices):**
- step-wise forward selection
- step-wise backward elimination
- combining forward selection and backward elimination
  - Combinatorial search – exponential computation cost

Histograms

- A popular data reduction technique
- Divide data into buckets and store average (sum) for each bucket



Clustering

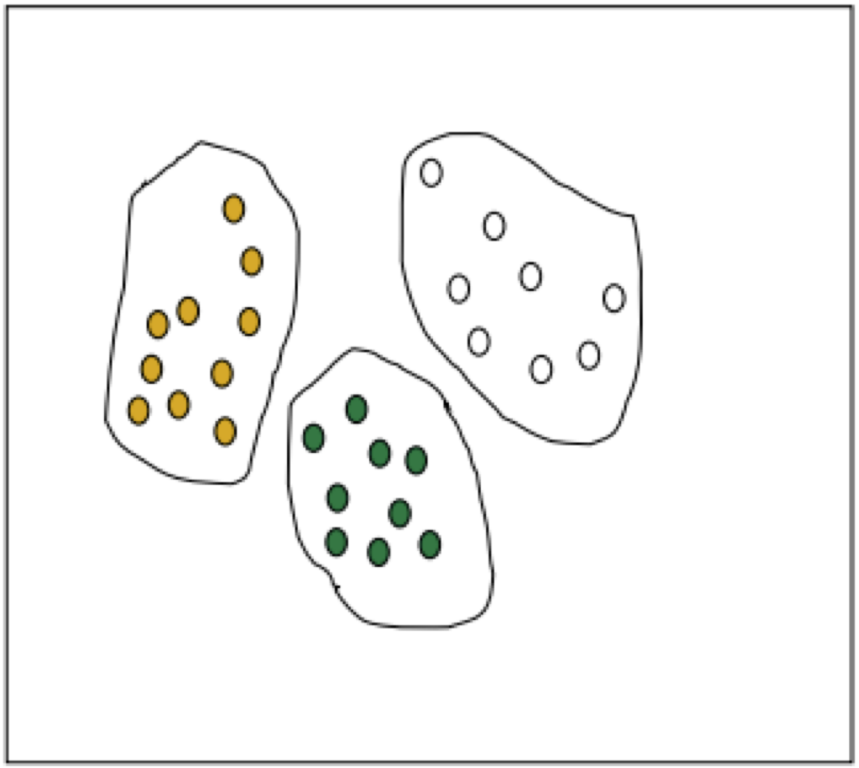
- Partition data set into clusters, and one can store cluster representation only
- Can be very effective if data is clustered but not if data is "smeared"
- There are many choices of clustering definitions and clustering algorithms. We will discuss them later.

Sampling

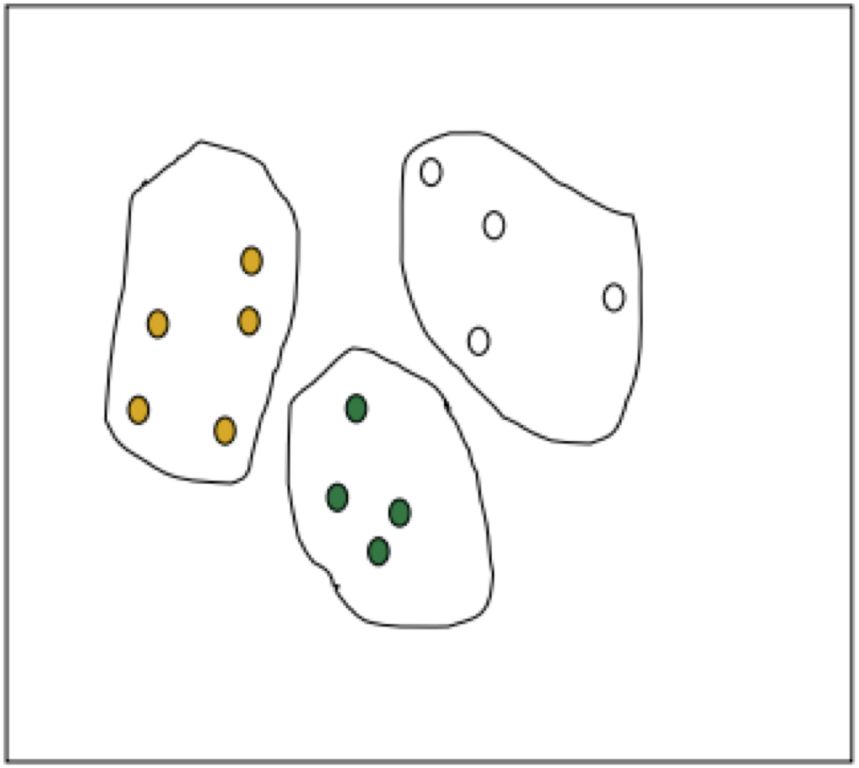
- **Choose a representative subset of the data**
  - Simple random sampling may have poor performance in the presence of skew.
- **Develop adaptive sampling methods**
  - Stratified sampling:
    - Approximate the percentage of each class (or subpopulation of interest) in the overall database
    - Used in conjunction with skewed data

Sampling

Raw Data



Cluster/Stratified Sample



Discretization

- **Three types of attributes:**
  - Nominal — values from an unordered set
  - Ordinal — values from an ordered set
  - Continuous — real numbers
- **Discretization:**
  - Divide the range of a continuous attribute into intervals because some data mining algorithms only accept categorical attributes.
  - It reduces the number of values for a given continuous attribute by dividing the range of the attribute into intervals. Interval labels can then be used to replace actual data values.

Two types of Discretization

Top-down discretization

- If the process starts by first finding one or a few points (called split points or cut points) to split the entire attribute range, and then repeats this recursively on the resulting intervals, then it is called top-down discretization or splitting.

Bottom-up discretization

- If the process starts by considering all of the continuous values as potential split-points, removes some by merging neighborhood values to form intervals, then it is called bottom-up discretization or merging.
- Discretization can be performed rapidly on an attribute to provide a hierarchical partitioning of the attribute values, known as a **concept hierarchy**.

Typical methods

1) Histogram Analysis

- Because histogram analysis does not use class information so it is an **unsupervised discretization technique**. Histograms partition the values for an attribute into disjoint ranges called buckets.

2) Cluster Analysis

- Cluster analysis is a popular data discretization method. A clustering algorithm can be applied to discrete a numerical attribute of A by partitioning the values of A into clusters or groups.
- Each initial cluster or partition may be further decomposed into several subcultures, forming a lower level of the hierarchy.

3) Binning

- Binning is a **top-down splitting technique** based on a specified number of bins. Binning is an unsupervised discretization technique.

Example: Binning

- **Attribute values (for one attribute e.g., age):**
  - 0, 4, 12, 16, 16, 18, 24, 26, 28
- **Equi-width binning** – for bin width of e.g., 10:
  - Bin 1: 0, 4 [-,10) bin
  - Bin 2: 12, 16, 16, 18 [10,20) bin
  - Bin 3: 24, 26, 28 [20,+) bin
  - - denote negative infinity, + denotes positive infinity
- **Equi-frequency binning** – for bin density of e.g., 3:
  - Bin 1: 0, 4, 12 [-, 14) bin
  - Bin 2: 16, 16, 18 [14, 21) bin
  - Bin 3: 24, 26, 28 [21,+] bin

Concept Hierarchy

- Concept hierarchies can be used to reduce the data by collecting and replacing low-level concepts with higher-level concepts.
- In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies. This organization provides users with the flexibility to view data from different perspectives.
- Data mining on a reduced data set means fewer input/output operations and is more efficient than mining on a larger data set.
- Because of these benefits, discretization techniques and concept hierarchies are typically applied before data mining, rather than during mining.

Data mining primitives

- **Data Mining:** Data Mining refers to extracting on mining knowledge from large amount of data.
- **Data Mining Primitives:** A data mining task can be specified in the form of a data mining query which is input to the data mining system.
- **A mining query is defined in terms of the following:**
  - Task-Relevant Data
  - The Kind Of Knowledge to be Mined
  - Background Knowledge: Concept Hierarchies
  - Interestingness Measures
  - Presentation and Visualization of Discovered Pattern

1) TASK-RELEVANT DATA

- The set of task relevant data can be collected a relational query involving operation like selection , projection , join and aggregation.
- The data collection process results in a new data relation called the initial data relation.
- The initial relation may or may not correspond to a physical relation in the database.
- Virtual relation are called views in the field of databases, the set of task-relevant data for data mining is called a minable view.

1) TASK-RELEVANT DATA

- The task-relevant data can be specified by providing the following information:
  - The names of the database or data warehouse to be used
  - The names of the tables or data cubes containing the relevant data
  - Condition for selection the relevant data
  - The relevant attributes or dimensions
  - The data retrieved be grouped by certain attributes , such as "grouped by data"
- The set of task relevant data can be specified by condition based data filtering ,slicing or dicing of the data cube
  - For eg: A concept hierarchy on item that specifies that "home entertainment " is at a higher concept level , composed of the lower concept level {"TV","CD player ", "VCR"} can be used in the collection of the task-relevant data.

2) THE KIND OF KNOWLEDGE TO BE MINED

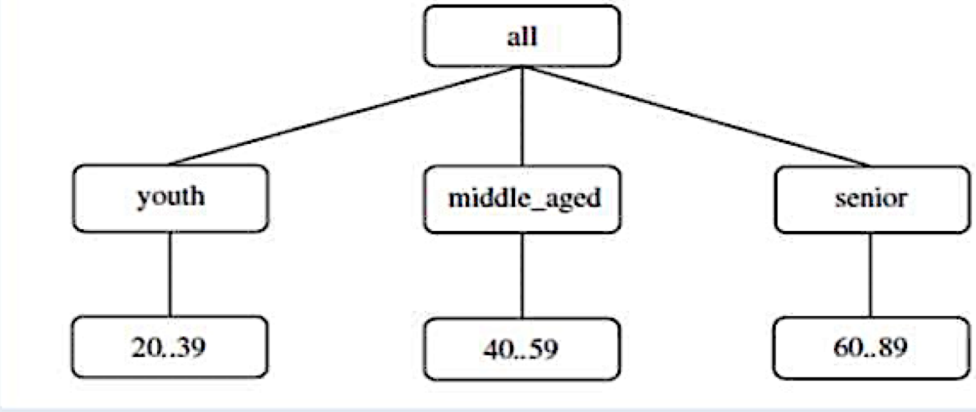
- The kinds of knowledge include concept description (characterization , discrimination), association , classification , prediction , clustering , and evolution analysis.
- These templates or metapatterns can be used to guide the discovery process.
- For eg:

age(X , “30…39”) ^ income(X, “40K…49K”) => buys (X,”VCR”)

[2.2%, 60%]

3) BACKGROUND KNOWLEDGE : CONCEPT HIERARCHIES

- Background knowledge is information about the domain to be mined that can be useful in the discovery process.
- Background knowledge known as concept hierarchies. concept hierarchies allows the discovery of knowledge at multiple levels of abstraction.
- concept hierarchies defines a sequence of mappings from a set of low-level concept to higher-level.



3) BACKGROUND KNOWLEDGE : CONCEPT HIERARCHIES

- Concept hierarchies is represented as a set of nodes organized in a tree , where each node, in itself , represents a concept.
- **There are four types of concept hierarchies:**
  - Schema hierarchies
  - Set grouping hierarchies
  - Operation-derived hierarchies
  - Rule –based hierarchies.

3) BACKGROUND KNOWLEDGE : CONCEPT HIERARCHIES

- Schema hierarchies: is a total or partial order among attributes in the database schema.
  - street < city < state < country
- Set grouping hierarchies: organizes a values for a given attribute or dimension into groups of constants or range values.
  - {young , middle-age) C all (age)
  - {20…39} C young
  - {40…59} C middle-aged
- Operation-derived hierarchies: include the decoding of information encoded string, information extraction from complex data objects.
  - login-name < department < university < country forming a email address.
- Rule –based hierarchies: set of rules and is evaluated dynamically based on the current database data and the rule definition.
  - low_profit_margin(X) <= price(X,P1) ^ cost (X,P2) ^ ((P1-P2) <$50)

4) INTERESTINGNESS MEASURES

- The number of uninteresting patterns returned by the process. This can be achieved by specifying interestingness measure that estimate the
  - simplicity,
  - certainty,
  - utility and
  - novelty
- Each measure is associated with a threshold that can be controlled by the user.

4) INTERESTINGNESS MEASURES

SIMPLICITY:

- Simplicity can be viewed as functions of the pattern structure defined in terms of the pattern size in bits or the number of attributes or operators appearing in the pattern. for eg: rule length.

CERTAINTY:

- Each discovery pattern should have a measure of certainty associated with it that assesses the validity or trustworthiness of the pattern. A certainty measure for associated rules of the form "A=>B", where A and B are set of items, is confidence.

- $$\text{confidence}(A \Rightarrow B) = \frac{\#\_\text{tuples\_containing\_both\_A\_and\_B}}{\#\_\text{tuples\_containing\_A}}$$

4) INTERESTINGNESS MEASURES

UTILITY:

- It can be estimated by a utility function such as support. The support of an associated pattern refers to the percentage of task-relevant data tuples for which the pattern is true. For associated rules of the form "A=>B" where A and B are set of items,
- $$\text{support}(A \Rightarrow B) = \frac{\#\_\text{tuples\_containing\_both\_A\_and\_B}}{\text{total\_\#\_of\_tuples}}$$

NOVELTY:

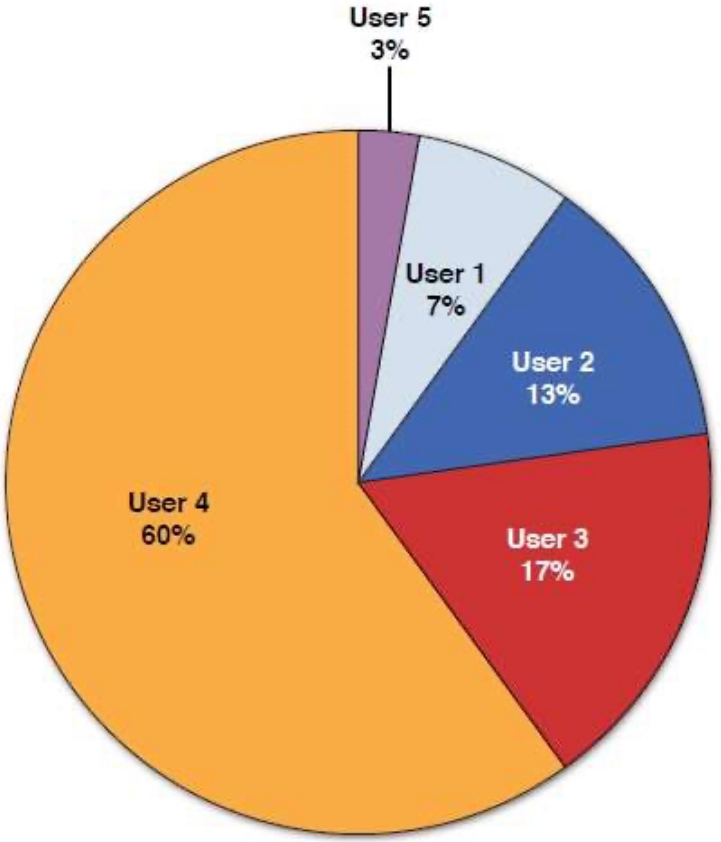
- It contribute new information or increased performed to the given pattern set. Novelty is removed redundant patterns.
- For eg: a data exception may be considered novel in it differs from that based on statistical model or user beliefs.
- location(X, "CANADA") => buys(X, "SONY_TV") [8%, 70%]

5) PRESENTATION AND VISUALIZATION OF DISCOVERED PATTERNS

- Data mining system should be able to display the discovery patterns in multiple patterns such as rules, tables, crosstabs, pie charts, decision tree, cubes, or other visual representations.
- Data mining system should employ concept hierarchies to implement drill-down and roll-up operation. So that users may discovery patterns at multiple levels of abstraction.
- In addition pivoting, slicing and dicing operation, the user in viewing generalized data and knowledge from different perspective.

5) PRESENTATION AND VISUALIZATION OF DISCOVERED PATTERNS

Various form of presenting and visualizing the discovered pattern:



Thank you !!!