Course : Data Engineering Techniques[AID30010]

MIT-WPU
।। विश्वशान्तिर्ध्रुवं ध्रुवा ।।

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

B Tech CSE AI DS
SY Sem IV

Disclaimer:

1. Information included in this slides came from multiple sources. We have tried our best to cite the sources. Please refer to the References to learn about the sources, when applicable.
2. The slides should be used only for academic purposes (e.g., in teaching a class), and should not be used for commercial purposes.

Course : Data Engineering Concepts [AID2PR03A]

TEAM: DET Panels : A,B,C,D,E,F

Dr. Devendra Joshi
Panel-A, D

Course : Data Engineering Techniques [AID30010]

Course Objectives:

1. To study the concepts of data engineering
2. To understand classification, clustering and association rule mining.
3. To learn the concepts and techniques of data warehousing
4. To understand the various data mining techniques for classification and clustering
5. To implement data pre-processing on any dataset.
6. To perform predictive analysis for any live data.

Course Outcomes:

On completion of course, students should be able to

1. Understand and apply data engineering concepts
2. Apply data pre-processing techniques on real world data
3. Apply the appropriate data warehousing architecture and tools for systematically organizing database
4. Analyse algorithms for classification and clustering for solving real world problems in data engineering

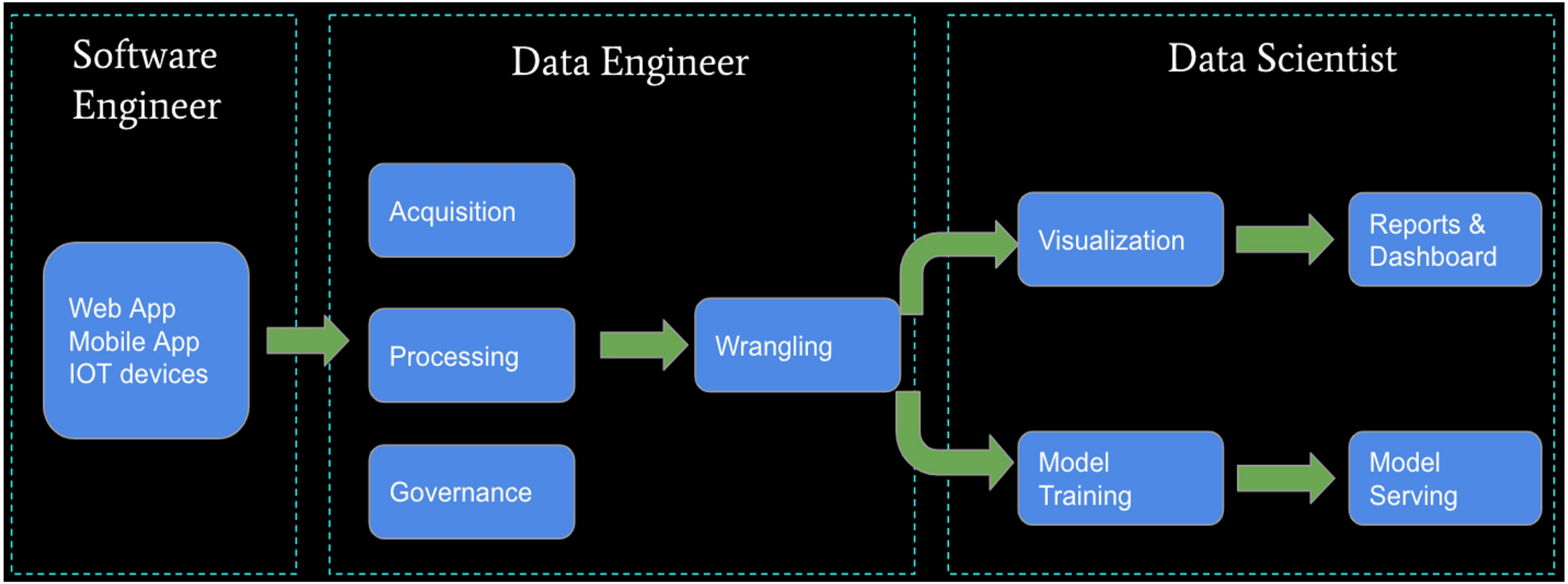
Course : Data Engineering Techniques [AID30010]

- Mid-term exam: 9-20 March, 2025.
- Term-end exam: 14 May-16 June, 2025. (as per Academic Calendar)

UNIT I - AN OVERVIEW

- Introduction to Data Engineering and Preprocessing
- Defining Data Engineering, Overview of the Data Engineering Ecosystem, Data Characteristics
- Types of Data: Structured, Unstructured, Semi-structured, Discrete, Continuous, Ordinal, Nominal, Qualitative, Quantitative, Time series data, Geographical data.
- Data Preprocessing: Data Cleaning, Data Integration, Data Reduction, Data Transformation, Data Discretization,
- Data Cleaning: Handling Missing Values, Noisy Data,
- Data Integration: Entity Identification problem, Redundancy and Correlation Analysis, Tuple Duplication
- Data Reduction: Attribute Subset Selection, Histograms, Sampling

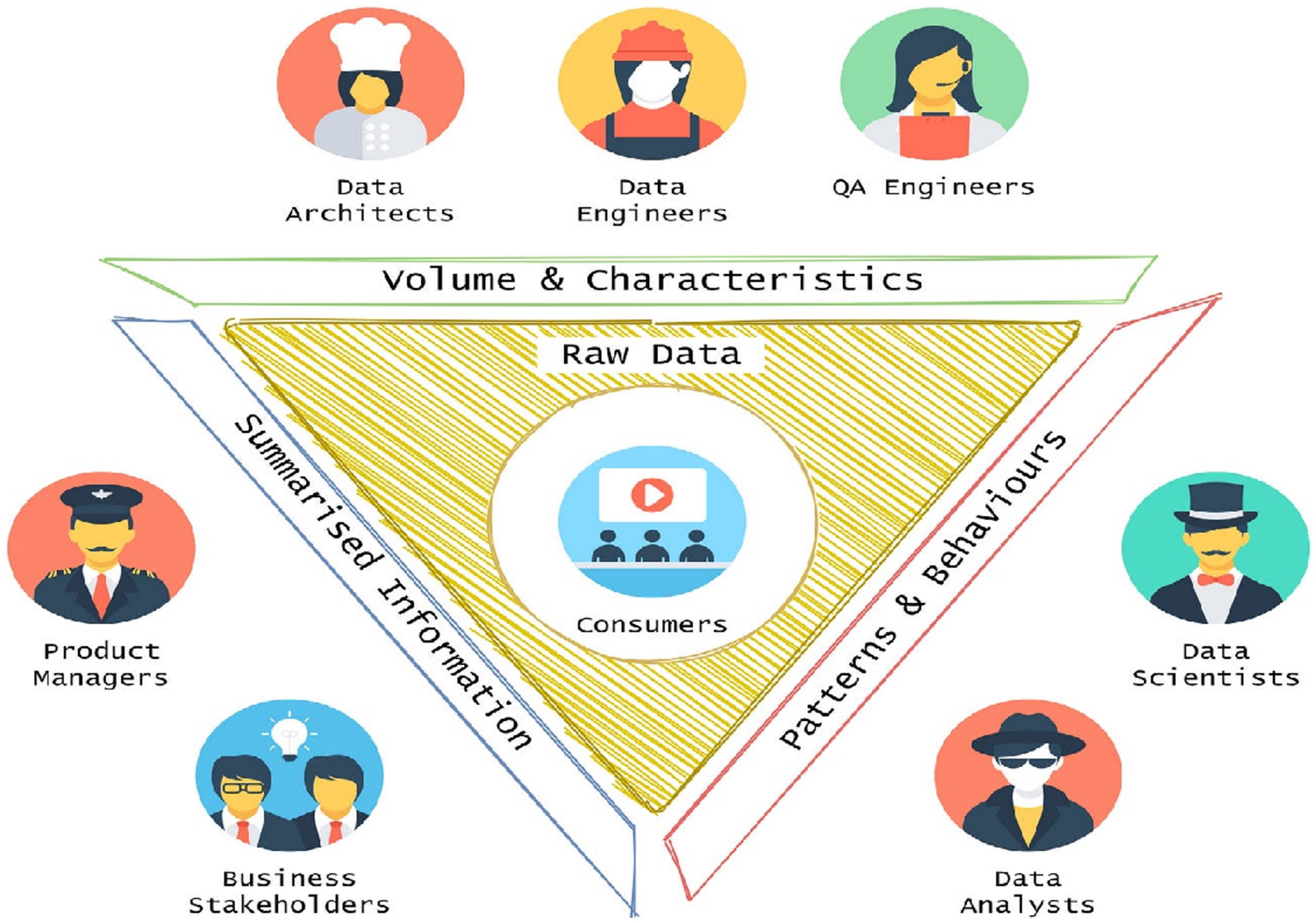
Introduction to Data Engineering



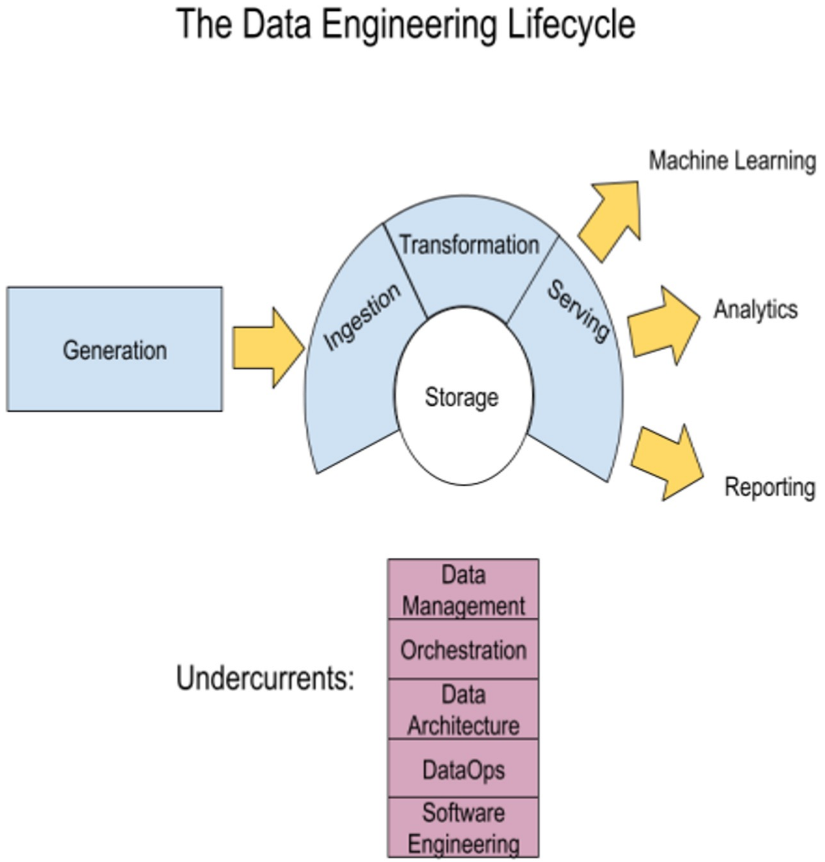
Introduction to Data Engineering

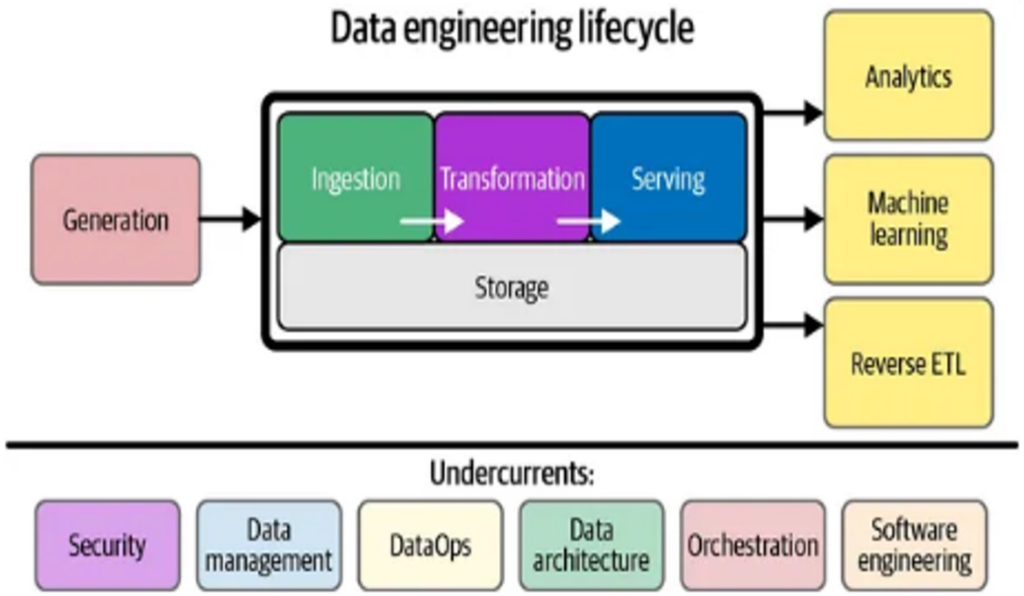
- Data engineering: The development, implementation, and maintenance of systems and processes that take in raw data and produce high-quality, consistent information and knowledge that supports downstream use cases, such as analysis and machine learning.
- Data engineering is the intersection of data management, data architecture, software engineering.
- Data engineers Manage the data engineering lifecycle beginning with ingestion and ending with serving data for use cases, such as analysis or machine learning.

Data Engineering Ecosystem



Data Engineering Lifecycle



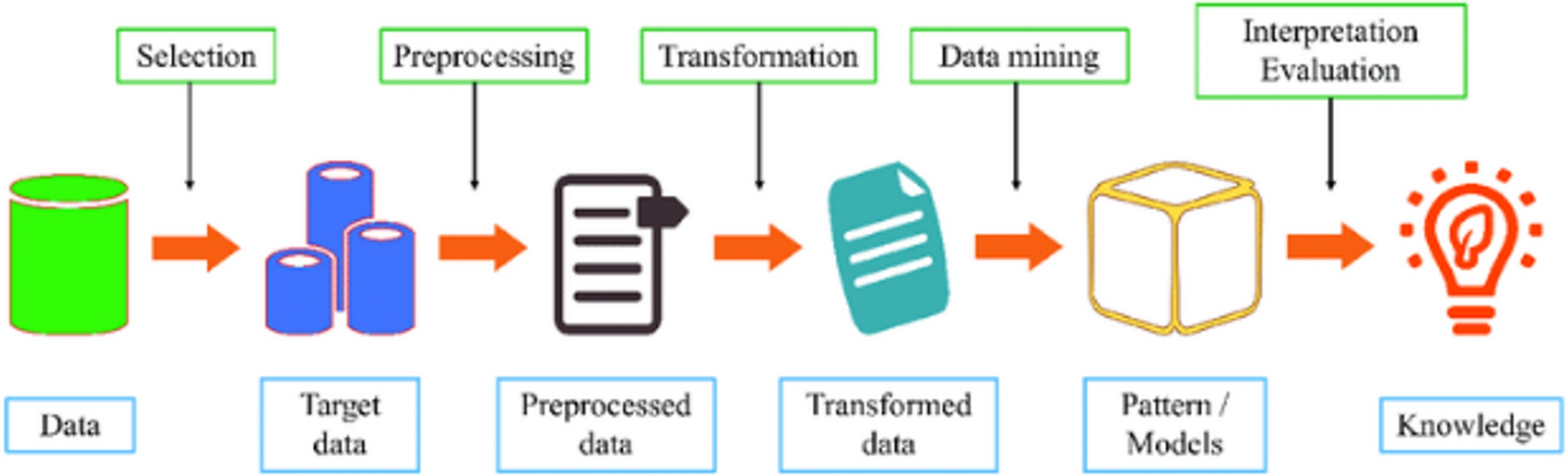


Evolution of Database Technology

- 1960s: Data collection, database creation, IMS and network DBMS
- 1970s: Relational data model, relational DBMS implementation
- 1980s:
  - RDBMS, advanced data models (extended-relational, OO, deductive, etc.)
  - Application-oriented DBMS (spatial, scientific, engineering, etc.)
- 1990s: Data mining, data warehousing, multimedia databases, and Web databases
- 2000s
  - Stream data management and data mining
  - Data mining, applications and Web technology

What is Data Mining?

- Data mining is also called *knowledge discovery and data mining*(KDD)
- Data mining is
  - Extraction of useful patterns from data sources, e.g.- Databases, texts, web, image
- Patterns must be:
  - Valid, novel, potentially useful, understandable



What Is Data Mining?

- Data mining (knowledge discovery from data)
  - Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data
  - Data mining: a misnomer?
- Alternative names
  - Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.
- Watch out: Is everything "data mining"?
  - Simple search and query processing
  - (Deductive) expert systems

Why is Data Mining important?

- Rapid computerization of businesses produce huge amount of data
- How to make best use of data?
- It is used to discover patterns and relationships in the data in order to help make better business decisions
- Data mining technology can generate new business opportunities by:
  - *Automated prediction of trends and behaviors*
  - *Automated discovery of previously unknown patterns*

Why Data Mining?

The Explosive Growth of Data: from terabytes to petabytes

- **Data and storage predictions for the year 2025(zettabytes)**
  - The storage industry will ship 42ZB of capacity over the next seven years.
  - 90ZB of data will be created on IoT devices by 2025.
  - By 2025, 49 percent of data will be stored in public cloud environments.
  - Nearly 30 percent of the data generated will be consumed in real-time by 2025.

Why Data Mining?

The Explosive Growth of Data: from terabytes to zettabytes

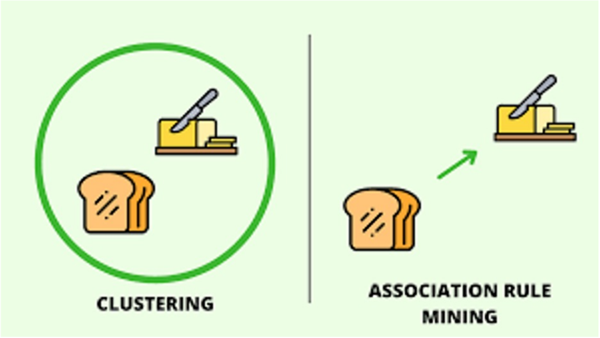
- Data collection and data availability
  - Automated data collection tools, database systems, Web, computerized society
- Major sources of abundant data
  - Business: Web, e-commerce, transactions, stocks, …
  - Science: Remote sensing, bioinformatics, scientific simulation, …
  - Society and everyone: news, digital cameras, YouTube
- <u>We are drowning in data, but starving for knowledge!</u>
- "Necessity is the mother of invention"—Data mining—Automated analysis of massive data sets

Example of Discovered Patterns

- Association rules:

  "80% of customers who buy *cheese* and *milk* also buy *bread*, and 5% of customers buy all of them together"

  Cheese, Milk→ Bread



Data mining Algorithms: can be characterized as consisting of three parts

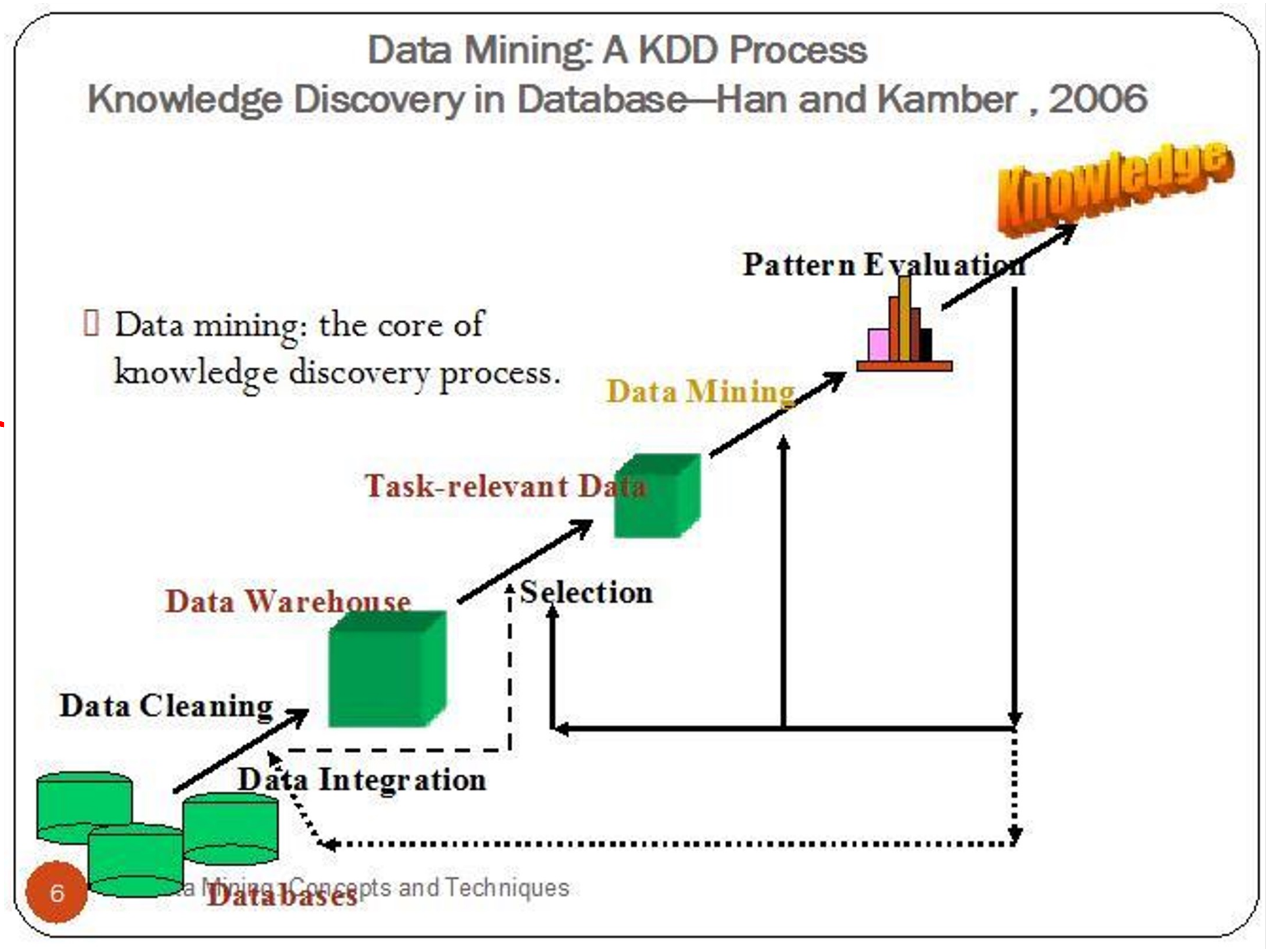
- Model: is to be fit on data.
- Preference: criteria to select one model over other
- Search: techniques to evaluate data point or searching of data.

Goal of Data Mining

- Goal of Data Mining is to provide
  efficient tools and techniques for KDD

Knowledge Discovery (KDD) Process

- This is a view from typical database systems and data warehousing communities
- Data mining plays an essential role in the knowledge discover process



Knowledge Discovery (KDD) Process

1. **Developing an understanding of:**
   - The application domain
   - The relevant prior knowledge
   - The goals of the end-user
   - Creating a target data set: selecting a data set, or focusing on a subset of variables, or data samples, on which discovery is to be performed.

2. **Data cleaning and preprocessing.**
   - Removal of noise or outliers.
   - Collecting necessary information to model or account for noise.
   - Strategies for handling missing data fields.
   - Accounting for time sequence information and known changes.

3. **Data reduction and projection**
   - Finding useful features to represent the data depending on the goal of the task.
   - Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.

Knowledge Discovery (KDD) Process

4. Choosing the data mining task.
    - Deciding whether the goal of the KDD process is classification, regression, clustering, etc.
5. Choosing the data mining algorithm(s)
    - Selecting method(s) to be used for searching for patterns in the data.
    - Deciding which models and parameters may be appropriate.
    - Matching a particular data mining method with the overall criteria of the KDD process.
6. Data mining.
    - Searching for patterns of interest in a particular representational form or a set of such representations as classification rules or trees, regression, clustering, and so forth.
7. Interpreting mined patterns.
8. Consolidating discovered knowledge.

Sequence of the steps

- **Data Cleaning:** To remove noise and inconsistent data.
- **Data Integration:** where multiple data sources may be combined.
- **Data Selection:** Where data relevant to the analysis task are retrieved from the database.
- **Data Transformation:** where data transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations.
- **Data Mining:** An essential process where intelligent methods are applied to extract data patterns.
- **Pattern evaluation:** To identify the truly interesting pattern representing knowledge based on interesting measures.
- **Knowledge Presentation:** where visualization and knowledge representation tech. are used to present mined knowledge to users.

Applications of Data Mining

- Data mining applications are widely used in
  - Direct marketing
  - Health industry
  - E-commerce
  - Customer relationship management (CRM)
  - Telecommunication industry and financial sector, etc..
- Data mining is available in various forms
  - Text mining
  - Web mining
  - Audio & video data mining
  - Pictorial data mining
  - Relational databases data mining
  - Social networks data mining

Data Warehouse and Data Mining

- Introduction to KDD
- **Data Preprocessing: An Overview**
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary

Why Is Data Preprocessing Important?

- No quality data, no quality mining results!
  - **Quality** decisions must be based on quality data
  - e.g., duplicate or missing data may cause incorrect or even misleading statistics.
- Data preparation, cleaning, and transformation comprises the majority of the work in a data mining application (90%).

Types of Data :Structured

- Structured data is data with a high degree of organization, usually stored in some sort of spreadsheet. Excel sheet, which is a prime example of structured data.

CUSTOMER

CUSTOMER_ID	LAST_NAME	FIRST_NAME	STREET	CITY	ZIP_CODE	COUNTRY
10302	Boucher	Leo	54, rue Royale	Nantes	44000	France
11244	Smith	Laurent	8489 Strong St	Las Vegas	83030	USA
11405	Han	James	636 St Kilda Road	Sydney	3004	Australia
11993	Mueller	Tomas	Berliner Weg 15	Tamm	71732	Germany
12111	Carter	Nataly	5 Tomahawk	Los Angeles	90006	USA
14121	Cortez	Nola	Av. Grande, 86	Madrid	28034	Spain
14400	Brown	Frank	165 S 7th St	Chester	33134	USA
14578	Wilson	Sarah	Seestreet #6101	Emory	1734	USA
14622	Jones	John	71 San Diego Ave	Arlington	69004	USA


```
productCode;orderNumber;quantityOrdered;priceEach;orderLineNumber;customerNumber;quantityInStock;buyPrice;MSRP;month;year;profit
S10_1678;287441.0;1057.0;2384.8799999999997;152.0;6498.0;222124;1366.6799999999992;2679.5999999999995;203.0;56108.0;1312.920000000000
S10_1949;287289.0;961.0;5524.66;162.0;7416.0;204540;2760.239999999999;6000.400000000003;200.0;56107.0;3240.159999999998
S10_2016;287432.0;999.0;3080.5299999999993;140.0;6885.0;185500;1931.72;3330.320000000001;202.0;56108.0;1398.6000000000006
S10_4698;287436.0;985.0;4824.07;142.0;6337.0;156296;2548.56;5422.479999999998;191.0;56109.0;2873.92
S10_4757;287364.0;1030.0;3478.8799999999997;195.0;8362.0;91056;2399.0400000000004;3808.0;196.0;56108.0;1408.959999999999
S10_4962;287304.0;932.0;3690.5700000000006;166.0;7072.0;190148;2895.760000000001;4136.7199999999975;202.0;56107.0;1240.9600000000003
S12_1099;277075.0;933.0;4656.049999999999;179.0;7313.0;1836;2574.1800000000003;5253.389999999999;195.0;54103.0;2679.21
S12_1108;276928.0;1019.0;5051.61;182.0;6109.0;97713;2580.9300000000003;5610.600000000003;189.0;54103.0;3029.6700000000005
S12_1666;287296.0;972.0;3453.6599999999994;171.0;7039.0;44212;2181.200000000001;3826.760000000001;201.0;56107.0;1645.5599999999995
S12_2823;287447.0;1028.0;3700.7300000000005;164.0;6971.0;279916;1855.5599999999997;4217.359999999998;194.0;56109.0;2361.799999999999
S12_3148;276923.0;963.0;3719.6199999999994;178.0;6084.0;186462;2406.7800000000007;4079.159999999999;188.0;54103.0;1672.380000000001
S12_3380;277079.0;925.0;2879.6300000000006;169.0;7350.0;246321;2029.3200000000008;3170.880000000001;195.0;54103.0;1141.5599999999995
S12_3891;276931.0;965.0;4271.910000000001;144.0;5950.0;28323;2242.35;4671.540000000002;190.0;54103.0;2429.1899999999996
S12_3990;277077.0;900.0;1911.22;223.0;7380.0;152901;861.8399999999998;2154.5999999999995;195.0;54103.0;1292.7600000000002
S12_4473;287330.0;1056.0;2909.259999999999;174.0;6450.0;171500;1559.6000000000008;3318.0;205.0;56107.0;1758.3999999999992
S12_4675;277085.0;992.0;2802.9999999999995;204.0;7735.0;197721;1585.7100000000003;3109.3199999999993;195.0;54103.0;1523.6100000000001
S18_1097;287292.0;999.0;2954.0900000000006;165.0;7321.0;73164;1633.2399999999996;3266.760000000001;200.0;56107.0;1633.5199999999995
S18_1129;277103.0;947.0;3275.25;165.0;7135.0;107325;2254.7700000000004;3821.5799999999995;195.0;54103.0;1566.8099999999995
S18_1342;287264.0;1111.0;2578.7799999999997;172.0;6865.0;243404;1697.3599999999988;2876.719999999998;198.0;56107.0;1179.3599999999997
S18_1367;287267.0;960.0;1340.1899999999998;157.0;7391.0;241780;679.2799999999999;1509.4800000000002;198.0;56107.0;830.1999999999996
S18_1589;256370.0;914.0;2792.4400000000005;160.0;6375.0;226050;1649.0000000000005;3111.000000000001;181.0;50094.0;1462.0000000000002
S18_1662;287387.0;1040.0;3894.9500000000007;172.0;9255.0;149240;2163.56;4415.32;197.0;56108.0;2251.7600000000007
S18_1749;256169.0;918.0;3842.0;176.0;7192.0;68100;2167.5000000000005;4250.0;172.0;50093.0;2082.4999999999995
S18_1889;277087.0;972.0;1850.3100000000002;175.0;7431.0;238302;1455.3000000000004;2079.0;195.0;54103.0;623.7000000000003
S18_1984;277105.0;917.0;3493.7099999999987;155.0;7137.0;263844;2535.03;3840.75;195.0;54103.0;1305.7199999999996
S18_2238;287333.0;986.0;4036.0;184.0;6483.0;132272;2842.2800000000016;4584.439999999999;205.0;56107.0;1742.1600000000003
S18_2248;256171.0;832.0;1357.3199999999997;160.0;7176.0;13500;832.4999999999997;1513.4999999999995;172.0;50094.0;681.0000000000001
S18_2319;287325.0;1053.0;3112.4600000000005;215.0;6300.0;231224;2096.0799999999986;3436.44;205.0;56107.0;1340.3599999999997
S18_2325;287246.0;957.0;3198.6099999999997;180.0;7452.0;261912;1637.4400000000003;3559.640000000002;198.0;56107.0;1922.200000000001
S18_2432;287310.0;998.0;1571.5199999999995;206.0;6512.0;56504;697.7599999999999;1701.5599999999997;202.0;56107.0;1003.8000000000004
S18_2581;287412.0;917.0;2112.86;175.0;7576.0;27776;1372.0;2365.44;200.0;56108.0;993.4400000000004
```

Data Types of Structured Data

<u>Data types in structed data:</u>

A structured data can have multiple data types in it. Let us look at a few of them.

- **Integers** – The data can be of integer or numeric format, which is a whole number. Example – Age, Number of runs scored, Number of cars owned.
- **Decimals** – The data can be of decimal type which is also known as fractions. Example – CGPA, Price.
- **Text** – The data can be of text format. Examples are Name, Location, Company Name.
- **Date** – The data can be of data format. It can of any data format depending on the problem statement. Example DOB, Order data

Semi structured Data

- Semi-structured is data which has some degree of organization in it.
- It is not as rigorously structured as structured data, but also not as messy as unstructured data.
- This degree of organization is typically achieved with some sort of tags or other elements with defined properties which introduce a hierarchy and system into a file.
- However, the order and amount of such structuring tags and elements may vary.
- Therefore, the structure imposed on a dataset it not as rigorous as in structured datasets where all data has to conform to the structure of the data table (spreadsheet).

```
{
    "EMPLOYEES": {
        "SALES": {
            "648229": {
              "NAME" : "Olivia Johnson"
              "DOB" : "1989-08-08"
            },
            "648666": {
              "NAME" : "Frank Mueller"
              "DOB" : "1985-05-11"
              "MISC" : "On paternal leave from 2019-01-01 until 2020-01-01"
            }
        }
    }
}
```

Unstructured data

- Unstructured data is data with no pre-defined organizational form or specific format. Or in other words, unstructured data is any data which is not structured or semi-structured.

Mercedes 300 SL Coupe

The commercial version of the Mercedes 300SL was first unveiled in 1954. This successful model then went into mass production soon after. With only 1,400 manufactured coupés in total, the gullwing was not an economical success, however.

Difference

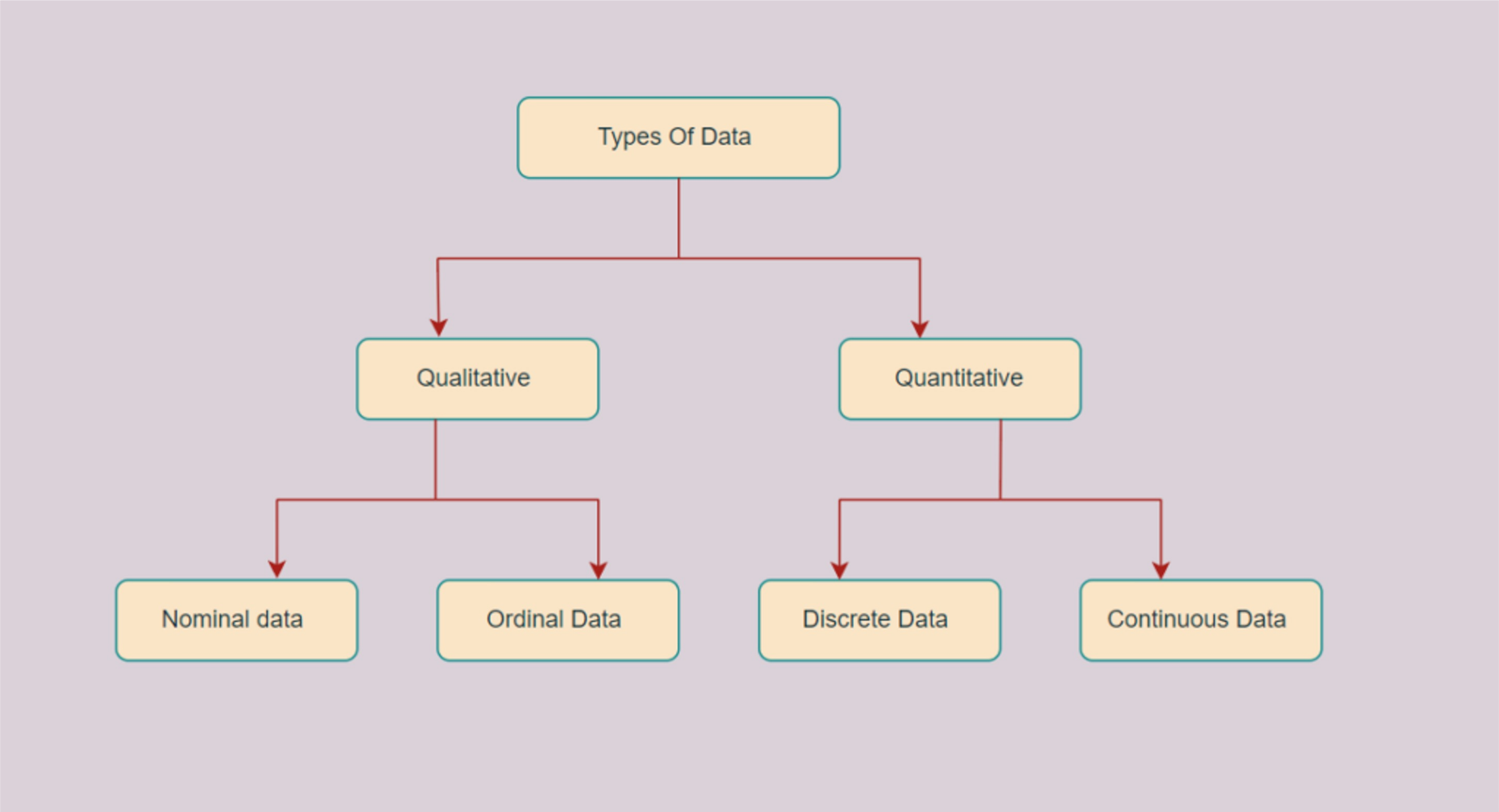
	Structured data	Semi-structured data	Unstructured data
What is it?	Data with a high degree of organization, typically stored in a spreadsheet-like manner	Data with some degree of organization	Data with no predefined organizational form and no specific format
To put it simply	Think of a spreadsheet (e.g. Excel) or data in a tabular format	Think of a TXT file with text that has some structure (headers, paragraphs, etc.)	Essentially anything that is not structured or semi-structured data (which is a lot)
Example formats	• Excel spreadsheets • Comma-separated value file (.csv) • Relational database tables	• Hypertext Markup Language (HTML) files • JavaScript Object Notation (JSON) files • Extensible Markup Language (XML) files	• Images such as .jpeg or .png files • Videos such as .mp4 or m4a files • Sound files such as .mp3 or .wav files • Plain text files • Word files • PDF files
Characteristics	• Data is structured in a spreadsheet-like manner (e.g. in a table) • Within that table, entries have the same format and a predefined length and follow the same order • Is easily machine-readable and can therefore be analysed without major pre-processing of the data • It is commonly said that around 20% of the world's data is structured	• Data is stored in files that have some degree of organization and structure • Tags or other markers separate elements and enforce hierarchies, but the size of elements can vary and their order is not important • Needs some pre-processing before it can be analysed by a computer • Has gained importance with the emergence of the World Wide Web	• Data that can take any form and thus be stored as any kind of file (formless) • Within that file, there is no structure of content • Typically needs major pre-processing before it can be analysed by a computer, but often easily consumable for humans (e.g. pictures, videos, plain texts) • Most of the data that is created today is unstructured

Structured vs Unstructured data:

Let us look at a few major differences between structured and unstructured data.

Structured data	Unstructured data
It is a quantitative data	It is a qualitative data
Easy to perform data analysis	Should be processed into proper form before performing analysis
Less than 20% of the total data in the world are in structured form	More than 80% of the total data in the world are in unstructured form
Can be either numeric or text	Can be text, numeric, images, videos, audio, and many more

Discrete, Continuous, Ordinal, Nominal, Qualitative, Quantitative , Time series data, Geographical data.



Discrete, Continuous, Ordinal, Nominal, Qualitative, Quantative , Time series data, Geographical data.

Qualitative or Categorical Data
Qualitative or Categorical Data is data that can't be measured or counted in the form of numbers.
These types of data are sorted by category, not by number.
These data consist of audio, images, symbols, or text. The gender of a person, i.e., male, female, or others, is qualitative data.
Qualitative data tells about the perception of people. This data helps market researchers understand the customers' tastes and then design their ideas and strategies accordingly.

Nominal Data
Nominal Data is used to label variables without any order or quantitative value. The color of hair can be considered nominal data, as one color can't be compared with another color.
The name "nominal" comes from the Latin name "nomen," which means "name." With the help of nominal data, we can't do any numerical tasks or can't give any order to sort the data. These data don't have any meaningful order; their values are distributed into distinct categories.

Ordinal Data
Ordinal data have natural ordering where a number is present in some kind of order by their position on the scale. These data are used for observation like customer satisfaction, happiness, etc., but we can't do any arithmetical tasks on them.

Quantitative Data
Quantitative data can be expressed in numerical values, making it countable and including statistical data analysis. These kinds of data are also known as Numerical data. It answers the questions like "how much," "how many," and "how often."
Quantitative data can be used for statistical manipulation. These data can be represented on a wide variety of graphs and charts, such as bar graphs, histograms, scatter plots, boxplots, pie charts, line graphs, etc.

Q1: Discrete Data
The term discrete means distinct or separate. The discrete data contain the values that fall under integers or whole numbers. The total number of students in a class is an example of discrete data. These data can't be broken into decimal or fraction values.
The discrete data are countable and have finite values; their subdivision is not possible. These data are represented mainly by a bar graph, number line, or frequency table.
Examples of Discrete Data :
Total numbers of students present in a class
Cost of a cell phone

Q2:**Continuous Data**
Continuous data are in the form of fractional numbers. It can be the version of an android phone, the height of a person, the length of an object, etc. Continuous data represents information that can be divided into smaller levels. The continuous variable can take any value within a range. **Examples of Continuous Data :** Height of a person

Discrete Vs Continuous Data

Discrete Data	Continuous Data
Discrete data are countable and finite; they are whole numbers or integers	Continuous data are measurable; they are in the form of fractions or decimal
Discrete data are represented mainly by bar graphs	Continuous data are represented in the form of a histogram
The values cannot be divided into subdivisions into smaller pieces	The values can be divided into subdivisions into smaller pieces
Discrete data have spaces between the values	Continuous data are in the form of a continuous sequence
Examples: Total students in a class, number of days in a week, size of a shoe, etc	**Example:** Temperature of room

Time Series Data

Time period	Sales
1/1/2021	124
1/2/2021	153
1/3/2021	130
1/4/2021	158
1/5/2021	120
1/6/2021	174
1/7/2021	155
1/8/2021	146
1/9/2021	126
1/10/2021	145
1/11/2021	155
1/12/2021	172
1/13/2021	198

- When the data is collected over equal intervals of time, it is called as time series data.
- The time intervals can be as small as seconds to as large as years. The only condition is the time-period should be of equal intervals and it should be clearly defined.
- A few examples of time series data are the stock prices of a company throughout the year, the sales data of a company for the past three years, the daily expenses a company has incurred over the past 6 months.
  - Characteristics of a time series data:
    Time series data has 4 characteristics, let us look at them one by one.
    1) Equal interval – As mentioned above, the time series data should be collected in equal interval of time and should not be uneven.
    2) Seasonality – When the time series data is influenced by seasonal factors such as rainfall, temperature, etc.
    3) Cyclicity - When the time series data is rising and falling in intervals those are not at fixed periods such as business cycles.
    4) Trend - When there is a long tern increase or decrease in the data. Based on the movement of data it can be called either an uptrend or a downtrend.

Understanding Data Attribute Types with

- For a customer object, attributes can be customer-id, address, etc
- First step before Data pre-processing - differentiate between different types of attributes and then pre-process the data.

- **Data attribute Types**
  - **Qualitative**
    - **Nominal**
    - **Ordinal**
    - **Binary**
  - **Quantitative**
    - **Numeric**

Qualitative Attributes

1. **Nominal Attributes –**
   - Values are name of things or some kind of symbols
   - Values of Nominal attributes represents some category or state and that’s why nominal attribute also referred as **categorical attributes** and
   - There is no order (rank, position) among values of nominal attribute

Attribute	Values
Colours	Black, Brown, White

Qualitative Attributes

2. **Binary Attributes :** Binary data has only 2 values/states.

For Example yes or no, true or false.

i. **Symmetric :** Both values are equally important (Gender). No preference on which should be coded as 0 or 1

ii. **Asymmetric :** Both values are not equally important. Most important outcome is coded as 1

Attribute	Values
Gender	Male , Female

Attribute	Values
Cancer detected	Yes, No
result	Pass , Fail

Qualitative Attributes

3. **Ordinal Attributes :**

- Values that have a meaningful sequence or ranking(order) between them
- But magnitude of values is not actually known

Attribute	Value
Grade	A,B,C,D,E,F

Quantitative Attributes

Numeric :

- A numeric attribute is quantitative because, it is a measurable quantity, represented as integer or real values.
- Numerical attributes are of 2 types, **interval** and **ratio**.

i) Interval-scaled attributes

- Have order
- Values can be added and subtracted but cannot be multiplied or divided
- Eg. Temperature of 10 degree Celsius should not be considered as twice hot as 5 degree Celsius since 10 degree Celsius is 50 degree Fahrenheit and 5 degree Celsius is 41 degree Fahrenheit which isn’t twice

Quantitative Attributes(Contd..)

- Interval data always appears in the form of numbers or numerical values where the distance between the two points is standardized and equal
- Eg. difference between 100 degrees Fahrenheit and 90 degrees Fahrenheit is the same as 60 degrees Fahrenheit and 70 degrees Fahrenheit.
- Don’t have a true zero

ii) **Ratio-scaled attributes**

- Has all properties of interval-scaled
- Have a true zero
- Values can be added , subtracted , multiplied & divided
- Eg. Weight, height,etc

Geographical data

- Geographical data is a type of information that captures the physical features of a place.
- This can include things like the location of a city or town, the layout of its streets and highways, and its natural features such as rivers and mountains
- Geographical data can be used for a variety of purposes, including mapping out areas for planning purposes.
- Geographical data is often used by social media networks to provide information about users' location.
- Construction companies may use geographical data to map out the locations of roads and highways
- Geographical data is often presented in a two-dimensional format on a paper map or globe. However, geographical data can also be represented in three dimensions through digital means such as virtual globes.
- Geographic information systems (GIS) are software applications that allow users to view, edit, and analyze geographical data in three dimensions.

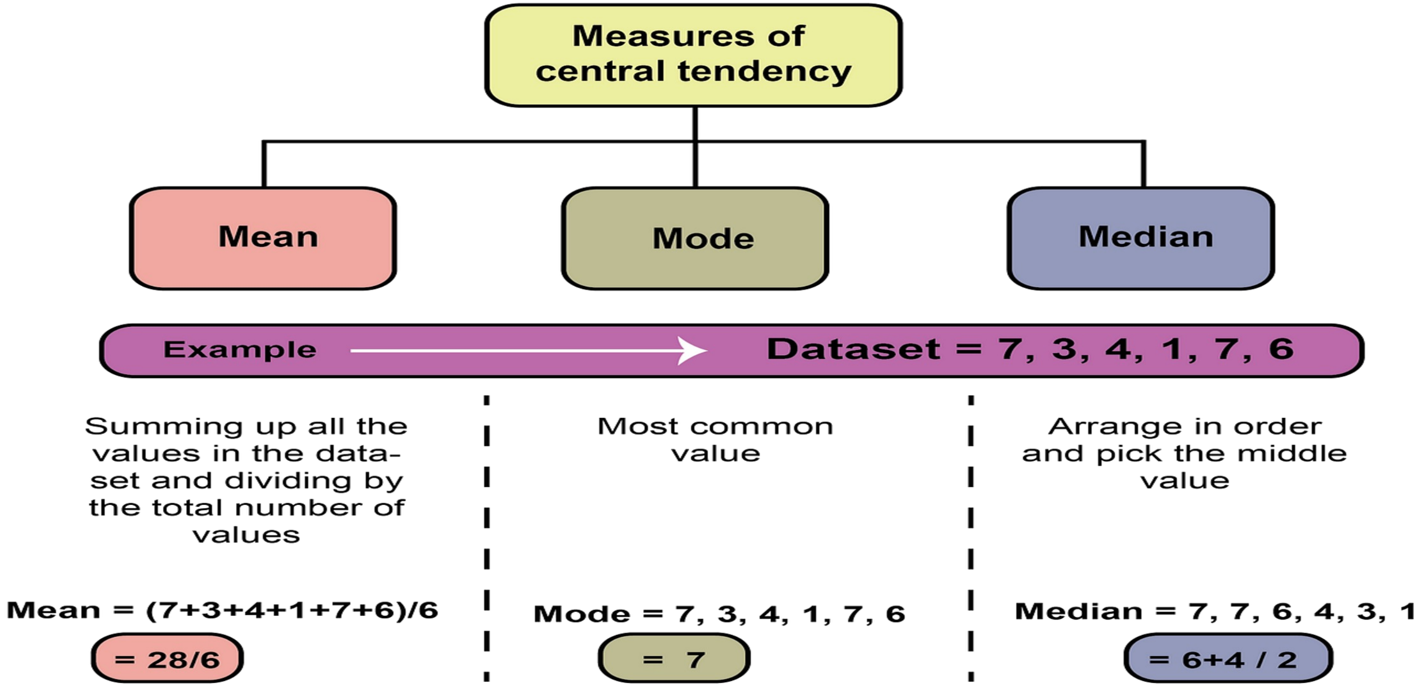
Exercise

- Classify the following attributes as discrete, or continuous. Also classify them as qualitative (nominal or ordinal) or quantitative (interval or ratio). Some cases may have more than one interpretation, so briefly indicate your reasoning if you think there may be some ambiguity.

1. Number of telephones in your house
2. Size of French Fries (Medium or Large or X-Large)
3. Ownership of a cell phone
4. Number of local phone calls you made in a month
5. Length of longest phone call
6. Length of your foot
7. Price of your textbook
8. Zip code
9. Temperature in degrees Fahrenheit
10. Temperature in degrees Celsius
11. Temperature in Kelvin

Measures of Central Tendency: Mean, Median, Mode, Mid-range

- Measures of central tendency are used to numerically summarize data by identifying the central position within that set of data. They are also called as summary statistic.



Mean

- Mean: The mean represents the average value of the dataset.
- Mean is calculated by taking the sum of the values and dividing with the numbers of values in a dataset
- The mean (or average) is the most popular and well-known measure of central tendency.
- It can be used with both discrete and continuous data, although its use is most often with continuous data
- An important property of the mean is that it includes every value in your data set as part of the calculation.
- Limitations of mean: The mean has one main disadvantage: it is particularly susceptible to the influence of outliers. These are values that are unusual compared to the rest of the data set by being especially small or large in numerical value

Example

- Ungrouped data: - 0 1 2 3 4 5 6 7 8 9 10 Mean = 5
- For grouped data

Grouped data:

Masses	Frequency(f)	X	FX
40-49	6	40+49/2 = 44.5	267
50-59	8	54.5	436
60-69	12	64.5	774
70-70	14	74.5	1043
80-89	7	84.5	591.5
90-99	3	94.5	283.5
Total	50		3395

Mean= 3395/50
67.9

Median

- Median is the middle value of the dataset in which the dataset is arranged in the ascending order or in descending order.
- When the dataset contains an even number of values, then the median value of the dataset can be found by taking the mean of the middle two values
- Consider the given dataset with the odd number of observations arranged in ascending order 2, 5, 6, 7, 9, 10, 12, 13, 15, 16, 18, 21, 23
- Here 12 is the middle or median number that has 6 values above it and 6 values below it
- Advantages of median: The median is less affected by outliners and skewed data than the mean and usually the preferred measure of central tendency when the distribution is not symmetrical
- Limitations of median: Unsuitable for fractions and percentage , it is not suitable for further algebraic calculation

Mode

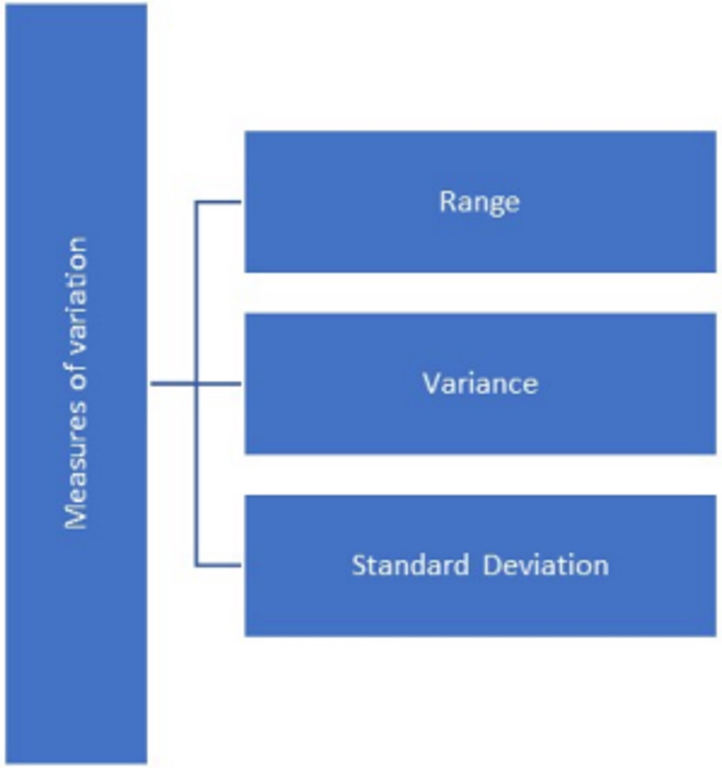
- The mode represents the frequently occurring value in the dataset.
- Sometimes the dataset may contain multiple modes, and in some cases, it does not contain any mode at all.
- Consider the given dataset 5, 4, 2, 3, 2, 1, 5, 4, 5 Since the mode represents the most common value. Hence, the most frequently repeated value in the given dataset is 5.
- Advantages of mode: The mode has an advantage over the median and the mean as it can be found for both numerical and categorical (non-numerical) data.
- Limitations of mode: In some distributions, the mode may not reflect the center of the distribution very well.

How to select measure

- Based on the properties of the data, the measures of central tendency are selected.
- If you have a symmetrical distribution of continuous data, all the three measures of central tendency hold good. But most of the times, the analyst uses the mean because it involves all the values in the distribution or dataset.
- If you have skewed distribution, the best measure of finding the central tendency is the median.
- If you have the original data, then both the median and mode are the best choice of measuring the central tendency.
- If you have categorical data, the mode is the best choice to find the central tendency

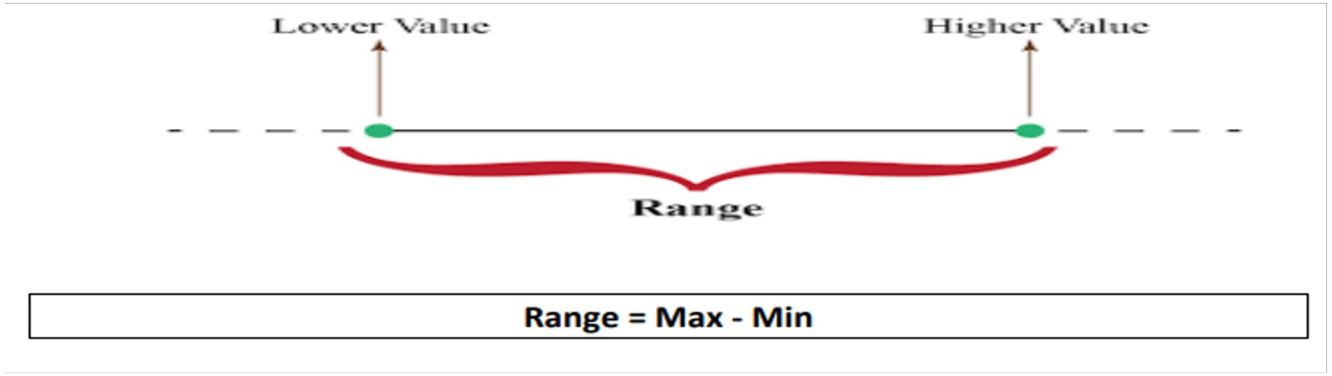
Measures of Dispersion: Range, Variance, Standard Deviation

- Measures of variation give information on the spread or variability or dispersion of the data values
- Finding the central value of the observations is important in this descriptive analysis.
- It is also important to find out the extent of variation near the center.
- There maybe two sets of data that may exhibit same position of the center, but that could differ with respect to their variability.



Range

- Range is referred to as the difference between the highest value and the lowest value in a data set.
- The range is extremely sensitive to outliers and can be a very coarse measurement of the spread of data.
- This means a single data value can greatly affect the analysis of the problem when using range and this can be a limitation to the statisticians who work using range.
- The below image is the pictorial representation of range



Range Example

Example 1:

- Participants in a race had the following finishing times in seconds: 16, 21, 28, 12, 30, 21, and 24. Compute the range.
- The maximum value is 30 and minimum value is 12,
- Hence the range is 30 – 12 = 18.

Example 2:

- The following is the data of measurements taken in meters. Find the range for the
- data. {10.2, 9.5, 12.6, 13, 7.9, 11.7, 8.1, 9, 11.8}.
- The range for the above data will be 13 – 7.9 = 5.1

Standard Deviation:

$$SD = \sqrt{\frac{\sum((x - mean)^2)}{N}}$$

- Standard Deviation: It gives us how spread out the observed values are from the mean. We can calculate the standard deviation using the below formula

Standard Deviation is always a positive value. Standard deviation is the square root of the variance. **Variance is the sum of the squared difference of the observed value from the mean.**

Standard deviation is a measure of the dispersion of a set of data from its mean.

If the data points are farther from the mean, there is higher deviation within the data set.

Standard deviation is calculated as the square root of variance by determining the variation between each data point relative to the mean.

Example

Example: Find the standard deviation for the below data;

{2, 4, 5, 7, 9, 11, 12, 14}

Sl	X	$X-\bar{X}$	$(X-\bar{X})^2$
1	2	$2-8=-6$	36
2	4	$4-8=-4$	16
3	5	$5-8=-3$	9
4	7	$7-8=-1$	1
5	9	$9-8=1$	1
6	11	$11-8=3$	9
7	12	$12-8=4$	16
8	14	$14-8=6$	36
			$\sum(X-\bar{X})^2=124$

$$\bar{X}=\frac{2+4+5+7+9+11+12+14}{8}=\frac{64}{8}=8$$

$$SD=\sqrt{\frac{\sum_{i=1}^{8}(Xi-8)^2}{8}}$$

$$SD=\sqrt{\frac{124}{8}}$$

$$SD=3.93$$

Variance

- Variance: The average deviation of all data values from the mean.
- The square of standard deviation is called variance.
- The difference between standard deviation and variance is only in terms of unit of measurement.
- For Example: If height is measured in meters and SD is 0.67 the unit of standard deviation is 0.67 meter, but the unit of variance is 0.45-meter square

Data Pre-processing: An Overview

- Data preprocessing is the process of transforming raw data into an understandable format.
- It is also an important step in data mining as we cannot work with raw data.
- The quality of the data should be checked before applying machine learning or data mining algorithms.
- The goal of data preprocessing is to improve the quality of the data and to make it more suitable for the specific data mining task.
- Data preprocessing is a process of **preparing the raw data** and making it **suitable for a machine learning model** and it is the first and crucial step while creating a machine learning model.

Why is Data Preprocessing Important?

- A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models.
- Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.
- *"How can the data be preprocessed in order to help improve the quality of the data and, consequently, of the mining results? How can the data be preprocessed so as to improve the efficiency and ease of the mining process?"*

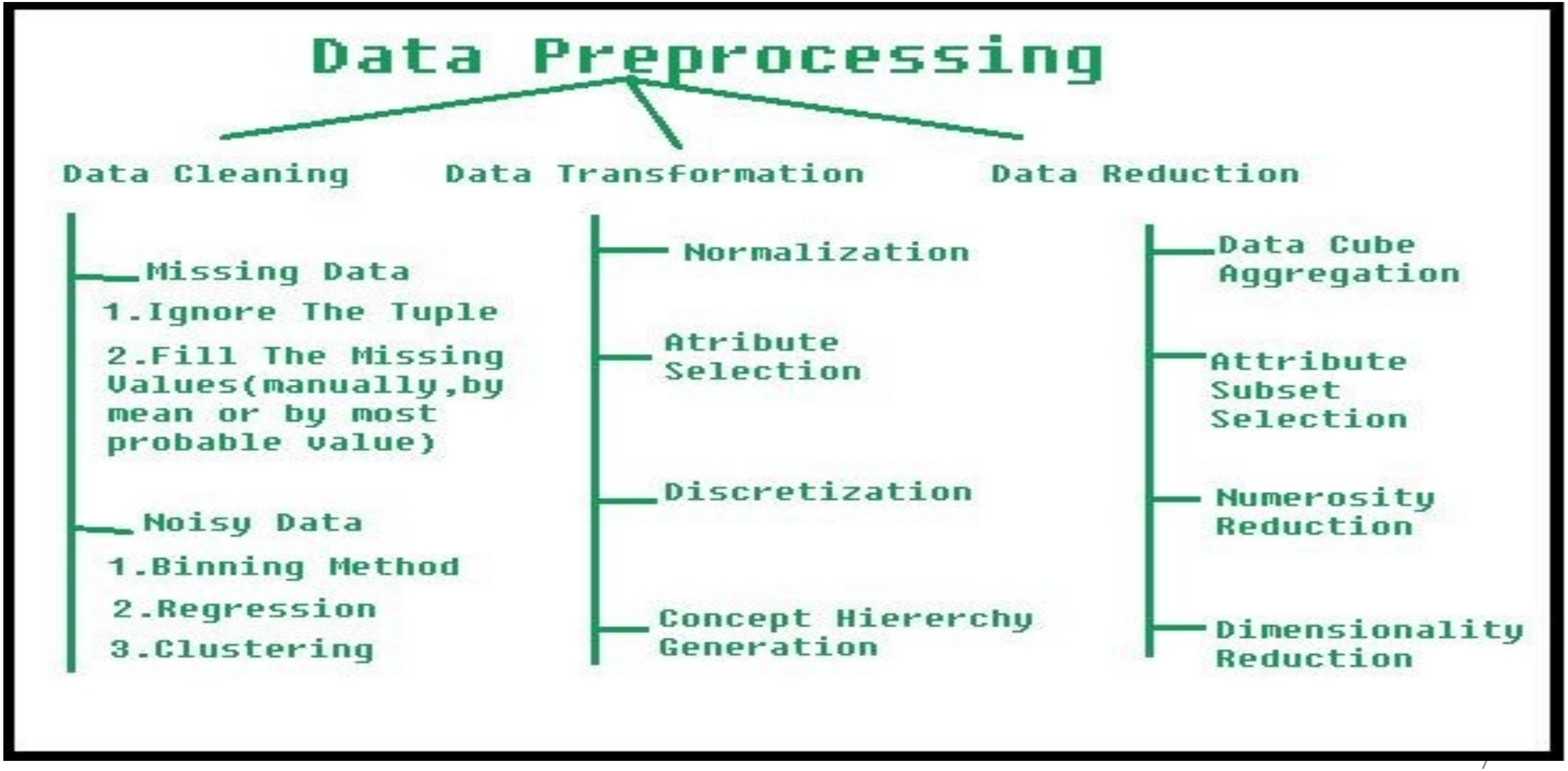
Why is Data Preprocessing Important?

- Preprocessing of data is mainly to check the data quality and it can be checked with the following parameters:
- **Accuracy**: To check whether the data entered is correct or not.
- **Completeness**: To check whether the data is available or not recorded.
- **Consistency:** To check whether the same data is kept in all the places that do or do not match.
- **Timeliness**: The data should be updated correctly.
- **Believability**: The data should be trustable.
- **Interpretability**: The understandability of the data.

Why is Data Preprocessing Important?

- *There are a number of data preprocessing techniques: Data cleaning can be applied to remove noise and correct inconsistencies in the data. Data integration merges data from multiple sources into a coherent data store, such as a data warehouse. Data reduction can reduce the data size by aggregating, eliminating redundant features, or clustering, for instance. Data transformations, such as normalization, may be applied, where data are scaled to fall within a smaller range like 0.0 to 1.0. This can improve the accuracy and efficiency of mining algorithms involving distance measurements.*

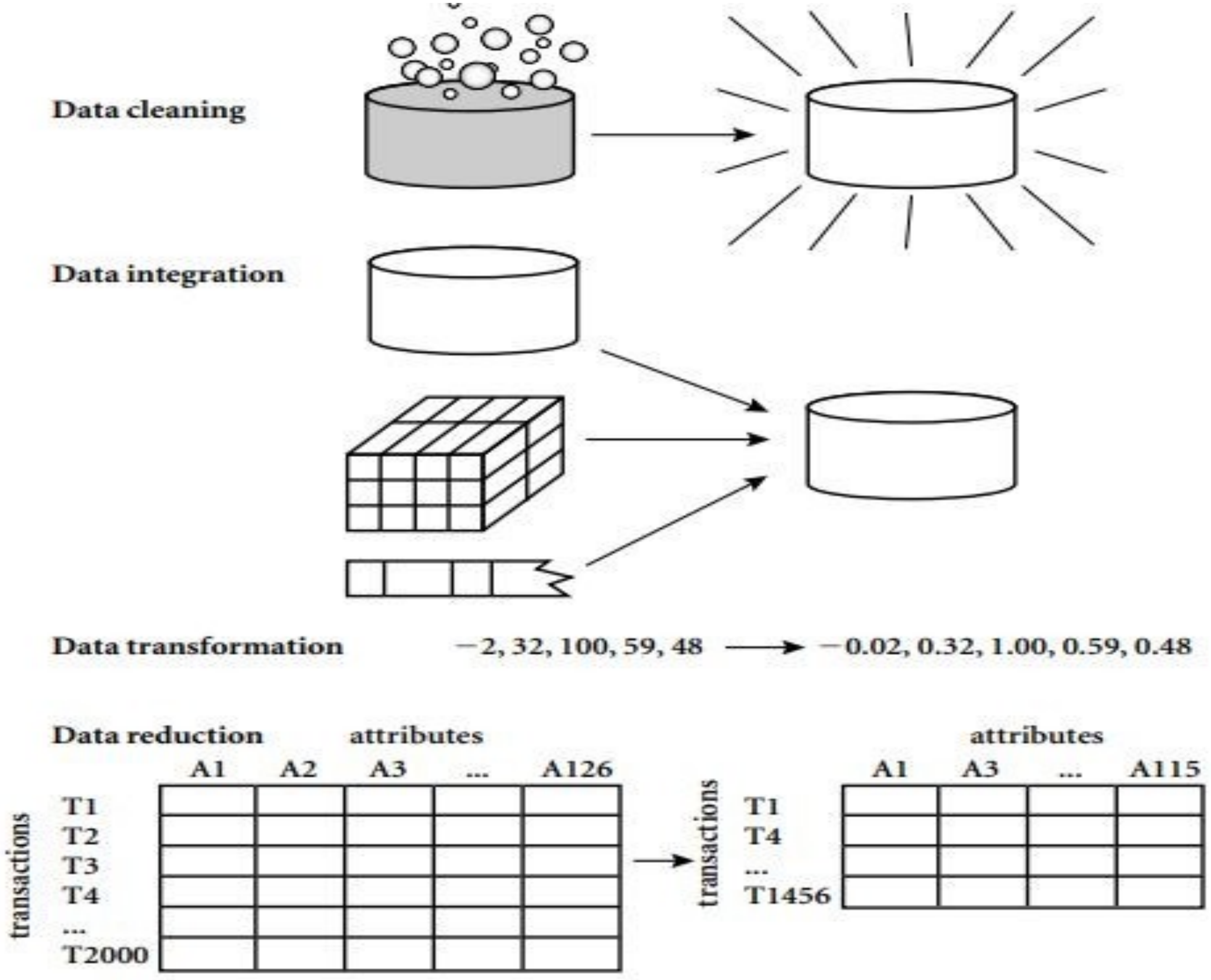
Why is Data Preprocessing Important?



Major Tasks in Data Preprocessing

There are 4 major tasks in data preprocessing – Data cleaning, Data integration, Data reduction, and Data transformation.

Major Tasks in Data Preprocessing



Data Cleaning

- Real-world data tend to be incomplete, noisy, and inconsistent. Data cleaning (or data cleansing) routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.
- Data cleaning is the process of removing incorrect data, incomplete data, and inaccurate data from the datasets, and it also replaces the missing values.
- Topics under Data Cleaning are: Handling missing values, Noisy data, Data cleaning as a process.

Data Cleaning

- **Missing or incomplete records:** Missing data sometimes appears as empty cells, values (e.g., NULL or N/A), or a particular character, such as a question mark

age	weight
[50-60)	?
[20-30)	[50-75)
[80-90)	?
[50-60)	?
[50-60)	?
[70-80)	?

Data Cleaning

- 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
  - Not register history or changes of the data.
- It is important to note that, a missing value may not always imply an error. (for example, Null-allow attri.)

Data Cleaning

- How can you go about filling in the missing values for this attribute? Let’s look at the following methods:
- **1. Ignore the tuple:** This is usually done when the class label is missing (assuming the mining task involves classification).
- This method is not very effective, unless the tuple contains several attributes with missing values.
- It is especially poor when the percentage of missing values per attribute varies considerably.
- By ignoring the tuple, we do not make use of the remaining attributes values in the tuple.
- **2. Fill in the missing value manually:** In general, this approach is time consuming and may not be feasible given a large data set with many missing values.

Data Cleaning

- **3. Use a global constant to fill in the missing value:** Replace all missing attribute values by the same constant, such as a label like "Unknown" or $-\infty$.
- If missing values are replaced by, say, "Unknown," then the mining program may mistakenly think that they form an interesting concept, since they all have a value in common—that of "Unknown."
- **4. Use a measure of central tendency for the attribute (such as the mean or median) to fill in the missing value:** measures of central tendency, which indicate the "middle" value of a data distribution.
- For normal (symmetric) data distributions, the mean can be used, while skewed data distribution should employ the median.

Data Cleaning

- **5. Use the attribute mean or median for all samples belonging to the same class as the given tuple:** For example, if classifying customers according to credit risk, we may replace the missing value with the average income value for customers in the same credit risk category as that of the given tuple.
- If the data distribution for a given class is skewed, the median value is a better choice.
- **6. Use the most probable value to fill in the missing value:** This may be determined with regression, inference-based tools using a Bayesian formalism, or decision tree induction.
- For example, using the other customer attributes in your data set, you may construct a decision tree to predict the missing values for income.

Data Cleaning: Noisy Data

Binning: Binning methods smooth a sorted data value by consulting its "neighborhood," that is, the values around it. The sorted values are distributed into a number of "buckets," or bins

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* Partition into (equi-depth) bins:

-Bin 1: 4, 8, 9, 15

-Bin 2: 21, 21, 24, 25

-Bin 3: 26, 28, 29, 34

* Smoothing by bin means:

-Bin 1: 9, 9, 9, 9 (4+8+9+15/4) =9

-Bin 2: 23, 23, 23, 23 (21+21+24+25/4)=23

-Bin 3: 29, 29, 29, 29 (26+28+29+34/4) =29

* Smoothing by bin boundaries:

-Bin 1: 4, 4, 4, 15

-Bin 2: 21, 21, 25, 25

-Bin 3: 26, 26, 26, 34

Data Cleaning

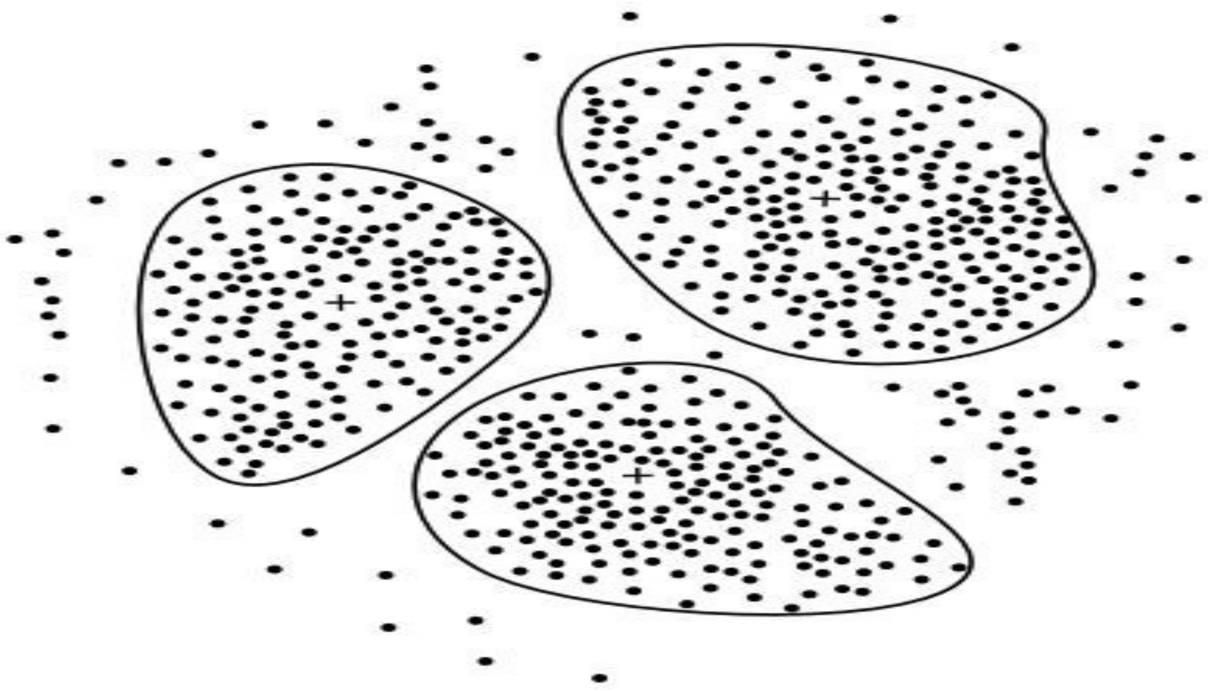
- In this example, the data for price are first sorted and then partitioned into equal-frequency bins of size 3 (i.e., each bin contains three values).
- In smoothing by bin means, each value in a bin is replaced by the mean value of the bin. For example, the mean of the values 4, 8, and 15 in Bin 1 is 9.
- Therefore, each original value in this bin is replaced by the value 9.
- Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median.
- In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries.
- Each bin value is then replaced by the closest boundary value.

Data Cleaning

- **Regression:** Data smoothing can also be done by conforming data values to a function, a technique known as regression.
- Linear regression involves finding the “best” line to fit two attributes (or variables), so that one attribute can be used to predict the other.
- Multiple linear regression is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

Data Cleaning

- **Outlier analysis:** Outliers may be detected by clustering, for example, where similar values are organized into groups, or “clusters.” Intuitively, values that fall outside of the set of clusters may be considered outliers



A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters. Each cluster centroid is marked with a "+", representing the average point in space for that cluster. Outliers may be detected as values that fall outside of the sets of clusters.

Data Cleaning as a Process

- *Missing values, noise, and inconsistencies contribute to inaccurate data. So far, we have looked at techniques for handling missing data and for smoothing data. "But data cleaning is a big job. What about data cleaning as a process? How exactly does one proceed in tackling this task? Are there any tools out there to help?"*
- The first step in data cleaning as a process is discrepancy (inconsistency) detection.
- Discrepancies can be caused by several factors, including poorly designed data entry forms that have many optional fields, human error in data entry, deliberate errors (e.g., respondents not wanting to disclose information about themselves), and data decay (e.g., outdated addresses).
- Discrepancies may also arise from inconsistent data representations and the inconsistent use of codes.
- Errors in instrumentation devices that record data, and system errors, are another source of discrepancies.
- Errors can also occur when the data are (inadequately) used for purposes other than originally intended and there may also be inconsistencies due to data integration (e.g., where a given attribute can have different names in different databases).

Data Cleaning as a Process

- *"So, how to proceed with discrepancy detection?" As a starting point, use any knowledge you may already have regarding properties of the data. Such knowledge or "data about data" is referred to as metadata.*
- For example, what are the data type and domain of each attribute? What are the acceptable values for each attribute? The basic statistical data descriptions those are useful here to grasp data trends and identify anomalies.
- For example, find the mean, median, and mode values.
- Are the data symmetric or skewed? What is the range of values? Do all values fall within the expected range? What is the standard deviation of each attribute?
- Values that are more than two standard deviations away from the mean for a given attribute may be flagged as potential outliers.
- Are there any known dependencies between attributes?

Data Cleaning as a Process

- Field overloading is another source of errors that typically results when developers squeeze new attribute definitions into unused (bit) portions of already defined attributes (e.g., using an unused bit of an attribute whose value range uses only, say, 31 out of 32 bits).
- The data should also be examined regarding unique rules, consecutive rules, and null rules.
- A unique rule says that each value of the given attribute must be different from all other values for that attribute.
- A consecutive rule says that there can be no missing values between the lowest and highest values for the attribute, and that all values must also be unique (e.g., as in check numbers).
- A null rule specifies the use of blanks, question marks, special characters, or other strings that may indicate the null condition (e.g., where a value for a given attribute is not available), and how such values should be handled.

Data Cleaning as a Process

- There are a number of different commercial tools that can aid in the step of discrepancy detection.
- Data scrubbing tools use simple domain knowledge (e.g., knowledge of postal addresses, and spell-checking) to detect errors and make corrections in the data.
- These tools rely on parsing and fuzzy matching techniques when cleaning data from multiple sources.
- Data auditing tools find discrepancies by analyzing the data to discover rules and relationships, and detecting data that violate such conditions.
- They are variants of data mining tools. For example, they may employ statistical analysis to find correlations, or clustering to identify outliers.

Data Cleaning as a Process

- ETL (extraction/transformation/loading) tools allow users to specify transforms through a graphical user interface (GUI).
- These tools typically support only a restricted set of transforms so that, often, we may also choose to write custom scripts for this step of the data cleaning process.
- The two-step process of discrepancy detection and data transformation (to correct discrepancies) iterates.
- This process, however, is error-prone and time consuming and some transformations may introduce more discrepancies while some nested discrepancies may only be detected after others have been fixed.

Data Cleaning as a Process

- New approaches to data cleaning emphasize increased interactivity.
- Potter's Wheel, for example, is a publicly available data cleaning tool that integrates discrepancy detection and transformation.
- Users gradually build a series of transformations by composing and debugging individual transformations, one step at a time, on a spreadsheet-like interface.
- The transformations can be specified graphically or by providing examples.
- Results are shown immediately on the records that are visible on the screen.
- The user can choose to undo the transformations, so that transformations that introduced additional errors can be "erased."
- The tool performs discrepancy checking automatically in the background on the latest transformed view of the data.
- Users can gradually develop and refine transformations as discrepancies are found, leading to more effective and efficient data cleaning.
- Another approach to increased interactivity in data cleaning is the development of declarative languages for the specification of data transformation operators.
- Such work focuses on defining powerful extensions to SQL and algorithms that enable users to express data cleaning specifications efficiently.

Exercise

- Suppose a group of 12 sales price records has been sorted as follows: 5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215 Partition them into three bins solve it by each of the following methods:

(a) Equi-depth partitioning

(b) Smoothing by bin boundaries

Exercise

- Solution:
  (a) equi-depth partitioning -Bin-1: 5, 10, 11, 13, Bin-2: 15, 35, 50, 55, Bin-03: 72, 92, 204, 215
  (b) Smoothing by bin boundaries-Smoothing by bin boundaries: Bin-1: 5,13,13,13 , Bin-2: 15,15,55,55, Bin-3:72,72,215,215

Data Integration

- The merging of data from multiple data stores.
- Careful integration can help reduce and avoid redundancies and inconsistencies in the resulting data set.
- This can help improve the accuracy and speed of the subsequent mining process.
- *The semantic heterogeneity and structure of data pose great challenges in data integration. How can we match schema and objects from different sources?*

Data Integration

Data integration techniques:

- Schema matching
- Instance conflict resolution
- Source selection
- Result merging
- Quality composition

Data Integration

- Combines data from multiple sources into a coherent store
- Careful integration can help reduce & avoid redundancies and inconsistencies
- This helps to improve accuracy & speed of subsequent data mining
- Heterogeneity & structure of data pose great challenges
- Issues that need to be addressed:
  1. How to match schema & objects from different sources? (Entity identification problem)
  2. Are any attributes correlated?
  3. Tuple duplication
  4. Detection & resolution of data value conflicts

Data Integration

Data integration techniques:

- Schema matching
- Instance conflict resolution
- Source selection
- Result merging
- Quality composition

- Entity Identification Problem, Redundancy and Correlation Analysis, Tuple Duplication, Data Value Conflict Detection and Resolution

84

Data Integration

- **Data integration**:
  - Combines data from multiple sources into a coherent store
  - Careful integration can help reduce & avoid redundancies and inconsistencies
  - This helps to improve accuracy & speed of subsequent data mining
  - Heterogeneity & structure of data pose great challenges
  - Issues that need to be addressed:
    1. How to match schema & objects from different sources? (Entity identification problem)
    2. Are any attributes correlated?
    3. Tuple duplication
    4. Detection & resolution of data value conflicts

85

Issues during Data Integration(Contd..)

1. Schema integration:

   Integrate metadata from different sources

   - e.g. customer_id in one database and cust_number in another

   Entity identification problem: Identify real world entities from multiple data sources,

   - e.g., Bill Clinton = William Clinton

2. Redundancy:
   - An attribute may be redundant if it can be derived or obtaining from another attribute or set of attribute.
   - Inconsistencies in attribute can also cause redundancies in the resulting data set.
   - Some redundancies can be detected by correlation analysis.

3. Detecting and resolving data value conflicts
   - For the same real world entity, attribute values from different sources are different
   - Possible reasons: different representations, different scales, e.g., metric vs. British units

Data Integration(Contd..)

- Data integration:
  - Combines data from multiple sources into a coherent store
- Schema integration: e.g., A.cust-id ≡ B.cust-#
  - Integrate metadata from different sources
- Entity identification problem:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
- **Redundancy**:
  - Inconsistencies in attribute or dimensions naming can cause redundancy
- Detecting and resolving data value conflicts
  - For the same real world entity, attribute values from different sources are different Possible reasons: different representations, different scales, e.g., metric vs. British units

34

Data Integration

Entity identification problem

- How can equivalent real-world entities from multiple data sources be matched up?
- Schema integration: e.g., A.cust-id ≡B.cust-number
- Integrate using **metadata**(meaning, data type, range of values,etc.) from different sources
- Ensure that any attribute functional dependencies & referential constraints in source system match those in target system

88

Handling Redundancy in Data Integration

Redundancy & Correlation Analysis:

- Redundant data occur often when integration of multiple databases
  - *Object identification*: The same attribute or object may have different names in different databases
  - *Derivable data:* One attribute may be a "derived" attribute in another table, e.g., annual revenue
- Redundant attributes may be able to be detected by *correlation analysis* and *covariance analysis*
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

89

Detection of Data Redundancy-Correlation

- Redundancies can be detected using following methods:
  - $\chi 2$ Test (Used for nominal Data or categorical or qualitative data)
  - Correlation coefficient and covariance (Used for numeric Data or quantitative data)
- The term "correlation" refers to a mutual relationship or association between quantities.
- In almost any business, it is useful to express one quantity in terms of its relationship with others.
  - For example, sales might increase when the marketing department spends more on TV advertisements,
  - Customer's average purchase amount on an e-commerce website might depend on a number of factors related to that customer.
- Often, correlation is the first step to understanding these relationships and subsequently building better business and statistical models.

Why is correlation a useful metric?

- Correlation can help in predicting one quantity from another
- Correlation can (but often does not, as we will see in some examples below) indicate the presence of a causal relationship
- Correlation is used as a basic quantity and foundation for many other modeling techniques
- More formally, correlation is a statistical measure that describes the association between random variables.
- There are several methods for calculating the correlation coefficient, each measuring different types of strength of association

Correlation Analysis (Numeric data)

- Evaluate correlation between 2 attributes, A & B, by computing **correlation coefficient(Pearson's product moment coefficient**)

$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^{n}(a_ib_i) - n\bar{A}\bar{B}}{n\sigma_A\sigma_B}$$

Where *n* is the number of tuples, a_i and b_i are the respective values of *A* and *B* in tuple *i*,

$\bar{A}$ and $\bar{B}$: the respective mean values of *A* and *B*,

σ_A and σ_B are the respective standard deviations of *A* and *B*

$\Sigma(a_ib_i)$ is the sum of the *AB* cross-product (i.e., for each tuple, the value for *A* is multiplied by the value for *B* in that tuple)

Correlation- Pearson Correlation Coefficient

- Pearson is the most widely used correlation coefficient.
- Pearson correlation measures the linear association between continuous variables.
- Pearson correlation coefficient

The **covariance** between A and B is defined as

$$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}.$$

$$E(A) = \bar{A} = \frac{\sum_{i=1}^{n} a_i}{n}$$

$$E(B) = \bar{B} = \frac{\sum_{i=1}^{n} b_i}{n}.$$

- The Correlations coefficient is a statistic and it can range between +1 and -1

Correlation Analysis (Numeric data)

- Note that $-1 \leq r_{A,B} \leq +1$
- If resulting value is greater than 0, then *A* and *B* are *positively correlated*, meaning that the values of *A* increase as the values of *B* increase
- Higher the value, the stronger the correlation
- Higher value may indicate that *A* (or *B*) may be removed as a redundancy
- If the resulting value is equal to 0, then *A* and *B* are *independent* and there is no
- correlation between them
- If the resulting value is less than 0, then *A* and *B* are *negatively correlated*, where the values of one attribute increase as the values of the other attribute decrease

Covariance Analysis(Numeric Data)

- Correlation and covariance are two similar measures for assessing how much two attributes change together
- Consider two numeric attributes *A* and *B*, and a set of *n* observations

  $$\{(a_1, b_1), \ldots, (a_n, b_n)\}$$

- Mean values of *A* and *B*, respectively, are also known as the **expected values** on *A* and *B*, that is

$$E(A) = \bar{A} = \frac{\sum_{i=1}^{n} a_i}{n}$$

$$E(B) = \bar{B} = \frac{\sum_{i=1}^{n} b_i}{n}.$$

- **covariance** between *A* and *B* is defined as

$$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^{n} (a_i - \bar{A})(b_i - \bar{B})}{n}.$$

Covariance Analysis(Numeric Data)

$$r_{A,B} = \frac{Cov(A,B)}{\sigma_A \sigma_B}$$

- Also, for simplified calculations

$$Cov(A,B) = E(A \cdot B) - \bar{A}\bar{B}.$$

- For two attributes *A* and *B* that tend to change together,
  - if *A* is larger than $\bar{A}$, then *B* is likely to be larger than $\bar{B}$. Hence, the covariance between *A* and *B* is ***positive***
  - if one of the attributes tends to be above its expected value when the other attribute is below its expected value, then the covariance of *A* and *B* is ***negative***
  - covariance value is 0 means attributes are independent

Correlation

- **Positive covariance**: If $\mathrm{Cov}_{A,B} > 0$, then A and B both tend to be larger than their expected values
- **Negative covariance**: If $\mathrm{Cov}_{A,B} < 0$ then if A is larger than its expected value, B is likely to be smaller than its expected value
- **Independence**: $\mathrm{Cov}_{A,B} = 0$ but the converse is not true:
  - Some pairs of random variables may have a covariance of 0 but are not independent. Only under some additional assumptions (e.g., the data follow multivariate normal distributions) does a covariance of 0 imply independence

$$Cov(A,B) = E(A \cdot B) - \bar{A}\bar{B}.$$

Covariance Analysis(Numeric Data)

Stock Prices for *AllElectronics* and *HighTech*

Time point	*AllElectronics*	*HighTech*
t1	6	20
t2	5	10
t3	4	14
t4	3	5
t5	2	5

- Table shows stock prices of two companies at five time points. If the stocks are affected by same industry trends, determine whether their prices rise or fall together?

Covariance Analysis(Numeric Data)

$$E(AllElectronics) = \frac{6+5+4+3+2}{5} = \frac{20}{5} = \$4$$

and

$$E(HighTech) = \frac{20+10+14+5+5}{5} = \frac{54}{5} = \$10.80.$$

we compute

$$Cov(AllElectroncis, HighTech) = \frac{6 \times 20 + 5 \times 10 + 4 \times 14 + 3 \times 5 + 2 \times 5}{5} - 4 \times 10.80$$
$$= 50.2 - 43.2 = 7.$$

Therefore, given the positive covariance we can say that stock prices for both companies rise together. ■

Correlation: example

An example of stock prices observed at five time points for *AllElectronics* and *HighTech*, a high-tech company. If the stocks are affected by the same industry trends, will their prices rise or fall together?

Time Point	All Electronics	High Tech
t1	2	5
t2	3	8
t3	5	10
t4	4	11
t5	6	14

- Suppose two stocks A and B have the following values in one week: (2, 5), (3, 8), (5, 10), (4, 11), (6, 14).
- Question: If the stocks are affected by the same industry trends, will their prices rise or fall together?
  - **E(A) = (2 + 3 + 5 + 4 + 6)/ 5 = 20/5 = 4**
  - **E(B) = (5 + 8 + 10 + 11 + 14) /5 = 48/5 = 9.6**
  - **Cov(A,B) = (2×5+3×8+5×10+4×11+6×14)/5 − 4 × 9.6 = 4**
- Thus, A and B rise together since Cov(A, B) > 0.

Data Value Conflict Detection and Resolution

- For the same real-world entity, attribute values from different sources may differ
  - Eg. Prices of rooms in different cities may involve different currencies
- Attributes may also differ on the abstraction level, where an attribute in one system is recorded at, say, a lower abstraction level than the "same" attribute in another.
  - Eg. total sales in one database may refer to one branch of All_Electronics, while an attribute of the same name in another database may refer to the total sales for All_Electronics stores in a given region.
- To resolve, data values have to be converted into consistent form

Data Transformation

- Smoothing: remove noise from data (binning, clustering, regression)
- Aggregation: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
  - Min-max normalization
  - Z-score normalization
  - Normalization by decimal scaling
- Attribute/feature construction
  - New attributes constructed from the given ones
  - E.g. add attribute *area* based on the attributes *height* and *width*

49

Data Transformation: Normalization

- Min-max normalization

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

- Z-score normalization

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

- Normalization by decimal scaling

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

Where *j* is the smallest integer such that Max(| |)<1

Data Transformation: Min Max-Normalization

- Min-max normalization: to [new_minA, new_maxA]
  - Performs a linear transformation on the original data.
  - Suppose that min_a and max_a are the minimum and maximum values of an attribute, *a*.
  - Min-max normalization maps a value, v_i, of *a* to the range $[new\_min_a, new\_max_a]$ by computing

    - Let ***income*** range \$12,000 to \$98,000 normalized to [0.0, 1.0].
    - Then \$73,000 is mapped to

$$\frac{73{,}600-12{,}000}{98{,}000-12{,}000}(1.0-0)+0=0.716$$

  - Eg. Suppose that the minimum and maximum values for the attribute income are \$12,000 and \$98,000, respectively. We would like to map income to the range [0.0, 1.0]. By min-max normalization, a value of \$73,600 for income is transformed to **0.716**

Data Transformation: Z-score Normalization

- Z-score normalization (uses mean & σ: standard deviation):
- Values for an attribute, A, are normalized based on the mean (i.e., average) and standard deviation of A

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A},$$

- Eg. Suppose that the mean and standard deviation of the values for the attribute income are $54,000 and $16,000, respectively.

- Let $\bar{A} = 54{,}000$, $\sigma_A = 16{,}000$, for the attribute ***income***
- With z-score normalization, a value of $73,600 for ***income*** is transformed to:

$$\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$$

Data Transformation: Decimal scaling Normalization

- Suppose that the recorded values of A ranges from -986 to 917
- The maximum absolute value of A is 986
- To normalize by decimal scaling, we therefore divide each value by 1000
- So that -986 normalize to -0.986 and
- 917 normalize to -0.917

Data Warehouse and Data Mining

- Introduction to KDD
- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- **Data Reduction**
- Data Transformation and Data Discretization
- Summary

Exercise

- Define Normalization.
- What is the value range of min-max. Use min-max normalization to normalize the following group of data: 8,10,15,20.
- Solution:

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

Marks	Marks after Min-Max normalization
8	0
10	0.16
15	0.58
20	1

Data Reduction Strategies

- Data is too big to work with
- 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
- Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.
- Data reduction strategies
  - Dimensionality reduction — remove unimportant attributes
  - Aggregation and clustering
  - Sampling

Data Reduction Strategies

- Obtains a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results
- Data reduction strategies
  - Data cube aggregation
  - Dimensionality reduction
  - Data compression
  - Numerosity reduction-Parametric and Non-Parametric methods
  - Discretization and concept hierarchy generation

Data Cube Aggregation

- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation capable to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

Data Cube Aggregation

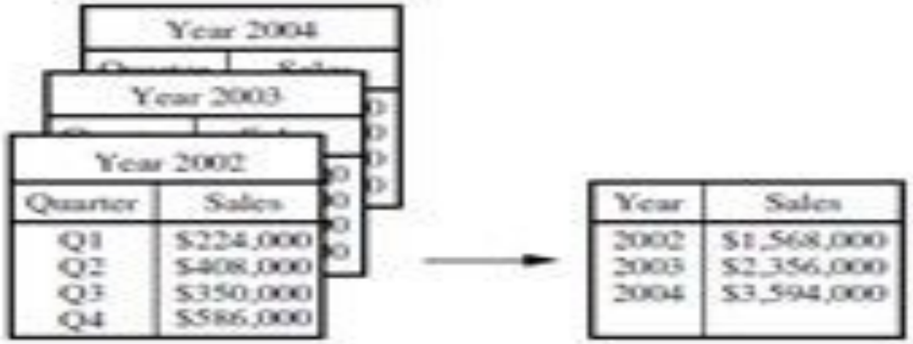


Figure 2.13 Sales data for a given branch of *AllElectronics* for the years 2002 to 2004. On the left, the sales are shown per quarter. On the right, the data are aggregated to provide the annual sales.

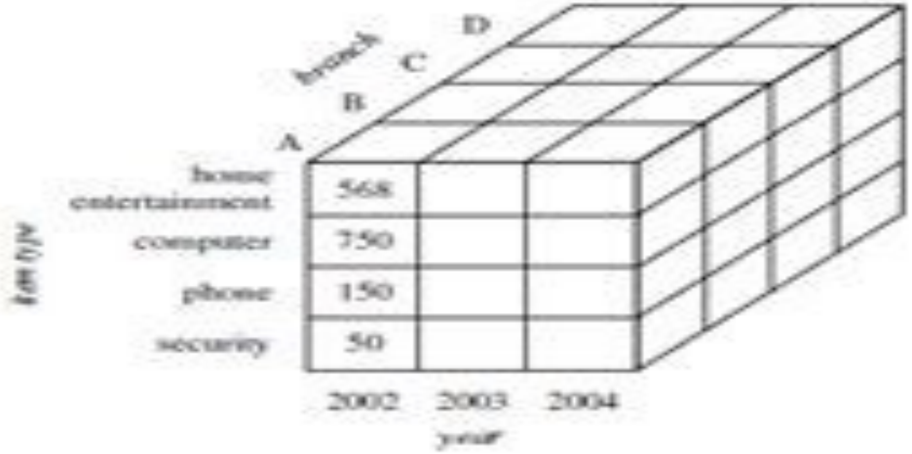


Figure 2.14 A data cube for sales at *AllElectronics*.

Data Reduction Strategies

- Data reduction strategies:
  - Dimensionality reduction, e.g., remove unimportant attributes
    - Wavelet transforms
    - Principal Components Analysis (PCA)
    - Feature subset selection, feature creation
  - Numerosity reduction (some simply call it: Data Reduction)
    - Regression and Log-Linear Models
    - Histograms, clustering, sampling
    - Data cube aggregation
  - Data compression

Data Reduction 1- Dimensionality Reduction

- Problem: Feature selection (i.e., attribute subset selection):
  - Select a minimum set of features 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
  - Nice side-effect: reduces # of attributes in the discovered patterns (which are now easier to understand)
- Solution: Heuristic methods (due to exponential # of choices) usually greedy:
  - step-wise forward selection
  - step-wise backward elimination
  - combining forward selection and backward elimination
  - decision-tree induction
  - E.g. Decision Tree

Dimensionality reduction :Attribute Subset Selection

- Reduces the data set size by removing irrelevant or redundant attributes (or dimensions)
- Goal of attribute subset selection is to find a minimum set of attributes
- Improves speed of mining as dataset size is reduced
- Mining on a reduced data set also makes the discovered pattern easier to understand
    - Duplicate information contained in one or more attributes
    - E.g., purchase price of a product and the amount of sales tax paid
- Irrelevant attributes
    - Contain no information that is useful for the data mining task at hand
    - E.g., students' telephone number is often irrelevant to the task of predicting students' CGPA

Attribute Subset Selection

Best and worst attributes are determined using **tests of statistical significance**

- A test of significance is a formal procedure for comparing observed data with a claim (also called a hypothesis), the truth of which is being assessed
- Claim is a statement about a parameter
- Results of a significance test are expressed in terms of a probability that measures how well the data and the claim agree

Heuristic (Greedy) methods for attribute subset selection

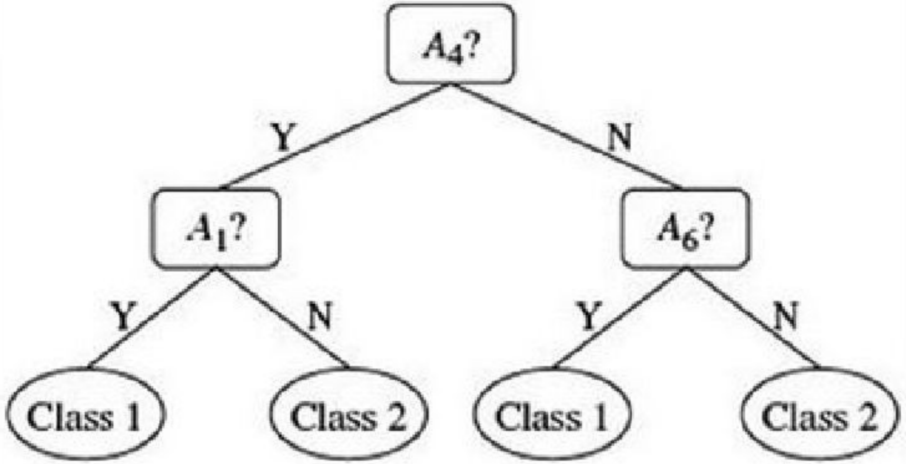
1. **Stepwise Forward Selection:**
   - Starts with an empty set of attributes as the reduced set
   - Best of the relevant attributes is determined and added to the reduced set
   - In each iteration, best of remaining attributes is added to the set
2. **Stepwise Backward Elimination:**
   - Here all the attributes are considered in the initial set of attributes
   - In each iteration, worst attribute remaining in the set is removed
3. **Combination of Forward Selection and Backward Elimination:**
   - Stepwise forward selection and backward elimination are combined
   - At each step, the procedure selects the best attribute and removes the worst from among the remaining attributes

Heuristic (Greedy) methods for attribute subset selection(cont)

4. Decision Tree Induction:

- This approach uses decision tree for attribute selection.
- It constructs a flow chart like structure having nodes denoting a test on an attribute.
- Each branch corresponds to the outcome of test and leaf nodes is a class prediction.
- The attribute that is not the part of tree is considered irrelevant and hence discarded

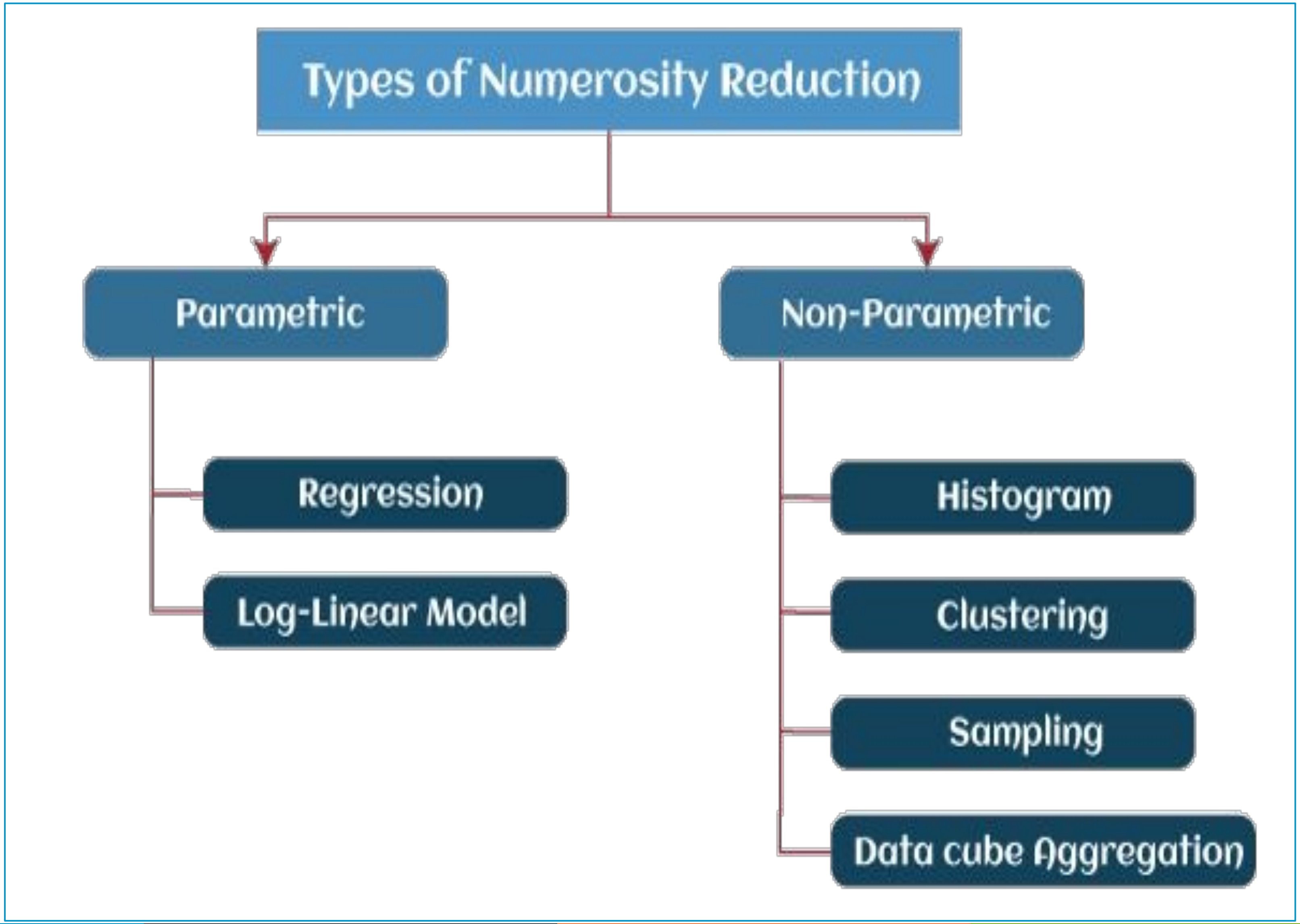
Example of Decision Tree Induction

Forward selection	Backward elimination	Decision tree induction
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: $\{A_1, A_4, A_6\}$	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: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  => Reduced attribute set: $\{A_1, A_4, A_6\}$

Data Reduction 2: Numerosity Reduction

- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - E.g.: Log-linear models: obtain value at a point in m-D space as the product on appropriate marginal subspaces
- Non-parametric methods
  - Do not assume models
  - Major families: histograms, clustering, sampling

Data Reduction 2: Numerosity Reduction



Regression and Log-Linear Models

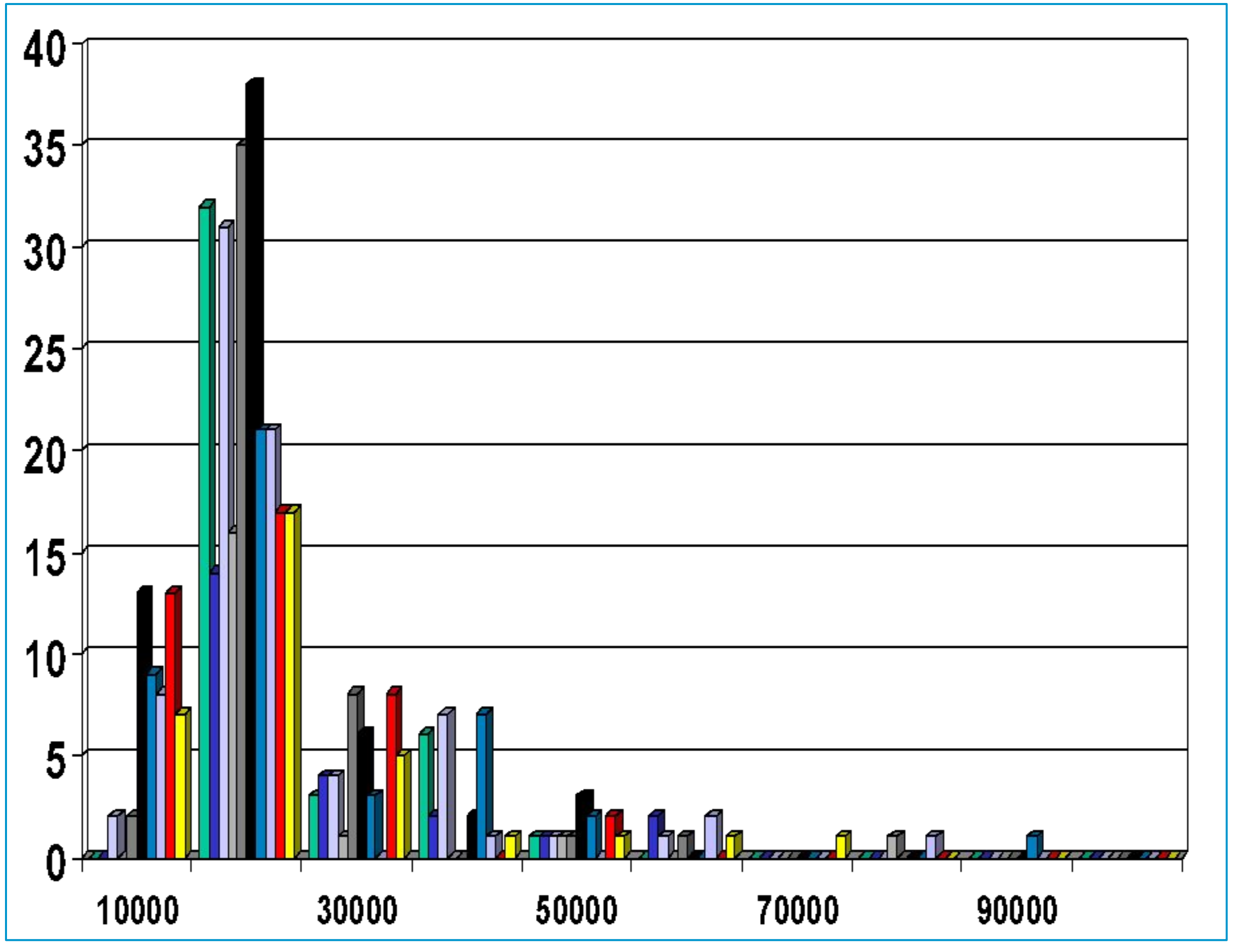
- Linear regression: Data are modeled to fit a straight line: $Y = \alpha + \beta X$
  - Often uses the least-square method to fit the line
- Multiple regression: allows a response variable y to be modeled as a linear function of multidimensional feature vector (predictor variables) $Y = b_0 + b_1 X_1 + b_2 X_2$.
- Log-linear model: approximates discrete multidimensional joint probability distributions $p(a, b, c, d) = \alpha_{ab} \beta_{ac} \chi_{ad} \delta_{bcd}$

Histograms

- Histograms (or frequency histograms) are at least a century old and are widely used.
- Plotting histograms is a graphical method for summarizing the distribution of a given attribute, X.
- Height of the bar indicates the frequency (i.e., count) of that X value
- Range of values for X is partitioned into disjoint consecutive subranges.
- Subranges, referred to as **buckets or bins**, are disjoint subsets of the data distribution for X.
- Range of a bucket is known as the width
- Typically, the buckets are of equal width.
- Eg. a price attribute with a value range of $1 to $200 can be partitioned into subranges 1 to 20, 21 to 40, 41 to 60, and so on.
- For each subrange, a bar is drawn with a height that represents the total count of items observed within the subrange

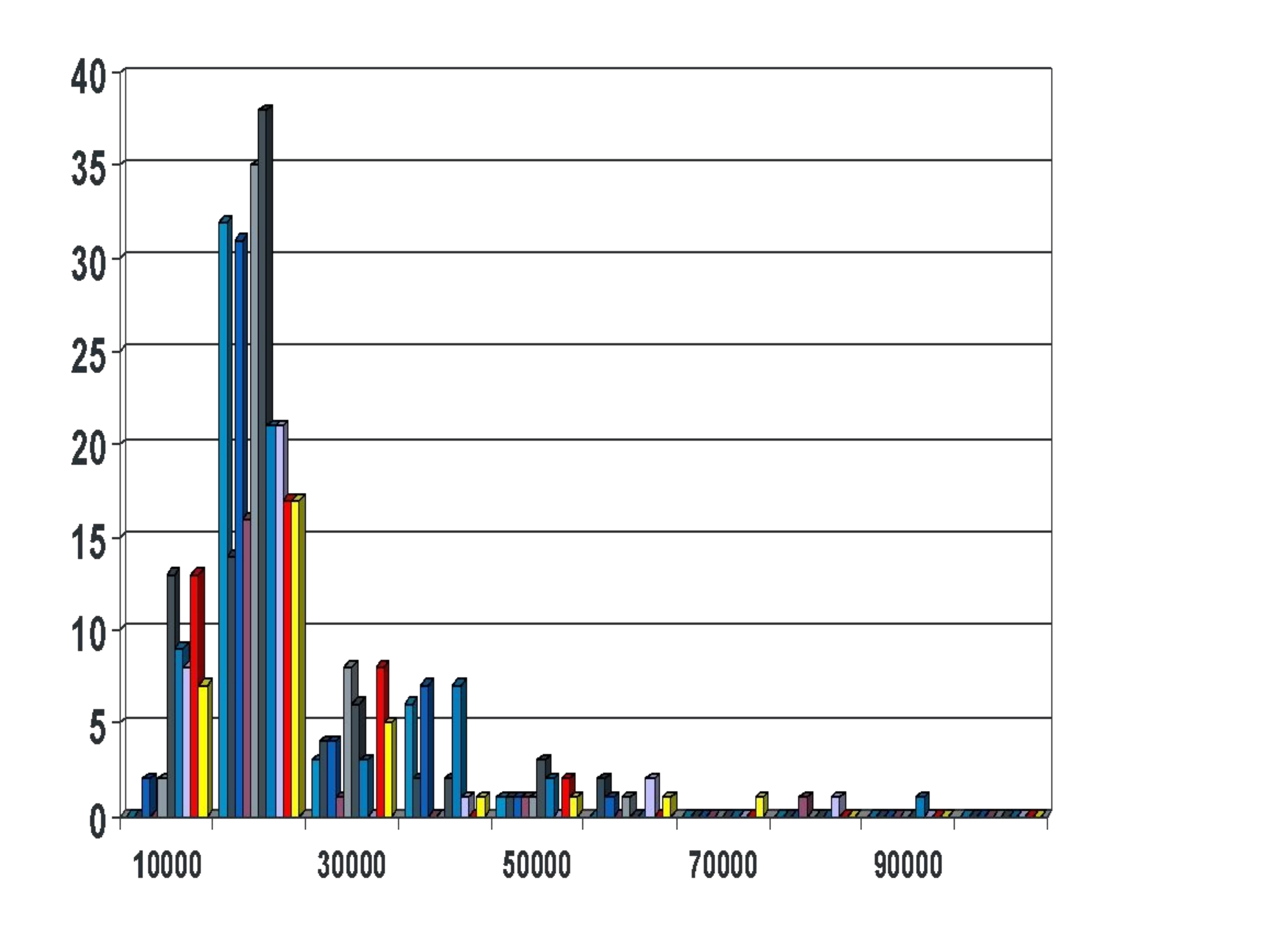
Histograms

- Approximate data distributions
- Divide data into buckets and store average (sum) for each bucket
- A bucket represents an attribute-value/frequency pair
- Can be constructed optimally in one dimension using dynamic programming
- Related to quantization problems.



Histogram

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)



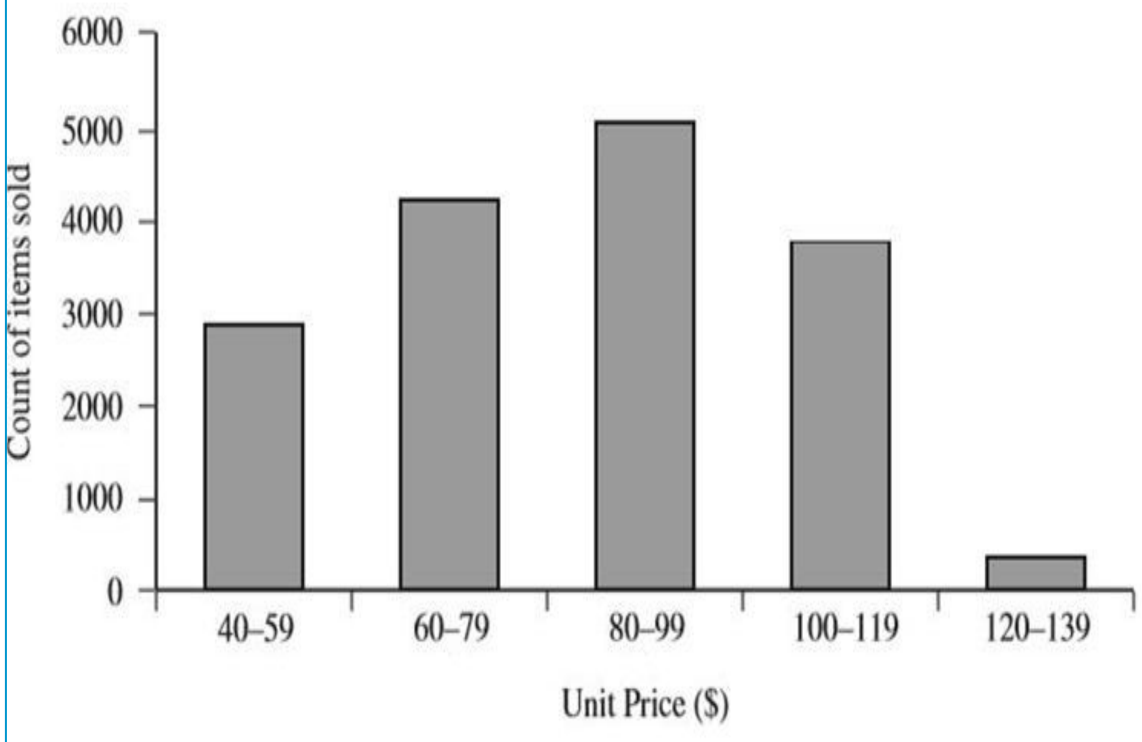
Histogram Analysis- Explanation & Example

- Plotting histograms, or frequency histograms, is a graphical method for summarizing the distribution of a given attribute.
- A histogram for an attribute A partitions the data distribution of A into disjoint subsets, or buckets. Typically, the width of each bucket is uniform.
- Each bucket is represented by a rectangle whose height is equal to the count or relative frequency of the values at the bucket.
- The following data are a list of AllElectronics prices for commonly sold items (rounded to the nearest dollar).

Histogram Analysis- Explanation & Example

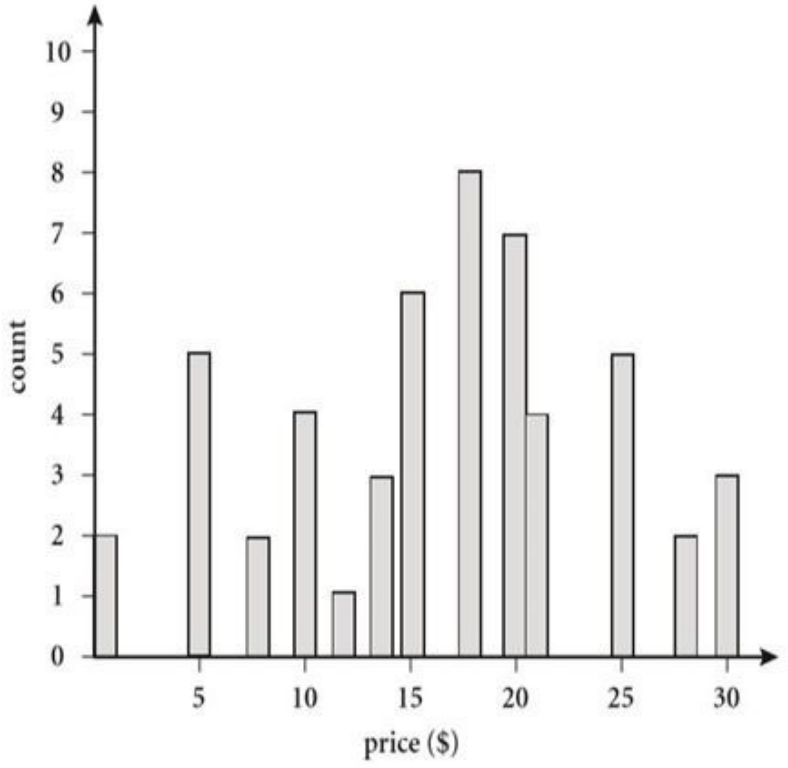
A set of unit price data for items sold at a branch of *AllElectronics*.

Unit price ($)	*Count of items sold*
40	275
43	300
47	250
..	..
74	360
75	515
78	540
..	..
115	320
117	270
120	350



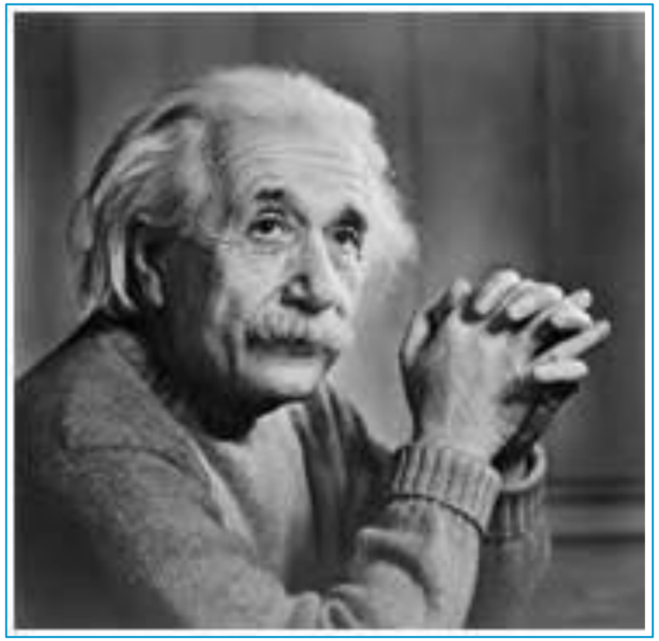
Histogram Analysis- Explanation & Example

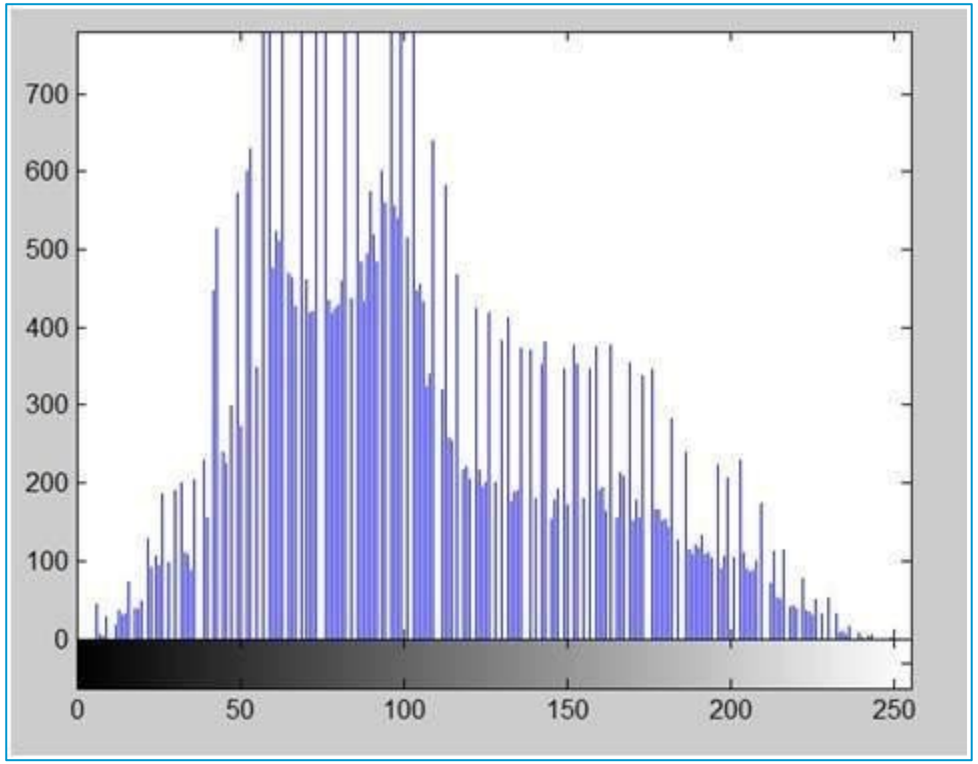
- The following data are a list of prices of commonly sold items at AllElectronics (rounded to the nearest dollar).
- The numbers have been sorted:
  - 1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, 30, 30.



A histogram for *price* using singleton buckets—each bucket represents one price-value/ frequency pair.

Histogram of an image : Application



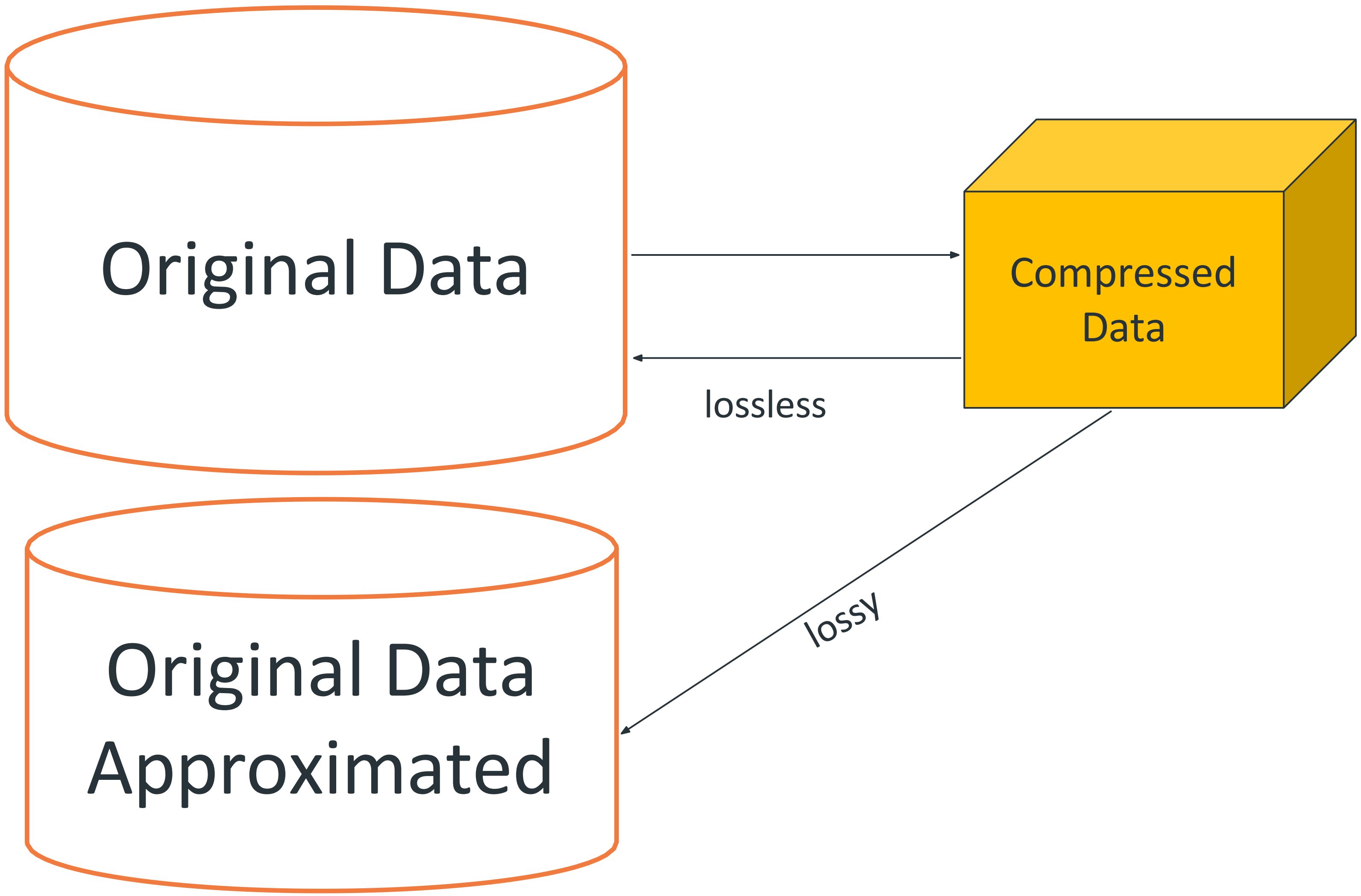


As you can see from the graph, that most of the bars that have high frequency lies in the first half portion which is the darker portion. That means that the image we have got is darker

Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless
  - But only limited manipulation is possible without expansion
- Audio/video, image compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence is not audio
  - Typically short and vary slowly with time

Data Compression



Clustering

- Partition data set into clusters, and store cluster representation only
- Quality of clusters measured by their diameter (max distance between any two objects in the cluster) or centroid distance (avg. distance of each cluster object from its centroid)
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering (possibly stored in multi-dimensional index tree structures (B+-tree, R-tree, quad-tree, etc))
- There are many choices of clustering definitions and clustering algorithms (further details later)

Sampling

- Sampling: obtaining a small sample *s* to represent the whole data set *N*
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Key principle: Choose a representative subset of the data
  - Simple random sampling may have very poor performance in the presence of skew
  - Develop adaptive sampling methods, e.g., stratified sampling:
  - In **stratified sampling, researchers divide subjects into subgroups called strata based on characteristics that they share (e.g., race, gender, educational attainment). Once divided, each subgroup is randomly sampled using another probability sampling method.**
  - Note: Sampling may not reduce database I/Os (page at a time)

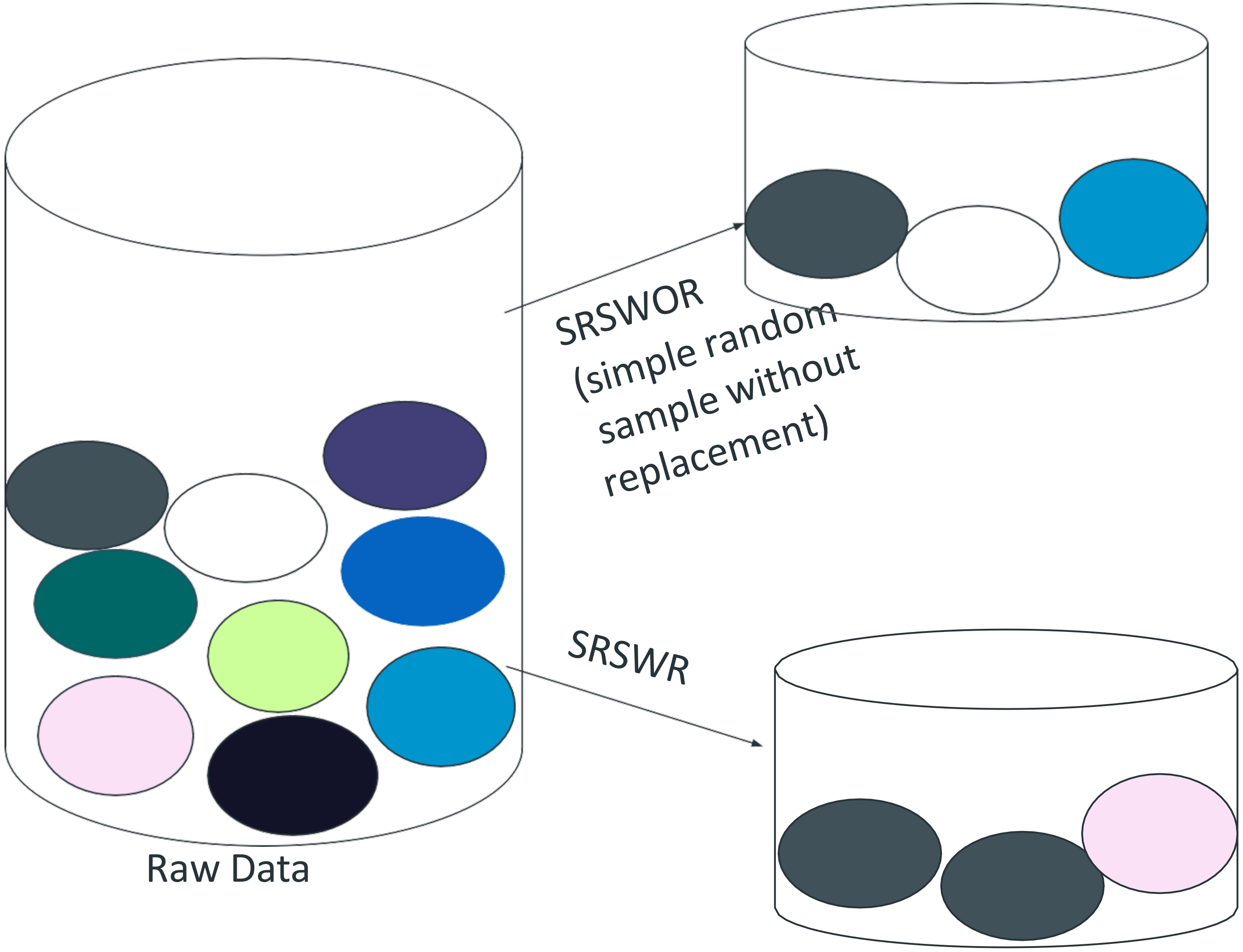
Sampling

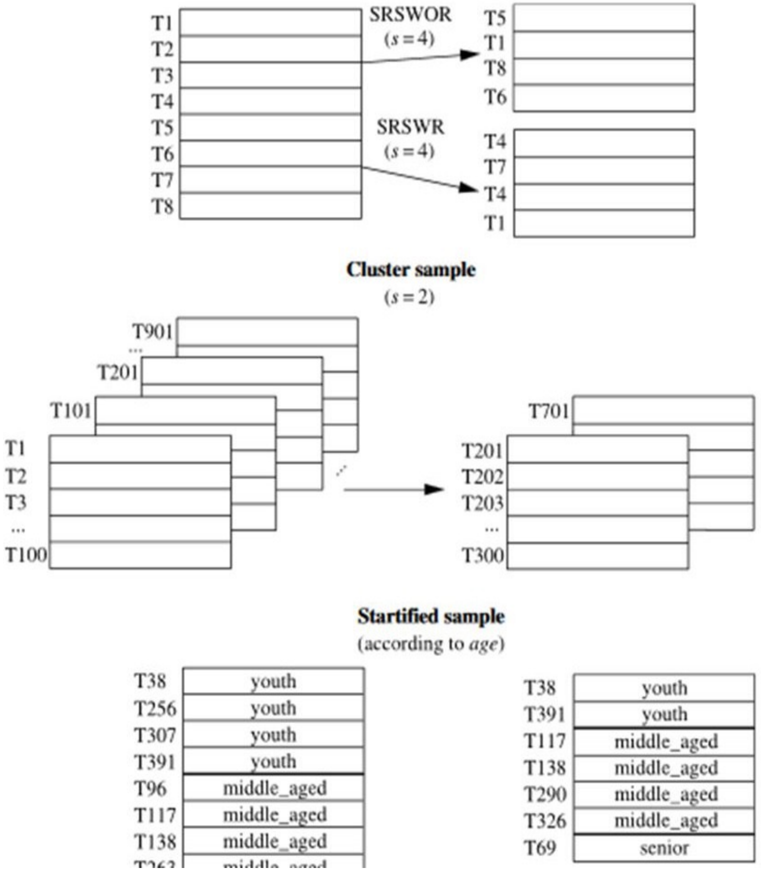
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Cost of sampling: proportional to the size of the sample, increases linearly with the number of dimensions
- Choose a representative subset of the data
  - Simple random sampling may have very poor 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 may not reduce database I/Os (page at a time).
- Sampling: natural choice for progressive refinement of a reduced data set.

Types of Sampling

- **Simple random sampling**
  - There is an equal probability of selecting any particular item
- **Sampling without replacement**
  - Once an object is selected, it is removed from the population
- **Sampling with replacement**
  - A selected object is not removed from the population
- **Stratified sampling:**
  - Partition the data set, and draw samples from each partition (proportionally, i.e., approximately the same percentage of the data)
  - Used in conjunction with skewed data

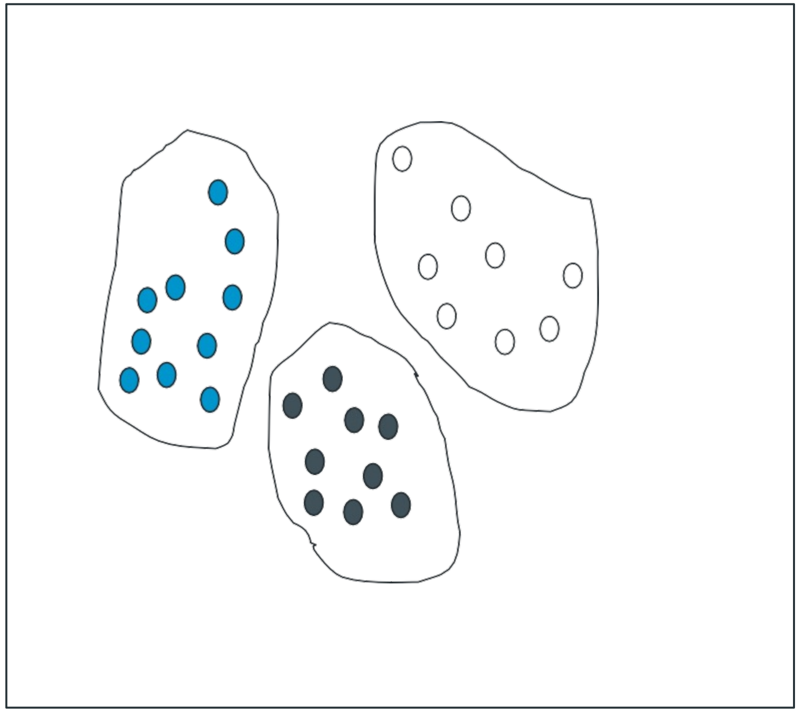
Types of Sampling



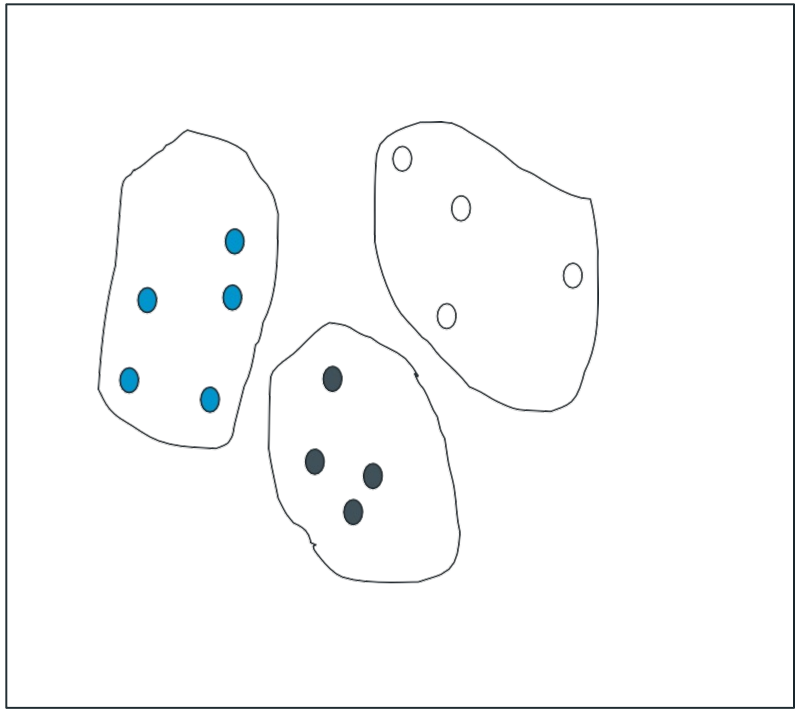


Sampling

Raw Data



Cluster/Stratified Sample



Discretization

- Three types of attributes
  - Nominal—values from an unordered set, e.g., color, profession
  - Ordinal—values from an ordered set, e.g., military or academic rank
  - Numeric—real numbers, e.g., integer or real numbers
- Discretization: Divide the range of a continuous attribute into intervals
- Discretization is one form of data transformation technique. It transforms numeric values to interval labels of conceptual labels. Ex. age can be transformed to (0-10,11-20….) or to conceptual labels like youth, adult, senior
  - Interval labels can then be used to replace actual data values
  - Reduce data size by discretization
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
  - Prepare for further analysis, e.g., classification

Data Discretization

- Three types of attributes:
  - Nominal — values from an unordered set
  - Ordinal — values from an ordered set
  - Continuous — real numbers
- Discretization/Quantization:
  - divide the range of a continuous attribute into intervals

  - Some classification algorithms only accept categorical attributes.
  - Reduce data size by discretization
  - Prepare for further analysis

Discretization and Concept Hierarchies

- Discretization
  - reduce 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.
- Concept Hierarchies
  - reduce the data by collecting and replacing low level concepts (such as numeric values for the attribute age) by higher level concepts (such as young, middle-aged, or senior).

Discretization and Concept Hierarchies : Numerical data

- Hierarchical and recursive decomposition using:
  - Binning (data smoothing)
  - Histogram analysis (numerosity reduction)
  - Clustering analysis (numerosity reduction)
- Segmentation by natural partitioning

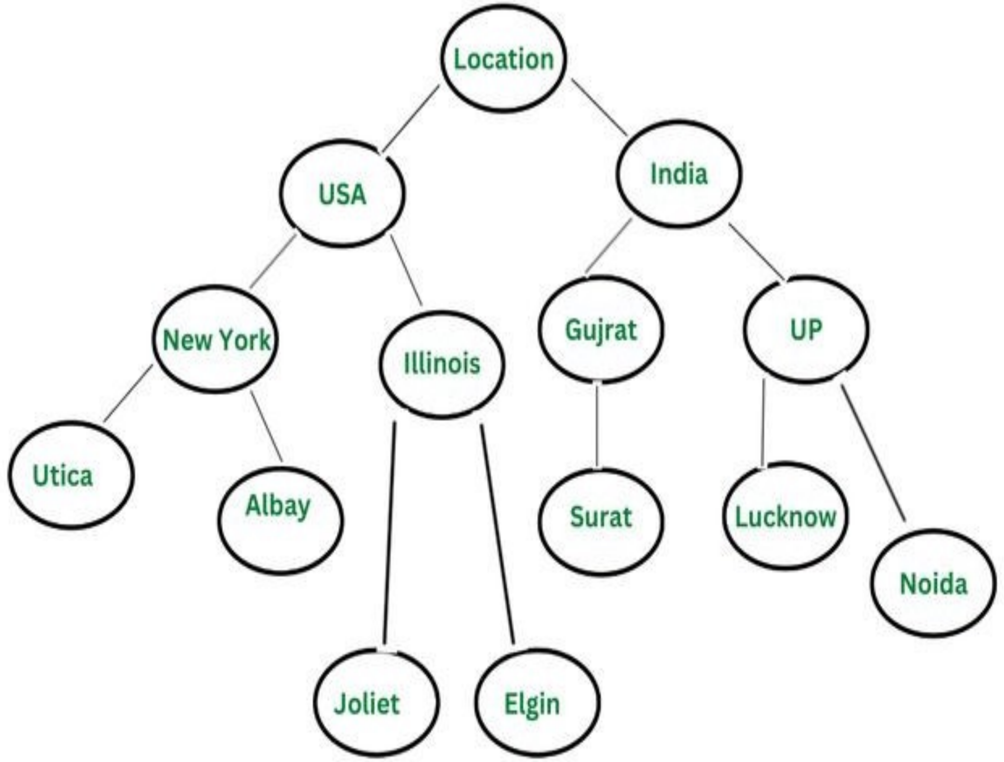
Data Discretization Methods

- Typical methods: All the methods can be applied recursively
  - Binning
    - Top-down split, unsupervised
  - Histogram analysis
    - Top-down split, unsupervised
  - Clustering analysis (unsupervised, top-down split or bottom-up merge)
  - Decision-tree analysis (supervised, top-down split)
  - Correlation (e.g., χ^2) analysis (unsupervised, bottom-up merge)

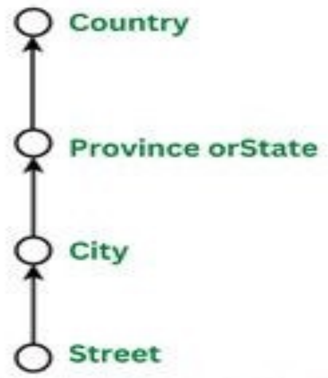
Concept of Hierarchy in Nominal Data

- In data mining, the concept of a concept hierarchy refers to the organization of data into a tree-like structure, where each level of the hierarchy represents a concept that is more general than the level below it.
- This hierarchical organization of data allows for more efficient and effective data analysis, as well as the ability to drill down to more specific levels of detail when needed.
- The concept of hierarchy is used to organize and classify data in a way that makes it more understandable and easier to analyze.
- The main idea behind the concept of hierarchy is that the same data can have different levels of granularity or levels of detail and that by organizing the data in a hierarchical fashion, it is easier to understand and perform analysis.

Concept of Hierarchy in Nominal Data



Concept Hierarchy for Dimension Location



Hierarchical Structure for Dimension Location

The top of the tree consists of the main dimension location and further splits into various sub-nodes. The root node is located, and it further splits into two nodes countries i.e. USA and India. These countries are further then splitted into more sub-nodes, that represent the province states ie. New York, Illinois, Gujarat, UP.

Need of Concept Hierarchy in Data Mining

- **Improved Data Analysis**: A concept hierarchy can help to organize and simplify data, making it more manageable and easier to analyze. By grouping similar concepts together, a concept hierarchy can help to identify patterns and trends in the data that would otherwise be difficult to spot.
- **Improved Data Visualization and Exploration**: A concept hierarchy can help to improve data visualization and data exploration by organizing data into a tree-like structure, allowing users to easily navigate and understand large and complex data sets.

Need of Concept Hierarchy in Data Mining

- **Improved Algorithm Performance**: The use of a concept hierarchy can also help to improve the performance of data mining algorithms. By organizing data into a hierarchical structure, algorithms can more easily process and analyze the data, resulting in faster and more accurate results.
- **Data Cleaning and Pre-processing**: A concept hierarchy can also be used in data cleaning and pre-processing, to identify and remove outliers and noise from the data.
- **Domain Knowledge**: A concept hierarchy can also be used to represent the domain knowledge in a more structured way, which can help in a better understanding of the data and the problem domain.

Applications of Concept Hierarchy

- **Data Warehousing**: Concept hierarchy can be used in data warehousing to organize data from multiple sources into a single, consistent and meaningful structure. This can help to improve the efficiency and effectiveness of data analysis and reporting.
- **Business Intelligence**: Concept hierarchy can be used in business intelligence to organize and analyze data in a way that can inform business decisions. For example, it can be used to analyze customer data to identify patterns and trends that can inform the development of new products or services.
- **Online Retail**: Concept hierarchy can be used in online retail to organize products into categories, subcategories and sub-subcategories, it can help customers to find the products they are looking for more quickly and easily.
- **Healthcare**: Concept hierarchy can be used in healthcare to organize patient data, for example, to group patients by diagnosis or treatment plan, it can help to identify patterns and trends that can inform the development of new treatments or improve the effectiveness of existing treatments.
- **Natural Language Processing**: Concept hierarchy can be used in natural language processing to organize and analyze text data, for example, to identify topics and themes in a text, it can help to extract useful information from unstructured data.
- **Fraud Detection**: Concept hierarchy can be used in fraud detection to organize and analyze financial data, for example, to identify patterns and trends that can indicate fraudulent activity.

Summary

- **Data quality**: accuracy, completeness, consistency, timeliness, believability, interpretability
- **Data cleaning**: e.g. missing/noisy values, outliers
- **Data integration** from multiple sources:
  - Entity identification problem
  - Remove redundancies
  - Detect inconsistencies
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

Reference

- Data Mining: Concepts and Techniques, Jiawei Han, Micheline Kamber, and Jian Pei, 3rd edition

Reference

- Data Mining: Concepts and Techniques, Jiawei Han, Micheline Kamber, and Jian Pei, 3rd edition
- https://towardsdatascience.com/tagged/data-engineering
- https://www.coursera.org/articles/
- https://www.nptel.org/