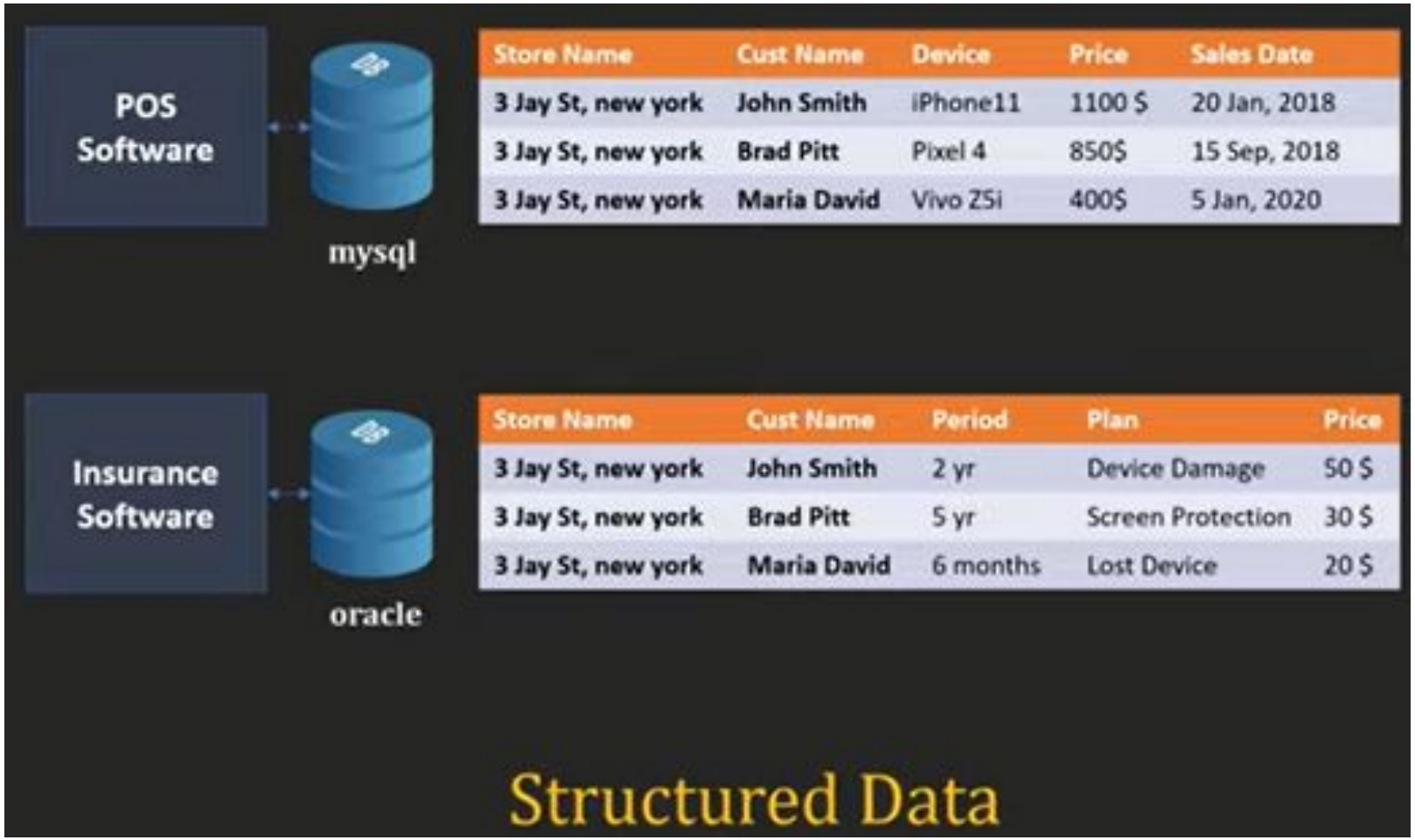
MODULE 1: INTRODUCTION TO DATA MINING

DATA WAREHOUSING & DATA MINING

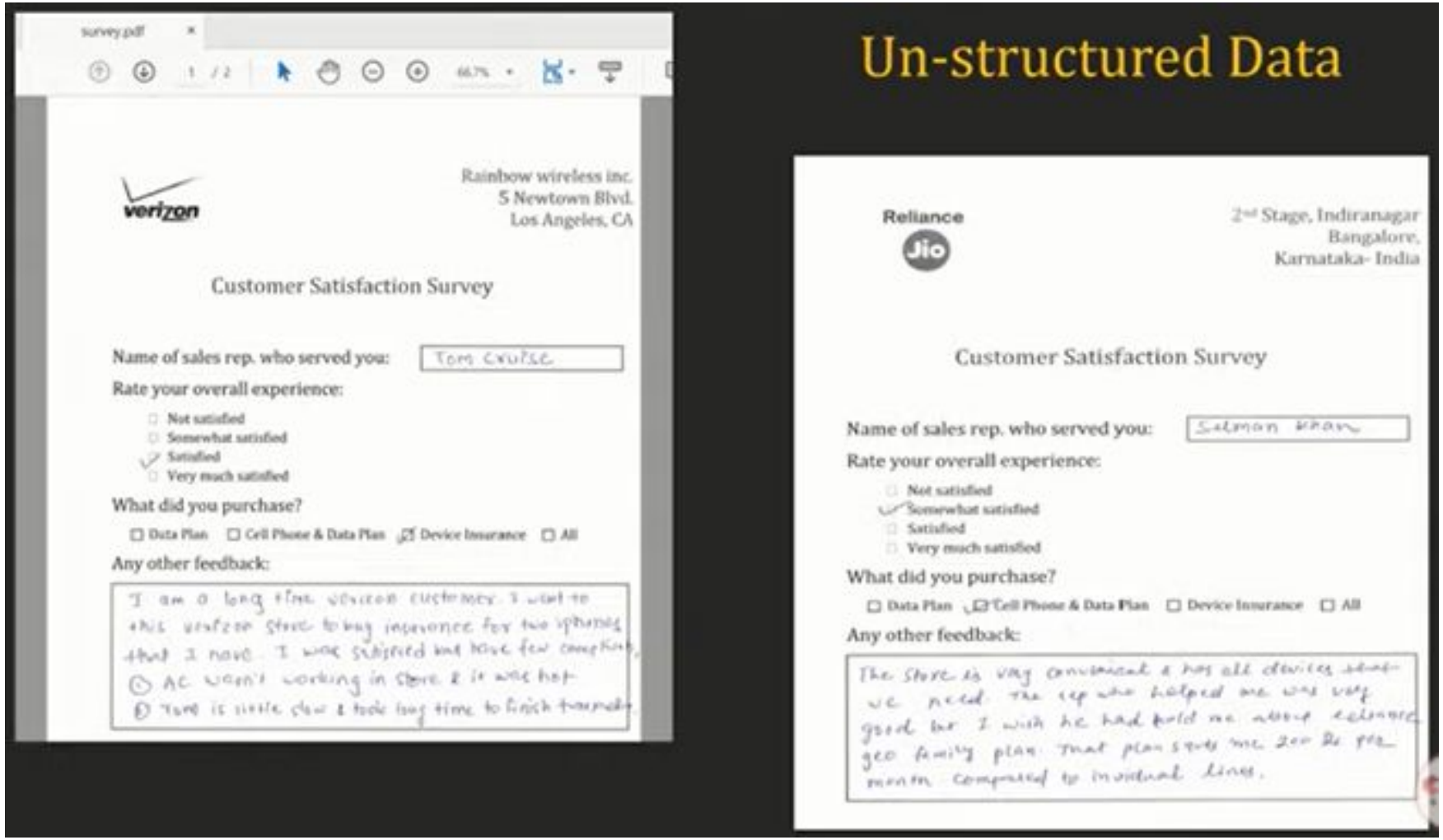
By,

Mrs. Apeksha Waghmare
AP, IT Dept. TCET

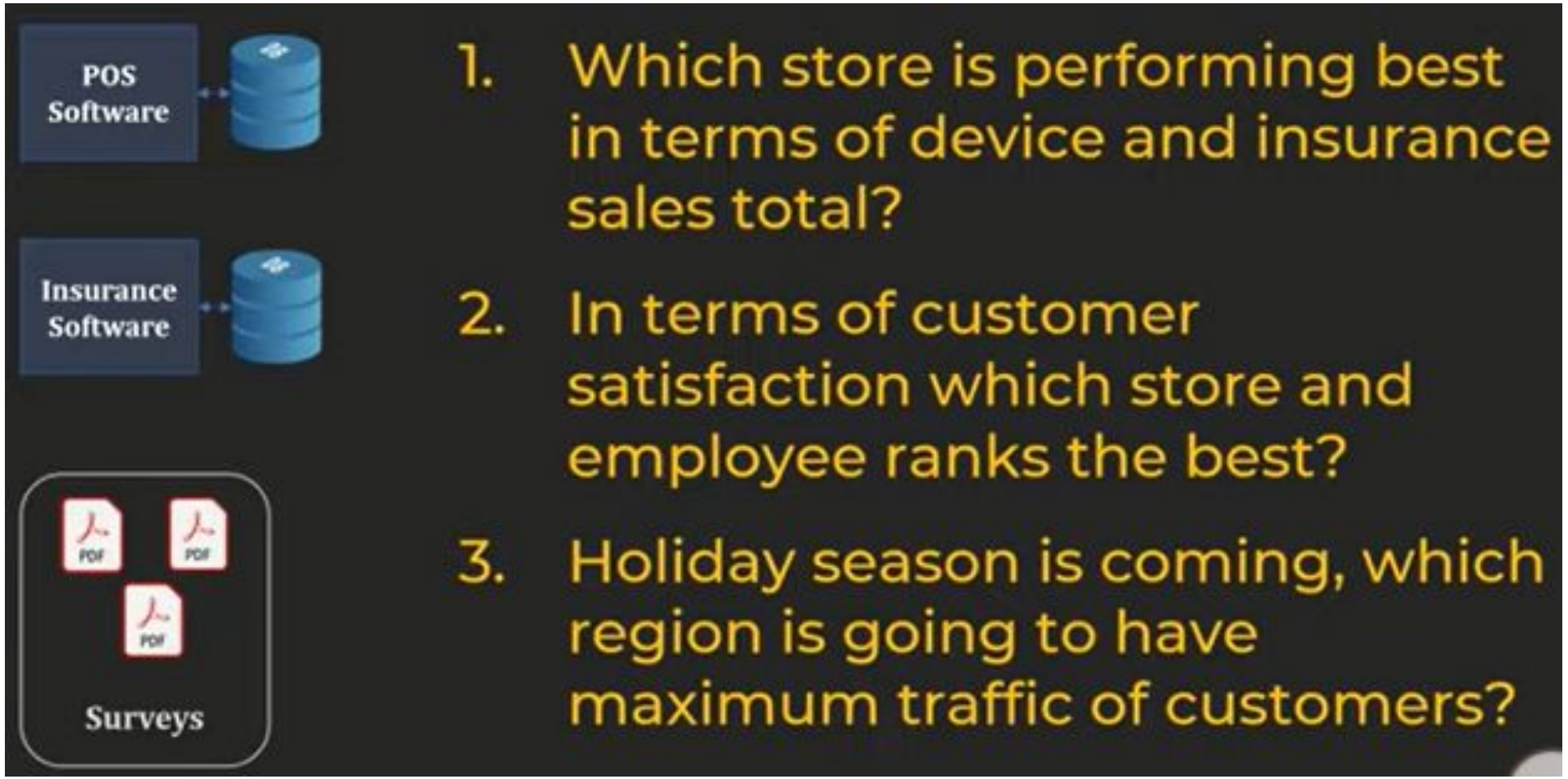
STRUCTURED DATA



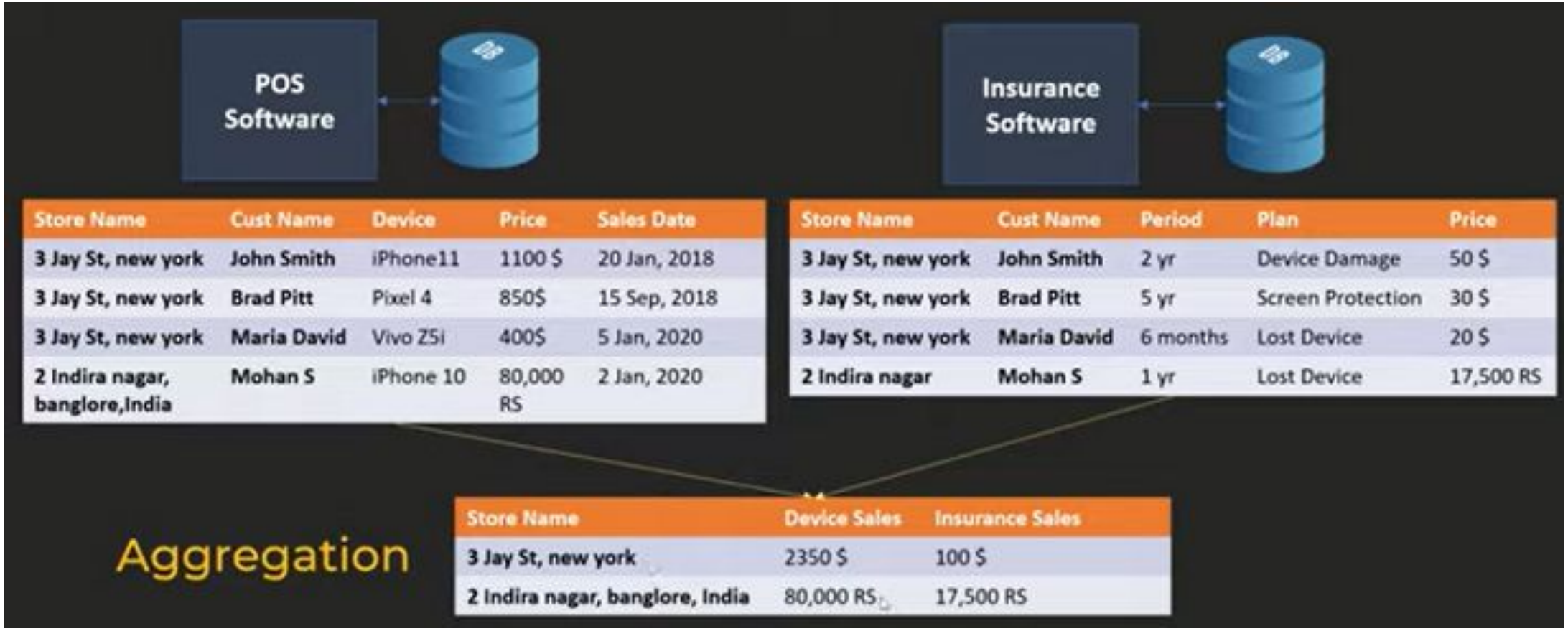
UN-STRUCTURED DATA



QUESTIONS TO BE ANSWERED…



AGGREGATION

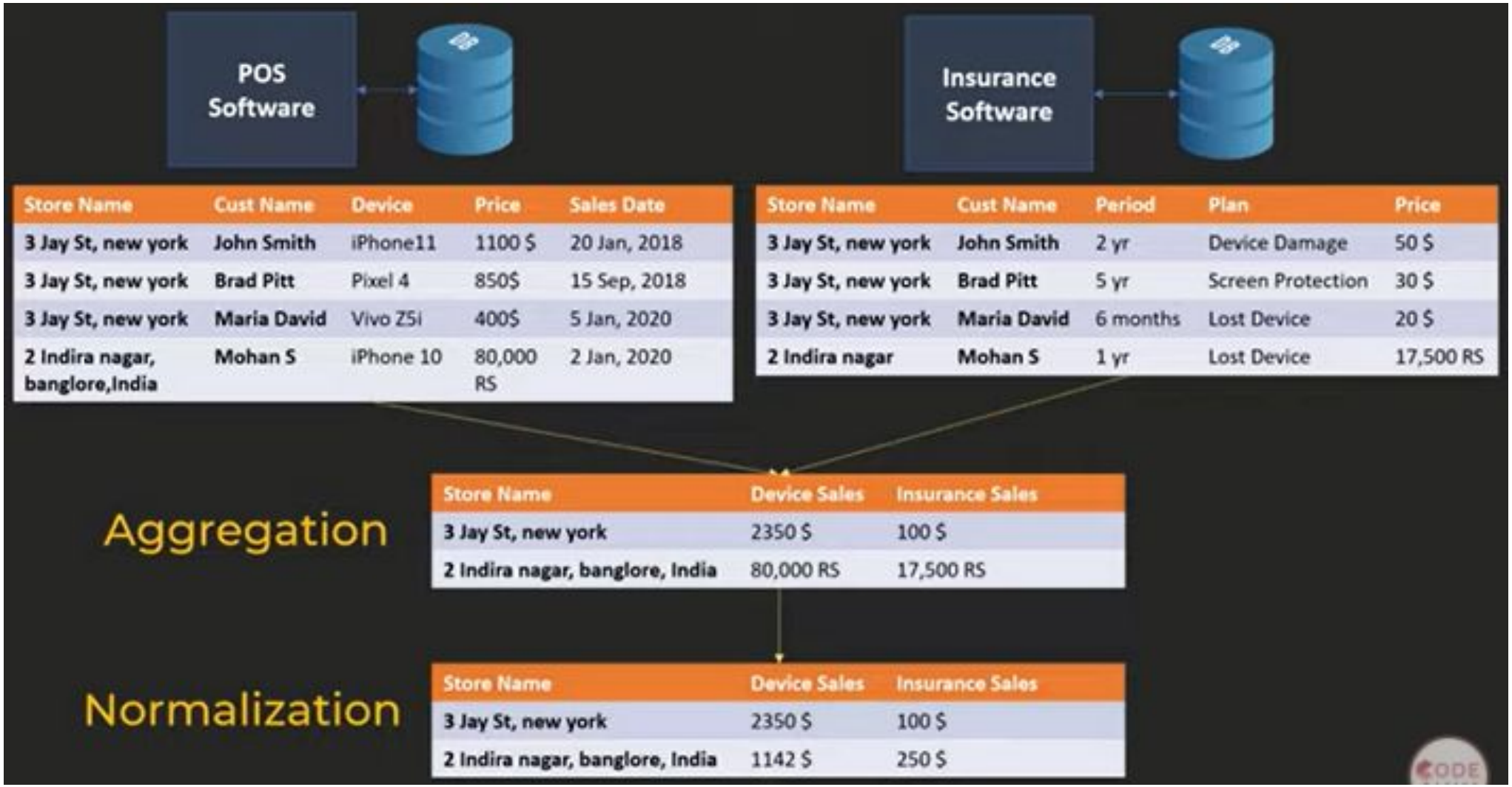


Store Name	Cust Name	Device	Price	Sales Date
3 Jay St, new york	John Smith	iPhone11	1100 $	20 Jan, 2018
3 Jay St, new york	Brad Pitt	Pixel 4	850$	15 Sep, 2018
3 Jay St, new york	Maria David	Vivo Z5i	400$	5 Jan, 2020
2 Indira nagar, banglore,India	Mohan S	iPhone 10	80,000 RS	2 Jan, 2020

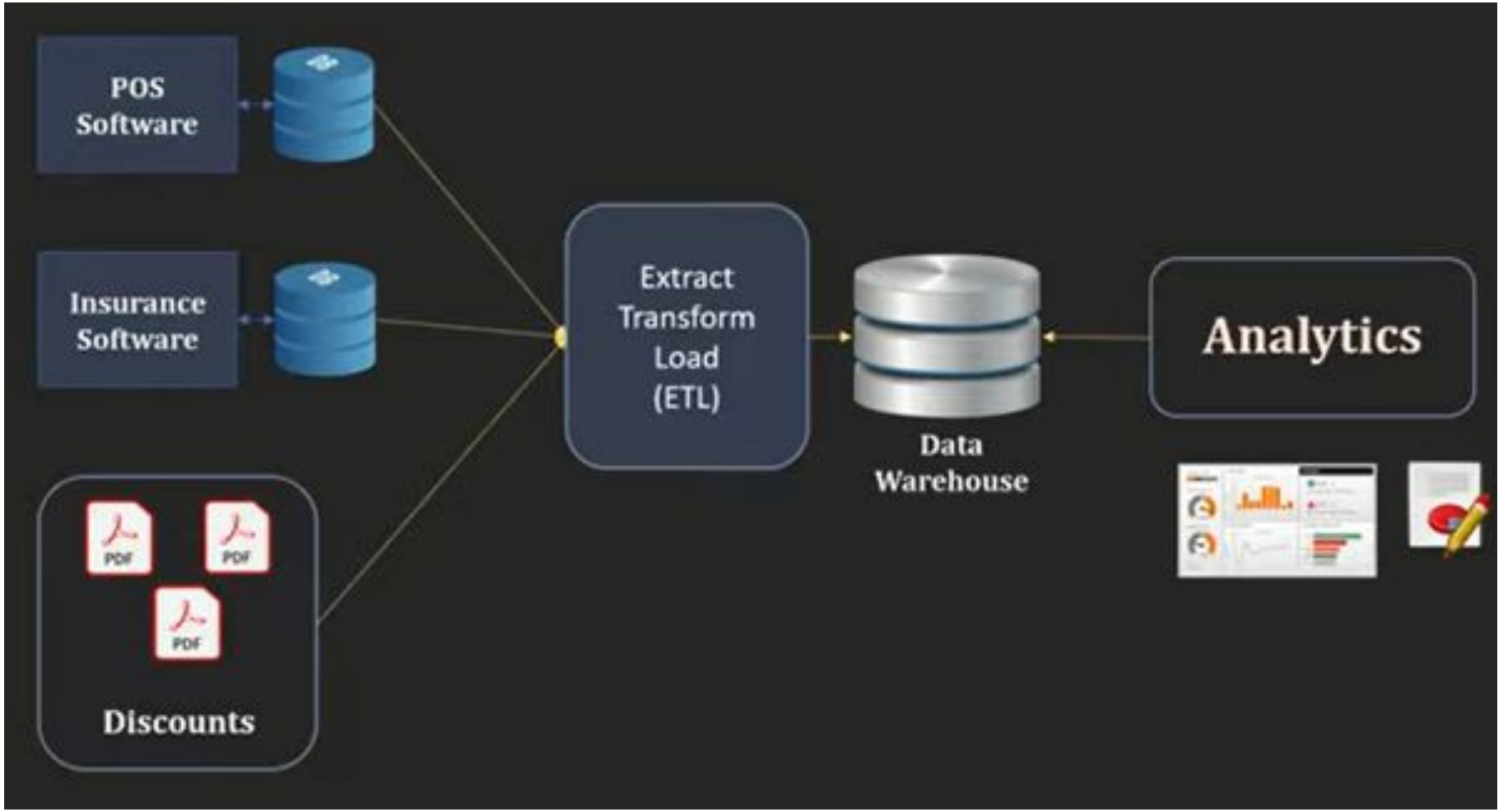
Store Name	Cust Name	Period	Plan	Price
3 Jay St, new york	John Smith	2 yr	Device Damage	50 $
3 Jay St, new york	Brad Pitt	5 yr	Screen Protection	30 $
3 Jay St, new york	Maria David	6 months	Lost Device	20 $
2 Indira nagar	Mohan S	1 yr	Lost Device	17,500 RS

Store Name	Device Sales	Insurance Sales
3 Jay St, new york	2350 $	100 $
2 Indira nagar, banglore, India	80,000 RS	17,500 RS

NORMALIZATION



DATA WAREHOUSE



ETL PROCESS

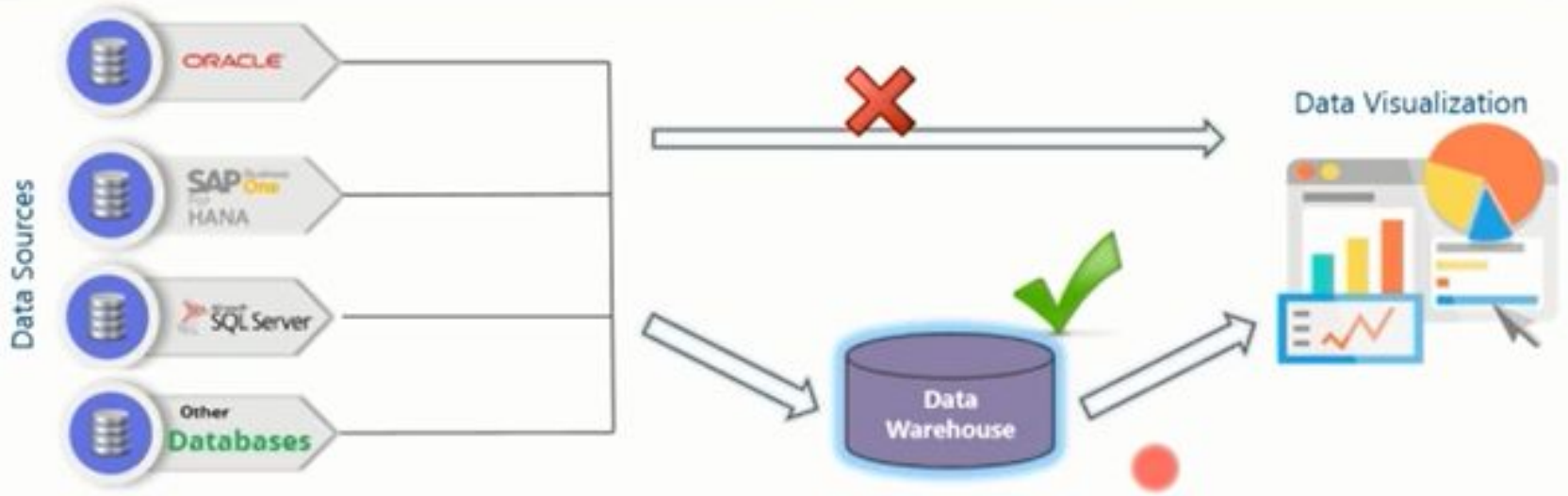
ETL → Extract, Transform & Load

ETL is the process of extracting the data from various sources, transforming this data to meet your requirement and then loading it into a target data warehouse.

ETL TOOLS

Why Data Warehouse?

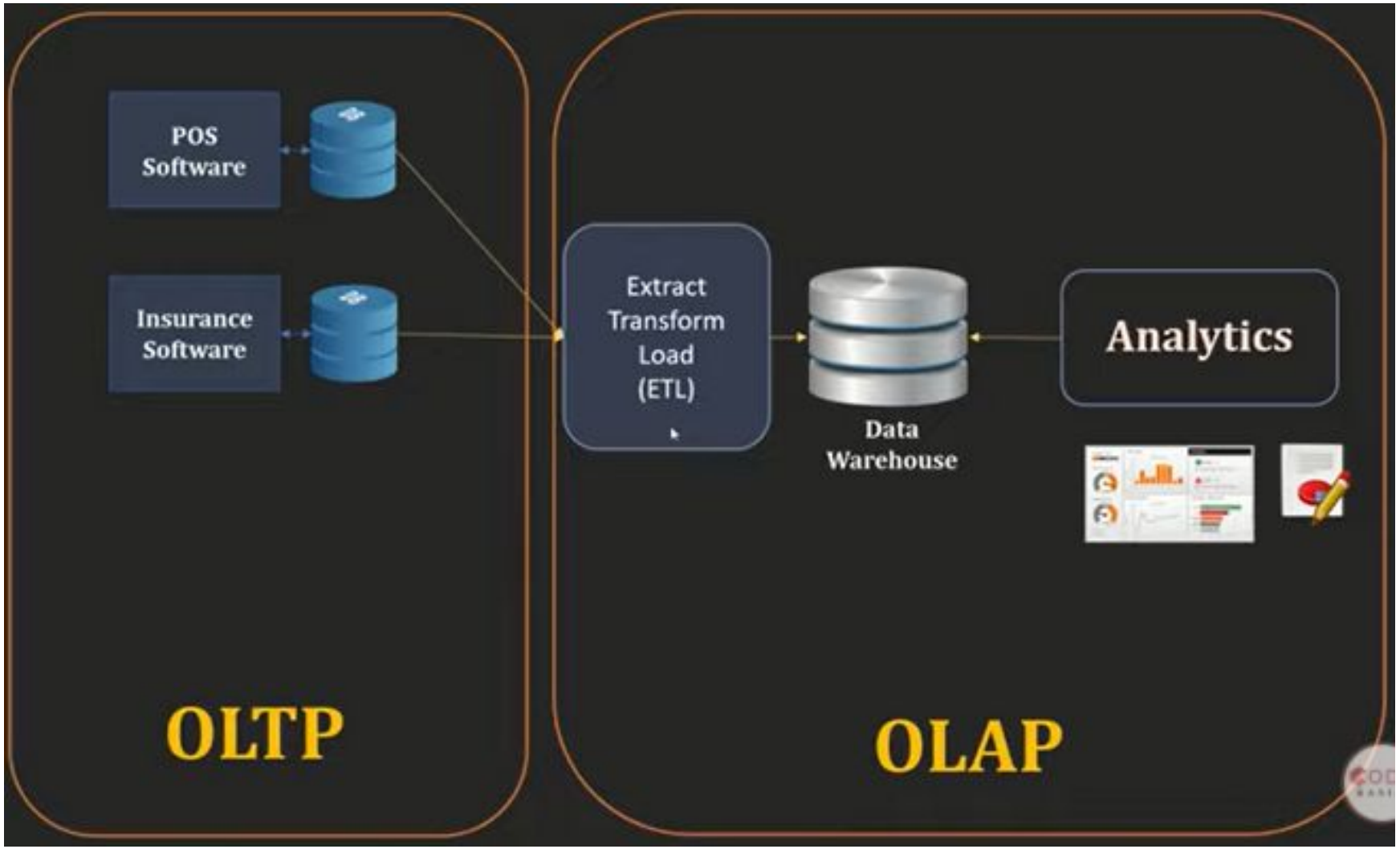
- Data collected from various sources & stored in various databases cannot be directly visualized.
- The data first needs to be **integrated** and then **processed** before visualization takes place.



DATA WAREHOUSES TOOLS

Enterprise Data Warehouses

OLTP/OLAP


POS Software
Insurance Software
Extract Transform Load (ETL)
Data Warehouse
Analytics
OLTP
OLAP

OLTP/OLAP

Information Systems:- OLTP (DB) vs. OLAP (DWH)

Relational Database (OLTP)	Analytical Data Warehouse (OLAP)
Contains current data	Contains historical data
Useful in running the business	Useful in analyzing the business
Based on Entity Relationship Model	Based on Star, Snowflake and Fact Constellation Schema
Provides primitive and highly detailed data	Provides summarized and consolidated data
Used for writing data into the database	Used for reading data from the data warehouse
Database size ranges from 100 MB to 1 GB	Data Warehouse size ranges from 100 GB to 1 TB
Fast; provides high performance	Highly flexible; but not fast
Number of records accessed is in tens	Number of records accessed is in millions
Ex: All bank transactions made by a customer	**Ex**: Bank transactions made by a customer at a particular time.

OLTP/OLAP

Information Systems:- OLTP (DB) vs. OLAP (DWH)

OLTP Examples:

1. A supermarket server which records every single product purchased at that market.
2. A bank server which records every time a transaction is made for a particular account.
3. A railway reservation server which records the transactions of a passenger.

OLAP Examples:

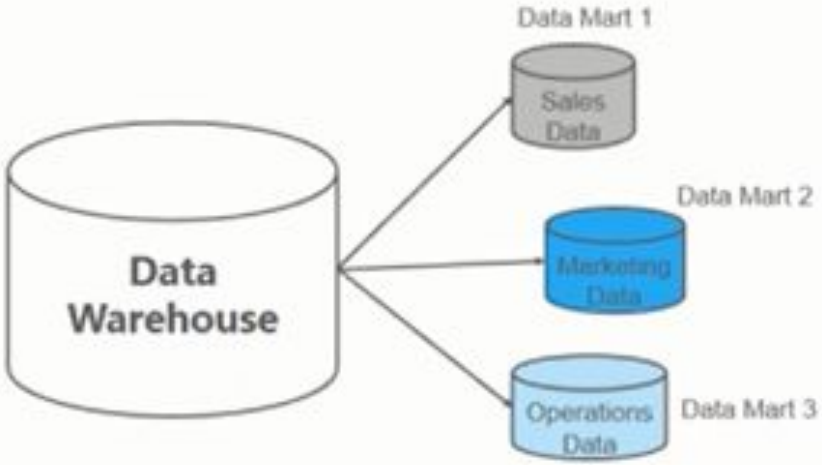
1. Bank Manager wants to know how many customers are utilizing the ATM of his branch. Based on this he may take a call whether to continue with the ATM or relocate it.
2. An insurance company wants to know the number of policies each agent has sold. This will help in better performance management of agents.

DATA MART

Data Mart

- Data mart is a smaller version of the Data Warehouse which deals with a single subject
- Data marts are focused on one area. Hence, they draw data from a limited number of sources
- Time taken to build Data Marts is very less compared to the time taken to build a Data Warehouse

Data Warehouse	Data Marts
Enterprise wide data	Department wide data
Multiple subject areas	Single subject area
Multiple data sources	Limited data sources
Occupies large memory	Occupies limited memory
Longer time to implement	Shorter time to implement



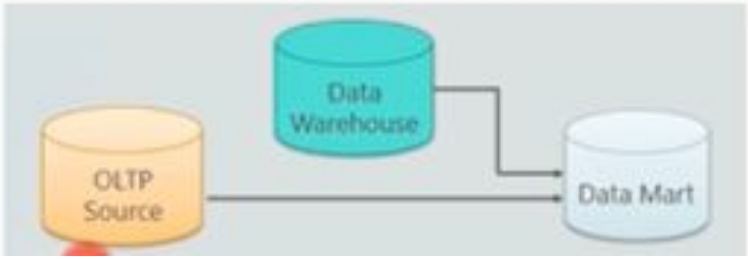
DATA MART

Types Of Data Mart

1. **Dependent Data Mart**
   - The data is first extracted from the OLTP systems and then populated in the central DWH
   - From the DWH, the data travels to the Data Mart

2. **Independent Data Mart**
   - The data is directly received from the source system
   - This is suitable for small organizations or smaller groups within an organization

3. **Hybrid Data Mart**
   - The data is fed both from OLTP systems as well as the Data Warehouse



METADATA

Metadata

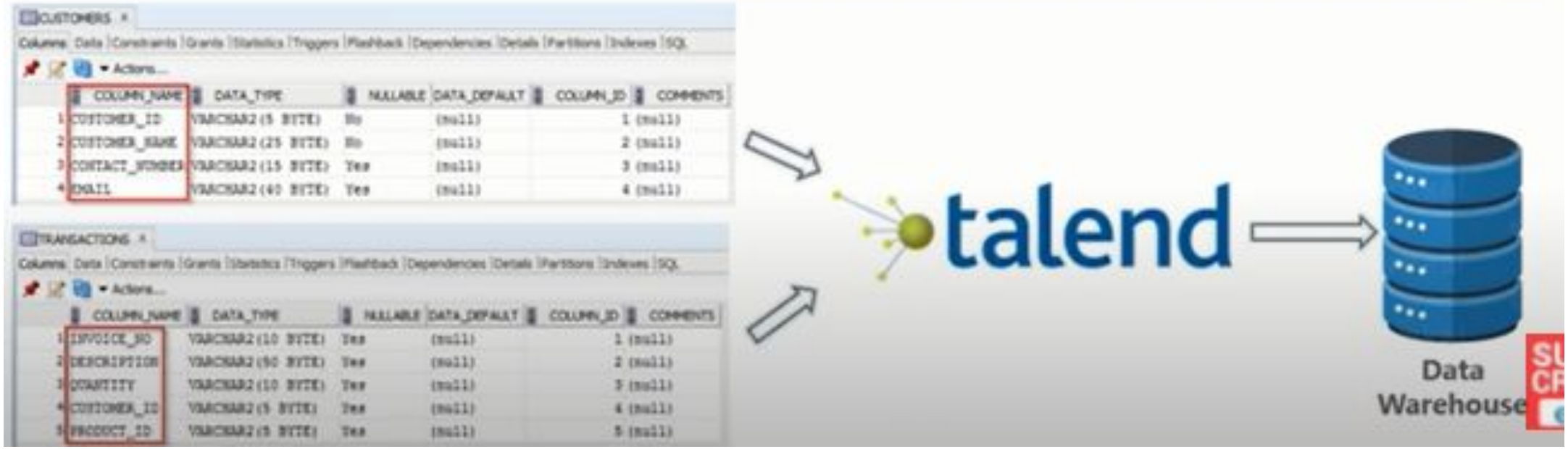
- Metadata is defined as data about data.
- Metadata in a DWH defines the source data i.e. Flat File, Relational Database and other objects.
- Metadata is used to define which table is source and target, and which concept is used to build business logic called transformation to the actual output.

TALEND ETL TOOL

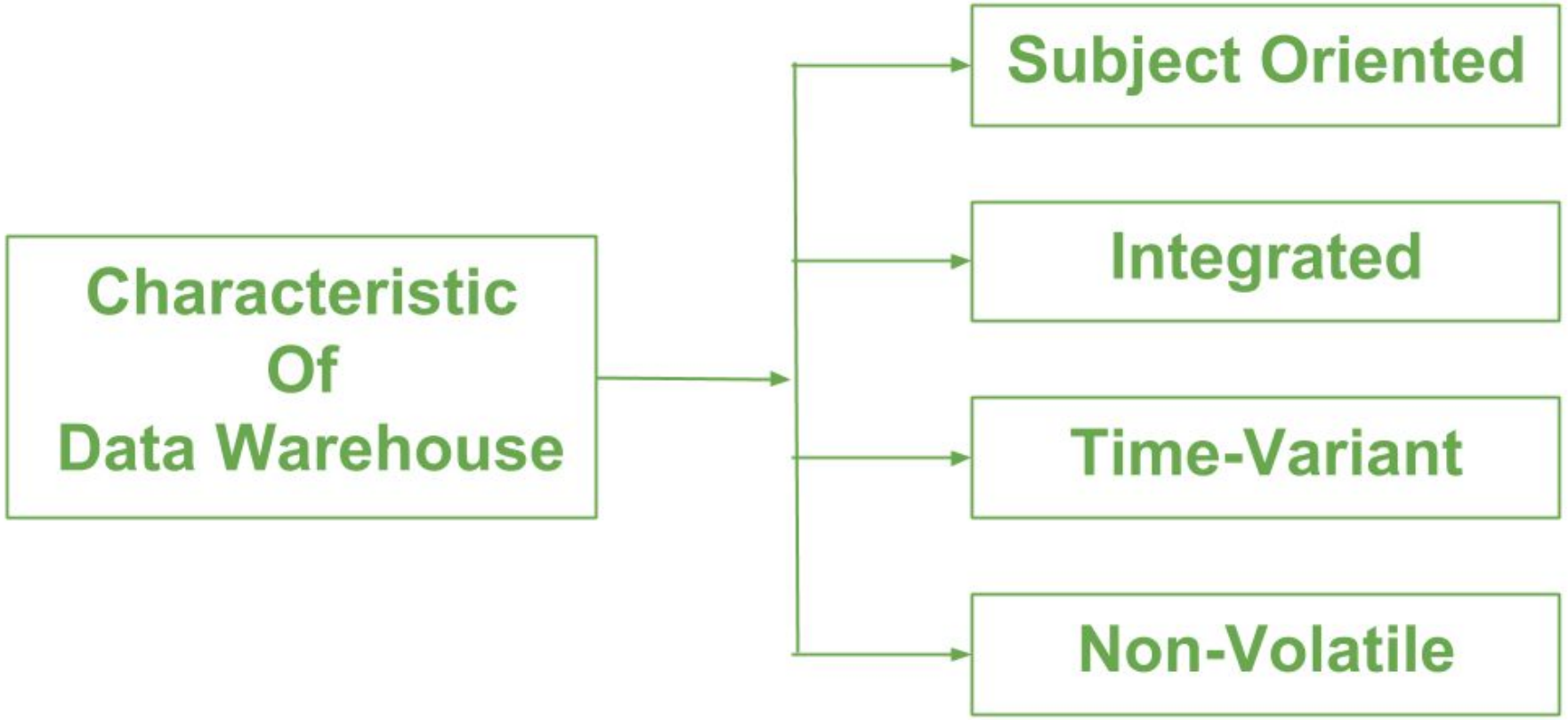
Problem Statement:

As a retail organization, you have details of 10,000 customer and 50,000 transactions. With this data you wish to find out

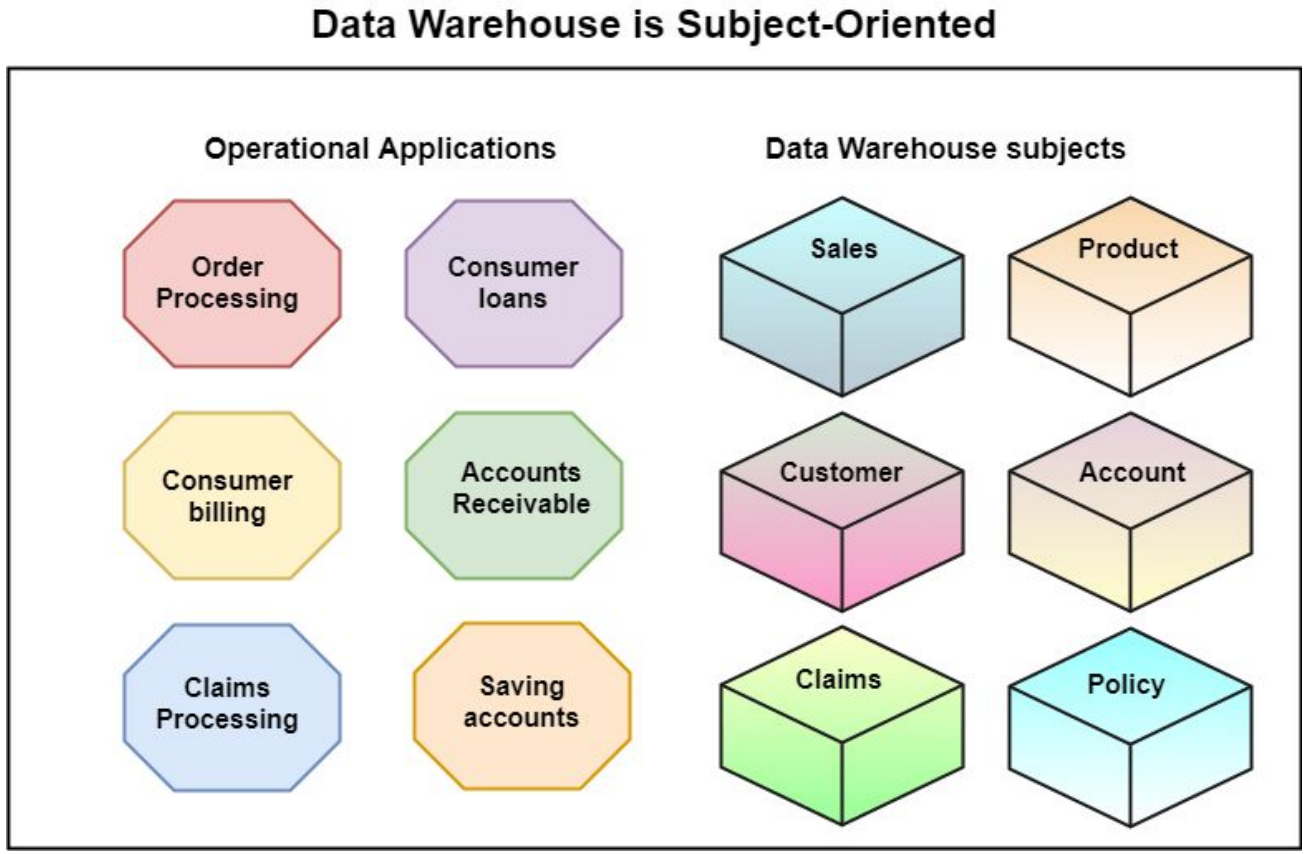
Customers who have low number of purchases.



CHARACTERISTICS OF DATA WAREHOUSE



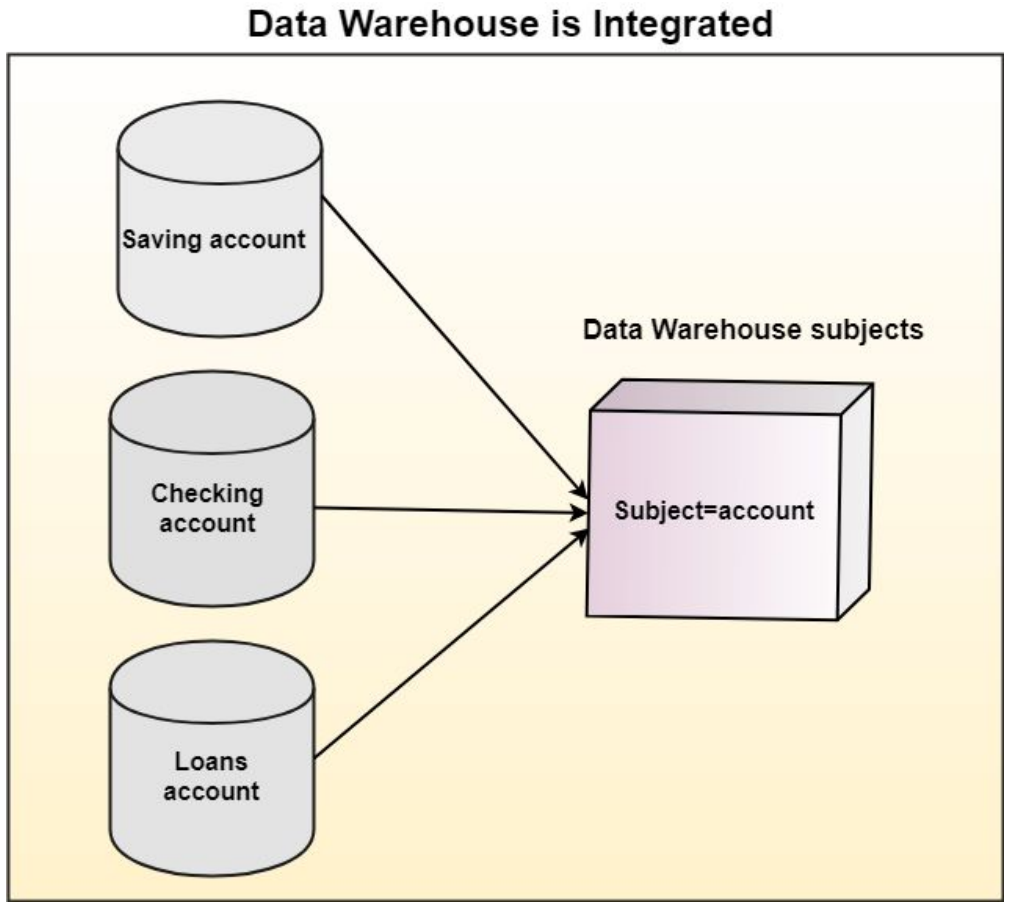
CHARACTERISTICS OF DATA WAREHOUSE



1. Subject-oriented –

- A data warehouse is always a subject oriented as it delivers information about a theme instead of organization's current operations.
- These themes can be sales, distributions, marketing etc.

CHARACTERISTICS OF DATA WAREHOUSE



2. Integrated

- A data warehouse integrates various heterogeneous data sources like RDBMS, flat files, and online transaction records.
- It requires performing data cleaning and integration during data warehousing to ensure consistency in naming conventions, attributes types, etc., among different data sources.

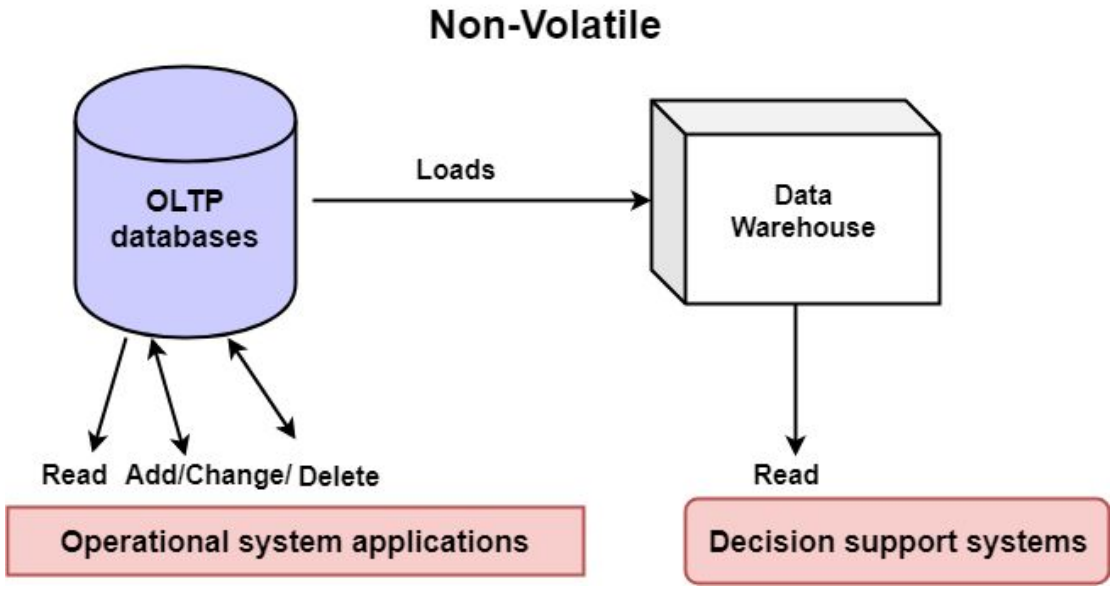
CHARACTERISTICS OF DATA WAREHOUSE

3. Time-Variant

Historical information is kept in a data warehouse. For example, one can retrieve files from 3 months, 6 months, 12 months, or even previous data from a data warehouse.

4. Non-Volatile

Non-Volatile defines that once data entered into the warehouse, and data should not change.



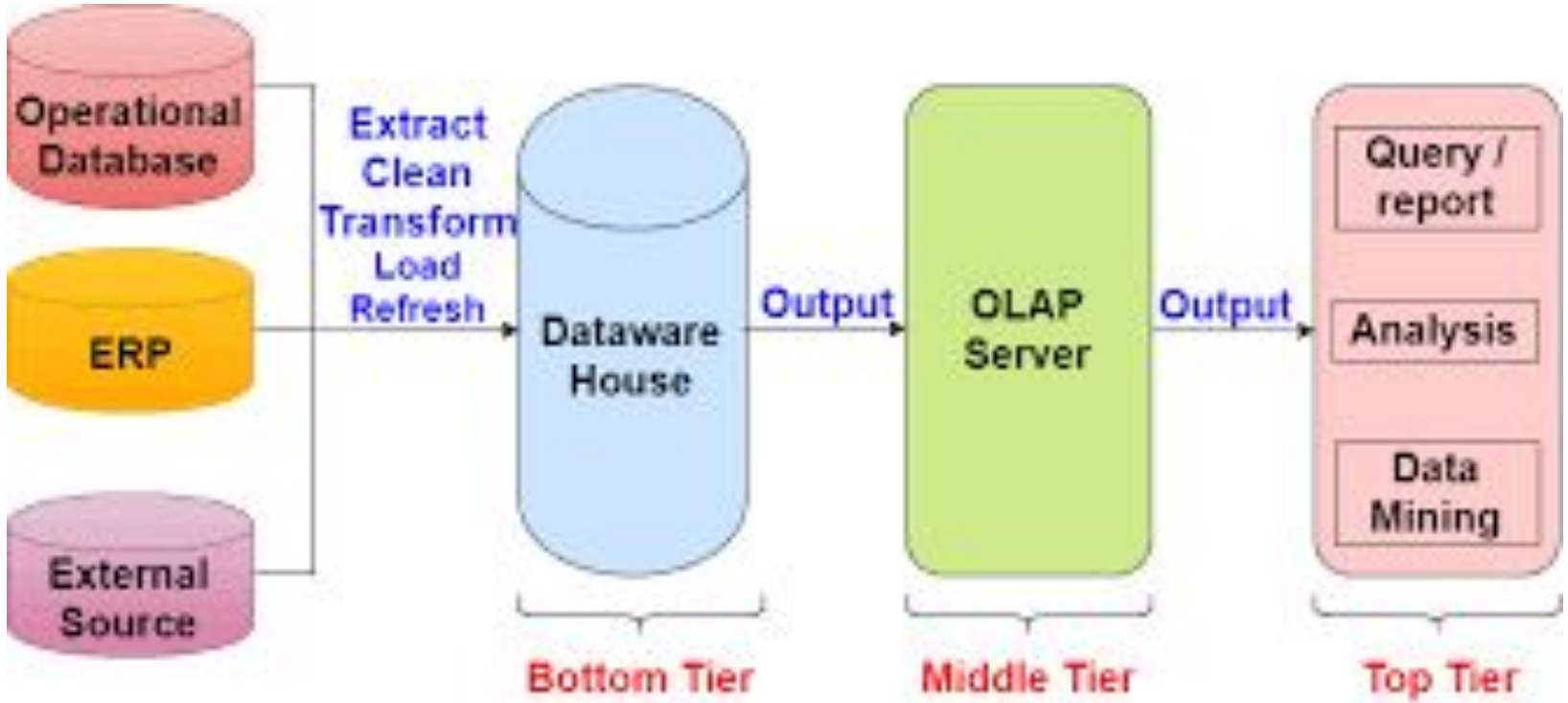
THREE-TIER DATA WAREHOUSE ARCHITECTURE

Bottom Tier: The database of the Data warehouse servers as the bottom tier. It is usually a relational database system. Data is cleansed, transformed, and loaded into this layer using back-end tools.

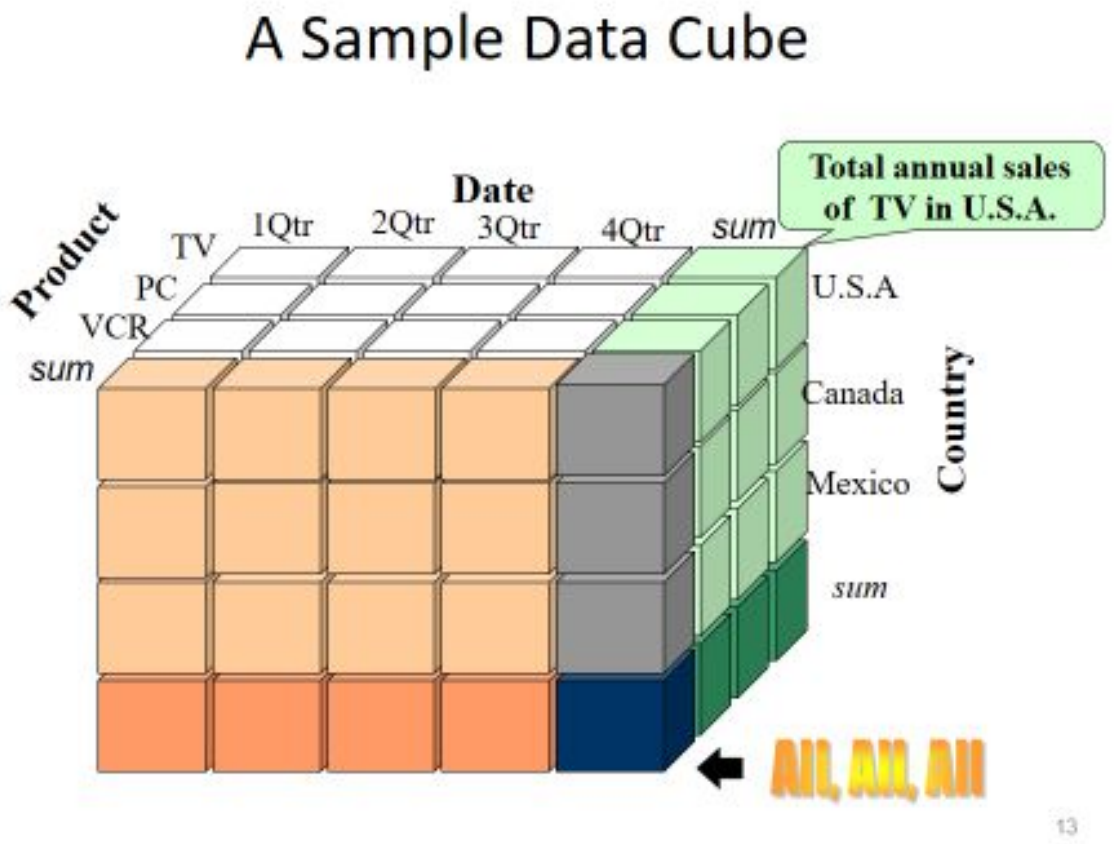
Middle Tier: The middle tier in Data warehouse is an OLAP server which is implemented using either ROLAP or MOLAP model. For a user, this application tier presents an abstracted view of the database. This layer also acts as a mediator between the end-user and the database.

Top-Tier: The top tier is a front-end client layer. Top tier is the tools and API that you connect and get data out from the data warehouse. It could be Query tools, reporting tools, managed query tools, Analysis tools and Data mining tools.

THREE-TIER DATA WAREHOUSE ARCHITECTURE



TYPES OF OLAP SERVERS



1. **Relational OLAP (ROLAP) – Star Schema based –**
   - In ROLAP data is stored in a relational database
2. **Multidimensional OLAP (MOLAP) – Cube based –**
   - MOLAP cubes are fast data retrieval, optimal for slicing and dicing and they can perform complex calculation.
3. **Hybrid OLAP (HOLAP)**
   - HOLAP is a combination of ROLAP and MOLAP
   - Cubes are smaller than MOLAP since detail data is kept in the relational database.

TYPICAL OLAP OPERATIONS

1. Roll up (drill-up): summarize data
   - by climbing up hierarchy or by dimension reduction

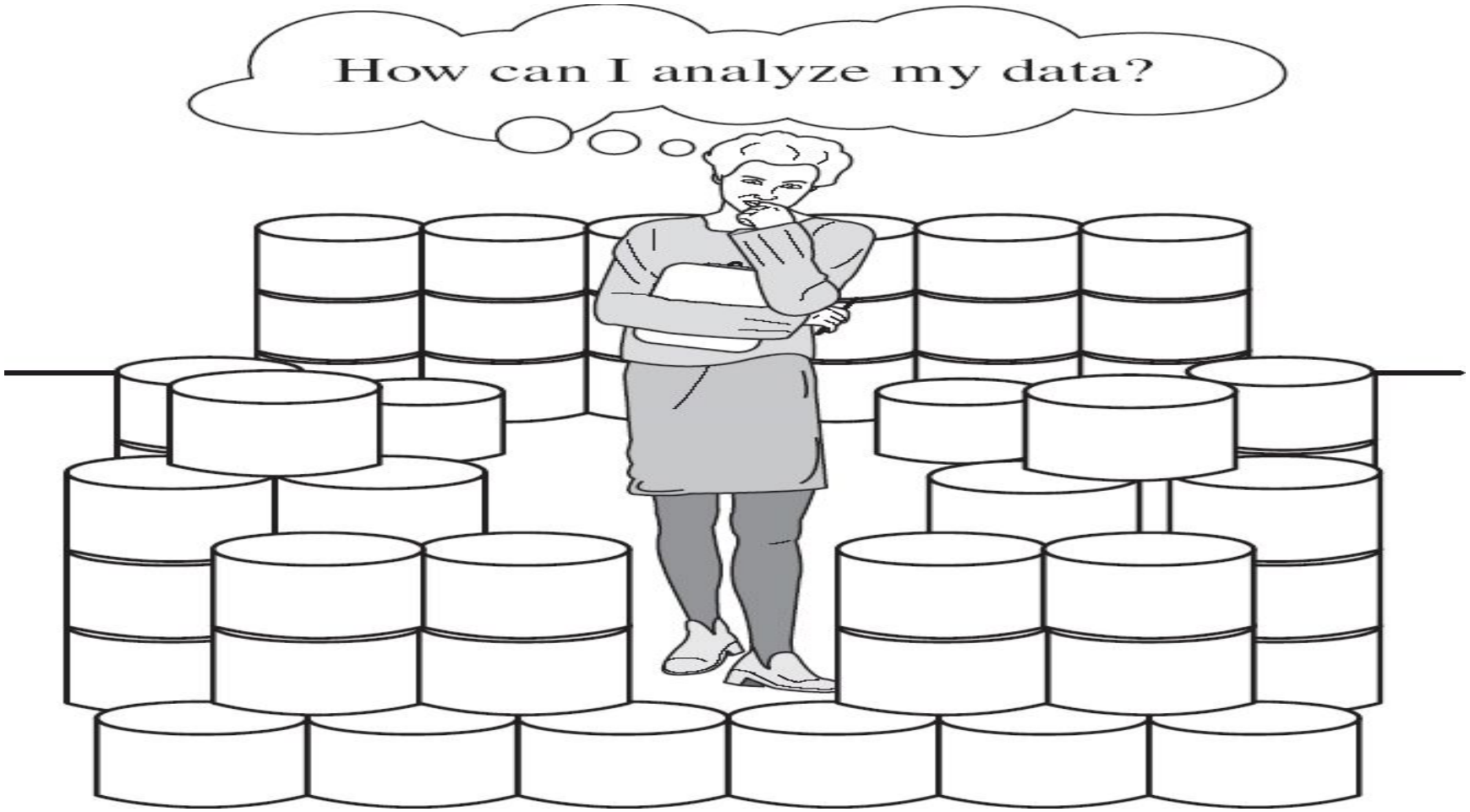
2. Drill down (roll down): reverse of roll-up
   - from higher level summary to lower level summary or detailed data

3. Slice : selection on one dimension of the given cube, resulting in a sub cube

4. Dice: define a sub cube by performing a selection on two or more dimensions

5. Pivot (rotate): rotates the data axes in order to provide an alternative presentation of the data.

Data Rich, Information Poor



WHAT IS DATA MINING?

Data mining :knowledge discovery from data(KDD)

- Extraction of interesting patterns or knowledge from huge amount of data
- It looks for hidden patterns within the data set and try to predict future behavior.
- This allows the business to take the data-driven decision

WHAT IS DATA MINING?

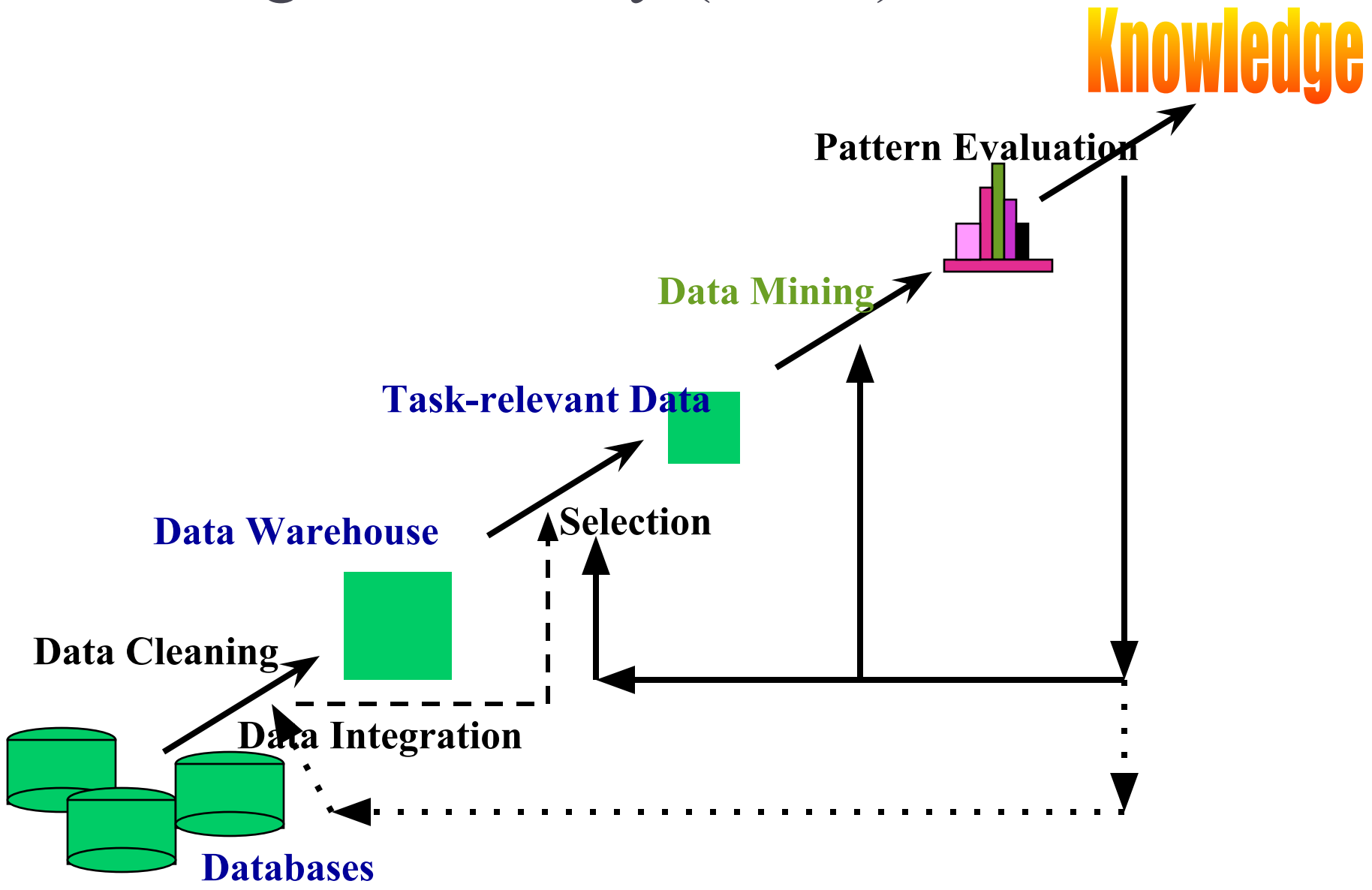
Data mining (knowledge discovery from data)

- Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data

Other names

- Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.

Knowledge Discovery (KDD) Process



KNOWLEDGE DISCOVERY FROM DATA

KDD process includes

1. Data cleaning (to remove noise and inconsistent data)

2. data integration (where multiple data sources may be combined)

3. data selection (where data relevant to the analysis task are retrieved from the database)

4. data transformation (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations)

KDD CONTINUED….

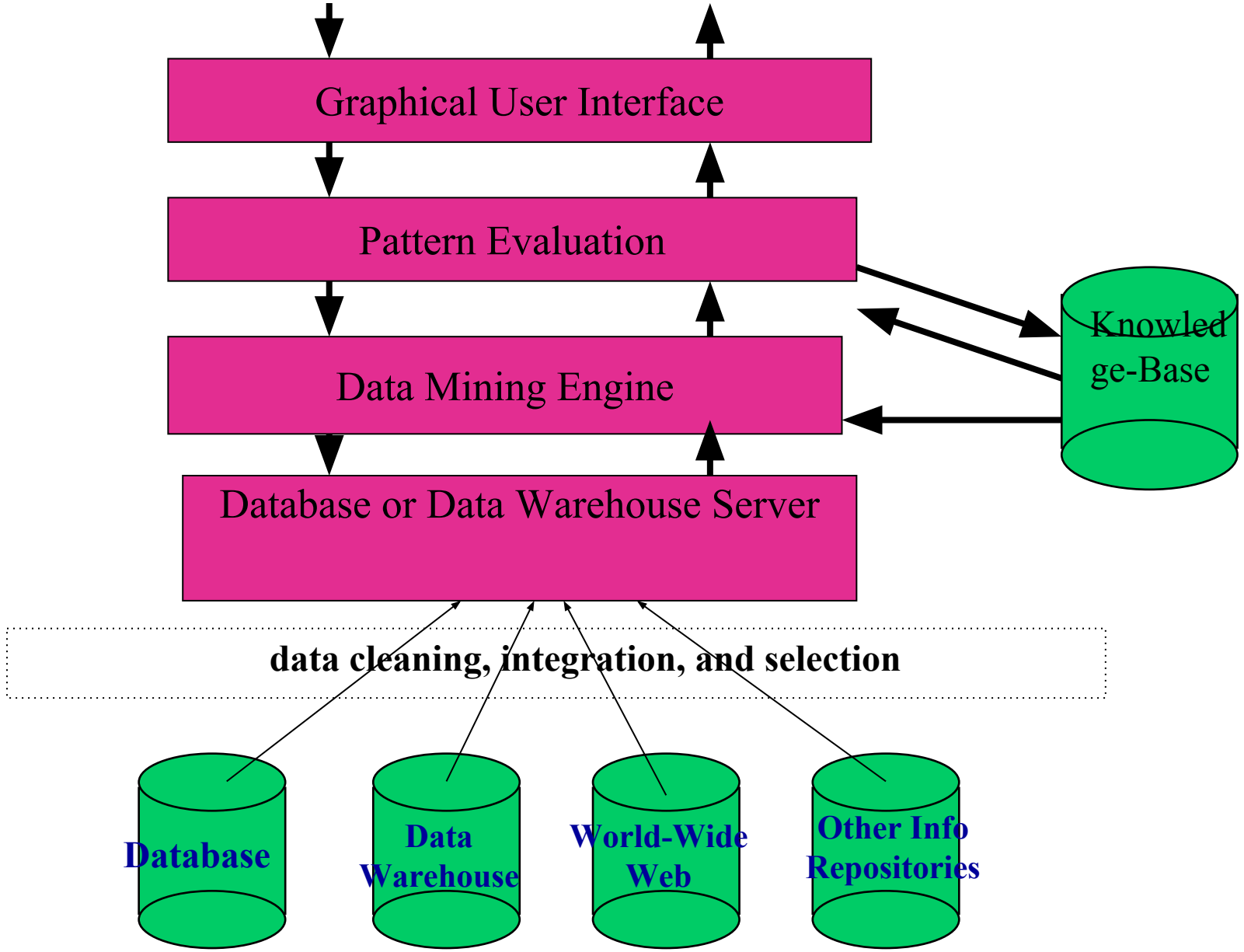
5. data mining (an essential process where intelligent methods are applied in order to extract data patterns.

6. pattern evaluation (to identify the truly interesting patterns representing knowledge based on some interestingness measures)

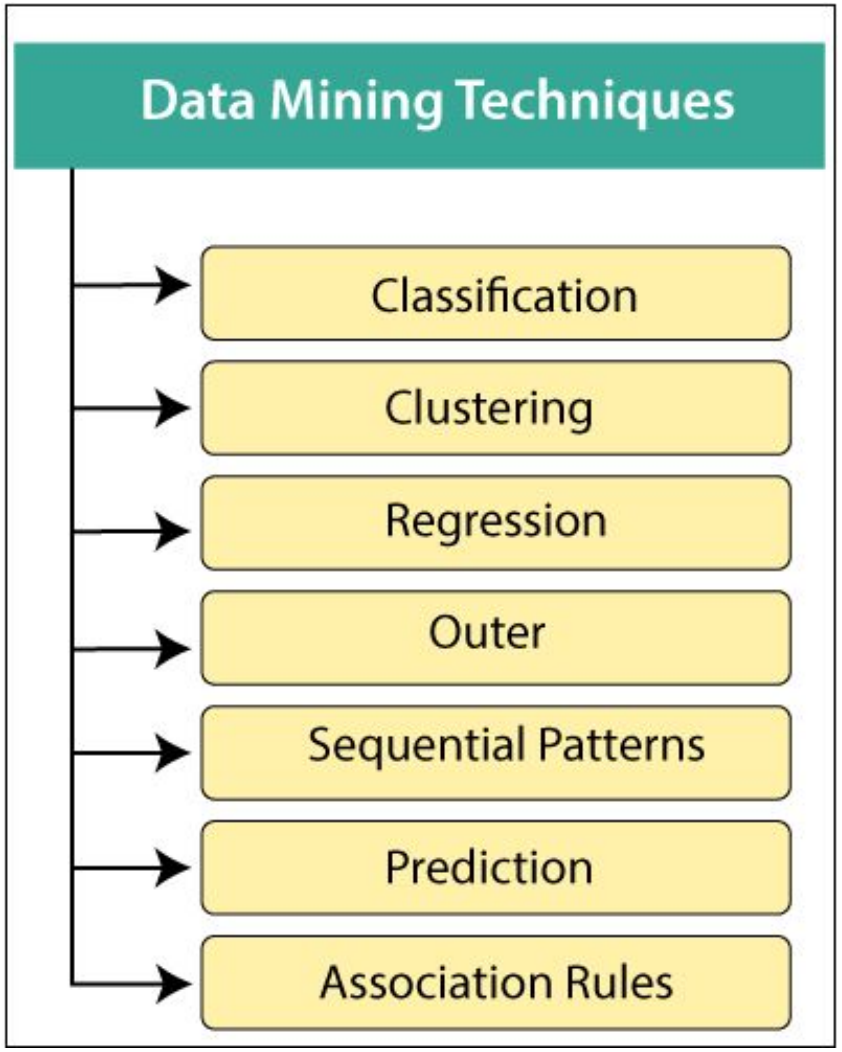
7. knowledge presentation (where visualization and knowledge representation techniques are used to present the mined knowledge to the user)

Data mining is a core of knowledge discovery process

ARCHITECTURE: TYPICAL DATA MINING SYSTEM



WHAT KIND OF PATTERN CAN BE MINED? (DATA MINING FUNCTIONALITIES)



Data mining techniques can be classified into two categories: descriptive and predictive

Descriptive: characterize the properties of the data in a target data set (class labels)

Predictive perform induction on the current data in order to make prediction

DATA MINING FUNCTIONALITIES

Class/Concept Description:

Data characterizing- summarization of the general features of a target class

Data discrimination- comparison of the target class with one or set of comparative classes

WHICH TECHNOLOGIES ARE USED?

Statistics

- Studies the collection , analysis, interpretation or explanation and presentation of data
- Statistical model are widely used to model data

Machine Learning

- Investigate how computers can learn based on data
- Computer is program to automatically learn to recognize complex pattern and make intelligent decision base on data
  - Supervised learning
  - Unsupervised learning
  - Semi-supervised learning

WHICH TECHNOLOGIES ARE USED?

Database systems and Data warehouse

- Focuses on the creation, maintenances and use of databases for organizations and end users
- Data warehouse consolidate data in multidimensional space

Information retrieval (IR)

- Science of searching a document or information which can be text or multimedia in a document which may reside on a web.

WHICH KIND OF APPLICATIONS ARE TARGETED?

Business Intelligence

- Provide historical, current, and predictive views of business operations

Web search engines

- It is a specialized computer server that searches for information on the web

MAJOR ISSUES IN DATA MINING

It is partition into five groups

1. Mining methodology
2. User interaction
3. Efficiency and scalability
4. Diversity of data types
5. Data mining and society

MAJOR ISSUES IN DATA MINING

1. Mining methodology issues

- Mining different kinds of knowledge in database
- Mining knowledge in multidimensional space
- Handling noisy or incomplete data
- Pattern evaluation

MAJOR ISSUES IN DATA MINING

2. User interaction issues

- Interactive mining of knowledge at multiple levels of abstraction
- Incorporation of background knowledge
- Data mining query languages and ad hoc data mining
- Presentation and visualization of data mining result

MAJOR ISSUES IN DATA MINING

3. Efficiency and scalability issues

- Efficiency and scalability of data mining algorithm
- Parallel, distributed and incremental mining algorithms

4. Diversity of database type issues

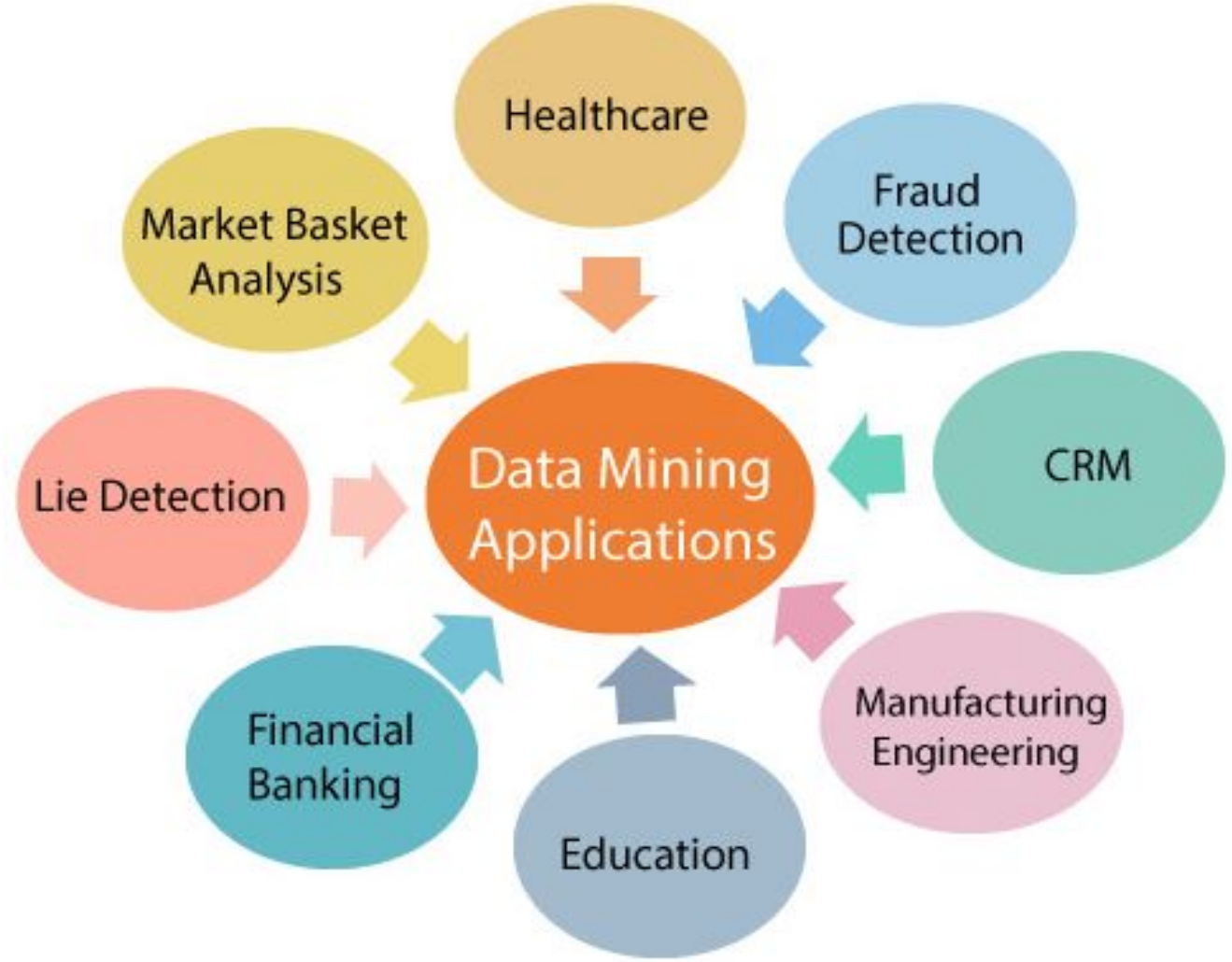
- Handling of relational and complex types of data
- Mining information from heterogeneous databases and global information system

MAJOR ISSUES IN DATA MINING

5. Data mining and Society

- Social impact of data mining
- Privacy preserving data mining
- Invisible data mining

DATA MINING APPLICATIONS



DATA MINING TOOLS


tcet
THAKUR COLLEGE OF ENGINEERING & TECHNOLOGY
Estd. in 2001


BRIGHT
tcet
IT
ENGINEERS

Thank You!!!

Data Exploration & Data Preprocessing

2 Chapter

Data Objects

- Data sets are made up of data objects.
- A **data object** represents an entity.
- Examples:
  - sales database: customers, store items, sales
  - medical database: patients, treatments
  - university database: students, professors, courses
- Also called *samples , examples, instances, data points, objects, tuples*.
- Data objects are described by **attributes**.
- Database rows -> data objects; columns ->attributes.

Attributes

- **Attribute (**or **dimensions, features, variables**):

  a data field, representing a characteristic or feature of a data object.
  - *E.g., customer _ID, name, address*
- Types:
  - Nominal
  - Binary
  - Ordinal
  - Numeric
    - Interval-scaled
    - Ratio-scaled

Attribute Types

1) **Nominal Attribute :**

- Related to names

- categories, states, or “names of things”
  - *Hair_color =* {*black, blond, brown, grey, red, white*}
  - marital status, occupation, ID numbers

- Also called as categorical Attribute

2) **Binary Attribute**

- Nominal attribute with only 2 states (0 and 1)
- 0 means attributes absent and 1 means attribute present
- Also called Boolean if two state corresponds to true and false
- Symmetric binary: both outcomes equally important and carry same weight.
  - e.g., gender
- Asymmetric binary: outcomes not equally important
  - e.g., medical test (positive vs. negative)
  - Convention: assign 1 to most important outcome (e.g., HIV positive)

- **Ordinal Attribute**

  - Values have a meaningful order (ranking) but magnitude between successive values is not known.
    - *Size =* {*small, medium, large*}*,*
    - grades=(A+, A, B, B+)

Numeric Attribute Types

Numeric Attribute:

- Quantitative (integer or real-valued)
- **Interval-scaled attributes**
  - Measured on a scale of **equal-sized units**
  - Values have order
    - E.g., *temperature in C˚or F˚, calendar dates*
  - No true zero-point
- **Ratio-scaled attributes**
  - Inherent **zero-point**
  - We can speak of values as being a multiple of another value

    (10 Kg is twice of 5 Kg).
    - e.g. *length, counts, monetary quantities*

Discrete vs. Continuous Attributes

- **Discrete Attribute**
  - Has only a finite or countably infinite set of values
    - E.g., zip codes, profession, or the set of words in a collection of documents
  - Sometimes, represented as integer variables
  - Note: Binary attributes are a special case of discrete attributes
- **Continuous Attribute**
  - Has real numbers as attribute values
    - E.g., temperature, height, or weight
  - Continuous attributes are typically represented as floating-point variables

Basic Statistical Descriptions of Data

- For successful data preprocessing it is essential to have overall picture of your data
- Statistical Descriptions can be used to identify propertied of the data
- Three area of basic SD of data-------

1) **Measuring the central tendency of data:**
   - mean, median, mode (where most of the value fall?)
2) **Measuring the dispersion of data:**

   Range, quartiles, variance, standard deviation, interquartile range, five number summery, box plot
3) **Graphic display of basic sd of data:**

   Quantile plot, quantile quantile plot, histogram, scatter plots

- Mean:
  - Effective measure of the center of a set of data
  - Let x1, x2, ….xn be set of n values for numeric attribute X like salary. The mean is given by---

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Mean = (x1+x2+….+xn)/n

  - Ex. 30,36,47,50,52,52,56,60,63,70,70,110

- Weighted arithmetic mean

$$\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

- Trimmed mean: chopping extreme values

Median:

1. For **odd number** of values (count), Middle value is considered.

2. For **even number** of values (count), average of two middle most values is considered.

3. For **interval**---

$$median = L_1 + (\frac{n/2 - (\sum freq)_l}{freq_{median}})width$$

age	*frequency*	
1–5	200	
6–15	450	
16–20	300	
21–50	1500	**Median interval**
51–80	700	
81–110	44	

Here

L1→ lower boundary of the median interval

N→ number of values in entire dataset

$(\sum freq)_1$→ sum of frequencies of all of the intervals that are lower than the median interval

Freq $_{median}$→ frequency of the median interval

Width→ width of the median interval

- L1→ 20
- N→ 3194
- $(\sum freq)_1$→ 950
- Freq $_{median}$→ 1500
- Width→ 30

Ans-→

Median=32.94 years

- Mode
  - A mode is defined as the value that has a higher frequency in a given set of values.
  - It is the value that appears the most number of times.
  - **Example**: In the given set of data: 2, 4, 5, 5, 6, 7, the mode of the data set is 5 since it has appeared in the set twice.

Bimodal, Trimodal & Multimodal (More than one mode)

- When there are two modes in a data set, then the set is called **bimodal**
  - For example, The mode of Set A = {2,2,2,3,4,4,5,5,5} is 2 and 5, because both 2 and 5 is repeated three times in the given set.
- When there are three modes in a data set, then the set is called **trimodal**
  - For example, the mode of set A = {2,2,2,3,4,4,5,5,5,7,8,8,8} is 2, 5 and 8
- When there are four or more modes in a data set, then the set is called **multimodal**

– Empirical formula for unimodal numeric data---

$$mean - mode = 3 \times (mean - median)$$

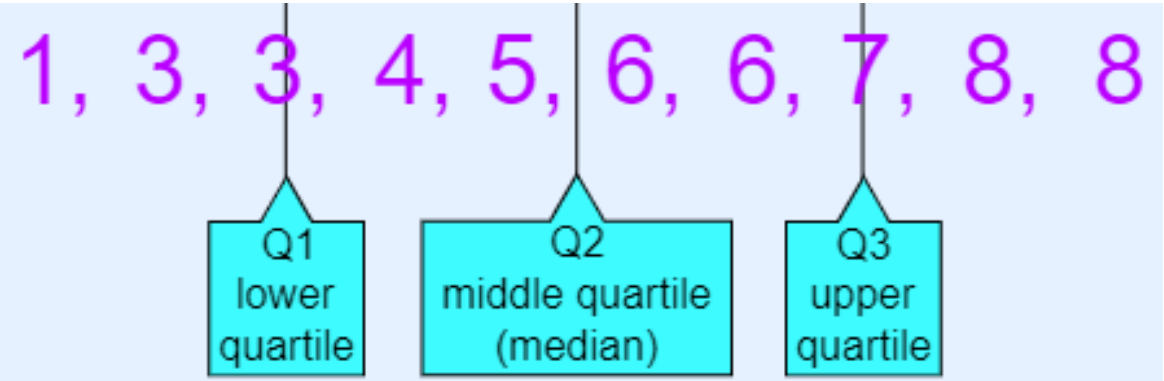
- Midrange
  - Average of largest and smallest value in dataset
  - Midrange = (Maximum Value + Minimum Value) / 2
  - **Example:** Consider the data set 110, 150, 180, 220, 270, 290, 310 and 390 as the prices of speakers. The minimum number is 110, and the maximum is 390.
  - Midrange=(390 + 110) / 2 = 250

Measuring the Dispersion of Data

- **Quartiles, outliers and boxplots:**
  - **Quartiles**: Q_1 (25^{th} percentile), Q_3 (75^{th} percentile)
  - **Inter-quartile range**: IQR = $Q_3 - Q_1$
  - **Five number summary**: {min, Q_1, median, Q_3, max}
  - **Boxplot**: ends of the box are the quartiles; median is marked; add whiskers, and plot outliers individually
  - **Outlier**: usually, a value higher/lower than 1.5 x IQR

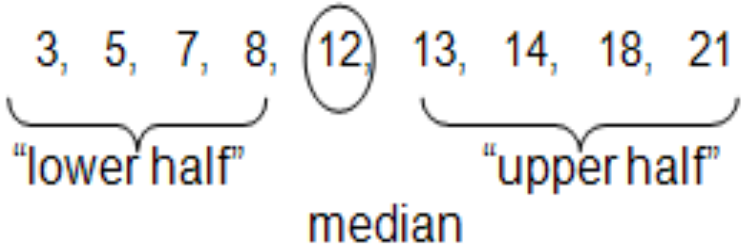
Quartiles

- Quartiles are the values (Q1, Q2,Q3,Q4) that divide a list of numbers into quarters or four parts.



Quartiles

- **Example 1:** Find the first and third quartiles of the data set {3, 7, 8, 5, 12, 14, 21, 13, 18}.
- **Total numbers in set=9**
- First, write data in increasing order: 3, 5, 7, 8, 12, 13, 14, 18, 21.
- The median is 12.
- The first quartile, Q_1, is the median of {3, 5, 7, 8}=6
- The third quartile, Q_3, is the median of {13,14,18,21}=16



$$Q_1 = \frac{5+7}{2} = \frac{12}{2} = 6$$

$$Q_3 = \frac{14+18}{2} = \frac{32}{2} = 16$$

Quartiles

- **Example 2:** Find the first and third quartiles of the set {3, 7, 8, 5, 12, 14, 21, 15, 18, 14}.

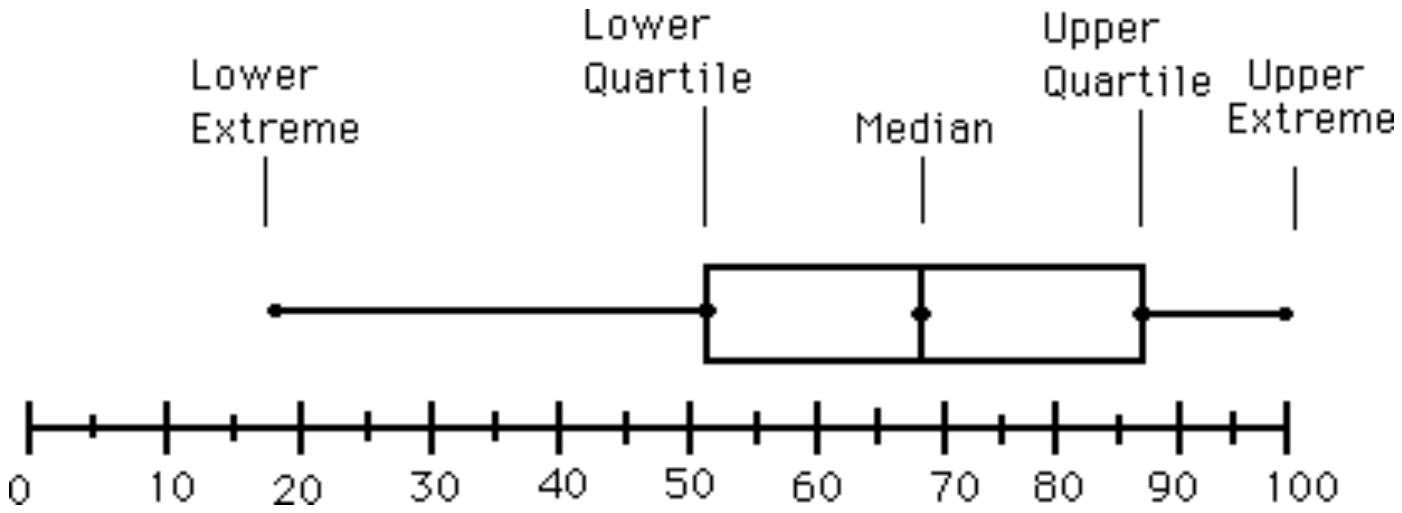
- Median (Q2) is 13 (it is the mean of 12 and 14)
- $\boldsymbol{Q_1 = 8}$
- $\boldsymbol{Q_3 = 17}$.

- Arrange the data in ascending order: {3, 5, 7, 8, 12, 14, 14, 15, 18, 21}
- Calculate the position of the first quartile (Q1): Q1 = (1/4) * (N + 1) where N is the number of data points. For this data set, N = 10 Q1 = (1/4) * (10 + 1) = (1/4) * 11 = 11/4 = 2.75
- The position of Q1 falls between the second and third data points.
- To find the first quartile (Q1), take the average of the values at the 2nd and 3rd positions: Q1 = (7 + 8) / 2 = 15 / 2 = 7.5
- So, the first quartile (Q1) is 7.5.
- Calculate the position of the third quartile (Q3): Q3 = (3/4) * (N + 1) Q3 = (3/4) * 11 = 33/4 = 8.25
- The position of Q3 falls between the 8th and 9th data points.
- To find the third quartile (Q3), take the average of the values at the 8th and 9th positions: Q3 = (15 + 18) / 2 = 33 / 2 = 16.5

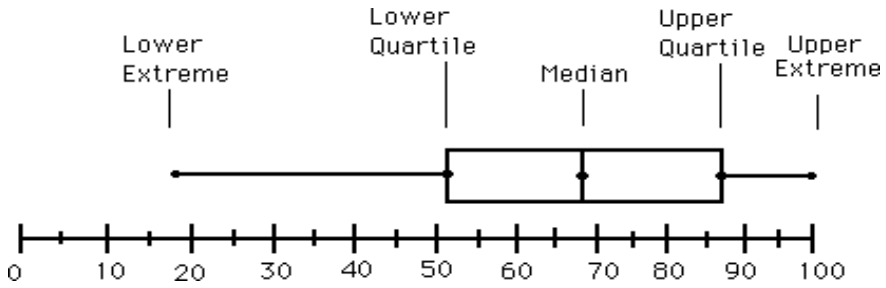
Boxplot

- Draw a number line and mark the positions of Q1, Q2 (median), and Q3.
- Draw a box from Q1 to Q3. This box represents the interquartile range (IQR).
- Draw a line (whisker) from the box to the minimum value (Min) and another line to the maximum value (Max), but within the lower and upper fences.

Boxplot


Lower Extreme
Lower Quartile
Median
Upper Quartile
Upper Extreme
0
10
20
30
40
50
60
70
80
90
100

Boxplot Analysis



- **Five-number summary** of a distribution
  - Minimum, Q1, Median, Q3, Maximum
- **Boxplot**
  - Data is represented with a box
  - The ends of the box are at the first and third quartiles, i.e., the height of the box is IQR
  - The median is marked by a line within the box
  - Whiskers: two lines outside the box extended to Minimum and Maximum
  - Outliers: points beyond a specified outlier threshold, plotted individually

Boxplot Example 1

Example: A sample of 10 boxes of raisins has these weights (in grams):

25, 28, 29, 29, 30, 34, 35, 35, 37, 38

Make a box plot of the data

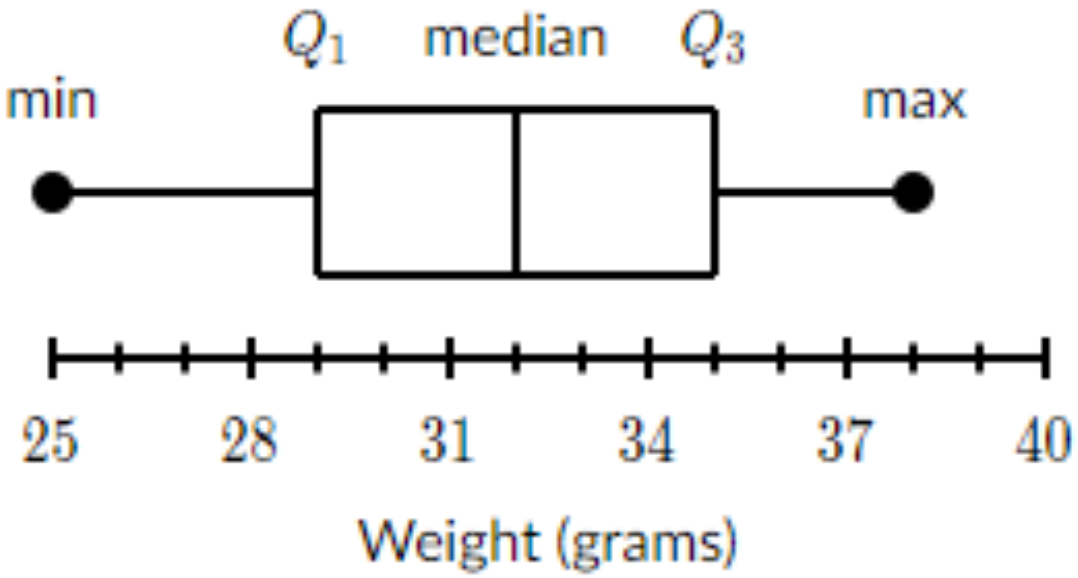
Solution:

Step 1: Order the data from smallest to largest.

Step 2: find the 5 number summary

(minimum, first quartile, median, third quartile, and maximum)

- (min, Q1,Q2,Q3,max)
- (25,29,32,35,38)

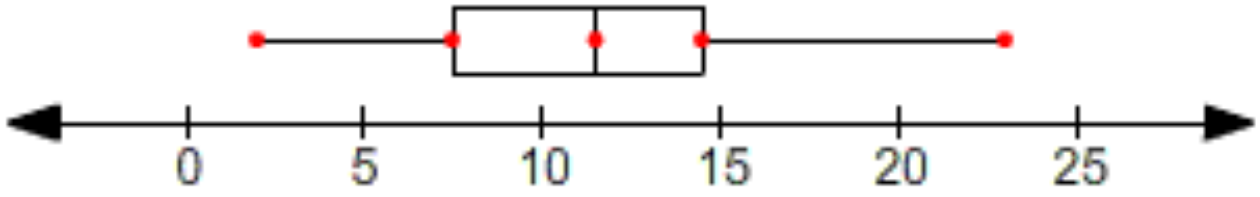


Five number summary: (25,29,32,35,38)

Ex 2

- Find Q1 , Q2 , and Q3 for the following data set, and draw a box-and-whisker plot.
- {2,6,7,8,8,11,12,13,14,15,22,23}

- Five number summary
- (2, 7.5, 11.5,14.5,23)



Boxplot Example 3

Ex 2. 30,36,47,50,52,52,56,60,63,70,70,110

- Draw the box plot

Ex 4

Find Q_1 , Q_2 , and Q_3 for the following data set. Identify any outliers, and draw a box-and-whisker plot.

$$\{5, 40, 42, 46, 48, 49, 50, 50, 52, 53, 55, 56, 58, 75, 102\}$$

There are 15 values, arranged in increasing order. So, Q_2 is the 8^{th} data point, 50 .

Q_1 is the 4^{th} data point, 46 , and Q_3 is the 12^{th} data point, 56 .

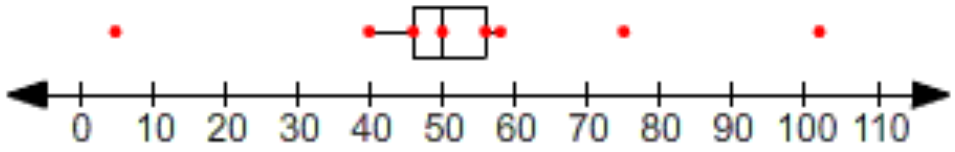
The interquartile range IQR is $Q_3 - Q_1$ or $56 - 47 = 10$.

Now we need to find whether there are values less than $Q_1 - (1.5 \times \text{IQR})$ or greater than $Q_3 + (1.5 \times \text{IQR})$

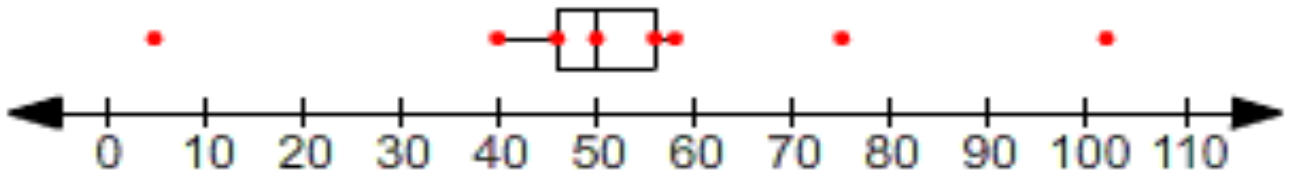
$$Q_1 - (1.5 \times \text{IQR}) = 46 - 15 = 31$$

$$Q_3 + (1.5 \times \text{IQR}) = 56 + 15 = 71$$

Since 5 is less than 31 and 75 and 102 are greater than 71 , there are 3 outliers.



Note that 40 and 58 are shown as the ends of the whiskers with outliers plotted separately as a dots.



Outliers

- If a data value is very far away from the quartiles (either much less than Q1 or much greater than Q3), it is sometimes designated an outlier .
- The standard definition for an outlier is a number which is less than Q1 or greater than Q3 by more than 1.5 times the interquartile range
- IQR=Q3−Q1
- That is, an outlier is any number

less than Q1−(1.5×IQR) **or**

greater than Q3+(1.5×IQR)

Variance & Standard Deviation

- Variance: **Variance** is the sum of squares of differences between all numbers and means.

$$Formula : \sigma^2 = \frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N}$$

- Where μ is Mean, N is the total number of elements or frequency of distribution.
- **Standard Deviation** is square root of variance. It is a measure of the extent to which data varies from the mean.
- The Standard Deviation is a measure of how spread out numbers are.

Example 1 – Standard deviation

A hen lays eight eggs. Each egg was weighed and recorded as follows

60 g, 56 g, 61 g, 68 g, 51 g, 53 g, 69 g, 54 g.

a. First, calculate the mean:

$$\overline{x} = \frac{\sum x}{n}$$

$$= \frac{472}{8}$$

$$= 59$$

b. Now, find the standard deviation.

Table 1. Weight of eggs, in grams

Weight (x)	$(x - \bar{x})$	$(x - \bar{x})^2$
60	1	1
56	-3	9
61	2	4
68	9	81
51	-8	64
53	-6	36
69	10	100
54	-5	25
472		**320**

Using the information from the above table, we can see that

$$\sum (x - \bar{x})^2 = 320$$

In order to calculate the standard deviation, we must use the following formula:

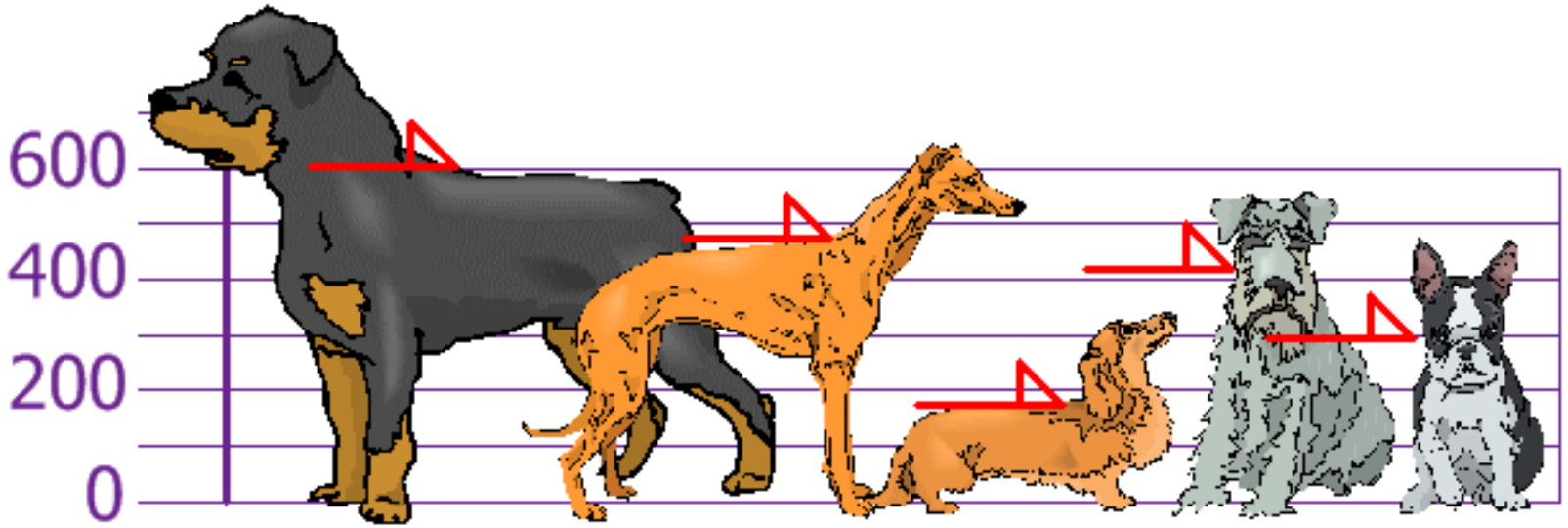
$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n}}$$

$$= \sqrt{\frac{320}{8}}$$

$$= 6.32 \text{ grams}$$

Example

You and your friends have just measured the heights of your dogs (in millimeters):



The heights (at the shoulders) are: 600mm, 470mm, 170mm, 430mm and 300mm.

Find out the Mean, the Variance, and the Standard Deviation.

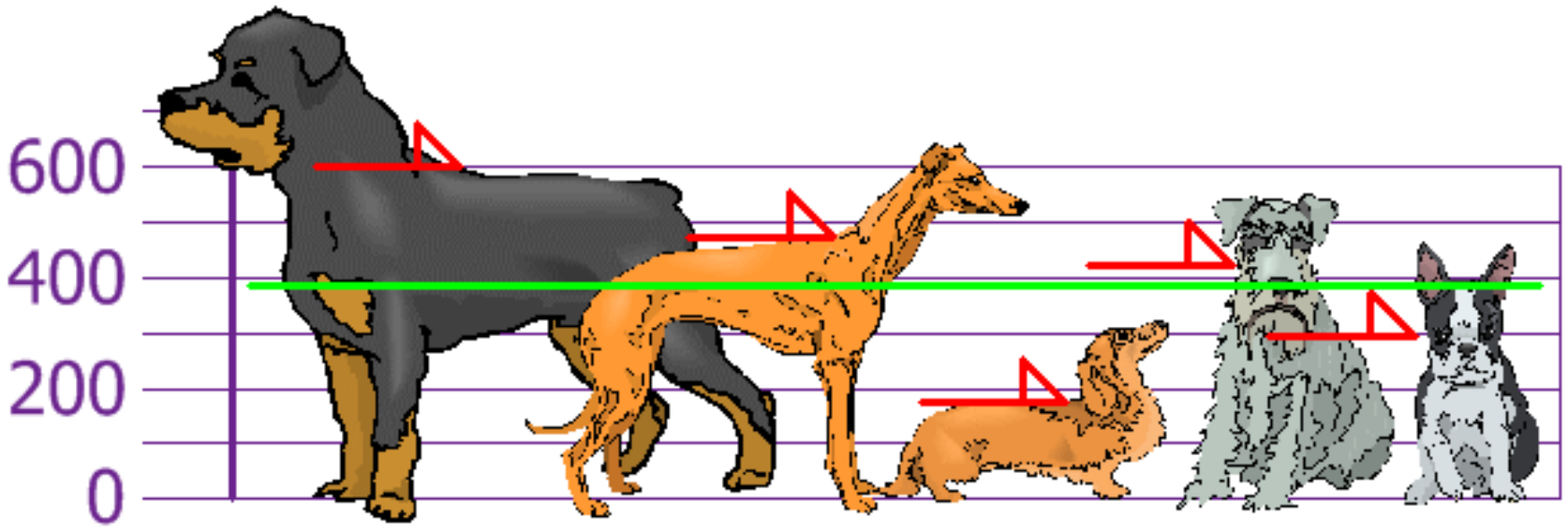
Answer:

$$\text{Mean} = \frac{600 + 470 + 170 + 430 + 300}{5}$$

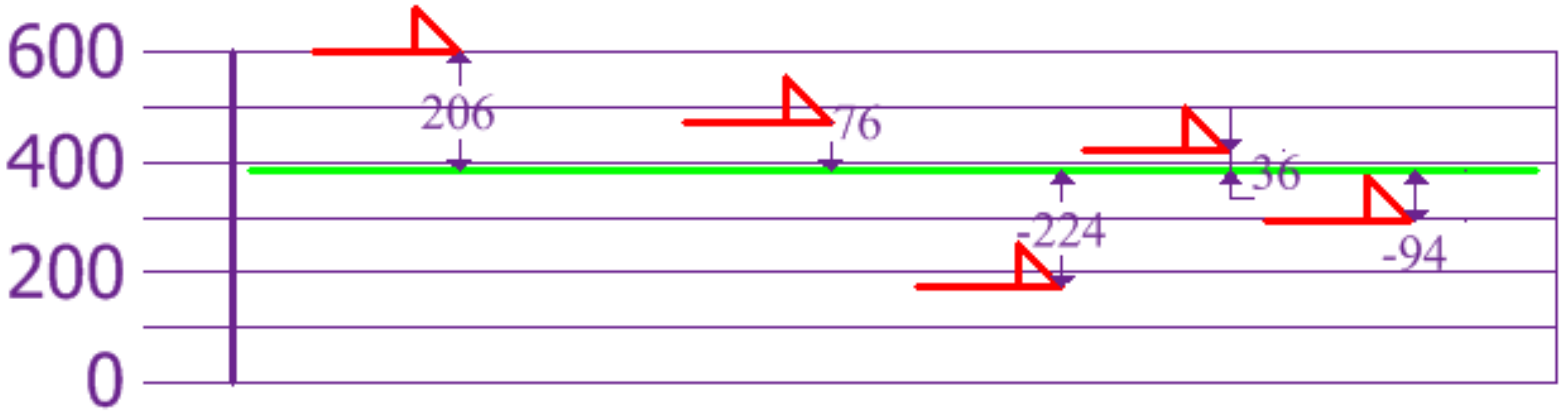
$$= \frac{1970}{5}$$

$$= 394$$

so the mean (average) height is 394 mm. Let's plot this on the chart:



Now we calculate each dog's difference from the Mean:



To calculate the Variance, take each difference, square it, and then average the result:

Variance

$$\sigma^2 = \frac{206^2 + 76^2 + (-224)^2 + 36^2 + (-94)^2}{5}$$

$$= \frac{42436 + 5776 + 50176 + 1296 + 8836}{5}$$

$$= \frac{108520}{5}$$

$$= 21704$$

So the Variance is **21,704**

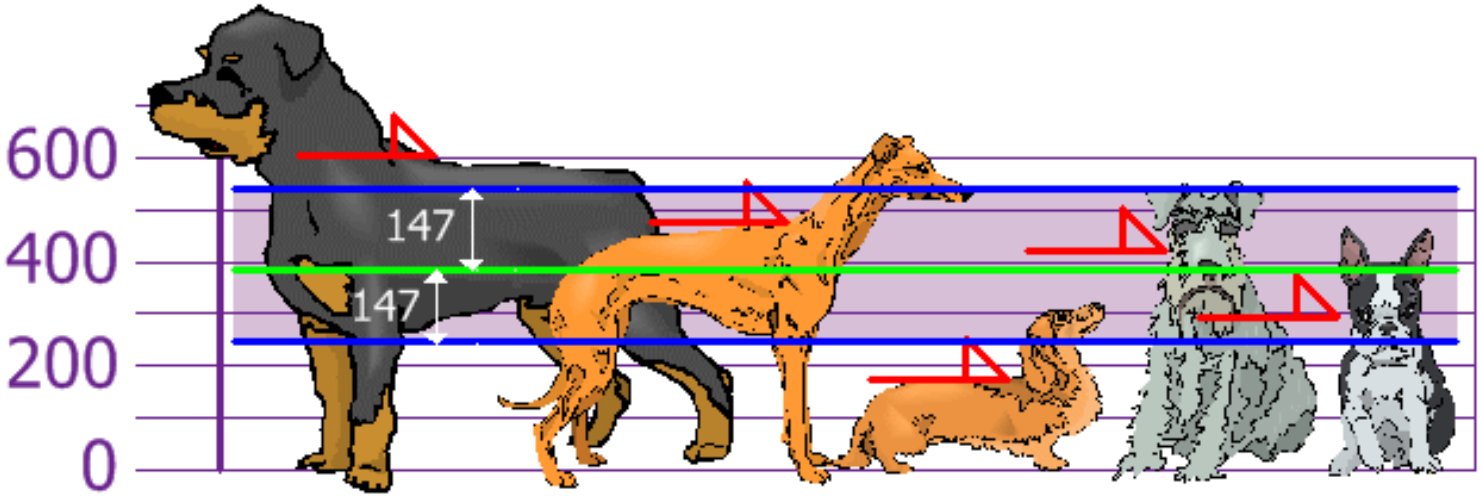
And the Standard Deviation is just the square root of Variance, so:

Standard Deviation

$$\sigma = \sqrt{21704}$$
$$= 147.32...$$
$$= \mathbf{147} \textit{ (to the nearest mm)}$$

And the good thing about the Standard Deviation is that it is useful. Now we can show which heights are within one Standard Deviation (147mm) of the Mean:

And the good thing about the Standard Deviation is that it is useful. Now we can show which heights are within one Standard Deviation (147mm) of the Mean:



So, using the Standard Deviation we have a "standard" way of knowing what is normal, and what is extra large or extra small.

Rottweilers **are** tall dogs. And Dachshunds **are** a bit short, right?

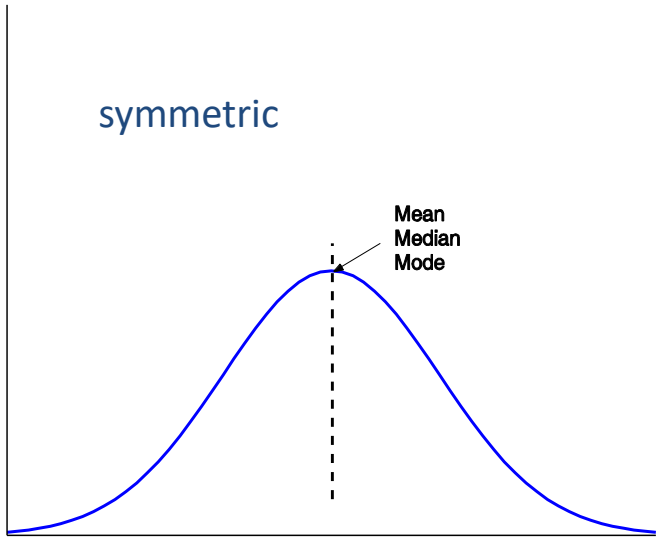
Key points about variance/standard deviation

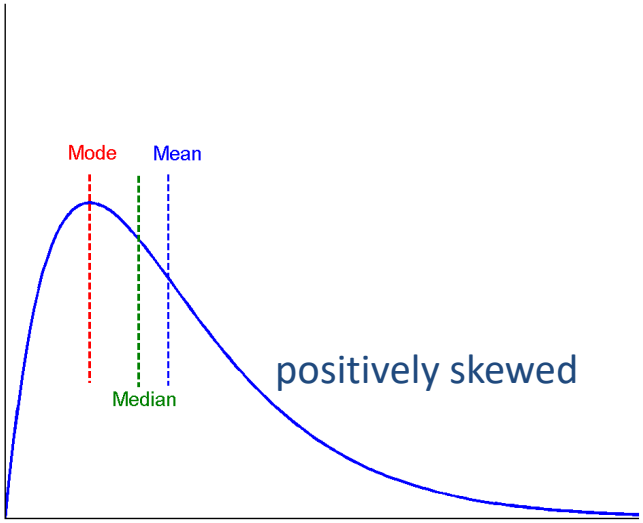
- Variance and standard deviation are measures of data dispersion
- They indicate how spread out a data distribution is.
- A **low standard** deviation means that the data observation tend to be very close to mean
- **High standard** deviation indicates data are spread out over a large range of values
- $\sigma = 0$ when there is no spread, that is , when all observation have the same value. Otherwise $\sigma > 0$

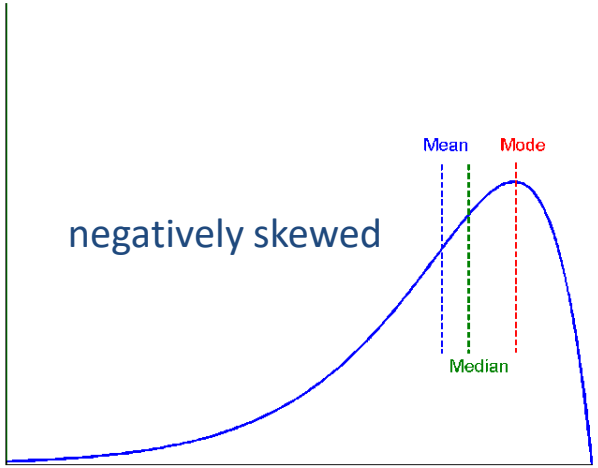
Symmetric vs. Skewed Data

- Median, mean and mode of symmetric, positively and negatively skewed data

"skewed to the left" (the long tail is on the left hand side): negatively skewed
"skewed to the right" (the long tail is on the right hand side): positively skewed





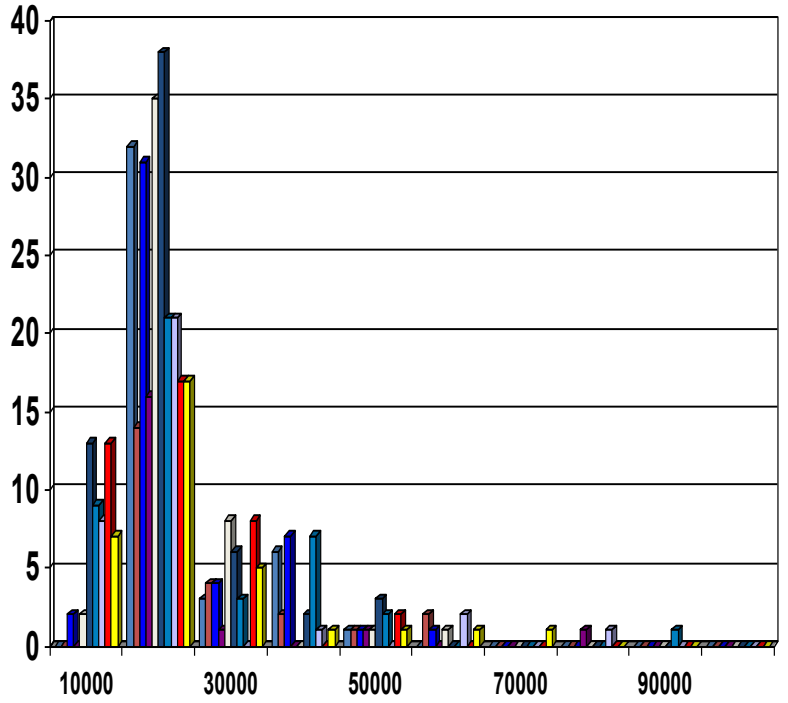


Graphic Displays of Basic Statistical Descriptions

- **Histogram**: x-axis are values, y-axis represents frequencies
- **Quantile plot**: each value x_i is paired with f_i indicating that approximately 100 f_i % of data are $\leq x_i$
- **Quantile-quantile (q-q) plot**: graphs the quantiles of one univariant distribution against the corresponding quantiles of another
- **Scatter plot**: each pair of values is a pair of coordinates and plotted as points in the plane

Histogram Analysis

- Histogram: Graph display of tabulated frequencies, shown as bars
- It shows what proportion of cases fall into each of several categories
- Differs from a bar chart in that it is the *area* of the bar that denotes the value, not the height as in bar charts, a crucial distinction when the categories are not of uniform width
- The categories are usually specified as non-overlapping intervals of some variable. The categories (bars) must be adjacent

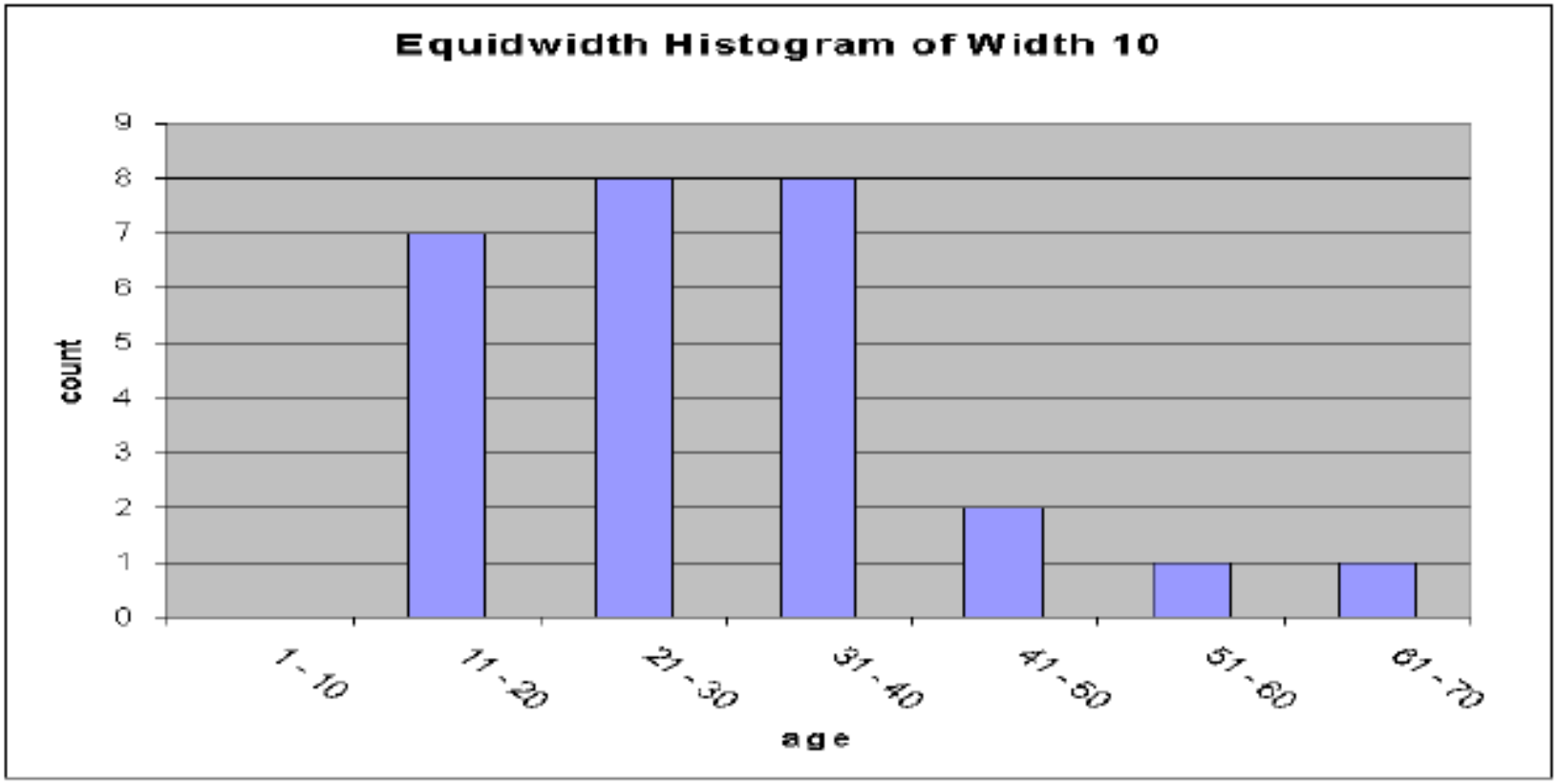


Example Histogram

- Dataset for Age:
  13,15,16,16,20,20,21,22,25,25,25,25,30,33,33,35,35,35,35,36,40,45,46,52,70

Age	Count/Frequency
13	1
15	1
16	2
20	2
21	1
22	1
25	4
30	1
33	2
35	4

Example Equidwidth Histogram



Example Histogram

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

Quantile plot

- Used to check whether your data is Normal
- To make a QQplot:

  For a sample of size n: $x_1, x_2, \ldots x_n$

  1. Order the data from smallest to largest:

     $x_{(1)}, x_{(2)}, \ldots x_{(n)}$ where $x_{(i)}$ is the i-th smallest

  2. Calculate the sample quantile

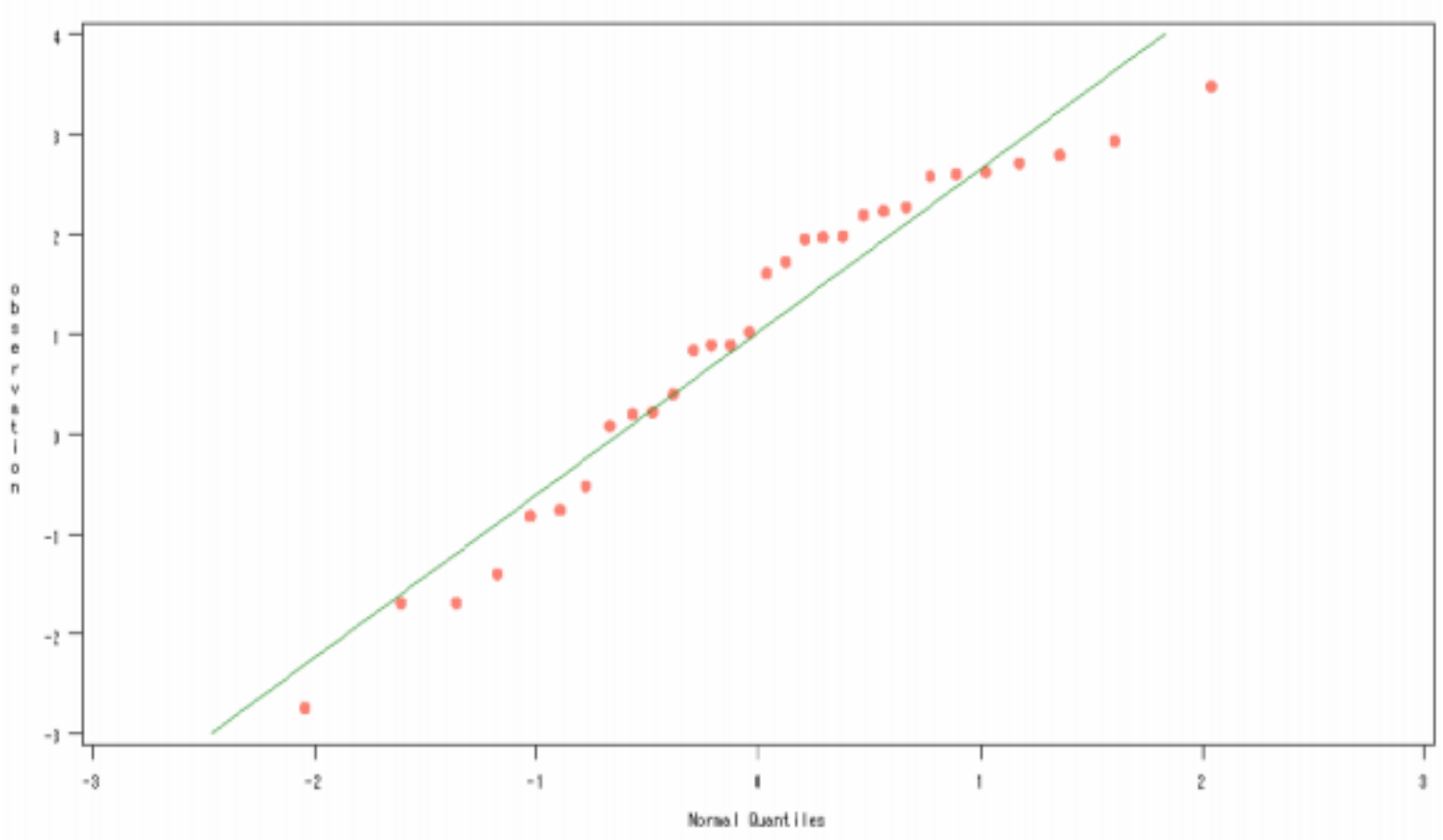
     Sample quantile is calculated as:

     $x_{(i)}$= $[(i-0.5)/n]$th sample quantile

  3. Plot the points ($[(i-0.5)/n]$th z-percentile, $x_{(i)}$)

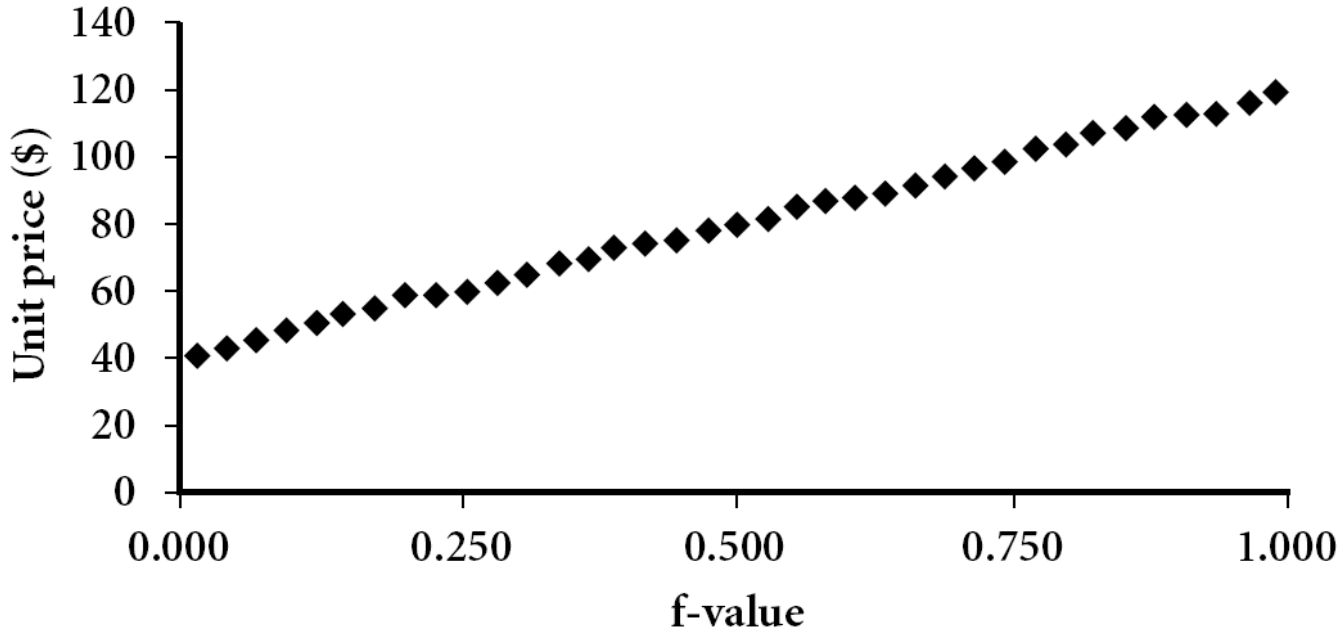
- If the data distribution is close to normal, the plotted points will lie close to a sloped straight line on the QQplot!

Quantile plot



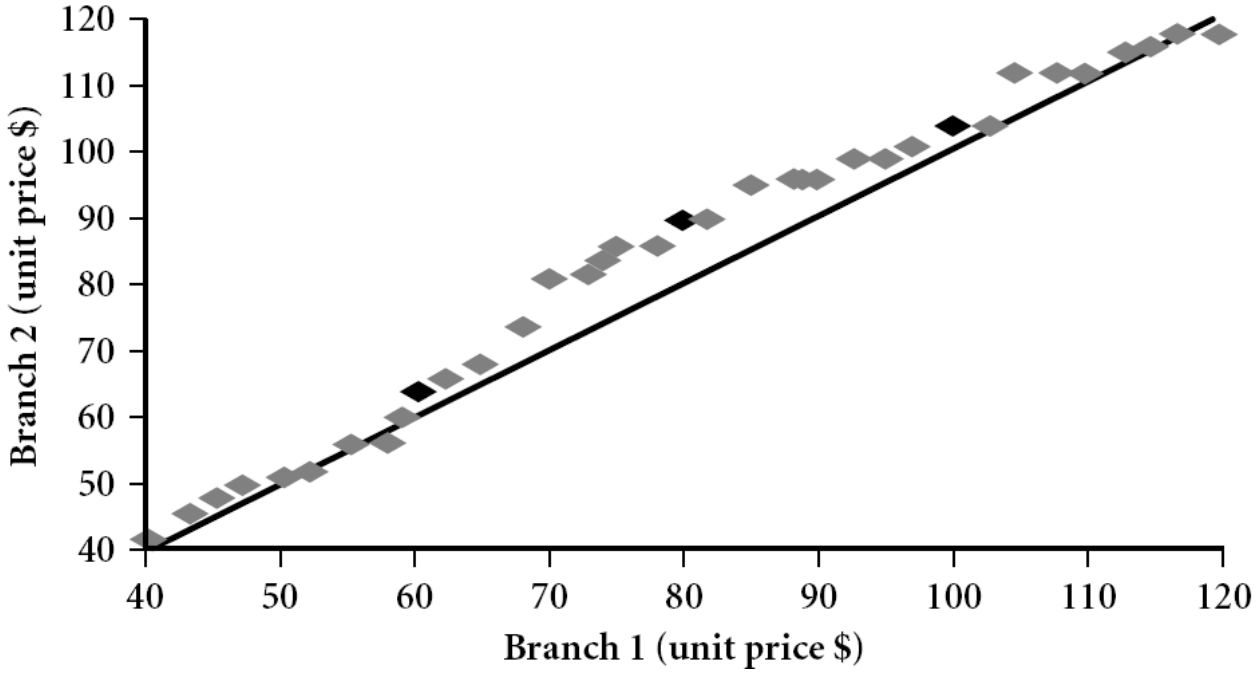
Quantile Plot

- Displays all of the data (allowing the user to assess both the overall behavior and unusual occurrences)
- Plots **quantile** information
  - For a data x_i data sorted in increasing order, f_i indicates that approximately 100 f_i% of the data are below or equal to the value x_i



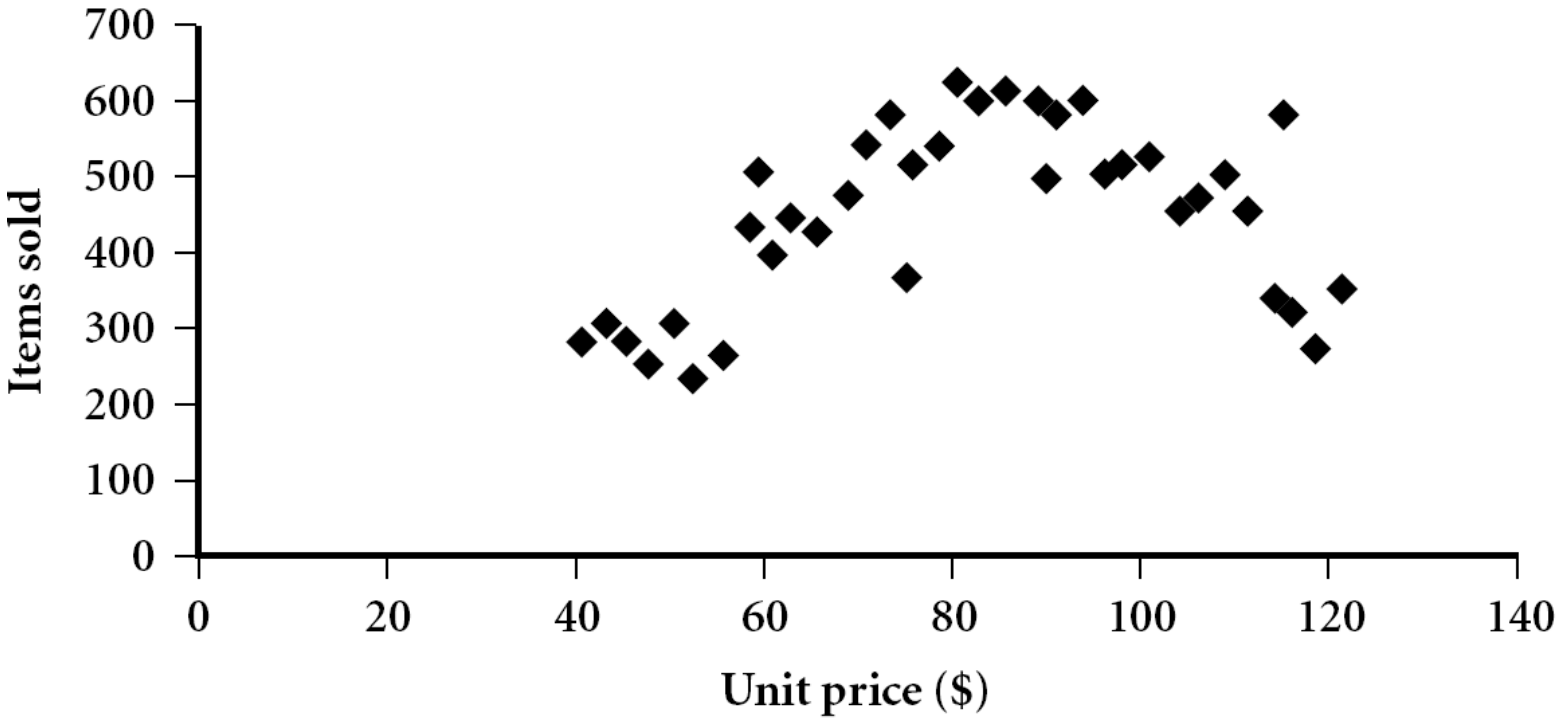
Quantile-Quantile (Q-Q) Plot

- Graphs the quantiles of one univariate distribution against the corresponding quantiles of another
- View: Is there is a shift in going from one distribution to another?
- Example shows unit price of items sold at Branch 1 vs. Branch 2 for each quantile. Unit prices of items sold at Branch 1 tend to be lower than those at Branch 2.

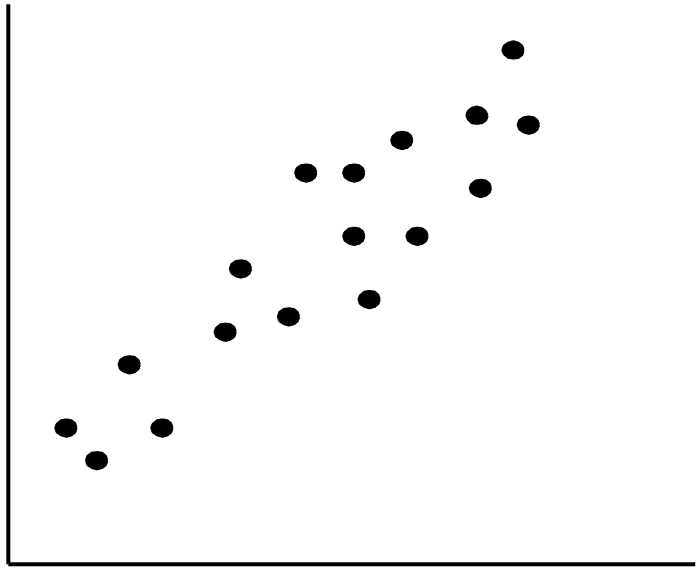


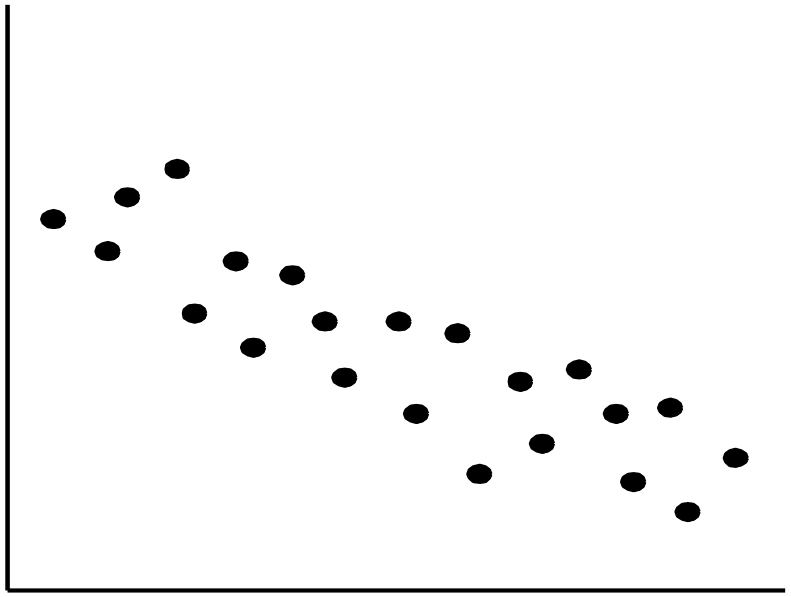
Scatter plot

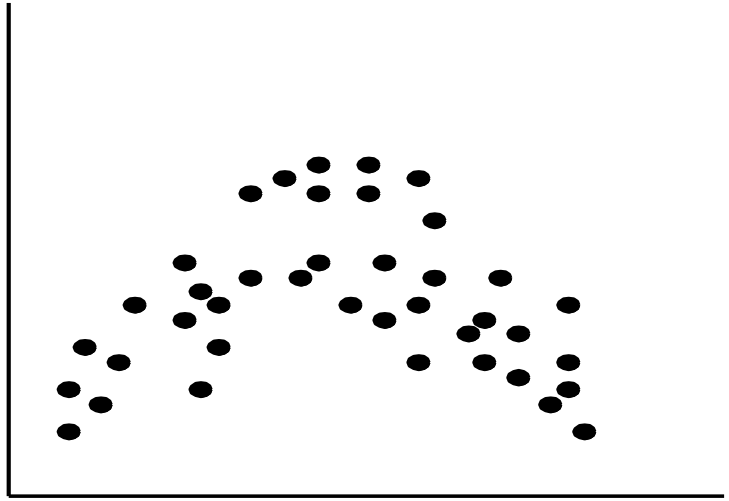
- Each pair of values is treated as a pair of coordinates and plotted as points in the plane



Positively and Negatively Correlated Data

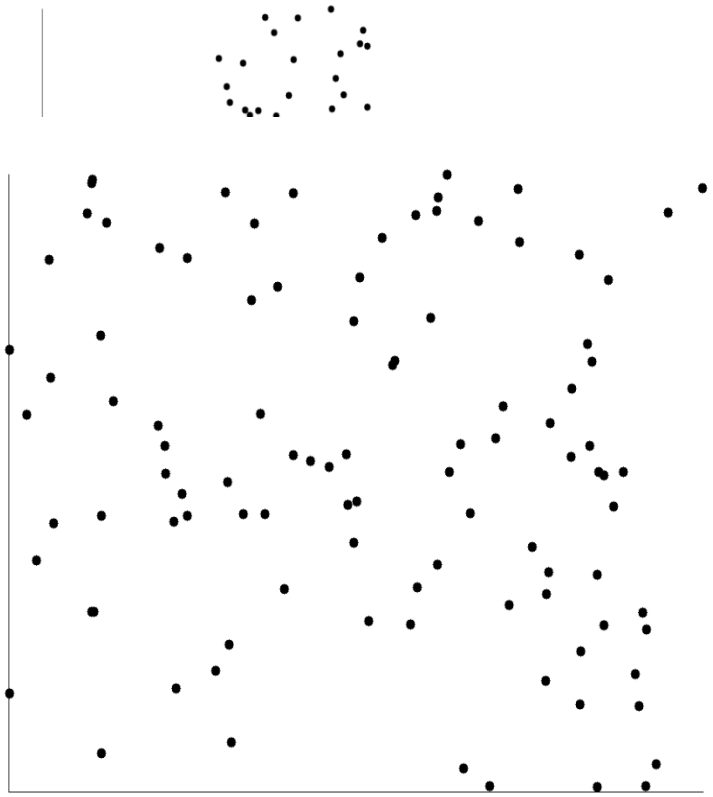


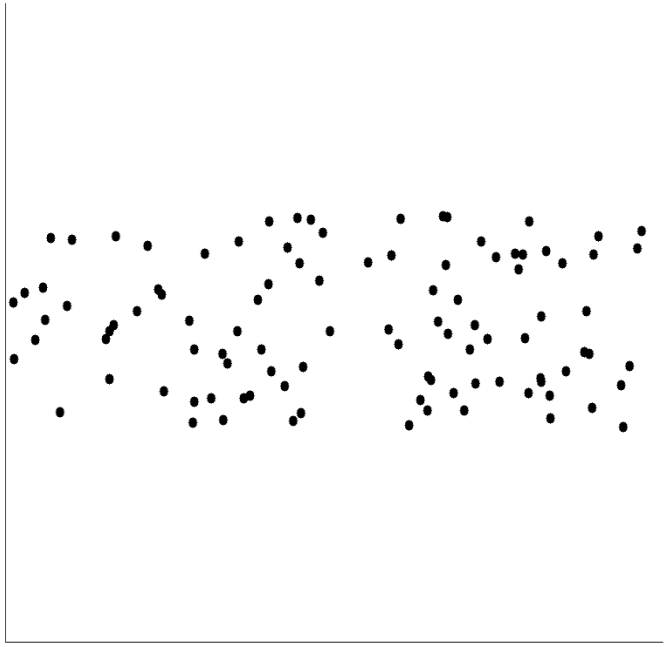




- The left half fragment is positively correlated
- The right half is negative correlated

Uncorrelat



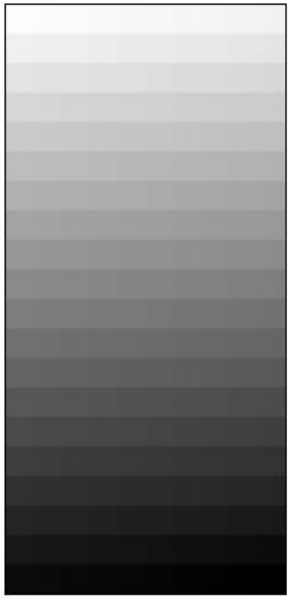


Data Visualization

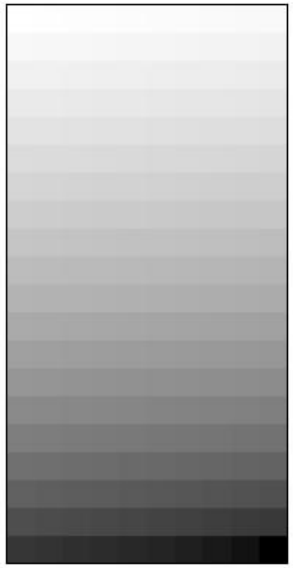
- Why data visualization?
  - Gain insight into an information space by mapping data onto graphical primitives
  - Provide qualitative overview of large data sets
  - Search for patterns, trends, structure, irregularities, relationships among data
  - Help find interesting regions and suitable parameters for further quantitative analysis
  - Provide a visual proof of computer representations derived
- Categorization of visualization methods:
  - Pixel-oriented visualization techniques
  - Geometric projection visualization techniques
  - Icon-based visualization techniques
  - Hierarchical visualization techniques
  - Visualizing complex data and relations

Pixel-Oriented Visualization Techniques

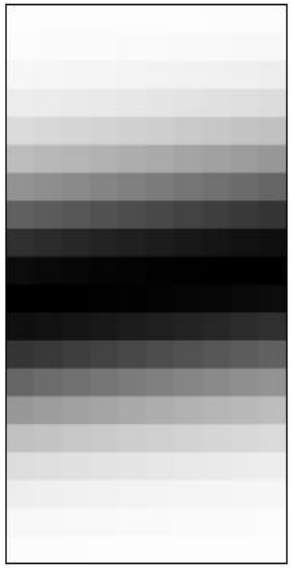
- For a data set of m dimensions, create m windows on the screen, one for each dimension
- The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows
- The colors of the pixels reflect the corresponding values



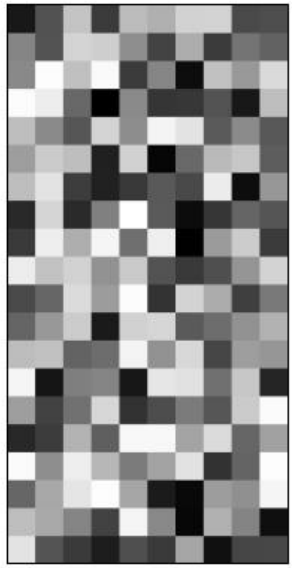
(a) Income



(b) Credit Limit



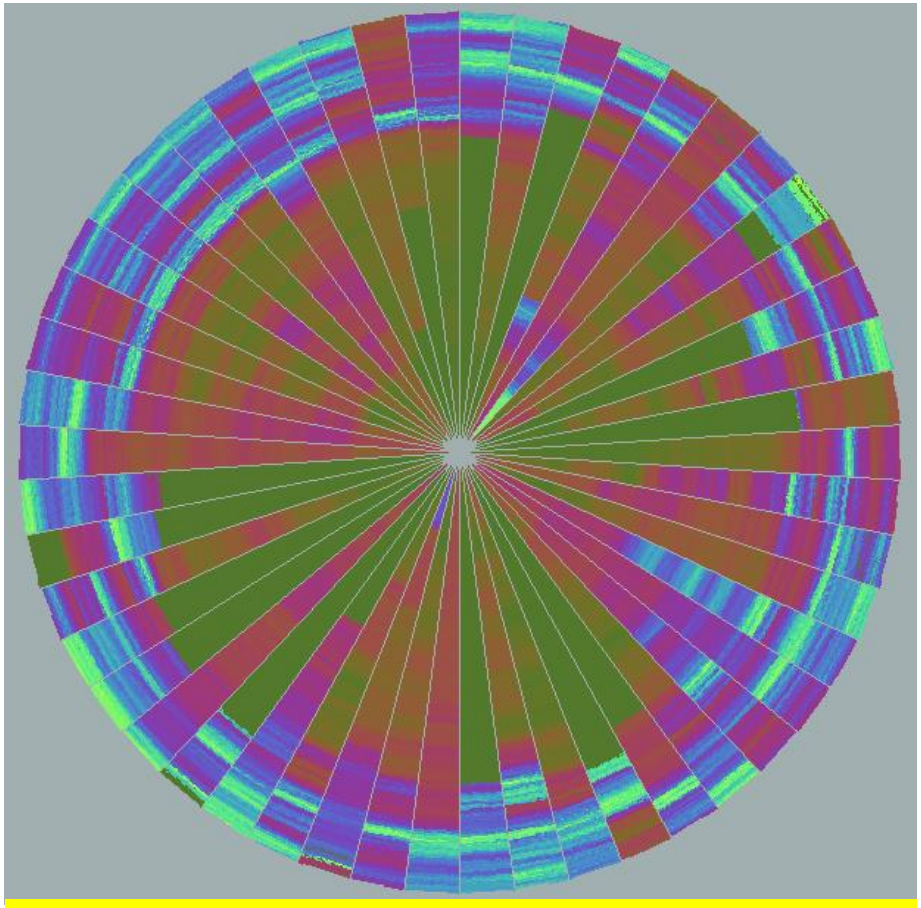
(c) transaction volume



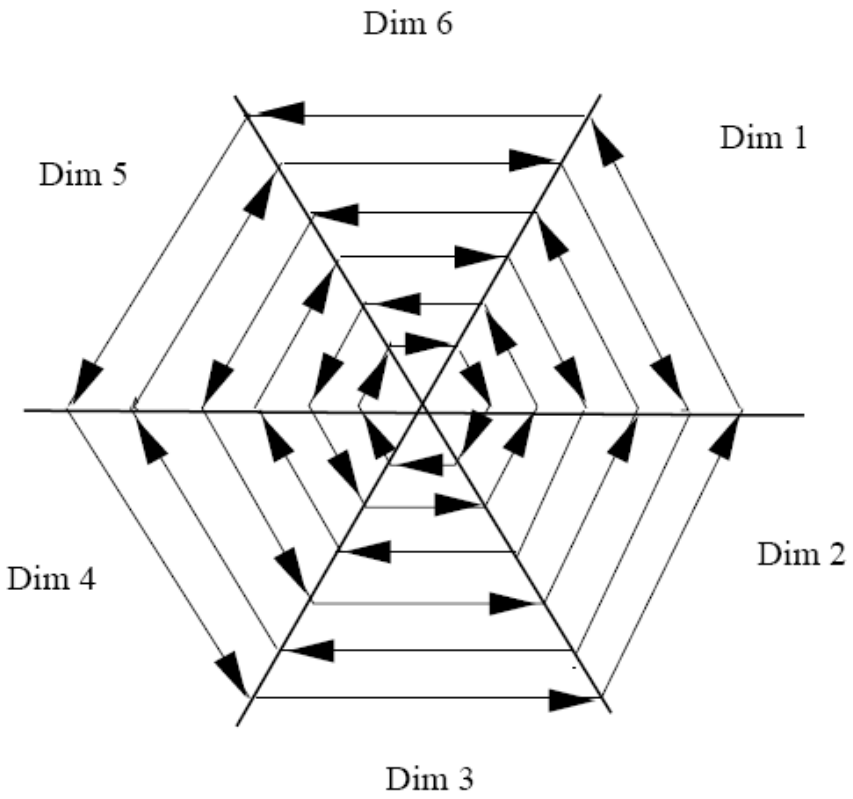
(d) age

Laying Out Pixels in Circle Segments

- To save space and show the connections among multiple dimensions, space filling is often done in a circle segment



Representing about 265,000 50-dimensional Data Items with the 'Circle Segments' Technique

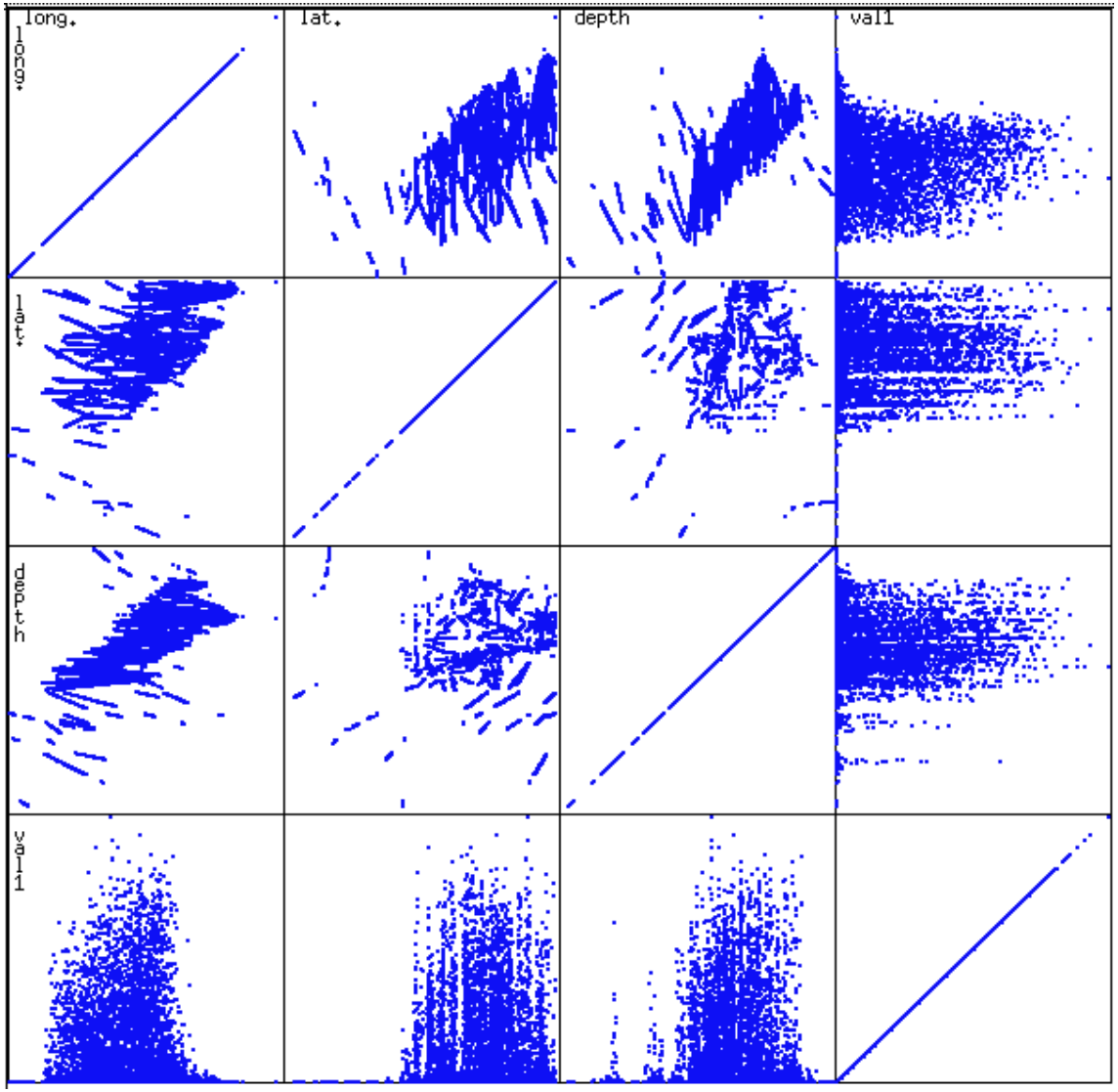


(b) Laying out pixels in circle segment

Geometric Projection Visualization Techniques

- Visualization of geometric transformations and projections of the data
- Methods
  - Direct visualization
  - Scatterplot and scatterplot matrices
  - Landscapes
  - Projection pursuit technique: Help users find meaningful projections of multidimensional data
  - Prosection views
  - Hyperslice
  - Parallel coordinates

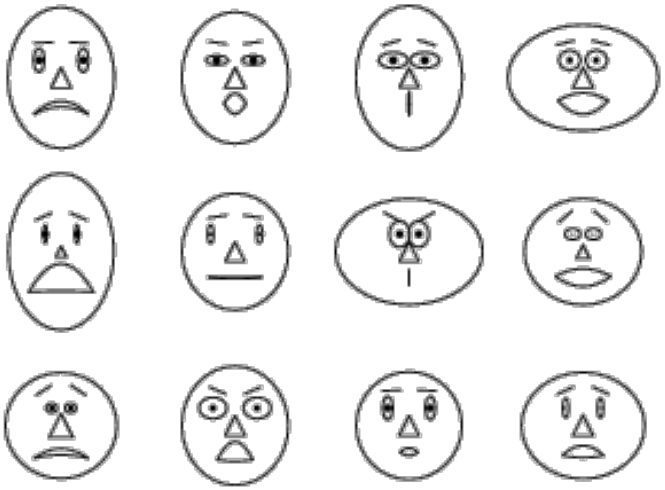
Scatterplot Matrices



Matrix of scatterplots (x-y-diagrams) of the k-dim. data [total of (k2/2-k) scatterplots]

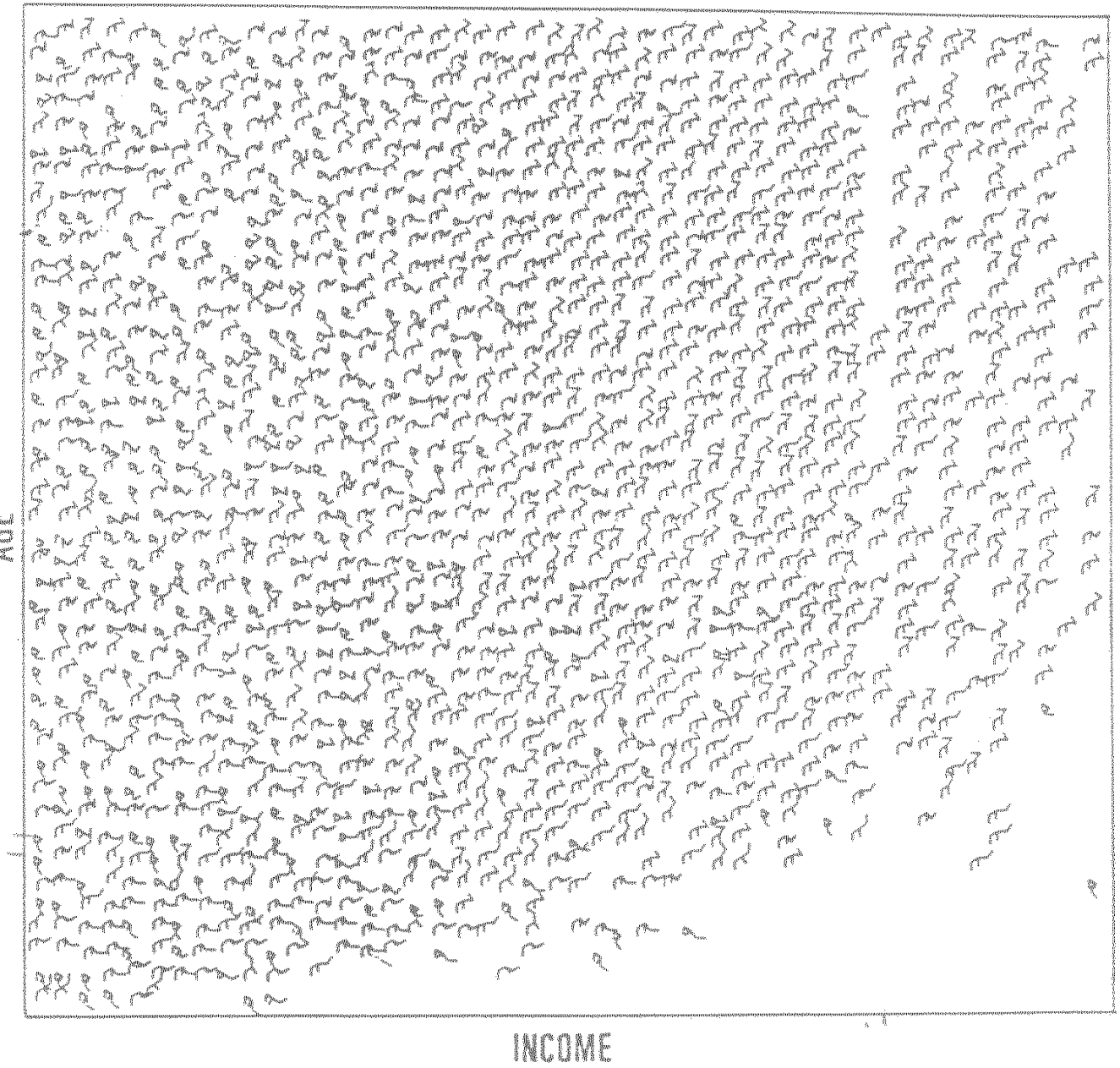
Chernoff Faces

- A way to display variables on a two-dimensional surface, e.g., let x be eyebrow slant, y be eye size, z be nose length, etc.
- The figure shows faces produced using 10 characteristics--head eccentricity, eye size, eye spacing, eye eccentricity, pupil size, eyebrow slant, nose size, mouth shape, mouth size, and mouth opening): Each assigned one of 10 possible values, generated using *Mathematica* (S. Dickson)



- REFERENCE: Gonick, L. and Smith, W. *The Cartoon Guide to Statistics.* New York: Harper Perennial, p. 212, 1993
- Weisstein, Eric W. "Chernoff Face." From *MathWorld*--A Wolfram Web Resource. mathworld.wolfram.com/ChernoffFace.html

Stick Figure



A census data figure showing age, income, gender, education, etc.

A 5-piece stick figure (1 body and 4 limbs w. different angle/length)

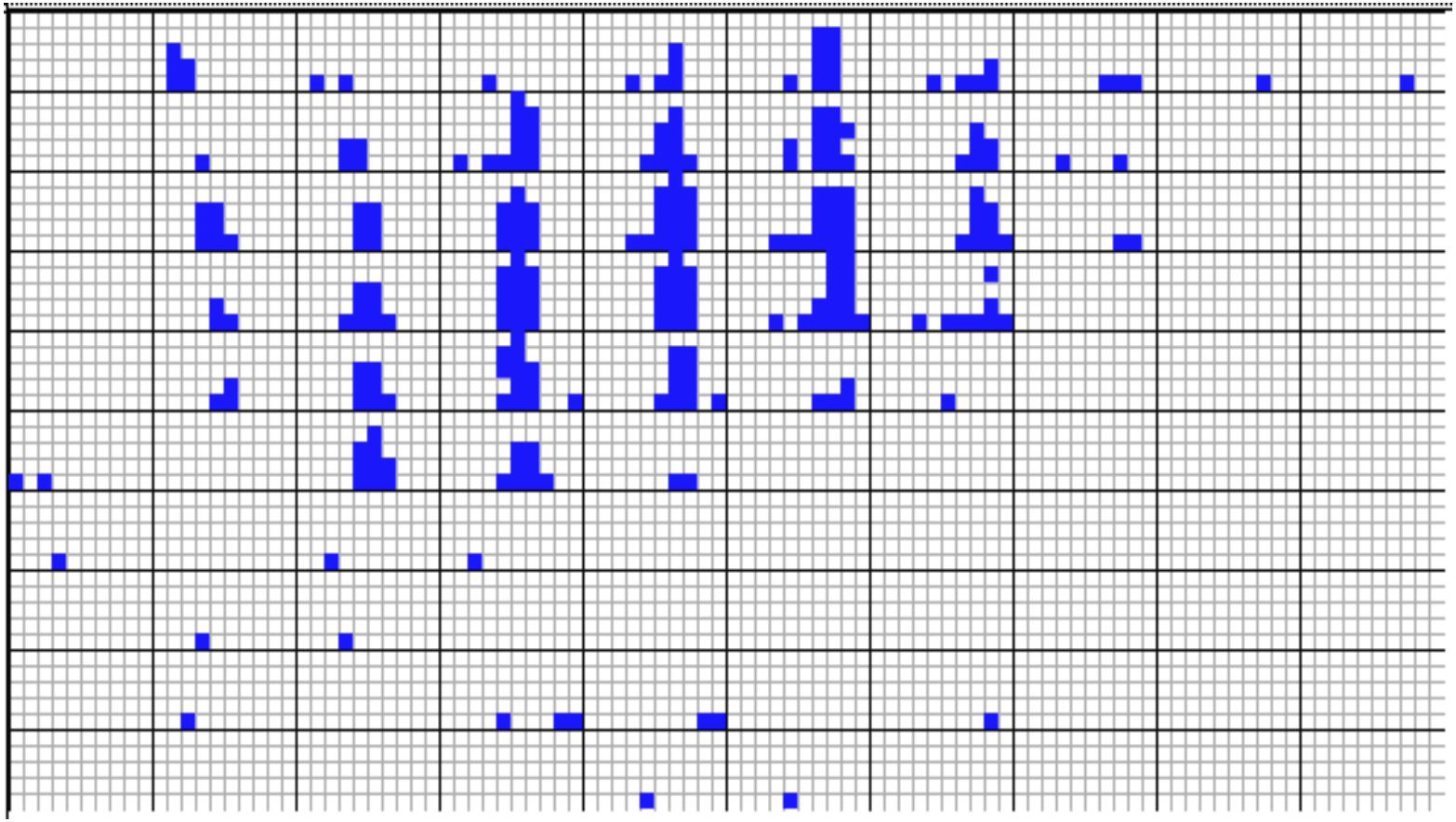
Hierarchical Visualization Techniques

- Visualization of the data using a hierarchical partitioning into subspaces
- Methods
  - Dimensional Stacking
  - Worlds-within-Worlds
  - Tree-Map
  - Cone Trees
  - InfoCube

Dimensional Stacking

- Partitioning of the n-dimensional attribute space in 2-D subspaces, which are ‘stacked’ into each other
- Partitioning of the attribute value ranges into classes. The important attributes should be used on the outer levels.
- Adequate for data with ordinal attributes of low cardinality
- But, difficult to display more than nine dimensions
- Important to map dimensions appropriately

Dimensional Stacking



Visualization of oil mining data with longitude and latitude mapped to the outer x-, y-axes and ore grade and depth mapped to the inner x-, y-axes

Measuring data Similarity and Dissimilarity

- **Similarity**
  - Numerical measure of how alike two data objects are
  - Value is higher when objects are more alike
  - Often falls in the range [0,1]
- **Dissimilarity** (e.g., distance)
  - Numerical measure of how different two data objects are
  - Lower when objects are more alike
  - Minimum dissimilarity is often 0
  - Upper limit varies
- **Proximity** refers to a similarity or dissimilarity

Proximity Measure for Nominal Attributes

Method 1: Simple matching

- m: number of matches,
- p: total number of variables

$$d(i,j) = \frac{p-m}{p}$$

Example:

Objects	code
1	Code A
2	Code B
3	Code C
4	Code A

Proximity Measure for Nominal Attributes

Objects	code
1	Code A
2	Code B
3	Code C
4	Code A

$$\begin{matrix} 0 & & & \\ d(2,1) & 0 & & \\ d(3,1) & d(3,2) & 0 & \\ d(4,1) & d(4,2) & d(4,3) & 0 \end{matrix}$$

$$d(i,j)=\frac{p-m}{p}$$

- d(i,j) is 0 if objects I and j match
- d(i,j) is 1 if objects I and j differ

$$\begin{matrix} 0 & & & \\ 1 & 0 & & \\ 1 & 1 & 0 & \\ 0 & 1 & 1 & 0 \end{matrix}$$

Proximity Measure for Binary Attributes

- Distance measure for symmetric binary variables:

$$d(i, j) = \frac{r + s}{q + r + s + t}$$

- Distance measure for asymmetric binary variables:

$$d(i, j) = \frac{r + s}{q + r + s}$$

Object *j*

Object *i*	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

q:11
r: 10
s: 01
t: 00

Dissimilarity between Binary Variables

- Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

  - Gender is a symmetric attribute
  - The remaining attributes are asymmetric binary
  - Let the values Y and P be 1, and the value N be 0

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

Output: results for Jack and mary are more similar

Dissimilarity of Numeric data

- Distance measure are commonly used for computing the dissimilarity of objects described by numeric attribute

- These measures include the Euclidean, Manhattan and Minkowski distance

Dissimilarity of Numeric data

- Let i and j are two objects described by p numeric attributes

$i = (x_{i1}, x_{i2}, \ldots, x_{ip})$

$j = (x_{j1}, x_{j2}, \ldots, x_{jp})$

- The Euclidean distance between objects I and j is defined as

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + \ldots + |x_{i_p} - x_{j_p}|^2)}$$

Dissimilarity of Numeric data

- Manhattan distance

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$

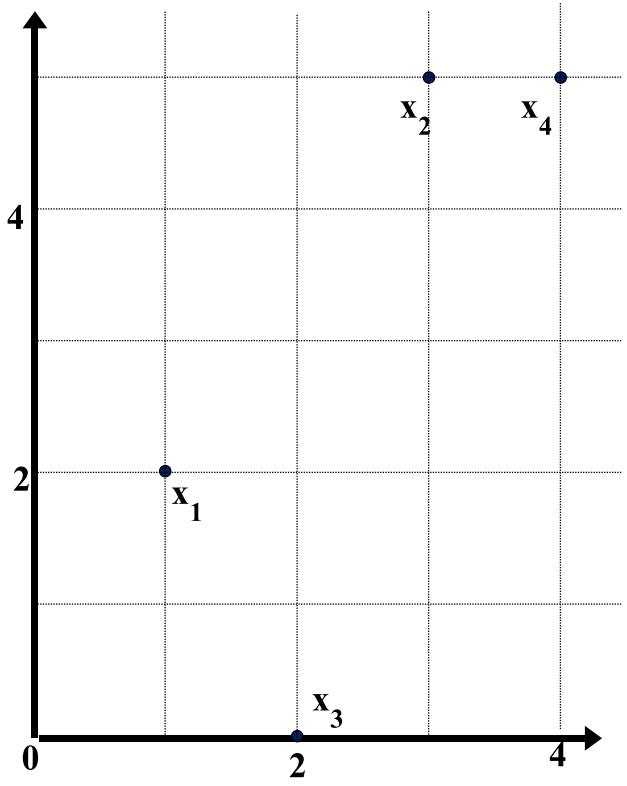
- Minkowski distance

$$d(i,j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \cdots + |x_{ip} - x_{jp}|^h}$$

Example: Data Matrix and Dissimilarity Matrix

Data Matrix

point	attribute 1	attribute 2
x1	1	2
x2	3	5
x3	2	0
x4	4	5



Dissimilarity Matrices

Manhattan (L_1)

L	x1	x2	x3	x4
x1	0			
x2	5	0		
x3	3	6	0	
x4	6	1	7	0

Euclidean (L_2)

L2	x1	x2	x3	x4
x1	0			
x2	3.61	0		
x3	2.24	5.1	0	
x4	4.24	1	5.39	0

d(x1,x1)=(1-1)+(2-2)=0
d(x2,x1)=(3-1)+(5-2)=5
d(x3,x1)=(2-1)+(0-2)=3
d(x4.x1)=(4-1)+(5-2)=6

Cosine Similarity

- A **document** can be represented by thousands of attributes, each recording the *frequency* of a particular word (such as keywords) or phrase in the document.

Document	*team*	*coach*	*hockey*	*baseball*	*soccer*	*penalty*	*score*	*win*	*loss*	*season*
Document1	5	0	3	0	2	0	0	2	0	0
Document2	3	0	2	0	1	1	0	1	0	1
Document3	0	7	0	2	1	0	0	3	0	0
Document4	0	1	0	0	1	2	2	0	3	0

- 0 value means that document do not share the word
- But this does not make them similar
- We need a measure that will focus on the words that the two documents do have in common.
- Cosine similarity is a measure of similarity that can be used to compare documents

Example: Cosine Similarity

- Cos (x, y) =sim (x, y)= ($x \bullet$ y) /||x|| ||y||

  → where • indicates vector dot product,

  →||x||: $(x_1*x_1+x_2*x_2+.....+x_p*x_p)^{0.5}$

- Ex: Find the **similarity** between documents 1 and 2.

  d_1 = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)
  d_2 = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)

  $d_1 \bullet d_2$ = 5*3+0*0+3*2+0*0+2*1+0*1+0*1+2*1+0*0+0*1 = 25

  $||d_1||$= $(5*5+0*0+3*3+0*0+2*2+0*0+0*0+2*2+0*0+0*0)^{0.5}=(42)^{0.5}$ = 6.481
  $||d_2||$= $(3*3+0*0+2*2+0*0+1*1+1*1+0*0+1*1+0*0+1*1)^{0.5}=(17)^{0.5}$ = 4.12

  cos(d1,d2)=25/(6.481*4.12)
  cos(d_1, d_2) = 0.94

Cosine Similarity

- Cosine value of zero means that two vectors are at 90 degree to each other and have no match

- The closer the cosine value to 1, the smaller is the angle and greater is the match between vectors

Ordinal Variables

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled
  - replace x_{if} by their rank $r_{if} \in \{1, \ldots, M_f\}$
  - map the range of each variable onto [0, 1] by replacing *i*-th object in the *f*-th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - compute the dissimilarity using methods for interval-scaled variables

Attributes of Mixed Type

- A database may contain all attribute types
  - Nominal, symmetric binary, asymmetric binary, numeric, ordinal
- One may use a weighted formula to combine their effects

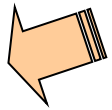
$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

  - f is binary or nominal:

    $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$, or $d_{ij}^{(f)} = 1$ otherwise
  - f is numeric: use the normalized distance
  - f is ordinal
    - Compute ranks r_{if} and
    - Treat z_{if} as interval-scaled

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

Chapter 2: Getting to Know Your Data

- Data Objects and Attribute Types
- Basic Statistical Descriptions of Data
- Data Visualization
- Measuring Data Similarity and Dissimilarity
- Summary 

Chapter 2: Data Preprocessing

Why Data Preprocessing?

- Data in the real world is dirty
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., occupation=“ ”
  - noisy: containing errors or outliers
    - e.g., Salary=“-10”
  - inconsistent: containing discrepancies in codes or names
    - e.g., Age=“30” Birthday=“03/07/1980”
    - e.g., Was rating “1,2,3”, now rating “A, B, C”
    - e.g., discrepancy between duplicate records

Why Is Data Dirty?

- Incomplete data may come from
  - “Not applicable” data value when collected
  - Different considerations between the time when the data was collected and when it is analyzed.
  - Human/hardware/software problems
- Noisy data (incorrect values) may come from
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission
- Inconsistent data may come from
  - Different data sources
  - Functional dependency violation (e.g., modify some linked data)
- Duplicate records also need data cleaning

Why Is Data Preprocessing Important?

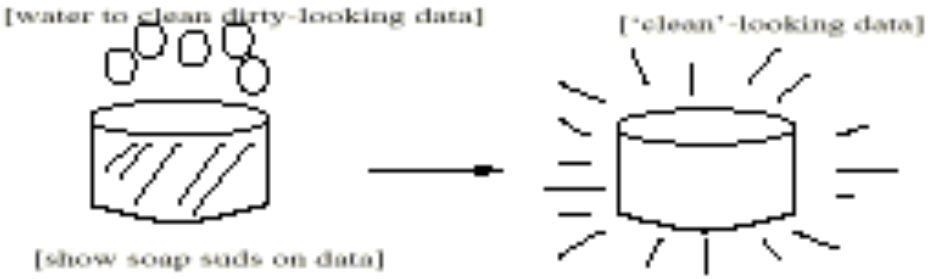
- No quality data, no quality mining results!
  - Quality decisions must be based on quality data
    - e.g., duplicate or missing data may cause incorrect or even misleading statistics.
  - Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse

Major Tasks in Data Preprocessing

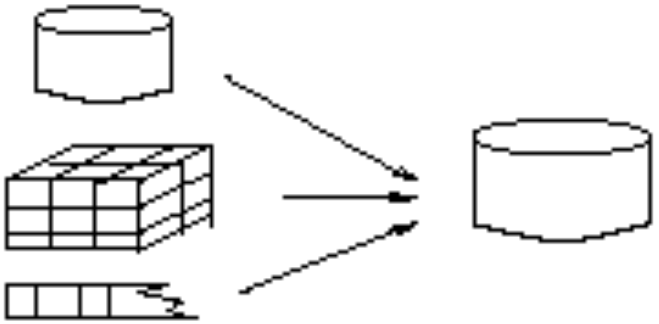
- Data cleaning
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- Data integration
  - Integration of multiple databases, data cubes, or files
- Data transformation
  - Normalization and aggregation
- Data reduction
  - Obtains reduced representation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

Forms of Data Preprocessing

Data Cleaning



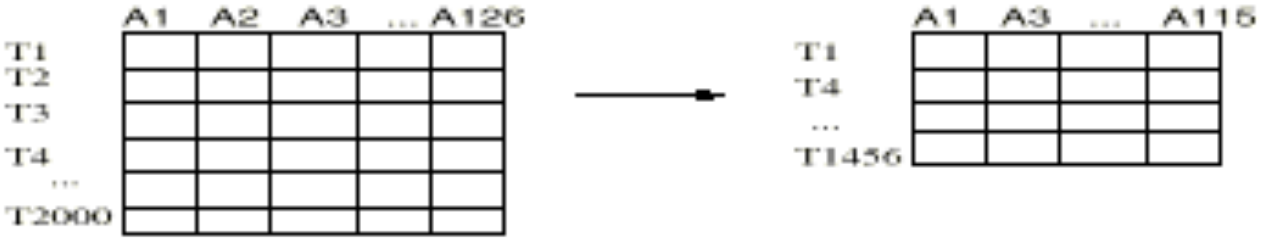
Data Integration



Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction



Data Cleaning

- Importance
  - “Data cleaning is one of the three biggest problems 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

Missing Data

- Data is not always available
  - E.g., many tuples have no recorded value 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
  - not register history or changes of the data
- Missing data may need to be inferred.

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
  - a global constant : e.g., “unknown”
  - the attribute mean
  - the attribute mean for all samples belonging to the same class: smarter
  - the most probable value: inference-based such as **Bayesian formula** or **decision tree**

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
  - technology limitation
  - inconsistency in naming convention

How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers

Binning method

- Equal-depth (frequency) partitioning
  - Divides the range into *N* intervals, each containing approximately same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky
- Equal-width (distance) partitioning
  - Divides the range into *N* intervals of equal size
  - if *A* and *B* are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$.
  - The most straightforward, but outliers may dominate presentation
  - Skewed data is not handled well

Binning Methods for Data Smoothing (Example)

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

* Partition into equal-frequency (equi-depth) bins:

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

1)Smoothing by bin means:

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

2)Smoothing by bin boundaries:

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

Q) 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** by each of the following methods.

(a) equal-frequency partitioning
(b) equal-width partitioning
(c) clustering

(a) equal-frequency partitioning

bin 1 -- 5,10,11,13
bin 2 -- 15,35,50,55
bin 3 -- 72,92,204,215

(b) equal-width partitioning

The width of each interval is (215 - 5)/3 = 70.
bin 1 -- 5,10,11,13,15,35,50,55,72 (5 to 75)
bin 2 -- 92 (76 to 146)
bin 3 -- 204,215 (147 to 217)

(c) clustering

We will use a simple clustering technique:

Partition the data along the 2 biggest gaps in the data.

bin 1	5,10,11,13,15
bin 2	35,50,55,72,92
bin 3	204,215

(ex. 5; 10; 11; 13; 15; 35; 50; 55; 72; 92; 204; 215;)

Data Integration

- 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
- 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

Handling Redundancy in Data Integration

- 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*
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Redundancy and Correlation Analysis

- Some redundancy can be detected by correlation analysis.

- Such analysis can measure how strongly one attributes implies the other based on the available data

  - For nominal data, x^2 (chi-square) test
  - For numeric data correlation coefficient and covariance

Correlation Analysis (Categorical Data)

- X^2 (chi-square) test

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- $Expected = \frac{(count\ A) * (count\ B)}{n}$

- The larger the X^2 value, the more likely the variables are related

Chi-Square Calculation: An Example

	Play chess	Not play chess	Sum (row): A
Like science fiction	250 (exp: 90)	200 (exp:360)	450
Not like science fiction	50 (exp:210)	1000(exp:840)	1050
Sum(col.): B	300	1200	1500 (n)

$$expected = \frac{(count\ A)*(count\ B)}{n} = \frac{450*300}{1500} = 90$$

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

- It shows that like_science_fiction and play_chess are correlated in the group

- We can get X^2 by:

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840}$$
$$= 284.44 + 121.90 + 71.11 + 30.48 = 507.93.$$

- For this 2 x 2 table, the degrees of freedom are (2-1)(2-1) = 1.
- For 1 degree of freedom, the X^2 value needed to reject the hypothesis at the 0.001 significance level is 10.828 (taken from the table of upper percentage points of the X^2 distribution, typically available from any textbook on statistics).

507.93 >10.82

Calculated value is more than tabulated value of X^2

So,

Null hypothesis: play chess and preferred reading are independent (not related)

is rejected and conclude that the two attributes are strongly correlated for the given group of people

Correlation Analysis (Numeric Data)

- Correlation coefficient (also called Pearson's product moment coefficient)

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

  - If $r_{A,B} > 0$,
    A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.
  - $r_{A,B} = 0$: independent;
  - $r_{A,B} < 0$: negatively correlated

Covariance (Numeric Data)

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

where $\overline{A}$ and $\overline{B}$ are the respective mean or **expected values** of A and B

Positive covariance: If $Cov_{A,B} > 0$, then A and B both tend to be larger than their expected values

Negative covariance: If $Cov_{A,B} < 0$ then if A is larger than its expected value, B is likely to be smaller than its expected value

Independence: $Cov_{A,B} = 0$ 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

Co-Variance: An Example

- It can be simplified in computation as

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

$$Cov(x,y) = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{N}$$

Example:

- 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?**
  - $\bar{A}$ = (2 + 3 + 5 + 4 + 6)/ 5 = 20/5 = 4
  - $\bar{B}$ = (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.

Days	Stock A	Stock B
Monday	2	5
Tuesday	3	8
Wednesday	5	10
Thursday	4	11
Friday	6	14

Data Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values so that each old value can be identified with one of the new values
- Methods
  - **Smoothing**: Remove noise from data
    - Techniques include binning, regression, clustering
  - **Attribute/feature construction**
    - New attributes constructed from the given ones
  - **Aggregation**: Summarization, data cube construction
    - Eg. Daily sales data aggregated to monthly and annual income

Data Transformation

- **Normalization**: attribute data are scaled to fall within a smaller range such as (-1.0 to 1.0) or (0.0 to 1.0)
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling

- **Discretization**: raw values of numeric attributes (e.g. age) are replaced by interval labels (0-10, 11-20) or conceptual labels(youth, adult, senior) resulting in Concept hierarchy for the numeric attributes

- **Concept hierarchy generation for nominal data:**
- Attribute such as street can be generalized to higher level concept, like city or country

Normalization

- **Min-max normalization**: it maps a value v of attribute A to new value v' in the range [new_min$_A$, new_max$_A$] by computing,

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

  - Ex. Let min and max value for the attribute income are \$12,000 and \$98,000 resp. Now map income to the range [0.0, 1.0]. Then the value \$73,600 for income is transformed as,

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

Normalization

- **Z-score normalization:** The values for attribute A are normalized based on mean and (μ) standard deviation(σ) of attribute A . Formula is given by,

$$v' = \frac{v - \mu_A}{\sigma_A}$$

- Ex. Let μ = 54,000, σ = 16,000 for attribute income. With z-score normalization, a value $73600 for income is transformed to ,

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

Normalization

- **Normalization by decimal scaling**: It transform the value by moving the decimal point of value of attribute A. The number of decimal points moved depends on the maximum absolute value of A.
- formula is given by, $$v' = \frac{v}{10^j}$$ Where j is the smallest integer such that Max(|v'|) < 1

- Eg. Let for attribute A range is -986 to 927. To normalize by decimal scaling, divide each value by 1000(i.e. j=3).
  Therefore -986 is normalized to -0.986 and 917 is normalized to 0.917.

Example

Use the two methods below to *normalize* the following group of data:

200*;* 300*;* 400*;* 600*;* 1000

(a) min-max normalization by setting *min* = 0 and *max* = 1

(b) z-score normalization

age	23	23	27	27	39	41	47	49	50
z-age	-1.83	-1.83	-1.51	-1.51	-0.58	-0.42	0.04	0.20	0.28
%fat	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2
z-%fat	-2.14	-0.25	-2.33	-1.22	0.29	-0.32	-0.15	-0.18	0.27

age	52	54	54	56	57	58	58	60	61
z-age	0.43	0.59	0.59	0.74	0.82	0.90	0.90	1.06	1.13
%fat	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7
z-%fat	0.65	1.53	0.0	0.51	0.16	0.59	0.46	1.38	0.77

Using the data for *age* given in Exercise 2.4, answer the following:

(a) Use min-max normalization to transform the value 35 for *age* onto the range [0:0*;* 1:0]

(b) Use z-score normalization to transform the value 35 for *age*, where the standard deviation of *age* is 12.94 years.

(c) Use normalization by decimal scaling to transform the value 35 for *age*.

Concept Hierarchy Generation

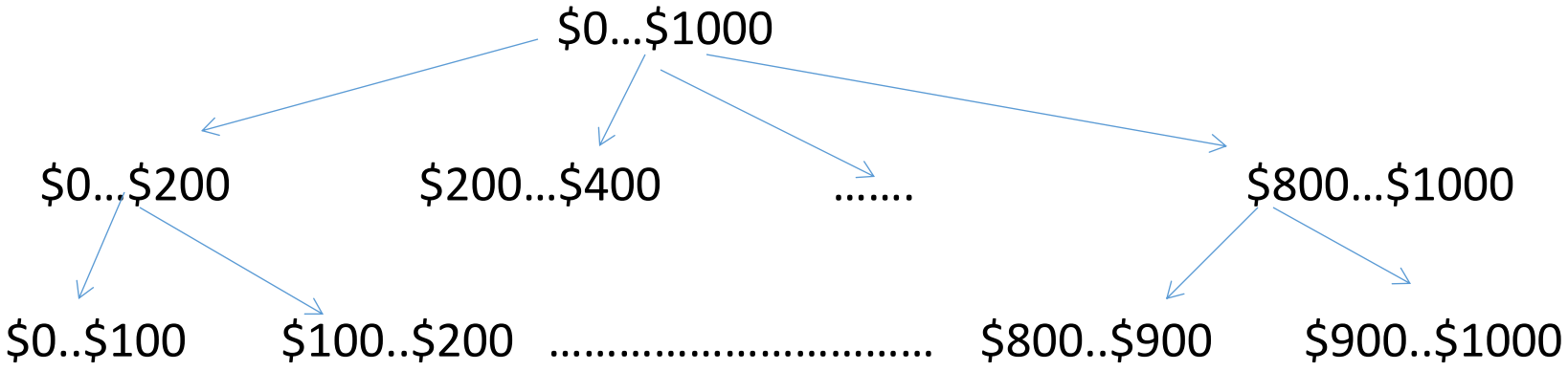
- **Concept hierarchy** organizes concepts (i.e., attribute values) hierarchically and is usually associated with each dimension in a data warehouse

- Concept hierarchies facilitate drilling and rolling in data warehouses to view data in multiple granularity

- Concept hierarchy formation: Recursively reduce the data by collecting and replacing low level concepts (such as numeric values for *age*) by higher level concepts (such as *youth, adult*, or *senior*)

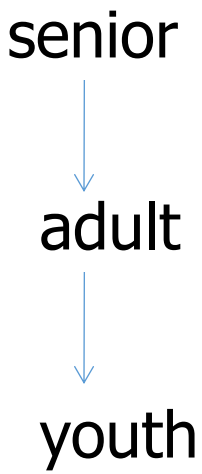
- Concept hierarchies can be explicitly specified by domain experts and/or data warehouse designers

Concept hierarchy for the numeric attributes (Discretization)



Concept hierarchy for attribute price (interval labels)

Concept hierarchy for attribute Age (conceptual labels)



Concept Hierarchy Generation for Nominal Data

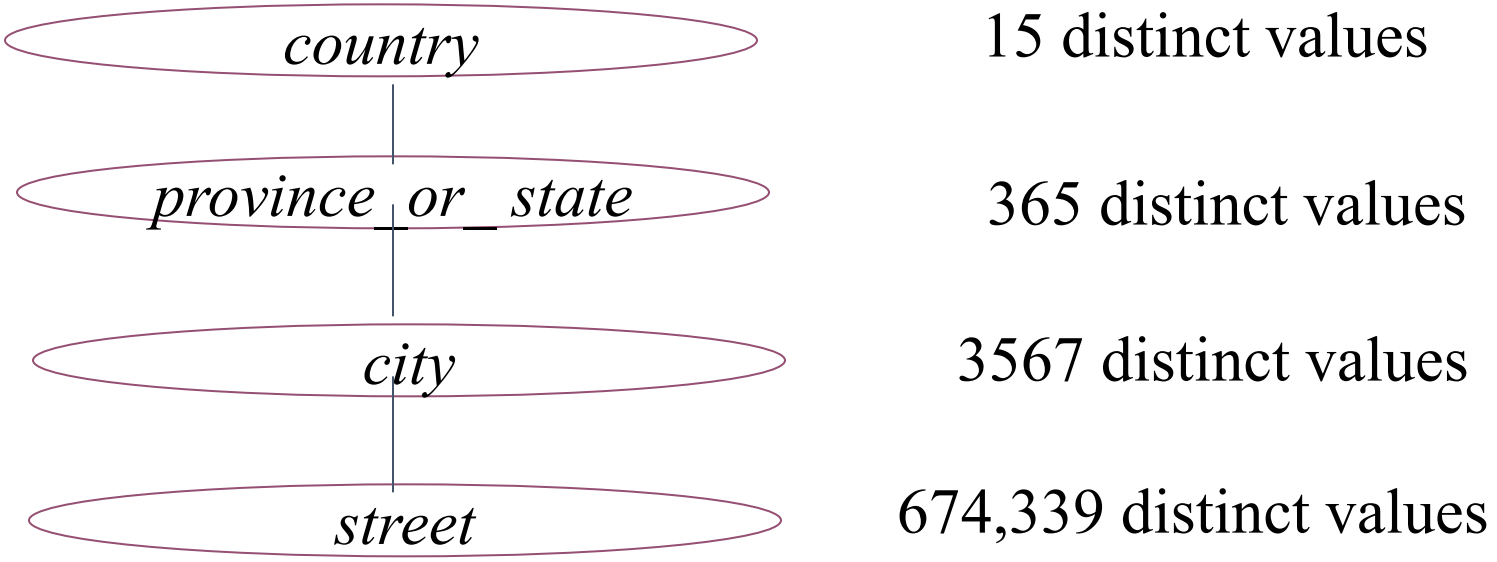
1. **Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts.**
   - *street* < *city* < *state* < *country*

2. **Specification of a hierarchy for a set of values by explicit data grouping**
   - {Punjab, Haryana, Delhi} < North_India
   - {Karnataka, Tamilnadu, kerala} < South_India

3. **Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values**

Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
  - The attribute with the most distinct values is placed at the lowest level of the hierarchy



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)

Data Reduction

- Why data reduction?
  - A database/data warehouse may store terabytes of data
  - Complex data analysis/mining may take a very long time to run on the complete data set

- Data reduction
  - Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results

Data reduction strategies

- Data cube aggregation:
- Eg. Total sales per year instead of per quarter

- Dimensionality reduction: original data is transform into smaller space. Encoding methods are used.
- Eg. Wavelet transform and principal component analysis

- Attribute subset selection: remove unimportant(irrelevant, redundant, weakly relevant) attributes
- Eg. For new CD purchase, customers phone number is irrelevant

Data reduction strategies

- Numerosity reduction : replace original data by alternatives smaller form of data representation
  - Parametric method: model is used to estimate the data
    - Eg. Regression, log-linear models
  - Non Parametric method
    - Eg. Histograms, clustering, sampling and Data cube aggregation

Data Cube Aggregation

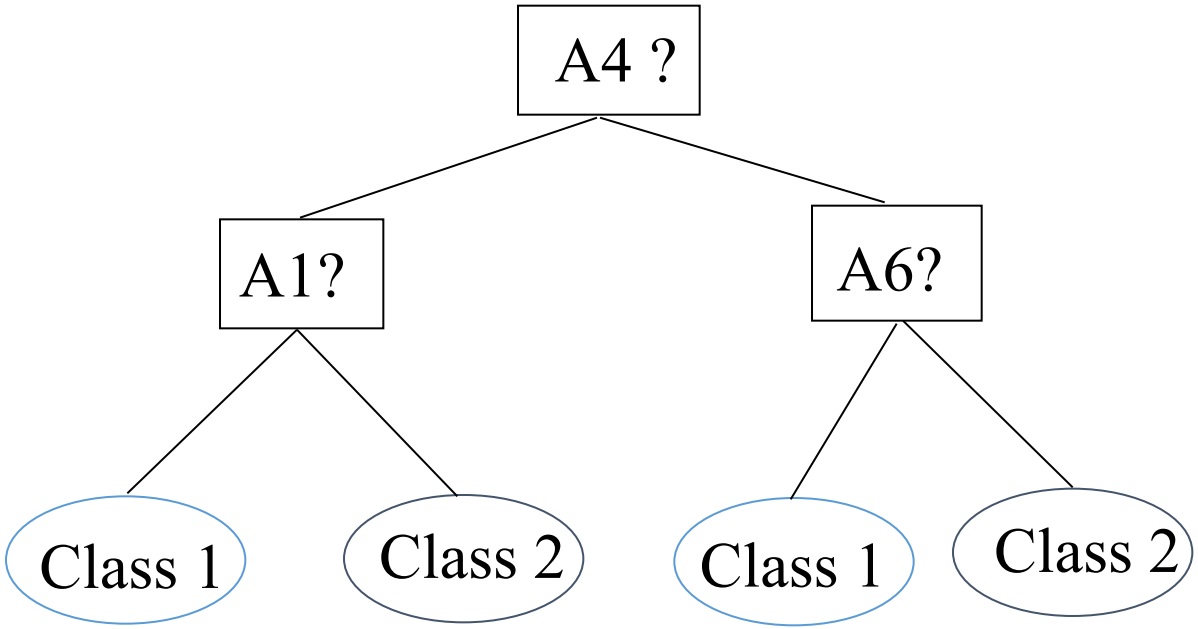
- The lowest level of a data cube (base cuboid)
  - The aggregated data for an individual entity of interest
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

Attribute Subset Selection

- 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
  - reduce # of patterns in the patterns, easier to understand
- Heuristic methods (due to exponential # of choices):
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

Example of Decision Tree Induction

Initial attribute set:
{A1, A2, A3, A4, A5, A6}



------> Reduced attribute set: {A1, A4, A6}

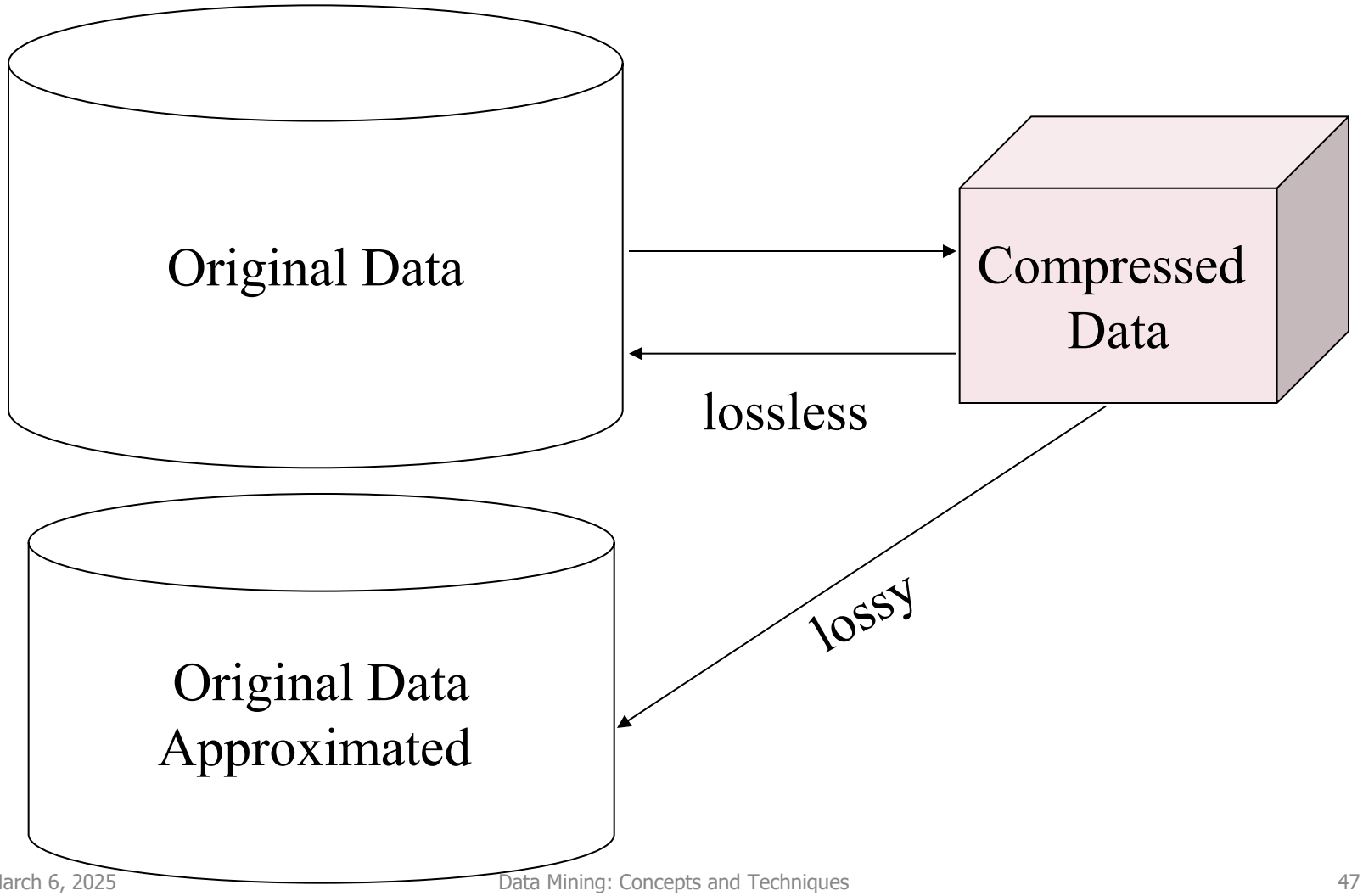
Heuristic Feature Selection Methods

- There are 2^d possible sub-features of d features
- Several heuristic feature selection methods:
  - Best single features under the feature independence assumption: choose by significance tests
  - Best step-wise feature selection:
    - The best single-feature is picked first
    - Then next best feature condition to the first, ...
  - Step-wise feature elimination:
    - Repeatedly eliminate the worst feature
  - Best combined feature selection and elimination
  - Optimal branch and bound:
    - Use feature elimination and backtracking

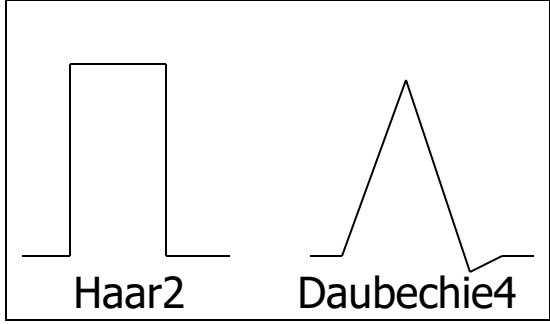
Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless
  - But only limited manipulation is possible without expansion
- Audio/video 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

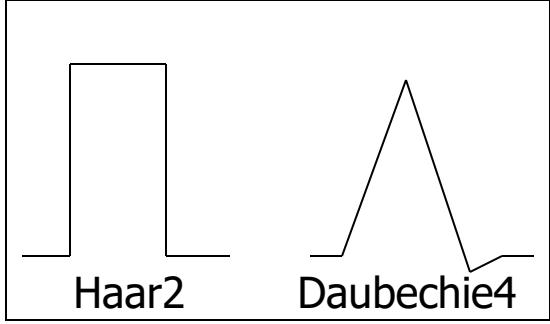


Dimensionality Reduction: Wavelet Transformation



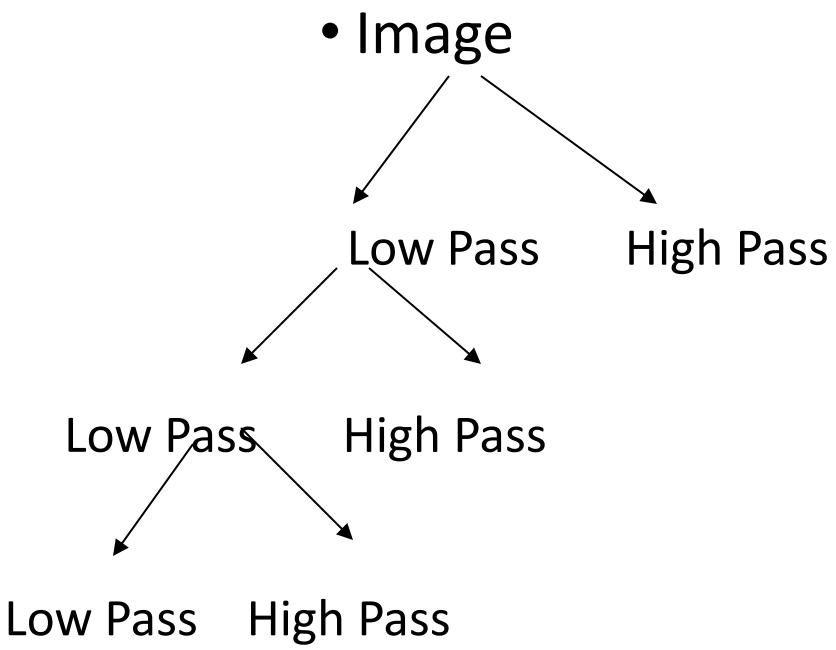
- Data encoding techniques or transformation are applied on the original data to obtain a reduced or compressed representation of the data.
- Reconstructed data can be lossy or lossless
- Lossy dimensionality reduction methods:
  - Wavelet transforms(DWT)
    - Haar tranform
  - Principle component analysis(PCA)
    - K-L method

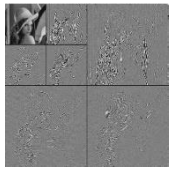
Dimensionality Reduction: Wavelet Transformation



- Discrete wavelet transform (DWT): linear signal processing, multi-resolutional analysis
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients
- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space
- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two set of data of length L/2
  - Applies two functions recursively, until reaches the desired length

DWT for Image Compression

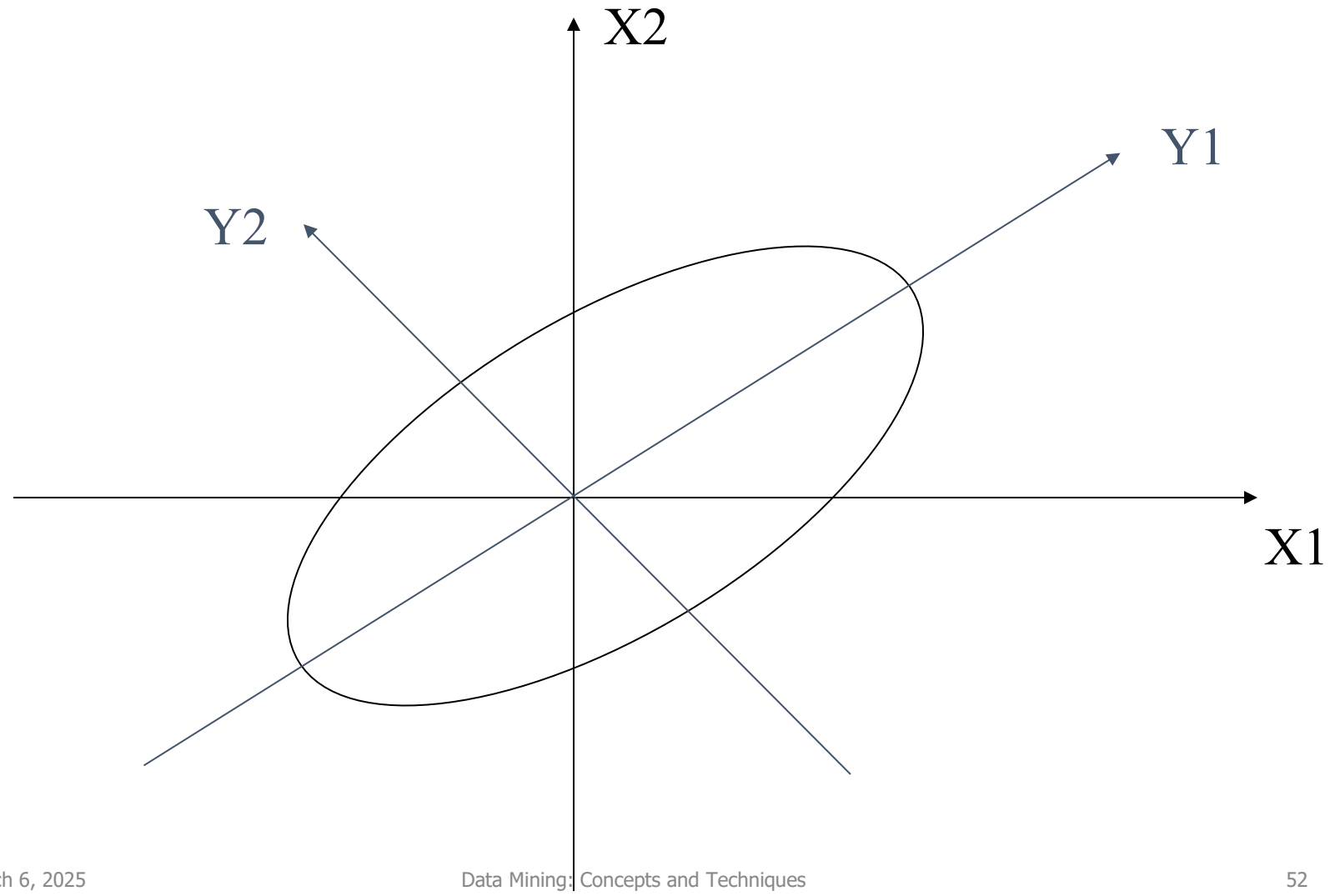




Dimensionality Reduction: Principal Component Analysis (PCA)

- Given *N* data vectors from *n*-dimensions, find $k \leq n$ orthogonal vectors (*principal components*) that can be best used to represent data
- Steps
  - Normalize input data: Each attribute falls within the same range
  - Compute *k* orthonormal (unit) vectors, i.e., *principal components*
  - Each input data (vector) is a linear combination of the *k* principal component vectors
  - The principal components are sorted in order of decreasing "significance" or strength
  - Since the components are sorted, the size of the data can be reduced by eliminating the weak components, i.e., those with low variance. (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data
- Works for numeric data only
- Used when the number of dimensions is large

Principal Component Analysis



Numerosity Reduction

- Reduce data volume by choosing alternative, smaller forms of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Example: Regression and 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 Method (1): Regression and Log-Linear Models

- Linear regression: Data are modeled to fit a straight line

  $y = wx + b$

Where, y and x--→numerical database attributes

w and b-→ regression coefficient

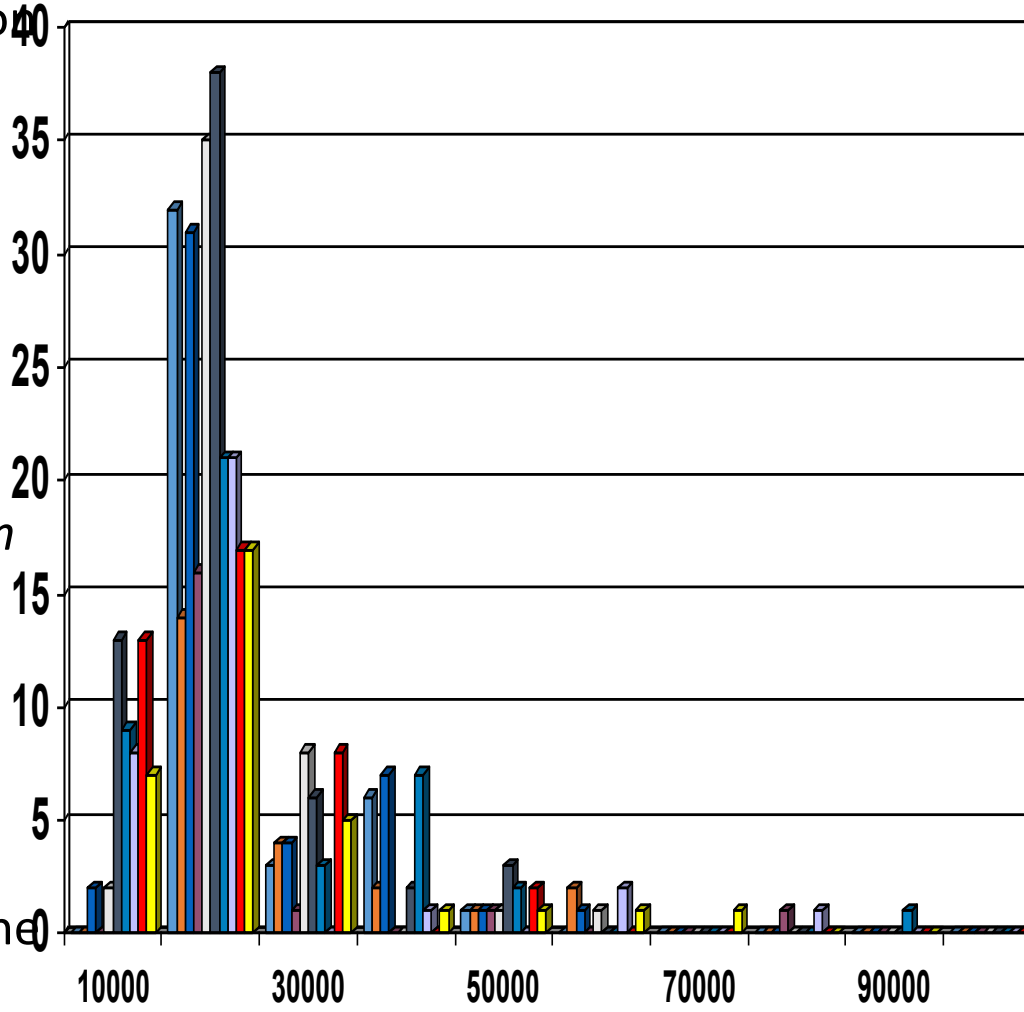
- Multiple regression: extension of linear regression method where y is modeled as a linear function of two or more predictor variable
- Log-linear model: estimate the probability of each point in a multidimentional space for a set of discretized attributes, based on the smaller subset of dimensional combinations

Regress Analysis and Log-Linear Models

- Linear regression: $Y = w X + b$
  - Two regression coefficients, *w* and *b,* specify the line and are to be estimated by using the data at hand
  - Using the least squares criterion to the known values of $Y_1, Y_2, \ldots, X_1, X_2, \ldots$.
- Multiple regression: $Y = b0 + b1 X1 + b2 X2.$
  - Many nonlinear functions can be transformed into the above
- Log-linear models:
  - The multi-way table of joint probabilities is approximated by a product of lower-order tables
  - Probability: $p(a, b, c, d) = \alpha ab \, \beta ac \chi ad \, \delta bcd$

Data Reduction Method (2): Histograms

- Histogram for an attribute A→ partitions data distribution of A into disjoint subsets or buckets.
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)
  - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)
  - MaxDiff: set bucket boundary between each pair for pairs have the β–1 largest differences



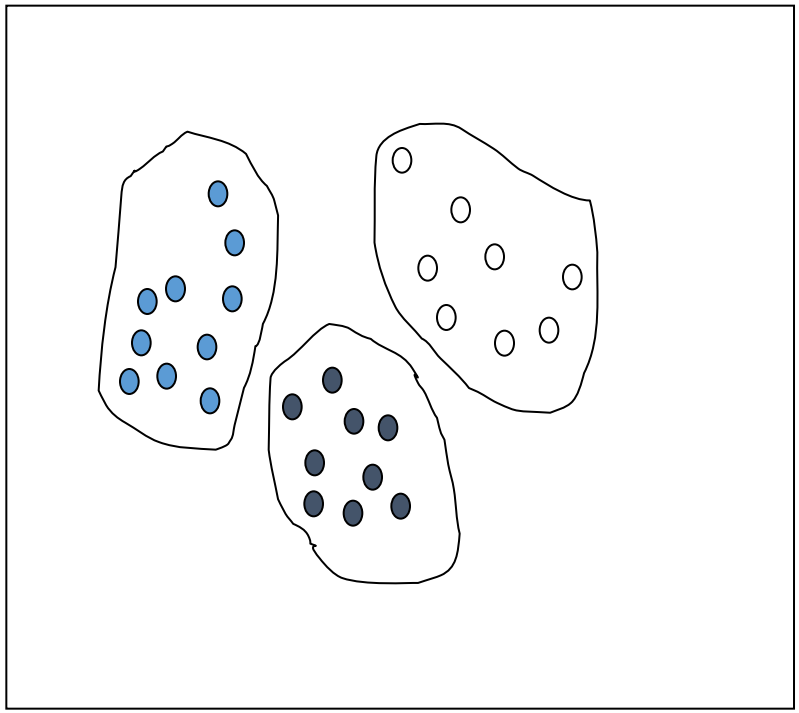
- List of prices of commonly sold items at AllElectronics(rounded to nearest dollar)
- 1,1,5,5,5,5,8,8,10,10,10,10,10,12,14,14,14,14,15,15,15,15,15,15

Data Reduction Method (3): Clustering

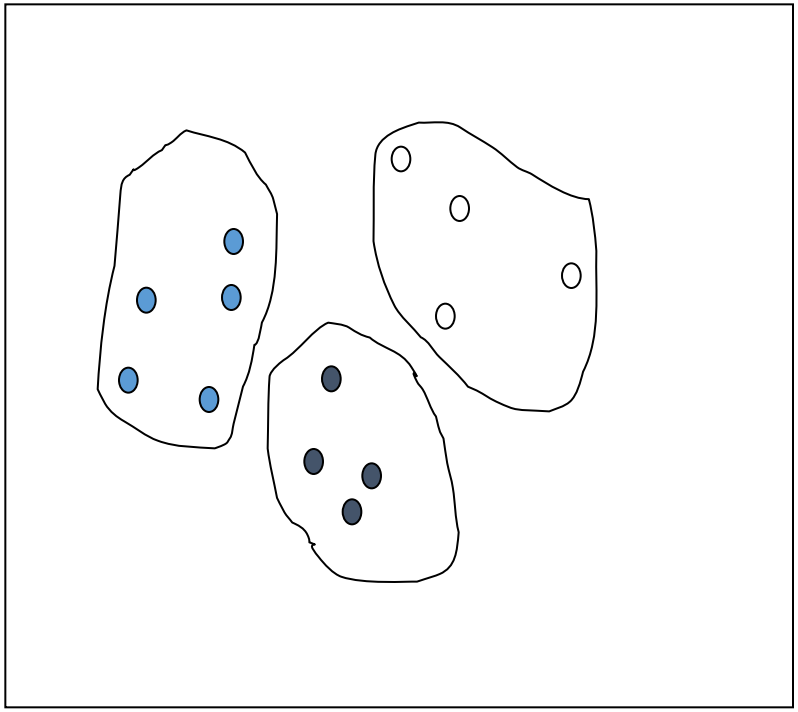
- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid and diameter) only
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms
- Cluster analysis will be studied in depth in Chapter 7

Sampling: Cluster or Stratified Sampling

Raw Data



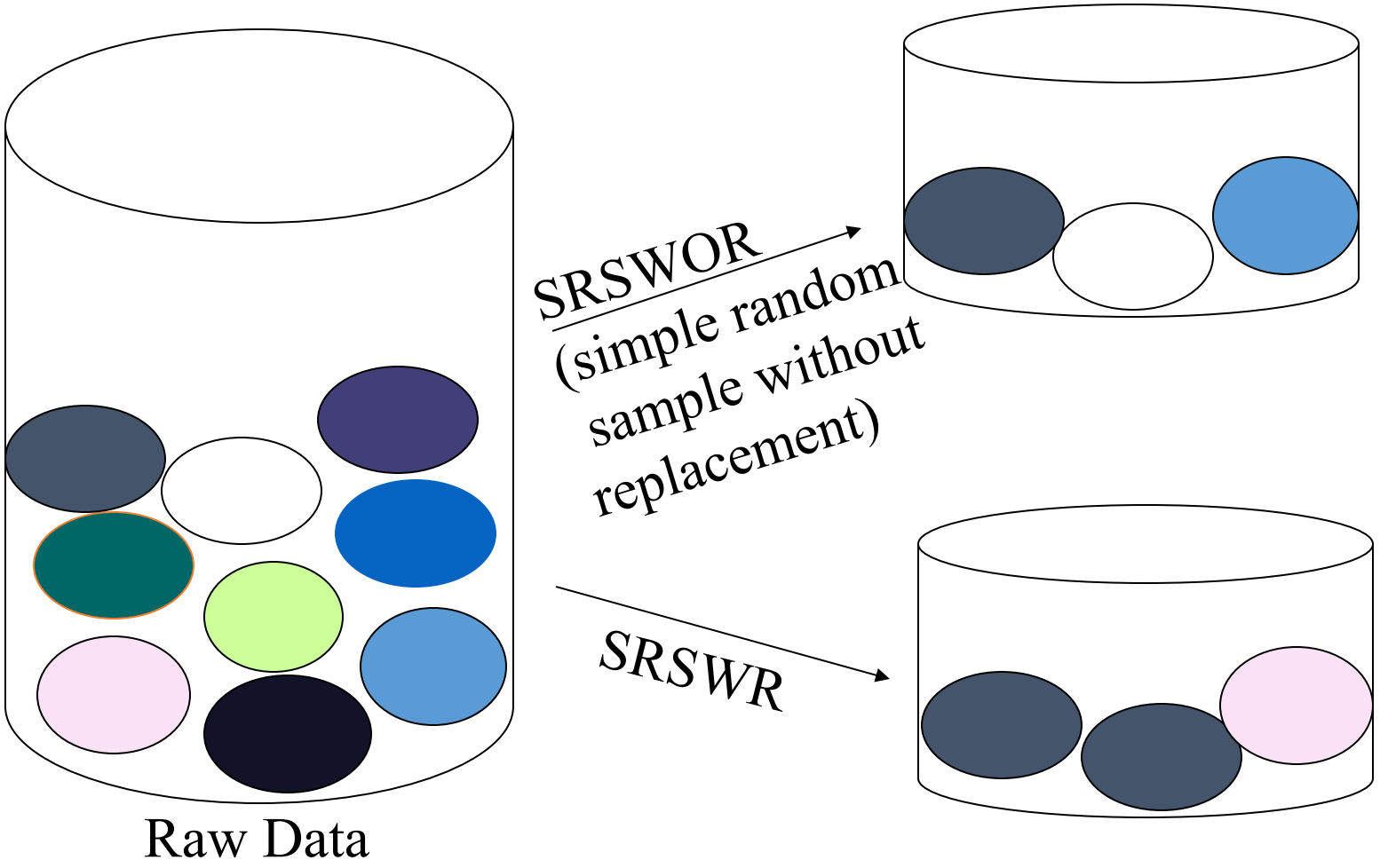
Cluster/Stratified Sample



Data Reduction Method (4): 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
- 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
- Note: Sampling may not reduce database I/Os (page at a time)

Sampling: with or without Replacement



Review questions

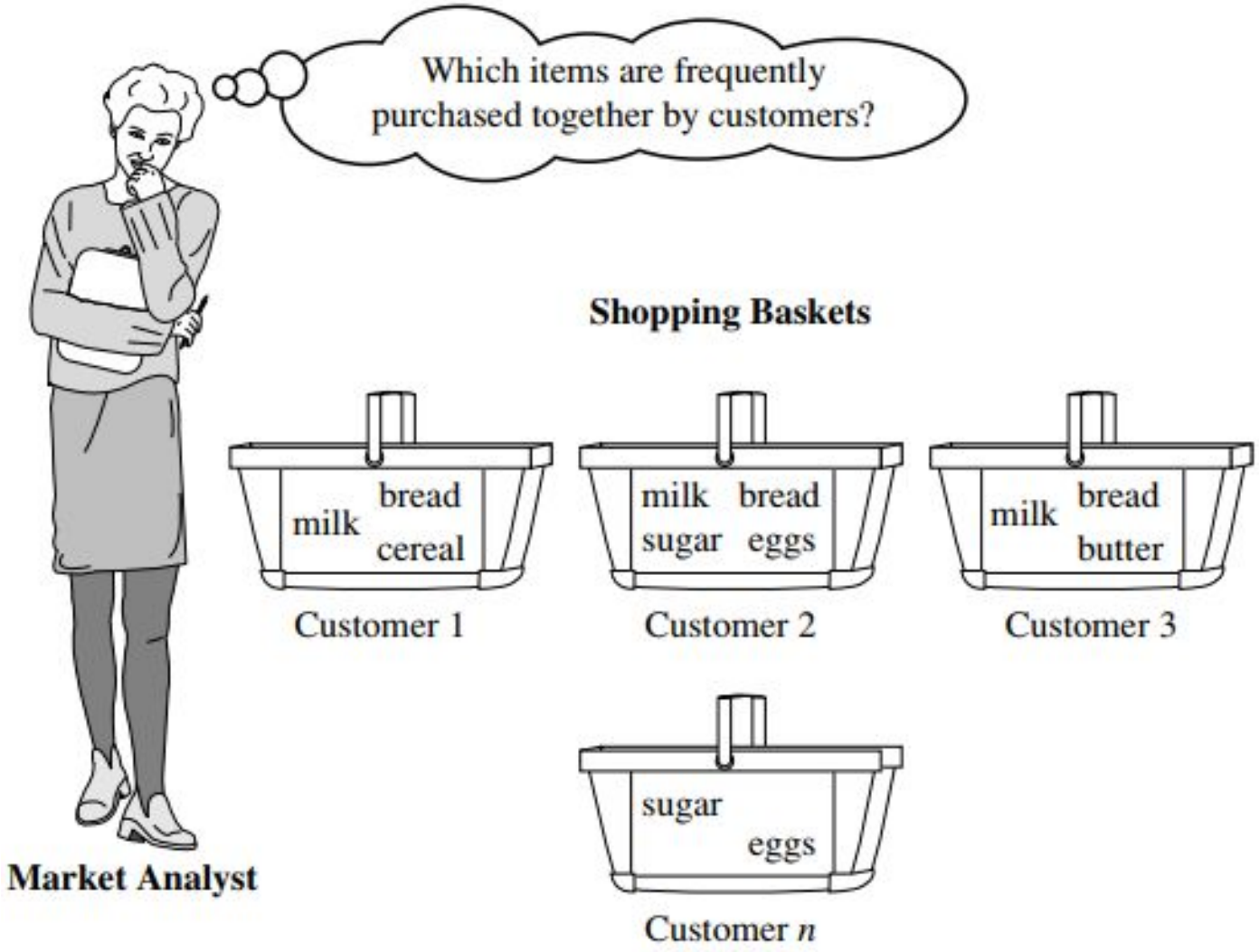
1. Describe the major tasks of data pre-processing
2. What is Noisy data? Describe the ways to handle noisy data
3. Define data transformation. Explain the methods for data transformation.
4. Why data needs to be normalized? Desribe the methods for data normalization.
5. "Suppose a group of 12 sales price records has been sorted as follows: 5; 10; 11; 13; 15; 35; 50; 55; 72; 92; 204; 215 Partition them into three bins by each of the following methods
   (a)equal-width partitioning

   (b) equal-frequency partitioning

6. There are two attributes: gender and preferred_reading. The observed frequency of each joint event is summarized in the contigency table shown in below table, where the numbers in parenthesis are the expected frequencies. Find the co-relation for the given attributes using chi-square test. (refer eg. From ppt)
7. Why data reduction is required? List down the techniques of data reduction

8. Elaborate the major tasks in data pre-processing
9. Illustrate different types of attributes with suitable example.
10. "For the given data {2,6,7,8,8,11,12,13,14,15,22,23} find the following:
    a) What is mean and median of the data?
    b) What is the mode of the data? Comment on data's modality.
    c) What is the midrange of the data?
    d) Find the Q1 and Q3 of the data
    e) Give the five-number summary of the data and also show a boxplot of the data.
11. Describe all the data transformation techniques in detail
12. Briefly outline how to compute the dissimilarity between objects described by the following: a) Nominal attributes b) Symmetric and Asymmetric binary attributes c) Ordinal attributes
13. Given two objects represented by the tuples (22,1,42,10) and (20,0,26,8):
    a) Compute the Euclidean distance between the two objects.
    B) Compute the Manhattan distance between the two objects.
    C) Compute the Minowski distance between the two objects, using q=3.

Chapter 03
Frequent Patterns Mining



Market Basket Analysis

Applications

- **Market Basket Analysis:** given a database of customer transactions, where each transaction is a set of items the goal is to find groups of items which are frequently purchased together.
- **Telecommunication** (each customer is a transaction containing the set of phone calls)
- **Credit Cards/ Banking Services** (each card/account is a transaction containing the set of customer's payments)
- **Medical Treatments** (each patient is represented as a transaction containing the ordered set of diseases)
- **Basketball-Game Analysis** (each game is represented as a transaction containing the ordered set of ball passes)

Market Basket Analysis

- Market Basket Analysis (Association Analysis) is a mathematical modeling technique based upon the theory that if you buy a certain group of items, you are likely to buy another group of items.

- It is used to analyze the customer purchasing behavior and helps in increasing the sales and maintain inventory by focusing on the point of sale transaction data.

- Given a dataset, the Apriori Algorithm trains and identifies product baskets and product association rules.

Association Rule Problem

- Given a database of transactions:

Transaction	Items
t_1	Bread,Jelly,PeanutButter
t_2	Bread,PeanutButter
t_3	Bread,Milk,PeanutButter
t_4	Beer,Bread
t_5	Beer,Milk

- Find all the association rules:

$X \Rightarrow Y$	s	α
Bread $\Rightarrow$ PeanutButter	60%	75%
PeanutButter $\Rightarrow$ Bread	60%	100%
Beer $\Rightarrow$ Bread	20%	50%
PeanutButter $\Rightarrow$ Jelly	20%	33.3%
Jelly $\Rightarrow$ PeanutButter	20%	100%
Jelly $\Rightarrow$ Milk	0%	0%

Association Rule Definitions

- $I=\{i_1, i_2, ..., i_n\}$: a set of all the items
- Transaction T: a set of items such that $T \subseteq I$
- Transaction Database D: a set of transactions
- A transaction $T \subseteq I$ contains a set $X \subseteq I$ of some items, if $X \subseteq T$
- ***An Association Rule***: is an implication of the form $X \Rightarrow Y$, where $X, Y \subseteq I$

Association Rule Definitions

Support:
This measurement technique measures how often multiple items are purchased and compared it to the overall dataset.

(Item A + Item B) / (Entire dataset)

Confidence:
This measurement technique measures how often item B is purchased when item A is purchased as well.

(Item A + Item B)/ (Item A)

Association Rule Definitions

Frequent pattern:
A pattern (a set of items, subsequences, substructures, etc.) that occurs frequently in a data set

- The **support** s of an itemset X is the percentage of transactions in the transaction database D that contain X.
- The support of the rule X ⇒ Y in the transaction database D is the support of the items set **X U Y** in D.
- The **confidence** of the rule X ⇒ Y in the transaction database D is the ratio of the number of transactions in D that contain **X U Y** to the number of transactions that contain **X** in D.

Association Rule Problem

- Given:
  - a set I of all the items;
  - a database D of transactions;
  - minimum support s;
  - minimum confidence c;
- Find:
  - all association rules $X \Rightarrow Y$ with a minimum support s and confidence c.

Problem Decomposition

Transaction ID	Items Bought
1	Shoes, Shirt, Jacket
2	Shoes,Jacket
3	Shoes, Jeans
4	Shirt, Sweatshirt

If the *minimum support* is 50%, then {Shoes,Jacket} is the only 2-itemset that satisfies the minimum support.

Frequent Itemset	Support
{Shoes}	75%
{Shirt}	50%
{Jacket}	50%
{Shoes, Jacket}	50%

If the *minimum confidence* is 50%, then the only two rules generated from this 2-itemset, that have confidence greater than 50%, are:

Shoes ⇒ Jacket Support=50%, Confidence= (2/3)=66%
Jacket ⇒ Shoes Support=50%, Confidence= (2/2)=100%

The Apriori Algorithm

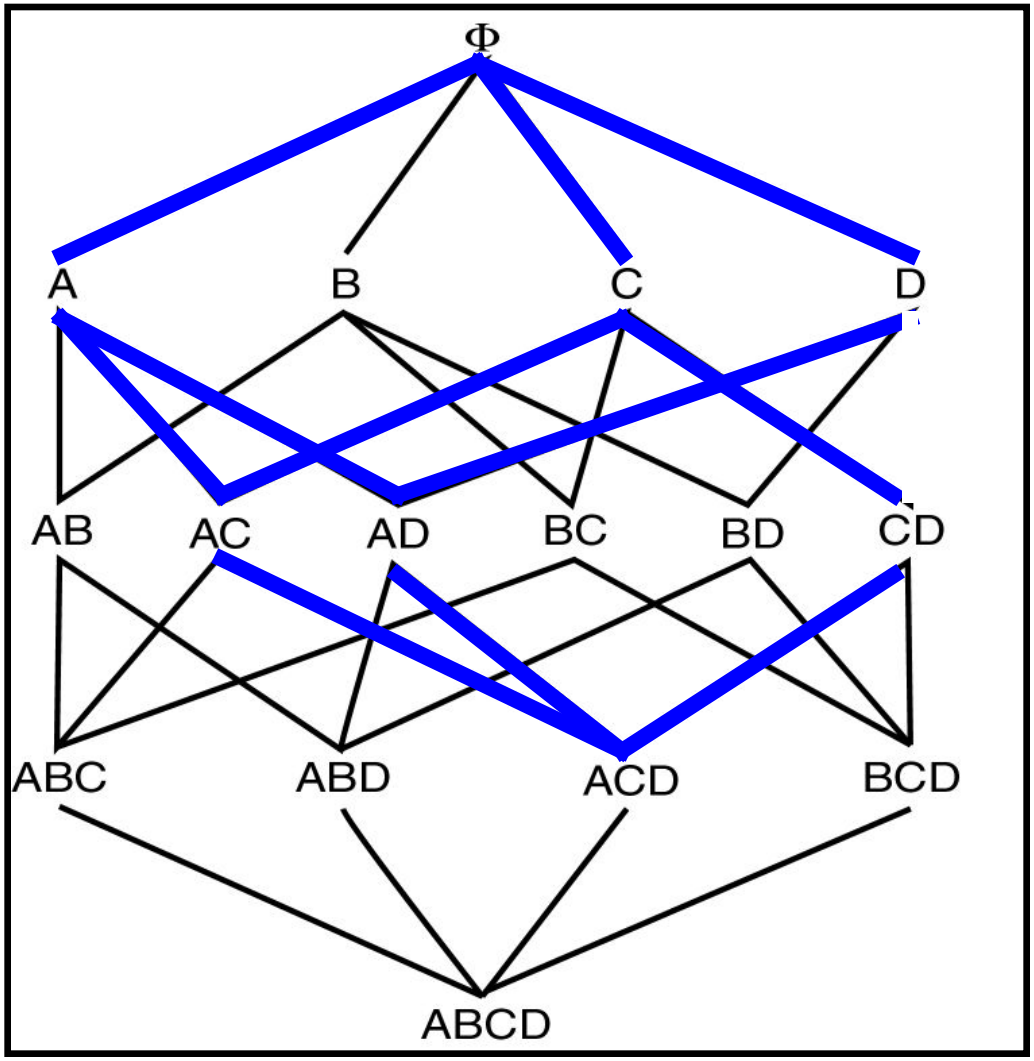
- ***Frequent Itemset Property:***

Any subset of a frequent itemset is frequent.

- ***Contrapositive:***

If an itemset is not frequent, none of its supersets are frequent.

Frequent Itemset Property



The Apriori Algorithm

- L_k: Set of frequent itemsets of size k (with min support)
- C_k: Set of candidate itemset of size k (potentially frequent itemsets)

L_1 = {frequent items};
for (k = 1; L_k !=∅; k++) **do**
    C_{k+1} = candidates generated from L_k;
    for each transaction t in database do
        increment the count of all candidates in C_{k+1} that are contained in t
    L_{k+1} = candidates in C_{k+1} with min_support
return $\cup_k L_k$;

The Apriori Algorithm — Example 1

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

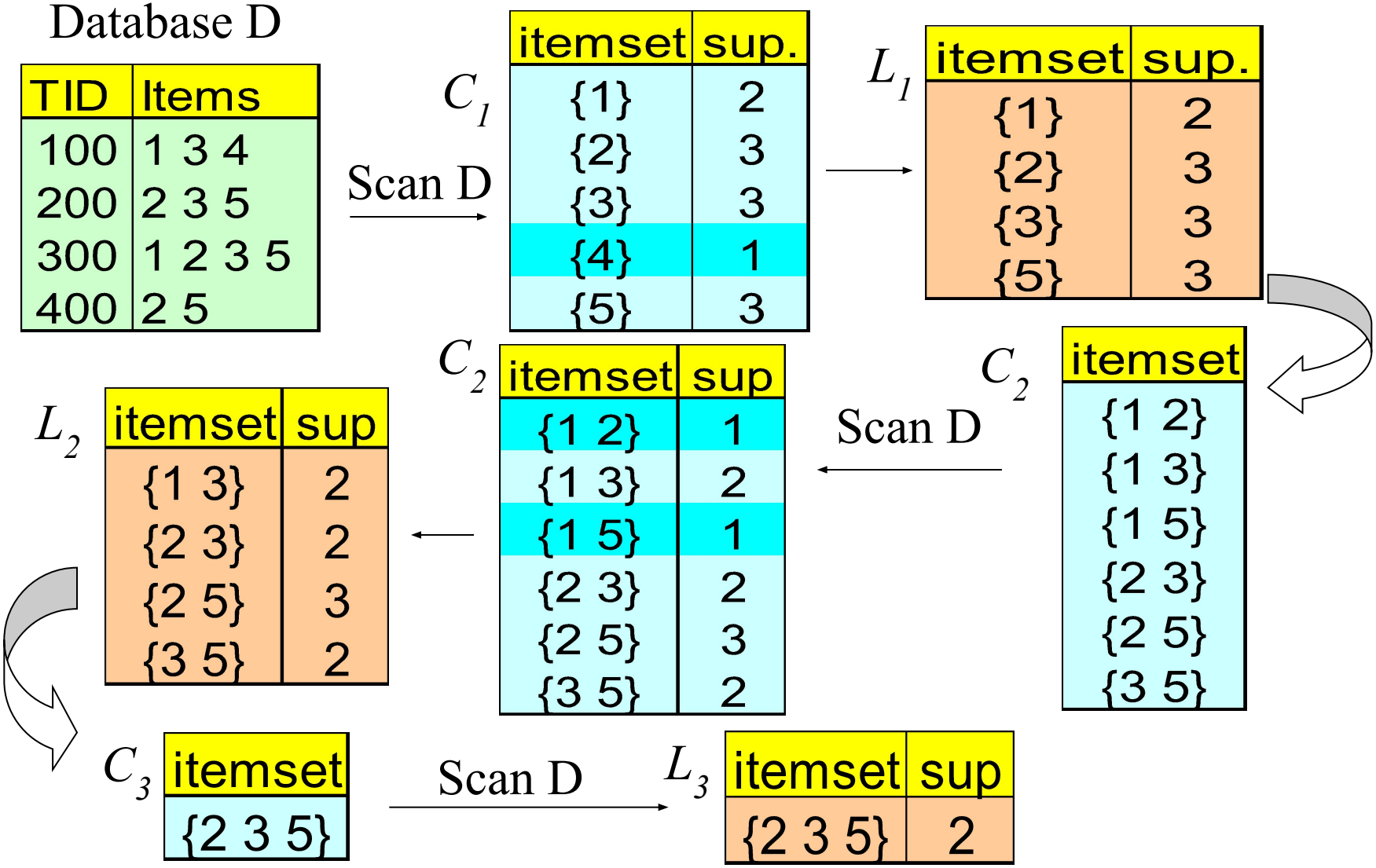
A dataset D has 4 transactions.

Let the minimum support be 50% and minimum confidence be 80%.

Find all the frequent item set and also generate association rule using Apriori algorithm

Min support count = min support threshold * total no. of transactions
= (50/100) * 4
= 2

Min support =50%



Therefore frequent item sets are

L= {2,3,5}

Now, for strong association rule:

Generate non-empty subset for L

S={2},{3},{5},{2,3},{2,5},{3,5}

Find S□(L-S)

For example: {2}□{3,5}

S□(L-S)	Confidence	Confidence(%)
{2}□{3,5}	2/3	66.66
{3}□{2,5}	2/3	66.66
{5}□{2,3}	2/3	66.66
{2,3}□{5}	2/2	100
{2,5}□{3}	2/3	66.66
{3,5}□{2}	2/2	100

Minimum confidence threshold=80%

Therefore strong association rules are_

{2,3}□{5}

{3,5}□{2}

Apriori Example 2

- Consider the following database with minimum support count=60%.Find all the frequent itemset using apriori algorithm and also generate strong association rules if minimum confidences= 50%.

Transaction ID	Items Bought
T1	{M, O, N, K, E, Y}
T2	{D, O, N, K, E, Y}
T3	{M, A, K, E}
T4	{M, U, C, K, Y}
T5	{C, O, O, K, I, E}

Hint : O is bought 4 times in total, but, it occurs in just 3 transactions.

Min support count = min support threshold * total no. of transactions

Min support count = (60/100) * 5

Min support count = 3

Step 1: Generate C1

Item	No of transactions
M	3
O	3
N	2
K	5
E	4
Y	3
D	1
A	1
U	1
C	2
I	1

Step 2: Generate L1 from C1

Item	Number of transactions
M	3
O	3
K	5
E	4
Y	3

Step 3: Generate C2 from L1 by Join Step

Item Pairs	Number of transactions
MO	1
MK	3
ME	2
MY	2
OK	3
OE	3
OY	2
KE	4
KY	3
EY	2

Step 4: Generate L2 from C2 by prune Step

Item Pairs	Number of transactions
MK	3
OK	3
OE	3
KE	4
KY	3

Step 5: Generate C3 from L2 by Join step

Item Set	Number of transactions
OKE	3
KEY	2

Step 6: Generate L3 from C3 by Prune step

Item Set	Number of transactions
OKE	3

For strong association rule_

L = { O, K, E}

Generate non empty subset of L

S = {O}, {K}, {E}, {OK}, {OE}, {KE}

Generate association rule S□(L-S)

Step 7: Finding association Rules with min confidences

Association Rule	Support	Confidence	Confidence(100 %)
O , K ⇒ E	**3**	3/3	100%
O , E ⇒ K	3	3/3	100%
K , E ⇒ O	3	3/4	75%
O ⇒ K , E	3	3/3	100%
K ⇒ O , E	3	3/5	60%
E ⇒ O , K	3	3/4	75%

All the association rules are having confidence more than 50%.

Therefore all rules are strong association rule

Apriori Example 3

- Consider the following database with minimum support count=50%.Find all the frequent itemset using apriori algorithm and also generate strong association rules if minimum confidences= 50%.

T id	Items Bought
1	A,B,D
2	A,D
3	A,C
4	B,D,E,F

Apriori Example 4

Tid	items
1	A B D
2	B C D
3	A B
4	B D
5	A B C

Find frequent item set and strong association rule

- min support = 30%
- min confidence = 75%

How to Generate Candidates

Input: L_{i-1} : set of frequent itemsets of size *i-1*

Output: C_i : set of candidate itemsets of size *i*

C_i = *empty set;*

for each itemset *J* in L_{i-1} **do**

  for each itemset *K* in L_{i-1} s.t. *K*<> *J* do

    if *i-2* of the elements in *J* and *K* are equal **then**

      if all subsets of $\{K \cup J\}$ are in L_{i-1} **then**

        $C_i = C_i \cup \{K \cup J\}$

return C_i;

Example of Generating Candidates

- L_3={*abc, abd, acd, ace, bcd*}
- Generating C_4 from L_3
  - *abcd* from *abc* and *abd*
  - *acde* from *acd* and *ace*
- Pruning:
  - *acde* is removed because *ade* is not in L_3
- C_4={*abcd*}

Example of Discovering Rules

Let use consider the 3-itemset {I1, I2, I5}:

$I1 \wedge I2 \Rightarrow I5$

$I1 \wedge I5 \Rightarrow I2$

$I2 \wedge I5 \Rightarrow I1$

$I1 \Rightarrow I2 \wedge I5$

$I2 \Rightarrow I1 \wedge I5$

$I5 \Rightarrow I1 \wedge I2$

Discovering Rules

for each frequent itemset I **do**

  for each subset C of I **do**

    if (support(I) / support(I - C) >= minconf) **then**

      output the rule (I - C) $\Rightarrow$ C,

      with confidence = support(I) / support (I - C)

        and support = support(I)

Example of Discovering Rules

TID	List of Item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Let use consider the 3-itemset {I1, I2, I5} with support of 0.22(2)%. Let generate all the association rules from this itemset:

$I1 \wedge I2 \Rightarrow I5$ *confidence*= 2/4 = 50%

$I1 \wedge I5 \Rightarrow I2$ *confidence*= 2/2 = 100%

$I2 \wedge I5 \Rightarrow I1$ *confidence*= 2/2 = 100%

$I1 \Rightarrow I2 \wedge I5$ *confidence*= 2/6 = 33%

$I2 \Rightarrow I1 \wedge I5$ *confidence*= 2/7 = 29%

$I5 \Rightarrow I1 \wedge I2$ *confidence*= 2/2 = 100%

Frequent Itemset with support count 2 is {I1,I2,I3} and {I1,I2,I5}

Association rule	support	confidence	Confidence %
1,5□2	2	2/2	100
2,5□1	2	2/2	100
5□1,2	2	2/2	100

Apriori Advantages/Disadvantages

- ***Advantages:***
  - Uses large itemset property.
  - Easily parallelized
  - Easy to implement.
- ***Disadvantages:***
  - Assumes transaction database is memory resident.
  - Requires many database scans.

What is FP Growth?

- FP Growth Stands for frequent pattern growth
- It is a scalable technique for mining frequent pattern in a database

FP Growth

- FP growth improves Apriority to a big extent
- Frequent Item set Mining is possible without candidate generation
- Only “two scan” to the database is needed

BUT HOW?

FP Growth

- Simply a two step procedure
  - Step 1: Build a compact data structure called the FP-tree
    - Built using 2 passes over the data-set.
  - Step 2: Extracts frequent item sets directly from the FP-tree

FP Growth

- Now Lets Consider the following transaction table

TID	List of Item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Note:

Assume min support = 2

FP Growth

- Now we will build a FP tree of that database
- Item sets are considered in order of their descending value of support count.

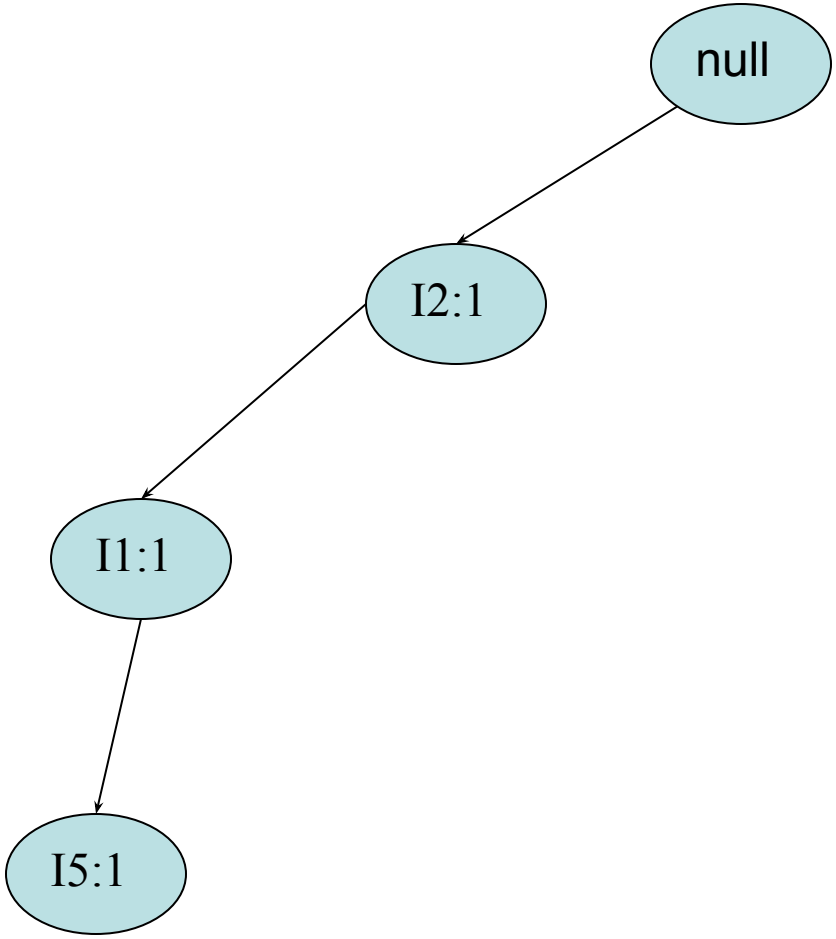
FP Growth

Items	Support count
I1	6
I2	7
I3	6
I4	2
I5	2

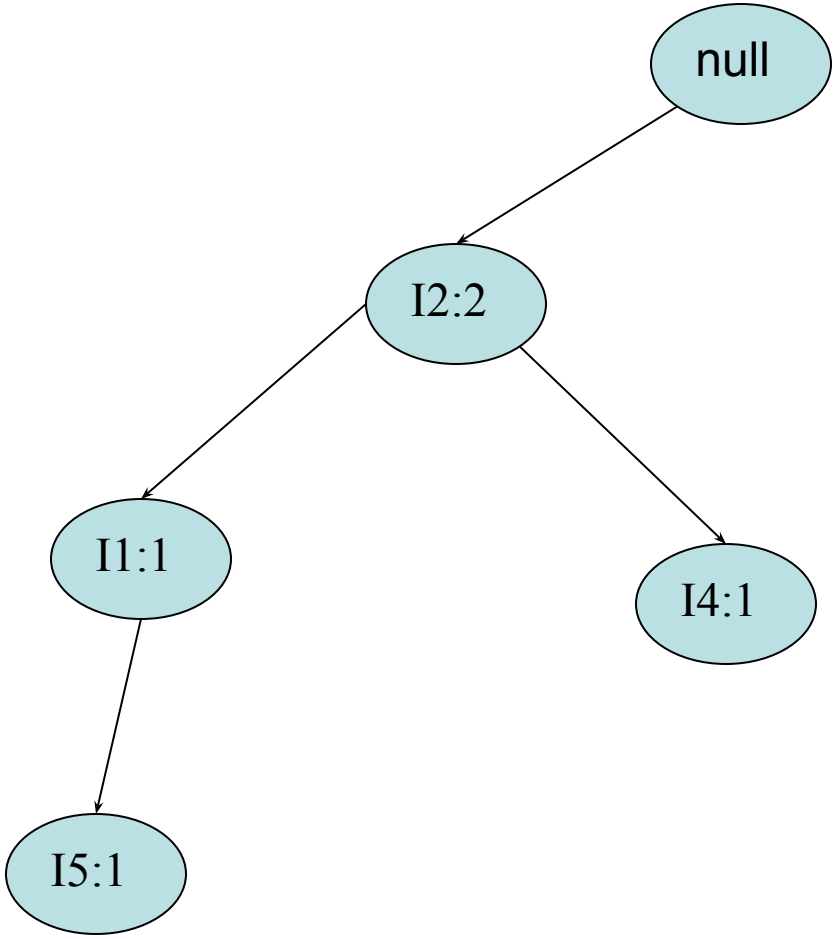
Items	Support count
I2	7
I1	6
I3	6
I4	2
I5	2

TID	List of Item_IDs
T100	I2, I1, I5
T200	I2, I4
T300	I2, I3
T400	I2, I1, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I2, I1, I3, I5
T900	I2, I1, I3

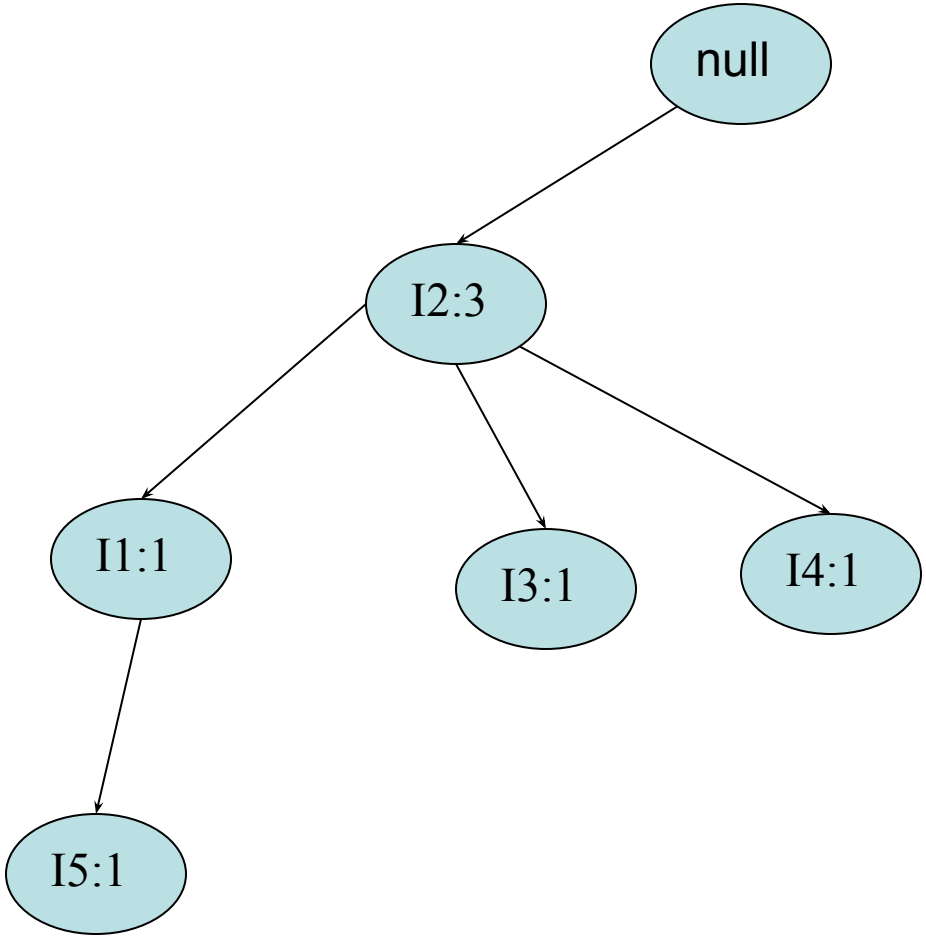
For Transaction:
I2,I1,I5



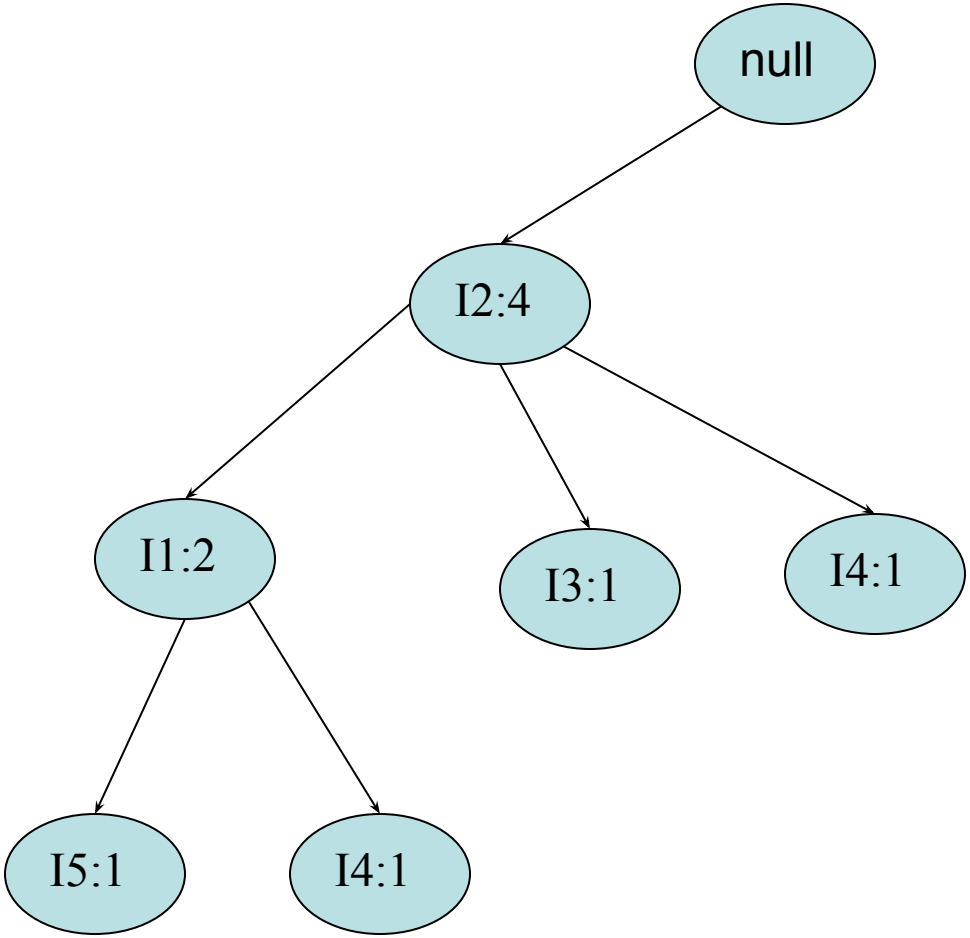
For Transaction:
I2,I4



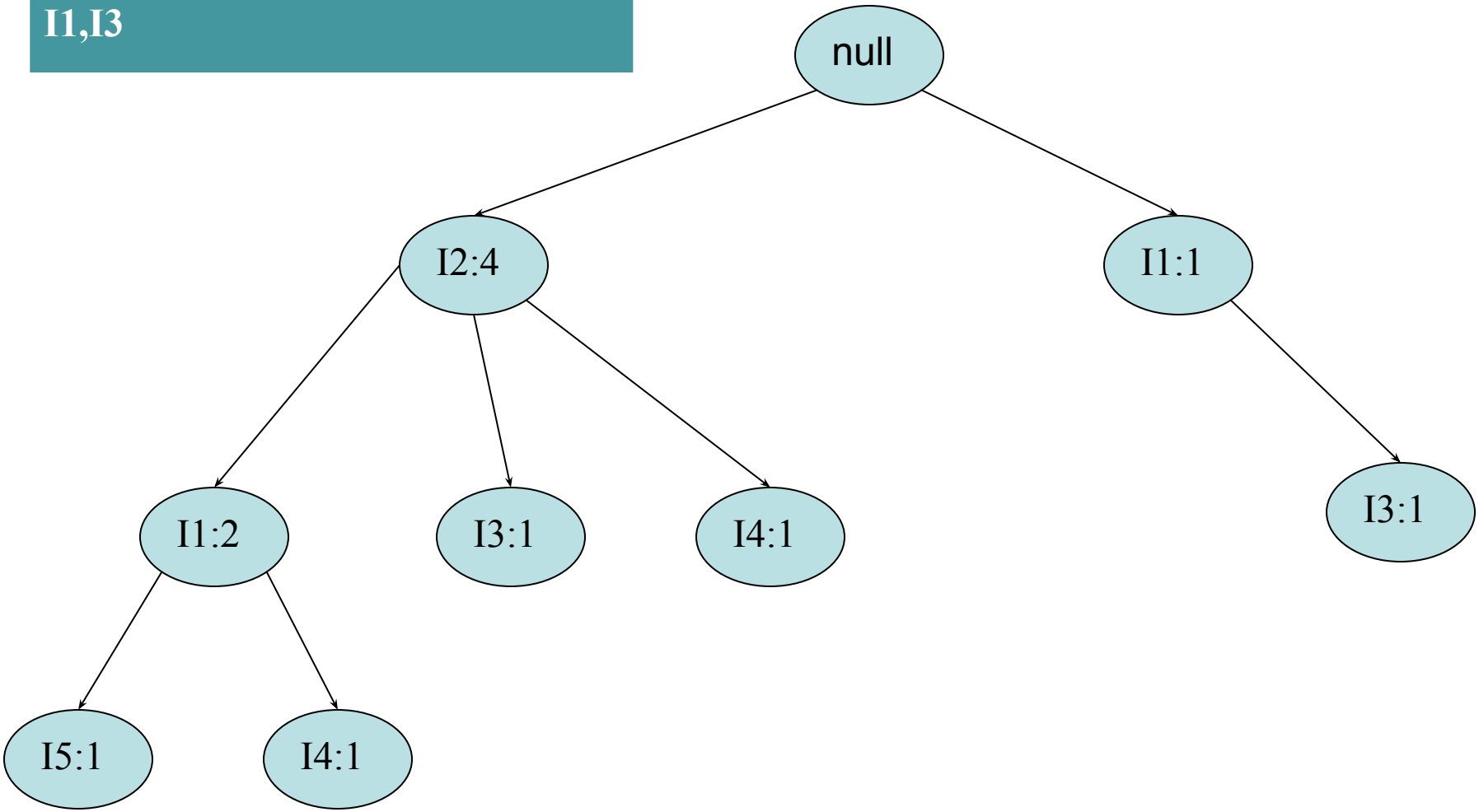
For Transaction:
I2,I3



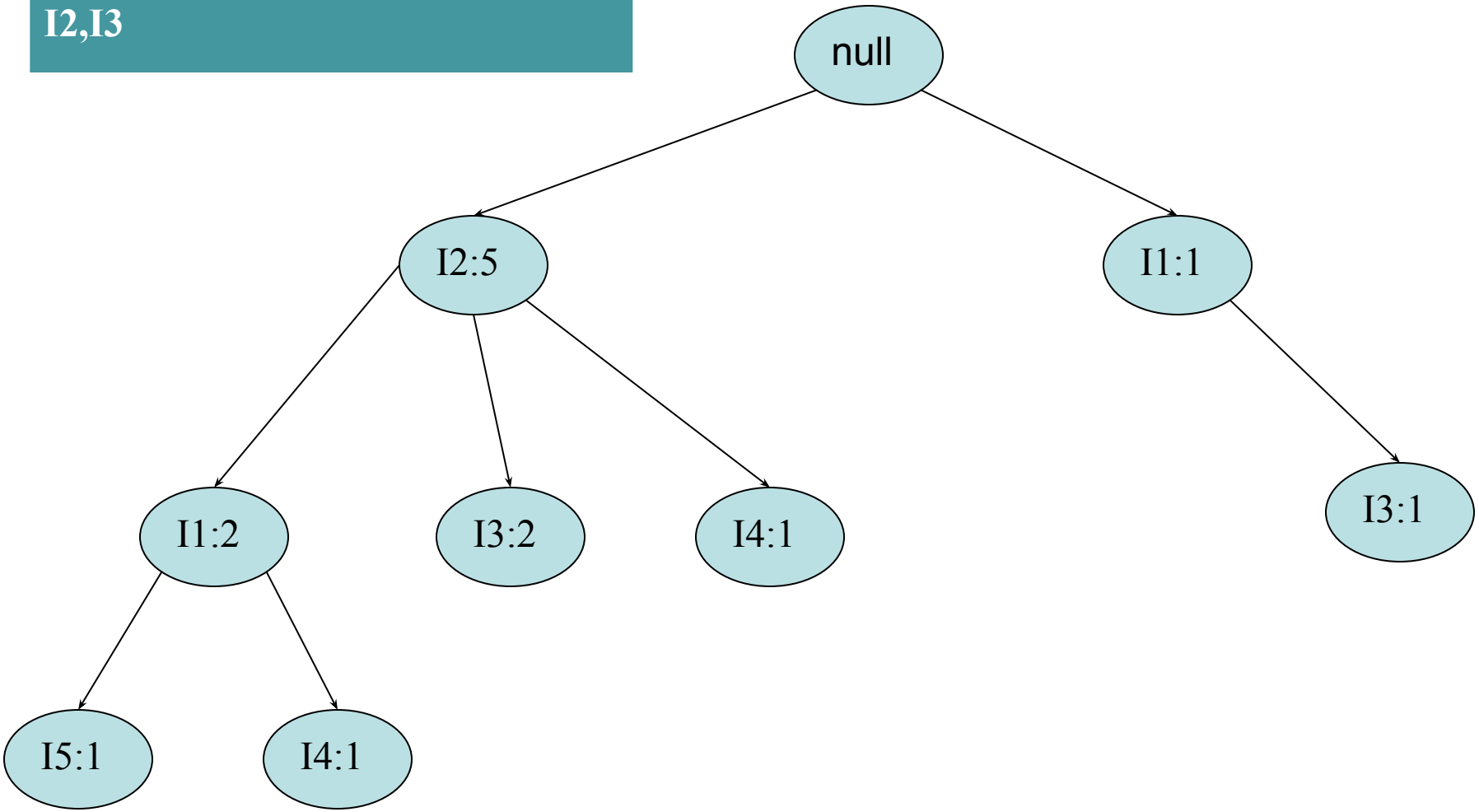
For Transaction:
I2,I1,I4



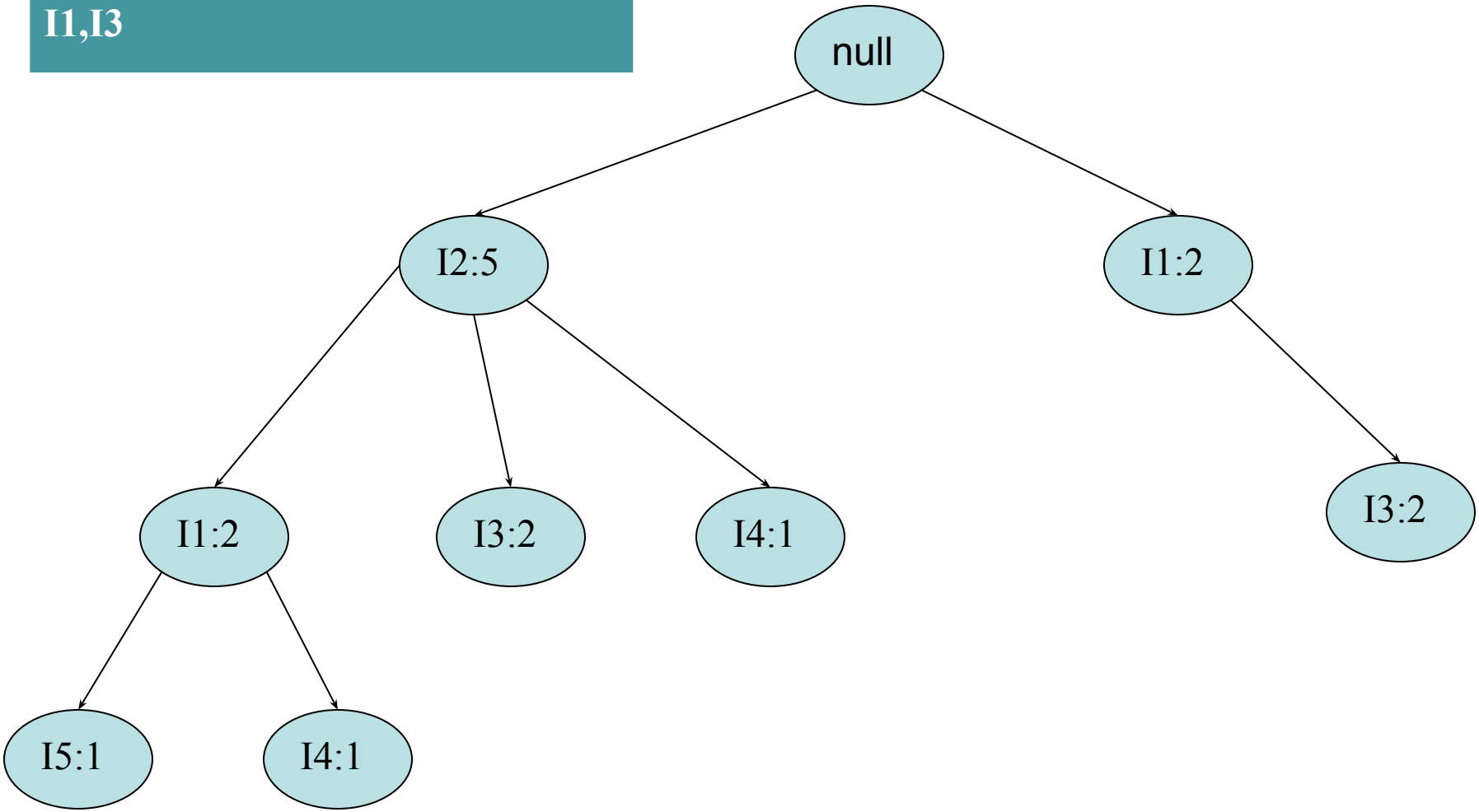
For Transaction:
I1,I3



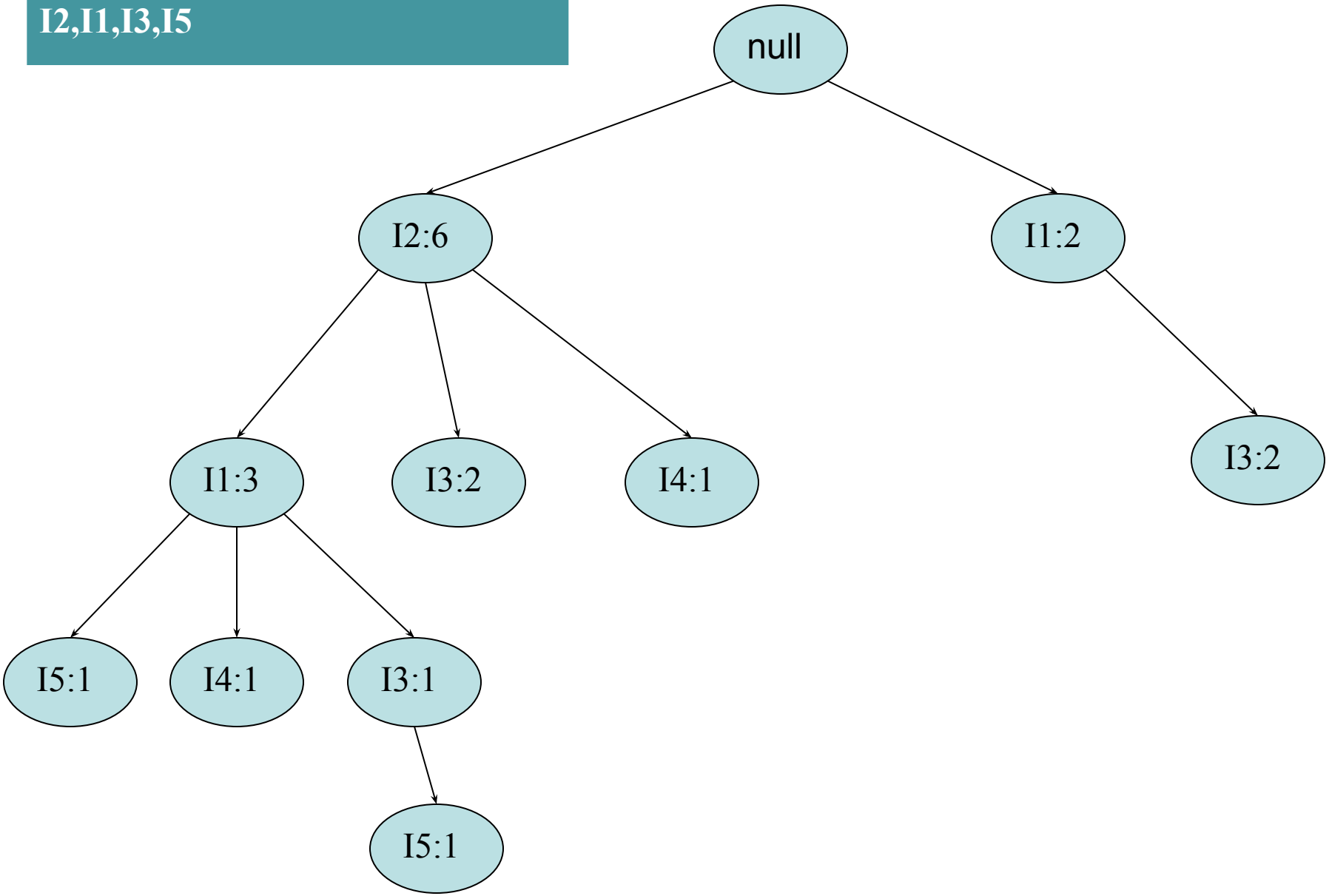
For Transaction:
I2,I3



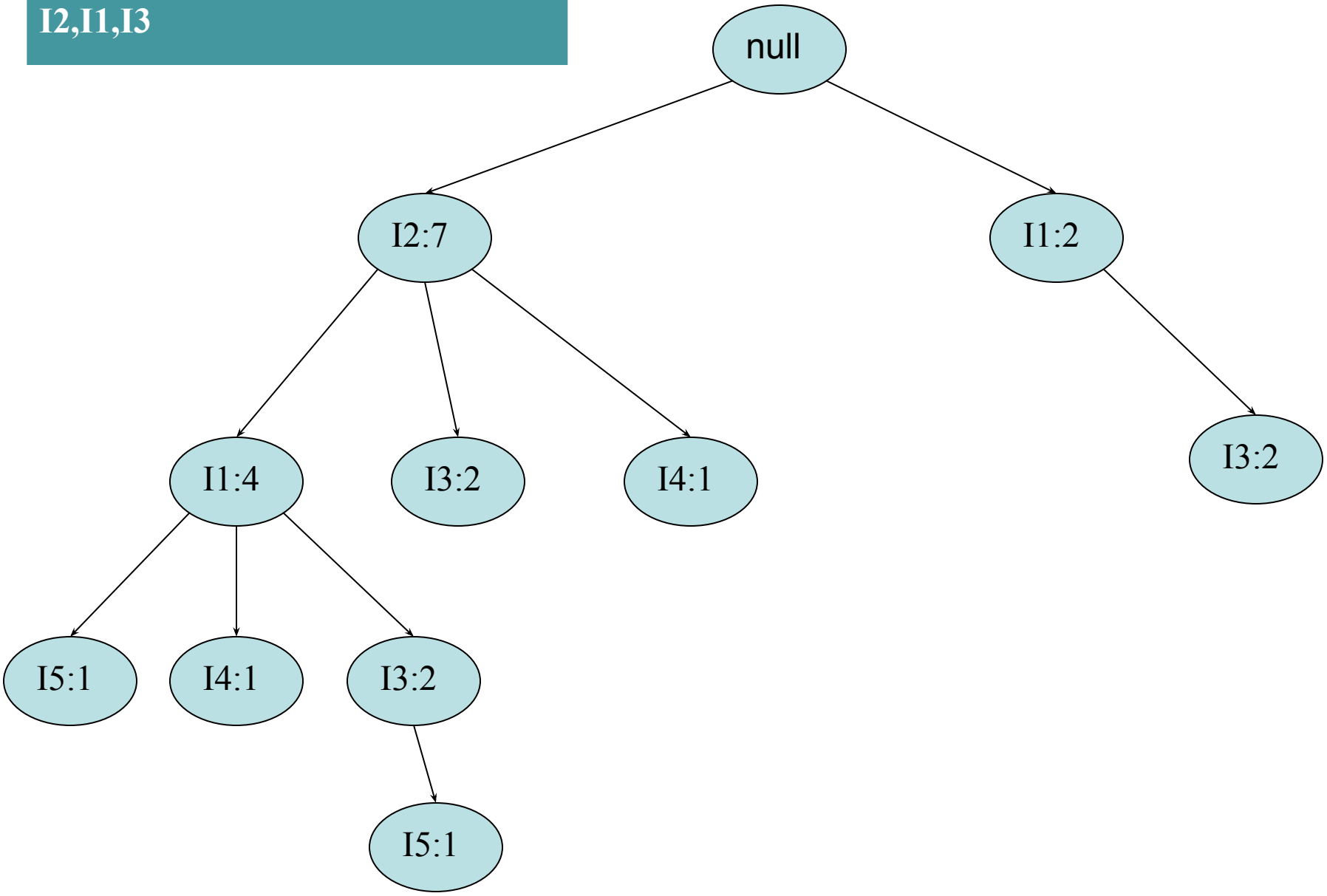
For Transaction:
I1,I3



For Transaction:
I2,I1,I3,I5

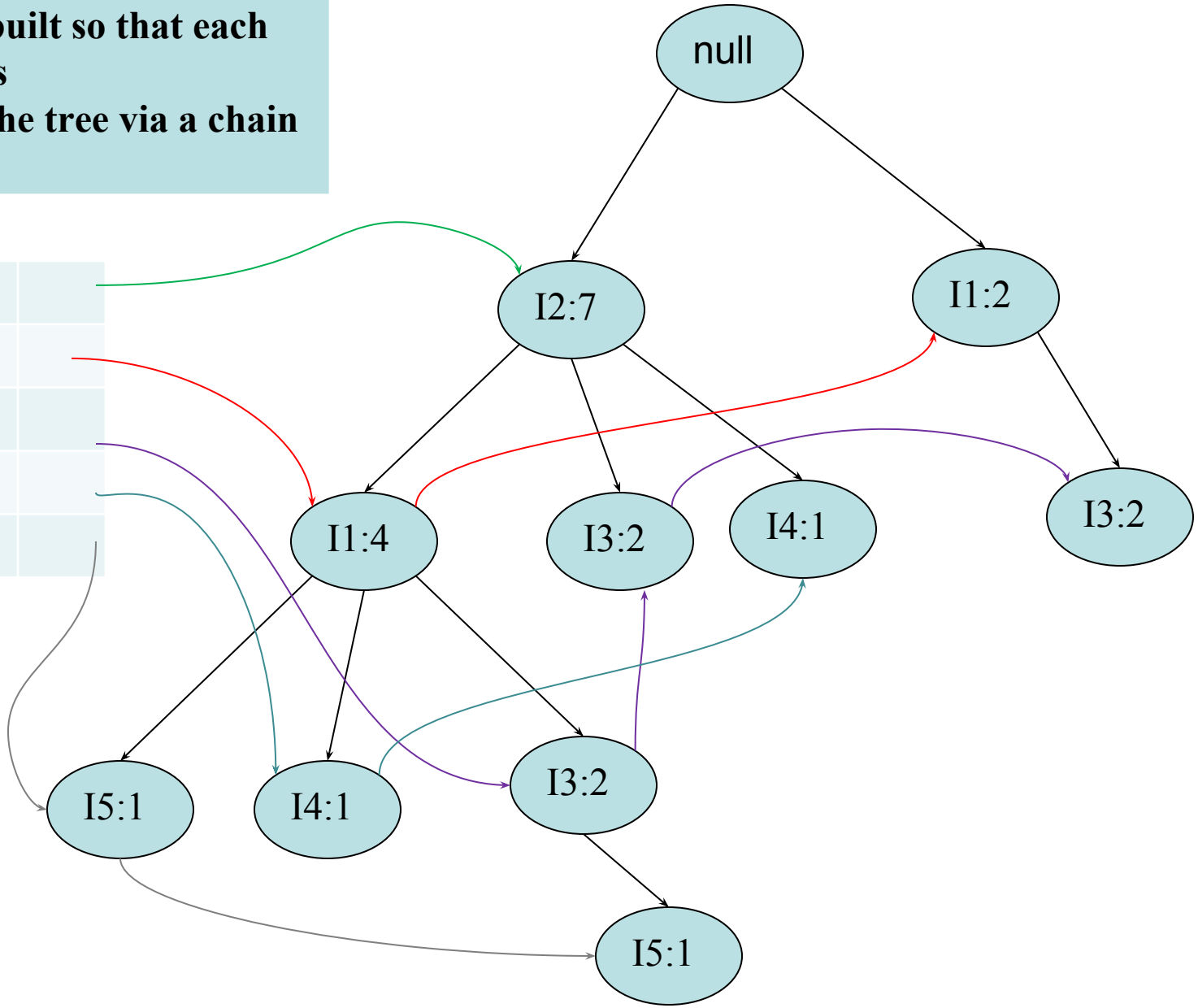


For Transaction:
I2,I1,I3



To facilitate tree traversal, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links.

I2	7	
I1	6	
I3	6	
I4	2	
I5	2	



FP Growth

- FP Tree Construction Over!!
  Now we need to find conditional pattern base and Conditional FP Tree for each item

Frequent Pattern Generated

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Pattern Generated
I5	{I2,I1 : 1},{I2,I1,I3 : 1}	{I2:2,I1:2}	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{I2,I1:1},{I2:1}	{I2:2}	{I2, I4: 2}
I3	{I2,I1:2},{I2:2},{I1:2}	{I2:4},{I1:2},{I1:2}	{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}
I1	{I2:4}	{I2:4}	{I2, I1: 4}
I2		Ignore as no Branch	

Conditional FP TREE: satisfying minimum support

Frequent Pattern Generated: RULES

Example 2: FP Growth

- Draw FP tree for the transaction items given below. Min. support=02

TId	Items
T1	b,e
T2	a,b,c,e
T3	b,c,e
T4	a,c
T5	a

Vertical Data formats to find frequent item sets

TID	List of Item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Table 6.3 The Vertical Data Format of the Transaction Data Set D of Table 6.1

itemset	TID_set
I1	{T100, T400, T500, T700, T800, T900}
I2	{T100, T200, T300, T400, T600, T800, T900}
I3	{T300, T500, T600, T700, T800, T900}
I4	{T200, T400}
I5	{T100, T800}

Table 6.4 2-Itemsets in Vertical Data Format

itemset	TID_set
{I1, I2}	{T100, T400, T800, T900}
{I1, I3}	{T500, T700, T800, T900}
{I1, I4}	{T400}
{I1, I5}	{T100, T800}
{I2, I3}	{T300, T600, T800, T900}
{I2, I4}	{T200, T400}
{I2, I5}	{T100, T800}
{I3, I5}	{T800}

Table 6.5 3-Itemsets in Vertical Data Format

itemset	TID_set
{I1, I2, I3}	{T800, T900}
{I1, I2, I5}	{T100, T800}