CHP - 02
DATA EXPLORATION

TOPICS

- Data objects and attribute types
- Basic statistical descriptions of data
- Data visualization
- Measuring data similarity and dissimilarity

KNOWING YOUR DATA

- What are the types of attributes or fields that make up your data?
- What kind of values does each attribute have?
- Which attributes are discrete, and which are continuous-valued?
- How are the values distributed?
- Are there ways we can visualize the data to get a better sense of it all?
- Can we spot any outliers?
- Can we measure the similarity of some data objects with respect to others?

TYPES OF DATA SETS

- Record
  - Relational records
  - Data matrix, e.g., numerical matrix
  - Document data, text documents
  - Transaction data
- Graph and network
  - World Wide Web
  - Social or information networks
- Ordered
  - Video data: sequence of images
  - Temporal data: time-series
  - Sequential Data: transaction sequences
  - Genetic sequence data
- Spatial, image and multimedia:
  - Spatial data: maps
  - Image data

DATA OBJECTS

- Data set (file) is a collection of data objects
- **Data object (records or rows)** represents an **entity**. Eg: In a **university** database, the objects may be **students**, **professors**, and **courses**
- Data **objects** are typically **described** by **attributes**
- Data objects can also be referred to as samples, examples, instances, data points, or objects
- If the data **objects** are stored in a **database**, they are **data tuples**. That is, the **rows** of a database correspond to the data **objects**, and the **columns** correspond to the **attributes**

ATTRIBUTE

- It is a **data field**, representing a **characteristic** or **feature** of a data **object** that **may vary** from one **object** to **another** or from one **time** to **another**
- Also called as dimension, feature, and variable
- Eg: Attributes describing a customer object **-** customer ID, name, and address
- **Observed values** for a given attribute are known as **observations**
- **Set of attributes** used to **describe** a given **object** is called an **attribute vector** or **feature vector**

ATTRIBUTE - UNIVARIATE

Super hero
Batman
Ironman
Superman
Captain America

- Distribution of data involving one attribute (or variable) is called **Univariate**

- In this case, data has only one variable

- Major purpose is to describe data - it takes data, summarizes that data and finds patterns in the data

Batman	Captain America	Iron Man	Superman
52	25	34	17

- Data is described using Frequency Distribution table, Histogram, Bar charts, Pie charts etc.

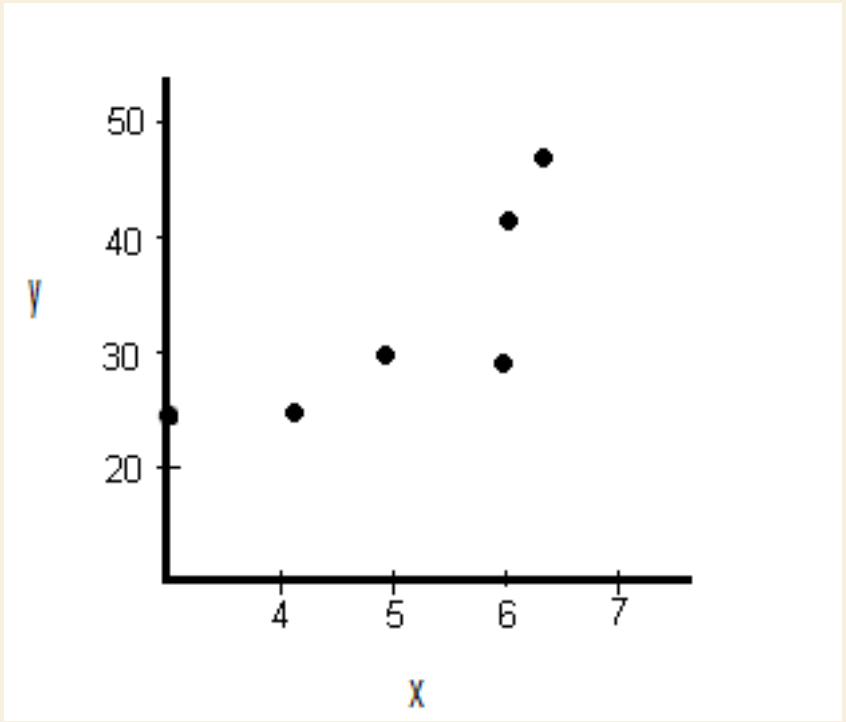
ATTRIBUTE - BIVARIATE

TEMPERATURE(IN CELSIUS)	ICE CREAM SALES
20	2000
25	2500
35	5000
43	7800

- **Bivariate** distribution involves **two** attributes
- Eg: Ice cream sales compared to temperature that day
- (X,Y)=(20,2000),(25,2500),(35,5000),(43,7800)
- Used to **find** out if there is a **relationship** between **two sets** of **values**
- **Temperature** would be the **independent** variable, and **ice-cream** sales would be the **dependent** variable

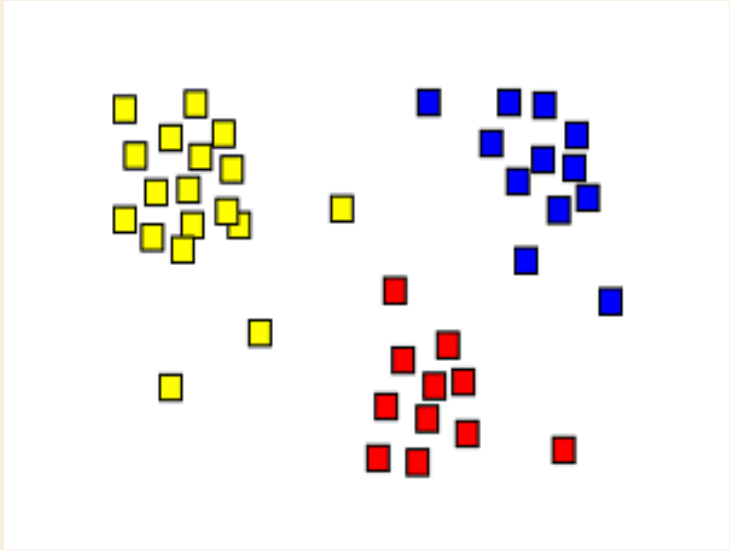
ATTRIBUTE - BIVARIATE

- Common types of bivariate analysis include:
  - Scatter plots
  - Correlation coefficient
  - Regression analysis



ATTRIBUTE - MULTIVARIATE

- Used to study more complex sets of data
- **Multivariate** distribution involves **more than two** attributes
- Common types of multivariate analysis include:
  - Cluster analysis



- **Type** of an attribute is determined by the set of possible values — nominal, binary, ordinal, or numeric — the attribute can have

NOMINAL ATTRIBUTES

- Values of a **nominal attribute** are **symbols** or **names** of things in **alphabetical** form and **not** in **integer**
- Allows only **qualitative classification** (**no quantitative** values)
- **Possible** to **represent** such **symbols** or "**names**" with **numbers** but mathematical **operations cannot** be **performed** on nominal attributes. Eg: one customer_id cannot be subtracted from another
- Each value represents some kind of category, code, or state, and so nominal attributes are also referred to as **categorical**
- Values **do not** have any **meaningful order**

EXAMPLE OF NOMINAL ATTRIBUTES

- Suppose that customer_id, hair_color and occupation are three attributes describing person objects
- customer_id = {E1234, E3427, E2780}
- hair_color = {black, brown, blond, red, auburn, gray, white}
- occupation= {engineer, teacher, dentist, programmer, farmer}
- All of the above attributes are nominal attributes

NOMINAL ATTRIBUTES

- For *hair_color*, we can also assign a code of 0 for *black*, 1 for *brown*, and so on

- Even though a **nominal** attribute **may** have **integers** as **values**, it is **not considered** a **numeric** attribute because the integers are **not meant** to be used **quantitatively**

- Mathematical **operations** on values of nominal attributes are **not meaningful**

EXAMPLE OF NOMINAL ATTRIBUTES

- What is your hair color?

1 – brown

2 – black

3 – bonde

4 – gray

5 – other

- Where do you live?

A – north of equator

B – south of equator

C – Neither, in the international space station

BINARY ATTRIBUTES

- **Binary attribute** is a **nominal** attribute with only **two** categories or states: **0** or **1**, where
  - **0** → means that an **attribute** is **absent**
  - **1** → means that an **attribute** is **present**

- Binary attributes are referred to as **Boolean** if the two states correspond to ***true*** and ***false***

EXAMPLE OF BINARY ATTRIBUTES

- Given the attribute smoker describing a patient object,
  - 1 → indicates that the patient smokes
  - 0 → indicates that the patient does not smoke

- Similarly, exam result of student has two possible outcomes

- The attribute **result** is binary, where
  - 1 → means the student has passed
  - 0 → means the student has failed

BINARY ATTRIBUTES - SYMMETRIC

- Binary attribute is **symmetric** if **both** of its **states** are **equally valuable** and **carry** the **same weight**; i.e., there is **no preference** on which **outcome** should be **coded** as **0** or **1**

- Eg: attribute *gender* having the states *male* and *female*

- Here, both are equally valuable and difficult to represent in terms of 0 or 1

BINARY ATTRIBUTES - ASYMMETRIC

- Binary attribute is **asymmetric** if the outcomes of the states are **not equally important**, such as the ***positive*** and ***negative*** outcomes of a medical test for **HIV**

- By convention, we code the most important outcome, which is usually the rarest one, by **1** (e.g., ***HIV positive***) and the other by **0** (e.g., ***HIV negative***)

ORDINAL ATTRIBUTES

- An **attribute** with possible **values** that have a **meaningful order** or ***ranking*** among them, but the **difference** between **each** one is **not** really **known**

- Subjective assessment of qualities that cannot be measured objectively

- Eg: How do you feel today?

1 **-** *very unhappy*

2 – unhappy

3 – OK

4 – happy

5 – very happy

ORDINAL ATTRIBUTES

- **Values** have a **meaningful sequence** (which corresponds to increasing amount of happiness)
- However, we cannot tell from the values what or how much is the difference between happy and very happy
- Other example of ordinal attributes include *grade* (e.g., ***A+, A, A-, B+,B-***)
- **Customer satisfaction** had the following ordinal categories: **0: very dissatisfied, 1: somewhat dissatisfied, 2: neutral, 3: satisfied, and 4: very satisfied**

ATTRIBUTES

- **Nominal**, **binary**, and **ordinal** attributes are **qualitative**
- They **describe** a **feature** of an **object without** giving an actual **size** or **quantity**
- **Values** of such **qualitative** attributes are typically **words** representing **categories**
- If **integers** are used, they **represent computer codes** for the **categories**, as opposed to **measurable quantities** (e.g., **0 for small drink size, 1 for medium, and 2 for large**)

NUMERIC ATTRIBUTES

- It is ***quantitative*** i.e. it is a **measurable quantity**, represented in **integer** or **real** values

- Numeric attributes can be ***interval-scaled*** or ***ratio-scaled***

INTERVAL SCALED ATTRIBUTES

- **Continuous** measurements on a **linear** scale i.e. **measured** on a **scale** of **equal-size** units

- **Numeric** scales in which we know not only the **order** but also the **exact differences** between the **values**

- Values of **interval-scaled** attributes have **order** and can be **positive**, **0**, or **negative**

- In addition to providing a **ranking/ordering** of values, such attributes allow us to **compare** and **quantify** the ***difference*** between **values** (value between each item)

INTERVAL SCALED ATTRIBUTES - EXAMPLE

- A ***Celsius temperature*** attribute is **interval-scaled** because the **difference** between **each value** is the **same**
- Suppose that we have the outdoor *temperature* value for a number of different days, where each day is an object
- By ordering the values, we obtain a ranking of the objects with respect to *temperature*

INTERVAL SCALED ATTRIBUTES - EXAMPLE

- For example, a temperature of **20^0C** is **five degrees higher** than a temperature of **15^0C**

- Calendar dates are another example. For instance, the years **2002** and **2010** are **eight** years apart

NEED FOR RATIO SCALED ATTRIBUTES

- Temperatures in Celsius and Fahrenheit do not have a true zero-point, that is, neither **0^0C** nor **0^0F** indicates **"no temperature"** which makes it impossible to compute ratios

- Although we can compute the *difference* between temperature values, we cannot talk of one temperature value as being a *multiple* of another

- Eg: $10^0 + 10^0 = 20^0$ this doesn't mean that 20^0C is twice as hot as 10^0C because there is no 0^0C or "no temperature" when it comes to Celsius scale

NEED FOR RATIO SCALED ATTRIBUTES

- Similarly, there is no true zero-point for calendar dates
- The year 0 does not correspond to the beginning of time
- This brings us to **ratio-scaled** attributes, for which a **true zero-point exists**

RATIO SCALED ATTRIBUTES

- **Interval scaled attributes + clear definition of zero**
- It is a **numeric** attribute with an **inherent zero-point**
- If a measurement is **ratio-scaled**, we can speak of a **value** as being a **multiple** (or ratio) of **another value**
- **Values** are **ordered**, and we can also compute the **difference** between **values**, as well as the **mean**, **median**, and **mode**

RATIO SCALED ATTRIBUTES - EXAMPLE

- Unlike temperatures in Celsius and Fahrenheit, the **Kelvin** (K) temperature scale has what is considered a true zero-point (**0^0K = -273.15^{0}C**)

- Other examples of ratio-scaled attributes include count attributes such as **years of experience** (e.g., the objects are employees) and **number of words** (e.g., the objects are documents)

- Other examples include attributes to measure **weight, height**, and **monetary** quantities (e.g., you are 100 times richer with $100 than with $1)

DIFFERENCE BETWEEN ATTRIBUTE TYPES

PROVIDES	NOMINAL	ORDINAL	INTERVAL	RATIO
“order” of values is known		Y	Y	Y
Can quantify the difference between each value			Y	Y
Can add or subtract values			Y	Y
Can multiply and divide values				Y
Has true zero				Y

DISCRETE ATTRIBUTES

- Has **finite** number of **values**
- It can be in **numerical** (integers or real numbers) or **categorical** form
- Example:

ATTRIBUTE	VALUE
Profession	Teacher, Doctor, Engineer, Clerk, Peon
Postal code	402022, 400056, 400060
Hair color	Black, Blonde, Red, Brown

CONTINUOUS ATTRIBUTES

- Can take on **any value between two specified values**
- Have **infinite** number of **values**
- **Continuous** values are **real** numbers (not categorical)
- They are represented as **floating point numbers**

ATTRIBUTE	VALUE
Height	4.7, 5.4, 6.1
Weight	40.3, 52.6

STATISTICAL DESCRIPTION OF DATA

- Used to **identify properties** of the **data** and **highlight** which data **values** should be treated as **noise** or **outliers**
- It includes:
  - Measuring the Central Tendency:
    - ✓Mean, Median, and Mode
  - Measuring the Dispersion of Data:
    - ✓Range, Quartiles, Interquartile Range, Variance, and Standard Deviation
  - Graphic Displays of Basic Statistical Descriptions of Data
    - ✓Quantile plot, Quantile-Quantile plot, Histograms, Scatter plots & data correlation

MEASURING THE CENTRAL TENDENCY

- Suppose that an **attribute *X***, like ***salary***, has been recorded for a set of **objects**

- Let $x_1, x_2, \ldots, x_N$ be the set of ***N observations* for *X*** referred to as the data set. To **plot** the **observations** for ***salary***, **where** would **most** of the **values fall**?

- **Central tendency** is a **measure** of **central location** that **describes** the **central position** within the **set of data**

- **Measures** of central tendency include the **mean, median, mode**, and **midrange**

MEAN

- Most **common** and **effective numeric measure** of the "**center**" of a set of **data** or **attribute values**

- It is the **average** of **numbers**. Also called as **arithmetic mean**

- Can be used with both **discrete** and **continuous** attributes but commonly used with **continuous** attributes

- Let $\mathbf{x_1, x_2, \ldots, x_N}$ be a set of **N values** or observations, such as for some numeric **attribute X**, like salary

- The **mean** of this set of values is $\frac{\sum_{i=1}^{N} x_i}{N} = \frac{x_1 + x_2 + x_3 + \ldots + xN}{N}$

MEAN FOR GROUPED DATA (ESTIMATED MEAN)

- Sometimes, each **value x_i** in a set may be **associated** with a **weight w_i** for **i = 1,…,N**

- The **weights reflect** the **significance**, **importance**, or **occurrence frequency attached** to their respective **values**

- In this case, we can compute the **weighted arithmetic mean** or **weighted average** as follows:

$$\overline{X} = \frac{\sum_{i=1}^{N} w_i x_i}{\sum_{i=1}^{N} w_i} = \frac{w_1 x_1 + w_2 x_2 + \ldots + wNxN}{w_1 + w_2 + \ldots + wN}$$

DRAWBACK OF MEAN

- **Not always** the **best** way of **measuring** the **center** of the data
- Major **problem** with the **mean** is its **sensitivity** to **extreme** (e.g., outlier) values that can **corrupt** the **mean**
- For **example**, the mean salary at a company may be substantially pushed up by that of a few highly paid managers
- To **offset** the **effect** caused by a **small number** of **extreme values**, we can instead use the **trimmed mean** which is the **mean obtained** after **chopping** off **values** at the **high** and **low extremes**

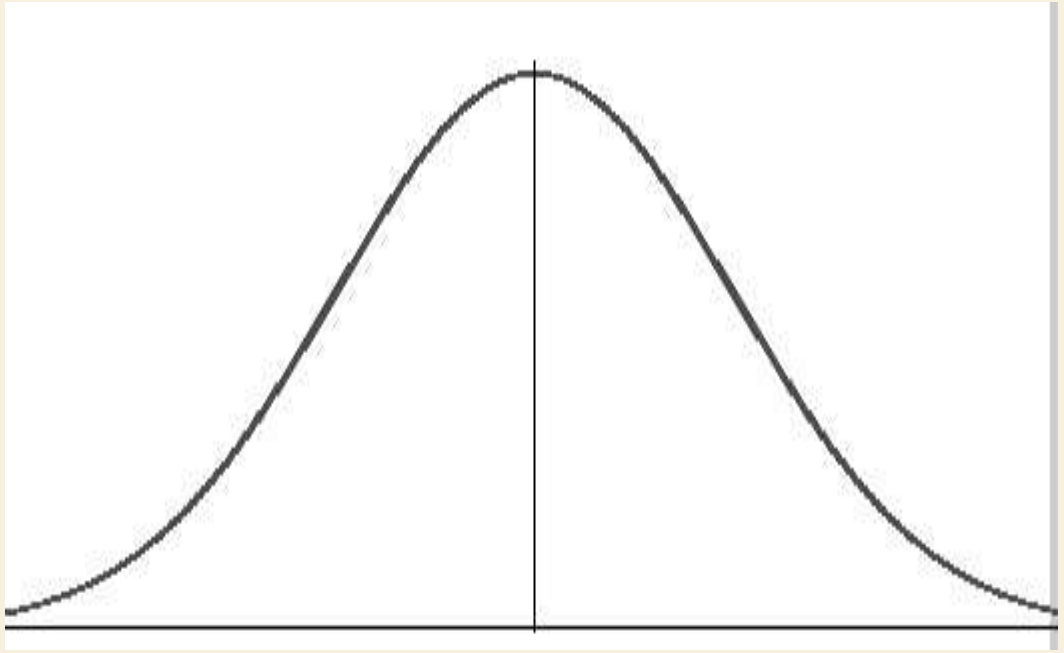
MEAN

- For **example**, we can **sort** the **values** observed for ***salary*** and **remove** the **top** and **bottom 2%** before **computing** the **mean**

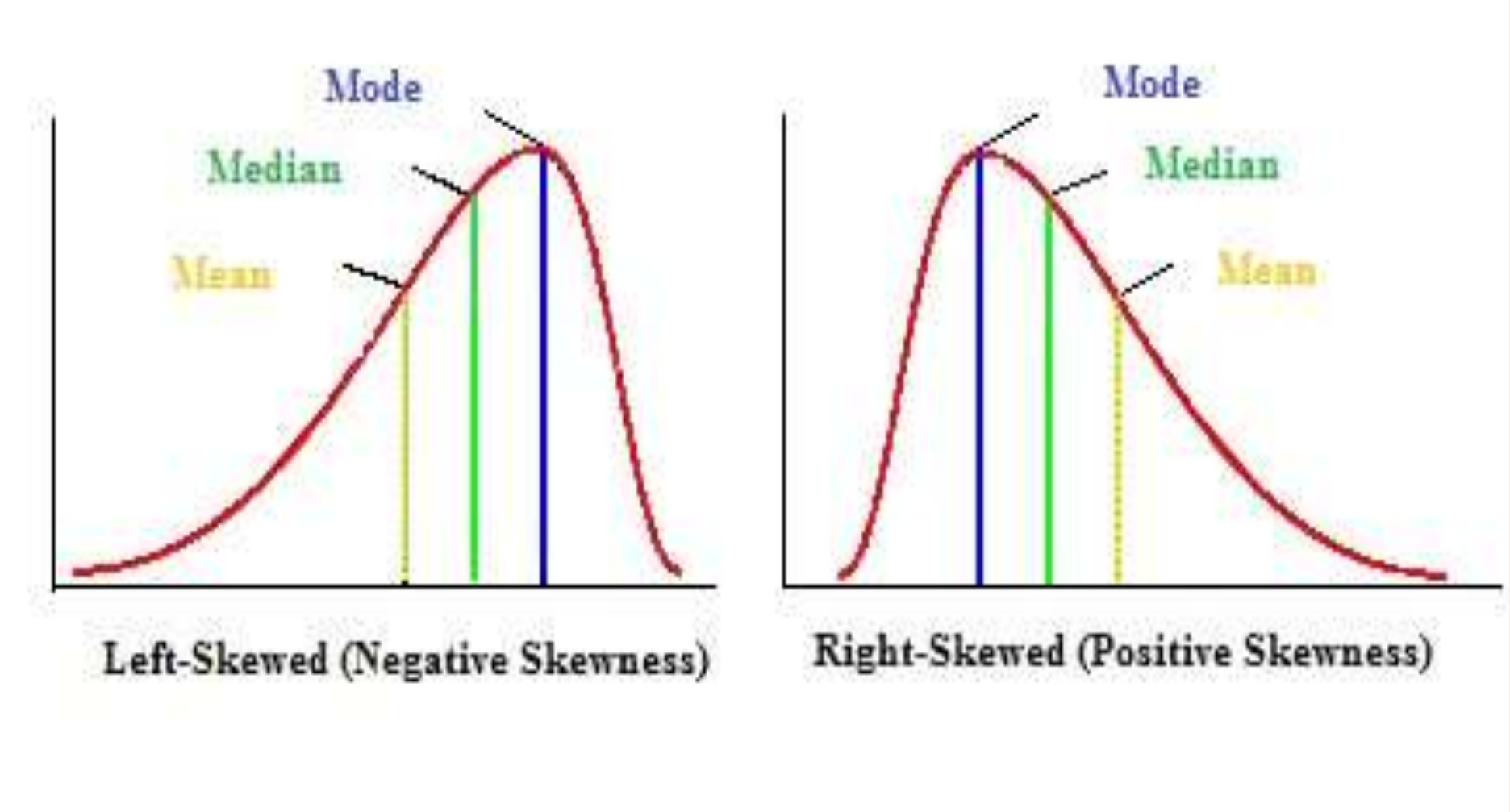
- **Avoid trimming** too **large portions** (such as 20%) at **both** ends, as this can **result** in the **loss** of **valuable information**

SKEWNESS OF DATA

- Normal Curve



SKEWNESS OF DATA



LEFT SKEWED DATA

- Left skewed distribution has a **long left tail**
- Also called **negatively-skewed** distributions
- There is a **long tail** in the **negative** direction on the number line i.e. the **scores fall towards** the **higher** side of the scale
- The **mean** is also to the **left** of the peak i.e. **median**
- **Mean** is **lesser** than the **median** which is **lesser** than the **mode**

RIGHT SKEWED DATA

- Right-skewed distribution has a **long right tail**
- Also called **positively-skewed** distributions
- There is a **long tail** in the **positive** direction on the number line i.e. the **scores fall** towards the **lower** side of the scale
- The **mean** is also to the **right** of the peak i.e. **median**
- **Mean** is **greater** than the **median** which is **greater** than the **mode**

MEDIAN

- **Median** is the **middle** value among **all values** in the **data set**
- For **skewed (asymmetric)** data, a **better measure** of the **center** of data is the **median**, which is the **middle** value in a set of **ordered data values**
- It is the **value** that **separates** the **higher half** of a **data** set from the **lower half**

MEDIAN

- Suppose that a given **data set** of ***N* values** for an **attribute *X*** is **sorted** in **increasing order**
- If ***N* is odd**, then the **median** is the ***middle value*** of the **ordered set**
- If ***N* is even**, then the **median** is **not unique**; it is **any value between the two middlemost values**
- If ***X*** is a **numeric** attribute in this case, by convention, the **median** is taken as the **average of the two middlemost values**

CALCULATING MEDIAN FOR ODD/EVEN NO. OF VALUES

- **Example:** Values are 9, 8, 5, 6, 3 (odd number of values)
  - Arrange values in order
  - 3, 5, 6, 8, 9
  - **Median** = 6

- **Example: Values are** 9, 8, 5, 6, 3, 4 (even number of values)
  - Arrange values in order
  - 3, 4, 5, 6, 8, 9
  - Add 2 middle values and calculate their mean
  - **Median** = (5+6)/2 = 5.5

MEDIAN FOR GROUPED DATA (ESTIMATED MEDIAN)

- Assume that data are grouped in intervals according to their x_i data values and the frequency (i.e. number of data values) of each interval is known

- For example, employees may be grouped according to their annual salary in intervals such as \$10,000–20,000, \$20,000–30,000, and so on

- Median of the entire data set i.e. median salary can be calculated by interpolation using the formula

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

MEDIAN

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

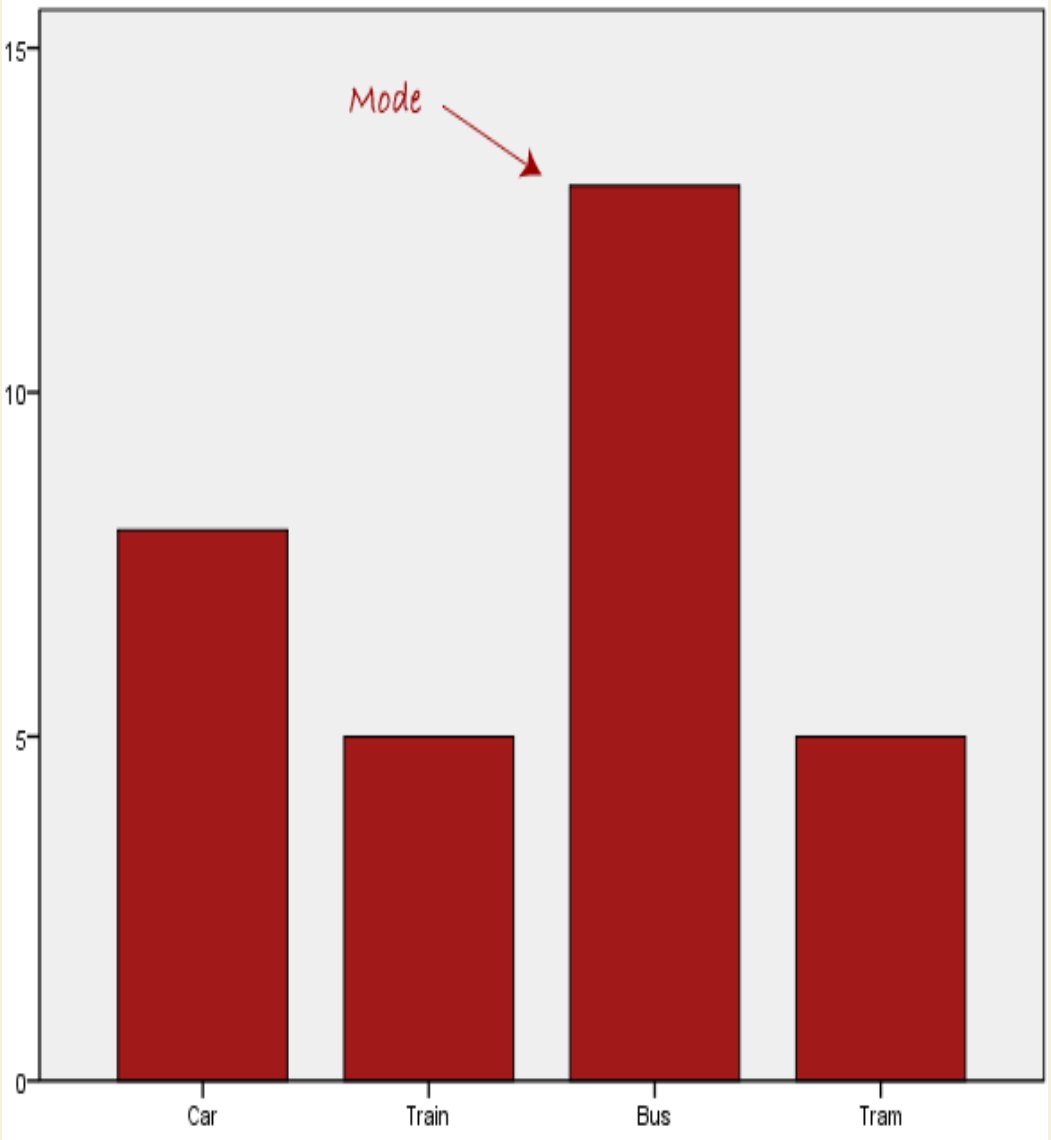
- where,
  - L_1 is the lower boundary of the median interval
  - N is the number of values in the entire data set
  - $(\sum freq)_l$ is the sum of the frequencies of all of the intervals that are lower than the median interval
  - $freq_{median}$ is the frequency of the median interval
  - *width* is the width of the median interval

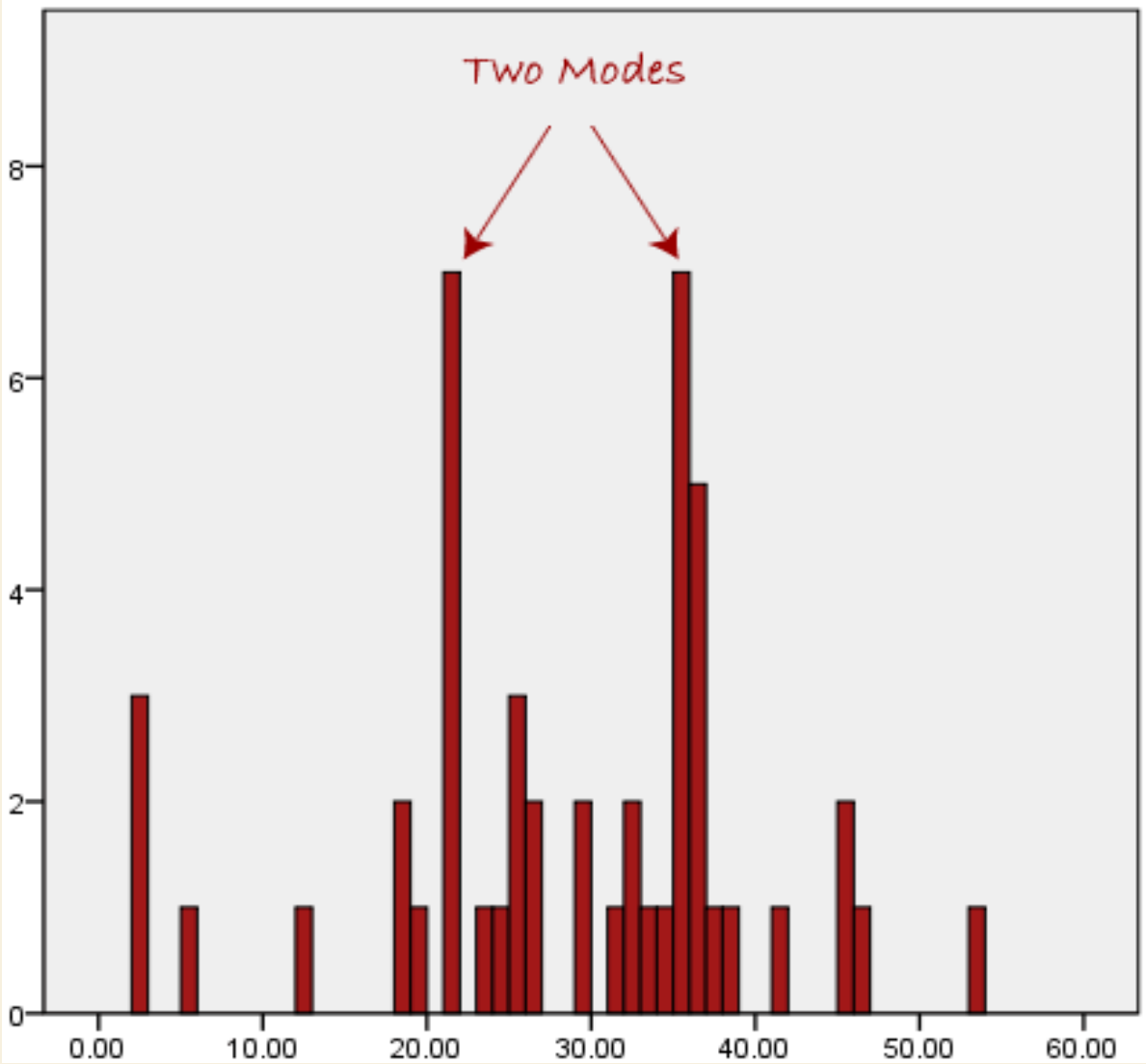
MODE

- The **mode** for a set of data is the value that occurs **most frequently** in the set
- It can be determined for **qualitative** and **quantitative** attributes
- Possible for the **greatest frequency** to correspond to several **different values**, which results in **more than one mode**
- Example: Values are 3, 6, 6, 8, 9

Mode = 6 (because 6 is occurring 2 times and all other values occur only one time)

MODE


Mode
15
10
5
0
Car
Train
Bus
Tram


Two Modes
8
6
4
2
0
0.00
10.00
20.00
30.00
40.00
50.00
60.00

MODE

- Data sets with one, two, or three modes are respectively called **unimodal**, **bimodal**, and **trimodal**
- In general, a data set with two or more modes is called **multimodal**
- At the other extreme, if each **data value** occurs **only once**, then there is **no mode**

MODE FOR GROUPED DATA (ESTIMATED MODE)

$$\text{Mode} = L_1 + \frac{\text{freq}_{\text{mode}} - \text{freq}_{\text{l}}}{(\text{freq}_{\text{mode}} - \text{freq}_{\text{l}}) + (\text{freq}_{\text{mode}} - \text{freq}_{h})} * \text{width}$$

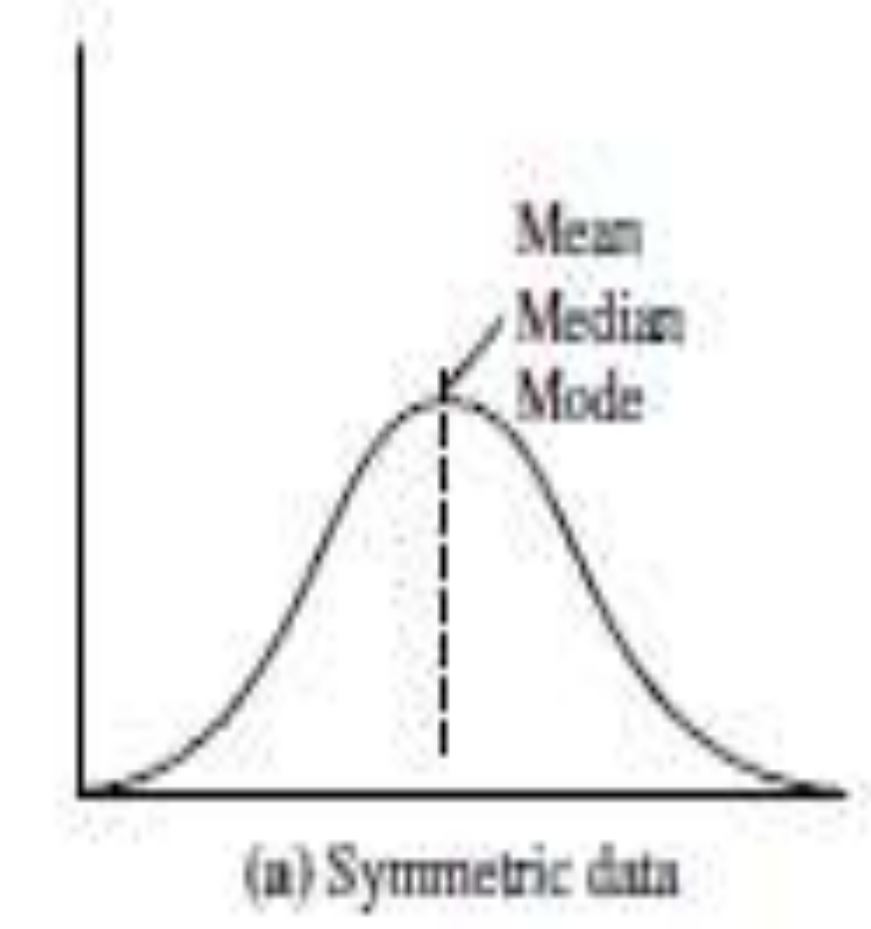
- where,
  - L_1 is the lower boundary of the mode interval
  - $freq_l$ is the frequency of the interval that is lower than the mode interval
  - $freq_h$ is the frequency of the interval that is higher than the mode interval
  - $freq_{mode}$ is the frequency of the mode interval
  - *width* is the width of the mode interval

MIDRANGE

- **Midrange** can also be used to assess the central tendency of a **numeric** data set
- It is the **average** of the **largest** and **smallest** values in the set
- This measure is easy to compute using the SQL aggregate functions, **max**() and **min**()
- It is given by **midrange = (largest_value + smallest_value) /2**

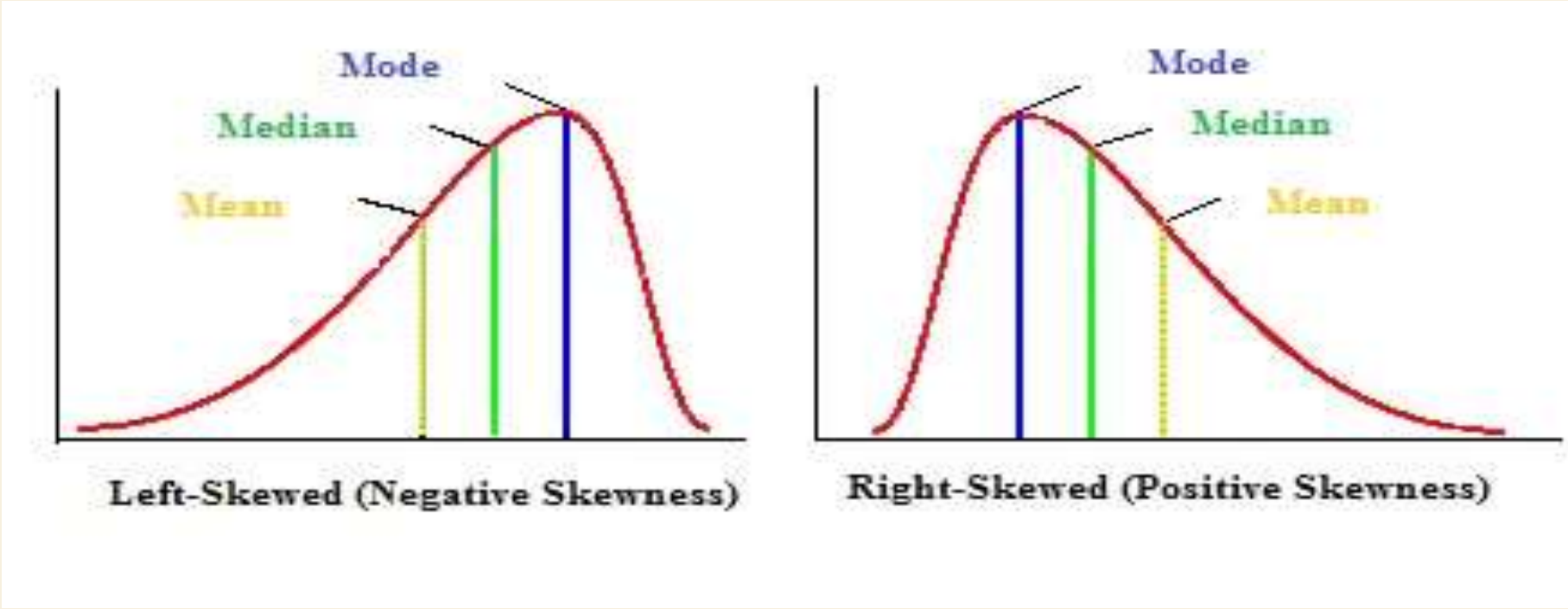
RELATION BETWEEN MEAN, MEDIAN & MODE

- In a unimodal frequency curve with perfect **symmetric** data distribution, the mean, median, and mode are all at the same center value



RELATION BETWEEN MEAN, MEDIAN & MODE

- Data in most real applications are not symmetric
- They may instead be either **positively skewed**, where the mode occurs at a value that is smaller than the median or **negatively skewed**, where the mode occurs at a value greater than the median



MEASURING DISPERSION OF DATA

- **Two data sets** can have the **same mean** but they can be **entirely different**. Thus to **describe** data, one needs to know the **extent** of **variability**

- Various **measures** used to **assess** the **dispersion** or spread of **numeric** data are as follows:
  - **Range, Quantiles, Quartiles, Percentiles, Interquartile range**

- The **five-number summary**, which can be displayed as a **boxplot**, is useful in identifying **outliers**

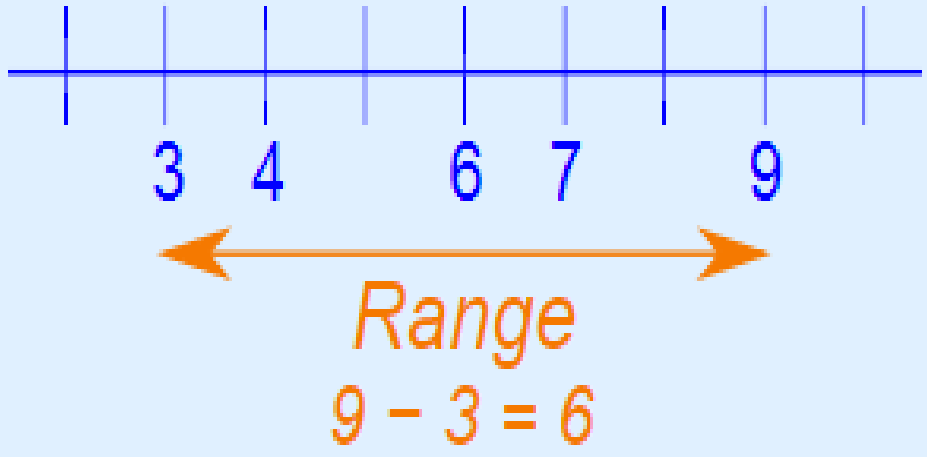
- **Variance** and **standard deviation** also indicate the **spread** of a data **distribution**

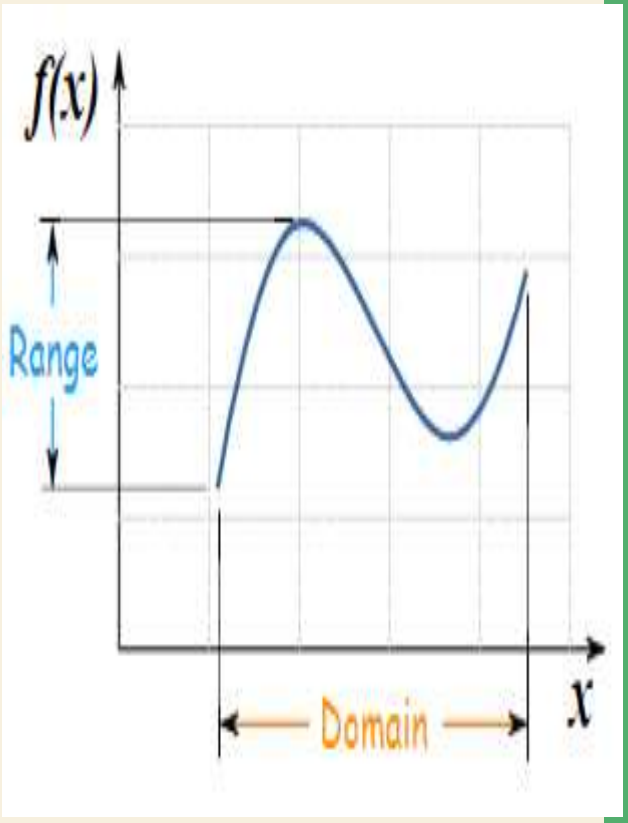
RANGE

- **Range** of the set is the **difference** between the **largest** (max()) and **smallest** (min()) **values**

Example: In **{4, 6, 9, 3, 7}** the lowest value is 3, and the highest is 9.

So the range is 9 – 3 = **6**.





QUANTILES

- Suppose that the **data** for **attribute *X*** are **sorted** in **increasing numeric order**
- We can **pick** certain **data points** so as to **split** the **data distribution** into **equal-sized consecutive sets**
- These **data points** taken at **regular intervals** of a **data distribution**, dividing the data into **essentially equal sized parts** are called as ***quantiles***
- **Quantile** represents **fraction** (or **percent**) of **data below** the **given value** i.e. **0.3** (or **30%**) **quantile** is the point at which **30% percent** of the **data** fall **below** and **70%** fall **above** that **value**

QUARTILES – TYPE OF QUANTILE

- **2-quantile** is the **single data point** **dividing** the **lower and upper halves** of the **data distribution**. It corresponds to the **median**

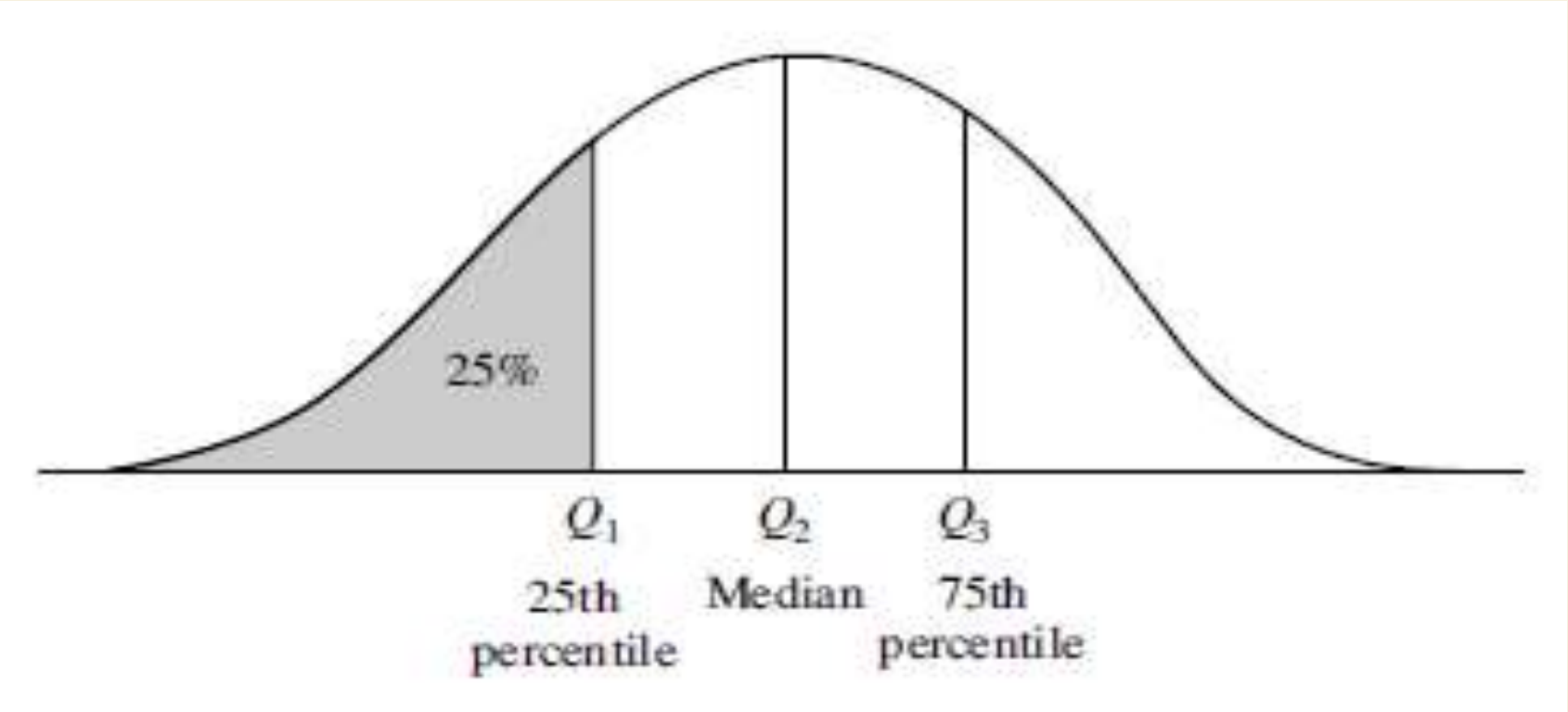
- **4-quantiles** are the **three data points** that split the data distribution into **four equal parts**; each part represents **one-fourth** of the data distribution. Each part is commonly referred to as **quartiles**

- **Quartiles** are also quantiles that **divide** the **data distribution** into **4 equal parts**

- **100-quantiles** i.e. **99 points** are called as **percentiles**; since they **divide** the data distribution into **100 equal-sized** consecutive **sets**

- **Median**, **quartiles**, and **percentiles** are the most **widely** used forms of **quantiles**

EXAMPLE ON QUARTILES



- The quantiles plotted are quartiles
- The three quartiles divide the distribution into four equal-size consecutive subsets
- The second quartile corresponds to the median

QUARTILES

- **Quartiles** give an **indication** of a **distribution's center and spread** (values above and below)

- **First quartile**, denoted by ***Q*1**, is the **25th percentile**. It cuts off the lowest **25%** of the data. **Center** of **smallest** and **median**.

- **Second quartile** is the **50th percentile**. As the median, it gives the **center** of the **data distribution**

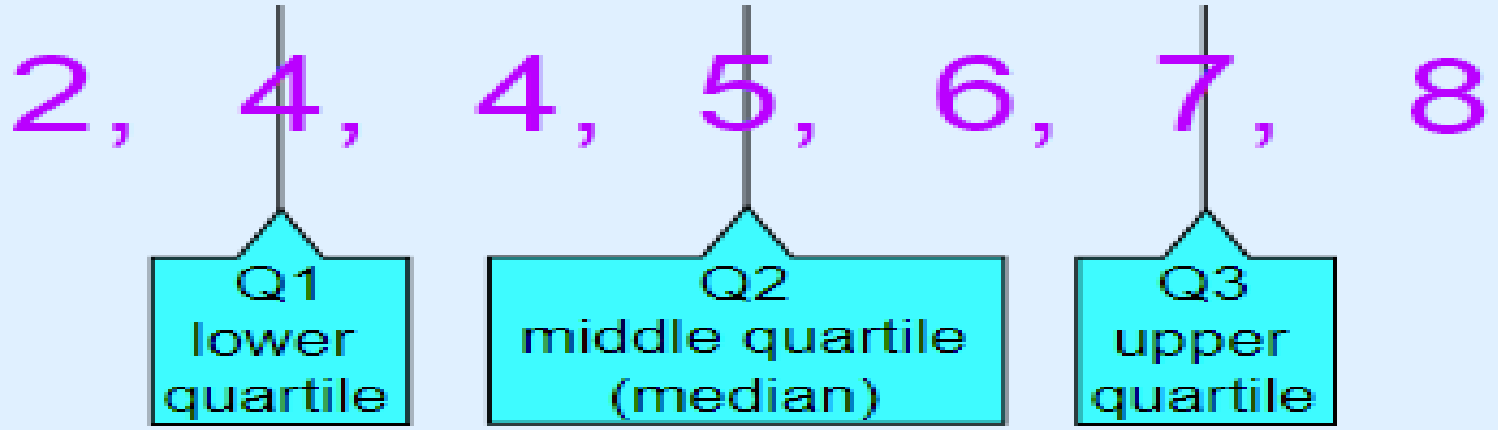
- **Third quartile**, denoted by ***Q*3**, is the **75th percentile**—it cuts off the lowest 75% (or highest 25%) of the data. **Center** of **median** and **largest** value

QUARTILES WITH ODD DATA

Example: 5, 7, 4, 4, 6, 2, 8

Put them in order: 2, 4, 4, 5, 6, 7, 8

Cut the list into quarters:



And the result is:

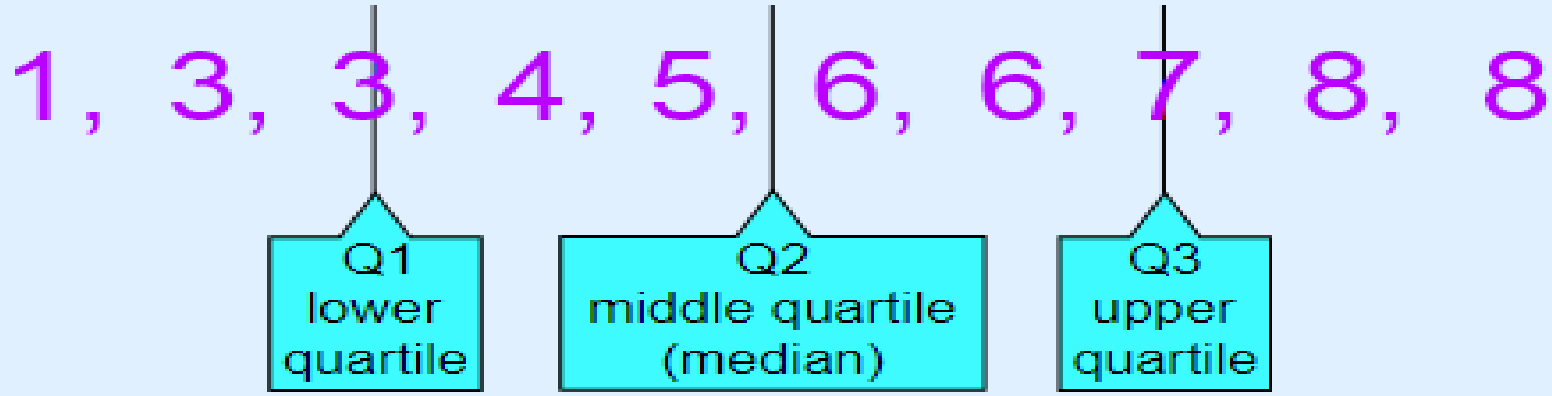
- Quartile 1 (Q1) = **4**
- Quartile 2 (Q2), which is also the Median, = **5**
- Quartile 3 (Q3) = **7**

QUARTILES WITH EVEN DATA

Example: 1, 3, 3, 4, 5, 6, 6, 7, 8, 8

The numbers are already in order

Cut the list into quarters:



In this case Quartile 2 is half way between 5 and 6:

$$Q2 = (5+6)/2 = \mathbf{5.5}$$

And the result is:

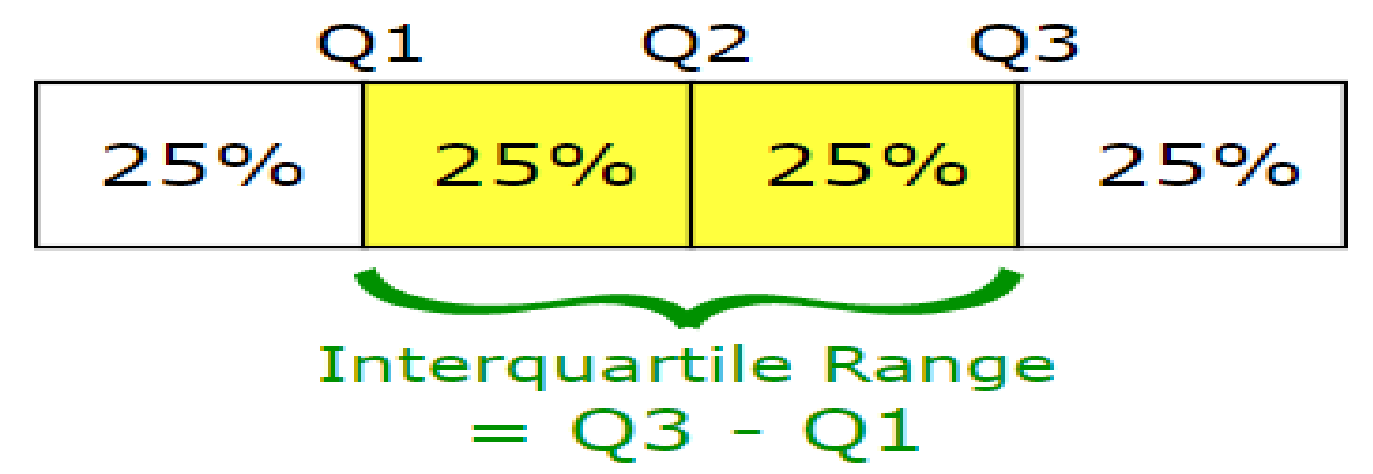
- Quartile 1 (Q1) = **3**
- Quartile 2 (Q2) = **5.5**
- Quartile 3 (Q3) = **7**

INTER QUARTILE RANGE

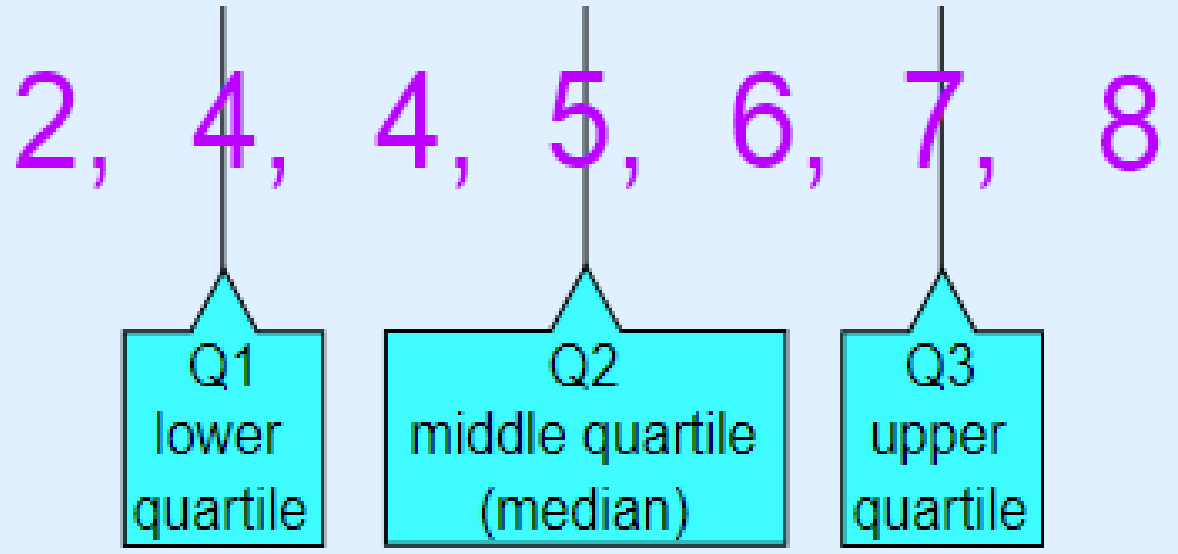
- The **distance** between the **first** and **third quartiles** is a simple measure of spread that gives the **range** covered by the **middle half** of the data

- This distance is called the **interquartile range** (**IQR**) and is defined as

IQR = Q3 – Q1



EXAMPLE ON INTER QUANTILE RANGE



The **Interquartile Range** is:

$$Q3 - Q1 = 7 - 4 = \mathbf{3}$$

FIVE NUMBER SUMMARY

- The "**five number summary**", or "**five statistical summary**", consists of:
  1) minimum
  2) first quartile Q1 (25% mark)
  3) median
  4) third quartile Q3 (75% mark)
  5) maximum

FIVE NUMBER SUMMARY

- **Obtain five number summary for the data set:** 18,15,27,6,9,2,1,5,7,12,19

✓**Step 1:** *Put the numbers in ascending order*.

1, 2, 5, 6, 7, 9, 12, 15, 18, 19, 27

✓**Step 2:** *Find the minimum and maximum* values from the data set

minimum = 1 and maximum = 27

✓**Step 3:** *Find the median*

median = 9

FIVE NUMBER SUMMARY

✓**Step 4:** *Place parentheses around the numbers* ***above and below*** *the median*

(1,2,5,6,7),9,(12,15,18,19,27)

✓**Step 5:** *Find Q1 and Q3*. Q1 can be thought of as a median in the lower half of the data, and Q3 can be thought of as a median for the upper half of data

(1,2,**5**,6,7), **9**, (12,15,**18**,19,27).

✓**Step 6:** *Write down your summary found in the above steps*.

minimum=1, Q1 =5, median=9, Q3=18, and maximum=27

BOX PLOT

- **Box plot** is a **plotting** of **data** in such a way that it is like a **box shape** and it represents the **five number summary**

- **Ends of the box** are at the (**1st and 3rd**) **quartiles** so that the **box length** is the **interquartile range**

- **Median** is marked by a **line within** the **box**

- **Rectangle** is drawn to represent the **second and third quartiles**, usually with a **vertical line inside** to **indicate** the **median** value

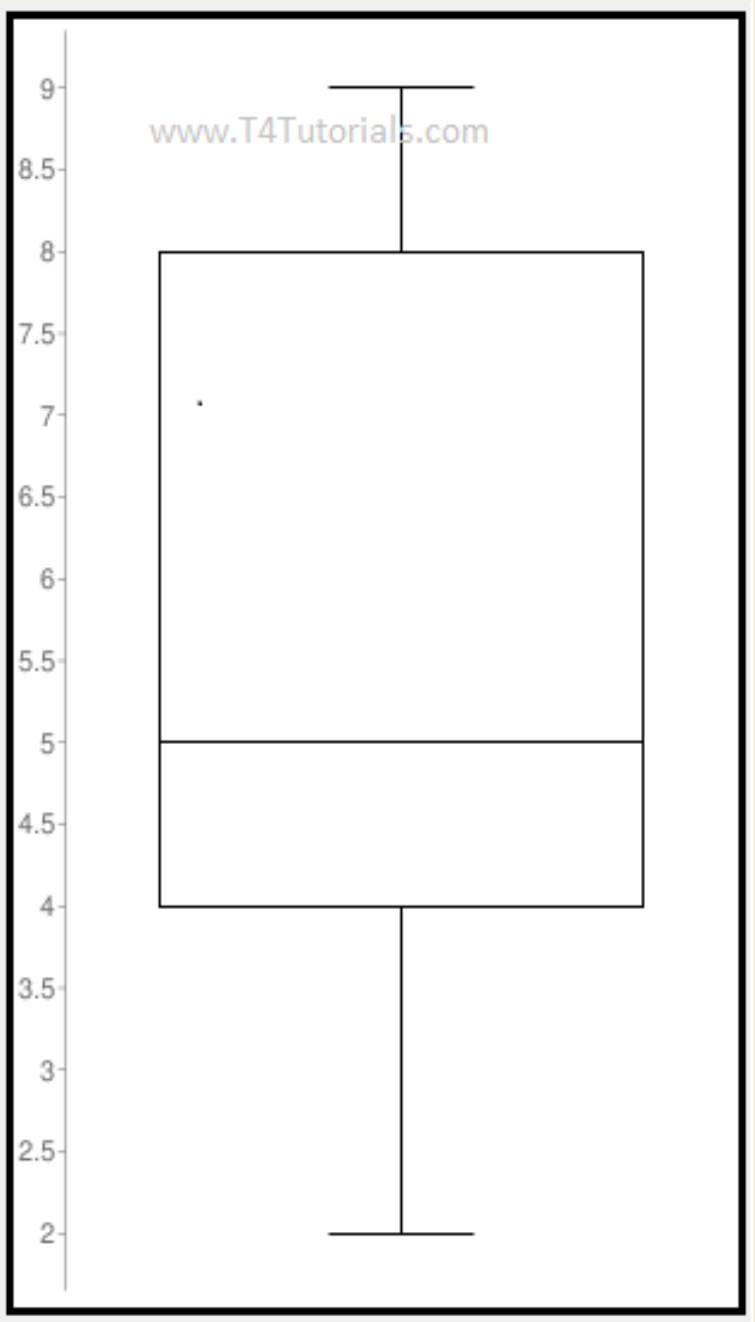
BOX PLOT

- **Lower** and **upper quartiles** are shown as **horizontal lines** either **side of the rectangle**

- **Maximum** and **minimum** values are on the **whiskers**

EXAMPLE ON BOX PLOT

- **Draw the box plot for the odd length data set.**
  - ✓Data = 2, 4, 5, 5, 8, 8, 9
  - ✓Q1: 4 – lower quartile
  - ✓Q2: 5 – middle quartile
  - ✓Q3: 8 – upper quartile
  - ✓Quartile range = Q3 – Q1 = 8 – 4 = 4
  - ✓Median = 5
  - ✓Maximum = 9
  - ✓Minimum = 2



EXAMPLE ON BOX PLOT

- **Draw the box plot for the even length data set.**
  - ✓Data = 8, 5, 2, 4, 8, 9, 5,7
  - ✓Arrange the values in sequence as follows:

    2, 4, 5, 5, 7, 8, 8, 9
  - ✓Q1: (4+5)/2 = 4.5 – lower quartile
  - ✓Q2: (5+7)/2 = 6 – middle quartile or median
  - ✓Q3: (8+8)/2 = 8 – upper quartile
  - ✓Quartile range = Q3 – Q1 = 8 – 4.5 = 3.5
  - ✓Median = (5+7)/2 = 6
  - ✓Maximum = 9
  - ✓Minimum = 2

VARIANCE AND STANDARD DEVIATION

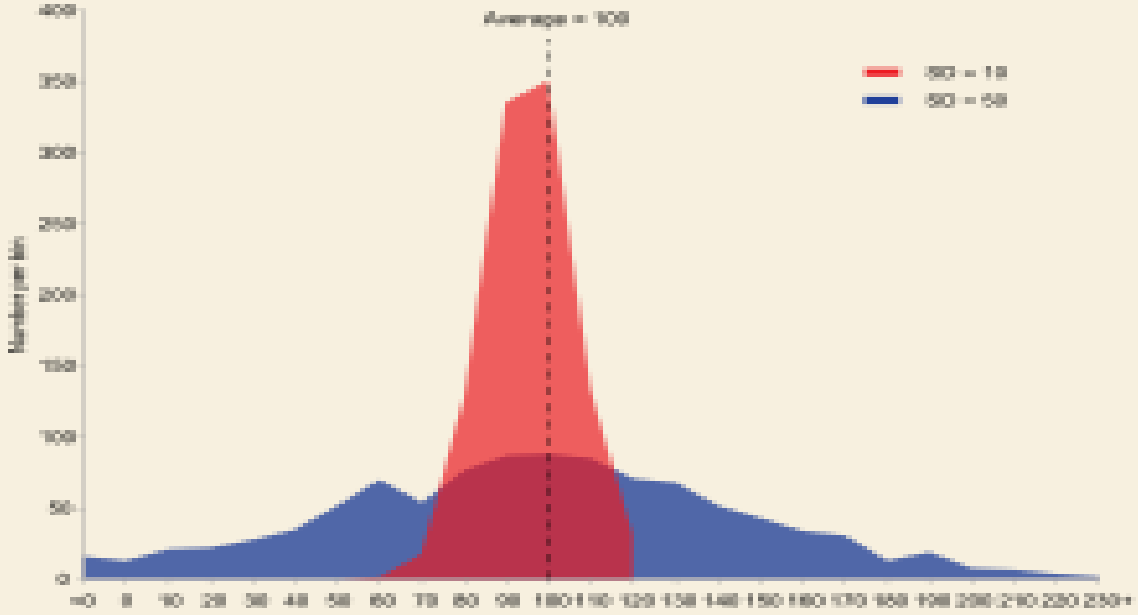
- Different **values** in the **data set** can be **spread** here and there **from** the **mean**
- **Variance** tells how **far** away are the **values from the mean**
- **Standard deviation** is the **square root** of the **variance**
- **Low standard deviation** means that the **data observations** tend to be very **close** to the **mean**
- **High standard deviation** indicates that the **data** are spread out over a **large range** of values **from** the **mean**

VARIANCE AND STANDARD DEVIATION

- Variance of N observations, $x_1, x_2, \ldots, x_N$, for a numeric attribute X is

$$\sigma^2 = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N}$$

where, $\overline{x}$ is the mean value of the observations



VARIANCE AND STANDARD DEVIATION

- Standard deviation σ, of the observations is given by the square root of the variance (σ^2)

- Basic properties of the standard deviation as a measure of spread are as follows:

  ✓**σ measures** spread about the **mean** and should be considered only when the **mean** is **chosen** as the **measure of center**

  ✓**σ = 0** only when there is **no spread**, that is, when **all observations** have the **same value**. Otherwise, $\sigma > 0$.

GRAPHIC DISPLAYS OF BASIC STATISTICAL DESCRIPTIONS OF DATA

- Graphic displays include:
  - Quantile plots
  - Quantile–Quantile plots
  - Histograms
  - Scatter plots

- Helpful for **visual** inspection of **data**, which is useful for data **preprocessing**

QUANTILE PLOT

- Simple and effective way to look at **univariate data distribution**

- Plots quantile information. Let x_i , for i = 1 to N, be the data sorted in increasing order for some ordinal or numeric attribute X

- Each observation, x_i , is paired with a percentage, f_i , which indicates that approximately f_i * 100% of the data are below the value, x_i

$$f_i = \frac{i - 0.5}{N}$$

- Allows to compare different distributions based on their quantiles

HISTOGRAM (BAR CHART)

- **Histogram** is a plot that shows the underlying **frequency distribution** (shape) of a given **attribute X**
- If X is **nominal** such as item_type then a **vertical bar** is drawn for each known value of X
- **Height** of the bar indicates the **frequency** of that X value
- This allows the **inspection** of the data for its **underlying distribution** (e.g., **normal** distribution), **outliers, skewness**, etc.

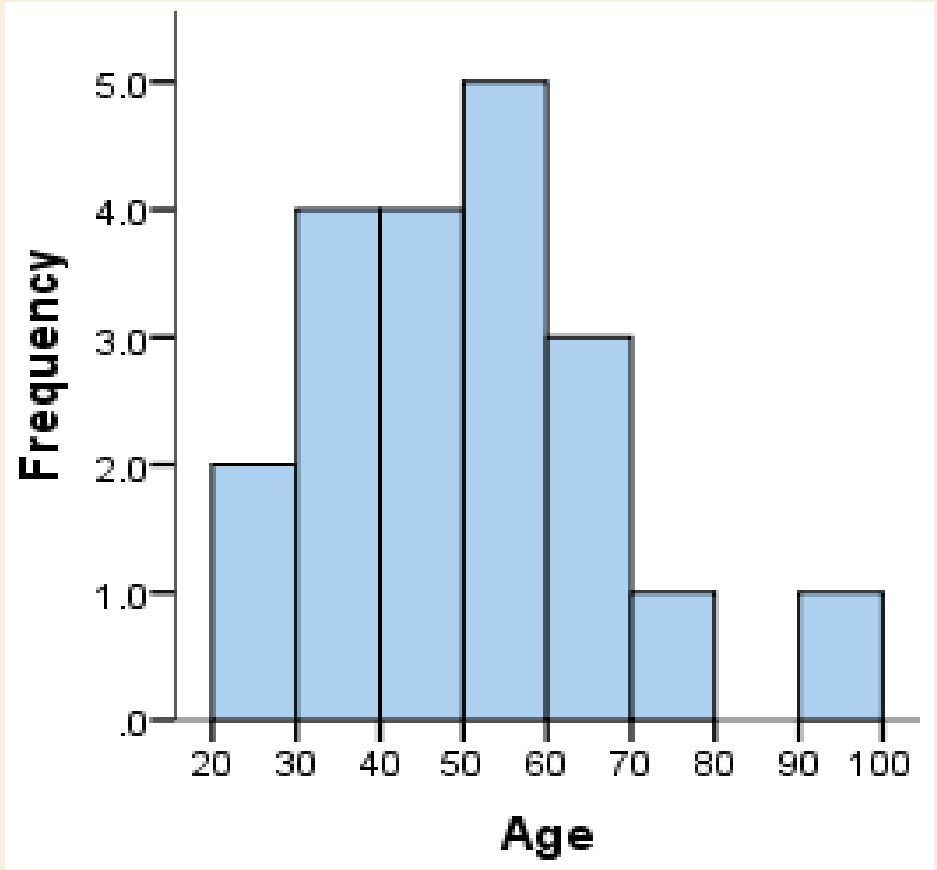
HISTOGRAM (BAR CHART)

- Ages are given as follows:

36	25	38	46	55	68	72	55	36	38
67	45	22	48	91	46	52	61	58	55

- Construct bins and calculate frequency

Bin	Frequency	Scores Included in Bin
20-30	2	25,22
30-40	4	36,38,36,38
40-50	4	46,45,48,46
50-60	5	55,55,52,58,55
60-70	3	68,67,61
70-80	1	72
80-90	0	-
90-100	1	91



SCATTER PLOTS & DATA CORRELATION

- Used to **determine** whether there is a **relationship, pattern or trend** between two **numeric** attributes

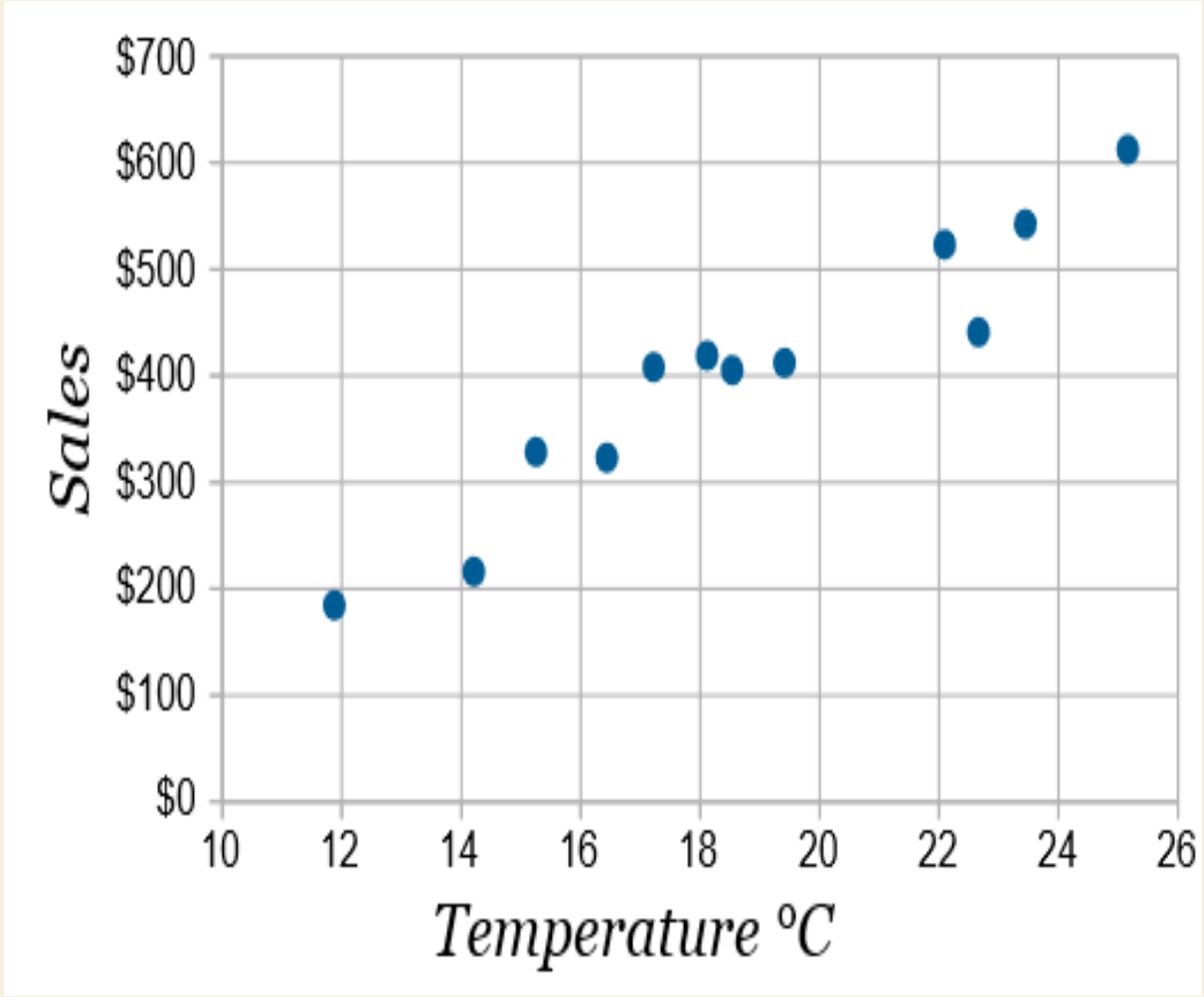
- To **construct** a scatter plot, each **pair of values** is treated as a **pair of coordinates** in an algebraic sense and **plotted as points in the plane**

- Used for **analysing bivariate** data to see **clusters** of points, **outliers** or **correlation** relationships

SCATTER PLOTS & DATA CORRELATION

Ice Cream Sales vs Temperature

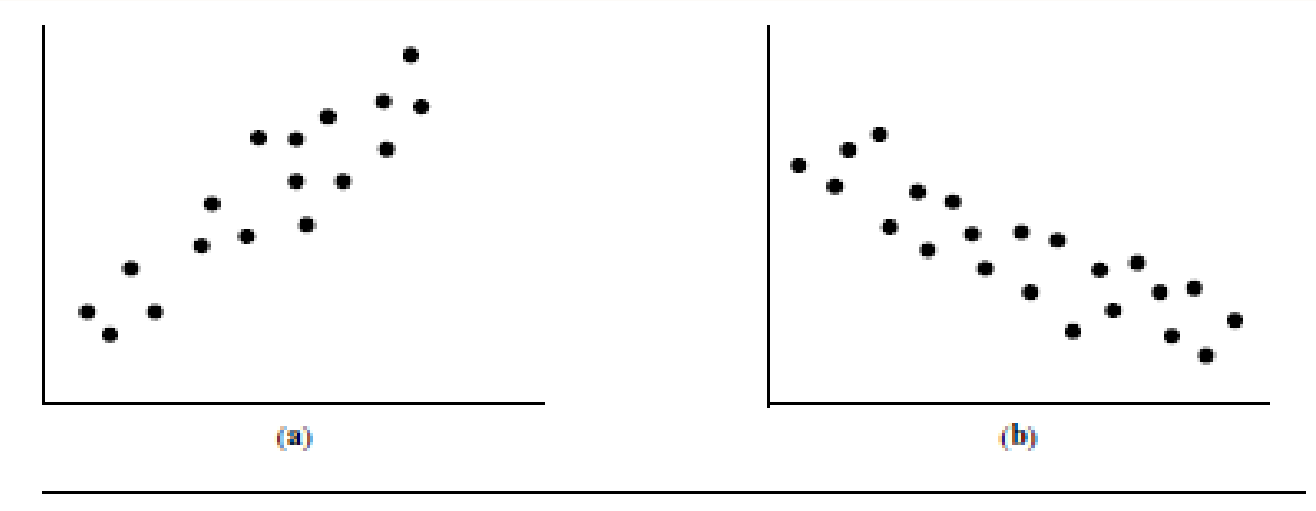
Temperature °C	Ice Cream Sales
14.2°	$215
16.4°	$325
11.9°	$185
15.2°	$332
18.5°	$406
22.1°	$522
19.4°	$412
25.1°	$614
23.4°	$544
18.1°	$421
22.6°	$445
17.2°	$408



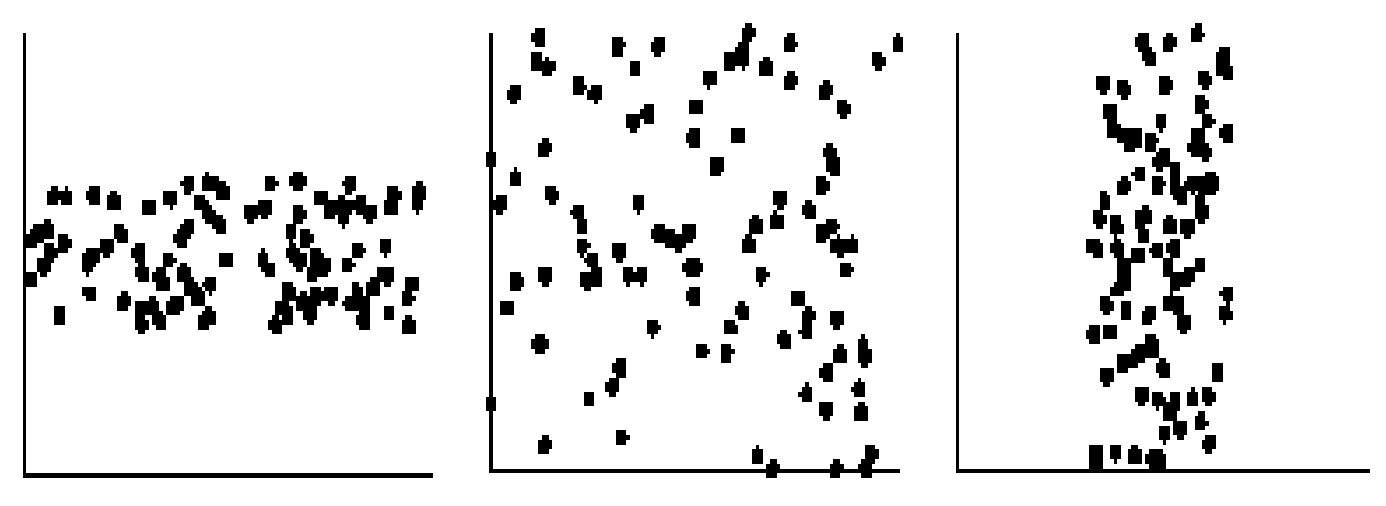
warmer weather leads to more sales

SCATTER PLOTS & DATA CORRELATION

- Fig. a) positive correlation
- Fig. b) negative correlation



- No correlation



DATA SIMILARITY & DISSIMILARITY

- **Clustering, outlier analysis**, and **nearest-neighbour classification** needs ways to assess how **alike or unalike** objects are in comparison to one another

- For eg: clusters of ***customer*** objects, resulting in groups of customers with similar characteristics (e.g., similar income, area of residence, and age)

DATA SIMILARITY & DISSIMILARITY

- **Similarity measure** for two objects, ***i* and *j***, will typically **return the value 0** if the **objects are unalike**

- **Higher** the **similarity** value, **greater** the **similarity** between objects. A value of **1** indicates **complete similarity**, that is, the objects are identical

- Dissimilarity measure works the opposite way. It returns a **value of 0** if the **objects are the same**

- **Higher** the **dissimilarity** value, the **more dissimilarity** between two objects

- **Proximity** refers to a similarity or dissimilarity

DATA MATRIX

- Stores the *n* data objects in the form of a relational table, or *n*-by-*p* matrix (*n* objects *p* attributes)

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

DISSIMILARITY MATRIX

- Stores a collection of **proximities** that are available for all pairs of n objects. It is often represented by an **n-by-n** table
- where **d(i, j)** is the measured **dissimilarity** or "difference" between objects **i and j**
- **Symmetric** matrix

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

DISSIMILARITY MATRIX

- **Measures of similarity** can often be expressed as a **function** of measures of **dissimilarity**

$$\mathbf{sim(i, j) = 1 - d(i, j)}$$

where sim(i, j) is the similarity between objects i and j

- Many clustering and nearest-neighbor algorithms operate on a dissimilarity matrix

- **Data** in the form of a **data matrix** can be **transformed** into a **dissimilarity matrix** before applying such algorithms

DISSIMILARITY OF NOMINAL ATTRIBUTES

- **Nominal attribute** can take on **two or more states**. For eg., ***map_color*** is a nominal attribute that may have five states: ***red, yellow, green, pink, blue***

- Let the **number of states** of a nominal attribute be ***M***

- **Dissimilarity** between two **objects i and j** can be computed based on the ratio of **mismatches**:

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

- where **m: # of matches** (i.e., the number of attributes for which i and j are in the same state), **p: total # of attributes**

- Alternatively, **similarity** can be computed as **sim(i, j) = 1 – d(i, j) = m/p**

EXAMPLE FOR DISSIMILARITY OF NOMINAL ATTRIBUTES

Object Identifier	test-1 (nominal)
1	code A
2	code B
3	code C
4	code A

- Here, p = 1, since there is only one attribute

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

DISSIMILARITY OF BINARY ATTRIBUTES

- **Contingency table for binary attributes**

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

- q is the number of attributes that equal 1 for both objects i and j
- r is the number of attributes that equal 1 for object i but equal 0 for object j
- s is the number of attributes that equal 0 for object i but equal 1 for object j
- t is the number of attributes that equal 0 for both objects i and j
- total number of attributes is p, where $p = q + r + s + t$

DISSIMILARITY OF BINARY ATTRIBUTES

- **Symmetric binary dissimilarity**

✓If **objects *i* and *j*** are described by **symmetric binary** attributes, then the dissimilarity between *i* and *j* is calculated as follows:

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

- **Asymmetric binary dissimilarity**

✓If objects *i* and *j* are described by asymmetric binary attributes, then the dissimilarity between *i* and *j* is calculated as follows:

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

DISSIMILARITY OF BINARY ATTRIBUTES

- **Jaccard coefficient** (similarity measure for asymmetric binary variables) is given by:

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

$$\textbf{sim(i, j) = 1 – d(i, j) = 1 - } \frac{r+s}{q+r+s}$$

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

EXAMPLE OF DISSIMILARITY OF BINARY ATTRIBUTES

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

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

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

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

DISSIMILARITY OF NUMERIC DATA

- **Minkowski distance (L_h norm)**
- It is a **generalization** of the **Euclidean** and **Manhattan** distances

- It is defined as follows:

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

where h is a real number such that **h > 1**

- It represents the **Manhattan** distance when **h = 1** and **Euclidean** distance when **h = 2**

DISSIMILARITY OF NUMERIC DATA

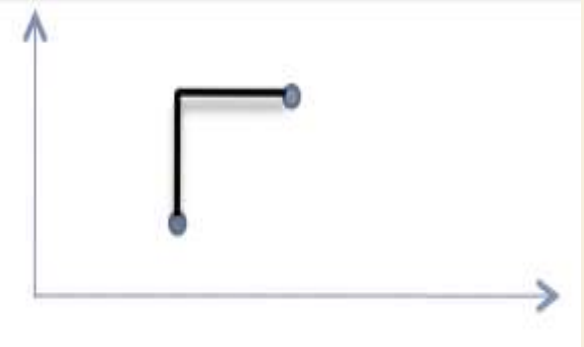
- **Euclidean distance (L_2 norm)**

✓Let $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ be two objects described by p numeric attributes

✓The Euclidean distance between objects i and j is defined as follows:

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

DISSIMILARITY OF NUMERIC DATA



- **Manhattan (city block) distance (L_1 norm)**

✓It is the distance in blocks between any two points

✓Eg., Hamming distance: the number of bits that are different between two binary vectors

✓The Manhattan distance between objects *i* and *j* is defined as follows:

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

DISSIMILARITY OF NUMERIC DATA

- Both the Euclidean and the Manhattan distance satisfy the following mathematical properties:

✓**Non-negativity:** $d(i, j) \geq 0$: Distance is a **non-negative** number

✓**Identity of indiscernibles:** $d(i, i) = 0$: Distance of an **object** to **itself** is **0**

✓**Symmetry:** $d(i, j) = d(j, i)$: Distance is a **symmetric** function

✓**Triangle inequality:** $d(i, j) \leq d(i, k) + d(k, j)$: Going directly from object i to object j in space is no more than making a detour over any other object k

DISSIMILARITY OF NUMERIC DATA

- **Supremum distance (Chebyshev distance, L_{max} norm, L_∞ norm)**

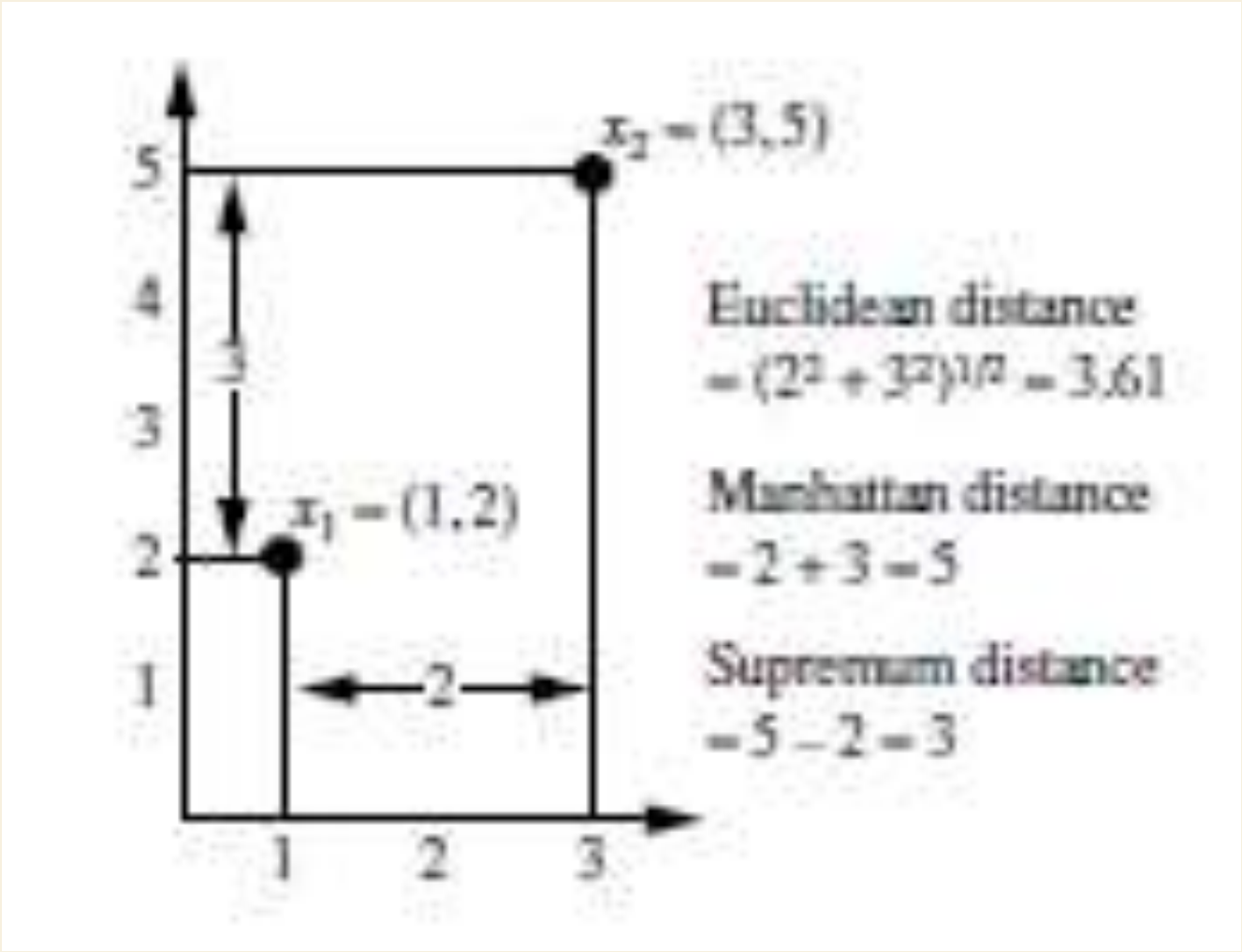
✓It is a **generalization** of the **Minkowski** distance for h → ∞

✓To compute it, we find the **attribute f** that gives the **maximum difference** in values **between** the **two objects**

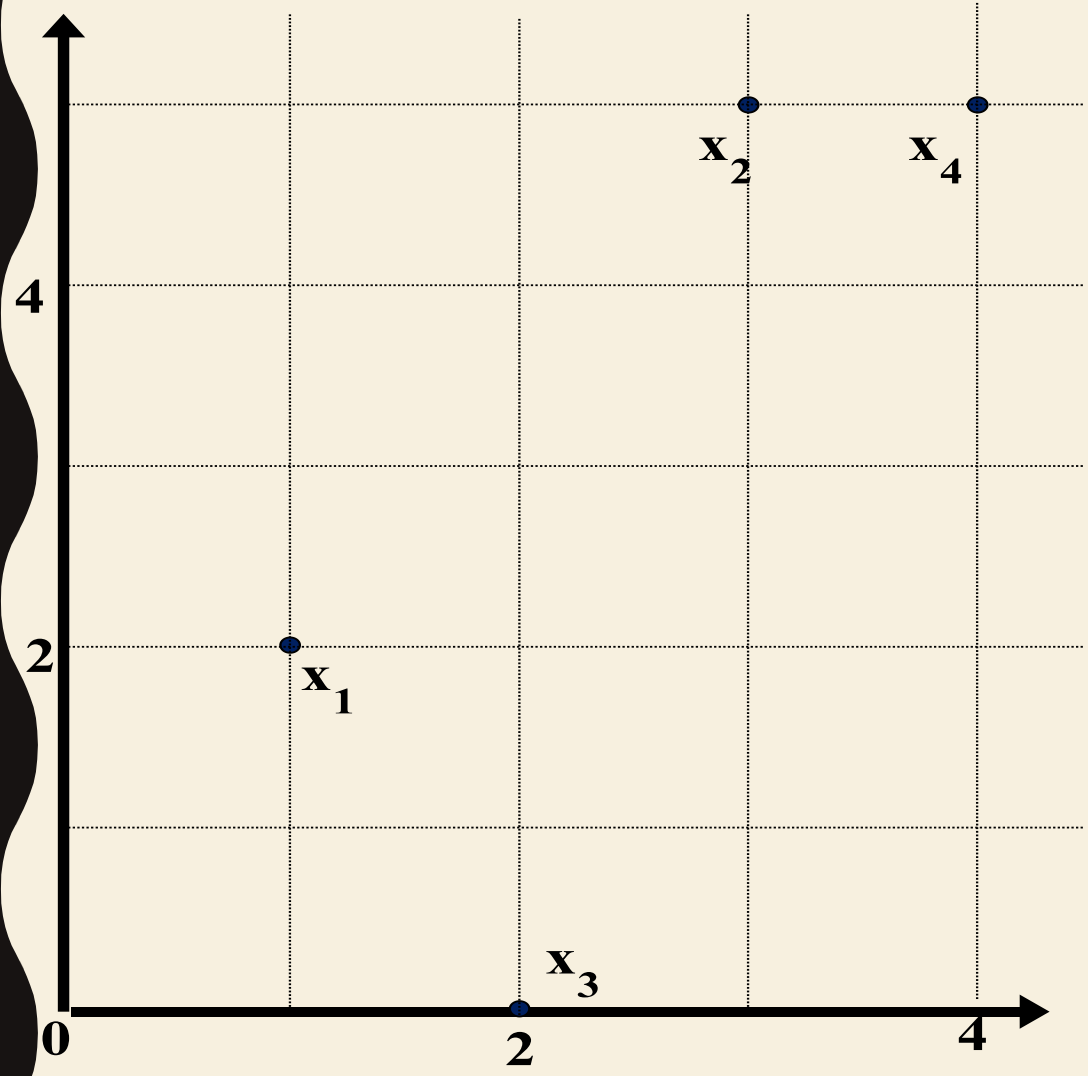
✓This difference is the Supremum distance, defined more formally as:

$$d(i,j) = \lim_{h\to\infty} \left(\sum_{f=1}^{p} |x_{if} - x_{jf}|^h \right)^{\frac{1}{h}} = \max_{f}^{p} |x_{if} - x_{jf}|$$

EXAMPLE ON DISSIMILARITY OF NUMERIC DATA



EXAMPLE: DATA MATRIX AND DISSIMILARITY MATRIX



Data Matrix

point	attribute1	attribute2
x1	1	2
x2	3	5
x3	2	0
x4	4	5

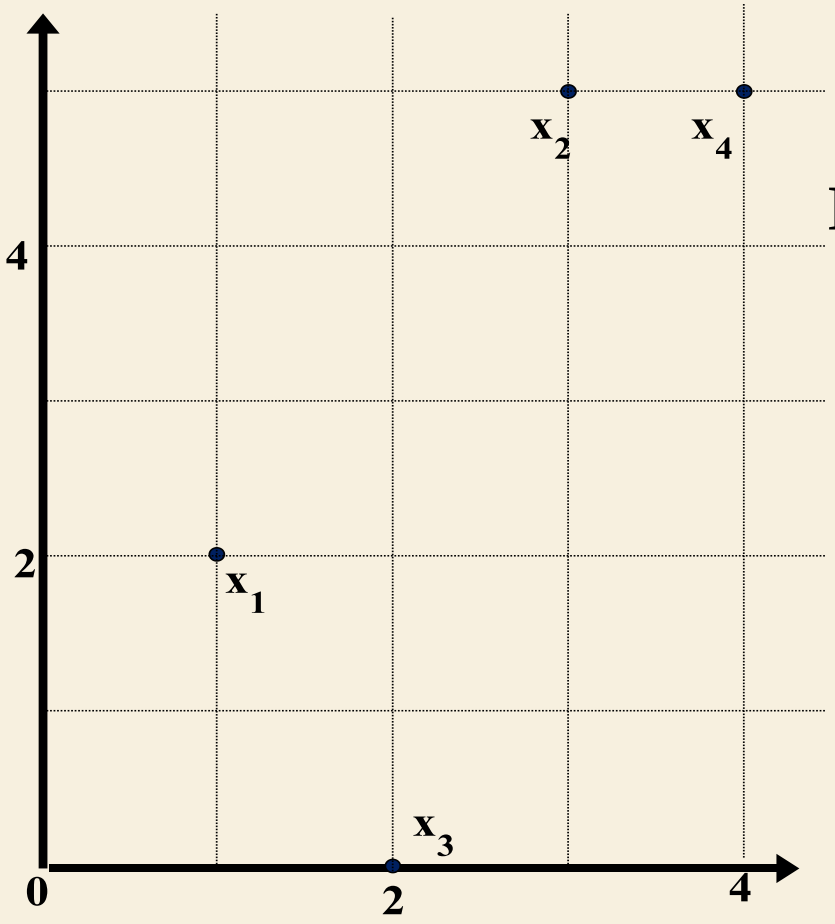
Dissimilarity Matrix

(with Euclidean Distance)

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

EXAMPLE: MINKOWSKI DISTANCE

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

Supremum

L_∞	x1	x2	x3	x4
x1	0			
x2	3	0		
x3	2	5	0	
x4	3	1	5	0

DISSIMILARITY OF ORDINAL DATA

- **Values** of an **ordinal** attribute have a **meaningful order** or ranking about them, yet the **magnitude** between successive **values** is **unknown**

- Suppose that **f is an attribute** from a **set of ordinal attributes** describing **n** objects

- **Dissimilarity** computation with respect to **f** involves the following steps:
  - ✓Value of f for the **i^{th}** object is **x_{if}** , and **f** has **M_f ordered** states, representing the ranking **1, … , M_f** . **Replace** each **x_{if}** by its corresponding **rank**, $r_{if} \in \{1,.., M_f\}$

DISSIMILARITY OF ORDINAL DATA

- Since each **ordinal** attribute can have a **different number of states**, it is often necessary to **map the range** of each attribute onto **[0.0, 1.0]** so that each **attribute** has **equal weight**. Such data **normalization** is performed by **replacing** the **rank r_{if}** of the **i^{th} object in the f^{th} attribute by**

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

- **Dissimilarity** can then be computed using **any** of the **distance measures** described for **numeric attributes**, using **z_{if}** to **represent** the **f** value for the **i^{th} object**

EXAMPLE - DISSIMILARITY OF ORDINAL DATA

- There are **3 states** for test-2: fair, good, and excellent, i.e., $\mathbf{M_f = 3}$

Object Identifier	test-2 (ordinal)
1	excellent
2	fair
3	good
4	excellent

- For **step 1**, if we **replace** each **value** for test-2 by its **rank**, the 4 objects are assigned the ranks 3, 1, 2, and 3, respectively

- **Step 2 normalizes** the **ranking** by **mapping** rank **1 to 0.0,** rank **2 to 0.5,** and rank **3 to 1.0**

EXAMPLE - DISSIMILARITY OF ORDINAL DATA

- For **step 3**, we can use, the Euclidean distance, which results in the following dissimilarity matrix:

Test 2	Ranking	Normalization
Excellent	3	1
Fair	1	0
Good	2	0.5
Excellent	3	1

$$\begin{bmatrix} 0 & & & \\ 1.0 & 0 & & \\ 0.5 & 0.5 & 0 & \\ 0 & 1.0 & 0.5 & 0 \end{bmatrix}$$

EXAMPLE - DISSIMILARITY OF ORDINAL DATA

- Therefore, objects 1 and 2 are the most dissimilar, as are objects 2 and 4 (i.e., d(2,1) = 1.0 and d(4,2) = 1.0)
- This makes intuitive sense since objects 1 and 4 are both excellent
- Object 2 is fair, which is at the opposite end of the range of values for test-2
- Similarity values for ordinal attributes can be interpreted from dissimilarity as sim(i, j) = 1 – d(i, j)

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

- A database may contain all attribute types viz. **nominal, symmetric binary, asymmetric binary, numeric, ordinal**

- Suppose that the **data set** contains **p attributes** of **mixed** type. The dissimilarity d(i, j) between objects i and j is defined as

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

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

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

- where the indicator $\delta_{ij}^{(f)} = 0$ if either
  - ✓ x_{if} or x_{jf} is missing
  - ✓ x_{if} or x_{jf} = 0 and f is asymmetric binary
- Otherwise $\delta_{ij}^{(f)} = 1$

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

- The contribution of attribute f to the dissimilarity between i and j (i.e., $d_{ij}^{(f)}$) is computed dependent on its type:

  - If f is numeric: $d_{ij}^{(f)} = \frac{|x_{if} - x_{jf}|}{max_h x_{hf} - min_h x_{hf}}$, where h runs over all nonmissing objects for attribute f.
  - If f is nominal or binary: $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$; otherwise, $d_{ij}^{(f)} = 1$.
  - If f is ordinal: compute the ranks r_{if} and $z_{if} = \frac{r_{if} - 1}{M_f - 1}$, and treat z_{if} as numeric.

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

Object Identifier	test-1 (nominal)	test-2 (ordinal)	test-3 (numeric)
1	code A	excellent	45
2	code B	fair	22
3	code C	good	64
4	code A	excellent	28

- We can use the dissimilarity matrices obtained for test-1 and test-2 attributes

- We need to compute the dissimilarity matrix for the third attribute i.e. test-3 (which is numeric)

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

- We need to compute the dissimilarity matrix for the third attribute test-3 (which is numeric) i.e. $d_{ij}^{(3)}$

- Let $max_h x_h = 64$ and $min_h x_h = 22$

- Difference between the two is used in above equation to normalize the values of the dissimilarity matrix i.e. 64-22 = 42

- The resulting dissimilarity matrix for test-3 is

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

Object Identifier	test-1 (nominal)	test-2 (ordinal)	test-3 (numeric)
1	code A	excellent	45
2	code B	fair	22
3	code C	good	64
4	code A	excellent	28

|45-45|/42 = 0			
|22-45| / 42 = 0.55	|22-22| / 42 = 0		
|64-45| / 42 = 0.45	|64-22| / 42 = 1	|64-64| / 42 = 0	
|28-45| / 42 = 0.4	|28-22| / 42 = 0.14	|28-64| / 42 = 0.86	|28-28| / 42 = 0

$$\begin{bmatrix} 0 & & & \\ 0.55 & 0 & & \\ 0.45 & 1.00 & 0 & \\ 0.40 & 0.14 & 0.86 & 0 \end{bmatrix}$$

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

Object Identifier	test-1 (nominal)	test-2 (ordinal)	test-3 (numeric)
1	code A	excellent	45
2	code B	fair	22
3	code C	good	64
4	code A	excellent	28

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

$$\begin{bmatrix} 0 & & & \\ 1.0 & 0 & & \\ 0.5 & 0.5 & 0 & \\ 0 & 1.0 & 0.5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & & & \\ 0.55 & 0 & & \\ 0.45 & 1.00 & 0 & \\ 0.40 & 0.14 & 0.86 & 0 \end{bmatrix}$$

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

- We can now use the dissimilarity matrices for the three attributes in our computation of given equation. The indicator $\delta_{ij}^{(f)} = 1$ for each of the 3 attributes, f

- We get, for eg. d(3,1) = [1(1) + 1(0.50) + 1(0.45)] / 3 = 0.65

- Resulting dissimilarity matrix obtained for the data described by 3 attributes of mixed types is

$$\begin{bmatrix} 0 & & & \\ 0.85 & 0 & & \\ 0.65 & 0.83 & 0 & \\ 0.13 & 0.71 & 0.79 & 0 \end{bmatrix}$$

DISSIMILARITY FOR ATTRIBUTES OF MIXED TYPE

- From the data, the objects 1 and 4 are the most similar, based on their values for *test*-1 and *test*-2
- This is confirmed by the dissimilarity matrix, where d(4, 1) is the lowest value for any pair of different objects
- Similarly, the matrix indicates that objects 1 and 2 are the least similar

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

- Thus, each **document** is an **object** represented by what is called a **term-frequency vector**

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

COSINE SIMILARITY

- **Term-frequency** vectors are typically very **long** and **sparse** (i.e., they have many 0 values)
- Applications using such structures include **information retrieval**, text **document clustering**, **biological taxonomy**, and **gene feature mapping**
- Traditional **distance measures do not work well** for such **sparse numeric** data
- Need a **measure** that will **focus** on the **words** that the **two documents** do have in **common**, and the **occurrence frequency** of such **words**

COSINE SIMILARITY

- **Cosine similarity** is a **measure** of **similarity** that can be used to **compare documents** or, say, give a ranking of documents with respect to a given vector of query words

- Let **x** and **y** be two **vectors** for **comparison**. Using the cosine measure as a similarity function, we have

$$\textbf{sim(x,y)} = \frac{\boldsymbol{x} \,.\boldsymbol{y}}{||\boldsymbol{x}||\,||\boldsymbol{y}||}$$

Where, ||x|| is Euclidean norm of vector $x = (x_1, x_2, \ldots, x_p)$ defined as $\sqrt{x_1^2 + x_2^2 + \cdots + x_2^p}$ i.e. the length of the vector

Similarly, ||y|| is Euclidean norm of vector y

COSINE SIMILARITY

- The **measure** computes the **cosine** of the **angle** between vectors **x and y**
- A **cosine** value of **0** means that the two **vectors** are at **90 degrees** to each other (orthogonal) and have **no match**
- The **closer** the **cosine** value to **1**, the **smaller** the **angle** and **greater** the **match** between **vectors**

EXAMPLE - COSINE SIMILARITY

- 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)

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

$\|\boldsymbol{d_1}\|$= (5*5+0*0+3*3+0*0+2*2+0*0+0*0+2*2+0*0+0*0)$^{\mathbf{0.5}}$=(42)$^{\mathbf{0.5}}$ = **6.481**

$\|\boldsymbol{d_2}\|$= (3*3+0*0+2*2+0*0+1*1+1*1+0*0+1*1+0*0+1*1)$^{\mathbf{0.5}}$=(17)$^{\mathbf{0.5}}$ = **4.12**

cos(d_1, d_2) = 0.94

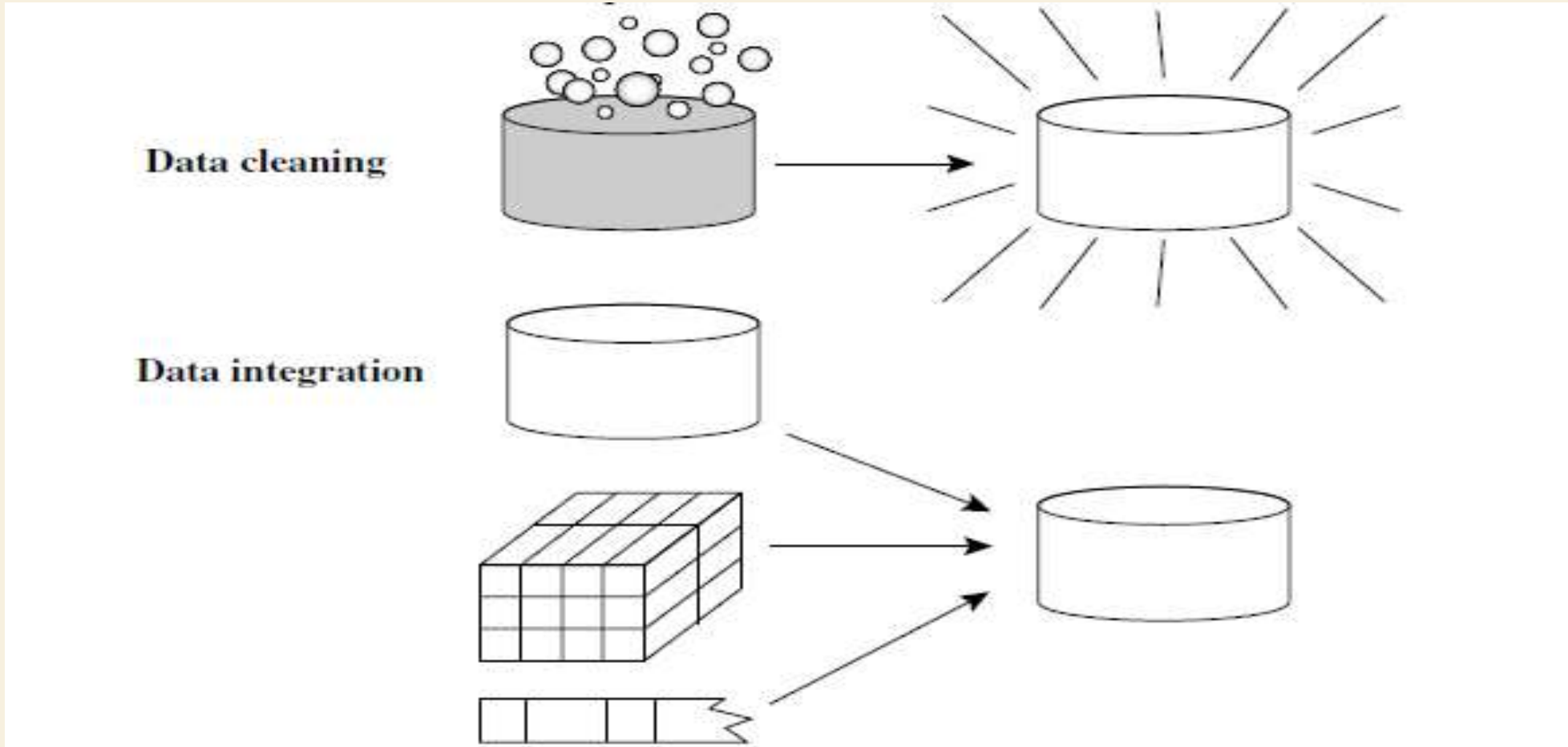
WHY PREPROCESSING?

- Data quality is determined by **accuracy**, **completeness**, **consistency**, **timeliness**, **believability** and **interpretability**

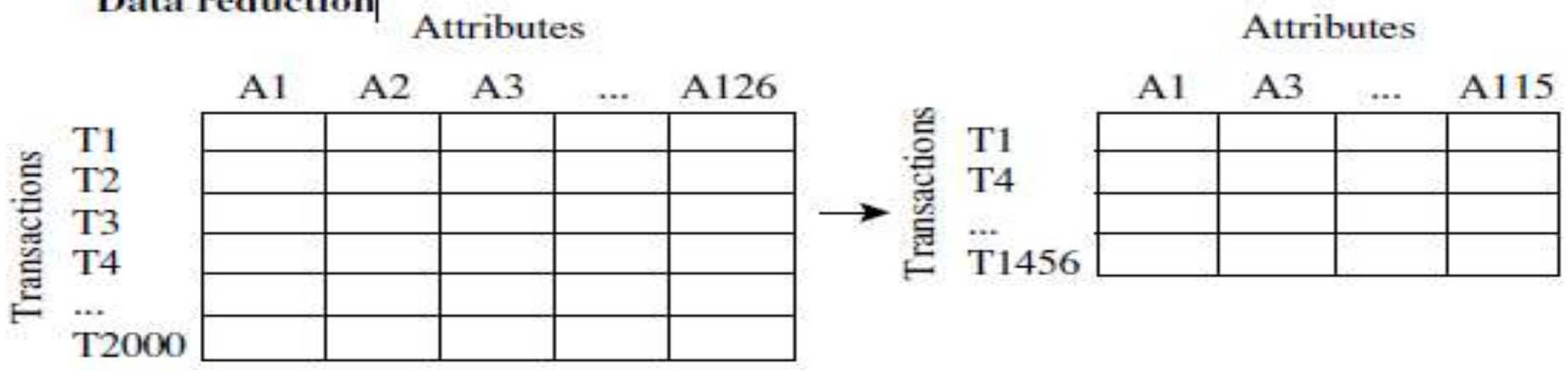
- To ensure data quality preprocessing is required as the real world data is:
  - **Incomplete** (lacking attributes of interest, or containing only aggregate data)
  - **Noisy** (containing errors or outlier values which deviate from the expected or meaningless information or unstructured data)
  - **Inconsistent** (same data is stored in different formats in two files)

MAJOR TASKS IN DATA PREPROCESSING

- **Data cleaning** – filling in missing values, smoothing noisy data, identifying or removing outliers, resolving inconsistencies
- **Data integration –** integrating data from multiple databases, data cubes or files
- **Data transformation –** normalization, aggregation, generalization
- **Data reduction –** removing irrelevant attributes, attribute construction
- **Data discretization** – replacing numerical attributes with nominal ones, concept hierarchy generation



Data reduction



DATA CLEANING

- Data cleaning routines attempt to:
  - **fill** in **missing values**
  - **smooth out noise** while **identifying outliers**
  - **correct inconsistencies** in the data

DATA CLEANING – MISSING VALUES

- Consider a dataset in which many tuples have no recorded value for several attributes such as **customer income**

- Methods for filling in the missing values for the attribute are discussed as follows:

DATA CLEANING – MISSING VALUES

1) Ignore the tuple:

- ✓Usually done when the **class label** is **missing** (assuming the mining task involves **classification**)

- ✓Method is **not** very **effective**, unless the **tuple** contains **several attributes** with **missing values**. It is **poor** when the **percentage** of **missing values** per **attribute varies considerably**

- ✓By ignoring the tuple, we do not make use of the remaining attributes' values in the tuple

- ✓Such data could have been useful to the task at hand

DATA CLEANING – MISSING VALUES

2) Fill in the missing value manually:

- ✓**Time** consuming
- ✓May **not** be **feasible** given a **large data set** with many **missing** values

DATA CLEANING – MISSING VALUES

3) Use a global constant to fill in the missing value:

✓**Replace** all **missing attribute values** by the same constant such as a label like ***"Unknown"*** or ∞

✓ If missing values are replaced by ***"Unknown,"*** then the **mining program** may **mistakenly** think that they form an **interesting concept**, since they all have a **value in common** that of *"Unknown."*

✓Hence, although this method is **simple**, it is **not foolproof**

DATA CLEANING – MISSING VALUES

4) Use a measure of central tendency for the attribute (e.g., the mean or median) to fill in the missing value:

- ✓Use measures of central tendency, which indicate the **“middle”** value of a **data distribution**

- ✓For **normal data distributions**, the **mean** can be used, while **skewed data distribution** should employ the **median**

DATA CLEANING – MISSING VALUES

5) Use the attribute mean or median for all samples belonging to the same class as the given tuple:

- ✓For example, if **classifying** customers according to ***credit risk***, we may **replace** the **missing value** with the **mean *income*** value for customers in the **same credit risk category** as that of the **given tuple**

- ✓If the **data distribution** for a given class is **skewed**, the **median** value is a better choice

DATA CLEANING – MISSING VALUES

6) Use the most probable value to fill in the missing value:

- ✓This may be determined with **regression**, inference-based tools using a **Bayesian** formalism, or **decision tree induction**

- ✓For example, using the other customer attributes in your data set, you may construct a decision tree to predict the missing values for income

DATA CLEANING – MISSING VALUES

- **Method 6**, is a **popular** strategy as in comparison to the other methods, it uses the **most information** from the **present data** to **predict missing values**

- By considering the other attributes' values in its **estimation** of the **missing** value for ***income***, there is a **greater** chance that the **relationships** between ***income*** and the other **attributes** are **preserved**

DATA CLEANING – MISSING VALUES

- **Missing values** may **not** always **imply** an **error** in the data
- Eg: when applying for credit card, candidates may be asked to provide their driver's license number. Candidates who do not have a driver's license may leave this field blank
- **Forms** should **allow** respondents to specify values such as **"not applicable"**
- Each **attribute** should have **rules** regarding the **null condition** that specify whether or not nulls are **allowed** and/or how such values should be **handled or transformed**
- **Good database** and **data entry procedure design** should help **minimize** the number of **missing values** or errors

DATA CLEANING – NOISY DATA

- Noise is a **random error** or **variance** in a measured variable

- **Statistical** description techniques (e.g., boxplots and scatter plots), and methods of data visualization can be used to **identify outliers**, which represent noise

- **Types of noise in data:**
  - Unknown encoding - Gender : E
  - Out of range values – Temperature : 1006, Age : 250

DATA CLEANING – NOISY DATA

- **Data smoothening** techniques used to **handle noisy data** are as follows:

✓**Binning**

- Equal-width (distance) partitioning
- Equal-depth (frequency) partitioning or Equal-height partitioning

✓**Regression**

- Linear Regression
- Multiple Regression

✓**Outlier analysis by clustering**

DATA CLEANING – NOISY DATA - BINNING

- Binning methods **smooth** a **sorted data value** by consulting its **neighborhood** i.e. the values around it and hence perform **local smoothing**

- **Sorted values** are **distributed** into a number of **"buckets,"** or ***bins***

- **Approaches of binning:**

❑ Equal-width (distance partitioning)

❑ Equal-depth (frequency) partitioning or Equal-height partitioning

DATA CLEANING – NOISY DATA - BINNING

- Equal-width (distance partitioning)

✓ Divides the range into N intervals of equal size

✓ bin width = (max value – min value) / N

✓Eg: For a set of observed values in the range of 0-100, the data can be placed into 5 bins as follows:

✓Bin width = (100 – 0) / 5 = 20

✓Bins formed are: [0-20], (20-40], (40-60], (60-80], (80-100]

DATA CLEANING – NOISY DATA - BINNING

❑ Equal-depth (frequency) partitioning or Equal-height partitioning

✓ Entire range is divided into N intervals, each containing approximately the same number of samples

✓ Eg: Price in INR – 4,8,9,15,21,21,24,25,26,28,29,32

✓Partition into equal depth bins taking N as 3

- Bin 1 : 4, 8, 9, 15
- Bin 2 : 21, 21, 24, 25
- Bin 3 : 26, 28, 29, 32

DATA CLEANING – NOISY DATA - BINNING

- **Types of Binning:**

✓*Smoothing by means* – **each value** in a **bin** is **replaced** by **mean** value of **bin**

✓*Smoothing by medians* **- each bin value** is **replaced** by the **bin median**

✓*Smoothing by boundaries* **- minimum** and **maximum values** in a given **bin** are identified as ***bin boundaries***. Each **bin value** is then **replaced** by the **closest boundary value**

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

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

DATA CLEANING – NOISY DATA - REGRESSION

- **Regression** is a **statistical measure** used to **determine** the **strength** of **relationship** between **one dependent variable** (Y) and a **series of independent changing variables**

- **Smooth** by **fitting** the data into **regression functions**

- Can also be used to **fill missing values** of **attributes**

DATA CLEANING – NOISY DATA - REGRESSION

- ***Linear regression*** involves finding the **"best" line** to **fit two attributes** (or variables) so that one attribute can be used to predict the other

- ***Multiple linear regression*** is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface

Simple Linear Regression

- It is a **statistical** method that allows us to **summarize** and study **relationships** between **two continuous (quantitative) variables:**
  - ✓**One variable**, denoted **x**, is regarded as the **predictor**, explanatory, or **independent variable**
  - ✓The **other variable**, denoted **y**, is regarded as **response**, **outcome**, or **dependent variable**
- Eg: Temperature and sales of ice cream

Simple Linear Regression – Best Fitting Line

- Equation for the best fitting line is: $Y = a + bX + \epsilon$

- Where,
  - Y is the response or the **dependent** variable
  - X is the predictor or the **independent** variable
  - b is the **slope** of the line
  - a is the **intercept** i.e. the value of y when x=0
  - ϵ is **regression** residual (difference between the **observed** value of the **dependent** variable y and the **predicted** value)

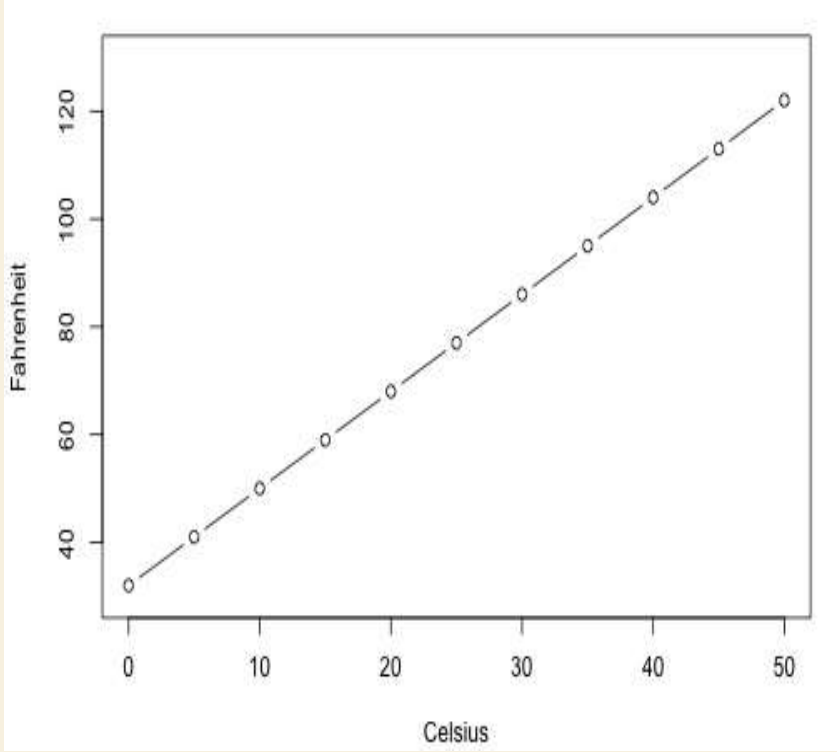
Simple Linear Regression

- Deals with **statistical** relationships rather than **deterministic** relationships

- **Deterministic relationship examples:**
  - Ohm's Law: I = V/r, where V = voltage applied, r = resistance, and I = current.
  - Circumference = $\pi \times$ diameter

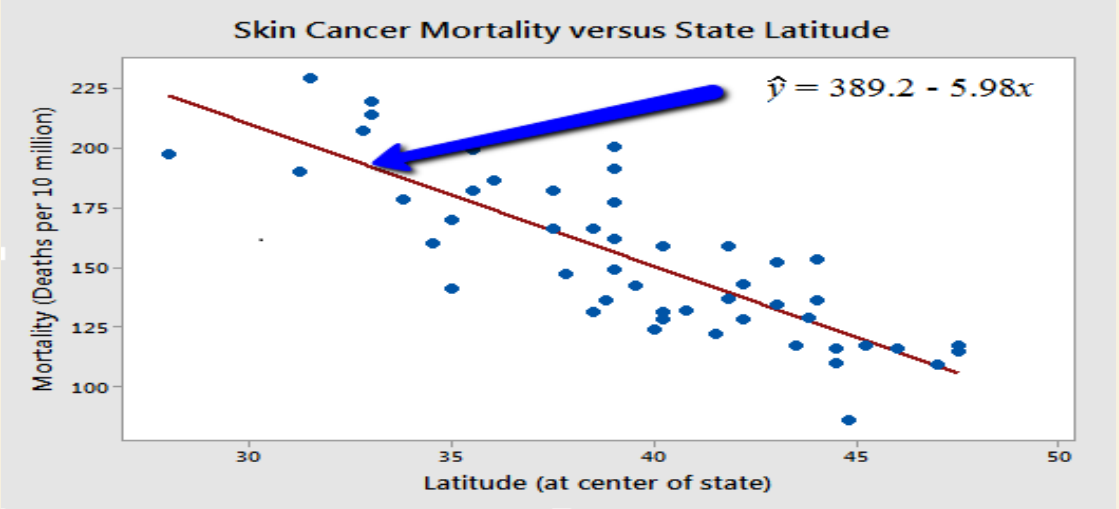
- For each of these **deterministic relationships**, the **equation exactly describes** the **relationship between the two variables**

- Knowing the value in Celsius we can determine the temperature in Fahrenheit exactly



Simple Linear Regression

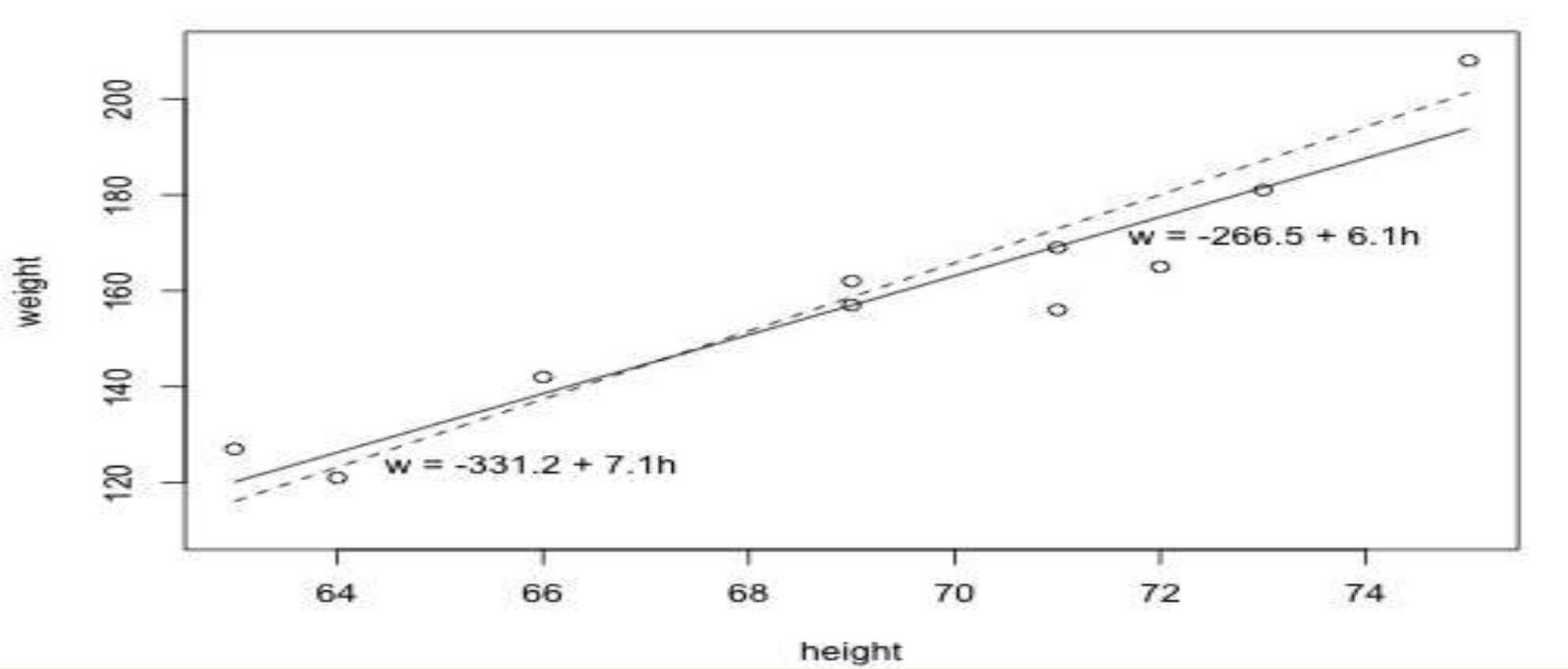
- **Statistical relationship:** Relationship between the variables is not perfect



- Higher latitudes - less exposed to harmful rays of the sun, and therefore, less risk of death due to skin cancer
- **Scatter** plot supports such a hypothesis. There appears to be a **negative linear** relationship between latitude and mortality due to skin cancer, but the **relationship** is **not perfect**
- Plot exhibits some "**trend**," but it also exhibits some "**scatter**." Therefore, it is a **statistical** relationship, not a **deterministic** one

Simple Linear Regression – Best Fitting Line

- Which line — the solid line or the dashed line — do you think best summarizes the trend between height and weight?



- Solid line i.e. w = -266.53 + 6.1376h seems to be more precise

Multiple Linear Regression (MLR)

- MLR has **atleast two** or **more predictors** unlike **SLR** which has a **single** predictor

- With **large number** of **predictors**, it is **more efficient** to use **matrices** to define the **regression model** and the subsequent analyses

Multiple Linear Regression (MLR)

- General form of multiple regression is given by:

$$Y = a + b_1X_1 + b_2X_2 + \ldots + b_{p-1}X_{p-1} + \epsilon$$

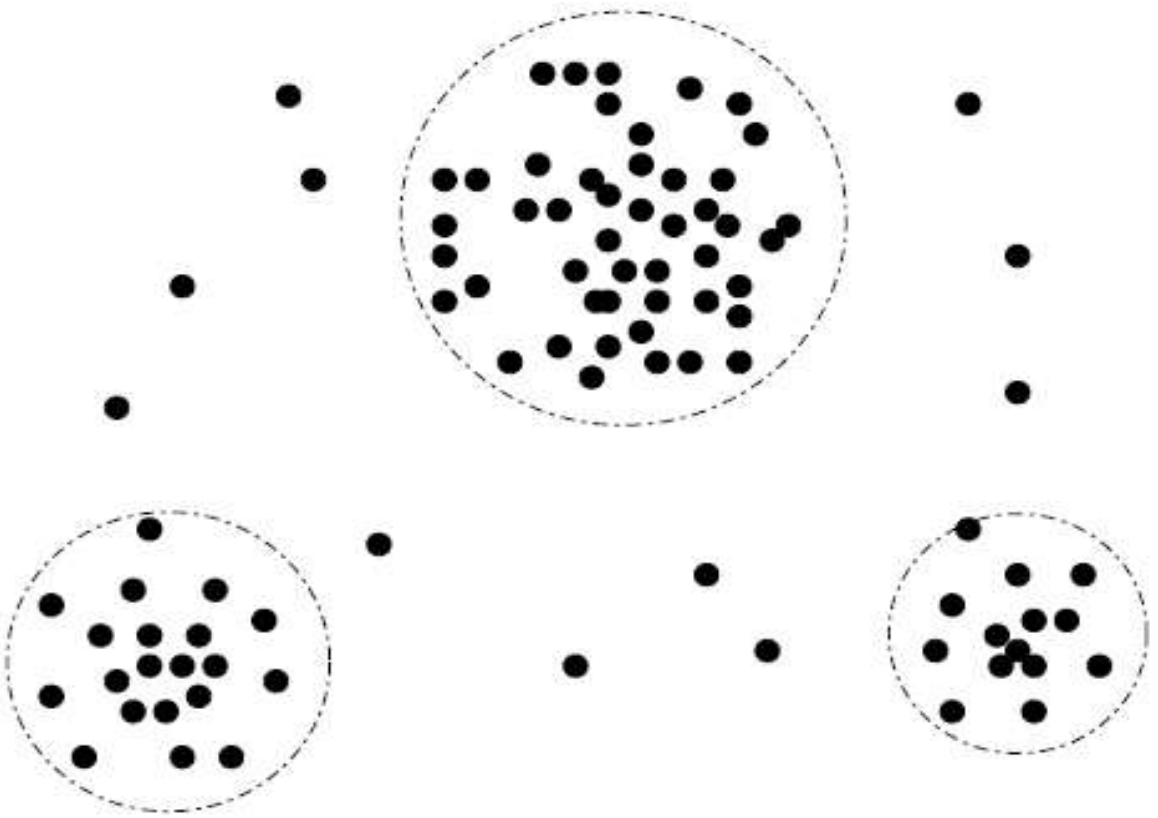
- Where,
  - Y – variable that is being predicted (**dependent** variable)
  - X – variable that is used to predict y (**independent** variable)
  - a is the **intercept**
  - $b_1 \ldots b_{p-1}$ are the **slopes**
  - ϵ is **regression residual** (difference between the observed value of the dependent variable y and the predicted value)

DATA CLEANING – NOISY DATA – OUTLIER ANALYSIS BY CLUSTERING

- Outliers may be detected by clustering, for example, where similar values are organized into groups, or “clusters”

- Values that fall outside of the set of clusters may be considered as outliers

DATA CLEANING – NOISY DATA – OUTLIER ANALYSIS BY CLUSTERING



A 2-D customer data plot with respect to customer locations in a city, showing three data clusters. Outliers may be detected as values that fall outside of the cluster sets.

DATA CLEANING – INCONSISTENT DATA

- Data with different formats:
  - Inconsistent entries – DoB : 20-Jan-1995, Age : 28
  - Inconsistent formats – DoB : 20-Jan-1995, DoJ : 12/2/2000

DATA INTEGRATION - ISSUES

- Data integration is the **merging** of **data** from **multiple data sources** like databases, data cubes and flat files

- Schema Integration

✓**Entity identification problem:** identify equivalent real world entities from multiple data sources

✓Eg: How can one be sure that customer_id in one database and cust_number in another refer to the same attribute i.e. A.cust-id ≡ B.cust-#

✓*Solution:* **Integrate metadata** from different sources

DATA INTEGRATION - ISSUES

- ✓Metadata for each attribute includes the **name, meaning, data type**, and **range of values** permitted for the attribute, and **null rules** for handling **blank, zero, or null values**

- ✓It can be used to **avoid errors** in **schema integration**. It may also be used to help **transform** the data (e.g., where data **codes** for **account_*type*** in one **database** may be *"**S**"* and *"**C**"* but ***1*** and ***2*** in another)

- ✓This step also relates to **data cleaning**

DATA INTEGRATION - ISSUES

- ✓When **matching attributes** from **one database to another** during integration, special attention must be paid to the **structure** (attribute **functional dependencies**) of the data

- ✓Eg: In one system, a **discount** may be **applied** to the **overall order**, whereas in another system it is **applied to each individual item within the order**

- ✓If this is not caught before integration, **items** in the target system may be **improperly discounted**

DATA INTEGRATION - ISSUES

- Redundancy and correlation analysis

✓Data redundancy occurs when data is integrated from multiple sources

✓**Object identification:** Same attribute or object may have different names in different databases, eg: roll_no, sap_id, enrol_no

✓**Derivable data:** One attribute may be a “derived” attribute from another attribute or set of attributes, eg: age, years of experience

✓**Redundant attributes** can be detected by **correlation analysis** and **covariance analysis**

DATA INTEGRATION - ISSUES

- **χ^2 Test**

✓Used for **nominal** data to determine **correlation** relationship between **two attributes**

✓Suppose **A** has **"c"** distinct values, namely $\mathbf{a_1, a_2, \ldots, a_c}$ and **B** has **"r"** distinct values, namely $\mathbf{b_1, b_2, \ldots, b_r}$

✓Data tuples described by A and B can be represented by a **contingency** table with **"c" values of A** making up the **columns** and **"r" values of B** making up the **rows**

DATA INTEGRATION - ISSUES

- **χ^2 Test**

✓Let (A_i , B_j) denote the joint event that attribute A takes on value a_i and attribute B takes on value b_j, i.e., where A = a_i , B = b_j

✓Each and every possible (A_i , B_j) joint event has its own cell (or slot) in the table

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

where, o_{ij} is the observed frequency (actual count) of the joint event (A_i , B_j)

e_{ij} is the expected frequency of (A_i , B_j)

DATA INTEGRATION - ISSUES

- **χ^2 Test**

✓ e_{ij} is calculated as $$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n}$$

✓ Where,

- n is the number of data tuples
- count($A=a_i$) is total number of tuples having value a_i for A
- count($B=b_j$) is total number of tuples having value b_j for B

✓ The χ^2 statistic **tests** the **hypothesis that A and B are independent** i.e. there is no correlation between them

DATA INTEGRATION - ISSUES

- χ^2 **Test**

✓Based on the **significance level** given, with **(r-1)*(c-1) degrees of freedom**, we find the χ^2 value from the Chi-Square distribution table

	Probability level (alpha)					
df	**0.5**	**0.10**	**0.05**	**0.02**	**0.01**	**0.001**
1	0.455	2.706	3.841	5.412	6.635	10.827
2	1.386	4.605	5.991	7.824	9.210	13.815
3	2.366	6.251	7.815	9.837	11.345	16.268
4	3.357	7.779	9.488	11.668	13.277	18.465
5	4.351	9.236	11.070	13.388	15.086	20.517

DATA INTEGRATION - ISSUES

- χ^2 **Test**

✓ If the **calculated value** is **greater** than the **probability value** obtained from **Chi-Square distribution table**, we **reject** the **hypothesis** that the **attributes** considered are **independent** and conclude that the **two attributes** are **correlated**

❑ χ^2 Test Example

✓Consider the contingency table below that contains the data classified by gender (Male or Female) and buying preferences (Young, Middle, Old). Level of significance is 0.05. Interpret the result.

	Buying Preferences			Row total
Gender	**Young age**	**Middle age**	**Old age**	
Male	200	150	50	400
Female	250	300	50	600
Column total	450	450	100	1000

❑ χ^2 Test Example

✓State the hypothesis:

1. H_0: Null hypothesis: Gender and buying preferences are independent
2. H_a: Alternative hypothesis: Gender and buying preferences are not independent

✓Analyze the sample data

Degrees of freedom DF = (r – 1) * (c – 1) = (2 – 1) * (3 – 1) = 2

where, r – no. of levels for row & c - no. of levels for column

❑ χ^2 Test Example

✓By using the contingency ta

	Buying Preferences			Row total
Gender	**Young age**	**Middle age**	**Old age**	
Male	200	150	50	400
Female	250	300	50	600
Column total	450	450	100	1000

	Young age	Middle age	Old age	total
Male (O)	200	150	50	400
Male(E)	(400 * 450) / 1000 = 180	(400 * 450) / 100 = 180	(400 * 100) / 1000 = 40	400
Male(O-E)	20	– 30	10	
Male$(O-E)^2$	400	900	100	
Male$(O-E)^2/E$	2.22	5	2.5	
Female(O)	250	300	50	600
Female(E)	(600 * 450) / 1000 = 270	(600 * 450) / 1000 = 270	(600 * 100) / 1000 =60	600
Female(O-E)	– 20	30	– 10	
Female$(O-E)^2$	400	900	100	
Female $(O-E)^2/E$	1.48	3.33	1.67	

❑ χ^2 Test Example

✓Calculating

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

✓ χ^2 = 2.22 + 5 + 2.5 + 1.48 + 3.33 + 1.67 = **16.2**

✓Based on the significance level given, with (r-1)*(c-1) degrees of freedom, we find the χ^2 value from the Chi-Square distribution table which is **5.991**

✓**Calculated value** is **greater** than the **probability value** obtained from **Chi-Square distribution** table, so we **reject** the hypothesis that the **attributes are independent**. There is **correlation** between **gender** and **buying preferences**.

DATA INTEGRATION - ISSUES

- **Correlation coefficient for numeric data (Pearson's product moment coefficient)**

✓Correlation ($r_{A,B}$) between two attributes A and B, is calculated as

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

✓where,

- n is the number of data tuples
- a_i and b_i are the respective values of A and B in tuple i,
- $\overline{A}$ and $\overline{B}$ are the respective mean values of A and B,
- σ_A and σ_B are the respective standard deviations of A and B
- $\sum(a_i b_i)$ is the sum of the AB cross product

DATA INTEGRATION - ISSUES

❑ **Correlation coefficient for numeric data** **Note:** $-1 \leq r_{A,B} \leq +1$

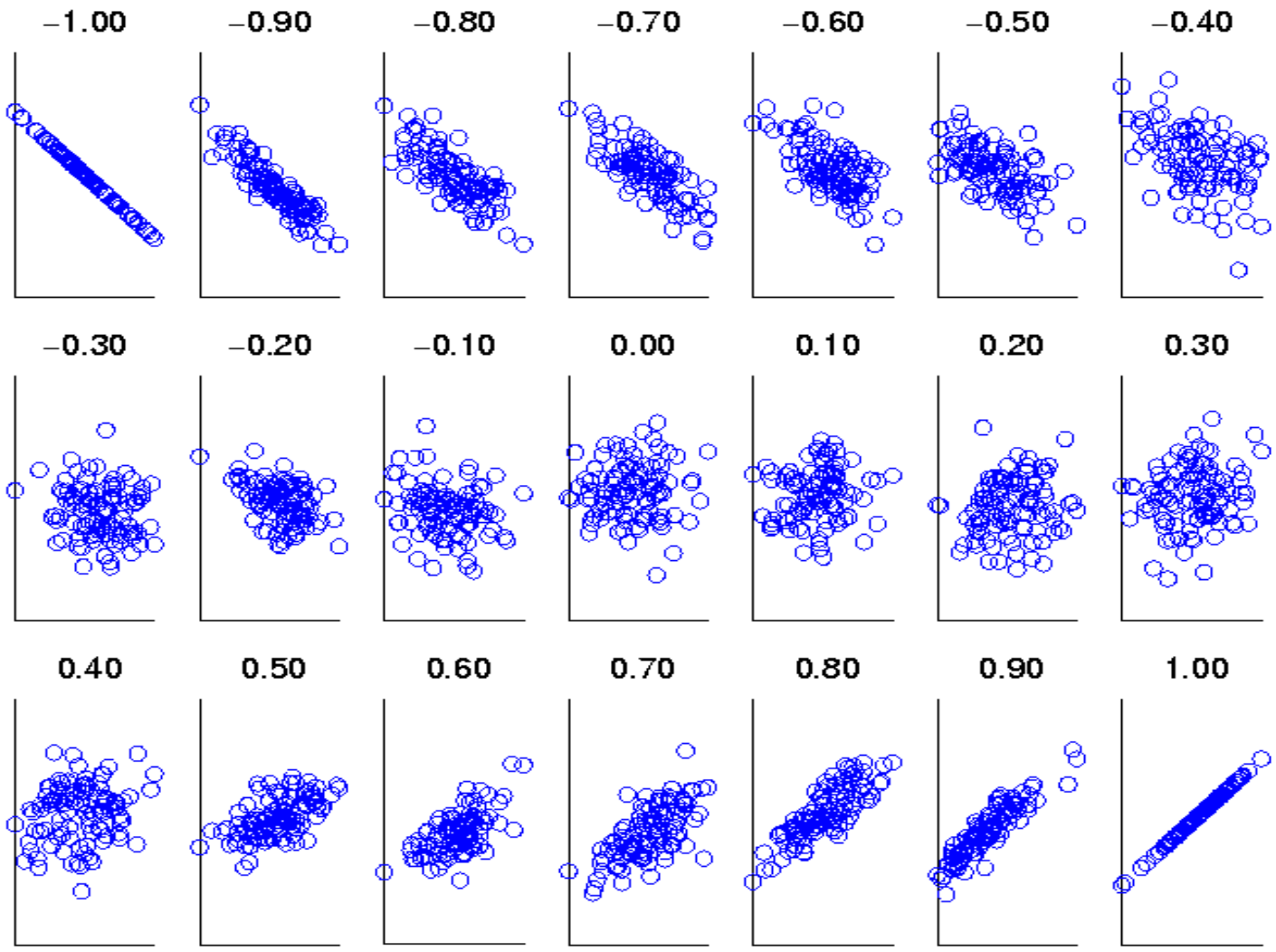
- If $\boldsymbol{r_{A,B} > 0}$, then *A* and *B* are ***positively correlated***, meaning that the values of ***A* increases as the values of *B* increases**

- **Higher** the value, **stronger the correlation**. Hence, a higher value may indicate that *A* (or *B*) may be **removed as a redundancy**

- If $\boldsymbol{r_{A,B} = 0}$, then *A* and *B* are ***independent*** and there is **no correlation** between them

- If $\boldsymbol{r_{A,B} < 0}$, then *A* and *B* are ***negatively correlated***, where the values of one **attribute increases** as the values of the other **attribute decreases**

VISUALLY EVALUATING CORRELATION



Scatter plots showing the similarity from –1 to 1.

DATA INTEGRATION - ISSUES

- Tuple Duplication

✓In addition to detecting redundancies between attributes, **duplication** should also be **detected** at the **tuple** level (e.g., where there are two or more identical tuples for a given unique data entry case)

✓**Inconsistencies** often arise between various **duplicates**, due to **inaccurate data entry** or **updating some but not all data occurrences**

DATA INTEGRATION - ISSUES

- Detecting and resolving data value conflicts

✓As data is collected from multiple sources, **attribute values** may be **different** for **same real world entity**

✓Possible reasons include different **representations, different scales**

✓Eg 1: weight attribute may be stored in metric units in one system and British imperial units in another

✓Eg 2: For a hotel chain, the price of rooms in different cities / countries may involve not only different currencies but also different services (e.g., free breakfast) and taxes

DATA INTEGRATION - ISSUES

- Detecting and resolving data value conflicts

✓Eg 3: When exchanging information between universities, for example, each university may have its own curriculum and grading scheme

✓One university may adopt a quarter system, offer three courses on database systems, and assign grades from A+ to F, whereas another may adopt a semester system, offer two courses on databases, and assign grades from 1 to 10

✓This makes information exchange difficult

DATA INTEGRATION - ISSUES

- Detecting and resolving data value conflicts

✓**Attributes** may also **differ** on the **abstraction level**, where an attribute in one system is recorded at, say, a lower abstraction level than the "same" attribute in another

✓Eg, the total sales in one database may refer to one branch of "XYZ Electronics", while an attribute of the same name in another database may refer to the total sales for All Electronics stores in a given region

DATA REDUCTION

- **Why data reduction?** — A database/data warehouse may store **terabytes** of data

- **Complex data analysis** may take a very **long time** to run on the complete data set making the analysis impractical or **infeasible**

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

DATA REDUCTION

- **Data reduction strategies**

✓Data Cube Aggregation

✓Dimensionality reduction, e.g., remove unimportant attributes

✓Numerosity reduction (some simply call it: Data Reduction)

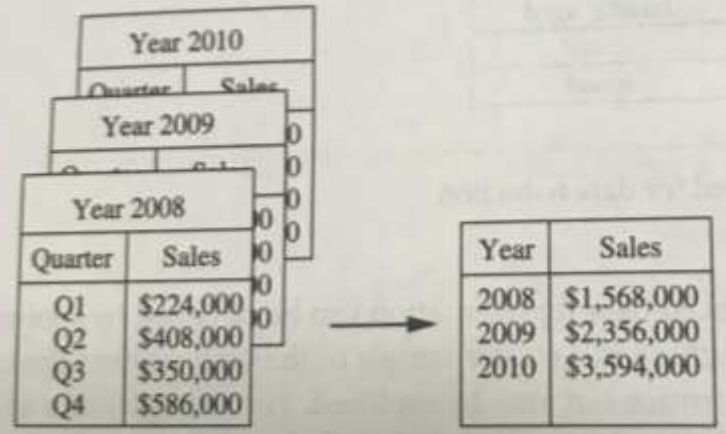
✓Data compression

DATA REDUCTION - DATA CUBE AGGREGATION

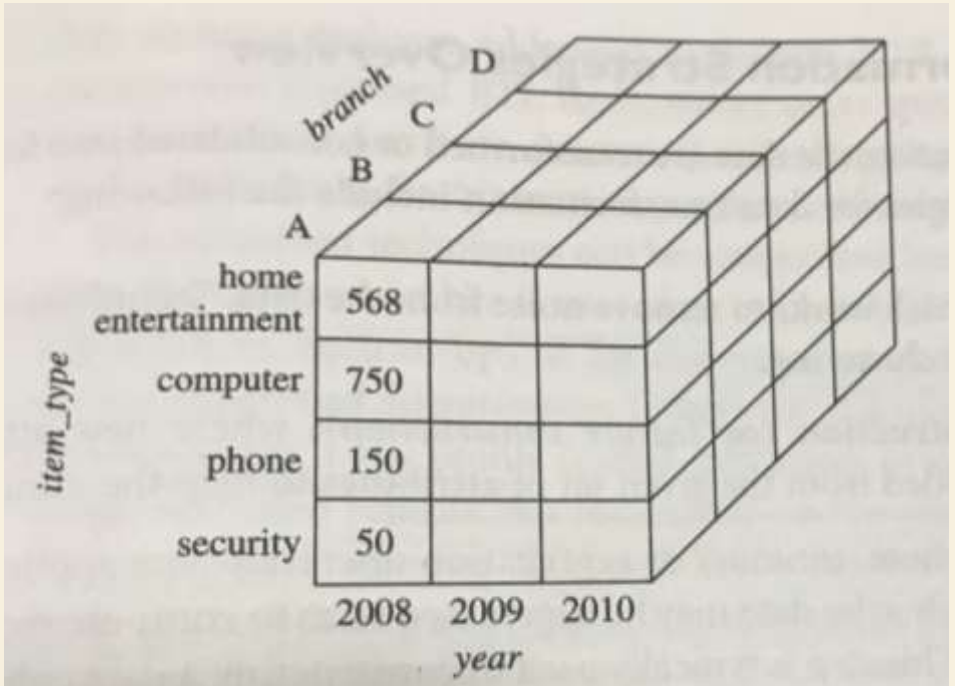
- ✓Combining two or more attributes (or objects) into a single attribute (or object)

- ✓Eg: Total annual sales of product is aggregated quarterly with the help of data cube

- ✓Purpose
  - Reduce the number of attributes or objects
  - Change of scale
  - Cities aggregated into regions, states, countries, etc.
  - Aggregated data tends to have less variability



e 3.10 Sales data for a given branch of *AllElectronics* for the years 2008 through 2010. On the *lef* the sales are shown per quarter. On the *right*, the data are aggregated to provide the annu sales.



A data cube for sales at *AllElectronics*.

DATA REDUCTION - DIMENSIONALITY REDUCTION

- ✓Data sets may contain large number of attributes that may be irrelevant or redundant
- ✓Dimensionality reduction removes the unwanted attributes resulting in a dataset that is smaller in size
- ✓Helps in reducing time and space complexity required by data mining technique
- ✓Data visualization becomes easy
- ✓Involves deleting inappropriate features or reducing noisy data

DATA REDUCTION - DIMENSIONALITY REDUCTION

- **Attribute subset selection techniques**

✓Process in which minimum set of attributes are selected such that their distribution represents the same as original data distribution considering all attributes

✓Different attribute subset selection techniques

- Stepwise forward selection
- Stepwise backward elimination
- Combination of forward selection and backward elimination
- Decision tree induction

DATA REDUCTION - DIMENSIONALITY REDUCTION

- **Attribute subset selection techniques -** Stepwise forward selection

✓Begins with an empty set of attributes as the reduced set (temporarily)

✓Next the best among the original attributes is determined and added to the reduced set

✓With each iteration the best among the remaining original attributes is added to the reduced set

DATA REDUCTION - DIMENSIONALITY REDUCTION

- **Attribute subset selection techniques -** Stepwise backward elimination

✓Begins with full set of attributes

✓In each step it finds the worst attribute and removes it from the set

- **Attribute subset selection techniques –** Combination of stepwise forward selection & stepwise backward elimination

✓First two methods are combined and the procedure at every step selects the best attribute and removes the worst

- For all of the above techniques, stopping criteria is different and it requires a threshold on the measure used to stop the attribute selection process

DATA REDUCTION - DIMENSIONALITY REDUCTION

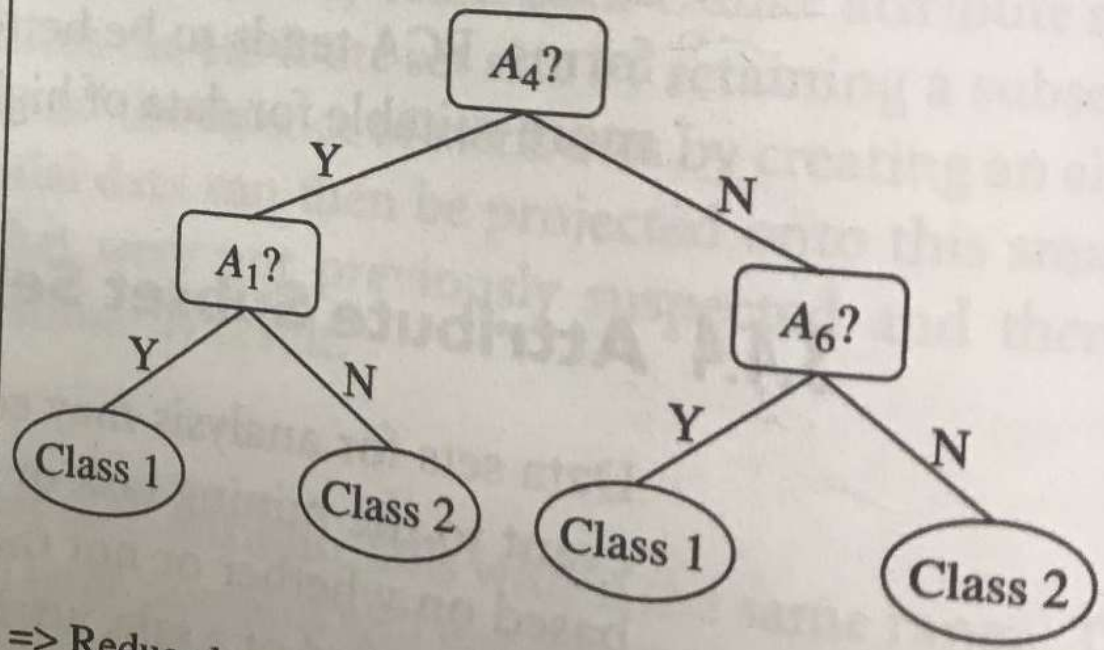
- **Attribute subset selection techniques –** Decision tree induction

✓ID3, C4.5 intended for classification

✓Constructs a tree like structure

✓Decision tree is a tree in which:

- Each internal node tests an attribute
- Each branch corresponds to attribute value
- Each leaf node assigns a classification

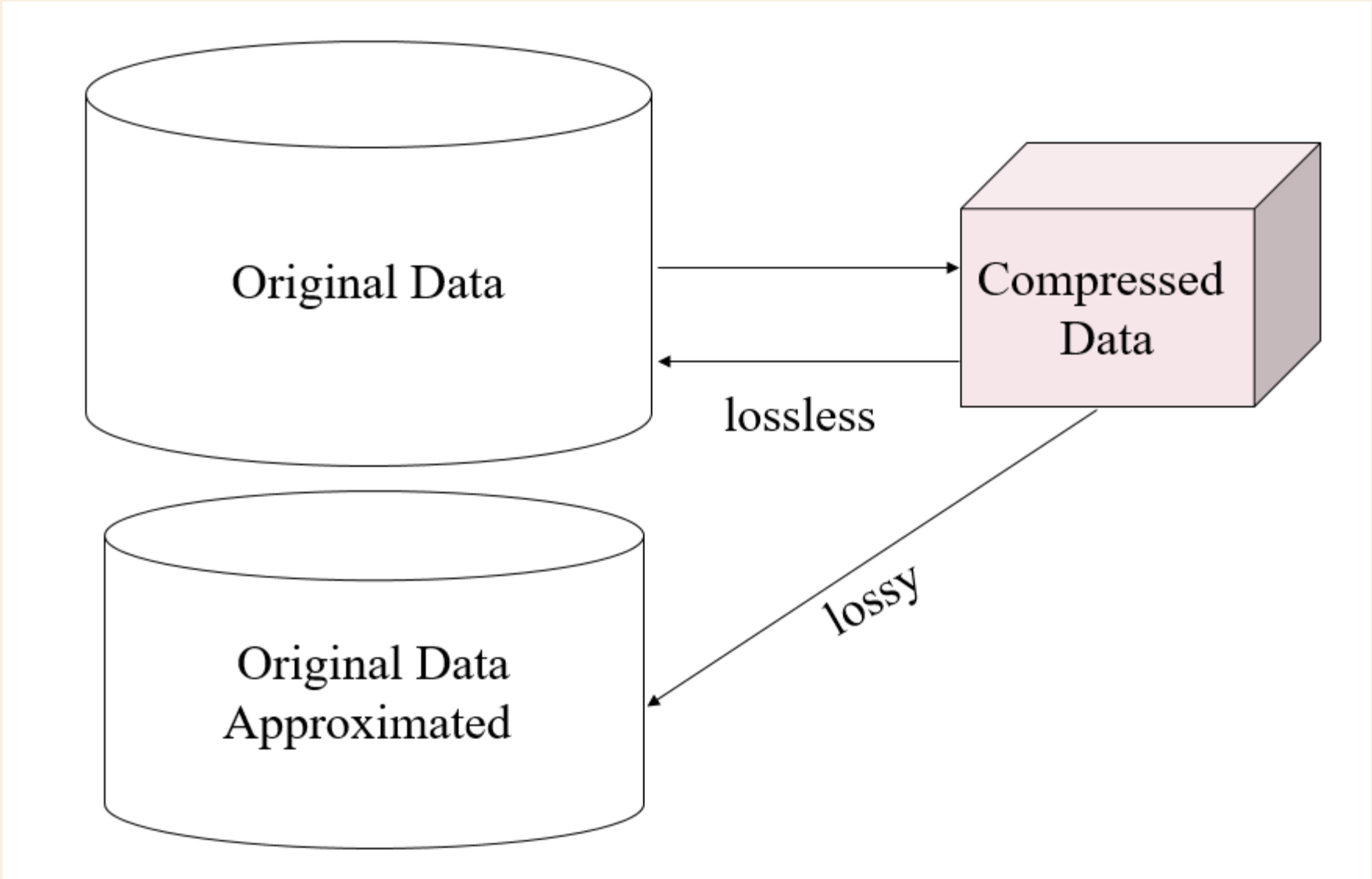
Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ $\Rightarrow \{A_1\}$ $\Rightarrow \{A_1, A_4\}$ $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4,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

3.6 Greedy (heuristic) methods for attribute subset selection.

[4]In machine

DATA REDUCTION – DATA COMPRESSION

- Process of reducing the number of bits needed to either store or transmit the data (text, graphics, audio, video)
- Done using various encoding techniques
- Classified as:
  - Lossy compression
  - Lossless compression


Original Data
Compressed Data
lossless
lossy
Original Data Approximated

DATA REDUCTION – DATA COMPRESSION

- **Lossy compression**

✓Can achieve higher compression ratio at the cost of data quality

✓Useful in applications where data loss is affordable

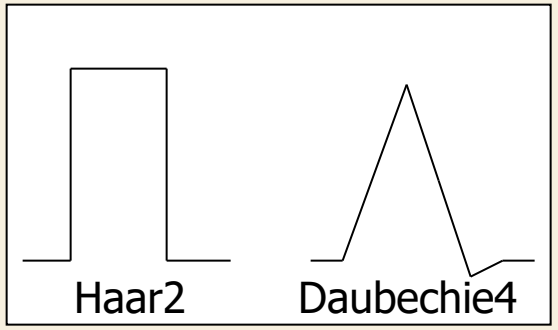
✓Mostly applied to digitized representations of analog phenomenon

✓Methods:

  – Wavelet transform
  – Principle component analysis

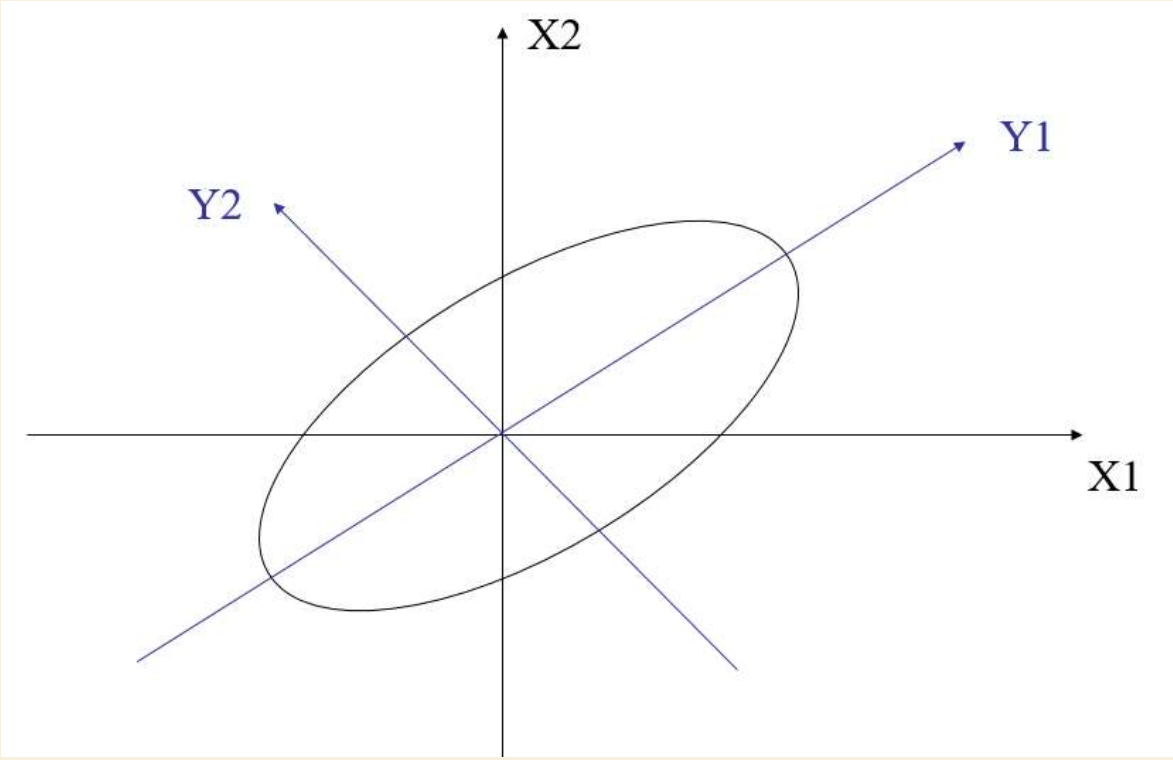
Wavelet Transformation

- 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(i.e. 2)
  - Selected values from the data sets obtained in the previous iterations are designated the wavelet coefficients of the transformed data.



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


X2
Y1
Y2
X1

DATA REDUCTION – DATA COMPRESSION

- **Lossless compression**

✓Generates an exact duplication of the input dataset after compress / decompress cycle

✓Uses Huffman coding, run length coding and arithmetic coding

DATA REDUCTION – NUMEROSITY REDUCTION

- Reduces the volume of data by choosing smaller forms for data representation

- **Techniques Used:**
  - Histograms
  - Clustering
  - Sampling

DATA REDUCTION – NUMEROSITY REDUCTION

- **Histograms – Types**

✓Equal width histograms – Divides the range into N intervals of equal size

✓Equal depth partitioning – Divides the range into N intervals, each containing approximately same number of samples

✓MaxDiff – Data is sorted and then the borders of the buckets are defined where the adjacent values have maximum difference

DATA REDUCTION – NUMEROSITY REDUCTION

- **Clustering**

✓Data mining technique used to group elements based on their similarity without prior knowledge of their class labels

- **Sampling**

✓Used in preliminary investigations as well as final analysis of data

✓Important as processing the entire data set is expensive and time consuming

DATA REDUCTION – NUMEROSITY REDUCTION

✓**Types:**

- Simple random sampling – there is an equal probability of selecting an item
- Sampling without replacement – as each item is selected, it is removed from the population
- Sampling with replacement – objects selected for the sample are not removed from the population. Same object can be selected multiple times.
- Stratified sampling – data is split into partitions and samples are drawn from each partition randomly

DATA TRANSFORMATION

- Integrating data from multiple sources introduces the problem of inconsistency which is handled by data transformation
- Commonly used process is **Attribute Naming Inconsistency** as data in different sources may have different attribute names
- Eg: Customer name can be referred to as cust_name or c_name
- In transformation, one set of data names is considered and used consistently in data warehouse

DATA TRANSFORMATION

- Once naming consistency is done, they must be converted to a common format

- **Conversion process involves:**
  - To ensure consistency uppercase representation may be used for mixed case text
  - Common format may be adopted for numerical data
  - Common representation may be used for measurements
  - Common format may be used for coded data (Eg: Male/Female, M/F)

DATA TRANSFORMATION

- **Data transformation can have following activities:**

✓Smoothing – Removal of noise from data

✓Aggregation – Summarization and data cube construction

✓Generalization – Data is replaced by higher level concepts using concept hierarchy

✓Normalization – Attribute scaling is performed for a specified range

DATA DISCRETIZATION

- Range of a continuous attribute is divided into intervals which reduces the number of values for a given continuous attribute
- Categorical attributes are accepted only by few classification algorithms
- Discretization reduces the size of data and prepares it for further analysis
- Actual data values may be replaced by interval labels
- Discretization is also performed using binning and histogram techniques