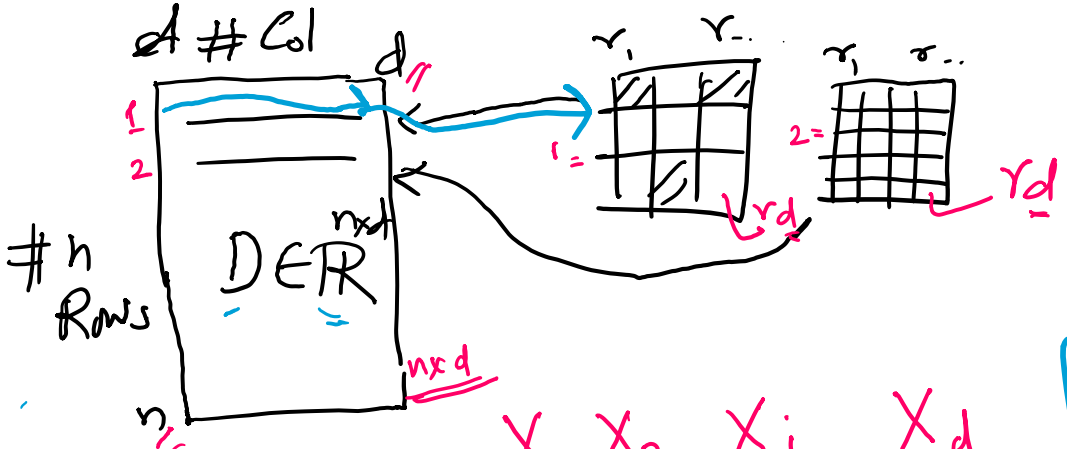
Data Mining

2. Data

Prof Sunil Bhirud

Data Matrix :



Actual

Row View :-

$$\vec{X_1} = \begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1j} \\ \vdots \\ x_{1d} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \end{bmatrix}^T$$

1 x d

$X_1 \; X_2 \; X_j \; X_d$

$$D = \begin{array}{c|cccc} & A_1 & A_2 & \cdots A_j \cdots & A_d \\ \hline \vec{x_1} & x_{11} & x_{12} & x_{1j} & x_{1d} \\ \vec{x_2} & x_{21} & x_{22} & x_{2j} & x_{2d} \\ \vdots & & & & \\ \vec{x_i} & x_{i1} & x_{i2} & x_{ij} & x_{id} \\ \vdots & & & & \\ \vec{x_n} & x_{n1} & x_{n2} & x_{nj} & x_{nd} \end{array}$$

n x d

Column View :-

$$A_1 = \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{n1} \end{bmatrix}^T$$

1 x n

& Important.

RW : 1 x d
colm : n x 1) View

What is Data?

- Data:
- Any observation that have been collected
- Collection of data objects and their attributes
- An attribute is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- A collection of attributes describe an object
  - Object is also known as record, point, case, sample, entity, or instance

Attributes / Variable / Field / Characteristics / Feature (Eye color, temperature etc..)

Data Objects/ Record / Point / Case / Sample / Entity / Instance

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

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.

Attribute Values

- Attribute values : Numbers or Symbols representing a characteristic or feature
  - E.g., customer _ID, Name, Address, Income etc.

- Distinction between attributes and attribute values
  - Same attribute can be mapped to different attribute values
    - Example: Attribute (height)- can be measured in feet or meters
  - Different attributes can be mapped to the same set of values
    - Example: Attribute (values)- for ID and age are integers
    - But properties of attribute values can be different
      - ID has no limit,
      - but age has a maximum and minimum value

Attribute Types

- **Nominal:** categories, states, or "names of things" : NOT Ordered
  - *Hair_color* = {*auburn, black, blond, brown, grey, red, white*}
  - marital status, occupation, ID numbers, zip codes
- **Binary**
  - Nominal attribute with only 2 states (0 and 1)
  - Symmetric binary: both outcomes equally important
    - 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:** **Ordered- Differences are Meaningless**
  - Values have a meaningful order (ranking) but magnitude between successive values is not known.
  - *Size* = {*small, medium, large*}, grades, army rankings, Color (Spectrum)

Numeric Attribute Types

- Quantity (integer or real-valued)
- **Interval : Ordered. Differences are Meaningful**
  - No natural Zero
  - Measured on a scale of **equal-sized units**
    - E.g., *temperature in C˚or F˚ (0˚ is a measured temperature), calendar dates*
  - No true zero-point
- **Ratio: Just like Interval**
  - Inherent **zero-point (Natural Zero)**
  - **Zero bank balance.**
  - 10 K˚ is twice as high as 5 K˚).
    - e.g., *temperature in Kelvin, length, counts, monetary quantities*

Properties of Attribute Values

- The type of an attribute depends on which of the following properties it possesses:
  - Distinctness: = ≠
  - Order: < >
  - Addition: + -
  - Multiplication: * /

  - Nominal attribute: distinctness
  - Ordinal attribute: distinctness & order
  - Interval attribute: distinctness, order & addition
  - Ratio attribute: all 4 properties

Attribute Type	Description	Examples	Operations
Nominal	The values of a nominal attribute are just different names, i.e., nominal attributes provide only enough information to distinguish one object from another. (=, ≠)	zip codes, employee ID numbers, eye color, sex: {*male*, *female*}	mode, entropy, contingency correlation, χ^2 test
Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	hardness of minerals, {*good*, *better*, *best*}, grades, street numbers	median, percentiles, rank correlation, run tests, sign tests
Interval	For interval attributes, the differences between values are meaningful, i.e., a unit of measurement exists. (+, -)	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, *t* and *F* tests
Ratio	For ratio variables, both differences and ratios are meaningful. (*, /)	temperature in Kelvin, monetary quantities, counts, age, mass, length, electrical current	geometric mean, harmonic mean, percent variation

Attribute Level	Transformation	Comments
Nominal	Any permutation of values	If all employee ID numbers were reassigned, would it make any difference?
Ordinal	An order preserving change of values, i.e., *new_value = f(old_value)* where *f* is a monotonic function.	An attribute encompassing the notion of good, better best can be represented equally well by the values {1, 2, 3} or by { 0.5, 1, 10}.
Interval	*new_value =a * old_value + b* where a and b are constants	Thus, the Fahrenheit and Celsius temperature scales differ in terms of where their zero value is and the size of a unit (degree).
Ratio	*new_value = a * old_value*	Length can be measured in meters or feet.

Discrete and Continuous Attributes

- Discrete Attribute
  - Finite or countably infinite set of values
  - Examples: zip codes, counts, or the set of words in a collection of documents
  - Often represented as integer variables.
  - Note: binary attributes are a special case of discrete attributes

- Continuous Attribute
  - Real numbers as attribute values (Infinite values)
  - Examples: temperature, height, or weight.
  - Practically, real values can only be measured and represented using a finite number of digits.
  - Continuous attributes are typically represented as floating-point variables.
  - Usually a Measurement…

Types of data sets

- **Record**
  - Data Matrix
  - Document Data
  - Transaction Data
- **Graph**
  - World Wide Web
  - Molecular Structures
- **Ordered**
  - Spatial Data
  - Temporal Data
  - Sequential Data
  - Genetic Sequence Data

Important Characteristics of Structured Data

- **Dimensionality**
  - Curse of Dimensionality
- **Sparsity**
  - Only presence counts
- **Resolution**
  - Patterns depend on the scale
- **Distribution**
  - Centrality and dispersion

Record Data

- Data that consists of a collection of records, each of which consists of a fixed set of attributes

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Data Matrix

- If data objects have the same fixed set of numeric attributes, then the data objects can be thought of as points in a multi-dimensional space, where each dimension represents a distinct attribute

- Such data set can be represented by an m by n matrix, where there are m rows, one for each object, and n columns, one for each attribute

Projection of x Load	Projection of y load	Distance	Load	Thickness
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

Document Data

- Each document becomes a `term' vector,
  - each term is a component (attribute) of the vector,
  - the value of each component is the number of times the corresponding term occurs in the document.

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

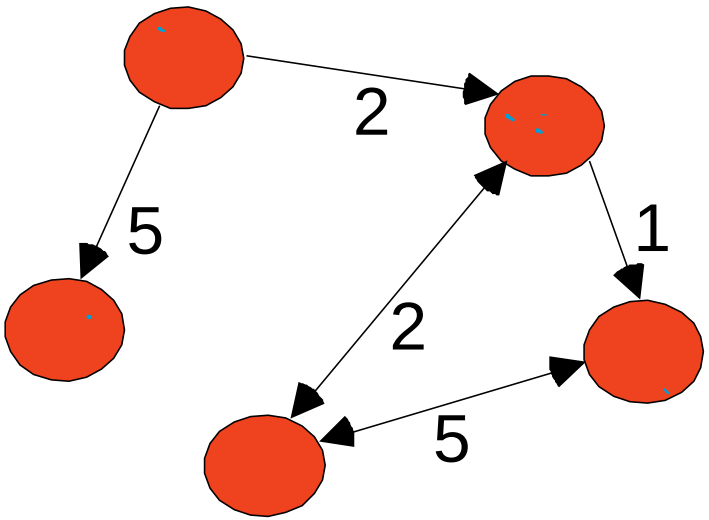
Transaction or Market Basket Data

- A special type of record data, where
  - each record (transaction) involves a set of items.
  - For example, consider a grocery store. The set of products purchased by a customer during one shopping trip constitute a transaction, while the individual products that were purchased are the items.

Transaction ID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Graph-Based Data

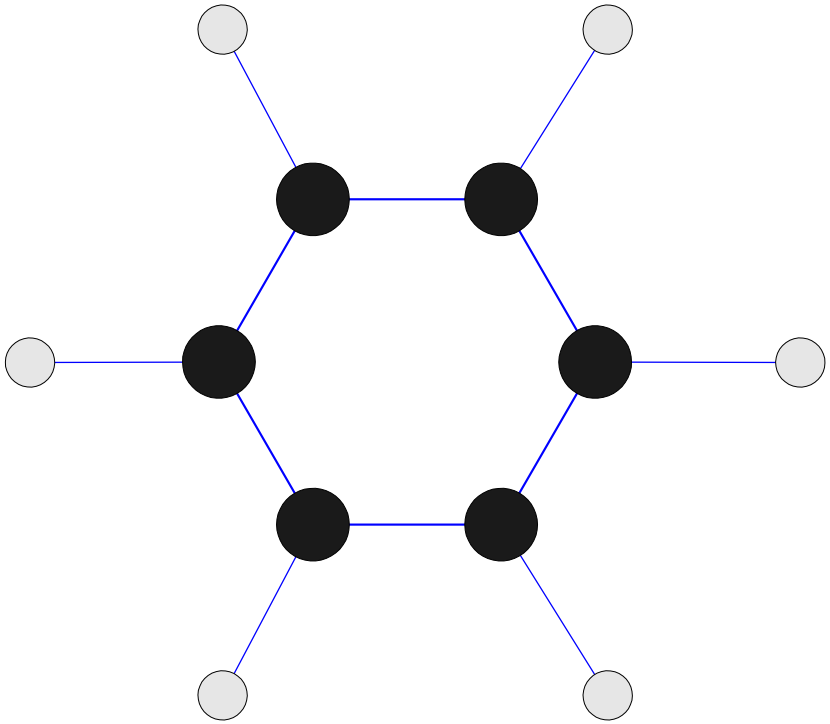
- Examples: Generic graph and HTML Links




```
<a href="papers/papers.html#bbbb">
Data Mining </a>
<li>
<a href="papers/papers.html#aaaa">
Graph Partitioning </a>
<li>
<a href="papers/papers.html#aaaa">
Parallel Solution of Sparse Linear System of Equations </a>
<li>
<a href="papers/papers.html#ffff">
N-Body Computation and Dense Linear System Solvers
```

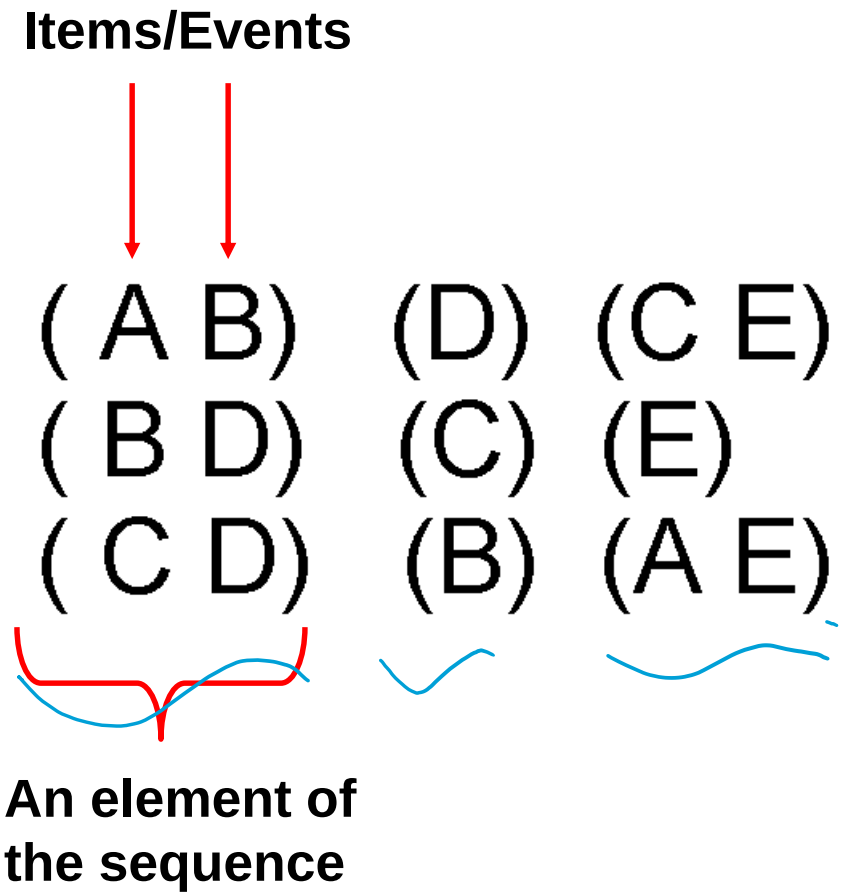
Chemical Data

- Benzene Molecule: C_6H_6



Ordered Data

- Sequences of transactions



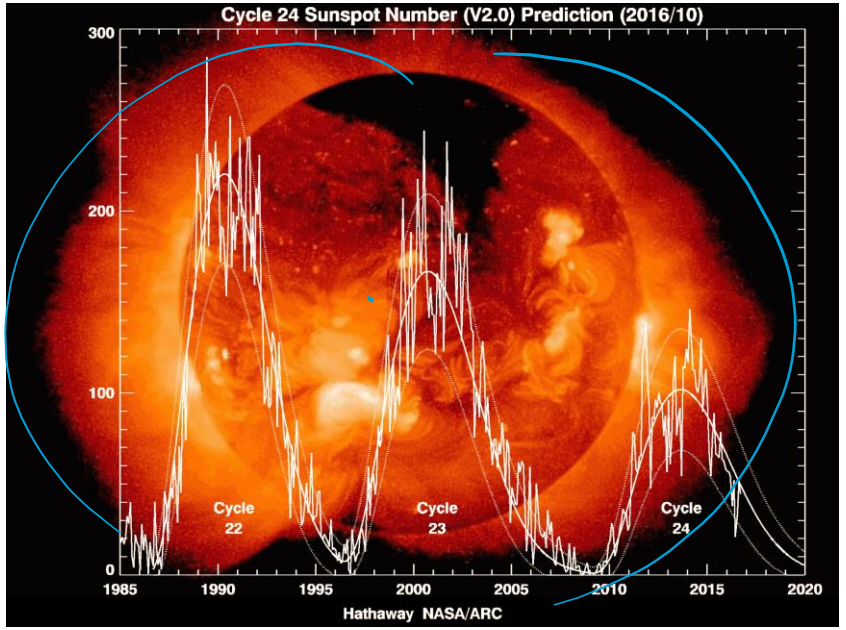
Ordered Data

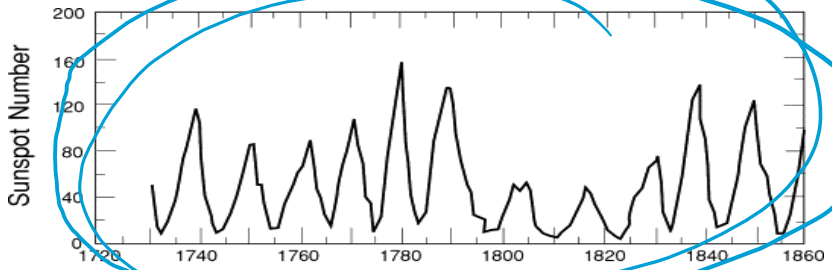
- Genomic sequence data

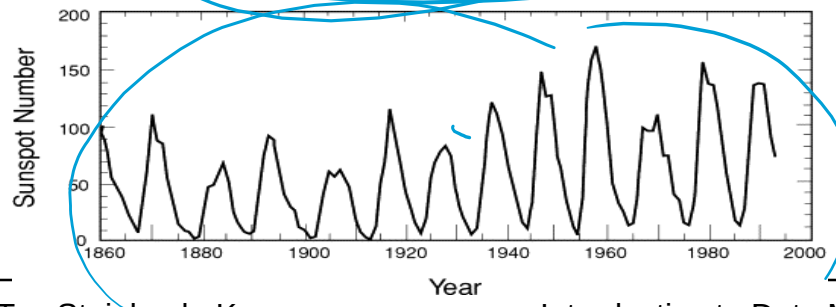
```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

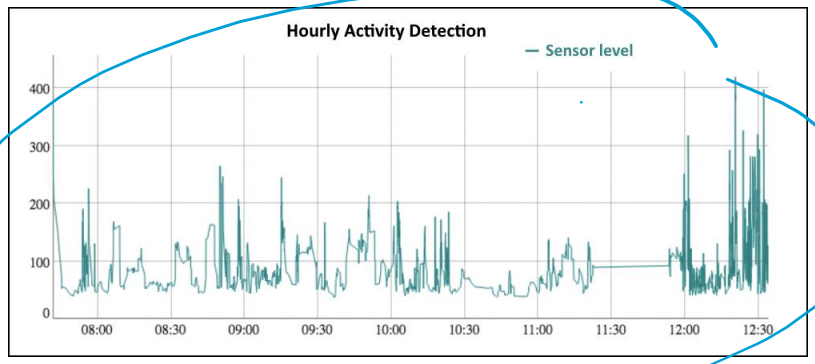
Ordered Data: Time Series Data:

- The **solar cycle** or **solar magnetic activity cycle** is a nearly periodic 11-year change in the Sun's activity measured in terms of variations in the number of observed sunspots on the solar surface. Sunspots have been observed since the early 17th century and the sunspot time series is the longest continuously observed (recorded) time series of any natural phenomena.





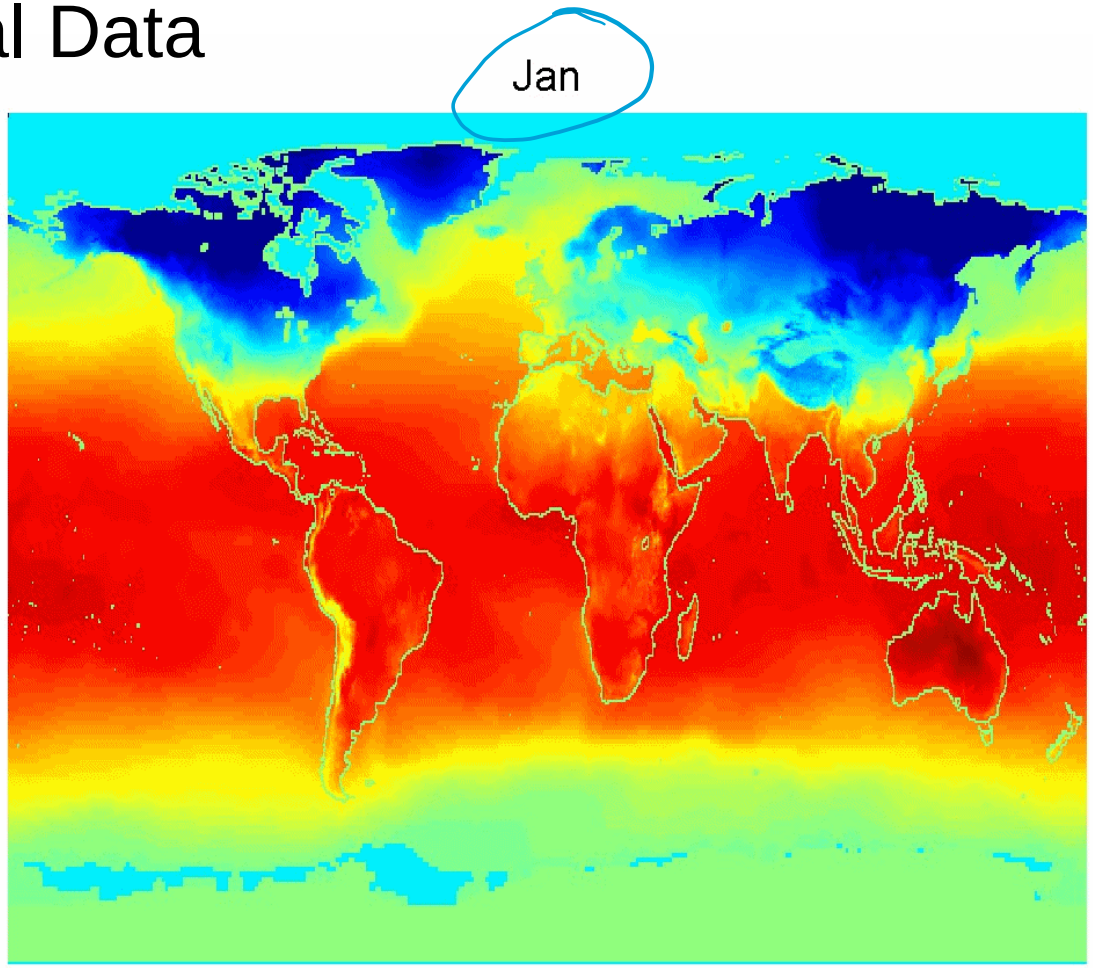




Ordered Data

- Spatio-Temporal Data

Average Monthly Temperature of land and ocean



Data Quality

- What kinds of data quality problems?
- How can we detect problems with the data?
- What can we do about these problems?

- Examples of data quality problems:
  - Noise and outliers
  - missing values
  - duplicate data

Measurement and Data Collection Issues

Measurement Error:

- Error is the difference between the measured value and the actual value.

Data Collection Error:

- Omitting data objects
- Omitting attribute values
- Taking in-appropriate data objects: while collecting one specific species of animal data, you also collect data of similar species of animals.

Measurement and Errors

Any observation is composed of the true value plus some random error value.

But is that reasonable?

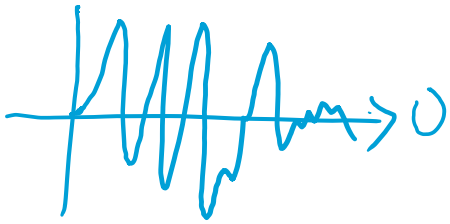
What if all error is random?
What if all error is not-random? (Systematic Error)

$$X = T + e_r + e_s$$

X: Observed Value
T: True Value
e_r: Random Error
e_s: Systematic Error

Random Error

Random errors: are caused by the sudden change in experimental conditions and noise and tiredness in the working persons.



- These errors are either positive or negative.
- It is due to factors which cannot be controlled.
- It may be too expensive to control them each time the experiment is conducted or the measurements are made.

Example: changes in humidity, unexpected change in temperature and fluctuation in voltage while taking readings.

- These errors may be reduced by taking the average of a large number of readings.
- Random Error is sometimes called as NOISE.

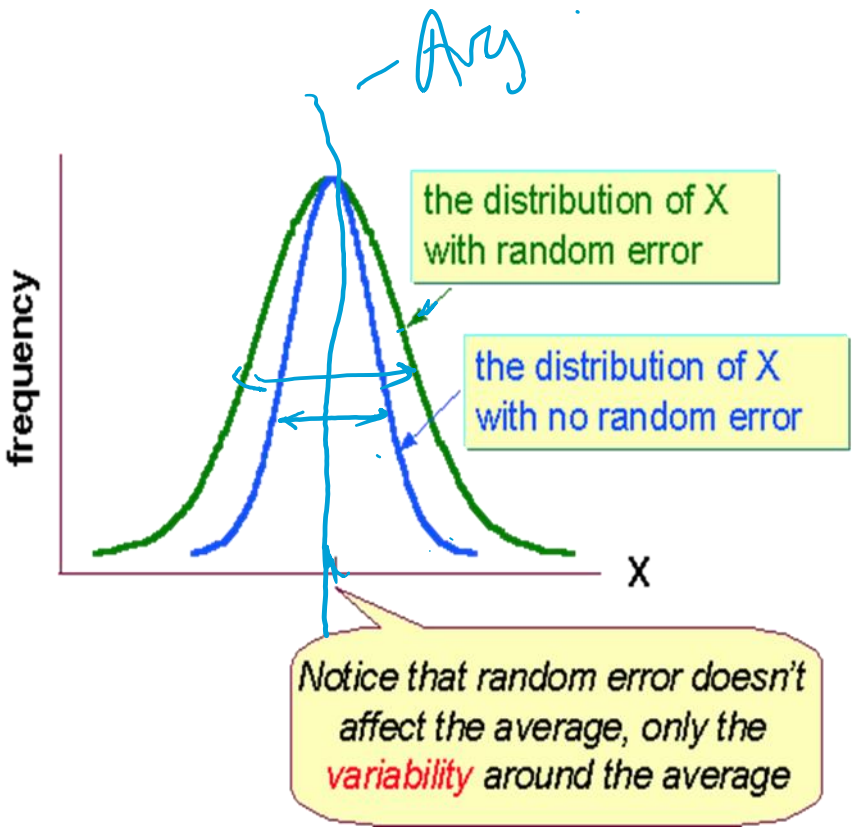
For example:

- It is common for digital balances to exhibit random error in their least significant digit.

- Three measurements of a single object might read something like 0.9111g, 0.9110g, and 0.9112g.

- The concept of random error is closely related to the concept of precision.

- The higher the precision of a measurement instrument, the smaller the variability (standard deviation) of the fluctuations in its readings.



Systematic Error

Systematic error:

It occur due to fault in the measuring device - are known as systematic errors.

- Usually they are called as Zero Error – a positive or negative error. Sometimes it is called as BIAS in instrument
- If the cause of the systematic error can be identified, then it usually can be eliminated.

Categories of Systematic Error:

- Instrumental Error: Imperfect calibration of measurement instruments (hysteresis or friction or Loading effect)

Categories of Systematic Error:

- Environmental Error:

  Interference of the environment with the measurement process i.e external conditions (pressure, temp, humidity, magnetic field)

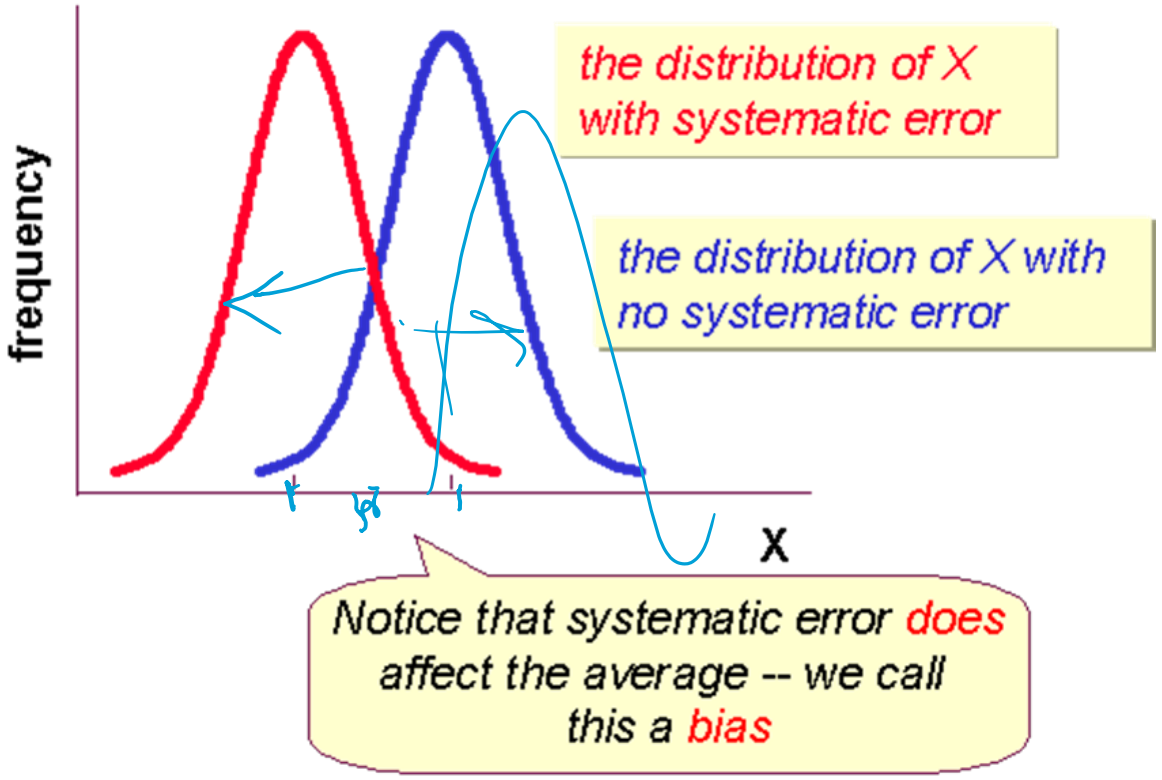
- Observational Errors:

  Wrong observation or reading in the instrument (Parallax)

- Theoretical Errors:

  While designing it is assumed that temp of the surrounding will not change the reading, which is not true (some procedures / instruments are sensitive to temp / environment / other specific conditions)

Systematic Error



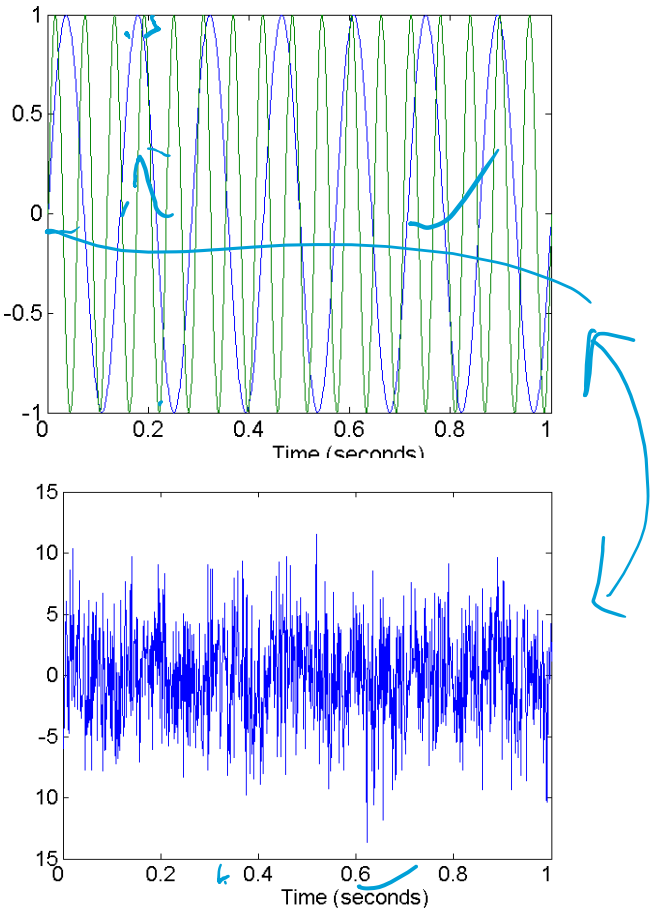
Noise and Artifacts

Noise: is the random component of a measurement.

Noise refers to modification of original values

Examples:

Distortion of a person's voice when talking on a poor phone and "snow" on television screen.



Two Sine Waves + Noise

Data Artifacts

Is a data flaw caused by equipment, techniques or conditions.

Sources of data flaw:

Hardware, software errors, electromagnetic interference and flaw in algorithm- prone to miscalculations

Types of Artefacts:

- Digital Artifacts : Digital camera that records a distorted or corrupted image.
- Visual Artifacts : Flaws in the visualizations such as user interfaces or streaming media
- Compression Artifacts : An Image become visibly distorted due to compression
- Noise Artifacts : Unwanted electrical fluctuations in radio reception
- Statistical Artifact : A flaw such as a bias in statistical data
- Radar Artifacts : Ghost objects in radar images/data/signal due to atmospheric effects or unfiltered echoes
- Sonic Artifacts : An unwanted sound as background noise on a film set. In some cases Artifacts are used a creative elements of music or films. Eg. Overdriving a bass signal for a fuzzy bass sound.

Precision

Precision: Closeness of repeated measurement (of the same quantity) to one another.

- Used for finding the consistency or reproducibility of the measurement.
- High precision: measurements are consistent or the repeated values of the reading are obtained.
- Low precision: value of the measurement varies.

Example:

Voltmeter readings :100V, 101V, 102V, 103V and 105V

The readings are nearly close to each other.

They are not exactly same because of the error.

The reading are close to each other, then we say that the readings are precise.

Bias:

Bias:

- A systematic variation of measurements from the quantity being measured.
- Difference between mean of the set of values and the known value of quantity being measured
- How close the measurements are to the true value.

Consider standard lab weighs of 1gm

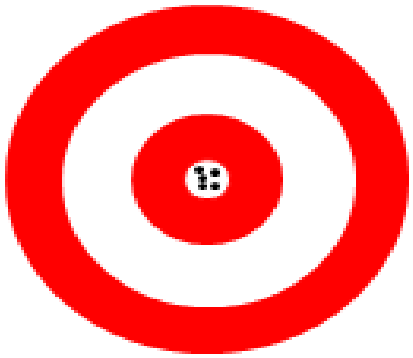
The weight of mass taken five times is { 1.015, 0.990, 1.013, 1.001, 0.986}

Mean=1.001 and hence the bias is 0.001

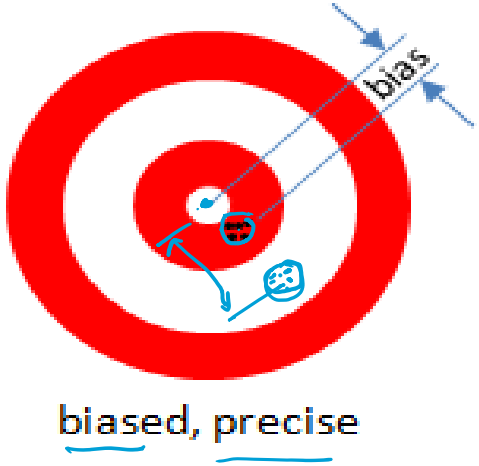
Precision is given by Standard Deviation=0.013

Bias and Precision

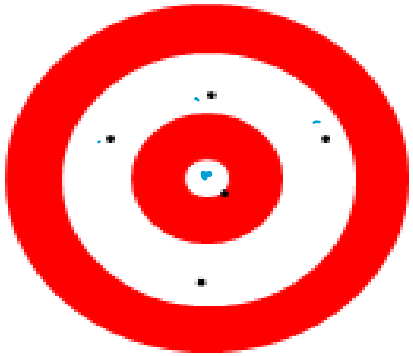
Being unbiased isn't always a good thing.



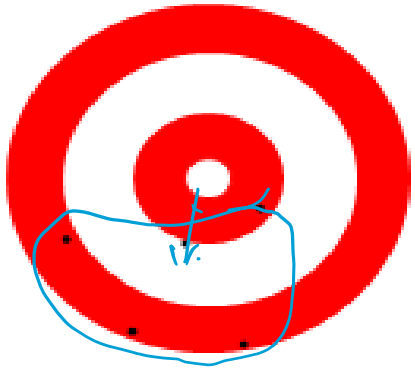
unbiased, precise



biased, precise



unbiased, imprecise

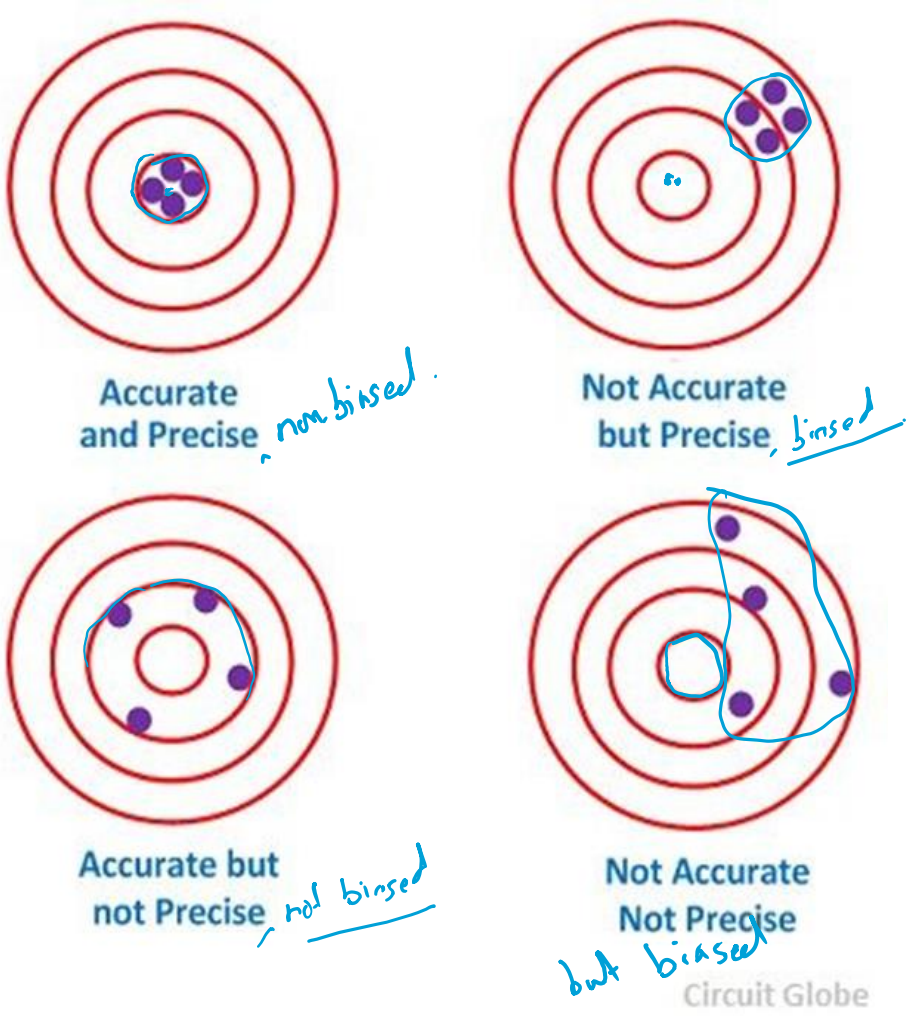


biased, imprecise

Accuracy

The closeness of the measurement value to the true/standard value of the quantity being measured.

It is the ability of the instrument to measure the accurate value.



Noise, Bias and Accuracy

Team A is *accurate*: The shots of the teammates are on the bull's-eye and close to one another.

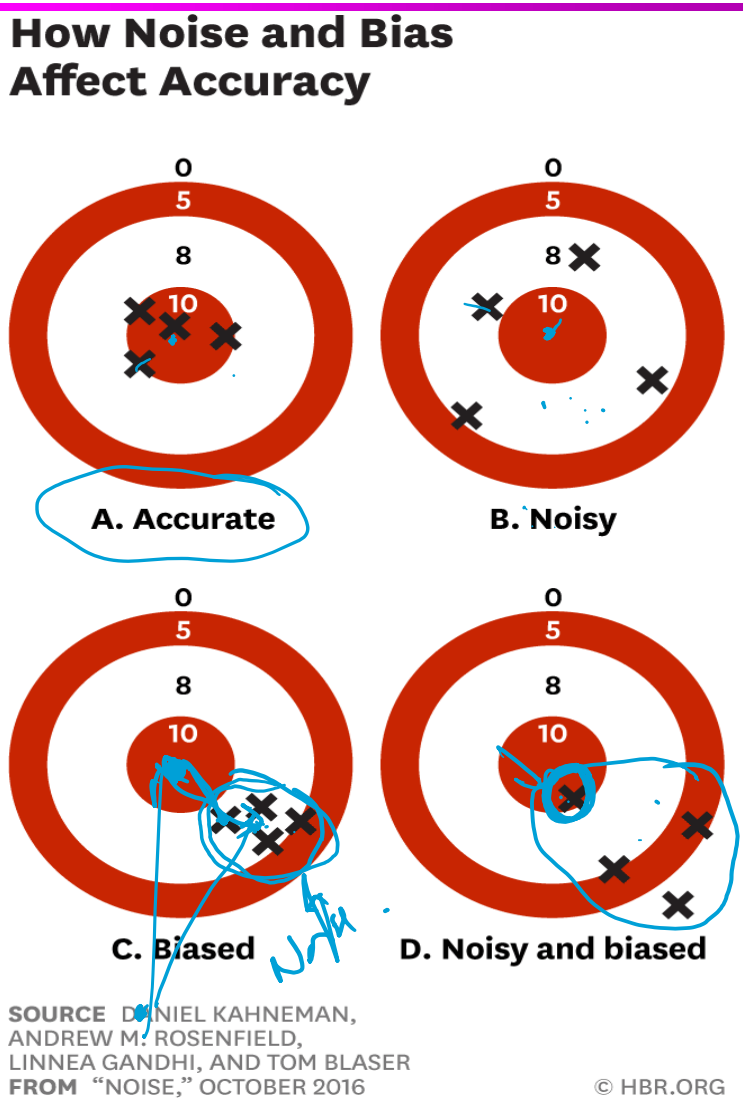
The other three teams are inaccurate but in distinctive ways:

Team B is *noisy:* The shots of its members are cantered around the bull's-eye but widely scattered.

Team C is *biased:* The shots all missed the bull's-eye but cluster together.

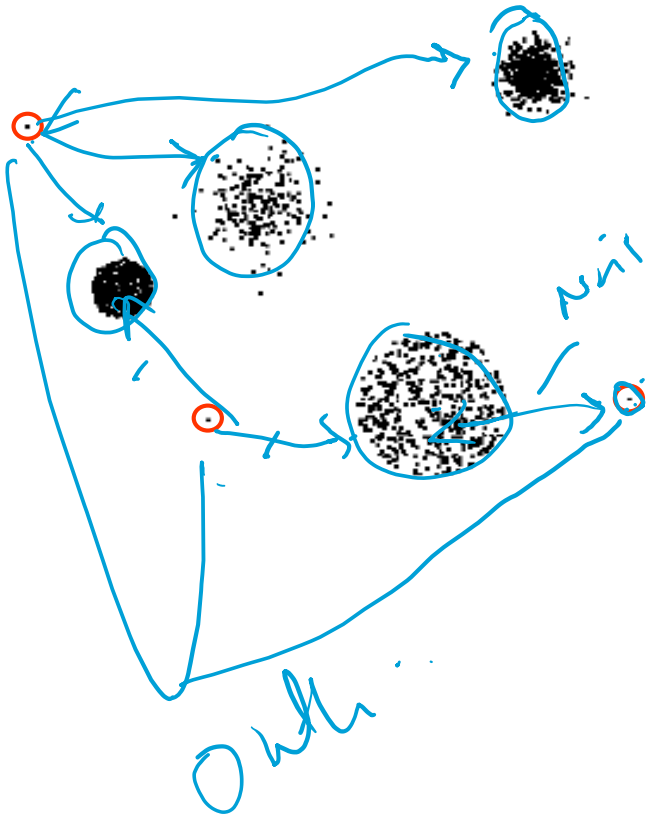
Team D is both *noisy* and *biased*.

As a comparison of teams A and B illustrates, an increase in noise always affects accuracy when there is no bias. When bias is present, increasing noise may actually cause a lucky hit, as happened for team D. Of course, no organization would put its trust in luck. Noise is always undesirable—and sometimes disastrous.



Outliers

- “Outliers” are data objects with characteristics that are considerably different than most of the other data objects in the data set
- “Outliers” is an observation that appears far away and diverges from an overall pattern in a sample.
- They are not necessarily wrong and are often the most interesting and informative observations in the sample



Outliers

Most common causes of outliers on a data set:

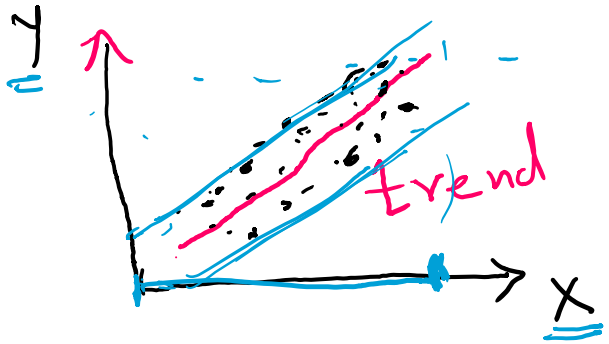
- Data entry errors (human errors)
- Measurement errors (instrument errors)
- Experimental errors (data extraction or experiment planning/executing errors)
- Intentional (dummy outliers made to test detection methods)
- Data processing errors (data manipulation or data set unintended mutations)
- Sampling errors (extracting or mixing data from wrong or various sources)
- Natural (not an error, novelties in data)

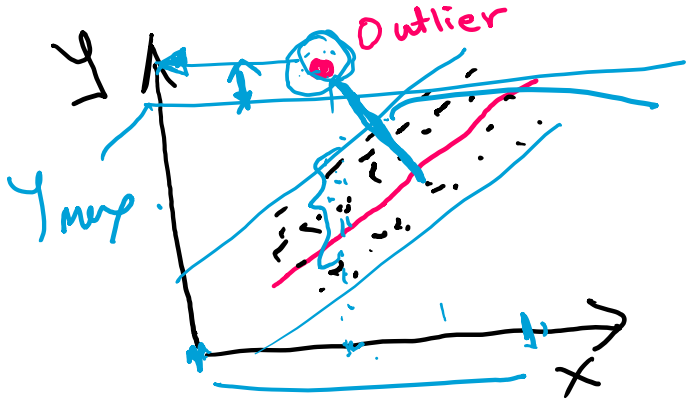
Some of the most popular methods for outlier detection are:

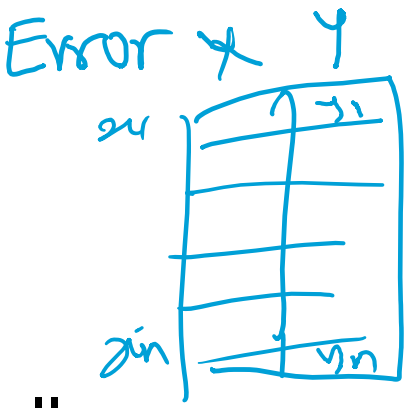
- Z-Score or Extreme Value Analysis (parametric)
- Probabilistic and Statistical Modeling (parametric)
- Linear Regression Models (PCA, LMS)
- Proximity Based Models (non-parametric)
- Information Theory Models
- High Dimensional Outlier Detection Methods (high dimensional sparse data)

Outlier and Leverage Point

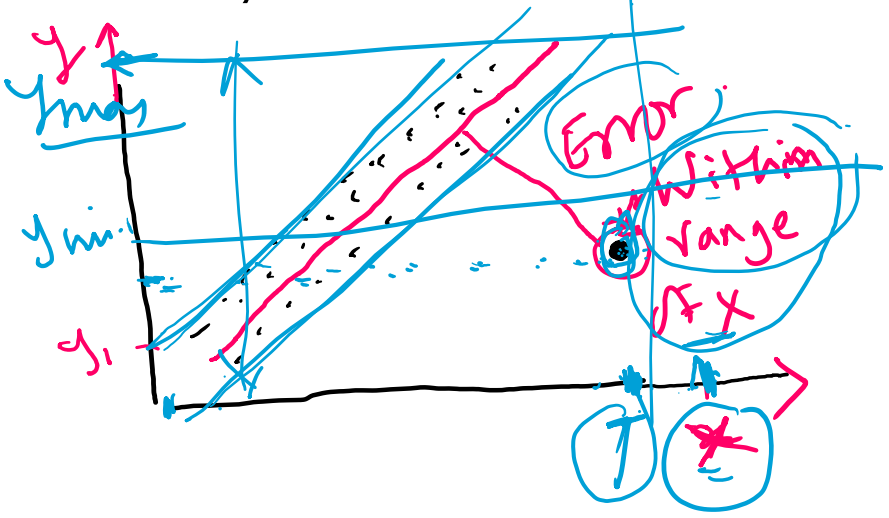
a. Outlier: Y- data that doesn't follow the general trend.





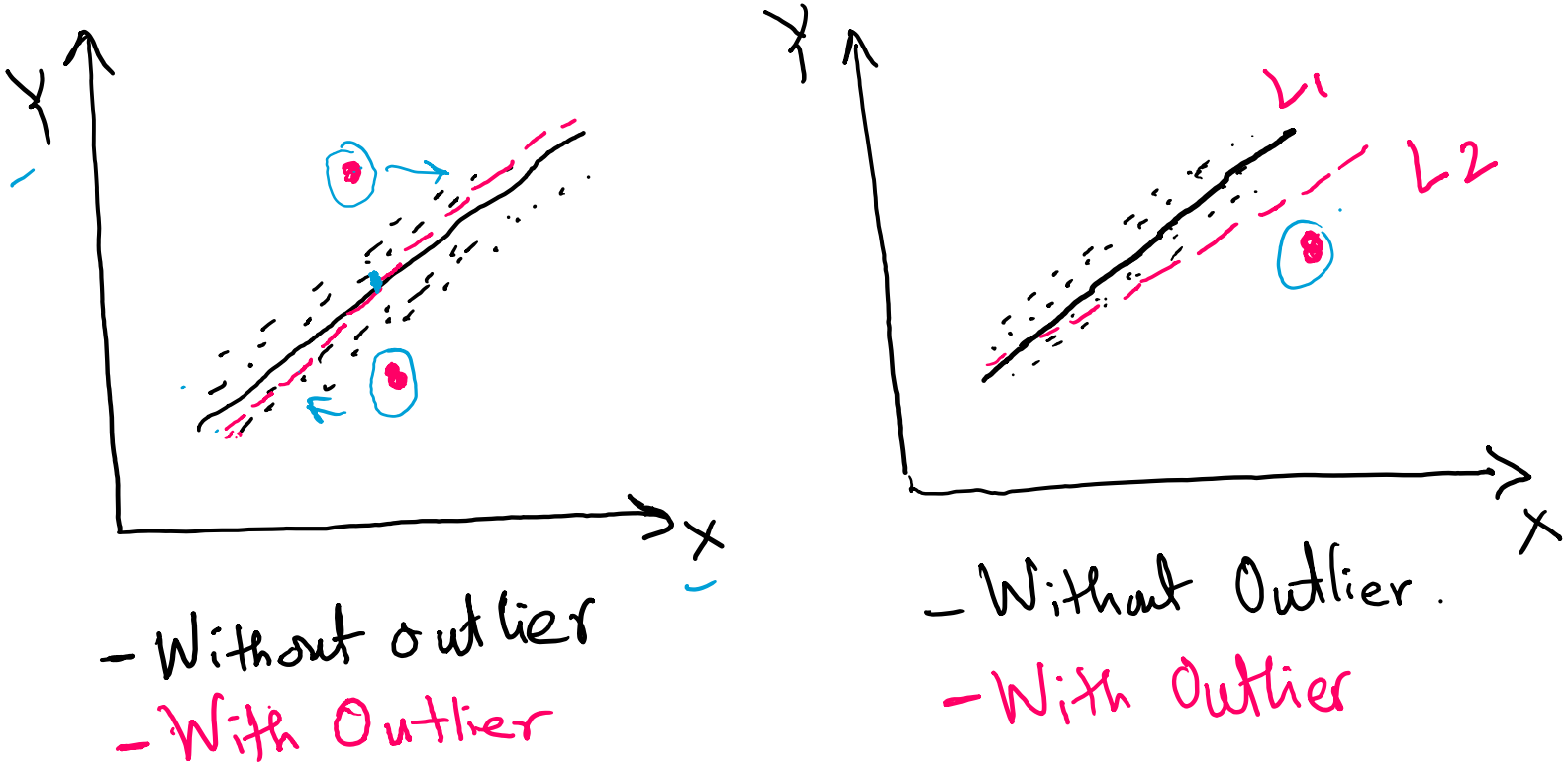


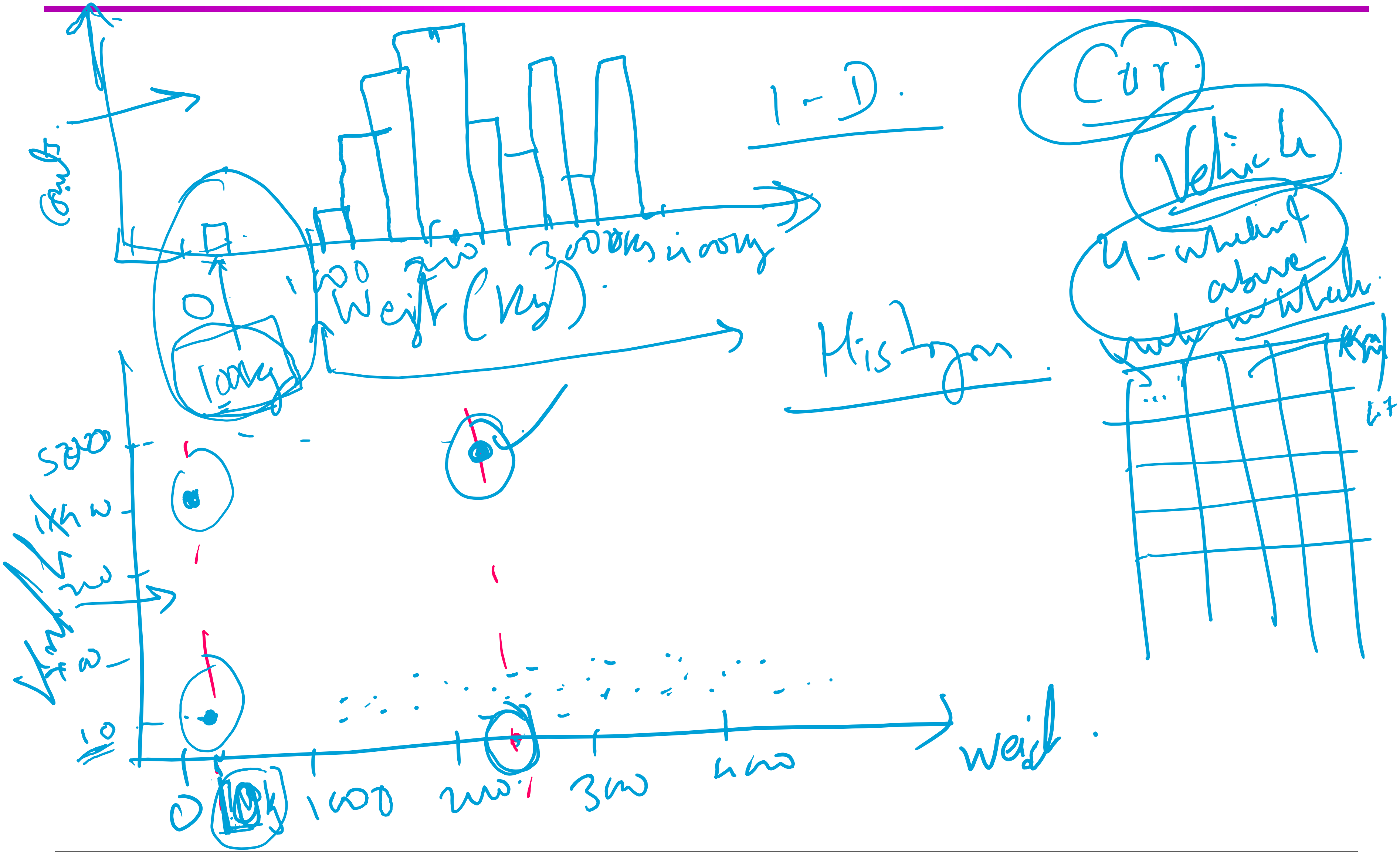
b. Leverage: Extreme X values, and Y follows trend



Outlier, Leverage Influence

Outlier and Leverage influence the predictions




Counts
Weight (Kg)
1-D.
Histogram
Car
Vehicle
4-wheeler
weight

Missing Values

- Reasons for missing values
  - Information is not collected (forgotten or lost)
    (e.g., people decline to give their age and weight,)
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)
  - It is of no interest to the instance
  - (measured parameter not related to patient condition)

- Handling missing values
  - Eliminate Data Objects
  - Estimate Missing Values
  - Ignore the Missing Value During Analysis
  - Replace with all possible values (weighted by their probabilities)

Dealing with Missing Data

Use what you know about

- Why data are missing ?
- Distribution of missing data

Decide on the best analysis strategy to yield the estimates

Deletion Methods

Delete all cases with incomplete data and conduct analysis using only complete cases.

Advantages: Simple

Disadvantage: Loss of data if we discard all incomplete cases. So, in-efficient.

NOTE: if you use complete case analysis, then changes summery statistics for other variable, too.

Example: 15% missing data

	Case 1				Case 2				Case 3			
	y1	y2	y3	y4	y1	y2	y3	y4	y1	y2	y3	y4
R1	NA1	NA2	NA3	NA4	NA1				NA1			
R2	NA5	NA6			NA2				NA2			
R3					NA3					NA3		
R4					NA4					NA4		
R5					NA5						NA5	
R6					NA6						NA6	
R7												
R8												
R9												
R10												

Case 1: Eliminate R1 and R2, Keep 8*4=32 data. 20%Loss
Case 2: Eliminate Y1 and Keep 10*3=30 data. 25% Loss
Case 3: Eliminate record R1 to R6, and Keep record 7 to 10 i.e. 4*4=16 data. 60% Loss

Listwise Deletion (Complete case analysis)

Only analyse cases with available data on each variable

Advantages: simple and compatible across analyses

Disadvantage: reduces statistical power (due to sample size), estimates may be biased.

Listwise deletion often produces unbiased regression slope estimates as long as missingness is not a function of outcome variable.

Pairwise Deletion (Available case analysis)

Analysis with all cases in which the variable of interest are present.

Advantages: Keeps as many cases as possible for each analysis, uses all information possible with each analysis.

Disadvantage: Cannot compare analyses because sample is different each time, sample size vary for each parameter estimation, can obtain nonsense results.

Compute the summary statistics using n_i observations not “n”

Compute correlation type statistics using complete pairs for both variables.

Example

List wise deletion

Gender	Manpower	Sales
M	25	343
F	--	378
M	33	245
F	--	289
M	25	--
--	29	295
M	26	299

Pair wise deletion

Gender	Manpower	Sales
M	25	343
F	--	378
M	33	245
F	--	289
M	25	--
--	29	295
M	26	299

Imputation Methods

1. Random sample from existing values:

Randomly generate an integer from 1 to n- $m_{\text{missing terms}}$, then replace the missing value with the corresponding observation that you chose randomly. (“m” number of missing points)

Case	: 1	2	3	4	5	6	7	8(n)
Y_1	: 3.4	3.9	2.6	1.9	2.2	3.3	1.7	2.4
Y_2	: 5.7	4.8	4.9	6.2	6.8	5.6	--	5.8

Randomly generate number between 1 and 7: Say 3

Replace Y_2,7 by Y_2,3 =4.9

Disadvantage: It may change the distribution of data

Imputation Method

2. Randomly sample from a reasonable distribution

e.g. If gender is missing and you have the information that there are about the same number of female and male in the population.

Gender $\sim$ Ber(p=0.5) or estimate p from the observed sample

Using random number generator from Bernoulli distribution for p=0.5, generate numbers for missing gender data

Disadvantage:

Distributional assumption may not be reliable (or correct even the assumption is correct, its representativeness is doubtful).

Imputation Methods

3. Mean / Mode Substitution

Replace missing value with the sample mean or mode. Then, run analyses as if all complete cases.

Advantages: We can complete case analyses

Disadvantage: Reduces variability, weakens the correlation estimates because it ignores the relationship between variables, it creates artificial band.

Unless the proportion of missing data is low, do not use this method.

Last Observation Carried Forward

This method is specific to longitudinal data problems.

For each individual, NAN are replaced by the last observed value of that variable. Then, analyse data as if data were fully observed.

Disadvantage: The covariance structure and distribution change seriously.

Observation Time

Cases	1	2	3	4	5	6
1	3.8	3.1	2.0	2.0	2.0	2.0
2	4.1	3.5	2.8	2.4	2.8	3.0
3	2.7	2.4	2.9	3.5	3.5	3.5

Imputation Methods

4. Dummy variable adjustment:

Create an indicator variable for missing value (1 for missing, 0 for observed)

Impute missing value to a constant (such as mean)

Include missing indicator in the regression

Advantage: Uses all information about missing observation

Disadvantage: results in biased estimates, not theoretically driven

Imputation Methods

5. Regression imputation:

Replace missing value with predicted score from regression equation.

Use complete cases to regress the variable with incomplete data on the other complete variables.

Advantages: Uses information from the observed data, gives better results than previous ones.

Disadvantage: Over-estimates model fit and correlation estimates, weakens variance.

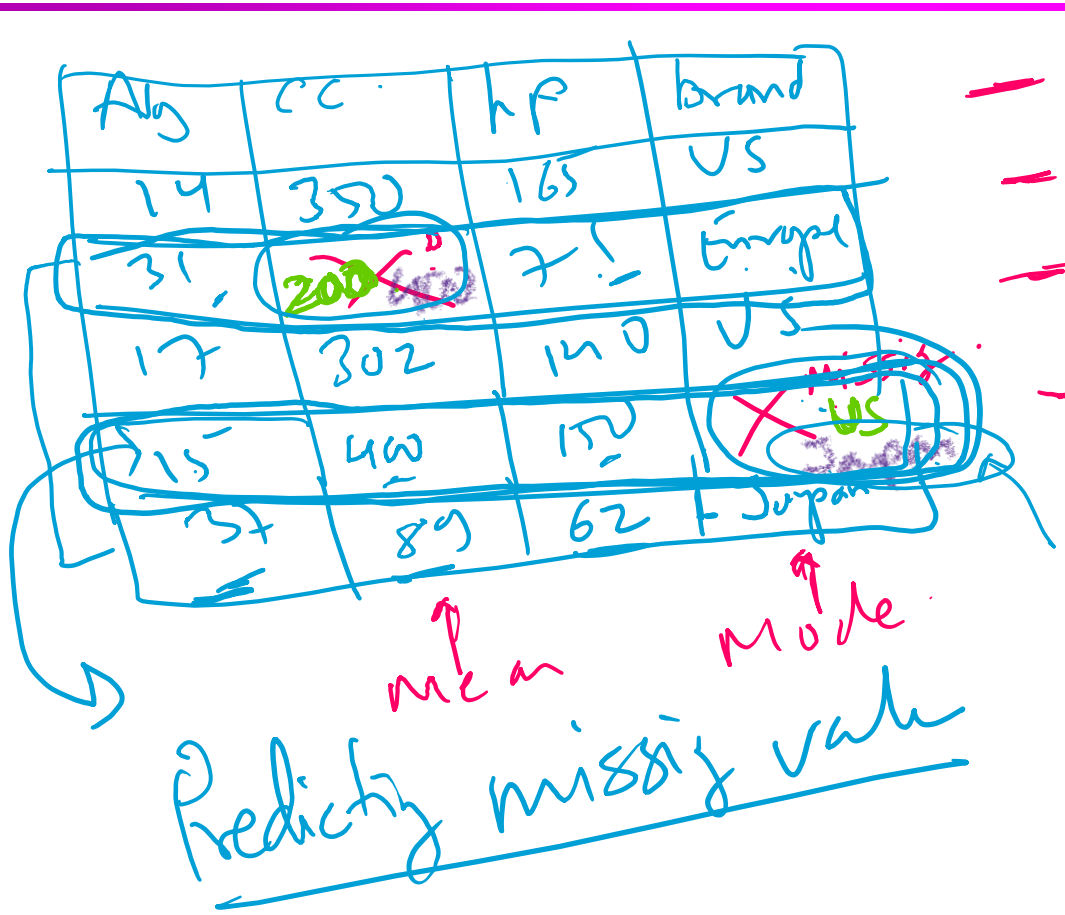
Imputation Methods

Problem:

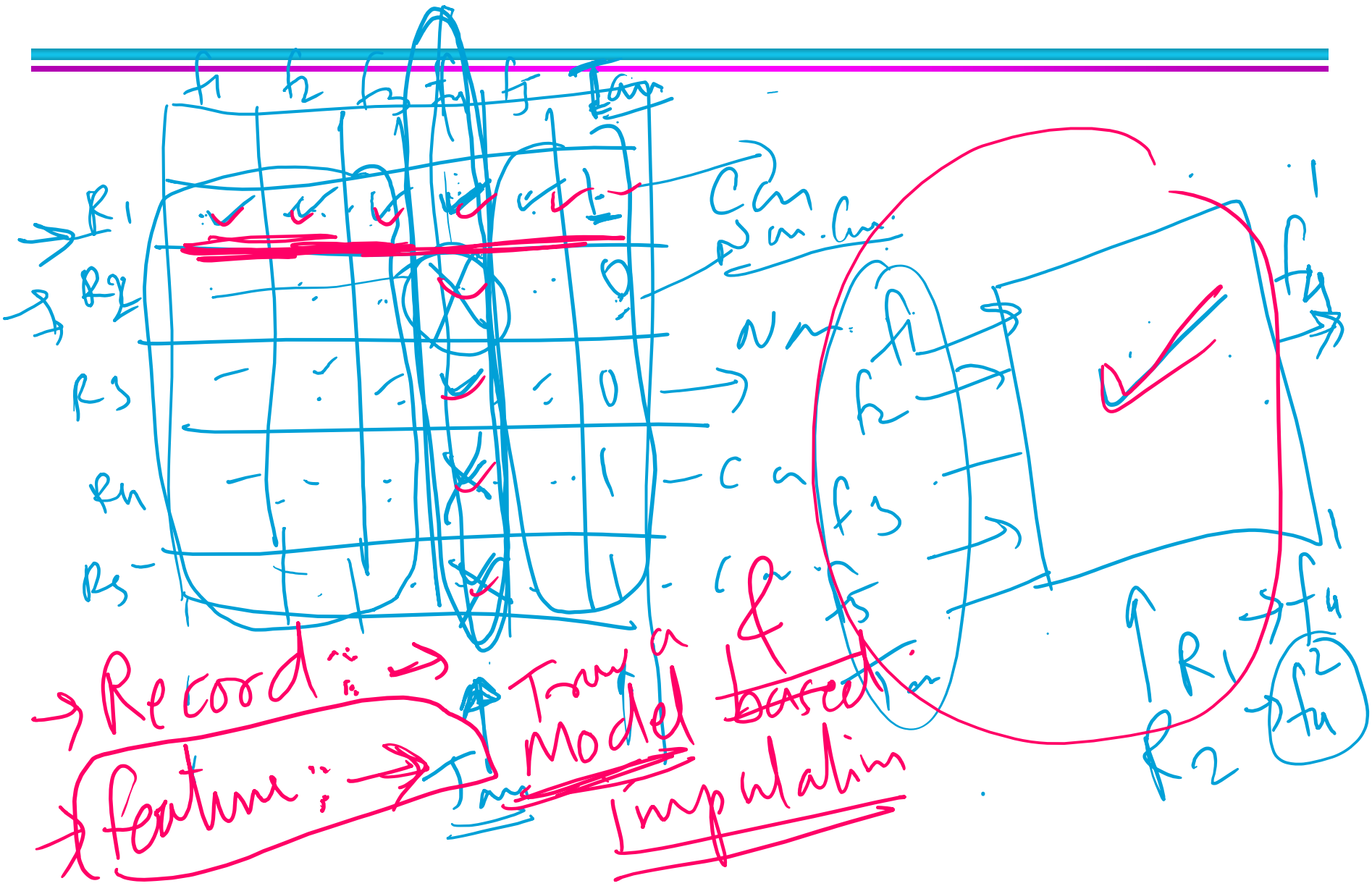
Regression assumes responses are normal distributed.

What if this assumption is unrealistic?

We can still use other models like logistic regression.



- Replace by some constant
- by mean.
- Ram replace from the observed values.
- Impute


Record
feature
Model
Implementation

Choice of the Imputation Method

1. Use a sample of your own dataset that does not contain any missing data (will serve as ground truth).
2. Introduce increasing proportions of missing data at random (e.g. 5–50 % in 5 % increments).
3. Reconstruct the missing data using the various methods.
4. Compute the sum of squared errors between the reconstructed and the original data, for each method and each proportion of missing data.
5. Repeat steps 1–4 a number of times (10 times for example) and compute the average performance of each method (average SSE).
6. Choose the method that performs best at the level of missing data in your dataset. E.g. if your data had 10 % of missing data, you would want to pick k-NN; at 40 % linear regression performs better (made-up data, for illustrative purpose only).

Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeous sources
- Examples:
  - Same person with multiple email addresses
- Data cleaning
  - Process of dealing with duplicate data issues

Data Preprocessing

- Aggregation
- Sampling
- Dimensionality Reduction
- Feature subset selection
- Feature creation
- Discretization and Binarization
- Attribute Transformation

Aggregation

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

- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More “stable” data
    - Aggregated data tends to have less variability (some user conducts many sessions, his/her behavior can be represented by taking average of all sessions)

Sampling

- Sampling is the main technique employed for data selection.
  - It is often used for both the preliminary investigation of the data and the final data analysis.

- Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming.

- Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming.

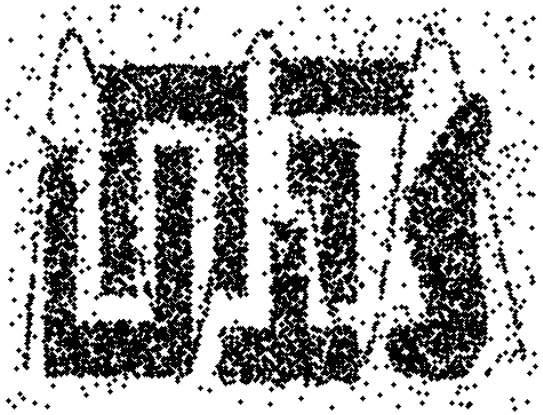
Sampling …

- The key principle for effective sampling is the following:
  - using a sample will work almost as well as using the entire data sets, if the sample is representative
  - A sample is representative if it has approximately the same property (of interest) as the original set of data

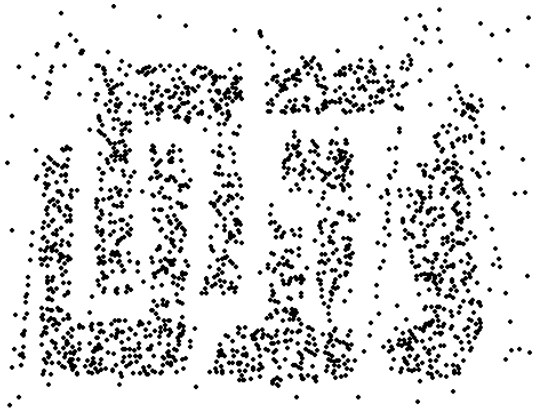
Types of Sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
- Sampling without replacement
  - As each item is selected, it is removed from the population
- Sampling with replacement
  - Objects are not removed from the population as they are selected for the sample.
    - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition

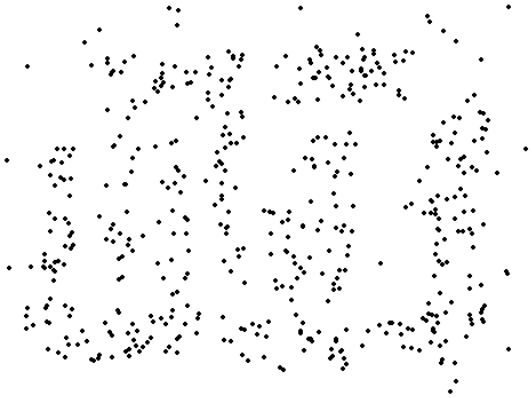
Sample Size



8000 points



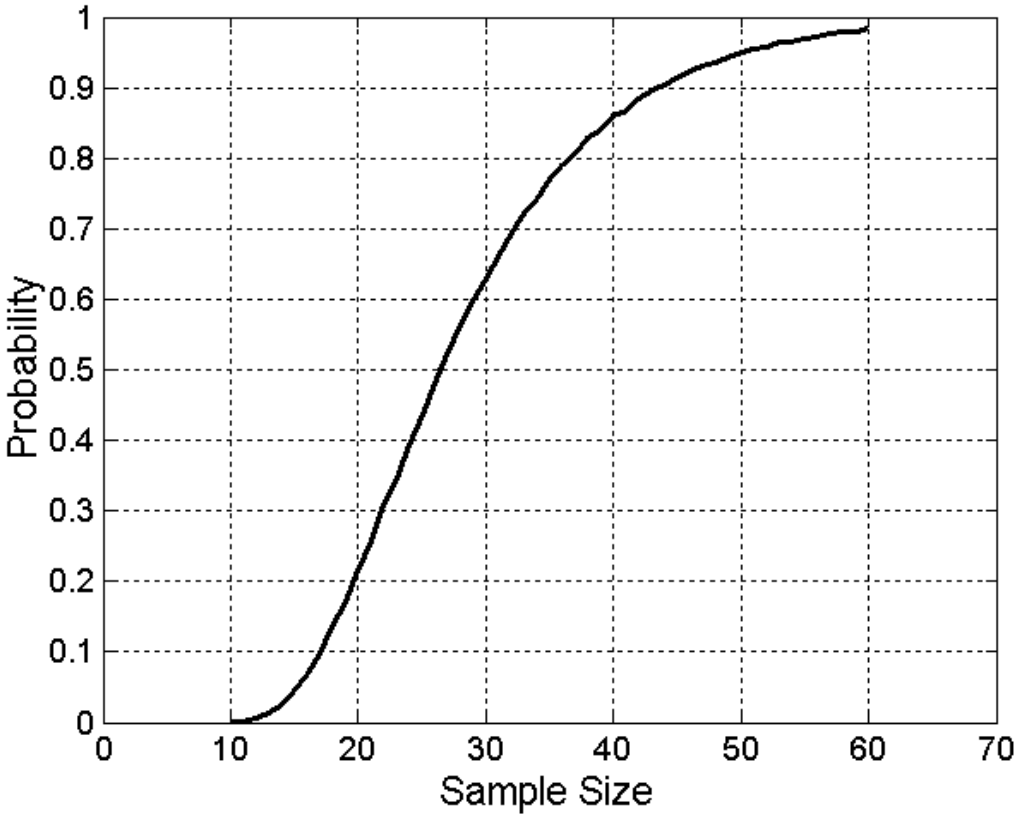
2000 Points



500 Points

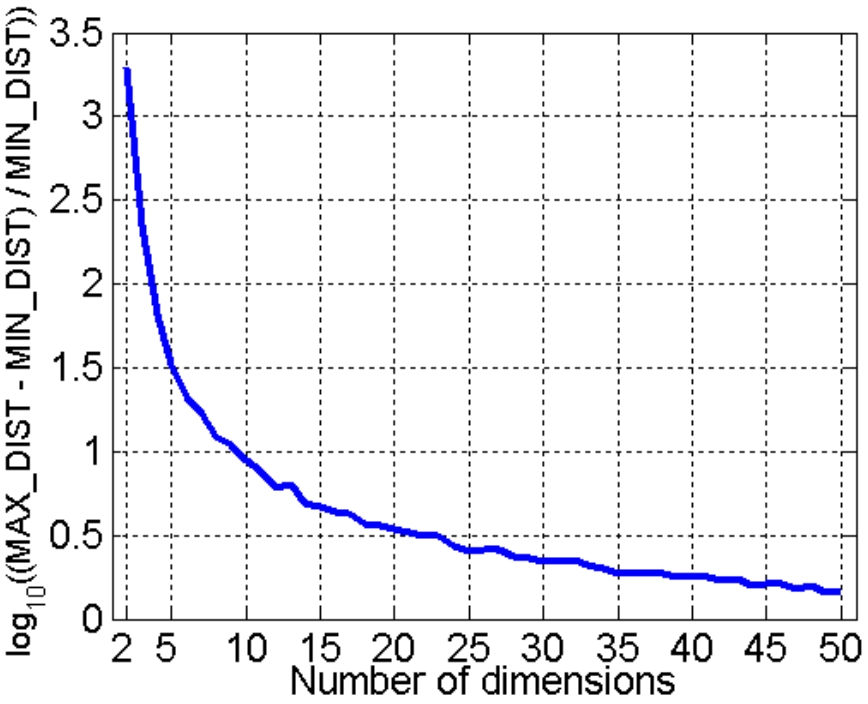
Sample Size

- What sample size is necessary to get at least one object from each of 10 groups.



Curse of Dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful



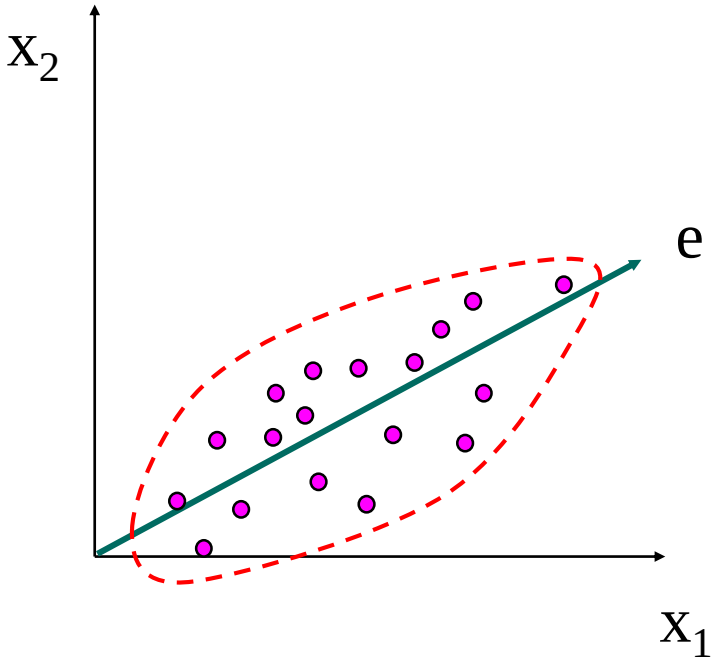
- **Randomly generate 500 points**
- **Compute difference between max and min distance between any pair of points**

Dimensionality Reduction

- Purpose:
  - Avoid curse of dimensionality
  - Reduce amount of time and memory required by data mining algorithms
  - Allow data to be more easily visualized
  - May help to eliminate irrelevant features or reduce noise
- Techniques
  - Principle Component Analysis
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

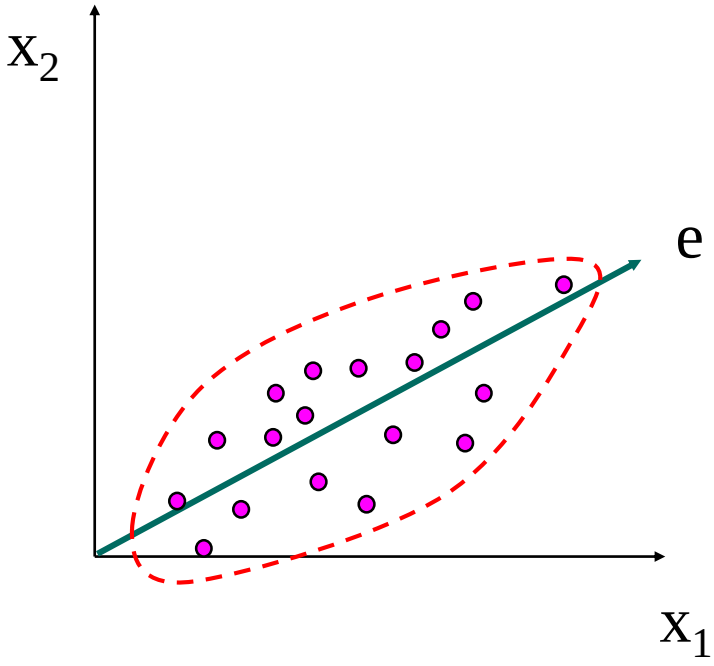
Dimensionality Reduction: PCA

- Goal is to find a projection that captures the largest amount of variation in data



Dimensionality Reduction: PCA

- Find the eigenvectors of the covariance matrix
- The eigenvectors define the new space



Feature Subset Selection

- Another way to reduce dimensionality of data

- Redundant features
  - duplicate much or all of the information contained in one or more other attributes
  - Example: purchase price of a product and the amount of sales tax paid

- Irrelevant features
  - contain no information that is useful for the data mining task at hand
  - Example: students' ID is often irrelevant to the task of predicting students' GPA

Feature Subset Selection

- Techniques:
  - Brute-force approch:
    - Try all possible feature subsets as input to data mining algorithm
  - Embedded approaches:
    - Feature selection occurs naturally as part of the data mining algorithm
  - Filter approaches:
    - Features are selected before data mining algorithm is run
  - Wrapper approaches:
    - Use the data mining algorithm as a black box to find best subset of attributes

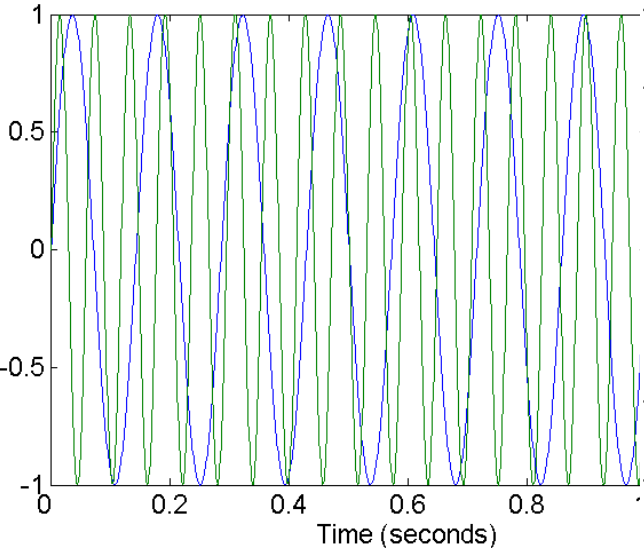
Feature Creation

- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes

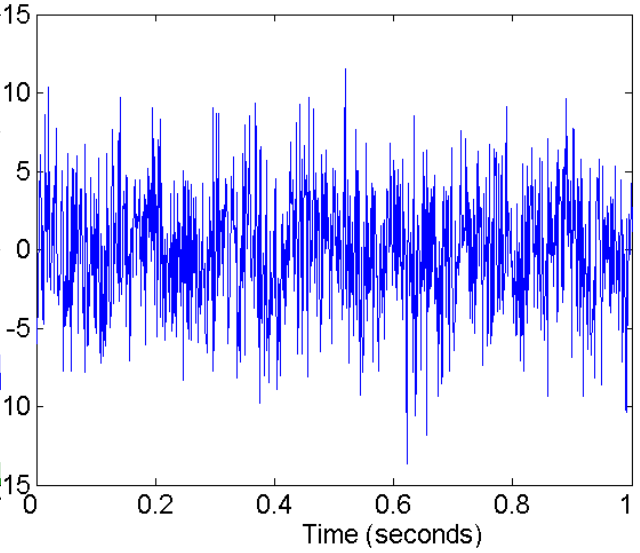
- Three general methodologies:
  - Feature Extraction
    - domain-specific
  - Mapping Data to New Space
  - Feature Construction
    - combining features

Mapping Data to a New Space

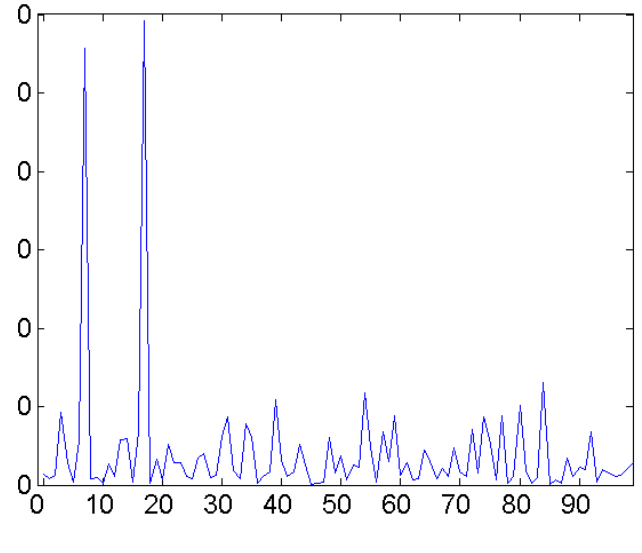
- Fourier transform
- Wavelet transform



Two Sine Waves



Two Sine Waves + Noise



Frequency

Discretization

- Three types of attributes
  - Nominal—values from an unordered set, e.g., color, profession
  - Ordinal—values from an ordered set, e.g., military or academic rank
  - Numeric—real numbers, e.g., integer or real numbers
- Discretization: Divide the range of a continuous attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Reduce data size by discretization
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
  - Prepare for further analysis, e.g., classification

Data Discretization

Data discretization transforms numeric data by mapping values to interval or concept labels.

- **Data discretization by binning**: This is a top-down unsupervised splitting technique based on a specified number of bins.
- **Data discretization by histogram analysis**: In this technique, a histogram partitions the values of an attribute into disjoint ranges called buckets or bins. It is also an unsupervised method.
- **Data discretization by cluster analysis**: In this technique, a clustering algorithm can be applied to discretize a numerical attribute by partitioning the values of that attribute into clusters or groups.
- **Data discretization by decision tree analysis**: Here, a decision tree employs a top-down splitting approach; it is a supervised method. To discretize a numeric attribute, the method selects the value of the attribute that has minimum entropy as a split-point, and recursively partitions the resulting intervals to arrive at a hierarchical discretization.
- **Data discretization by correlation analysis**: This employs a bottom-up approach by finding the best neighboring intervals and then merging them to form larger intervals, recursively. It is supervised method.

Simple Discretization: Binning

- Equal-width (distance) partitioning
  - Divides the range into *N* intervals of equal size: uniform grid
  - 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
- 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

Binning Methods for Data Smoothing

The binning method can be used for smoothing the data (removing noise)

Unsorted data for price in dollars

Before sorting: 8 16, 9, 15, 21, 21, 24, 30, 26, 27, 30, 34

First of all, sort the data

After Sorting: 8, 9, 15, 16, 21, 21, 24, 26, 27, 30, 30, 34

Smoothing the data by equal frequency bins (Equal Depth)

Bin 1: 8, 9, 15, 16

Bin 2: 21, 21, 24, 26,

Bin 3: 27, 30, 30, 34

Smoothing by bin mean value

For Bin 1:

(8+ 9 + 15 +16 / 4) = 12

Bin 1 = 12, 12, 12, 12

For Bin 2:

(21 + 21 + 24 + 26 / 4) = 23

Bin 2 = 23, 23, 23, 23

For Bin 3:

(27 + 30 + 30 + 34 / 4) = 30

Bin 3 = 30, 30, 30, 30

Smoothing by bin boundaries

How to smooth data by bin boundaries?

Put the minimum on the left side and maximum on the right side.

Middle values in bin boundaries move to its closest neighbor value with less distance.

Unsorted data for price in dollars:

Data: 8, 9, 15, 16, 21, 21, 24, 26, 27, 30, 30, 34

Smoothing the data by equal frequency bins

Bin 1: 8, 9, 15, 16

Bin 2: 21, 21, 24, 26,

Bin 3: 27, 30, 30, 34

Smooth data after bin Boundary

Before bin Boundary: Bin 1: 8, 9, 15, 16

Here, 8 is the minimum value and 16 is the maximum value.9 is near to 8, so 9 will be treated as 8. 15 is more near to 16 and farther away from 8. So, 15 will be treated as 16.

After bin Boundary: Bin 1: 8, 8, 16, 16

Before bin Boundary: Bin 2: 21, 21, 24, 26,

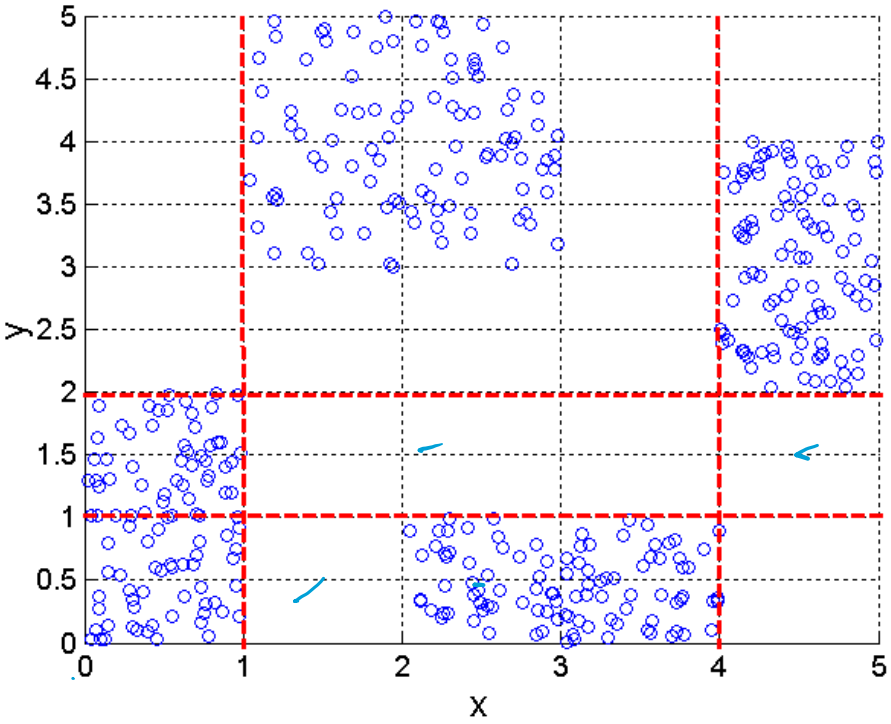
After bin Boundary: Bin 2: 21, 21, 26, 26,

Before bin Boundary: Bin 3: 27, 30, 30, 34

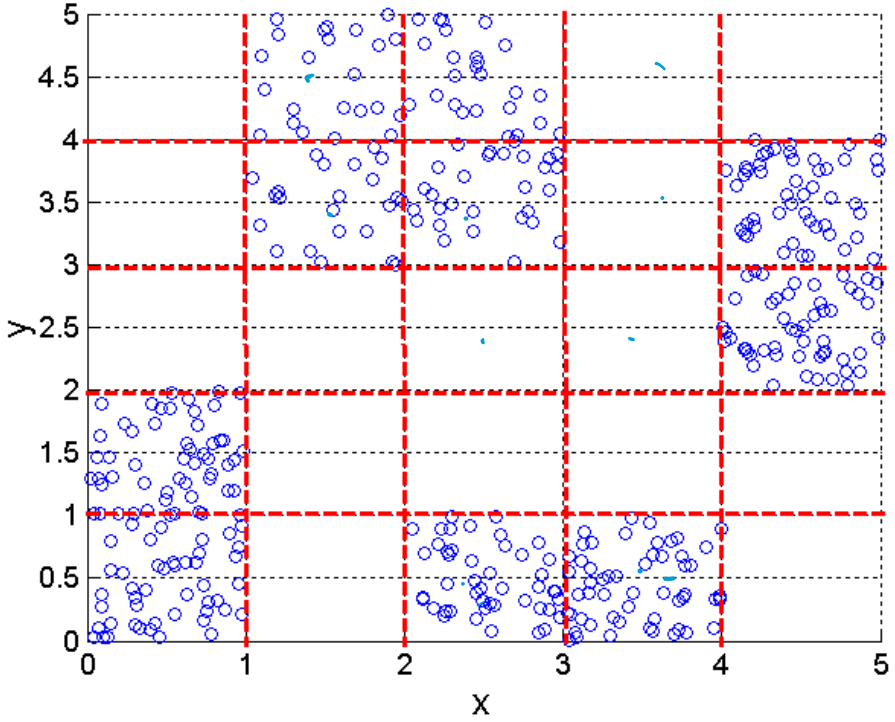
After bin Boundary: Bin 3: 27, 27, 27, 34

Discretization Using Class Labels

- **Entropy based approach**

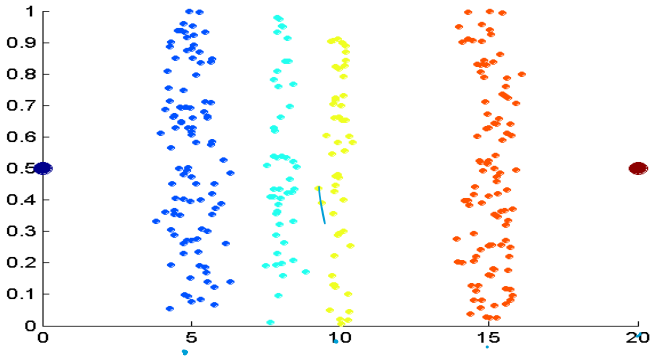


3 categories for both x and y

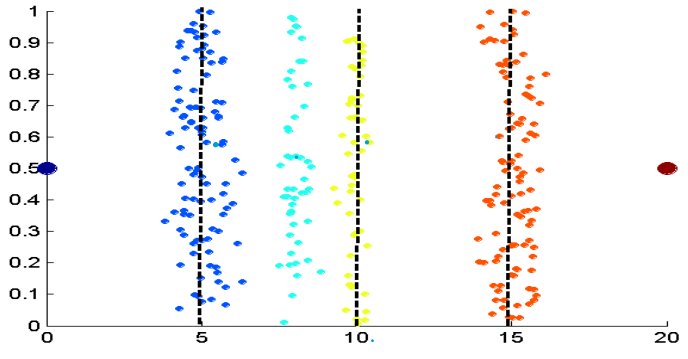


5 categories for both x and y

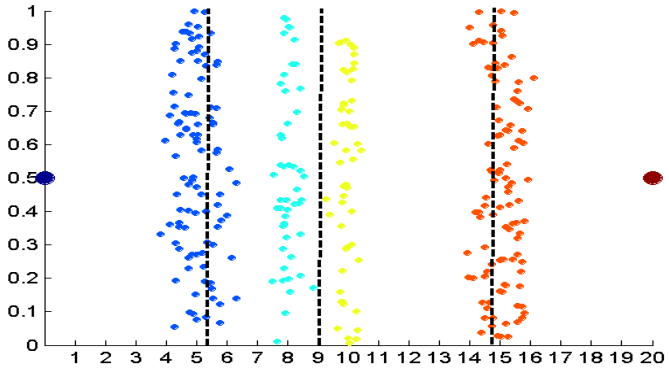
Discretization Without Using Class Labels



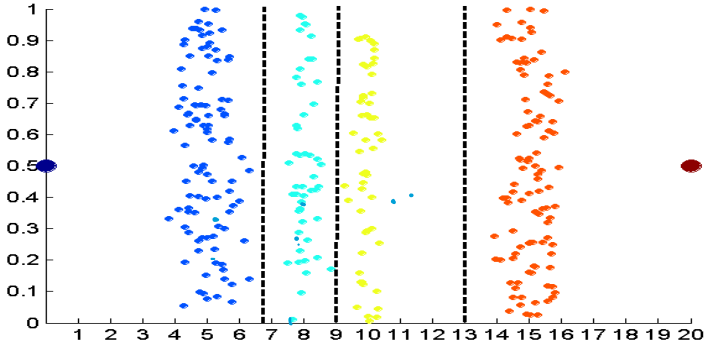
Data



Equal interval width



Equal frequency



K-means

Advantages (Pros) of data smoothing

Data smoothing clears the understandability of different important hidden patterns in the data set.

Data smoothing can be used to help predict trends. Prediction is very helpful for getting the right decisions at the right time.

Data smoothing helps in getting accurate results from the data.

Cons of data smoothing

Data smoothing doesn’t always provide a clear explanation of the patterns among the data.

It is possible that certain data points being ignored by focusing the other data points.

Discretization by Classification & Correlation Analysis

- Classification (e.g., decision tree analysis)
  - Supervised: Given class labels, e.g., cancerous vs. benign
  - Using *entropy* to determine split point (discretization point)
  - Top-down, recursive split
- Correlation analysis (e.g., Chi-merge: χ^2-based discretization)
  - Supervised: use class information
  - Bottom-up merge: find the best neighboring intervals (those having similar distributions of classes, i.e., low χ^2 values) to merge
  - Merge performed recursively, until a predefined stopping condition

Normalization

Normalization: used to scale the data of an attribute in range -1.0 to 1.0 or 0.0 to 1.0. It is generally useful for classification algorithms.

Need of Normalization –
Required when we are dealing with attributes on a different scale
It may lead to a dilution in effectiveness of an important equally important attribute (on lower scale) because of other attribute having values on larger scale.

Normalized to bring all the attributes on the same scale.

person_name	Salary	Year_of_ experience	Expected Position Level
Aman	100000	10	2
Abhinav	78000	7	4
Ashutosh	32000	5	8
Dishi	55000	6	7
Abhishek	92000	8	3
Avantika	120000	15	1
Ayushi	65750	7	5

The attributes salary and year_of_experience are on different scale and hence attribute salary can take high priority over attribute year_of_experience in the model.

Min-Max Normalization

Normalization means transforming or mapping the data to a smaller or common range. All attributes gain an equal weight after this process.

- **Min-max normalization**: This preserves the relationships among the original data values and performs a linear transformation on the original data. The applicable ones of the actual maximum and minimum values of an attribute will be normalized in 0 to 1.

$$v' = \frac{v - \min(A)}{\max(A) - \min(A)}(\text{new\_max}(A) - \text{new\_min}(A)) + \text{new\_min}(A)$$

- Where, A is the attribute data,
  Min(A), Max(A) are the minimum and maximum absolute value of A respectively.
  v' is the new value of each entry in data.
  v is the old value of each entry in data.
  new_max(A), new_min(A) is the max and min value of the range(i.e boundary value of range required) respectively.

z-score normalization (Zero-Mean)

- **z-score normalization (Zero-Mean)**: Here the values for an attribute are normalized based on the mean and standard deviation of that attribute.
- It is useful when the actual minimum and maximum of an attribute to be normalized are unknown.

$$v' = \frac{v - \overline{A}}{\sigma_A}$$

- v' and v is the new and old of each entry in data respectively. σ_A, and $\overline{A}$ is the standard deviation and mean of A respectively.
- Takes care of outliers but does not provide data normalization with an identical scale.

Comparison of Min-Max Normalization and Z-Score Normalization

Min-max normalization	Z-score normalization
Not very well efficient in handling the outliers	Handles the outliers in a good way.
Min-max Guarantees that all the features will have the exact same scale.	Helpful in the normalization of the data but not with the *exact* same scale.

Normalization by decimal scaling

Normalization by decimal scaling: This normalizes by moving the decimal point of values of attribute.

It normalizes by moving the decimal point of values of the data. To normalize the data by this technique, we divide each value of the data by the maximum absolute value of data.

Where, *j* is the smallest integer such that max(|v_i'|)<1.

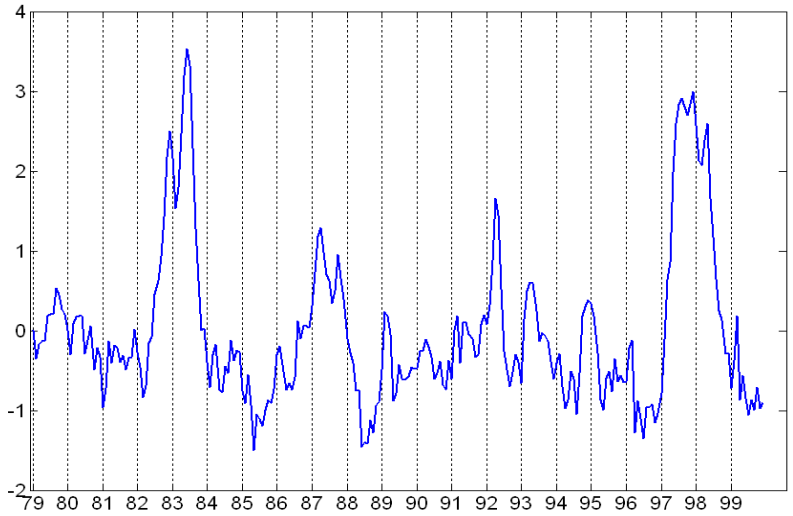
$$v_i' = \frac{v_i}{10^j}$$

- *Ex: input data is: -10, 201, 301, -401, 501, 601, 701*
- *To normalize the above data,*
  ***Step 1:** Maximum absolute value in given data(m): 701*
  ***Step 2:** Divide the given data by 1000 (i.e j=3=no of digits comprising the number)*
- ***Result:** The normalized data is: -0.01, 0.201, 0.301, -0.401, 0.501, 0.601, 0.701*

- It follows that the means of the normalized data will always be between 0 and 1.

Attribute Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
  - Simple functions: x^k, log(x), e^x, |x|
  - Standardization and Normalization



Similarity and Dissimilarity

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

Similarity/Dissimilarity for Simple Attributes

p and *q* are the attribute values for two data objects.

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert p-q \rvert}{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{\lvert p-q \rvert}{n-1}$
Interval or Ratio	$d = \lvert p - q \rvert$	$s = -d$, $s = \frac{1}{1+d}$ or $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

Table 5.1. Similarity and dissimilarity for simple attributes

Distance Measures

- Remember K-Nearest Neighbor are determined on the bases of some kind of “distance” between points.

- Two major classes of distance measure:
  1. *Euclidean* : based on position of points in some *k* -dimensional space.
  2. *Noneuclidean* : not related to position or space.

Distance Measures

For a pair of vectors (data points, or objects, or rows of a table), we can use some distance measures to compute how different or similar the vectors are.

Euclidean distance:

The distance between points

$A(x_1, y_1)$ and $B(x_2, y_2)$ is equal to

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$A(f_1, f_2, f_3, f_4, \ldots, f_n)$ $B(g_1, g_2, g_3, g_4, \ldots, g_n)$

$$\sqrt{(f_1 - g_1)^2 + (f_2 - g_2)^2 + (f_3 - g_3)^2 + (f_4 - g_4)^2 + \ldots + (f_n - g_n)^2}$$

$$\sqrt{\sum_{i=1}^{n} (f_i - g_i)^2}$$

Therefore, the distance between Row 2 and Row 5 is equal to

$$\sqrt{(5-9)^2 + (4-2)^2} = 4.472135954999579$$

	Feature 1	Feature 2
Row 1	10	3
Row 2	5	4
Row 3	10	4
Row 4	8	6
Row 5	9	2

L_2 Norm = $\|A - B\| = \sqrt{\sum_{i=1}^{n} (f_i - g_i)^2}$

Manhattan distance

A simpler difference in every dimension can be computed without squaring the differences and without using the square root.

That is, just sum up the differences between the two vectors in every dimension.

The measure that just sums up the differences in each dimension of two points or vectors

$A(f_1, f_2, f_3, f_4, \ldots, f_n)$ and $B(g_1, g_2, g_3, g_4, \ldots, g_n)$

Manhattan Distance between A and B =

$$|A - B|_1 = \sum_{i=1}^{n} |f_i - g_i|$$

Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$dist = \left(\sum_{i=1}^{n} |f_i - g_i|^r \right)^{\frac{1}{r}}$$

Minkowski Distance: Examples

- $r = 1$. City block (Manhattan, taxicab, L_1 norm) distance.
  - A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors
- $r = 2$. Euclidean distance
- $r \rightarrow \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component of the vectors
- Do not confuse r with n, i.e., all these distances are defined for all numbers of dimensions.

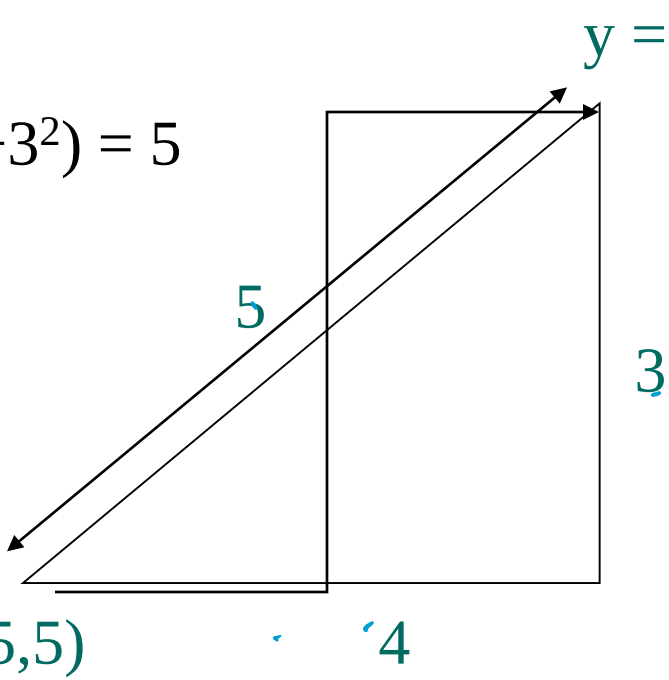
Another Euclidean Distance

- L_∞ *norm* : d(x,y) = the maximum of the differences between *x* and *y* in any dimension.

Examples L_1 and L_2 norms

L_2-norm:

dist(x,y) = $\sqrt{(4^2+3^2)} = 5$

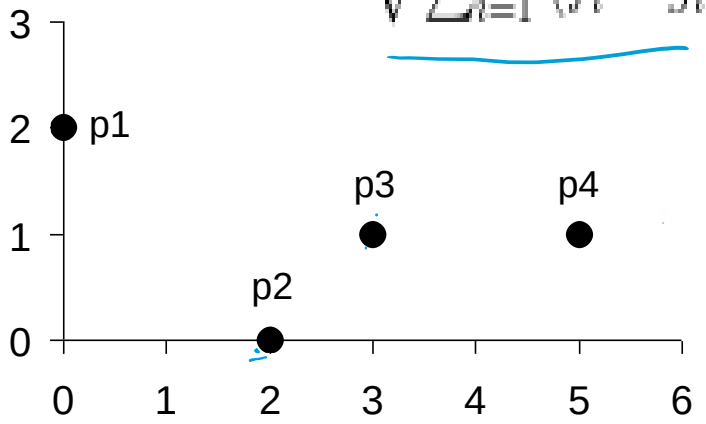


L_1-norm:

dist(x,y) = 4+3 = 7

Euclidean Distance

$$\sqrt{\sum_{i=1}^{n} (f_i - g_i)^2}$$



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

Minkowski Distance

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

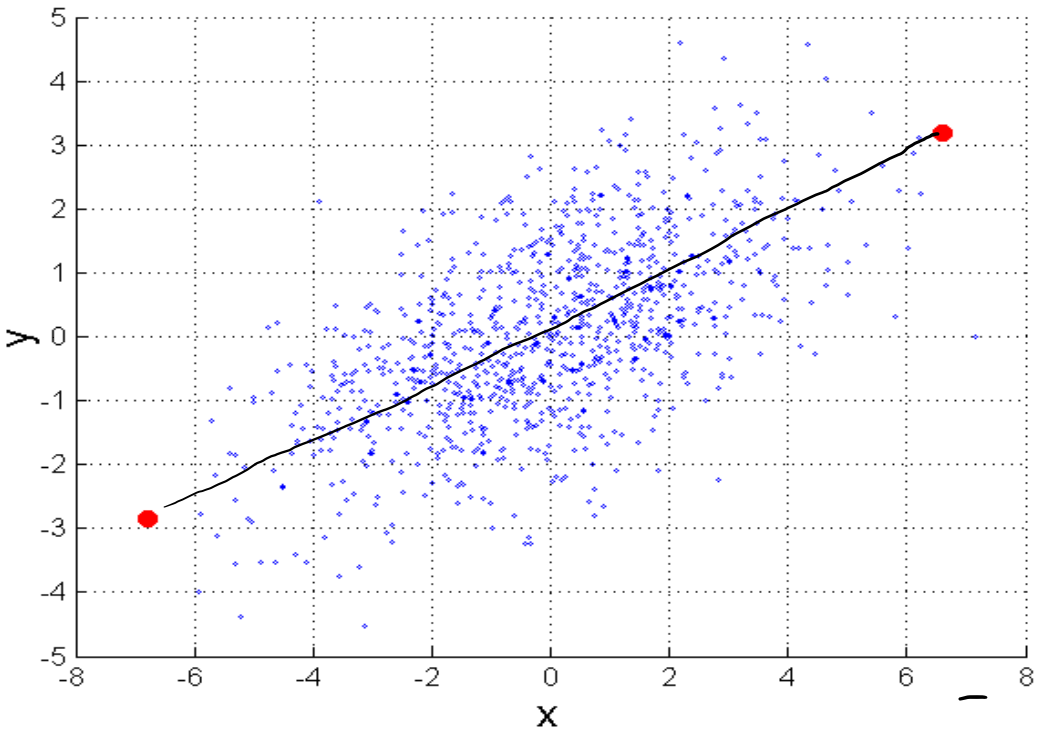
L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Distance Matrix

Mahalanobis Distance

$$mahalanobis(p,q) = (p-q)\Sigma^{-1}(p-q)^T$$



Σ is the covariance matrix of the input data *X*

$$\Sigma_{j,k} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{ij} - \overline{X}_j)(X_{ik} - \overline{X}_k)$$

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.

Hamming Distance

Hamming distance between two strings (same length) is the number of positions where the strings have different letters.

Example: compute the distance between the following two strings.

apricot
Abrikop

The two strings are different in three positions : 2, 5, and 7.

Hamming distance (apricot, abrikop) = 3

Used to compute the distance between binary bit vectors of the same length.

Sometimes, binary streams are repeated during transmission to increase reliability.

Hamming distance is a true distance measure and satisfies the four distance properties.

Edit distance (LCS-based)

The edit distance between two strings is the minimum number of edits (delete or insert) required to convert one string to the other.

Please note that given an infinite number of edits it is always possible to convert one string to another. We are talking about the “minimum” number of edits.

Example: consider that we have two strings S_1 and S_2.

S_1=xyzmn S_2=yzmopn

Let us try to convert S_1 to S_2.

1. Delete x from S_1. S_1 is now yzmn.
2. Insert o in the fourth position. S_1 now becomes yzmon.
3. Insert p in the fifth position. S_1 now becomes yzmopn.
4. Notice that S_1 has become S_2=yzmopn.

To convert S_1 to S_2, we needed three edits (one delete and two insertions.) We cannot do this conversion with any lesser edits than 3.

Therefore, the edit distance between xyzmn and yzmopn is = 3.

edit distance (S_1, S_2)=edit distance (S_2, S_1)

The alternative way to compute the edit distance: based on the length of the longest common subsequence (LCS)

$$\text{edit distance } (S_1, S_2) = |S_1| + |S_2| - 2 * |\text{LCS } (S_1, S_2)|$$

$|S_1|$ or $|S_2|$ indicates the length of the corresponding string.
$|\text{LCS } (S_1, S_2)|$ is the length of the longest common subsequence between both strings S_1 and S_2 from left to right

Example: consider strings S_1=xyzmn and S_2=yzmopn.

yzm is present in both the strings from left to right.

However, the longest sequence that is present in both the strings from left to right is yzmn.

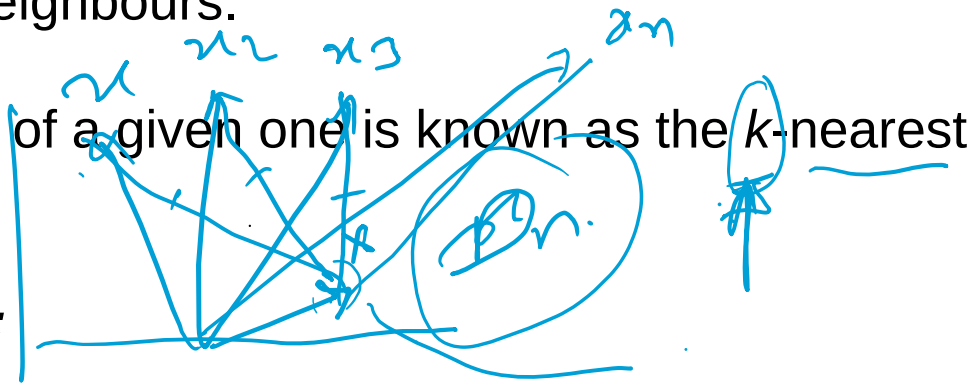
Edit distance(S_1, S_2) = edit distance(xyzmn, yzmopn)

$$= |xyzmn| + |yzmopn| - 2\,|yzmn| = 5+6-2*4 = 3$$

Nearest Neighbours

Given a data point, finding several closest points is called the computation of the nearest neighbours.

Finding *k* nearest data points of a given one is known as the *k*-nearest neighbours *(knn)* problem.



Problem statement for *knn:*

Given a vector and a data set D, order all vectors of D such that

$$D = \{x_1, x_2, x_3, \ldots, x_n\} \quad \text{distance}(x, x_i) \leq \text{distance}(x, x_{i+1})$$

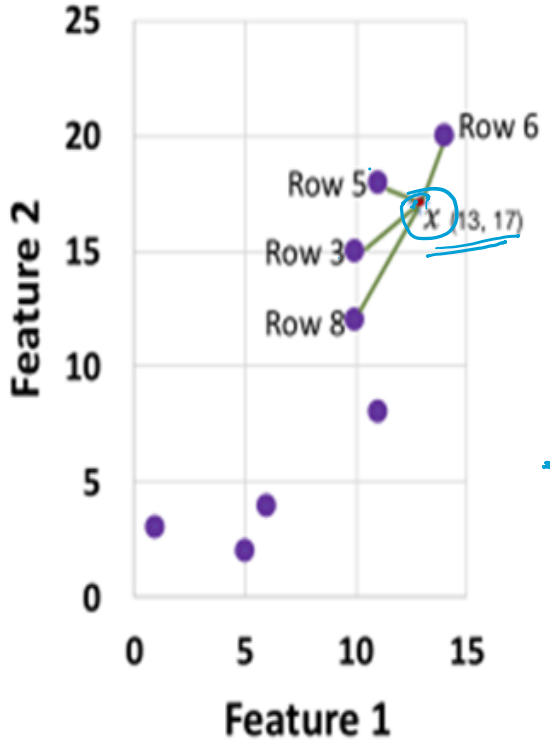
Return the first K vectors

$$S = \{x_1, x_2, x_3, \ldots, x_k\}$$

For numerical objects, the length of must be equal to the number of features/dimensions in D to be able to use a distance or similarity measure. *knn* returns the indices (row serial number) of the top *k*-nearest neighbours, instead of returning *k* complete vectors from the data.

Example of *knn*

	Feature 1	Feature 2
Row 1	5	2
Row 2	1	3
Row 3	10	15
Row 4	6	4
Row 5	11	18
Row 6	14	20
Row 7	11	8
Row 8	10	12



With k=4 and a given point x, a *knn* function will return the following row IDs.

Row 5
Row 6
Row 3
Row 8

Row 5 is the 1st nearest neighbor.
Row 6 is the 2nd nearest neighbor.
Row 3 is the 3rd nearest neighbor.
Row 8 is the 4th nearest neighbor.

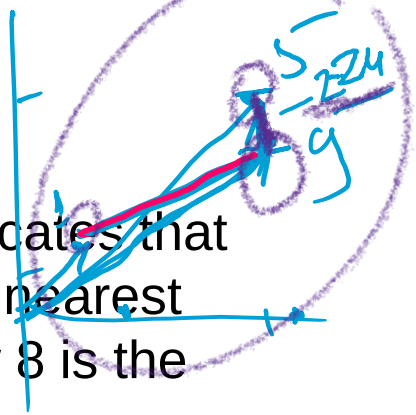
Find the 4 nearest neighbours of the point x=(13,17)

Find the 4 nearest neighbours of the point x=(13,17)

COMPUTE THE DISTANCE BETWEEN THE GIVEN POINT $x = (13, 17)$ AND EACH OF THE DATA POINTS IN THE DATA TABLE.

	Feature 1	Feature 2	Euclidean distance between x and row i	
Row 1	5	2	$\sqrt{(13-5)^2+(17-2)^2}=17.00$	Row 1 is the 7th nearest neighbor
Row 2	1	3	$\sqrt{(13-1)^2+(17-3)^2}=18.44$	Row 2 is the 8th nearest neighbor
Row 3	10	15	$\sqrt{(13-10)^2+(17-15)^2}=3.61$	Row 3 is the 3rd nearest neighbor
Row 4	6	4	$\sqrt{(13-6)^2+(17-4)^2}=14.76$	Row 4 is the 6th nearest neighbor
Row 5	11	18	$\sqrt{(13-11)^2+(17-18)^2}=2.24$	Row 5 is the 1st nearest neighbor
Row 6	14	20	$\sqrt{(13-14)^2+(17-20)^2}=3.16$	Row 6 is the 2nd nearest neighbor
Row 7	11	8	$\sqrt{(13-11)^2+(17-8)^2}=9.22$	Row 7 is the 5th nearest neighbor
Row 8	10	12	$\sqrt{(13-10)^2+(17-12)^2}=5.83$	Row 8 is the 4th nearest neighbor

First Compute Euclidean distance of x with each row.
For K=4 select first 4 from the ascending orders.



knn will return an array with content [5, 6, 3, 8], which indicates that Row 5 is the first nearest neighbour, Row 6 is the second nearest neighbour, Row 3 is the third nearest neighbour, and Row 8 is the fourth nearest neighbour.

Back to k-Nearest Neighbor (Pseudo-code)

- Missing values Imputation using k-NN.
- Input: Dataset **(*D*)**, size of ***K***

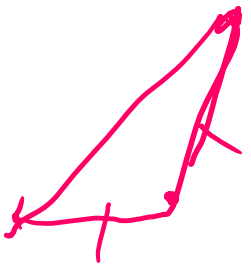
- for each record **(*x*)** with at least one missing value in ***D***.
  - for each data object **(*y*)** in ***D***.
    - Take the Distance ***(x,y)***
    - Save the distance and **y** in array Similarity **(*S*)** array.
  - Sort the array ***S*** in descending order
  - Pick the top **K** data objects from ***S***
    - Impute the missing attribute value (s) of ***x*** on the basis of known values of ***S*** (use Mean/Median or MOD).

K-Nearest Neighbor Drawbacks

- The major drawbacks of this approach are the
  - Choice of selecting exact distance functions.
  - Considering all attributes when attempting to retrieve the similar type of examples.
  - Searching through all the dataset for finding the same type of instances.
  - Algorithm Cost: ?

Common Properties of a Distance

- Distances, such as the Euclidean distance, have some well known properties.



1. $d(p, q) \geq 0$ for all p and q and $d(p, q) = 0$ only if $p = q$. (Positive definiteness, can not be negative)
2. $d(p, q) = d(q, p)$ for all p and q. (Symmetry)
3. $d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q, and r. (Triangle Inequality) The distance from one point to another cannot be greater than the distance between the same two points via another point. This is commonly known as the triangle inequality property - the length of one side of a triangle cannot be greater than the sum of the other two sides

where $d(p, q)$ is the distance (dissimilarity) between points (data objects), p and q.

- A distance that satisfies these properties is a metric

Common Properties of a Similarity

- Similarities, also have some well-known properties.

1. $s(p, q) = 1$ (or maximum similarity) only if $p = q$.

2. $s(p, q) = s(q, p)$ for all p and q. (Symmetry)

where $s(p, q)$ is the similarity between points (data objects), p and q.

Similarity Between Binary Vectors

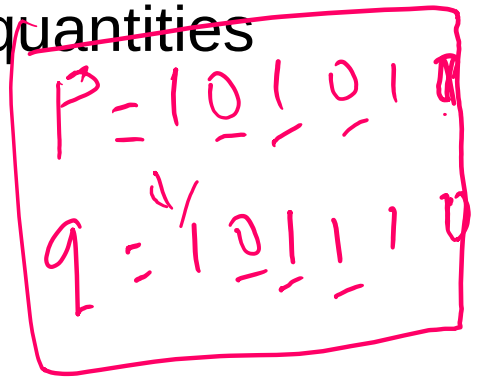
- Common situation is that objects, *p* and *q*, have only binary attributes
- Compute similarities using the following quantities

  M_{01} = the number of attributes where p was 0 and q was 1

  M_{10} = the number of attributes where p was 1 and q was 0

  M_{00} = the number of attributes where p was 0 and q was 0

  M_{11} = the number of attributes where p was 1 and q was 1



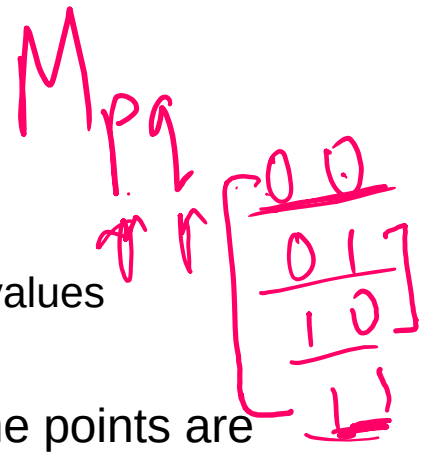
- Simple Matching and Jaccard Coefficients

  SMC = number of matches / number of attributes

  $= (M_{11} + M_{00}) / (M_{01} + M_{10} + M_{11} + M_{00})$

  J = number of 11 matches / number of not-both-zero attributes values

  $= (M_{11}) / (M_{01} + M_{10} + M_{11})$



- The high similarity between a pair of points indicates that the points are nearby. Low similarity indicates a large distance.

SMC versus Jaccard: Example

p = 1 0 0 0 0 0 0 0 0 0

q = 0 0 0 0 0 0 1 0 0 1

M_{01} = 2 (the number of attributes where p was 0 and q was 1)

M_{10} = 1 (the number of attributes where p was 1 and q was 0)

M_{00} = 7 (the number of attributes where p was 0 and q was 0)

M_{11} = 0 (the number of attributes where p was 1 and q was 1)

$$SMC = (M_{11} + M_{00})/(M_{01} + M_{10} + M_{11} + M_{00}) = (0+7) / (2+1+0+7) = 0.7$$

$$J = (M_{11}) / (M_{01} + M_{10} + M_{11}) = 0 / (2 + 1 + 0) = 0$$

Jaccard index or Jaccard coefficient

- Jaccard index/coefficient/similarity is generally computed between two sets of items.
- It is a ratio of commonality between the sets over all the items.
- If *X* and *Y* are two sets, then the Jaccard index between two sets is computed using the ratio of the size of the intersection and the size of the union of the two sets.

$$\text{Jaccard}(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

If *X= {a, b, c}*, and *Y= {b, c, d, e}* then, the size of the intersection between *X* and *Y* is:

$$|X \cap Y| = |\{b, c\}| = 2 \qquad |X \cup Y| = |\{a, b, c, d, e\}| = 5$$

$$\text{Jaccard}(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} = \frac{2}{5} = 0.4$$

Jaccard similarity varies between 0 (No similarity) to 1 (Similar).

Weighted Jaccard index/coefficient/similarity

- Jaccard index computed between two vectors/data points/objects is called a weighted Jaccard index.
- Given *X* and *Y* — two vectors each of length *n* — the formula for weighted Jaccard index or similarity between them is:

$$\text{Jaccard}(X, Y) = \frac{\sum_{k=1}^{n} \min(X_k, Y_k)}{\sum_{k=1}^{n} \max(X_k, Y_k)}$$

	Feature 1	Feature 2	Feature 3	Feature 4
Row 1	10	3	3	5
Row 2	5	4	5	3
Row 3	9	4	6	4
Row 4	8	6	2	6
Row 5	20	15	10	20

$$\text{Jaccard}(\text{Row 1}, \text{Row 3}) = \frac{9+3+3+4}{10+4+6+5} = \frac{19}{25} = 0.76$$

$$\text{Jaccard}(\text{Row 1}, \text{Row 5}) = \frac{10+3+3+5}{20+15+10+20} = \frac{21}{65} = 0.323076923$$

That means Row 3 is more like Row 1 than Row 5

The set-based Jaccard similarity discussed earlier is a special case of weighted Jaccard similarity — in the set-based Jaccard similarity, the weight of an item (feature) can be either 1 (present) or 0 (absent.)

Consider $x = \{a, b, c\}$.

	a	b	c	d	e
x	1	1	1	0	0
y	0	1	1	1	0

$$\text{Jaccard coeff.}(x, y) = \frac{0+1+1+0+0}{1+1+1+1+1}$$

$$= \frac{2}{5} = 0.4$$

Proximity Measure for Binary Attributes

- A contingency table for binary data

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

- Distance measure for symmetric binary variables:

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

- Distance measure for asymmetric binary variables:

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

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

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

- **Note: Jaccard coefficient is the same as "coherence":**

$$coherence(i,j) = \frac{sup(i,j)}{sup(i) + sup(j) - sup(i,j)} = \frac{q}{(q+r)+(q+s)-q}$$

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

i \ j	1	0
1	q	r
0	s	t

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

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

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

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

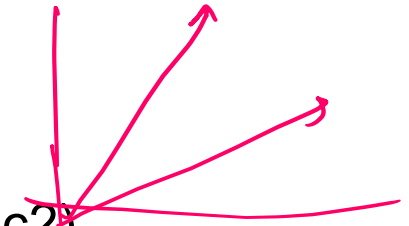
Extended Jaccard Coefficient (Tanimoto)

- Variation of Jaccard for continuous or count attributes
  - Reduces to Jaccard for binary attributes

$$T(p,q) = \frac{p \bullet q}{\|p\|^2 + \|q\|^2 - p \bullet q}$$

Cosine Similarity

- Classical approach for computational linguistics is to measure similarity based on the content overlap between documents.
- For this we will represent documents as bag-of-words, so each document will be a sparse vector. And define measure of overlap as angle between vectors:
  - Similarity (doc1, doc2) =cos(θ) = doc1doc2 / |doc1||doc2|
- By *cosine distance/dissimilarity* we assume following:
  - distance (doc1, doc2) = 1 − similarity (doc1,doc2)
- It is important to note, however, that this is not a proper distance metric in a mathematical sense as it does not have the triangle inequality property and it violates the coincidence axiom.



Cosine Similarity

- If d_1 and d_2 are two document vectors, then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / \|d_1\| \|d_2\| ,$$

where $\bullet$ indicates vector dot product and $\| d \|$ is the length of vector d.

- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**

d_2 = **1 0 0 0 0 0 0 1 0 2**

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

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

$\|d_2\| = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{0.5} = (6)^{0.5} = 2.245$

$$\cos(d_1, d_2) = .3150$$

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

- Applications: information retrieval, biologic taxonomy, gene feature mapping, ...
- Cosine measure: If d_1 and d_2 are two vectors (e.g., term-frequency vectors), then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2|| ,$$

where $\bullet$ indicates vector dot product, $||d||$: the length of vector d

Example: Cosine Similarity

- $\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \, ||d_2||$,
  where $\bullet$ indicates vector dot product, $||d|$: the length of vector d

- 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)

$d_1 \bullet d_2 = 5*3+0*0+3*2+0*0+2*1+0*1+0*1+2*1+0*0+0*1 = 25$
$||d_1|| = (5*5+0*0+3*3+0*0+2*2+0*0+0*0+2*2+0*0+0*0)^{0.5} = (42)^{0.5} = 6.481$
$||d_2|| = (3*3+0*0+2*2+0*0+1*1+1*1+0*0+1*1+0*0+1*1)^{0.5} = (17)^{0.5} = 4.12$
$\cos(d_1, d_2) = 0.94$

General Approach for Combining Similarities

- Sometimes attributes are of many different types, but an overall similarity is needed.

1. For the k^{th} attribute, compute a similarity, s_k, in the range $[0, 1]$.
2. Define an indicator variable, δ_k, for the k_{th} attribute as follows:

$$\delta_k = \begin{cases} 0 & \text{if the } k^{th} \text{ attribute is a binary asymmetric attribute and both objects have a value of 0, or if one of the objects has a missing values for the } k^{th} \text{ attribute} \\ 1 & \text{otherwise} \end{cases}$$

3. Compute the overall similarity between the two objects using the following formula:

$$similarity(p, q) = \frac{\sum_{k=1}^{n} \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

Using Weights to Combine Similarities

- May not want to treat all attributes the same.
  - Use weights w_k which are between 0 and 1 and sum to 1.

$$similarity(p,q) = \frac{\sum_{k=1}^{n} w_k \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

$$distance(p,q) = \left(\sum_{k=1}^{n} w_k |p_k - q_k|^r \right)^{1/r}$$

Correlation

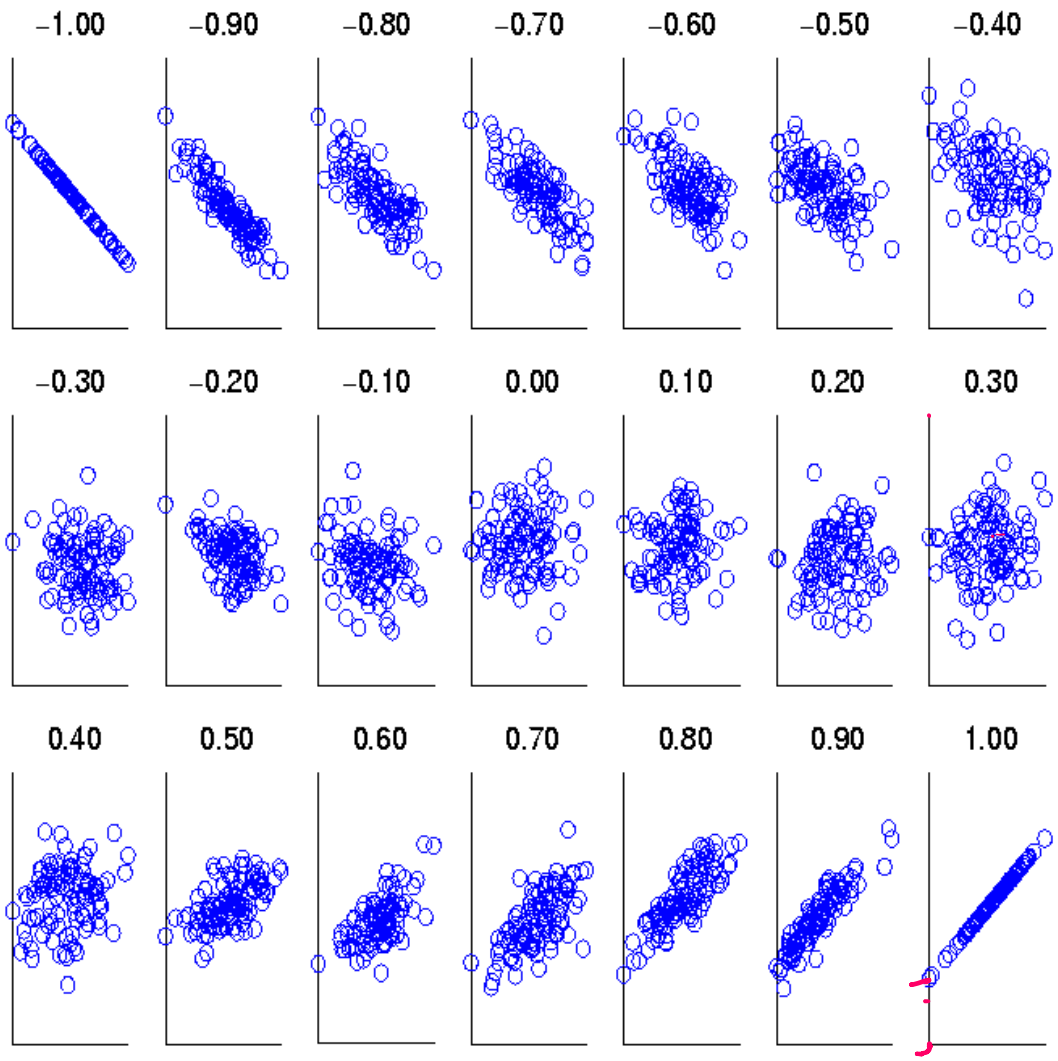
- Correlation measures the linear relationship between objects
- To compute correlation, we standardize data objects, p and q, and then take their dot product

$$p'_k = (p_k - mean(p)) / std(p)$$

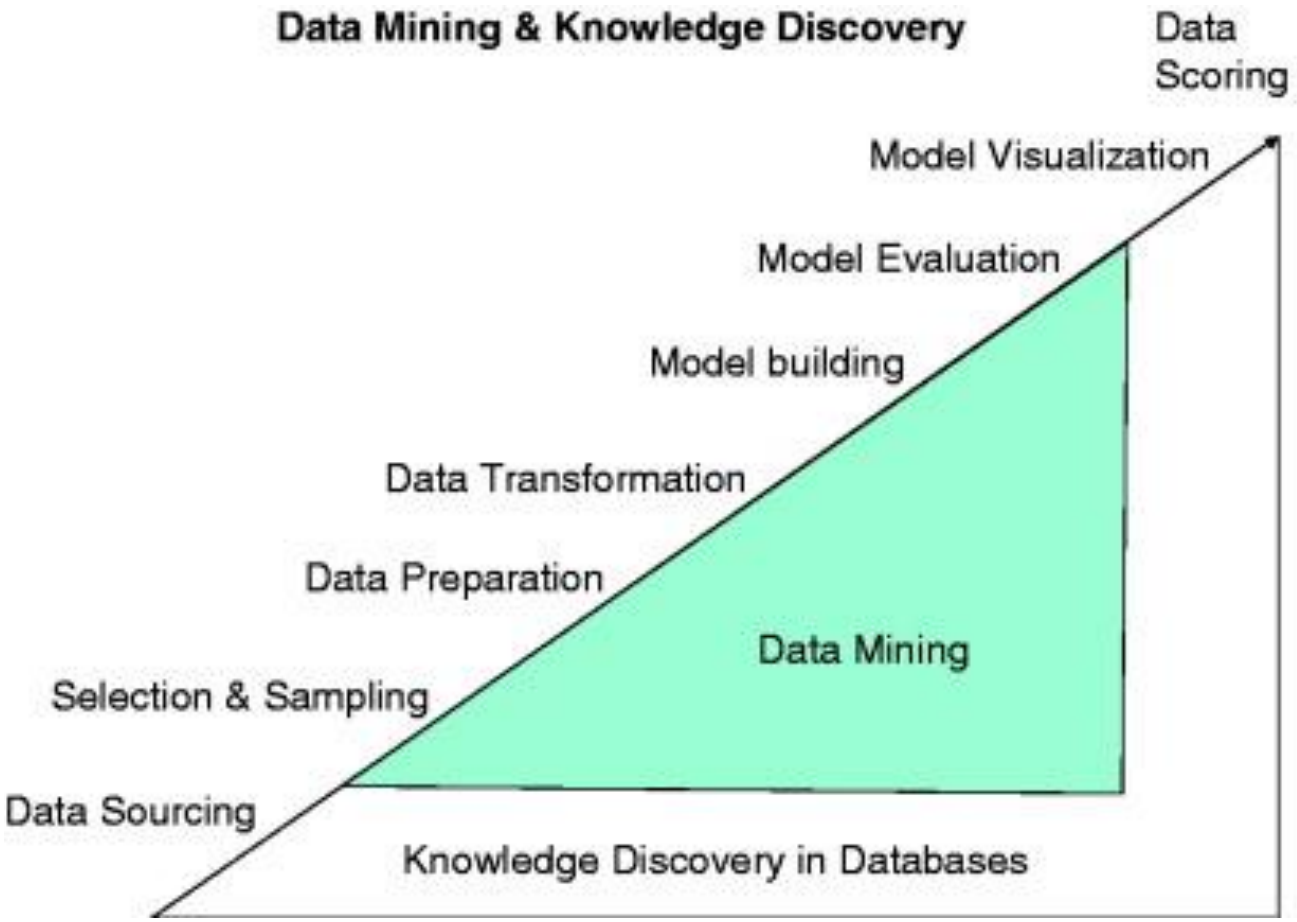
$$q'_k = (q_k - mean(q)) / std(q)$$

$$correlation(p, q) = p' \bullet q'$$

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.


Data Mining & Knowledge Discovery
Data Scoring
Model Visualization
Model Evaluation
Model building
Data Transformation
Data Preparation
Data Mining
Selection & Sampling
Data Sourcing
Knowledge Discovery in Databases