UNIVERSITY OF LIVERPOOL

Data-related Issues

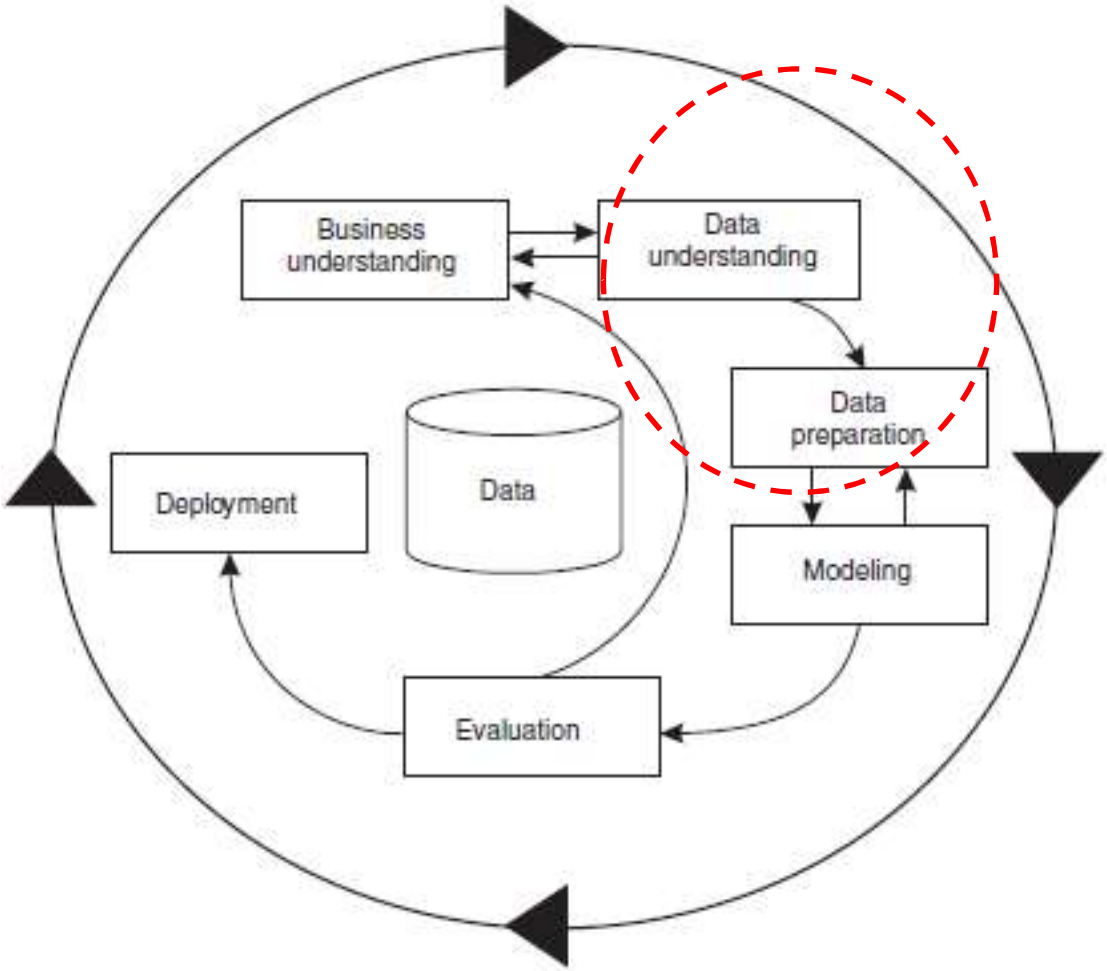
Prof. Dongping Song

University of Liverpool Management School

Email: Dongping.song@liv.ac.uk

Tan, P.N., Steinbach, M., and Kumar, V. (2014) Introduction to Data Mining, Pearson.

The CRISP-DM Process Model



CRISP=Cross-Industry Standard Process

Mariscal et al (2010). A survey of data mining and knowledge discovery process models and methodologies, Knowledge Engineering Review, 25, 137-166.

Learning Outcomes

- Data Set, Object, Attribute, Attribute Value
  - Types of Attributes and Properties of Attribute Values
  - Characteristics of Data Set
- Types of Data Set
- Data Quality & Techniques
- Data Pre-processing Techniques
- Similarity and Dissimilarity
  - Definitions and techniques

What Is a Data Set ?

- Data Set
  - Collection of objects
- Objects (have attributes)
  - Record, point, case, sample, entity or item
- Attributes (describe objects)
  - Variable, field, characteristic, feature or observation
- Attribute value & its properties

Example: Data Set

- A data set is a collection of **data objects** and their attributes
- An object is also known as record, point, case, sample, entity, or instance
- An attribute is a **property** or characteristic of an object
  - E.g. eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- A collection of attributes describe an object

Attributes

Objects

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

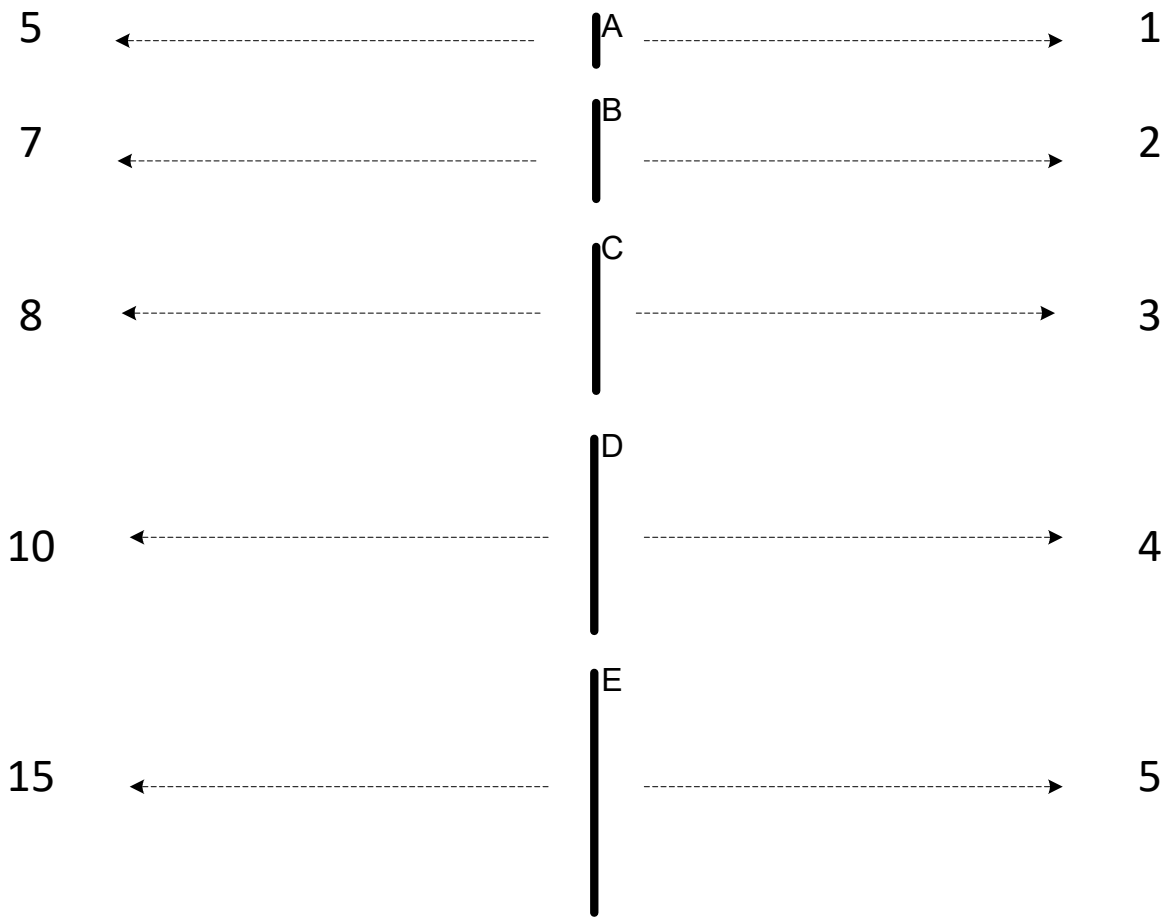
Attribute Values

- Attribute values are **numbers or symbols** assigned to an attribute
- Distinction between attributes and attribute values
  - Same attribute can be mapped to different attribute values, e.g. height can be measured in feet or meters
  - Different attributes can be mapped to the same set of values, e.g. 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

Example: Attribute Values

- Length of a Line segment: the way you measure an attribute is somewhat may not match the attributes properties.



Types of Attributes

There are different types of attributes:

- **Nominal**
  - E.g. ID numbers, eye color, zip codes
- **Ordinal**
  - E.g. rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}
- **Interval**
  - E.g. calendar dates, temperatures in Celsius or Fahrenheit.
- **Ratio**
  - E.g. temperature in Kelvin, length, time, counts

Properties of Attribute Values

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

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

Attribute Type	Description	Examples	Operations
Nominal	The values of a nominal attribute are just different names. (=, ≠)	zip codes, employee ID numbers, eye color, sex	mode, entropy, contingency correlation, χ^2 test
Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	hardness of minerals, grades, street numbers	median, percentiles, rank correlation, run tests, sign tests
Interval	The differences between values are meaningful, i.e., a unit of measurement exists. (+, -)	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, 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 one-to-one mapping, e.g. 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.

Qualitative

Quantitative

Discrete and Continuous Attributes

- **Discrete Attribute**
  - Has only a **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**
  - Has **real numbers** as attribute 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.

Example: Containership at Southampton -- Data

ShipID	Speed	GT	Carrier	Route	planned arrival	Actual arrival	Planned depature	Actual departure
1	19.4	161635	O3	FAL1	12/01/2015 06:00	12/01/2015 13:18	13/01/2015 06:00	13/01/2015 14:15
2	20.3	166269	O3	FAL1	19/01/2015 06:00	19/01/2015 10:00	20/01/2015 06:00	20/01/2015 10:03
3	21.1	173022	O3	FAL1	26/01/2015 06:00	26/01/2015 06:52	27/01/2015 06:00	27/01/2015 09:15
4	19.9	152991	O3	FAL1	02/02/2015 06:00	02/02/2015 12:30	03/02/2015 06:00	01/02/2015 08:02
5	28.4	175343	O3	FAL1	09/02/2015 06:00	09/02/2015 06:45	10/02/2015 06:00	10/02/2015 07:39
6	21.0	170269	O3	FAL1	16/02/2015 06:00	16/02/2015 07:30	17/02/2015 06:00	17/02/2015 07:13
7	20.0	165343	O3	FAL1	23/02/2015 06:00	23/02/2015 14:10	24/02/2015 06:00	24/02/2015 20:04
8	21.9	177991	O3	FAL1	02/03/2015 06:00	02/04/2015 07:50	03/03/2015 06:00	03/03/2015 11:50
9	19.5	140259	O3	FAL1	09/03/2015 06:00	09/03/2015 14:15	10/03/2015 06:00	10/03/2015 19:30
10	15.5		O3	FAL1	16/03/2015 06:00	18/03/2015 10:05	17/03/2015 06:00	19/03/2015 23:35

Important Characteristics of Data Sets

- **Dimensionality**
  - Curse of Dimensionality
  - Dimensionality reduction

the difficulties associated with analyzing high-dimensional data

- **Sparsity**
  - With asymmetric features
  - Only presence counts

- 1% of the entries are non-zero
- computation time and storage

- **Resolution**
  - Levels of resolution
  - Patterns depend on the scale

- surface of the Earth
- atmospheric pressure

A pattern may be buried in noise if too fine, or not detectable if too coarse.

Types of Data Sets

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

Time Series data: A special type of sequential data in which each record is a series of measurements taken over time

Record Data

- For the most basic form of record data, there is no explicit relationship among records or data fields
- Often stored in flat files or relational databases

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

Record Data: 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

Record Data: 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.

Sparse Data Matrix

	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

Record Data: Transaction Data

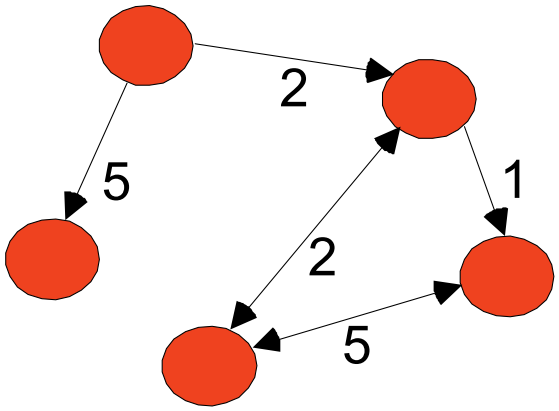
- A special type of record data, where
  - each record (transaction) involves **a set of items**.
  - For example, consider a grocery store. The set of products purchased by a customer during one shopping trip constitute a transaction, while the individual products that were purchased are the items.

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

Can be converted into Sparse Data Matrix in a database

Graph-based Data

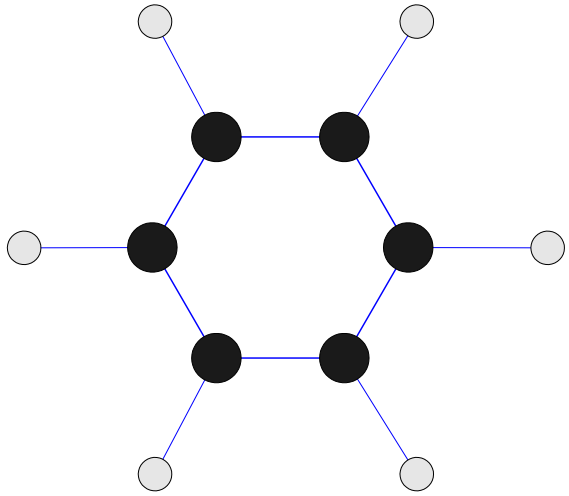
- Examples: Generic graph and HTML Links



The graph captures relations among data objects.

- Chemical data: Benzene Molecule: C_6H_6

  The nodes are atoms and the links between nodes are chemical bonds.

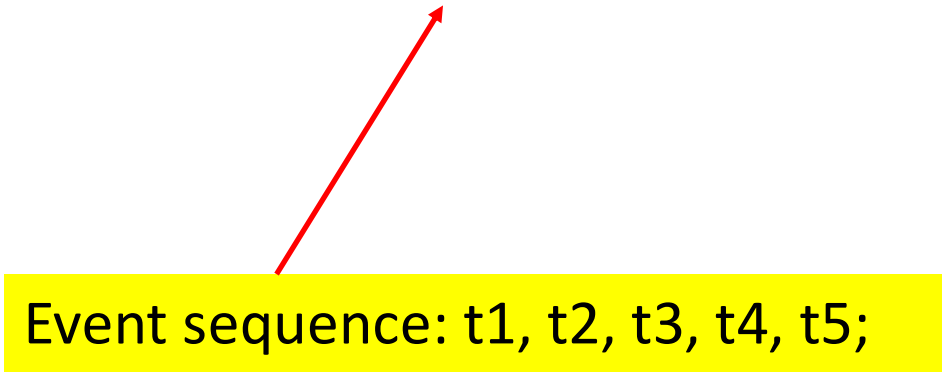


The data objects themselves are represented as graphs. Its structure is associated with properties.

Ordered Data: Sequential Data

- Sequential data (temporal data), e.g. transactions

Customer	Time and items purchased
C1	(t1: A,B); (t2: C,D); (t5: A,E)
C2	(t3: A,D); (t4: E)
C3	(t2: A,C)



Ordered Data: Sequence Data

- A section of human genetic code:

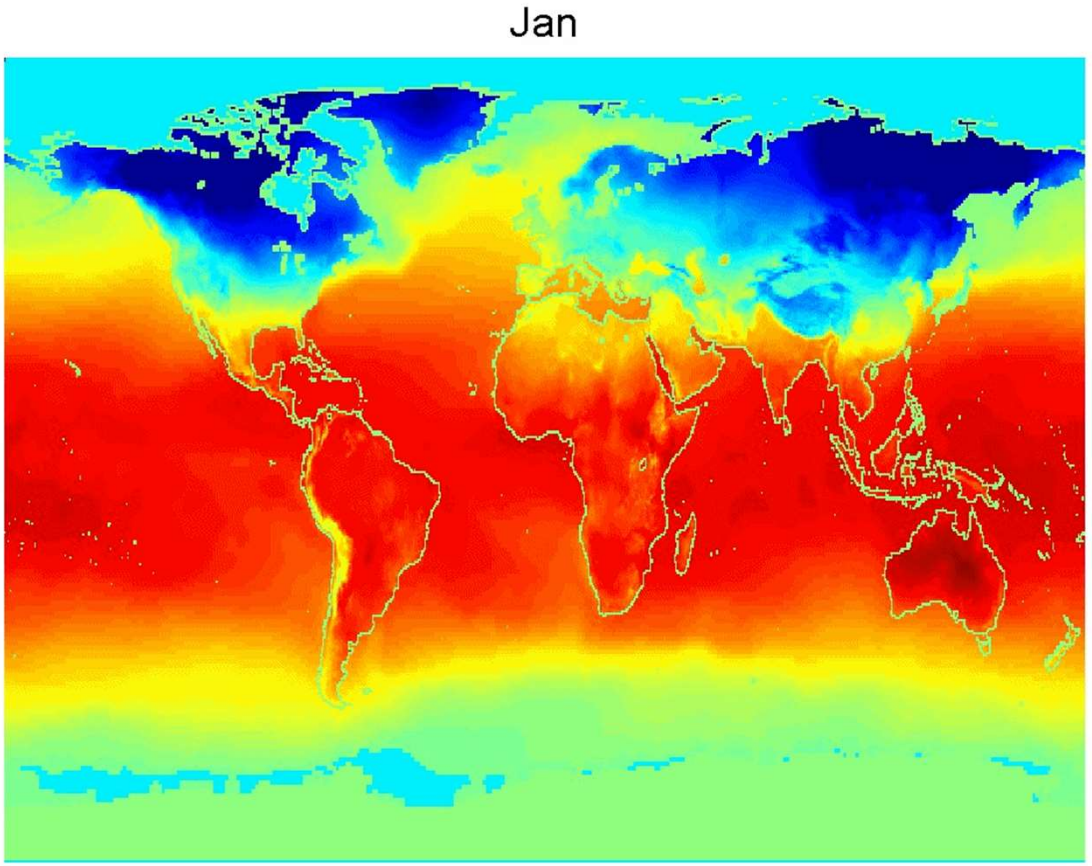
```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

Positions of letters/words are in an ordered sequence

Ordered Data: Spatial Data

- Spatial Temperature Data

Average Monthly **Temperature** of land and ocean



Discussion Question 1

Q1. You are approached by the marketing director of a local company, who believes that he has devised a foolproof way to measure customer satisfaction. He explains his scheme as follows: “It’s so simple that I can’t believe that no one has thought of it before. I just keep track of the number of customer complaints for each product. I read in a data mining book that counts are ratio attributes, and so, my measure of product satisfaction must be a ratio attribute. But when I rated the products based on my new customer satisfaction measure and showed them to my boss, he told me that I had overlooked the obvious, and that my measure was worthless. I think that he was just mad because our best-selling product had the worst satisfaction since it had the most complaints. Could you help me set him straight?”

Discussion Question 2

Q2. A few months later, you are again approached by the same marketing director. This time, he has devised a better approach to measure the extent to which a customer prefers one product over other, similar products. He explains, “When we develop new products, we typically create several variations and evaluate which one customers prefer. Our standard procedure is to give our test subjects all of the product variations at one time and then ask them to rank the product variations in order of preference. However, our test subjects are very indecisive, especially when there are more than two products. As a result, testing takes forever. I suggested that we perform the comparisons in pairs and then use these comparisons to get the rankings. Thus, if we have three product variations, we have the customers compare variations 1 and 2, then 2 and 3, and finally 3 and 1. Our testing time with my new procedure is a third of what it was for the old procedure, but the employees conducting the tests complain that they cannot come up with a consistent ranking from the results. And my boss wants the latest product evaluations, yesterday. I should also mention that he was the person who came up with the old product evaluation approach. Can you help me?”

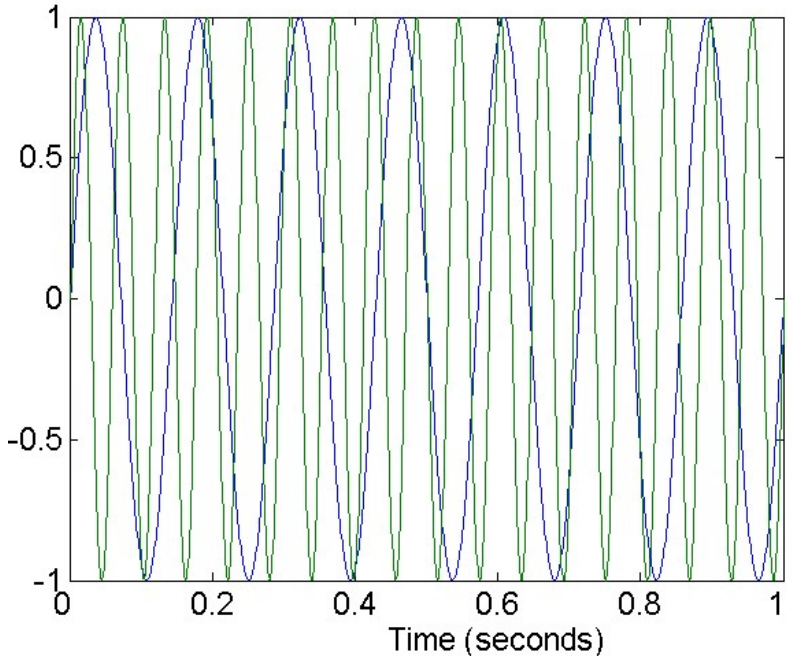
Data Quality

- What kinds of data quality problems exist?
- 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
  - Inconsistent values

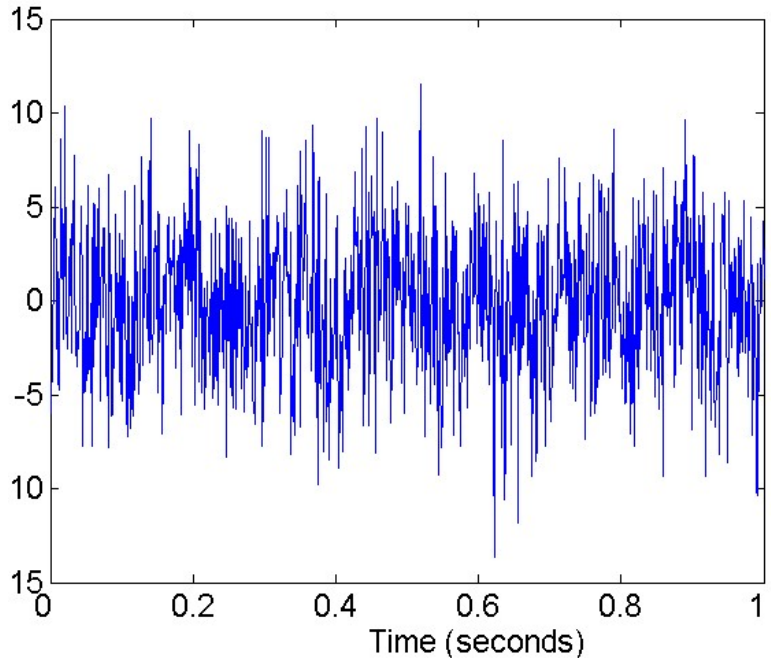
Data cleaning: detection and correction of data quality problems.

Data Quality: Noise

- Noise refers to **random modification** of original values
  - Examples: distortion of a person's voice when talking on a poor phone and "snow" on television screen



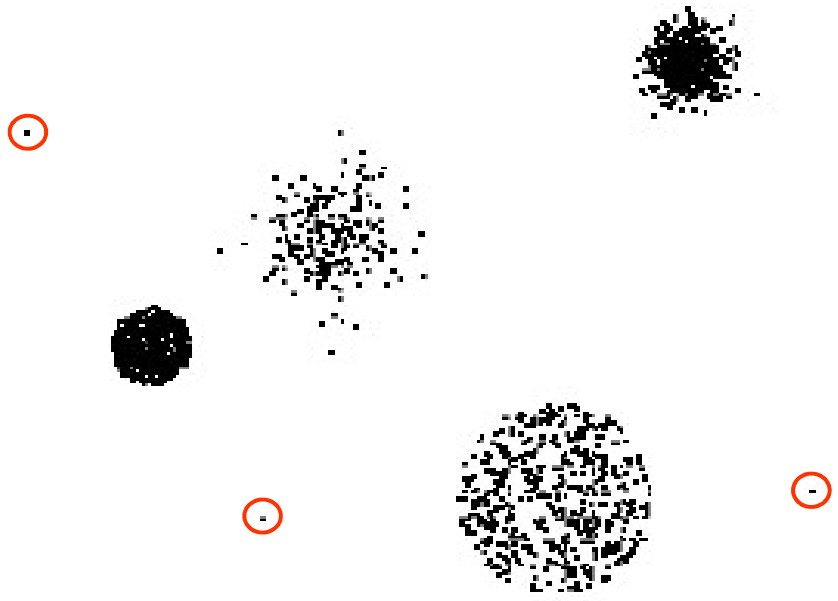
Two Sine Waves



Two Sine Waves + Noise

Data Quality: Outliers

- Outliers are data objects with characteristics that are **considerably different** from most of the other data objects in the data set



Data Quality: Missing Values

- **Reasons** for missing values
  - Information is not collected
    (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)
- **Approaches** to handle missing values
  - Eliminate data objects
  - Estimate missing values (e.g. time series)
  - Ignore the missing value during analysis (by algorithms)

Data Quality: Inconsistent Values & Duplicate Data

- Data object can contain inconsistent values, e.g. zip code and city.
- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeneous sources
  - Same person with multiple email addresses
  - **De-duplication**: the process of dealing with duplicate data issues

Example: Containership at Southampton – data quality

ShipID	Speed	GT	Carrier	Route	planned arrival	Actual arrival	Planned depature	Actual departure
1	19.4	161635	O3	FAL1	12/01/2015 06:00	12/01/2015 13:18	13/01/2015 06:00	13/01/2015 14:15
2	20.3	166269	O3	FAL1	19/01/2015 06:00	19/01/2015 10:00	20/01/2015 06:00	20/01/2015 10:03
3	21.1	173022	O3	FAL1	26/01/2015 06:00	26/01/2015 06:52	27/01/2015 06:00	27/01/2015 09:15
4	19.9	152991	O3	FAL1	02/02/2015 06:00	02/02/2015 12:30	03/02/2015 06:00	01/02/2015 08:02
5	28.4	175343	O3	FAL1	09/02/2015 06:00	09/02/2015 06:45	10/02/2015 06:00	10/02/2015 07:39
6	21.0	170269	O3	FAL1	16/02/2015 06:00	16/02/2015 07:30	17/02/2015 06:00	17/02/2015 07:13
7	20.0	165343	O3	FAL1	23/02/2015 06:00	23/02/2015 14:10	24/02/2015 06:00	24/02/2015 20:04
8	21.9	177991	O3	FAL1	02/03/2015 06:00	02/04/2015 07:50	03/03/2015 06:00	03/03/2015 11:50
9	19.5	140259	O3	FAL1	09/03/2015 06:00	09/03/2015 14:15	10/03/2015 06:00	10/03/2015 19:30
10	15.5		O3	FAL1	16/03/2015 06:00	18/03/2015 10:05	17/03/2015 06:00	19/03/2015 23:35

Discussion 3

Data Preprocessing

Data preprocessing is to make the data more suitable for data mining and machine learning, e.g.

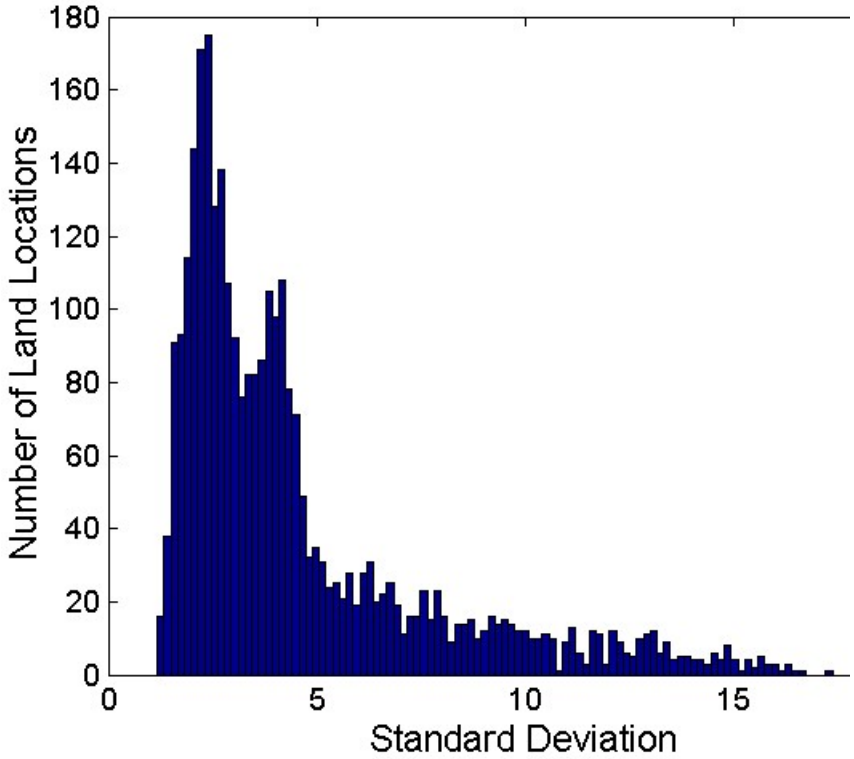
- (Data cleaning, i.e. deal with data quality issues)
- Aggregation
- Sampling
- Dimensionality Reduction
- Feature subset selection
- Feature creation
- Discretization and Binarization
- Attribute Transformation

Aggregation

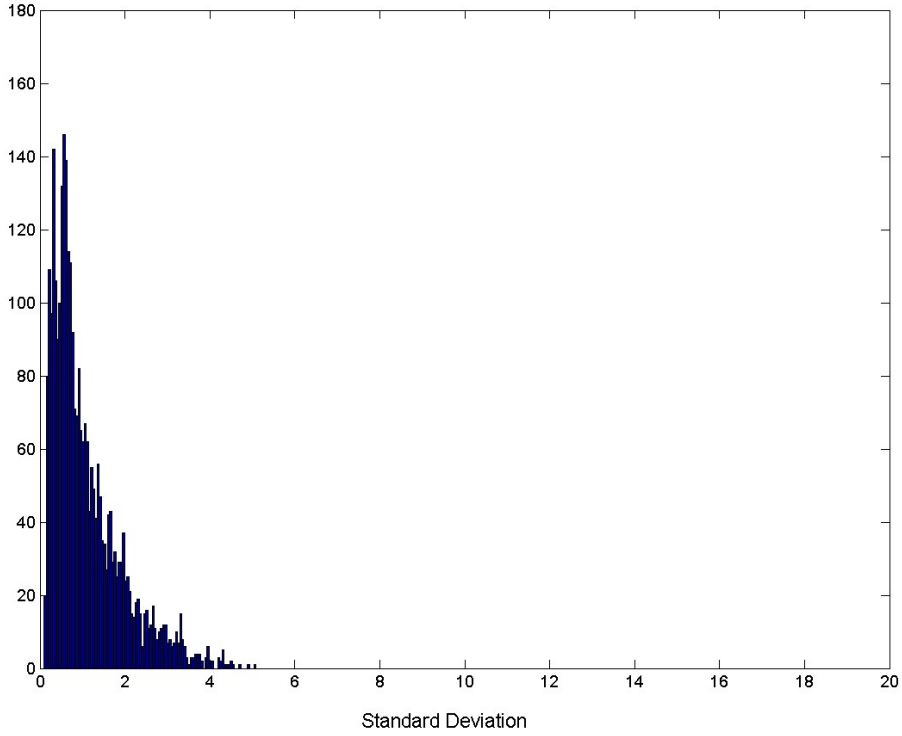
- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc.
  - More “stable” data
    - Aggregated data tends to have less variability

Example: Aggregation

Variation of Precipitation in Australia



Standard Deviation of Average Monthly Precipitation (rain)



Standard Deviation of Average Yearly Precipitation

Sampling

- **Sampling** is the process of understanding characteristics of data or models based on a subset of the original data. It is used extensively in all aspects of data exploration and mining
- Why sample?
  - Obtaining the entire set of "data of interest" is too expensive or time consuming
  - Obtaining the entire set of data may not be necessary (and hence a waste of resources)

A sample is representative if it has approximately the same property (of interest) as the original set of data

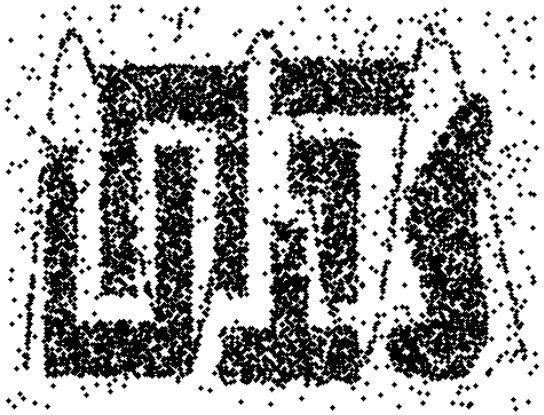
Sampling Approaches

- 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.
- **Stratified sampling**
  - Split the data into several partitions; then draw random samples from each partition

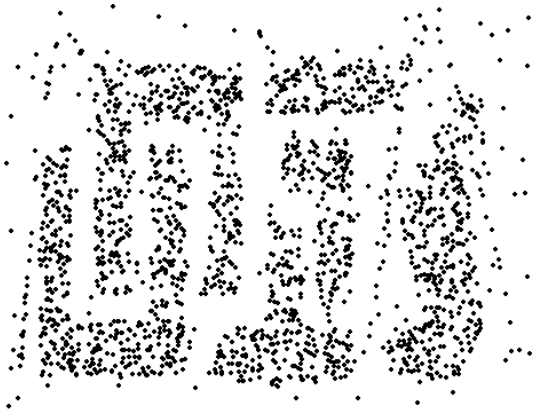
Stratified Sampling

- When subpopulations vary considerably, it is advantageous to sample each subpopulation (stratum) independently
- **Stratification** is the process of grouping members of the population into relatively homogeneous subgroups before sampling
- The strata should be mutually exclusive. The strata should also be collectively exhaustive
- Then **random sampling** is applied within each stratum. This often improves the representativeness of the sample by reducing sampling error

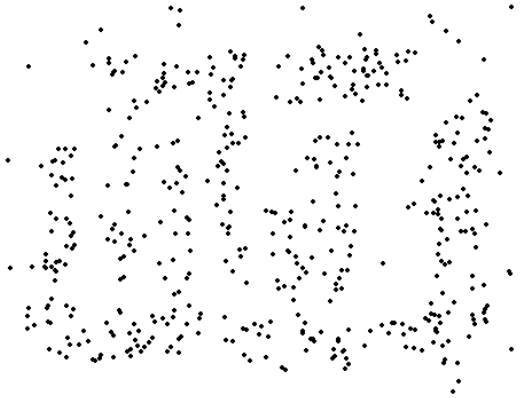
Loss of Structure with Sampling



8000 points



2000 Points

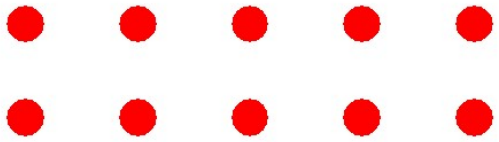


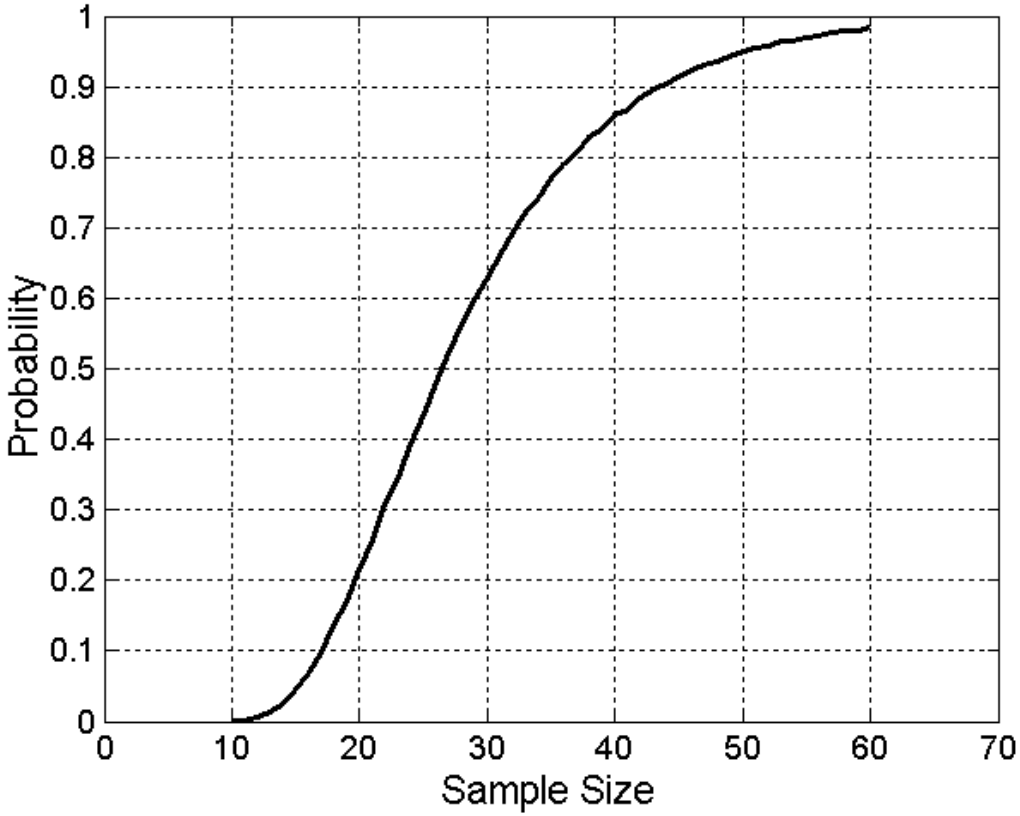
500 Points

Patterns may be missed in smaller sample sizes.
Appropriate sample size is important. How?

Sample Size

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





Curse of Dimensionality

- The **curse of dimensionality** refers to the phenomenon that many types of data analysis become significantly harder as the dimensionality of the data increases
  - Consider a set of documents, where each document is represented by a vector whose attributes are the frequencies with which each word occurs in the document. There are typically thousands of attributes
- Difficulty for **classification**: When dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Difficulty for **clustering**: Definitions of density and distance between points become less meaningful

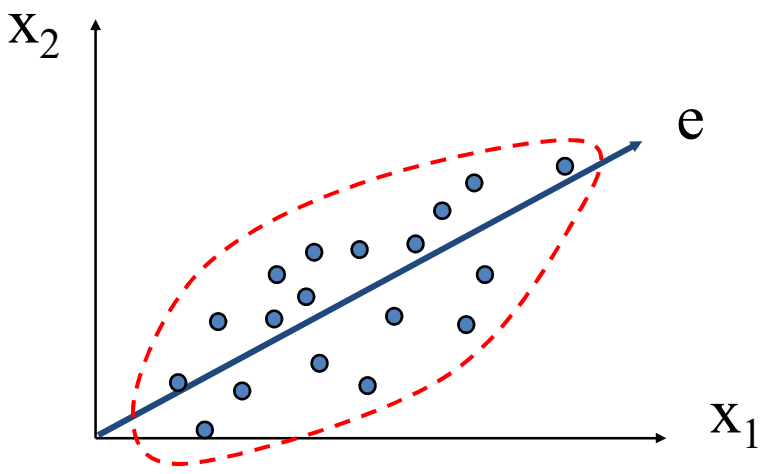
Dimensionality Reduction

- 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
  - Principal Component Analysis (PCA)
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

PCA: 1. are linear combinations of the original attributes; 2. are orthogonal to each other, 3. capture the maximum amount of variation in the data

Dimensionality Reduction: PCA

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

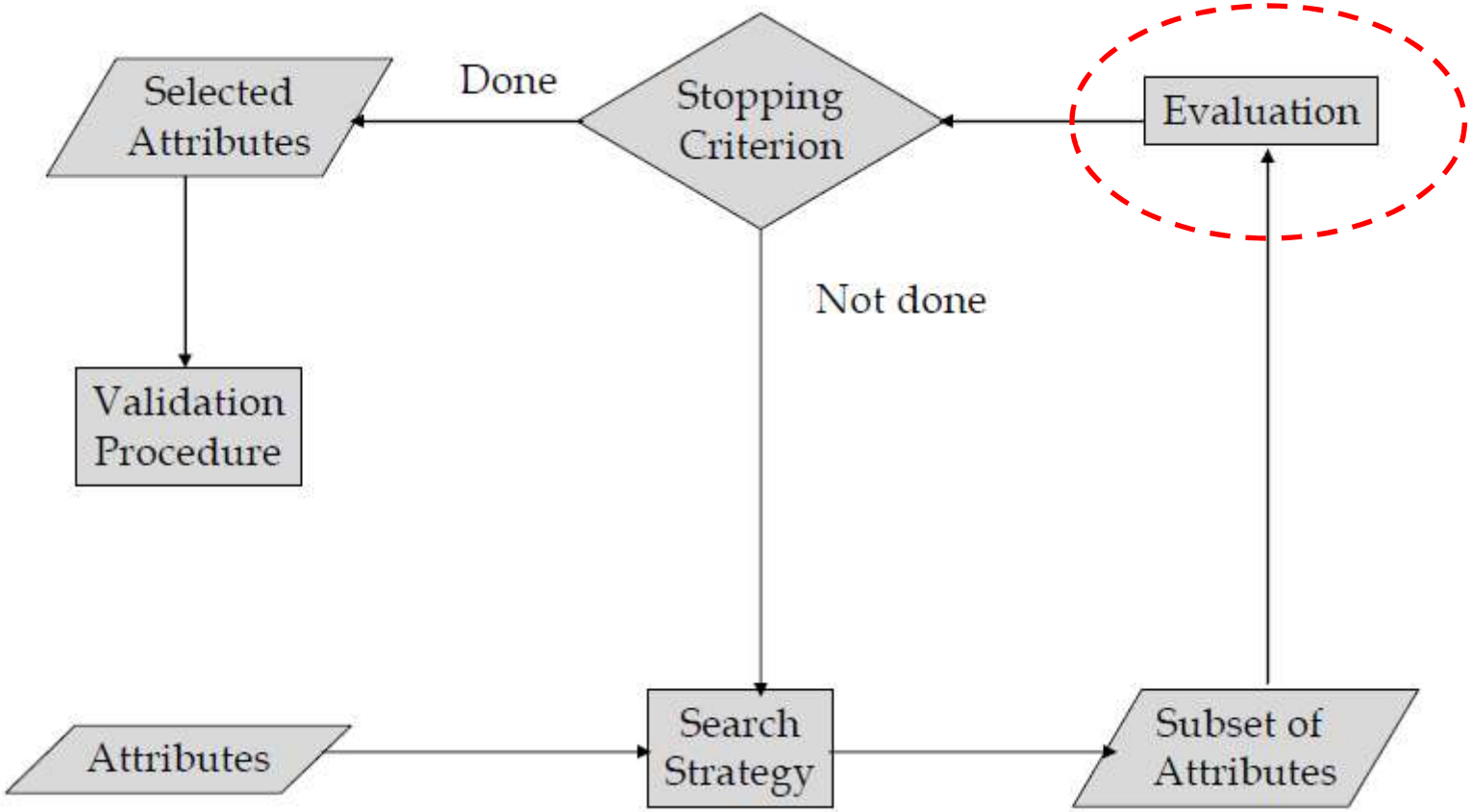
    Try all possible feature subsets as input to data mining algorithm
  - Embedded approaches

    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

Architecture for Feature Subset Selection: Filter & Wrapper



Feature Creation

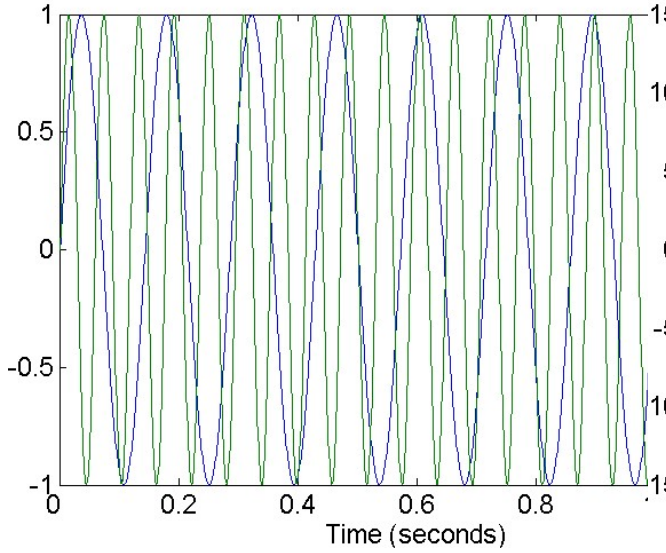
- Create **new attributes** that can capture the important information in a data set much more efficiently than the original attributes
- Three general methodologies:
  - **Feature Extraction**
    - domain-specific, e.g. photographs
  - **Mapping Data to New Space**
  - **Feature Construction**
    - combining features, e.g. density=mass/volume

Feature Creation: Feature Extraction

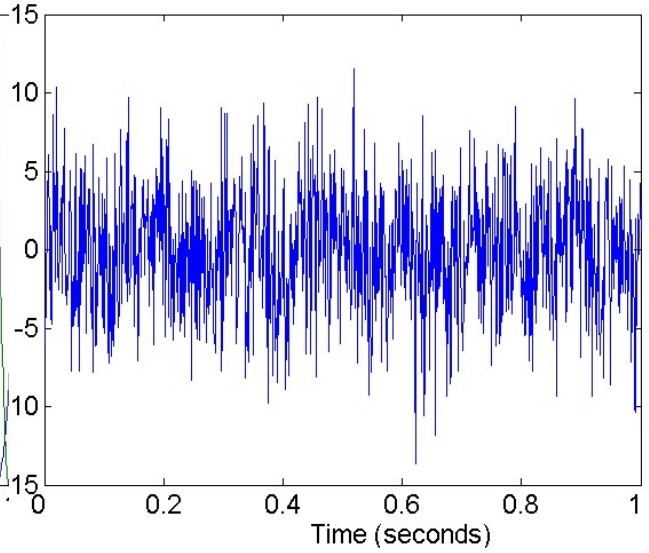
- One approach to dimensionality reduction is feature extraction, which is creation of a new, smaller set of features from the original set of features
- For example, consider a set of photographs, where each photograph is to be classified whether its human face or not
  - The raw data is set of pixels, and as such is not suitable for many classification algorithms
  - However, if data is processed to provide high-level features like presence or absence of **certain types of edges or areas** correlated with presence of human faces, then a broader set of classification techniques can be applied to the problem

Feature Creation: Mapping Data to a New Space

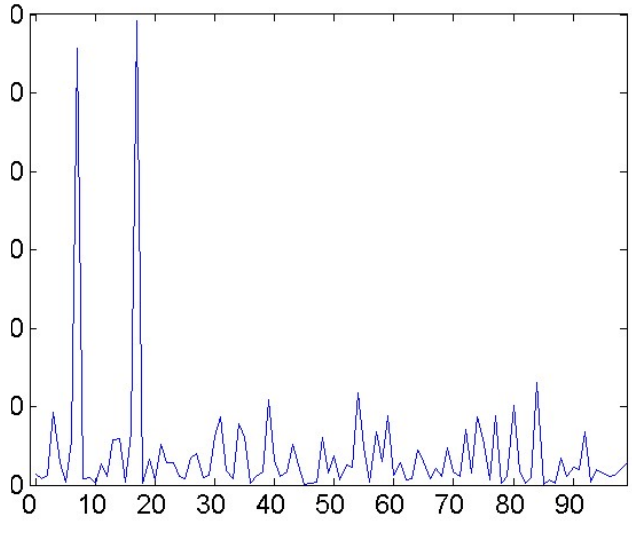
- Fourier transform
- Wavelet transform



Two Sine Waves



Two Sine Waves + Noise



Frequency

Feature Creation: Feature Construction

- Sometimes features have the necessary information, but not in the **form** necessary for the data mining algorithm. In this case, one or more new features constructed out of the original features may be useful
- Example, there are two attributes that record volume and mass of a set of objects
- Suppose there exists a classification model based on material of which the objects are constructed
- Then a **density feature** constructed from the original two features would help classification

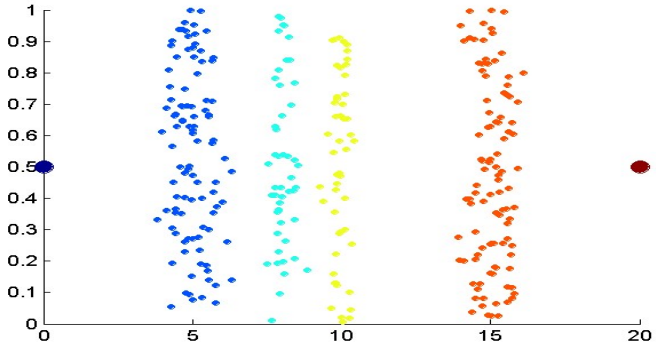
Example: Containership at Southampton– Feature selection, creation

ShipID	Speed	GT	Carrier	Route	planned arrival	Actual arrival	Planned depature	Actual departure
1	19.4	161635	O3	FAL1	12/01/2015 06:00	12/01/2015 13:18	13/01/2015 06:00	13/01/2015 14:15
2	20.3	166269	O3	FAL1	19/01/2015 06:00	19/01/2015 10:00	20/01/2015 06:00	20/01/2015 10:03
3	21.1	173022	O3	FAL1	26/01/2015 06:00	26/01/2015 06:52	27/01/2015 06:00	27/01/2015 09:15
4	19.9	152991	O3	FAL1	02/02/2015 06:00	02/02/2015 12:30	03/02/2015 06:00	01/02/2015 08:02
5	28.4	175343	O3	FAL1	09/02/2015 06:00	09/02/2015 06:45	10/02/2015 06:00	10/02/2015 07:39
6	21.0	170269	O3	FAL1	16/02/2015 06:00	16/02/2015 07:30	17/02/2015 06:00	17/02/2015 07:13
7	20.0	165343	O3	FAL1	23/02/2015 06:00	23/02/2015 14:10	24/02/2015 06:00	24/02/2015 20:04
8	21.9	177991	O3	FAL1	02/03/2015 06:00	02/04/2015 07:50	03/03/2015 06:00	03/03/2015 11:50
9	19.5	140259	O3	FAL1	09/03/2015 06:00	09/03/2015 14:15	10/03/2015 06:00	10/03/2015 19:30
10	15.5		O3	FAL1	16/03/2015 06:00	18/03/2015 10:05	17/03/2015 06:00	19/03/2015 23:35

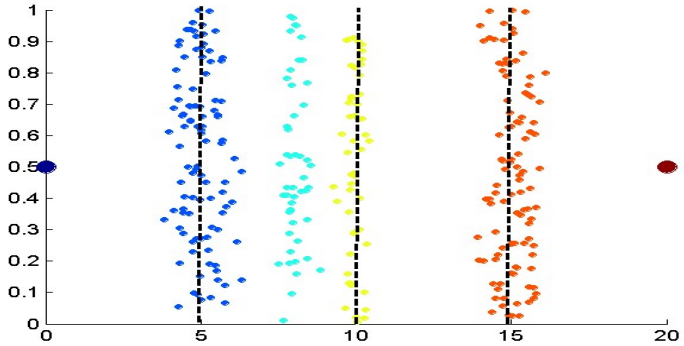
Discretization and Binarization

- Discretization is the process of **converting a continuous attribute to a discrete attribute**
- Why is it needed? -- Some algorithms require that the data be in the form of categorical or binary attributes.
- Categorical attributes → discrete or binary attributes
- Continuous attributes: a common example is rounding off real numbers to integers
- Unsupervised discretization and supervised discretization

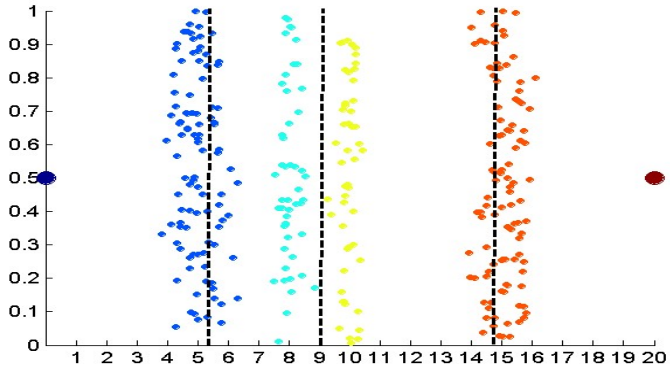
Discretization Without Using Class Labels



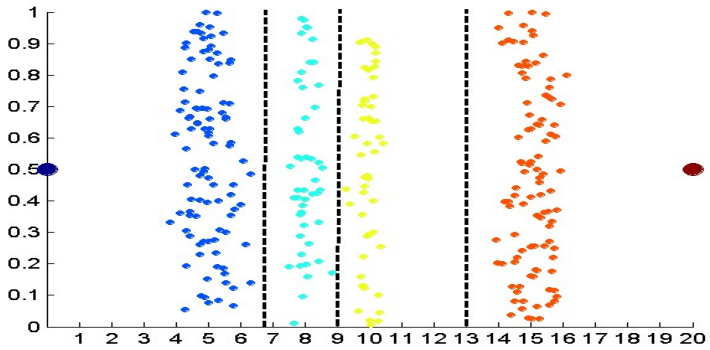
Data



Equal interval



Equal frequency

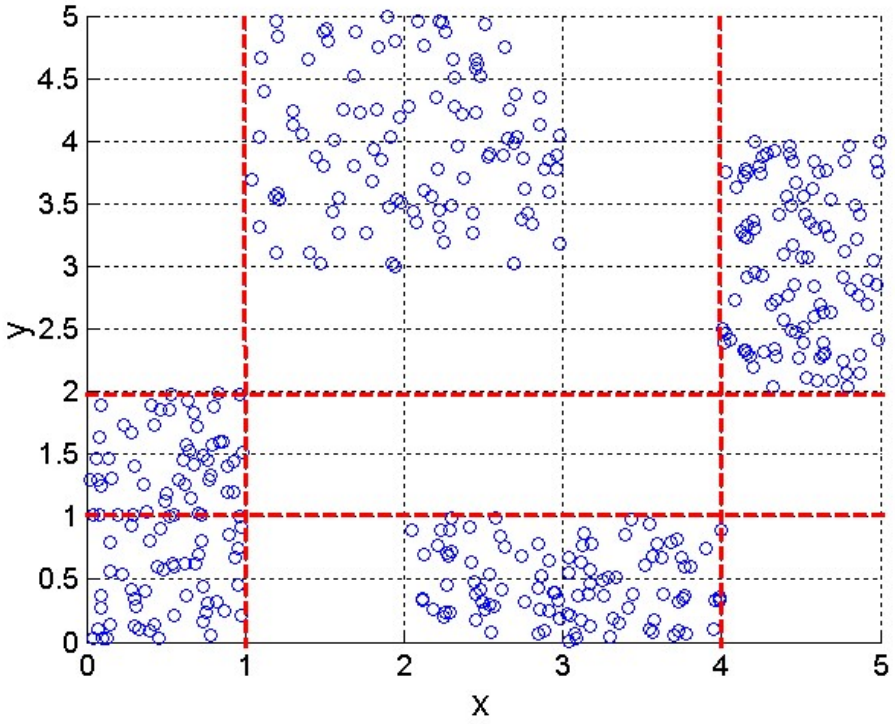


K-means

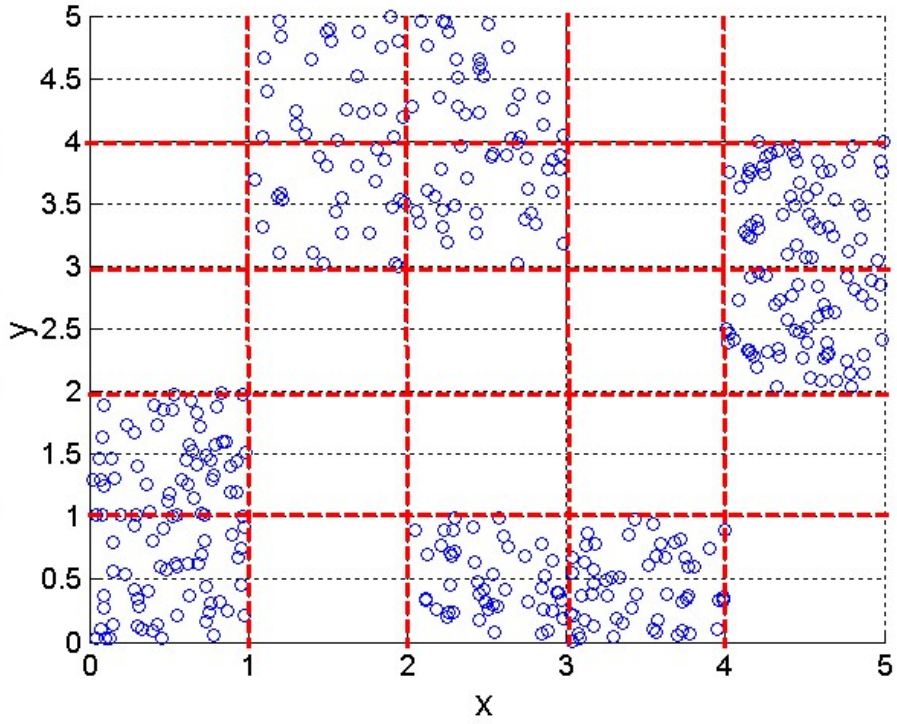
Unsupervised discretization does not use class info

Discretization Using Class Labels

- Entropy based approach



3 categories for both x and y

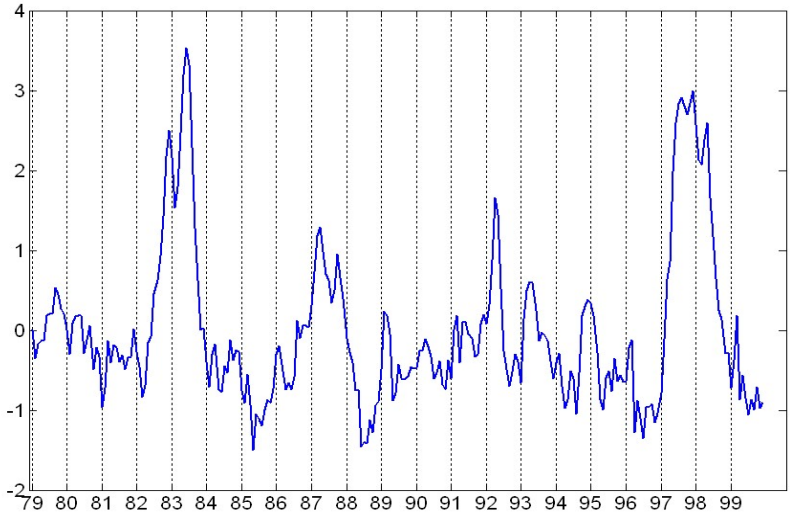


5 categories for both x and y

Supervised discretization uses additional info – same class in the same group AMSP

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



Example: Containership at Southampton-- Discretization

Speed	GT	Carrier	arrDelay	depDelay
19.4	161635	O3	0.30	0.34
20.3	166269	O3	0.17	0.17
21.1	173022	O3	0.04	0.14
19.9	152991	O3	0.27	0.08
21.0	170269	O3	0.06	0.05
20.0	165343	O3	0.34	0.59
21.9	177991	O3	0.08	0.24
19.5	140259	O3	0.34	0.56
15.2	131332	O3	2.08	1.74
21.1	173022	O3	0.05	0.18
21.4	175343	O3	0.05	-0.04
21.0	170269	O3	0.04	0.07
15.0	125688	O3	2.08	2.06

Converting continuous attributes **Speed, GT, arrDelay, depDelay** into discrete attributes.

Speed:
= IF(A2<15, "low", IF(A2<18, "medium","high"))
arrDelay:
= IF(D2<0.5), “No”, “Yes”)

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 similar
  - Minimum dissimilarity is often 0, upper limit varies
- **Proximity** refers to a similarity or dissimilarity
- **Correlation**

Similarity/Dissimilarity for Simple Attributes

Let *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 $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

Euclidean Distance

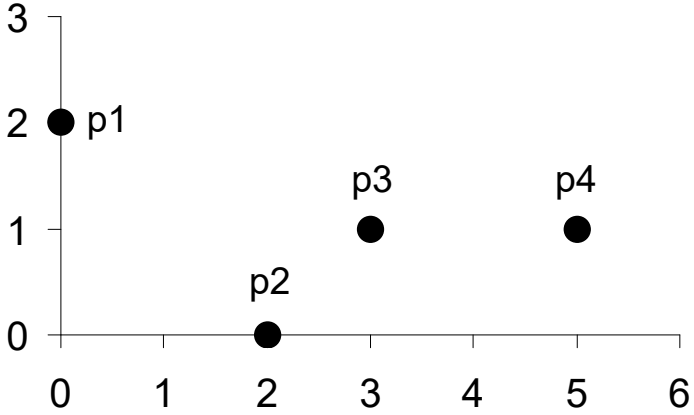
- **Euclidean Distance**

$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

Where n is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects p and q.

- Standardization is necessary, if scales differ.

Euclidean Distance



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

- **Minkowski Distance** is a generalization of Euclidean Distance

$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

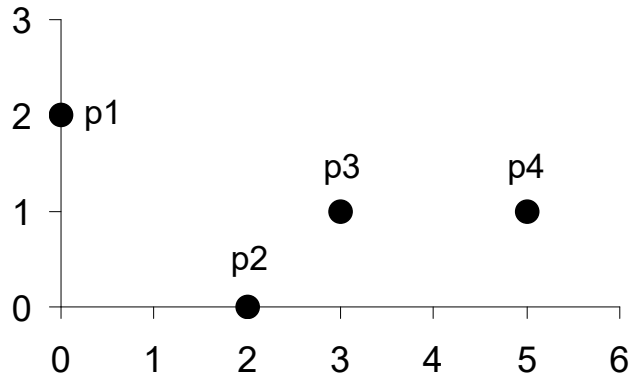
Where *r* is a parameter, *n* is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects *p* and *q*.

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.

Minkowski Distance: Examples

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



L1	p1	p2	p3	p4
p1				
p2				
p3				
p4				

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

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 $p = q$. (**Positivity**)
  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**)

  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 Coefficient (SMC) and Jaccard Coefficient

  SMC = number of matches / number of attributes

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

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

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

SMC versus Jaccard: Example

Two customer transactions:

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

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

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)

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

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

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

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

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

$\cos(d_1, d_2) = 5 / (\text{qsrt}(42)*\text{sqrt}(6)) = 0.3150$

Extended Jaccard Coefficient (Tanimoto)

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

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

Correlation

- **Correlation** measures statistical relationship (e.g. dependency or association) between objects
- **Pearson's correlation coefficient** between X and Y, is defined as

$$\text{Corr}(X,Y) = \frac{\text{covariance}(X,Y)}{\text{stdDev}(X)*\text{stdDev}(Y)} = \frac{s_{xy}}{s_x * s_y}$$

$$\rho_{X,Y} = \operatorname{corr}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y},$$

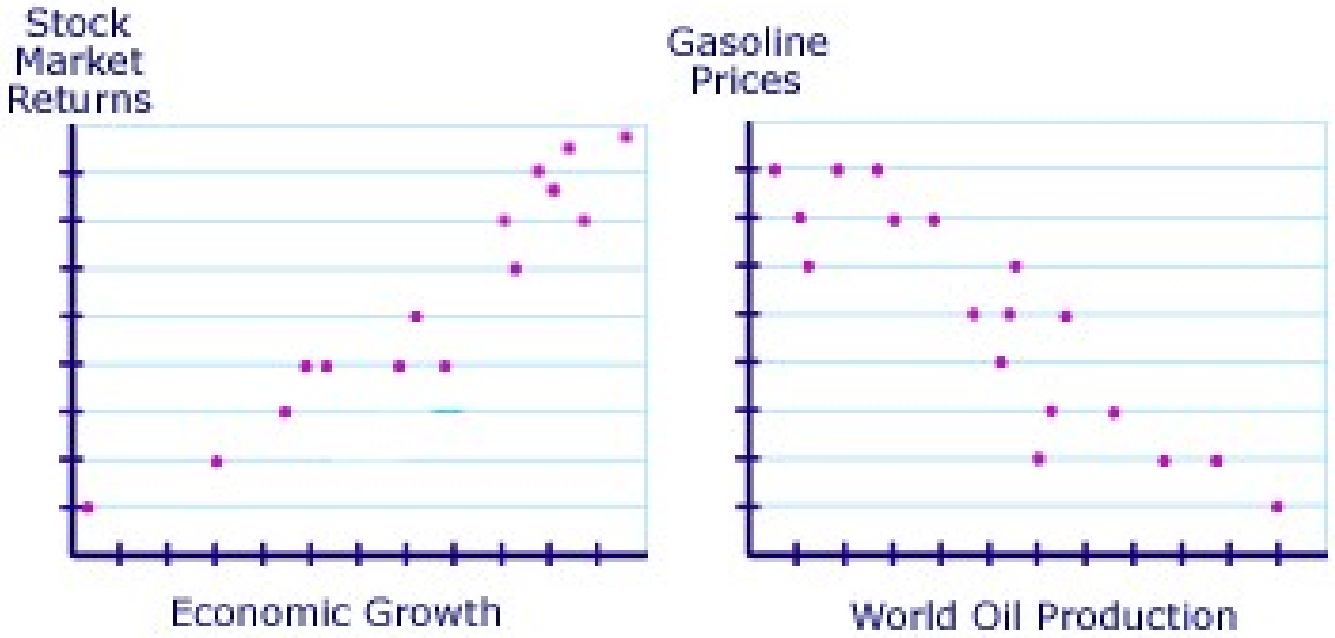
Sample Correlation Coefficient

$$\rho_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

or

$$\rho_{x,y} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{(n-1)s_x s_y} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}}.$$

Correlation: Examples



Covariance: Example

Economic Growth % (x_i)	S & P 500 Returns % (y_i)
2.1	8
2.5	12
4.0	14
3.6	10

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

$$\bar{x} = \frac{2.1 + 2.5 + 4.0 + 3.6}{4}$$

$$= \frac{12.2}{4}$$

$$= 3.1$$

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

$$\bar{y} = \frac{8 + 12 + 14 + 10}{4}$$

$$= \frac{44}{4}$$

$$= 11$$

$$COV(x, y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

$$= \frac{(2.1 - 3.1)(8 - 11) + \ldots}{4\text{-}1}$$

$$= \frac{(-1)(-3) + (-0.6)(1) + (0.9)(3) + \ldots}{3}$$

$$= \frac{3 + (-0.6) + 2.7 + (-0.5)}{3}$$

$$= \frac{4.6}{3}$$

$$= 1.53$$

Correlation: Example

$$s_x = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$$

$$s_x = \sqrt{\frac{(2.1 - 3.1)^2 + (2.5 - 3.1)^2 + \ldots}{4\text{-}1}}$$

$$= \sqrt{\frac{(-1)^2 + (-0.6)^2 + (0.9)^2 + (0.5)^2}{3}}$$

$$= \sqrt{\frac{1 + 0.36 + 0.81 + 0.25}{3}} = 0.90$$

$$s_y = \sqrt{\frac{(8 - 11)^2 + (12 - 11)^2 + \ldots}{4\text{-}1}}$$

$$= \sqrt{\frac{(-3)^2 + (1)^2 + (3)^2 + (-1)^2}{3}}$$

$$= \sqrt{\frac{9 + 1 + 9 + 1}{3}} = 2.58$$

Economic Growth % (x_i)	S & P 500 Returns % (y_i)
2.1	8
2.5	12
4.0	14
3.6	10

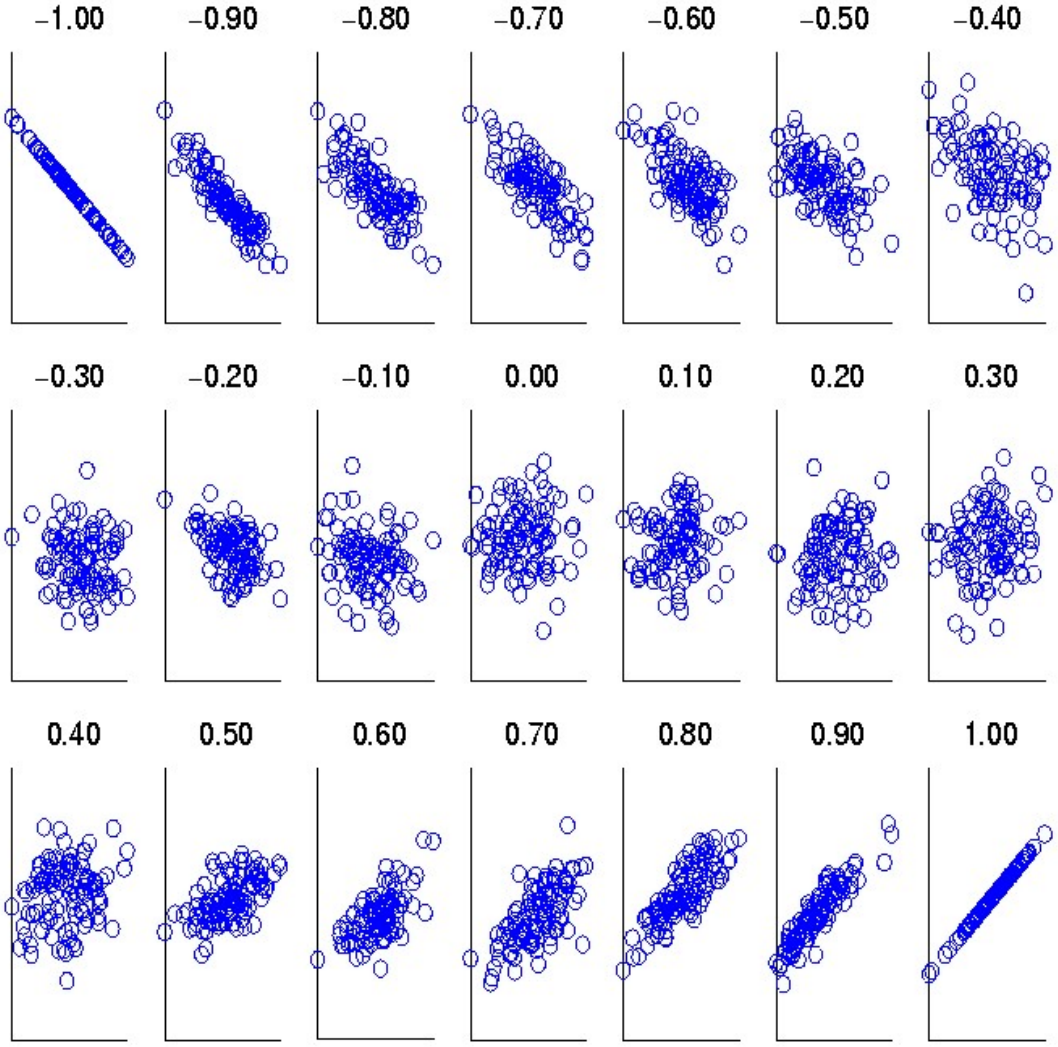
$COV(x,y) = 1.53$
$s_x = 0.90$
$s_y = 2.58$

$$\rho_{x,y} = \frac{COV(x,y)}{s_x s_y}$$

$$= \frac{1.53}{(.90)(2.58)}$$

$$= .66$$

Visually Evaluating Correlation



X and Y have 30 attributes randomly generated with similarity ranging from –1 to 1.

General Approach for Combining Similarities

- Sometimes attributes are of many **different types**, but an overall similarity is needed.
- Similarity algorithm of heterogeneous objects:

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.

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

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

Summary

- Data Set, Object, Attribute, Attribute Value
- Types of Data Set
- Data Quality & Techniques
- Data Pre-processing Techniques
- Similarity and Dissimilarity