CS 584
Data Mining

Data

2/2/16

What is Data ?

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**

What is Data?

- Information that can be easily processed.
- 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

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
  - Different attributes can be mapped to the same set of values

Types of Attributes

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

Properties of Attribute Values

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

Nominal attribute: ?
Ordinal attribute: ?
Interval attribute: ?
Ratio attribute: ?

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. ($=$, $\neq$)	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
  - 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 using integer variables.
  - Note: binary attributes are a special case of discrete attributes

- Continuous (numeric) 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.

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

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

How would you represent

- Document 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.

	team	coach	pla y	ball	score	game	wi n	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

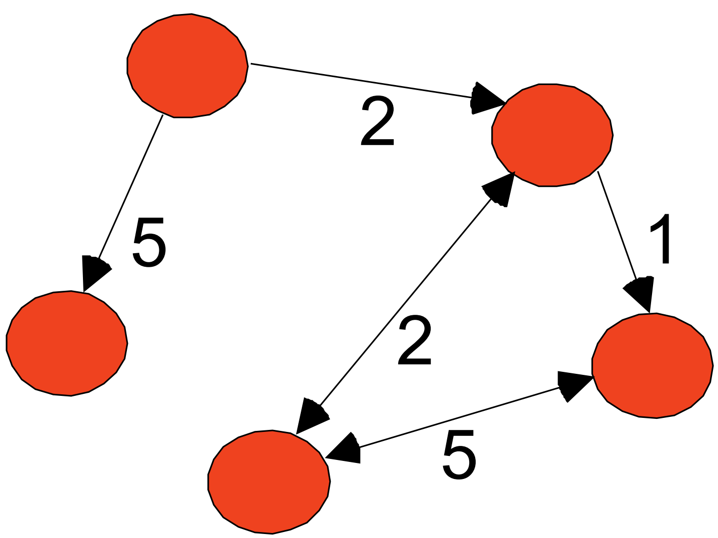
Transaction Data

- 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

Graph Data

- Examples: Generic graph and HTML Links



Data Mining

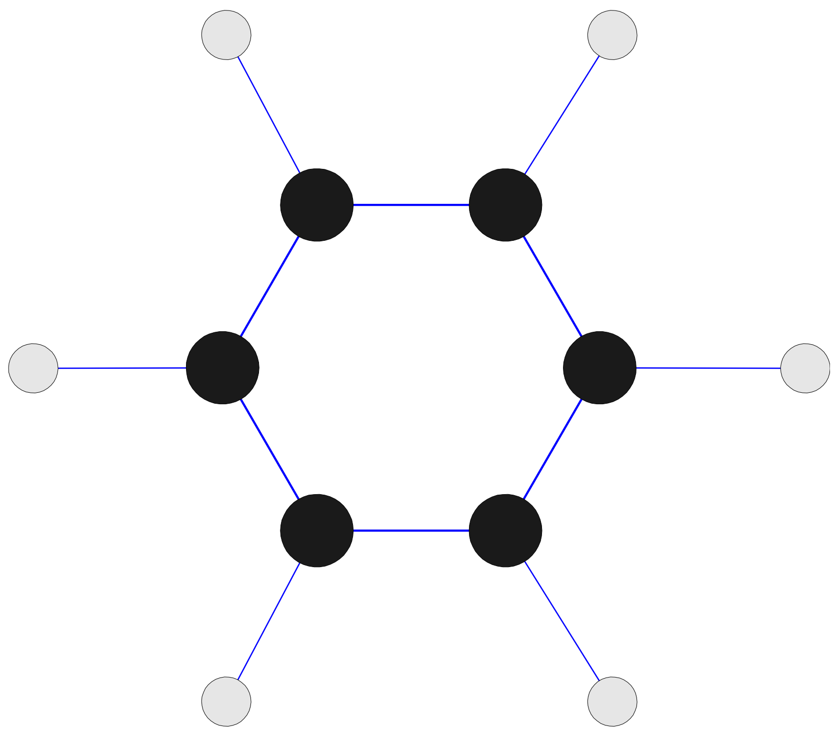
Graph Partitioning

Parallel Solution of Sparse Linear System of Equations

<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

Items/Events

(A B) (D) (C E)

(B D) (C) (E)

(C D) (B) (A E)

An element of the sequence

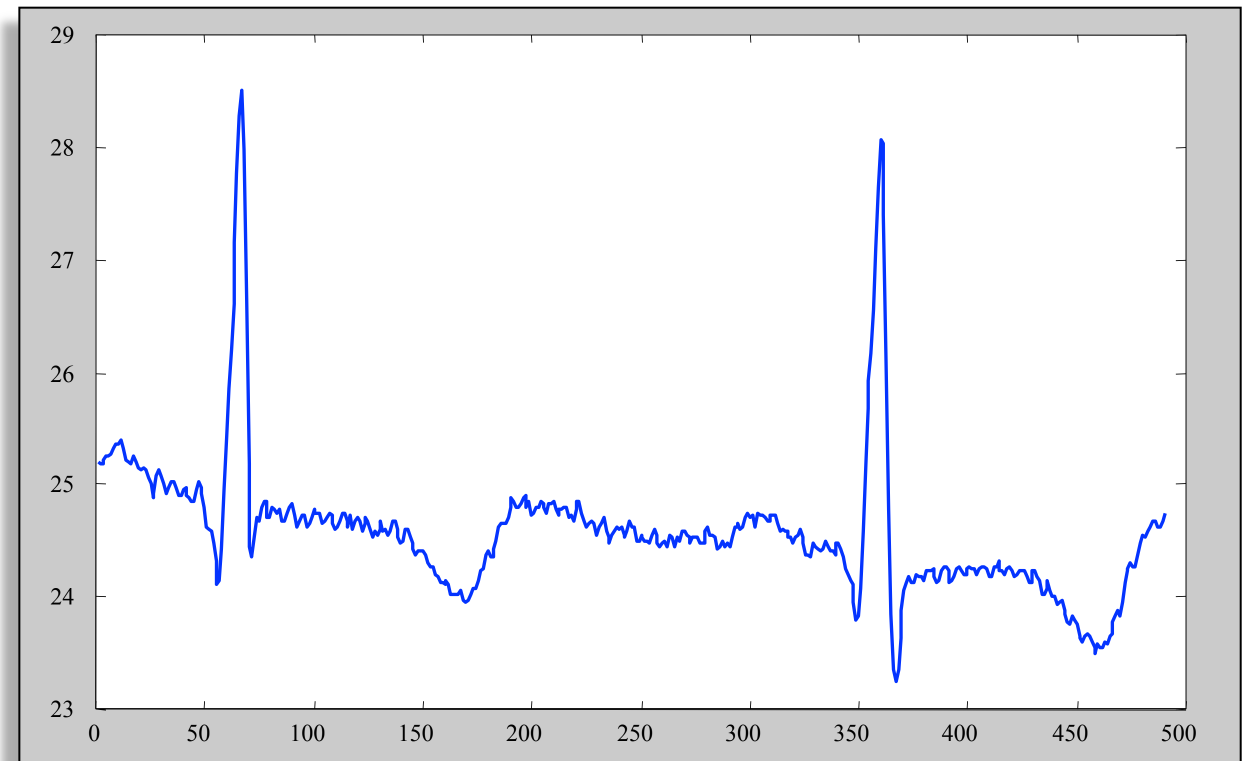
Ordered Data

- Genomic sequence data

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

Ordered Data

- Time Series



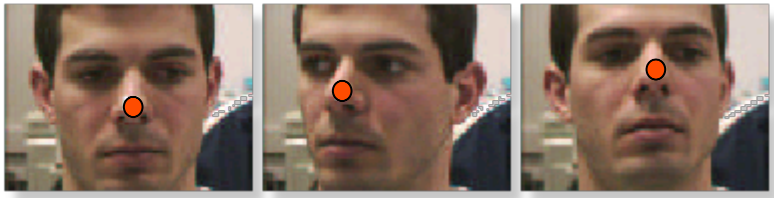
25.1750
25.2250
25.2500
25.2500
25.2750
25.3250
25.3500
25.3500
25.4000
25.4000
25.3250
25.2250
25.2000
25.1750

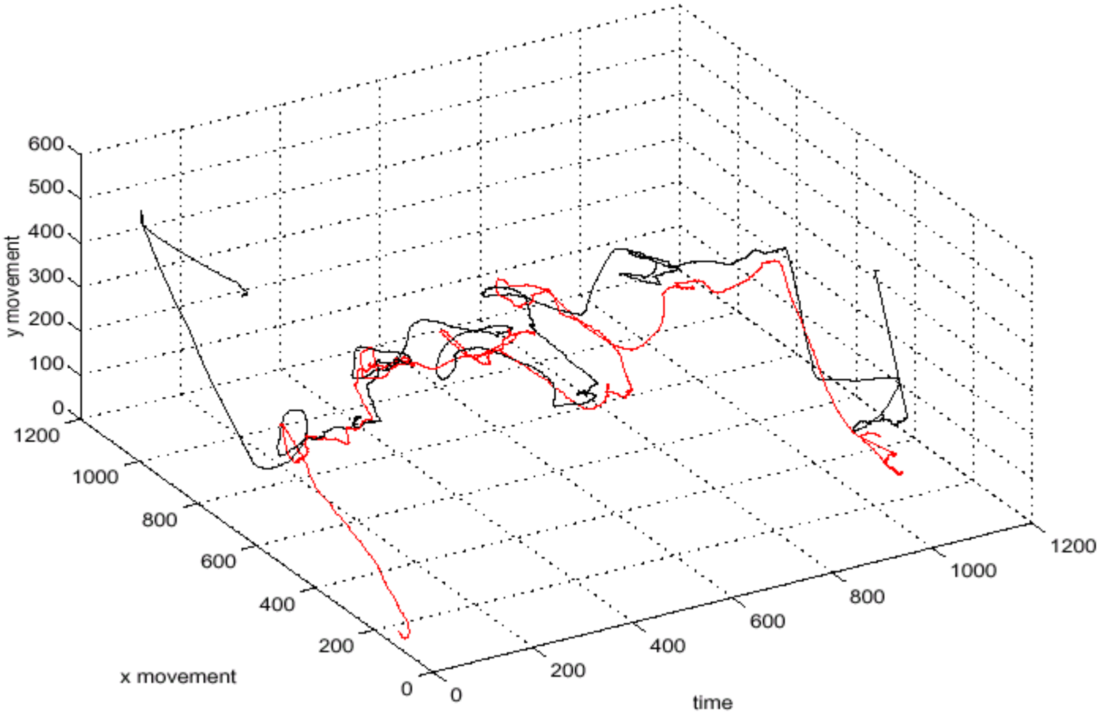
..

..
24.6250
24.6750
24.6750
24.6250
24.6250
24.6250
24.6750
24.7500

Many Time Series Contain Spatial Information (Trajectories)

- Video tracking / Surveillance
  - Visual tracking of body features (2D time-series)
  - Sign Language recognition (3D time-series)
- GPS tracking
- Hurricane tracks

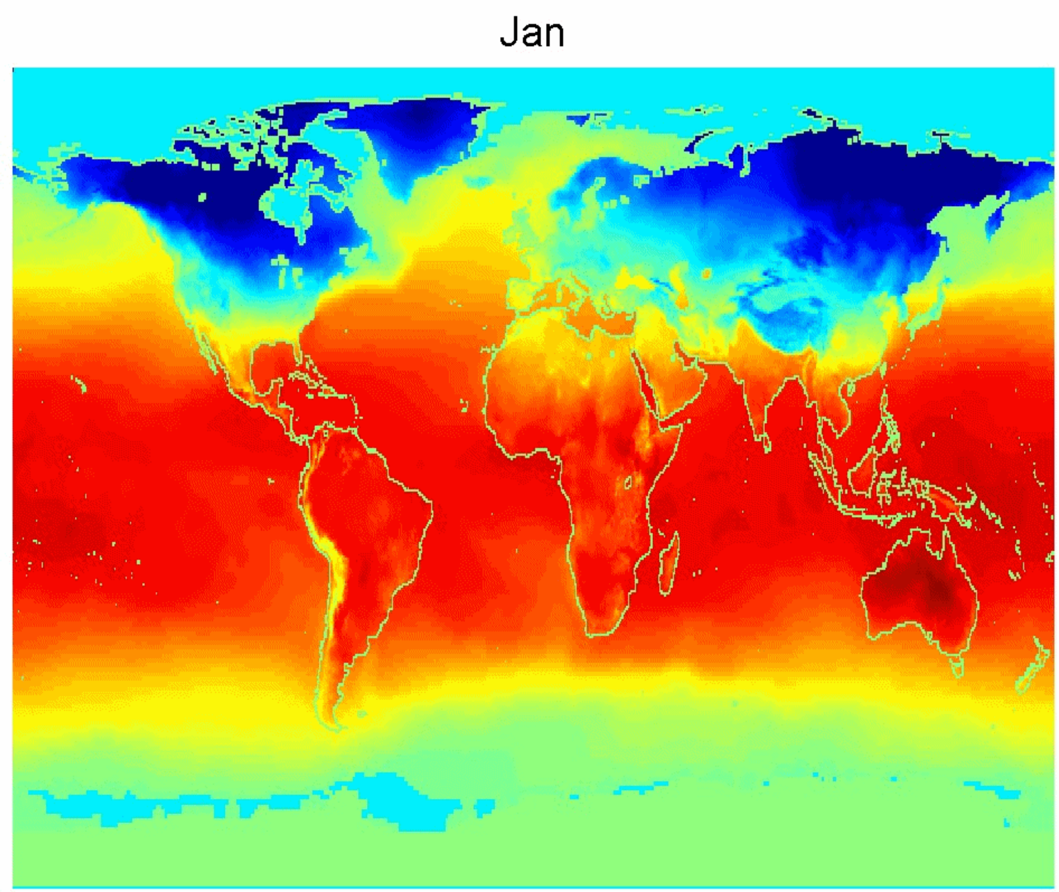




Courtesy of Dr. Michail Vlachos

Ordered Data

- Spatio-Temporal Data



Average Monthly Temperature of land and ocean

Image Data

- Can be represented as (color) histograms
- Frequency count of each individual color
- Most commonly used color feature representation

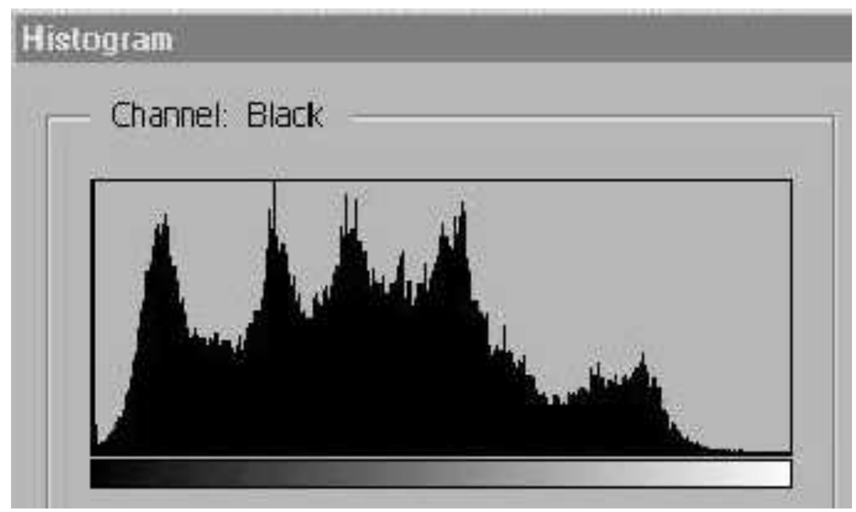


Image **Corresponding histogram**

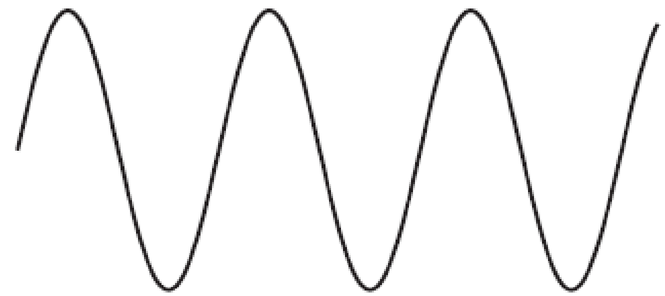
Data Quality

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

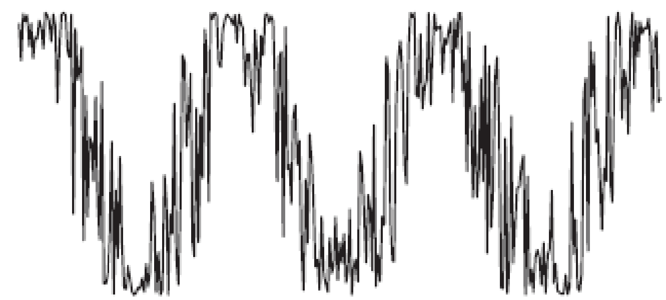
- Examples of data quality problems:
  – Noise and outliers
  – missing values
  – duplicate data

Noise

- Noise refers to modification of original values
  - Random collection of error.
  - Examples: distortion of a person’s voice when talking on a poor phone and “snow” on television screen



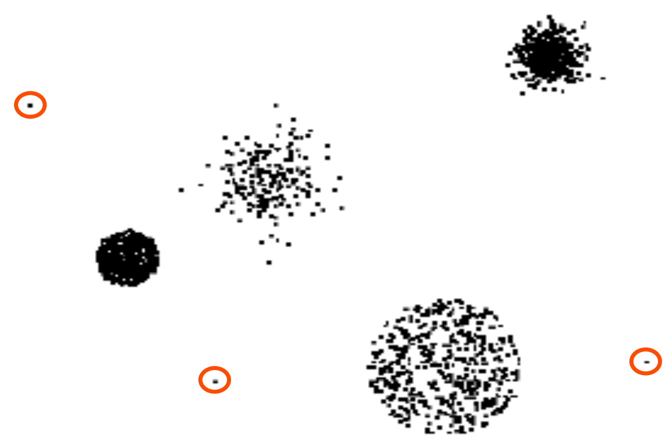
(a) Time series.



(b) Time series with noise.

Outliers

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



Missing Values (Think)

- Reasons for missing values?
- Handling missing values (How? Think)

Missing Values (Think)

- 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)

- Handling missing values (How? Think)
  - Eliminate Data Objects
  - Estimate Missing Values
  - Ignore the Missing Value During Analysis

Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeneous sources

- Examples:
  - Same person with multiple email addresses

- Data cleaning
  - Process of dealing with duplicate data issues

Data Preprocessing

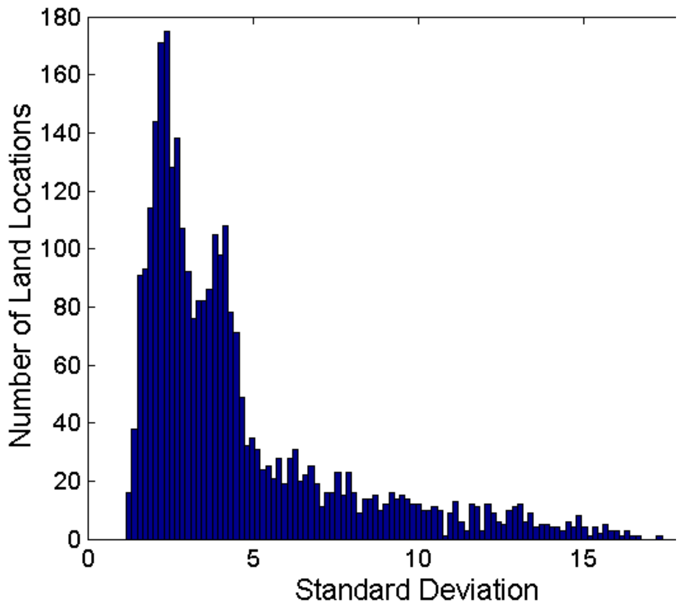
- Aggregation
- Sampling
- Dimensionality Reduction
- Feature subset selection
- Feature creation
- Discretization and Binarization
- Attribute Transformation

Aggregation (LESS IS MORE)

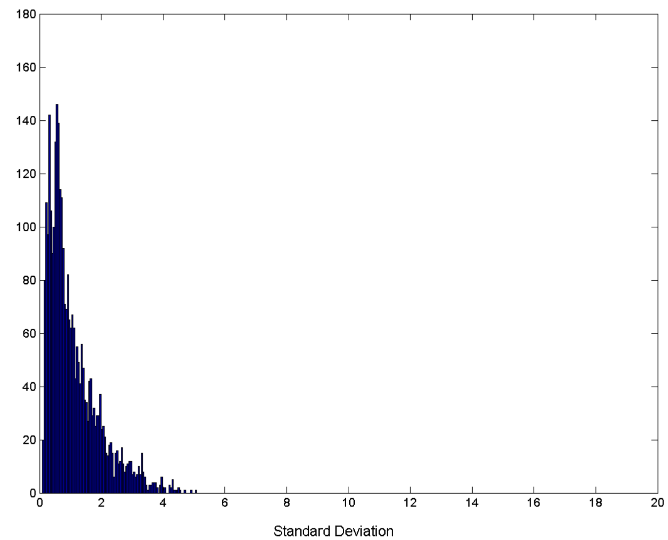
- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More "stable" data
    - Aggregated data tends to have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

Aggregation

Variation of Precipitation in Australia



Standard Deviation of Average Monthly Precipitation



Standard Deviation of Average Yearly Precipitation

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.

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

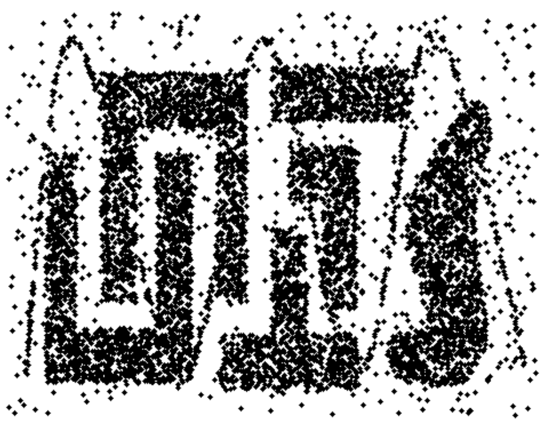
Sampling …

- 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

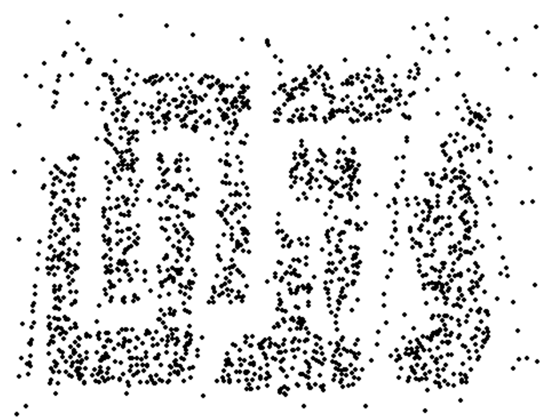
Types of Sampling

- 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

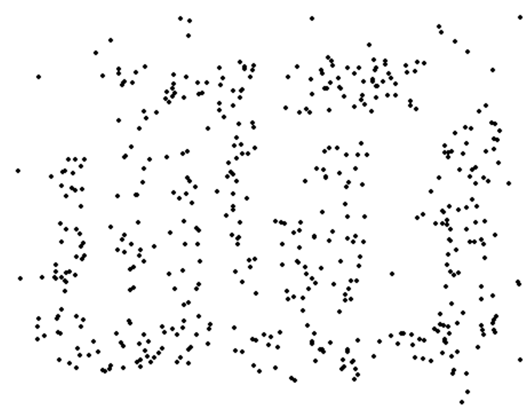
Sample Size



8000 points



2000 Points



500 Points

Curse of Dimensionality

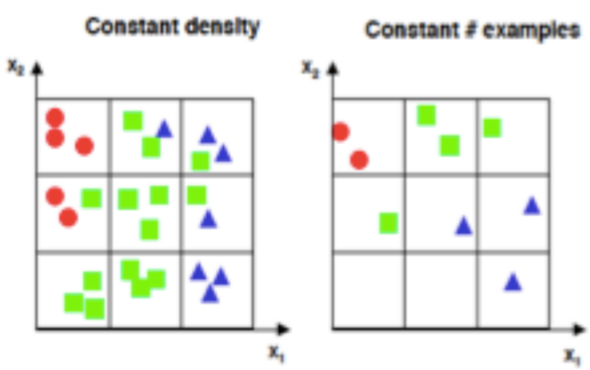
- When dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Also distances between objects gets skewed
  - More dimensions that contribute to the notion of distance or proximity which makes it uniform. This leads to trouble in clustering and classification settings.

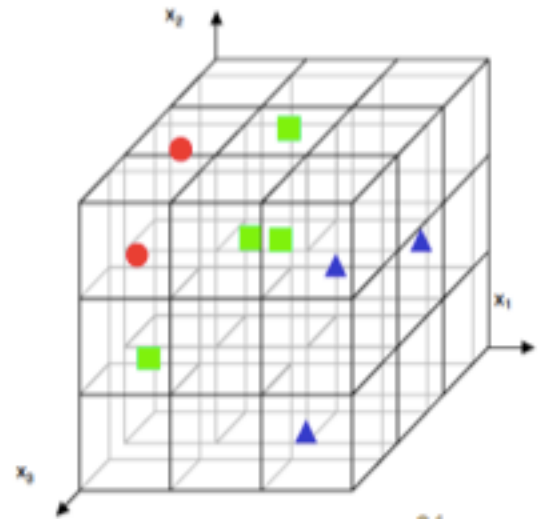
Driving the point ..

- Consider a 3-class classification problem.
- In our toy problem, we decide to start with one feature and divide the real line into 3 segments.

- After we have done this, we notice that there exist too much overlap between classes. So we add another feature.

- We decide to preserve the granularity of each axis, so the # of bins goes from 3 (in 1D) to $3^2 = 9$ (in 2D).
  - At this point we are faced with a decision: do we maintain the density of each cell, or do we keep the same number of examples as in 1D?

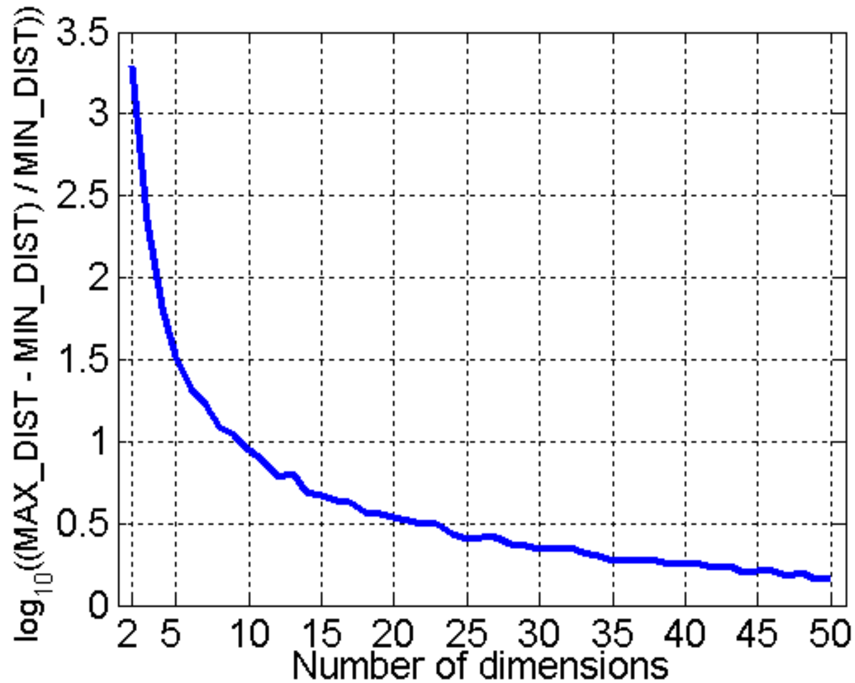




  - Moving to 3 features makes the problem worse.
    - The # of bins becomes $3^3 = 27$ (in 3D).
    - For the same density, the number of examples becomes…?
    - For the same number of examples, the 3D scatter plot looks almost empty.

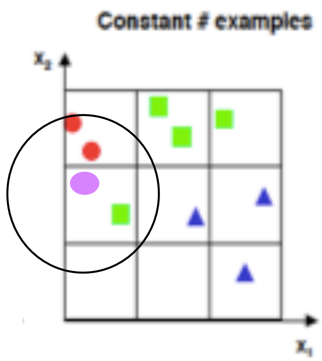
Curse of Dimensionality

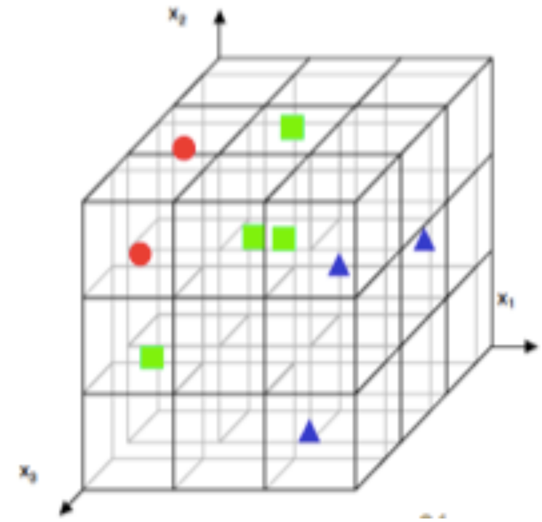
- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful



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

- Curse of dimensionality in indexing.

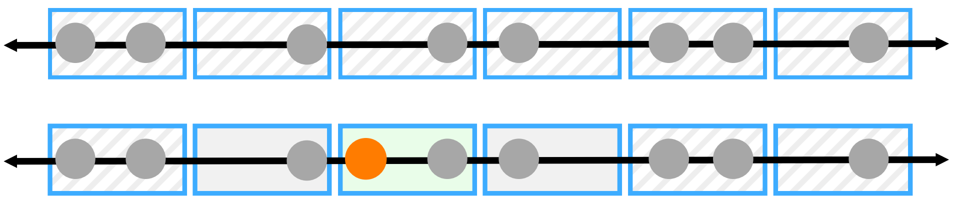




- Recall that if we decide to preserve the granularity of each axis, the # of bins goes from 3 (in 1D) to $3^2 = 9$ (in 2D) to $3^3 = 27$ (in 3D).
- We can treat this multi-dimensional grid as an index structure. Now suppose that, given a query point (the purple circle in the center cell), we want to find the closest point to the query.
- Obviously, we want to check the cell that the point resides in. The closest point may be in a neighboring cell, so we have to check those too.

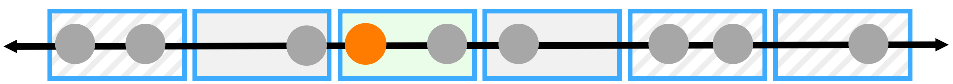
Simplified example to illustrate curse of dimensionality:
How many additional (nonempty) cells must we examine before we are guaranteed to find the best match?

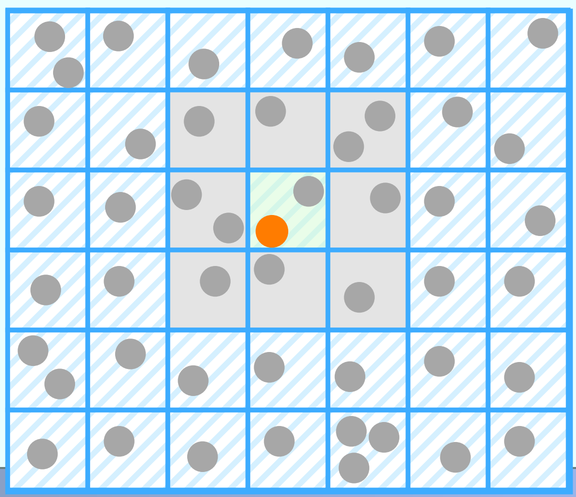
For the one dimensional case, the answer is clearly 2...



If we project a **query** into n-dimensional space, how many additional (nonempty) cells must we examine before we are guaranteed to find the best match?

For the one dimensional case, the answer is clearly 2...

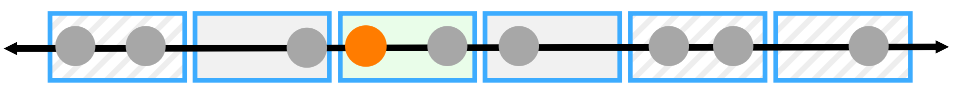


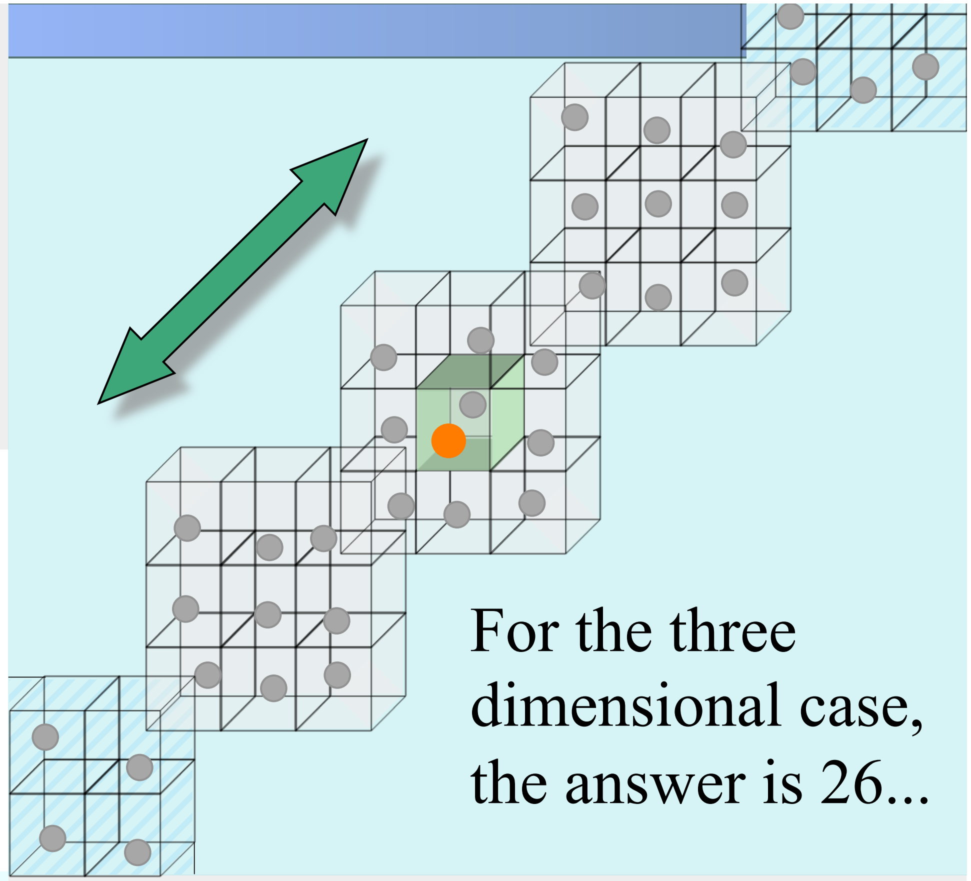


For the two dimensional case, the answer is 8...

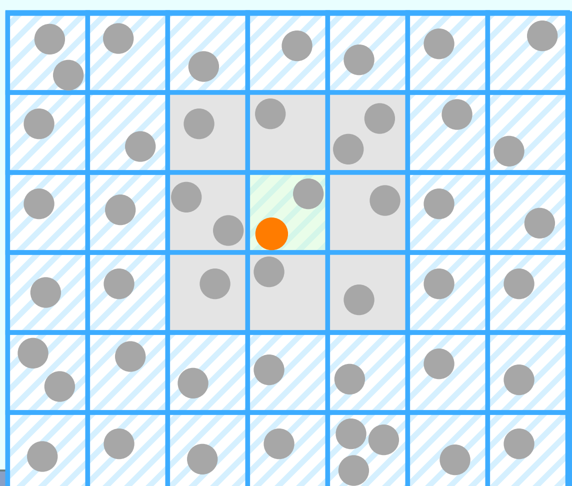
If we project a **query** into n-dimensional space, how many additional (nonempty) cells must we examine before we are guaranteed to find the best match?

For the one dimensional case, the answer is clearly 2...





For the three dimensional case, the answer is 26...



For the two dimensional case, the answer is 8...

More generally, in n-dimension space we must examine $3^n - 1$ cells

$n = 21 \Rightarrow 10{,}460{,}353{,}201$ cells

*The cells are also known as “MBR” (minimum bounding rectangles) as in R-trees.

Dimensionality Reduction

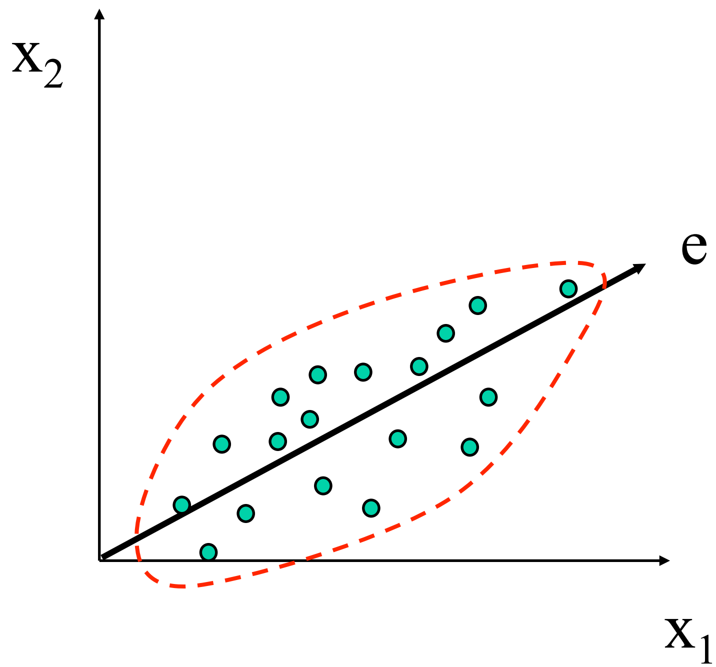
- Purpose:
  - Avoid curse of dimensionality
  - Reduce amount of time and memory required by data mining algorithms
  - Allow data to be more easily visualized
  - May help to eliminate irrelevant features or reduce noise
- Techniques
  - Principle Component Analysis
  - Singular Value Decomposition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

Principal Component Analysis

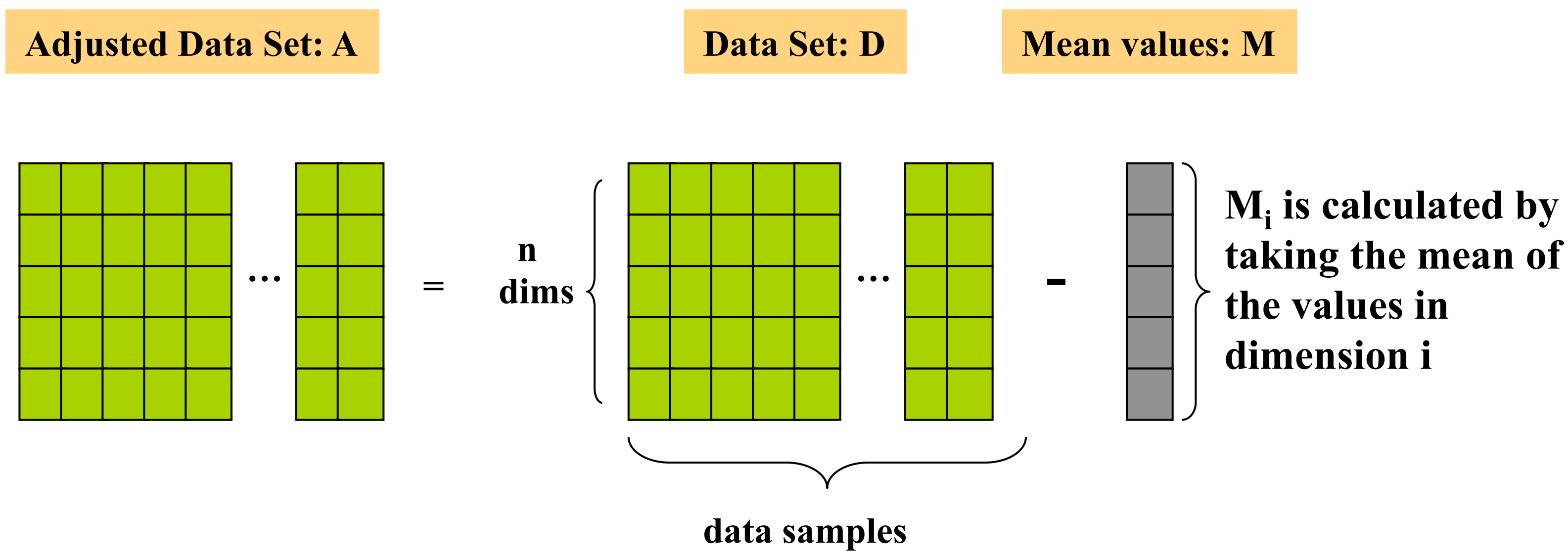
- Goal of PCA
  - To reduce the number of dimensions.
  - Transfer interdependent variables into single and independent components.
- What does PCA do ?
  - Transforms the data into a lower dimensional space, by constructing dimensions that are linear combinations of the input dimensions/ features.
  - Find independent dimensions along which data have the largest variance.

Dimensionality Reduction: PCA

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



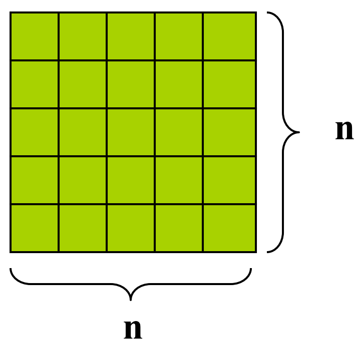
PCA: #1 Calculate Adjusted Data Set



Adapted from Musawir Ali, UCF

PCA: #2 Calculate Co-variance matrix, C, from Adjusted Data Set, A

Co-variance Matrix: C



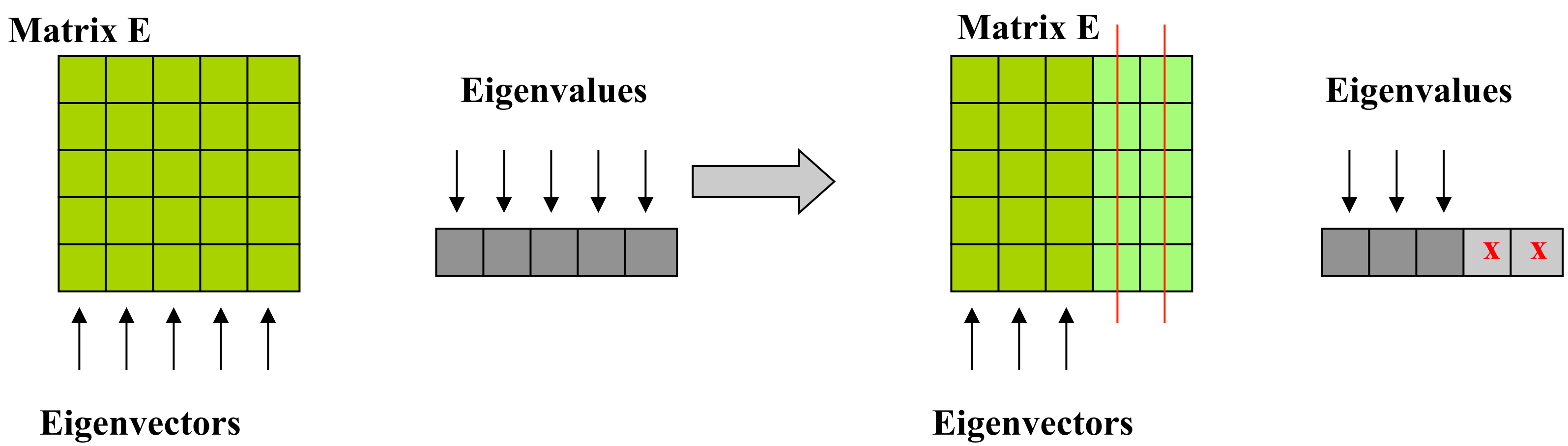
$C_{ij} = cov(i,j)$

Note: Since the means of the dimensions in the adjusted data set, A, are 0, the covariance matrix can simply be written as:

$$C = (A A^T) / (n-1)$$

Adapted from Musawir Ali, UCF

PCA: #3 Calculate eigenvectors and eigenvalues of C



If some eigenvalues are 0 or very small, we can essentially discard those eigenvalues and the corresponding eigenvectors, hence reducing the dimensionality of the new basis.

Adapted from Musawir Ali, UCF

PCA: #4 Transforming data set to the new basis

$F = E^T A$

where:

• F is the transformed data set

• E^T is the transpose of the E matrix containing the eigenvectors

• A is the adjusted data set

Note that the dimensions of the new dataset, F, are less than the data set A

To recover A from F:

$(E^T)^{-1} F = (E^T)^{-1} E^T A$

$(E^T)^T F = A$

$EF = A$

*** E is orthogonal, therefore $E^{-1} = E^T$**

Adapted from Musawir Ali, UCF

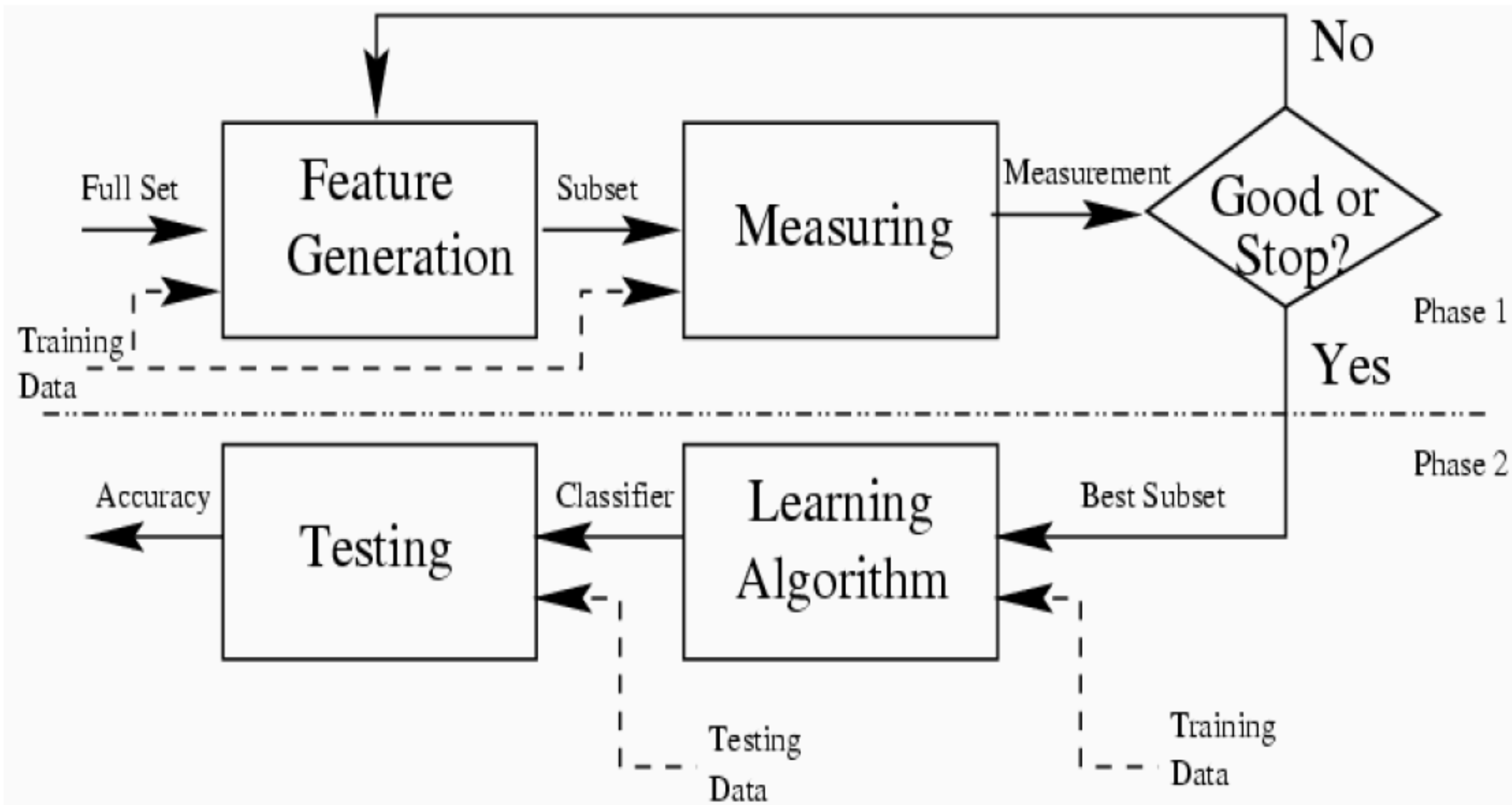
Feature Subset Selection

- 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
  - Feature Weighting

Filter Approach



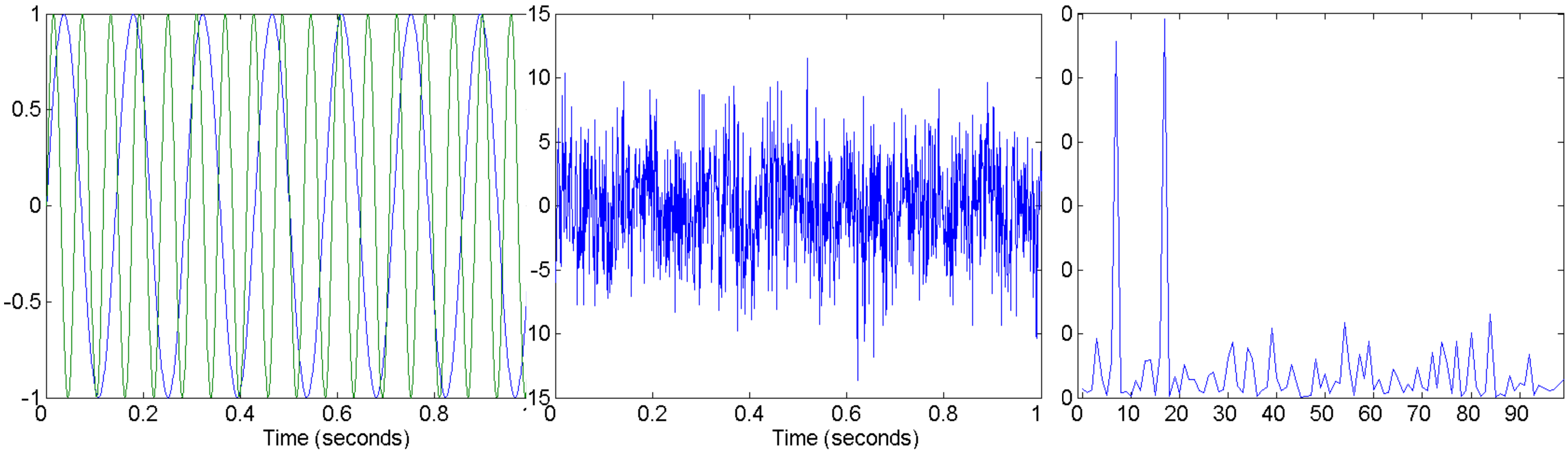
Feature Creation

- 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

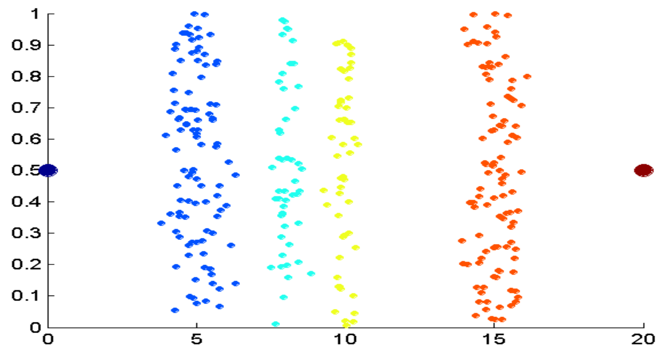
- Fourier transform
- Wavelet transform



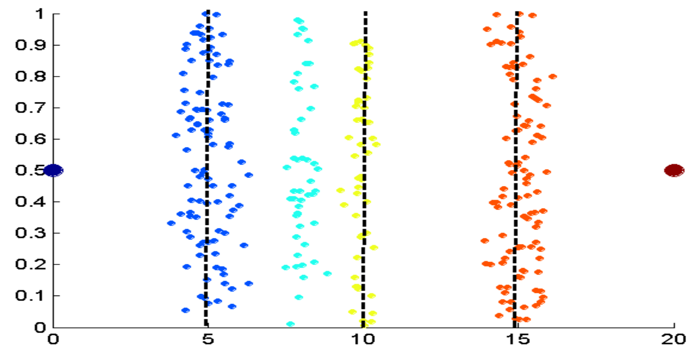
Two Sine Waves **Two Sine Waves + Noise** **Frequency**

Discretization

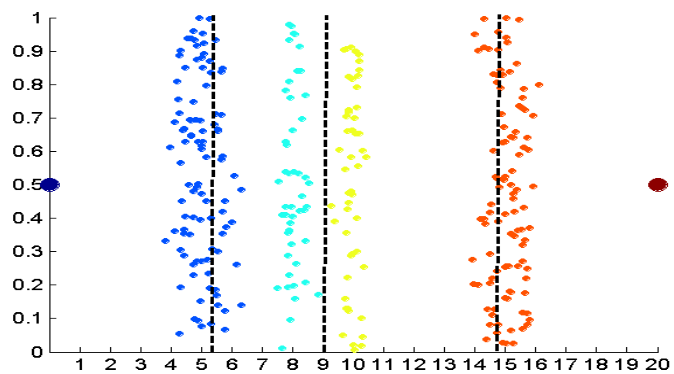
Without using class labels (unsupervised)



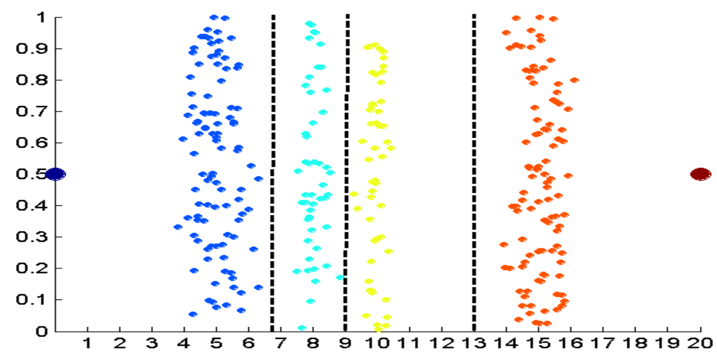
Data



Equal interval width



Equal frequency



K-means

Discretization Using Class Labels

- Entropy based approach:
  - If you have class labels, compute the entropy per discretized bin, and then try to minimize the same.
  - The entropy e_i for the i^{th} bin is given by (k = # of classes):

$$e_i = \sum_{j=1}^{k} p_{ij} \log_2 p_{ij}$$

where p_{ij} = prob(class j in the i^{th} interval)

  - If entropy = 0 then it is a pure grouping
  - Total entropy: weighted average of all e_i

$$e = \sum_{i=1}^{n} w_i e_i$$

where n is the number of intervals

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

Dangers of Dimensionality Reduction

- https://cs.gmu.edu/~jessica/DimReducDanger.htm

What is Similarity?

The quality or state of being similar; likeness; resemblance; as, a similarity of features. **Webster's Dictionary**

Similarity is hard to define, but…
"*We know it when we see it*"

The real meaning of similarity is a philosophical question.

We will take a more pragmatic approach.

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

Defining Distance Measures

Definition: Let O_1 and O_2 be two objects from the universe of possible objects. The distance (dissimilarity) is denoted by $D(O_1,O_2)$

What properties should a distance measure have?

- $D(A,B) = D(B,A)$ *Symmetry*
- $D(A,A) = 0$ *Constancy of Self-Similarity*
- $D(A,B) = 0$ Iff $A = B$ *Positivity*
- $D(A,B) \leq D(A,C) + D(B,C)$ *Triangular Inequality*

Measures for which all properties hold are referred to as distance *metrics*.

Intuitions behind desirable distance measure properties I

$D(\mathrm{A,B}) = D(\mathrm{B,A})$ *Symmetry*

Otherwise you could claim:

"Fairfax is close to D.C., but D.C is not close to Fairfax."

Intuitions behind desirable distance measure properties II

$D(\mathrm{A},\mathrm{A}) = 0$ *Constancy of Self-Similarity*

Otherwise you could claim:

"Fairfax is closer to D.C than D.C. itself!".

Intuitions behind desirable distance measure properties III

D(A,B) = 0 iff A=B *Positivity*

Otherwise you could claim:

"Fairfax is exactly at the same location as DC"

Intuitions behind desirable distance measure properties IIII

$D(\text{A,B}) \leq D(\text{A,C}) + D(\text{B,C})$ *Triangular Inequality*

Otherwise you could claim:

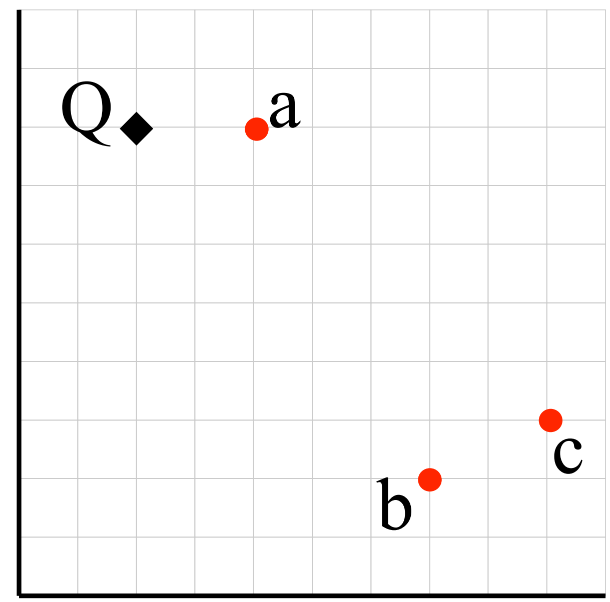
"My house is very close to Fairfax, your house is very close to Fairfax, but my house is very far from your house".

Why is the Triangular Inequality so Important?

Virtually all techniques to index data require the triangular inequality to hold.

Suppose I am looking for the closest point to Q, in a database of 3 objects.

Further suppose that the triangular inequality holds, and that we have pre-computed a table of distances between all the items in the database.



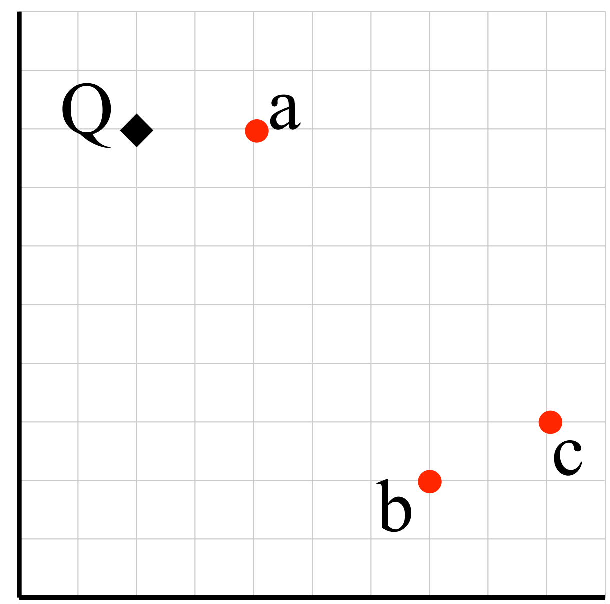
	a	b	c
a		6.70	7.07
b			2.30
c			

Why is the Triangular Inequality so Important?

Virtually all techniques to index data require the triangular inequality to hold.

I find **a** and calculate that it is 2 units from Q, it becomes my *best-so-far*. I find **b** and calculate that it is **7.81** units away from Q.

I don't have to calculate the distance from Q to **c**!



I know $D(Q,\mathbf{b}) \leq D(Q,\mathbf{c}) + D(\mathbf{b},\mathbf{c})$

$$D(Q,\mathbf{b}) - D(b,\mathbf{c}) \leq D(Q,\mathbf{c})$$

$$\mathbf{7.81} - \mathbf{2.30} \leq D(Q,\mathbf{c})$$

$$5.51 \leq D(Q,\mathbf{c})$$

So I know that **c** is at least 5.51 units away, but my *best-so-far* is only 2 units away.

	a	b	c
a		6.70	7.07
b			**2.30**
c			

Euclidean Distance

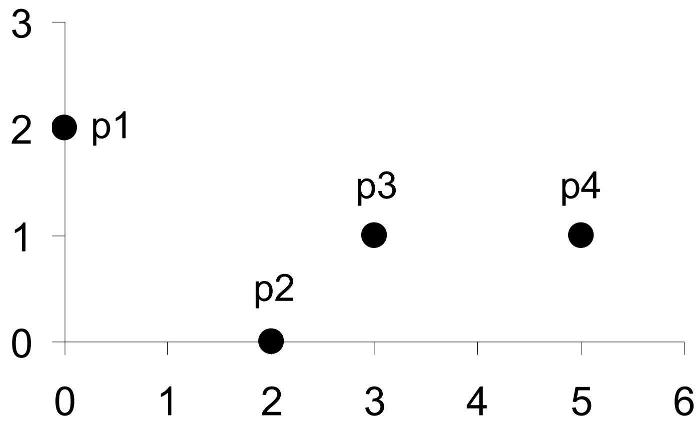
- Euclidean Distance

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

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 or normalization is necessary, if scales differ.
  - Min-max normalization
  - Z-normalization

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, L1 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
- $r \to \infty$. "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

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

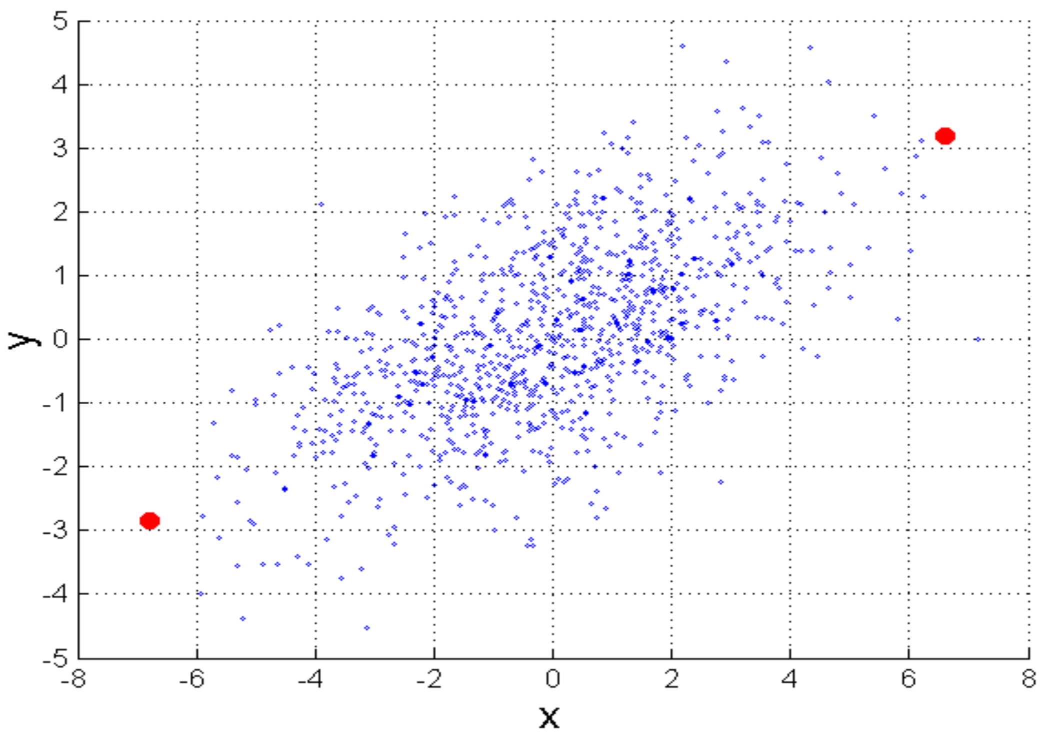
L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

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

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



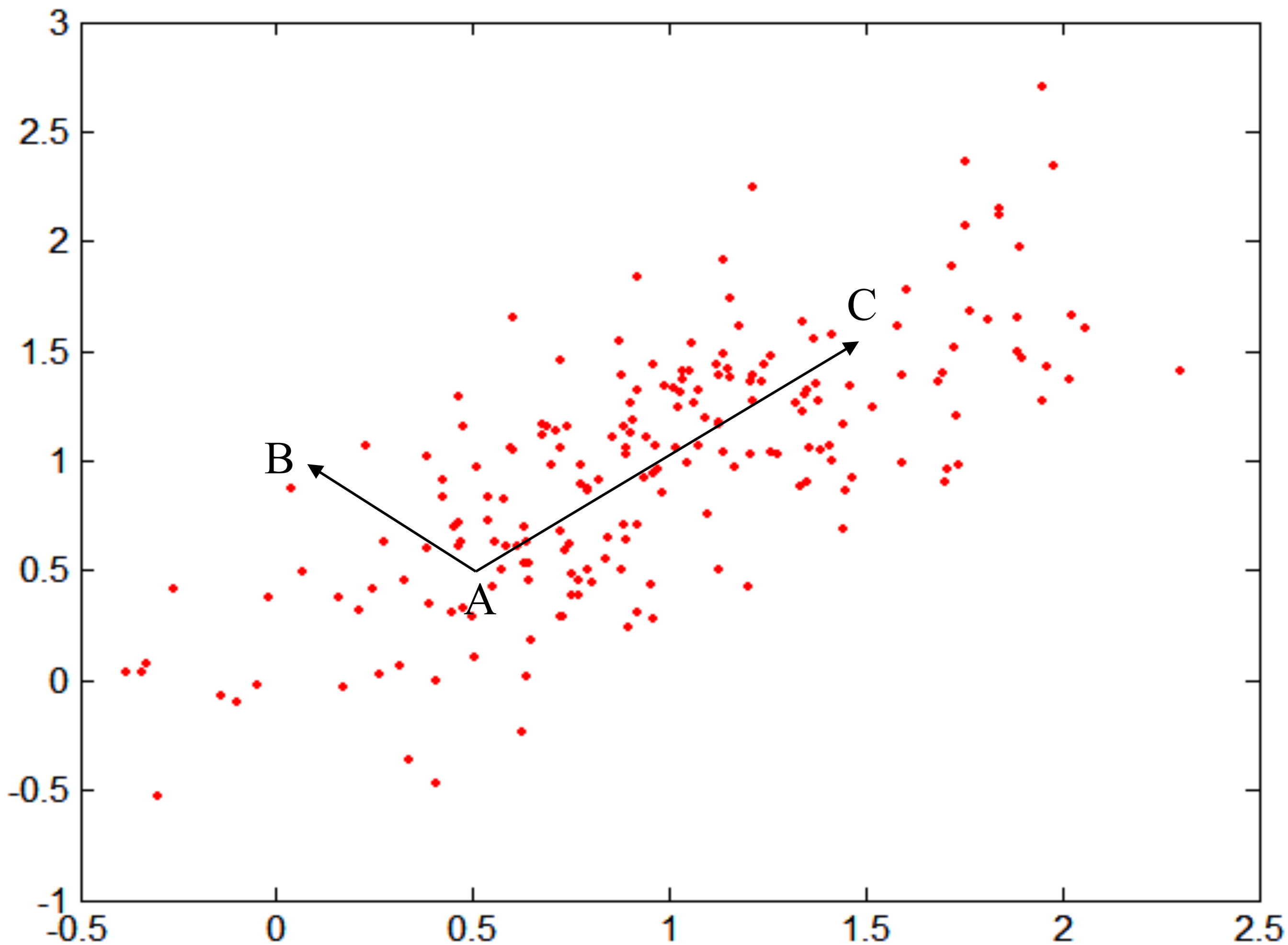
Σ is the covariance matrix of the input data X

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

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

*** In some literature, this is the “squared” distance**

Mahalanobis Distance



Covariance Matrix:

$$\Sigma = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

C: (1.5, 1.5)

Mahal(A,B) = 5

Mahal(A,C) = 4

Common Properties of Similarity

- Similarities also have some well known properties.
  - $s(p, q) = 1$ (or maximum similarity) only if $p = q$.
  - $s(p, q) = s(q, p)$ 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
  $= (M_{11}) / (M_{01} + M_{10} + M_{11})$

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)

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

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

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 • 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**

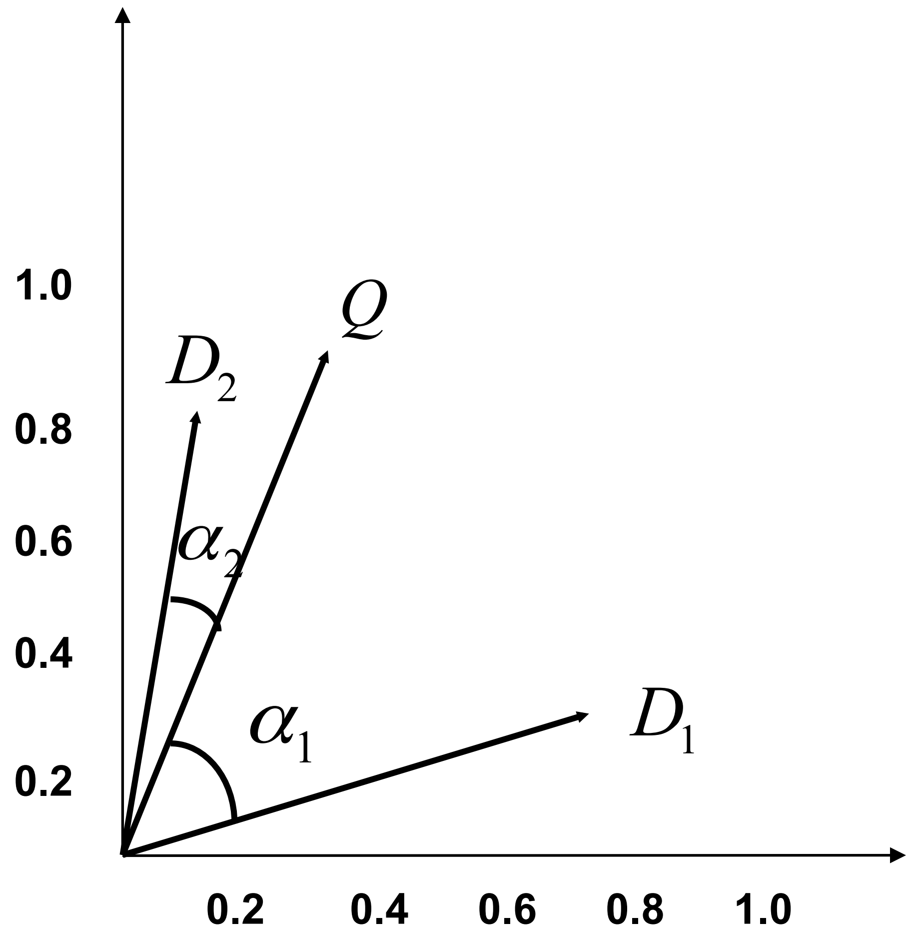
$d_1 \bullet d_2 = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$

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

$\|d_2\| = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{\mathbf{0.5}} = (6)^{\mathbf{0.5}} = 2.45$

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

Cosine Similarity



$$D_1 = (0.8, 0.3)$$

$$D_2 = (0.2, 0.7)$$

$$Q = (0.4, 0.8)$$

$$\cos\alpha_1 = 0.74$$

$$\cos\alpha_2 = 0.98$$

Extended Jaccard Coefficient (Tanimoto)

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

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

Correlation

Correlation measures the linear relationship between objects

$$corr(x, y) = \frac{\text{Covariance}(x, y)}{\text{standard\_dev(x)*standard\_dev(y)}}$$

$$= \frac{S_{xy}}{S_x S_y}$$

Correlation (cont.)

$$\text{covariance(x,y)}=\frac{1}{n-1}\sum_{k=1}^{n}(x_k-\bar{x})(y_k-\bar{y})$$

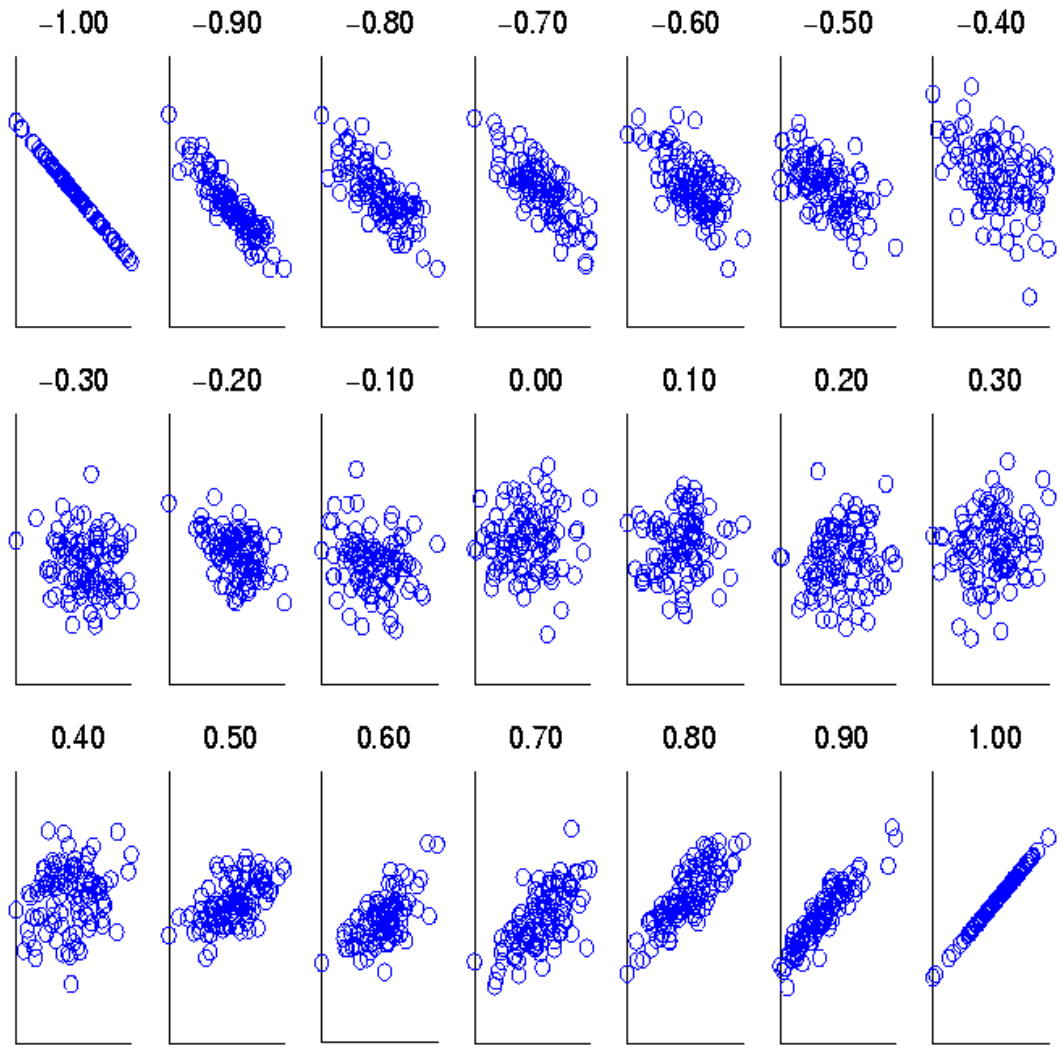
$$\text{standard\_dev(x)}=\mathrm{S_x}=\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(x_k-\bar{x})^2}$$

$$\text{standard\_dev(y)}=\mathrm{S_y}=\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(y_k-\bar{y})^2}$$

Exercise

- x = (1 1 0 0 0), y = (0 0 0 1 1). Compute their correlation.

Visually Evaluating Correlation



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:

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

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 wk which are between 0 and 1 and sum to 1.

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

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

Which similarity function to use ?

- Depends on the application.
  - Analyze the attributes.
  - See their properties, min, max, etc
  - See their dependency on other attributes
  - Do you need similarity or distance ?
  - Do you need a metric ?
  - Try several functions.
  - Combine/merge.
- Active area of research!