Data Science (CS3206)

Lecture #6

Measuring Data Similarity and Dissimilarity

Today’s discussion…

- Similarity and Dissimilarity
- Data Matrix and Dissimilarity Matrix
- Proximity Measure for Nominal Attributes
- Proximity Measure for Binary Attributes
- Distance on Numeric Data
- Proximity Measure for Ordinal Variables
- Dissimilarity for Attributes of Mixed Types

Why to measure Similarity and Dissimilarity

- In applications, such as clustering, outlier analysis, and nearest-neighbor classification, we need ways to assess how alike or unalike objects are in comparison to one another

- For example, a store may want to search for clusters of customer objects, resulting in groups of customers with similar characteristics (e.g., similar income, area of residence, and age).

- Such information can then be used for marketing

- A cluster is a collection of data objects such that the objects within a cluster are similar to one another and dissimilar to the objects in other clusters

Similarity and Dissimilarity

- **Similarity**
  - Numerical measure of how alike two data objects are
  - Value is higher when objects are more alike
  - Often falls in the range [0,1] (0 if the objects are not same)
- **Dissimilarity** (e.g., distance)
  - Numerical measure of how different two data objects are
  - Lower when objects are more alike
  - Minimum dissimilarity is often 0 (objects are same)
  - Upper limit varies
- **Proximity** refers to a similarity or dissimilarity

Data Matrix and Dissimilarity Matrix

- Data matrix
  - object-by-attribute structure
  - n data points with p dimensions
  - Two-mode matrix

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

- Dissimilarity matrix
  - n data points, but registers only the distance
  - A triangular matrix
  - Single mode matrix

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

sim(i, j) = 1 − d(i, j)

Proximity Measure for Nominal Attributes

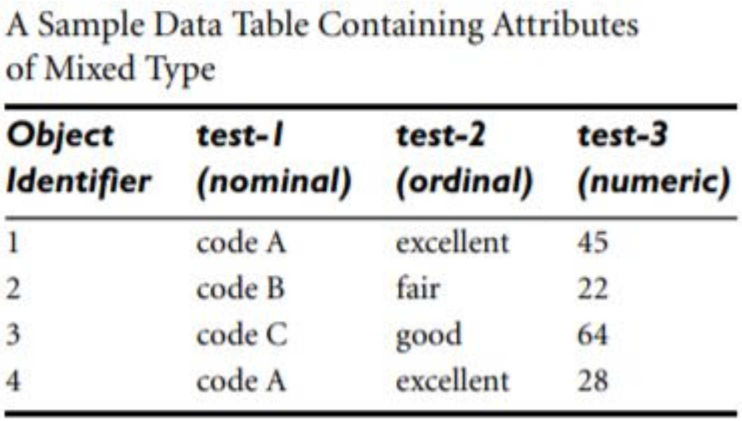
- Can take 2 or more states, e.g., red, yellow, blue, green (generalization of a binary attribute)

  Let the number of states M.
  denoted by letters, symbols, or a set of integers, such as 1, 2,..., M. Notice that such integers are used just for data handling and do not represent any specific ordering.

- Method 1: Simple matching
  - *m*: # of matches, *p*: total # of variables

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

Proximity Measure for Nominal Attributes



A Sample Data Table Containing Attributes of Mixed Type

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

test-1 is nominal
set p = 1

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

Similarity can be computed as

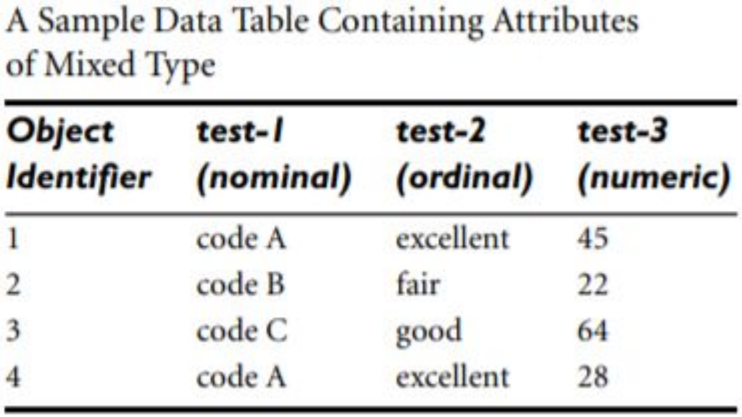
$$sim(i,j) = 1 - d(i,j) = \frac{m}{p}.$$

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

all objects are dissimilar except objects 1 and 4

Proximity Measure for Nominal Attributes

- Method 2: Use a large number of binary attributes
  - creating a new binary attribute for each of the *M* nominal states



A Sample Data Table Containing Attributes of Mixed Type

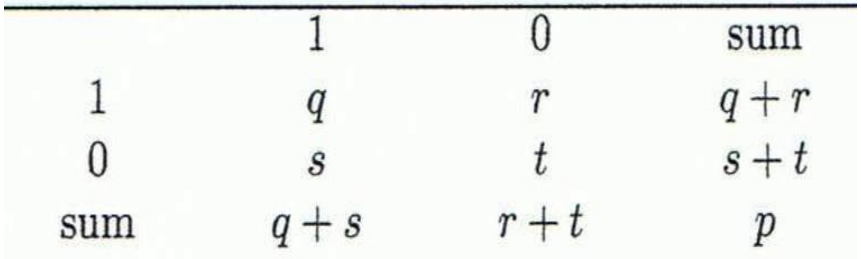
Object Identifier	*test-1 (nominal)*	*test-2 (ordinal)*	*test-3 (numeric)*
1	code A	excellent	45
2	code B	fair	22
3	code C	good	64
4	code A	excellent	28

object id	**code A**	**code B**	**code C**
1	1	0	0
2	0	1	0
3	0	0	1
4	1	0	0

Proximity Measure for Binary Attributes

- A 2*2 contingency table for binary data

Object *j*



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

where
q is the number of attributes that equal 1 for both objects i and j
r is the number of attributes that equal 1 for object i but equal 0 for object j
s is the number of attributes that equal 0 for object i but equal 1 for object j
t is the number of attributes that equal 0 for both objects i and j.

The total number of attributes is p, where p = q + r + s + t

Proximity Measure for Binary Attributes

- Distance measure for symmetric binary variables:

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

- Distance measure for asymmetric binary variables:

  Given two asymmetric binary attributes, the agreement of two 1s (a positive match) is then considered more significant than that of two 0s (a negative match). The number of negative matches, t, is considered unimportant and is thus ignored eg disease test

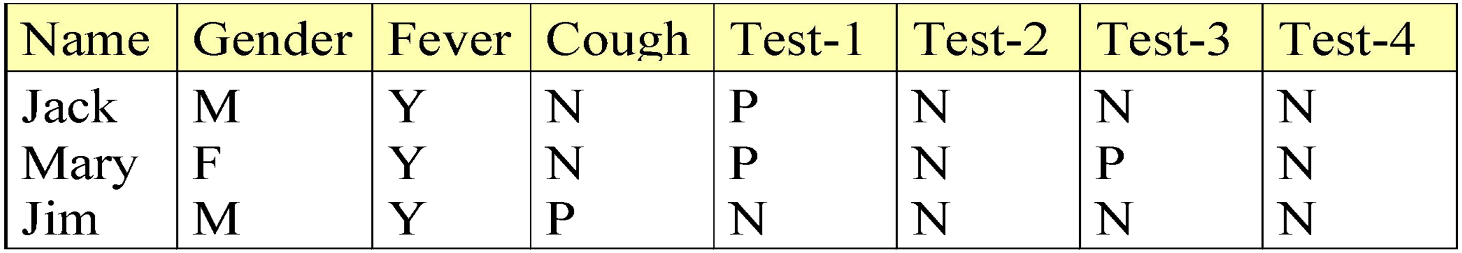
$$d(i, j) = \frac{r+s}{q+r+s}$$

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

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

Dissimilarity between Binary Variables

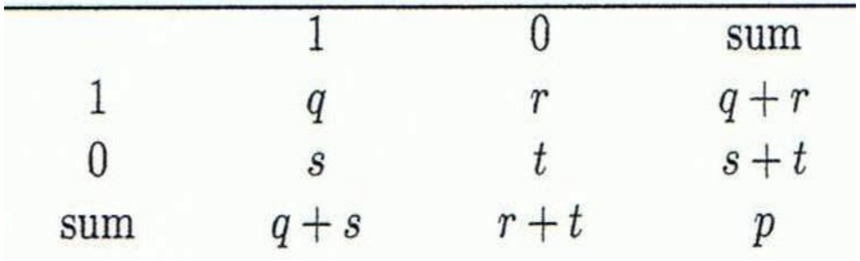
- Example



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 0

Object *j*



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

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

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

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

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

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

Jim and Mary are unlikely to have a similar disease because they have the highest dissimilarity value among the three pairs.

Of the three patients, Jack and Mary are the most likely to have a similar disease

Distance on Numeric Data

- *Euclidean, Manhattan, and Minkowski distances*

  The data are normalized before applying distance calculations

  Transforming the data to fall within a smaller or common range, such as [−1,1] or [0.0, 1.0]

  Consider a height attribute, for example, which could be measured in either meters or inches

  In general, expressing an attribute in smaller units will lead to a larger range for that attribute

  Thus tend to give such attributes greater effect or "weight."

  Normalizing the data attempts to give all attributes an equal weight

Distance on Numeric Data: Euclidean distance

- *Euclidean distance*: The most popular distance measure
  (i.e., straight line or "as the crow flies")

Let i = (xi1, xi2,..., xip) and j = (xj1, xj2,..., xjp) be two objects described by p numeric attributes.

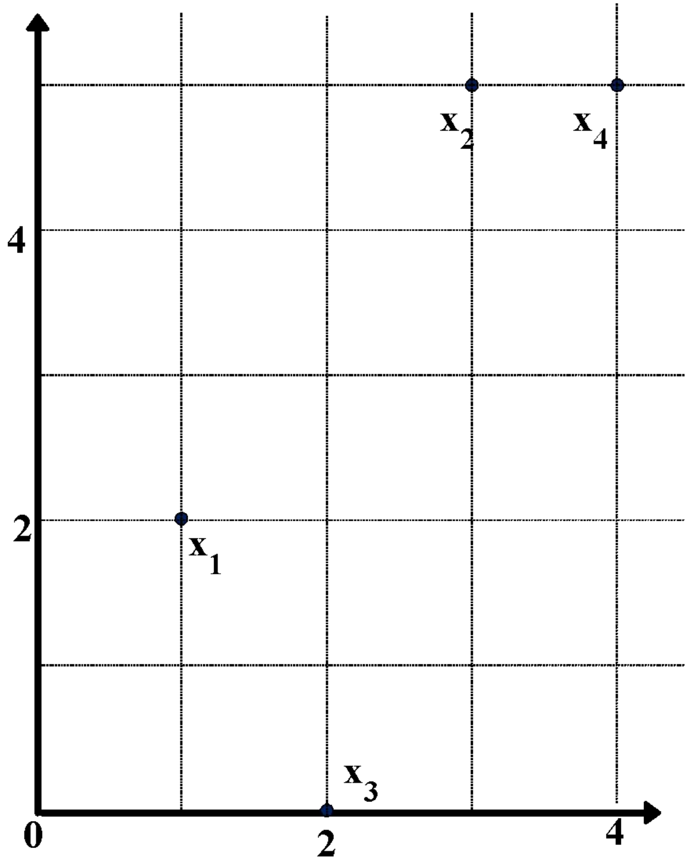
The Euclidean distance between objects i and j is defined as

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2}.$$

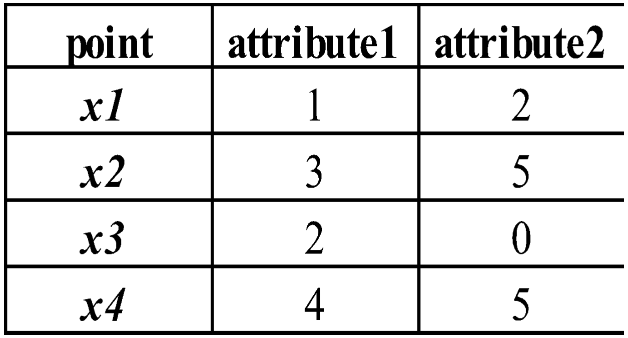
Let x1 = (1, 2) and x2 = (3, 5)

$$\sqrt{2^2 + 3^2} = 3.61$$

Distance on Numeric Data: Euclidean distance



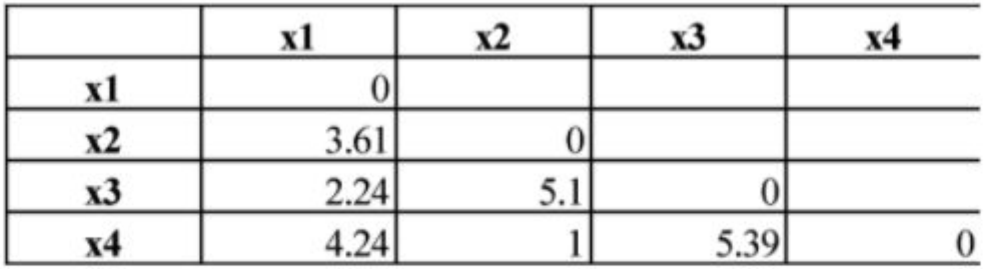
Data Matrix



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

Dissimilarity Matrix

(with Euclidean Distance)



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

Distance on Numeric Data: Manhattan Distance

- *Manhattan distance*: (or city block) named so because it is the distance in blocks between any two points in a city (such as 2 blocks down and 3 blocks over for a total of 5 blocks).

- It is defined as

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

Let x1 = (1, 2) and x2 = (3, 5)

The Manhattan distance between the two is 2 + 3 = 5

Distance on Numeric Data: Manhattan Distance

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

- Properties
  - Non-negativity: $d(i, j) > 0$ if $i \neq j$
  - Identity of indiscernibles: $d(i, i) = 0$
  - Symmetry: $d(i, j) = d(j, i)$
  - Triangle Inequality: $d(i, j) \leq d(i, k) + d(k, j)$

- A distance that satisfies these properties is a metric

Distance on Numeric Data: Minkowski Distance

- *Minkowski distance*: a generalization of the Euclidean and Manhattan distances

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

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects,

h is the order (the distance so defined is also called L-h norm)

h is a real number such that h ≥ 1

Special Cases of Minkowski Distance

- *h* = 1: Manhattan (city block, L_1 norm) distance
  - E.g., the Hamming distance: the number of bits that are different between two binary vectors

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

- *h* = 2: (L_2 norm) Euclidean distance

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

- *h* → ∞. “supremum” (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component (attribute) of the vectors

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

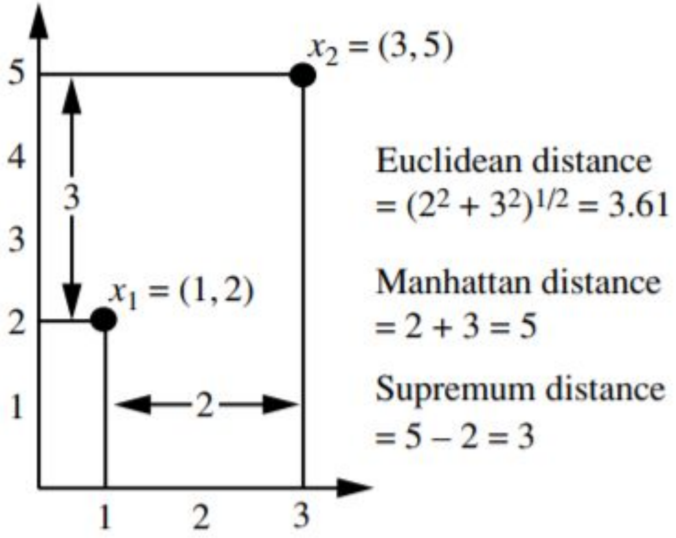
find the attribute f that gives the maximum difference in values between the two objects

Special Cases of Minkowski Distance

Supremum distance

Let’s use the same two objects, x1 = (1, 2) and x2 = (3, 5)

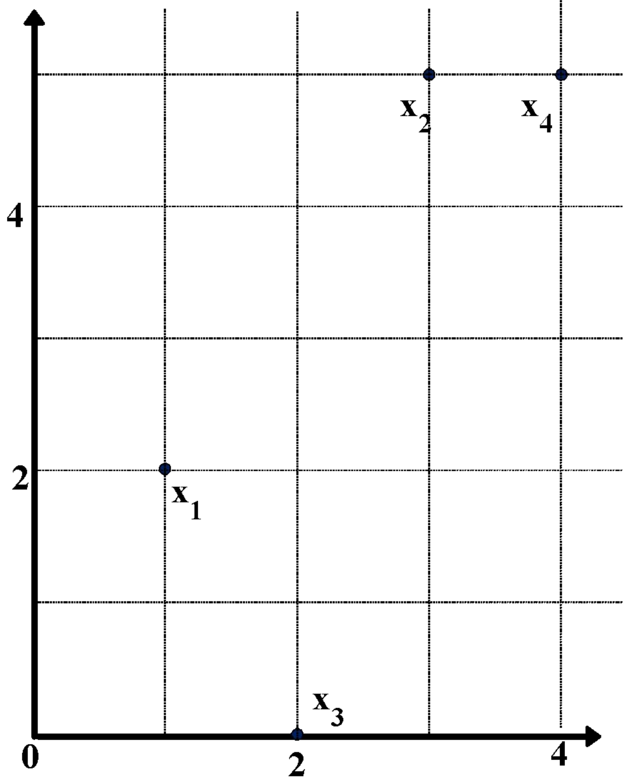
The second attribute gives the greatest difference between values for the objects, which is 5 − 2 = 3.



Euclidean, Manhattan, and supremum distances between two objects.

Example: Minkowski Distance

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



Dissimilarity Matrices

Manhattan (L_1)

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

Euclidean (L_2)

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

Supremum

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

Proximity Measure for Ordinal Variables

- An ordinal variable have a meaningful order or ranking about them
- Order is important, e.g., rank
- the magnitude between successive values is unknown

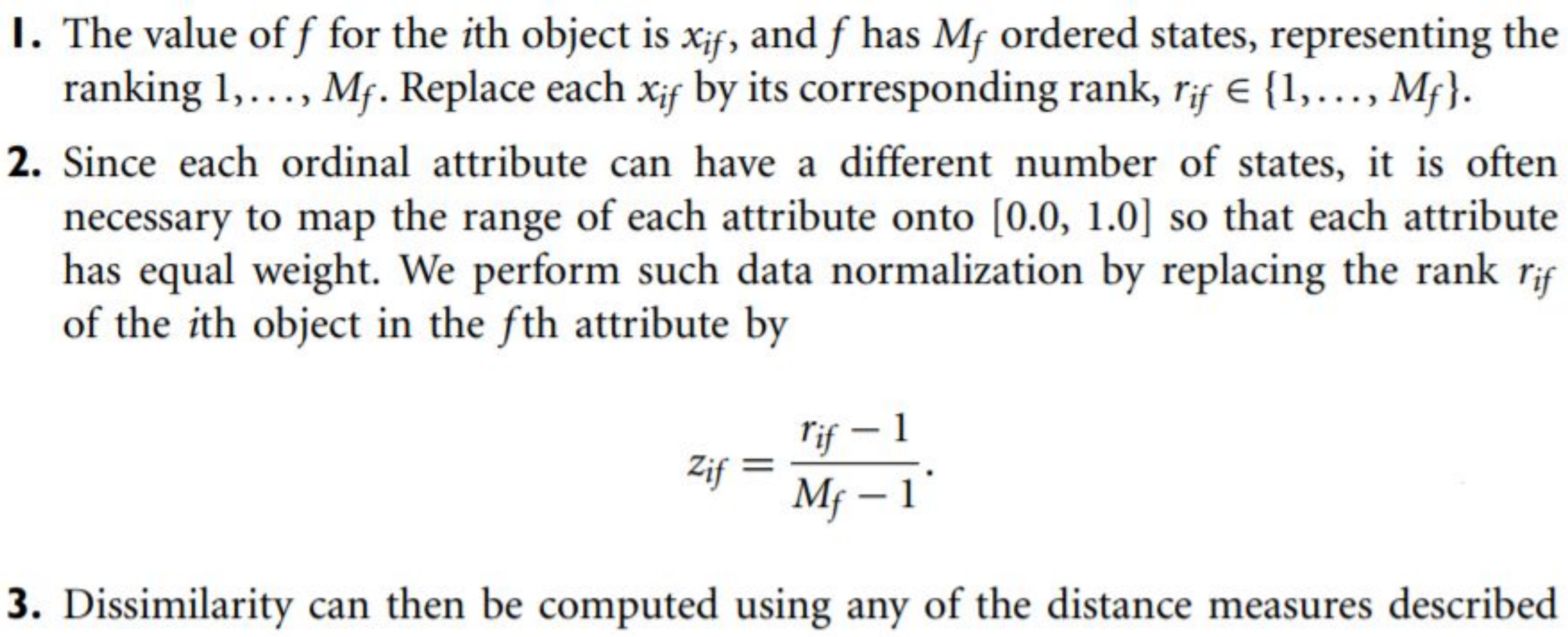
For example, the range of temperature (in Celsius) can be organized into the following states: −30 to −10, −10 to 10, 10 to 30

cold temperature, moderate temperature, and warm temperature

Let M represent the number of possible states that an ordinal attribute can have. These ordered states define the ranking 1,..., Mf .

Proximity Measure for Ordinal Variables

Suppose that f is an attribute from a set of ordinal attributes describing n objects. The dissimilarity computation with respect to f involves the following steps:



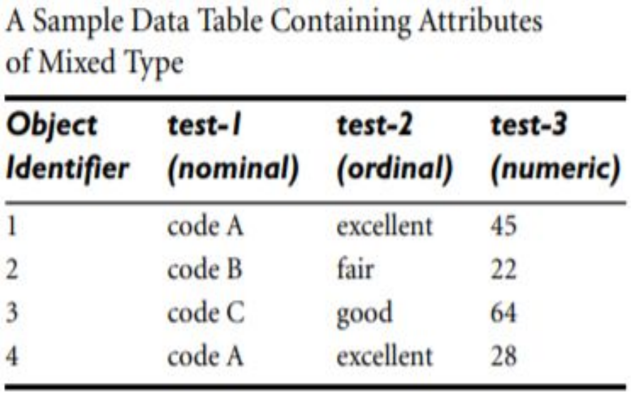
1. The value of f for the ith object is x_{if}, and f has M_f ordered states, representing the ranking $1, \ldots, M_f$. Replace each x_{if} by its corresponding rank, $r_{if} \in \{1, \ldots, M_f\}$.
2. Since each ordinal attribute can have a different number of states, it is often necessary to map the range of each attribute onto [0.0, 1.0] so that each attribute has equal weight. We perform such data normalization by replacing the rank r_{if} of the ith object in the fth attribute by

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

3. Dissimilarity can then be computed using any of the distance measures described

Proximity Measure for Ordinal Variables

Suppose that we have the sample data shown



A Sample Data Table Containing Attributes of Mixed Type

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

three states for test-2: fair, good, and excellent, that is, Mf = 3

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

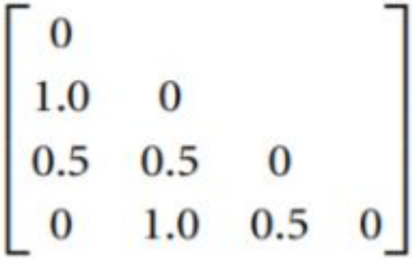
Step 2 normalizes the ranking by mapping rank 1 to 0.0, rank 2 to 0.5, and rank 3 to 1.0

For step 3, we can use, say, the Euclidean distance

objects 1 and 2 are the most dissimilar, as are objects 2 and 4 (i.e., d(2,1) = 1.0 and d(4,2) = 1.0).

This makes intuitive sense since objects 1 and 4 are both excellent.

Object 2 is fair, which is at the opposite end of the range of values for test-2



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

Dissimilarity for Attributes of Mixed Types

- A database may contain all attribute types
  - Nominal, symmetric binary, asymmetric binary, numeric, ordinal
- Performing separate analysis for each type.
- Feasible if these analyses derive compatible results
- Preferable approach is to process all attribute types together, performing a single analysis
- combine the different attributes into a single dissimilarity matrix, bringing all of the meaningful attributes onto a common scale of the interval [0.0, 1.0]

Dissimilarity for Attributes of Mixed Types

- One may use a weighted formula to combine their effects

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

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

  - *f* is binary or nominal:

    $d_{ij}^{(f)} = 0$

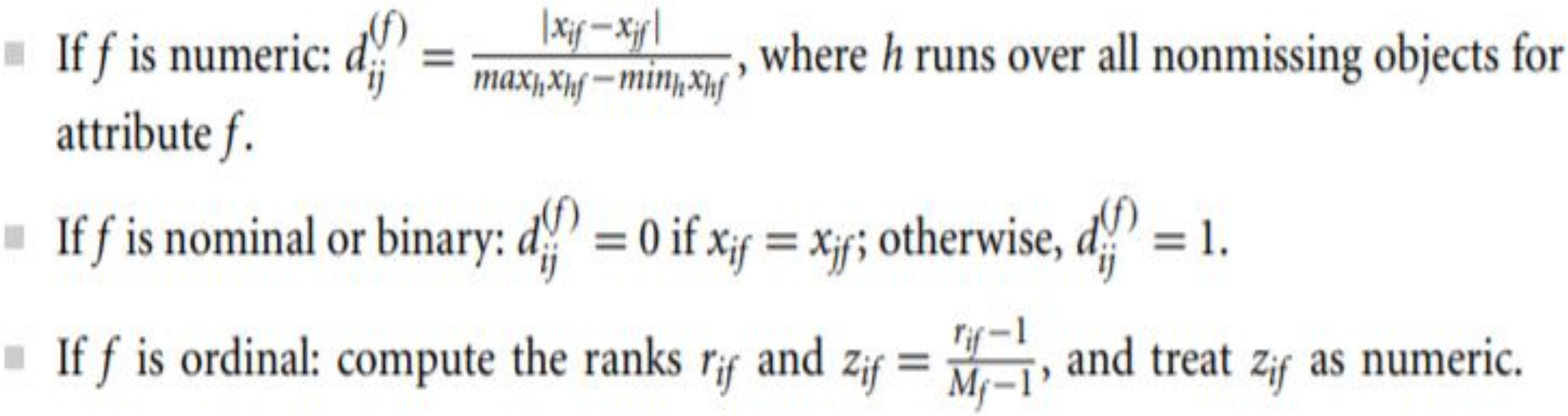
    if either (1) x_{if} or x_{jf} is missing or

    (2)$x_{if} = x_{jf} =0$ and attribute f is asymmetric binary

    $d_{ij}^{(f)} = 1$ otherwise

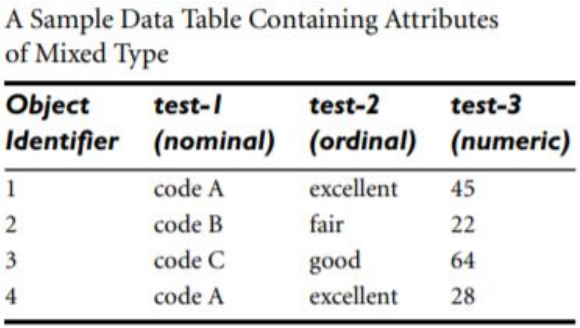
Dissimilarity for Attributes of Mixed Types

The contribution of attribute f to the dissimilarity between i and j is computed dependent on its type



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

Dissimilarity for Attributes of Mixed Types



A Sample Data Table Containing Attributes of Mixed Type

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

test-1

$$\begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ 1 & 1 & 0 & \\ 0 & 1 & 1 & 0 \end{bmatrix}.$$

test-2

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

Dissimilarity matrix for the third attribute, test-3 (which is numeric).
Let maxhxh = 64 and minhxh = 22

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

The indicator $d_{ij}^{(f)} = 1$ for each of the three attributes, f .

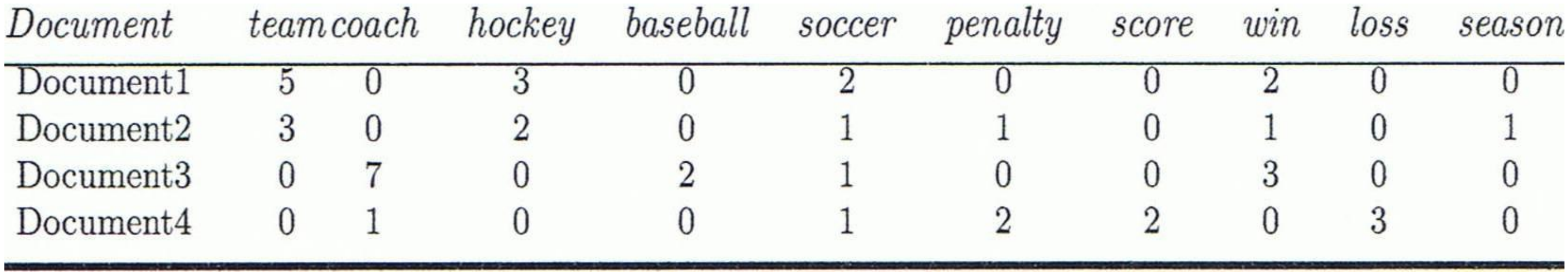
$$d(3,1) = \frac{1(1)+1(0.50)+1(0.45)}{3} = 0.65.$$

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

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.

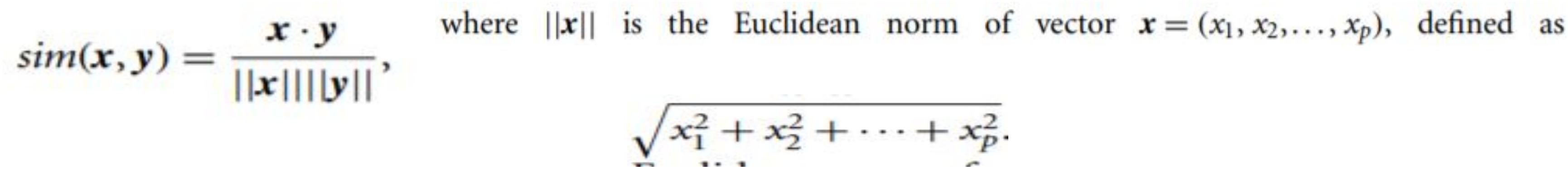
- Applications: information retrieval, biologic taxonomy, gene feature mapping



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

Cosine Similarity

Cosine similarity is a measure of similarity that can be used to compare documents or, say, give a ranking of documents with respect to a given vector of query words. Let x and y be two vectors for comparison



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

where $||\boldsymbol{x}||$ is the Euclidean norm of vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)$, defined as

$$\sqrt{x_1^2 + x_2^2 + \cdots + x_p^2}.$$

The measure computes the cosine of the angle between vectors x and y.

A cosine value of 0 means that the two vectors are at 90 degrees to each other (orthogonal) and have no match.

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

Example: Cosine Similarity

- $\cos(d_1, d_2) = (d_1 \cdot d_2) / ||d_1|| \, ||d_2||$,
  where · 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)

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

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

$\cos(d_1, d_2) = 0.94$