Course: Data Mining
Lecturer: Hend Muslim Jasim
Date: December ,2025
Reference: " Data Mining Concepts and Techniques” Third Edition, Jiawei Han, et al.

Lecture 3: Data Similarity Measures

1. Introduction to Data Similarity and Dissimilarity

In data mining, we often need to quantify how similar or different data objects are from each other. This is crucial for many applications:

- **Clustering**: Grouping similar objects together.
- **Classification**: Assigning labels based on similarity to known examples
- **Outlier Detection**: Identifying objects that are highly dissimilar to others.
- **Information Retrieval**: Finding documents similar to a query.

Key Concepts:

- **Similarity**: Numerical measure of how alike two objects are (higher values indicate greater similarity)
- **Dissimilarity**: Numerical measure of how different two objects are (lower values indicate greater similarity)
- **Distance**: Often used interchangeably with dissimilarity

2. Data Structures for Proximity Measurement

a. Data Matrix

- An **n × p matrix** representing n objects with p attributes
- Also called **object-by-attribute structure**
- Used when we have the raw data values

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

b. Dissimilarity Matrix

- An **n × n matrix** storing pairwise dissimilarities
- Also called **object-by-object structure**
- Used by many clustering and classification algorithms

Properties:

- d(i,i) = 0 (identity: object is identical to itself)
- d(i,j) = d(j,i) (symmetry)
- d(i,j) ≥ 0 (non-negativity)

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

Course: Data Mining
Lecturer: Hend Muslim Jasim
Date: December ,2025
Reference: " Data Mining Concepts and Techniques" Third Edition, Jiawei Han, et al.

3. Proximity Measures for Different Attribute Types

3.1 Nominal Attributes

Characteristics:

- Values are categories without meaningful order
- Examples: color, gender, product type

Dissimilarity Measure:

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

Where:
p = total number of attributes
m = number of matching attributes
The larger the number of mismatches, the higher the dissimilarity.

Similarity Measure:

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

The larger the number of matches, the higher the similarity.

Example:

If two objects have 3 matching attributes out of 5:

$$d(i,j) = \frac{5-3}{5} = 0.4$$

$$\text{sim}(i,j) = \frac{3}{5} = 0.6$$

3.2 Binary Attributes

Characteristics:

- Only two states: 0 or 1
- Can be symmetric or asymmetric

Contingency Table:

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

Where:

- q = number of attributes where both objects have 1
- r = number where i=1, j=0
- s= number where i=0, j=1
- t = number where both objects have 0
- p= total number of attributes

Course: Data Mining
Lecturer: Hend Muslim Jasim
Date: December ,2025
Reference: " Data Mining Concepts and Techniques" Third Edition, Jiawei Han, et al.

Symmetric Binary Dissimilarity : both states (1 and 0) equally important

Formula : **d(i,j) = (r + s)/(q + r + s + t) = (mismatches)/total attributes)**

Example: comparing two friends' likes/dislikes for movies. If they disagree on two out of ten movies, dissimilarity = 2/10 = 0.2.

Asymmetric Binary Dissimilarity (positive matches more important): means one state matters more than the other. For example: disease test → positive (1) is more important than negative (0). Ignore negative matches (t) because two people both being negative is less interesting.

Formula: d(I , j) = (r + s)/(q + r + s)

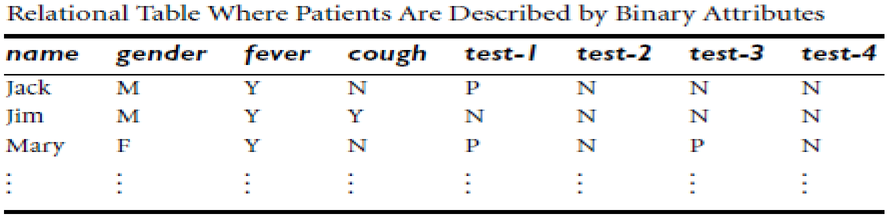
Asymmetric Binary Similarity: Instead of dissimilarity, we can measure **similarity**.

- This is called the **Jaccard coefficient**, widely used in information retrieval, ecology, etc.
- **Formula**: sim (i, j) =q /(q+r+s)= 1-d(i,j)

Example: two friends like 5 of the same movies out of 8 they have opinions on

similarity = 5/8 = 0.625

Example: **Dissimilarity between binary attributes.** Suppose that a patient record table contains the attributes *name, gender, fever, cough, test-1, test-2, test-3*, and *test-4*, where *name* is an object identifier, *gender* is a symmetric attribute, and the remaining attributes are asymmetric binary. For asymmetric attribute values, let the values *Y* (*yes*) and *P* (*positive*) be set to 1, and the value *N* (*no* or *negative*) be set to 0. Suppose that the distance between objects



Relational Table Where Patients Are Described by Binary Attributes

name	*gender*	*fever*	*cough*	*test-1*	*test-2*	*test-3*	*test-4*
Jack	M	Y	N	P	N	N	N
Jim	M	Y	Y	N	N	N	N
Mary	F	Y	N	P	N	P	N
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Attributes:(name, gender, fever, cough, test-1, test-2, test-3, test-4)

- **Gender**: symmetric
- **Others (fever, cough, tests)**: asymmetric binary

Encoding:
Y (yes) / P (positive) → 1
N (no/negative) → 0
For **asymmetric attributes only** (fever, cough, test-1, test-2, test-3, test-4), we compute distances
Data (asymmetric part, 6 attributes):

- **Jack**: 1, 0, 1, 0, 0, 0 → (Y,N,P,N,N,N)
- **Jim**: 1, 1, 0, 0, 0, 0 → (Y,Y,N,N,N,N)
- **Mary**: 1, 0, 1, 0, 1, 0 → (Y,N,P,N,P,N)

Compute d(Jack, Jim)

Compare Jack & Jim across 6 asymmetric attributes:

1. fever: both 1 → q=1
2. cough: Jack=0, Jim=1 → s=1 (for Jack)
3. test-1: Jack=1, Jim=0 → r=1
4. test-2: both 0 → ignored (asymmetric)
5. test-3: both 0 → ignored
6. test-4: both 0 → ignored

So: q=1, r=1, s=1

D= (r+s) /(q+r+s)= (1+1)/ (1+1+1)= 2/3 = 0.67

Compute d(Jack, Mary)

1. fever: both 1 → q=1
2. cough: both 0 → ignore
3. test-1: both 1 → q=2
4. test-2: both 0 → ignore
5. test-3: Jack=0, Mary=1 → s=1
6. test-4: both 0 → ignore

So: q=2, r=0, s=1

D= (r+s) /(q+r+s)= (0+1)/ (2+0+1)= 1/3 = 0.33

Compute d(Jim, Mary)

1. fever: both 1 → q=1
2. cough: Jim=1, Mary=0 → r=1
3. test-1: Jim=0, Mary=1 → s=1
4. test-2: both 0 → ignore
5. test-3: Jim=0, Mary=1 → s=2
6. test-4: both 0 → ignore

So: q=1, r=1, s=2

D= (r+s) /(q+r+s)= (1+2)/ (1+1+2)= 3/4 = 0.75

These measurements suggest that 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.

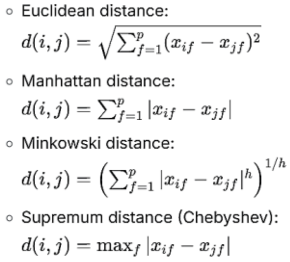
- **Jim & Mary** have highest dissimilarity (0.75) → least similar.
- **Jack & Mary** have lowest dissimilarity (0.33) → most similar in terms of disease indicators.

Course: Data Mining
Lecturer: Hend Muslim Jasim
Date: December ,2025
Reference: " Data Mining Concepts and Techniques" Third Edition, Jiawei Han, et al.

3.3 Numeric Attributes

Numeric attributes are **quantitative measurements** (like age, salary, temperature) that support arithmetic operations. Choosing the right distance measure for numeric data is crucial for clustering, classification, and similarity search.

Common Distance Measures:



- Euclidean distance:
  $d(i,j) = \sqrt{\sum_{f=1}^{p}(x_{if} - x_{jf})^2}$
- Manhattan distance:
  $d(i,j) = \sum_{f=1}^{p} |x_{if} - x_{jf}|$
- Minkowski distance:
  $d(i,j) = \left(\sum_{f=1}^{p} |x_{if} - x_{jf}|^h\right)^{1/h}$
- Supremum distance (Chebyshev):
  $d(i,j) = \max_f |x_{if} - x_{jf}|$

3.4 Ordinal Attributes

Handling ordinal data requires mapping the ordered states to ranks and then normalizing.

- **Process:**
  1. **Replace values with ranks:** Map each ordinal state (e.g., fair, good, excellent) to a rank (e.g., 1, 2, 3).
  2. **Normalize ranks:** Scale the ranks to the interval **[0.0, 1.0]** to give all attributes equal weight. Formula**: z_ij = (rank_ij - 1) / (M_f - 1),** where M_f is the number of possible states.
  3. **Compute dissimilarity:** Use any numeric distance measure (like Euclidean distance) on the normalized values z_ij.
- **Example 2.21:** Objects with values {excellent, fair, good, excellent} are mapped to ranks {3, 1, 2, 3}, then normalized to {1.0, 0.0, 0.5, 1.0}. Euclidean distance is then applied.

3.5 Mixed Attribute Types

Combined Dissimilarity Measure:

Real-world datasets often contain multiple attribute types. The solution is to combine them into a single dissimilarity measure.

- **Overall Formula:** A weighted average of the dissimilarities from each attribute:

d(i, j) = [Σ (δ_ij^(f) * d_ij^(f))] / [Σ δ_ij^(f)]

Indicator δ_ij^(f): This is 1 if attribute f is valid for comparison for objects i and j. It's 0 if:

a) the value is missing for either object, or

b) the attribute is asymmetric binary and both values are 0.

- **Attribute-Specific Contribution (d_ij^(f)):**
  - **Numeric:** Normalized absolute difference: |x_if - x_jf| / (max_h x_hf - min_h x_hf). This scales the difference to [0,1].
  - **Nominal/Binary:** Simple matching. d=0 if values are the same, d=1 if they differ.
  - **Ordinal:** Treat as numeric after converting values to normalized ranks (z_if).

4. Special Similarity Measures

4.1 Cosine Similarity

- This measure is crucial for using Text documents, term-frequency vectors, and high-dimensional sparse data
- **Why it's needed:** Traditional distances (e.g., Euclidean) perform poorly on sparse vectors. Two documents might share many zero counts for uncommon words, which shouldn't make them similar. Cosine similarity focuses on the angle between vectors, ignoring the magnitude (length), which is heavily influenced by zeros.
- **Formula:** sim(x, y) = (x · y) / (||x|| * ||y||)

  x · y is the dot product (sum of the products of corresponding attributes).

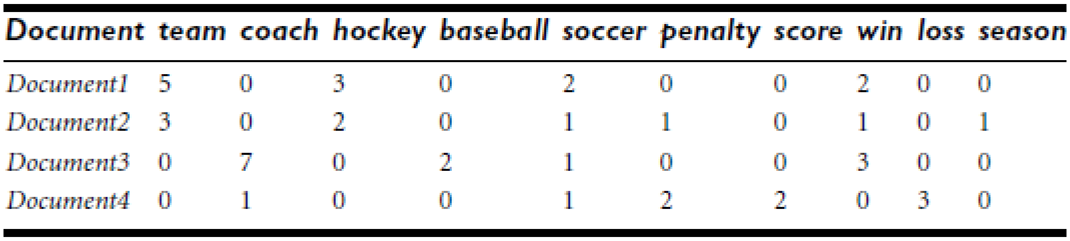
  ||x|| is the Euclidean norm (length) of vector x.

- **Interpretation:** It measures the **cosine of the angle** between two vectors.

  1: Vectors point in the same direction (perfect similarity).
  0: Vectors are orthogonal (no similarity).
  -1: opposite direction (dissimilar)

It is a **nonmetric** measure because it doesn't satisfy all properties of a metric distance (like the triangle inequality).

Example: Calculates the cosine similarity between two document vectors from Table



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

Document vectors

- Document1: (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)
- Document2: (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)
- Dot product: 5×3+0×0+3×2+...=255×3+0×0+3×2+...=25
- Norms: ∥*x*∥=6.48, ∥*y*∥=4.12
- Cosine similarity: 25/(6.48×4.12)=0.9425/(6.48×4.12)=0.94 (very similar)
-

- **Special Case - Binary Attributes:** For binary vectors, cosine similarity has a clear interpretation:

  x · y = number of attributes where both x and y are 1.
  ||x|| * ||y|| is related to the geometric mean of the counts of 1s in each vector.

4.2 Tanimoto Coefficient

A common variation for binary data:
$\text{sim}(x,y) = (x \cdot y) / (x \cdot x + y \cdot y - x \cdot y)$.

This is the ratio of shared attributes to the total attributes possessed by either object.

5. Practical Considerations

Data Normalization

- Important when attributes have different scales
- Prevents attributes with larger ranges from dominating
- Common methods: min-max, z-score, decimal scaling

Weighting

- Attributes can be weighted based on importance
- Weighted Euclidean distance:

$$d(i, j) = \sqrt{w_1|x_{i1} - x_{j1}|^2 + w_2|x_{i2} - x_{j2}|^2 + \cdots + w_m|x_{ip} - x_{jp}|^2}.$$

Handling Missing Values

- Exclude missing attributes from calculations
- Use indicator variables to track available data
- Impute missing values before proximity calculation

Summary

- **Similarity/dissimilarity measures** are fundamental to many data mining tasks
- **Choice of measure** depends on attribute types and application requirements
- **Numeric attributes**: Euclidean, Manhattan, Minkowski distances
- **Categorical attributes**: Matching coefficients, Jaccard similarity
- **Text/sparse data**: Cosine similarity
- **Mixed types**: Combined dissimilarity measures

Understanding these measures allows you to choose the right approach for your specific data mining task and ensures meaningful results.