Data Similarity & Quality Assessment

Foundations of Data Mining and Quality Control

The Recommendation Problem

Imagine you're Netflix. How do you mathematically determine that *The Dark Knight* is more similar to *Inception* than to *The Notebook*?

Learning Objectives:

- Calculate similarity measures for various data types.
- Distinguish between data and dissimilarity matrices.
- Identify and resolve data quality issues.

Data Matrix vs. Dissimilarity Matrix

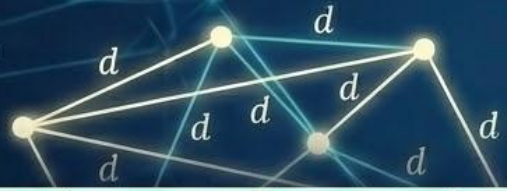
Data Matrix ($n \times p$)

Contains raw attribute values for each object.

Object	Age	Income	Ed.
P1	25	50k	Bach
P2	30	65k	Mast
P3	35	80k	PhD

Dissimilarity Matrix ($n \times n$)

Contains pairwise distances (d). Symmetric, diagonal is 0.



	P1	P2	P3
P1	0	0.4	0.7
P2	0.4	0	0.3
P3	0.7	0.3	0

Proximity Measures: Categorical

Nominal

Simple Matching

$$d = \frac{p - m}{p}$$

All mismatches are equally important. Used for distinct categories like City or Color.

Symmetric Binary

Simple Matching

$$a+d/Total$$

Both states (0 and 1) are equally important (e.g., Gender).

Asymmetric Binary

Jaccard Coefficient

$$J = \frac{a}{a + b + c}$$

Ignores negative matches (d). Vital for rare events like diseases or purchases.

```
Customer1: {Gender: M, City: NY, Product: A}
Customer2: {Gender: M, City: LA, Product: B}


d = (3 - 1)/3 = 2/3 = 0.667
```

		Object j	
		1	0
Object i	1	a	b
	0	c	d

where: a = both 1, d = both 0, b/c = mismatches

Example 2.18. Dissimilarity between nominal attributes. Suppose that we have the sample data of Table 2.4, except that only the *object-identifier* and the attribute *test-1* are available, where *test-1* is nominal. (We will use *test-2* and *test-3* in later examples.) Let's compute the dissimilarity matrix Eq. (2.14), that is,

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

Since here we have one nominal attribute, *test-1*, we set $p = 1$ in Eq. (2.16) so that $d(i, j)$ evaluates to 0 if objects i and j match, and 1 if the objects differ. Thus, we get

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

From this, we see that all objects are dissimilar except objects 1 and 4 (i.e., $d(4, 1) = 0$). □

Table 2.4 A sample data table containing attributes of mixed types.

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

- 100 patients: 90 healthy (0), 10 sick (1)
- Two tests agree on 8 sick patients (a=8)
- Jaccard = 8/(8+1+1) = 0.8 (more meaningful than simple matching)

Example 2.19. Dissimilarity between binary attributes. Suppose that a patient record table (Table 2.6) 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 binary attribute, and the remaining attributes are asymmetric binary. □

Table 2.6 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
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Numeric Distance: Minkowski

The Formula & Types

$$d(i,j) = \left(\sum_{k=1}^{p} |x_{ik} - x_{jk}|^{r} \right)^{\frac{1}{r}}$$

r = 1: Manhattan (City Block)
Sum of absolute differences. Path follows grid lines.

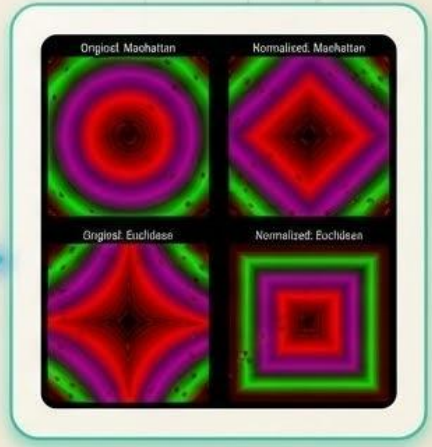
r = 2: Euclidean (Straight Line)
Standard straight-line distance.

r = ∞: Chebyshev (Max dimension diff)
d = max(|x1-x2|, |y1-y2|)

Normalization Critical

Without normalization, attributes with larger ranges (e.g., Income) dominate results, skewing distance calculations.

$$x_{new} = \frac{x - min}{max - min}$$



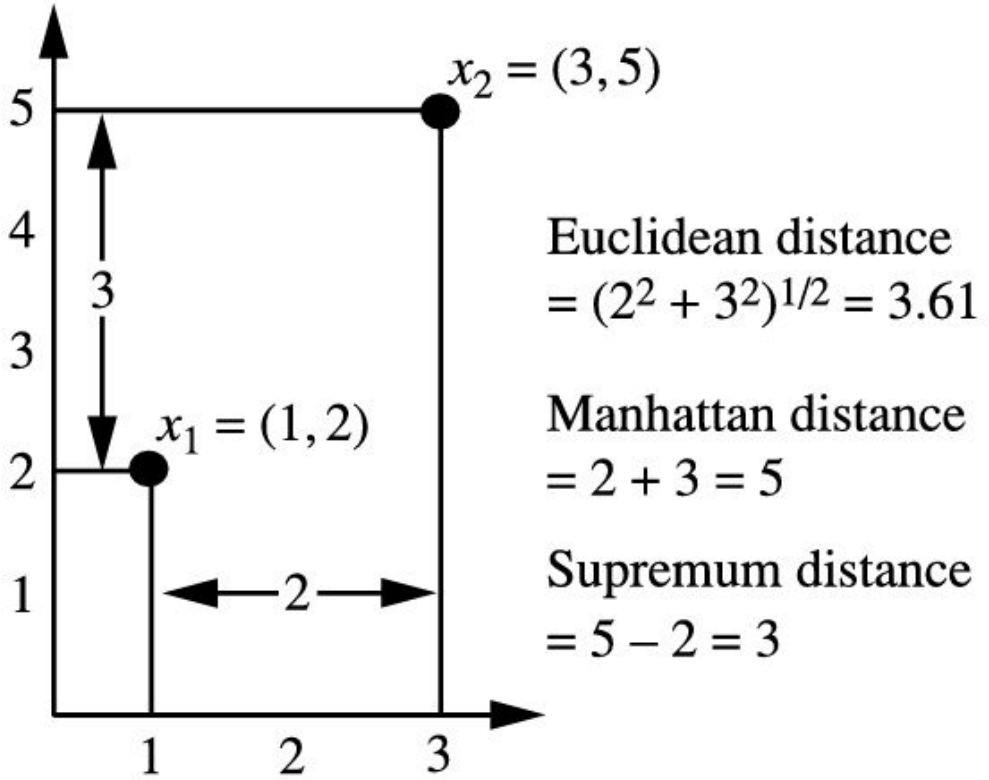


FIGURE 2.10

Euclidean, Manhattan, and supremum distances between two objects.

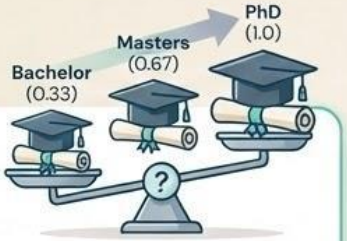
```
Point A: (Age=25, Income=50K)
Point B: (Age=35, Income=80K)

After normalization (Age:20-60, Income:30-100K):
A_norm: (0.125, 0.286)
B_norm: (0.375, 0.714)
```

```
Euclidean: √[(0.25)² + (0.428)²] = 0.495
```

Example 2.20. Euclidean distance and Manhattan distance. Let $x_1 = (1, 2)$ and $x_2 = (3, 5)$ represent two objects as shown in Fig. 2.10. The Euclidean distance between the two is $\sqrt{2^2 + 3^2} = 3.61$. The Manhattan distance between the two is $2 + 3 = 5$. □

Example 2.21. Supremum distance. Let's use the same two objects, $x_1 = (1, 2)$ and $x_2 = (3, 5)$, as in Fig. 2.10. The second attribute gives the greatest difference between the values for the objects. That is, $max\{(|3 - 1|, |5 - 2|\} = 3$. This is the supremum distance between the two objects. □

Ordinal & Mixed Types



Ordinal Attributes

Order matters, but magnitude is vague.

Procedure:

1. Map values to ranks (1 to M).
2. Normalize ranks to [0.0, 1.0].
3. Use numeric measures on normalized values.

Mixed Attributes

Combining nominal, numeric, and binary.

$$d(i,j) = \frac{\sum \delta_{ij} d_{ij}}{\sum \delta_{ij}}$$

Compute individual attribute distances first, then calculate a weighted average.

Example: Education level:

- High School: 1
- Bachelor's: 2
- Master's: 3
- PhD: 4

Normalized: HS=0, Bachelor=0.33, Master=0.67, PhD=1.0

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

Example 2.22. Dissimilarity between ordinal attributes. Suppose that we have the sample data shown earlier in Table 2.4, except that this time only the *object-identifier* and the continuous ordinal attribute, *test-2*, are available. There are three states for *test-2*: *fair*, *good*, and *excellent*, that is, $M_f = 3$. For step 1, if we 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 defined in Eq. (2.21), which results in the following dissimilarity matrix:

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

Therefore 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*. □

```
Customer1: {Age:30(numeric), Gender:M(nominal), Income:70K(numeric)}
Customer2: {Age:25, Gender:M, Income:50K}

d_age = |30-25|/max_age_diff = 5/40 = 0.125
d_gender = 0 (match)
d_income = |70-50|/max_income_diff = 20/60 = 0.333


Overall = (0.125 + 0 + 0.333)/3 = 0.152
```

Example 2.23. Dissimilarity between attributes of mixed types. Let's compute a dissimilarity matrix for the objects in Table 2.4. Now we will consider *all* of the attributes, which are of different types. In Examples 2.18 and 2.22, we worked out the dissimilarity matrices for each of the individual attributes. The procedures we followed for *test-1* (which is nominal) and *test-2* (which is ordinal) are the same as outlined earlier for processing attributes of mixed types. Therefore we can use the dissimilarity matrices obtained for *test-1* and *test-2* later when we compute Eq. (2.27). First, however, we need to compute the dissimilarity matrix for the third attribute, *test-3* (which is numeric). That is, we must compute $d_{ij}^{(3)}$. Following the case for numeric attributes, we let $max_h x_h = 64$ and $min_h x_h = 22$. The difference between the two is used in Eq. (2.27) to normalize the values of the dissimilarity matrix. The resulting dissimilarity matrix for *test-3* is

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

We can now use the dissimilarity matrices for the three attributes in our computation of Eq. (2.27). The indicator $\delta_{ij}^{(f)} = 1$ for each of the three attributes, f. We get, for example, $d(3, 1) = \frac{1(1)+1(0.50)+1(0.45)}{3} = 0.65$. The resulting dissimilarity matrix obtained for the data described by the three attributes of mixed types is:

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

From Table 2.4, we can intuitively guess that objects 1 and 4 are the most similar, based on their values for *test-1* and *test-2*. This is confirmed by the dissimilarity matrix, where $d(4, 1)$ is the lowest value for any pair of different objects. Similarly, the matrix indicates that objects 1 and 2 are the least similar. □

Cosine Similarity

For Text & High Dimensions

Measures the **orientation** of vectors, not the magnitude.

Angle θ close to 0, Cos(θ) close to 1 **(Similar)**

Angle θ close to 90, Cos(θ) close to 0 **(Orthogonal)**

Angle θ close to 180, Cos(θ) close to -1 **(Opposite)**

$$\cos(\theta) = \frac{\boldsymbol{A} \cdot \boldsymbol{B}}{\|\boldsymbol{A}\| \, \|\boldsymbol{B}\|}$$

- Range: -1 (Opposite) to 1 (Identical).
- 0 indicates orthogonality (no shared terms).
- Common in document clustering and search engines.

```
Doc1: [2, 0, 1, 3]  (counts of: data, mining, science, analysis)
Doc2: [1, 2, 0, 2]

cosθ = (2*1 + 0*2 + 1*0 + 3*2) / (√14 * √9)

     = 8 / (3.74 * 3) = 0.713
```

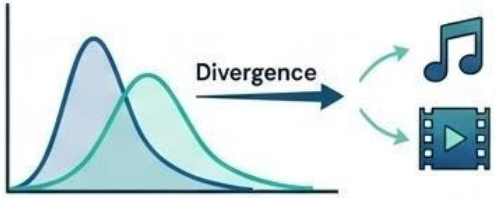
Example 2.24. Cosine similarity between two term-frequency vectors. Suppose that $\boldsymbol{x}$ and $\boldsymbol{y}$ are the first two term-frequency vectors in Table 2.7. That is, $\boldsymbol{x} = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)$ and $\boldsymbol{y} = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)$. How similar are $\boldsymbol{x}$ and $\boldsymbol{y}$? Using Eq. (2.28) to compute the cosine similarity between the two vectors, we get:

$$\begin{aligned}
\boldsymbol{x} \cdot \boldsymbol{y} &= 5 \times 3 + 0 \times 0 + 3 \times 2 + 0 \times 0 + 2 \times 1 + 0 \times 1 + 0 \times 0 + 2 \times 1 \\
&\quad + 0 \times 0 + 0 \times 1 = 25 \\
||\boldsymbol{x}|| &= \sqrt{5^2 + 0^2 + 3^2 + 0^2 + 2^2 + 0^2 + 0^2 + 2^2 + 0^2 + 0^2} = 6.48 \\
||\boldsymbol{y}|| &= \sqrt{3^2 + 0^2 + 2^2 + 0^2 + 1^2 + 1^2 + 0^2 + 1^2 + 0^2 + 1^2} = 4.12 \\
sim(\boldsymbol{x}, \boldsymbol{y}) &= 0.94.
\end{aligned}$$

Therefore if we were using the cosine similarity measure to compare these documents, they would be considered quite similar. □

Advanced Measures & Semantics

KL Divergence

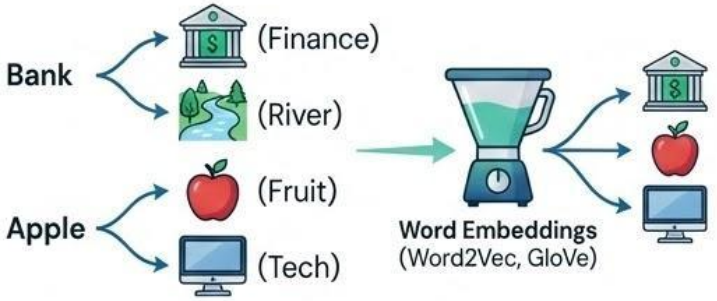


Measures how one probability distribution diverges from another.

Used to compare user preference distributions (e.g., Genre preferences).

$$D_{KL}(P \,||\, Q) = \sum P(i) \log \frac{P(i)}{Q(i)}$$

Hidden Semantics



Mathematical similarity ≠ Semantic similarity.

Solution: Use Word Embeddings to capture context.

- ✓ **Polysemy:** 'Bank' (Finance) vs 'Bank' (River).
- ✓ **Context:** 'Apple' (Fruit) vs 'Apple' (Tech).

```
P = [0.5, 0.3, 0.2]
Q = [0.4, 0.4, 0.2]

D_KL = 0.5*log(0.5/0.4) + 0.3*log(0.3/0.4) + 0.2*log(0.2/0.2)

     = 0.5*0.223 + 0.3*(-0.288) + 0 = 0.028
```

Example 2.25. Computing the KL divergence by smoothing. Suppose there are two sample distributions P and Q as follows: $P : (a : 3/5, b : 1/5, c : 1/5)$ and $Q : (a : 5/9, b : 3/9, d : 1/9)$. To compute the KL divergence $D_{KL}(P||Q)$, we introduce a small constant ϵ, for example $\epsilon = 10^{-3}$, and define a smoothed version of P and Q, P' and Q', as follows.

The sample set observed in P, $SP = \{a, b, c\}$. Similarly, $SQ = \{a, b, d\}$. The union set is $SU = \{a, b, c, d\}$. By smoothing, the missing symbols can be added to each distribution accordingly, with the small probability ϵ. Thus we have $P' : (a : 3/5 - \epsilon/3, b : 1/5 - \epsilon/3, c : 1/5 - \epsilon/3, d : \epsilon)$ and $Q' : (a : 5/9 - \epsilon/3, b : 3/9 - \epsilon/3, c : \epsilon, d : 1/9 - \epsilon/3)$. $D_{KL}(P', Q')$ can be computed easily. □

Data Quality & Cleaning

Garbage In = Garbage Out

Key Quality Dimensions

The four pillars of high-quality data.

Accuracy

Correctness & Precision.

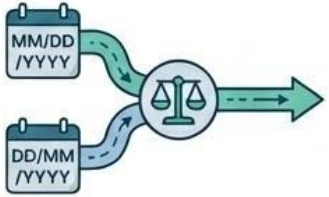
Issue: Age = 150

Completeness

Missing values.

Issue: Null Address

Consistency



Uniform format.

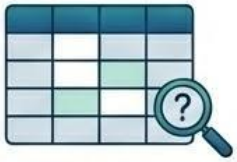
Issue: MM/DD vs DD/MM

Timeliness

Up-to-dateness.

Issue: 2010 Census data

Data Cleaning Techniques

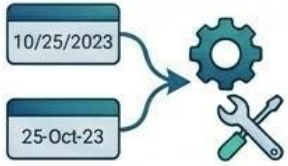


Missing Data

- ✓ Ignore tuple (if minimal).
- ✓ Impute (Mean/Median/Mode).
- ✓ Predict using Regression/ML.

Noisy Data

- ✓ Binning, Regression, or Clustering to remove outliers.



Inconsistent Data

- ✓ Standardize formats and resolve redundancy using domain rules.

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

FIGURE 2.11

Data smoothing with different binning methods.

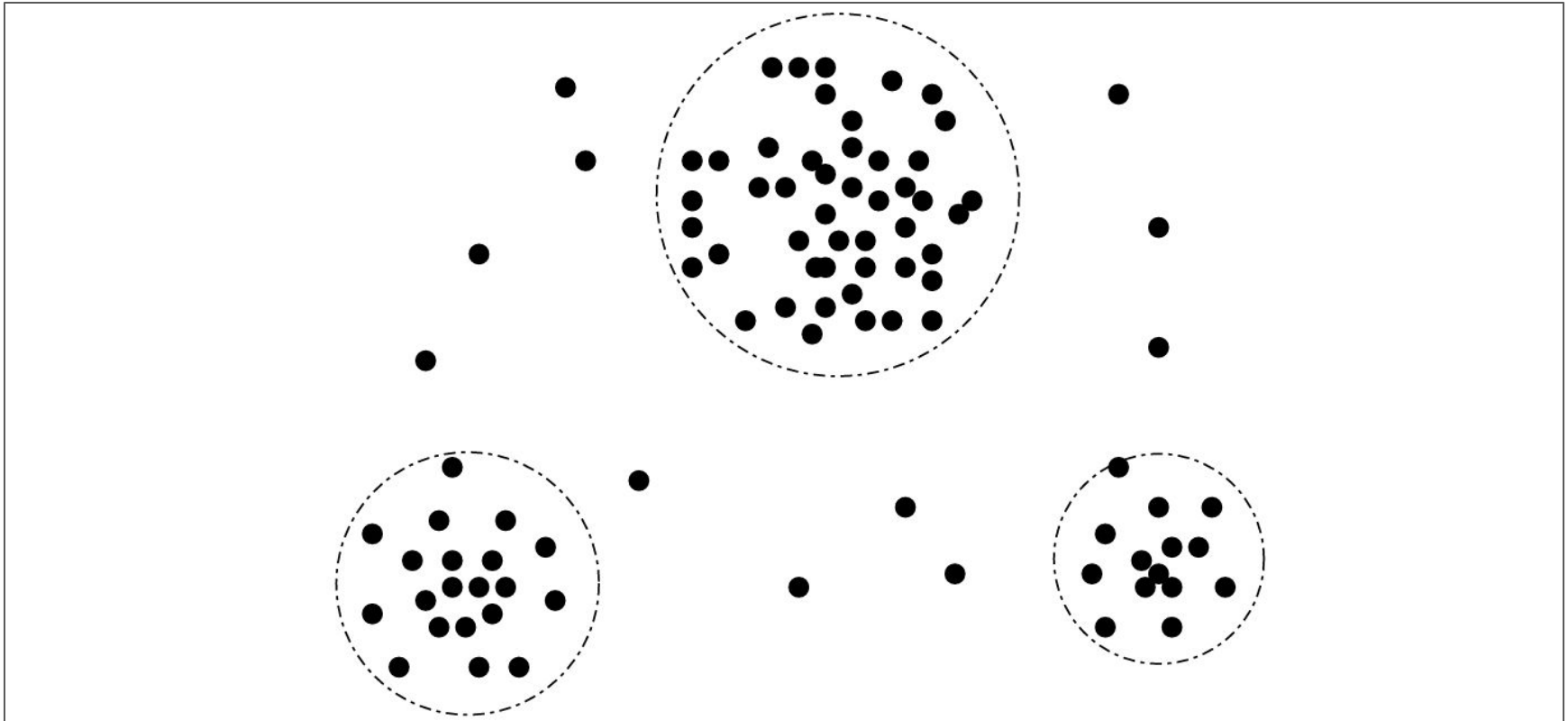


FIGURE 2.12

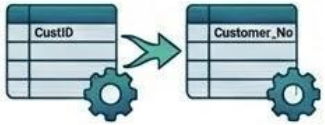
A 2-D customer data plot with respect to customer locations in a city, showing three data clusters. Outliers may be detected as values that fall outside of the cluster sets.

```
Input: "123 Main St., NY, 10001"
        "123 Main Street, New York, NY 10001"
        "123 Main St. Apt 4B, NYC 10001-1234"
```

Data Integration

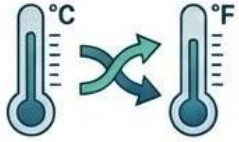
The Challenge

Schema Integration

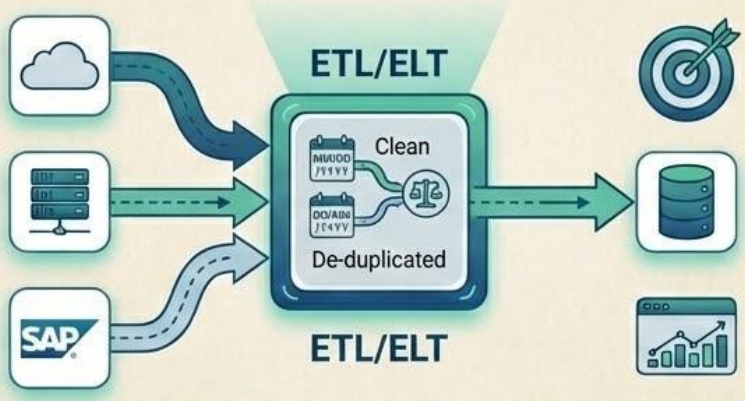


Different names (CustID vs Customer_No).

Value Conflicts



Celsius vs Fahrenheit.

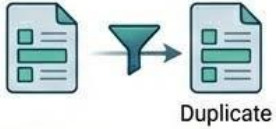


Entity Identification

Bill Gates, DB1 William Gates, DB2

Is 'Bill Gates' in DB1 the same as 'William Gates' in DB2?

Redundancy

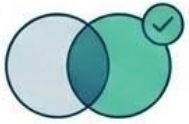


Correlation analysis required to avoid duplicate weight.

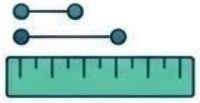
Key Takeaways

Similarity Measures

- ✓ Nominal: Simple Matching.



- ✓ Binary: Jaccard (Asymmetric).



- ✓ Numeric: Minkowski (Normalize first!).

- ✓ Text: Cosine Similarity.

Quality & Integration

- ✓ Cleaning: Essential step before mining to handle missing or noisy data.

- ✓ Integration: Combined data yields greater insights but requires careful schema matching.

- ✓ Garbage In = Garbage Out.

- ✓ Essential step before mining.