| | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **P1** | 0 | 1.12 | 2.12 | 3.91 | 6.02 | 5.59 | 5.70 | 5.41 | 4.03 | 1.58 | 2.06 | 3.16 |
| **P2** | 1.12 | 0 | 1.12 | 2.83 | 5 | 4.47 | 4.61 | 4.47 | 3.16 | 2.06 | 1.41 | 2.50 |
| **P3** | 2.12 | 1.12 | 0 | 1.80 | 3.91 | 3.64 | 3.61 | 3.35 | 3.20 | 3.16 | 2.06 | 2.92 |
| **P4** | 3.91 | 2.83 | 1.80 | 0 | 2.24 | 2 | 1.80 | 2 | 3.16 | 4.72 | 3.16 | 3.50 |
| **P5** | 6.02 | 5 | 3.91 | 2.24 | 0 | 2.24 | 1.12 | 1 | 5 | 6.95 | 5.39 | 5.59 |
| **P6** | 5.59 | 4.47 | 3.64 | 2 | 2.24 | 0 | 1.12 | 2.83 | 3.16 | 6.92 | 4.24 | 4.03 |
| **P7** | 5.70 | 4.61 | 3.61 | 1.80 | 1.12 | 1.12 | 0 | 1.80 | 4.03 | 6.40 | 4.72 | 4.74 |
| **P8** | 5.41 | 4.47 | 3.35 | 2 | 1 | 2.83 | 1.80 | 0 | 5.10 | 6.50 | 5.10 | 5.50 |
| **P9** | 4.03 | 3.16 | 3.20 | 3.16 | 5 | 3.16 | 4.03 | 5.10 | 0 | 3.64 | 2 | 1.12 |
| **P10** | 1.58 | 2.06 | 3.16 | 4.72 | 6.95 | 6.92 | 6.40 | 6.50 | 3.64 | 0 | 1.80 | 2.55 |
| **P11** | 2.06 | 1.41 | 2.06 | 3.16 | 5.39 | 4.24 | 4.72 | 5.10 | 2 | 1.80 | 0 | 1.12 |
| **P12** | 3.16 | 2.50 | 2.92 | 3.50 | 5.59 | 4.03 | 4.74 | 5.50 | 1.12 | 2.55 | 1.12 | 0 |

## 1. How DBSCAN works

**DBSCAN = Density-Based Spatial Clustering of Applications with Noise.**

The central idea is very simple:

> **A cluster is a region where points are sufficiently dense, while isolated points are treated as noise.**

Unlike K-Means, DBSCAN does **not** require us to specify the number of clusters beforehand.

It has two important parameters:

### Parameter 1 — $\epsilon$

$$\epsilon = \text{maximum distance for two points to be considered neighbors}$$

Here:

$$\boxed{\epsilon = 1.9}$$

So if

$$d(P_i, P_j) \leq 1.9$$

then $P_i$ and $P_j$ are neighbors.

$\epsilon = 1.9.$

### Parameter 2 — MinPts

$$\boxed{\text{MinPts} = 4}$$

A point is a **core point** when its ε-neighborhood contains at least 4 points.

**Important:** the point itself is counted.

Thus:

$$|N_\epsilon(P_i)| \geq 4$$

means $P_i$ is a core point.

---

## 2. The three types of DBSCAN points

### A. Core point

A point is **core** if:

$$|N_\epsilon(P)| \geq \text{MinPts}$$

Here:

$$|N_\epsilon(P)| \geq 4$$

Example:

$$N_\epsilon(P_2) = \{P_1, P_2, P_3, P_{11}\}$$

Therefore:

$$|N_\epsilon(P_2)| = 4$$

so:

$$\boxed{P_2 = \text{Core}}$$

### B. Border point

A point is **border** if:

1. It is **not** a core point.
2. It lies within ε of a core point.

For example, if:

$$d(P_1, P_2) = 1.12$$

and $P_2$ is core, then $P_1$ is a border point.

### C. Noise point

A point is noise if:

- it is not core, and
- it is not within ε of any core point.

Therefore:

$$\boxed{\text{Noise} = \text{Non-core point not density-connected to a core}}$$

## 3. The DBSCAN algorithm — conceptually

For every point:

### Step 1

Calculate distances to other points.

### Step 2

Find all points satisfying:

$$d(P_i, P_j) \leq \epsilon$$

### Step 3

Count the ε-neighborhood including the point itself.

### Step 4

$$
\begin{cases}
|N_\epsilon(P_i)| \geq 4 & \Rightarrow \text{Core} \\
|N_\epsilon(P_i)| < 4, \ \text{but adjacent to core} & \Rightarrow \text{Border} \\
\text{otherwise} & \Rightarrow \text{Noise}
\end{cases}
$$

### Step 5

Start a cluster from a core point.

### Step 6

Expand the cluster through **density-connected core points** and their border points.

This last step is extremely important. **DBSCAN does not simply put every point that is geographically close into one cluster.** Cluster expansion occurs through the density structure.

---

## 4. Data

contains 12 points:

$$P_1, P_2, \ldots, P_{12}$$

| | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **P1** | 0 | 1.12 | 2.12 | 3.91 | 6.02 | 5.59 | 5.70 | 5.41 | 4.03 | 1.58 | 2.06 | 3.16 |
| **P2** | 1.12 | 0 | 1.12 | 2.83 | 5 | 4.47 | 4.61 | 4.47 | 3.16 | 2.06 | 1.41 | 2.50 |
| **P3** | 2.12 | 1.12 | 0 | 1.80 | 3.91 | 3.64 | 3.61 | 3.35 | 3.20 | 3.16 | 2.06 | 2.92 |
| **P4** | 3.91 | 2.83 | 1.80 | 0 | 2.24 | 2 | 1.80 | 2 | 3.16 | 4.72 | 3.16 | 3.50 |
| **P5** | 6.02 | 5 | 3.91 | 2.24 | 0 | 2.24 | 1.12 | 1 | 5 | 6.95 | 5.39 | 5.59 |
| **P6** | 5.59 | 4.47 | 3.64 | 2 | 2.24 | 0 | 1.12 | 2.83 | 3.16 | 6.92 | 4.24 | 4.03 |
| **P7** | 5.70 | 4.61 | 3.61 | 1.80 | 1.12 | 1.12 | 0 | 1.80 | 4.03 | 6.40 | 4.72 | 4.74 |
| **P8** | 5.41 | 4.47 | 3.35 | 2 | 1 | 2.83 | 1.80 | 0 | 5.10 | 6.50 | 5.10 | 5.50 |
| **P9** | 4.03 | 3.16 | 3.20 | 3.16 | 5 | 3.16 | 4.03 | 5.10 | 0 | 3.64 | 2 | 1.12 |
| **P10** | 1.58 | 2.06 | 3.16 | 4.72 | 6.95 | 6.92 | 6.40 | 6.50 | 3.64 | 0 | 1.80 | 2.55 |
| **P11** | 2.06 | 1.41 | 2.06 | 3.16 | 5.39 | 4.24 | 4.72 | 5.10 | 2 | 1.80 | 0 | 1.12 |
| **P12** | 3.16 | 2.50 | 2.92 | 3.50 | 5.59 | 4.03 | 4.74 | 5.50 | 1.12 | 2.55 | 1.12 | 0 |

## 5. How to read this matrix

We only care about:

$$\boxed{d \le 1.9}$$

For example, for $P_2$:

$$d(P_2, P_1) = 1.12$$

$$d(P_2, P_3) = 1.12$$

$$d(P_2, P_{11}) = 1.41$$

All three satisfy:

$$d \leq 1.9$$

Therefore:

$$N_\epsilon(P_2) = \{P_1, P_2, P_3, P_{11}\}$$

Notice that $P_2$ **itself is included**.

Hence:

$$|N_\epsilon(P_2)| = 4$$

and therefore $P_2$ is core.

## 6. Complete ε-neighborhood calculations

Now we perform the actual DBSCAN calculation point by point.

### P1

Distances ≤ 1.9:

$$P1 \to P2 = 1.12$$

$$P1 \to P10 = 1.58$$

Including itself:

$$N_\epsilon(P_1) = \{P_1, P_2, P_{10}\}$$

Therefore:

$$|N_\epsilon(P_1)| = 3$$

$$3 < 4$$

So:

$$\boxed{P_1 \text{ is not core}}$$

---

### P2

$$P2 \to P1 = 1.12$$

$$P2 \to P3 = 1.12$$

$$P2 \to P11 = 1.41$$

Therefore:

$$N_\epsilon(P_2) = \{P_1, P_2, P_3, P_{11}\}$$

Count:

$$4$$

Thus:

$$\boxed{P_2 = \text{Core}}$$

---

### P3

$$P3 \to P2 = 1.12$$

$$P3 \to P4 = 1.80$$

Including itself:

$$N_\epsilon(P_3) = \{P_2, P_3, P_4\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_3 = \text{Non-core}}$$

---

### P4

$$P4 \to P3 = 1.80$$

$$P4 \to P7 = 1.80$$

Including itself:

$$N_\epsilon(P_4) = \{P_3, P_4, P_7\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_4 = \text{Non-core}}$$

---

### P5

$$P5 \to P7 = 1.12$$

$$P5 \to P8 = 1.00$$

Including itself:

$$N_\epsilon(P_5) = \{P_5, P_7, P_8\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_5 = \text{Non-core}}$$

---

## P6

From the **actual displayed matrix**:

$$d(P_6, P_5) = 2.24$$

$$d(P_6, P_7) = 1.12$$

$$d(P_6, P_8) = 2.83$$

Only $P_7$ is within ε.

Therefore:

$$N_\epsilon(P_6) = \{P_6, P_7\}$$

Count:

$$2$$

Hence:

$$\boxed{P_6 = \text{Non-core}}$$

$$2.24 > 1.9$$

and

$$2.83 > 1.9$$

Therefore $P_5$ and $P_8$ cannot be ε-neighbors of $P_6$.

---

## 7. P7

From the matrix:

$$d(P_7, P_4) = 1.80$$

$$d(P_7, P_5) = 1.12$$

$$d(P_7, P_6) = 1.12$$

$$d(P_7, P_8) = 1.80$$

Including itself:

$$N_\epsilon(P_7) = \{P_4, P_5, P_6, P_7, P_8\}$$

Therefore:

$$|N_\epsilon(P_7)| = 5$$

Since:

$$5 \geq 4$$

we obtain:

$$\boxed{P_7 = \text{Core}}$$

---

## 8. P8

$$d(P_8, P_5) = 1.00$$

$$d(P_8, P_7) = 1.80$$

Including itself:

$$N_\epsilon(P_8) = \{P_5, P_7, P_8\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_8 = \text{Non-core}}$$

---

## 9. P9

$$d(P_9, P_{12}) = 1.12$$

Including itself:

$$N_\epsilon(P_9) = \{P_9, P_{12}\}$$

Count:

$$2 < 4$$

So P9 is not core.

But we must wait until we know whether P12 is connected to a core point.

---

## 10. P10

$$d(P_{10}, P_1) = 1.58$$

$$d(P_{10}, P_{11}) = 1.80$$

Including itself:

$$N_\epsilon(P_{10}) = \{P_1, P_{10}, P_{11}\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_{10} = \text{Non-core}}$$

---

## 11. P11

$$d(P_{11}, P_2) = 1.41$$

$$d(P_{11}, P_{10}) = 1.80$$

$$d(P_{11}, P_{12}) = 1.12$$

Including itself:

$$N_\epsilon(P_{11}) = \{P_2, P_{10}, P_{11}, P_{12}\}$$

Count:

$$4$$

Therefore:

$$\boxed{P_{11} = \text{Core}}$$

---

## 12. P12

$$d(P_{12}, P_9) = 1.12$$

$$d(P_{12}, P_{11}) = 1.12$$

Including itself:

$$N_\epsilon(P_{12}) = \{P_9, P_{11}, P_{12}\}$$

Count:

$$3 < 4$$

Therefore:

$$\boxed{P_{12} = \text{Non-core}}$$

---

## 13. Complete calculation table

This is the most important table for an examination.

| Point | ε-neighbors excluding itself | Self included? | Neighborhood count | Core? | Border? | Noise? |
|---|---|---|---|---|---|---|
| **P1** | P2, P10 | Yes | 3 | No | **Yes** | No |
| **P2** | P1, P3, P11 | Yes | 4 | **Yes** | No | No |
| **P3** | P2, P4 | Yes | 3 | No | **Yes** | No |
| **P4** | P3, P7 | Yes | 3 | No | **Yes** | No |
| **P5** | P7, P8 | Yes | 3 | No | **Yes** | No |
| **P6** | P7 | Yes | 2 | No | **Yes** | No |

| Point | ε-neighbors excluding itself | Self included? | Neighborhood count | Core? | Border? | Noise? |
|---|---|---|---|---|---|---|
| **P7** | P4, P5, P6, P8 | Yes | 5 | **Yes** | No | No |
| **P8** | P5, P7 | Yes | 3 | No | **Yes** | No |
| **P9** | P12 | Yes | 2 | No | No | **Yes** |
| **P10** | P1, P11 | Yes | 3 | No | **Yes** | No |
| **P11** | P2, P10, P12 | Yes | 4 | **Yes** | No | No |
| **P12** | P9, P11 | Yes | 3 | No | **Yes** | No |

So the **core points are**:

$$\boxed{P_2,\ P_7,\ P_{11}}$$

The **border points are**:

$$\boxed{P_1, P_3, P_4, P_5, P_6, P_8, P_{10}, P_{12}}$$

The **noise point is**:

$$\boxed{P_9}$$

---

## 14. Now the most important part: cluster formation

Finding core points is **not the end of DBSCAN**.

We must determine which core points belong to the same density-connected cluster.

The core points are:

$$P_2, \quad P_7, \quad P_{11}$$

Now examine distances between them.

### P2 and P7

$$d(P_2, P_7) = 4.61$$

Therefore:

$$4.61 > 1.9$$

They are **not directly density-connected**.

---

### P2 and P11

$$d(P_2, P_{11}) = 1.41$$

Therefore:

$$1.41 < 1.9$$

So:

$$\boxed{P_2 \leftrightarrow P_{11}}$$

They belong to the same cluster.

---

### P7 and P11

$$d(P_7, P_{11}) = 4.72$$

Therefore:

$$4.72 > 1.9$$

They are not directly connected.

There is also no chain of core points connecting P7 to P2/P11.

Therefore:

$$\boxed{\{P_2, P_{11}\}}$$

and

$$\boxed{\{P_7\}}$$

are two separate core regions.

---

## 15. Cluster 1

Start with core point:

$$P_2$$

Its ε-neighborhood:

$$\{P_1, P_2, P_3, P_{11}\}$$

P11 is itself core.

Expand P11:

$$N_\epsilon(P_{11}) = \{P_2, P_{10}, P_{11}, P_{12}\}$$

Therefore cluster 1 becomes:

$$\boxed{C_1 = \{P_1, P_2, P_3, P_{10}, P_{11}, P_{12}\}}$$

Core:

$$P_2, P_{11}$$

Border:

$$P_1, P_3, P_{10}, P_{12}$$

---

## 16. Cluster 2

Start with core:

$$P_7$$

Its neighborhood:

$$N_\epsilon(P_7) = \{P_4, P_5, P_6, P_7, P_8\}$$

P7 is the only core point in this neighborhood.

Therefore:

$$\boxed{C_2 = \{P_4, P_5, P_6, P_7, P_8\}}$$

Core:

$$P_7$$

Border:

$$P_4, P_5, P_6, P_8$$

---

## 17. What happens to P9?

P9 has:

$$N_\epsilon(P_9) = \{P_9, P_{12}\}$$

P12 is **not core**.

Therefore P9 is not connected to a core point.

Hence:

$$\boxed{P_9 = \text{Noise}}$$

This agrees with the PDF's conclusion that P9 is the genuine noise/outlier point.

---

## 18. Final DBSCAN result

Using the **distance matrix actually shown in the PDF**:

| Cluster | Core points | Border points | Noise |
|---|---|---|---|
| **Cluster 1** | P2, P11 | P1, P3, P10, P12 | — |
| **Cluster 2** | P7 | P4, P5, P6, P8 | — |
| **Noise** | — | — | **P9** |

Therefore:

$$\boxed{C_1 = \{P_1, P_2, P_3, P_{10}, P_{11}, P_{12}\}}$$

$$\boxed{C_2 = \{P_4, P_5, P_6, P_7, P_8\}}$$

$$\boxed{Noise = \{P_9\}}$$

---

## 19. Very important correction to the supplied PDF

The PDF's written solution states:

## 20. Why this distinction is extremely important

A common student mistake is:

> "If A is close to B and B is close to C, then A, B and C must be in the same DBSCAN cluster."

Not necessarily.

DBSCAN uses **density connectivity**, not merely ordinary geometric connectivity.

Consider:

$$P_2 \to P_1$$

and

$$P_1 \to P_3$$

Even though P1 is not core, it does not create a new dense expansion path.

The important expansion mechanism is through **core points**.

This is why understanding:

- ε-neighborhood
- core point
- directly density-reachable
- density-reachable

- density-connected

is much more important than simply memorizing "DBSCAN finds clusters."

---

## 21. The mathematical DBSCAN terminology you should know

### Directly density-reachable

$q$ is directly density-reachable from $p$ if:

$$q \in N_\epsilon(p)$$

and $p$ is core.

---

### Density-reachable

A point can be reached through a sequence:

$$p_1 \to p_2 \to p_3 \to \cdots \to p_n$$

where the relevant intermediate points satisfy the DBSCAN core condition.

---

### Density-connected

Two points $p$ and $q$ are density-connected if there exists some point $o$ such that both $p$ and $q$ are density-reachable from $o$.

This is the deeper mathematical reason DBSCAN forms clusters.

---

## 22. solution

**Given**

$$\epsilon = 1.9$$

$$MinPts = 4$$

Core condition:

$$|N_\epsilon(P)| \geq 4$$

From the distance matrix:

| Point | ε-neighborhood including itself | Count | Classification |
|---|---|---|---|
| P1 | P1,P2,P10 | 3 | Border |
| P2 | P1,P2,P3,P11 | 4 | **Core** |
| P3 | P2,P3,P4 | 3 | Border |
| P4 | P3,P4,P7 | 3 | Border |
| P5 | P5,P7,P8 | 3 | Border |
| P6 | P6,P7 | 2 | Border |
| P7 | P4,P5,P6,P7,P8 | 5 | **Core** |
| P8 | P5,P7,P8 | 3 | Border |
| P9 | P9,P12 | 2 | **Noise** |
| P10 | P1,P10,P11 | 3 | Border |
| P11 | P2,P10,P11,P12 | 4 | **Core** |

| Point | ε-neighborhood including itself | Count | Classification |
|---|---|---|---|
| P12 | P9,P11,P12 | 3 | Border |

Hence:

$$\boxed{\text{Core} = \{P2, P7, P11\}}$$

$$\boxed{\text{Border} = \{P1, P3, P4, P5, P6, P8, P10, P12\}}$$

$$\boxed{\text{Noise} = \{P9\}}$$

Final clusters:

$$\boxed{C_1 = \{P1, P2, P3, P10, P11, P12\}}$$

$$\boxed{C_2 = \{P4, P5, P6, P7, P8\}}$$

---

## 23. important Points

These are much more interesting than ordinary "What is DBSCAN?" questions.

### Q1. Why doesn't DBSCAN require the number of clusters?

Because clusters emerge from **density connectivity**. There is no $K$ parameter like K-Means.

---

### Q2. What happens if ε is too small?

Too many points become:

$$\boxed{\text{noise}}$$

and genuine clusters may fragment.

---

### Q3. What happens if ε is too large?

Separate dense regions can become connected, causing:

$$\boxed{\text{cluster merging}}$$

This is exactly the type of issue visible in density-based clustering.

---

### Q4. What happens if MinPts is increased?

The definition of "dense" becomes stricter.

Consequences can include:

- fewer core points,
- more border/noise points,
- smaller or fewer clusters.

---

### Q5. Why is MinPts counted including the point itself?

Because the DBSCAN neighborhood is conventionally defined as:

$$N_\epsilon(p) = \{q : d(p,q) \le \epsilon\}$$

and $d(p,p) = 0$, so $p \in N_\epsilon(p)$.

This matters enormously in borderline cases.

For example, P2 has exactly:

points including itself, so it qualifies as core.

### Q6. Is DBSCAN deterministic?

**Cluster structure is generally deterministic for fixed data, distance metric, ε and MinPts, but assignment of a border point lying within ε of multiple clusters can depend on processing/order conventions.**

This is an advanced and very good interview question.

---

### Q7. Can DBSCAN find non-spherical clusters?

**Yes.**

This is one of its major advantages over K-Means.

DBSCAN can discover arbitrarily shaped density-connected regions.

---

### Q8. Why can DBSCAN fail on datasets having different densities?

Because a single global ε and MinPts may not work simultaneously for both a dense cluster and a sparse cluster.

For example:

$$\text{Cluster A density} \gg \text{Cluster B density}$$

A single ε may either:

- split Cluster B, or
- merge/saturate Cluster A.

This is one reason algorithms such as **OPTICS** and **HDBSCAN** are important extensions/alternatives.

### Q9. Why should features often be scaled before DBSCAN?

Because DBSCAN depends directly on distances.

Suppose:

$$X \in [0, 100000]$$

while:

$$Y \in [0, 1]$$

Then Euclidean distance may be dominated by $X$.

Therefore preprocessing such as standardization or robust scaling can materially change the resulting clusters.

---

### Q10. Does DBSCAN always use Euclidean distance?

No.

The algorithm requires a distance/similarity measure, and the choice depends on the data.

Examples include:

- Euclidean distance
- Manhattan distance
- cosine distance
- domain-specific distances

The metric must be appropriate for the data geometry.

---

### Q11. Why is DBSCAN particularly useful for anomaly detection?

Because it explicitly identifies observations that do not belong to sufficiently dense regions as:

$$\boxed{\text{noise}}$$

However:

> **DBSCAN noise is not automatically equivalent to a business-defined anomaly.**

That distinction is important in real Data Science.

---

### Q12. Can DBSCAN work well in very high dimensions?

Often not.

In high-dimensional spaces, distances can become less discriminative due to the **curse of dimensionality** and distance concentration.

Therefore dimensionality reduction or an appropriate representation/metric may be required.

---

### Q13. How do you select ε in a real project?

A standard approach is a **k-distance plot**.

For:

$$k \approx MinPts$$

calculate the distance to each point's $k$-th nearest neighbor, sort these distances, and look for an "elbow" or knee.

That region can provide a useful candidate ε.

But it is a heuristic, not a universal optimum.

### Q14. Why is K-Means sometimes better than DBSCAN?

If:

- clusters are approximately spherical,
- noise is limited,
- $K$ is meaningful,
- and the dataset is large,

K-Means may be simpler and computationally attractive.

Algorithm choice should depend on the data-generating structure, not on the fact that one algorithm is theoretically more sophisticated.

---

### Q15. Why is DBSCAN potentially computationally expensive?

The algorithm needs neighborhood queries.

With an appropriate spatial index, performance can be substantially improved. Without effective indexing, pairwise distance computations can approach:

$$O(n^2)$$

for $n$ observations.

In high-dimensional spaces, indexing structures can also become less effective.

---

## 24. Important Points

### 1. "Why can a border point belong to a cluster even though it isn't dense enough to be a core point?"

Because cluster membership is based on **density reachability from a core region**, not on every point independently satisfying the MinPts criterion.

---

### 2. "Why doesn't ordinary graph connectivity completely explain DBSCAN?"

Because DBSCAN's connectivity is constrained by **core-point density conditions**. Merely having pairwise edges between arbitrary non-core points does not create unlimited cluster expansion.

---

### 3. "What happens when two core points are within ε?"

They are directly density-connected and therefore belong to the same DBSCAN cluster.

In this problem:

$$d(P_2, P_{11}) = 1.41 < 1.9$$

so P2 and P11 belong to the same cluster.

---

### 4. "Why is P6 NOT a core point here despite the supplied solution saying it is?"

Because the actual matrix gives:

$$d(P_6, P_5) = 2.24 > 1.9$$

and:

$$d(P_6, P_8) = 2.83 > 1.9$$

Only P7 lies within ε.

Thus:

$$|N_\epsilon(P_6)| = 2$$

not 4.

This is exactly the kind of **data-validation/error-detection question** a strong Data Scientist should catch.

---

### 5. "What is the biggest practical weakness of standard DBSCAN?"

A single global pair:

$$(\epsilon, MinPts)$$

can struggle when the dataset contains **clusters with substantially different densities**.

That is one of the key reasons more advanced density-based methods such as **OPTICS and HDBSCAN** are important.

---

## Final takeaway

For this particular dataset:

$$\boxed{\epsilon = 1.9, \quad MinPts = 4}$$

Using the **actual distance matrix** the correct result is:

$$\boxed{\text{Core} = \{P2, P7, P11\}}$$

$$\boxed{\text{Cluster 1} = \{P1, P2, P3, P10, P11, P12\}}$$

$$\boxed{\text{Cluster 2} = \{P4, P5, P6, P7, P8\}}$$

$$\boxed{\text{Noise} = \{P9\}}$$