



## Knowledge Discovery in Databases with Exercises Summer Semester 2025

# Exercise Sheet 5: Clustering

### About this Exercise Sheet

This exercise sheet focuses on the content of lecture 8. *Clustering*.

It includes both theoretical exercises on K-means (Exercise 1) and DBSCAN (Exercise 2) and a practical data science exercise (Exercise 3).

The exercise sheet is designed for a two-week period, during which the tasks can be completed flexibly.

The sample solution will be published after the two weeks have elapsed.

### Preparation

Before participating in the exercise, you must prepare the following:

#### 1. Install Python and pip on your computer

- Detailed instructions can be found in `1-Introduction-Python-Pandas.pdf`.

#### 2. Download provided additional files

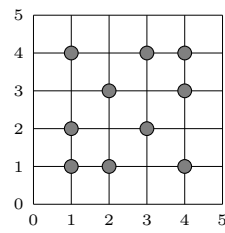
- Download `Additional-Files-Student.zip` from StudOn
- Extract it to a folder of your choice.

#### 3. Install required Python packages

- Open a terminal and navigate to the folder where you extracted the files.
- Run the command `pip install -r requirements.txt` within the extracted additional files folder to install the required Python packages.

## Exercise 1: K-means

Given is a set of points in a two-dimensional space:



Points:

● (1,1), (1,2), (1,4), (2,1), (2,3),  
(3,2), (3,4), (4,1), (4,3), (4,4)

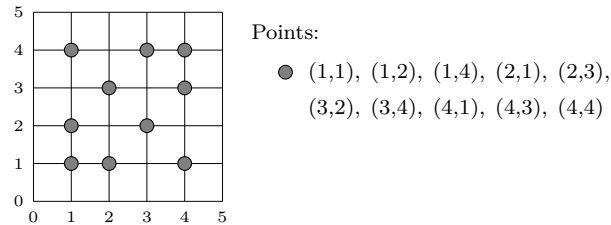
Use **K-means** to cluster the given points into three clusters. Use the **Euclidean distance** as the metric defining the similarity between points.

Write down **all** intermediate steps.

## Exercise 2: DBSCAN

### Task 1: Basic Terms

Given is a set of points in a two-dimensional space:



#### Task 1.1: Core Points

Determine whether (1,1), (2,1), (2,3), and (1,4) are **core points** if a density based clustering algorithm like **DBSCAN** is initialized with  $\varepsilon = 1$  and  $MinPts = 2$  and applied on the given point set. The distance is calculated using the Euclidean distance.

#### Task 1.2: Direct Density Reachability

Determine which of the points in the point set are **directly density reachable** from the core point (1,2) if a density based clustering algorithm like **DBSCAN** is initialized with  $\varepsilon = 1$  and  $MinPts = 2$ . The distance is calculated using the Euclidean distance.

#### Task 1.3: Density Reachability

##### Task 1.3.1: Basic Density Reachability

Determine whether (1,1), (2,1), (2,3), and (4,4) are **density reachable** from the core point (1,2) if a density based clustering algorithm like **DBSCAN** is initialized with  $\varepsilon = 1$  and  $MinPts = 2$ . The distance is calculated using the Euclidean distance.

##### Task 1.3.2: Reversal of Density Reachability

Determine whether (3,4) is density reachable from (4,4) and whether (4,4) is density reachable from (3,4) if a density based clustering algorithm like **DBSCAN** is initialized with  $\varepsilon = 1$  and  $MinPts = 3$ . The distance is calculated using the Euclidean distance.

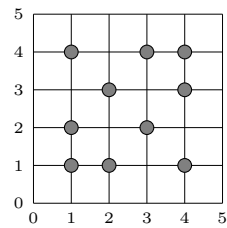
**Be careful:**  $MinPts$  was increased in this task. Thus you have to reevaluate whether points are core points or not.

#### Task 1.4: Density Connectivity

Determine whether (1,1), (3,2), (4,3), and (4,4) are **density connected** to the point (3,4) if a density based clustering algorithm like **DBSCAN** is initialized with  $\varepsilon = 1$  and  $MinPts = 3$ . The distance is calculated using the Euclidean distance.

## Task 2: Application of DBSCAN

Given is a set of points in a two-dimensional space:



Points:

● (1,1), (1,2), (1,4), (2,1), (2,3),  
(3,2), (3,4), (4,1), (4,3), (4,4)

Apply the **DBSCAN** algorithm known from the lecture on the given point set while using  $\varepsilon = 1$  and  $MinPts = 2$ .

Write down **all** intermediate steps.

## Exercise 3: Clustering in Python

This exercise comprises practical data science tasks and thus utilizes a Jupyter Notebook:

1. Open `Clustering-in-Python.ipynb`.
2. Take a look at the tasks (blue boxes) in the notebook and try to solve them.

If you are unfamiliar with how to open a Jupyter Notebook, please refer to Exercise 1 of `1-Introduction-Python-Pandas.pdf`.