

Module 1: Basics and Background

Lecture 2: Advanced Database Queries

Prev topic

Next topic

Next page

Prev page

The Lecture Contains:

Join

- Example of join

Skyline queries

- Example of skyline query

## Module 1: Basics and Background

## Lecture 2: Advanced Database Queries

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

## Join

- **Join**: Given two databases  $D_1, D_2$ , a distance function  $d$  and range  $r$  or number of nearest neighbours  $k$ , return object pairs

$T = (O_i \in D_1, O_j \in D_2)$  such that  $(O_i, O_j)$  satisfy  $r$  or  $k$  conditions

- $r$  and  $k$  conditions are defined similar to range and kNN searches

## Module 1: Basics and Background

## Lecture 2: Advanced Database Queries

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

## Join

- **Join**: Given two databases  $D_1, D_2$ , a distance function  $d$  and range  $r$  or number of nearest neighbours  $k$ , return object pairs

$T = (O_i \in D_1, O_j \in D_2)$  such that  $(O_i, O_j)$  satisfy  $r$  or  $k$  conditions

- $r$  and  $k$  conditions are defined similar to range and kNN searches
- If  $D_1$  and  $D_2$  are same, then the problem becomes **self-join**

## Module 1: Basics and Background

## Lecture 2: Advanced Database Queries

Prev topic

Next topic

Prev page

Next page

## Join

- **Join**: Given two databases  $D_1, D_2$ , a distance function  $d$  and range  $r$  or number of nearest neighbours  $k$ , return object pairs

$T = (O_i \in D_1, O_j \in D_2)$  such that  $(O_i, O_j)$  satisfy  $r$  or  $k$  conditions

- $r$  and  $k$  conditions are defined similar to range and kNN searches
- If  $D_1$  and  $D_2$  are same, then the problem becomes **self-join**
- Problem can be defined over m databases
- A general "distance" function  $d(O_1, O_2, \dots, O_m) \rightarrow \mathcal{R}$  is defined
- Answer set  $T$  contains m-tuplets of the form

$(O_{1i} \in D_1, O_{2j} \in D_2, \dots, O_{mk} \in D_m)$

Prev topic

Next topic

Next page

Prev page

Example of join

| Database 1<br>Objects  | Database 2<br>Objects  |
|------------------------|------------------------|
| O <sub>11</sub> (3, 7) | O <sub>21</sub> (2, 9) |
| O <sub>12</sub> (3, 6) | O <sub>22</sub> (1, 4) |
| O <sub>13</sub> (5, 8) | O <sub>23</sub> (4, 5) |

$d$  is Manhattan distance.

Range search:  $r = 2$

kNN search:  $k = 2$

Example of join

| Database 1<br>Objects  | Database 2<br>Objects  |
|------------------------|------------------------|
| O <sub>11</sub> (3, 7) | O <sub>21</sub> (2, 9) |
| O <sub>12</sub> (3, 6) | O <sub>22</sub> (1, 4) |
| O <sub>13</sub> (5, 8) | O <sub>23</sub> (4, 5) |

| Distances       | O <sub>21</sub> | O <sub>22</sub> | O <sub>23</sub> |
|-----------------|-----------------|-----------------|-----------------|
| O <sub>11</sub> | 3               | 5               | 3               |
| O <sub>12</sub> | 4               | 4               | 2               |
| O <sub>13</sub> | 4               | 8               | 4               |

$d$  is Manhattan distance.

Range search:  $r = 2$

kNN search:  $k = 2$

Range search answer:  $T = \{(O_{12}, O_{23})\}$

kNN search answer:  $T = \{(O_{12}, O_{23}), (O_{11}, O_{21})\}$

## Module 1: Basics and Background

## Lecture 2: Advanced Database Queries

Prev topic

Next topic

Prev page

Next page

## Skyline queries

- **Skyline query**: Given  $D$ , dominating functions  $\succ_j \forall_{j=1 \dots k}$  for each dimension, return all objects  $T = \{O_g\}$  such that

$$\nexists O_h \in D, O_h \succ O_g$$

- Domination of an object by another object is defined by

$$O_{gj} \succ O_{hj} \Leftrightarrow \exists j, O_{gj} \succ O_{hj} \text{ and } \nexists_{1 \neq j}, O_{gj} \succ O_{hj}$$

- Dominating functions for individual dimensions may be  $>, \geq, <, \leq$

[Prev topic](#)  
[Next page](#)

[Next topic](#)

[Prev page](#)

Example of skyline query

| Object         | Dim1 | Dim2 |
|----------------|------|------|
| O <sub>1</sub> | 7    | 3    |
| O <sub>2</sub> | 2    | 2    |
| O <sub>3</sub> | 8    | 5    |
| O <sub>4</sub> | 9    | 2    |
| O <sub>5</sub> | 4    | 1    |
| O <sub>6</sub> | 3    | 4    |

Dominating functions are  $\leq$  for dimension 1 and  $\geq$  for dimension 2.



Example of skyline query

| Object         | Dim1 | Dim2 |
|----------------|------|------|
| O <sub>1</sub> | 7    | 3    |
| O <sub>2</sub> | 2    | 2    |
| O <sub>3</sub> | 8    | 5    |
| O <sub>4</sub> | 9    | 2    |
| O <sub>5</sub> | 4    | 1    |
| O <sub>6</sub> | 3    | 4    |

| Object         | Dim1 | Dim2 | Dominated by   |
|----------------|------|------|--|
| O <sub>1</sub> | 7    | 3    | O <sub>6</sub>   |
| O <sub>2</sub> | 2    | 2    | -  |
| O <sub>3</sub> | 8    | 5    | -  |
| O <sub>4</sub> | 9    | 2    | O <sub>1</sub> ,O <sub>2</sub> ,O <sub>3</sub> ,O <sub>6</sub> |
| O <sub>5</sub> | 4    | 1    | O <sub>2</sub> ,O <sub>6</sub>                                 |
| O <sub>6</sub> | 3    | 4    | -  |

Dominating functions are  $\leq$  for dimension 1 and  $\geq$  for dimension 2.

Skyline query answer:  $T = \{O_2, O_3, O_6\}$



