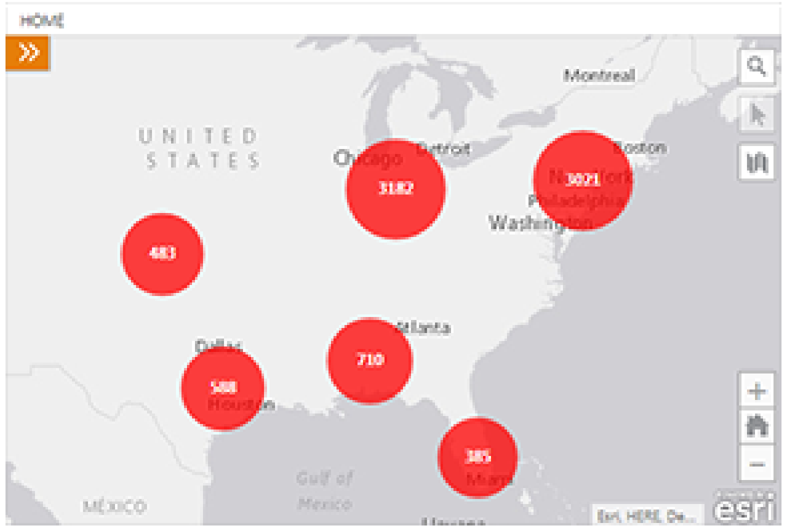
Algorithmen für geographische Informationssysteme

Clustering

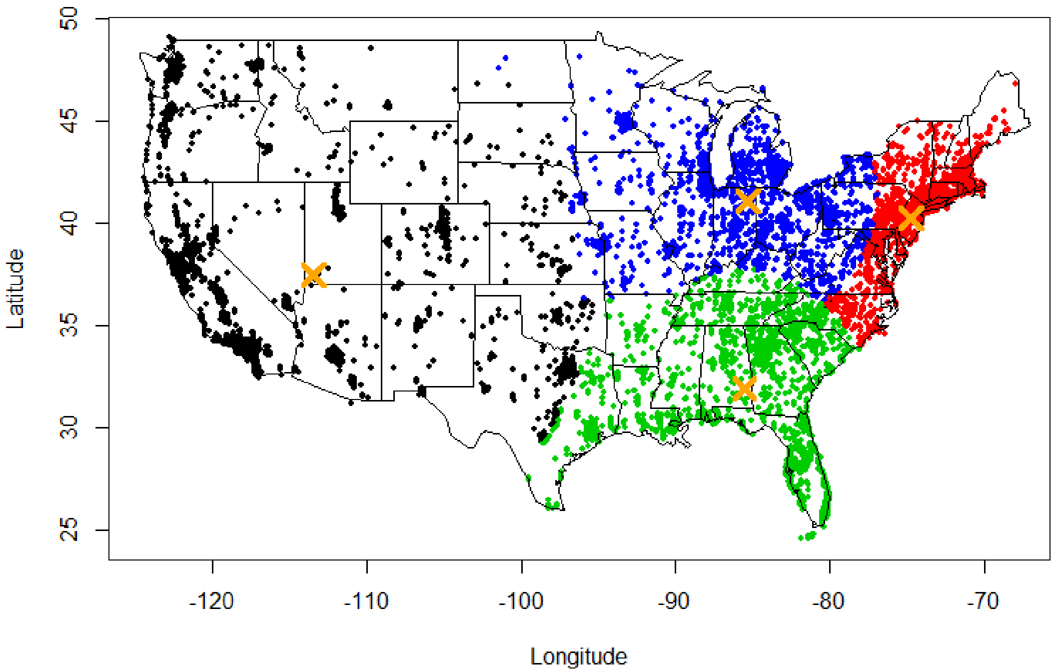
Faster DBSCAN and HDBSCAN in Low-Dimensional Euclidean Spaces

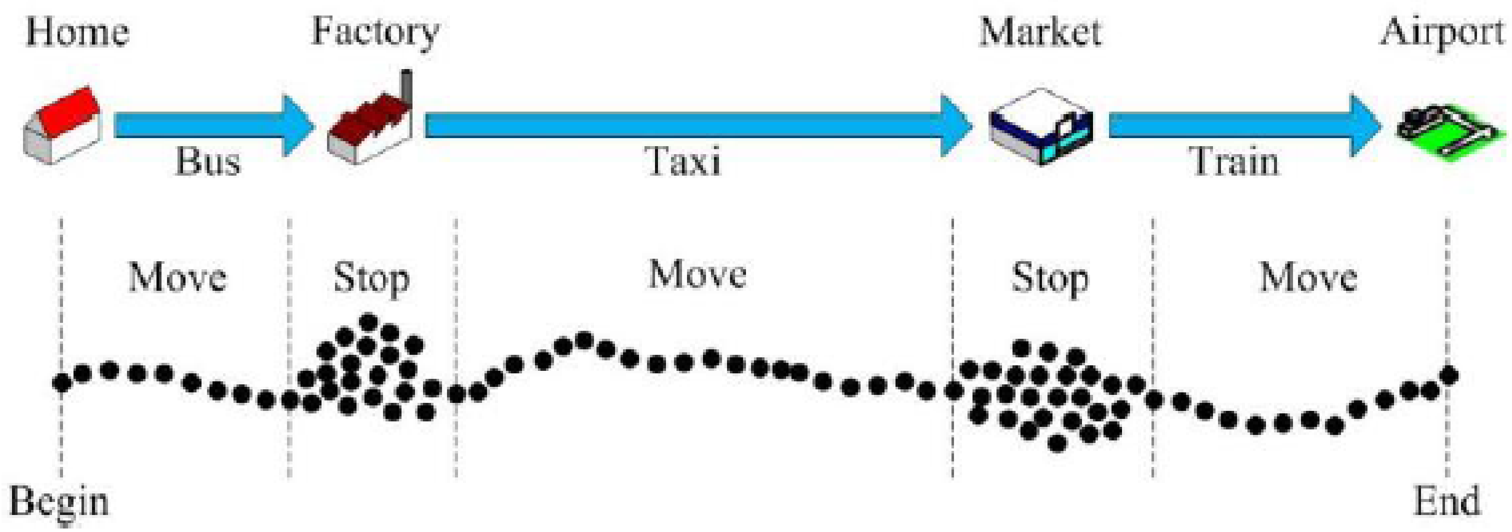
Alexander Wolff

Slides by Thomas van Dijk

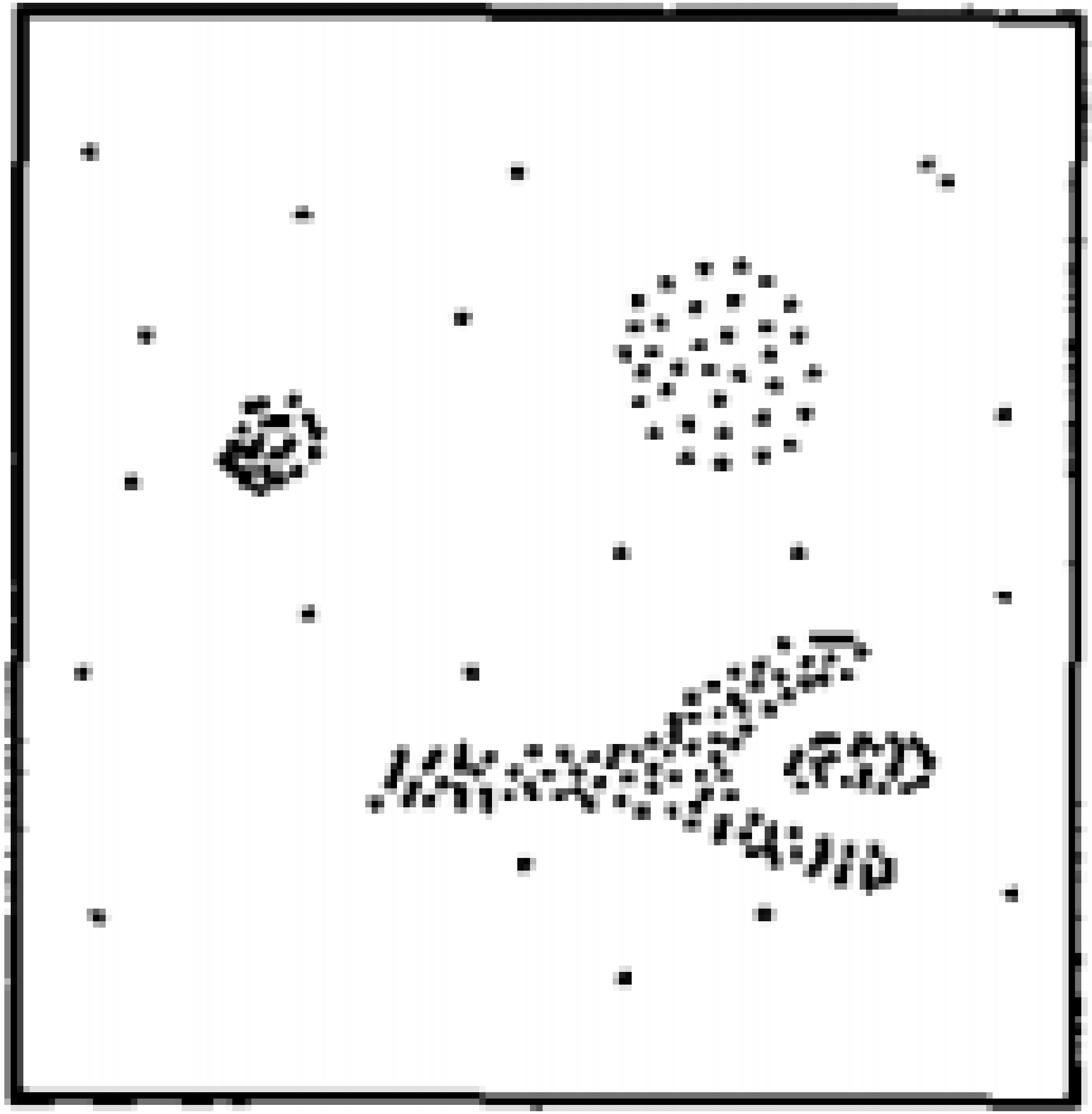

HOME
Montreal
UNITED STATES
Chicago
Detroit
Boston
Washington
Atlanta
Dallas
Houston
3182
3021
483
588
710
385
MÉXICO
esri


HOME
Montreal
UNITED STATES
Chicago
Detroit
Boston
New York
Philadelphia
Washington
Atlanta
Dallas
Houston
Miami
MÉXICO
esri


Latitude
Longitude
50
45
40
35
30
25
-120
-110
-100
-90
-80
-70


Home
Factory
Market
Airport
Bus
Taxi
Train
Move
Stop
Move
Stop
Move
Begin
End

Clustering



database 3

DBSCAN: Objectives

1. “Minimal requirements of domain knowledge to determine the input parameters, because appropriate values are often not known in advance when dealing with large databases.”

2. “Discovery of clusters with arbitrary shape, because the shape of clusters in spatial databases may be spherical, drawn-out, linear, elongated etc.”

3. “Good efficiency on large databases, i.e., on databases of significantly more than just a few thousand objects.”

DBSCAN

Given: data points X, distance function $d(\cdot, \cdot)$, thresholds ε and k.

Def. The **ε-neighborhood** of a point $p \in X$ is

$$N_\varepsilon(p) = \{ q \in X \mid d(p, q) \leqslant \varepsilon \}.$$

Def. A point $p \in X$ is called a **core point** iff $|N_\varepsilon(p)| \geqslant k$.

Def. A point $p \in X$ is **directly density-reachable** from a point q iff:

$$p \in N_\varepsilon(q) \qquad |N_\varepsilon(q)| \geqslant k$$ (q is a core point)

Not a symmetric relation!

Def. A point $p \in X$ is **density reachable** from a point q if there exists a chain of direct density-reachability from q to p.

Def. A point $p \in X$ is **density connected** to a point q if there exists a (core) point r such that both p and q are density-reachable from r.

DBSCAN example

Legend

$k = 3$

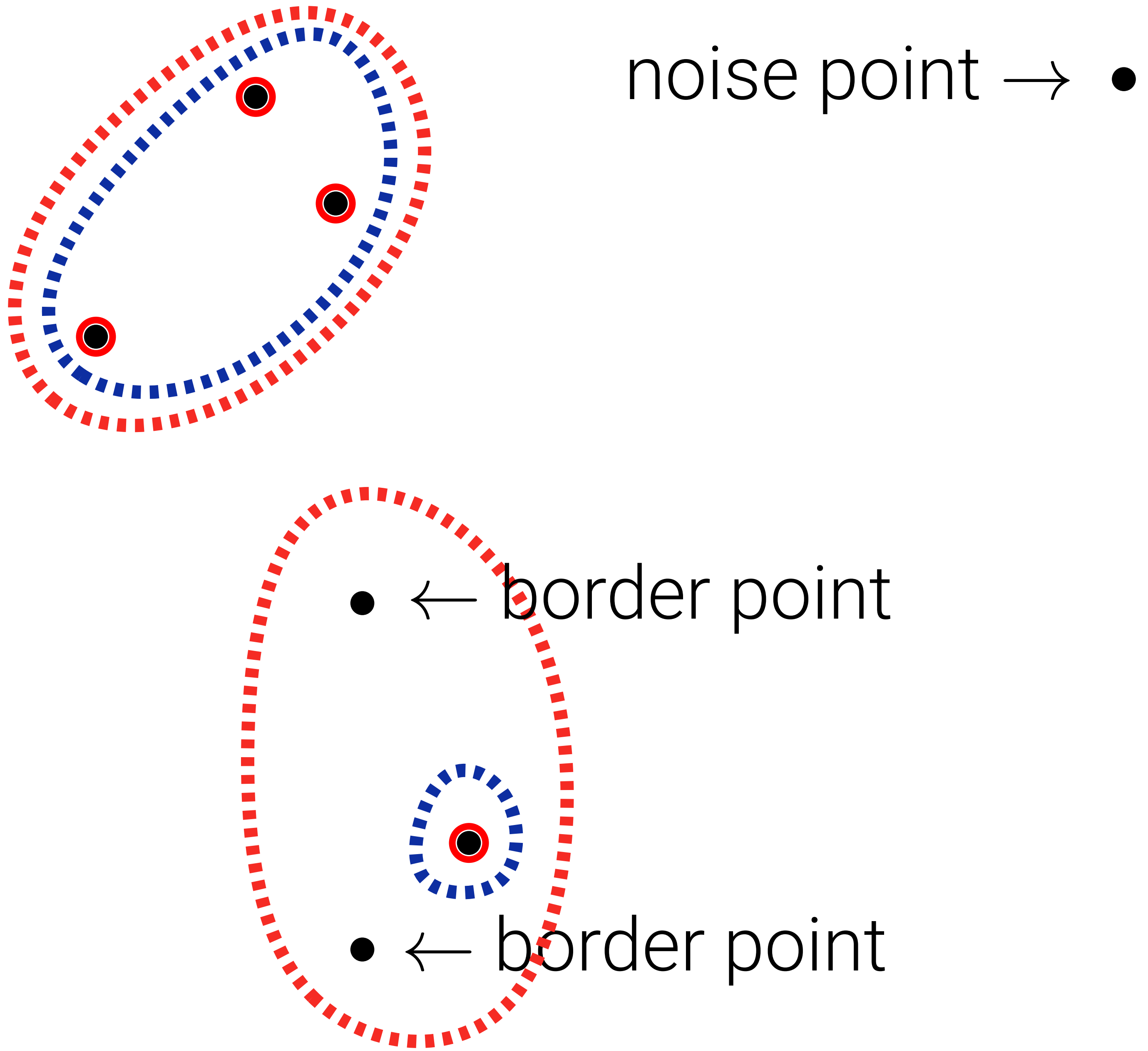
Distance ε

Core points

Density connected

DBSCAN clustering

DBSCAN* clustering



DBSCAN:
DBSCAN*:

$\mathbf{p}$ and $\mathbf{q}$ are in the same cluster $\Leftrightarrow$ $\mathbf{p}$ and $\mathbf{q}$ are density connected (and core pts.)

DBSCAN example

Legend

$k = 3$

Distance ε

Core points

Density connected

DBSCAN clustering

DBSCAN* clustering

Runtime

Naive algorithm runs in $\mathcal{O}(n^2)$ time.

"Since the Eps-neighborhoods are expected to be small compared to the size of the whole data space, the average run time complexity of a single region query is $\mathcal{O}(\log n)$. [...] Thus, the average run time complexity of DBSCAN is $\mathcal{O}(n \log n)$."

DBSCAN:
p and q are in the same cluster $\Leftrightarrow$ p and q are density connected

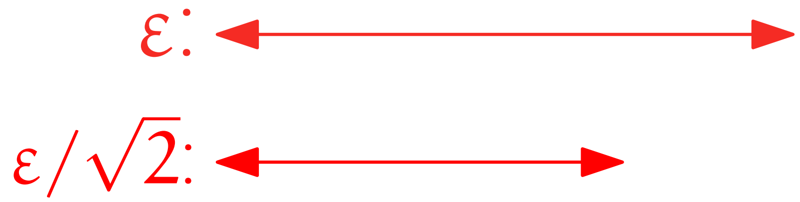
DBSCAN*:
(and core pts.)

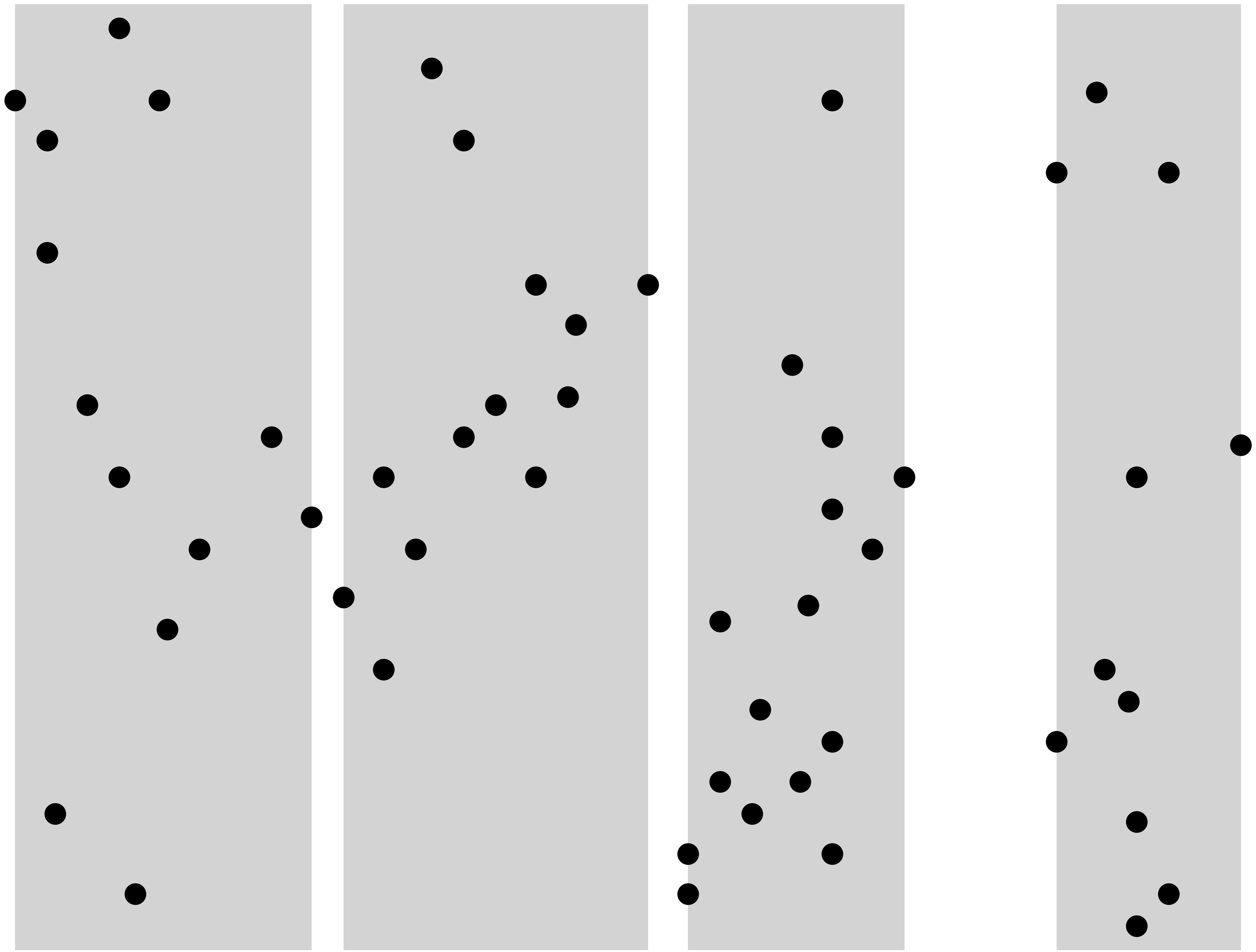
De Berg, Gunawan, Roeloffzen (2017)

Everywhere: ε free, k fixed constant, Euclidean distances

	2D	dD	
DBSCAN	$\mathcal{O}(n \log n)$ ←	$\mathcal{O}(n^{2-\frac{2}{\lceil d/2 \rceil+1}+\gamma})$	$\gamma > 0$
HDBSCAN	$\mathcal{O}(n \log n)$ expected	$\times$	

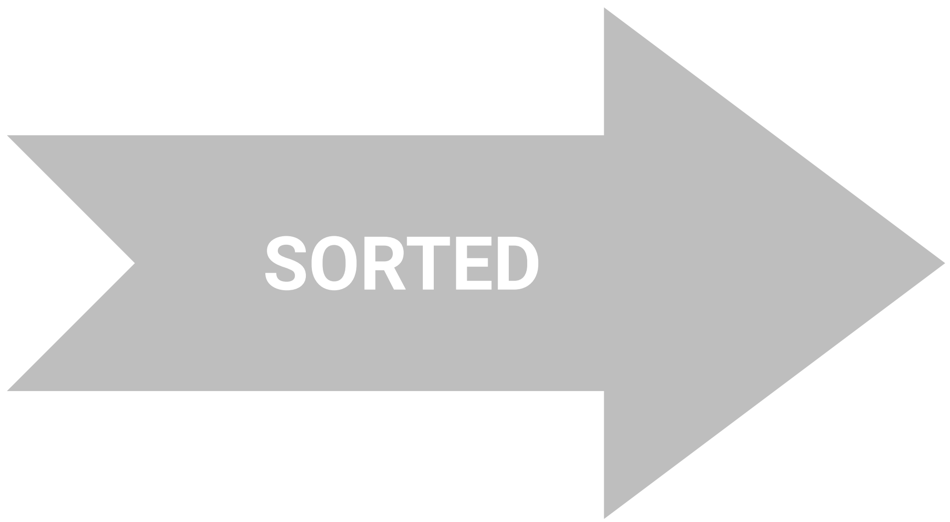
Box graph $\mathcal{G}_{\text{box}}$



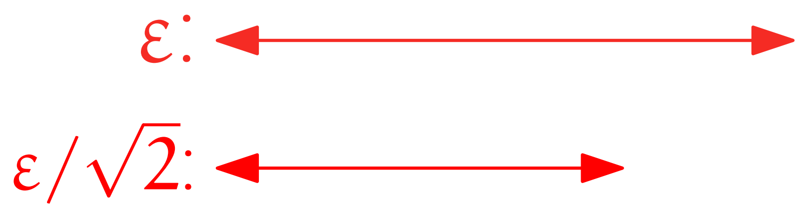


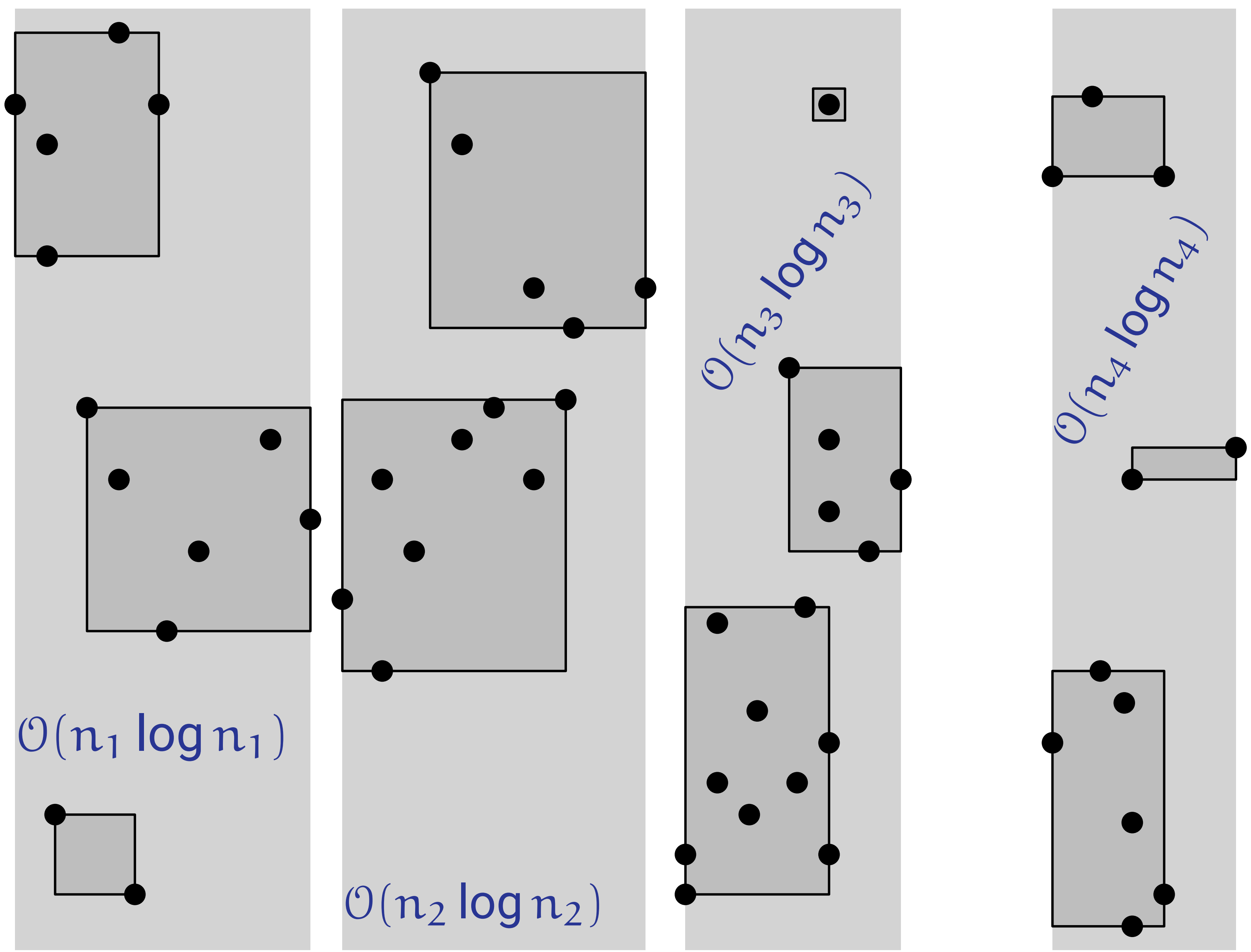
1. Construct boxes

Add points as long as strip width $\leqslant \varepsilon/\sqrt{2}$.



Box graph $\mathcal{G}_{box}$





1. Construct boxes

Add points as long as strip width $\leqslant \varepsilon/\sqrt{2}$.

Per strip: add points to box as long as height $\leqslant \varepsilon/\sqrt{2}$.

Runtime:

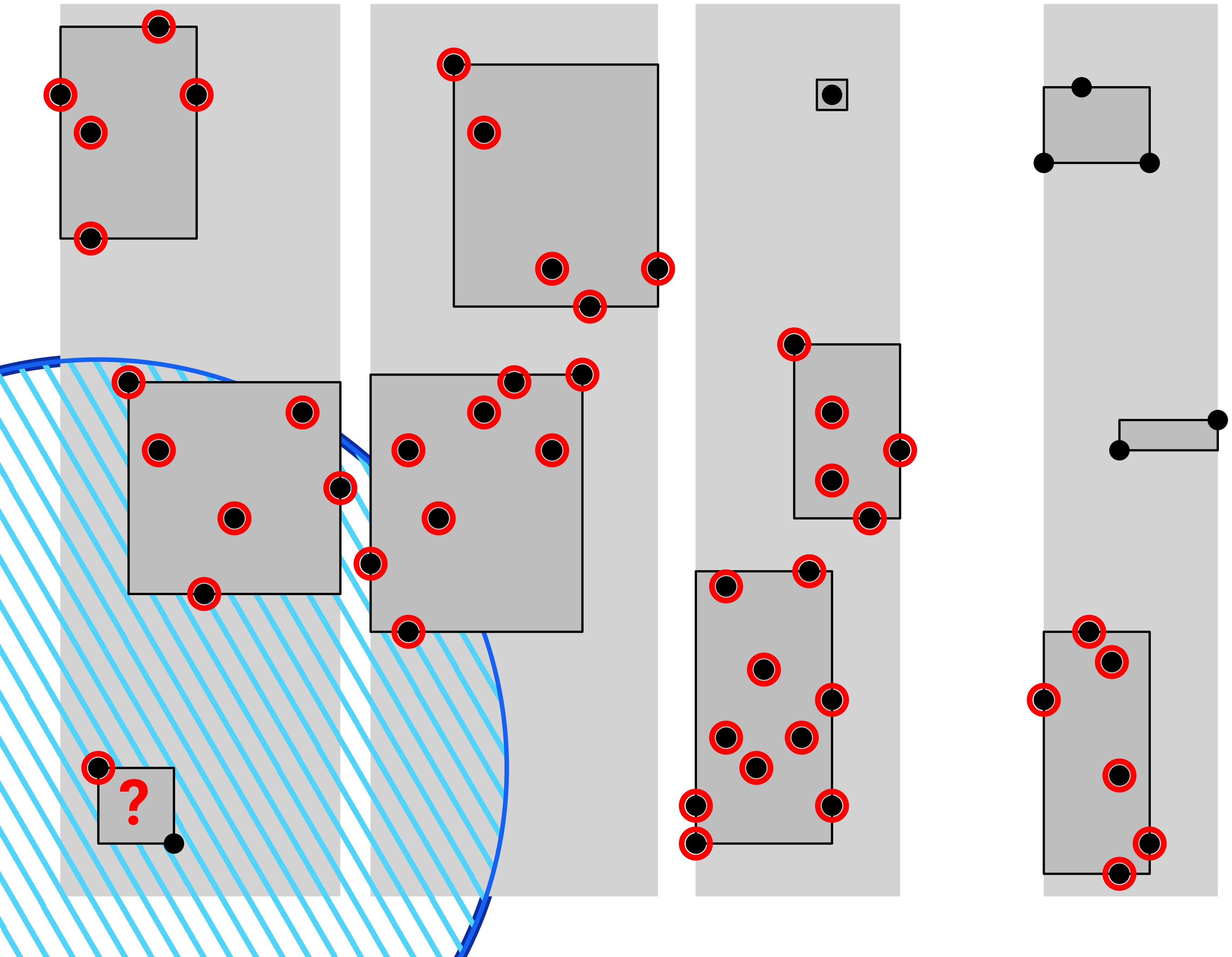
- Sort by x
  $\mathcal{O}(n \log n)$
- Sort by y per strip
  $\sum_j \mathcal{O}(n_j \log n_j)$
- Total
  $\mathcal{O}(n \log n)$

Box graph $\mathcal{G}_{box}$

$k = 4$

ε: ⟷

$\varepsilon/\sqrt{2}$: ⟷



Property of single boxes

All points within a box... are in ε-neighbourhood.

(Box width & height are each $\leqslant \varepsilon/\sqrt{2}$.)

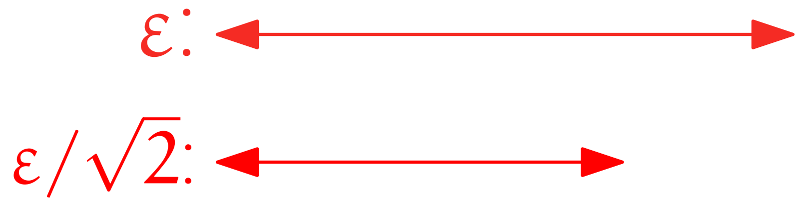
In boxes with at least k points, ...

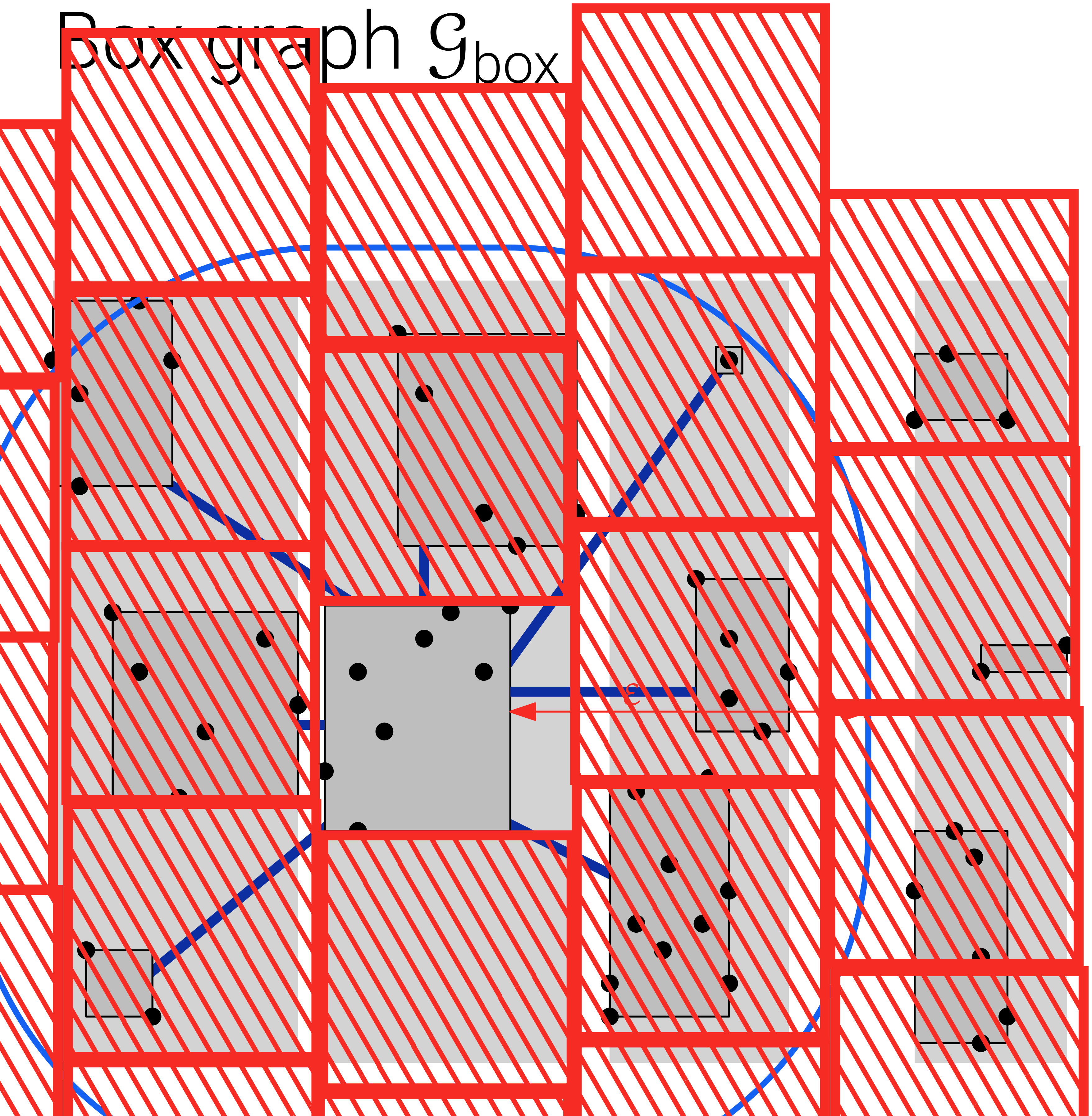
all points are core points.

In boxes with fewer than k points, ...

points **can** be core points.

Box graph $\mathcal{G}_{\text{box}}$





Property of box pairs

Connect boxes with edge if distance between **boxes** is at most ε.

Nonneighbours in $\mathcal{G}_{\text{box}}$: none of these points are in ε-neighbourhood.

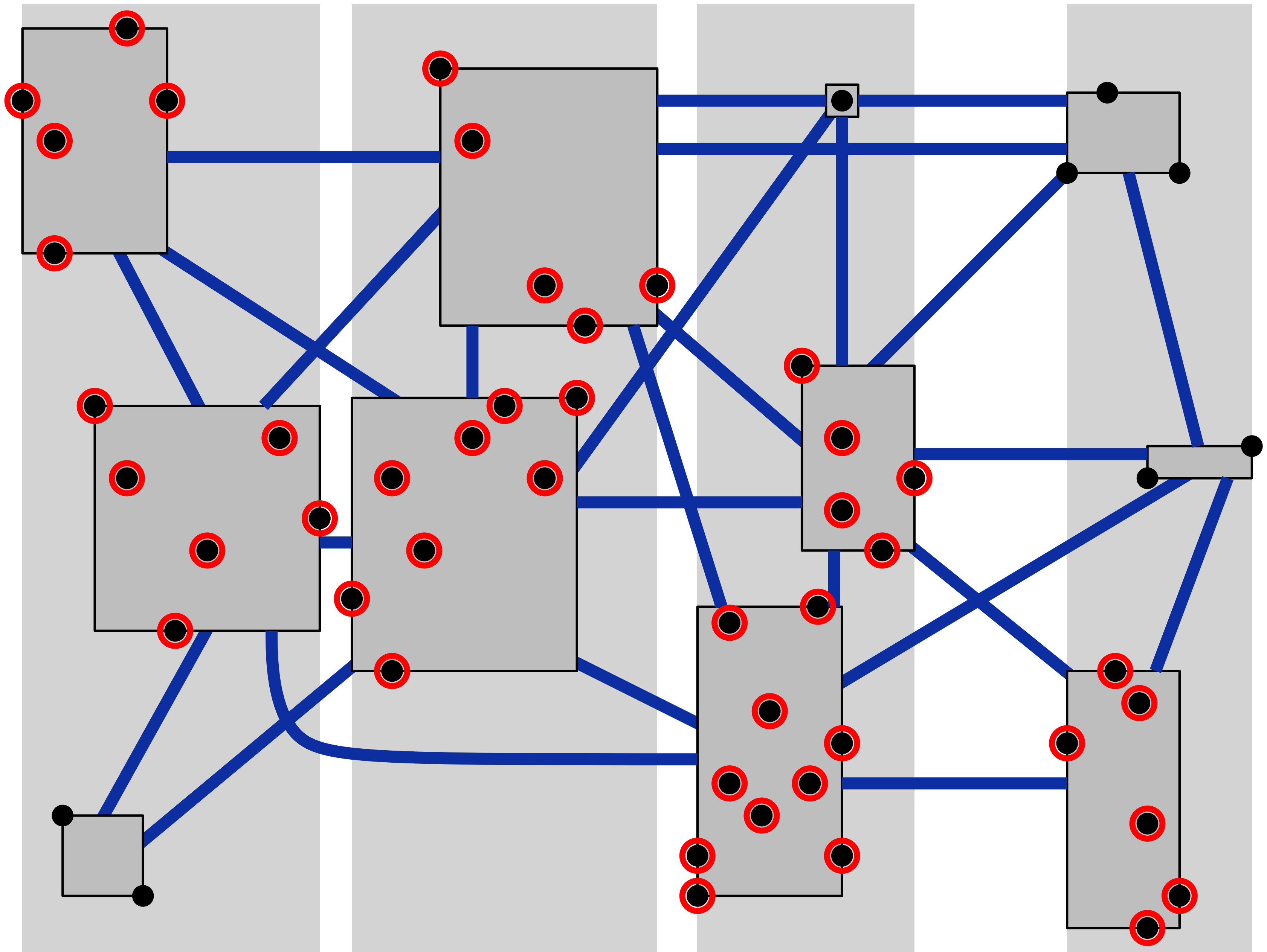
How many neighbours can a box have? **22** $\in \mathcal{O}(1)$

Box graph $\mathcal{G}_{box}$

$k = 4$

ε:

$\varepsilon/\sqrt{2}$:



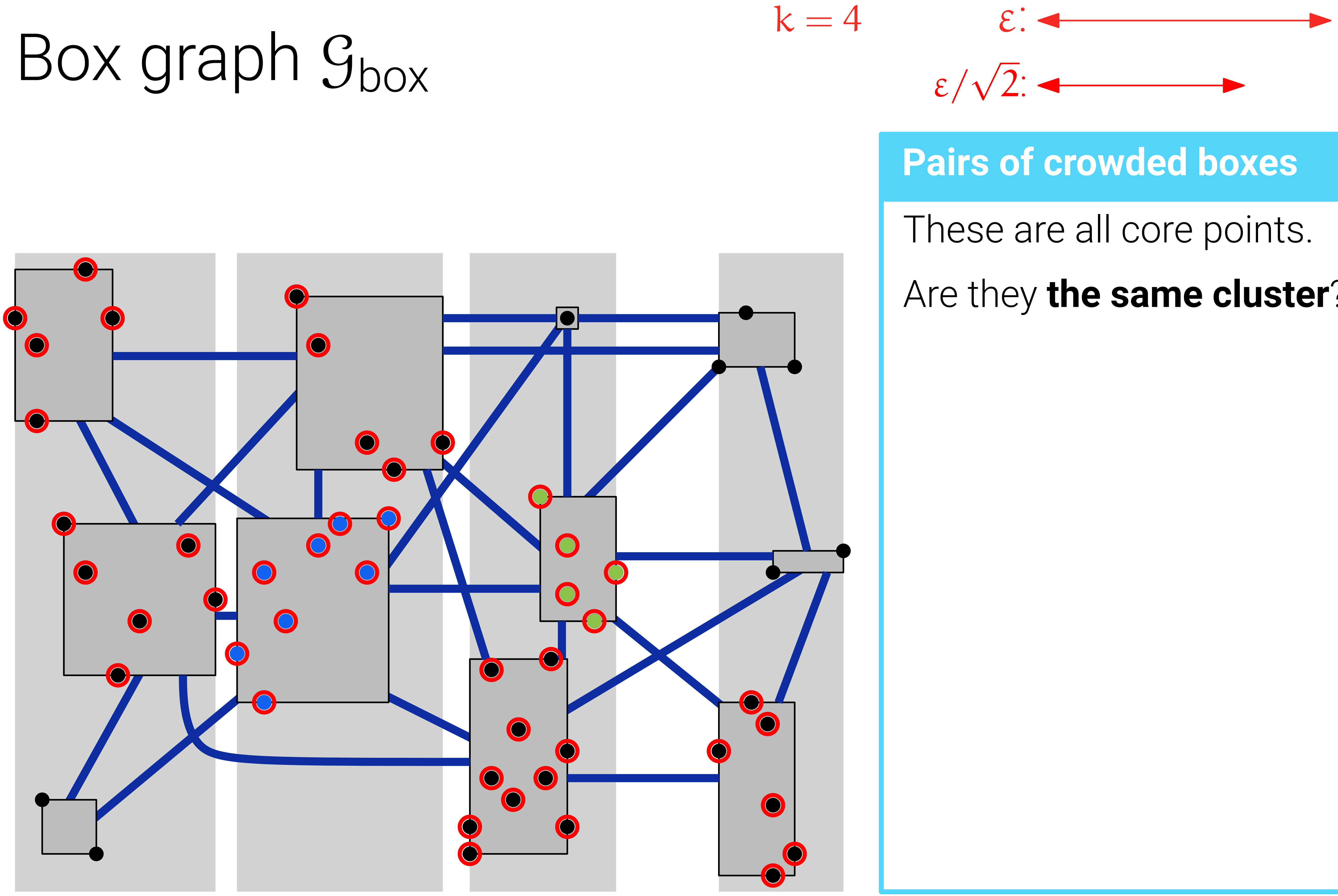
2. Find all core points

Already have all core points in "crowded" boxes.

For all "sparse" boxes:
  For all neighbour boxes:
    ... check all pairs.

Total runtime?
  Other box is sparse:
    $\mathcal{O}(k^2) = \mathcal{O}(1)$
  Other box is crowded:
    Charge to crowded box:
    Point in crowded box checked $\leqslant 22k$ times (!!)

Box graph $\mathcal{G}_{box}$

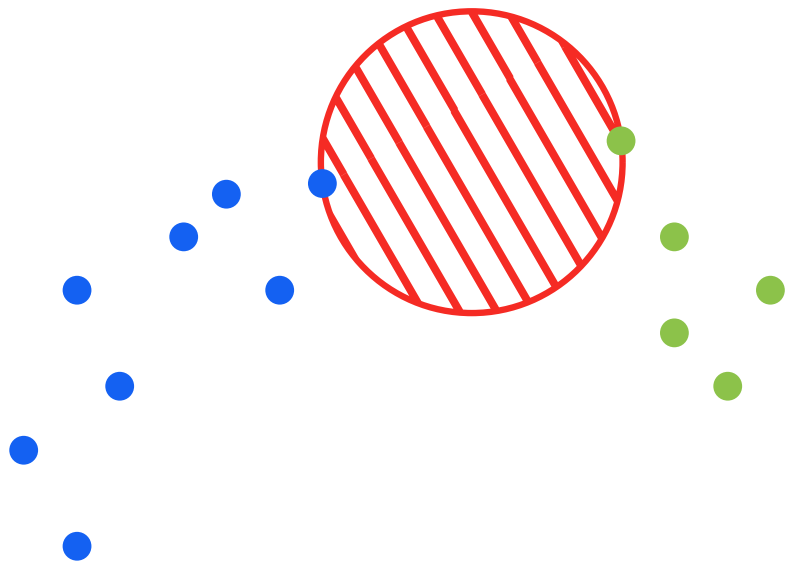


Pairs of crowded boxes

These are all core points.

Are they **the same cluster**?

Box graph $\mathcal{G}_{\text{box}}$



Pairs of crowded boxes

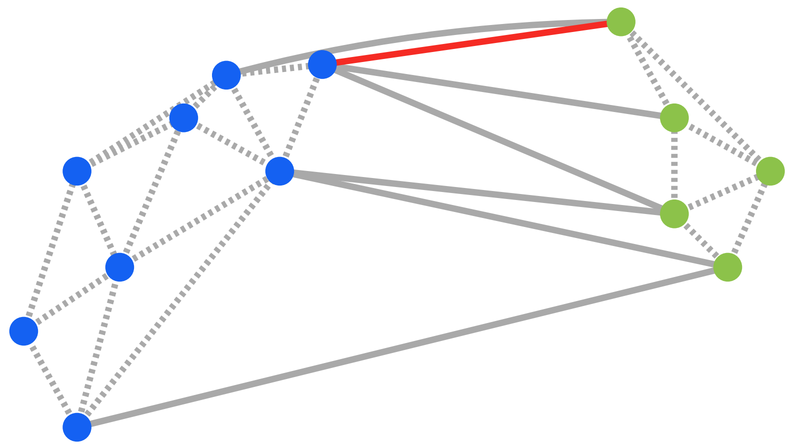
These are all core points.

Are they **the same cluster**?

BICHROMATIC CLOSEST PAIR

In Euclidean 2D?

Box graph $\mathcal{G}_{\text{box}}$



Pairs of crowded boxes

These are all core points.

Are they **the same cluster**?

BICHROMATIC CLOSEST PAIR

In Euclidean 2D?

Delaunay triangulation (DT) contains this edge!

DT has $\mathcal{O}(n)$ edges, takes $\mathcal{O}(n \log n)$ time for n pts.

Charge to edges in $\mathcal{G}_{\text{box}}$:

Edge ij gets charged $c_{ij}(n_i + n_j)\log(n_i + n_j)$.

Total charge is $\mathcal{O}(n \log n)$ since $\sum_{ij \text{ is edge}} n_i \leqslant 22kn_i$.

Results

Everywhere: ε free, k fixed constant, Euclidean distances

	2D	dD	
DBSCAN	$\mathcal{O}(n \log n)$	$\mathcal{O}(n^{2-\frac{2}{\lceil d/2 \rceil+1}+\gamma})$	$\gamma > 0$
HDBSCAN	$\mathcal{O}(n \log n)$ expected	$\times$	

1. Construct $\mathcal{G}_{\text{box}}$
2. Find core points
3. Merge clusters

(4. Assign border points.)

now

BICHROMATIC CLOSEST POINT instead of Delaunay triangulation.

(Agarwal, Edelsbrunner, Schwarzkopf, 1991)

HDBSCAN

[McInnes, Healy, Astels: JOSS 2017]

Use DBSCAN* and sweep ε from 0 to ∞.

Initially all points are noise; eventually everything is one cluster.

Three types of "events":

- Noise point becomes core point. Call this value $d_{\text{core}}(p)$.
- New cluster forms.
- Two clusters merge

Events only happen when $\varepsilon = d(p, q)$ for some p, q.

Store this tree structure of cluster creation and merges: HDBSCAN.

Mutual reachability

Starting at which value of ε will these points be in the same cluster?

Both need to be core points, so at least $d_{\text{core}}(p)$ and $d_{\text{core}}(q)$.
Either $\varepsilon \geqslant d(p, q)$, or they must be connected through other points.

Def. Let $d_{\text{mr}}(p, q) = \textbf{max}\{\, d_{\text{core}}(p),\; d_{\text{core}}(q),\; d(p, q)\,\}$.

Def. Mutual reachability graph $\mathcal{G}_{\text{mr}}$: complete, edge weights d_{mr}.

Algorithm:

1. Compute d_{core} for all points. $\mathcal{O}(n \log n)$ time [Vaidya, 1989]
2. Construct $\mathcal{G}_{\text{mr}}$ and compute a minimum spanning tree $\mathcal{T}$.
3. Convert $\mathcal{T}$ into HDBSCAN tree. (by Kruskal's algorithm)

2. Construct $\mathcal{G}_{\text{mr}}$ and compute an MST.

Cannot look at all edges: too slow.

k^{th}-order

Def. $\{p, q\}$ is a Delaunay edge "iff" there exists a circle with:

- p and q on the boundary
- ~~no~~ $\leqslant k$ points in its interior

a "k-OD edge"

Theorem (Gudmundsson, Hammer, van Kreveld, 2002)
The k^{th}-order Delaunay graph has $\mathcal{O}(n(k+1))$ edges and can be computed in $\mathcal{O}(n(k+1)\log n)$ expected time by randomized incremental construction.

Claim: The MST of $\mathcal{G}_{\text{mr}}$ uses only k-OD edges.

The MST of $\mathcal{G}_{mr}$ uses only k-OD edges.

Consider applying Kruskal's algorithm to $\mathcal{G}_{mr}$:

- Looks at edges in order of increasing cost.
- With weights d_{mr} this corresponds to the HDBSCAN events.

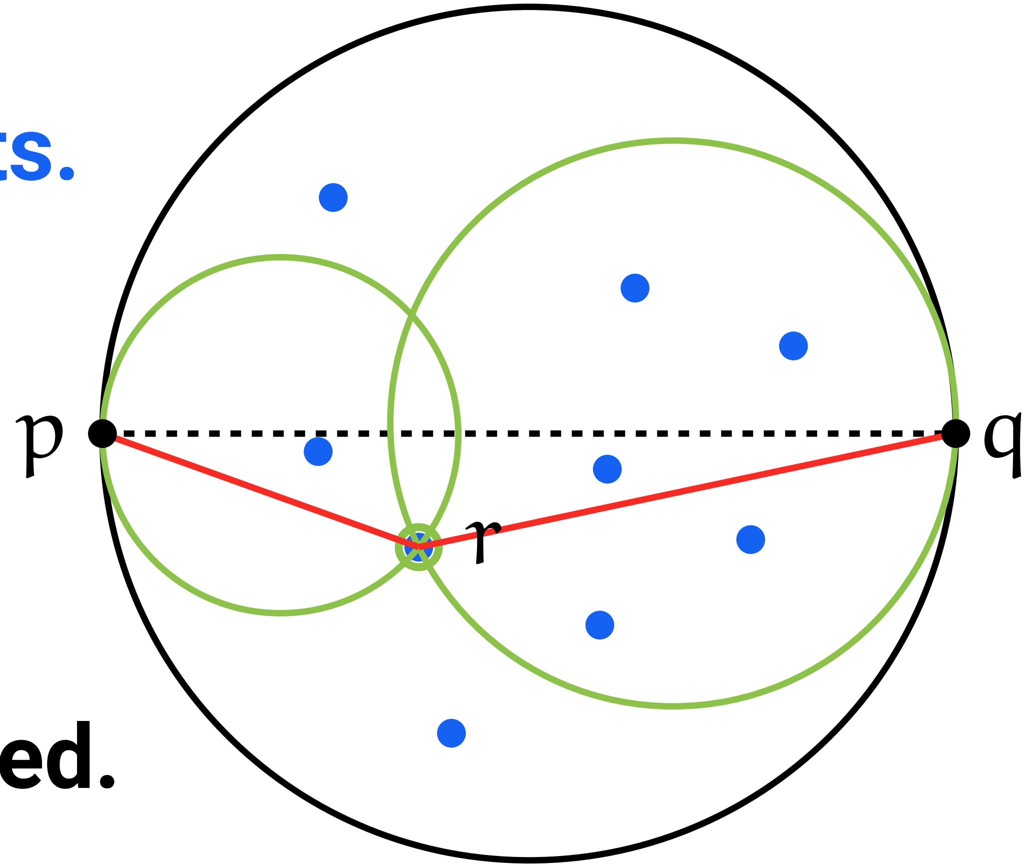
Claim: Whenever Kruskal looks at a non-k-OD edge $\{p, q\}$, p and q are already in the same cluster, and thus ignores the edge.

Not a k-OD edge, so more than k points.

Pick any point.

Recurse until only k-OD edges.

Kruskal has already considered those edges, so p and q are already connected.



Results

Everywhere: ε free, k fixed constant, Euclidean distances

	2D	dD	
DBSCAN	$\mathcal{O}(n \log n)$	$\mathcal{O}(n^{2-\frac{2}{\lceil d/2 \rceil+1}+\gamma})$	$\gamma > 0$
HDBSCAN	$\mathcal{O}(n \log n)$ expected	$\times$	