Clustering with Few Disks to Minimize the Sum of Radii

(accepted at SoCG '24)

Mikkel Abrahamsen Sarita de Berg Lucas Meijer
André Nusser Leonidas Theocharous

Inria CNRS Université Côte d'Azur

Introduction

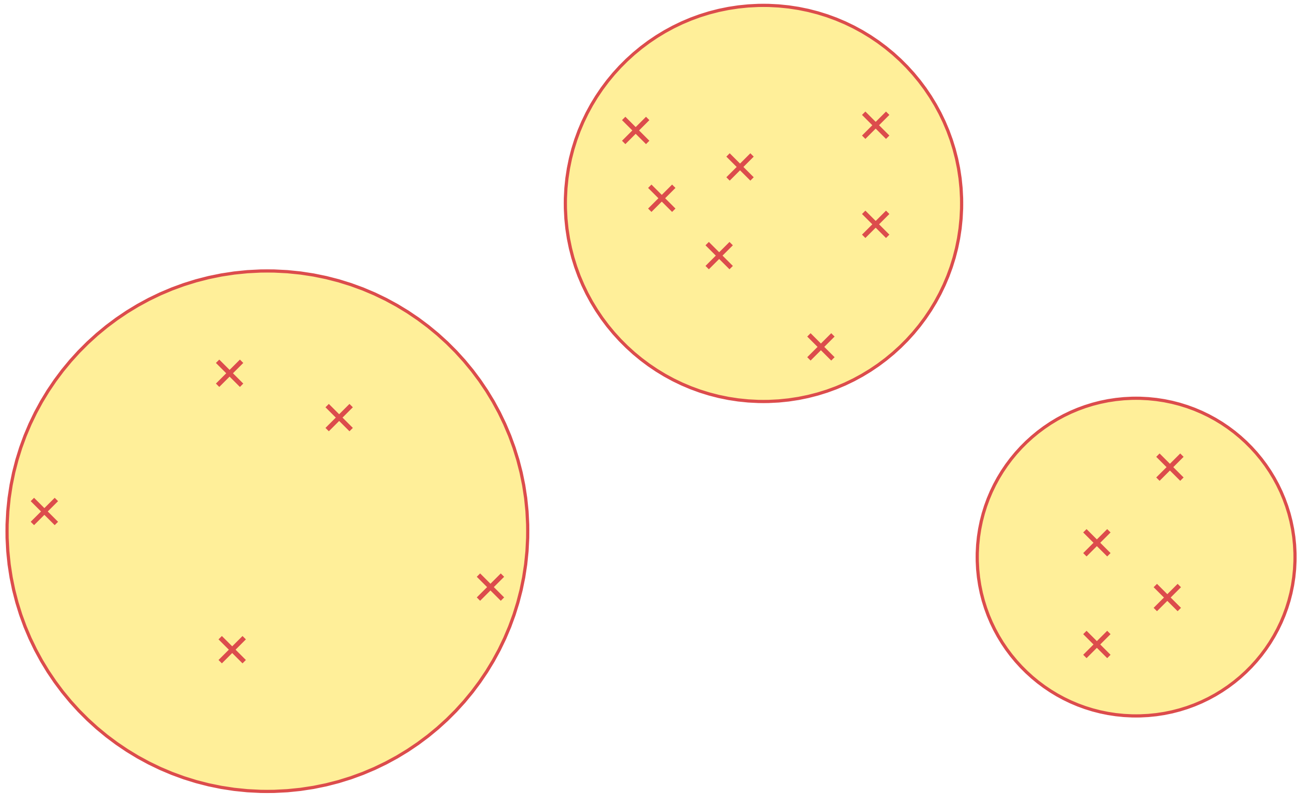
Clustering in the Plane

Given: Set of points P in the plane

Clustering in the Plane

Given: Set of points P in the plane

Goal: Cover P with k disks, minimizing the *objective*

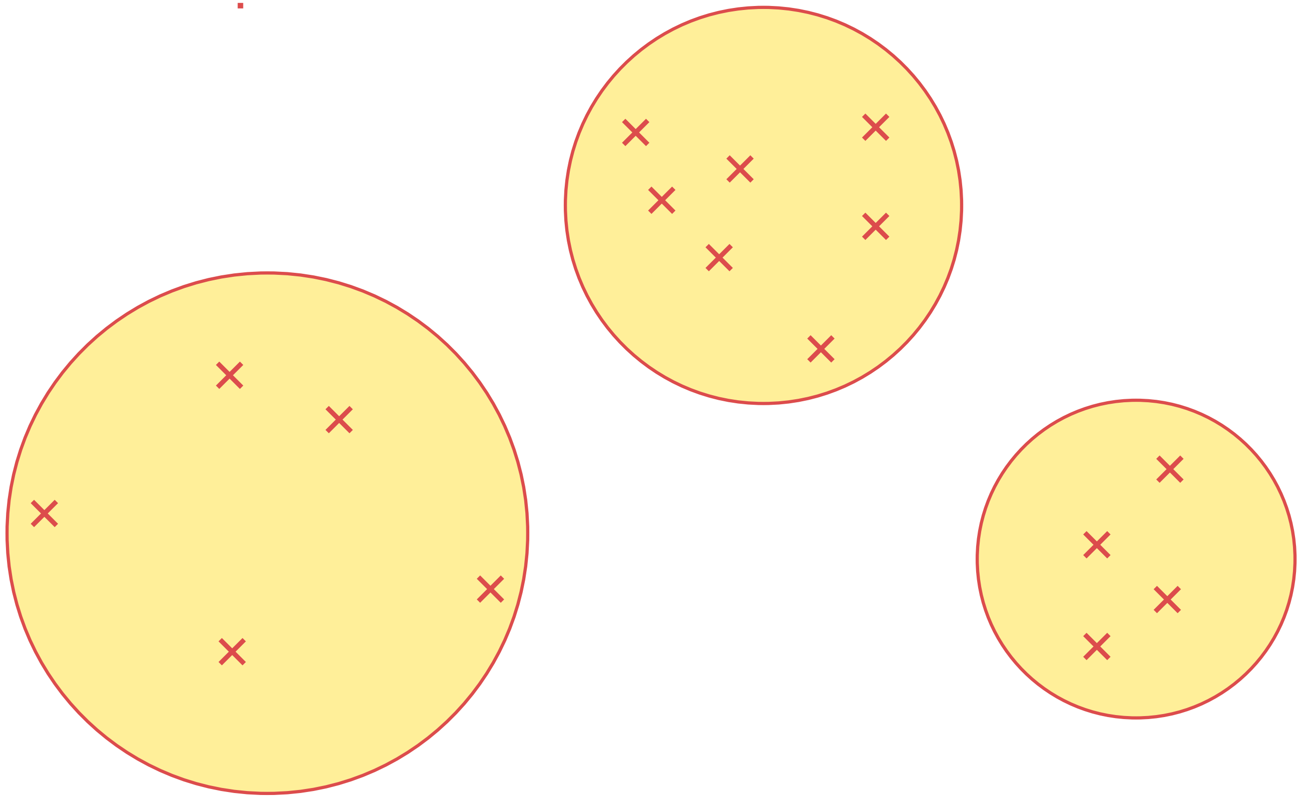


Clustering in the Plane

Given: Set of points P in the plane

Goal: Cover P with k disks, minimizing the *objective*

?



Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

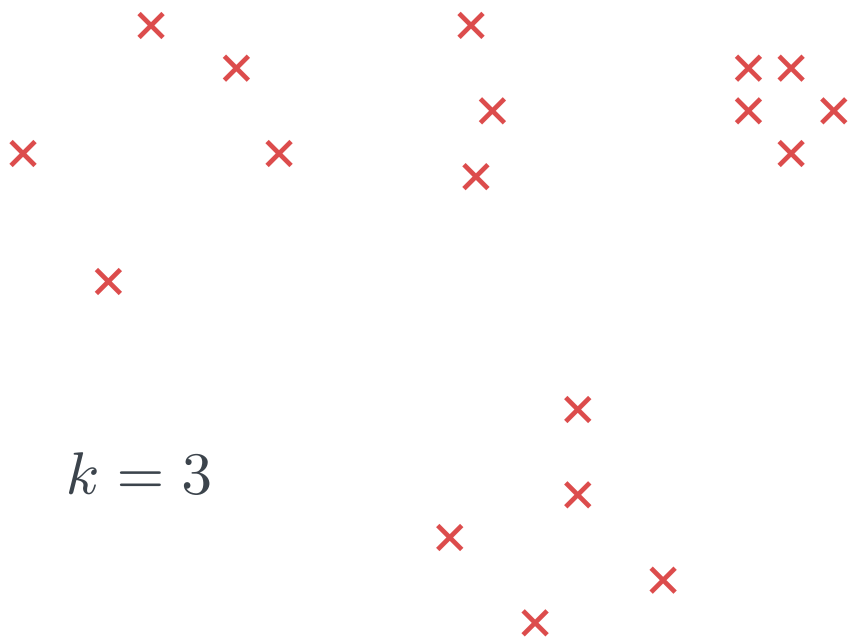
$$\max_i r(C_i).$$

Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\max_i r(C_i).$$



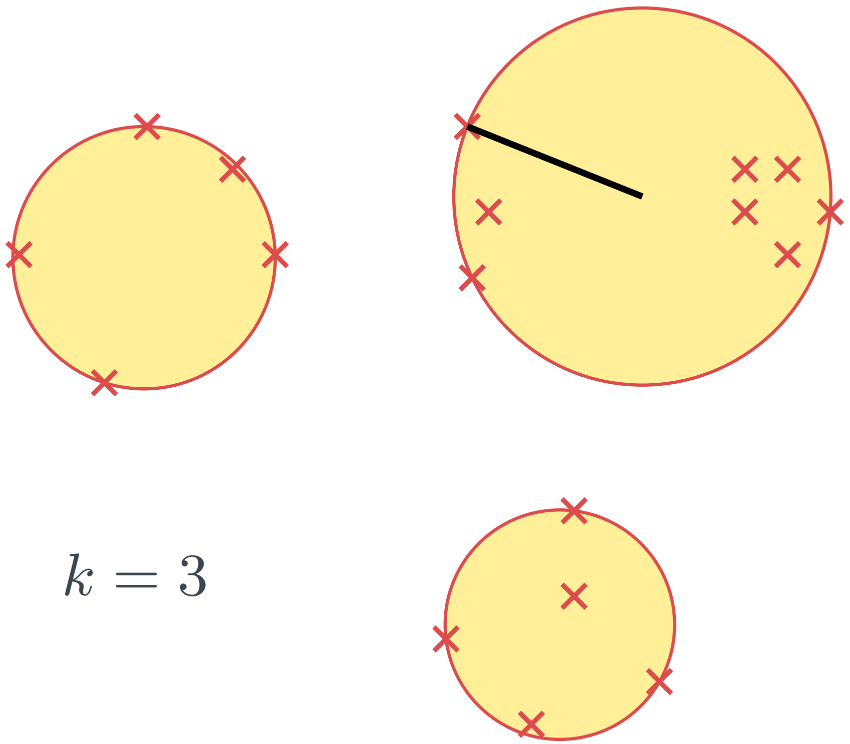
$k = 3$

Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\max_i r(C_i).$$



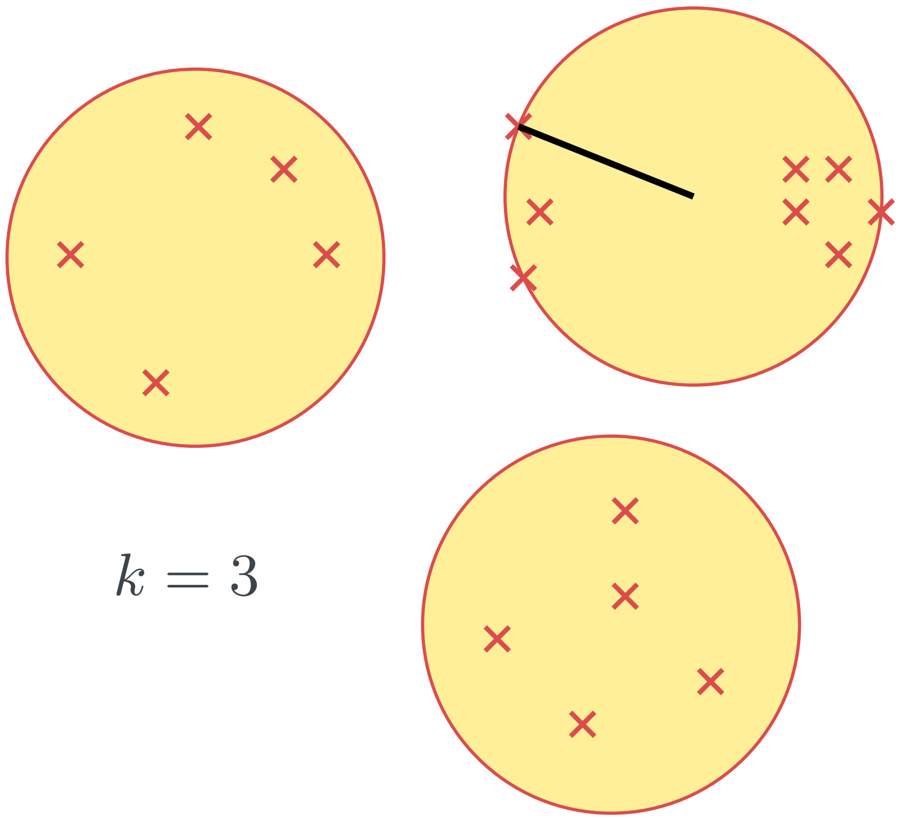
$k = 3$

Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\max_i r(C_i).$$

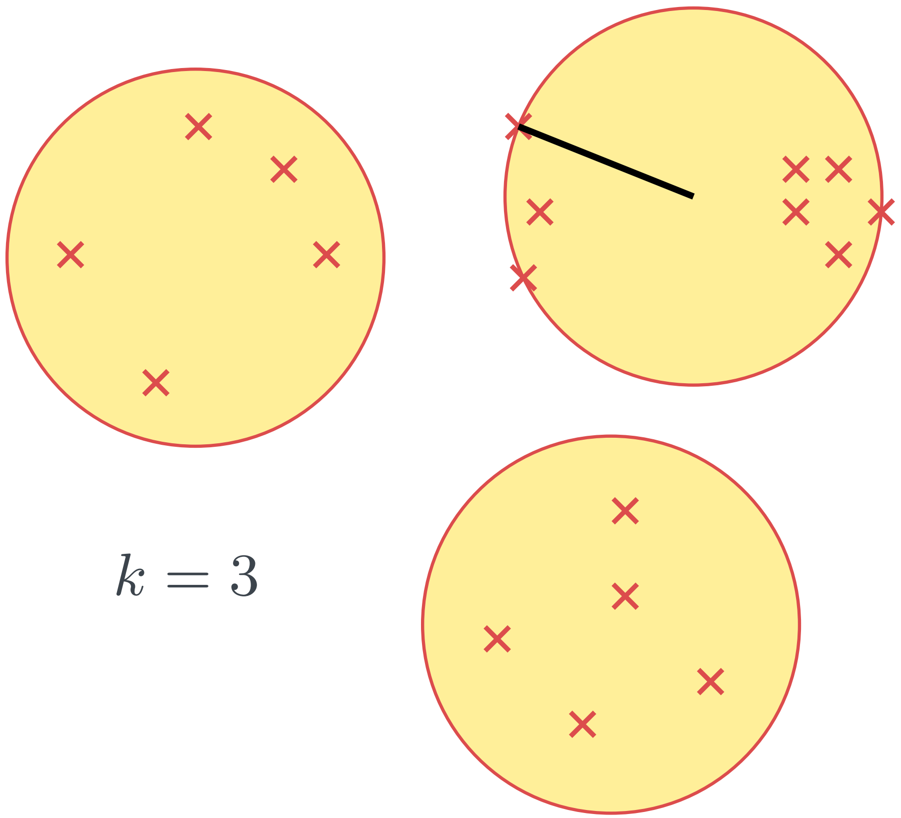


Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\max_i r(C_i).$$



k-MinSumRadius

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

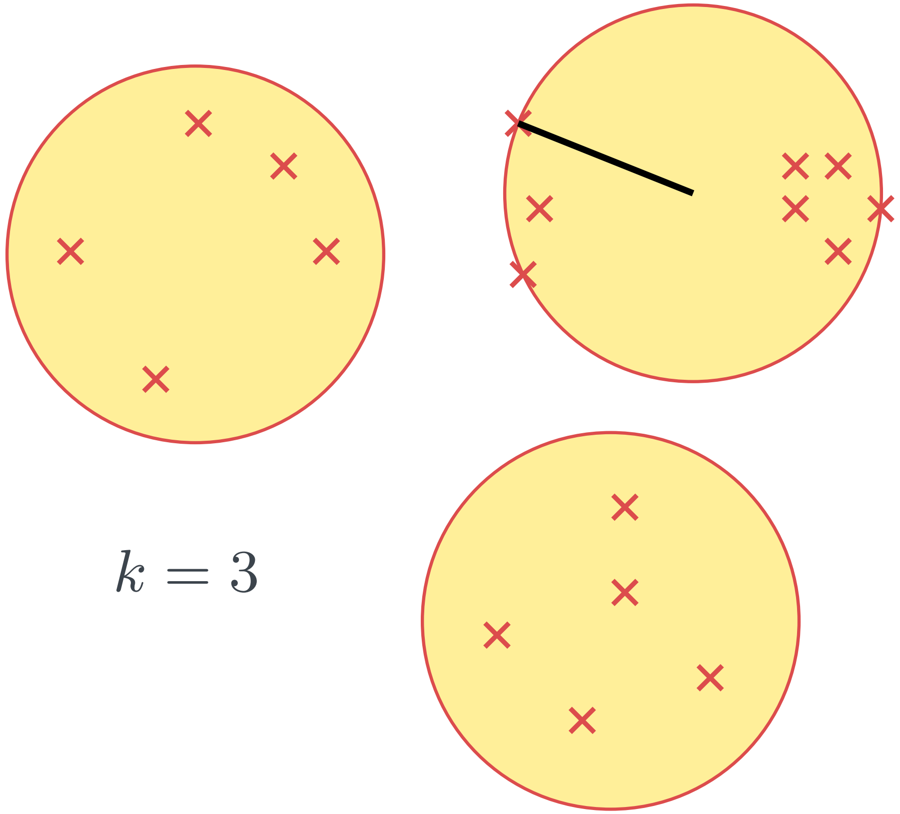
$$\sum_i r(C_i).$$

Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

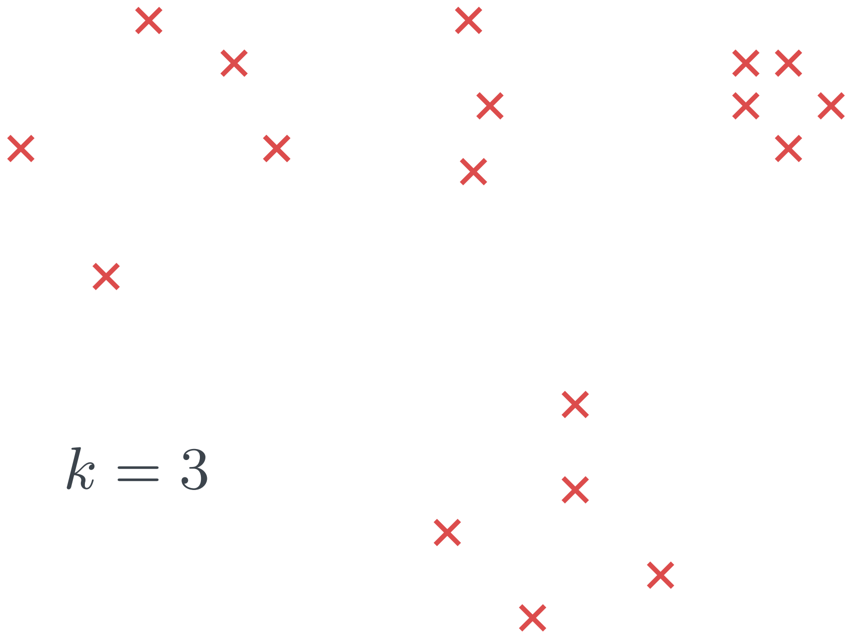
$$\max_i r(C_i).$$



k-MinSumRadius

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\sum_i r(C_i).$$

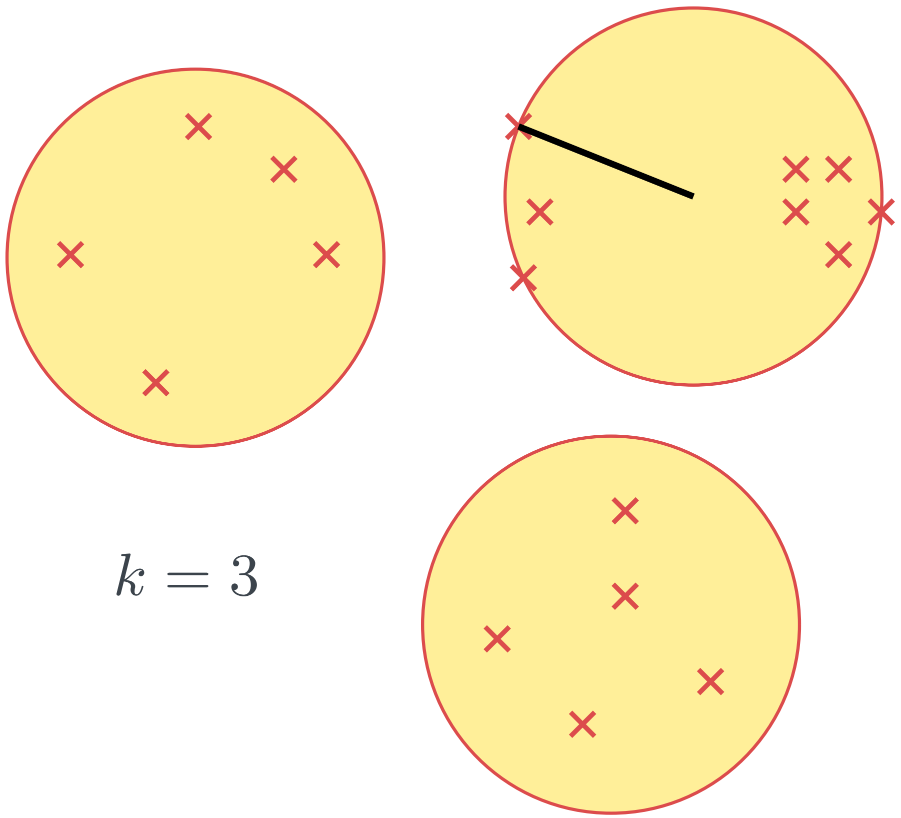


Clustering in the Plane

k-Center

Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

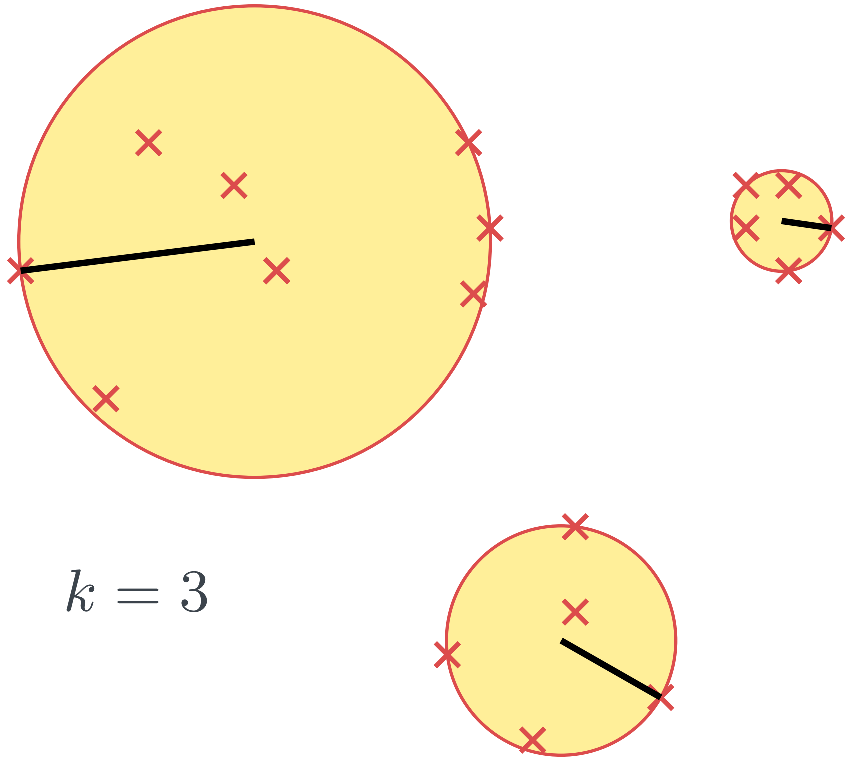
$$\max_i r(C_i).$$



k-MinSumRadius

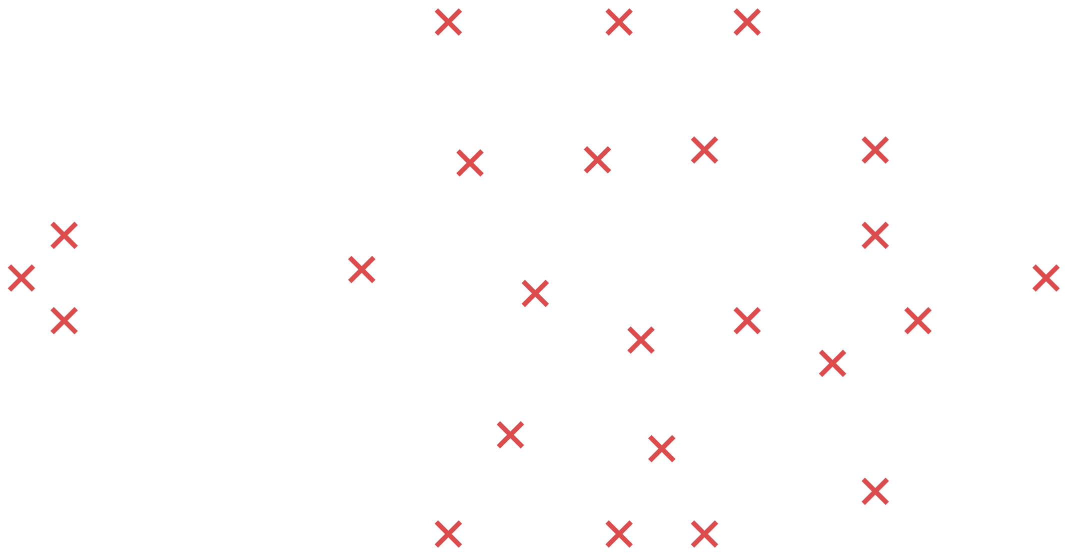
Problem: Find disks $C_1, \ldots, C_k$ covering P minimizing

$$\sum_i r(C_i).$$



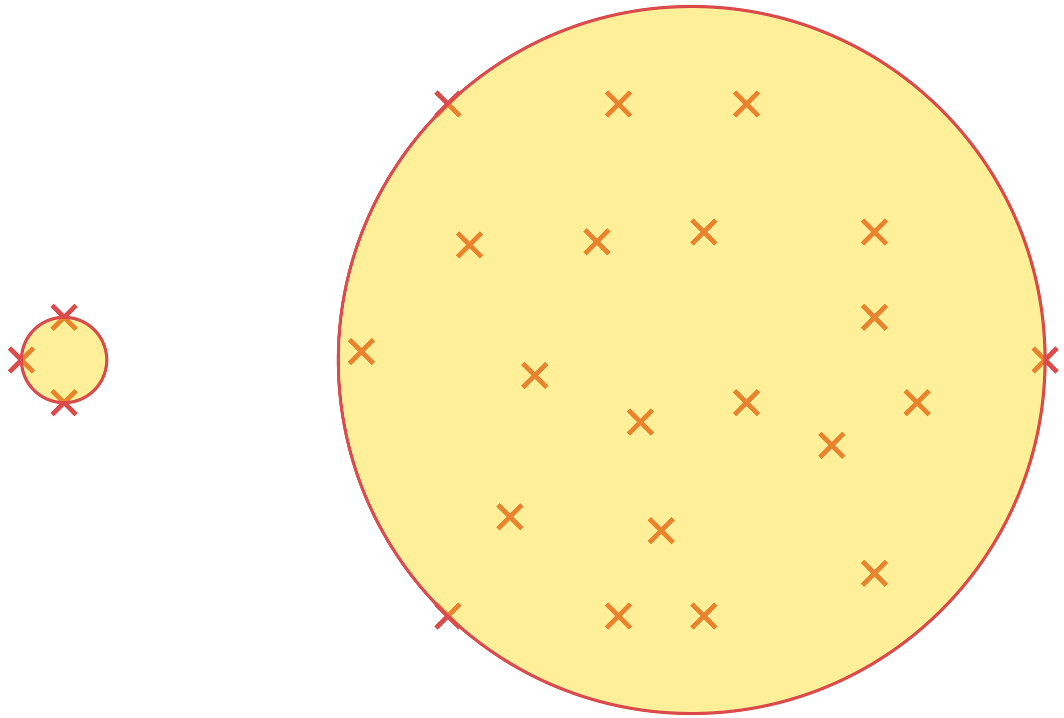
Clustering in the Plane

k-Center vs. k-MinSumRadius



Clustering in the Plane

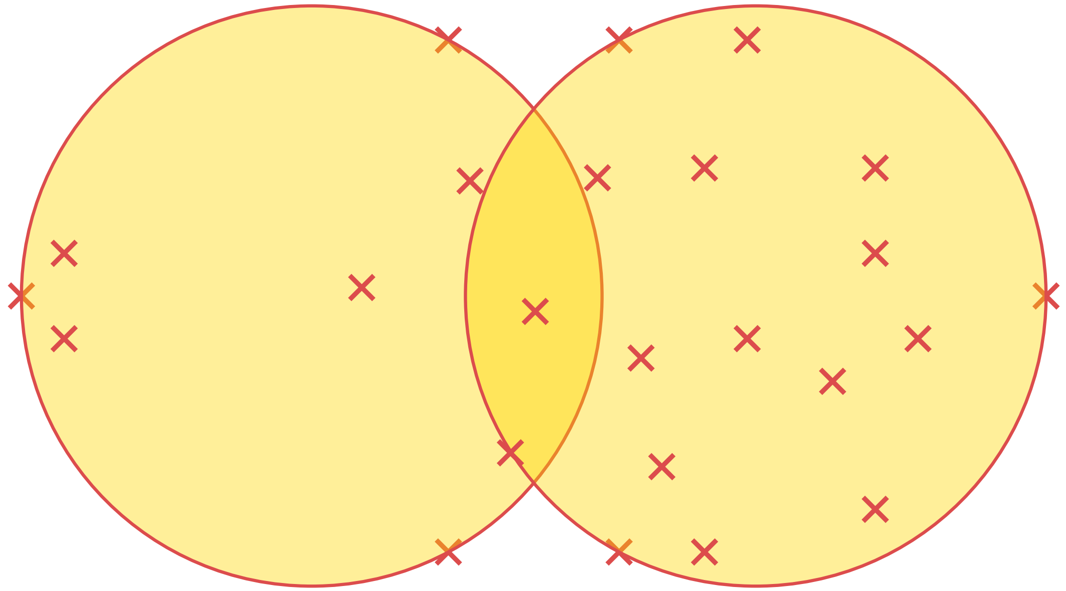
k-Center vs. k-MinSumRadius



k-MinSumRadius solution

Clustering in the Plane

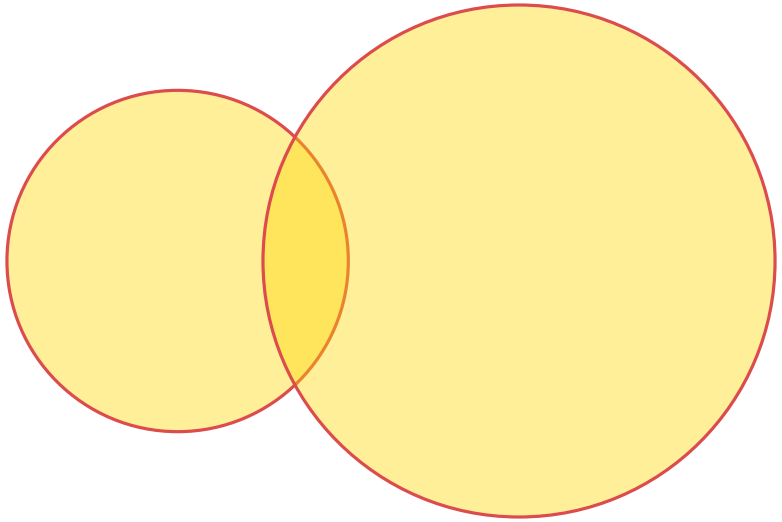
k-Center vs. k-MinSumRadius



k-Center solution

Clustering in the Plane

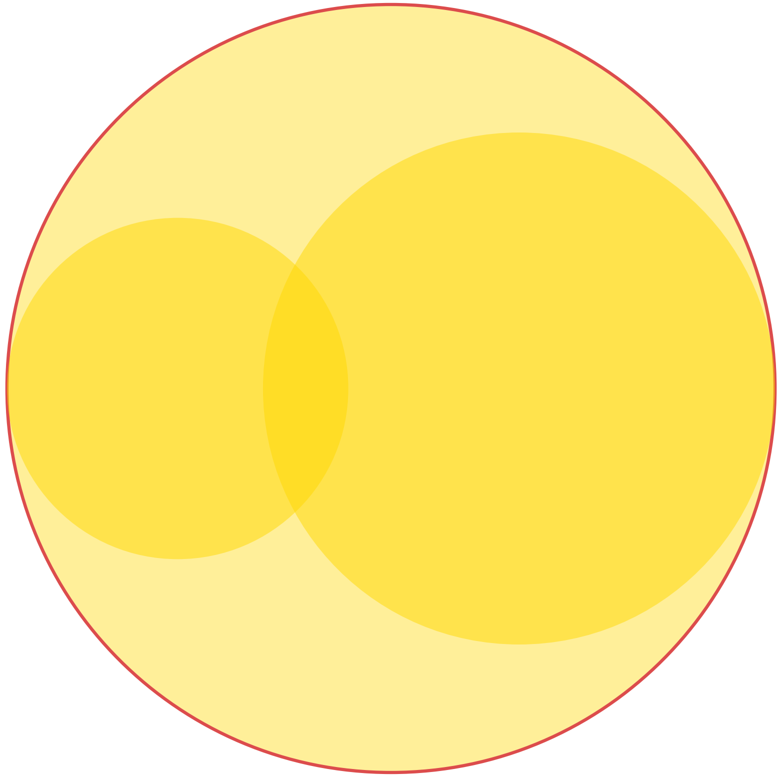
k-Center vs. k-MinSumRadius



k-MinSumRadius clusters do **not** intersect

Clustering in the Plane

k-Center vs. k-MinSumRadius



k-MinSumRadius clusters do **not** intersect

Clustering in the Plane

k-Center vs. k-MinSumRadius

k-Center

NP-hard with k part of input [Megiddo, Supovit '84]

k-MinSumRadius

$\mathcal{O}(n^{881})$ algorithm [Gibson, Kanade, Krohn, Pirwani, Varadarajan '12]

Clustering in the Plane

k-Center vs. k-MinSumRadius

k-Center

NP-hard with k part of input [Megiddo, Supovit '84]

$k = 2$:

near-quadratic:

- [Eppstein '92] [Katz, Sharir '93] [Agarwal, Sharir '94] [Jaromczyk, Kowaluk '94]

near-linear:

- [Sharir '96] [Eppstein '97] [Chan '99] [Wang '20] [Cho, Oh '20]

k-MinSumRadius

$\mathcal{O}(n^{881})$ algorithm [Gibson, Kanade, Krohn, Pirwani, Varadarajan '12]

$k = 2$:

$\mathcal{O}(n^2 \log^2 n)$ algorithm [Eppstein '92]

Our Results

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$		
$d > 2$		

* We omit $\log \log$ factors here

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$	$\mathcal{O}(n \log^2 n)$	
$d > 2$		

* We omit $\log \log$ factors here

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(n^2 \log^2 n)$
$d > 2$		

* We omit $\log \log$ factors here

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(n^2 \log^2 n)$
$d > 2$	$\mathcal{O}(n^{1-1/(\lceil d/2 \rceil + 1) + \varepsilon})$	—

* We omit $\log \log$ factors here

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(n^2 \log^2 n)$
$d > 2$	$\mathcal{O}(n^{1-1/(\lceil d/2 \rceil+1)+\varepsilon})$	—

→ All results rely on similar **structural lemmas**!

* We omit $\log \log$ factors here

Our Results

k-MinSumRadius Algorithms

	$k = 2$	$k = 3$ *
$d = 2$	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(n^2 \log^2 n)$
$d > 2$	$\mathcal{O}(n^{1-1/(\lceil d/2 \rceil+1)+\varepsilon})$	—

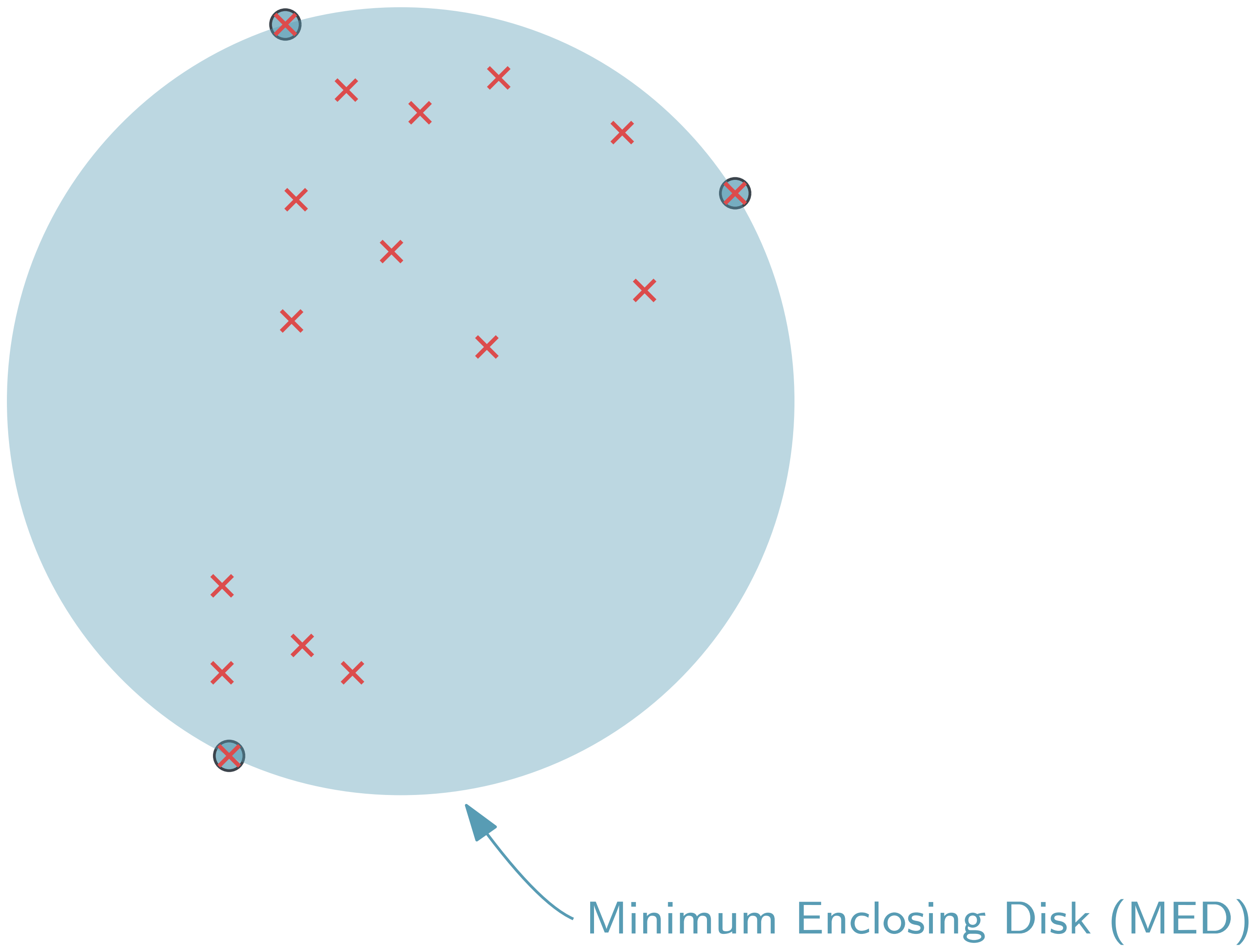
$\rightarrow$ All results rely on similar **structural lemmas**!

* We omit $\log \log$ factors here

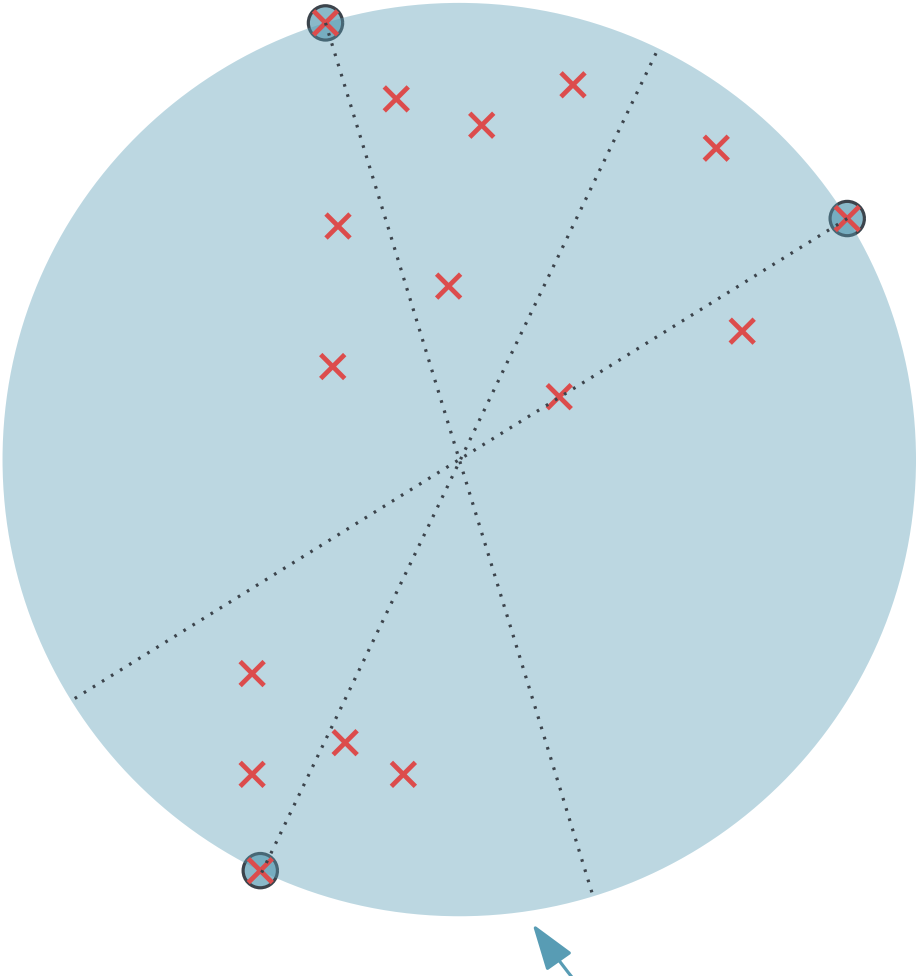
Structural Lemmas

Lemmas

Lemmas

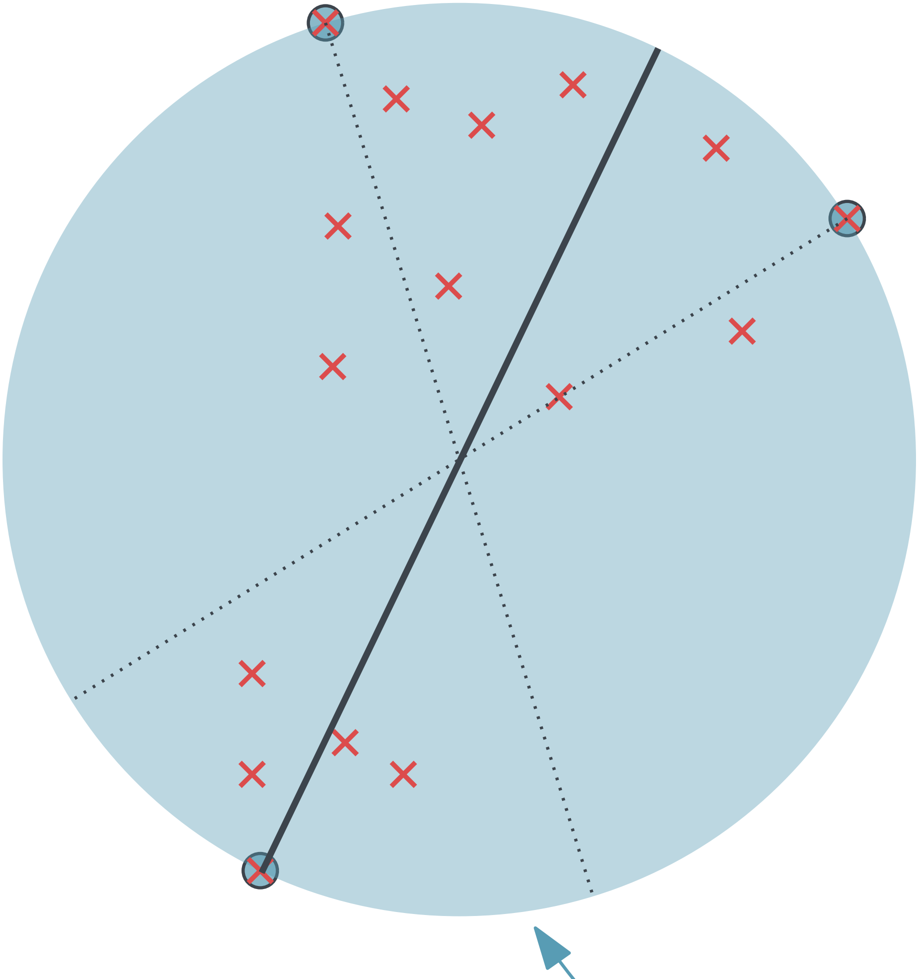


Lemmas



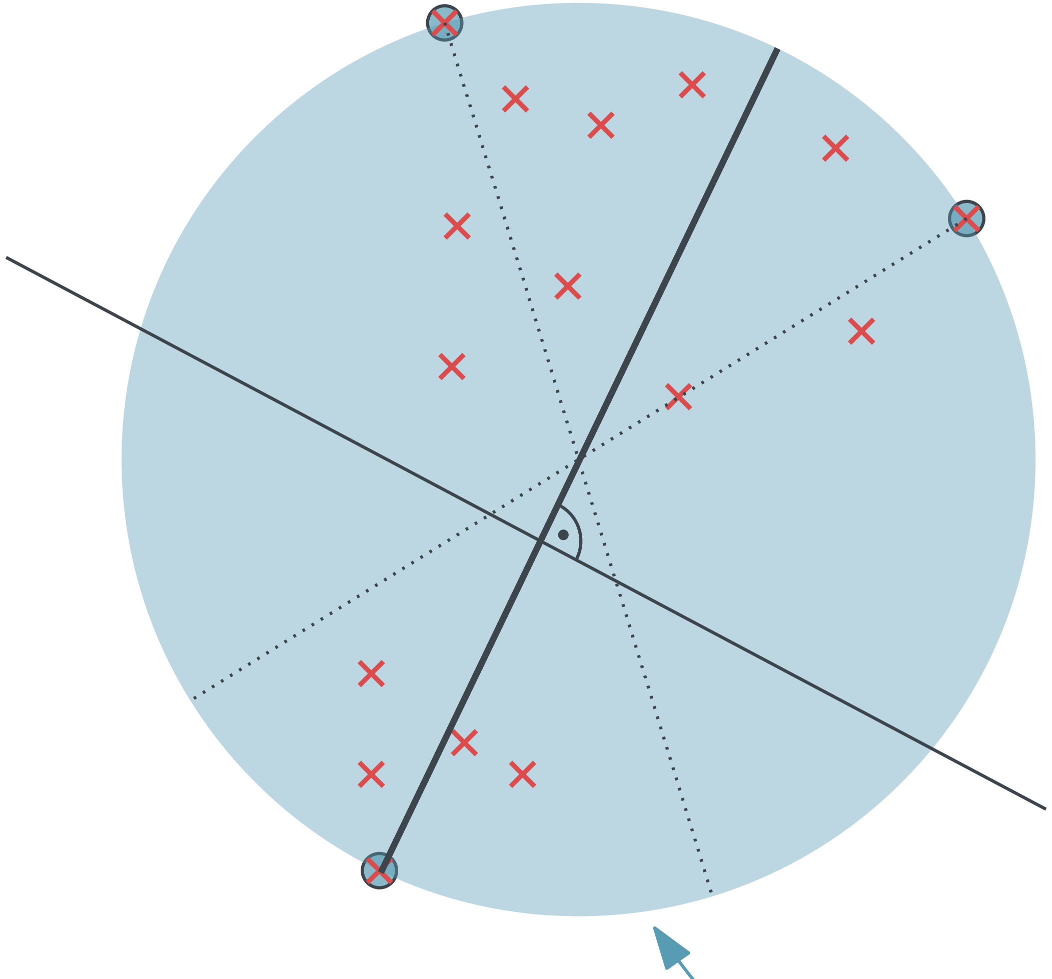
Minimum Enclosing Disk (MED)

Lemmas



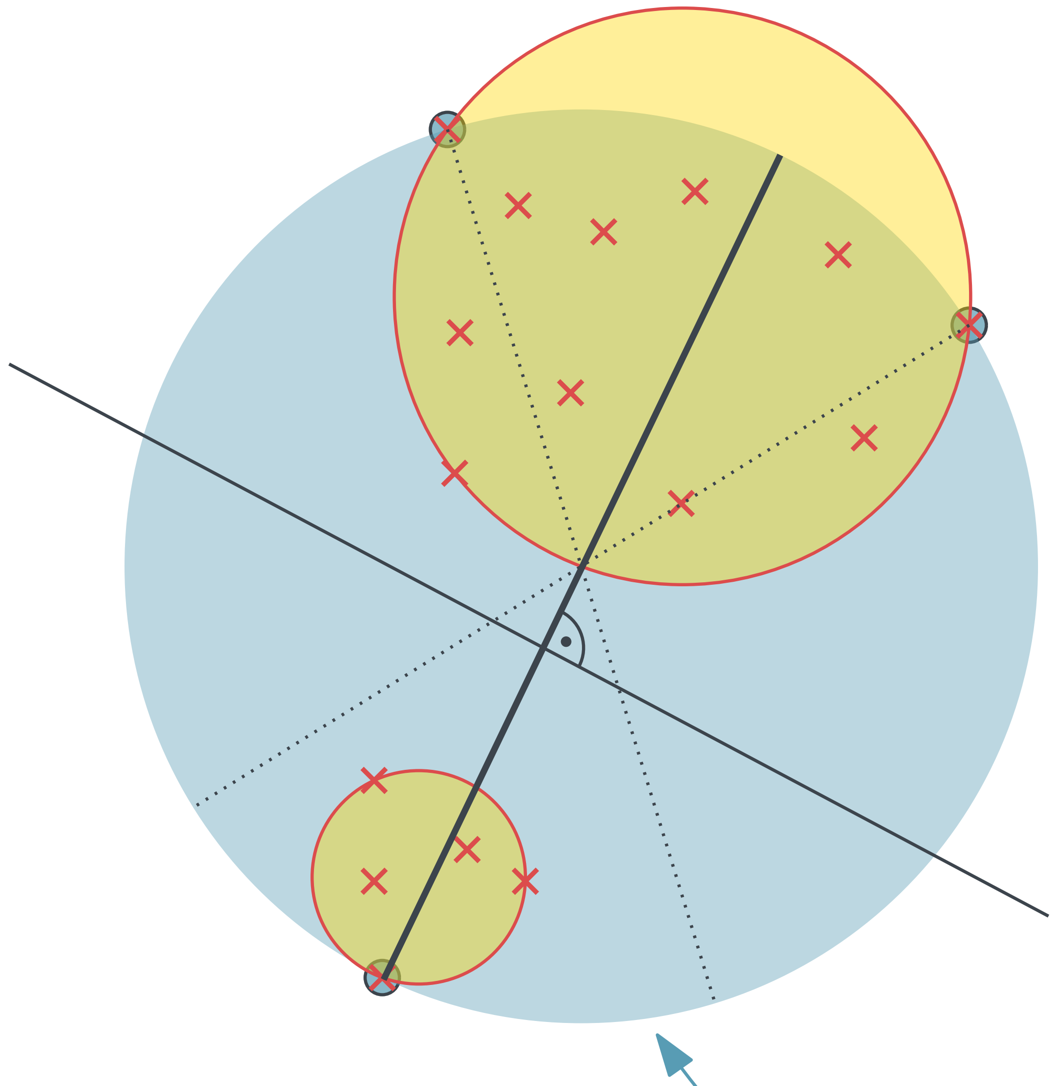
Minimum Enclosing Disk (MED)

Lemmas



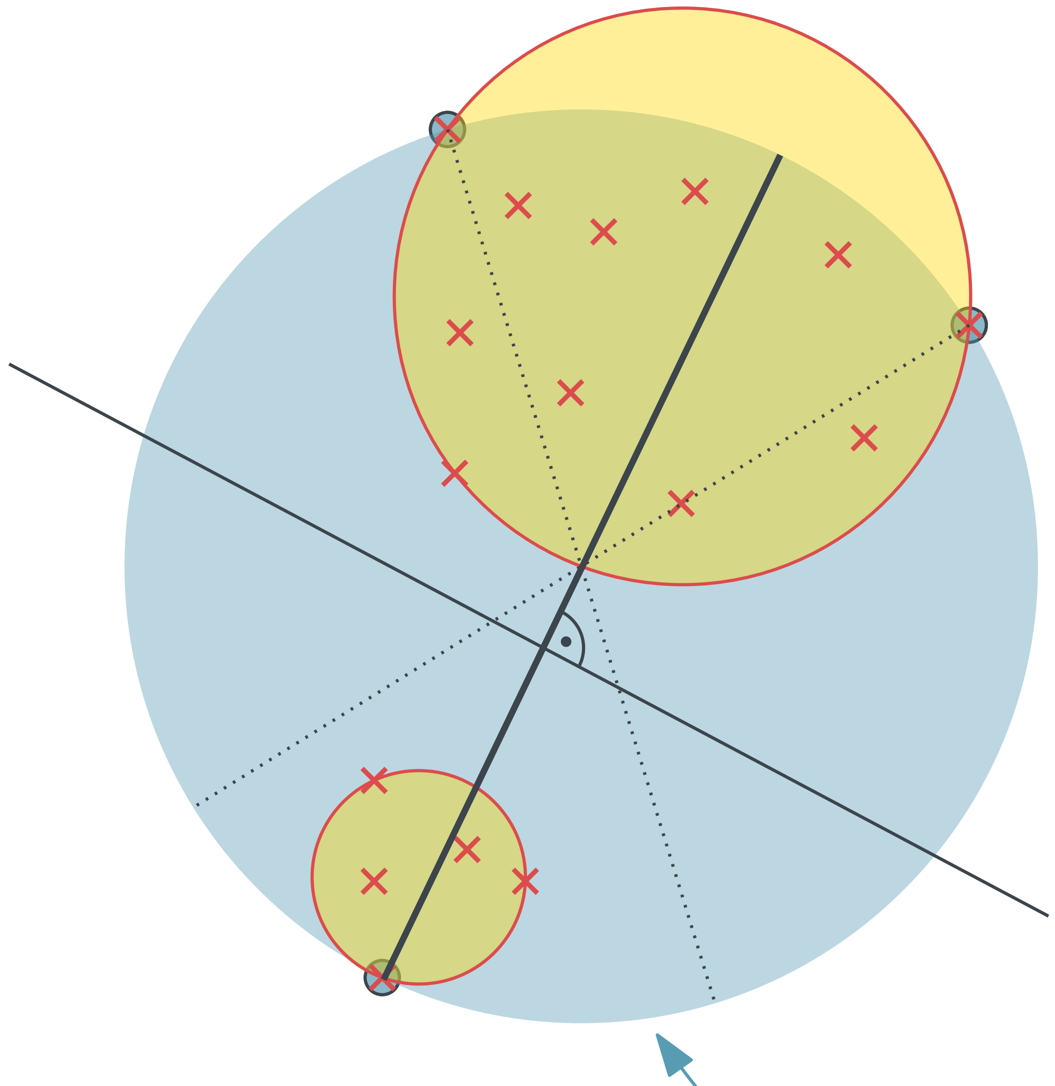
Minimum Enclosing Disk (MED)

Lemmas



Minimum Enclosing Disk (MED)

Lemmas

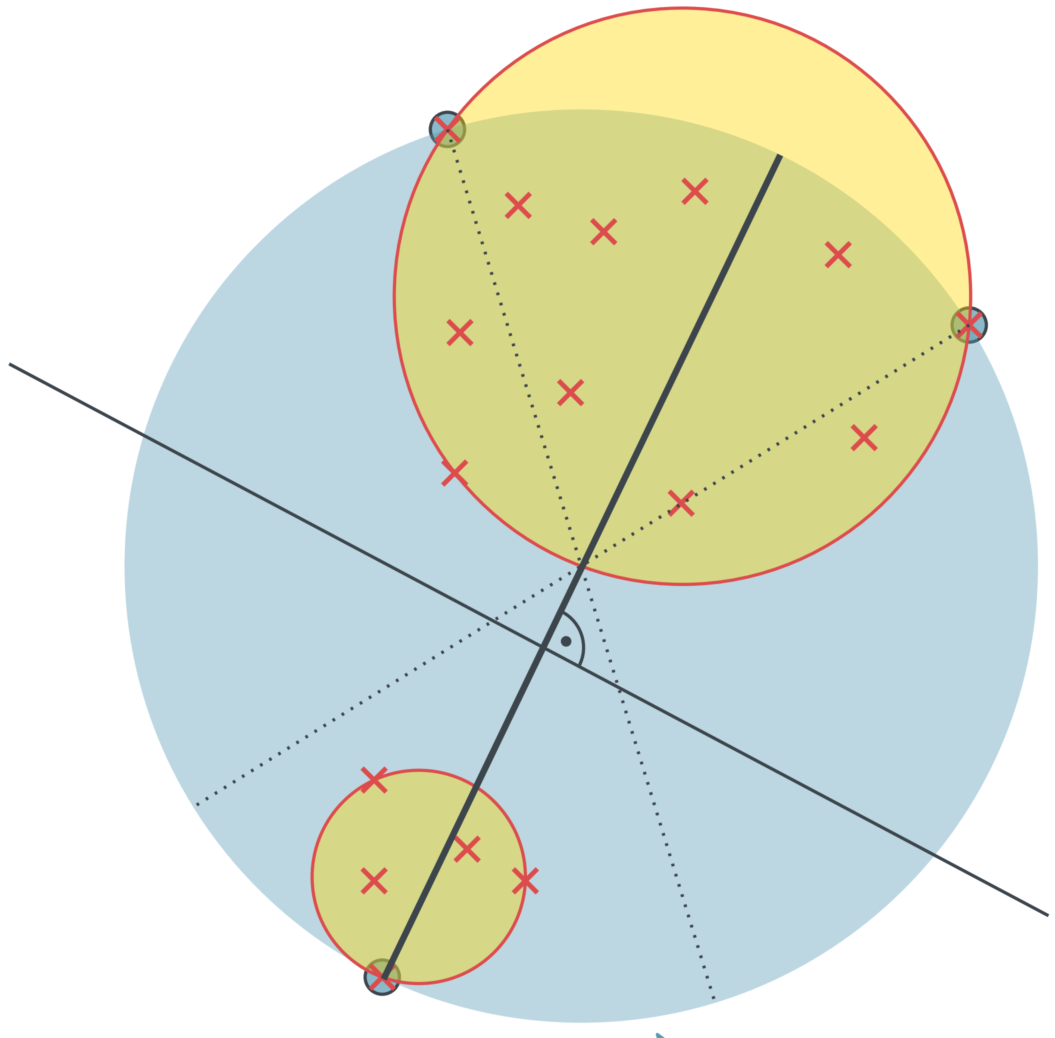


Minimum Enclosing Disk (MED)

Lemma

Given a 2-MinSumRadius instance, an orthogonal (——) to one of the diagonals (⋯⋯) separates the clusters in an optimal solution.

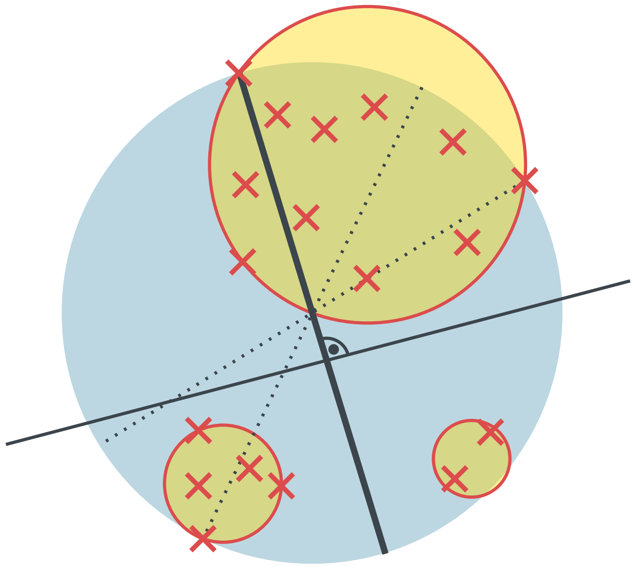
Lemmas



Minimum Enclosing Disk (MED)

Lemma

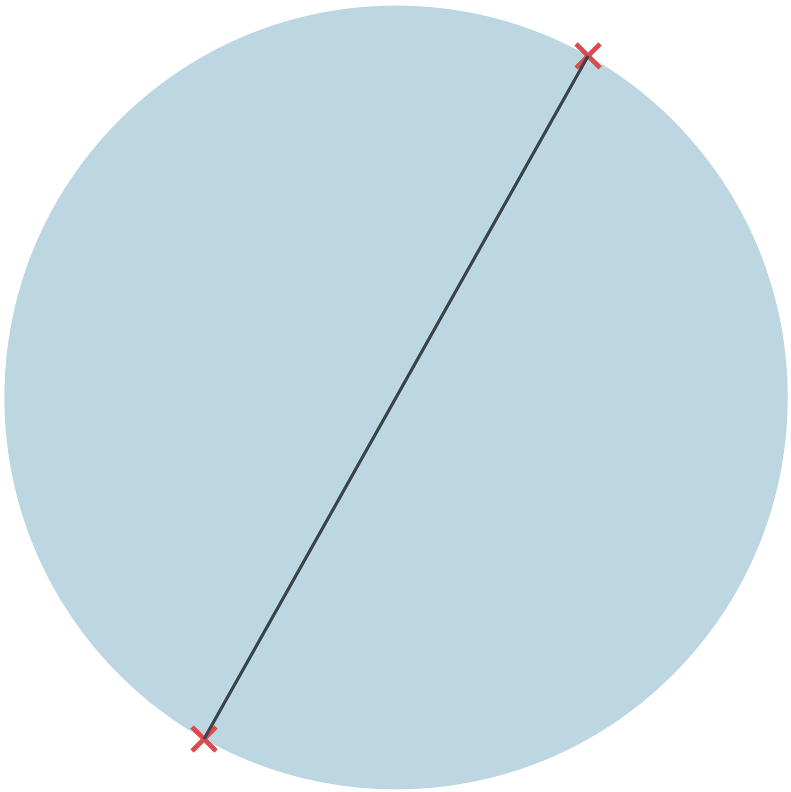
Given a ~~2~~ 3-MinSumRadius instance, an orthogonal (——) to one of the diagonals (⋯⋯) separates ~~the clusters~~ one cluster from the other two in an optimal solution.



Proof Sketch

$$k = 2$$

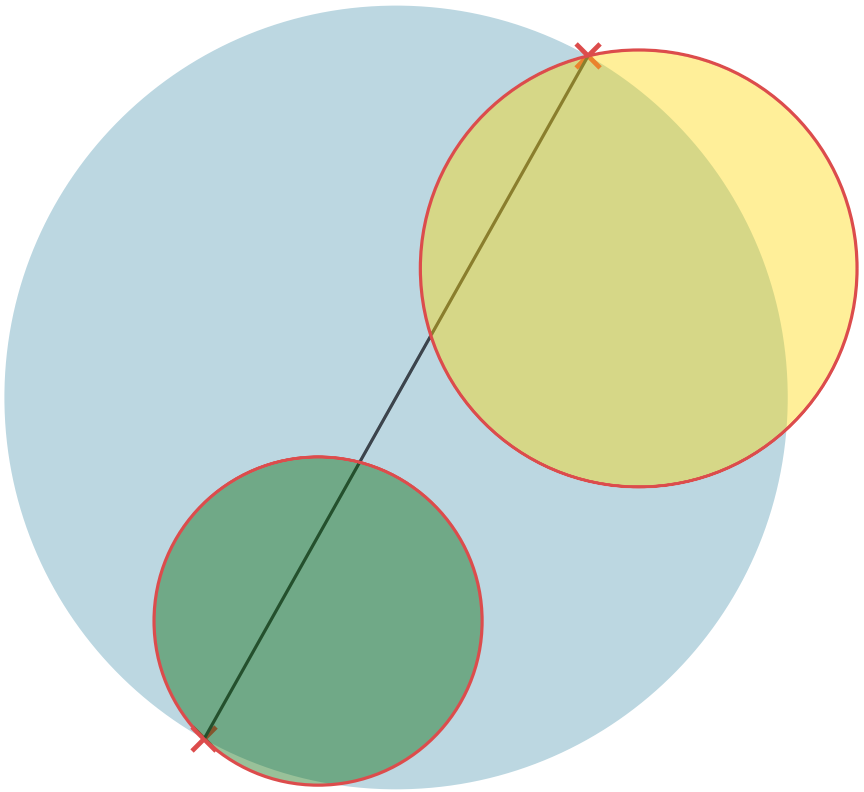
Case 1: two points define MED



Proof Sketch

$$k = 2$$

Case 1: two points define MED

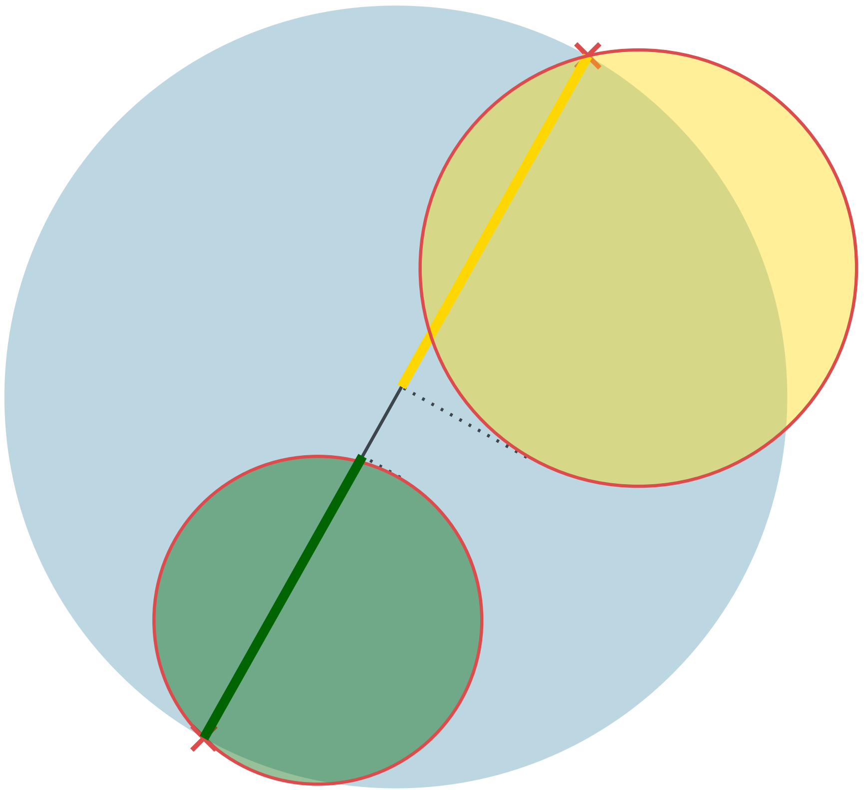


$\rightarrow$ assume points are in different clusters

Proof Sketch

$$k = 2$$

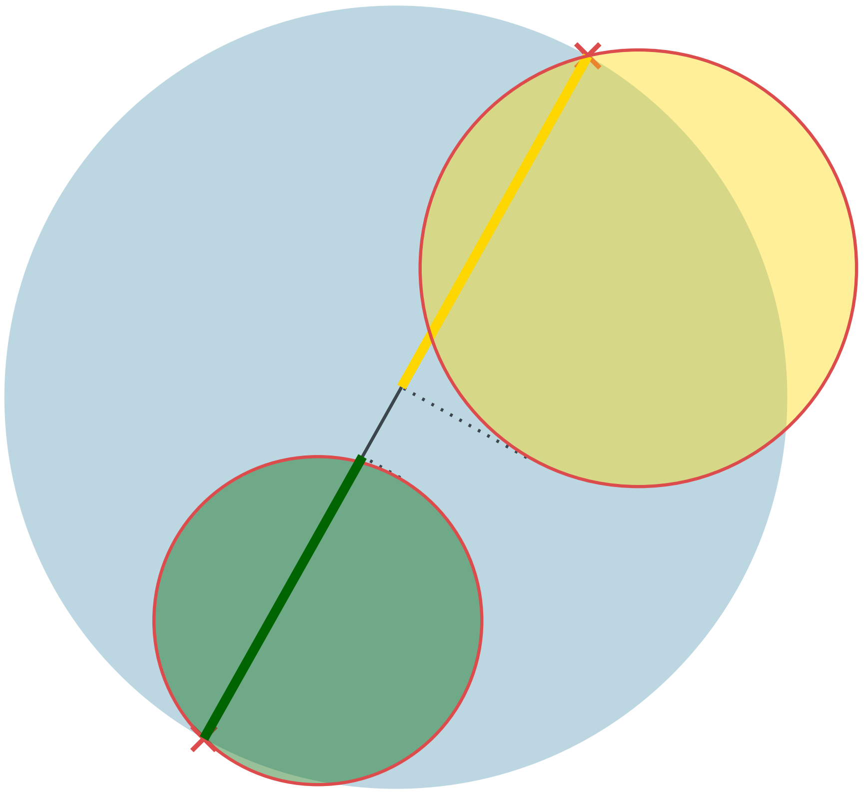
Case 1: two points define MED



Proof Sketch

$$k = 2$$

Case 1: two points define MED

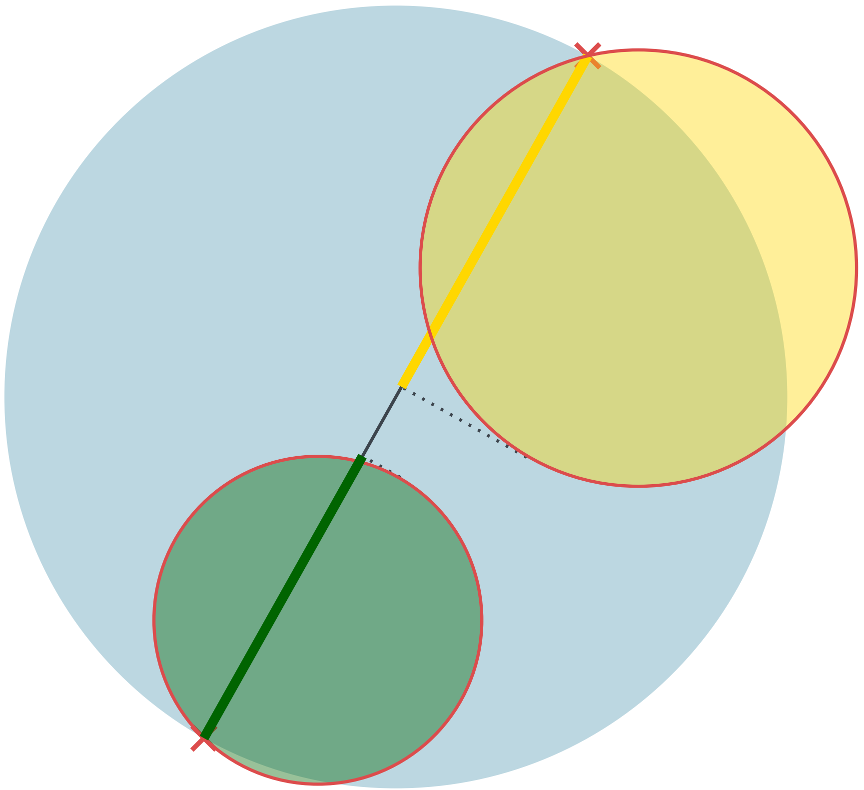


/ and / intersect

Proof Sketch

$k = 2$

Case 1: two points define MED



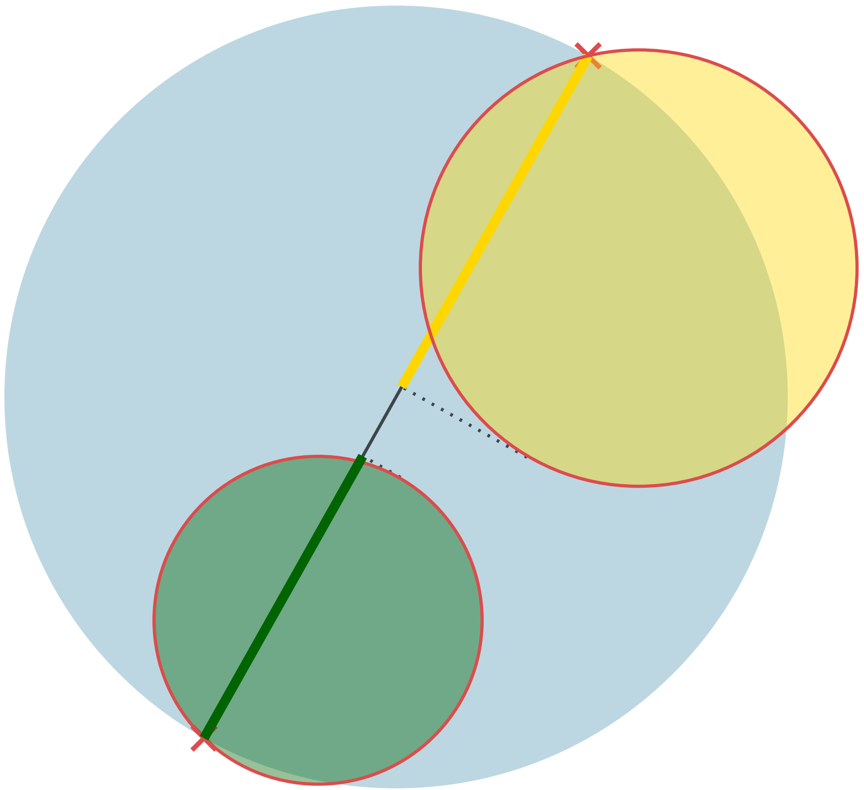
/ and / intersect

$\implies$ len(—) $+$ len(—) $>$ len(—) $\lightning$

Proof Sketch

$$k = 2$$

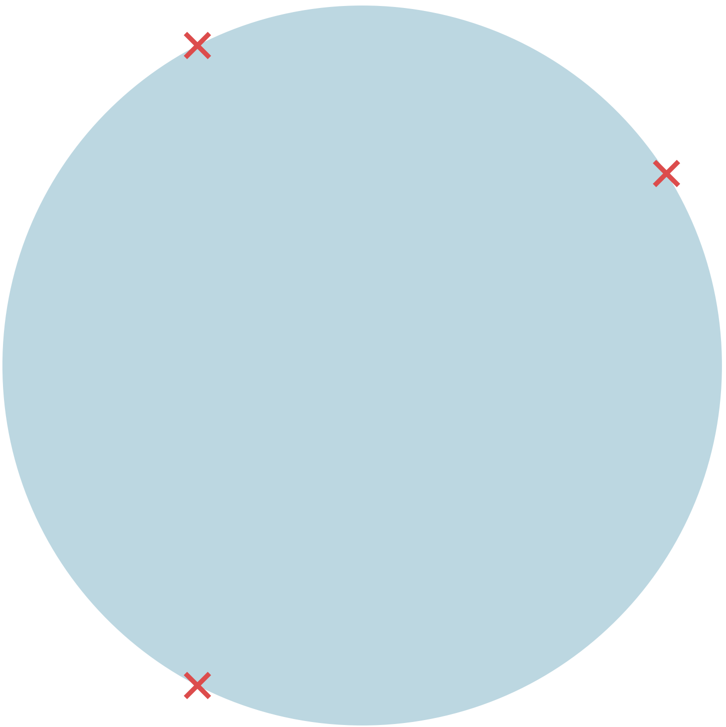
Case 1: two points define MED



/ and / intersect

$\Longrightarrow$ len(—) + len(—) > len(—) ↯

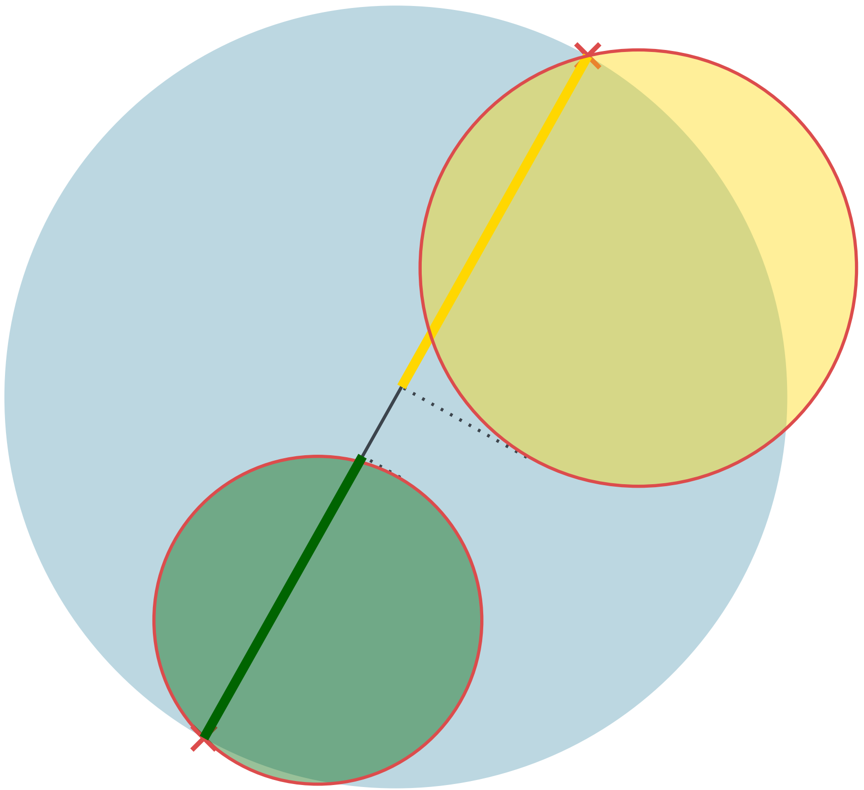
Case 2: three points define MED



Proof Sketch

$k = 2$

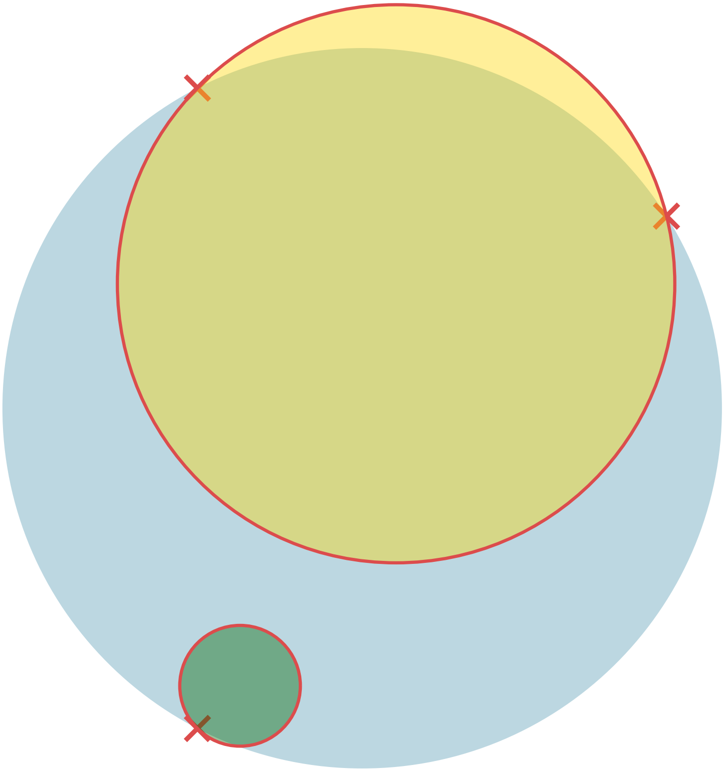
Case 1: two points define MED



/ and / intersect

$\implies$ len(—) $+$ len(—) $>$ len(—) ↯

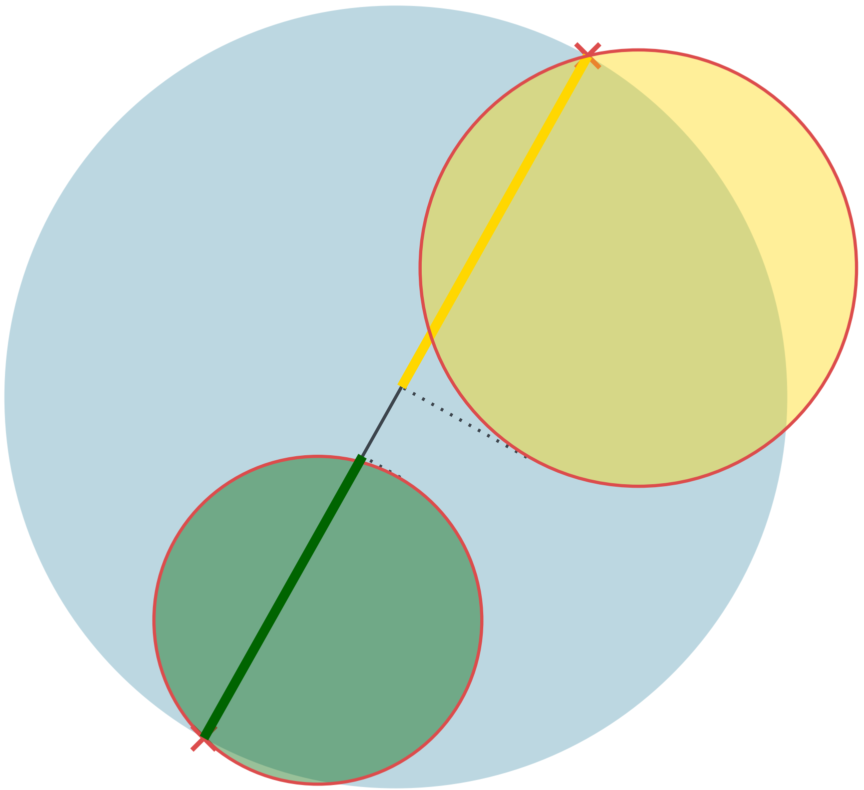
Case 2: three points define MED



Proof Sketch

$k = 2$

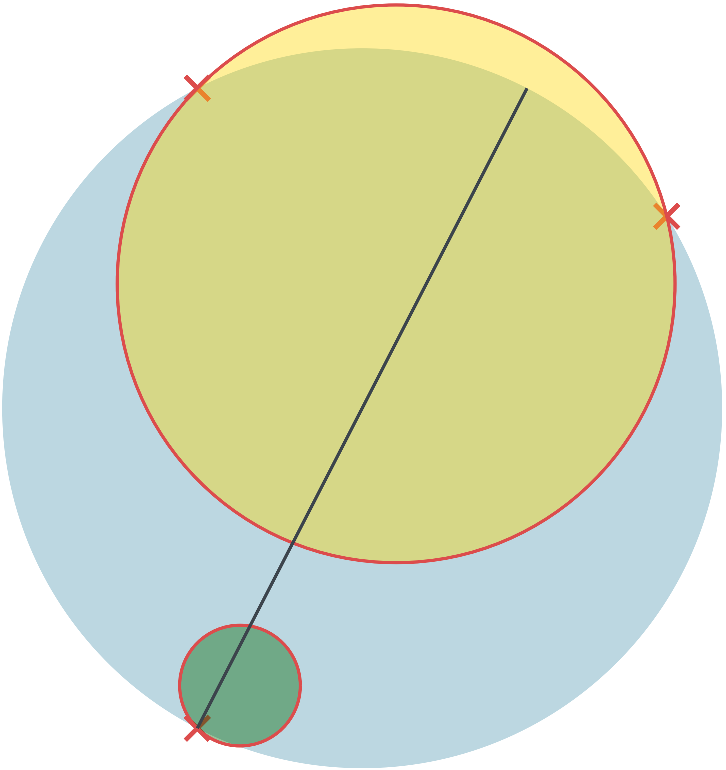
Case 1: two points define MED



/ and / intersect

$\implies$ len(—) + len(—) > len(—) ↯

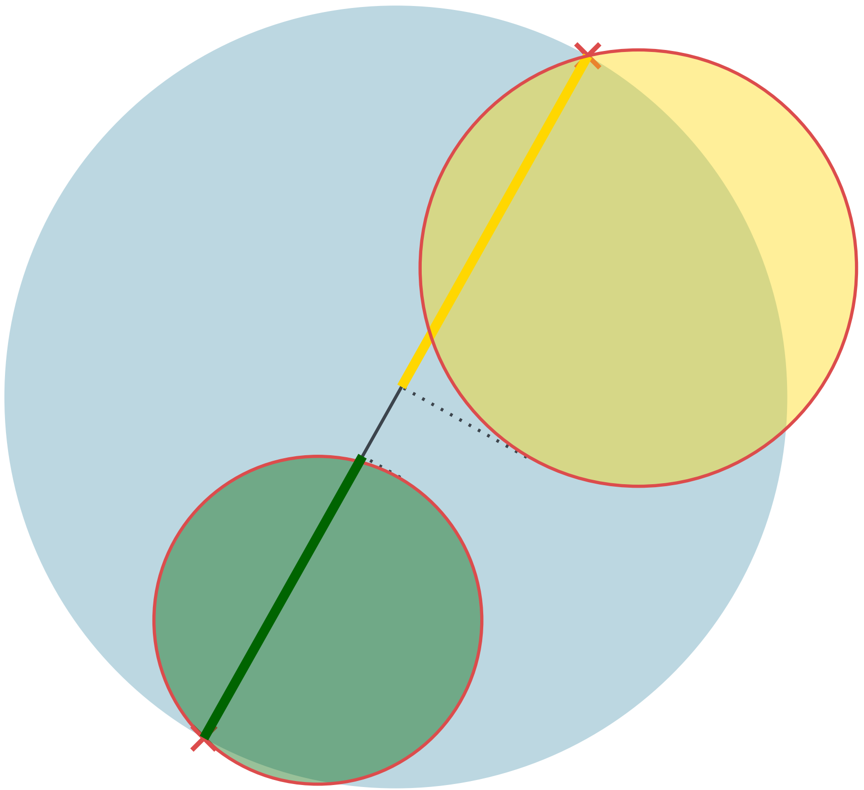
Case 2: three points define MED



Proof Sketch

$k = 2$

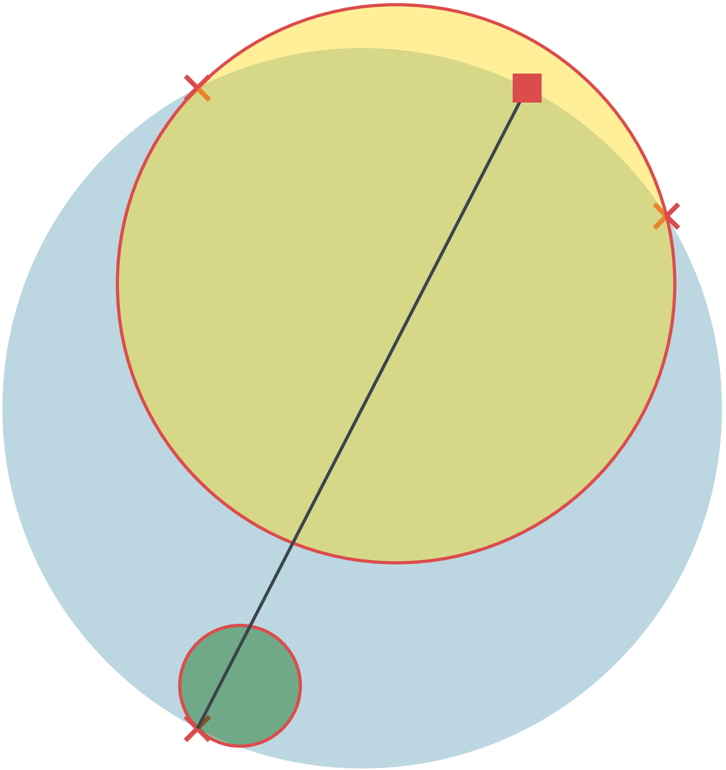
Case 1: two points define MED



/ and / intersect

$\implies$ len(—) $+$ len(—) $>$ len(—) ↯

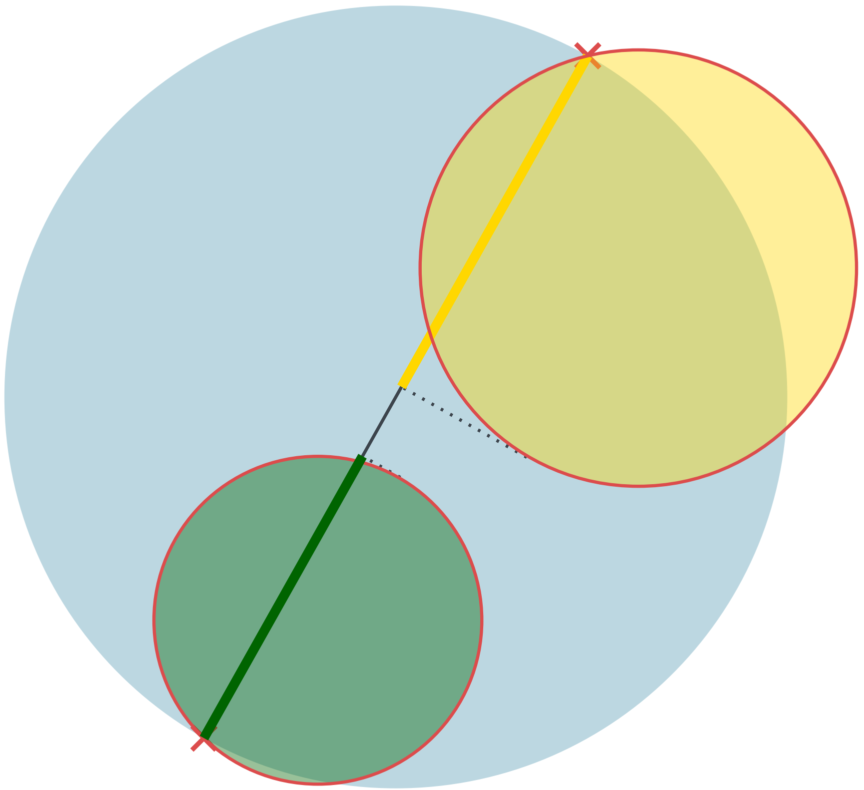
Case 2: three points define MED



Proof Sketch

$k = 2$

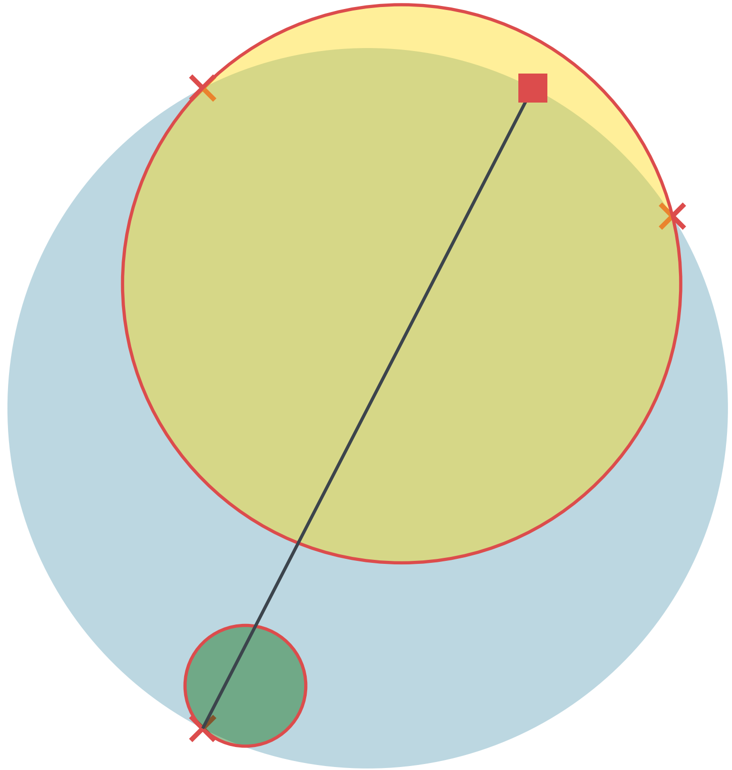
Case 1: two points define MED



/ and / intersect

$\implies$ len(—) + len(—) > len(—) ↯

Case 2: three points define MED



■ ∈ ○ $\implies$ reduction to **Case 1**

Algorithms

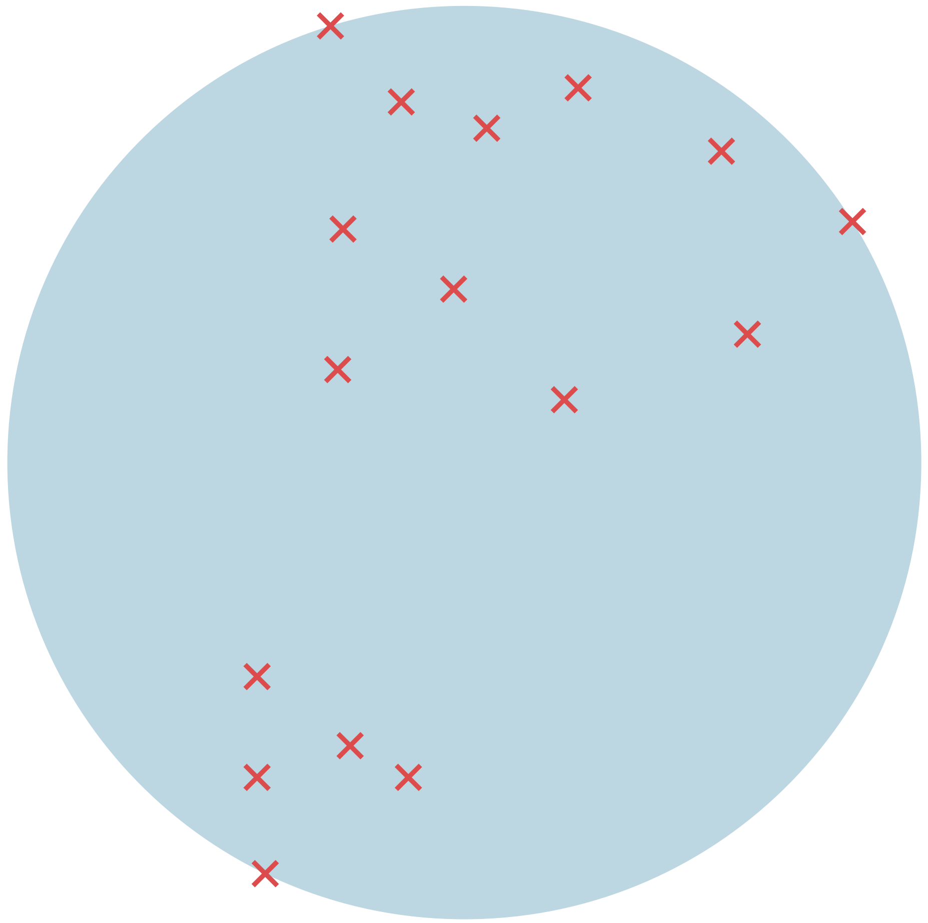
Algorithm

$k = 2$

Running time:

Algorithm

$k = 2$

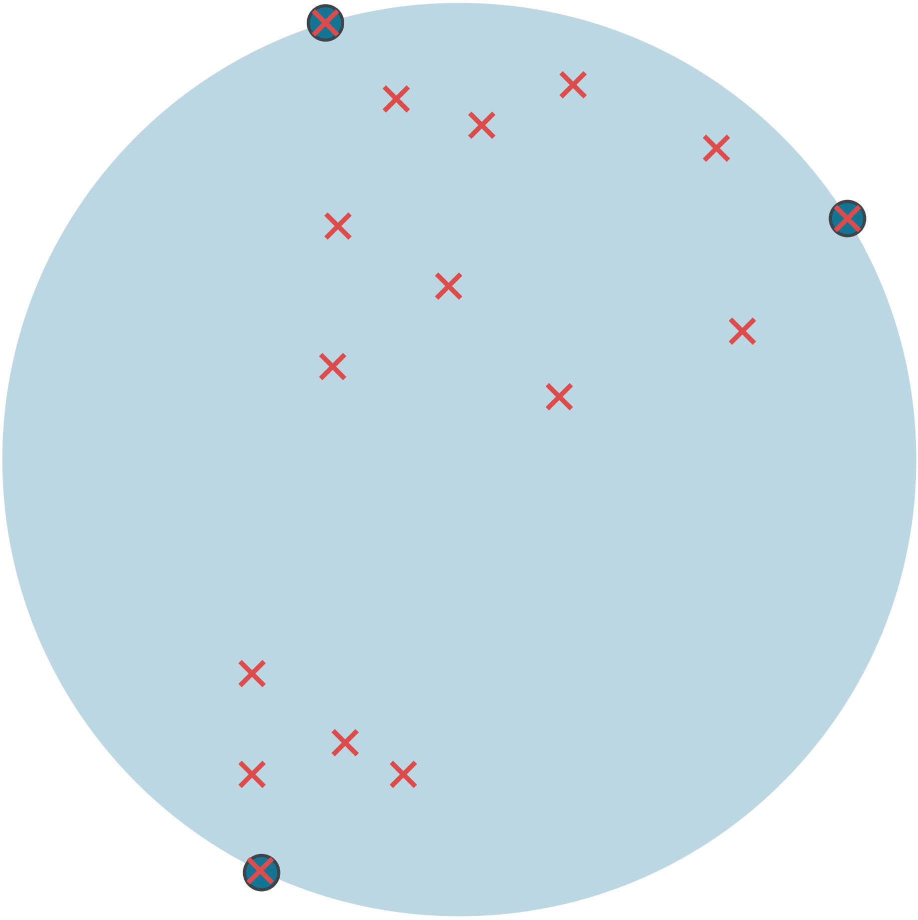


Running time:

$O(n)$

Algorithm

$k = 2$

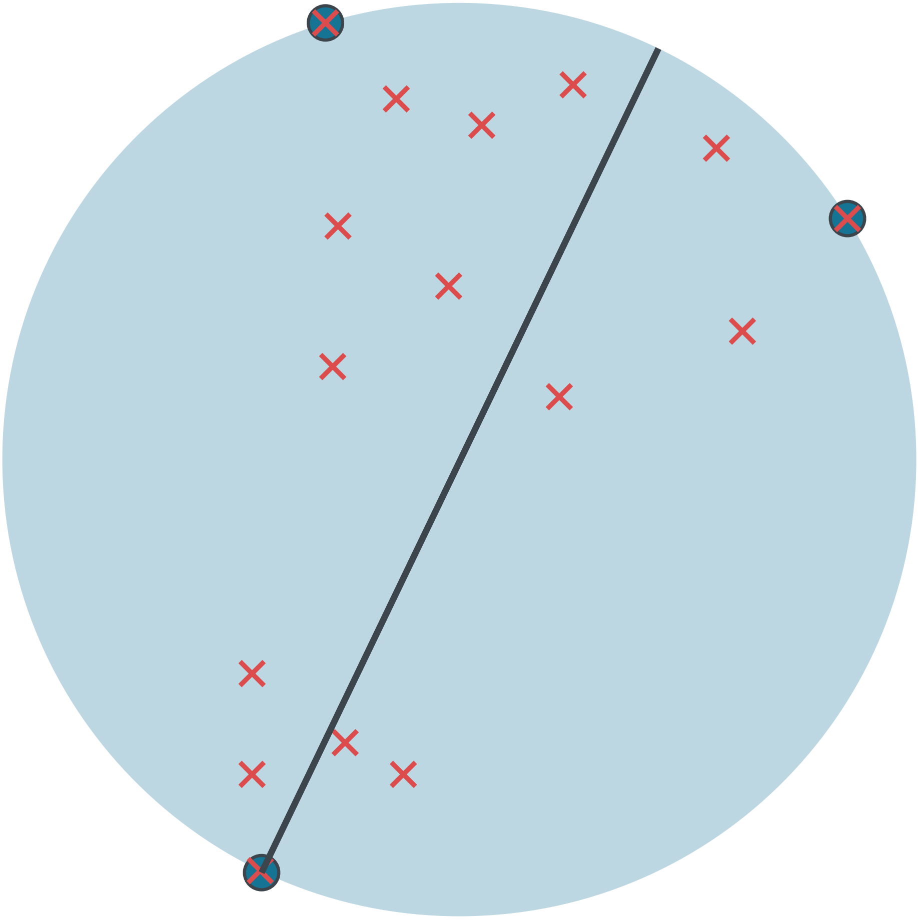


Running time:

$O(n)$

Algorithm

$k = 2$

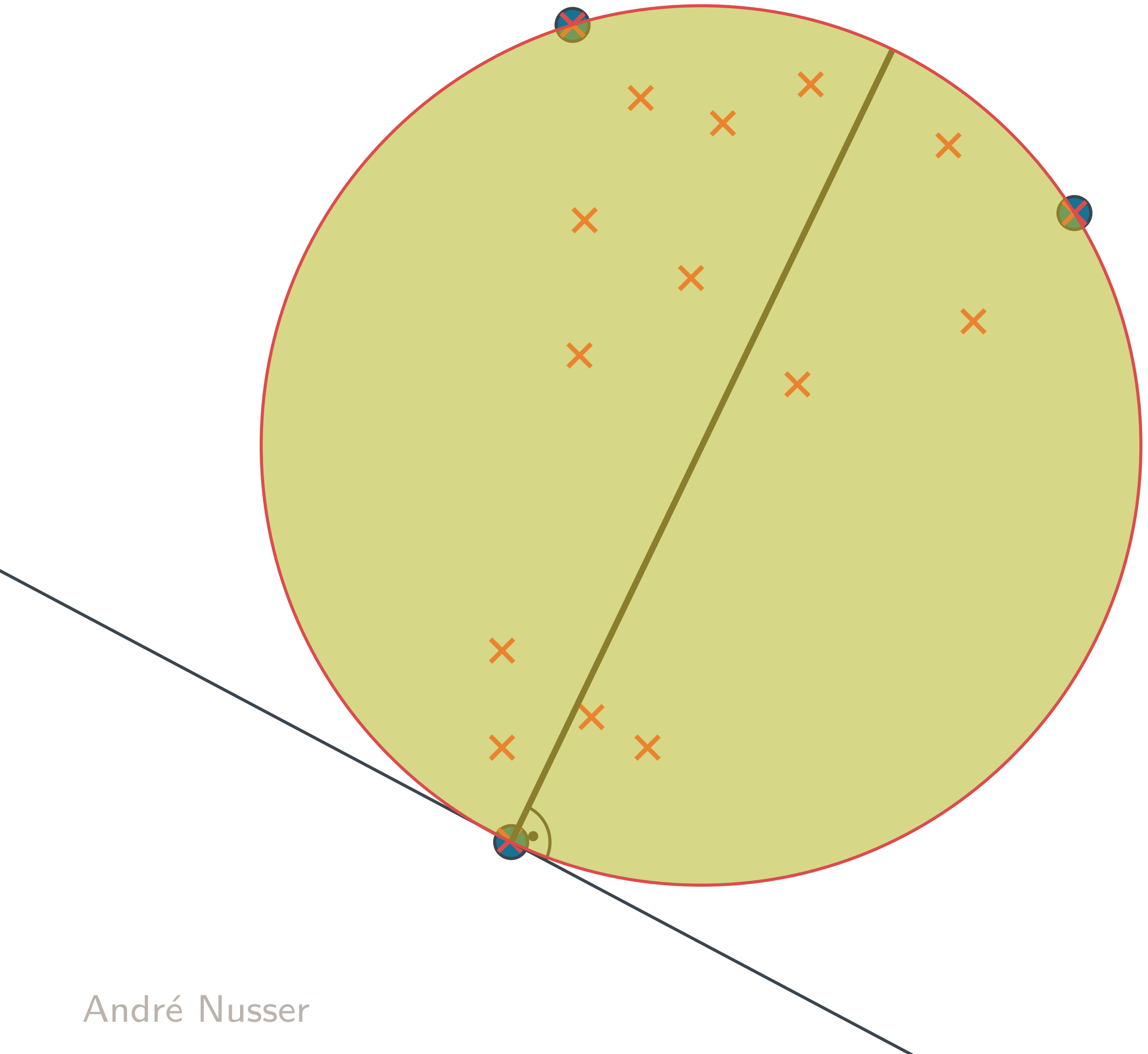


Running time:

$O(n) + 3 \cdot$

Algorithm

$$k = 2$$

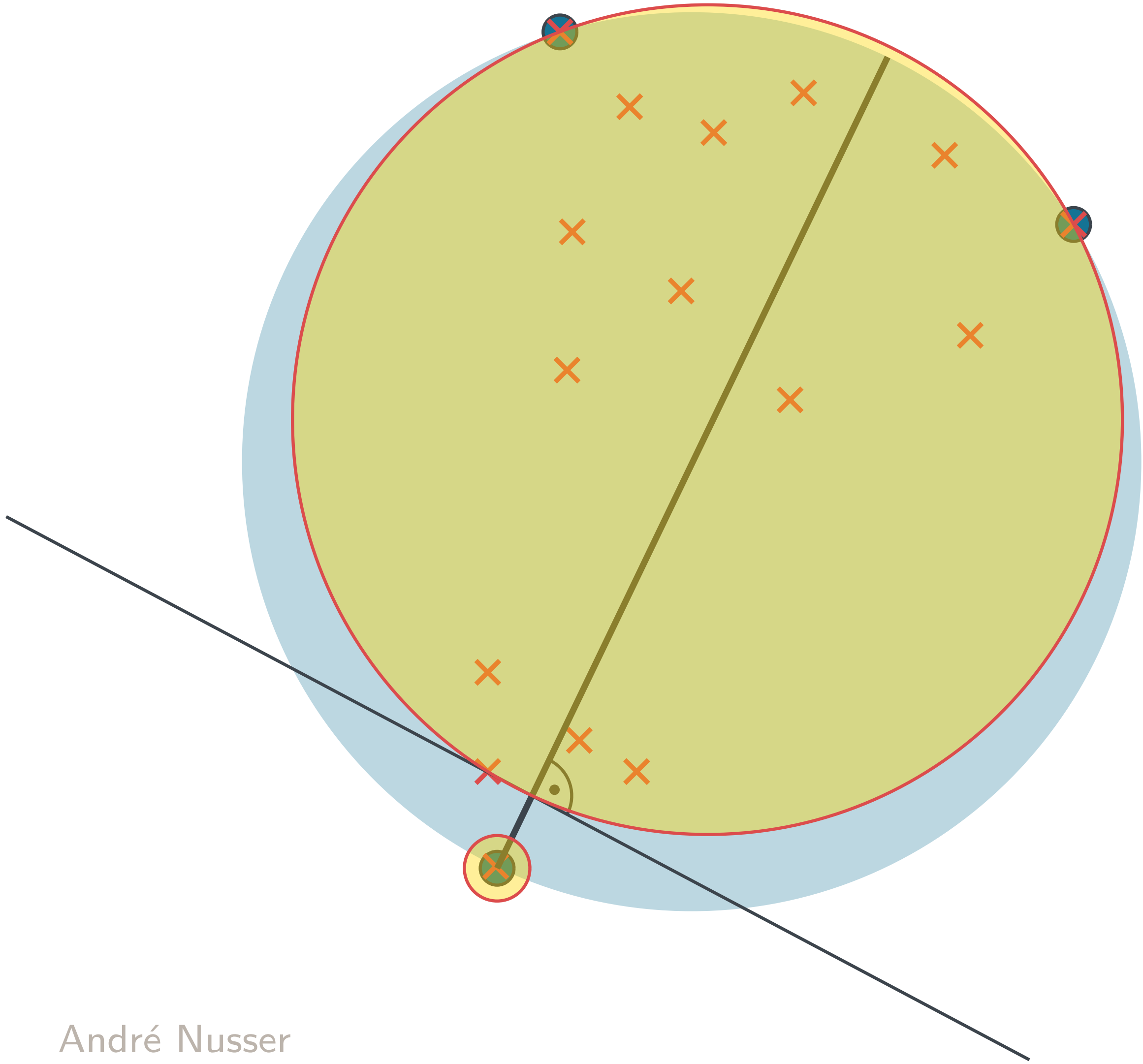


Running time:

$O(n) + 3 \cdot$

Algorithm

$k = 2$

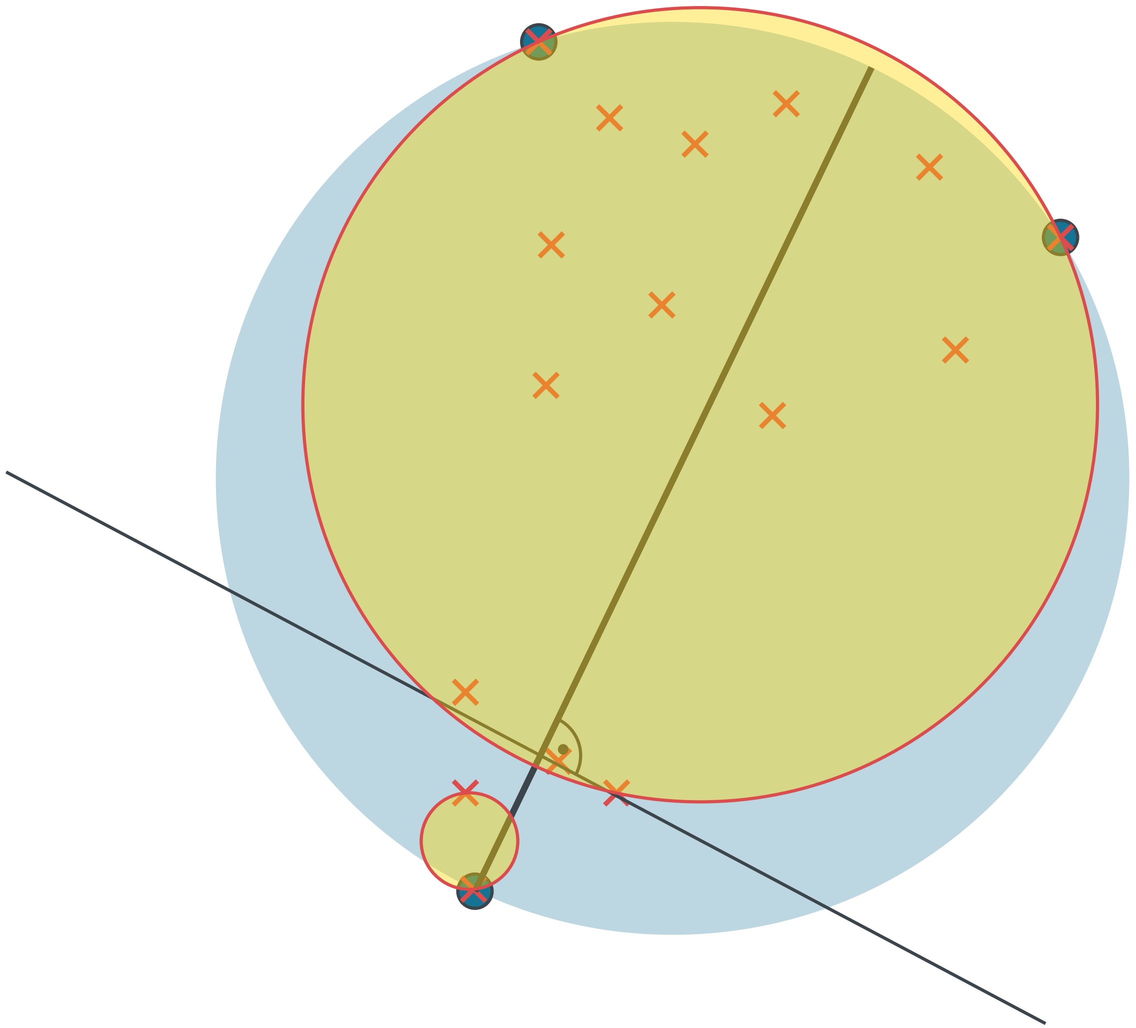


Running time:

$O(n) + 3 \cdot$

Algorithm

$k = 2$

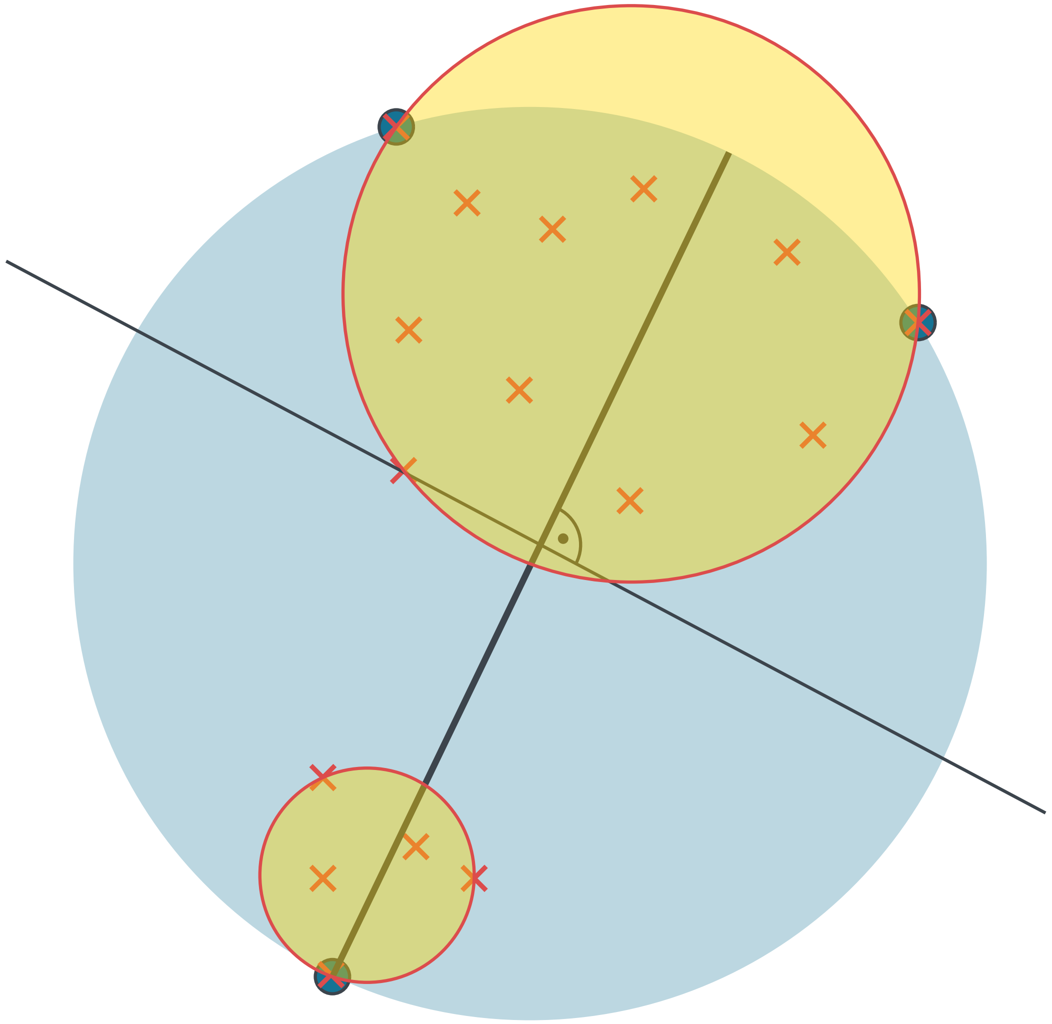


Running time:

$O(n) + 3 \cdot$

Algorithm

$k = 2$

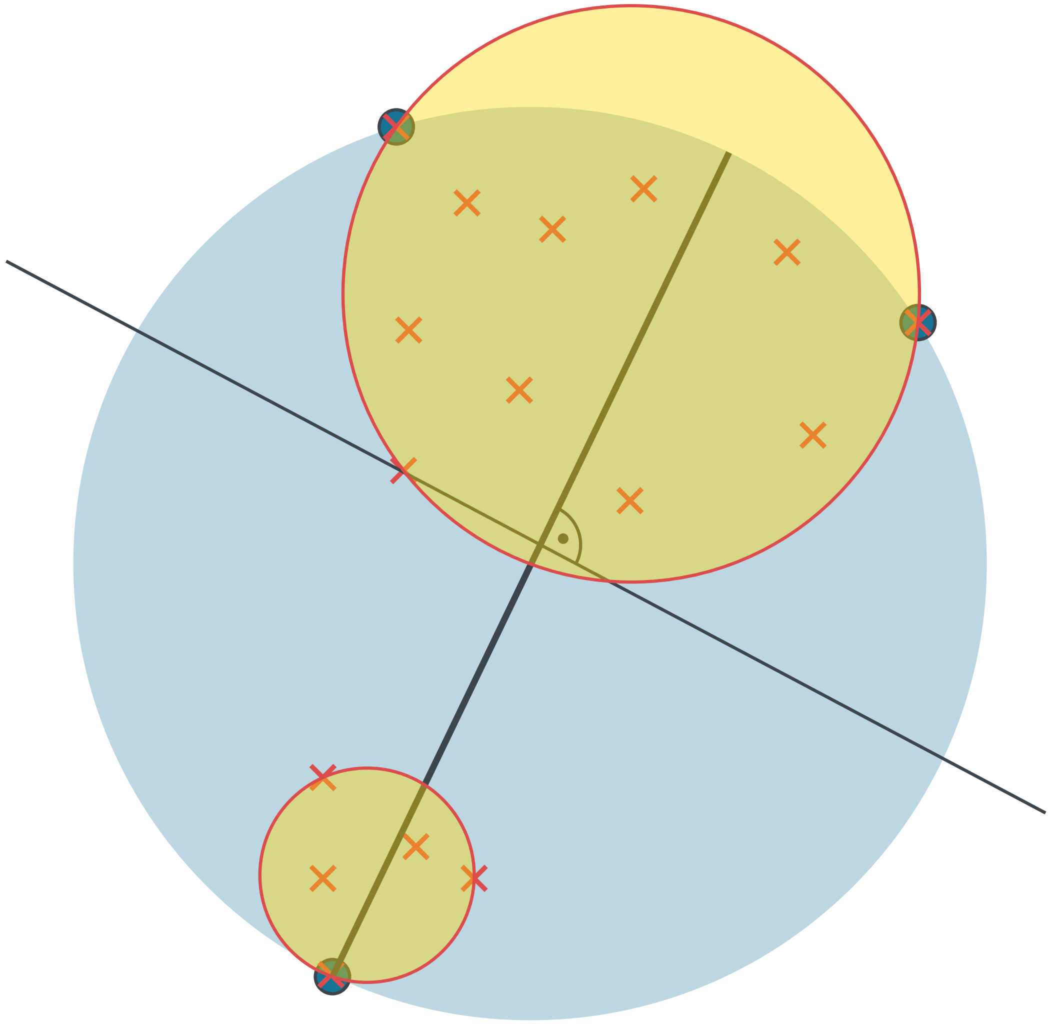


Running time:

$O(n) \; + \; 3 \cdot n \cdot$

Algorithm

$k = 2$



*

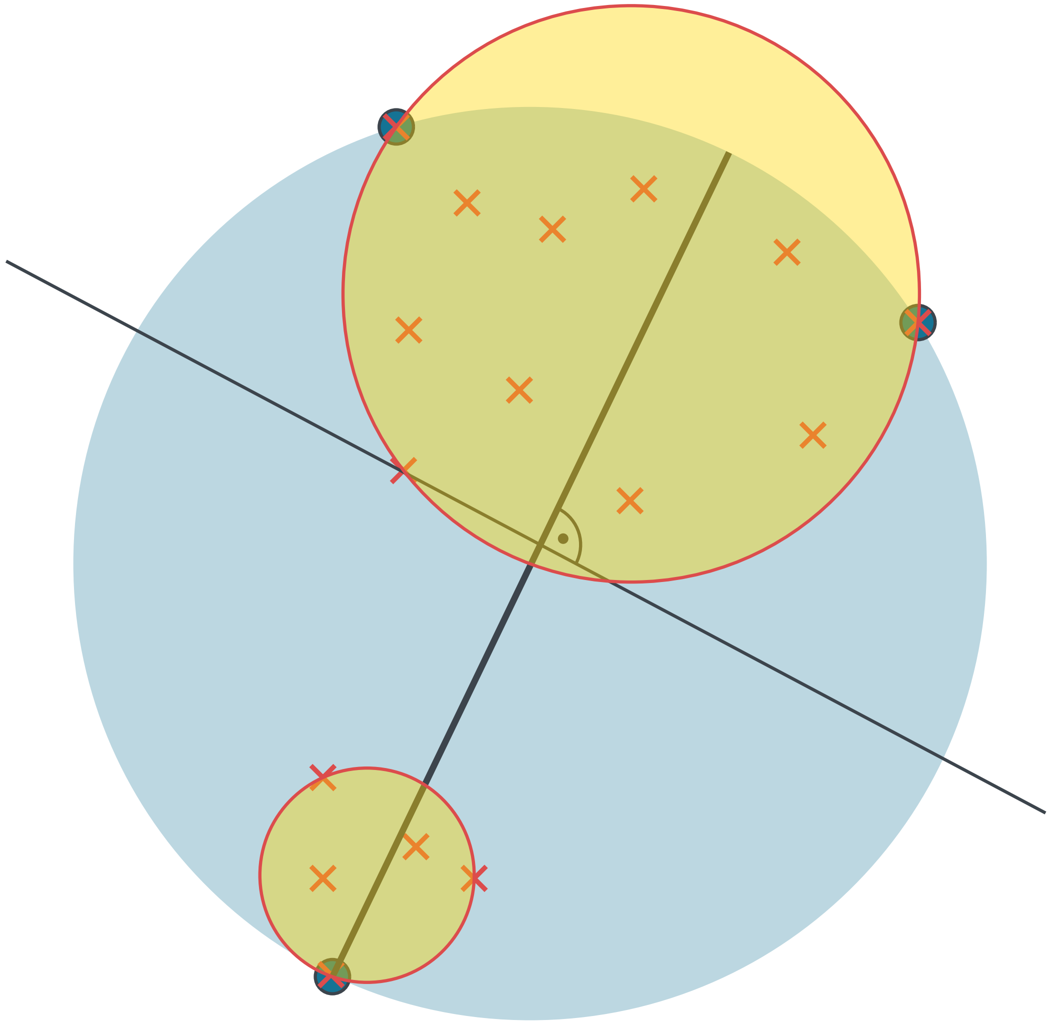
Running time:

$O(n) + 3 \cdot n \cdot O(\log^2 n)$

* We omit $\log \log$ factors here

Algorithm

$k = 2$



*

Running time:

$$O(n) \; + \; 3 \cdot n \cdot O(\log^2 n)$$
$$=$$
$$O(n \cdot \log^2 n)$$

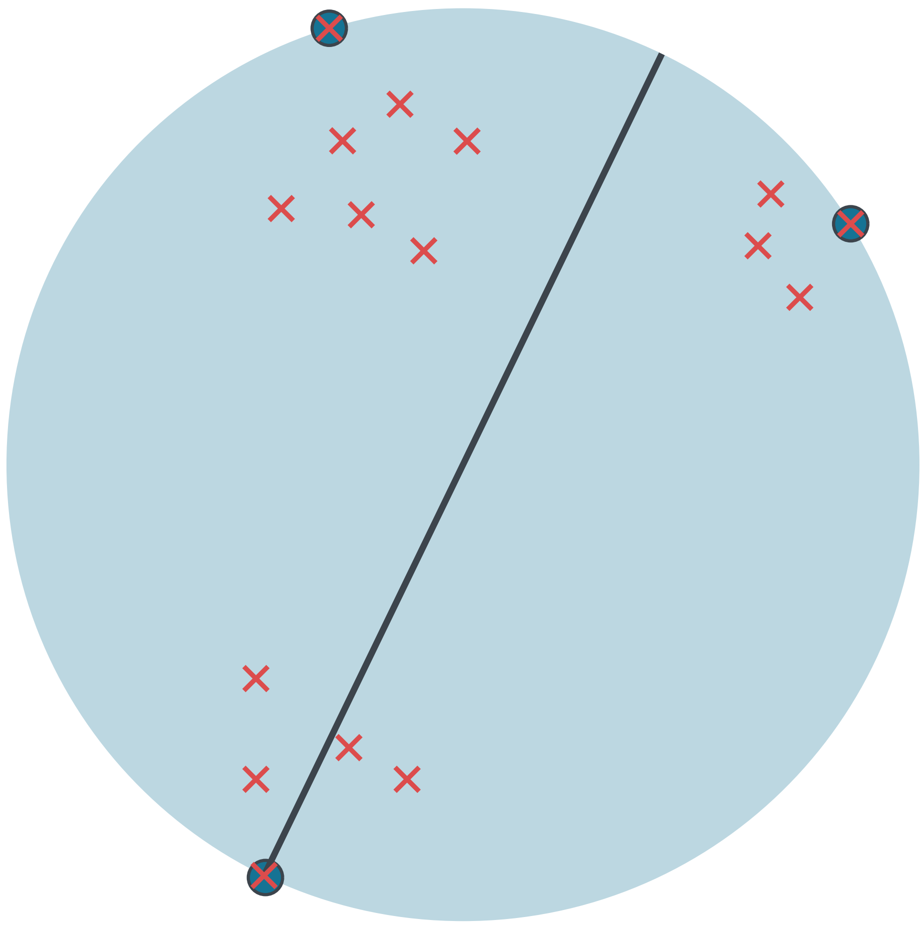
* We omit $\log \log$ factors here

Algorithm

$$k = 3$$

Algorithm

$k = 3$

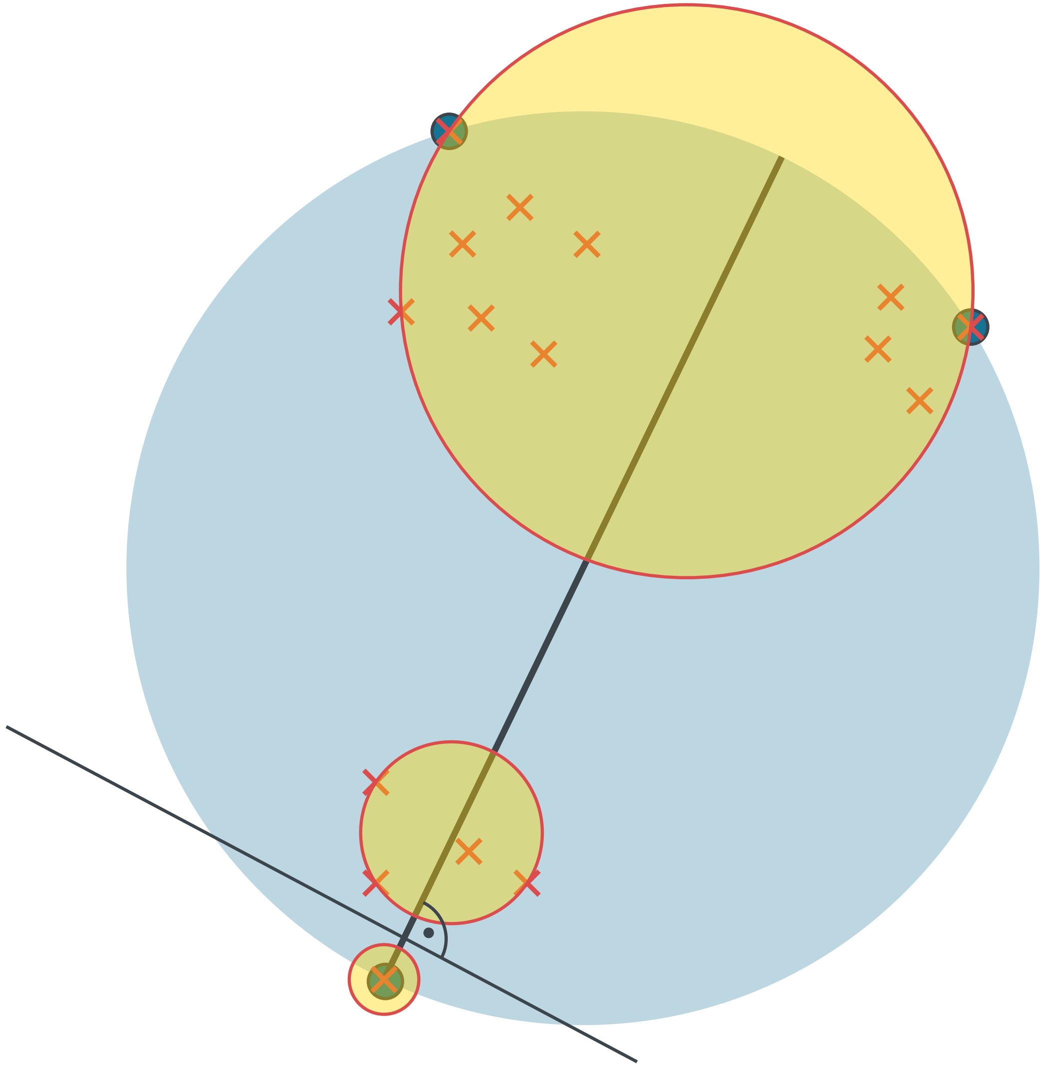


Running time:

$O(n) + 3 \cdot n \cdot$

Algorithm

$k = 3$



*

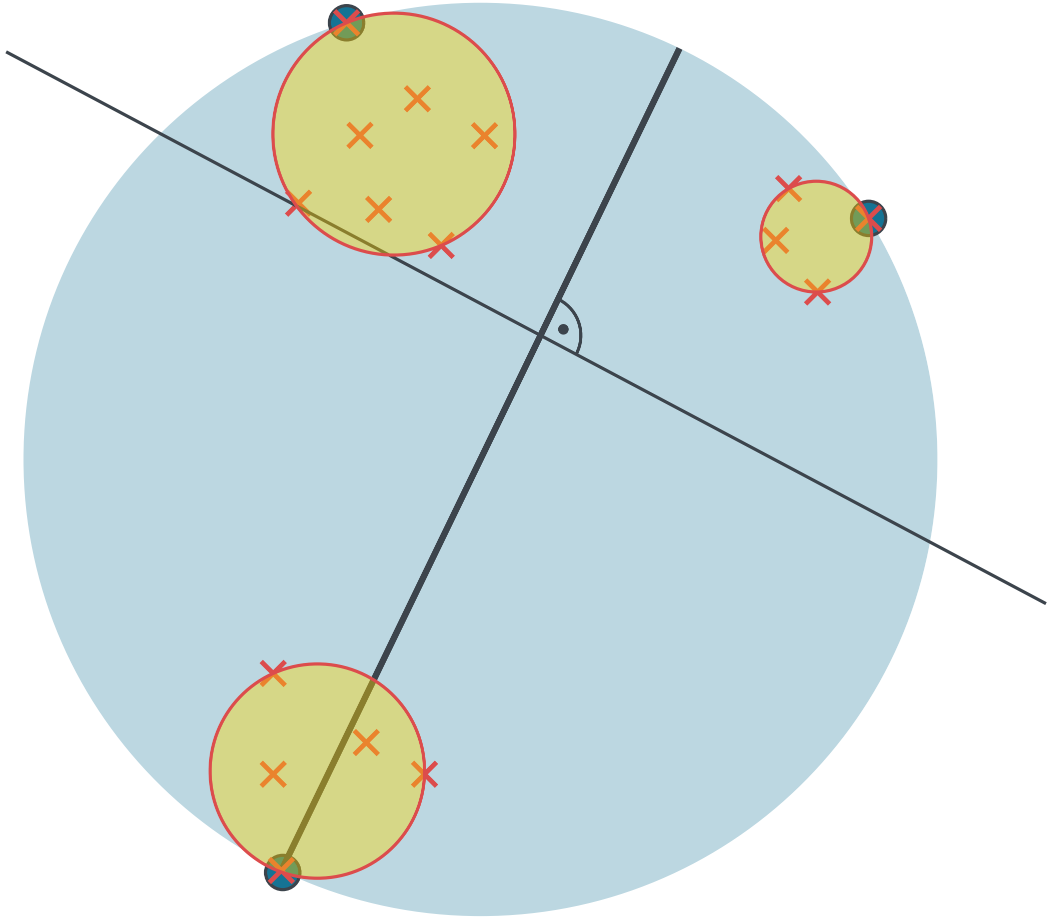
Running time:

$$O(n) \ + \ 3 \cdot n \cdot (O(n) + O(n \log^2 n))$$

* We omit $\log \log$ factors here

Algorithm

$$k = 3$$



*

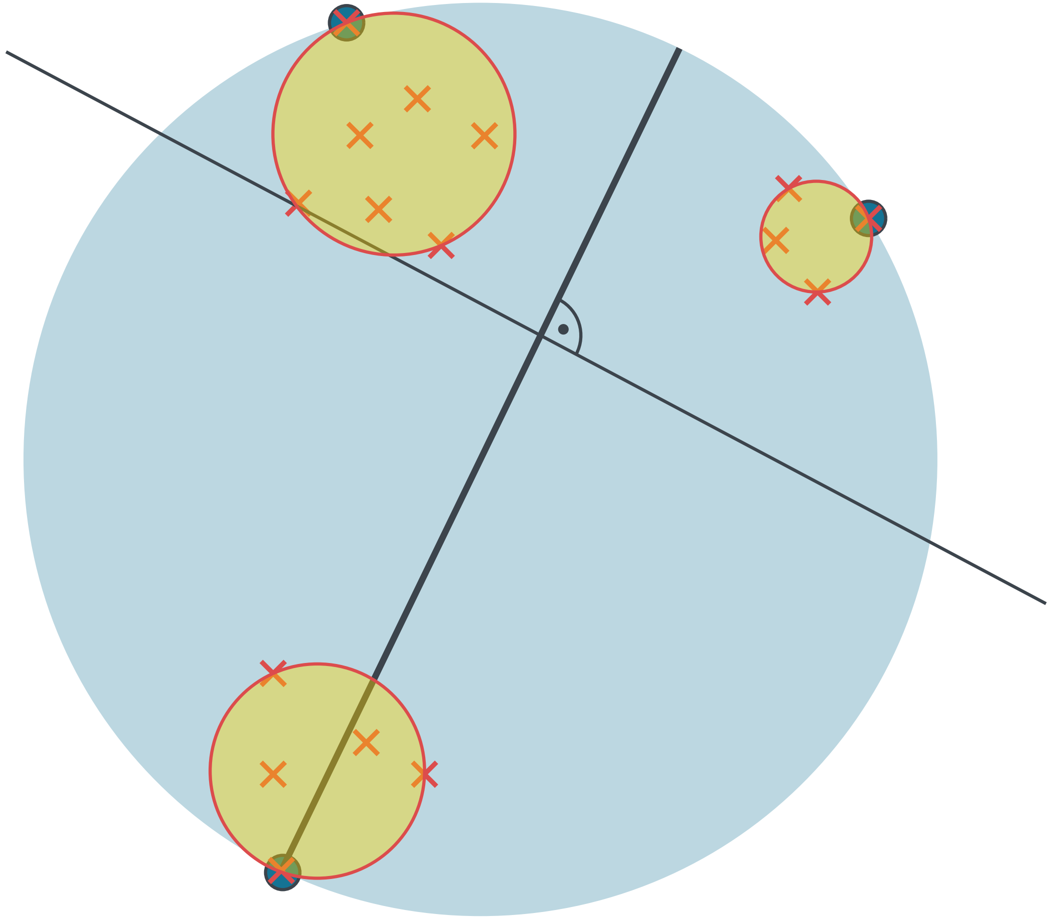
Running time:

$$O(n) \; + \; 3 \cdot n \cdot (O(n) + O(n \log^2 n))$$

* We omit $\log \log$ factors here

Algorithm

$k = 3$



*

Running time:

$$O(n) \; + \; 3 \cdot n \cdot (O(n) + O(n \log^2 n))$$
$$=$$
$$O(n^2 \cdot \log^2 n)$$

* We omit $\log\log$ factors here

Thank You!

Questions?

Unofficial Paper Soundtrack:

Full Circle
by **Half Moon Run**