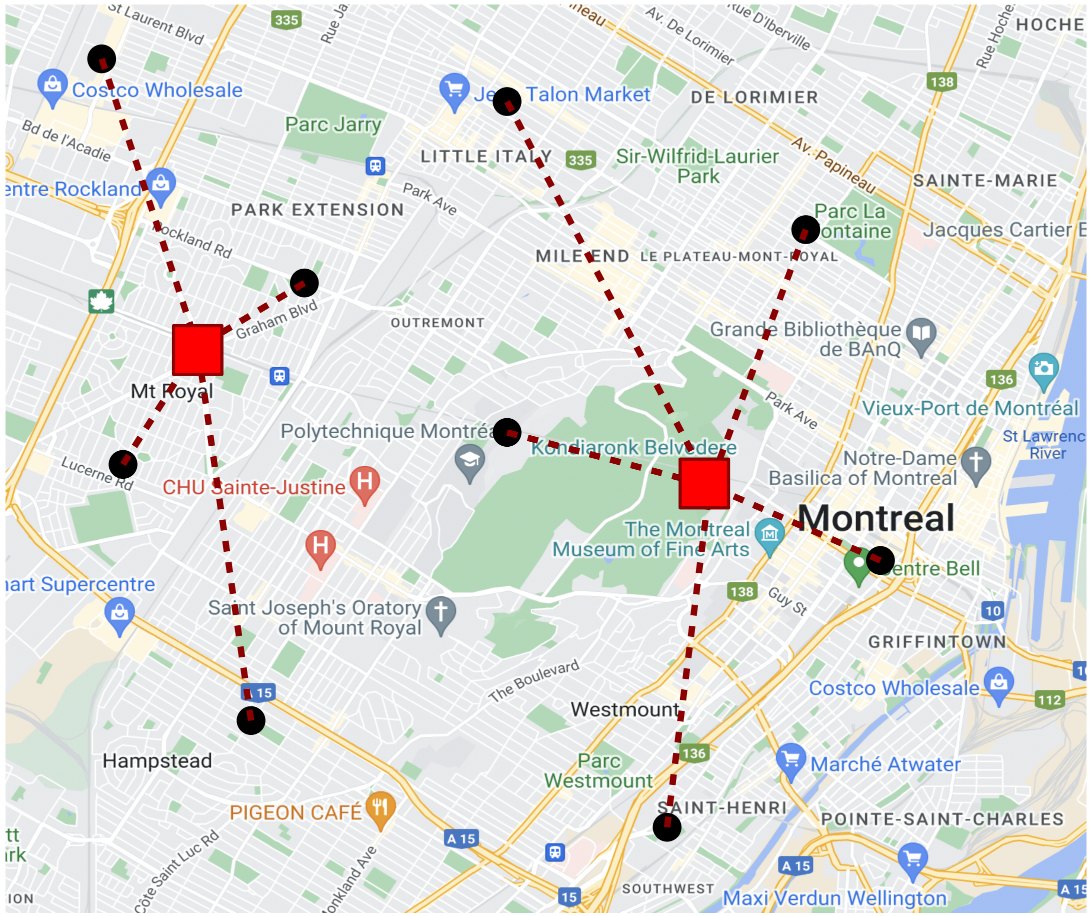
Parallel Line Centers with Guaranteed Separation

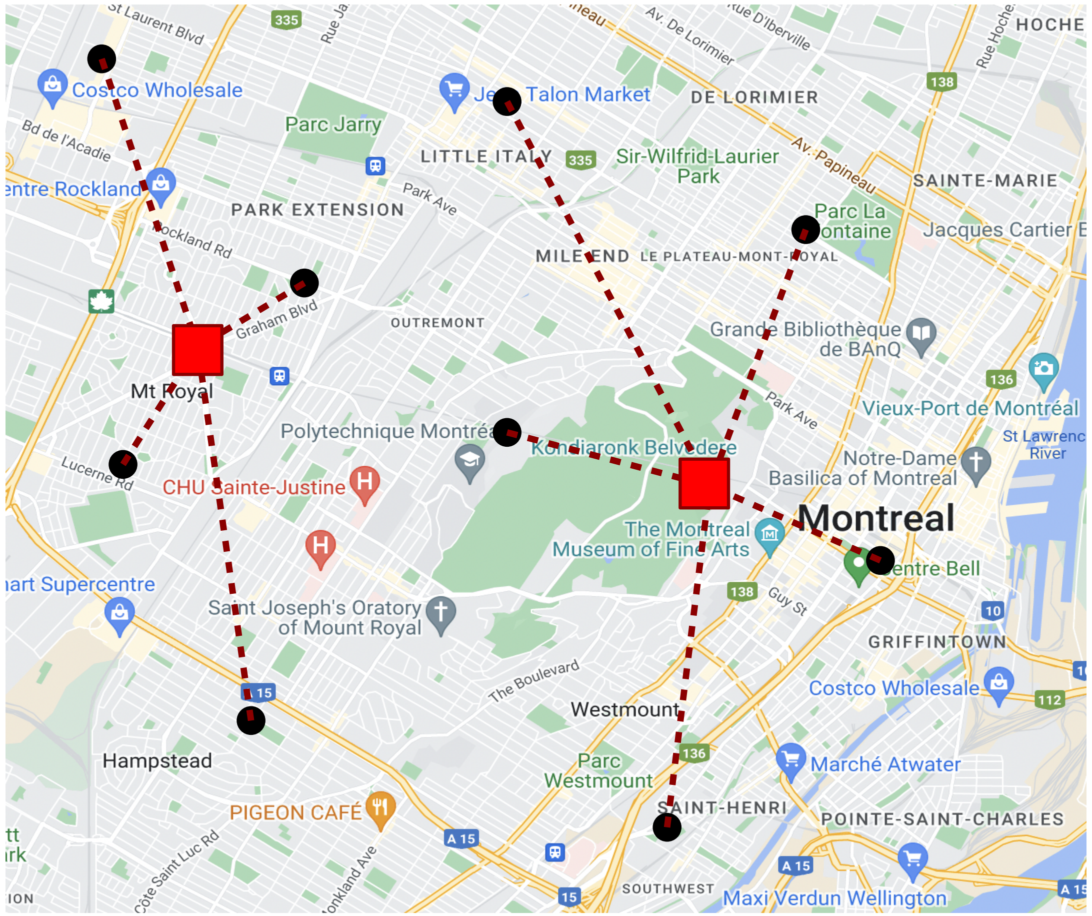
CCCG 2023
August 3

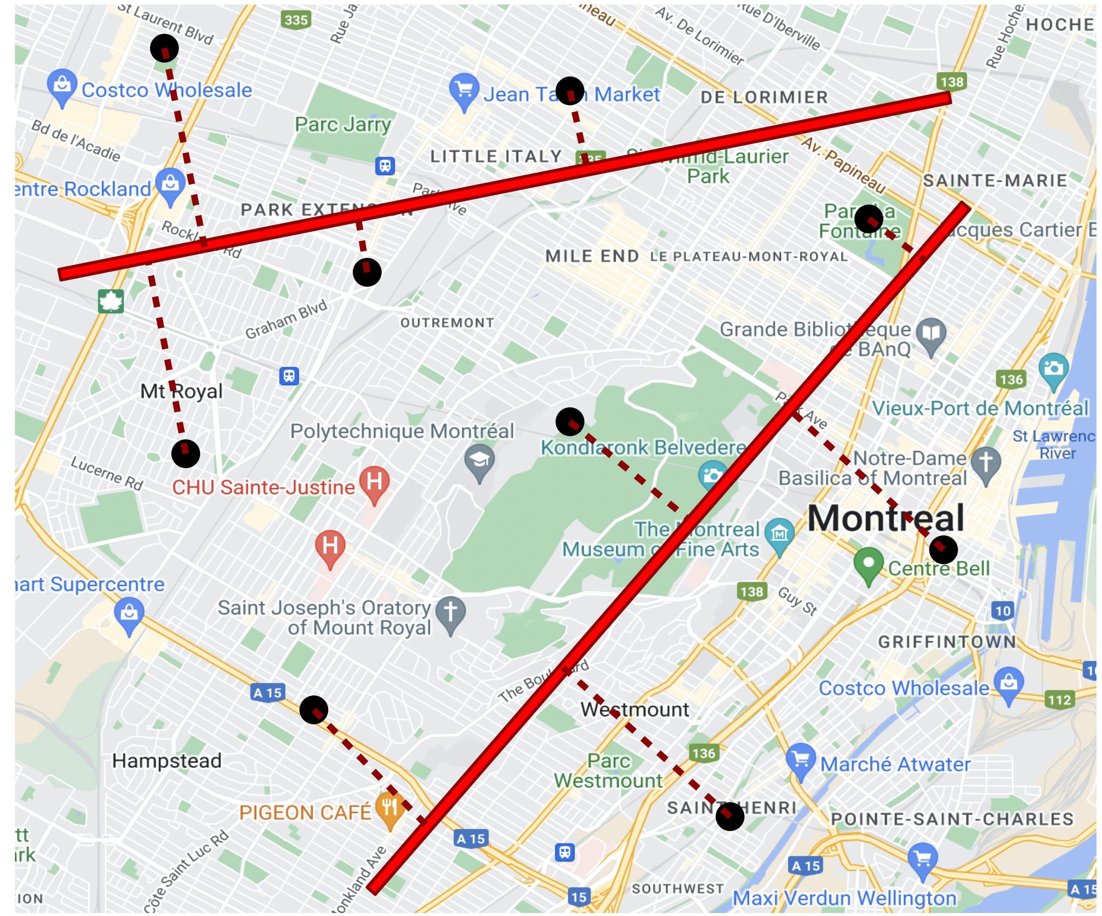
Chaeyoon Chung, Taehoon Ahn, Sang Won Bae, Hee-Kap Ahn

Facility-location problems

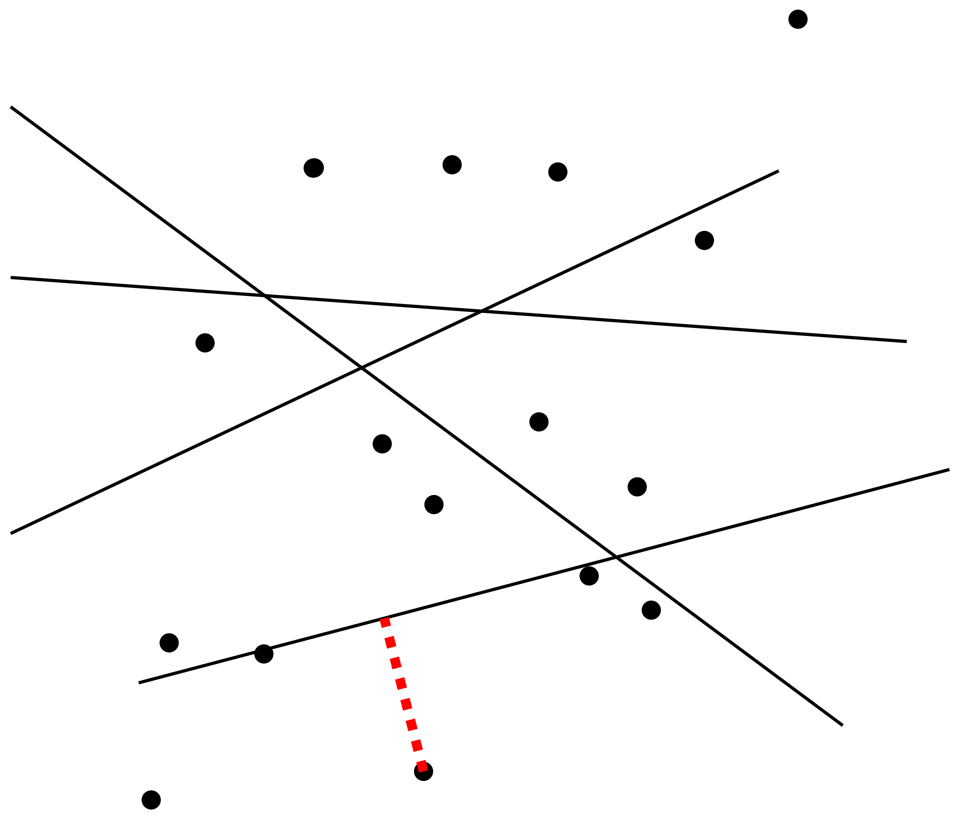


Facility-location problems



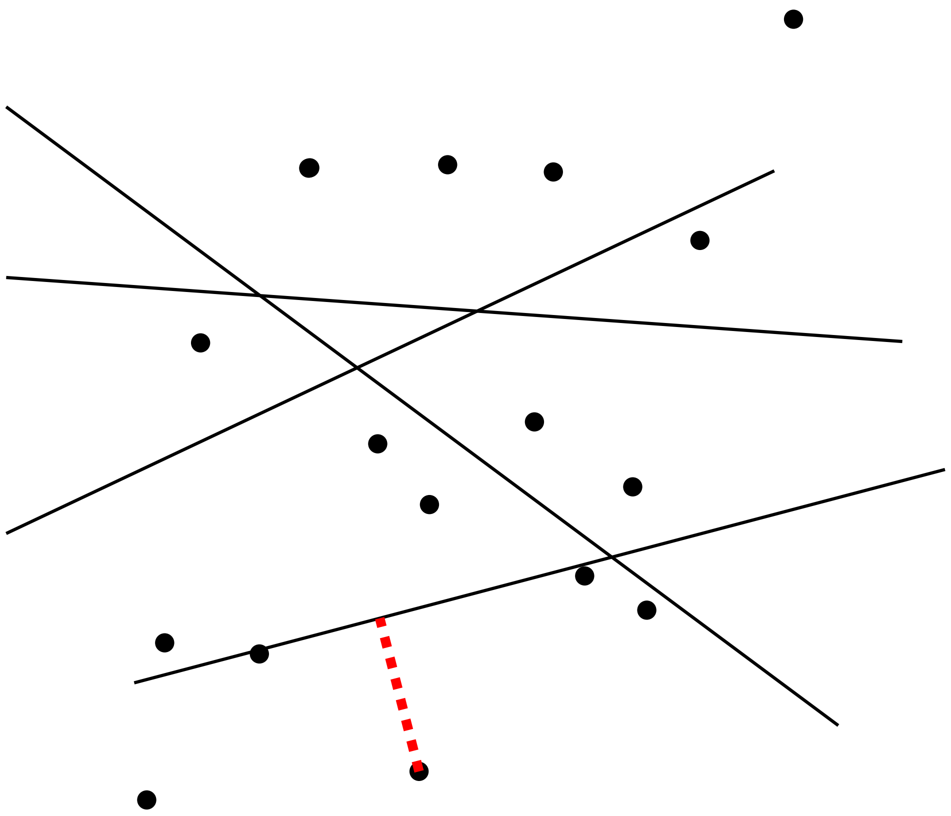


k-line-center problem

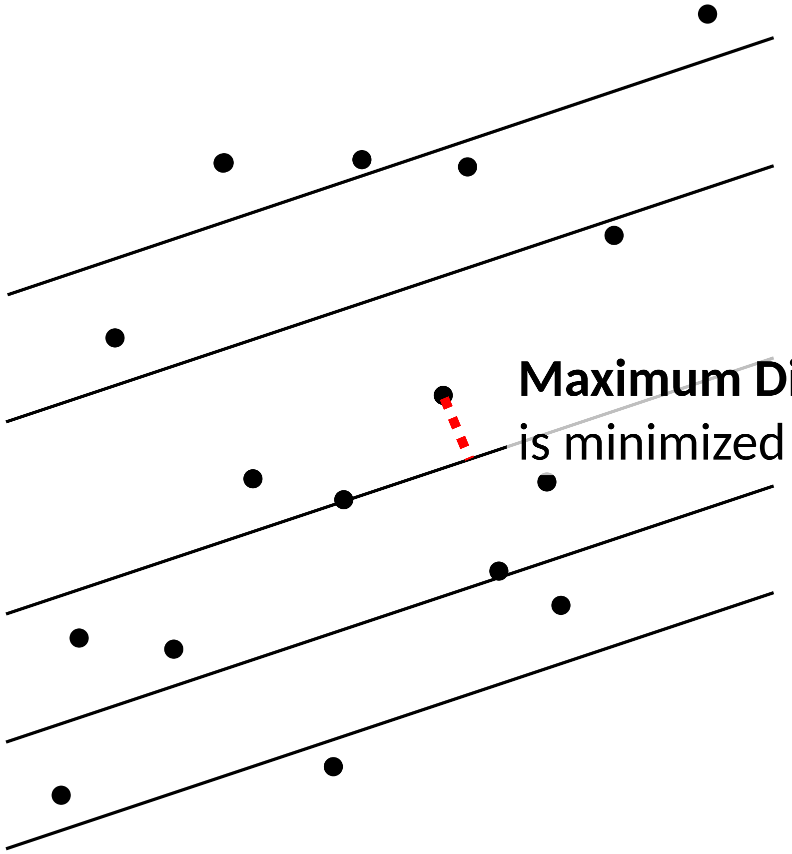


1. Introduction

k-line-center problem

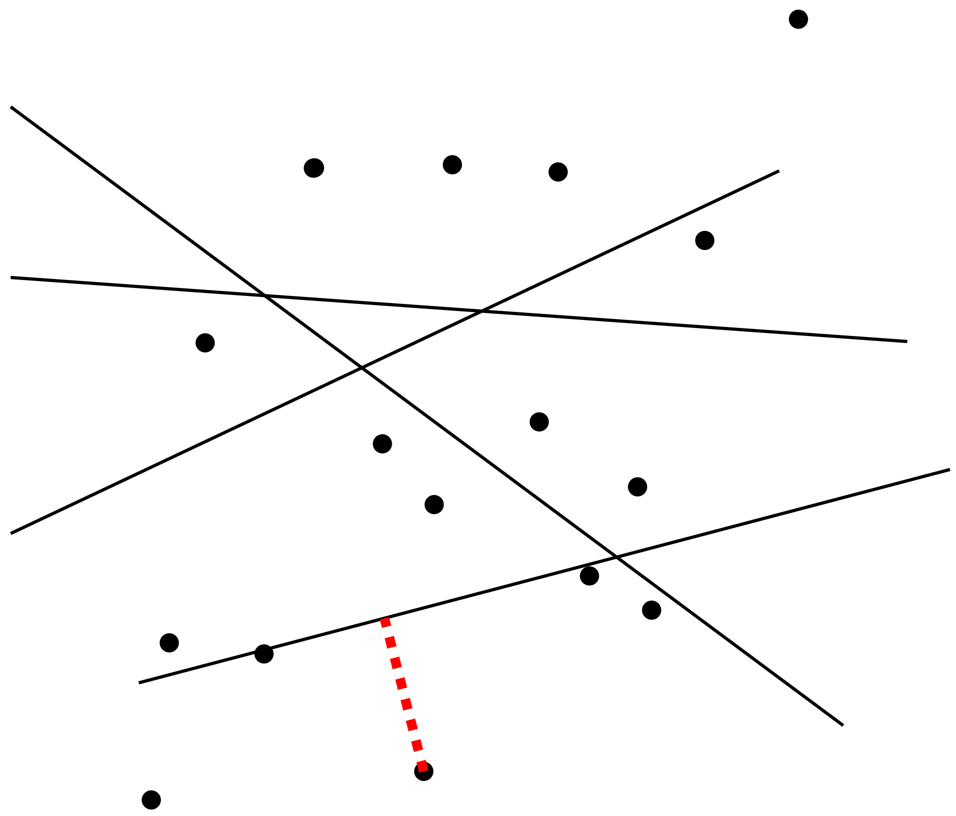


k-**parallel**-line-center problem

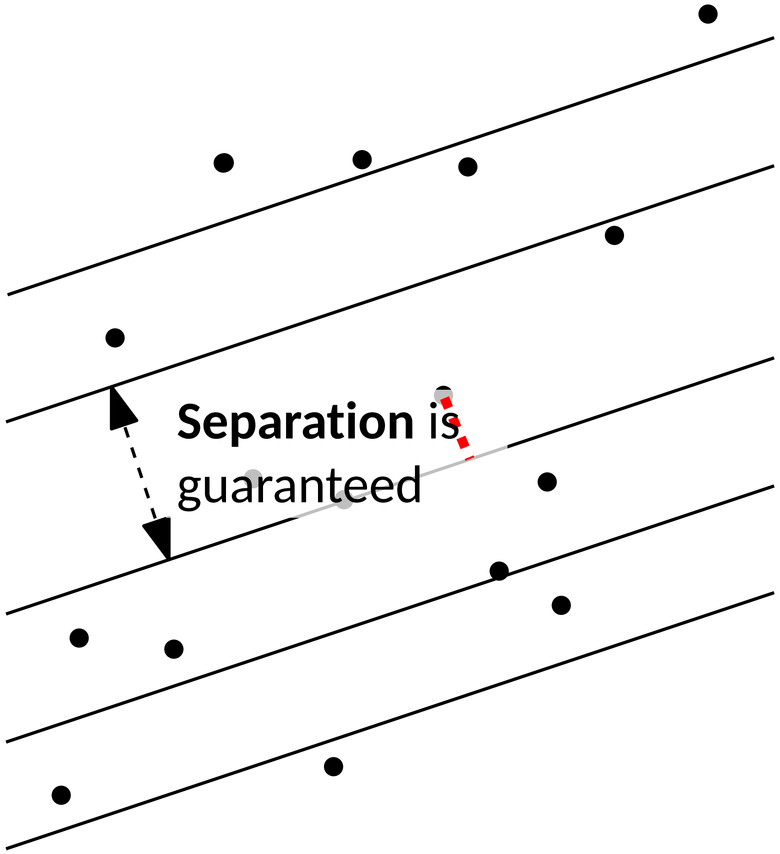


1. Introduction

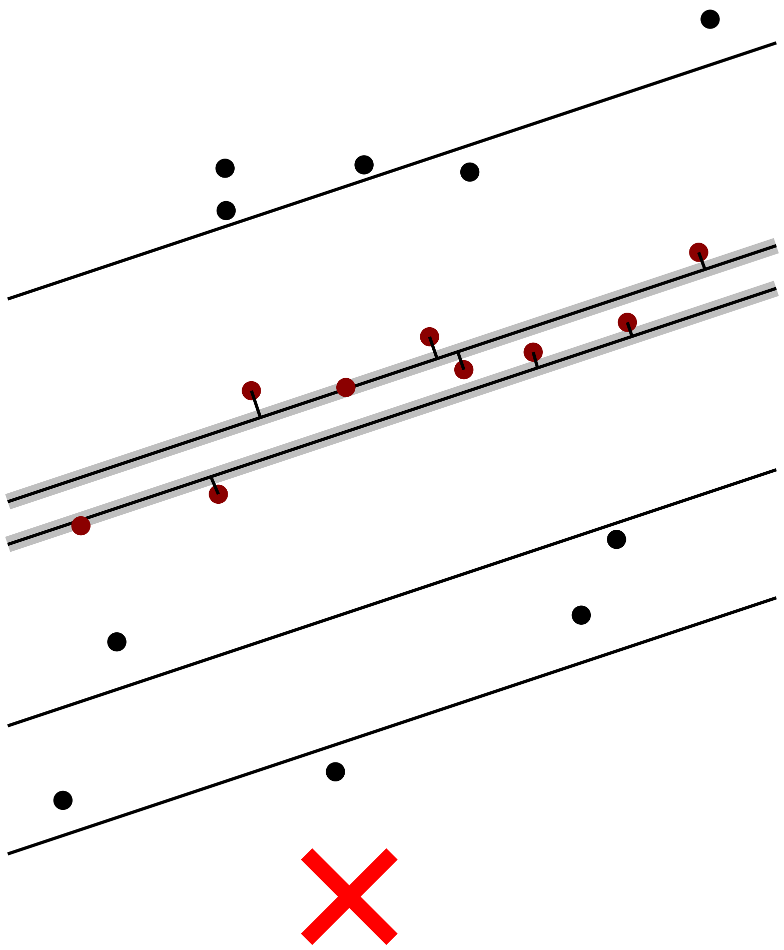
k-line-center problem



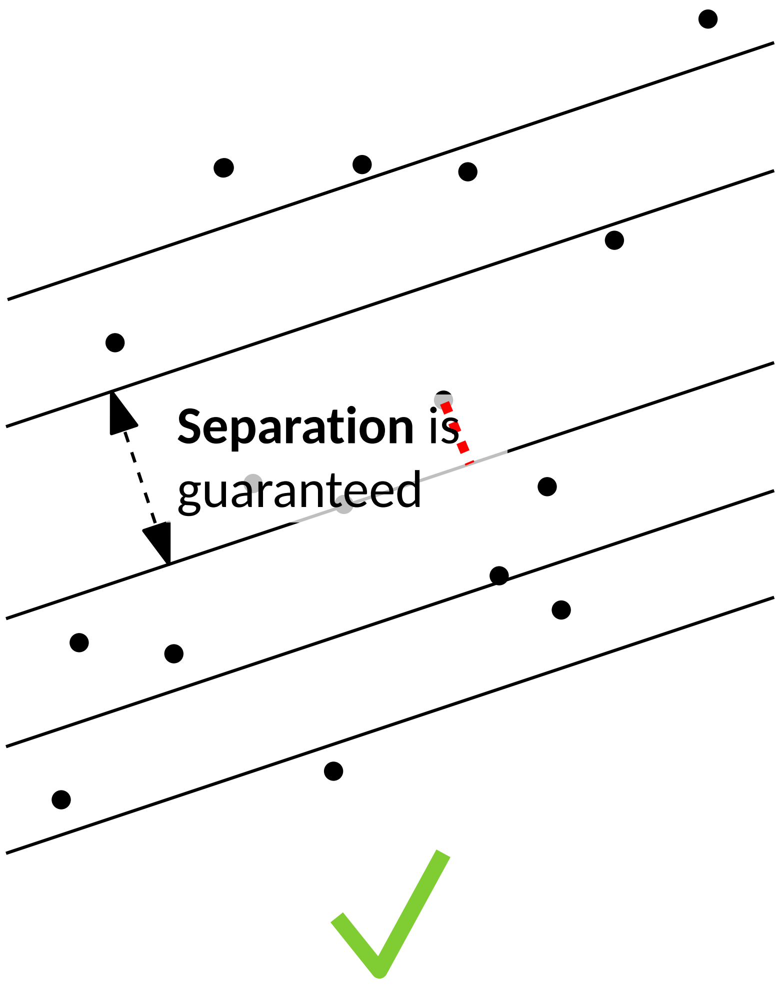
k-**parallel**-line-center problem



1. Introduction

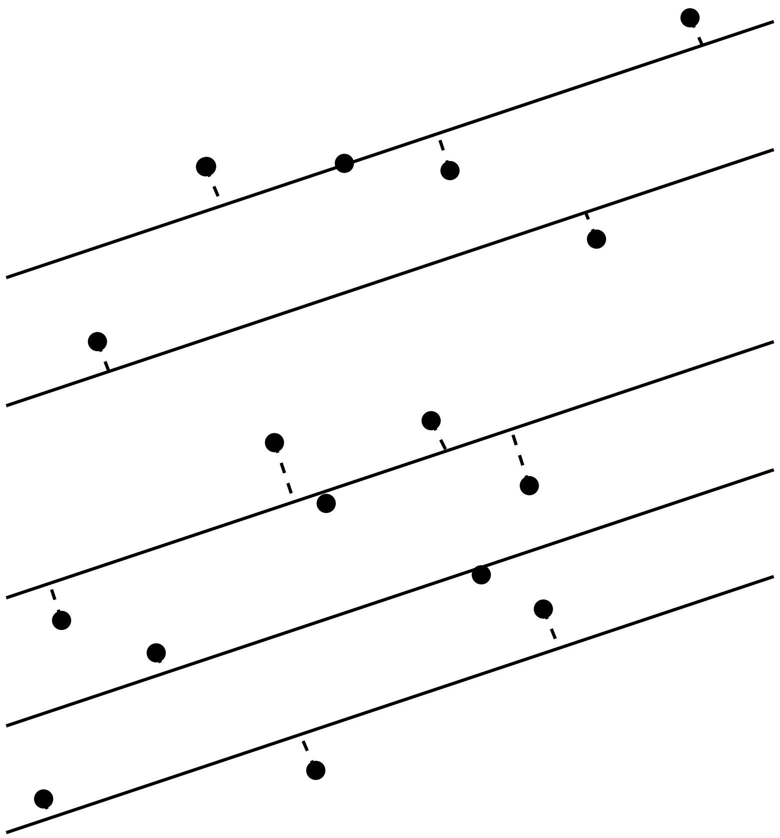


k-**parallel**-line-center problem

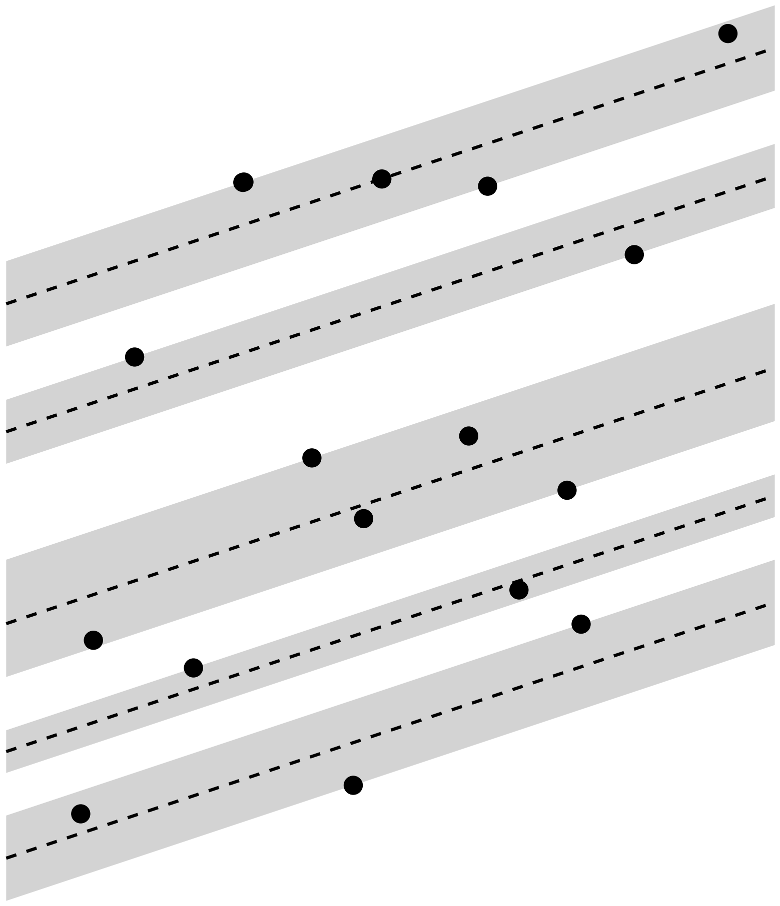


For a given point set P,

k-parallel-line-center of P

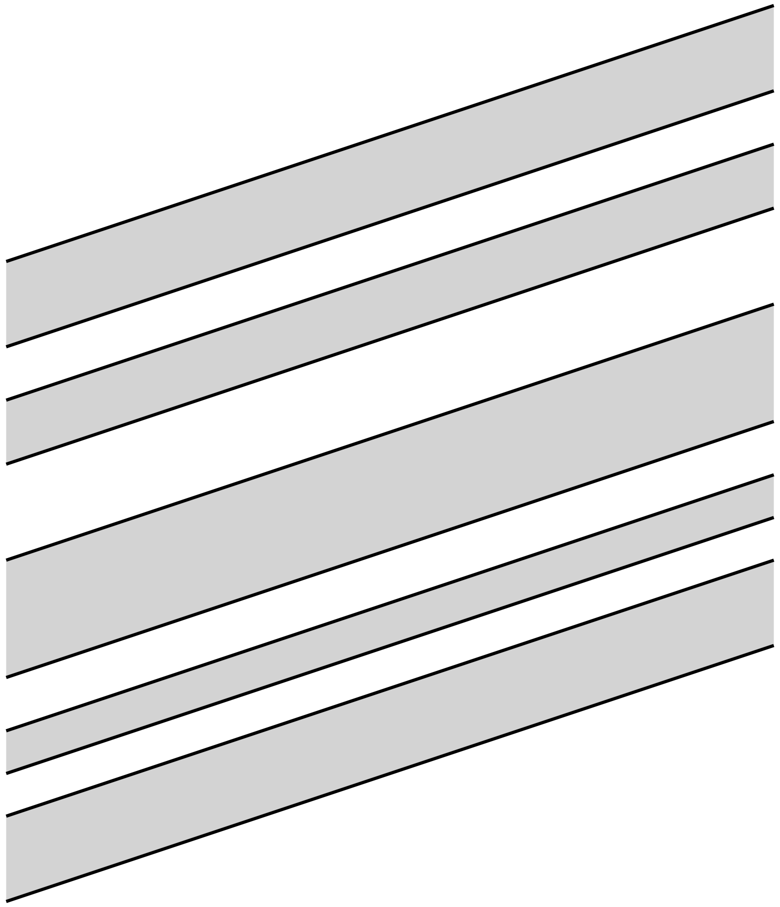


k-**slab cover** of P



Given a set of *k* slabs,

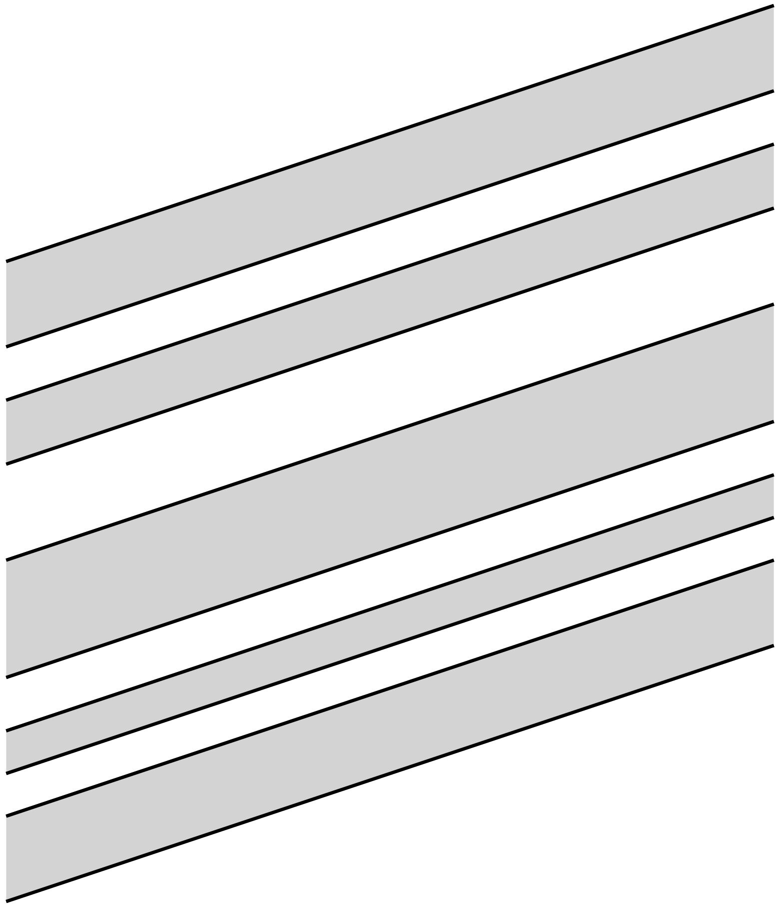
***k*-slab**



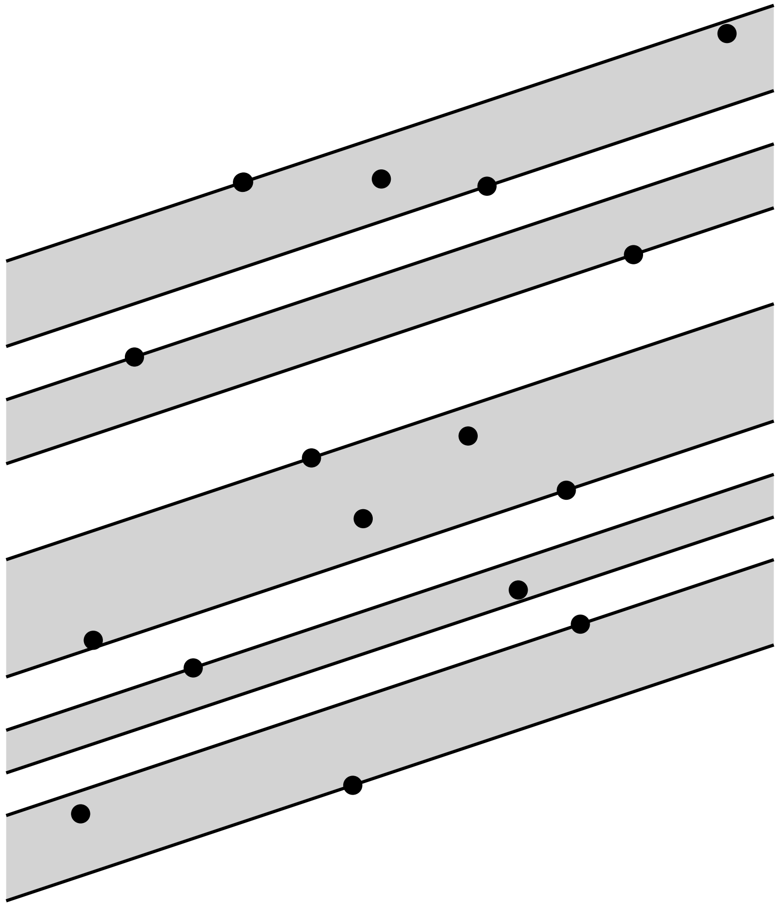
Given a set of k slabs,

Given a set P of points,

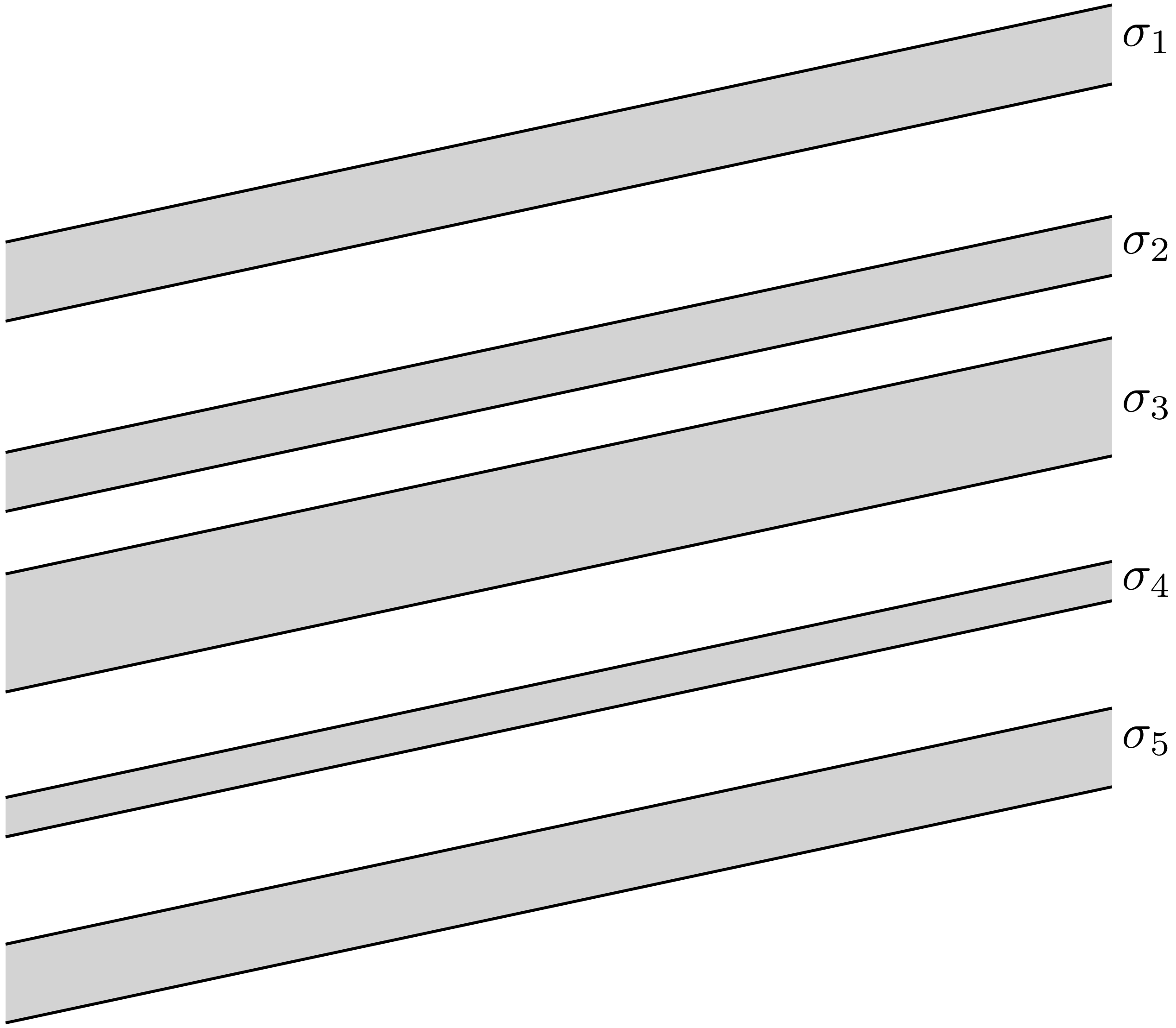
***k*-slab**



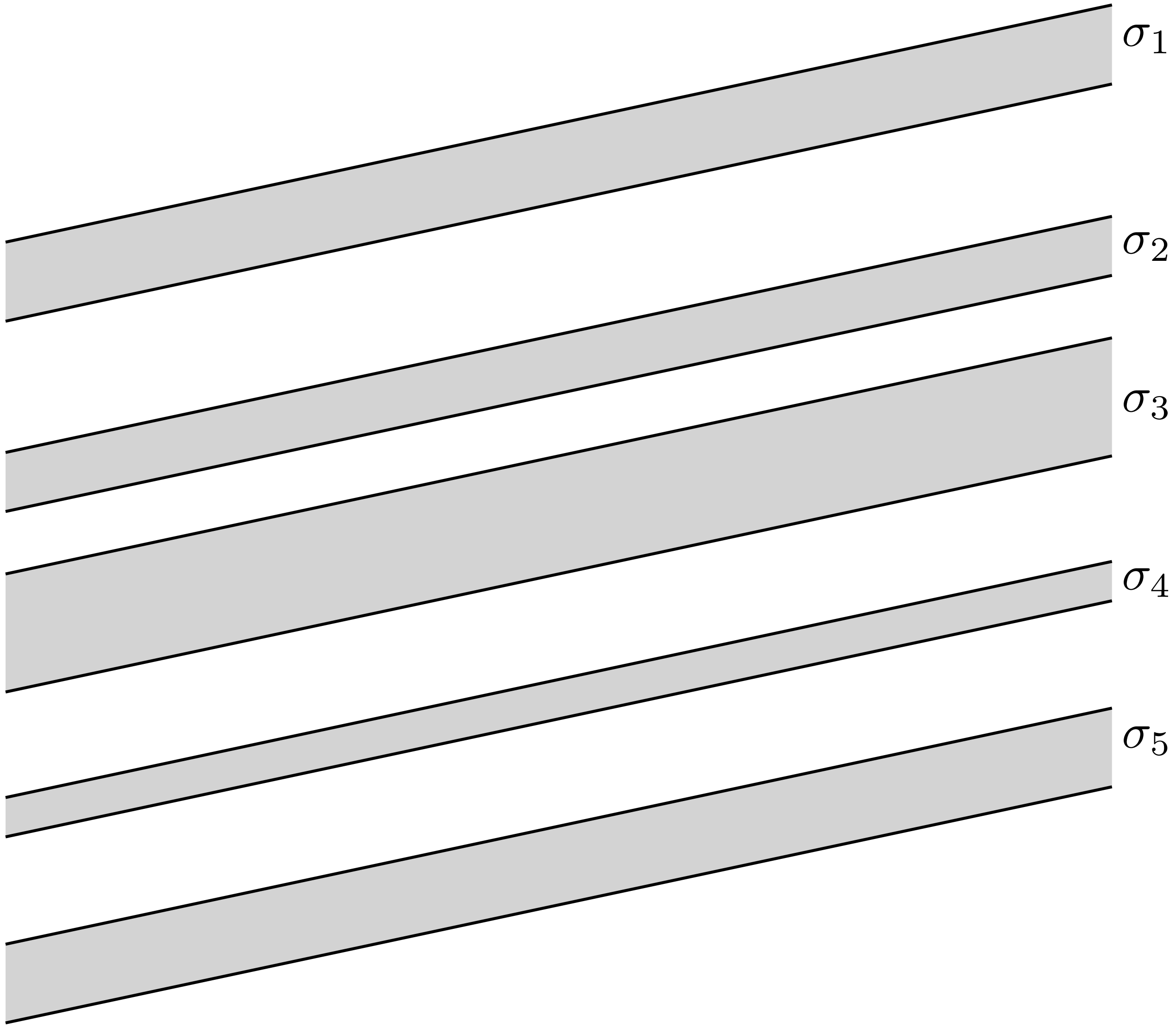
***k*-slab cover** of P

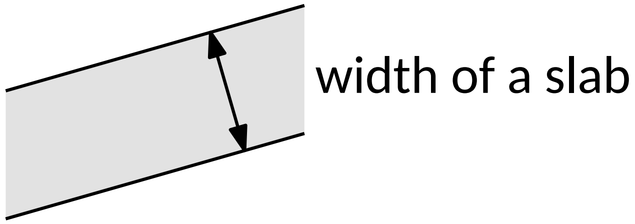


k-slab $\mathsf{S} = (\sigma_1, \ldots, \sigma_k)$ when $k = 5$

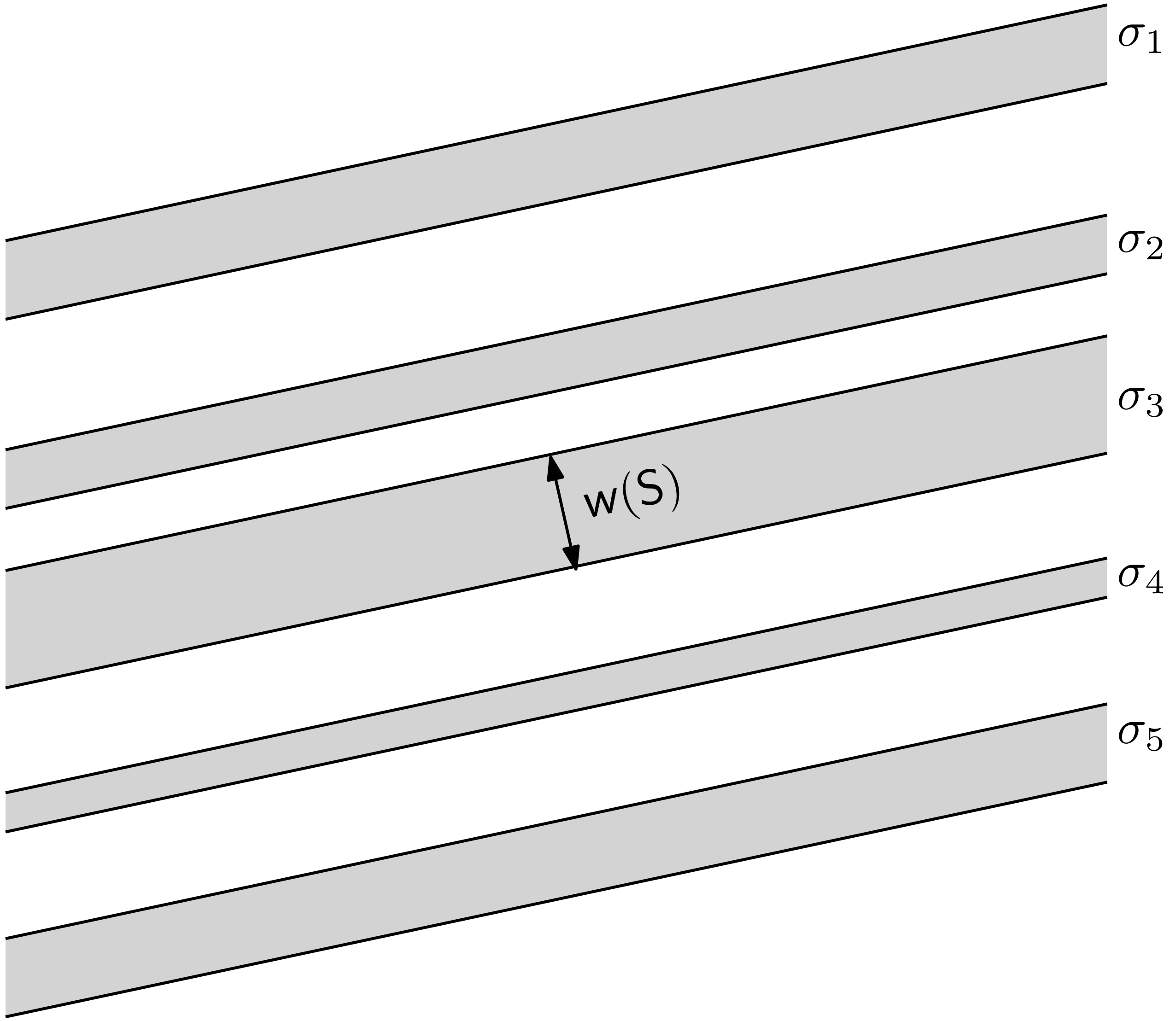


k-slab $S = (\sigma_1, \ldots, \sigma_k)$ when $k = 5$





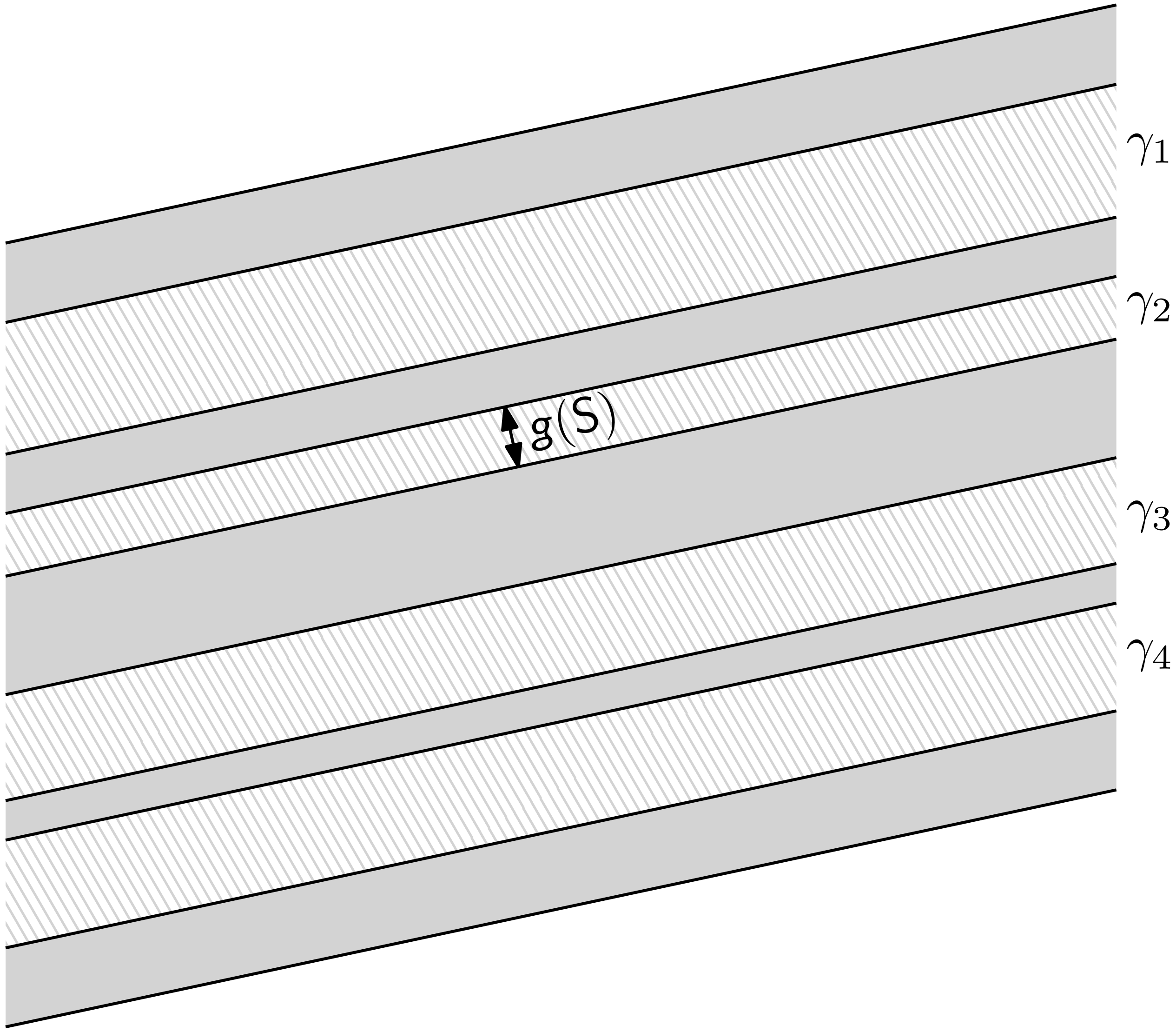
k-slab $S = (\sigma_1, \dots, \sigma_k)$ when $k = 5$



The width of S

- $w(S) := \max\{w(\sigma_1), w(\sigma_2), \dots\}$

k-slab $S = (\sigma_1, \ldots, \sigma_k)$ when $k = 5$



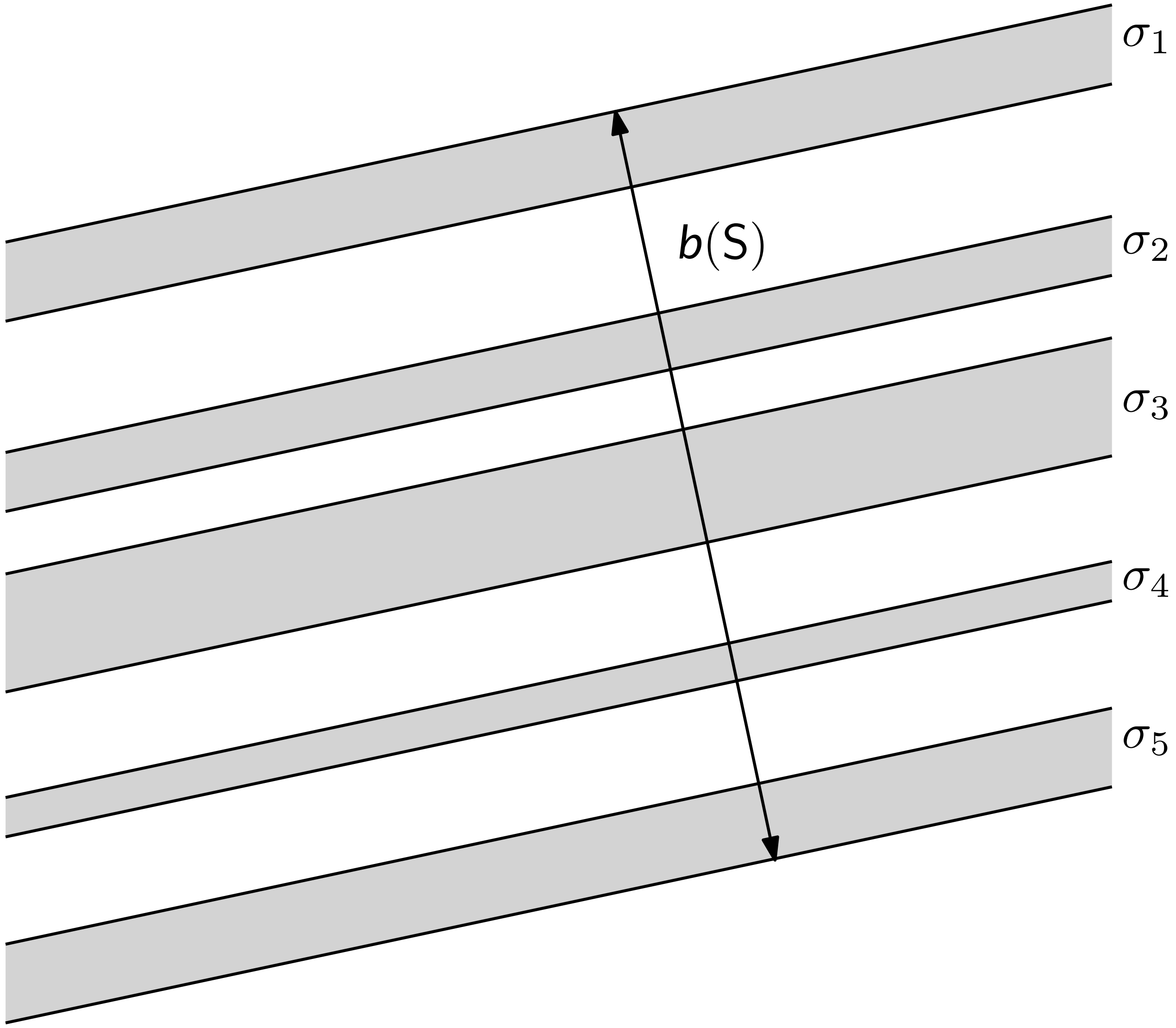
The width of S

- $w(S) := \max\{w(\sigma_1), w(\sigma_2), \ldots\}$

The gap-width of S

- $g(S) := \min\{w(\gamma_1), w(\gamma_2), \ldots\}$

k-slab $S = (\sigma_1, \ldots, \sigma_k)$ when $k = 5$



The width of S

- $w(S) := \max\{w(\sigma_1), w(\sigma_2), \ldots\}$

The gap-width of S

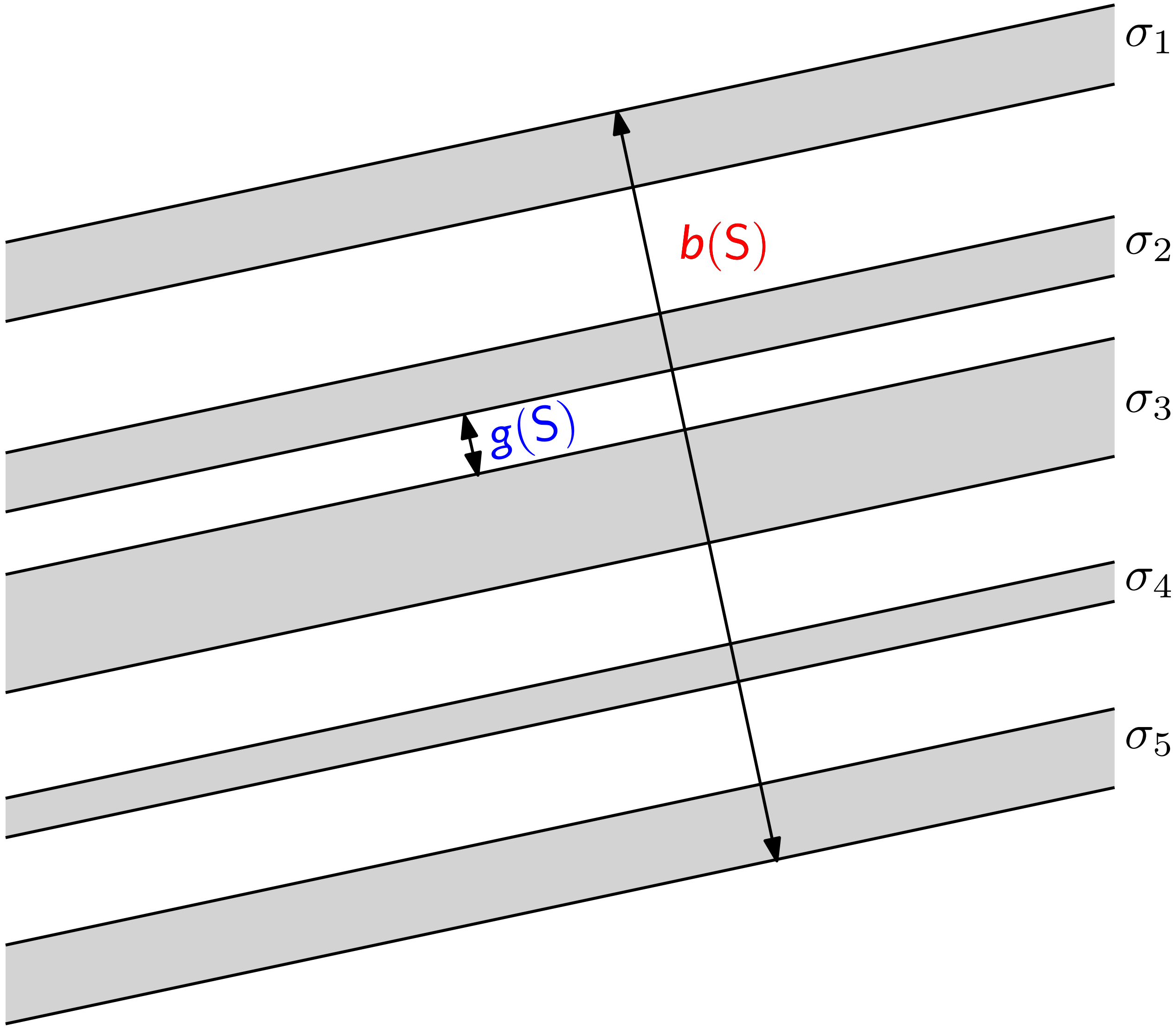
- $g(S) := \min\{w(\gamma_1), w(\gamma_2), \ldots\}$

The breadth of S

- $b(S)$

1. Introduction

k-slab $S = (\sigma_1, \ldots, \sigma_k)$ when $k = 5$



The width of S

- $w(S) := \max\{w(\sigma_1), w(\sigma_2), \ldots\}$

The gap-width of S

- $g(S) := \min\{w(\gamma_1), w(\gamma_2), \ldots\}$

The breadth of S

- $b(S)$

The gap-ratio of S

- $\rho(S) := g(S)/b(S)$

1. Introduction

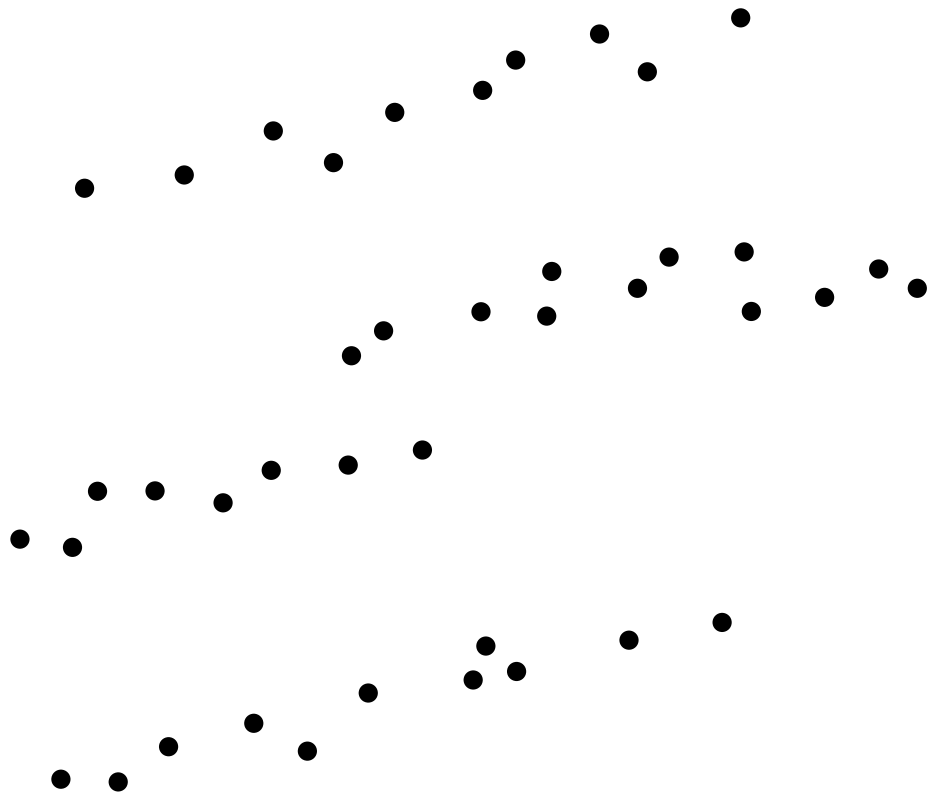
For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

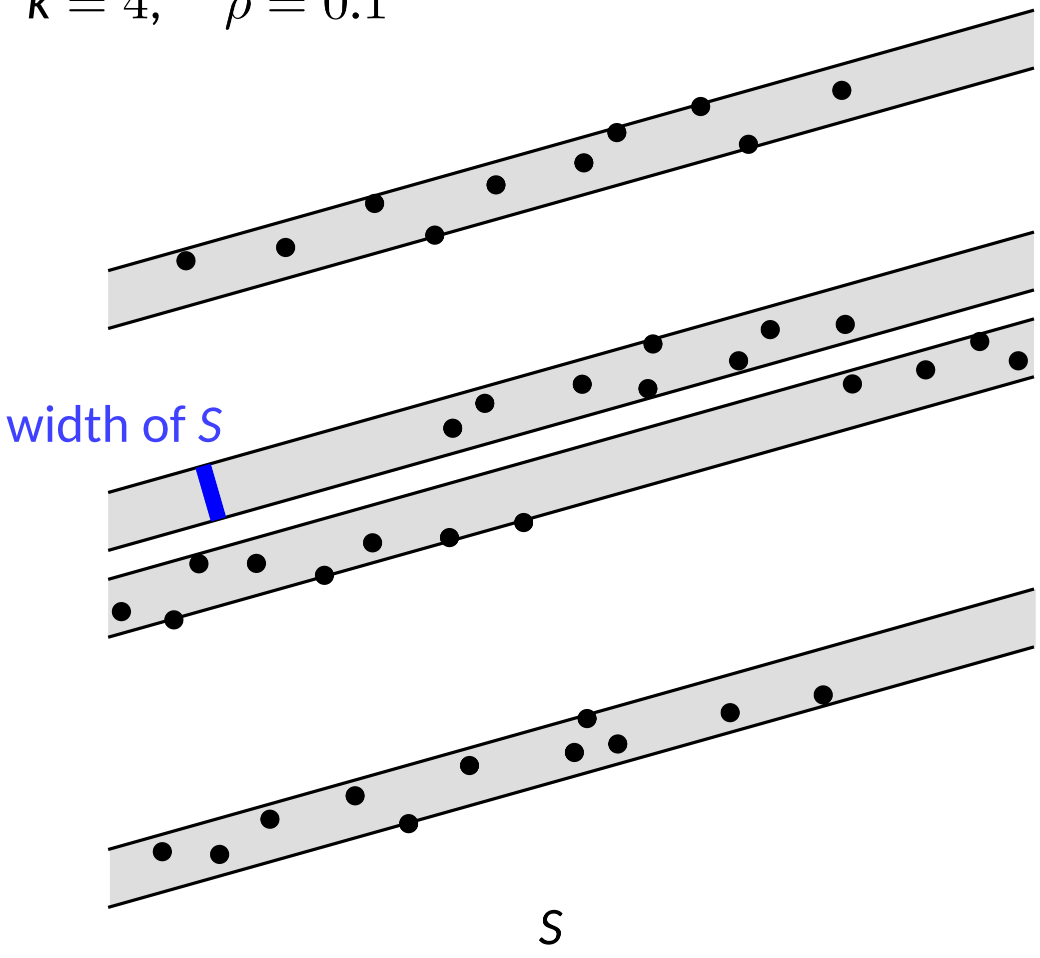
$k = 4, \quad \rho = 0.1$



For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$

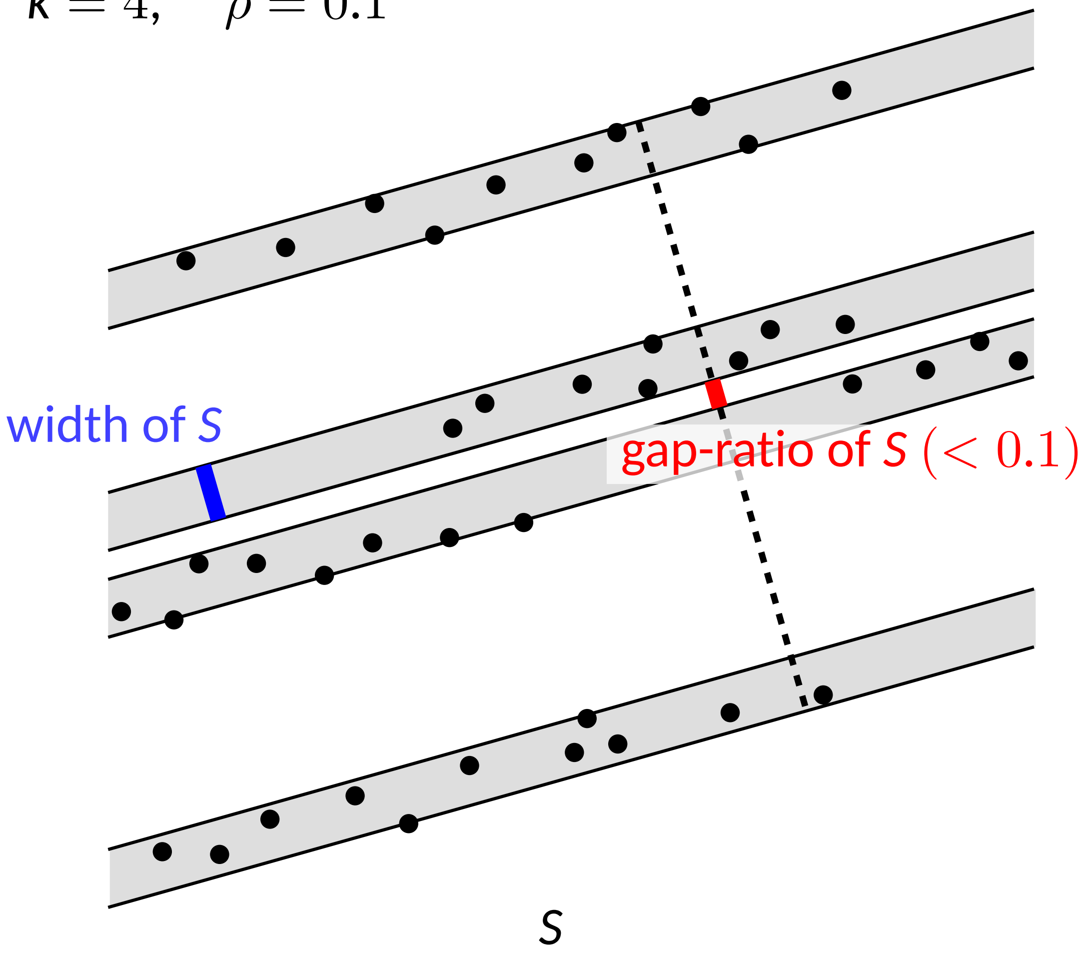


1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

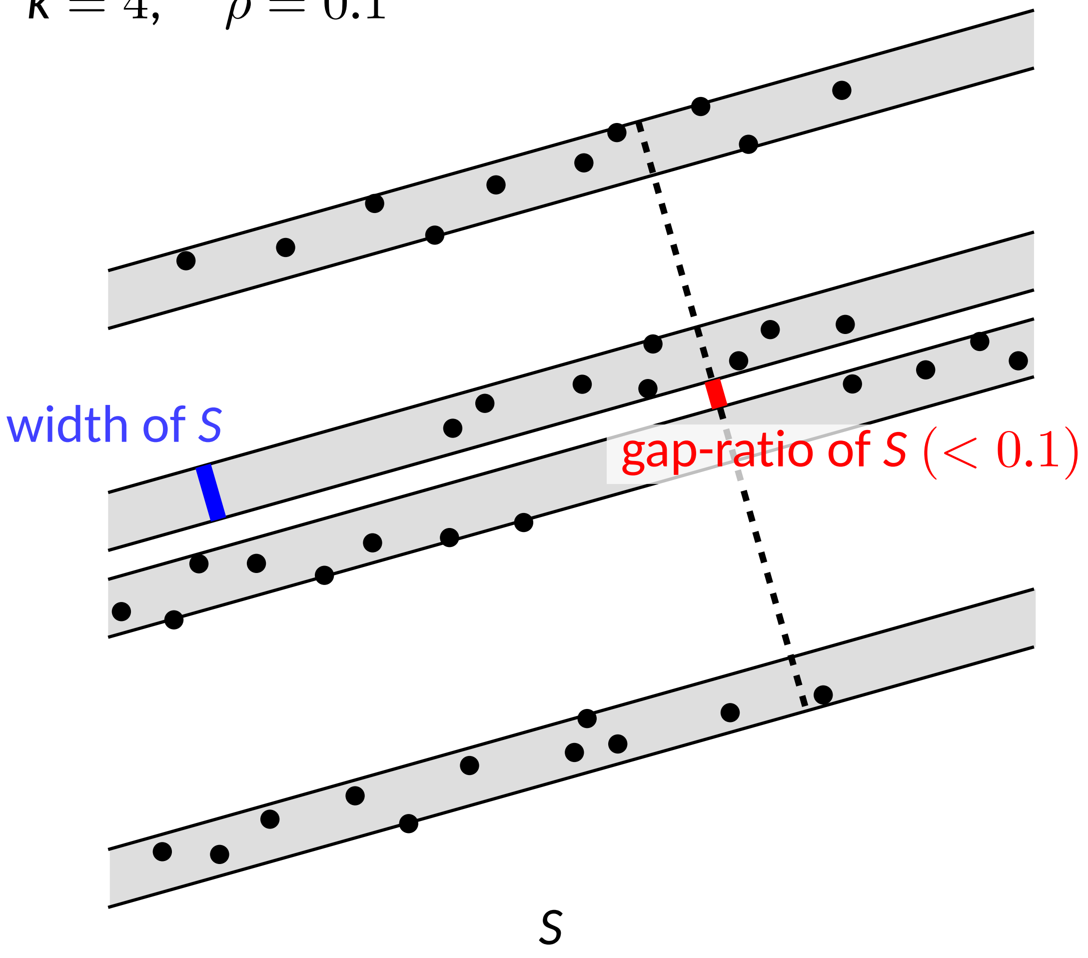
$k = 4, \quad \rho = 0.1$

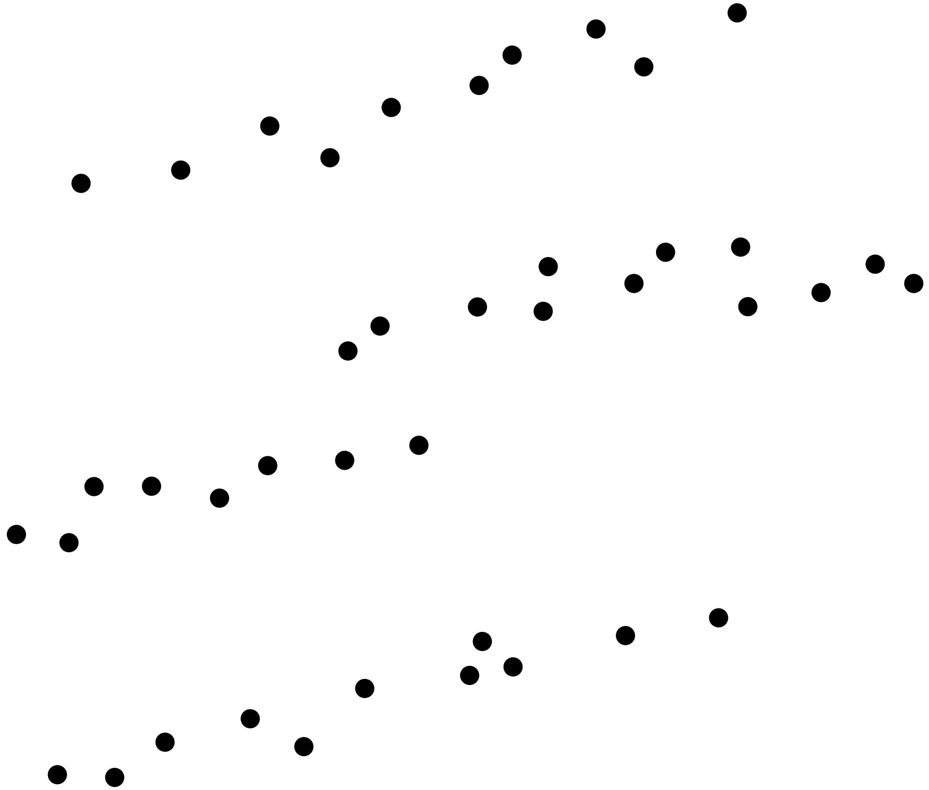


For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$

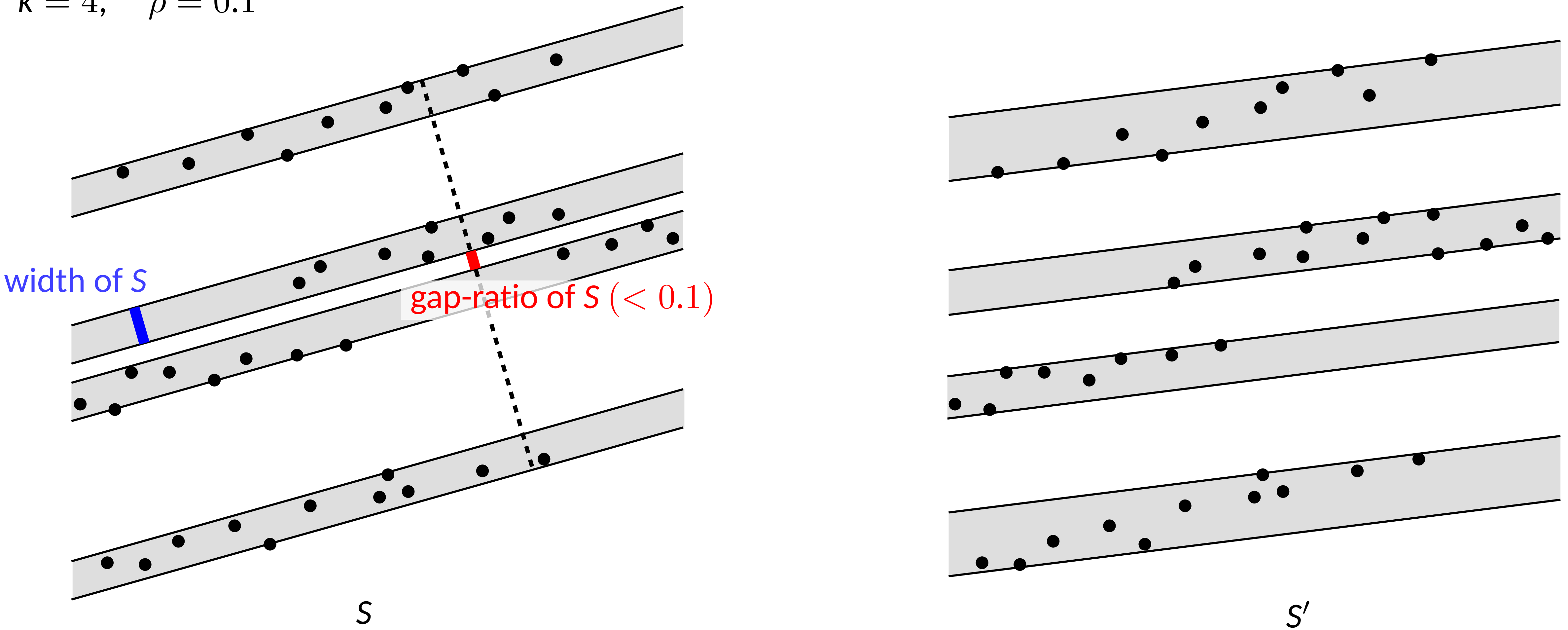




For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$

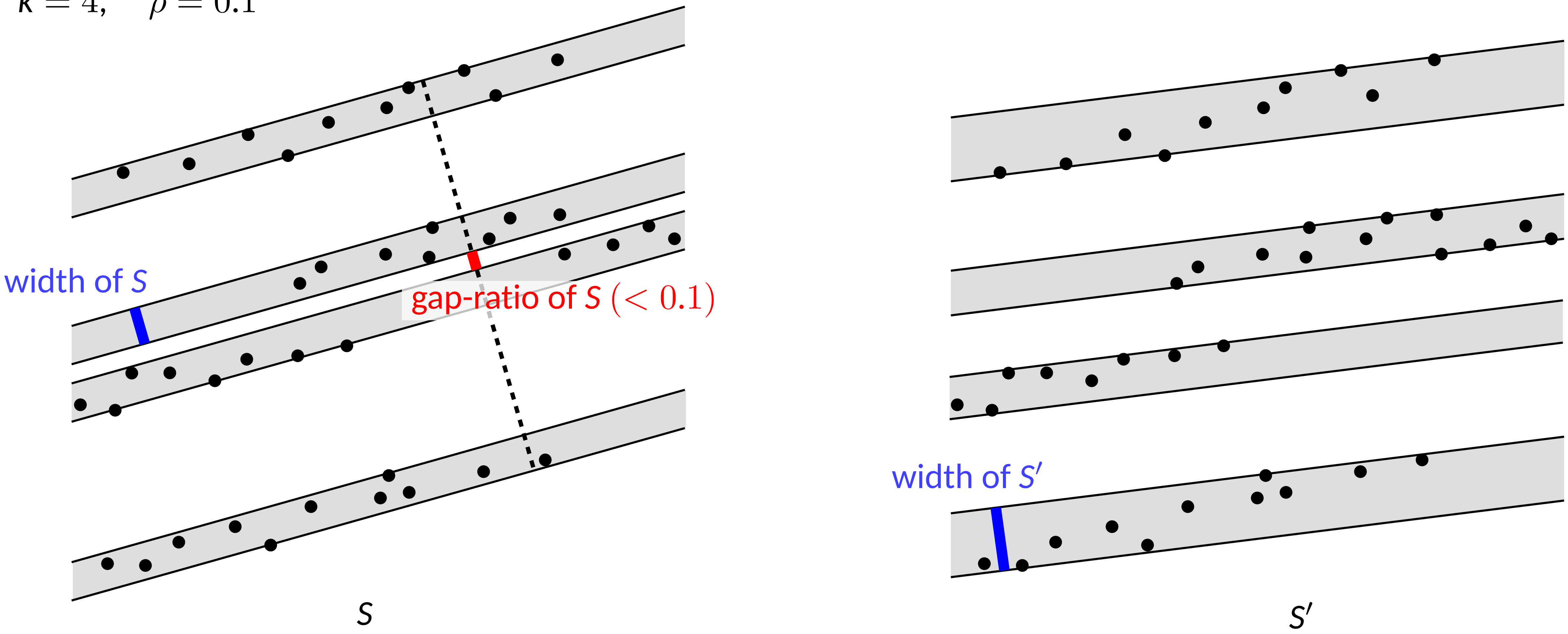


1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$

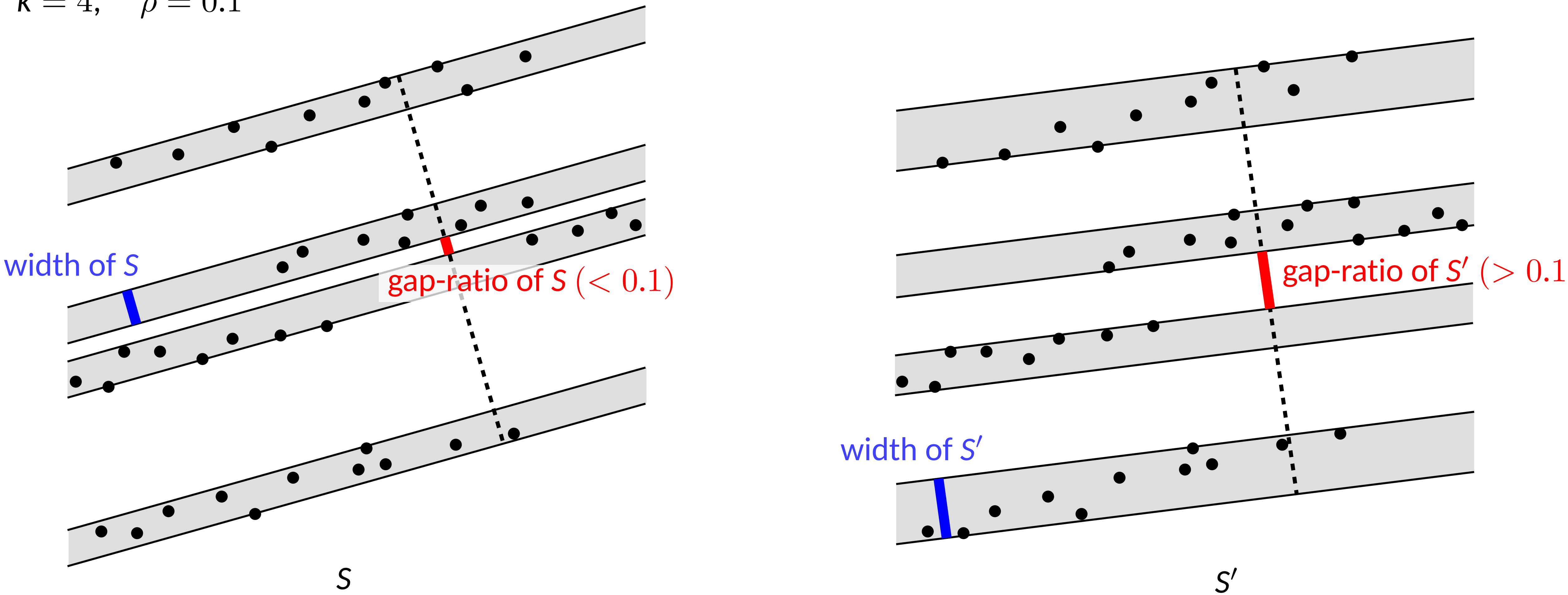


1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$

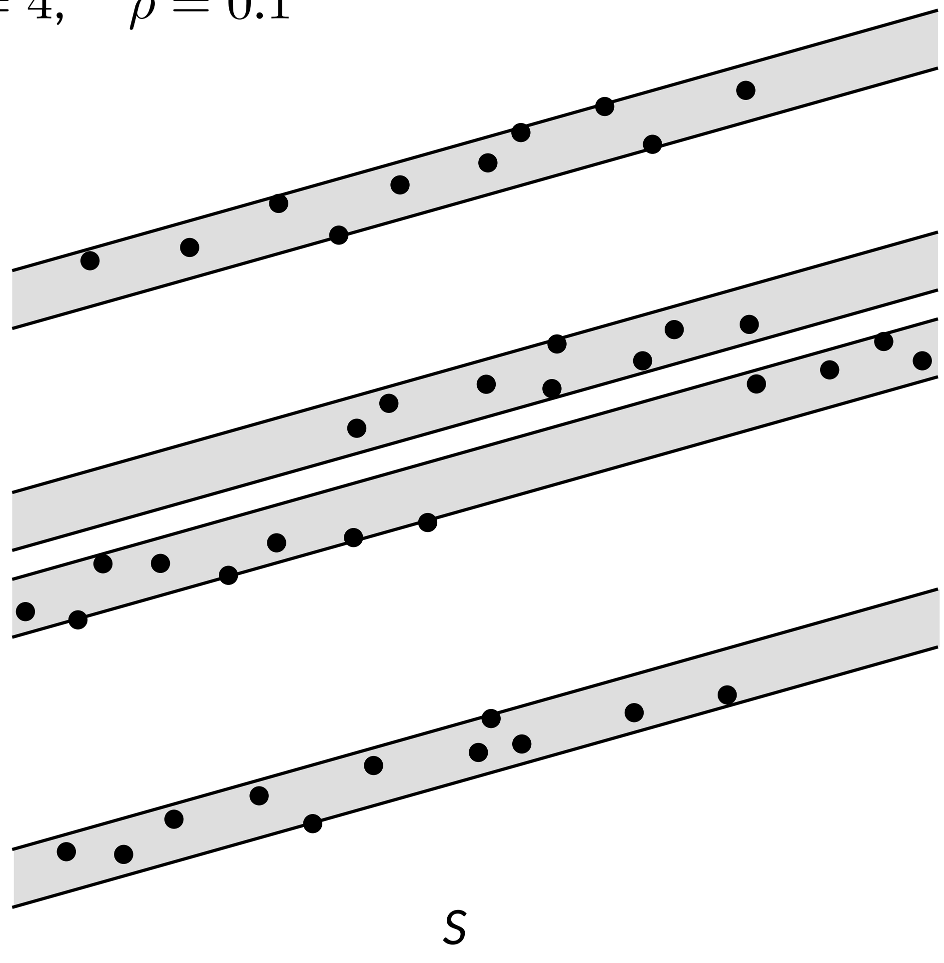


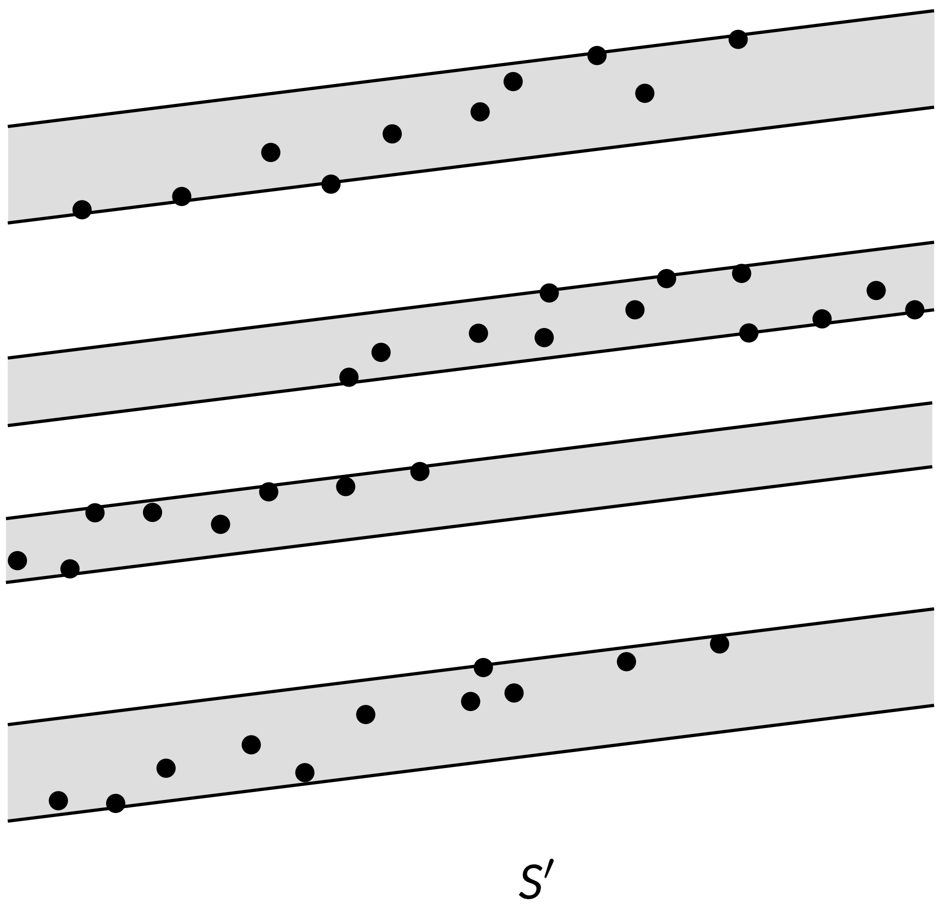
1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

$k = 4, \quad \rho = 0.1$





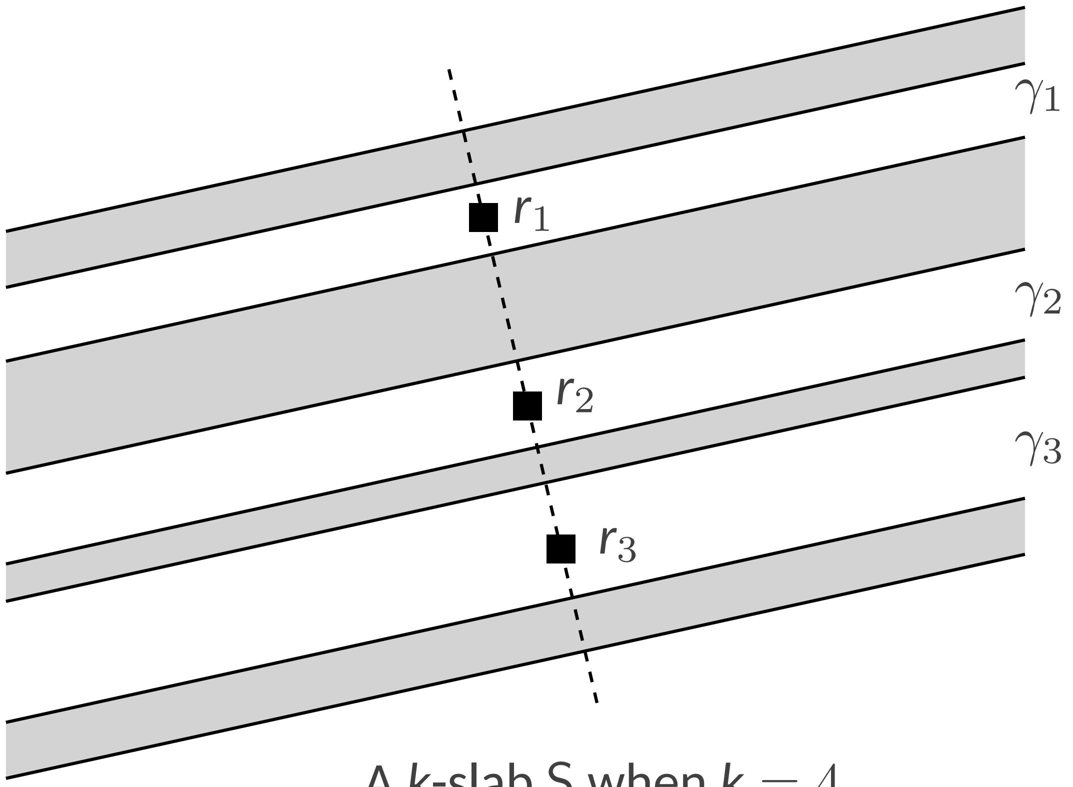
For given *k* $\geqslant 2$, a set *P* of *n* points, and a real $\rho \in (0, 1]$,

find a minimum-width *k*-slab cover of *P* whose gap-ratio is at least ρ .

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

A **separator** of a k-slab S:
a sequence of $k - 1$ points on a common line each of which lies in its distinct gap.



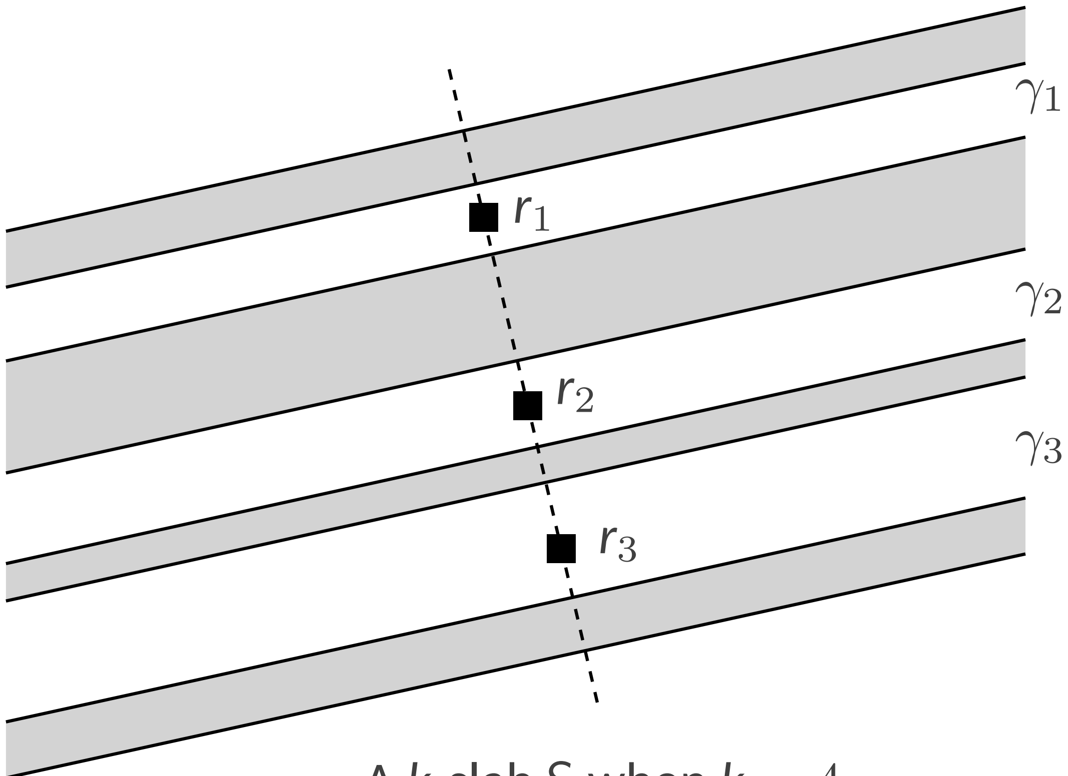
A k-slab S when $k = 4$

R $= (r_1, r_2, r_3)$
It holds that $r_i \in \gamma_i$ for each $i = 1, \ldots, k - 1$.

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

A **separator** of a k-slab S:
a sequence of $k - 1$ points on a common line each of which lies in its distinct gap.



A k-slab S when $k = 4$

$R = (r_1, r_2, r_3)$
It holds that $r_i \in \gamma_i$ for each $i = 1, \ldots, k - 1$.

A k-slab S **respects** the separator R .

1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

Step 1. An algorithm to compute a minimum-width k-slab cover of P which **respects** a given separator R in $O(kn \log n)$ time and $O(n)$ space.

Step 2. An algorithm to compute $O(\rho^{-k})$ candidate separators.

Step 3. Compute a minimum-width k-slab cover of P by testing $O(\rho^{-k})$ candidate separators in $O(\rho^{-k} kn \log n)$ time and $O(n)$ space.

For given *k* $\geqslant 2$, a set *P* of *n* points, and a real $\rho \in (0, 1]$,

find a minimum-width *k*-slab cover of *P* whose gap-ratio is at least ρ .

Step 1. An algorithm to compute a minimum-width *k*-slab cover of *P* which **respects** a given separator R in $O(kn \log n)$ time and $O(n)$ space.

Step 2. An algorithm to compute $O(\rho^{-k})$ candidate separators.

Step 3. Compute a minimum-width *k*-slab cover of *P* by testing $O(\rho^{-k})$ candidate separators in $O(\rho^{-k} kn \log n)$ time and $O(n)$ space.

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

Step 1. An algorithm to compute a minimum-width k-slab cover of P which **respects** a given separator R in $O(kn \log n)$ time and $O(n)$ space.

Step 2. An algorithm to compute $O(\rho^{-k})$ candidate separators.

Step 3. Compute a minimum-width k-slab cover of P by testing $O(\rho^{-k})$ candidate separators in $O(\rho^{-k} kn \log n)$ time and $O(n)$ space.

1. Introduction

For given $k \geqslant 2$, a set P of n points, and a real $\rho \in (0, 1]$,

find a minimum-width k-slab cover of P whose gap-ratio is at least ρ .

Step 1. An algorithm to compute a minimum-width k-slab cover of P which **respects** a given separator R in $O(kn \log n)$ time and $O(n)$ space.

Step 2. An algorithm to compute $O(\rho^{-k})$ candidate separators.

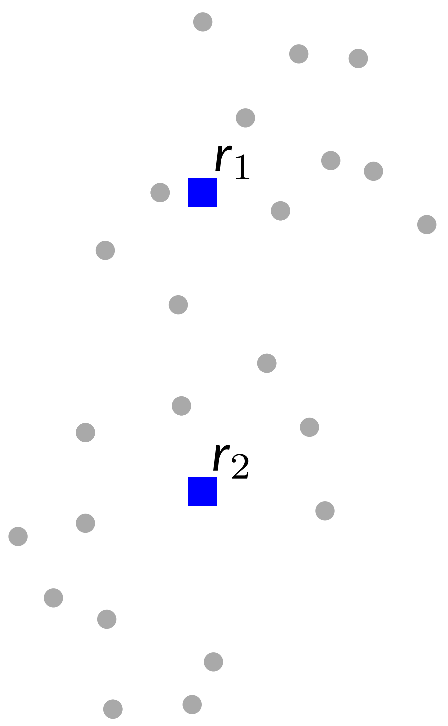
Step 3. Compute a minimum-width k-slab cover of P by testing $O(\rho^{-k})$ candidate separators in $O(\rho^{-k} kn \log n)$ time and $O(n)$ space.

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

2. A min-width k-slab cover for a Given Separator

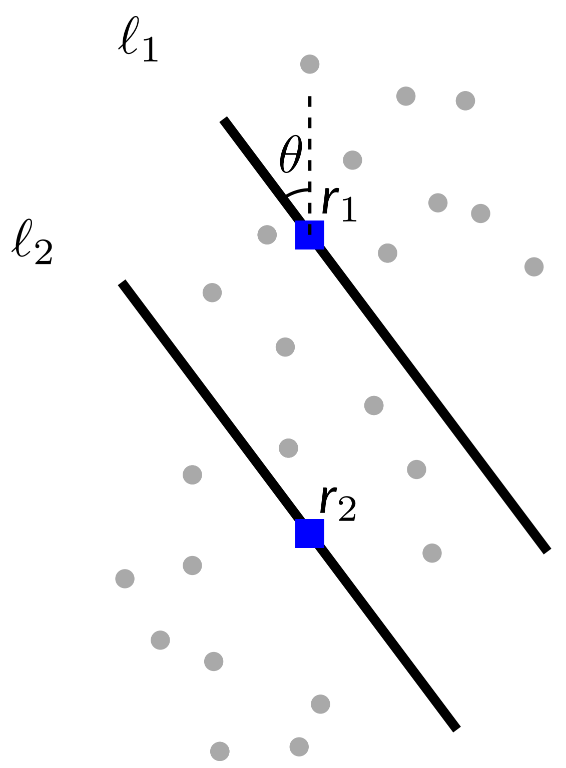
Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



$k = 3$

2. A min-width k-slab cover for a Given Separator

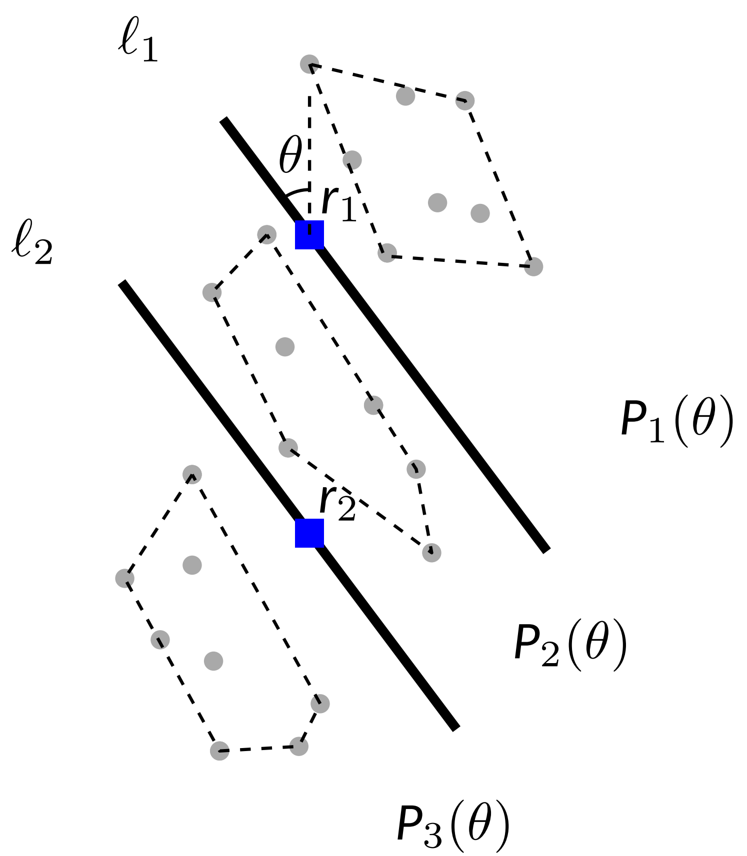
Step 1. Given a separator $\mathsf{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



$k = 3$

2. A min-width k-slab cover for a Given Separator

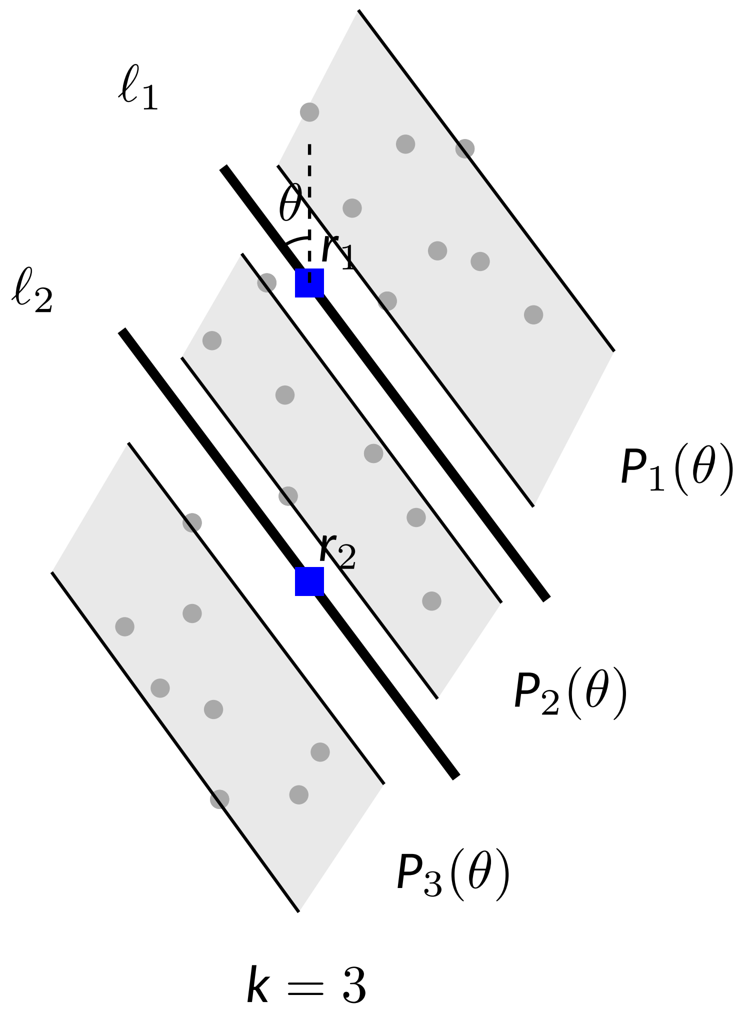
Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



$k = 3$

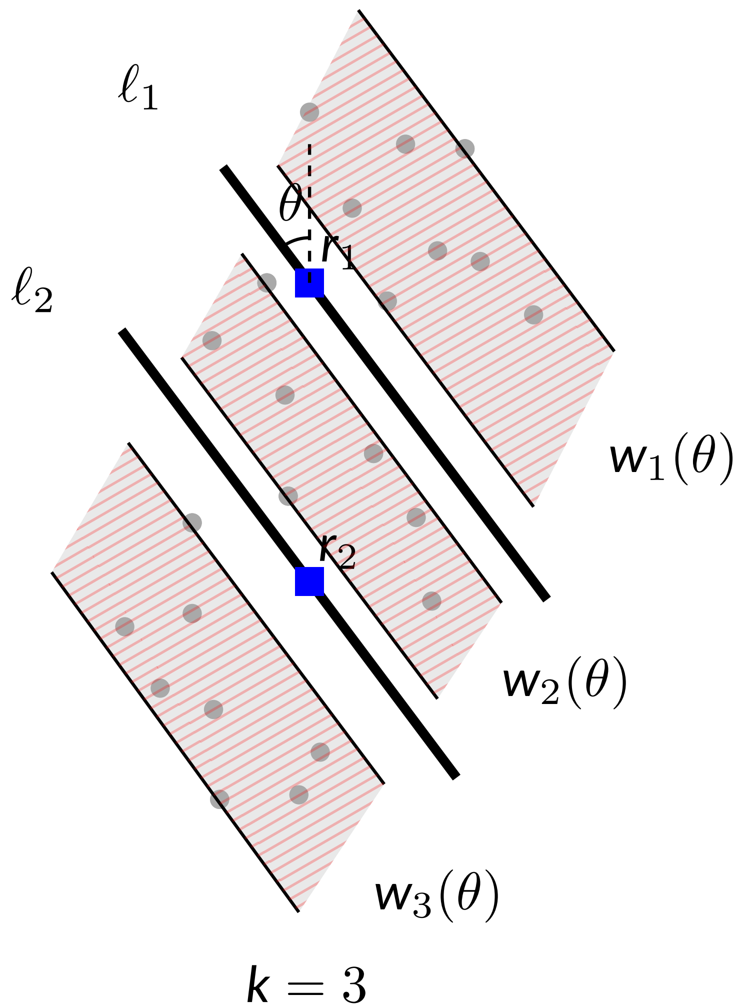
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



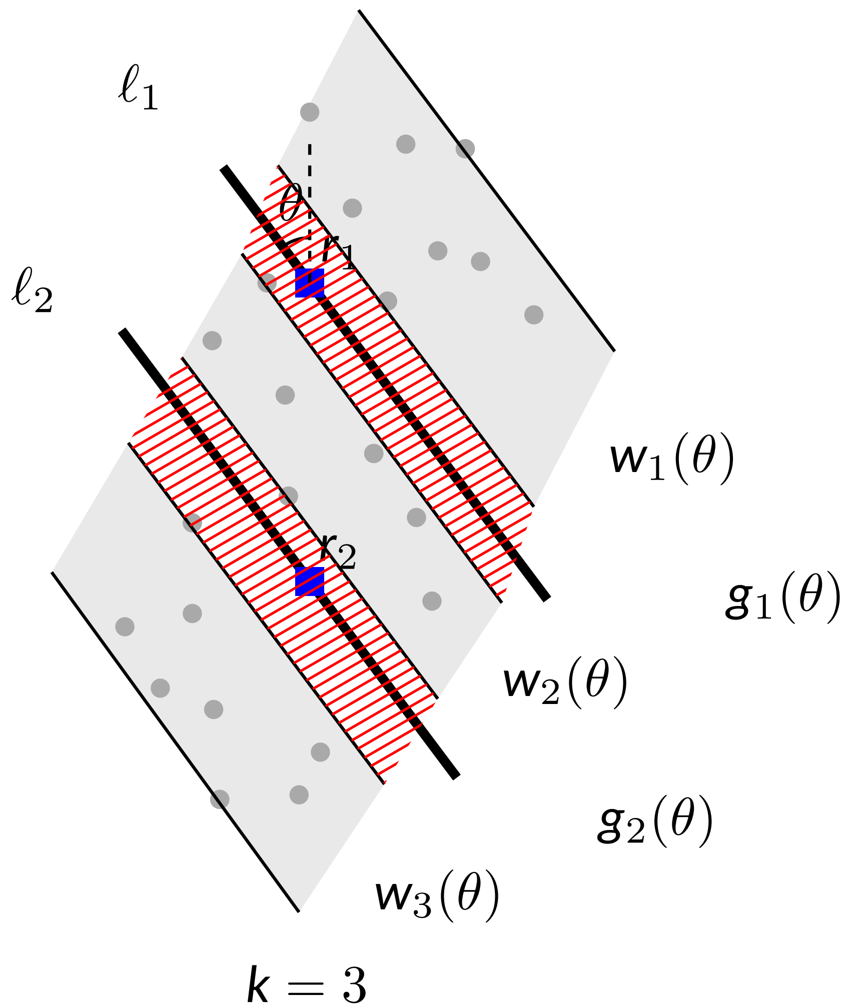
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



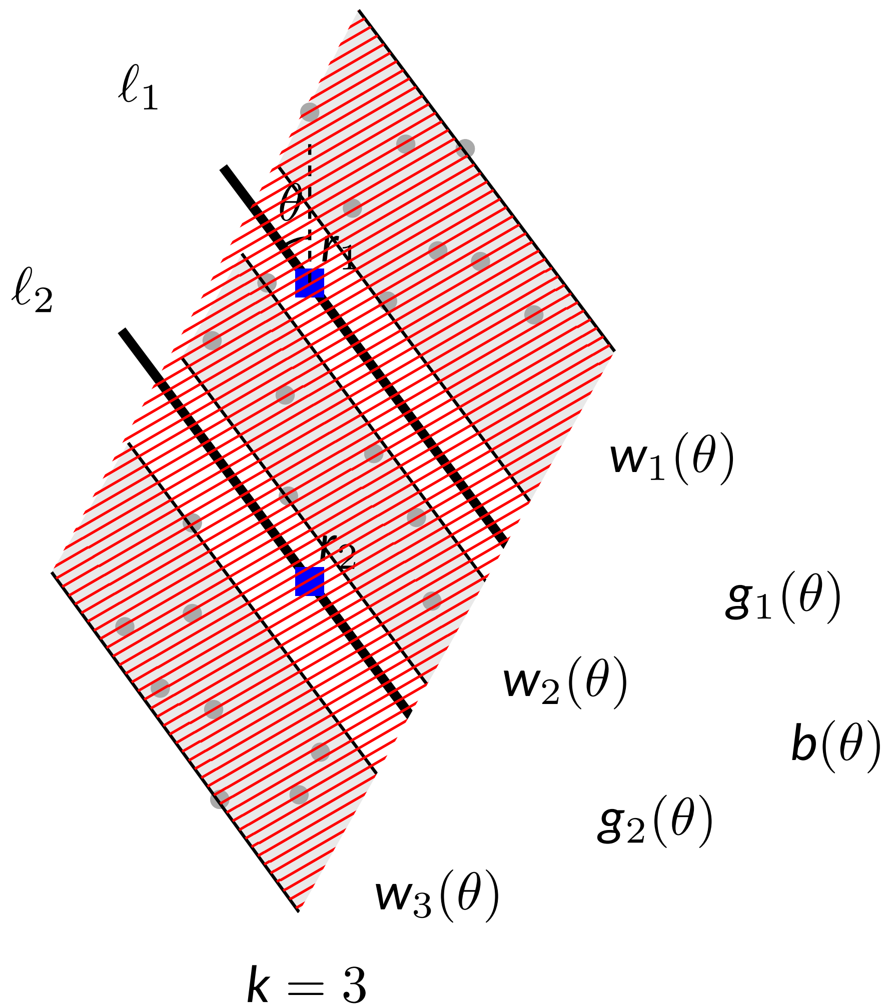
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



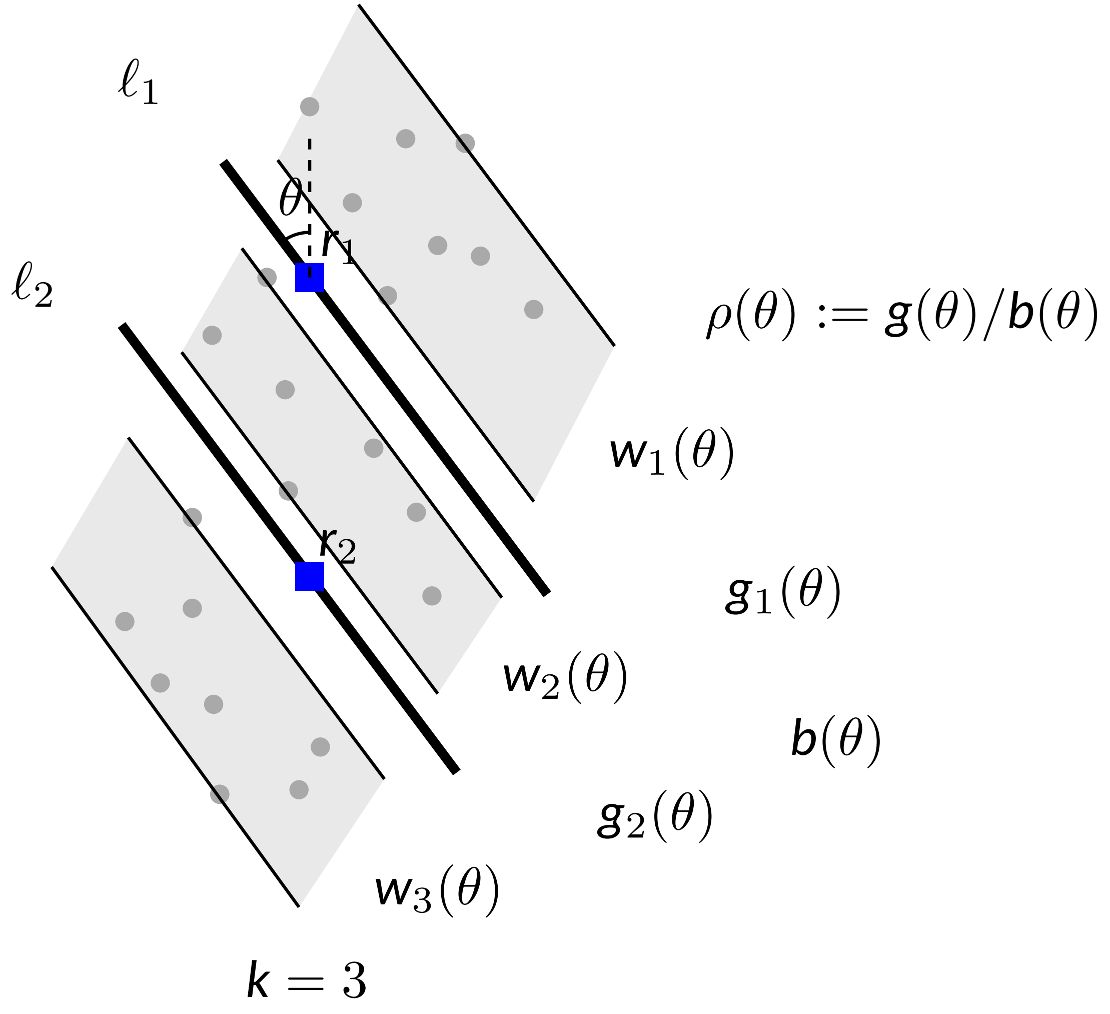
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



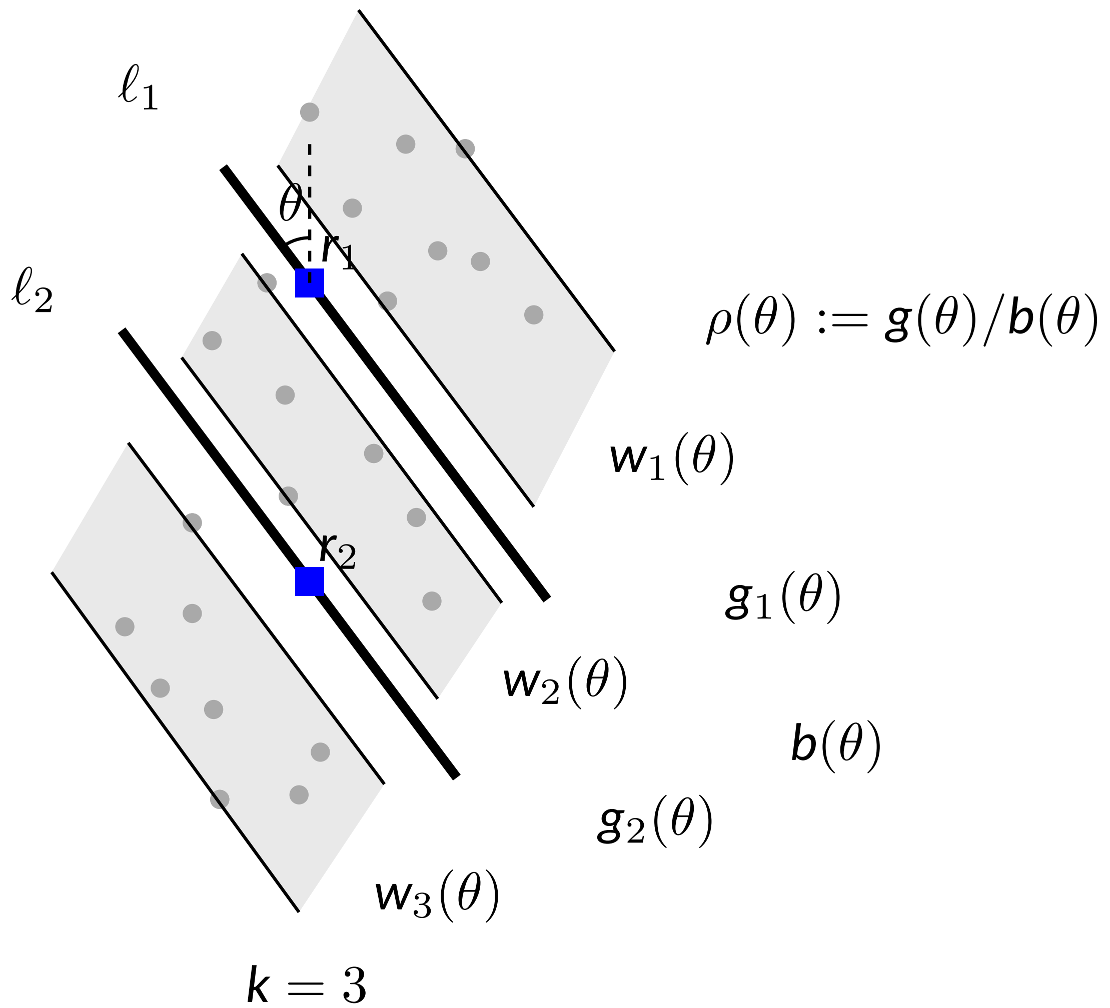
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



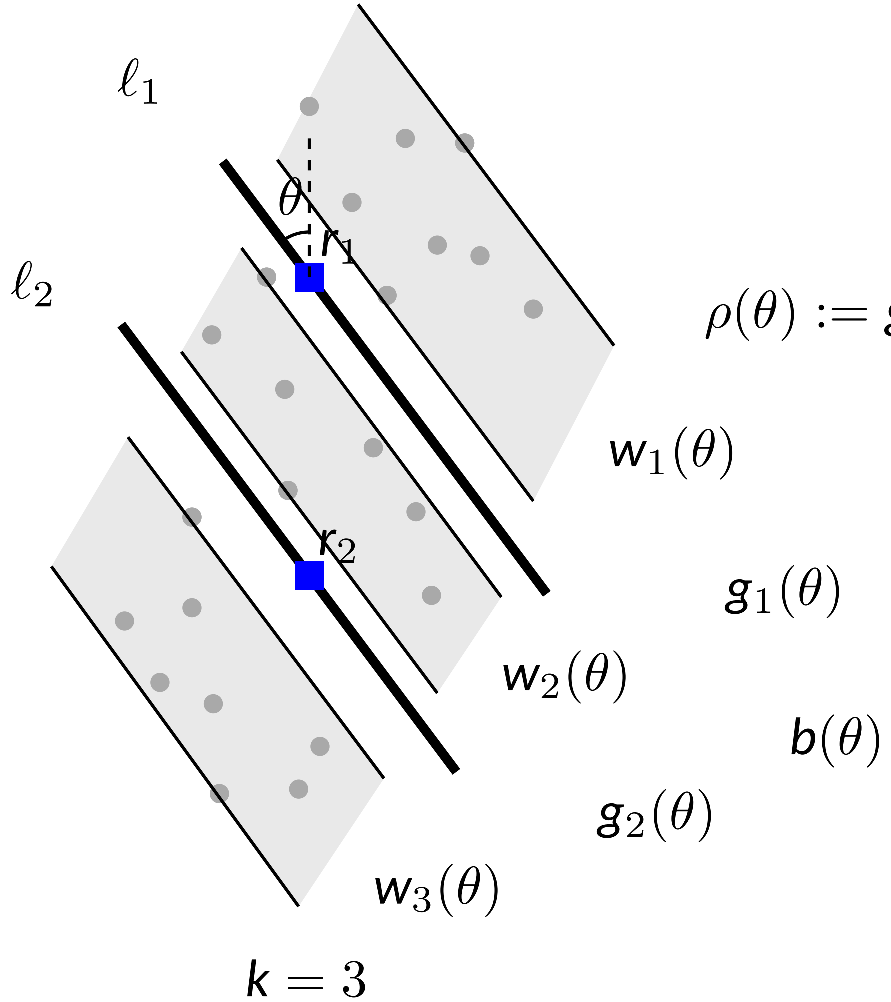
Each of the function

- $w_1, \ldots, w_k$,
- $g_1, \ldots, g_{k-1}$
- b

: piecewise sinusoidal with $O(n)$ breakpoints

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



Each of the function

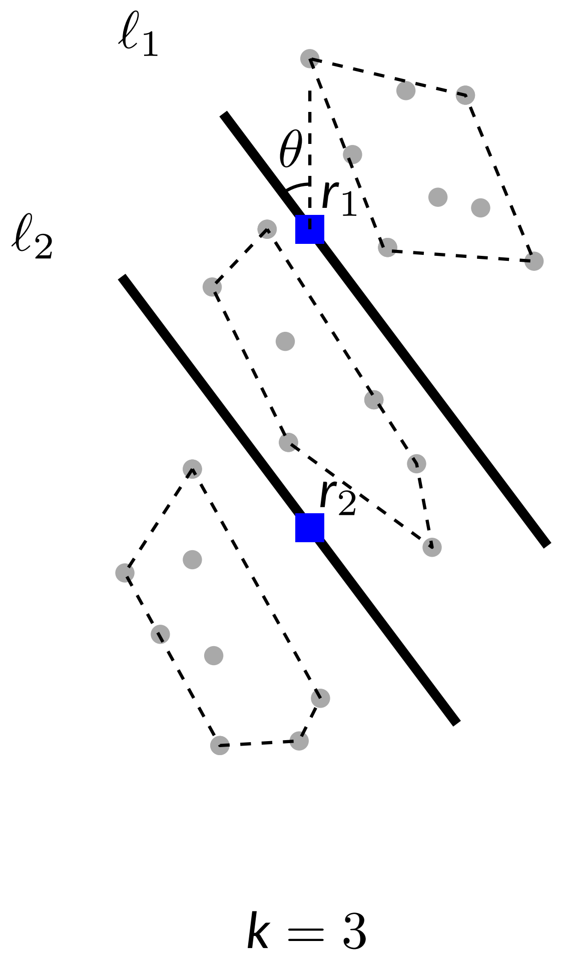
- $w_1, \dots, w_k$,
- $g_1, \dots, g_{k-1}$
- b

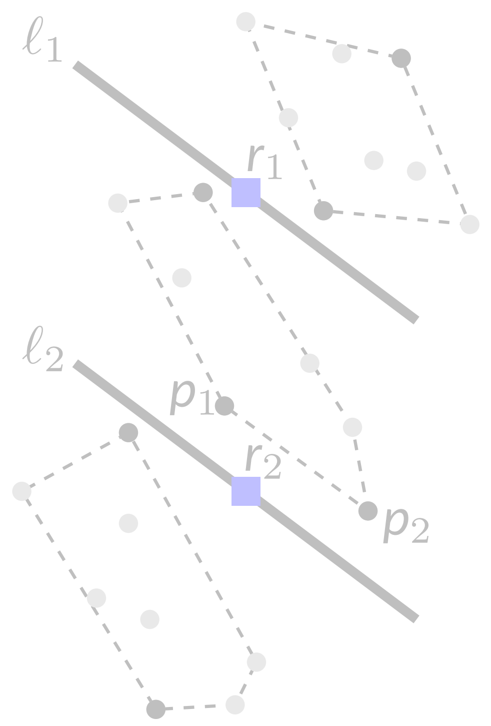
: piecewise sinusoidal with $O(n)$ breakpoints

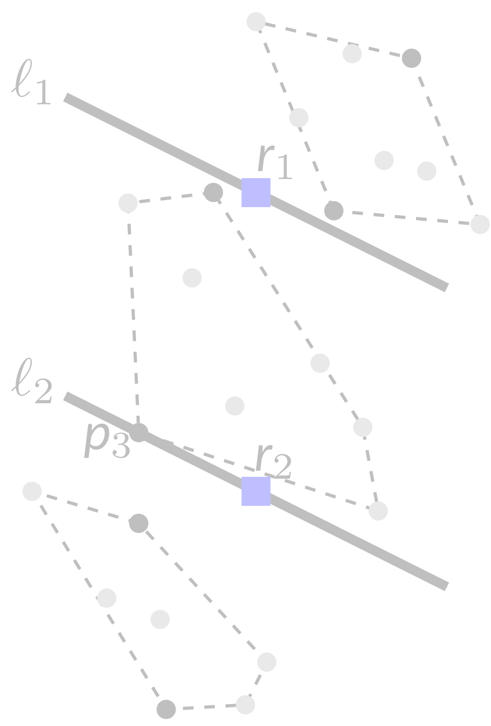
$\rightarrow$ We can compute **the width, gap-width, breadth**, and **the gap ratio** for a fixed orientation.

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

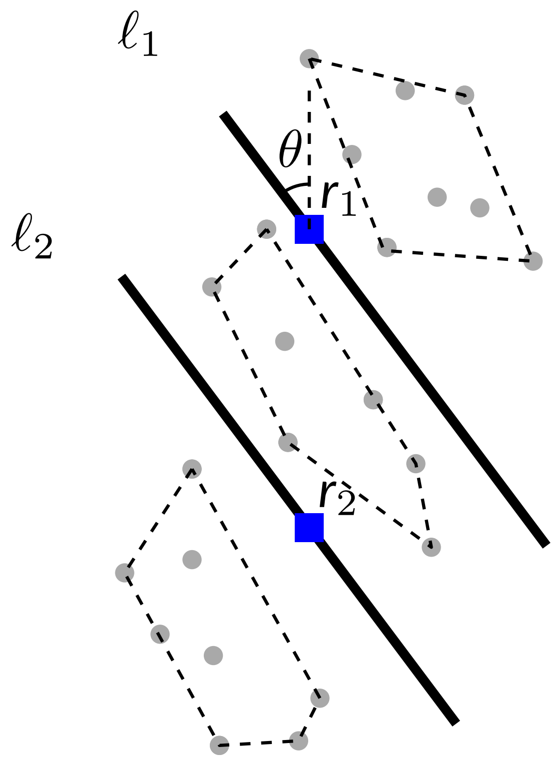




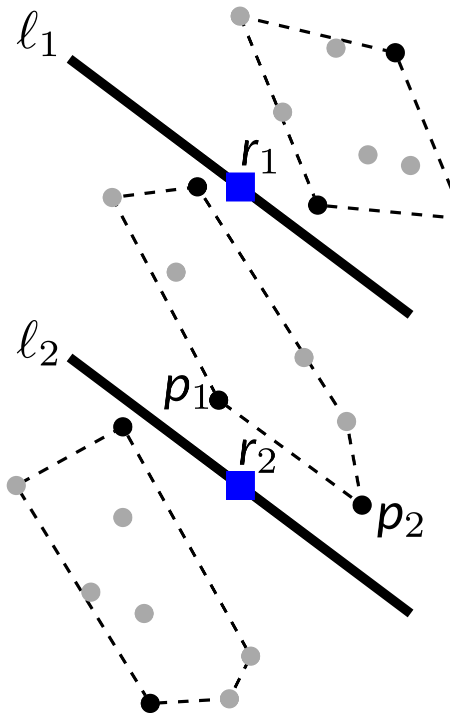


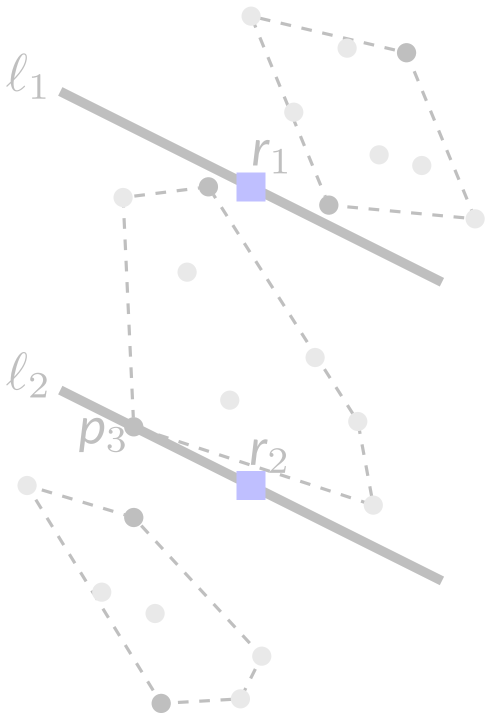
2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



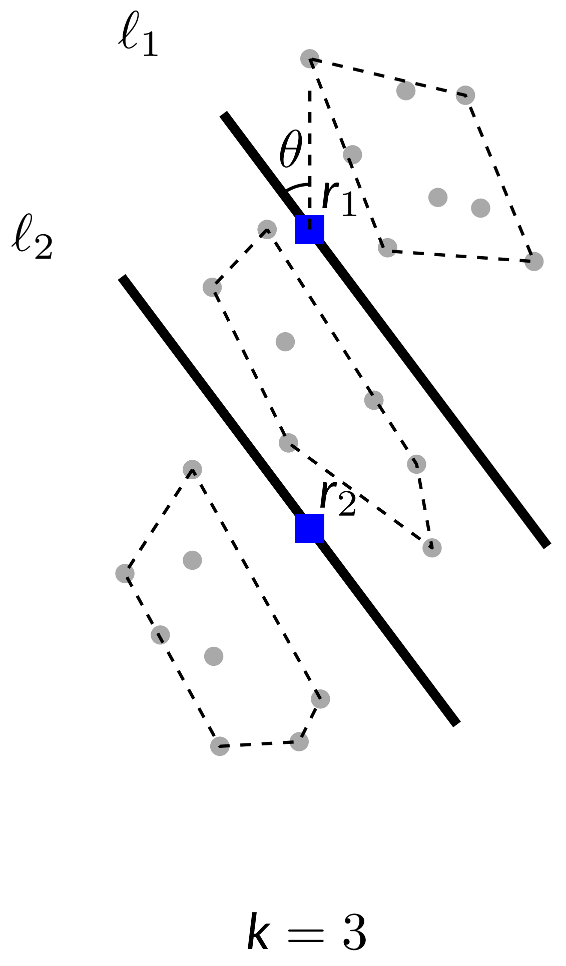
$k = 3$

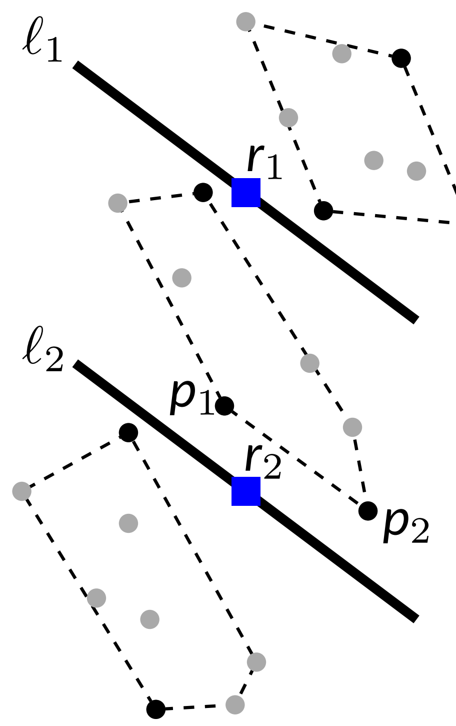


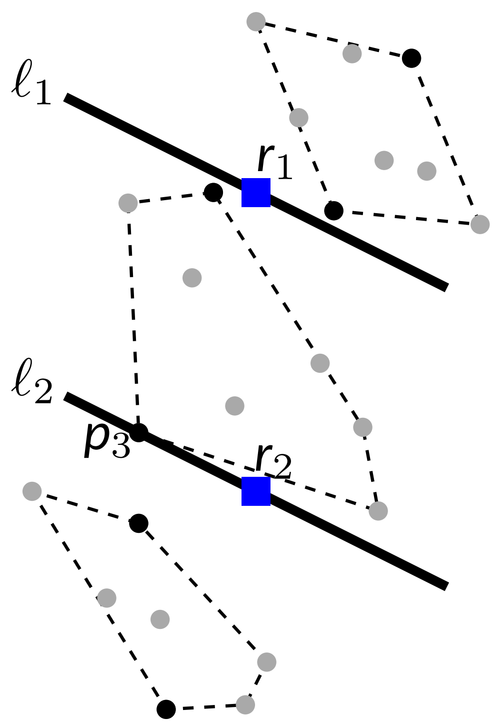


2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

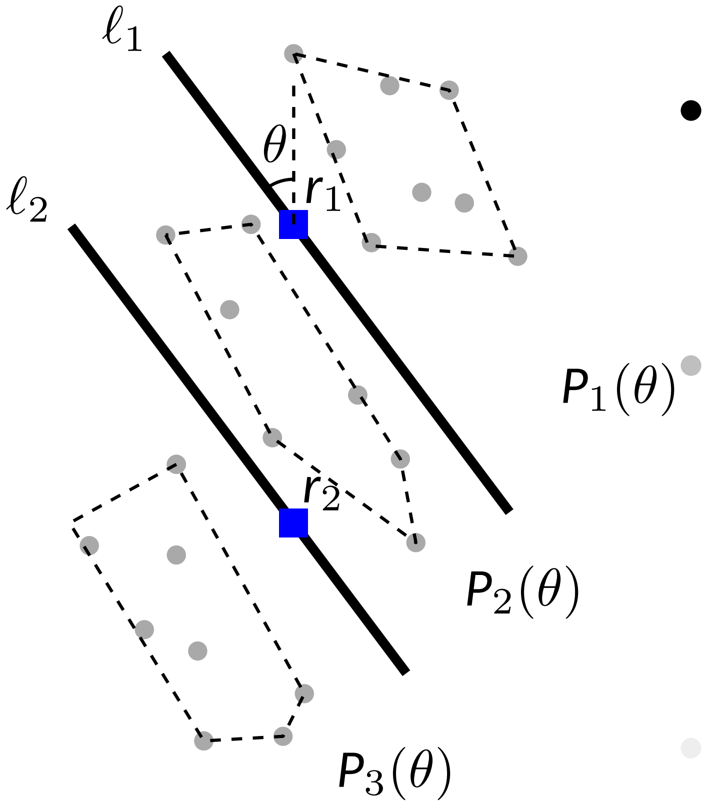






2. A min-width k-slab cover for a Given Separator

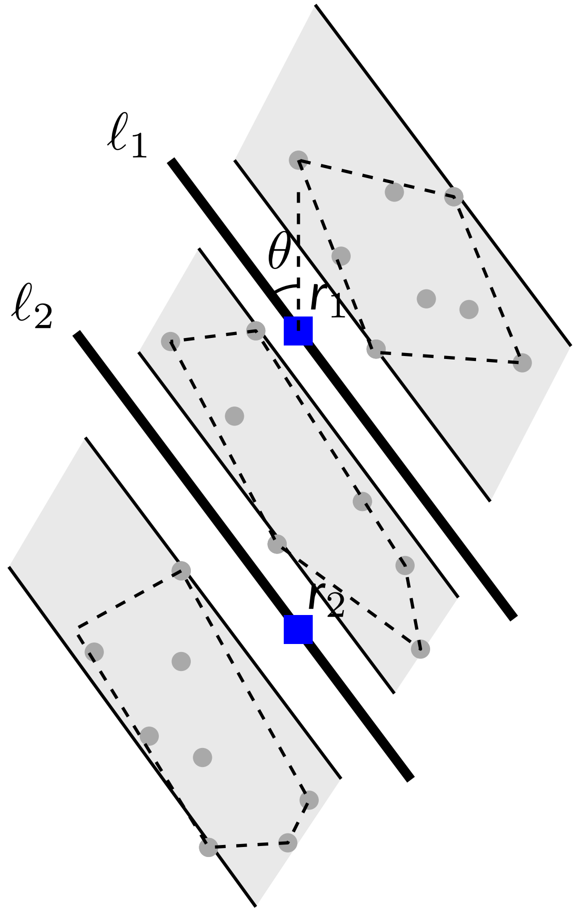
Step 1. Given a separator $R = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



- A fully dynamic structure CH_i for each $i = 1, \dots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.
- $2k$ lists ($W_1, \dots, W_k, G_1, \dots, G_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: $W_1, \dots, W_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: $G_1, \dots, G_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B
- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

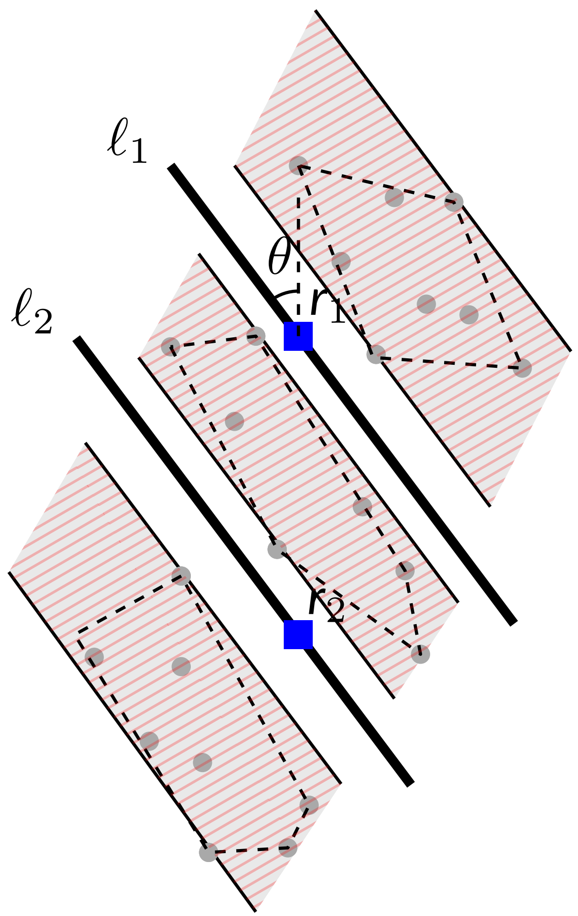
Step 1. Given a separator $R = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



- A fully dynamic structure CH_i for each $i = 1, \dots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.
- $2k$ lists ($W_1, \dots, W_k, G_1, \dots, G_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: $W_1, \dots, W_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: $G_1, \dots, G_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B
- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

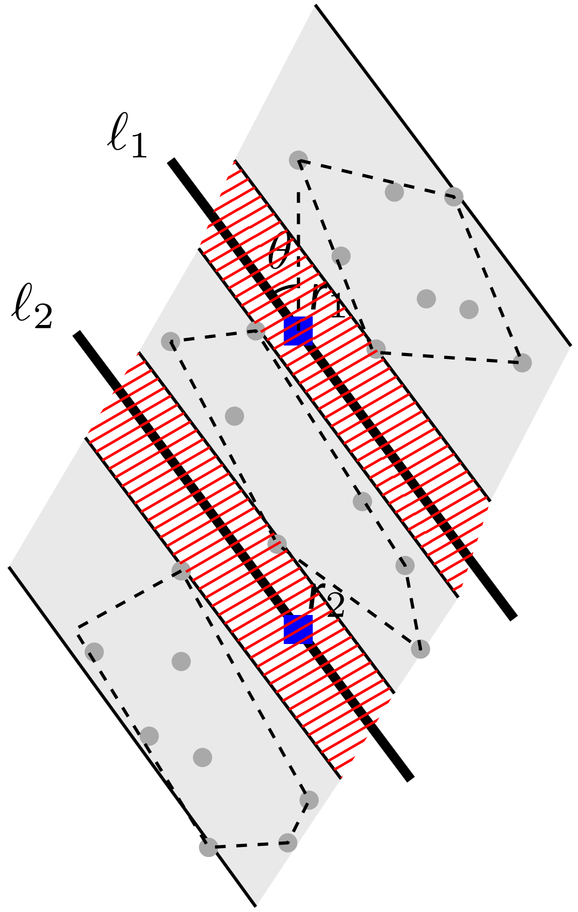
Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



- A fully dynamic structure CH_i for each $i = 1, \ldots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.
- $2k$ lists $(W_1, \ldots, W_k, G_1, \ldots, G_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \ldots, k$: $W_1, \ldots, W_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \ldots, k-1$: $G_1, \ldots, G_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B
- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



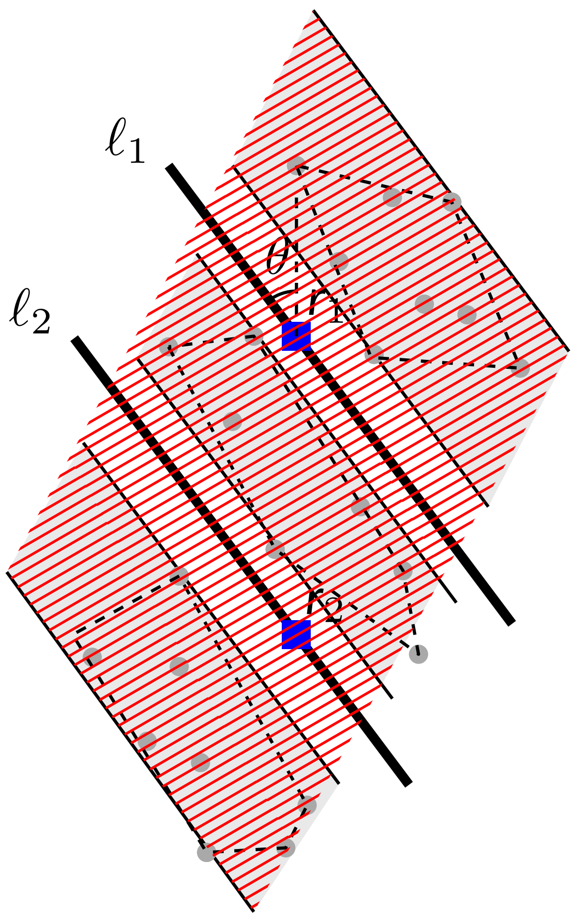
- A fully dynamic structure CH_i for each $i = 1, \ldots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.

- $2k$ lists ($W_1, \ldots, W_k, G_1, \ldots, G_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \ldots, k$: $W_1, \ldots, W_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \ldots, k-1$: $G_1, \ldots, G_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B

- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



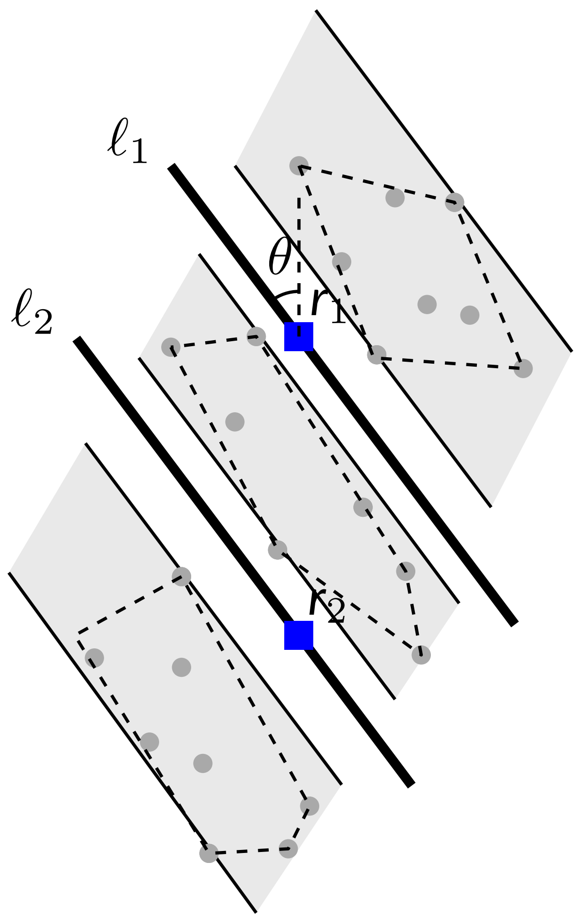
- A fully dynamic structure CH_i for each $i = 1, \dots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.

- $2k$ lists ($\mathrm{W}_1, \dots, \mathrm{W}_k, \mathrm{G}_1, \dots, \mathrm{G}_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: $\mathrm{W}_1, \dots, \mathrm{W}_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: $\mathrm{G}_1, \dots, \mathrm{G}_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B

- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

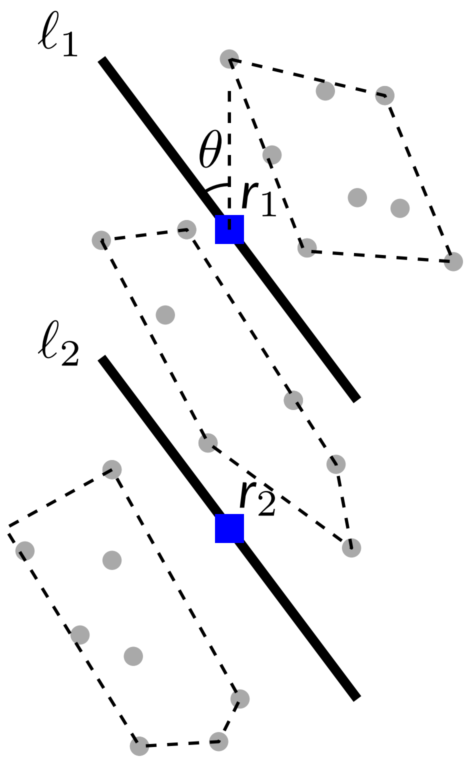
Step 1. Given a separator R $= (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

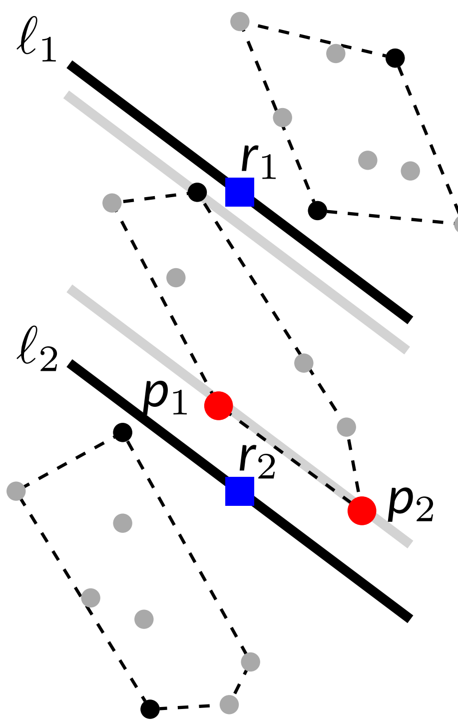


- A fully dynamic structure CH_i for each $i = 1, \dots, k$ [G. S. Brodal and R. Jacob, 2002]
  - $O(\log n)$ time queries using $O(n)$ space.
- $2k$ lists (W$_1, \dots,$ W$_k$, G$_1, \dots,$ G$_{k-1}$, and B)
  - k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: W$_1, \dots,$ W$_k$
  - $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: G$_1, \dots,$ G$_{k-1}$
  - 1 list: Stores a function of $b(\theta)$: B
- The two extreme points $q_i^+(\theta)$ and $q_i^-(\theta)$ for each $P_i(\theta)$

2. A min-width k-slab cover for a Given Separator

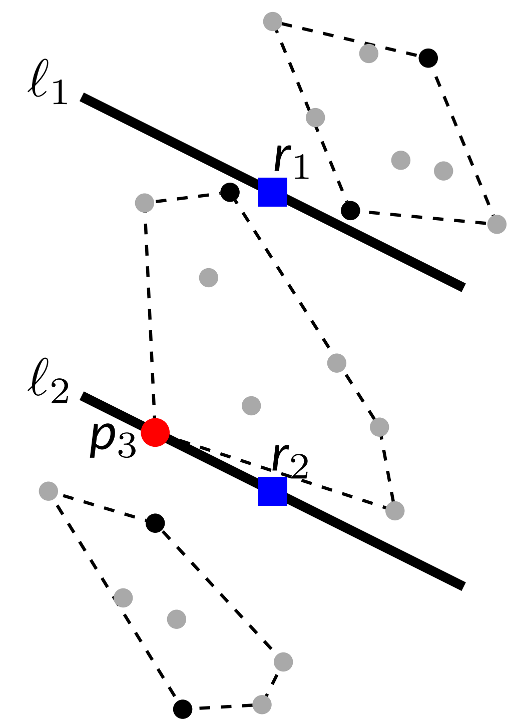
Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.





1) Slab event

: when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$

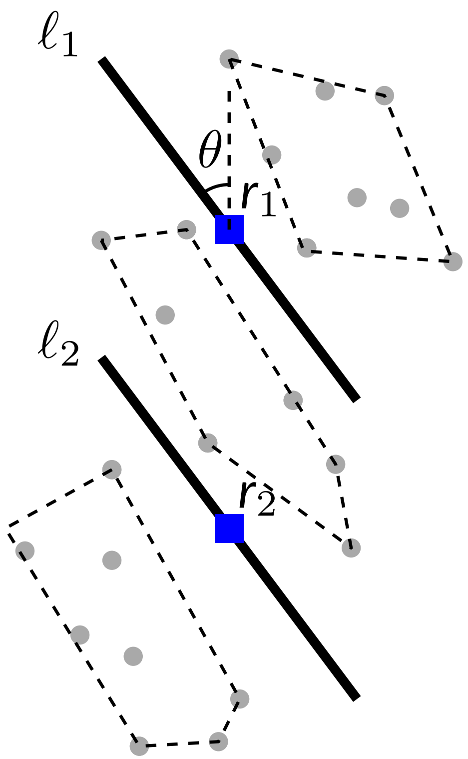


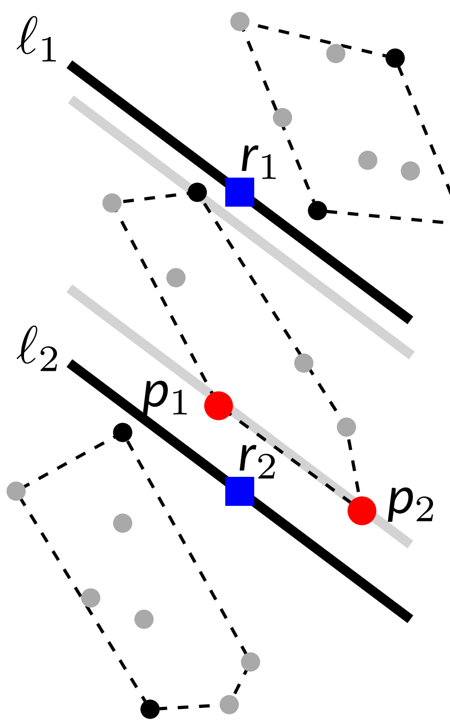
2) Cross event

: when a point in P lies on $\ell_i(\theta)$

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

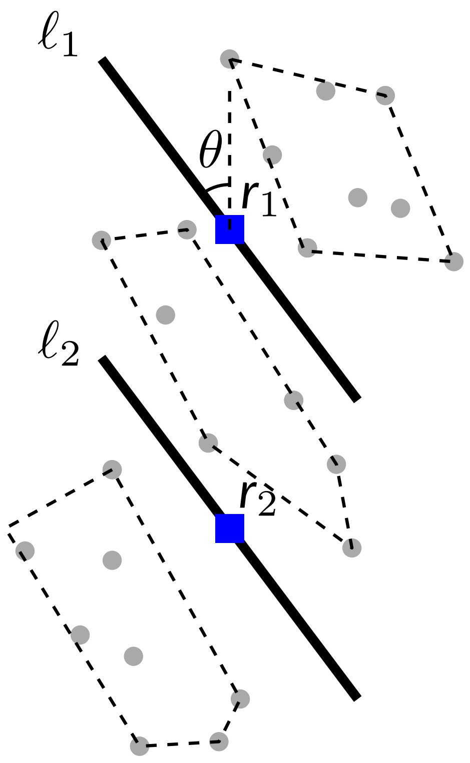


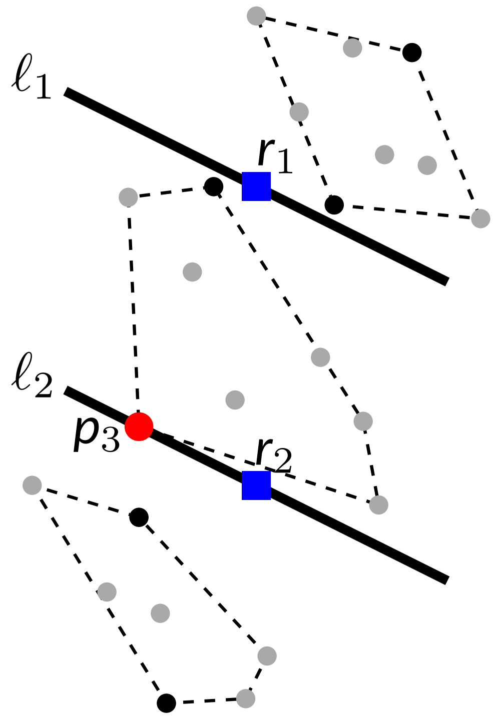


- Update W_i
- Update $\mathrm{G}_{i-1}, \mathrm{G}_i$
- Update B, if needed

1) Slab event
: when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.





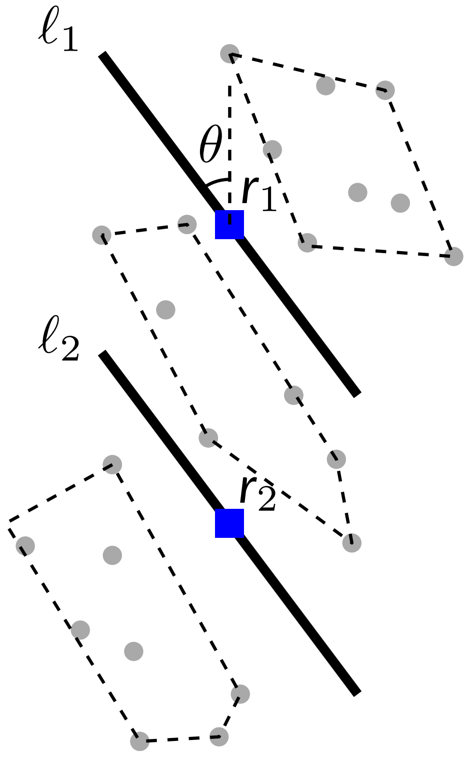
- Update CH_i, C_{i+1}
- Update W_i, W_{i+1}
- Update G_i

2) Cross event

: when a point in P lies on $\ell_i(\theta)$

2. A min-width k-slab cover for a Given Separator

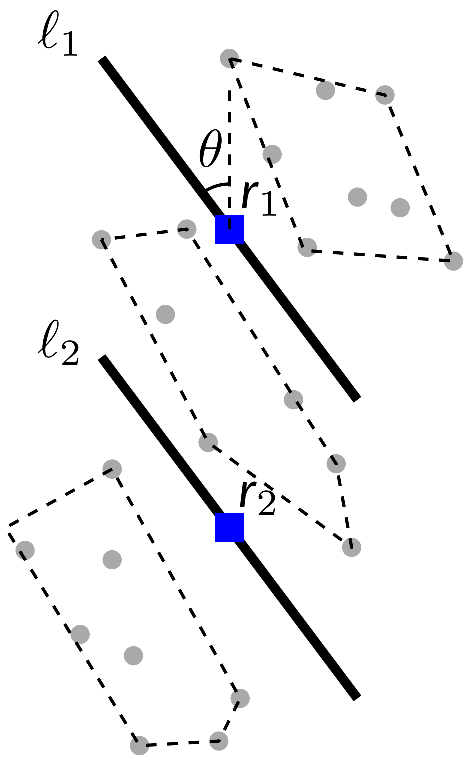
Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



1) **Slab event** : when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$
2) **Cross event** : when a point in P lies on $\ell_i(\theta)$

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

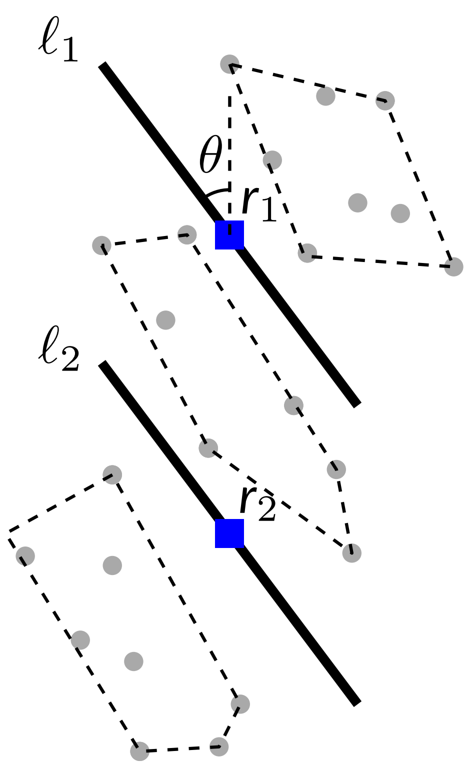


1) **Slab event** : when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$
2) **Cross event** : when a point in P lies on $\ell_i(\theta)$

- The number of events is $O(kn)$
- It takes $O(\log n)$ time for each event. [G. S. Brodal and R. Jacob, 2002]

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



1) **Slab event** : when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$
2) **Cross event** : when a point in P lies on $\ell_i(\theta)$

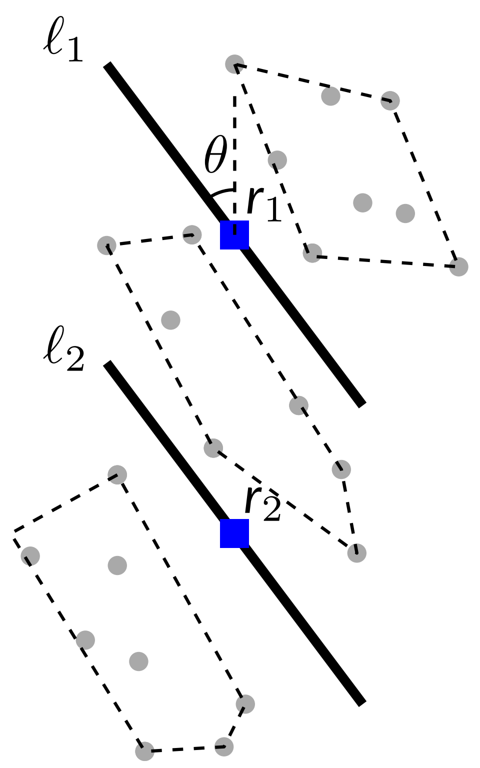
- The number of events is $O(kn)$
- It takes $O(\log n)$ time for each event. [G. S. Brodal and R. Jacob, 2002]

$2k$ lists ($\mathrm{W}_1, \dots, \mathrm{W}_k, \mathrm{G}_1, \dots, \mathrm{G}_{k-1}$, and B)

- k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: $\mathrm{W}_1, \dots, \mathrm{W}_k$
- $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: $\mathrm{G}_1, \dots, \mathrm{G}_{k-1}$
- 1 list: Stores a function of $b(\theta)$: B

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \dots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



1) **Slab event** : when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$
2) **Cross event** : when a point in P lies on $\ell_i(\theta)$

- The number of events is $O(kn)$
- It takes $O(\log n)$ time for each event. [G. S. Brodal and R. Jacob, 2002]

$2k$ lists ($W_1, \dots, W_k, G_1, \dots, G_{k-1}$, and B)

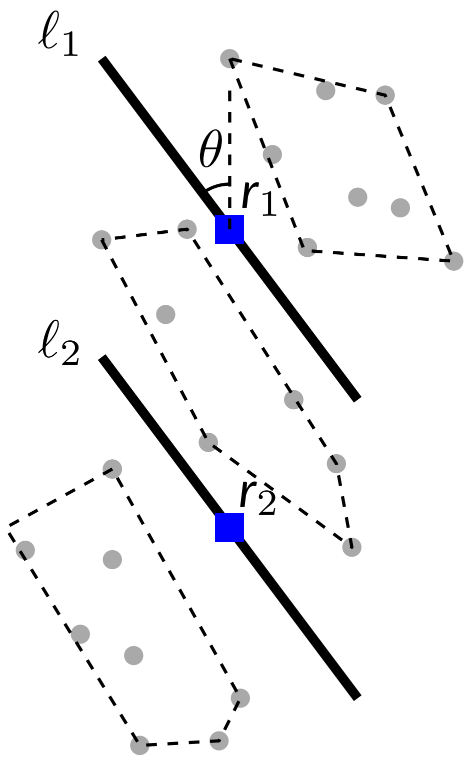
- k lists: Store functions of $w_i(\theta)$ for $i = 1, \dots, k$: $W_1, \dots, W_k$
- $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \dots, k-1$: $G_1, \dots, G_{k-1}$
- 1 list: Stores a function of $b(\theta)$: B

$\rightarrow$ Evaluate $W_1, \dots, W_k, G_1, \dots, G_{k-1}$, and B

: Compute the exact orientation θ where the gap-ratio is at least ρ and the width is minimized.

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $R = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.



1) **Slab event** : when two or more points of P are contained in a boundary line of slab of $P_i(\theta)$
2) **Cross event** : when a point in P lies on $\ell_i(\theta)$

- The number of events is $O(kn)$
- It takes $O(\log n)$ time for each event. [G. S. Brodal and R. Jacob, 2002]

$2k$ lists ($W_1, \ldots, W_k, G_1, \ldots, G_{k-1}$, and B)

- k lists: Store functions of $w_i(\theta)$ for $i = 1, \ldots, k$: $W_1, \ldots, W_k$
- $k-1$ lists: Store functions of $g_i(\theta)$ for $i = 1, \ldots, k-1$: $G_1, \ldots, G_{k-1}$
- 1 list: Stores a function of $b(\theta)$: B

$\rightarrow$ Evaluate $W_1, \ldots, W_k, G_1, \ldots, G_{k-1}$, and B **every *n* events**

: Compute the exact orientation θ where the gap-ratio is at least ρ and the width is minimized.

2. A min-width k-slab cover for a Given Separator

Step 1. Given a separator $\mathrm{R} = (r_1, \ldots, r_{k-1})$, compute a minimum-width k-slab cover of P respecting R.

Theorem. A minimum-width k-slab cover of P respecting R can be computed in $O(kn \log n)$ time and $O(n)$ space.

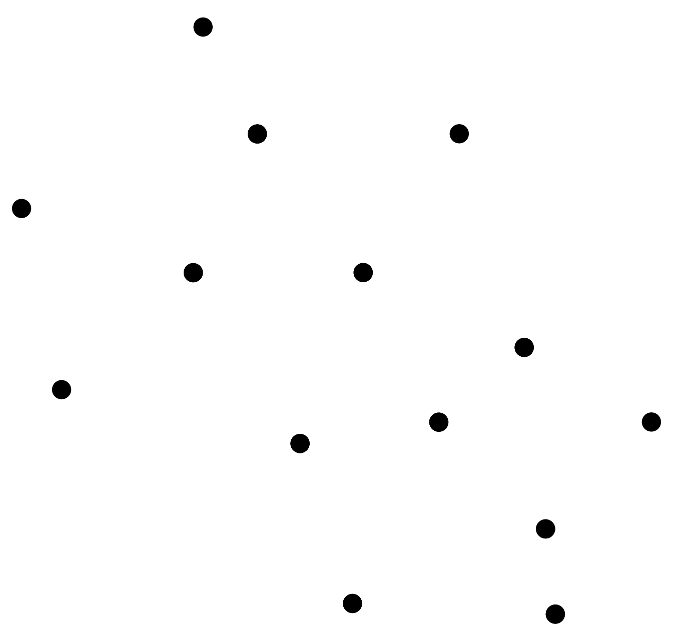
3. Computing Candidate Separators

Step 2. Given a set *P* of *n* points, compute candidates for an optimal separator.

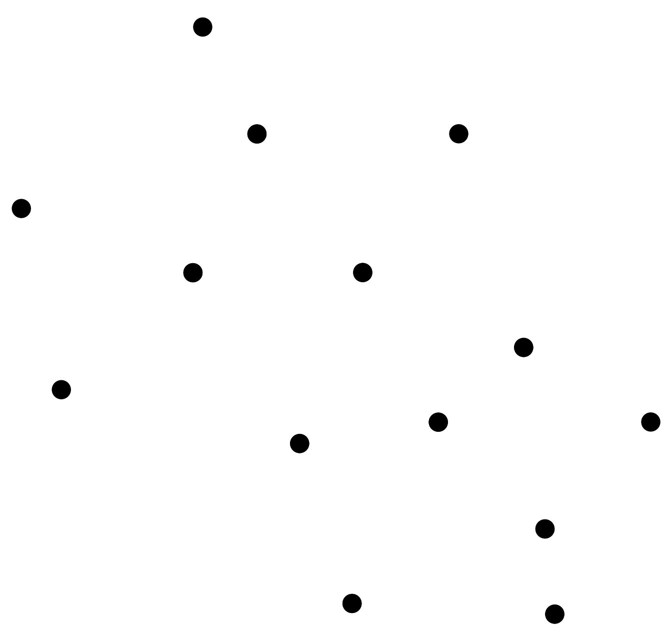
3. Computing Candidate Separators

Step 2. Given a set *P* of *n* points, compute candidates for an optimal separator.

The **directional width** of *P* in orientation θ, $d_\theta(P)$

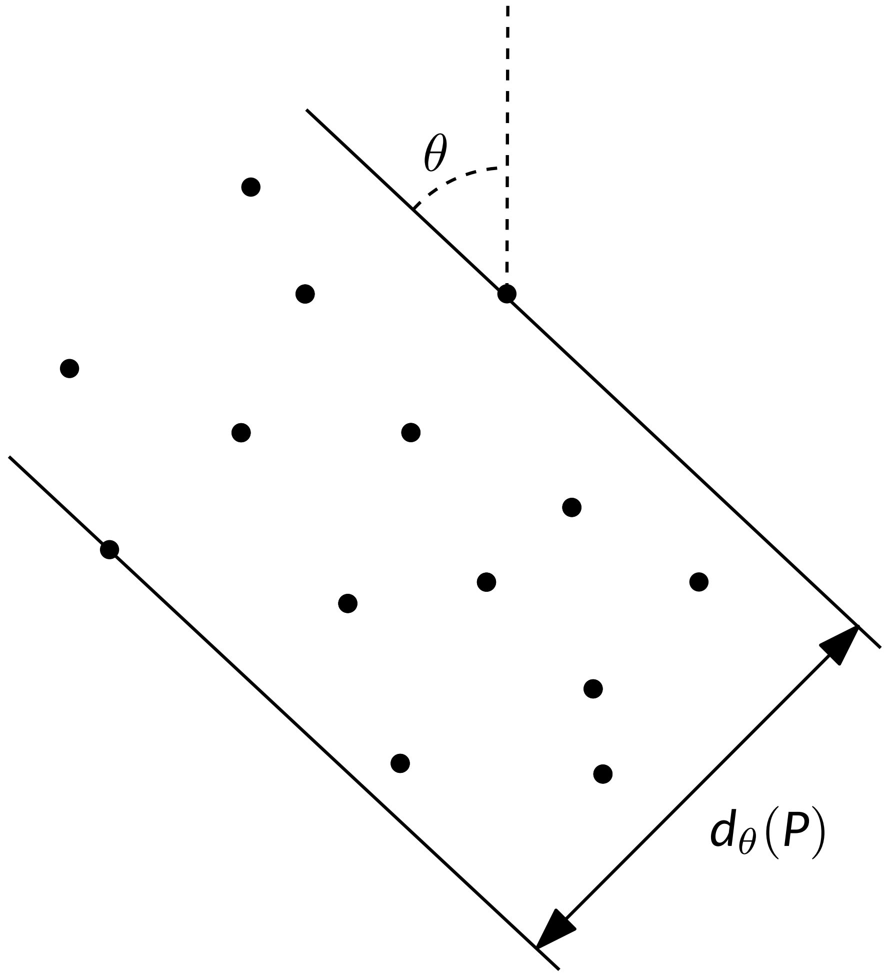


An ϵ**-coreset** for the directional width of *P*

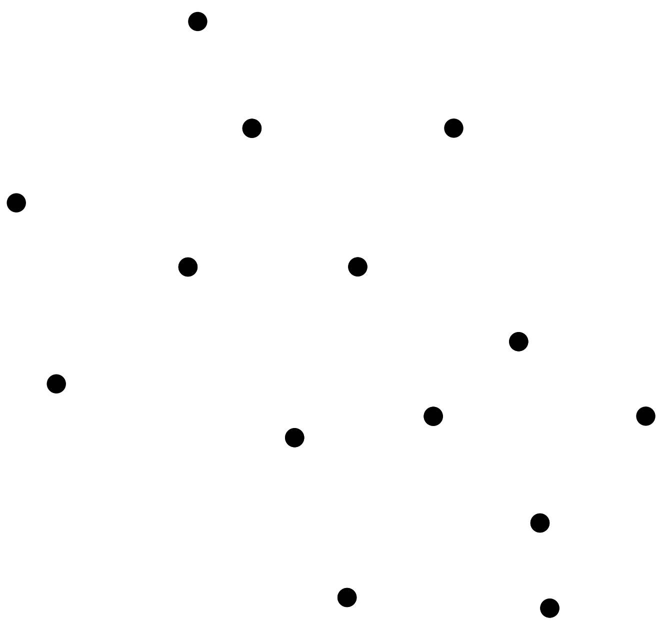


Step 2. Given a set P of n points, compute candidates for an optimal separator.

The **directional width** of P in orientation θ, $d_\theta(P)$



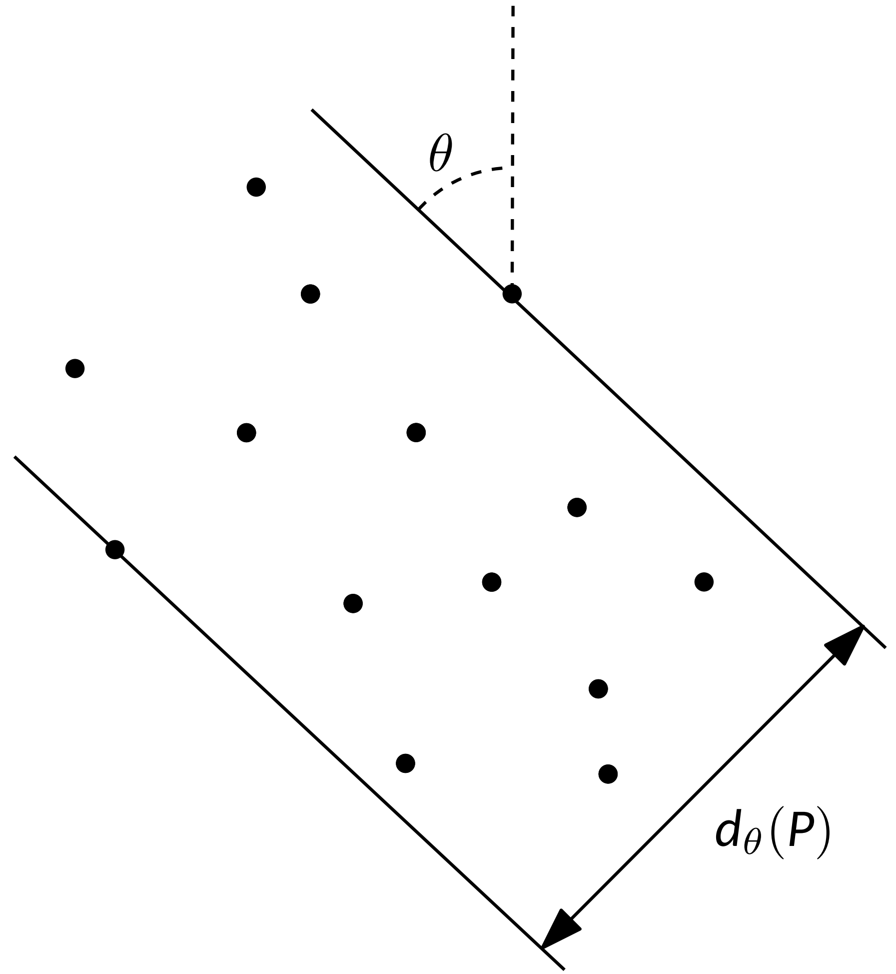
An ϵ**-coreset** for the directional width of P



3. Computing Candidate Separators

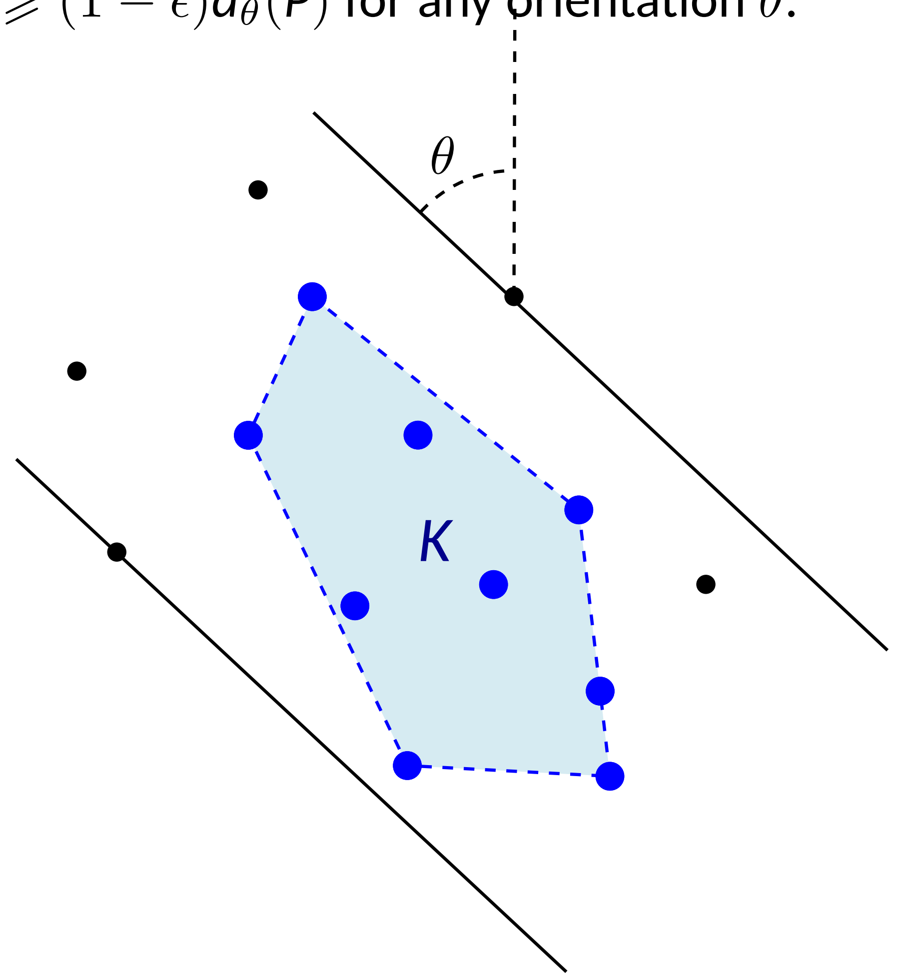
Step 2. Given a set P of n points, compute candidates for an optimal separator.

The **directional width** of P in orientation θ, $d_\theta(P)$



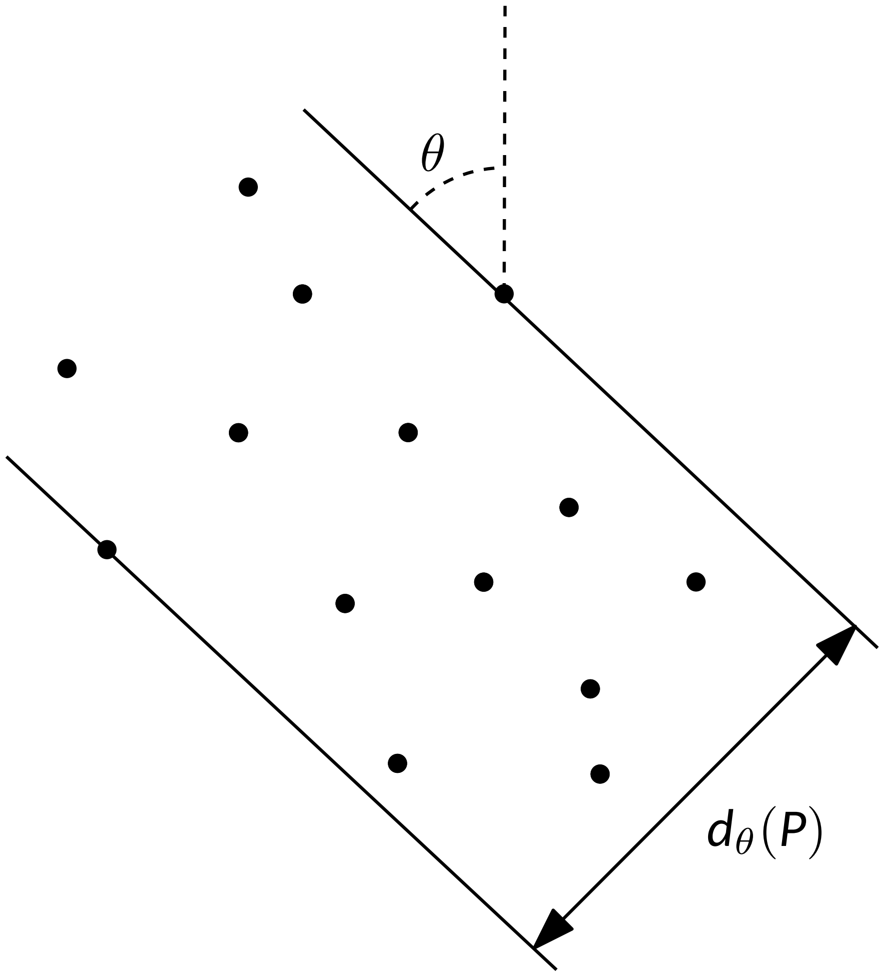
An ϵ**-coreset** for the directional width of P

$d_\theta(K) \geqslant (1-\epsilon)d_\theta(P)$ for any orientation θ.



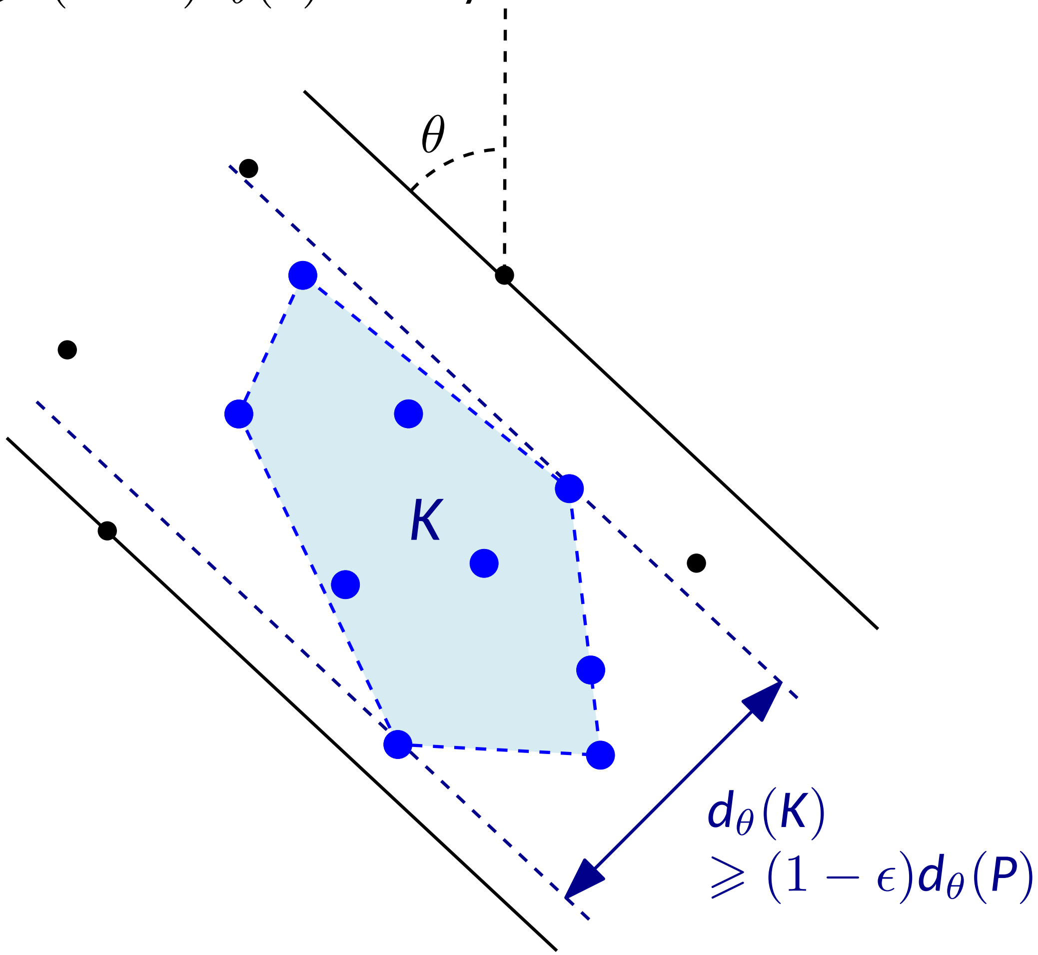
Step 2. Given a set P of n points, compute candidates for an optimal separator.

The **directional width** of P in orientation θ, $d_\theta(P)$



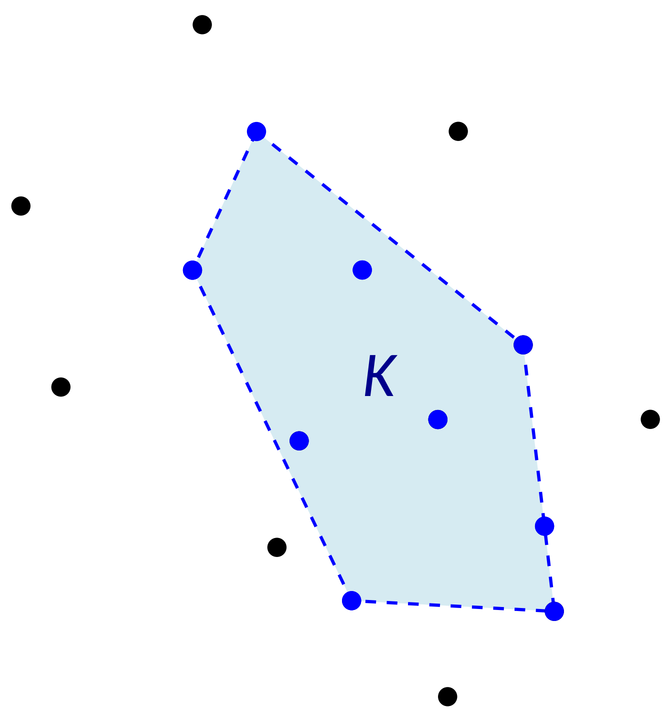
An ϵ**-coreset** for the directional width of P

$d_\theta(K) \geqslant (1-\epsilon)d_\theta(P)$ for any orientation θ.



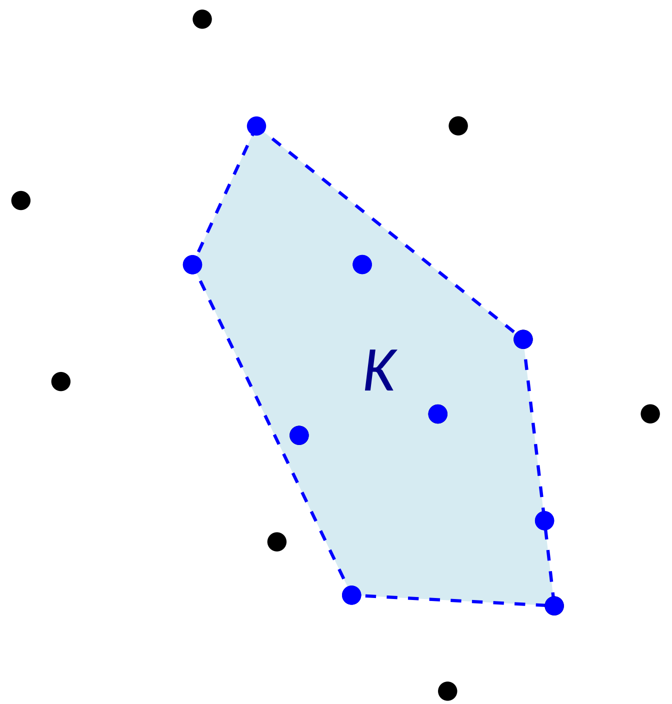
Step 2. Given a set P of n points, compute candidates for an optimal separator.

Let $\mathbf{K} \subset P$ be a $(\rho/2)$-coreset for directional width of P.



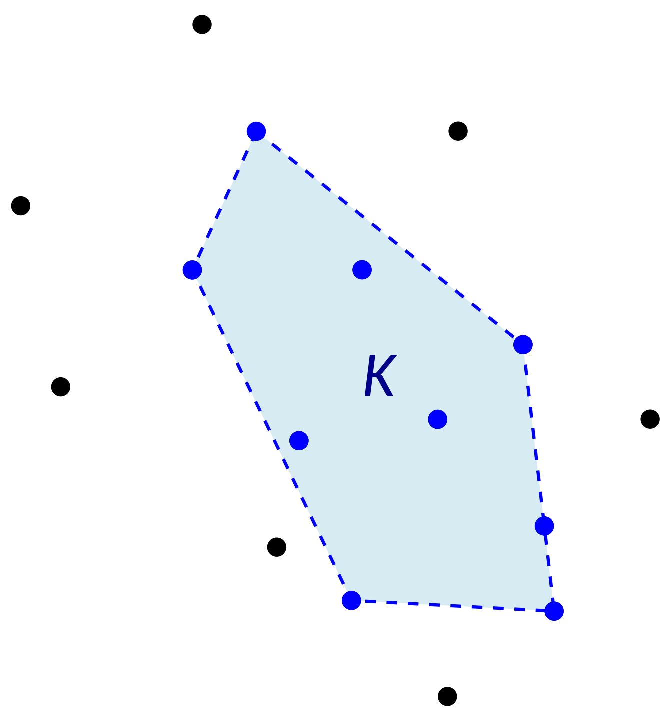
Step 2. Given a set P of n points, compute candidates for an optimal separator.

Let **K** $\subset P$ be a $(\rho/2)$-coreset for directional width of P. $\rightarrow$ K of size $O(1/\rho)$ can be computed in $O(n)$ time.



Step 2. Given a set P of n points, compute candidates for an optimal separator.

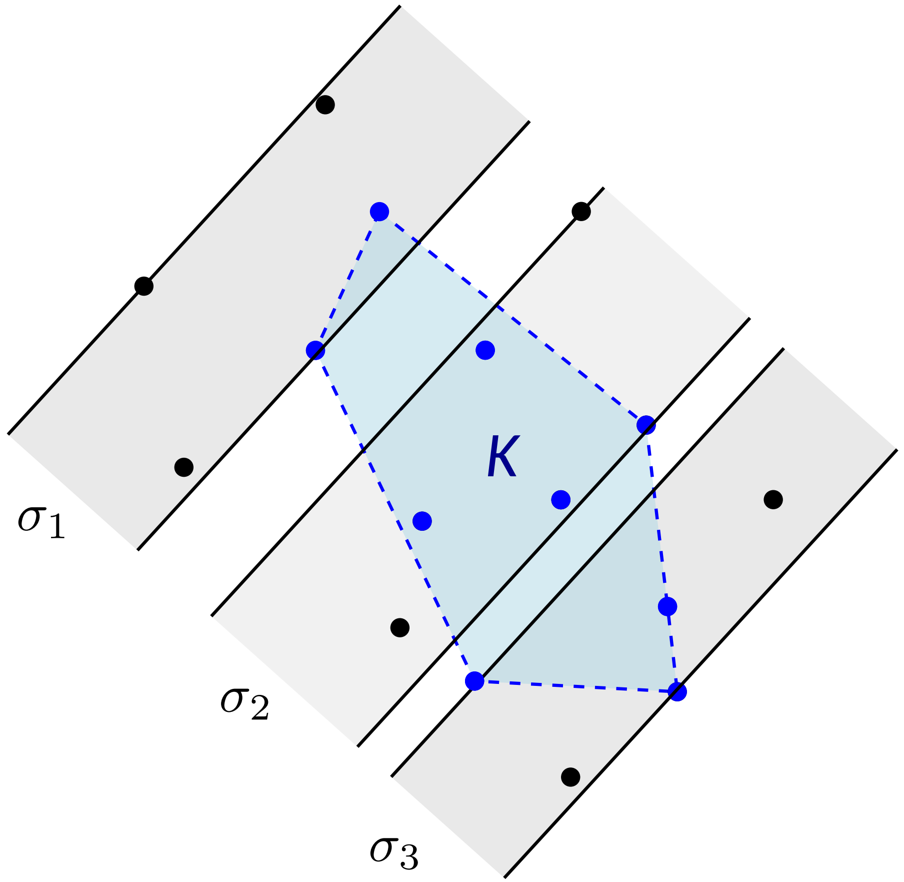
Let $\mathbf{K} \subset P$ be a $(\rho/2)$-coreset for directional width of P. $\rightarrow$ K of size $O(1/\rho)$ can be computed in $O(n)$ time.



Lemma. For any k-slab cover $S = (\sigma_1, \dots, \sigma_k)$ of P, it holds that $K \cap \sigma_1 \neq \emptyset$ and $K \cap \sigma_k \neq \emptyset$.

Step 2. Given a set P of n points, compute candidates for an optimal separator.

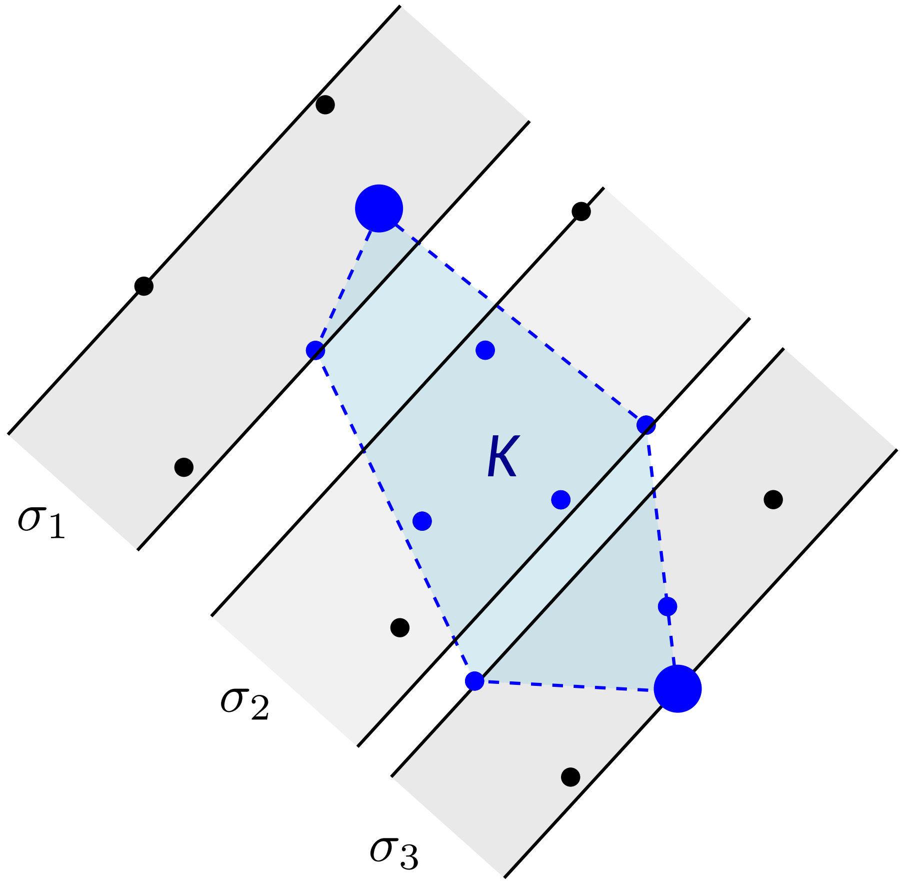
Let $\mathbf{K} \subset P$ be a $(\rho/2)$-coreset for directional width of P. $\rightarrow$ K of size $O(1/\rho)$ can be computed in $O(n)$ time.



Lemma. For any k-slab cover $S = (\sigma_1, \ldots, \sigma_k)$ of P, it holds that $K \cap \sigma_1 \neq \emptyset$ and $K \cap \sigma_k \neq \emptyset$.

Step 2. Given a set P of n points, compute candidates for an optimal separator.

Let $\mathbf{K} \subset P$ be a $(\rho/2)$-coreset for directional width of P. $\rightarrow K$ of size $O(1/\rho)$ can be computed in $O(n)$ time.



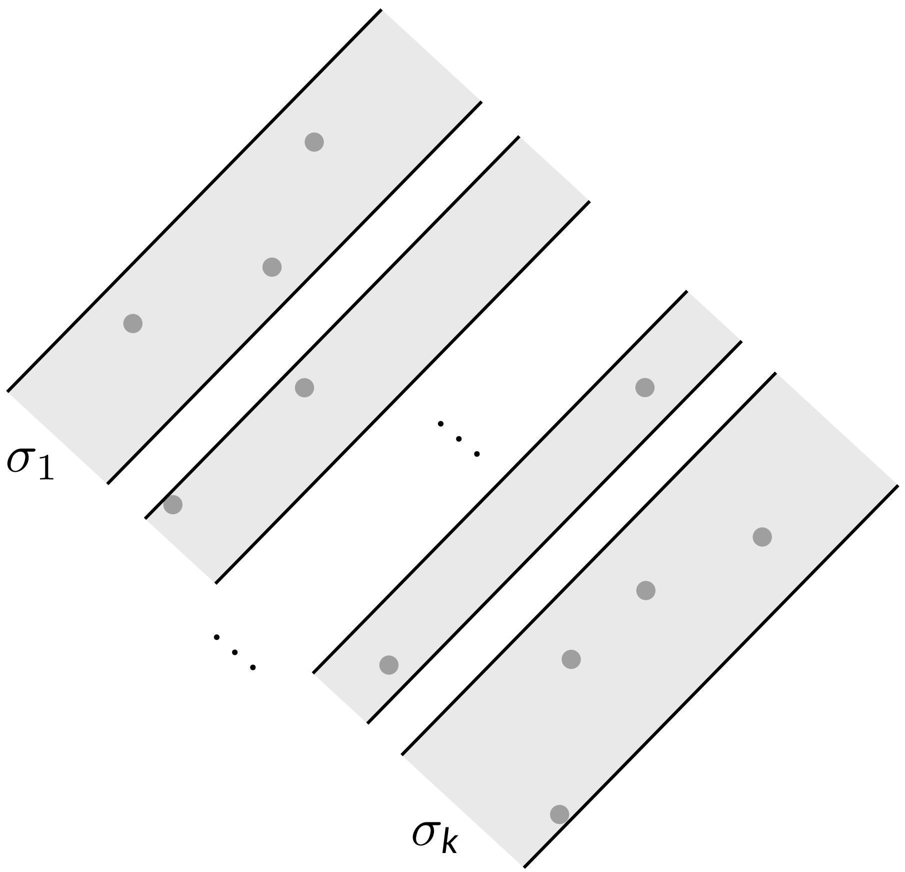
Lemma. For any k-slab cover $S = (\sigma_1, \ldots, \sigma_k)$ of P, it holds that $K \cap \sigma_1 \neq \emptyset$ and $K \cap \sigma_k \neq \emptyset$.

Corollary. There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.

3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

For an arbitrary k-slab cover S of P,

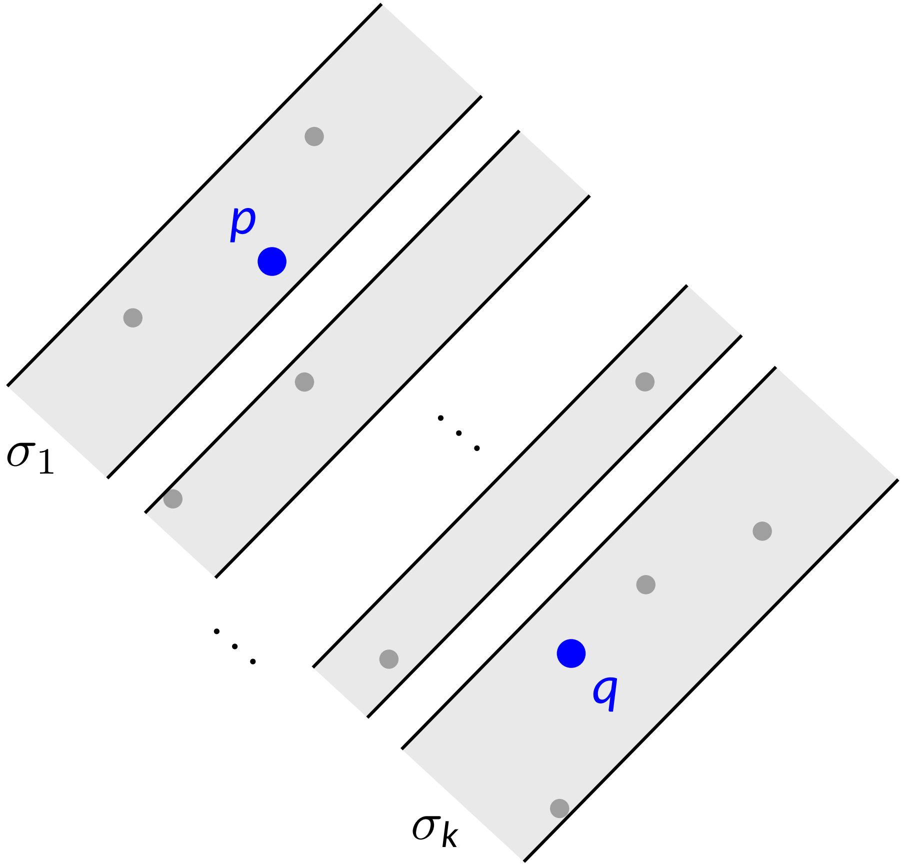


3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

For an arbitrary k-slab cover S of P,

Let p and q be any two points such that $p \in \sigma_1$ and $q \in \sigma_k$.

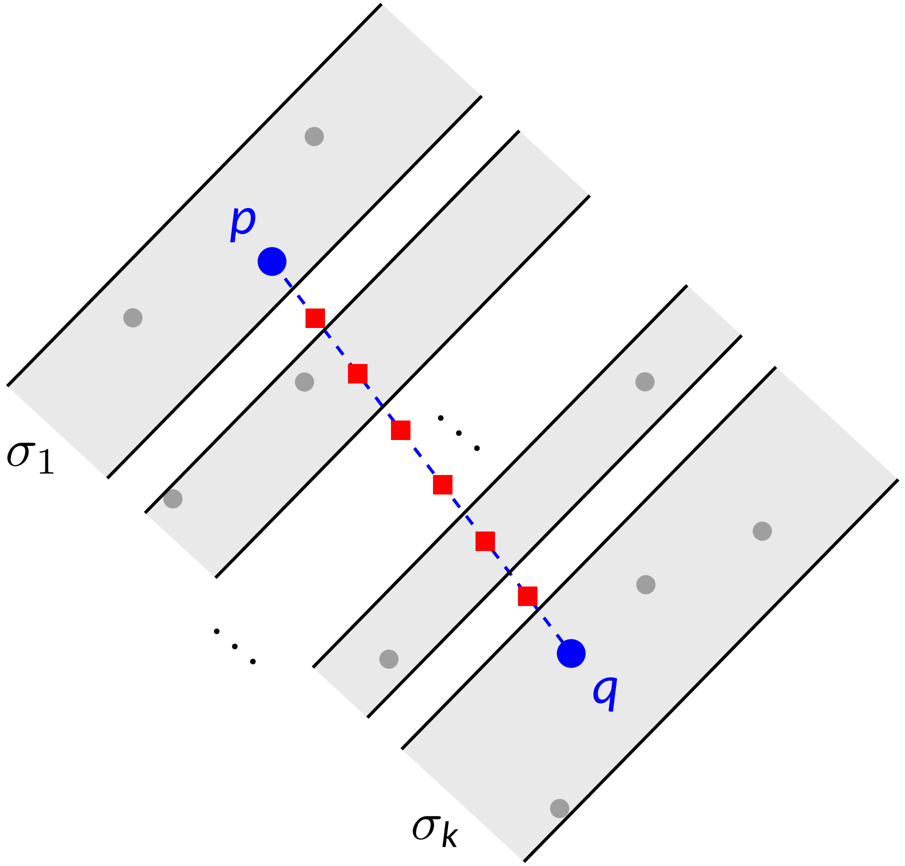


3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

For an arbitrary k-slab cover S of P,

Let p and q be any two points such that $p \in \sigma_1$ and $q \in \sigma_k$.



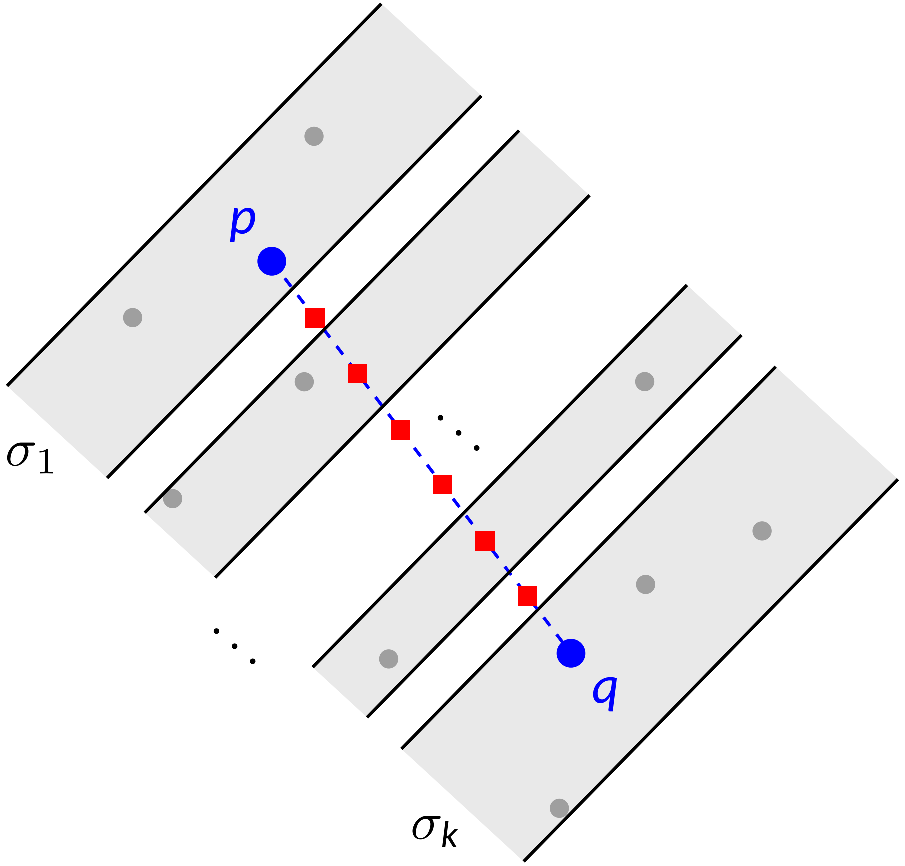
R_{pq} : $\lceil 1/\rho \rceil$ equidistant points on $\overline{pq}$

3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

For an arbitrary k-slab cover S of P,

Let p and q be any two points such that $p \in \sigma_1$ and $q \in \sigma_k$.



R_{pq} : $\lceil 1/\rho \rceil$ equidistant points on $\overline{pq}$

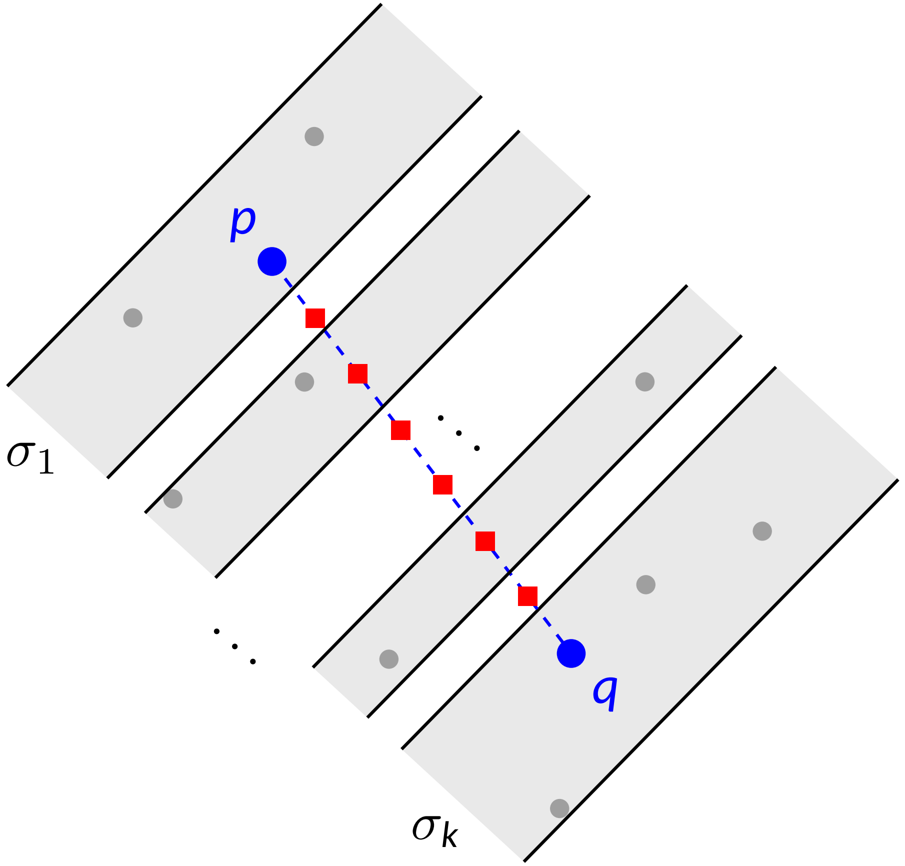
Lemma. Each gap of S contains at least one point in R_{pq}

3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

For an arbitrary k-slab cover S of P,

Let p and q be any two points such that $p \in \sigma_1$ and $q \in \sigma_k$.



R_{pq} : $\lceil 1/\rho \rceil$ equidistant points on $\overline{pq}$

Lemma. Each gap of S contains at least one point in R_{pq}

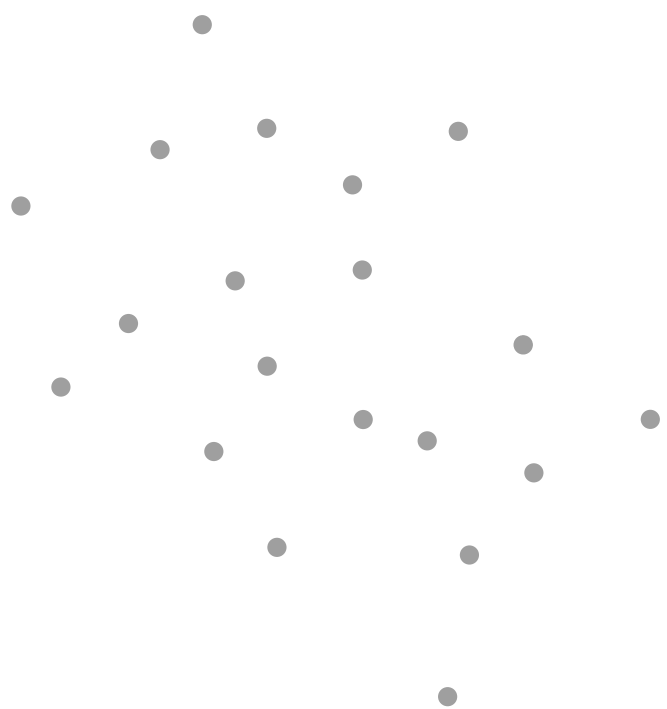
↓

There is a separator $(r_1, \ldots, r_{k-1})$ of S such that $r_i \in R_{pq}$

3. Computing Candidate Separators

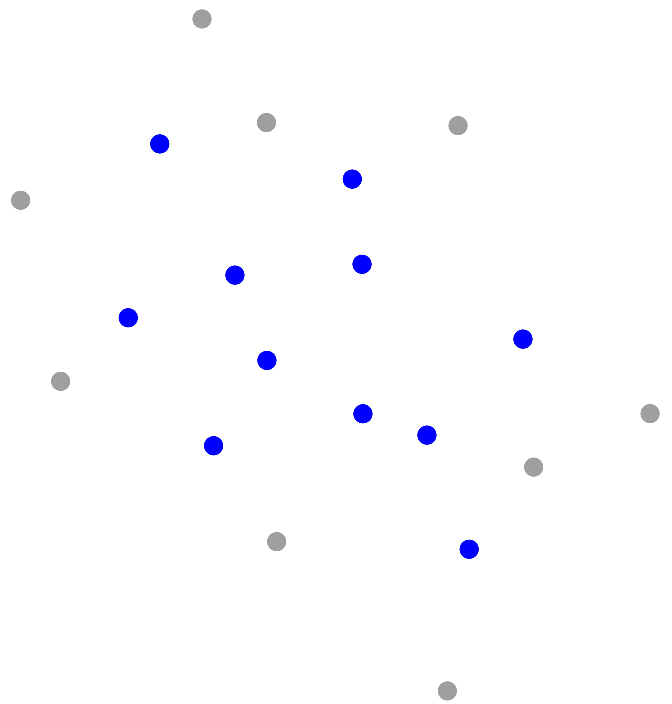
Step 2. Given a set *P* of *n* points, compute candidates for an optimal separator.

1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$



Step 2. Given a set P of n points, compute candidates for an optimal separator.

1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$



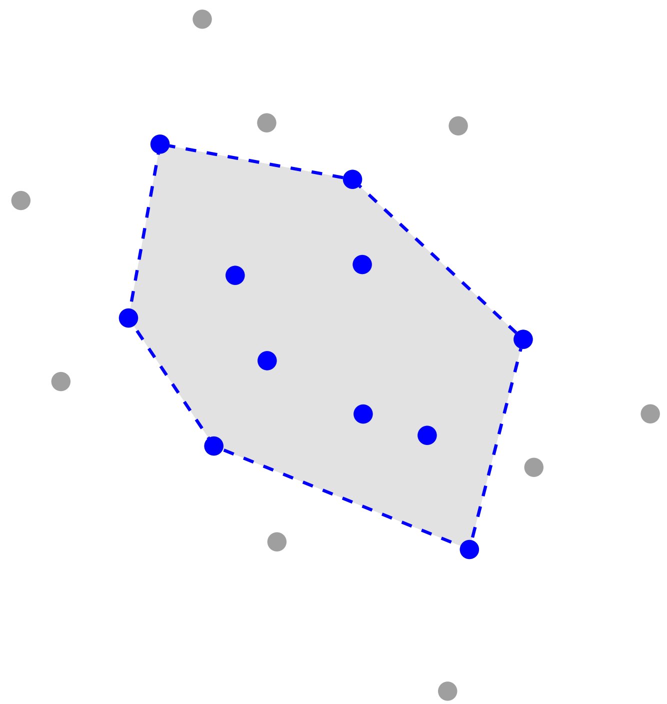
3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$

2) Compute all antipodal pairs of the convex hull of K

Corollary. There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.



3. Computing Candidate Separators

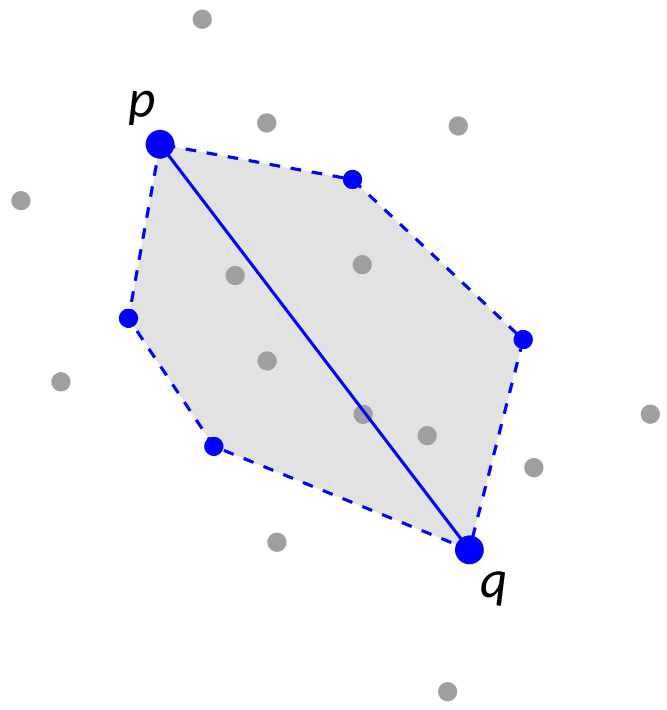
Step 2. Given a set P of n points, compute candidates for an optimal separator.

1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$

2) Compute all antipodal pairs of the convex hull of K

Corollary. There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.

3) For every antipodal pair (p, q), generate $\lceil 1/\rho \rceil$ equidistant points



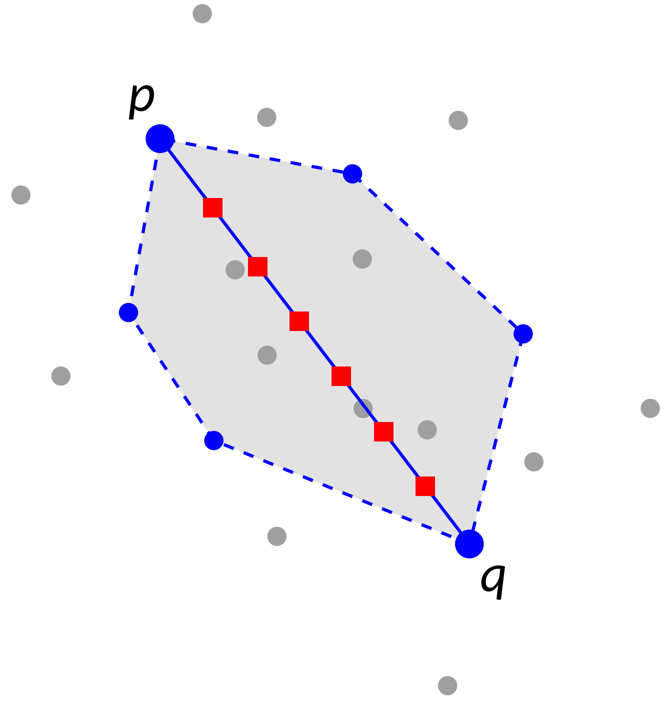
Step 2. Given a set P of n points, compute candidates for an optimal separator.

1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$

2) Compute all antipodal pairs of the convex hull of K ←

Corollary. There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.

3) For every antipodal pair (p, q), generate $\lceil 1/\rho \rceil$ equidistant points



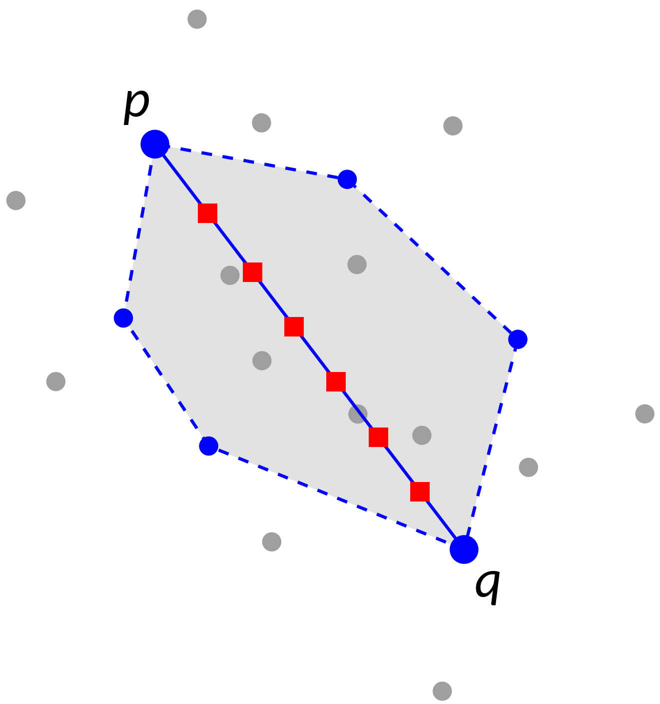
3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

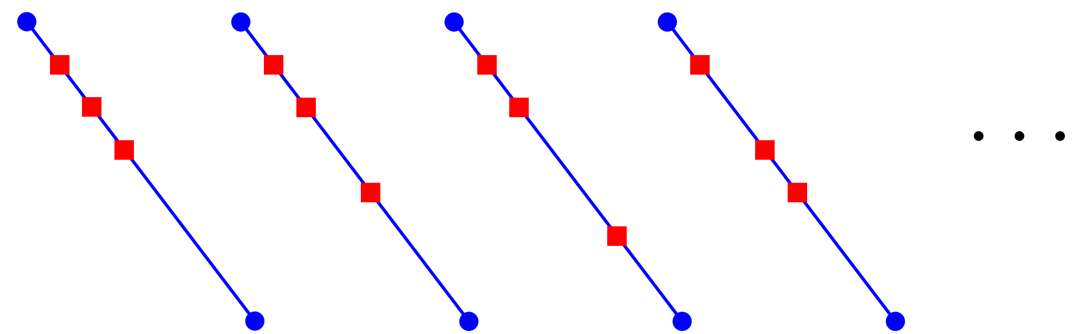
1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$

2) Compute all antipodal pairs of the convex hull of K ← **Corollary.** There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.

3) For every antipodal pair (p, q), generate $\lceil 1/\rho \rceil$ equidistant points & consider all possible $(k-1)$-combinations of the points as candidate separators ← **Lemma.** Each gap of S contains at least one point in R_{pq}



When $k = 4$



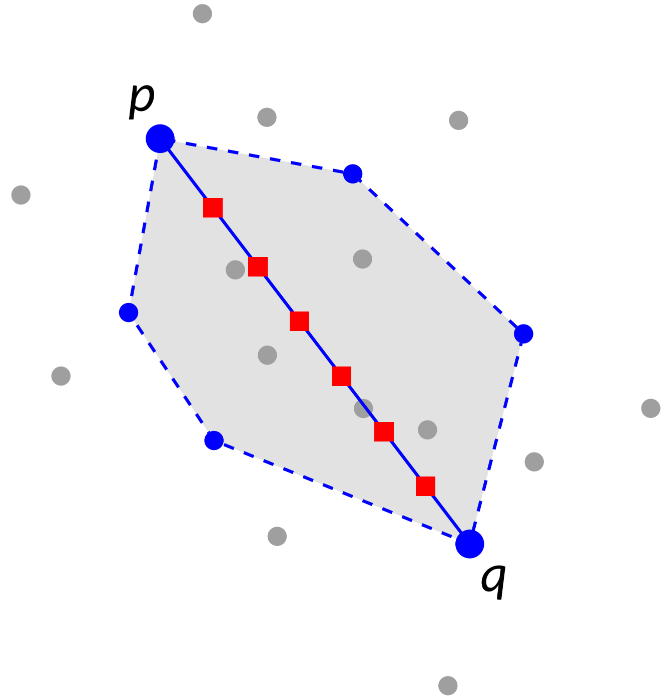
3. Computing Candidate Separators

Step 2. Given a set P of n points, compute candidates for an optimal separator.

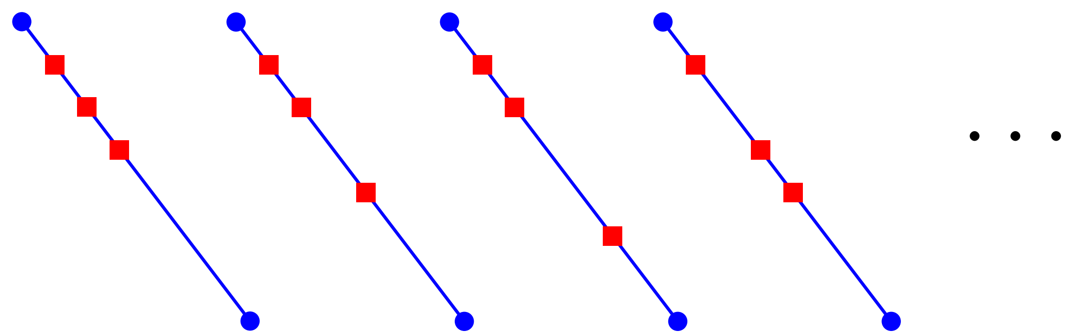
1) Compute a $(\rho/2)$-coreset $K \subseteq P$ of size $O(1/\rho)$

2) Compute all antipodal pairs of the convex hull of K ← **Corollary.** There is an antipodal pair (p, q) of K such that $p \in \sigma_1$ and $q \in \sigma_k$.

3) For every antipodal pair (p, q), generate $\lceil 1/\rho \rceil$ equidistant points & consider all possible $(k-1)$-combinations of the points as candidate separators ← **Lemma.** Each gap of S contains at least one point in R_{pq}



When $k = 4$



$|K| = O(1/\rho)$, # of combinations per antipodal pair = $O(1/\rho^{k-1})$

$\rightarrow$ # of candidate separators is $O(1/\rho^k)$

4. Algorithm

1) Given a separator R, we can compute a minimum width k-slab cover of P respecting R in $O(kn \log n)$ time and $O(n)$ space.

2) We can compute $O(\rho^{-k})$ candidate separators in $O(n \log n + \rho^{-k})$ time.

4. Algorithm

1) Given a separator R, we can compute a minimum width k-slab cover of P respecting R in $O(kn \log n)$ time and $O(n)$ space.

2) We can compute $O(\rho^{-k})$ candidate separators in $O(n \log n + \rho^{-k})$ time.

Theorem. A minimum-width k-slab cover of P whose gap-ratio is at least ρ can be computed in $O(\rho^{-k} \cdot kn \log n)$ time and $O(n)$ space, if exists.

Thank you!