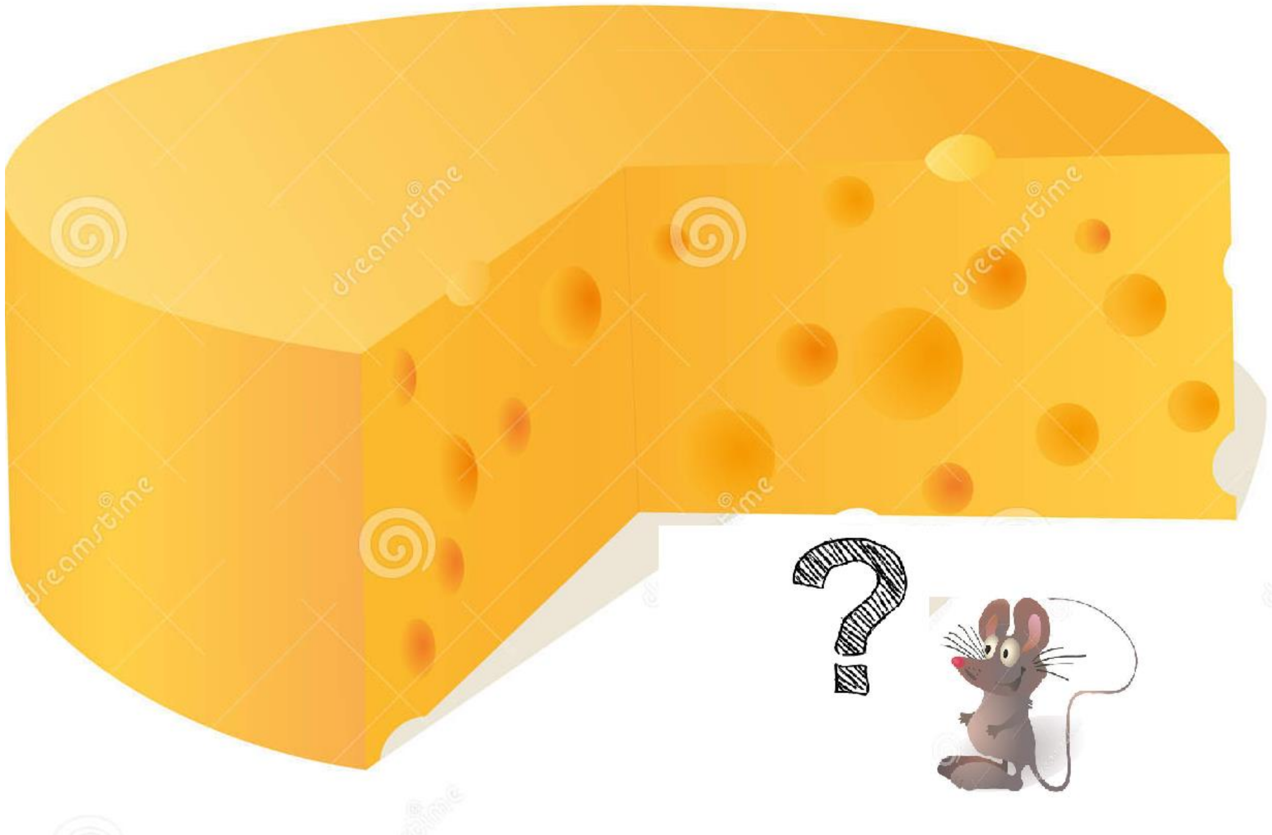
Big Data Class



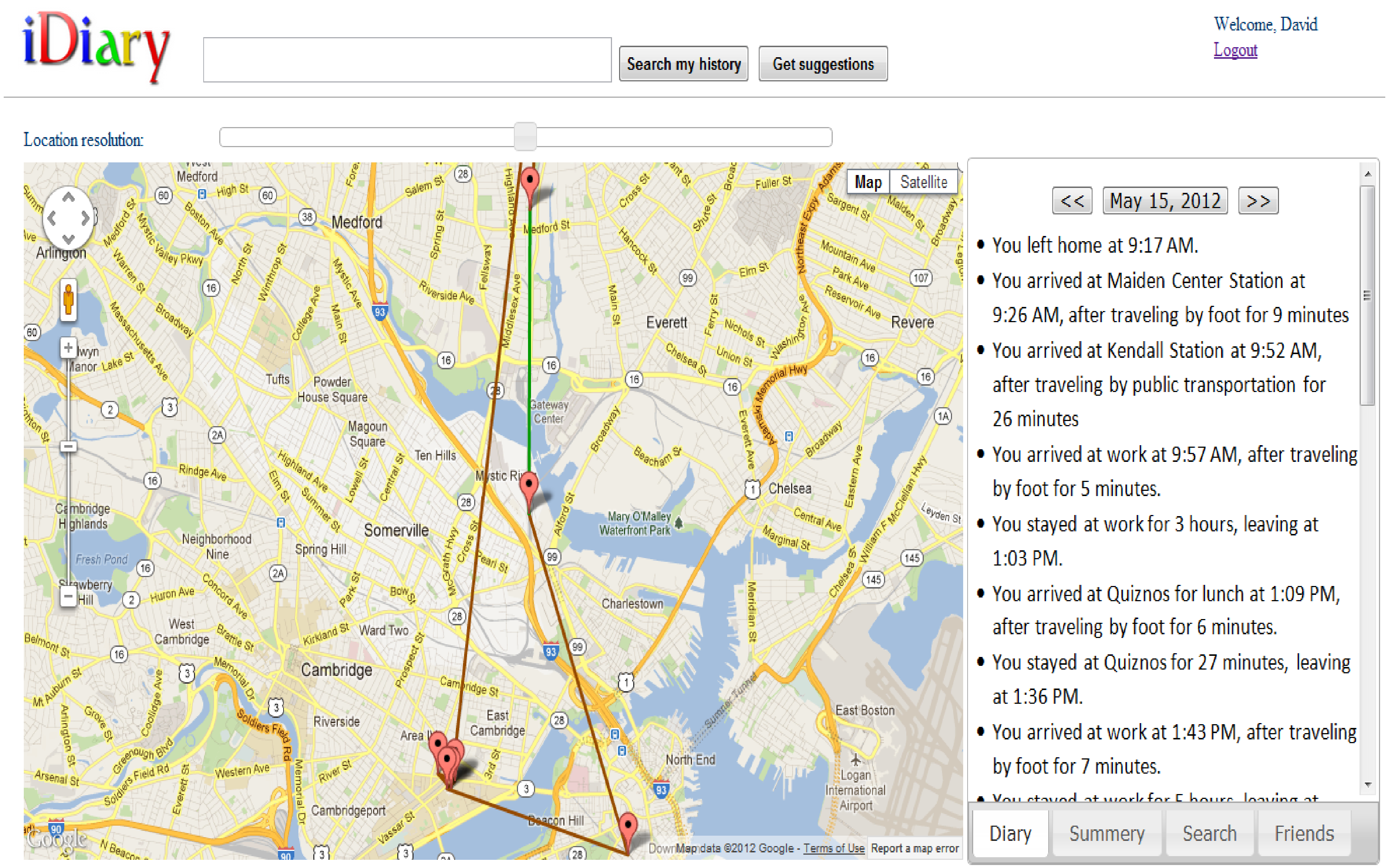
LECTURER: DAN FELDMAN

TEACHING ASSISTANTS:

IBRAHIM JUBRAN

ALAA MAALOUF

אוניברסיטת חיפה
University of Haifa
جامعة حيفا


iDiary
Search my history
Get suggestions
Welcome, David
Logout
Location resolution:
Map
Satellite
<<
May 15, 2012
>>
You left home at 9:17 AM.
You arrived at Maiden Center Station at 9:26 AM, after traveling by foot for 9 minutes
You arrived at Kendall Station at 9:52 AM, after traveling by public transportation for 26 minutes
You arrived at work at 9:57 AM, after traveling by foot for 5 minutes.
You stayed at work for 3 hours, leaving at 1:03 PM.
You arrived at Quiznos for lunch at 1:09 PM, after traveling by foot for 6 minutes.
You stayed at Quiznos for 27 minutes, leaving at 1:36 PM.
You arrived at work at 1:43 PM, after traveling by foot for 7 minutes.
Diary
Summery
Search
Friends
Medford
Arlington
Everett
Revere
Somerville
Chelsea
Charlestown
Cambridge
East Cambridge
North End
Beacon Hill
East Boston
Logan International Airport
Map data ©2012 Google - Terms of Use
Report a map error

iDiary

restaurants July 11 | Search my history | Get suggestions

Welcome, Gary
Logout

Restaurants you visited on July 11th, 2012

1. ## Anna's Taqueria

   You were here on July 11th from 7:03 PM to 7:31 PM, with John Smith, Foo Bar, and 3 OTHERS.

   You have been here 142 OTHER TIMES.

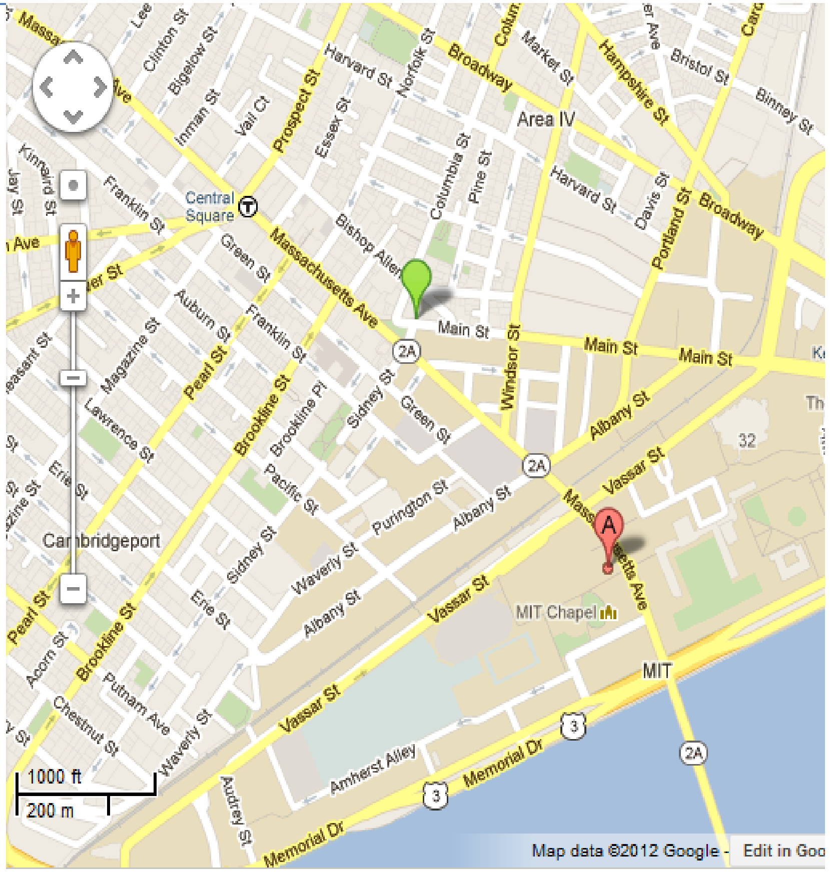
   VIEW SIMILAR RESTAURANTS

2. ## Toscanini's Ice Cream

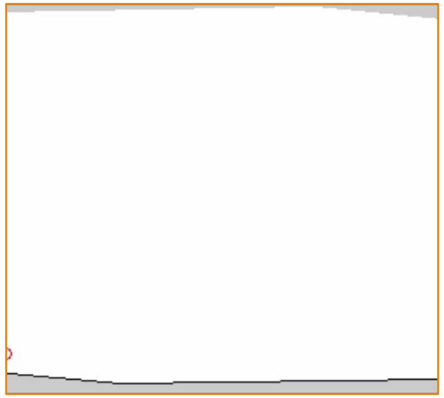
   You were here on July 11th from 7:44 PM to 7:58 PM, with Tim Yang, John Smith, and 4 OTHERS.

   You have been here 17 OTHER TIMES.

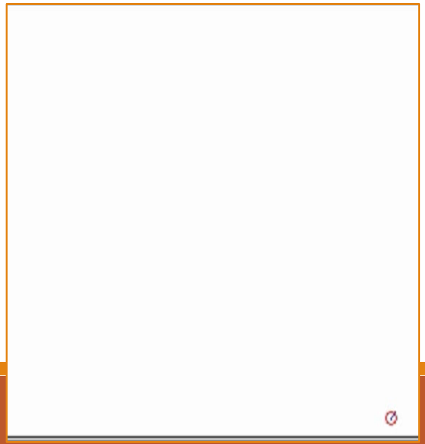
   VIEW SIMILAR RESTAURANTS

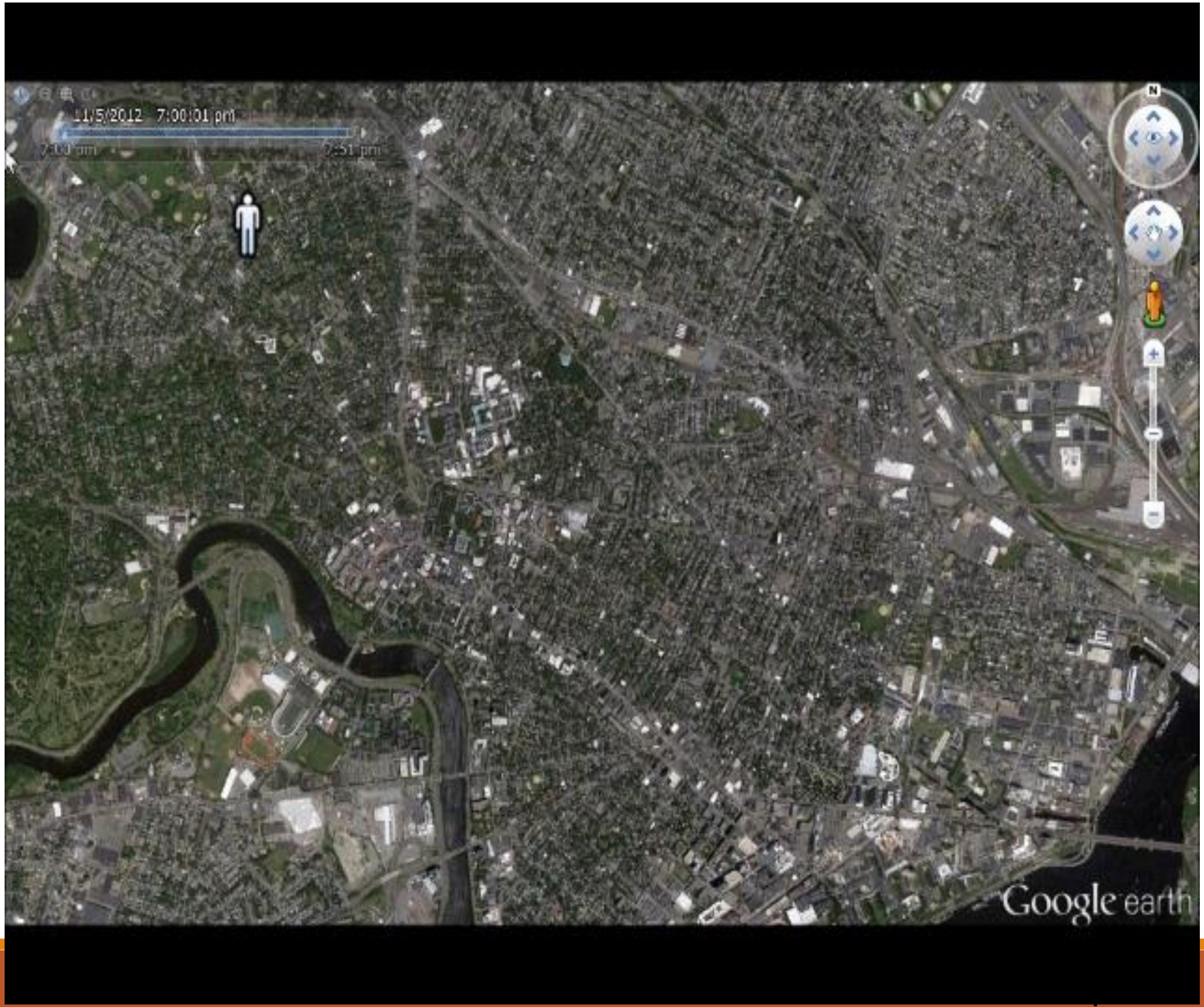


longitude
time
latitude

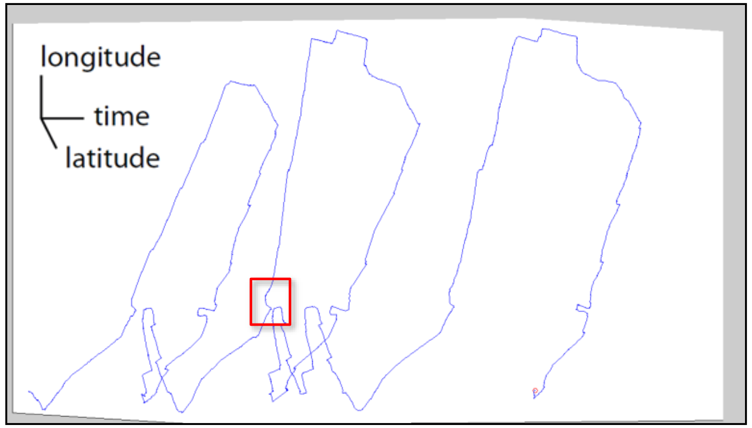


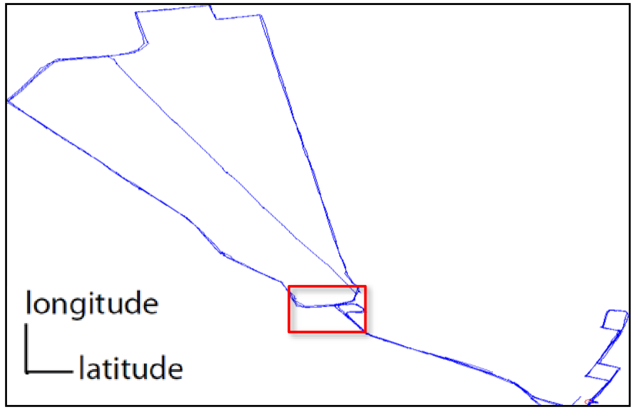
longitude
latitude

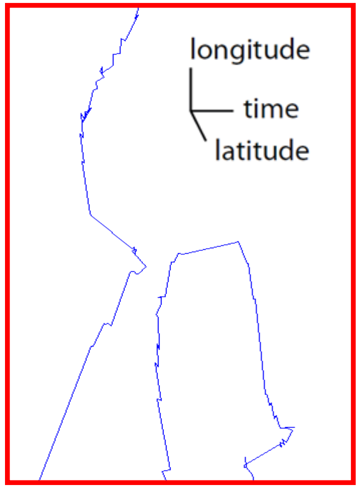




Big Data — Big Noise

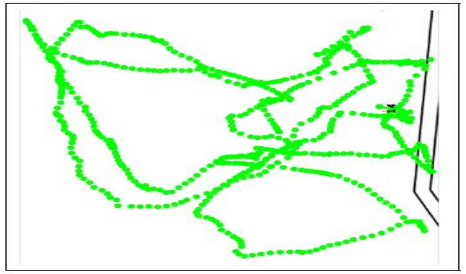


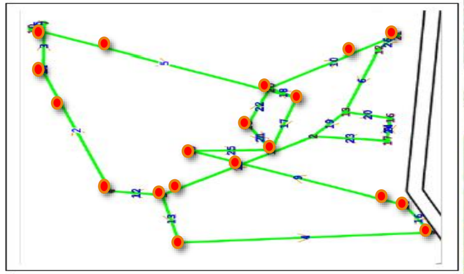




GPS Compression

[Feldman, Wu, Julian, Sung & Rus.]





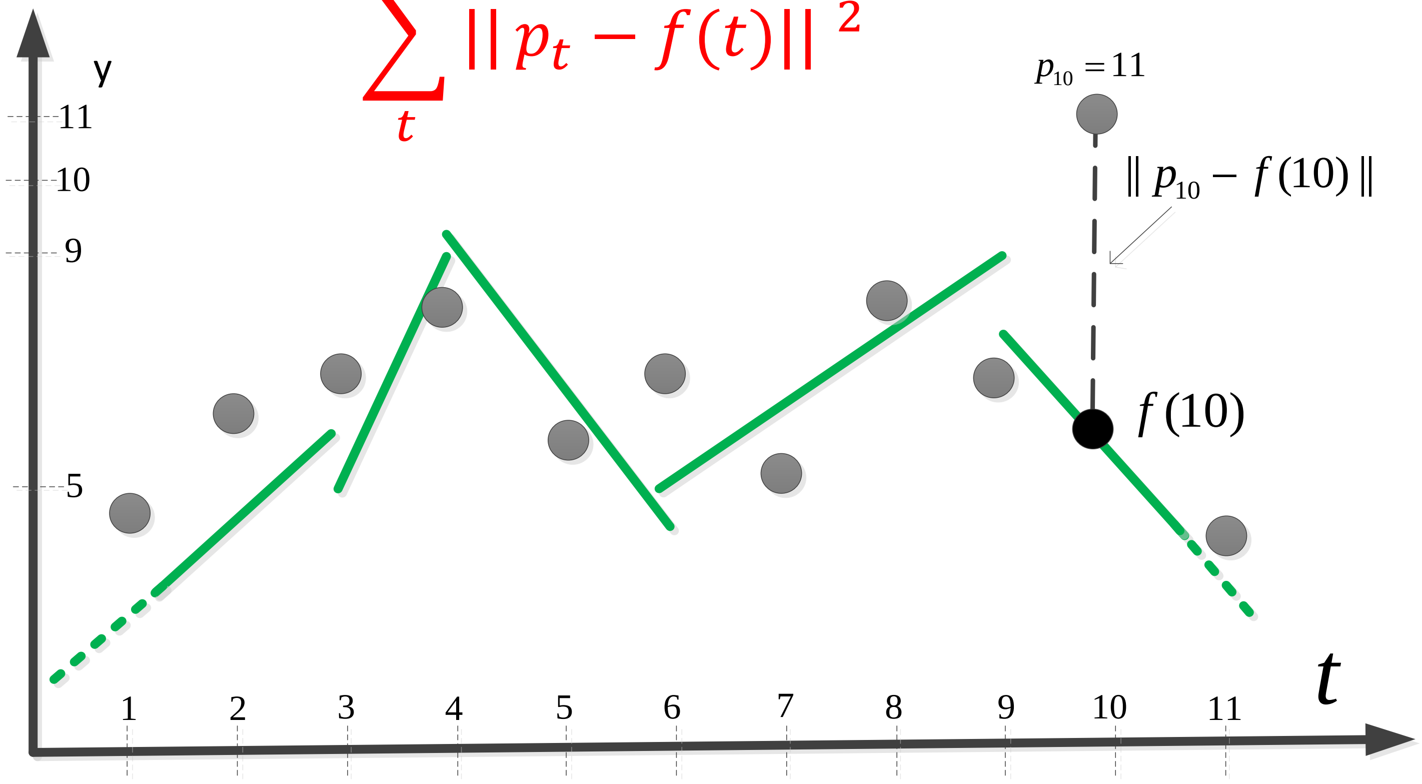
- Quadrobot collects data using attached smartphone
- Terabytes

  = 4 hours of image snapshots from Quadrobot
- Challenge: Real-time compression

k-Segment mean

The k-segment f^* that minimizes the fitting cost from points to a *d*-dimensional signal

$$\sum_t ||p_t - f(t)||^2$$

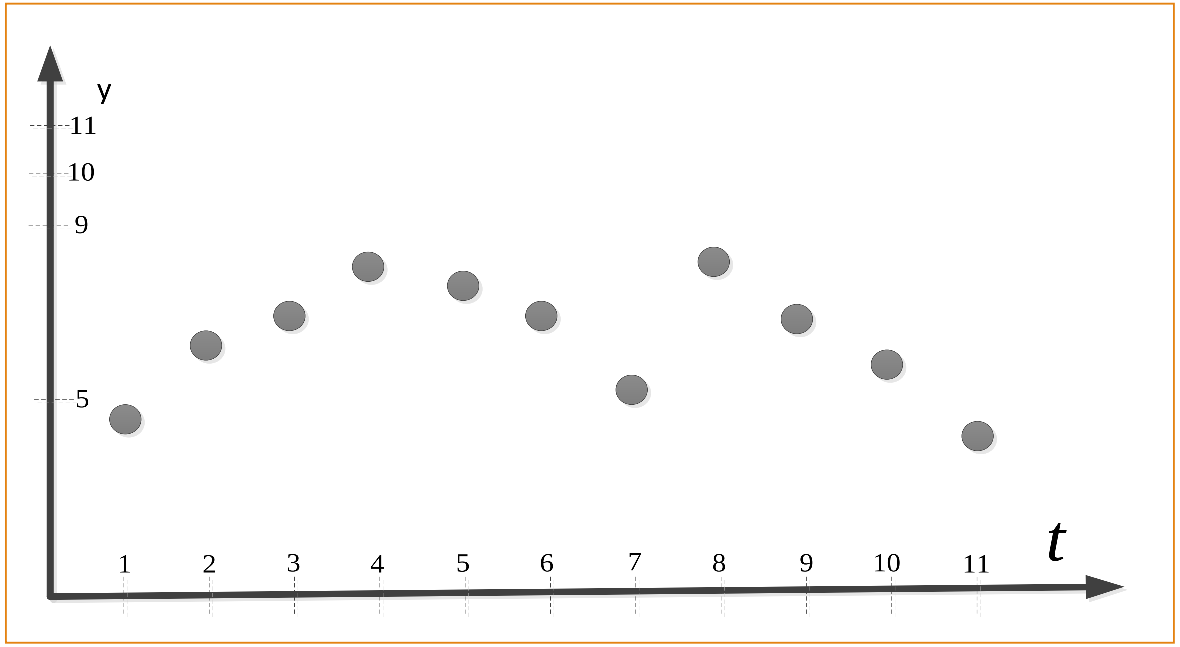


$k = 5$

$d = 1$

k – Segment Queries

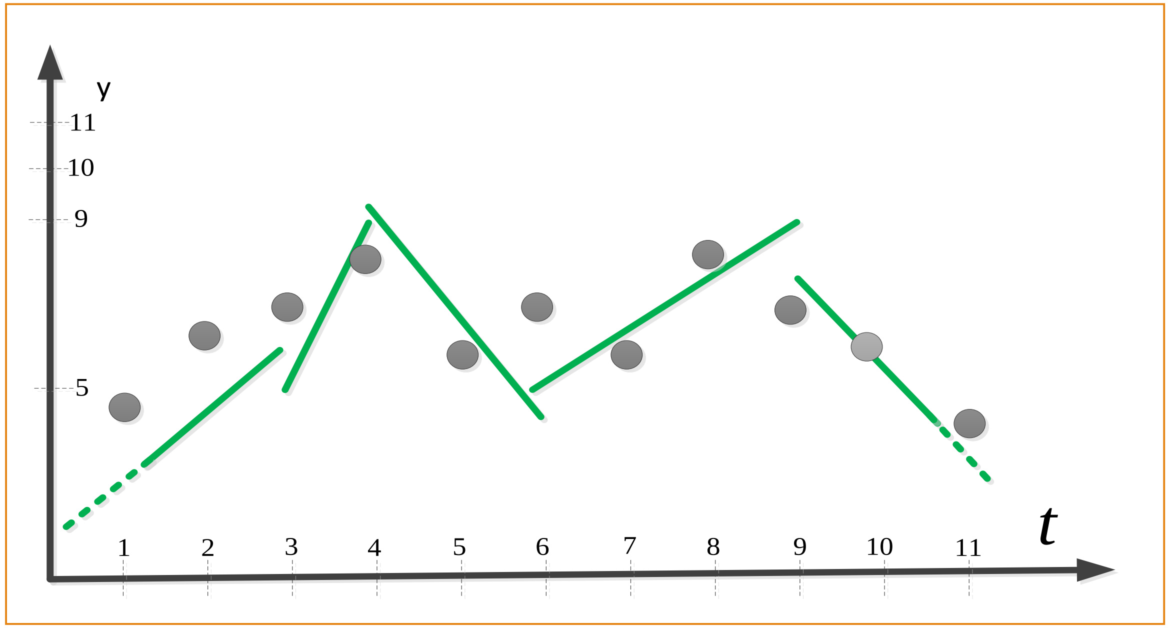
Input: *d*-dimensional signal *P* over time



k – Segment Queries

Input: *d*-dimensional signal *P* over time

Query: *k* segments over time



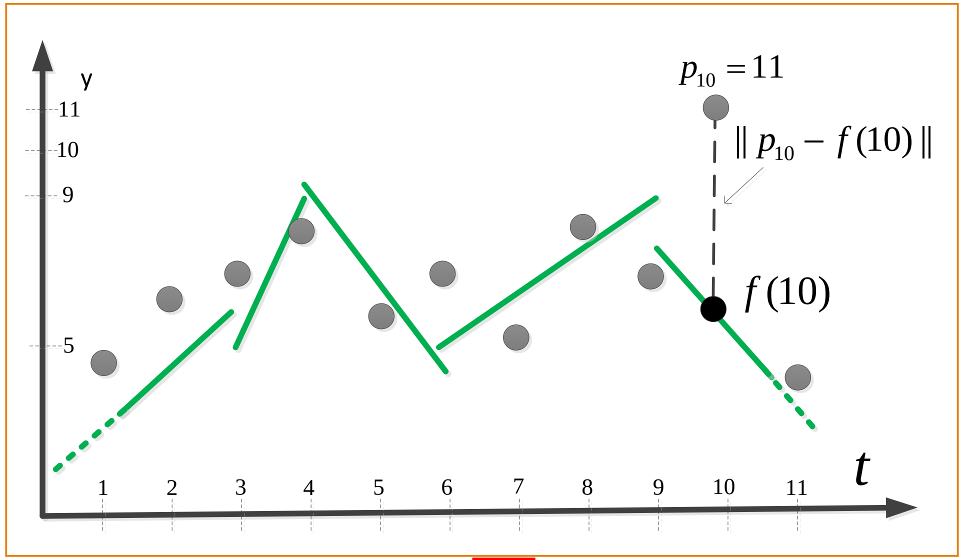
k-Piecewise linear function *f* over *t*

k – Segment Queries

Input: *d*-dimensional signal *P* over time

Query: *k* segments over time

Output: Sum of squared distances from *P*



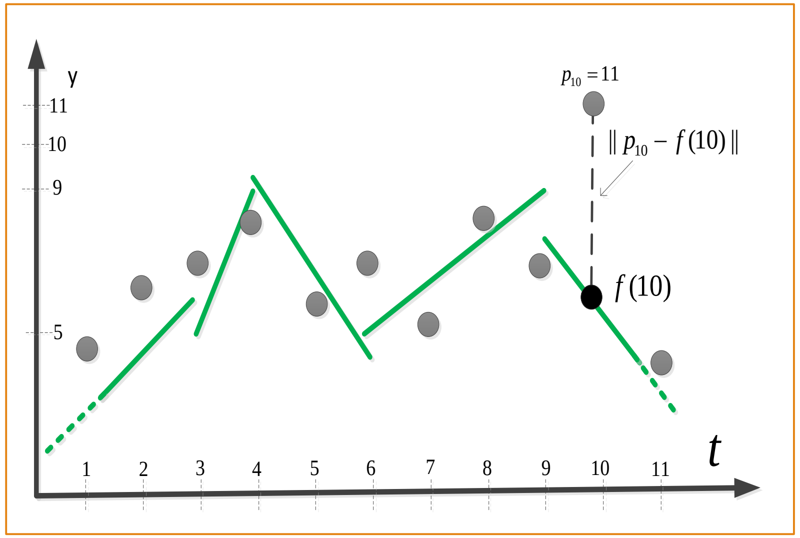
$$\text{cost}(P, f) := \sum_{t} \|f(t) - p_t\|^2$$

Coreset

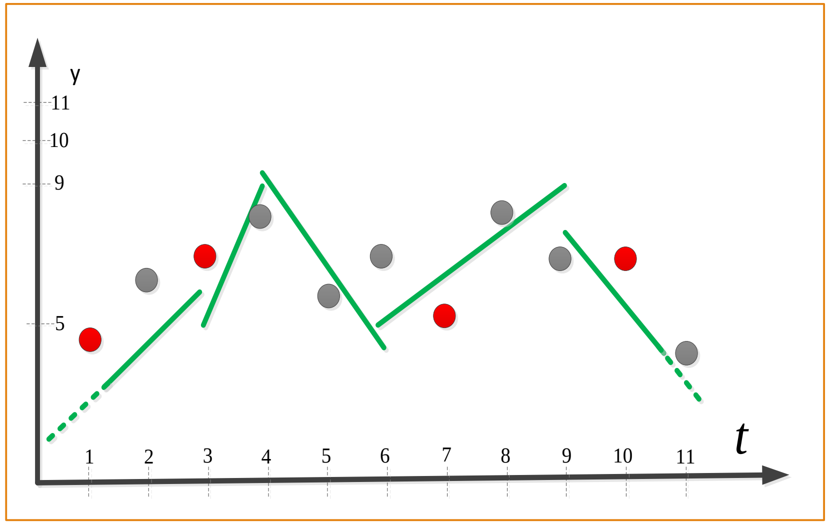
A weighted set C such that
for every k-segment f :

Different cost function

$$\text{cost}(P, f) \sim \text{cost}_w(C, f)$$



~

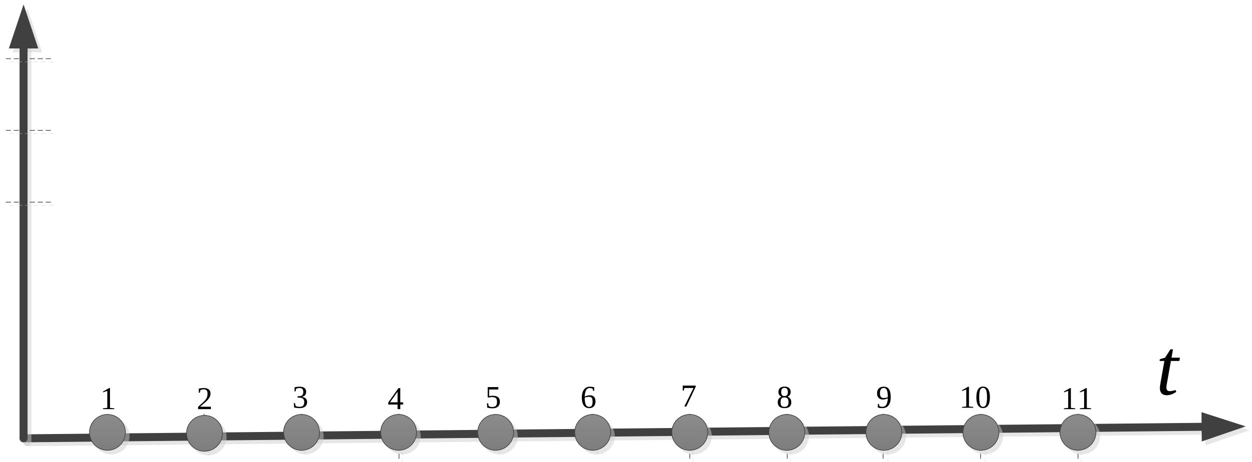


$$\sum_{t} \|f(t) - p_t\|^2 \quad (1 \pm \epsilon) \quad \sum_{p_t \in C} w(p_t) \cdot \|f(t) - p_t\|^2$$

Observation:

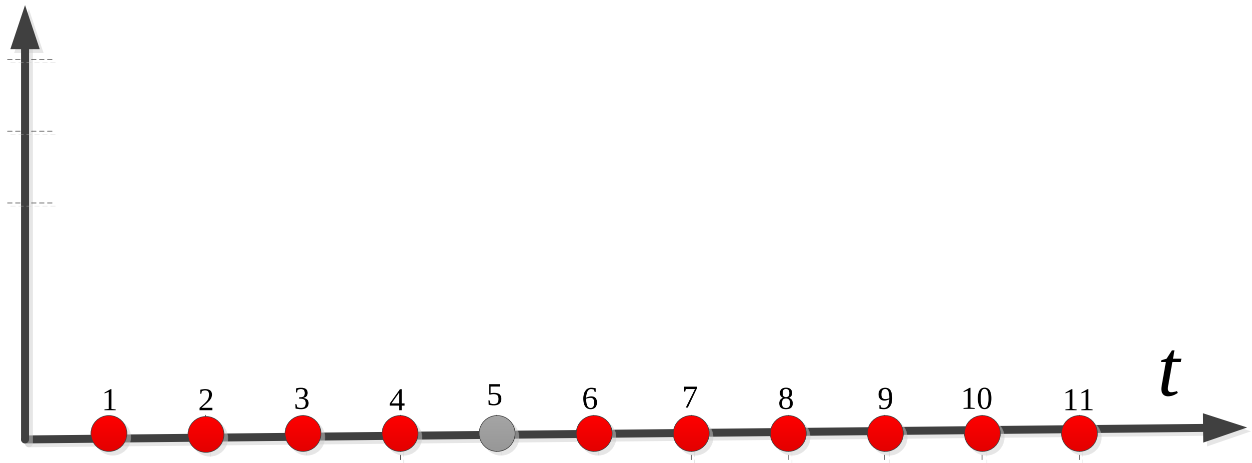
No small coreset $C \subset P$ exists for k-segment queries

Input P: *n* points on the *x*-axis



Input P: *n* points on the *x*-axis

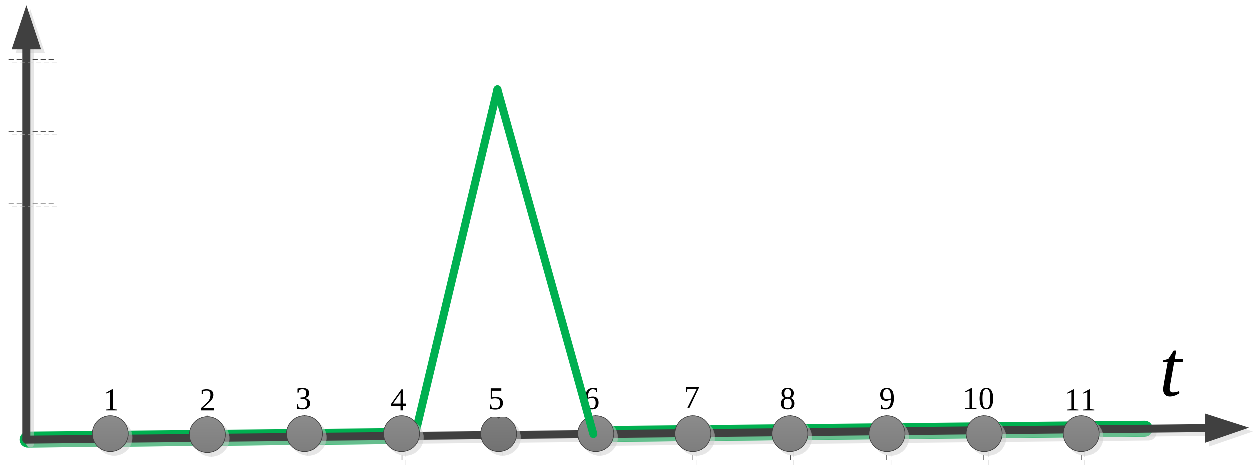
Coreset *C*: all points except one



Input P: n points on the x-axis

Coreset C: all points except one

Query f: covers all except this one



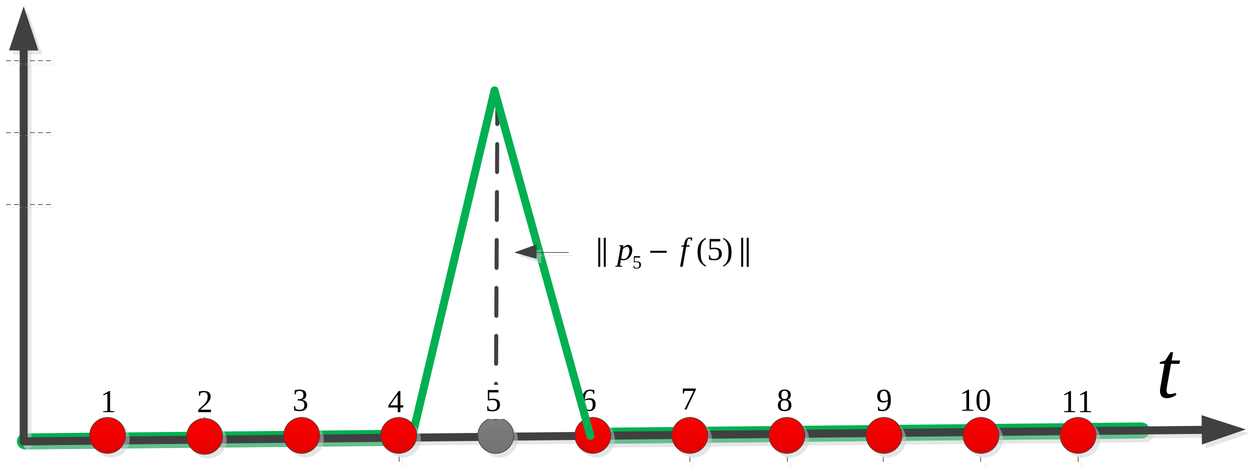
Input P: n points on the x-axis

Coreset C: all points except one

Query f: covers all except this one

$\text{Cost}(P, f) > 0$

$\text{Cost}(C, f) = 0$

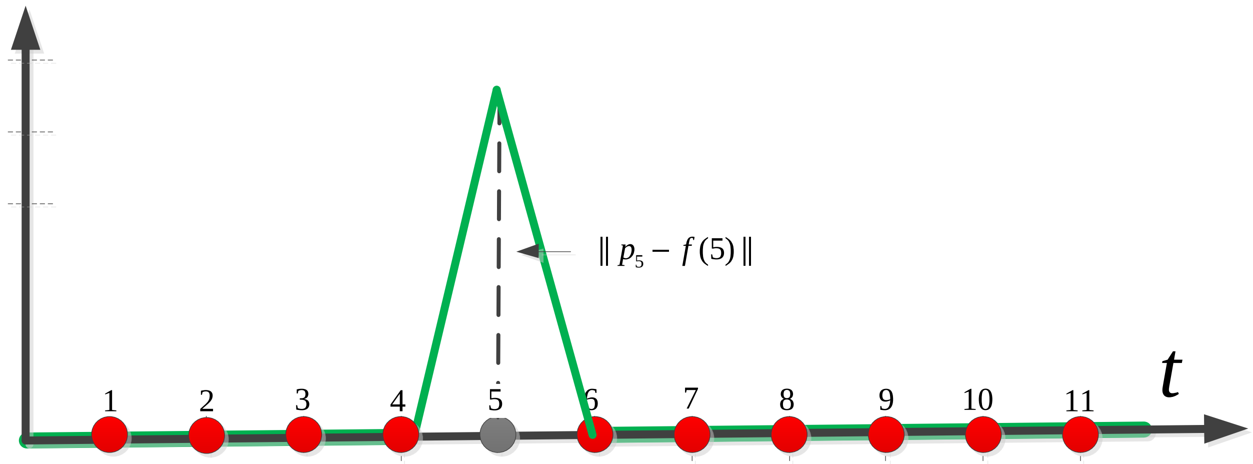


Input P: n points on the x-axis

Coreset C: all points except one

Query f: covers all except this one

$$\frac{\text{Cost}(P, f) > 0}{\text{Cost}(C, f) = 0}$$ → Unbounded factor approximation



For every point *p*:

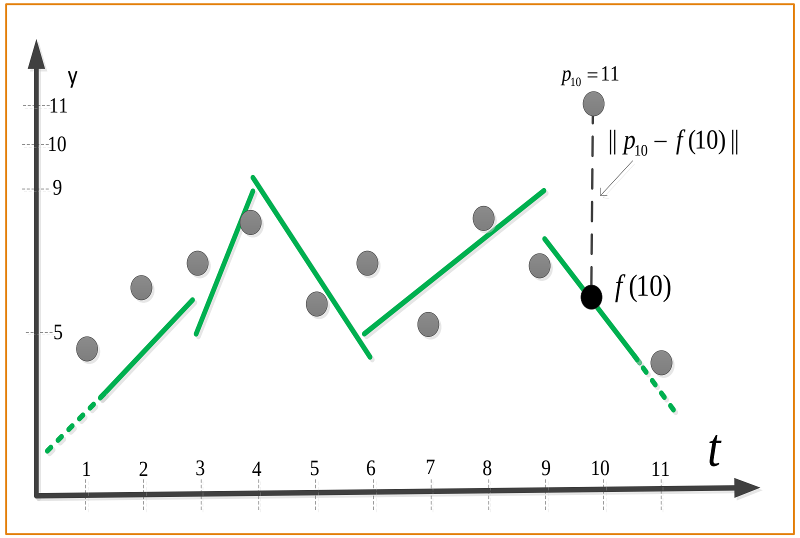
Sensitivity(*p*) = $\max_{q \in Q} \frac{dist(p,q)}{\sum_{p'} dist(p',q)} = 1$

Total sensitivities: *n*

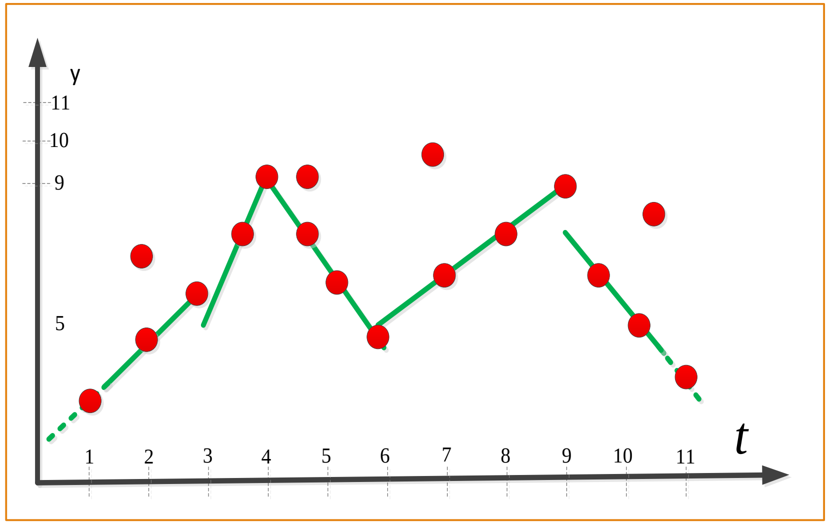
Definition: Coreset

A weighted set $C \not\subset P$ such that
for every k-segment f:

$$\text{cost}(P, f) \sim \text{cost}_w(C, f)$$



~

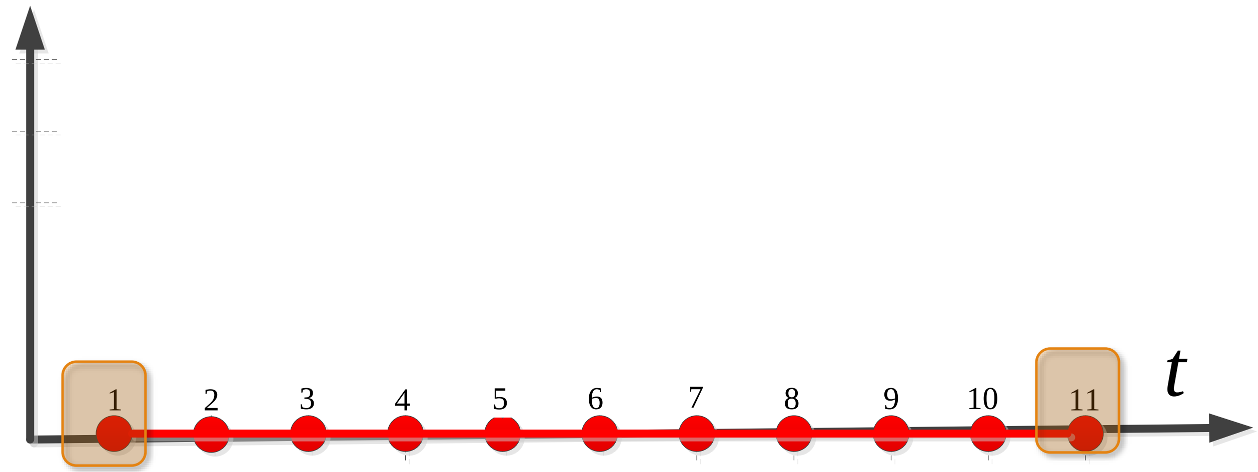


$$\sum_t \|f(t) - pt\|$$

$$\sum_{p_t \in C} w(p_t) \cdot \|f(t) - pt\|$$

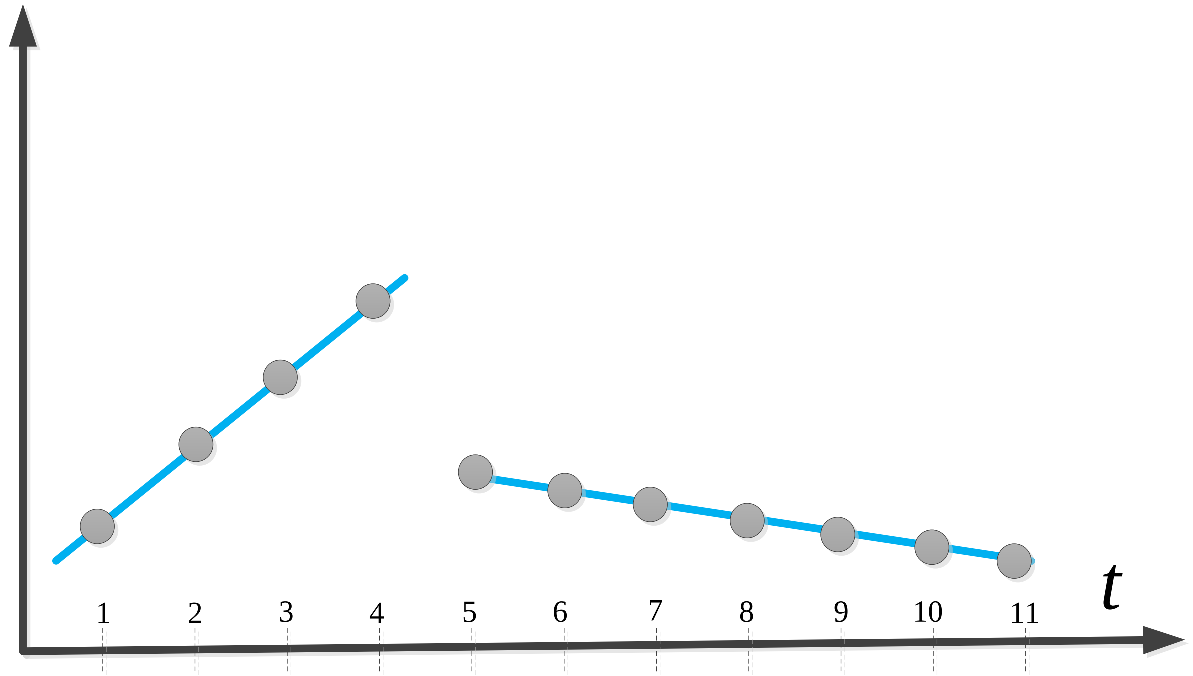
Observation:

Points on a segment can be stored by the two indexes of their end-points



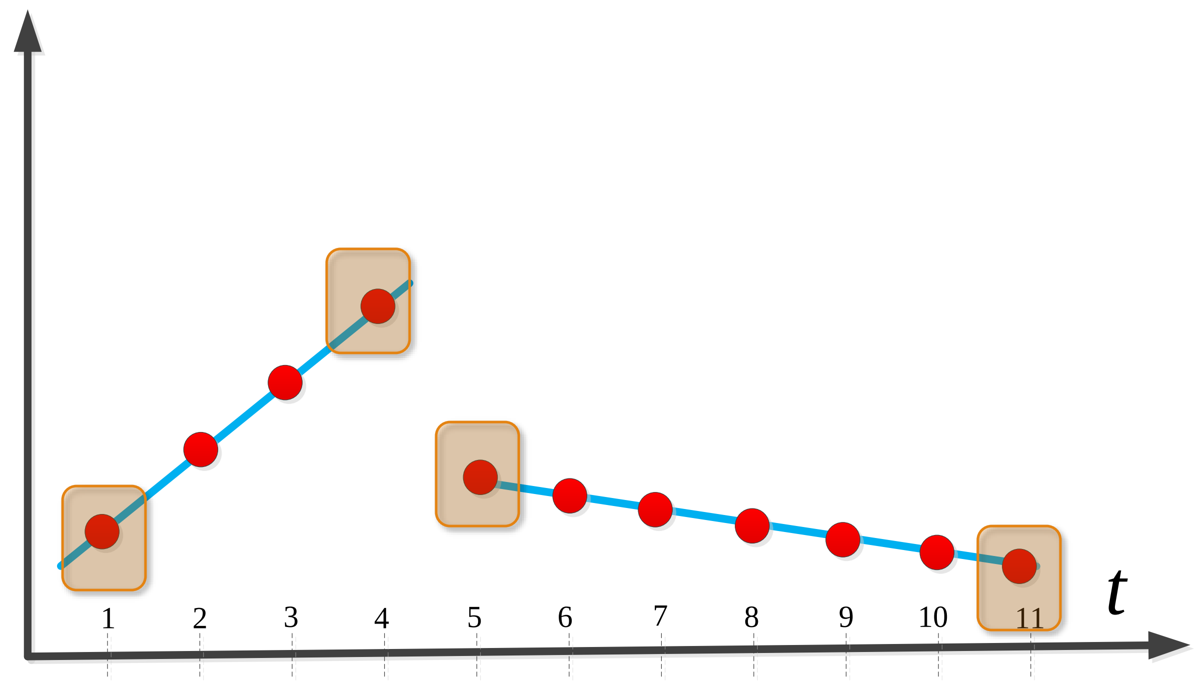
Observation:

Points on a segment can be stored by the two indexes of their end-points and the slope of the segment



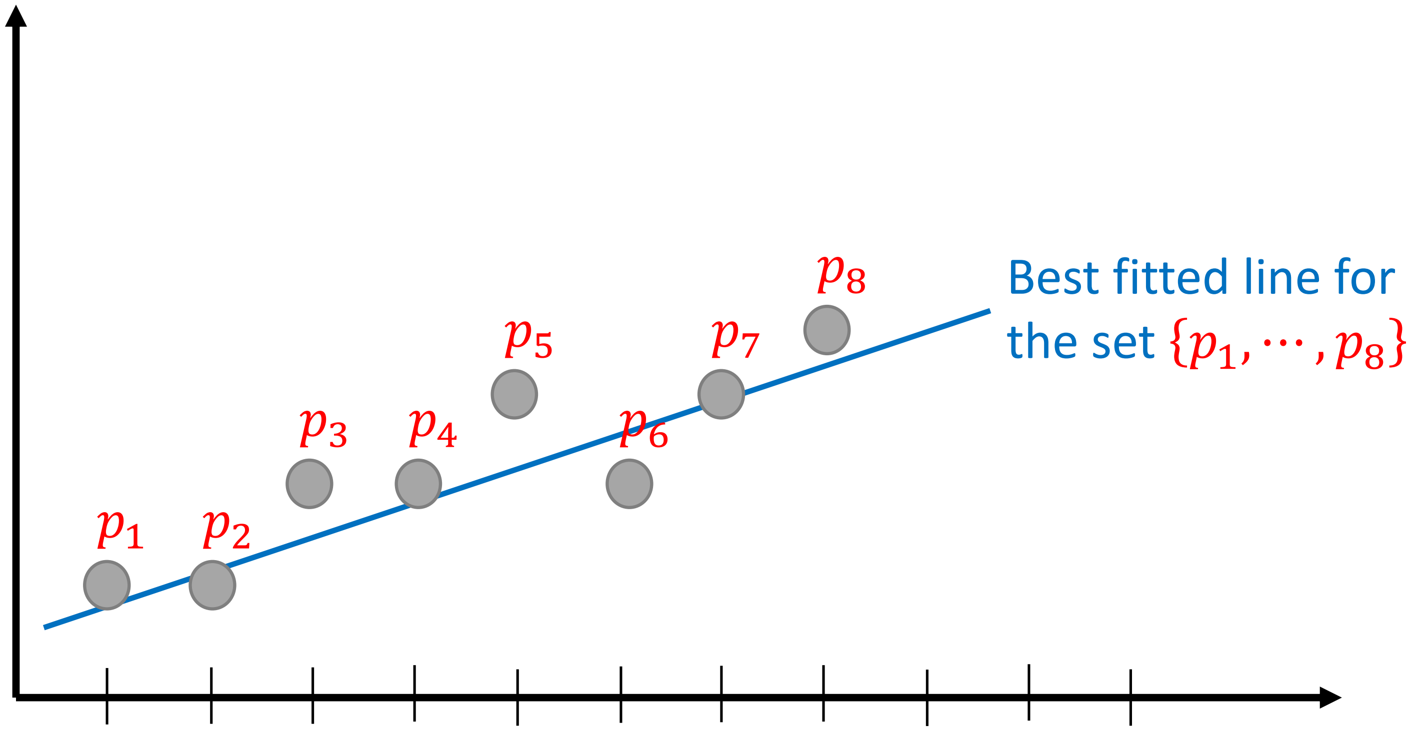
Observation:

Points on a segment can be stored by the two indexes of their end-points and the slope of the segment



Observation 2:

We can solve optimally for $k = 1$ (1 segment) by solving a simple linear regression problem on the set of points.



Coreset for k-segment Mean

Definition: $(\boldsymbol{k}, \boldsymbol{\varepsilon})$**-coreset**

Let $P \subseteq \mathbb{R}^{d+1}$ be a signal, $k \geq 1$ and $\varepsilon > 0$.

Let D be a set of items, and $cost'(D,\cdot)$ be a function that maps every k-segment f to a non-negative number. Then $(D, cost')$ is a (k, ϵ)-coreset for P if for every k-segment f we have

$$(1-\varepsilon)cost(P, f) \leq cost'(D, f) \leq (1+\varepsilon)cost(P, f).$$

Coreset for k-segment Mean

Definition: ***cost'(D, f)***

See main algorithm to understand D better

Let $D = \{(C_i, g_i, b_i, e_i)\}_{i=1}^{m}$ where for every $i \in [m]$ we have $C_i \subseteq \mathbb{R}^{d+1}$, $g_i : \mathbb{R} \to \mathbb{R}^d$ and $b_i, e_i \in \mathbb{R}$ such that $b_i \leq e_i$. For a k-segment $f : \mathbb{R} \to \mathbb{R}^d$ and $i \in [m]$ we say that C_i is served by one segment of f if $\{f(t) | b_i \leq t \leq e_i\}$ is a linear segment. We denote by $Good(D, f) \subseteq [m]$ the union of indices i such that C_i is served by one segment of f. We also define $L_i = \{g_i(t) | b_i \leq t \leq e_i\}$, the projection of C_i on g_i. We define $cost'(D, f)$ as

$$\sum_{i \in Good(D,f)} cost(C_i, f) + \sum_{i \in [m] \setminus Good(D,f)} cost(L_i, f) .$$

Our Main Compression Theorem

[ACM GIS'12, with C. Sung, and D. Rus]

Theorem:

For every discrete signal P of n points in R^d, there is a (k, ε)-coreset for P of space $O\left(\frac{k\log n}{\epsilon^2}\right)$ that can be computed in the big data model, and can be computed in $O\left(\frac{dn}{\epsilon^4}\right)$ time.

See Algorithm $BALANCEDPARTITION$.

K – segments Bicriteria

Algorithm 1: BICRITERIA(P, k)

Input: A set $P \subseteq \mathbb{R}^{d+1}$ and an integer $k \geq 1$
Output: An $(O(\log n), O(\log n))$-approximation to the k-segment mean of P.

if $n \leq 2k + 1$ **then**
  $f :=$ a 1-segment mean of P;
  return f;
Set $t_1 \leq \cdots \leq t_n$ and $p_1, \cdots, p_n \in \mathbb{R}^d$ such that $P = \{(t_1, p_1), \cdots, (t_n, p_n)\}$
$m \leftarrow \{t \in \mathbb{R} \mid (t, p) \in P\}$
Partition P into $4k$ sets $P_1, \cdots, P_{2k} \subseteq P$ such that for every $i \in [2k - 1]$:
  (i) $|\{t \mid (t,p) \in P_i\}| = \left\lfloor \frac{m}{4k} \right\rfloor$, and (ii) if $(t, p) \in P_i$ and $(t', p') \in P_{i+1}$ then $t < t'$.
for $i := 1$ *to* $4k$ **do**
  Compute a 2-approximation g_i to the 1-segment mean of P_i
$Q :=$ the union of $k + 1$ signals P_i with the smallest value $\text{cost}(P_i, g_i)$ among $i \in [2k]$.
$h :=$ BICRITERIA$(P \setminus Q, k)$; Repartition the segments that did not have a good approximation
Set

$$f(t) := \begin{cases} g_i(t) & \exists (t,p) \in P_i \text{ such that } P_i \subseteq Q \\ h(t) & \text{otherwise} \end{cases}.$$

return f;

<u>Input:</u>
A signal $P \subseteq \mathbb{R}^d$, and an integer k.

<u>Output:</u>
An (α, β)-approximation f' to the k-segment mean of P.

$$\alpha, \beta = O(\log n)$$

K – segments Algorithm

$\sigma = cost(P, f')$
where f' is the output of the Bicriteria algorithm

Algorithm 2: BALANCEDPARTITION(P, ε, σ)

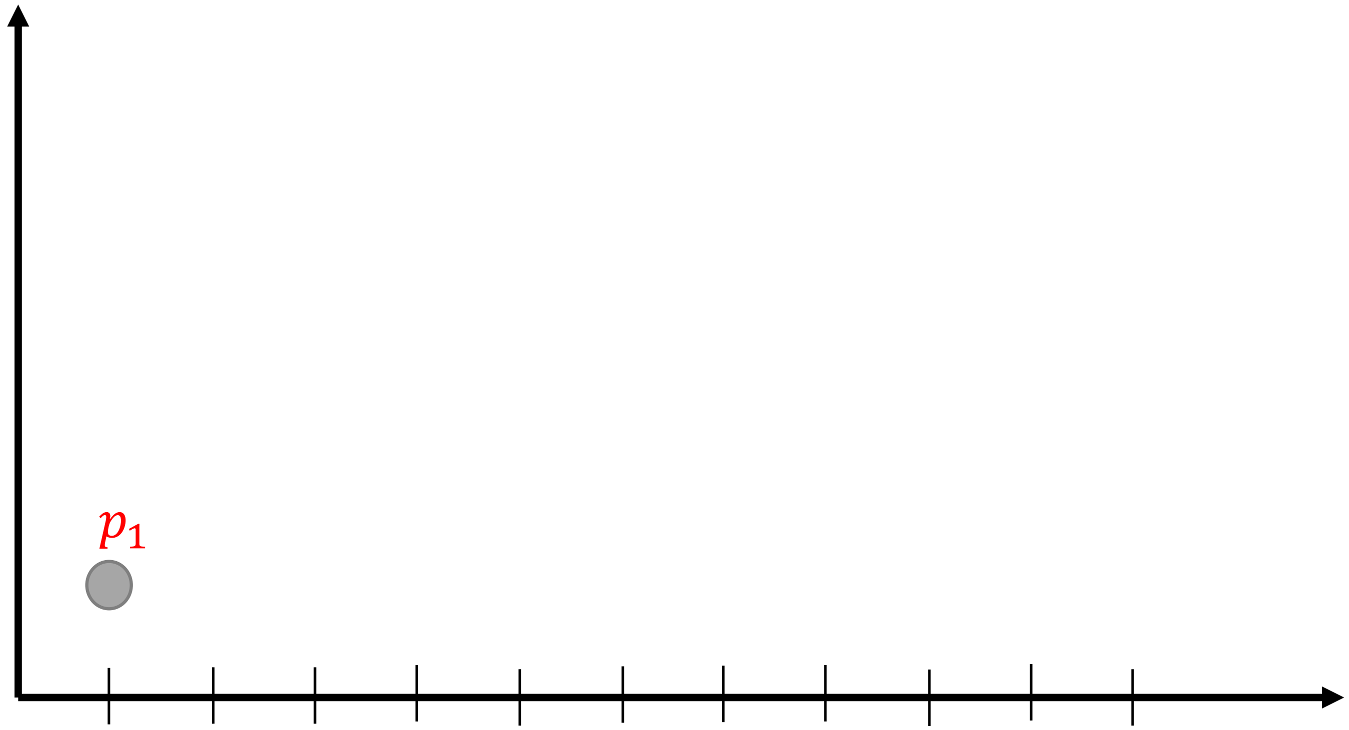
```
Input: A set P = {(1, p_1), ··· , (n, p_n)} in R^{d+1}
an error parameters ε ∈ (0, 1/10) and σ > 0.
Output: A set D that satisfies Theorem 4.
Q := ∅; D = ∅ ; p_{n+1} := an arbitrary point in R^d ;
for i := 1 to n + 1 do
    Q := Q ∪ {(i, p_i)}; Add new point to tuple
    f* := a linear approximation of Q;   λ := cost(Q, f*)
    if λ > σ or i = n + 1 then
        T := Q \ {(i, p_i)} ; take all the new points into tuple
        C := a (1, ε/4)-coreset for T; Approximate points by a local
        representation
        g := a linear approximation of T, b := i − |T|, e := i − 1; save
        endpoints
        D := D ∪ {(C, g, b, e)} ; save a tuple
        Q := {(i, p_i)} ; proceed to new point
return D
```

We will show how to compute a $(1, \varepsilon)$-coreset later

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1)\}$



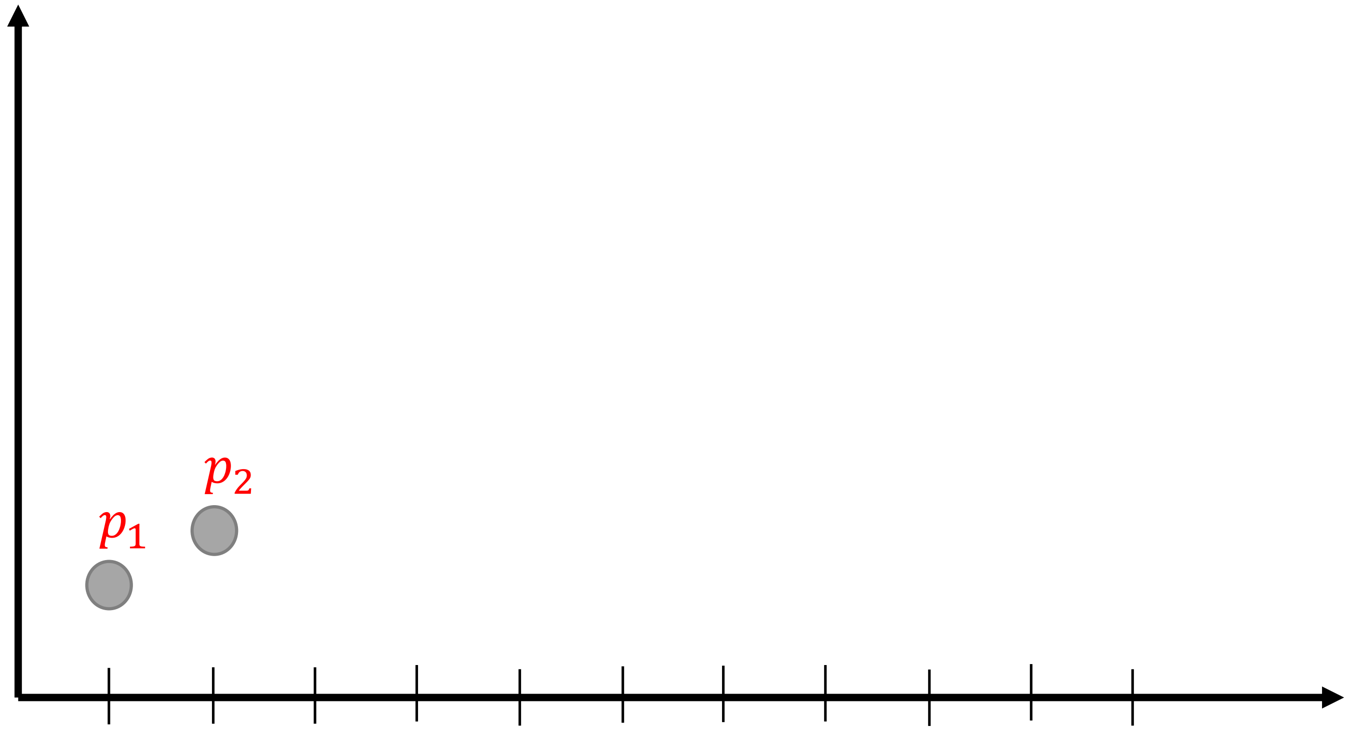
For $i := 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2)\}$



For $i \coloneqq 1 \to n$ do

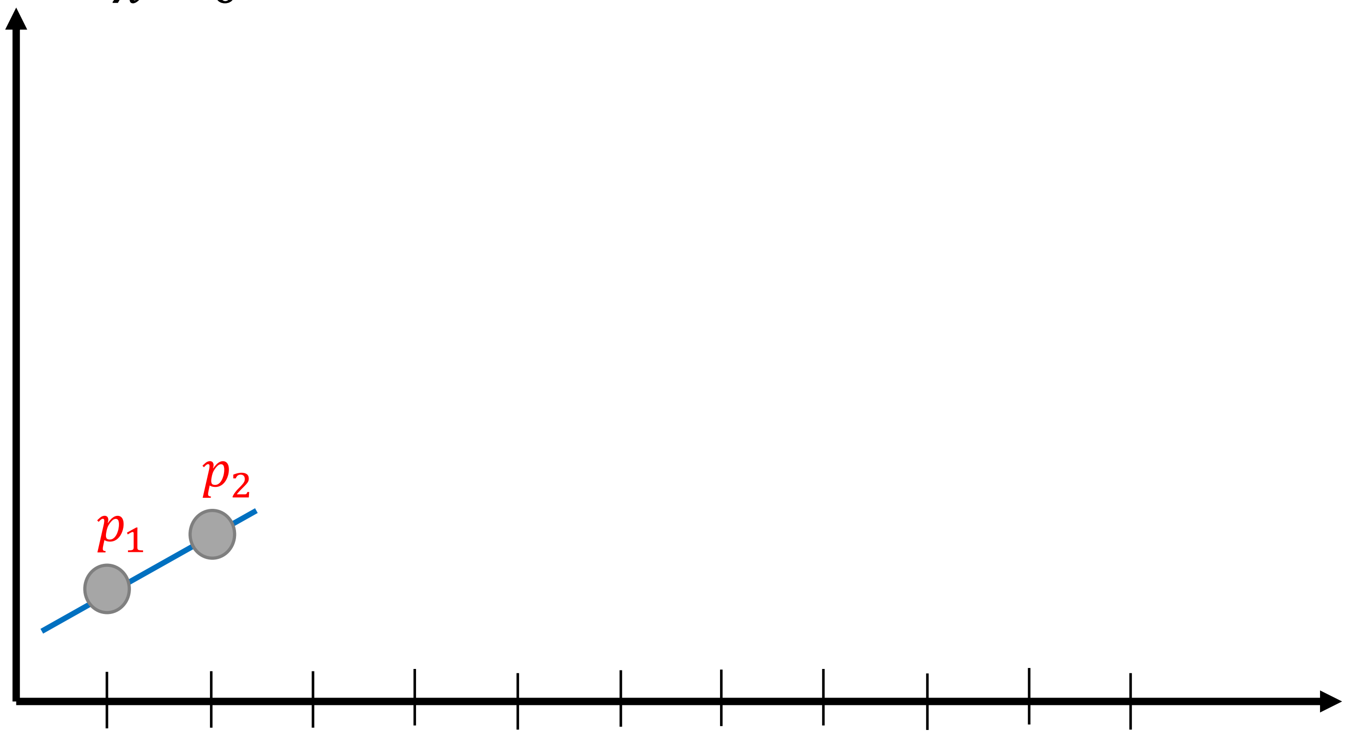
- $Q = Q \cup \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2)\}$

$\lambda = 0$



For $i \coloneqq 1 \rightarrow n$ do

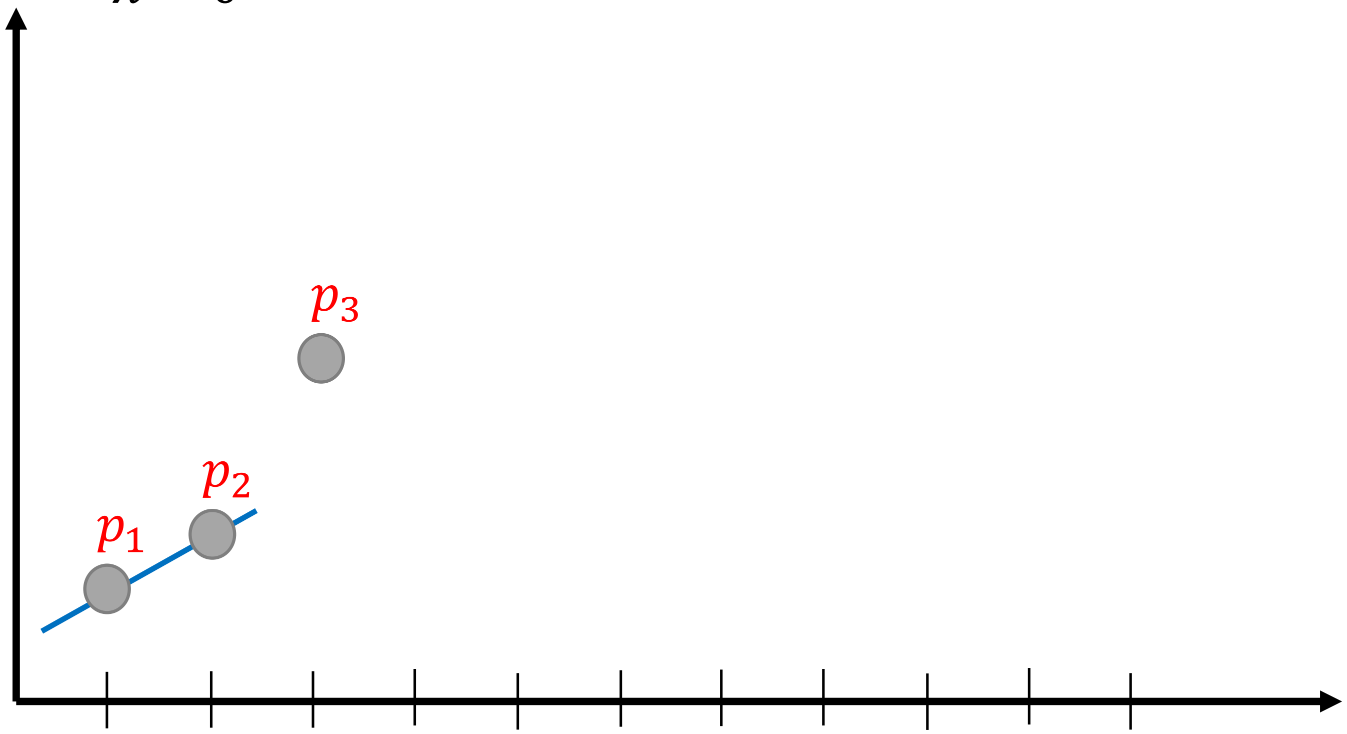
- $Q = Q \cup \{(i, p_i)\}$
- f^* = a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3)\}$

$\lambda = 0$



For $i \coloneqq 1 \rightarrow n$ do

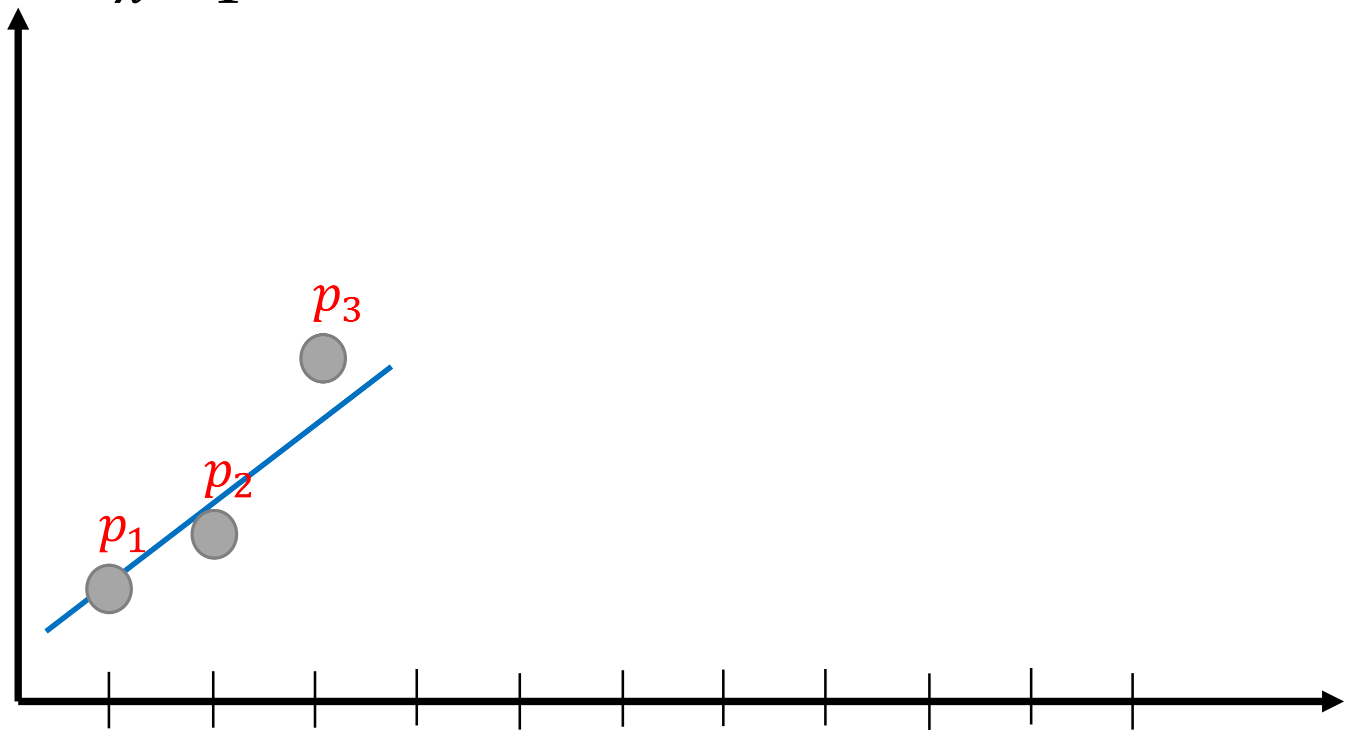
- $Q = Q \cup \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3)\}$

$\lambda = 1$



For $i := 1 \rightarrow n$ do

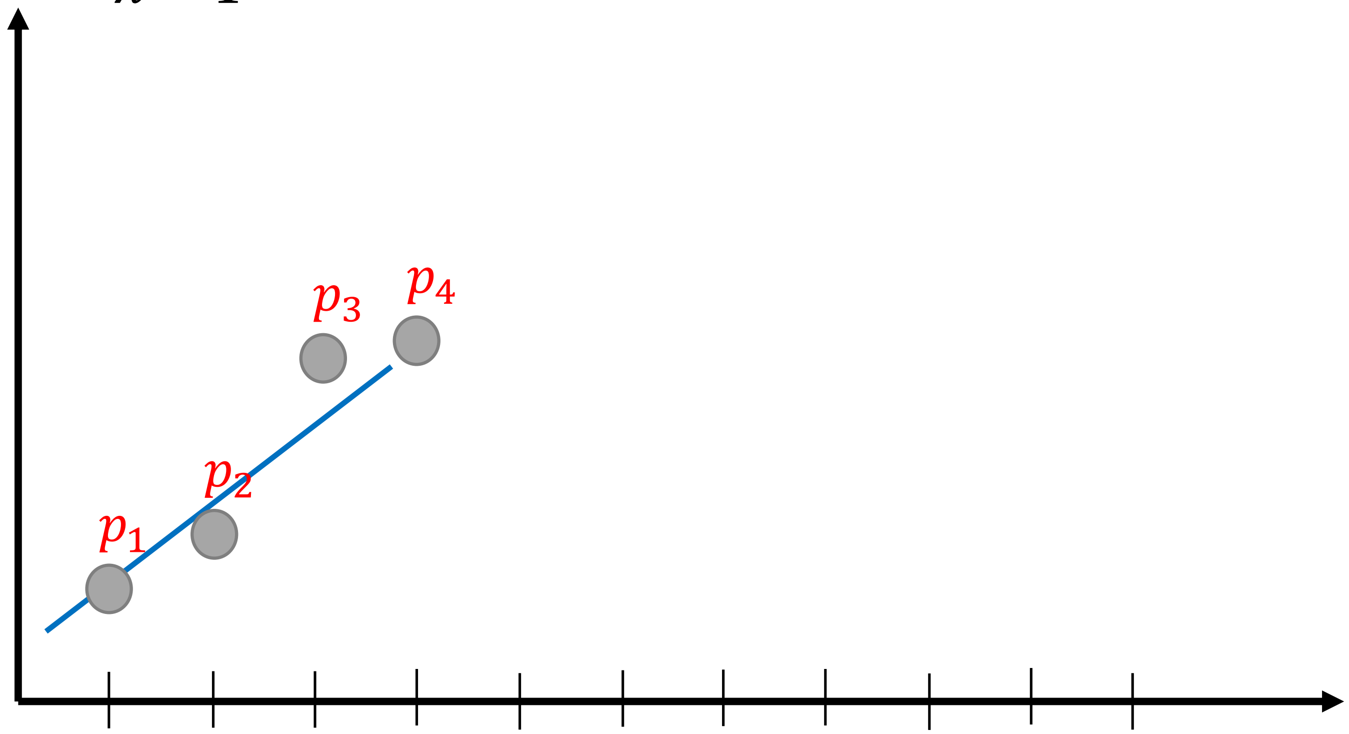
- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4)\}$

$\lambda = 1$



For $i := 1 \rightarrow n$ do

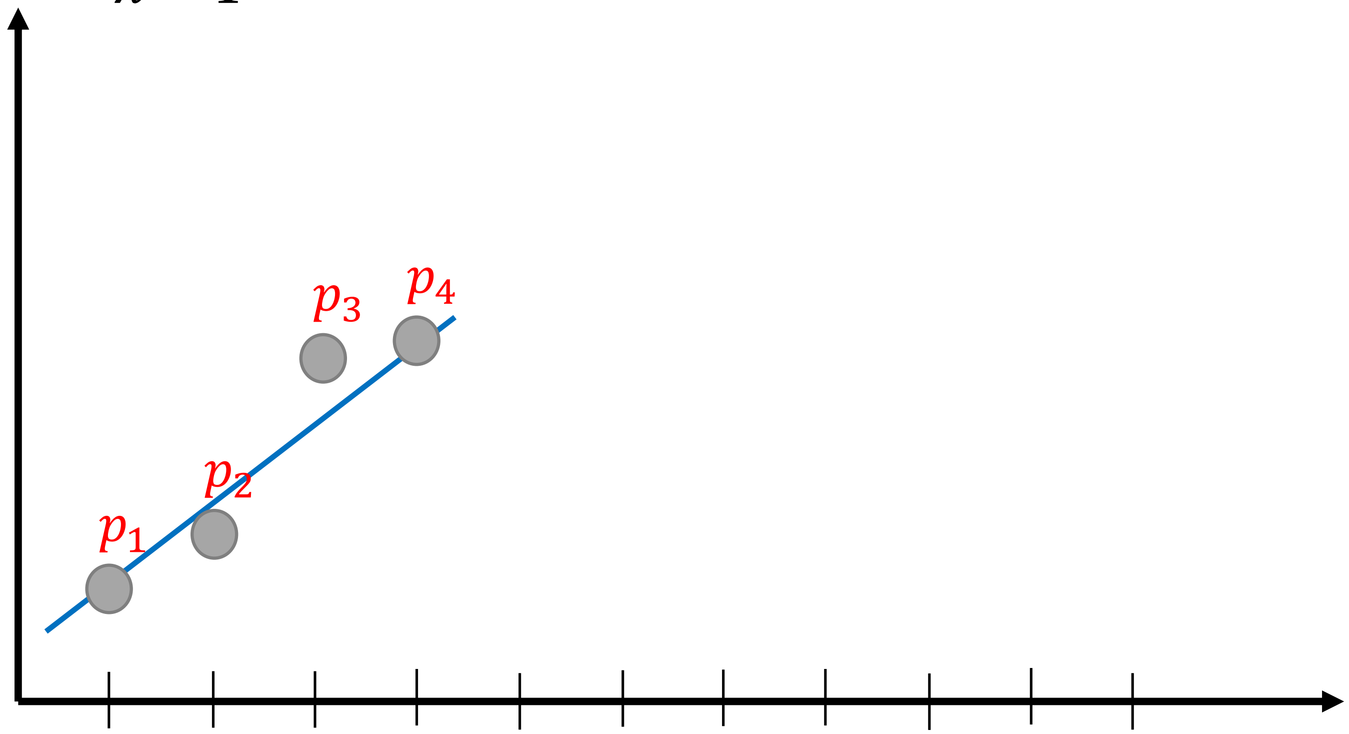
- $Q = Q \cup \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4)\}$

$\lambda = 1$



For $i := 1 \rightarrow n$ do

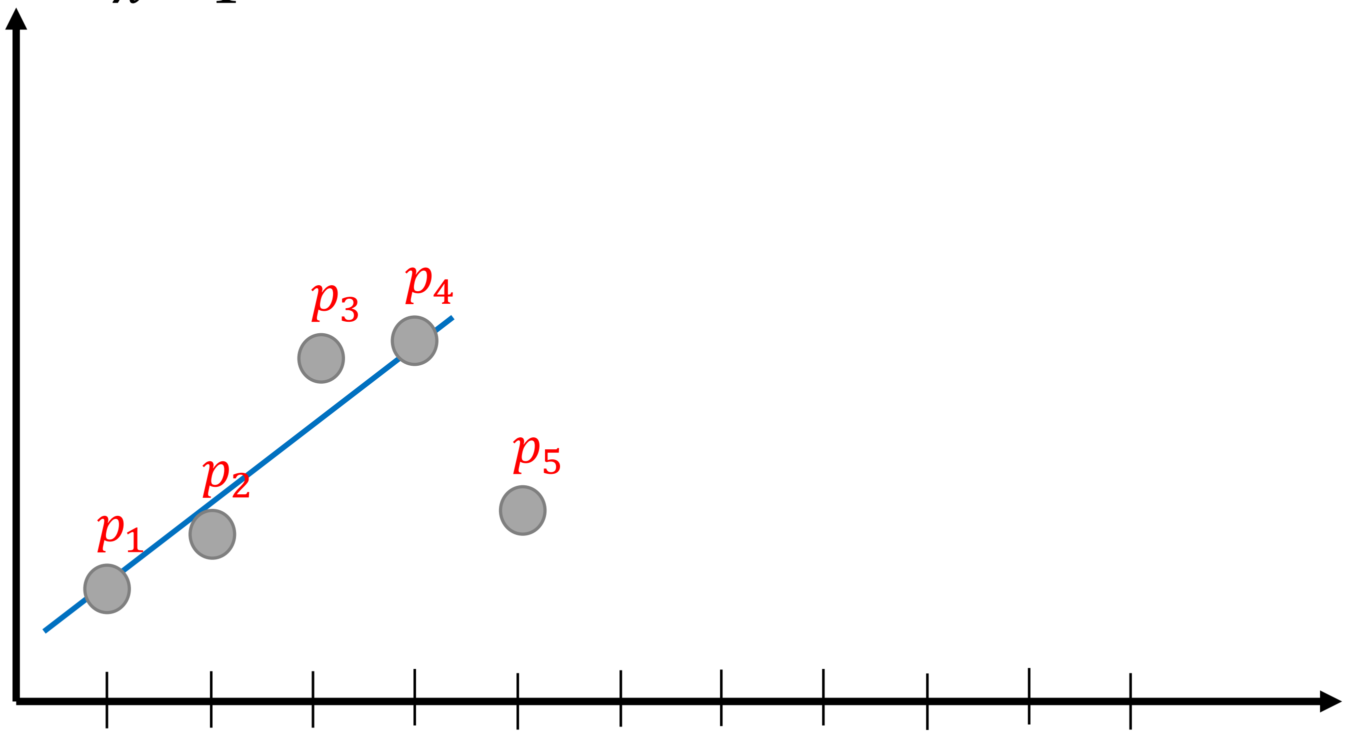
- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 1$



For $i := 1 \rightarrow n$ do

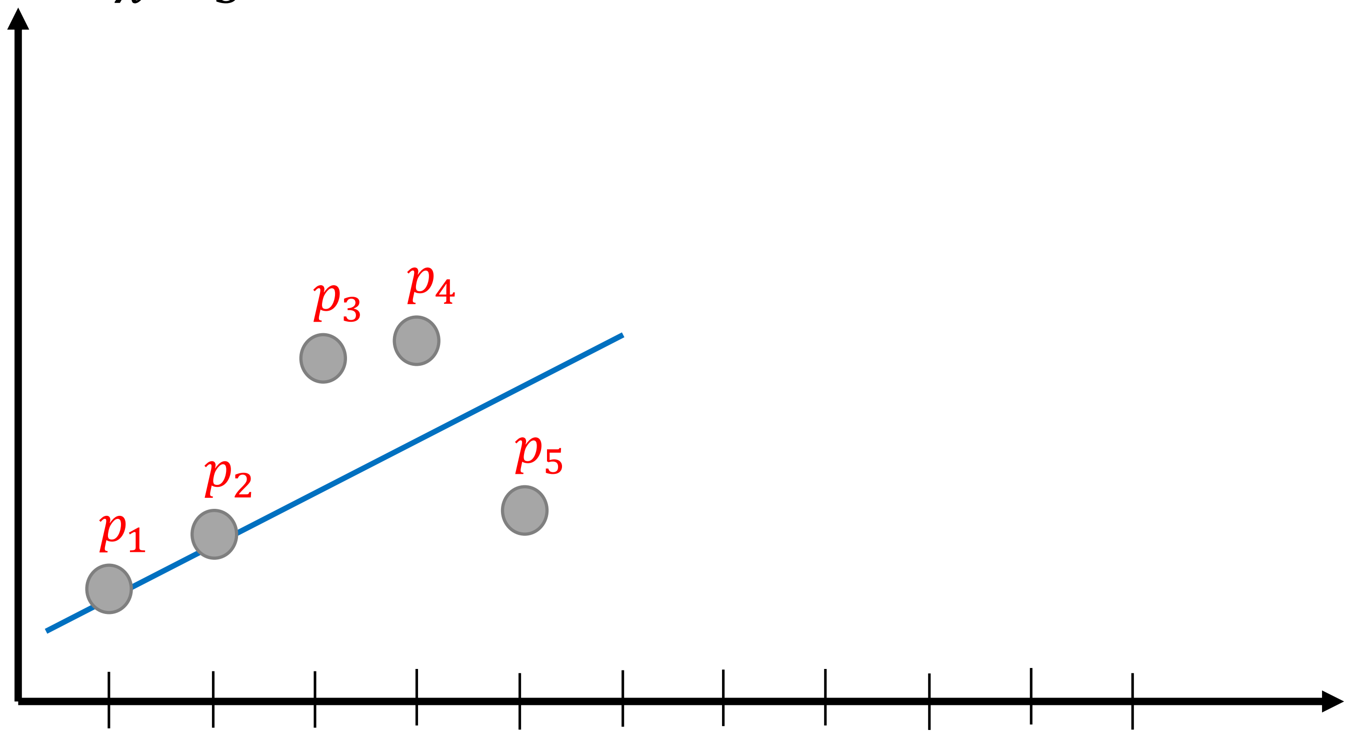
- $Q = Q \cup \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 5$



For $i \coloneqq 1 \rightarrow n$ do

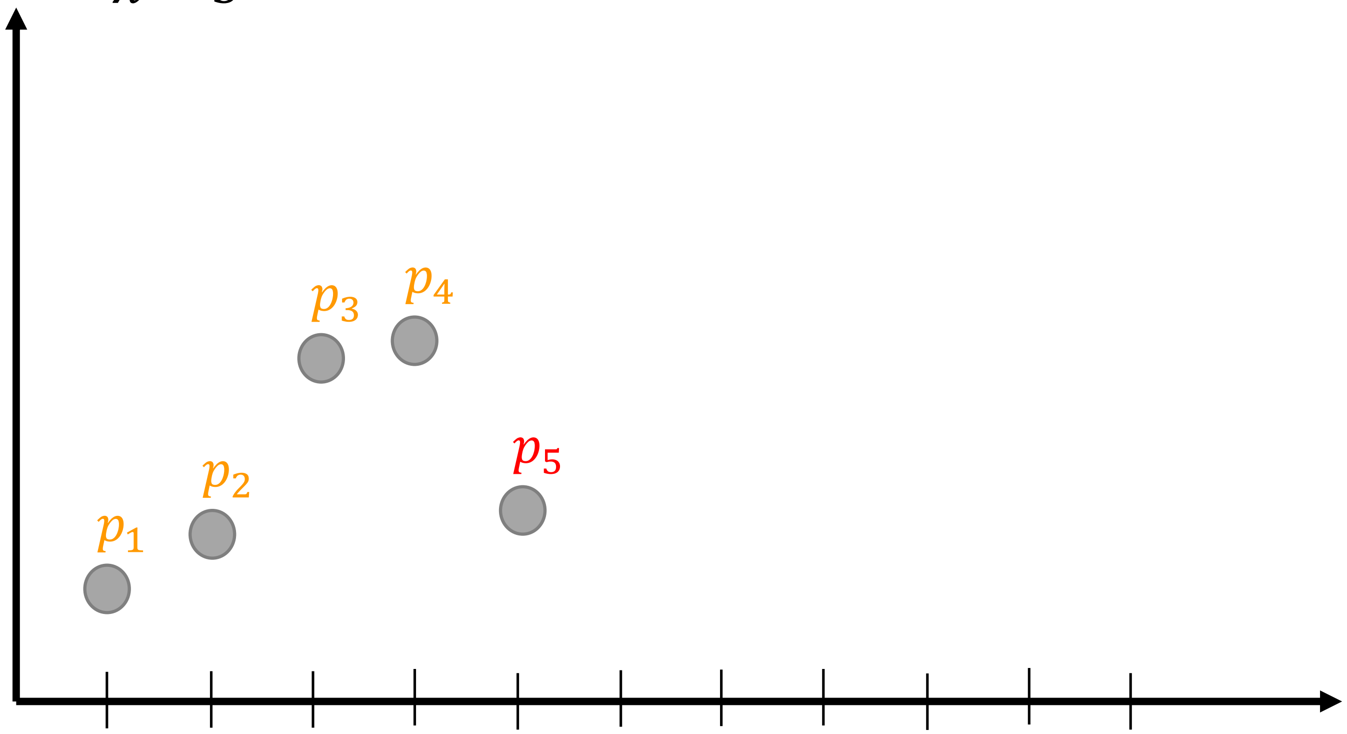
- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 5$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

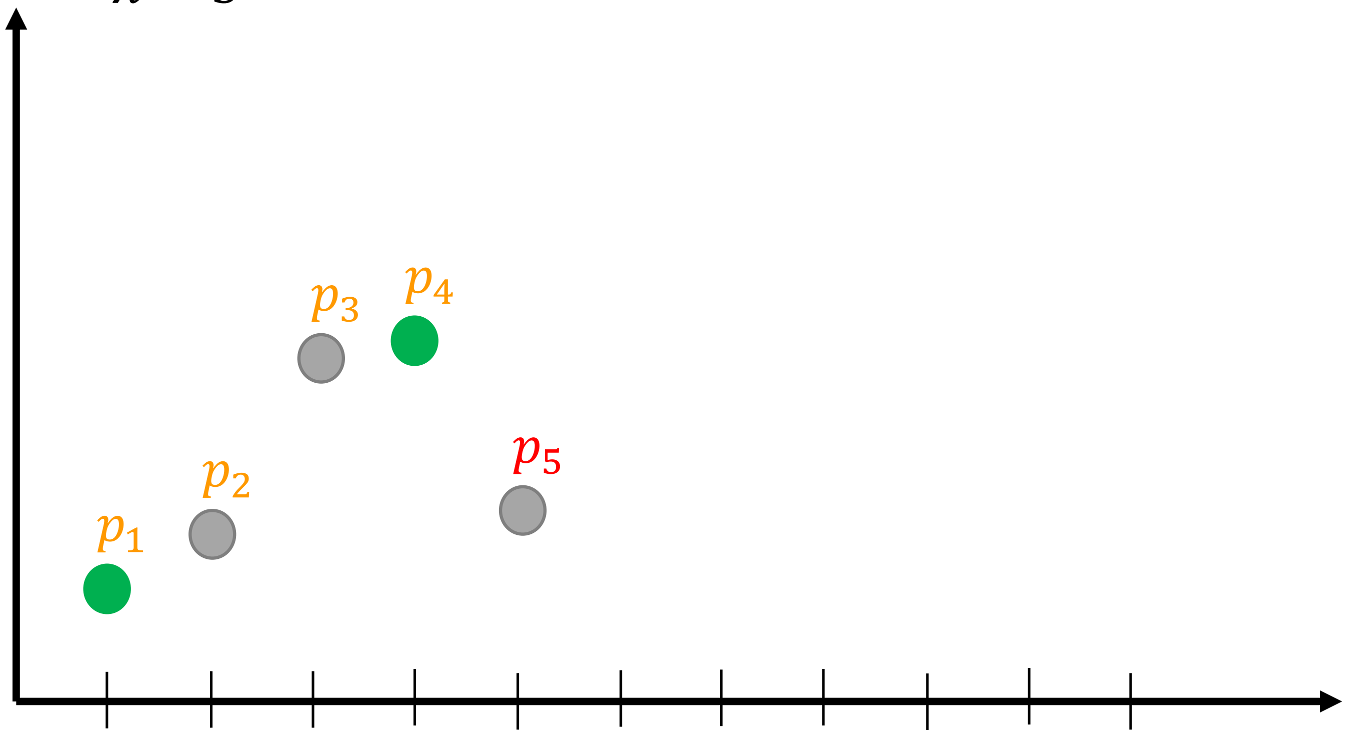
- $T = Q \setminus \{(i, p_i)\}$

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 5$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

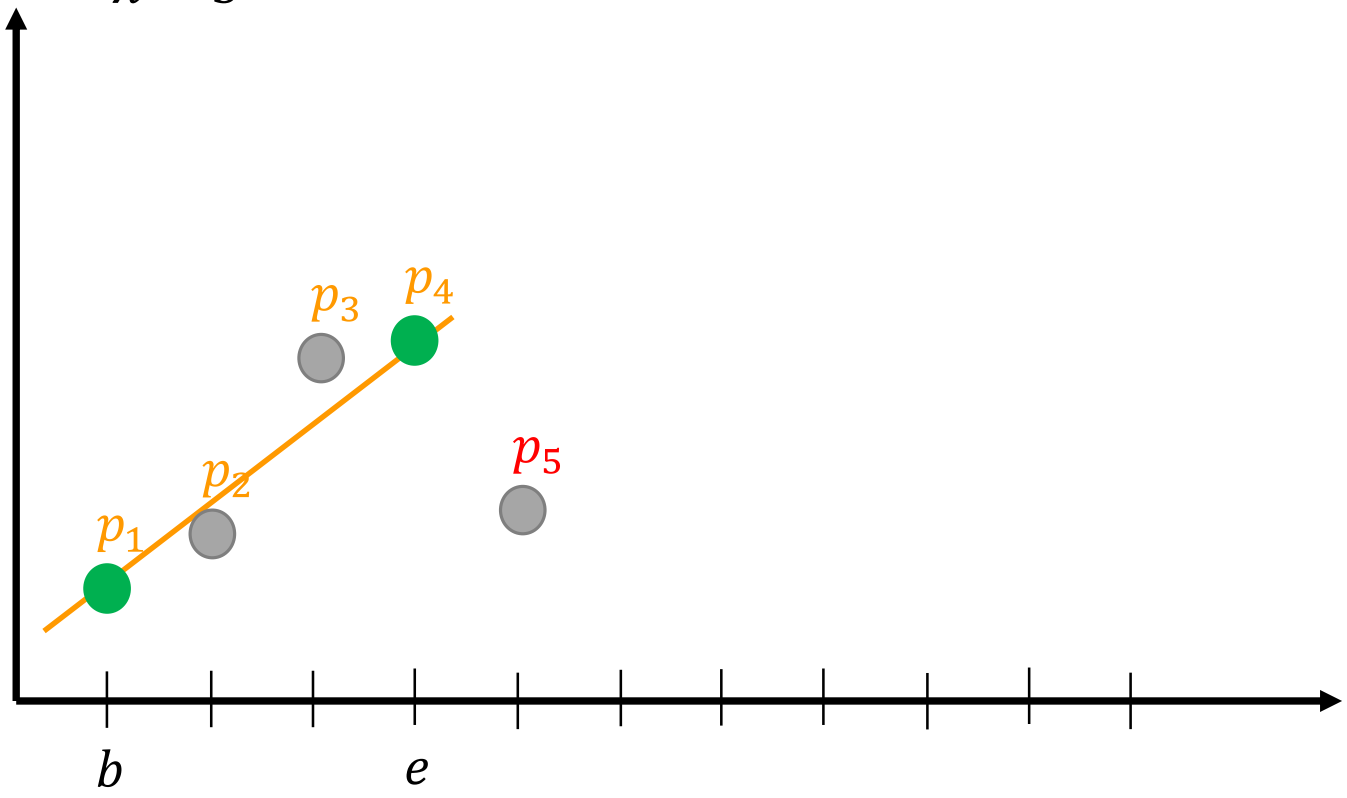
- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.

K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 5$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.

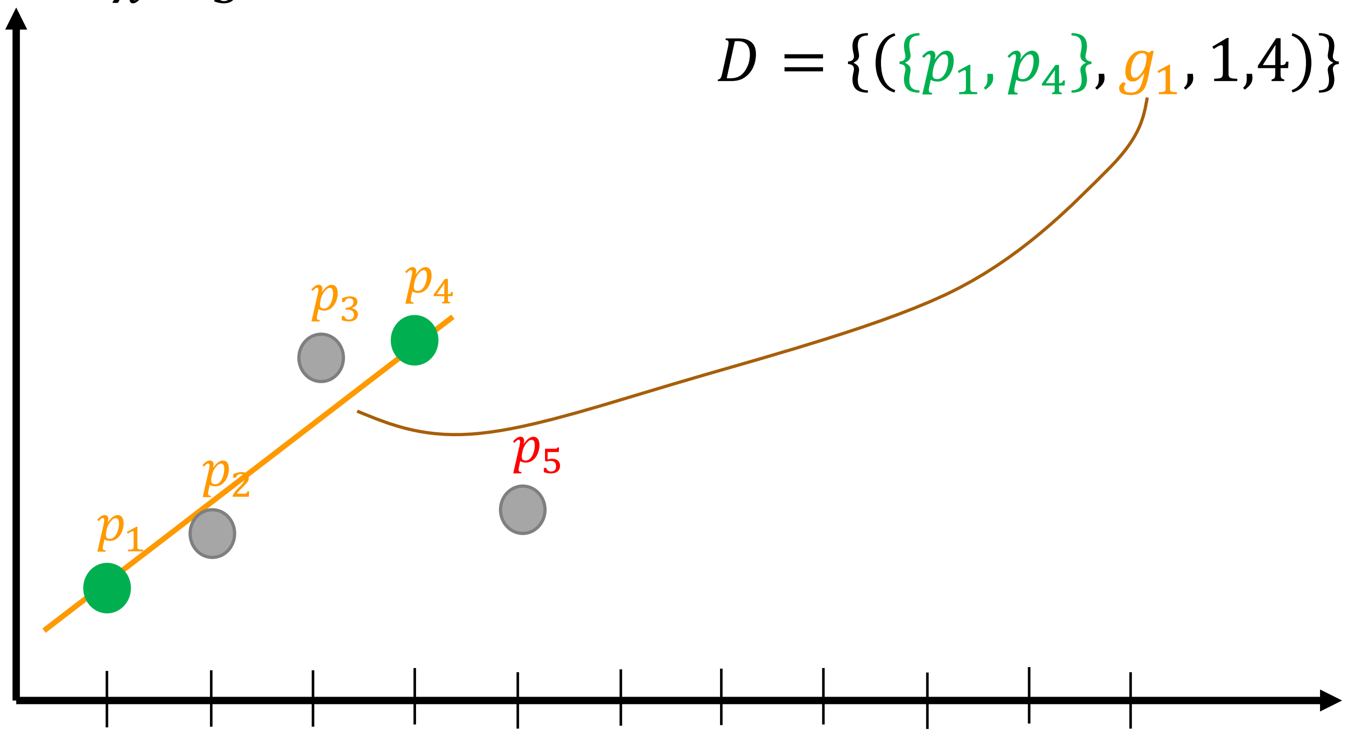
K – segments Algorithm

$\sigma = 2$

$Q = \{(1, p_1), (2, p_2), (3, p_3), (4, p_4), (5, p_5)\}$

$\lambda = 5$

$D = \{(\{p_1, p_4\}, g_1, 1{,}4)\}$



For $i \coloneqq 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.
- $D = D \cup \{(C, g, b, e)\}$.

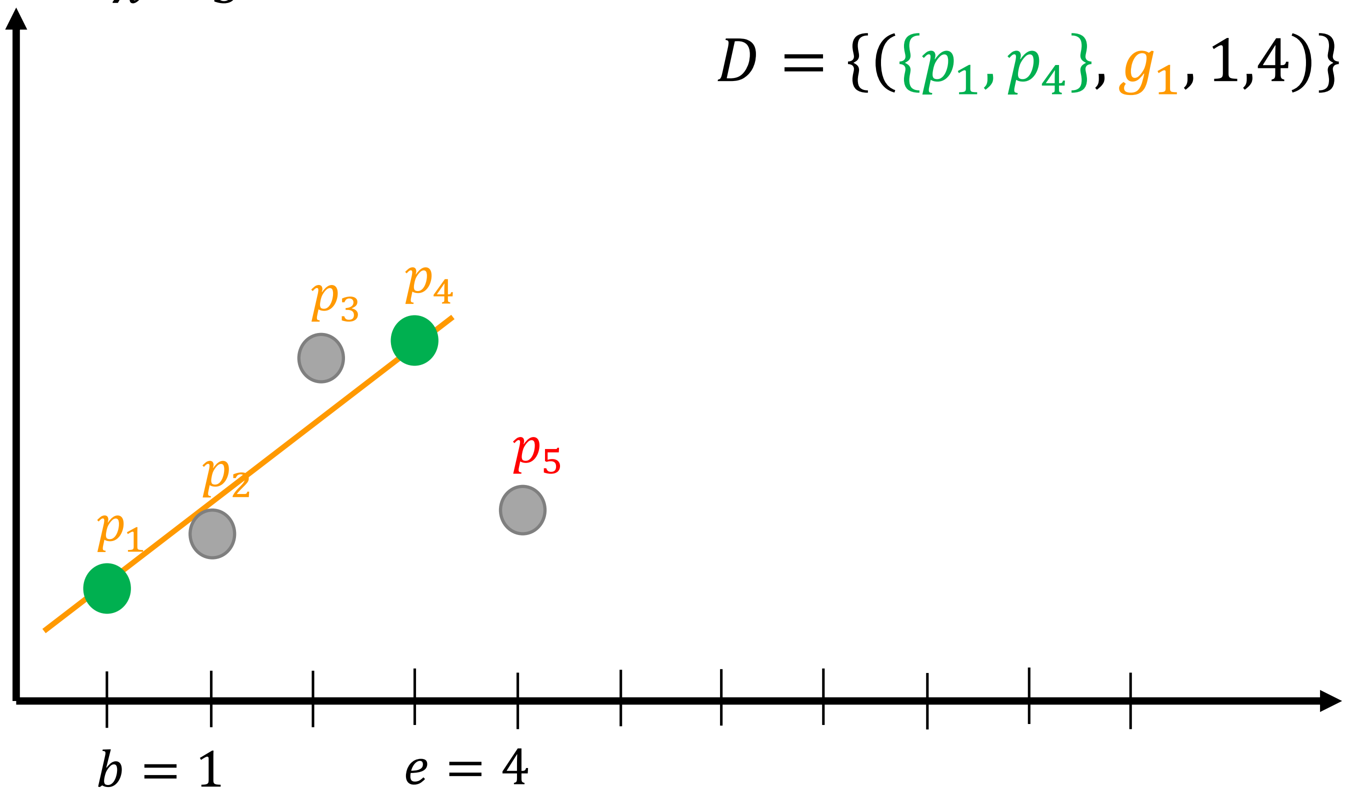
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5)\}$

$\lambda = 5$

$D = \{(\{p_1, p_4\}, g_1, 1{,}4)\}$



For $i \coloneqq 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.
- $D = D \cup \{(C, g, b, e)\}$.
- $Q = \{(i, p_i)\}$.

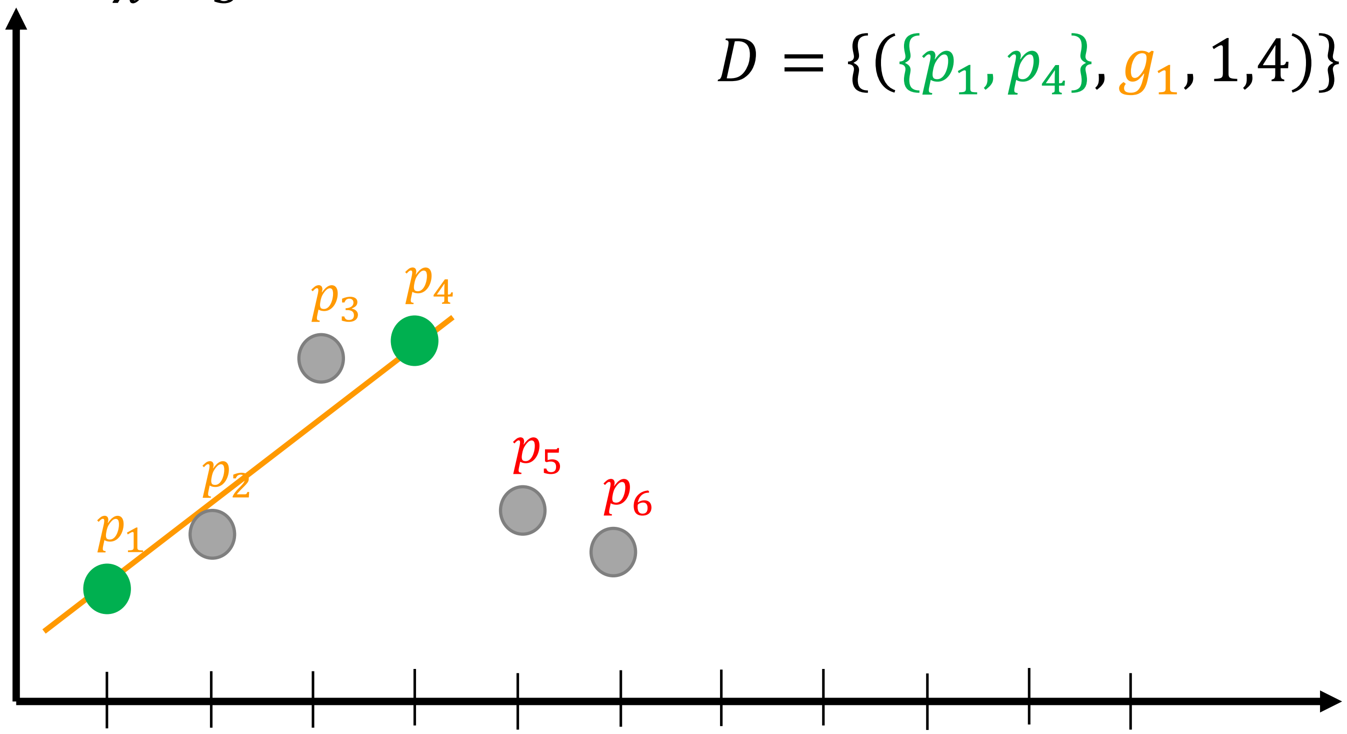
$K -$ segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6)\}$

$\lambda = 5$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i := 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$

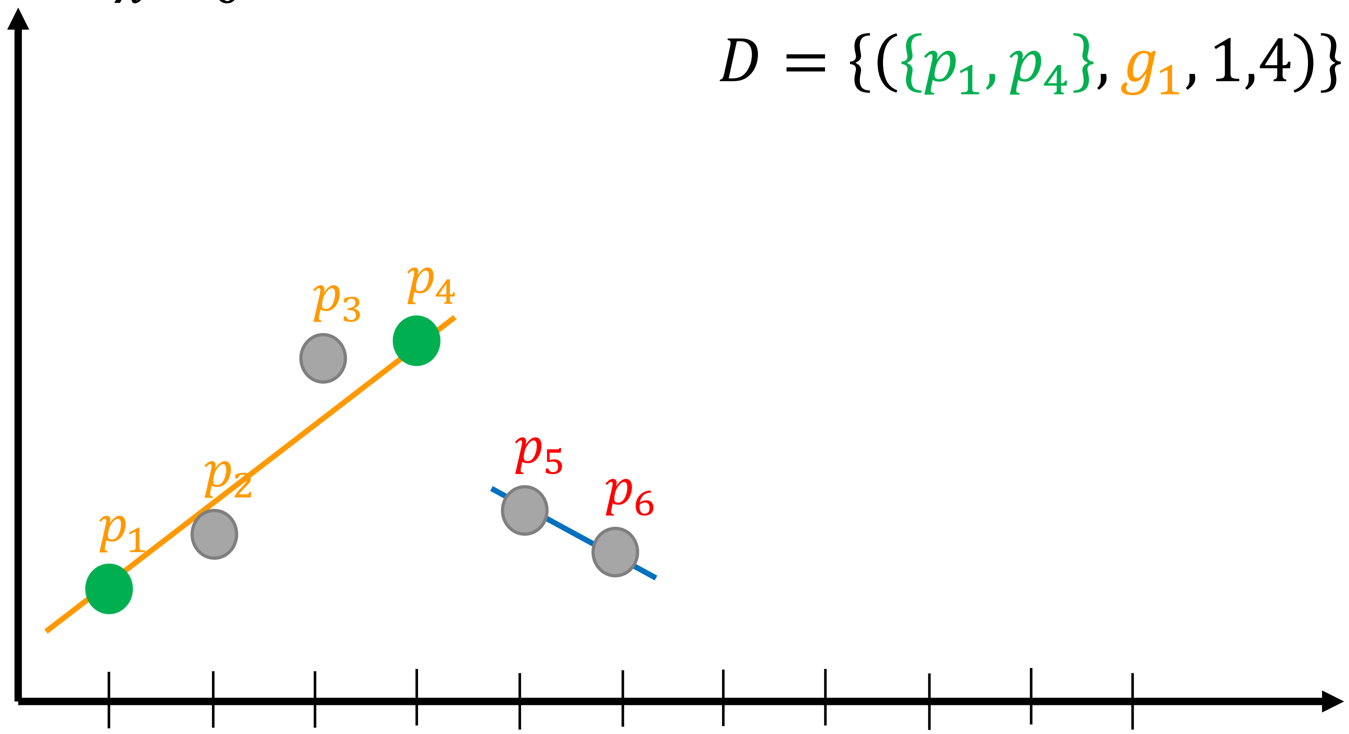
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6)\}$

$\lambda = 0$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

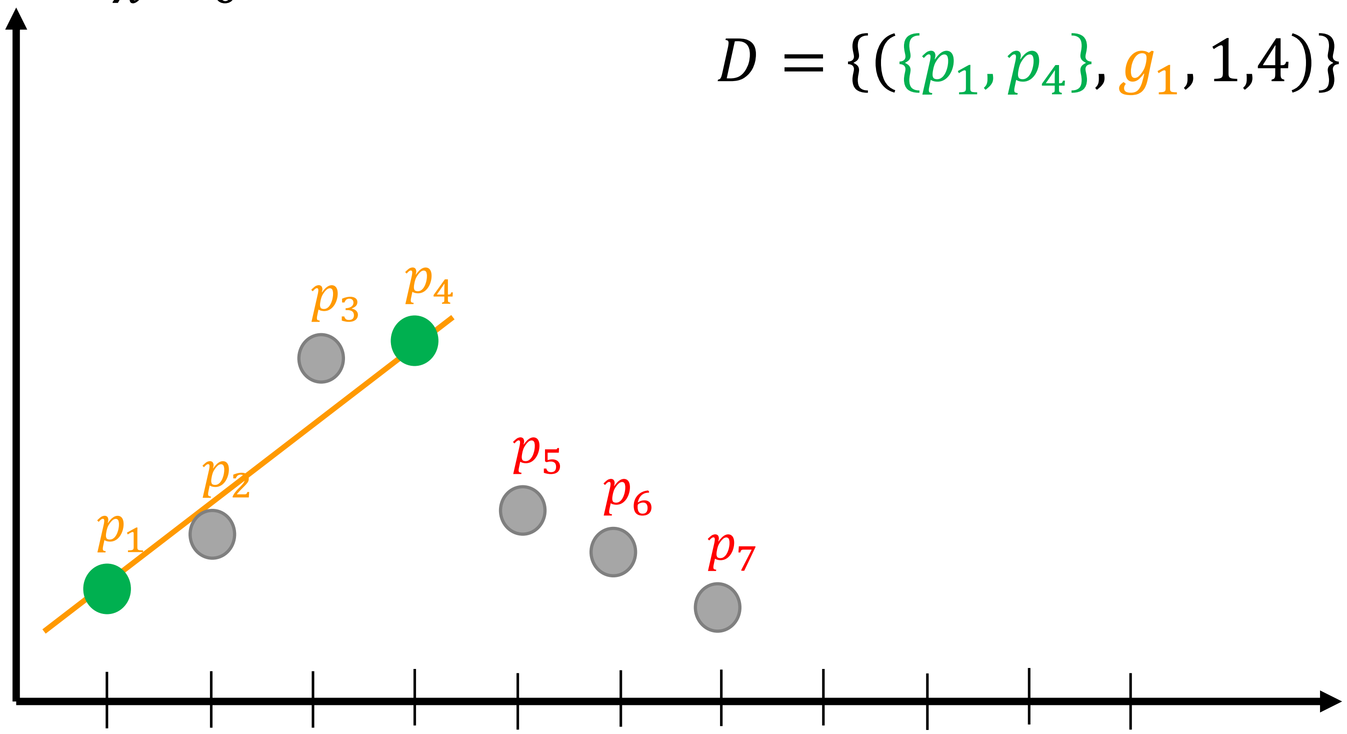
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7)\}$

$\lambda = 0$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i := 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$

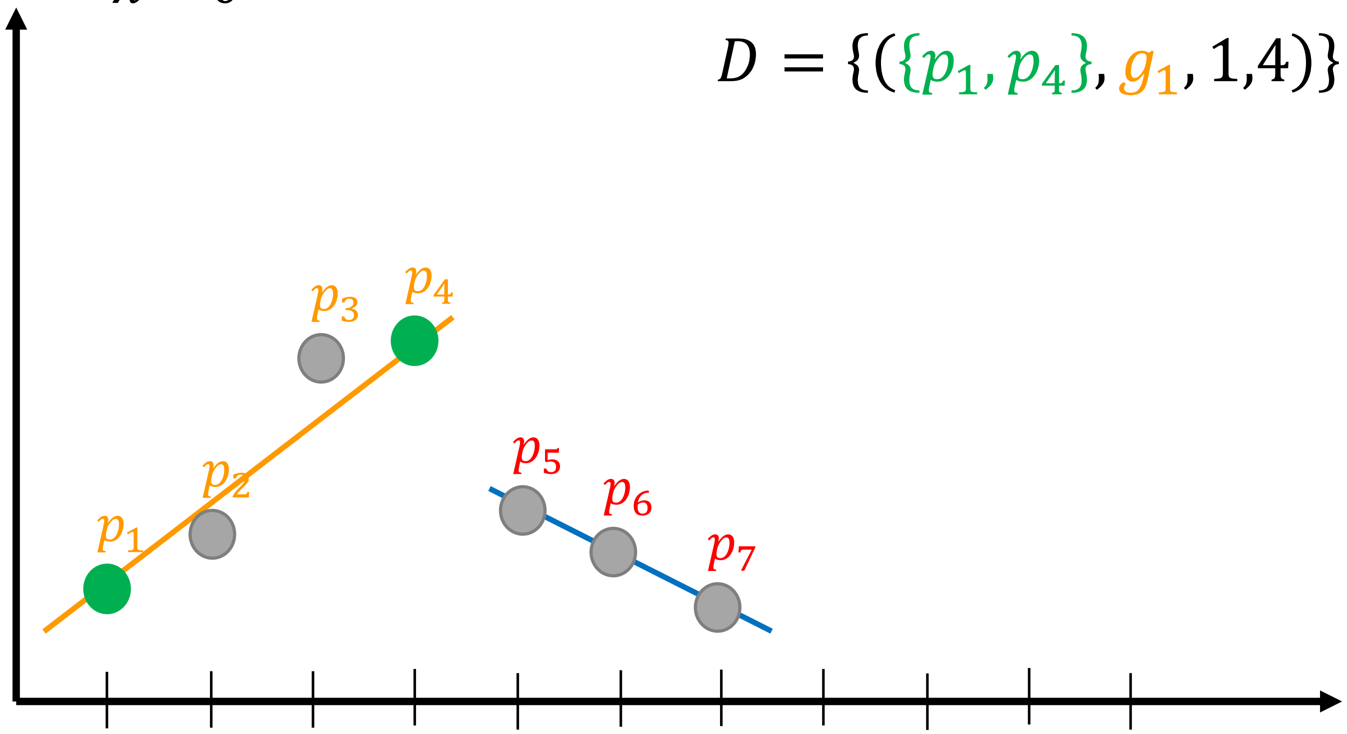
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7)\}$

$\lambda = 0$

$D = \{(\{p_1, p_4\}, g_1, 1, 4)\}$



For $i := 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

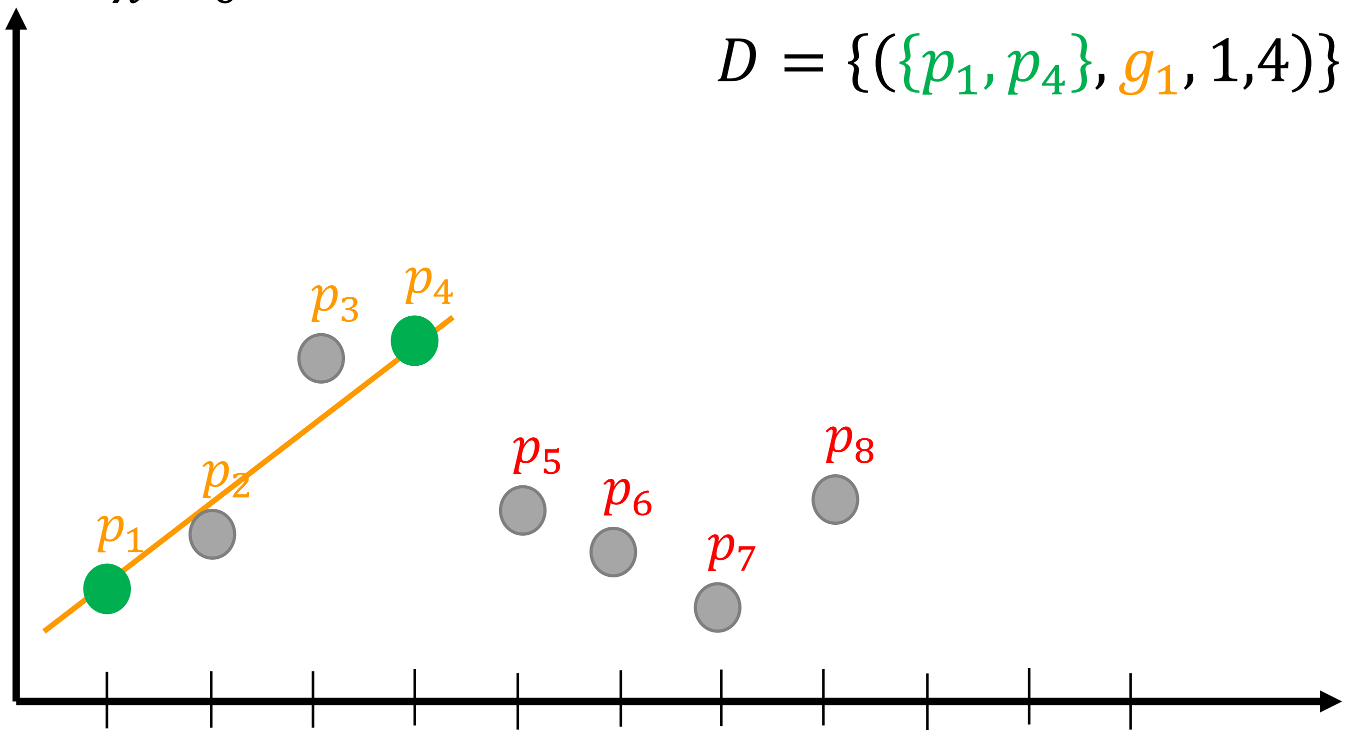
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 0$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i := 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$

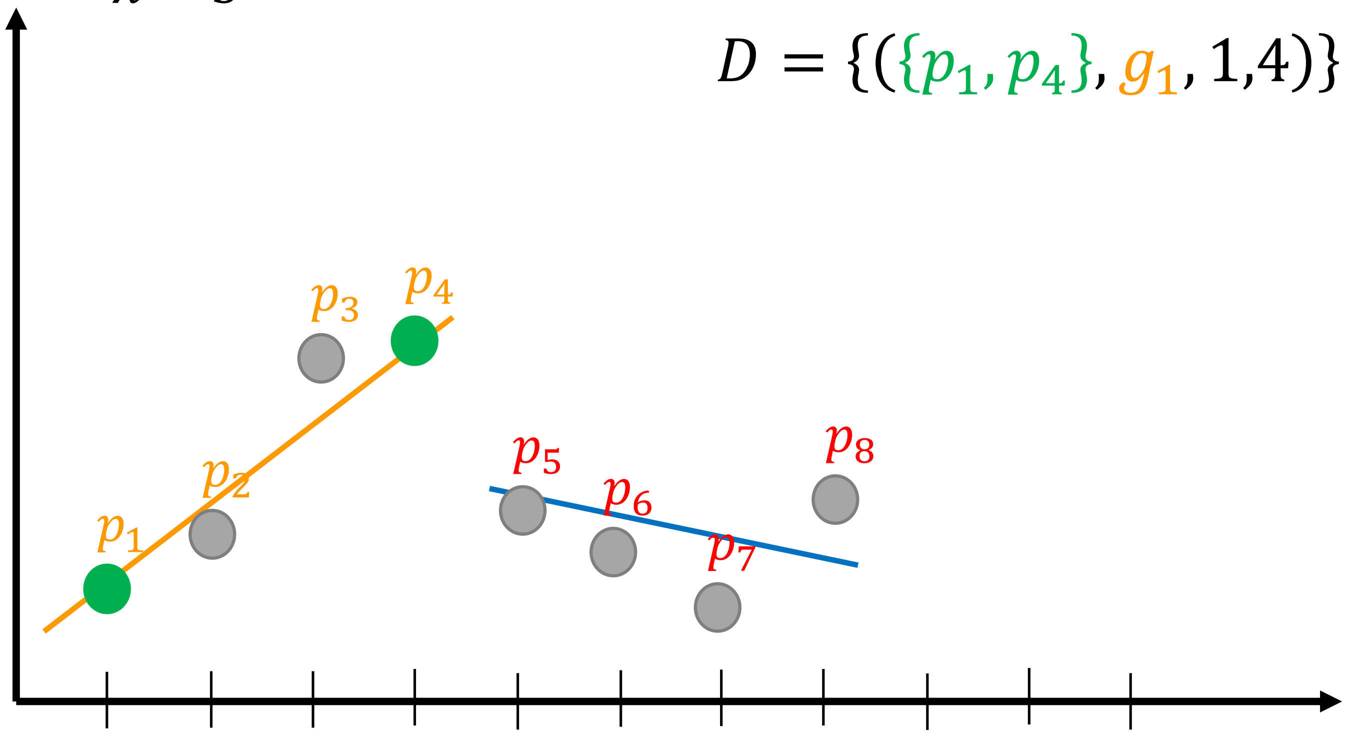
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i \coloneqq 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

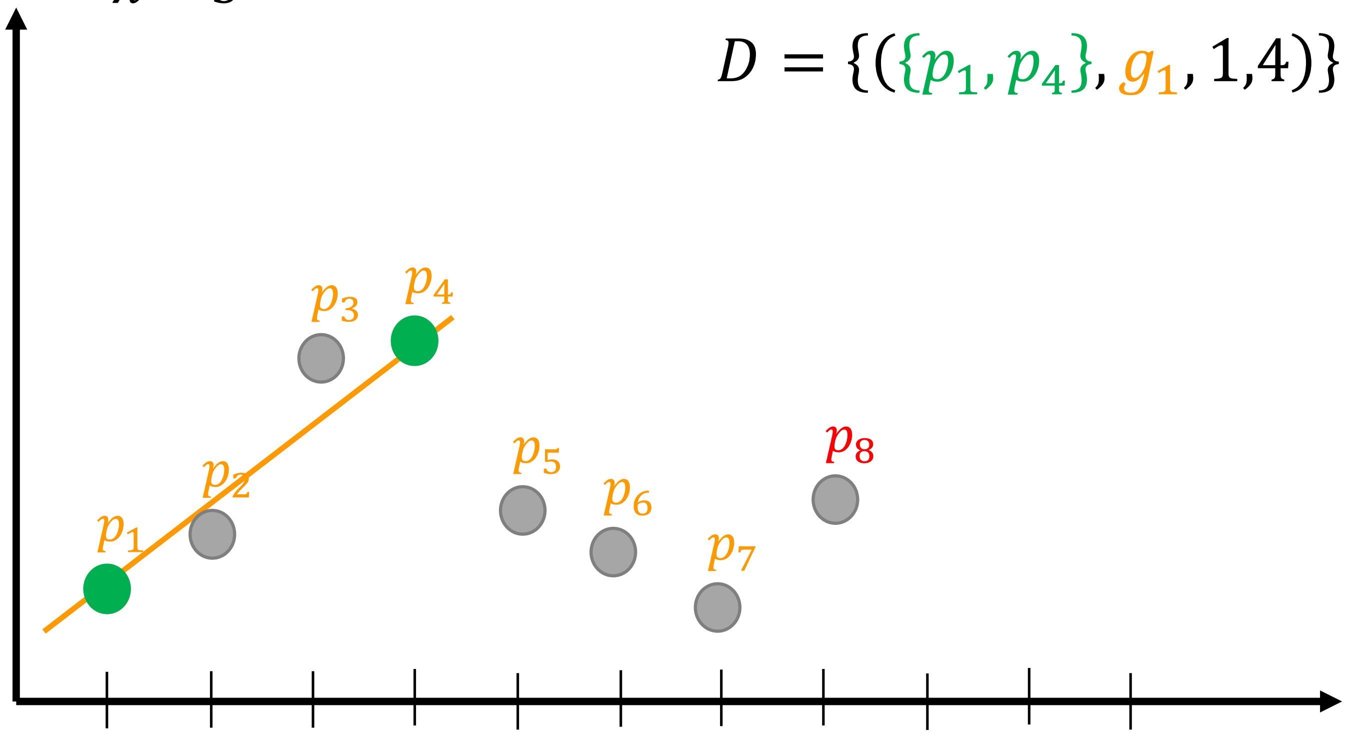
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

  if $\lambda > \sigma$

  - $T = Q \setminus \{(i, p_i)\}$

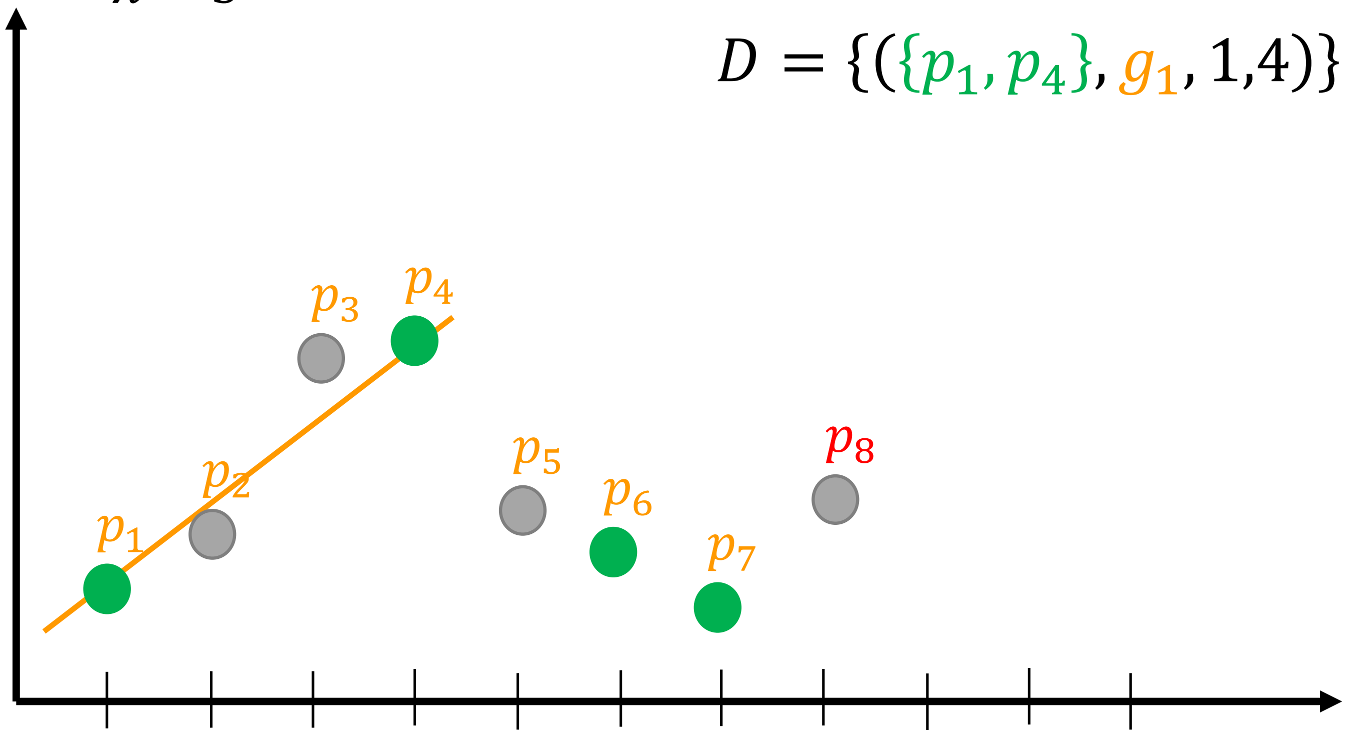
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1,4)\}$



For $i \coloneqq 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- f^* = a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

 if $\lambda > \sigma$

 - $T = Q \setminus \{(i, p_i)\}$
 - $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.

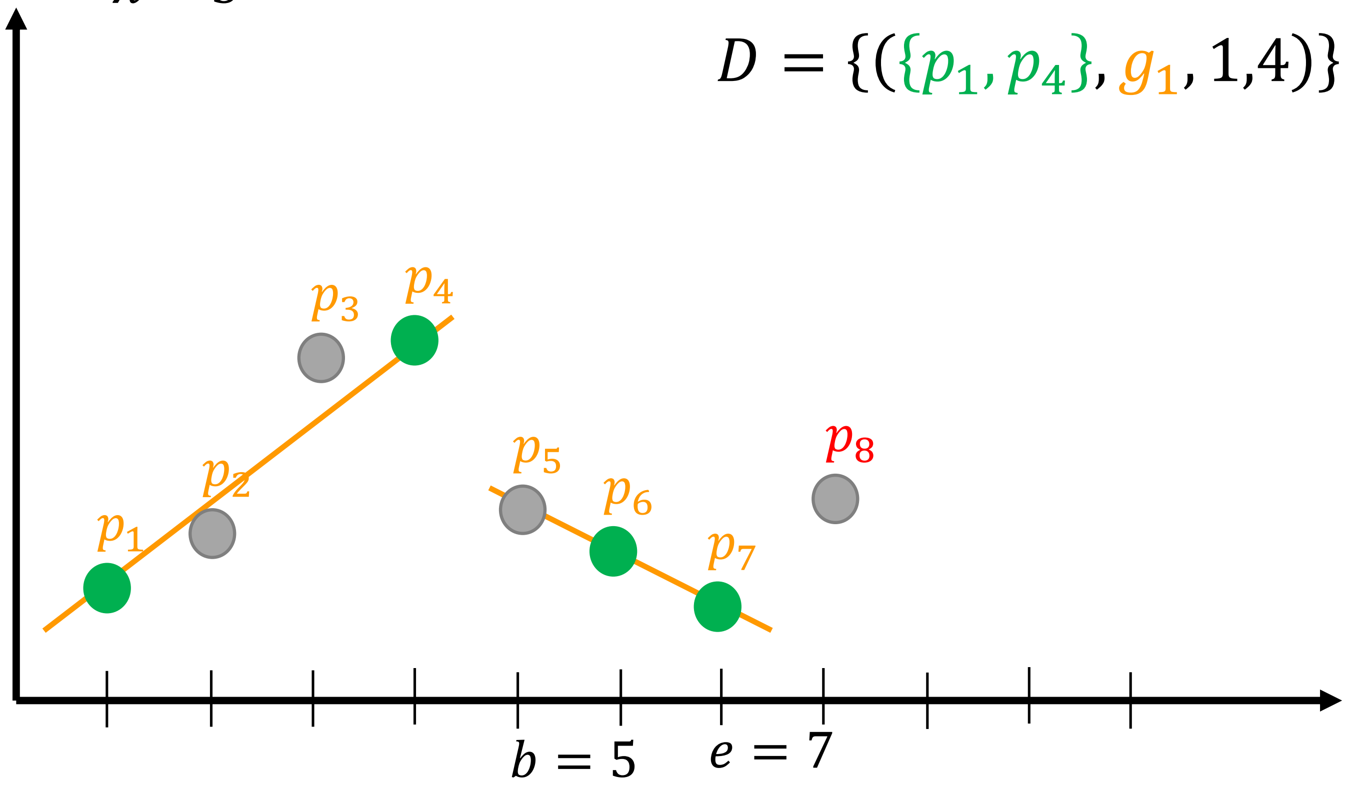
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1{,}4)\}$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.

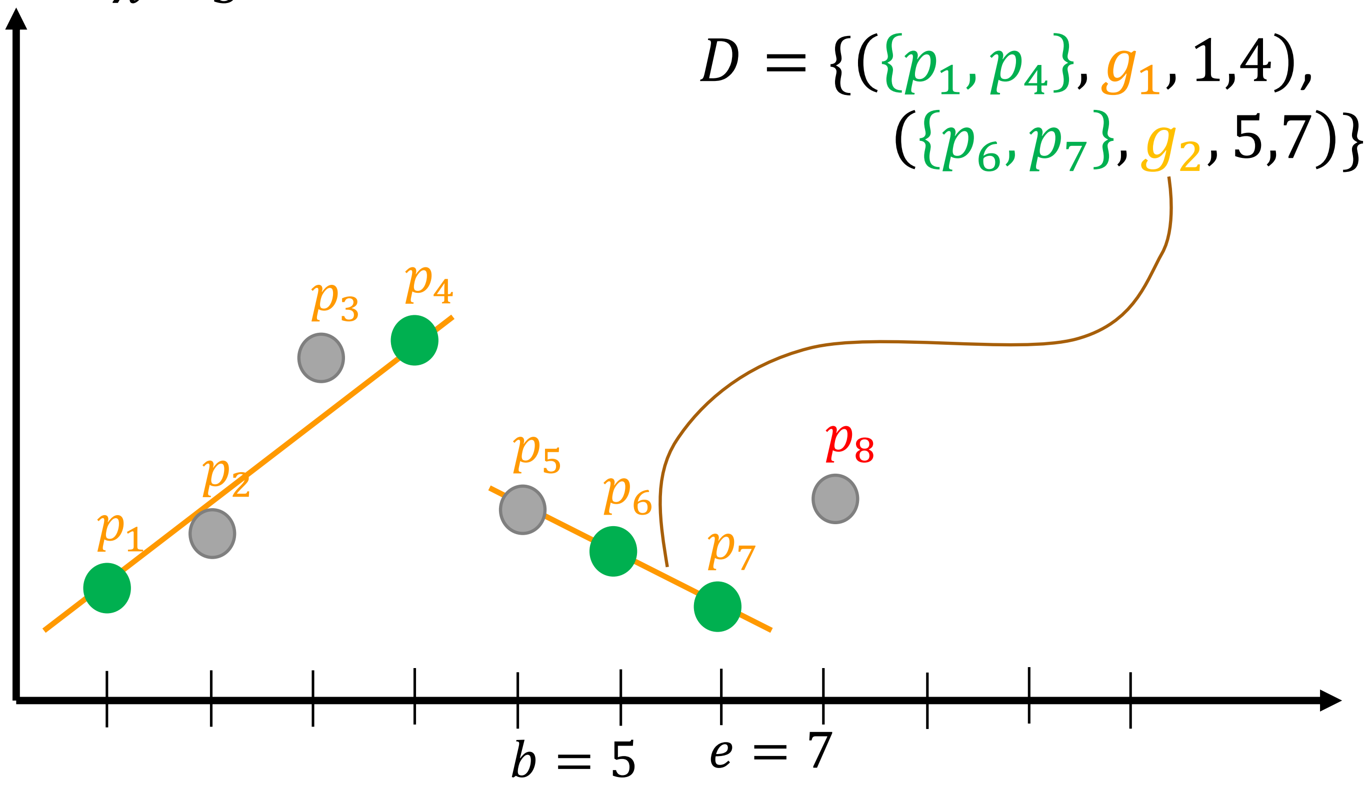
K – segments Algorithm

$\sigma = 2$

$Q = \{(5, p_5), (6, p_6), (7, p_7), (8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1{,}4),$
$(\{p_6, p_7\}, g_2, 5{,}7)\}$



For $i \coloneqq 1 \to n$ do

- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$

- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.
- $D = D \cup \{(C, g, b, e)\}$.

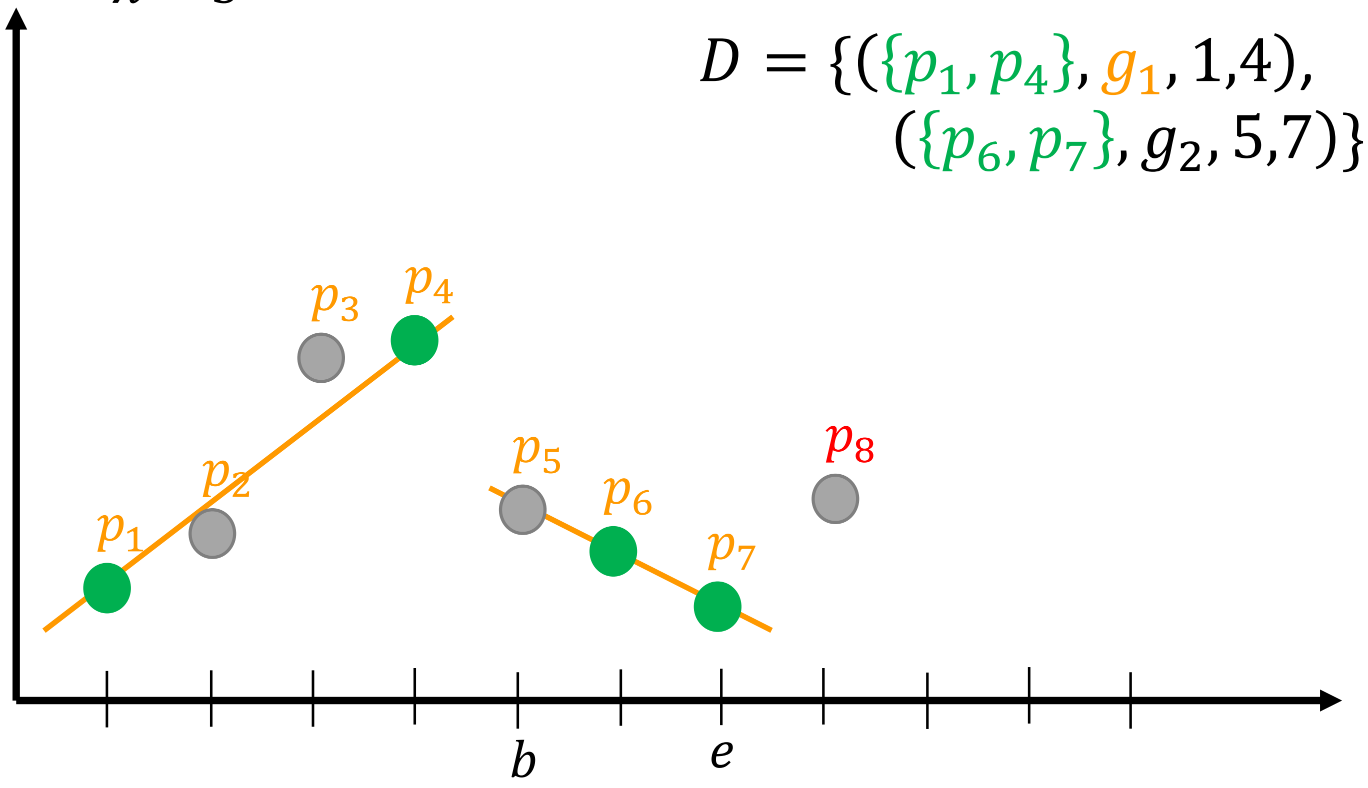
K – segments Algorithm

$\sigma = 2$

$Q = \{(8, p_8)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1,4),$
$(\{p_6, p_7\}, g_2, 5,7)\}$



For $i \coloneqq 1 \to n$ do
- $Q = Q \cup \{(i, p_i)\}$
- $f^* =$ a linear approx. of Q.
- $\lambda = cost(Q, f^*)$

if $\lambda > \sigma$
- $T = Q \setminus \{(i, p_i)\}$
- $C = \left(1, \frac{\epsilon}{4}\right)$-coreset for T.
- $g =$ a linear approx. of T +save endpoints.
- $D = D \cup \{(C, g, b, e)\}$.
- $Q = \{(i, p_i)\}$.

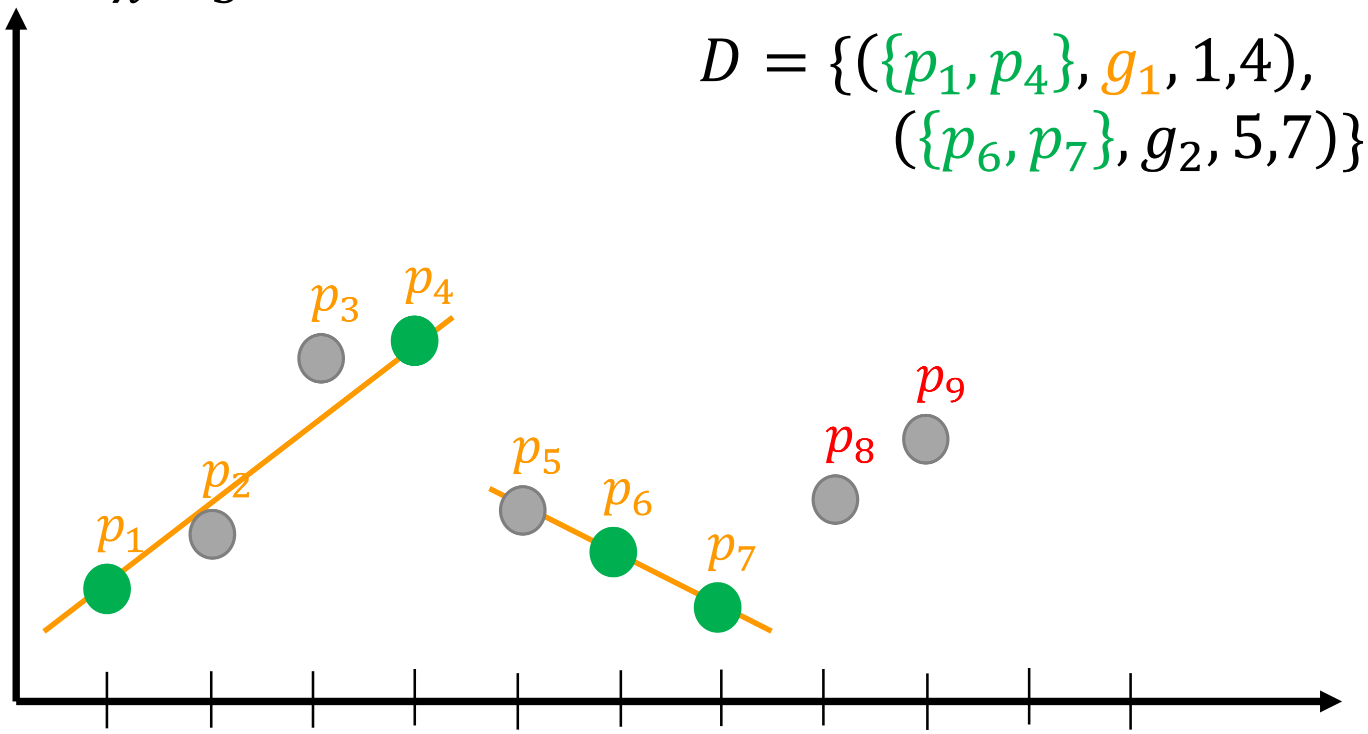
K – segments Algorithm

$\sigma = 2$

$Q = \{(8, p_8), (9, p_9)\}$

$\lambda = 3$

$D = \{(\{p_1, p_4\}, g_1, 1,4),$
$(\{p_6, p_7\}, g_2, 5,7)\}$



For $i \coloneqq 1 \rightarrow n$ do

- $Q = Q \cup \{(i, p_i)\}$

⋮

K – segments Algorithm

Algorithm 2: BALANCEDPARTITION(P, ε, σ)

Input: A set $P = \{(1, p_1), \cdots, (n, p_n)\}$ in $\mathbb{R}^{d+1}$
an error parameters $\varepsilon \in (0, 1/10)$ and $\sigma > 0$.
Output: A set D that satisfies Theorem 4.

$Q := \emptyset$; $D = \emptyset$; p_{n+1}:= an arbitrary point in $\mathbb{R}^d$;
for $i := 1$ *to* $n + 1$ **do**
    $Q := Q \cup \{(i, p_i)\}$; Add new point to tuple
    $f^* :=$ a linear approximation of Q; $\lambda := \text{cost}(Q, f^*)$
    if $\lambda > \sigma$ **or** $i = n + 1$ **then**
        $T := Q \setminus \{(i, p_i)\}$; take all the new points into tuple
        $C :=$ a $(1, \varepsilon/4)$-coreset for T; Approximate points by a local representation
        $g :=$ a linear approximation of T, $b := i - |T|$, $e := i - 1$; save endpoints
        $D := D \cup \{(C, g, b, e)\}$; save a tuple
        $Q := \{(i, p_i)\}$; proceed to new point
return D

We now prove that the output D of our algorithm is a (k, ε)-coreset for P.

K – segments Algorithm (Proof)

Proof Let $m = |D|$ and f be a k-segment. We denote the ith coreset segment in D by (C_i, g_i, b_i, e_i) for every $i \in [m]$. For every $i \in [m]$ we have that C_i is a $(1, \varepsilon/4)$-coreset for a corresponding subset $T = T_i$ of P. By the construction of D we also have $P = T_1 \cup \cdots \cup T_m$.

Using Definition 3 of $\text{cost}'(D, f)$, $\text{Good}(D, f)$ and L_i, we thus have

$$
\begin{aligned}
&|\text{cost}(P, f) - \text{cost}'(D, f)| \\
&= \left| \sum_{i=1}^{m} \text{cost}(T_i, f) - \left(\sum_{i \in \text{Good}(D,f)} \text{cost}(C_i, f) + \sum_{i \in [m] \setminus \text{Good}(D,f)} \text{cost}(L_i, f) \right) \right| \\
&= \left| \sum_{i \in \text{Good}(D,f)} (\text{cost}(T_i, f) - \text{cost}(C_i, f)) + \sum_{i \in [m] \setminus \text{Good}(D,f)} (\text{cost}(T_i, f) - \text{cost}(L_i, f)) \right| \\
&\leq \sum_{i \in \text{Good}(D,f)} |\text{cost}(T_i, f) - \text{cost}(C_i, f)| + \sum_{i \in [m] \setminus \text{Good}(D,f)} |\text{cost}(T_i, f) - \text{cost}(L_i, f)|,
\end{aligned} \tag{2}
$$

where the last inequality is due to the triangle inequality. We now bound each term in the right hand side.

For every $i \in \mathrm{Good}(D, f)$ we have that C_i is a $(1, \varepsilon/4)$-coreset for T_i, so

$$|\mathrm{cost}(T_i, f) - \mathrm{cost}(C_i, f)| \le \frac{\varepsilon \mathrm{cost}(T_i, f)}{4}. \tag{3}$$

For every $i \in [m] \setminus \mathrm{Good}(D, f)$, we have

$$\begin{aligned}
|\mathrm{cost}(T_i, f) - \mathrm{cost}(L_i, f)| &= \left| \sum_{(p,t) \in T_i} \|p - f(t)\|^2 - \sum_{t=b_i}^{e_i} \|g_i(t) - f(t)\|^2 \right| \\
&= \left| \sum_{(p,t) \in T_i} \left(\|p - f(t)\|^2 - \|g_i(t) - f(t)\|^2 \right) \right| && (4) \\
&\le \sum_{(p,t) \in T_i} \left| \|p - f(t)\|^2 - \|g_i(t) - f(t)\|^2 \right| && (5) \\
&\le \sum_{(p,t) \in T_i} \left(\frac{12\|g_i(t) - p\|^2}{\varepsilon} + \frac{\varepsilon \|p - f(t)\|^2}{2} \right) && (6) \\
&= \frac{12 \mathrm{cost}(T_i, g_i)}{\varepsilon} + \frac{\varepsilon \mathrm{cost}(T_i, f)}{2} \le \frac{24\sigma}{\varepsilon} + \frac{\varepsilon \mathrm{cost}(T_i, f)}{2}, && (7)
\end{aligned}$$

where (5) is by the triangle inequality, and (6) is by the weak triangle inequality (see (Feldman et al., 2013, Lemma 7.1)). The inequality in (7) is because by construction $\mathrm{cost}(T, f^*) \le \sigma$ for some 2-approximation f^* of the 1-segment mean of T. Hence, $\mathrm{cost}(T, g_i) \le 2\mathrm{cost}(T, f^*) \le 2\sigma$.

Plugging (7) and (3) in (2) yields

$$
\begin{aligned}
|\mathrm{cost}(P,f)-\mathrm{cost}'(D,f)| &\le \sum_{i\in \mathrm{Good}(D,f)} \frac{\varepsilon \mathrm{cost}(T_i,f)}{4} + \sum_{i\in [m]\setminus \mathrm{Good}(D,f)} \left(\frac{24\sigma}{\varepsilon}+\frac{\varepsilon}{2}\mathrm{cost}(T_i,f)\right)\\
&\le \left(\frac{\varepsilon}{4}+\frac{\varepsilon}{2}\right)\mathrm{cost}(P,f)+\frac{24k\sigma}{\varepsilon},
\end{aligned}
$$

where in the last inequality we used that fact that $|[m]\setminus \mathrm{Good}(D,f)| \le k-1 < k$ since f is a k-segment. Substituting σ yields

$$
\begin{aligned}
|\mathrm{cost}(P,f)-\mathrm{cost}'(D,f)| &\le \frac{3\varepsilon}{4}\mathrm{cost}(P,f)+\frac{\varepsilon \mathrm{cost}(P,h)}{4\alpha}\\
&\le \frac{3\varepsilon}{4}\mathrm{cost}(P,f)+\frac{\varepsilon \mathrm{cost}(P,f)}{4} = \varepsilon \mathrm{cost}(P,f).
\end{aligned}
$$

Bound on $|D|$: Let $i \in [m-1]$, consider the values of T, Q and λ during the execution of Line 7 when $T = T_i$ is constructed. Let $Q_i = Q$ and $\lambda_i = \lambda$. The cost of the 1-segment mean of Q_i is at least $\lambda_i/2 > \sigma/2 > 0$, which implies that $|Q_i| \ge 3$ and thus $|T_i| \ge 1$. Since Q_{i-1} is the union of T_{i-1} with the first point of T_i we have $Q_{j-1} \subseteq T_{i-1} \cup T_j$. By letting g^* denote a 1-segment mean of $T_{i-1} \cup T_i$ we have

$$
\mathrm{cost}(T_{i-1}\cup T_i, g^*) \ge \mathrm{cost}(Q_{i-1}, g^*) \ge \lambda_i/2 > \sigma/2.
$$

Bound on $|D|$: Let $i \in [m-1]$, consider the values of T, Q and λ during the execution of Line 7 when $T = T_i$ is constructed. Let $Q_i = Q$ and $\lambda_i = \lambda$. The cost of the 1-segment mean of Q_i is at least $\lambda_i/2 > \sigma/2 > 0$, which implies that $|Q_i| \geq 3$ and thus $|T_i| \geq 1$. Since Q_{i-1} is the union of T_{i-1} with the first point of T_i we have $Q_{j-1} \subseteq T_{i-1} \cup T_j$. By letting g^* denote a 1-segment mean of $T_{i-1} \cup T_i$ we have

$$\mathrm{cost}(T_{i-1} \cup T_i, g^*) \geq \mathrm{cost}(Q_{i-1}, g^*) \geq \lambda_i/2 > \sigma/2.$$

Suppose that for our choice of $i \in [m-1]$, the points in $T_{i-1} \cup T_j$ are served by a single segment of h, i.e, $\{h(t) \mid b_{i-1} \leq t \leq e_i\}$ is a linear segment. Then

$$\mathrm{cost}(T_{i-1}, h) + \mathrm{cost}(T_i, h) = \mathrm{cost}(T_{i-1} \cup T_i, h) \geq \mathrm{cost}(T_{i-1} \cup T_i, g^*) > \sigma/2. \tag{8}$$

Let $G \subseteq [m-1]$ denote the union over all values $i \in [m-1]$ such that i is both even and satisfies (8). Summing (8) over G yields

$$\mathrm{cost}(P, h) = \sum_{i \in [m]} \mathrm{cost}(T_i, h) \geq \sum_{i \in G} (\mathrm{cost}(T_{i-1}, h) + \mathrm{cost}(T_i, h)) \geq |G|\sigma/2. \tag{9}$$

Since h is a (βk)-segment, at most $(\beta k)-1$ sets among $T_1, \cdots, T_m$ are not served by a single segment of h, so $|G| \geq (m-\beta k)/2$. Plugging this in (9) yields $\text{cost}(P,h) \geq (m-\beta k)\sigma/4$. Rearranging,

$$m \leq \frac{4\text{cost}(P,h)}{\sigma} + \beta k = O\left(\frac{k\alpha}{\varepsilon^2}\right) + \beta k. \tag{10}$$

Running time:

In a few slides we will show an algorithm to compute a $(1,\varepsilon)$-coreset C in time $O\left(\frac{nd}{\varepsilon^4}\right)$ for n points. This algorithm is dynamic and supports insertions of a new point in $O\left(\frac{d}{\varepsilon^4}\right)$ time. Therefore, updating the 1-segment mean f^* and the coreset C can be done in $O\left(\frac{d}{\varepsilon^4}\right)$ time per point, and the overall time is $O\left(\frac{nd}{\varepsilon^4}\right)$ time.

Coreset for 1-segment Mean

Algorithm 7: 1-SEGMENTCORESET(P)

Input: A signal $P = \{(t_1, p_1), \cdots, (t_n, p_n)\}$ in $\mathbb{R}^{d+1}$.

Output: A $(1, 0)$-coreset (C, w) that satisfies Claim 15.

Set $X \in \mathbb{R}^{n \times (d+2)}$ to be matrix whose ith row is $(1, t_i, p_i)$ for every $i \in [n]$.

Compute the thin SVD $X = U\Sigma V^T$ of X.

Set $u \in \mathbb{R}^{d+2}$ to be the leftmost column of ΣV^T.

Set $w := \frac{\|u\|^2}{d+2}$. `/* w > 0 since ||Σ|| = ||X|| > 0 */`

Set $Q, Y \in \mathbb{R}^{(d+2)\times(d+2)}$ to be unitary matrices whose leftmost columns are $u/\|u\|$ and $(\sqrt{w}, \cdots, \sqrt{w})/\|u\|$ respectively.

Set $B \in \mathbb{R}^{(d+2)\times(d+1)}$ to be the $(d+1)$ rightmost columns of $YQ^T\Sigma V^T/\sqrt{w}$.

Set $C \subseteq \mathbb{R}^{d+1}$ to be the union of the rows in B ;

return (C, w)

Coreset for 1-segment Mean

Claim 15: Accurate (1,0)-coreset

Let $P \subseteq \mathbb{R}^{d+1}$ be a signal, $k \geq 1$. Let (C, w) be an output of a call to 1-$SEGMENCORESET(P)$. Then (C, w) is a $(1,0)$-coreset for P of size $|C| = d + 1$. Formally, for every 1-segment f we have

$$cost(P, f) = w \cdot cost(C, f).$$

Moreover, C and w can be computed in $O(nd^2)$ time.

The size and running time of the above $(1,0)$-coreset C might be too large. Therefore, we then show how to construct a $(1, \varepsilon)$-coreset of size $O\left(\frac{1}{\varepsilon^2}\right)$ that takes $O\left(\frac{nd}{\varepsilon^4}\right)$ time.

Coreset for 1-segment Mean

Proof:

Let f be a 1-segment. Hence, there are row vectors $a, b \in \mathbb{R}^d$ such that $f(t) = a + b \cdot t$ for every $t \in \mathbb{R}$. Bu definition of Q and Y we have that $\frac{YQ^T u}{\|u\|} = \frac{(\sqrt{w},\ldots,\sqrt{w})^T}{\|u\|}$. The leftmost column of $YQ^T\Sigma V^T$ is thus $YQ^T u = (\sqrt{w}, \ldots, \sqrt{w})^T$.

Therefore,

$$cost(P, f) = \sum_{(t,p)\in P}^{n} \|f(t) - p\|^2 = \sum_{(t,p)\in P} \|a + b \cdot t - p\|^2$$

$$= \left\| \begin{bmatrix} 1 & t_1 & p_1 \\ & \vdots & \\ 1 & t_n & p_n \end{bmatrix} \begin{bmatrix} a \\ b \\ -I \end{bmatrix} \right\|^2 = \left\| U\Sigma V^T \begin{bmatrix} a \\ b \\ -I \end{bmatrix} \right\|^2 = \left\| YQ^T\Sigma V^T \begin{bmatrix} a \\ b \\ -I \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} \sqrt{w} & \\ \vdots & \sqrt{w}B \\ \sqrt{w} & \end{bmatrix} \begin{bmatrix} a \\ b \\ -I \end{bmatrix} \right\|^2$$

$$w \left\| \begin{bmatrix} 1 & \\ \vdots & B \\ 1 & \end{bmatrix} \begin{bmatrix} a \\ b \\ -I \end{bmatrix} \right\|^2 = w \cdot \sum_{(t,p)\in B} \|a + b \cdot t - p\|^2 = w \cdot cost(C, f).$$

Coreset for 1-segment Mean

Smaller coreset with less computation time.

Theorem: $(1, \varepsilon)$-coreset

Let $P \subseteq \mathbb{R}^{d+1}$ and let $\varepsilon > 0$.

A $(1, \varepsilon)$-coreset $C \subseteq \mathbb{R}^{d+1}$ for P of size $|C| = O\left(\frac{1}{\varepsilon^2}\right)$ can be computed in $O\left(\frac{nd}{\varepsilon^4}\right)$ time.

Proof It was proven in Feldman et al. (2013) that a coreset for P and a family of query shapes, where each shape is spanned by $O(1)$ vectors in $\mathbb{R}^d$, can be computed by projecting P on a $(1/\varepsilon^2)$ dimensional subspace S that minimizes the sum of squared distances to P up to a $(1+\varepsilon)$ factor. The resulting coreset approximates the sum of squared distances to every such shape up to a factor of $(1+\varepsilon)$. The size of this coreset is n, the same as the input size, however the coreset is contained in an $O(1/\varepsilon^2)$ dimensional subspace. We then compute a $(1, 0)$-coreset C for this low dimensional set of n points in $s = O(1/\varepsilon^2)$ space using Algorithm 7, as per Claim 15. This will take additional $O(ns^2)$ time and the resulting coreset will be of size $O(s)$.

The subspace S can be computed deterministically in $O(nd/\varepsilon^4)$ using a recent result of Ghashami and Phillips (2014).. ■

Coreset for 1-segment Mean

Corollary:

Let $\varepsilon \in (0,1)$. A $(1+\varepsilon)$-approximation to the 1-segment mean of P can be computed in $O\left(\frac{nd}{\varepsilon^4}\right)$ time.

Proof:

Based on the previous Theorem, we can compute a $(1,\varepsilon)$-coreset C of size $|C| = O\left(\frac{1}{\varepsilon^2}\right)$ in $O\left(\frac{nd}{\varepsilon^4}\right)$ time. Then, using the singular value decomposition (solving linear regression), it is easy to compute a 1-segment mean f of C in $O(d \cdot |C|^2) = O\left(\frac{d}{\varepsilon^4}\right)$ time. Let f^* be a 1-segment mean of P and f be a 1-segment mean of C. Then

$$cost(P,f) \le (1+\varepsilon)cost(C,f) \le (1+\varepsilon)cost(C,f^*) \le (1+\varepsilon)^2 cost(P,f^*) \le (1+3\varepsilon)cost(P,f^*).$$

Replacing ε with $\frac{\varepsilon}{3}$ in the above proof proves the corollary.