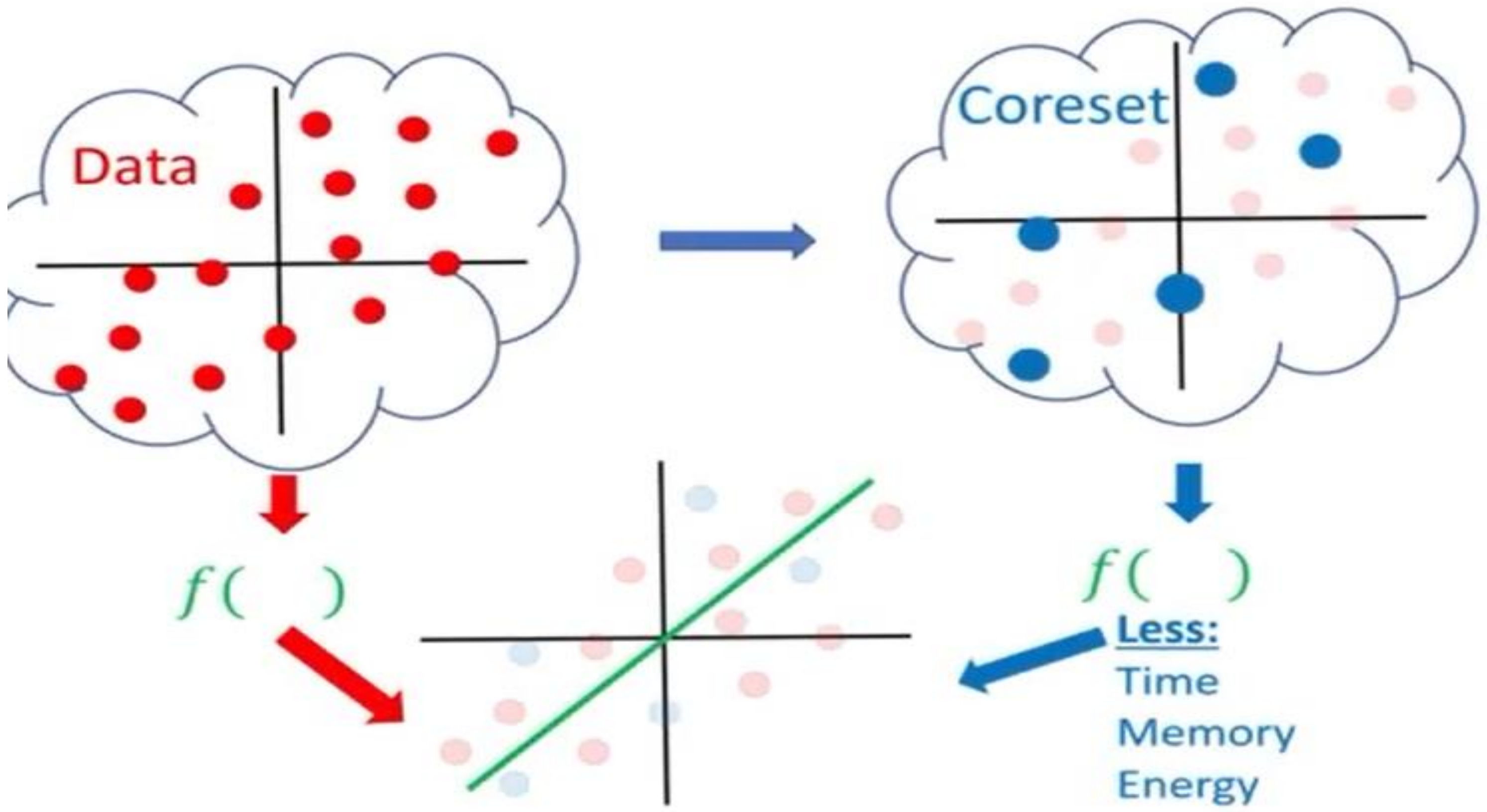


Introduce to Coreset

Outline of presentation:

I Motivation & Coreset

II How to compute a Coreset?

III Applications

I: Motivation & Coreset

I.1 Motivation

I.2 What is coreset?

I.3 Coreset properties

I.1: Motivation

Motivation: Big Data

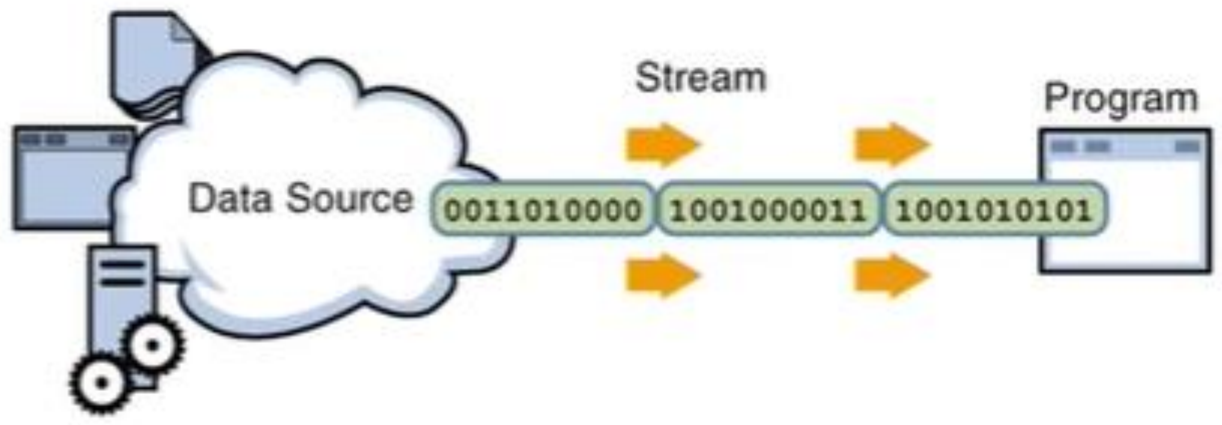
How BIG is Big Data?

Limited hardware
- Computation: IoT
- Energy: smartphones

Limited time
- Real-time decision making

Motivation: Streaming & Distributed Data

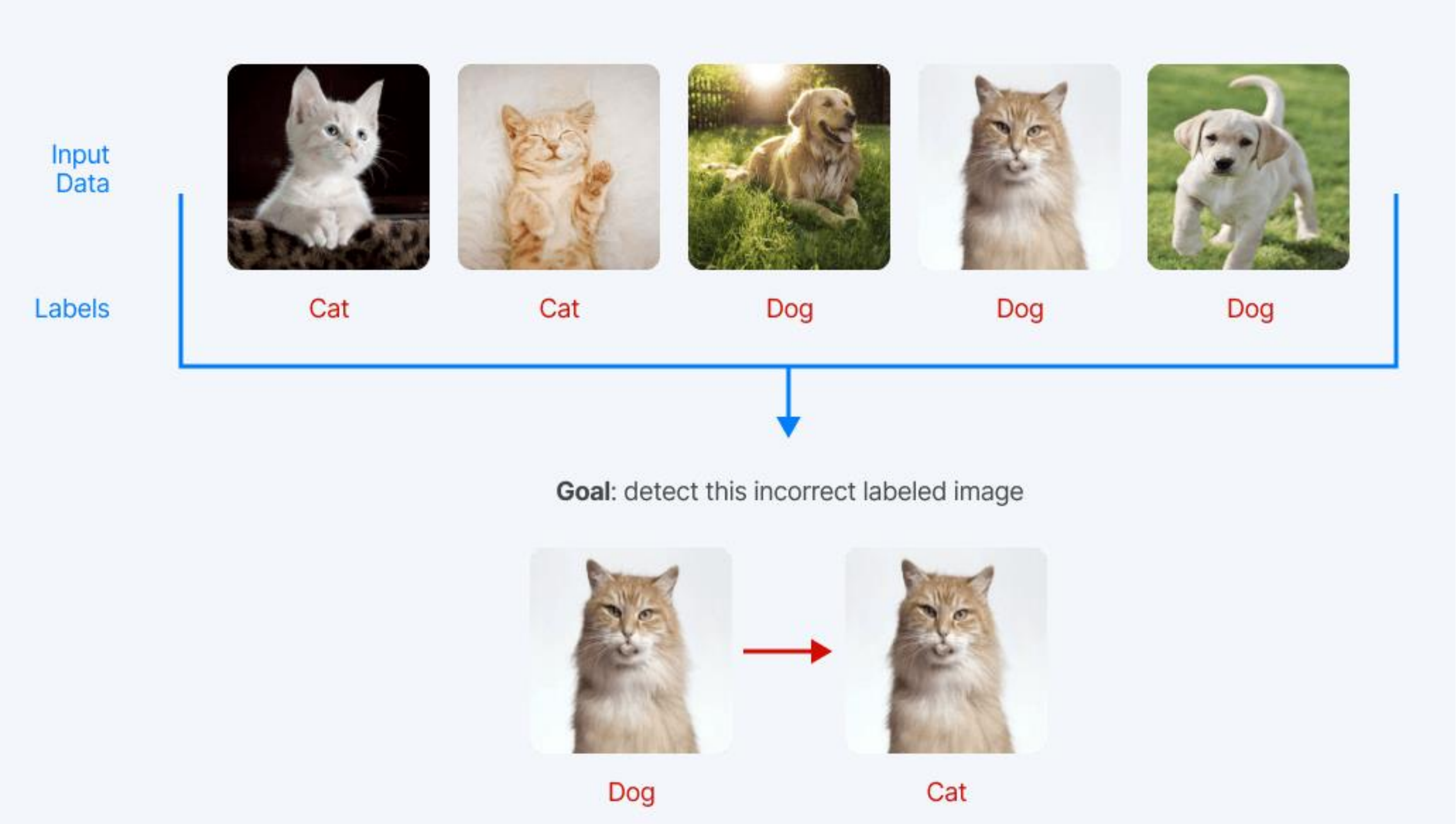
New computation model



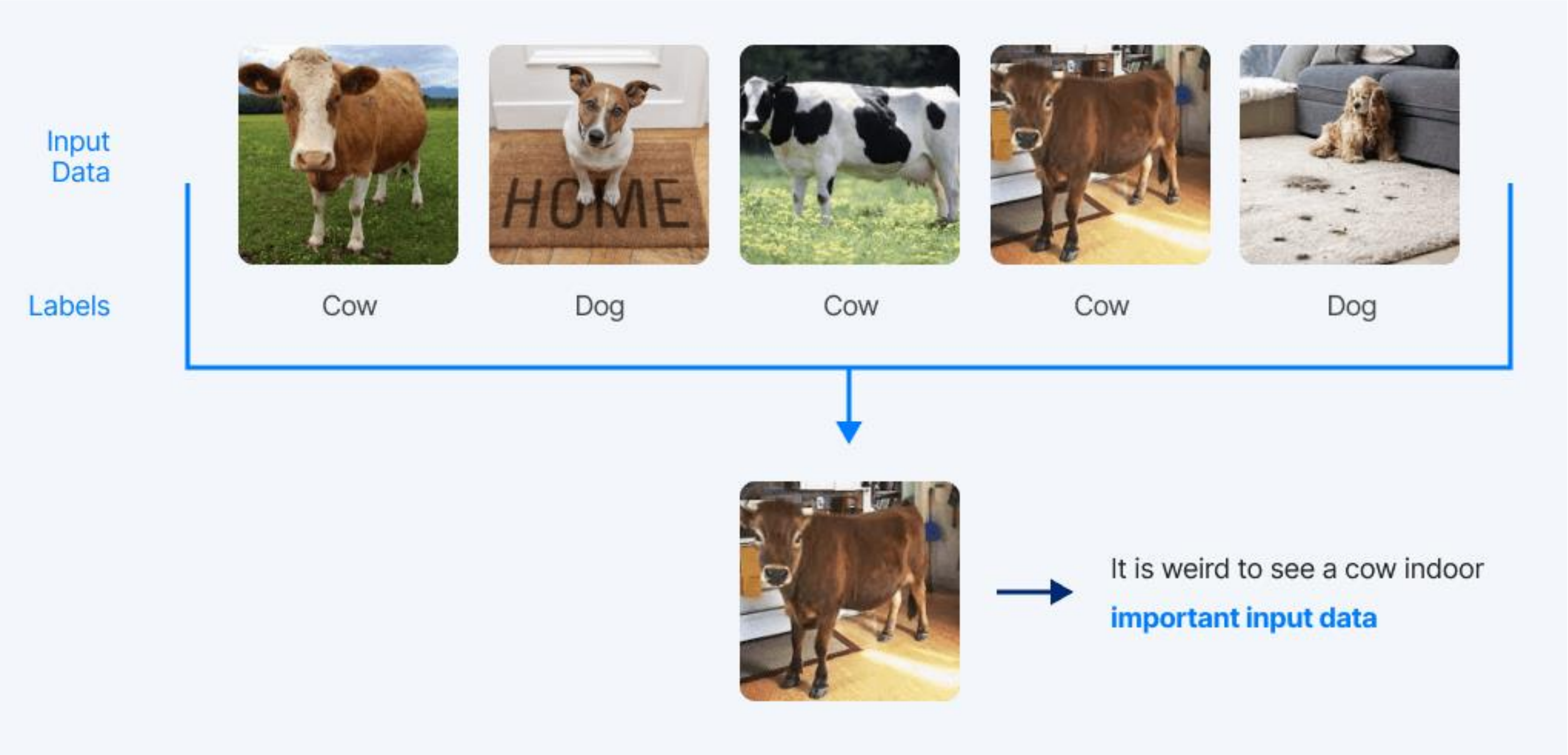
Streaming real-time data
- Data is being received in chunks
- Or one by one

Distributed data
- Across multiple machine
- Or in the cloud

Motivation: Incorrectly labeled data

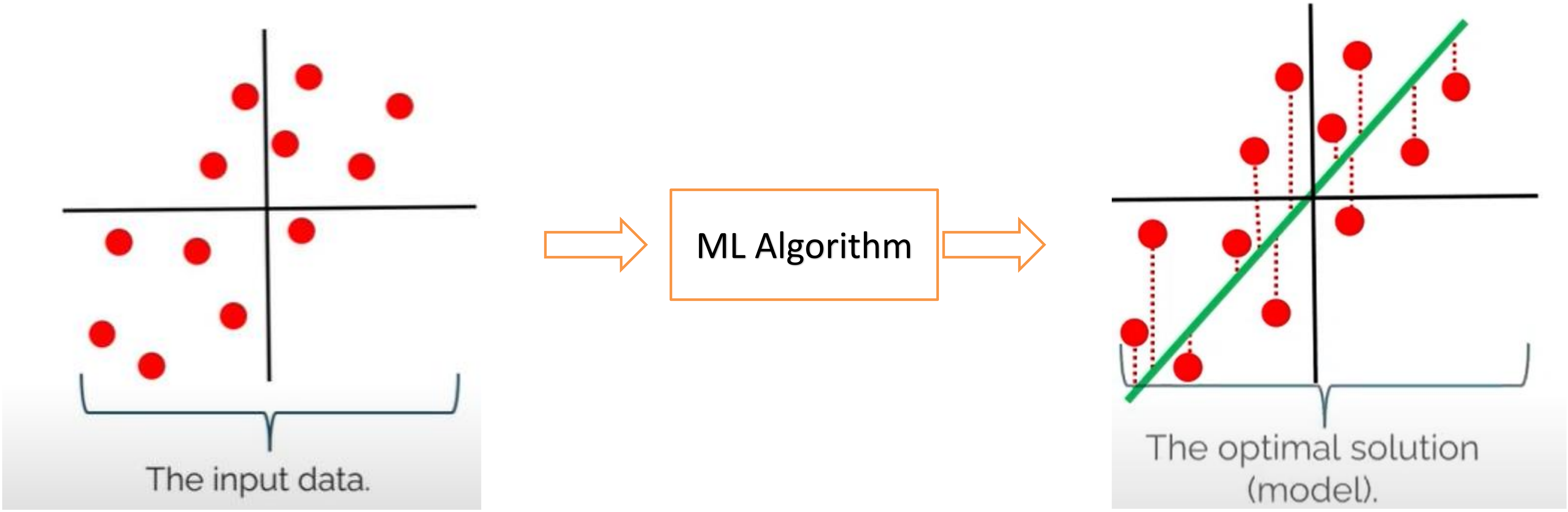


Motivation: Data quality



I.2: What is Coreset?

A general Machine Learning Procedure

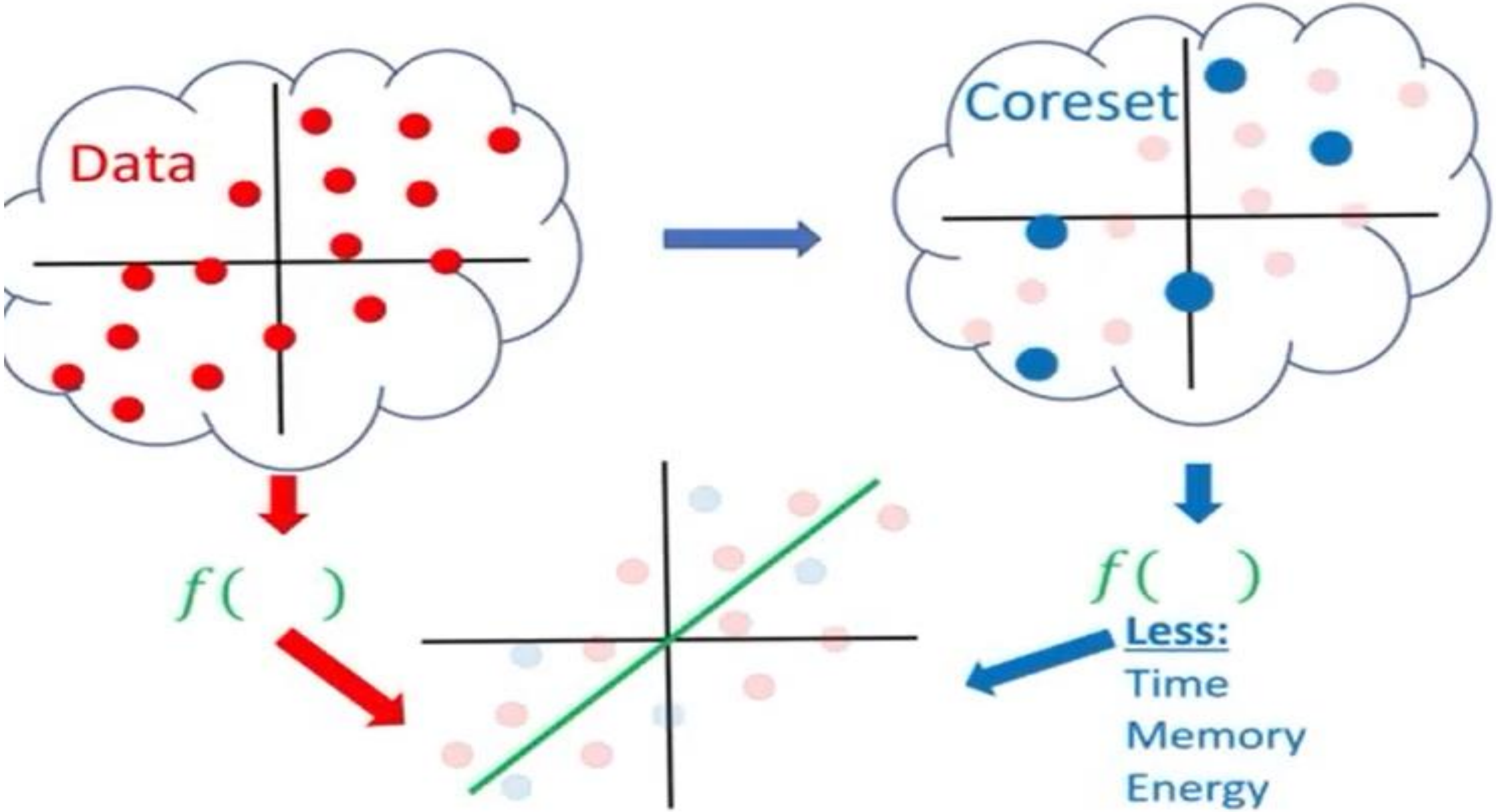


New approach is needed

Common approach: Design faster algorithms

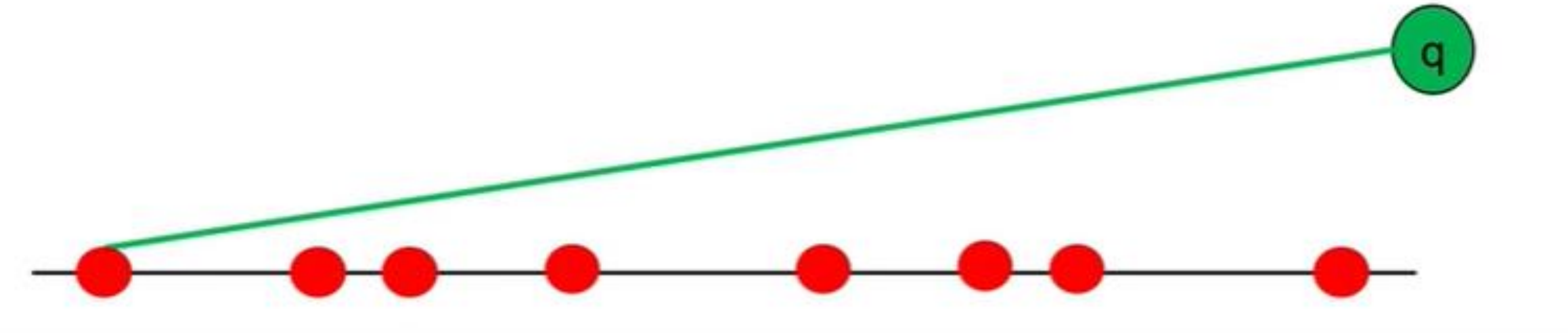
Alternative approach: Make the data smaller

A Coreset



Warm-up

Give a set P of n point on a line, another point q

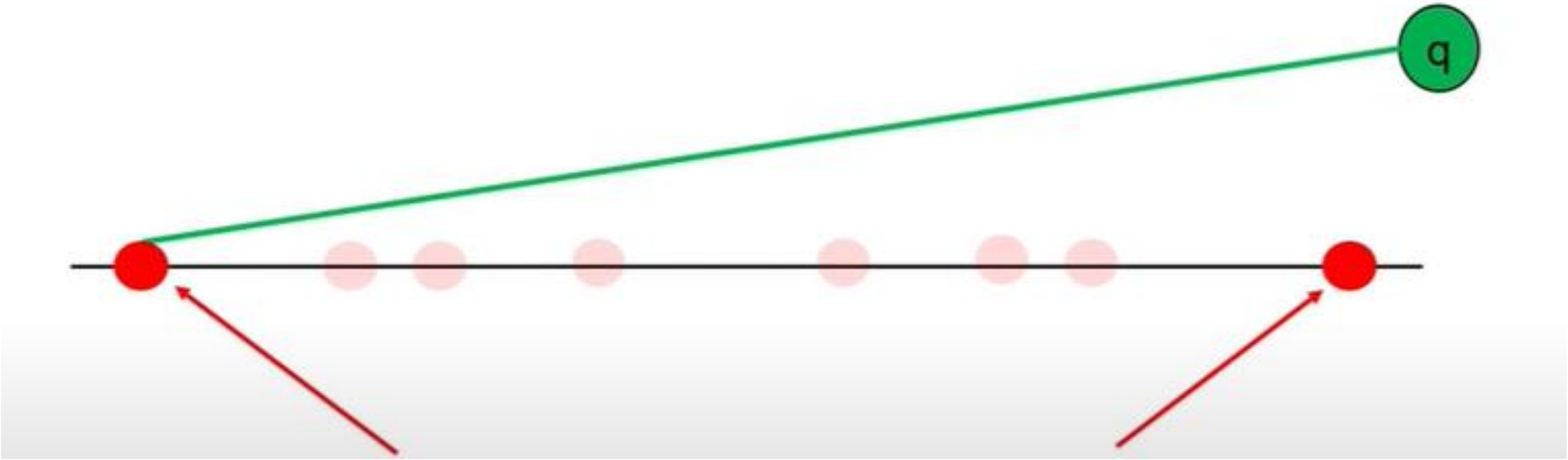


Goal:

Return the farthest point in P from q in O(1) time

Warm-up

Give a set P of n point on a line, another point q



The farthest point is one of the two end points
=> Delete all others points

Defining a problem

We need to define:

The input set (denote by P)

The query set (denote by X)

The cost function: $cost: P \times X \to [0, \infty)$ we need to optimize

Defining a problem

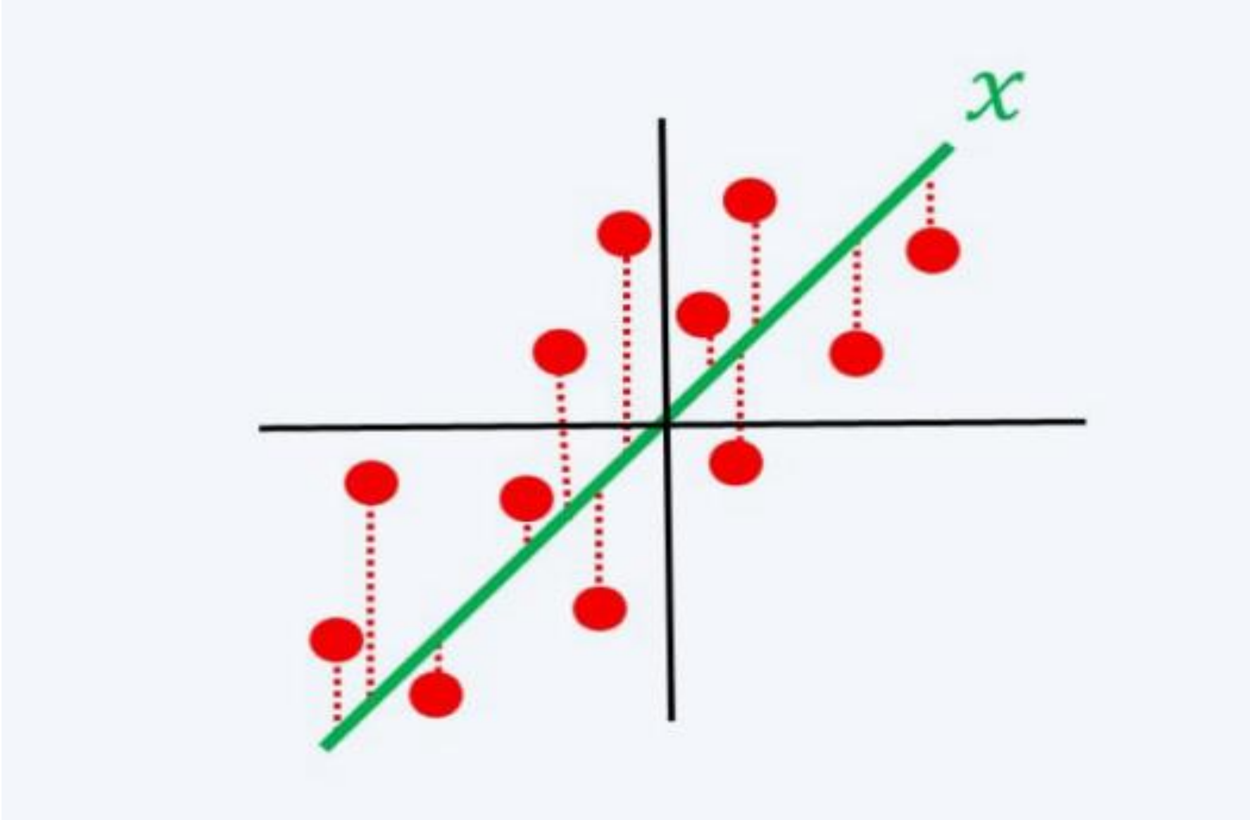
Ex: Linear regression

The input set: P of n points in R^d

and their label $y\colon P \to R$

The query set X is composed of every vector in R^d

The cost function to optimize: $\sum_{p \in P} \left(p^T x - y(p)\right)^2$



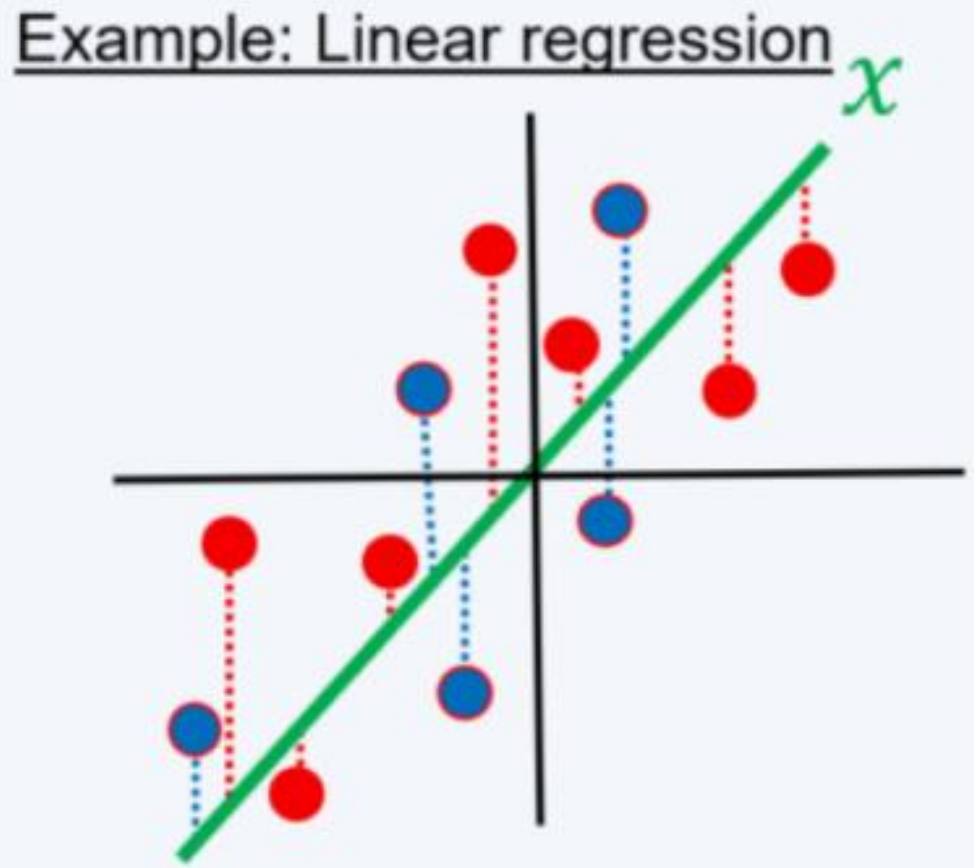
Defining a coreset

Let P be an input set

Let X be the query set

Let $cost: P \times X \rightarrow [0, \infty)$
be a cost function

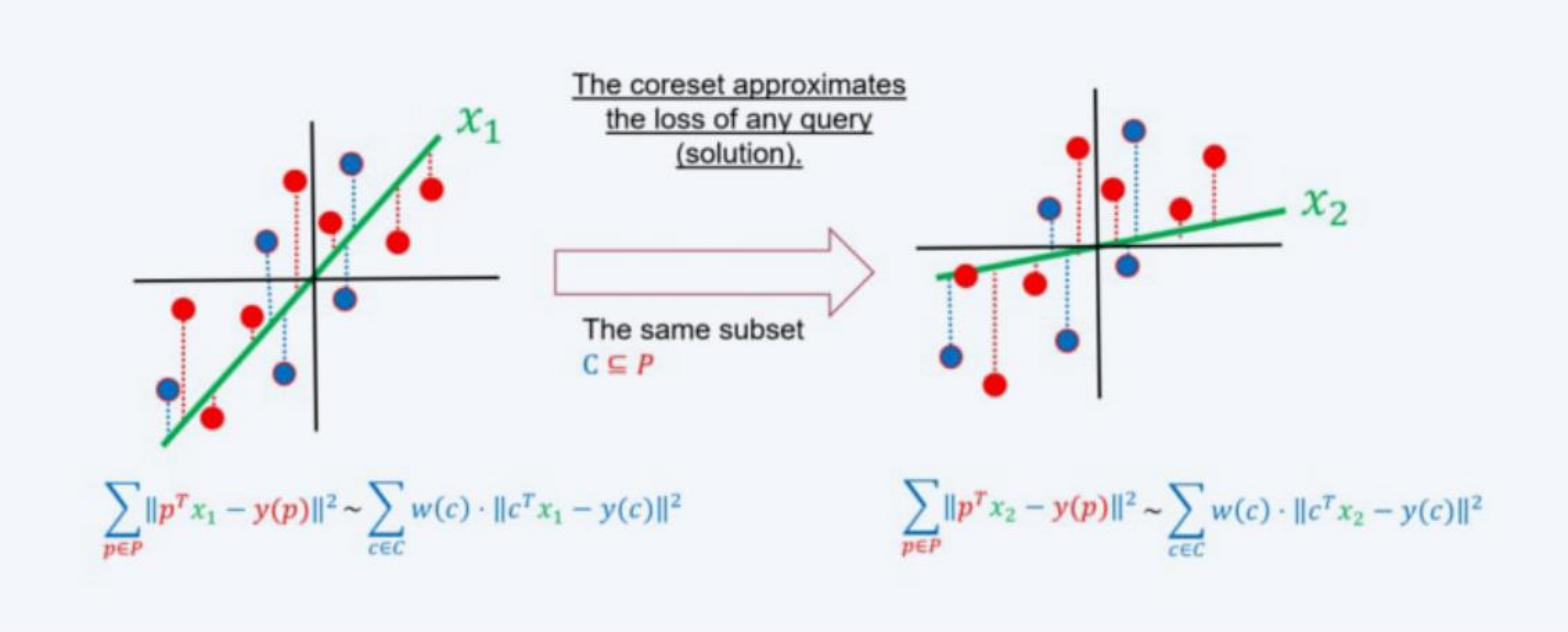
A coreset is pair of (C, w)
where $C \subset P$ and w: C -> R
such that for every query
$x \in X$: cost(P, x) ≈ cost((C, w), x)



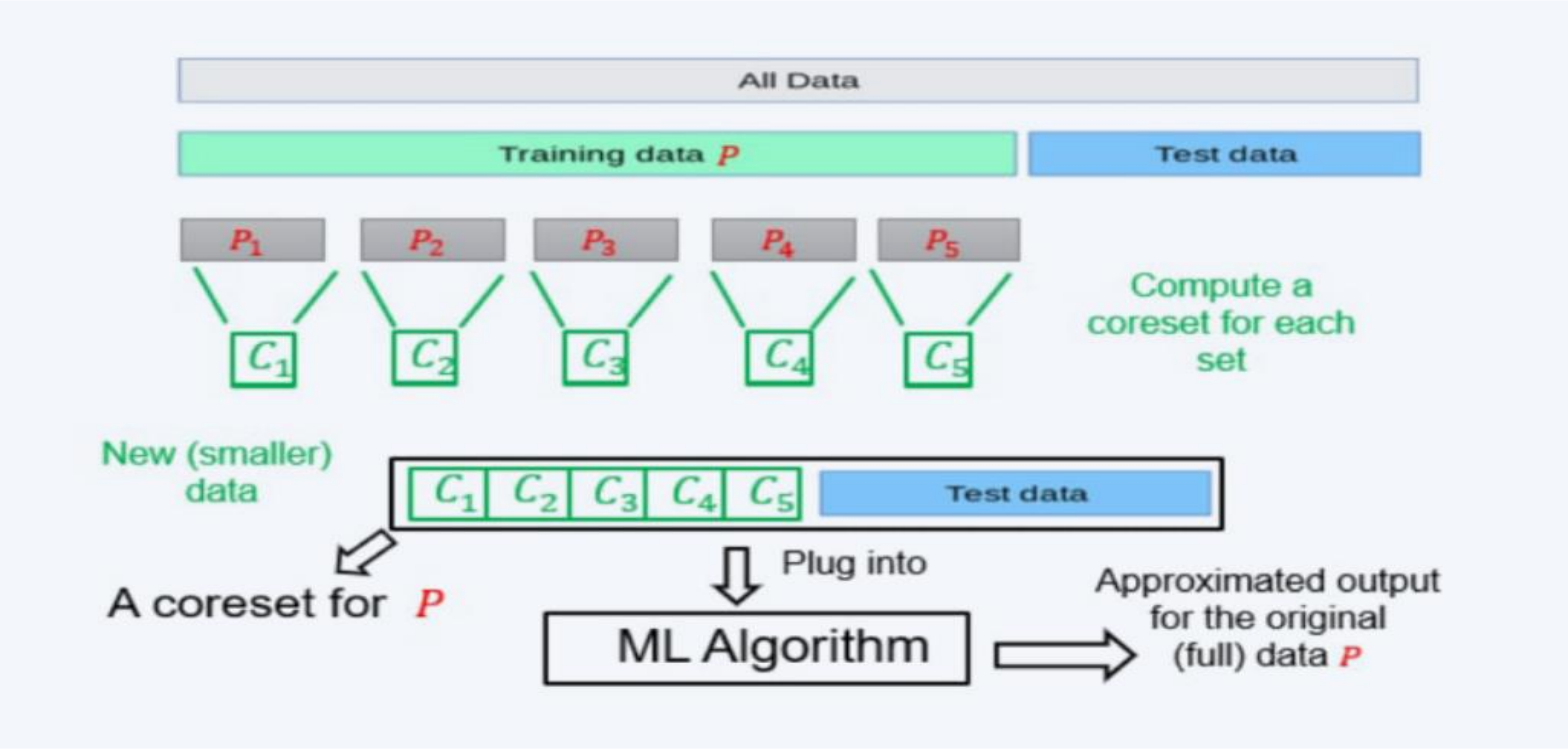
$$\sum_{p \in P} \|p^T x - y(p)\|^2 \sim \sum_{c \in C} w(c) \cdot \|c^T x - y(c)\|^2$$

I.3: Coreset properties

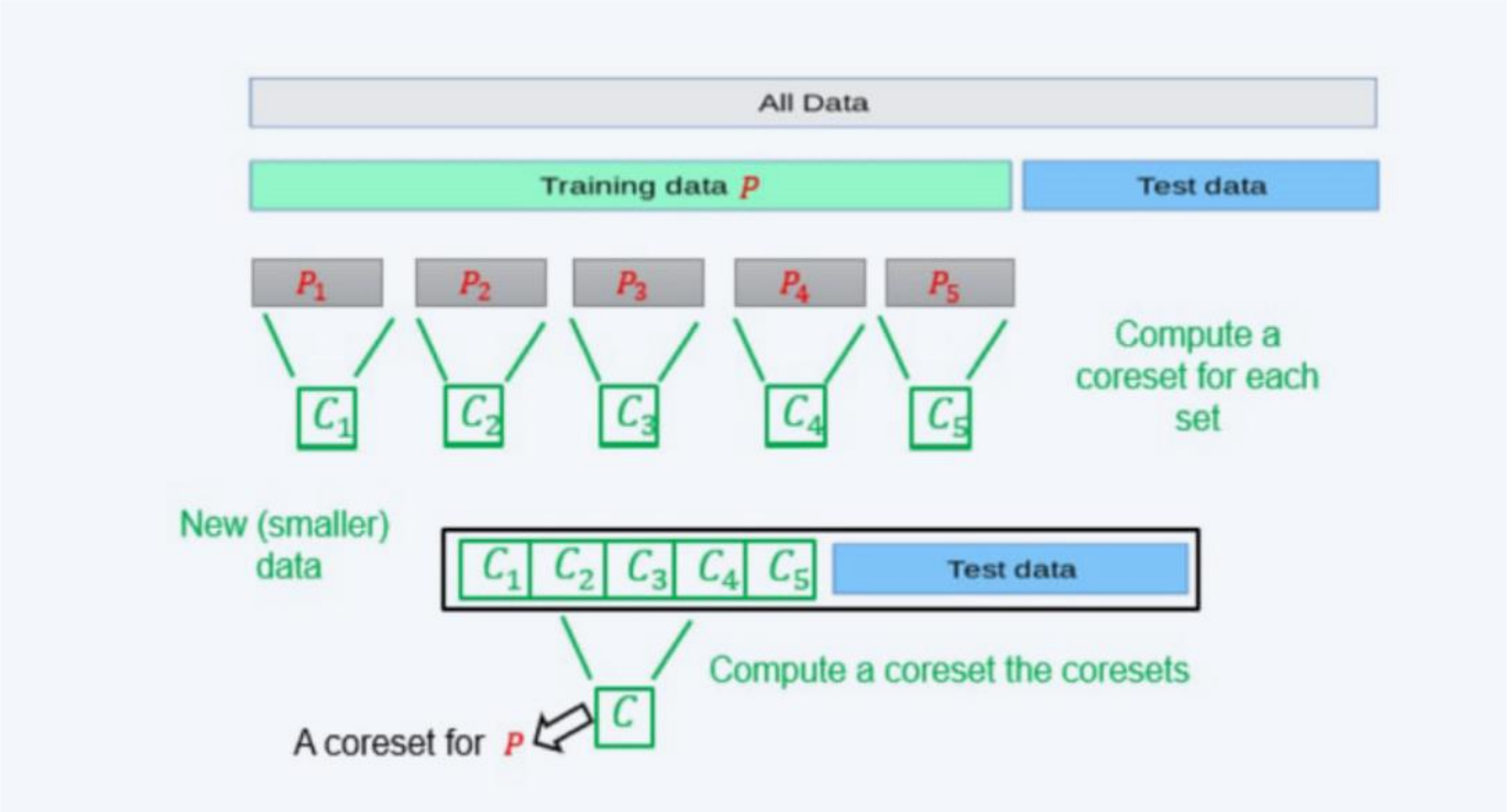
Coreset approximate any query (solution)



Union of Coreset is Coreset



Coreset of Coreset is Coreset



II: How to compute a Coreset?

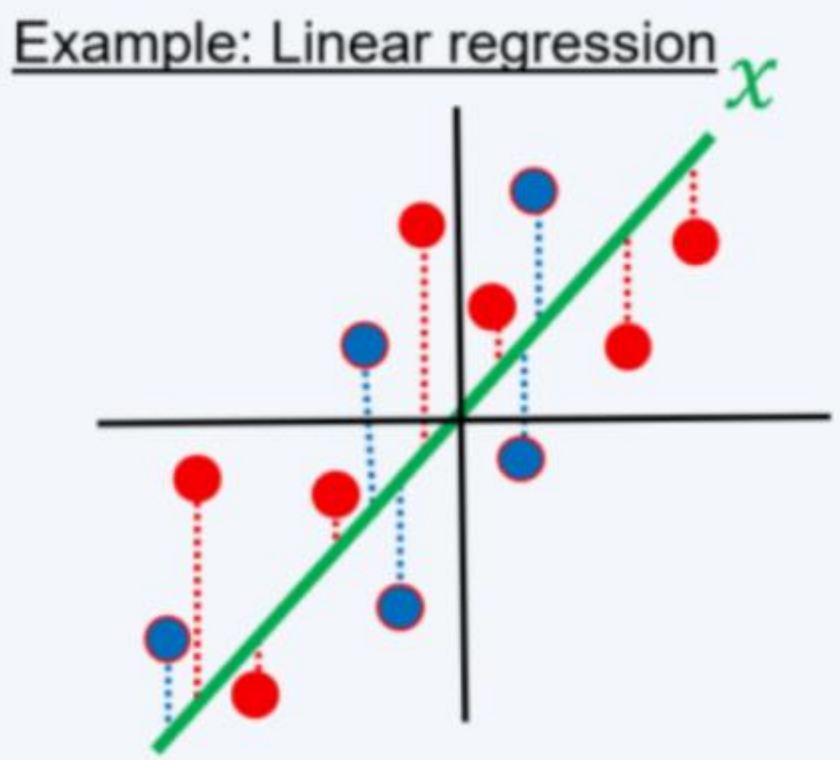
Computing a Coreset

- Let P be an input set.
- Let X be the query set.
- Let $cost: P \times X \rightarrow [0, \infty)$ be a cost function.

For every p ∈ P let s(p) ∈ [0, 1] be a number defining its importance/sensitivity.

1. Set $t = \sum_{p \in P} s(p)$
2. C := Sample m (m > 1) points according to s(p) / t
3. For every p ∈ C: $w(p) = \frac{t}{s(p)|C|}$.

With high probability (C, w) is a coreset for (P, X, cost), C ⊂ P, w: C -> R such that for every query x ∈ X: cost(P, x) ≈ cost((C, w), x)



$$\sum_{p \in P} \|p^T x - y(p)\|^2 \sim \sum_{c \in C} w(c) \cdot \|c^T x - y(c)\|^2$$

How to compute the importance s(p)?

- For every $p \in P$:

$$s(p) \geq \sup_{x \in X} \frac{cost(p,x)}{cost(P,x)}.$$

- To explain the definition of sensitivity/importance let's look at the following two cases, when $s(p)$ is high and when $s(p)$ is low and see the difference between them.

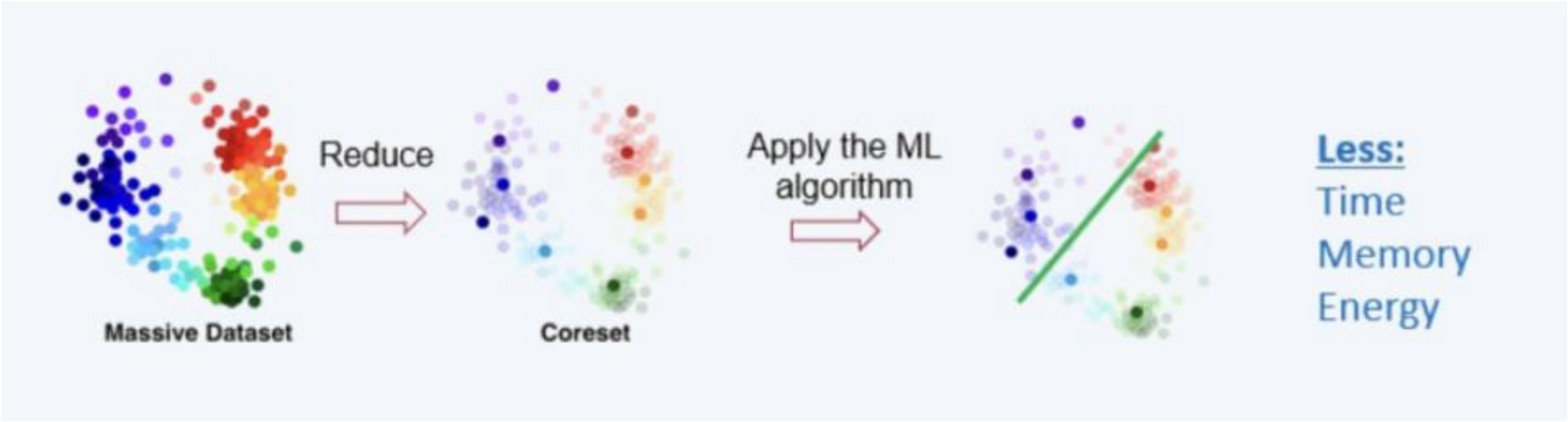
If $s(p)$ is high	*If $s(p)$ is low*
There exists a solution $x \in X$, such that P affects the loss too much (P is important).	For every solution $x \in X$, P does not affect the loss, hence, it is not important.
$\exists x \in X$: $cost(p,x)$ affects $cost(P,x)$.	$\forall x \in X$: $cost(p,x)$ **does not affect** $cost(P,x)$.

III: Applications

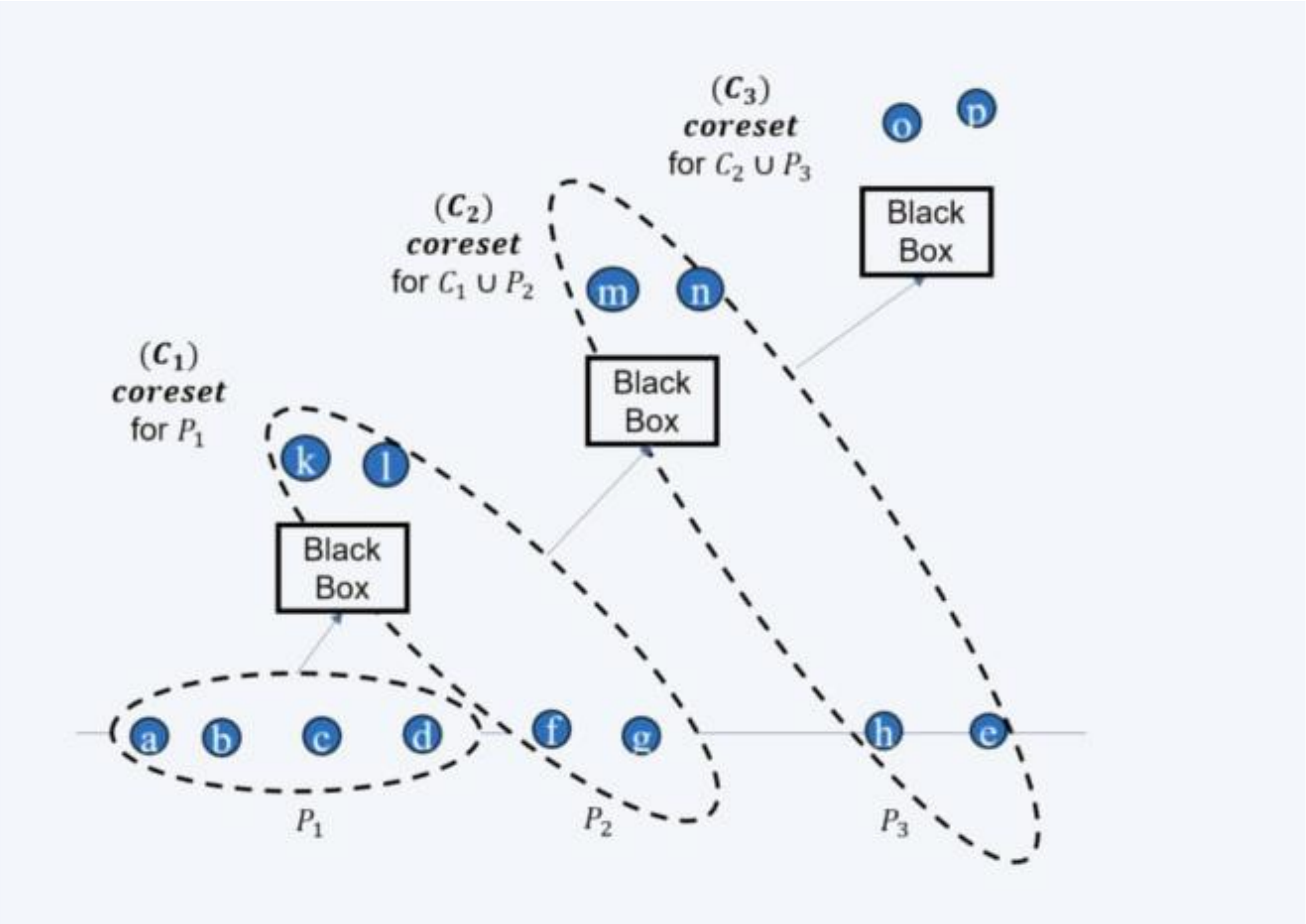
Limited hardware/time

Limited hardware
- Computation: IoT
- Energy: smartphones,

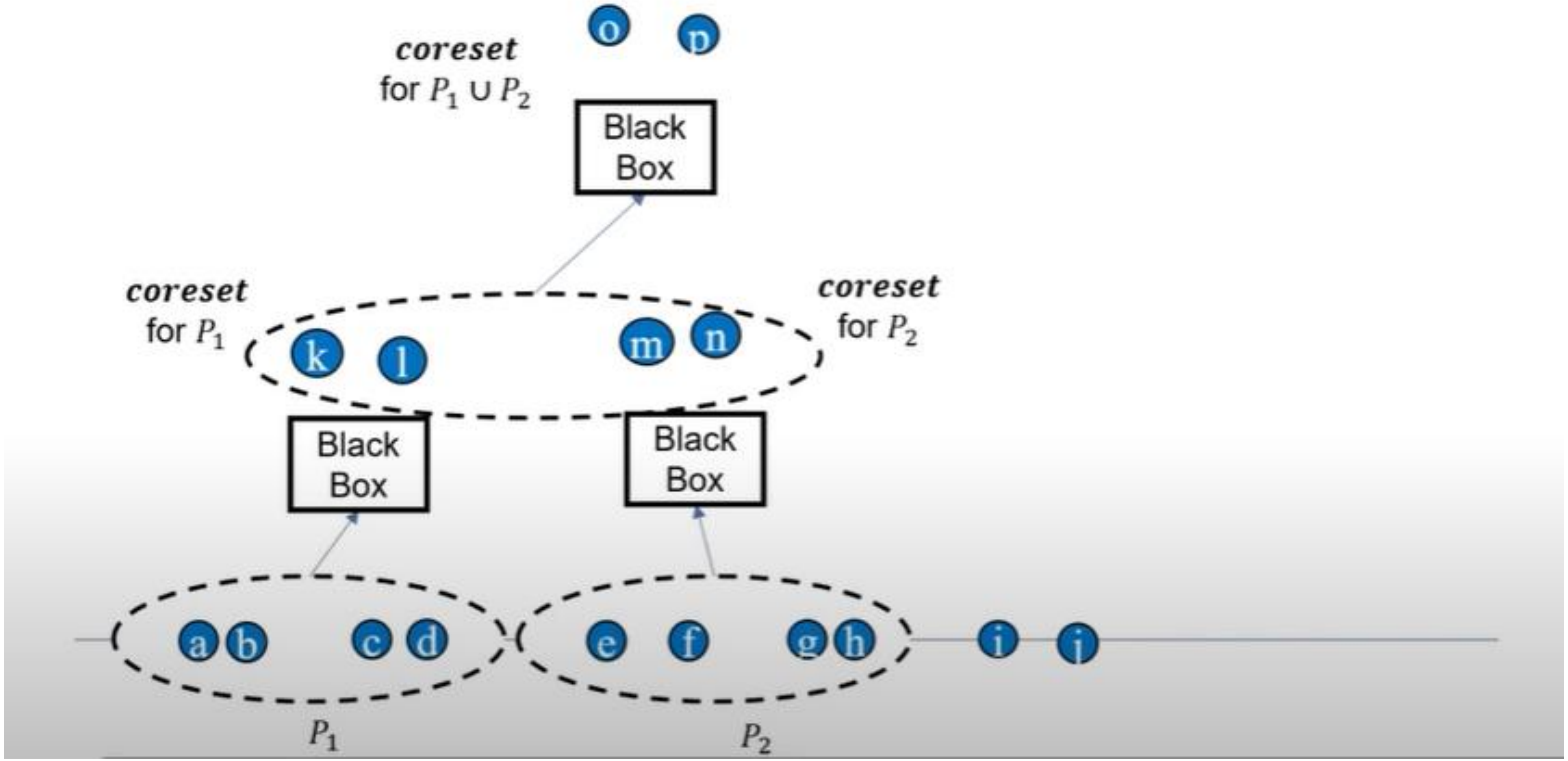
Limited time
- Real-time decision making



Streaming Coresets



Streaming tree + Distributed data



High-Quality Data