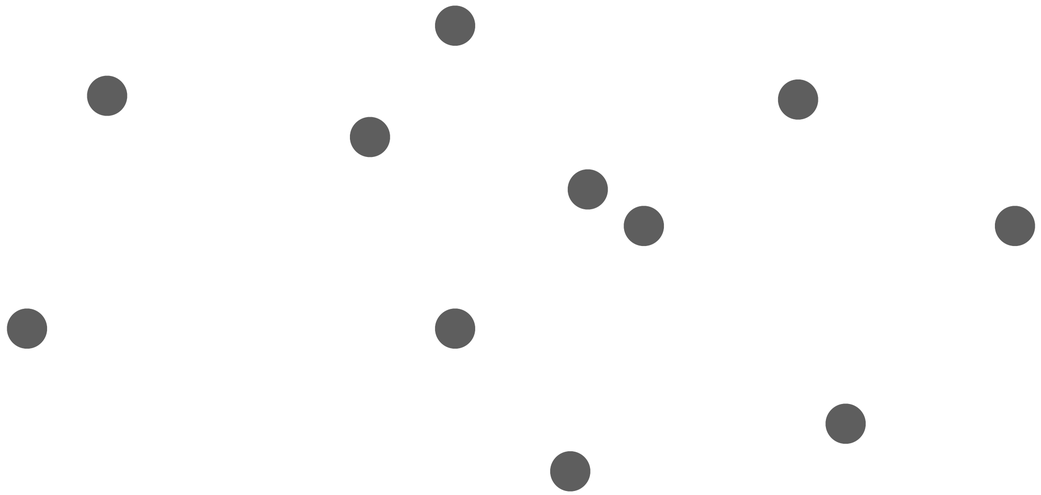
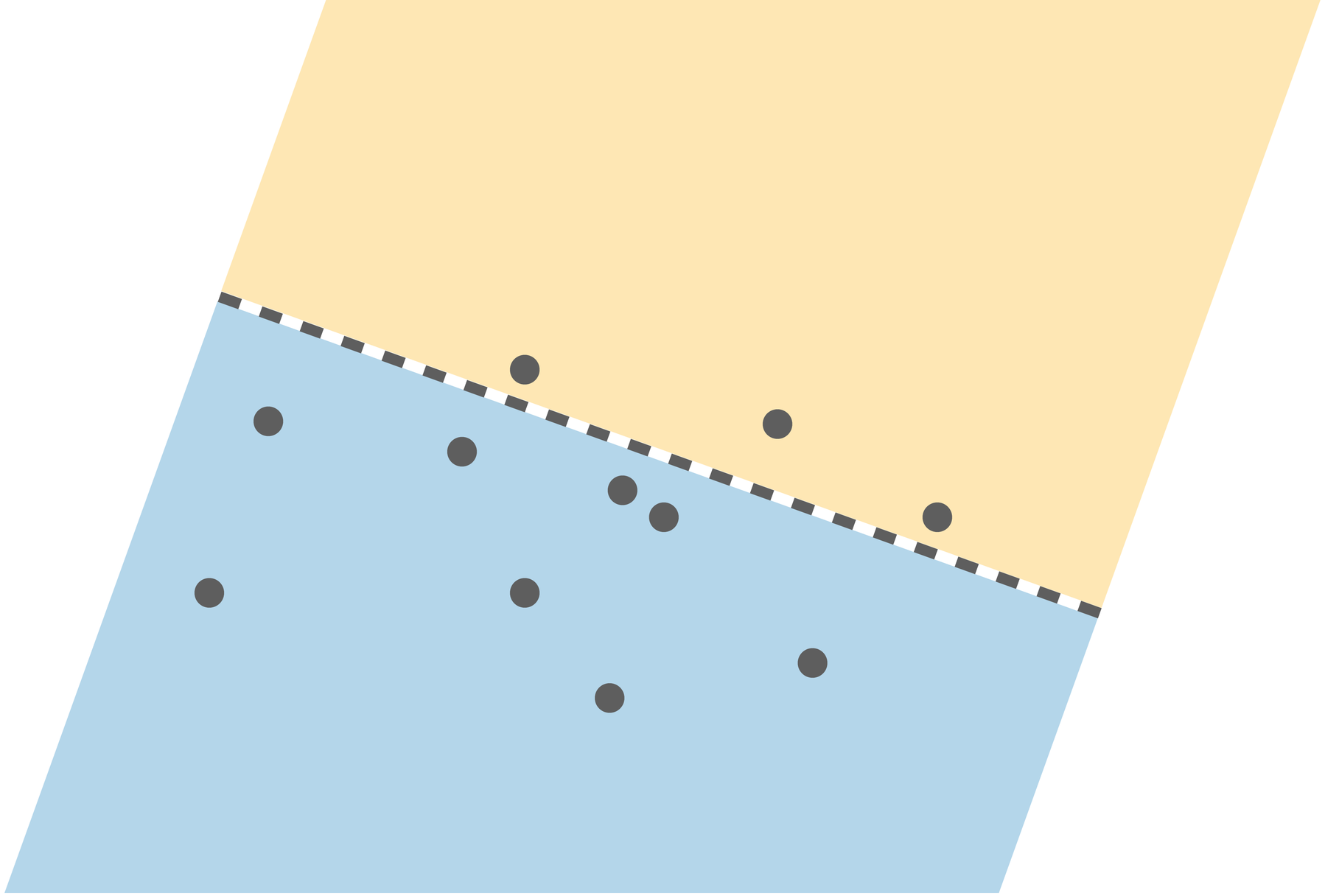
Active Learning Halfspaces without Synthetic Data

Hadley Black, Kasper Green Larsen, Arya Mazumdar, Barna Saha, and Geelon So

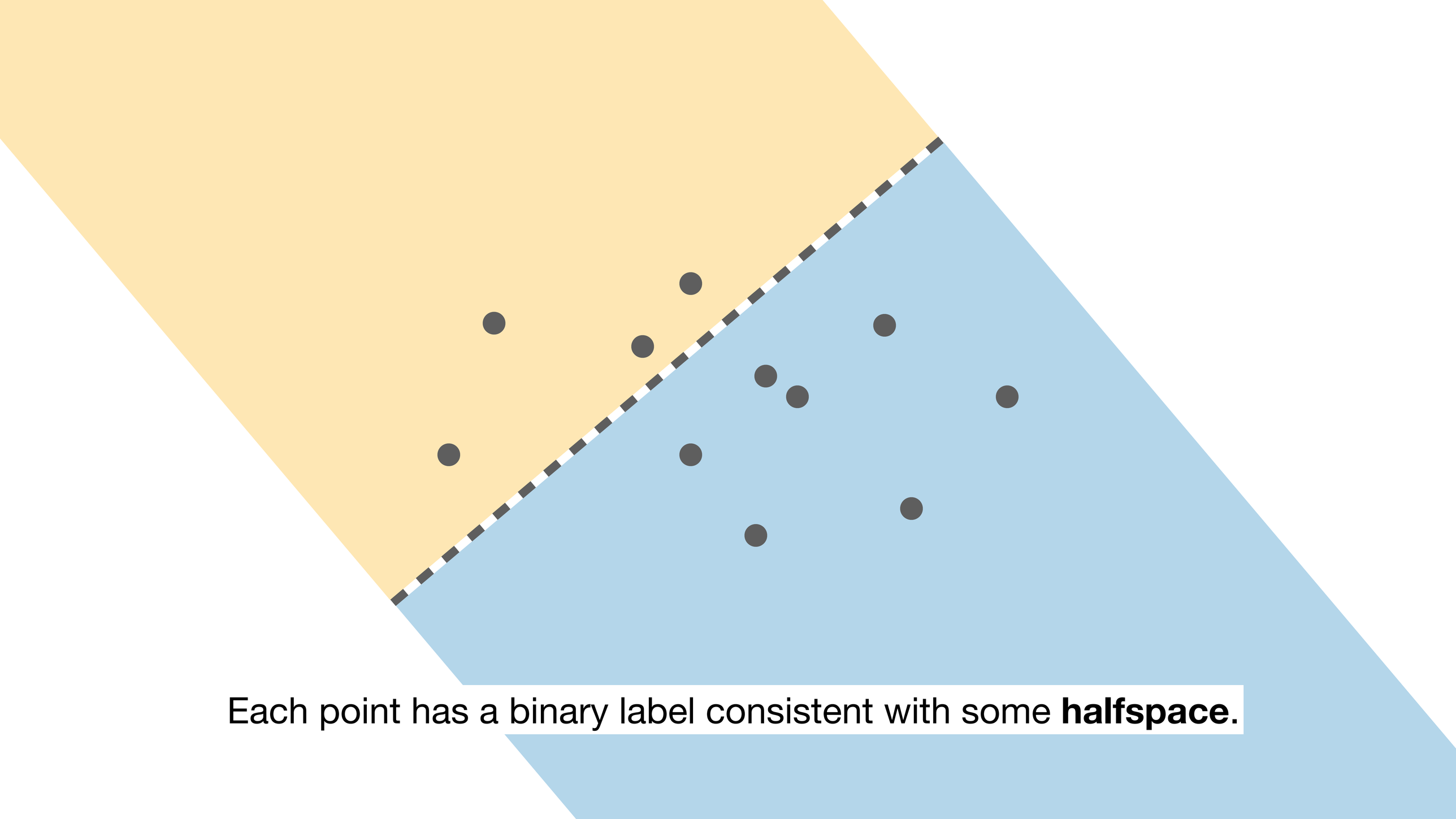
Problem



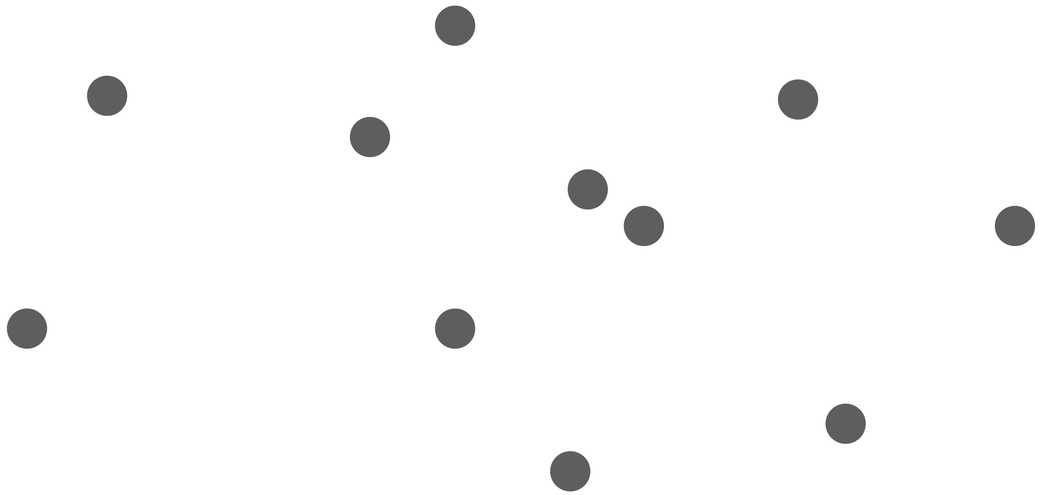
Let $X \subset \mathbb{R}^d$ be an **arbitrary** point set of size n.



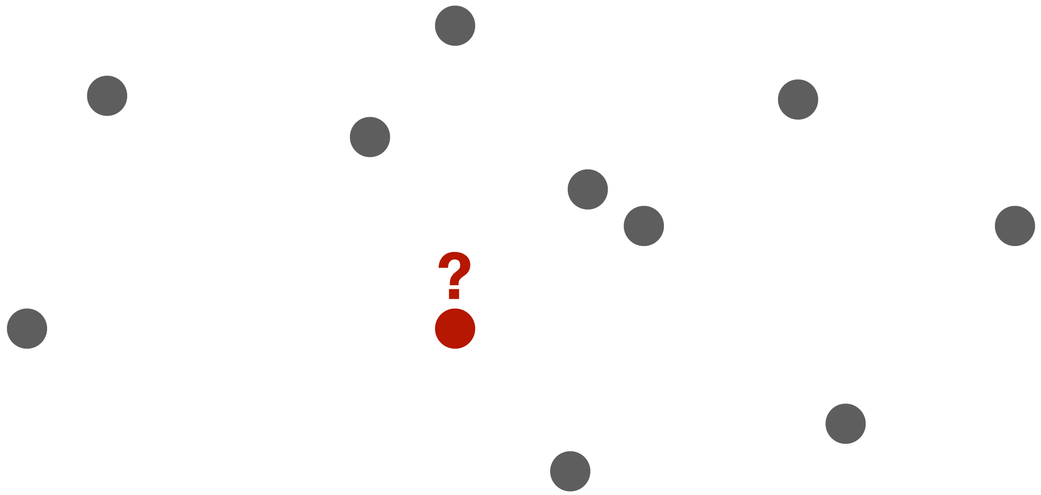
Each point has a binary label consistent with some **halfspace**.



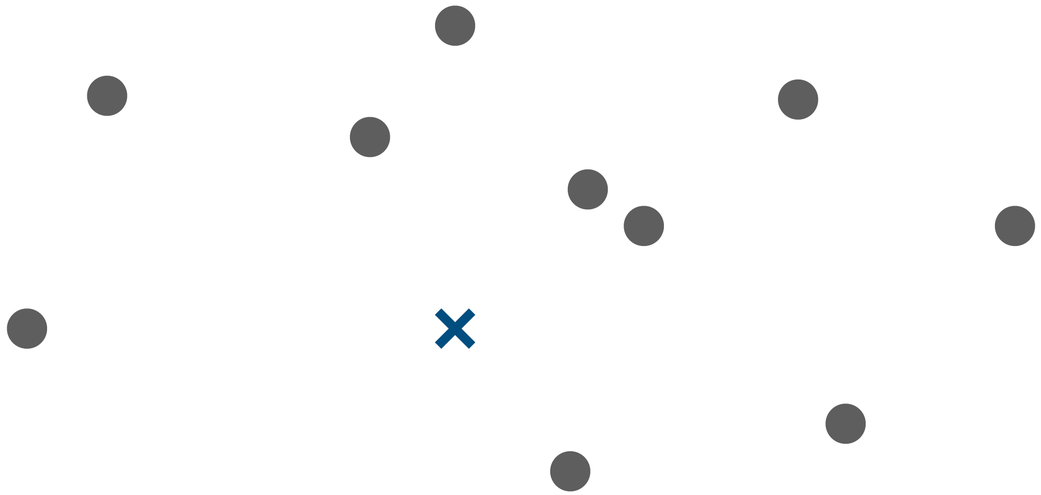
Each point has a binary label consistent with some **halfspace**.



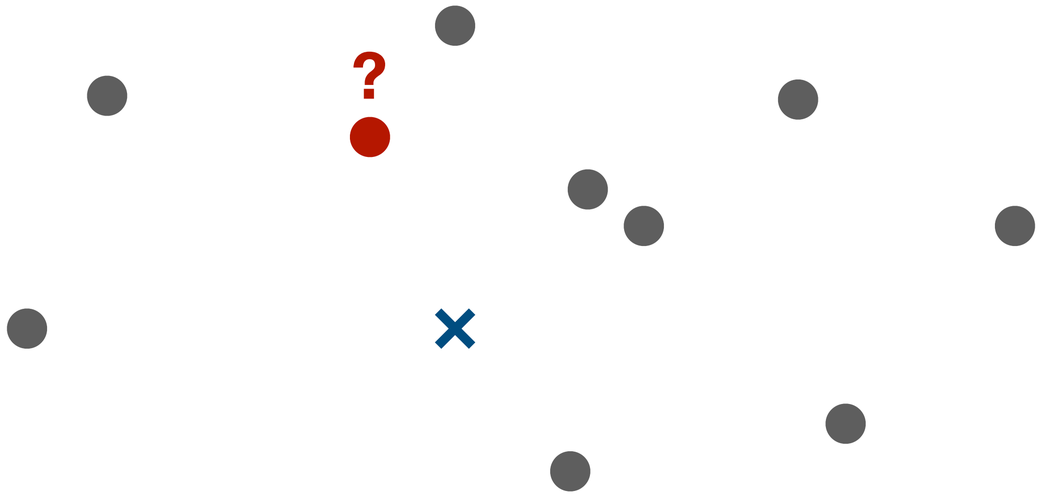
We do not initially know the halfspace…



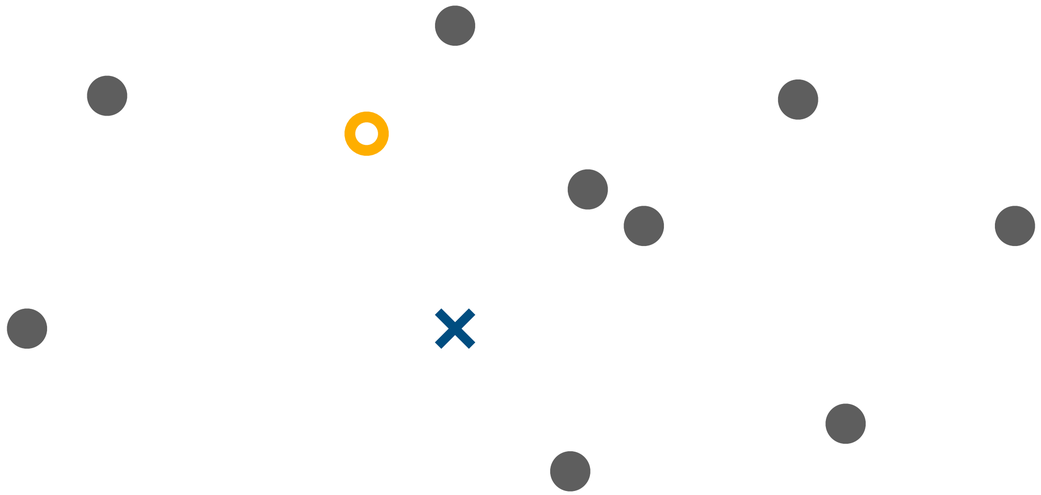
We do not initially know the halfspace, but we can make **label queries**.



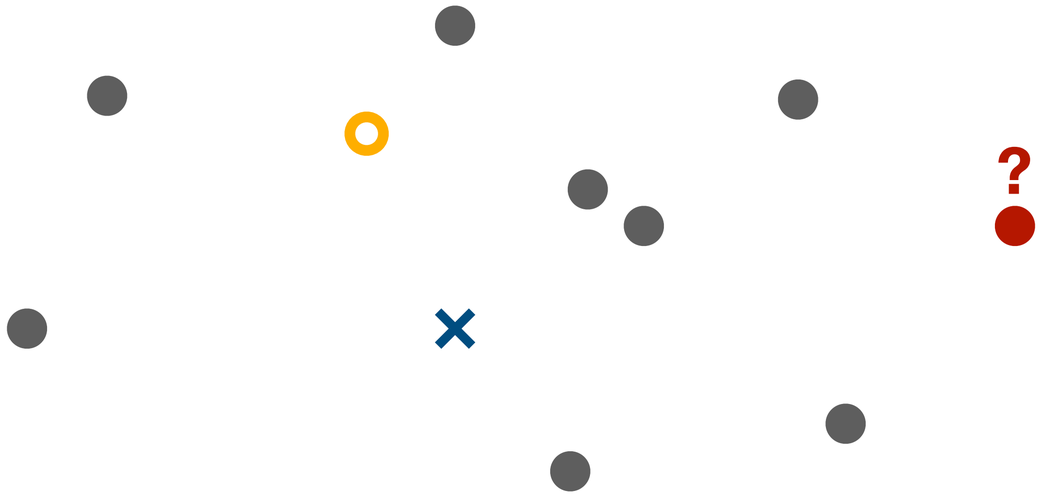
We do not initially know the halfspace, but we can make **label queries**.



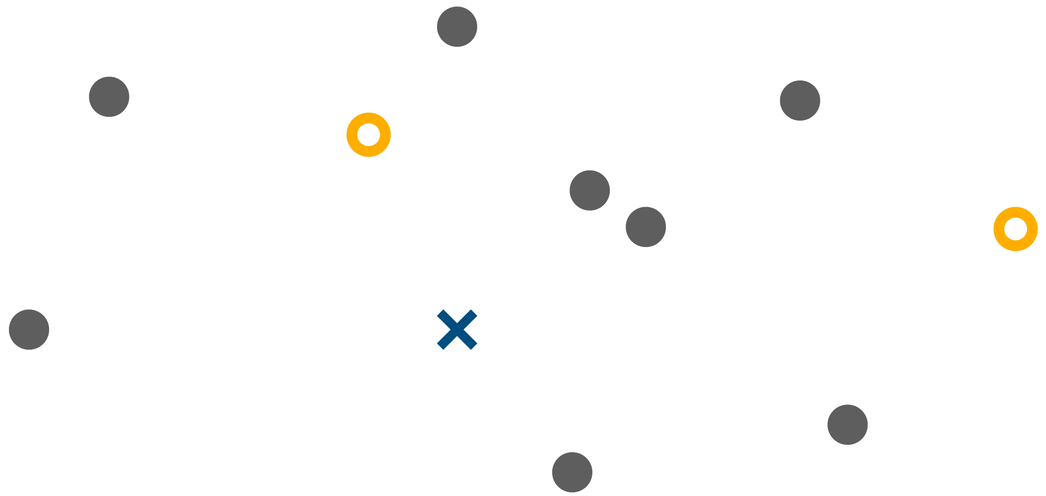
We do not initially know the halfspace, but we can make **label queries**.



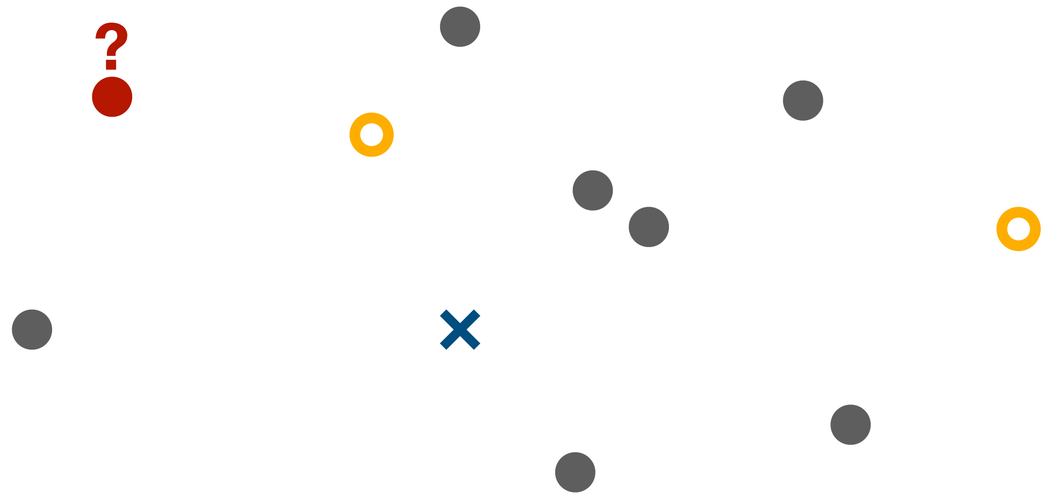
We do not initially know the halfspace, but we can make **label queries**.



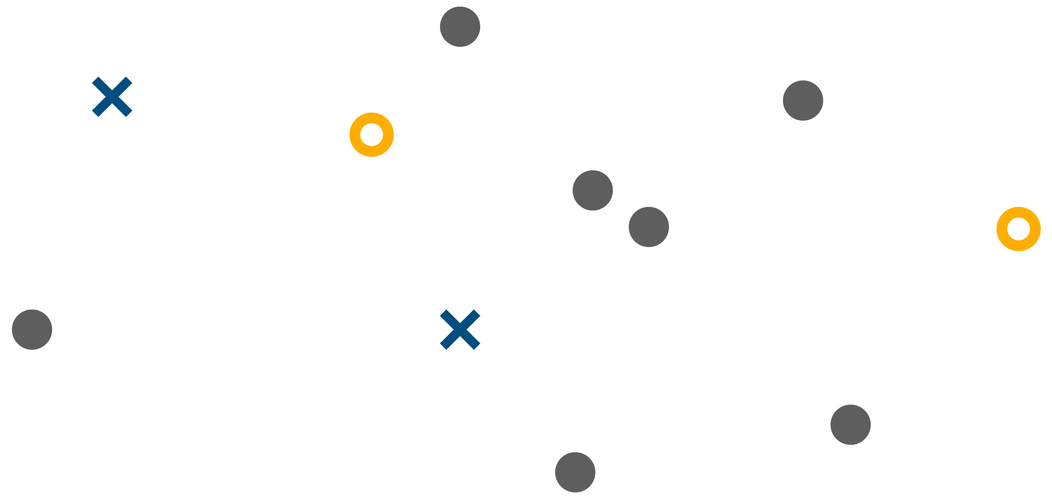
We do not initially know the halfspace, but we can make **label queries**.



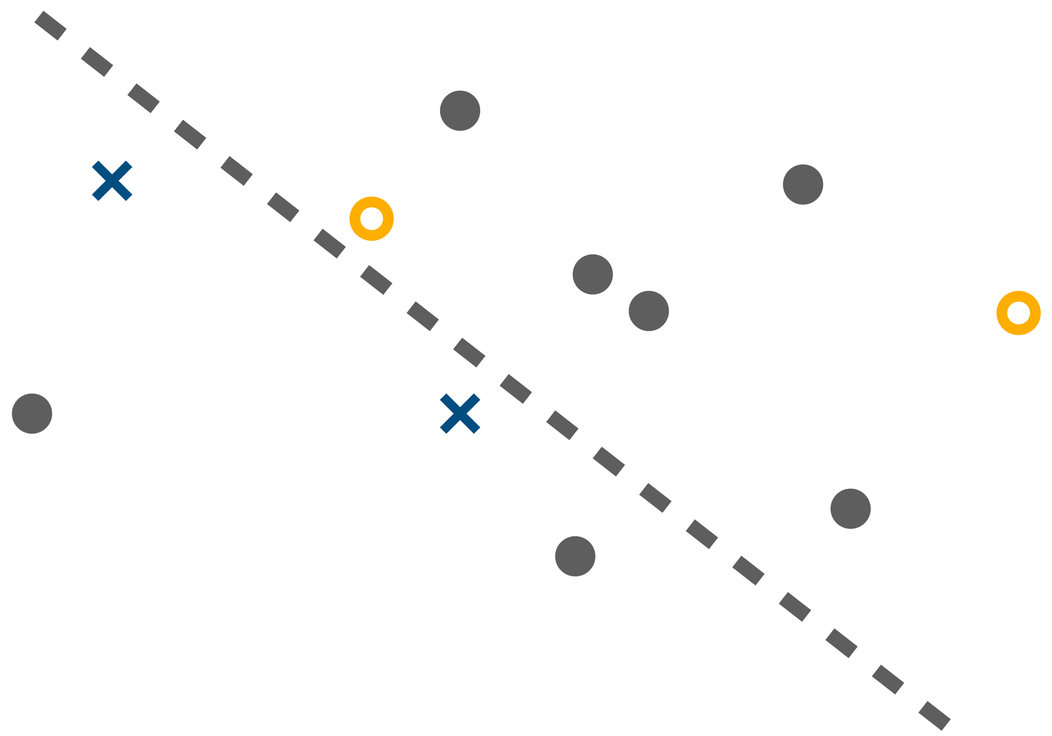
We do not initially know the halfspace, but we can make **label queries**.



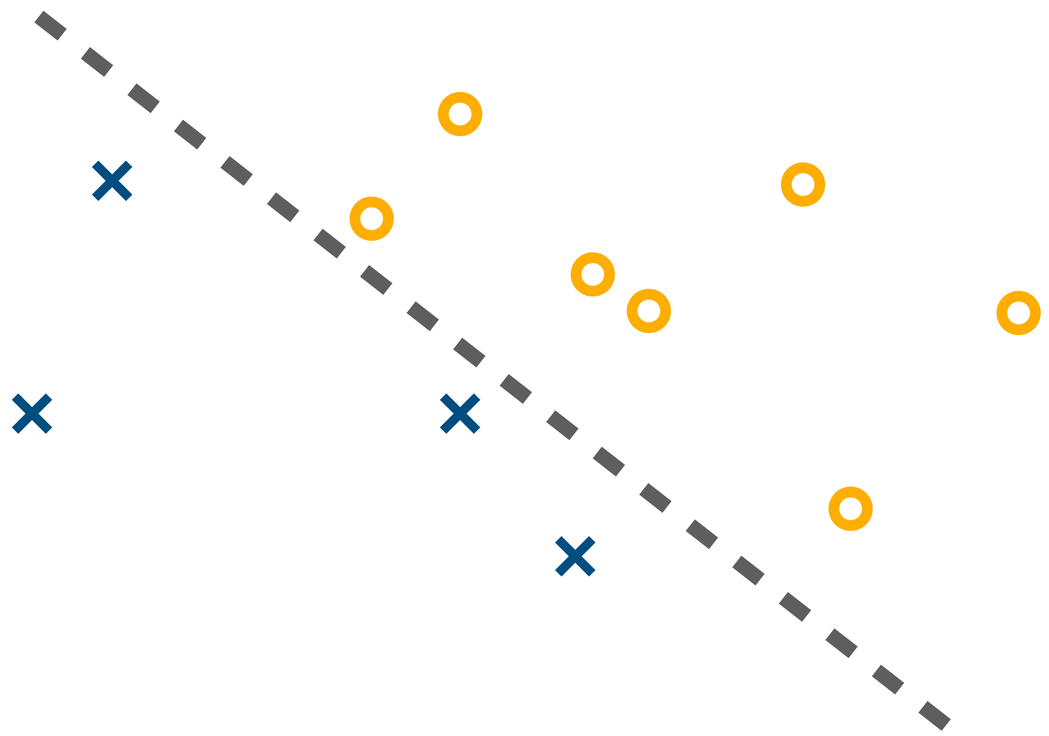
We do not initially know the halfspace, but we can make **label queries**.



We do not initially know the halfspace, but we can make **label queries**.



At this point, we can infer the rest of the labels.



Goal: label all points in X using as few queries as possible.

Active Learning Halfspaces

- Let $X \subset \mathbb{R}^d$ be an **arbitrary point set** of size n.

Active Learning Halfspaces

- Let $X \subset \mathbb{R}^d$ be an **arbitrary point set** of size n.
- Each point $x_i \in X$ has an associated label y_i

$$y_i = \mathbf{1}\{\langle u, x_i \rangle > \tau\}.$$

Active Learning Halfspaces

- Let $X \subset \mathbb{R}^d$ be an **arbitrary point set** of size n.
- Each point $x_i \in X$ has an associated label y_i

$$y_i = \mathbf{1}\{\langle u, x_i \rangle > \tau\}.$$

  - The label is consistent with an unknown non-homogeneous **halfspace**.

Active Learning Halfspaces

- Let $X \subset \mathbb{R}^d$ be an **arbitrary point set** of size n.
- Each point $x_i \in X$ has an associated label y_i

$$y_i = \mathbf{1}\{\langle u, x_i \rangle > \tau\}.$$

  - The label is consistent with an unknown non-homogeneous **halfspace**.
- We may make **label queries** to points only in X.

Active Learning Halfspaces

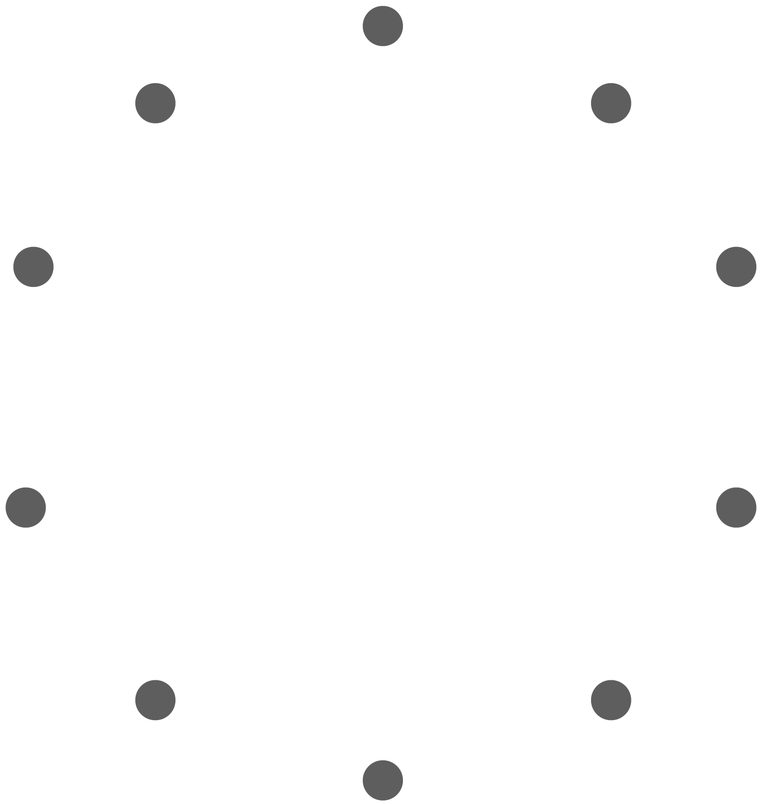
- Let $X \subset \mathbb{R}^d$ be an **arbitrary point set** of size n.
- Each point $x_i \in X$ has an associated label y_i

$$y_i = \mathbf{1}\{\langle u, x_i \rangle > \tau\}.$$

  - The label is consistent with an unknown non-homogeneous **halfspace**.
- We may make **label queries** to points only in X.
- **Goal:** label all of X using as few adaptively-chosen label queries as possible.

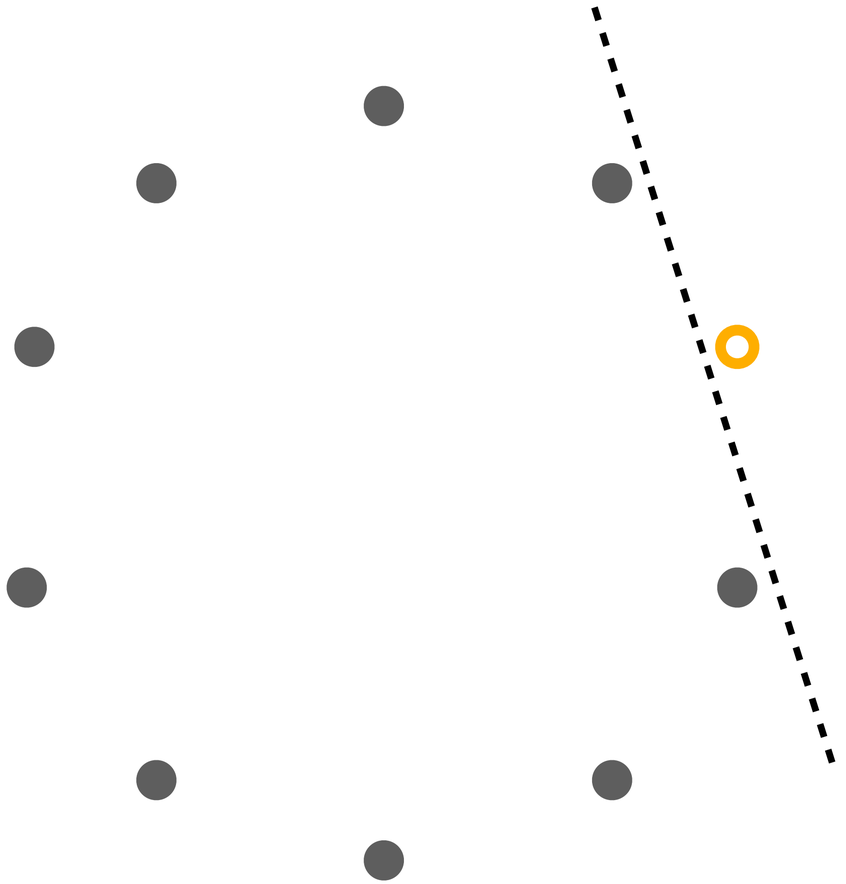
Prior Work: Lower Bound

An $\Omega(n)$ Lower Bound



In the worst-case, $\Omega(n)$ queries are needed even in $\mathbb{R}^2$ (Dasgupta 2004).

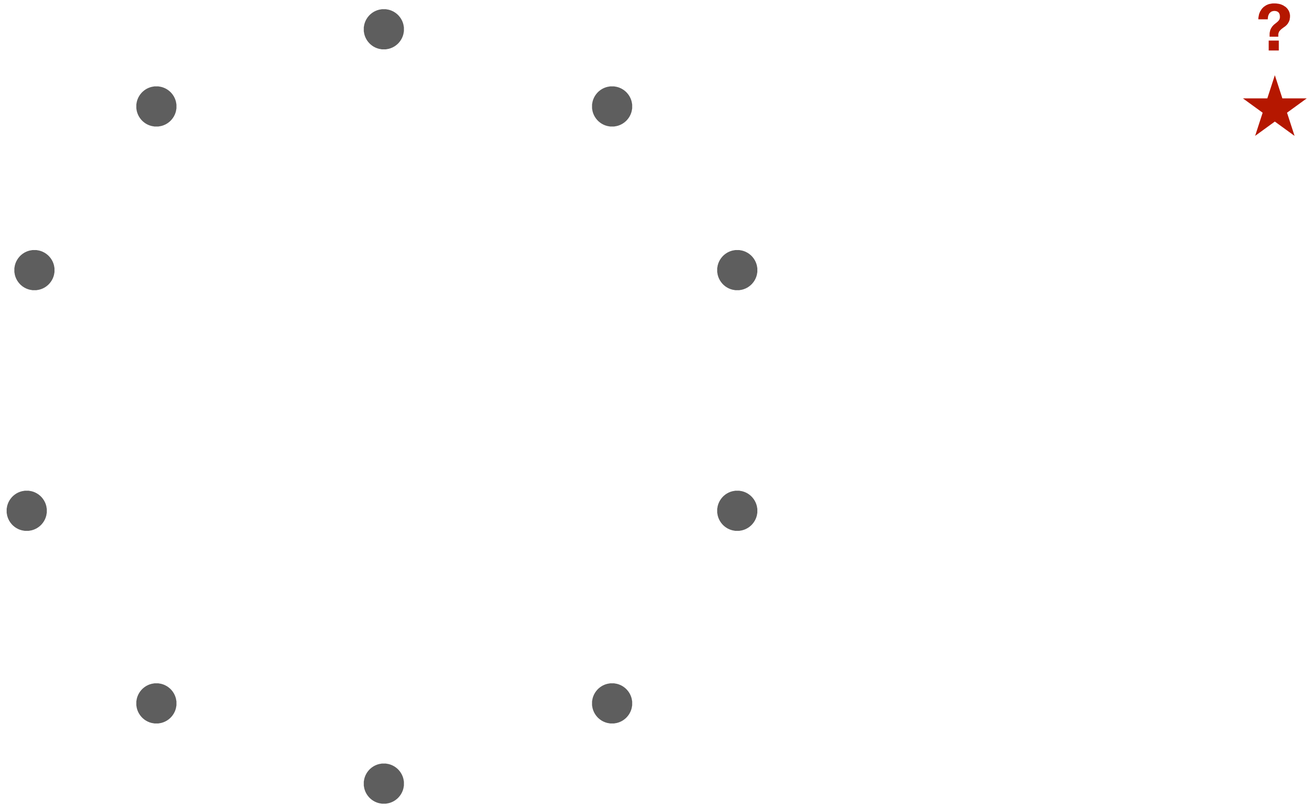
An $\Omega(n)$ Lower Bound



The halfspace can isolate a single point; we need to query exactly that point.

Prior Work: Circumventing the Lower Bound

Active Learning with Point Synthesis



The **membership query** oracle can label synthetic data outside of X.

Active Learning with Point Synthesis

- Information-theoretic lower bound of $\Omega(d \log n)$.

Active Learning with Point Synthesis

- Information-theoretic lower bound of $\Omega(d \log n)$.
- Hopkins, Kane, Lovett, & Mahajan (2020) give an active learning strategy with

$$O(d \log^2 d \log n)$$ membership queries.

Active Learning with Point Synthesis

- Information-theoretic lower bound of $\Omega(d \log n)$.
- Hopkins, Kane, Lovett, & Mahajan (2020) give an active learning strategy with

$$O(d \log^2 d \log n)$$ membership queries.

- Downside: synthetic data may not have a sensible label (Lang & Baum 1992).

Other Ways to Circumvent Lower Bound

- Distributional assumptions
  - (Dasgupta, Kalai, & Monteleoni 2009), (Balcan & Long 2013),…

Other Ways to Circumvent Lower Bound

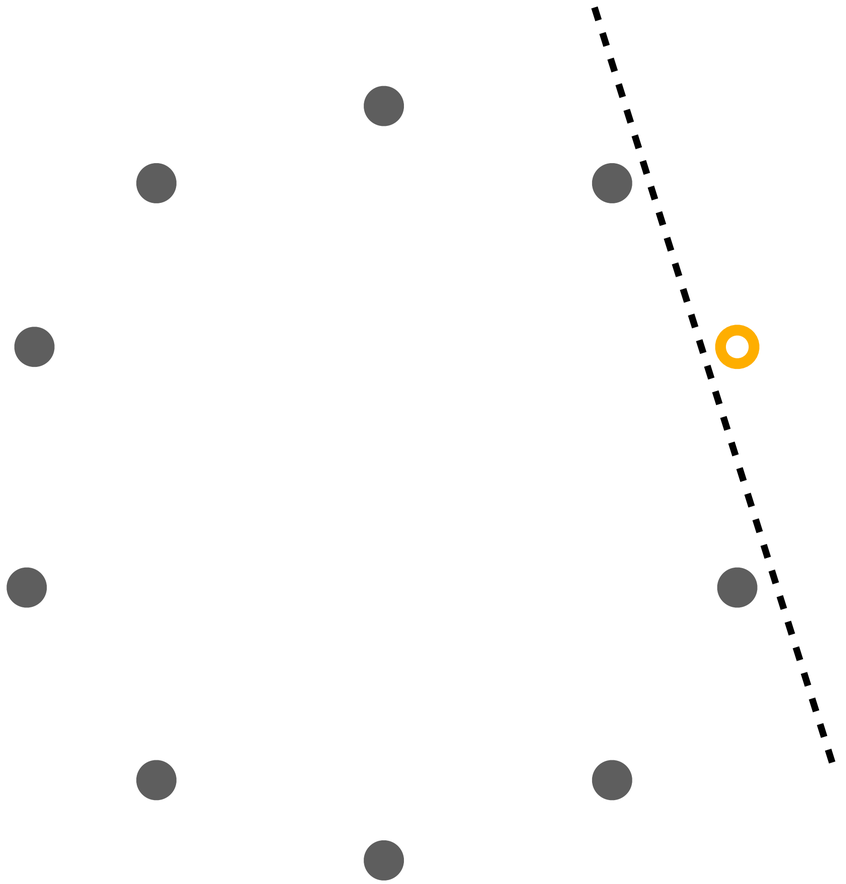
- Distributional assumptions
  - (Dasgupta, Kalai, & Monteleoni 2009), (Balcan & Long 2013),…
- Margin-based approaches
  - (Balcan, Broder, & Zhang 2007), (Gonen, Sabato, & Shalev-Shwartz 2013),…

Other Ways to Circumvent Lower Bound

- Distributional assumptions
  - (Dasgupta, Kalai, & Monteleoni 2009), (Balcan & Long 2013),…
- Margin-based approaches
  - (Balcan, Broder, & Zhang 2007), (Gonen, Sabato, & Shalev-Shwartz 2013),…
- **Structured subclass of halfspaces**
  - (This work)

This Work

Source of Hardness



Intuitively, there are n **directions** in this hard example.

Parametrize Hardness via Directionality

Parametrize Hardness via Directionality

Bounded directionality assumption

The normal vector of the halfspace is one out of D **known directions**

Parametrize Hardness via Directionality

Bounded directionality assumption

The normal vector of the halfspace is one out of D **known directions**

$$h^\star(x) \equiv \mathbf{1}\left\{\langle u^\star, x\rangle > \tau^\star\right\}$$

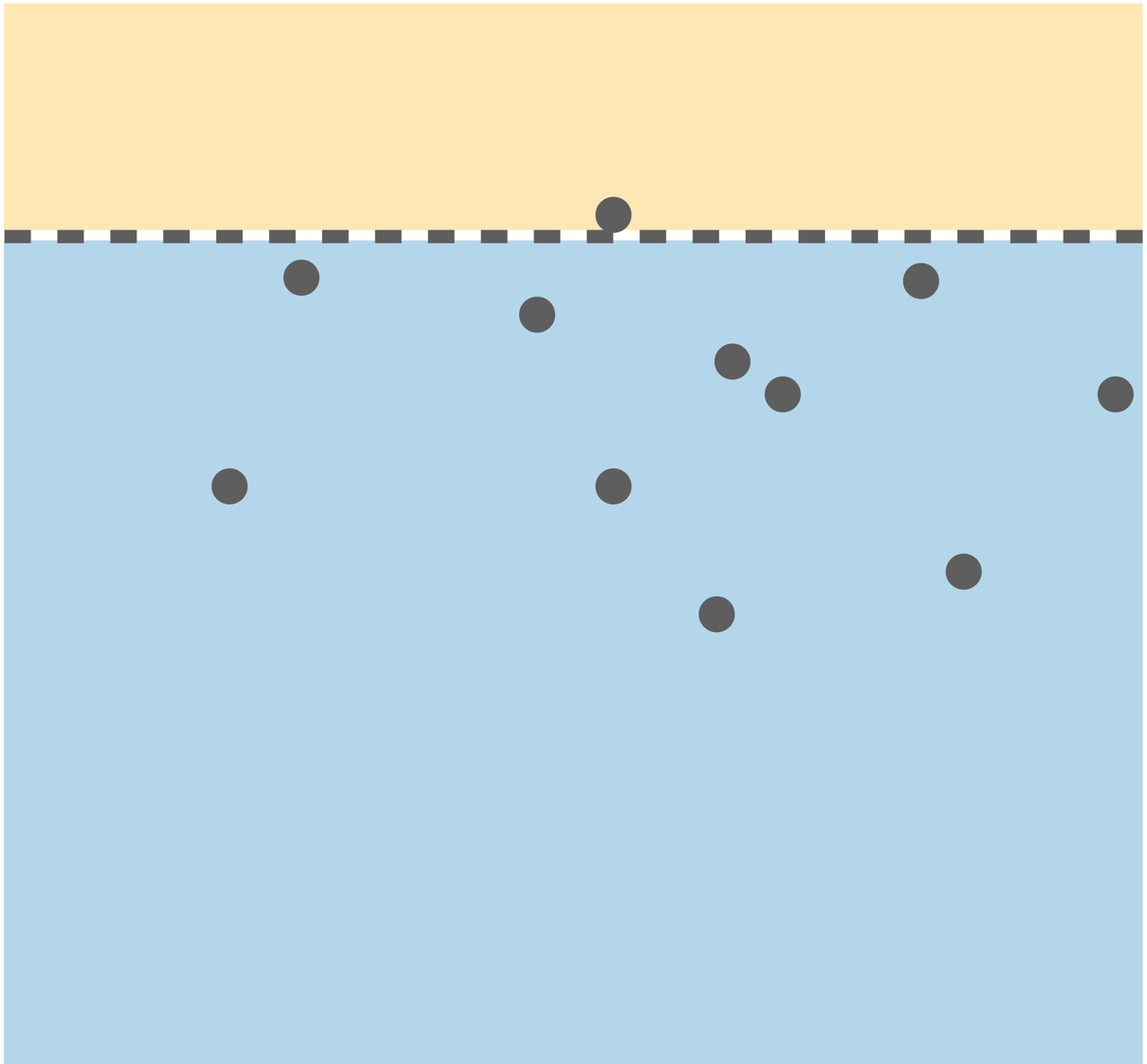
where $u^\star \in \{u_1, \ldots, u_D\}$.

Canonical Class: Axis-Aligned Halfspaces

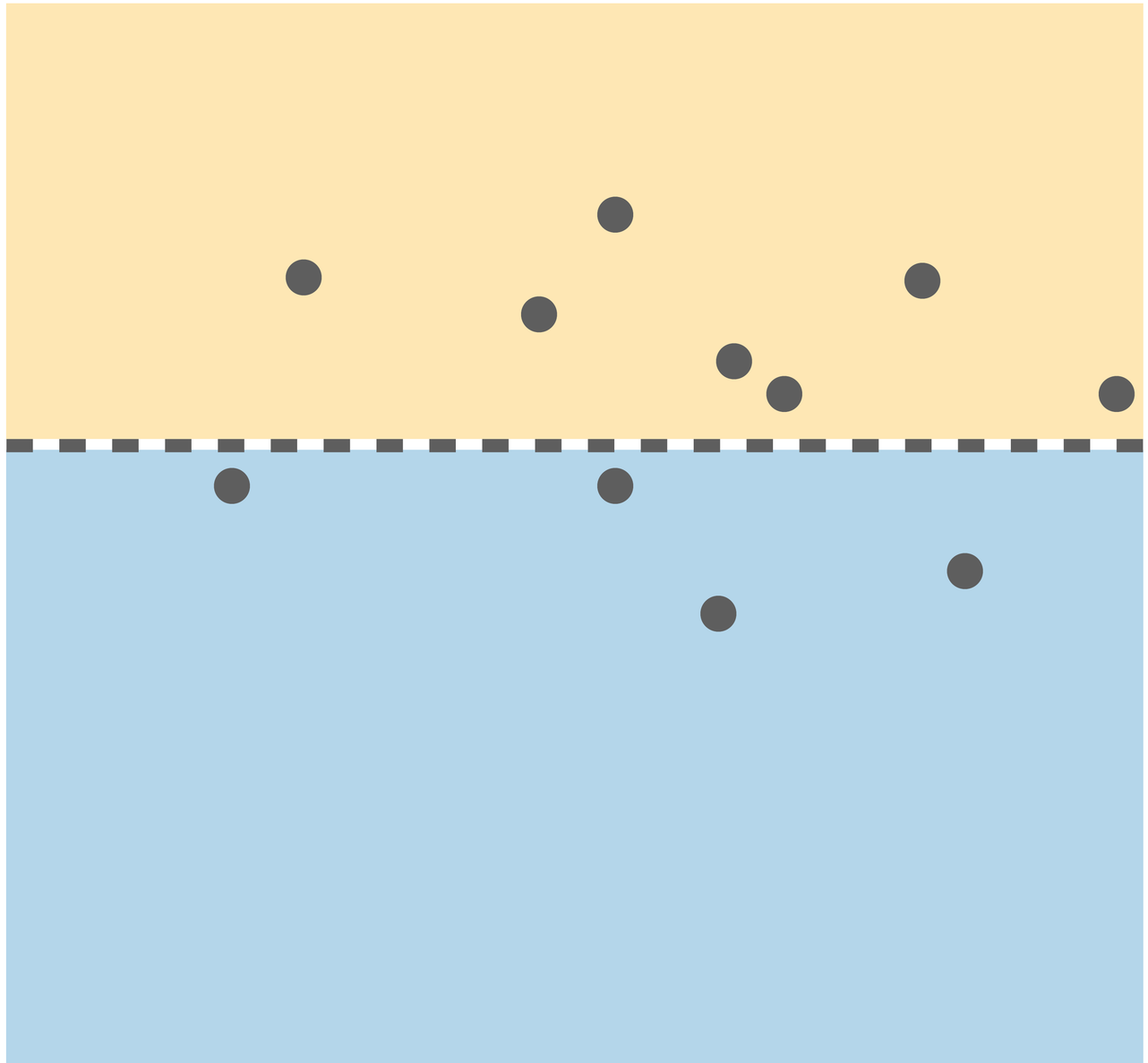
Axis-aligned halfspaces, or **decision stumps**, are a canonical class of halfspaces with bounded directionality:

- Number of directions D = number of dimensions d.
- The normal vectors are the standard basis of $\mathbb{R}^d$.

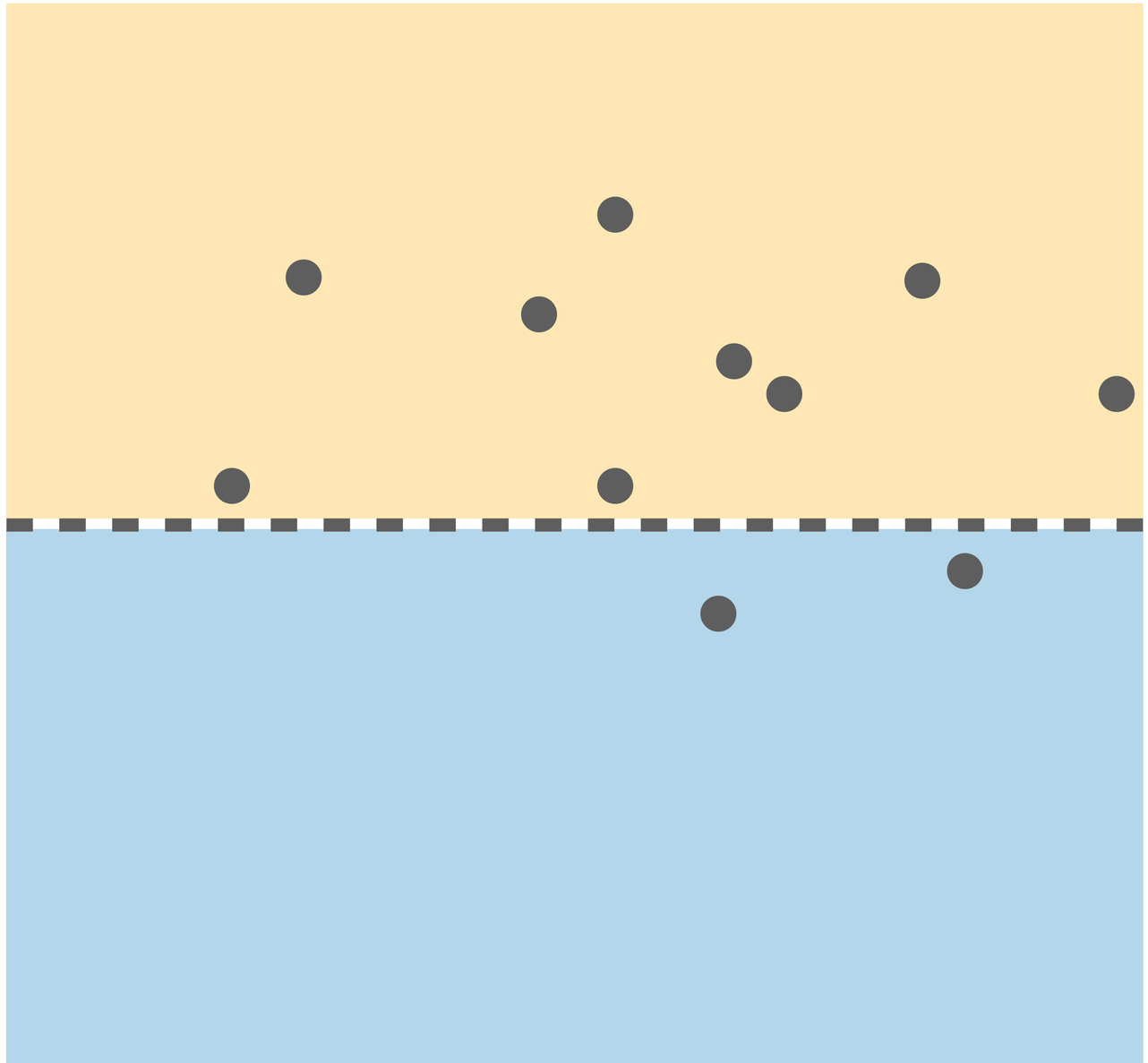
Canonical Class: Axis-Aligned Halfspaces



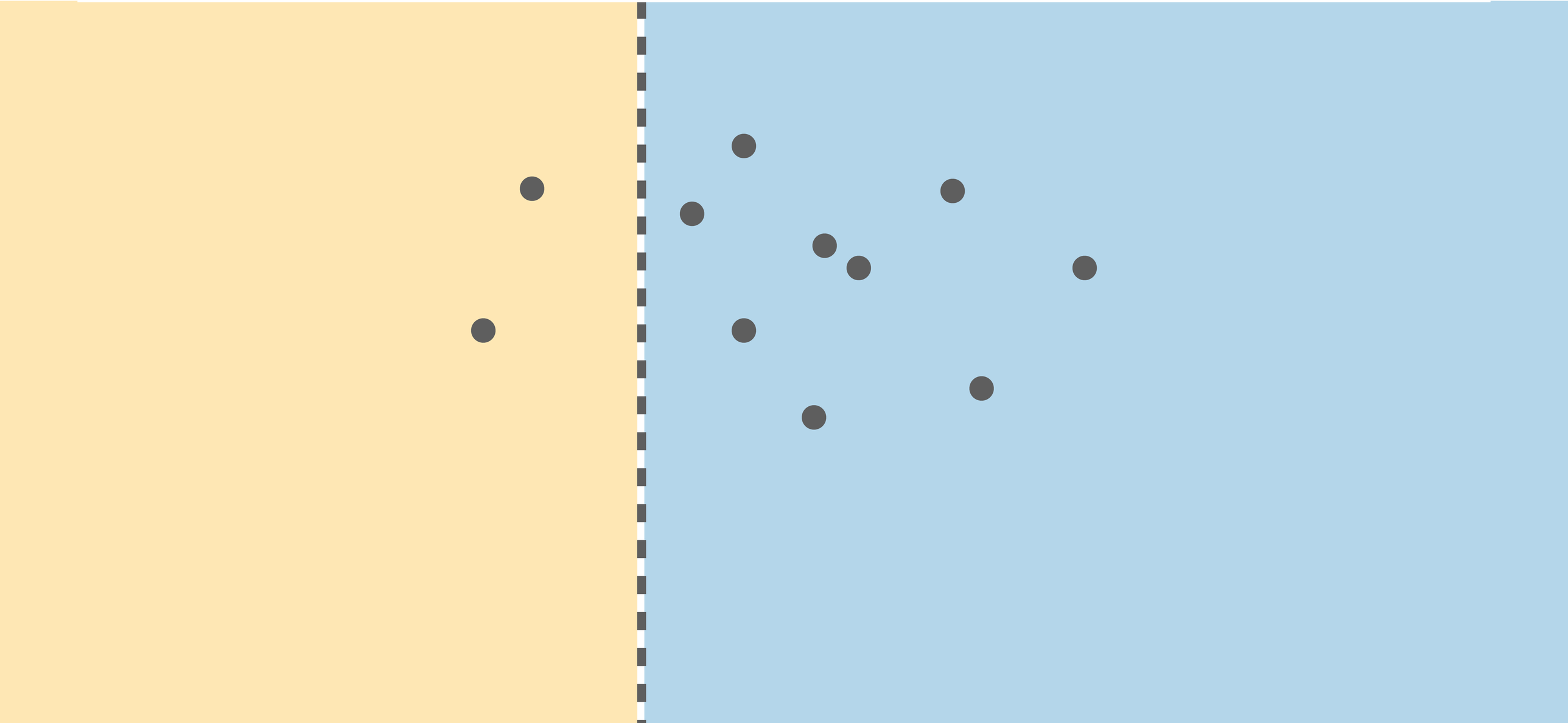
Canonical Class: Axis-Aligned Halfspaces



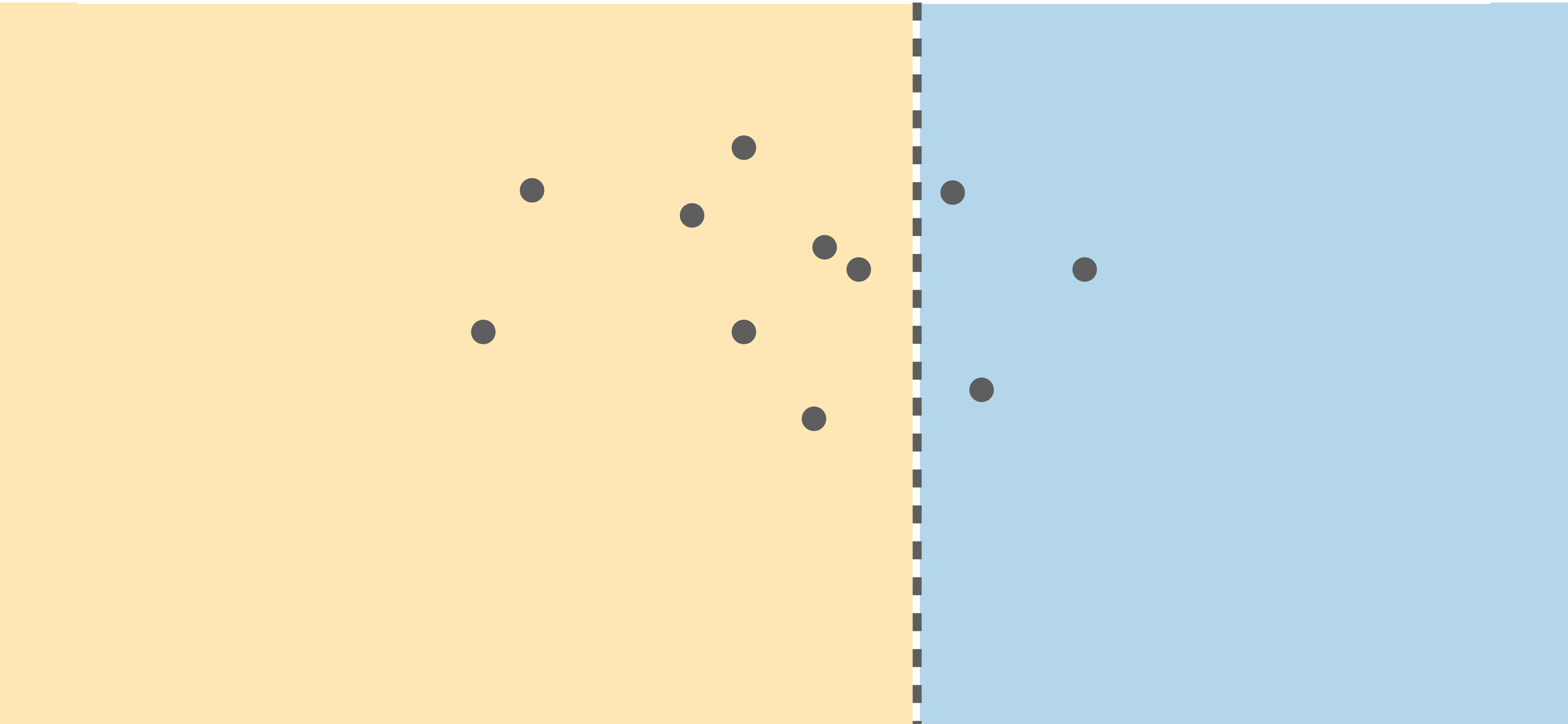
Canonical Class: Axis-Aligned Halfspaces



Canonical Class: Axis-Aligned Halfspaces

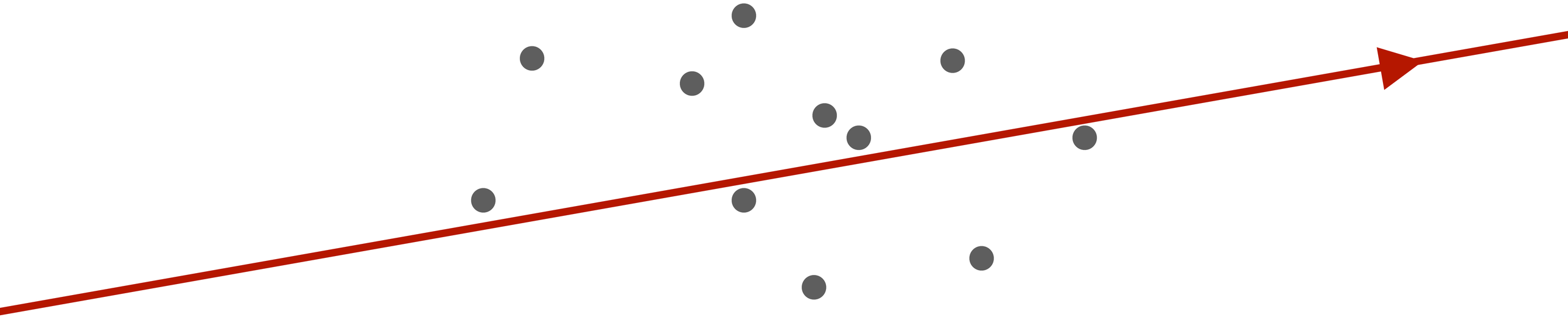


Canonical Class: Axis-Aligned Halfspaces



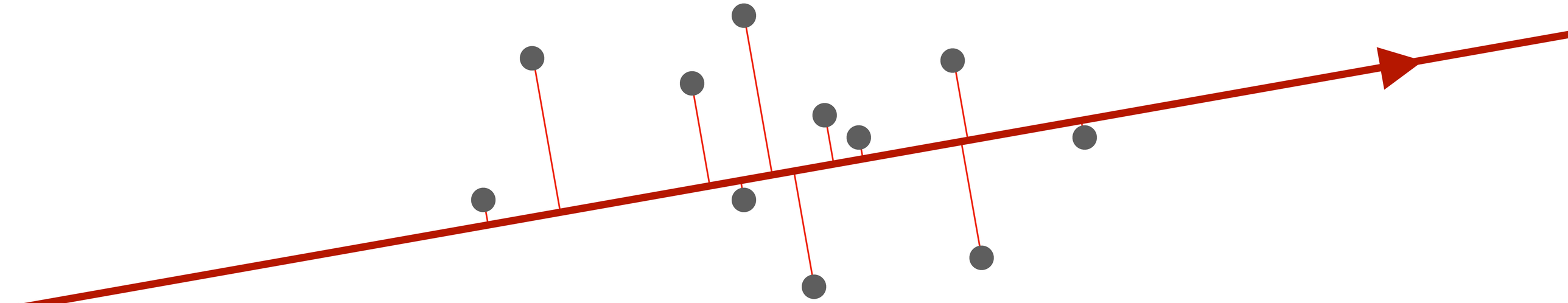
Algorithms and Analysis

Connection to Monotonicity



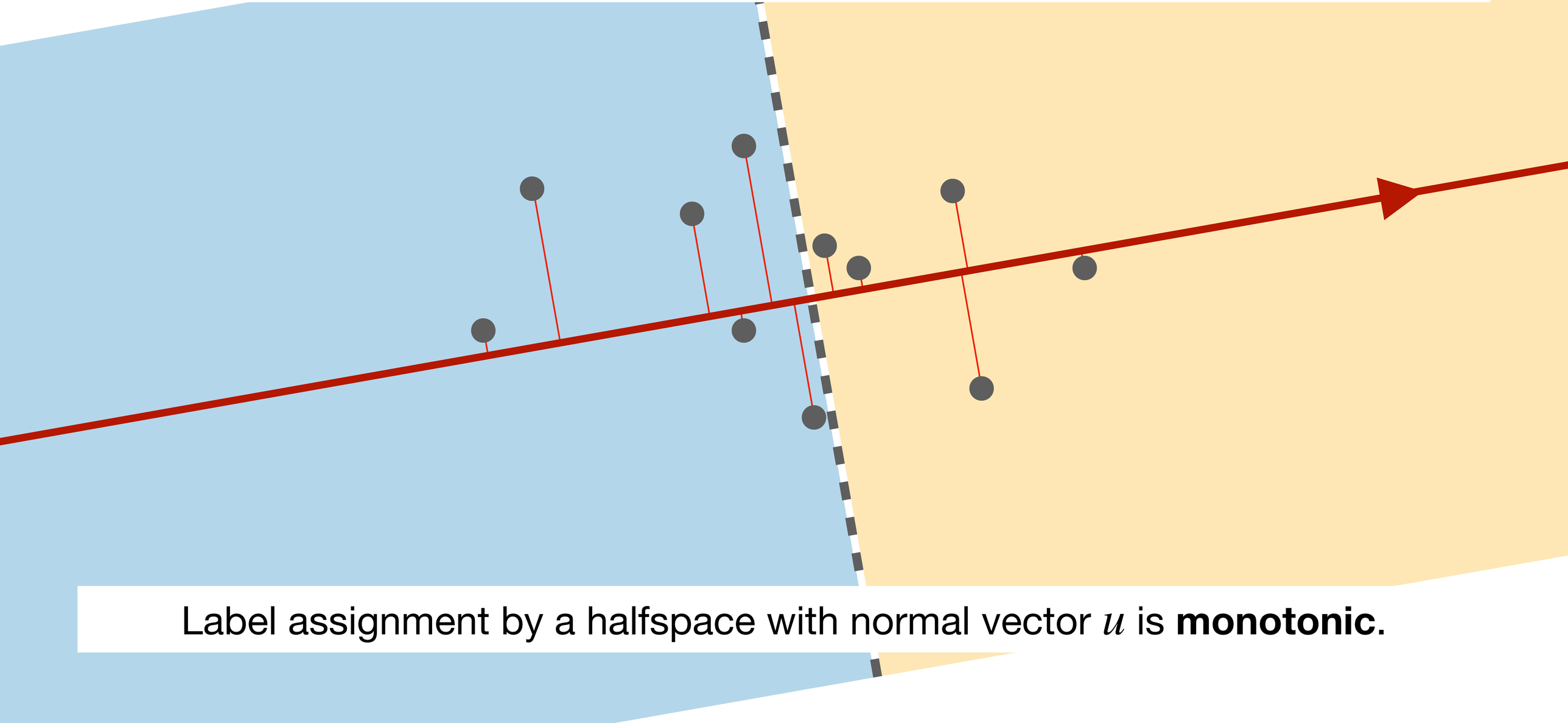
Each **direction** u induces an ordering of the points.

Connection to Monotonicity



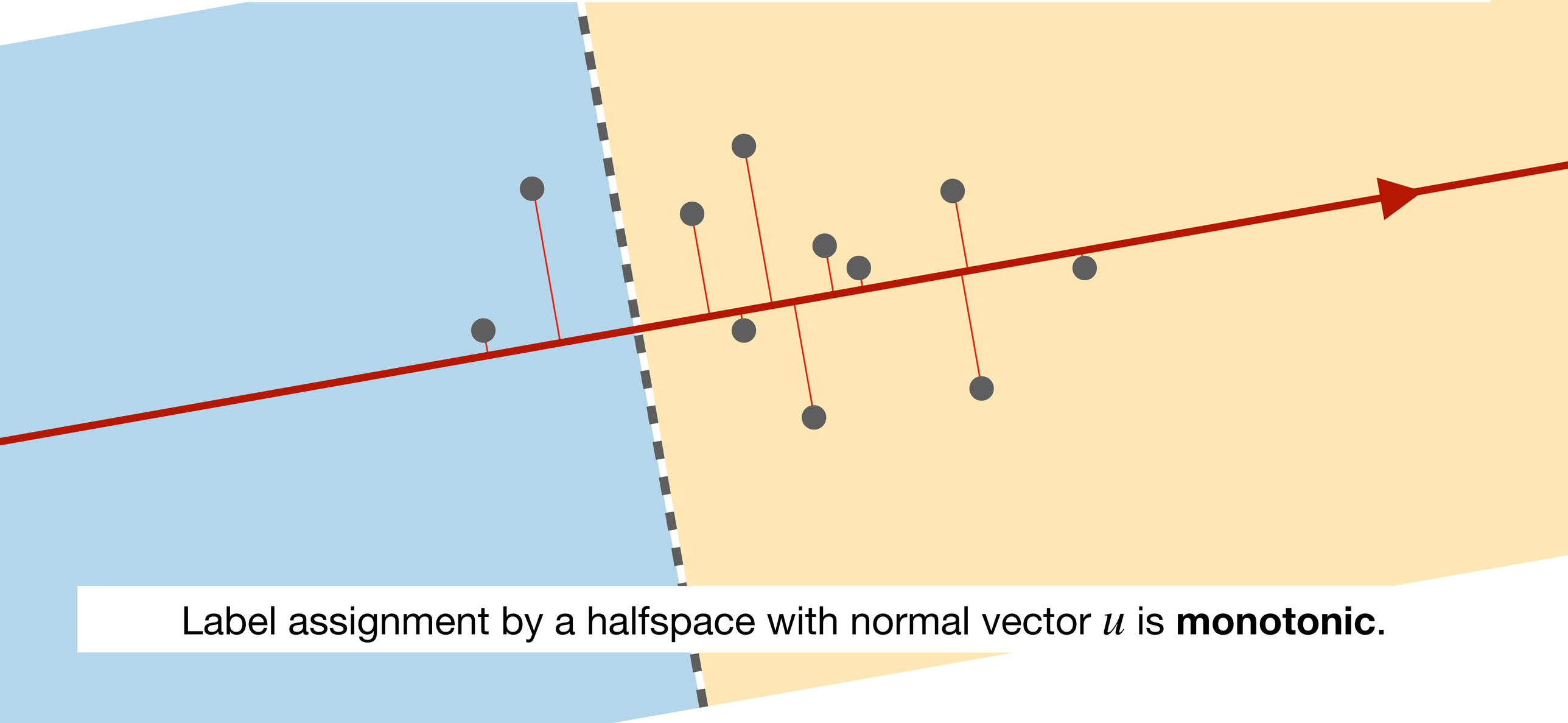
Each **direction** u induces an ordering of the points.

Connection to Monotonicity



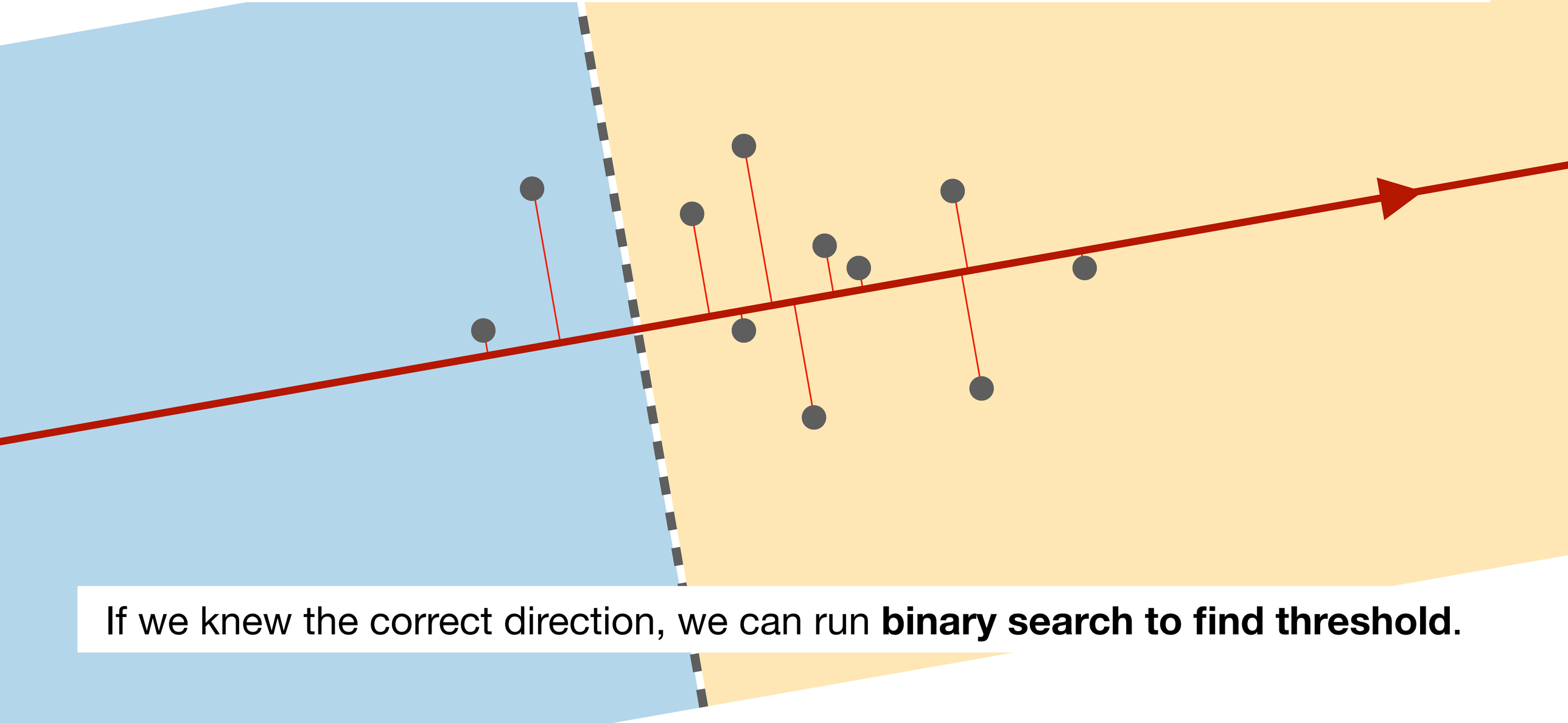
Label assignment by a halfspace with normal vector u is **monotonic**.

Connection to Monotonicity



Label assignment by a halfspace with normal vector u is **monotonic**.

Connection to Monotonicity



If we knew the correct direction, we can run **binary search to find threshold**.

Naive Upper Bound: $O(D \log n)$

1. **Generate candidates hypotheses**
   Run a binary search for each candidate direction $u \in \{u_1, \ldots, u_D\}$.

Naive Upper Bound: $O(D \log n)$

1. **Generate candidates hypotheses**
   Run a binary search for each candidate direction $u \in \{u_1, \ldots, u_D\}$.
   - This yields a collection of hypotheses $V = \{h_1, \ldots, h_D\}$.

Naive Upper Bound: $O(D \log n)$

1. **Generate candidates hypotheses**
   Run a binary search for each candidate direction $u \in \{u_1, \ldots, u_D\}$.
   - This yields a collection of hypotheses $V = \{h_1, \ldots, h_D\}$.
2. **Prune hypotheses**
   While the disagreement region of V is non-empty, query a point in it.

Naive Upper Bound: $O(D \log n)$

1. **Generate candidates hypotheses**
   Run a binary search for each candidate direction $u \in \{u_1, \ldots, u_D\}$.
   - This yields a collection of hypotheses $V = \{h_1, \ldots, h_D\}$.

2. **Prune hypotheses**
   While the disagreement region of V is non-empty, query a point in it.
   - The label will rule out at least one hypothesis from V.

Naive Upper Bound: $O(D \log n)$

1. **Generate candidates hypotheses** $D \log n$ queries
   Run a binary search for each candidate direction $u \in \{u_1, \ldots, u_D\}$.
   - This yields a collection of hypotheses $V = \{h_1, \ldots, h_D\}$.
2. **Prune hypotheses** D queries
   While the disagreement region of V is non-empty, query a point in it.
   - The label will rule out at least one hypothesis from V.

Lower Bound: $\Omega(D + \log n)$

- An $\Omega(D)$ lower bound follows from Dasgupta (2004).

Lower Bound: $\Omega(D + \log n)$

- An $\Omega(D)$ lower bound follows from Dasgupta (2004).
- An $\Omega(\log n)$ lower bound holds even when $D = 1$ (binary search is optimal).

What is the Answer?

Previously-known label complexity:

$$\Omega(D + \log n) \quad \text{vs.} \quad O(D \log n)$$

What is the Answer?

Previously-known label complexity:

$$\Omega(D + \log n) \quad \text{vs.} \quad O(D \log n)$$

This work:

$$\Theta(D + \log n)$$

Algorithmic Idea: Parallel Binary Search

We can achieve at least one of the following using < 10 adaptive queries:

- Rule out a direction
- Generate a candidate threshold for one direction
- Recover the labels for a constant fraction of points

Generalized Question

Learning Union of Hypothesis Classes. Let $\mathscr{H}_1, \ldots, \mathscr{H}_D$ be hypothesis classes that are each learnable using an aggressive active learning strategy.

- How sample efficiently can we learn their union?

Thank You!

Additional Results/Implications

- A proper, realizable $(\varepsilon,0)$-PAC learner using

$$O\left(D+\log\frac{1}{\varepsilon}\right) \text{ queries.}$$

- A proper mildly-noise-tolerant (ε,δ)-PAC learner with constant δ using

$$O\left(D+\log\frac{1}{\varepsilon}\right)\cdot\log D \text{ queries.}$$