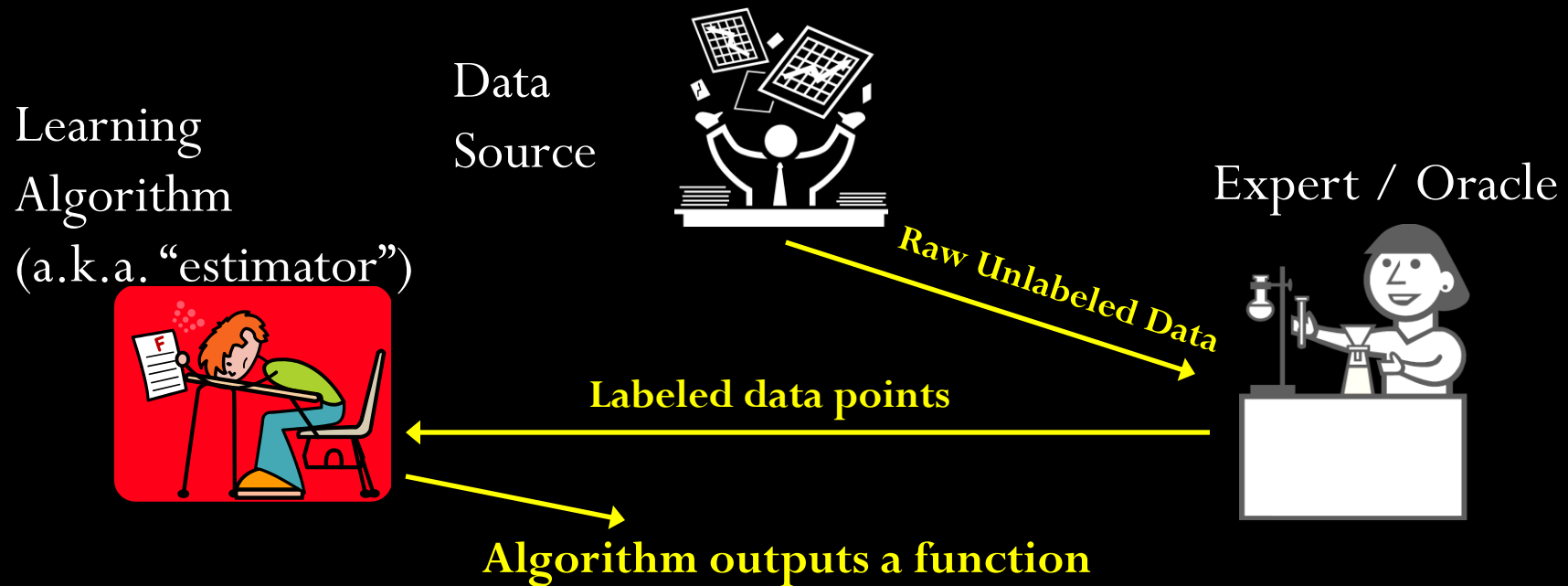


# Theory of Active Learning

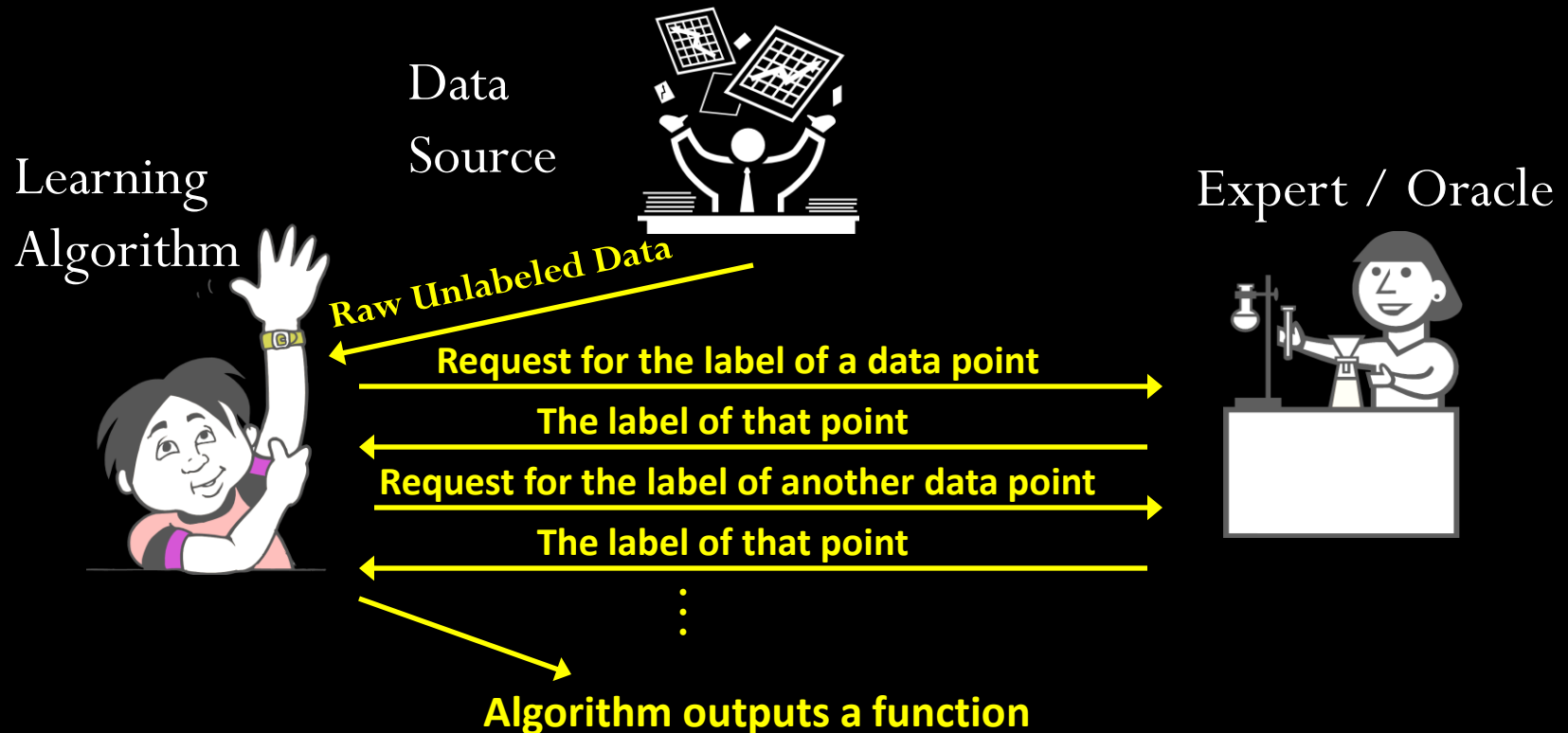
Steve Hanneke  
Carnegie Mellon University



# What is Passive Learning?



# What is Active Learning?

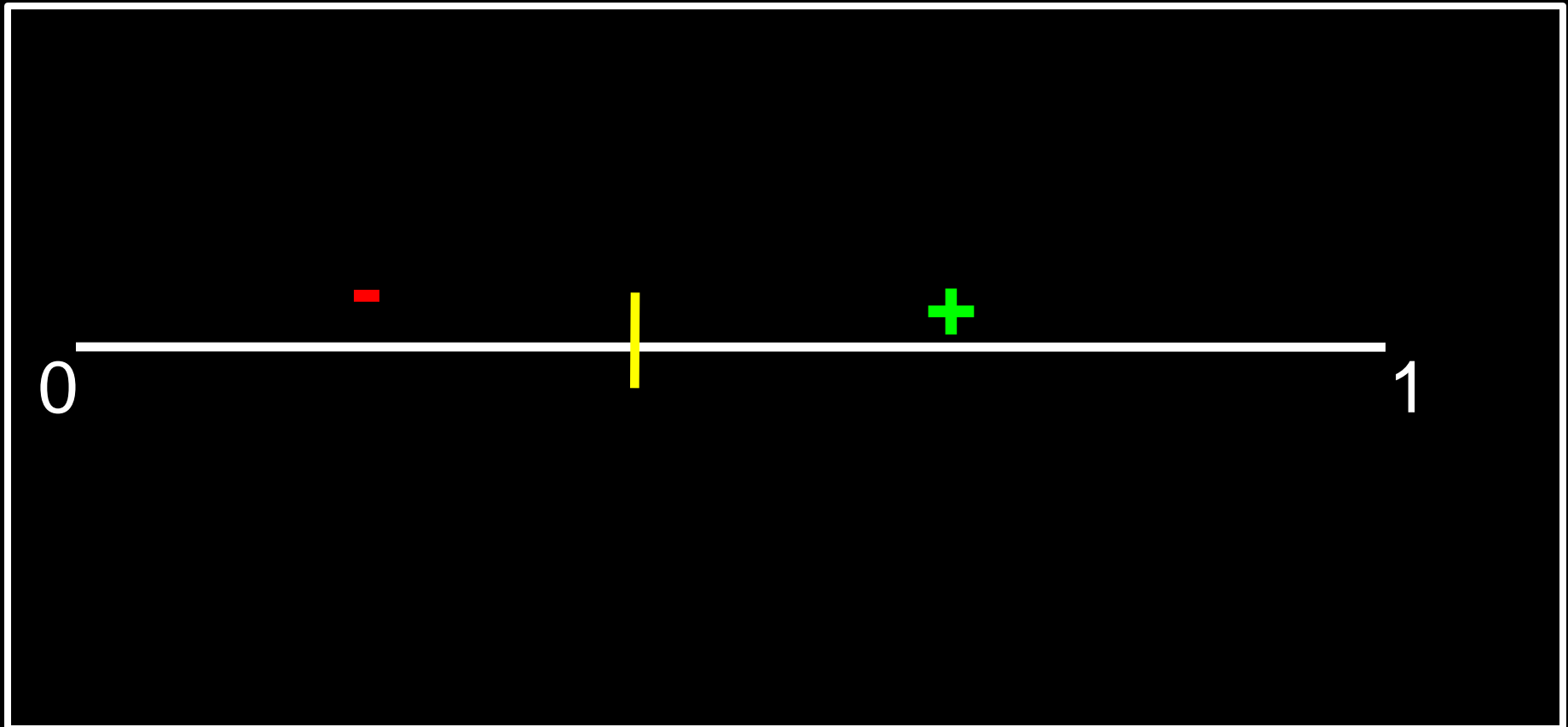


# What is Active Learning?



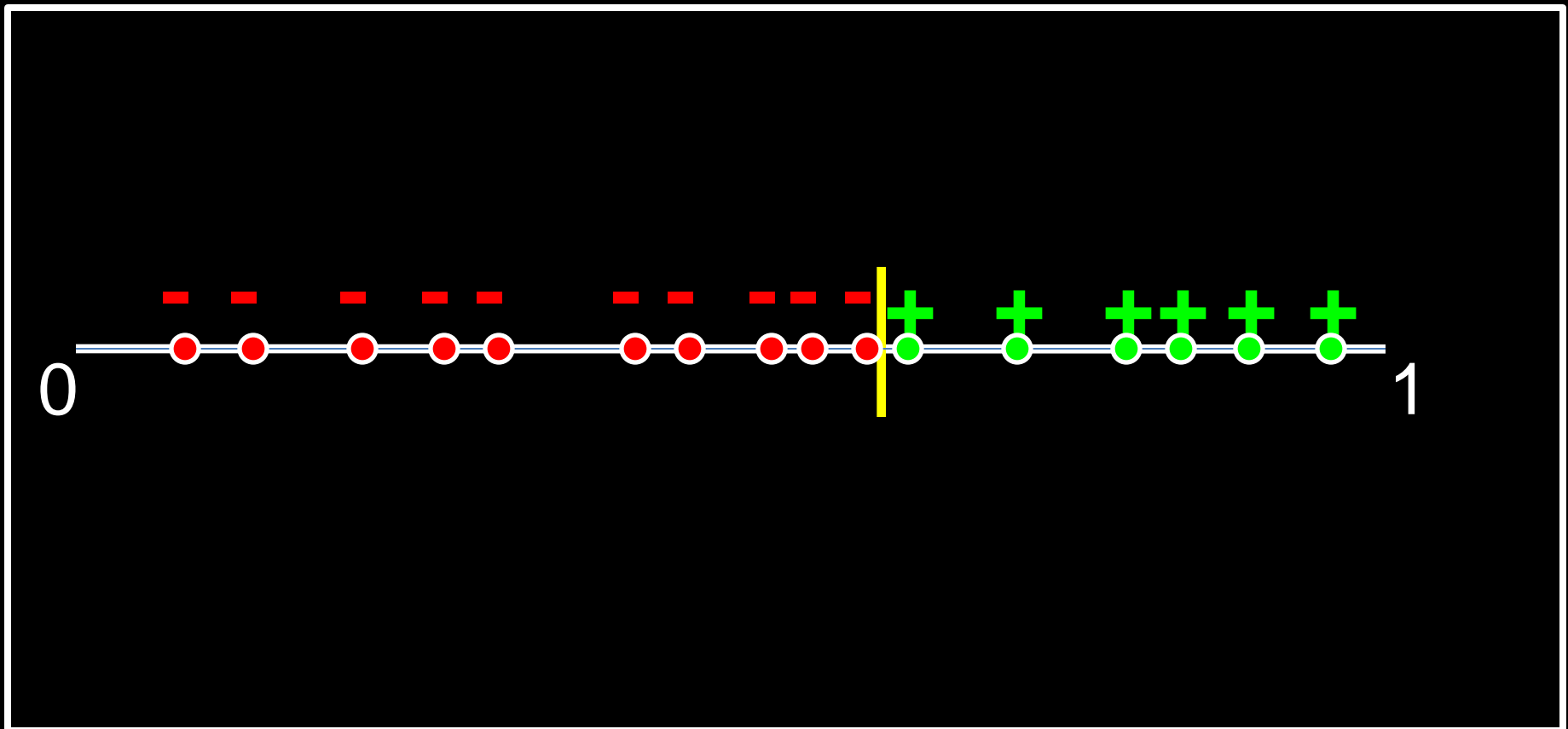
# Example: thresholds

An Example: 1-dimensional threshold functions.



# Example: thresholds

How many **random labeled points** are sufficient to find the threshold, to within  $\pm \varepsilon$ ? (with prob  $1-\delta$ )



# Example: thresholds

How many **random labeled points** are sufficient to find the threshold, to within  $\pm \varepsilon$ ? (with prob  $1-\delta$ )

If we get a  $-$  and  $+$  both within  $\varepsilon$  of  $z$ ,  
any point between them is within  $\varepsilon$  of  $z$ .

Analysis

0

Given  $m$  random points:

$X_1, X_2, \dots, X_m$

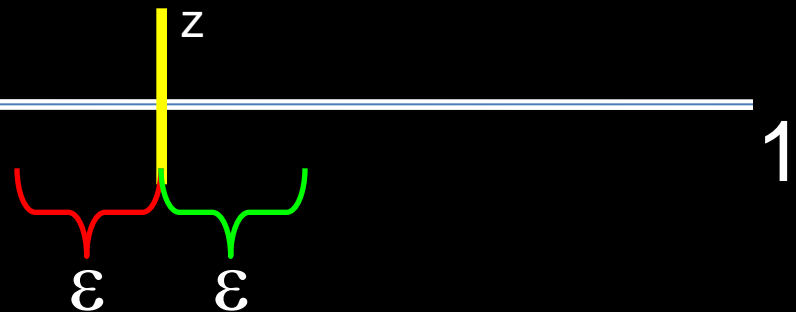
then  $\mathbb{P}(z \leq X_i \leq z + \varepsilon) = \varepsilon$

so  $\mathbb{P}(\exists i \leq m : z \leq X_i \leq z + \varepsilon) = 1 - (1 - \varepsilon)^m \geq 1 - e^{-\varepsilon m}$ .

Similarly,  $\mathbb{P}(\exists i \leq m : z - \varepsilon \leq X_i < z) \geq 1 - e^{-\varepsilon m}$ .

For any  $m \geq \frac{1}{\varepsilon} \ln \left( \frac{2}{\delta} \right)$ ,

with prob  $\geq 1 - 2e^{-\varepsilon m} \geq 1 - \delta$ ,  $\exists +$  and  $-$  within  $\varepsilon$  of  $z$ .



# Example: thresholds

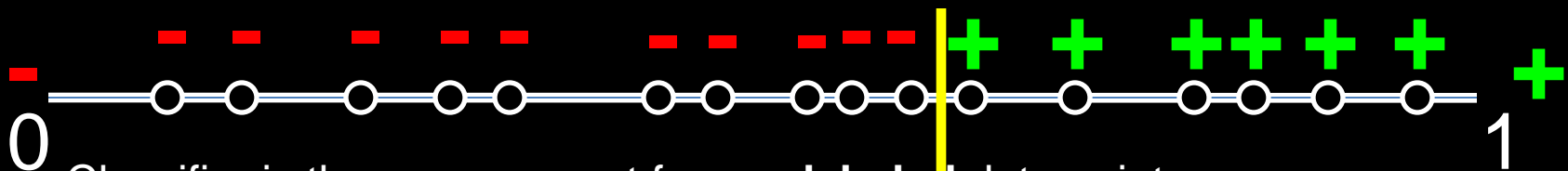
How many **active label requests** are sufficient to find the threshold, to within  $\pm \varepsilon$ ? (with prob  $1-\delta$ )

How can we use active learning to reduce the number of labels?

Take  $m$  random **unlabeled** data points

Repeatedly request the label of the median between closest known + and -

Take the mid-point threshold between adjacent +/- points



Classifier is the **same** we get from  $m$  **labeled** data points

But requested only  $\log(m)$  labels!

Label complexity: Recall  $m = \frac{1}{\varepsilon} \ln \left( \frac{2}{\delta} \right)$  points suffice for passive.

So  $\log_2 \left( \frac{1}{\varepsilon} \ln \left( \frac{2}{\delta} \right) \right)$  label requests suffice.



# Topics We Will Cover

- Disagreement-based active learning algorithms for binary classification
- Active learning with and without noisy labels
- Will focus on provable results (theory)

# Formal Definitions

There is a *target distribution*  $\mathcal{D}_{XY}$  over  $\mathcal{X} \times \{-1, 1\}$ , with marginal  $\mathcal{D}$  over  $\mathcal{X}$ .

There is also a *hypothesis class*  $\mathbb{C}$  of classifiers  $h : \mathcal{X} \rightarrow \{-1, 1\}$ . (a.k.a. *concept space*)

Define the *error rate* of a classifier  $\text{er}(h) = \mathbb{P}[h(X) \neq Y]$  for  $(X, Y) \sim \mathcal{D}_{XY}$ .

There are iid data points  $(X_1, Y_1), (X_2, Y_2), \dots \sim \mathcal{D}_{XY}$  but the  $Y_i$  are hidden from the learning algorithm until individually requested.

# Formal Definitions

Define the *error rate* of a classifier  
 $er(h) = \mathbb{P}[h(X) \neq Y]$  for  $(X, Y) \sim \mathcal{D}_{XY}$ .

There are iid data points  $(X_1, Y_1), (X_2, Y_2), \dots \sim \mathcal{D}_{XY}$   
but the  $Y_i$  are hidden from the learning algorithm  
until individually requested.

# Formal Definitions

Define the *error rate* of a classifier

$$er(h) = \mathbb{P}[h(X) \neq Y] \text{ for } (X, Y) \sim \mathcal{D}_{XY}.$$

There are iid data points  $(X_1, Y_1), (X_2, Y_2), \dots \sim \mathcal{D}_{XY}$  but the  $Y_i$  are hidden from the learning algorithm until individually requested.

Algorithm chooses  $X_i$  to request label, receives  $Y_i$ ; repeat up to  $n$  times.

We will be trying to choose  $h$  to achieve small *excess error*:  $er(h) - \inf_{g \in \mathbb{C}} er(g)$ .

# Formal Definitions

**Definition:** An algorithm  $A(n)$  achieves *label complexity*  $\Lambda(\varepsilon, \delta, \mathcal{D}_{XY})$  if it outputs a classifier  $h_n$  after at most  $n$  label requests, and  $\forall \varepsilon, \delta \in (0, 1), \forall n \geq \Lambda(\varepsilon, \delta, \mathcal{D}_{XY}),$

$$\mathbb{P}[er(h_n) \leq \varepsilon] \geq 1 - \delta.$$

# Important Distinctions

- **Noise-free**, a.k.a. the “realizable case”:

$\mathcal{D}_{XY} \in \mathcal{Realizable}(\mathbb{C})$  means  $\exists f \in \mathbb{C}$  with  $er(f) = 0$ .  
 $f$  is called the *target function*.

- **Noisy**, a.k.a. the “agnostic case”:

$\mathcal{D}_{XY}$  can be arbitrary.

Background:

Passive learning in the realizable case



# Empirical Risk Minimization

For  $\mathcal{L} = \{(x_1, y_1), \dots, (x_m, y_m)\} \in (\mathcal{X} \times \{-1, +1\})^m$ , and  $h : \mathcal{X} \rightarrow \{-1, +1\}$ , the *empirical error rate* is

$$\text{er}_{\mathcal{L}}(h) = \frac{1}{|\mathcal{L}|} \sum_{i=1}^{|\mathcal{L}|} \mathbb{I}[h(x_i) \neq y_i]$$

**Definition:** A *passive learning algorithm* takes as input any finite sequence  $\mathcal{L} \in \bigcup_m (\mathcal{X} \times \{-1, +1\})^m$ , and returns a classifier  $h : \mathcal{X} \rightarrow \{-1, +1\}$ .

**Definition:** For a fixed hypothesis class  $\mathbb{C}$ , a passive learning algorithm  $\mathcal{A}$  is called an *empirical risk minimization* algorithm (ERM) if, for any  $\mathcal{L} \in \bigcup_m (\mathcal{X} \times \{-1, +1\})^m$ ,  $\mathcal{A}(\mathcal{L}) \in \arg \min_{h \in \mathbb{C}} \text{er}_{\mathcal{L}}(h)$ .



# Empirical Risk Minimization

For  $\mathcal{L} = \{(x_1, y_1), \dots, (x_m, y_m)\} \in (\mathcal{X} \times \{-1, +1\})^m$ ,  
and  $h : \mathcal{X} \rightarrow \{-1, +1\}$ , the *empirical error rate* is

$$\text{er}_{\mathcal{L}}(h) = \frac{1}{|\mathcal{L}|} \sum_{i=1}^{|\mathcal{L}|} \mathbb{I}[h(x_i) \neq y_i]$$

Some “shorthand” notation:

$$\text{er}_m(h) = \text{er}_{\mathcal{L}}(h) \text{ for } \mathcal{L} = \{(X_1, Y_1), \dots, (X_m, Y_m)\}.$$

# VC Dimension

Definition:  $\mathbb{C}$  *shatters*  $\{x_1, x_2, \dots, x_m\} \in \mathcal{X}^m$  if  $\forall y_1, y_2, \dots, y_m \in \{-1, +1\}$ ,  $\exists h \in \mathbb{C}$  such that  $\forall i \in \{1, \dots, m\}$ ,  $h(x_i) = y_i$ .

Definition: The *VC dimension* of  $\mathbb{C}$  is the largest  $m \in \mathbb{N}$  s.t.  $\exists \{x_1, \dots, x_m\} \in \mathcal{X}^m$  that  $\mathbb{C}$  shatters.

We will denote the VC dimension by  $d$ .

If  $d < \infty$ , then  $\mathbb{C}$  is called a *VC class*.

# VC Dimension

Definition:  $\mathbb{C}$  *shatters*  $\{x_1, x_2, \dots, x_m\} \in \mathcal{X}^m$  if  $\forall y_1, y_2, \dots, y_m \in \{-1, +1\}$ ,  $\exists h \in \mathbb{C}$  such that  $\forall i \in \{1, \dots, m\}$ ,  $h(x_i) = y_i$ .

Definition: The *VC dimension* of  $\mathbb{C}$  is the largest  $m \in \mathbb{N}$  s.t.  $\exists \{x_1, \dots, x_m\} \in \mathcal{X}^m$  that  $\mathbb{C}$  shatters.

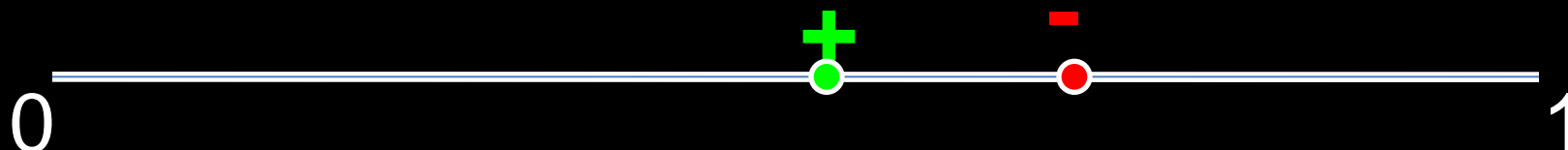


Thresholds can shatter one point

# VC Dimension

Definition:  $\mathbb{C}$  *shatters*  $\{x_1, x_2, \dots, x_m\} \in \mathcal{X}^m$  if  $\forall y_1, y_2, \dots, y_m \in \{-1, +1\}$ ,  $\exists h \in \mathbb{C}$  such that  $\forall i \in \{1, \dots, m\}$ ,  $h(x_i) = y_i$ .

Definition: The *VC dimension* of  $\mathbb{C}$  is the largest  $m \in \mathbb{N}$  s.t.  $\exists \{x_1, \dots, x_m\} \in \mathcal{X}^m$  that  $\mathbb{C}$  shatters.



No threshold agrees with these labels, for any two points

So  $d=1$  for threshold classifiers

# VC Dimension

- For intervals,  $d=2$
- For unions of  $k$  intervals,  $d=2k$
- For linear separators in  $k$  dimensions,  $d=k+1$
- For axis-aligned rectangles in  $k$  dimensions,  $d=2k$

# Label Complexity of ERM

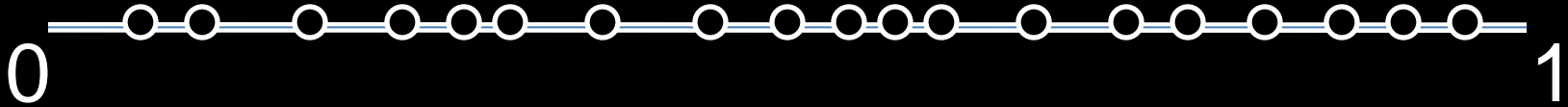
**Lemma 1** [Vapnik, 82]: For a constant  $c_p \in [1, \infty)$ ,  $\forall \ell \in \mathbb{N}$ , for any distribution  $P$  on  $\mathcal{X}$  and  $X'_1, \dots, X'_\ell$  iid  $P$ , with probability  $1 - \alpha$ , if  $V = \{h \in \mathbb{C} : \forall i \leq \ell, h(X'_i) = f(X'_i)\}$ , then  $\sup_{h \in V} P(x : h(x) \neq f(x)) \leq \frac{c_p}{\ell} (d \ln(e\ell/d) + \ln(1/\alpha))$ .

**Corollary:** For a constant  $c'_p \in [1, \infty)$ , for any  $\mathcal{D}_{XY}$  in  $\text{Realizable}(\mathbb{C})$  ERM achieves label complexity  $\Lambda(\varepsilon, \delta, \mathcal{D}_{XY}) = \frac{c'_p}{\varepsilon} \left( d \ln \left( \frac{1}{\varepsilon} \right) + \ln \left( \frac{1}{\delta} \right) \right) = \tilde{O} \left( \frac{1}{\varepsilon} \right)$ .

# Disagreement-Based Active Learning



# A Simple Strategy





# CAL

A simple idea from Cohn, Atlas & Ladner (1994).

Algorithm: **CAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0$

1. For  $m = 1, 2, \dots$

2.     If  $\exists h_1, h_2 \in V_t$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,

3.         Request  $Y_m$

4.          $t \leftarrow t + 1$

5.          $V_t \leftarrow \{h \in V_{t-1} : h(X_m) = Y_m\}$

6.         If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_n$

# CAL

A simple idea from Cohn, Atlas & Ladner (1994).

Algorithm: **CAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0$

1. For  $m = 1, 2, \dots$

2.     If  $\exists h_1, h_2 \in V_t$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,

3.         Request  $Y_m$

4.          $t \leftarrow t + 1$

5.          $V_t \leftarrow \{h \in V_{t-1} : h(X_m) = Y_m\}$

6.         If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_n$

# Disagreement Coefficient

For  $\mathcal{H} \subseteq \mathbb{C}$ , define the *region of disagreement*

$$\text{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\},$$

the  $r$ -ball centered at  $f$ :

$$\text{B}(f, r) = \{h \in \mathbb{C} : \mathbb{P}(x : h(x) \neq f(x)) \leq r\},$$

and the *disagreement coefficient* of  $f$  with respect to  $\mathbb{C}$ :

$$\theta_f(r_0) = \sup_{r > r_0} \frac{\mathbb{P}(\text{DIS}(\text{B}(f, r)))}{r}.$$

Sometimes abbreviate:  $\theta_f = \theta_f(0)$ .

# Disagreement Coefficient

$$\text{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\},$$

$$\text{B}(f, r) = \{h \in \mathbb{C} : \mathbb{P}(x : h(x) \neq f(x)) \leq r\},$$

$$\theta_f(r_0) = \sup_{r > r_0} \frac{\mathbb{P}(\text{DIS}(\text{B}(f, r)))}{r}.$$

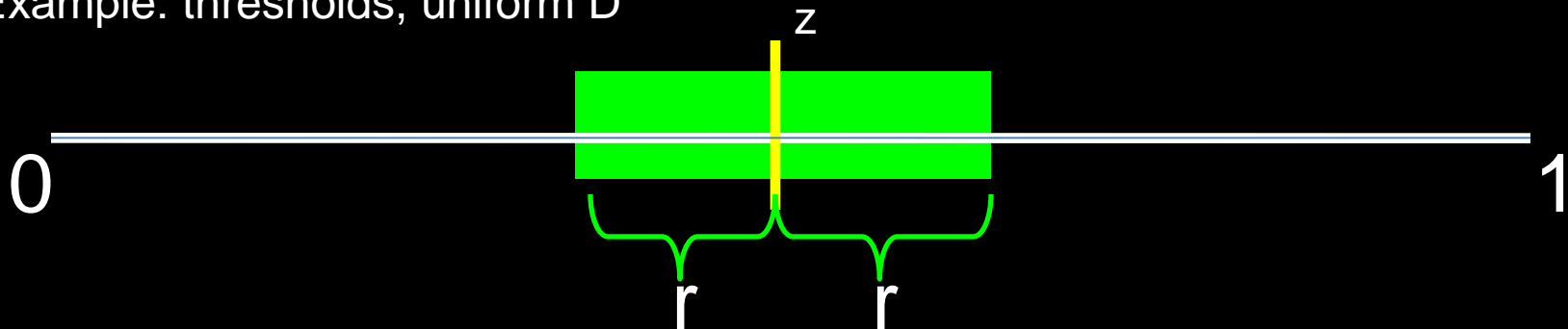
# Disagreement Coefficient

$$\text{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\},$$

$$\text{B}(f, r) = \{h \in \mathbb{C} : \mathbb{P}(x : h(x) \neq f(x)) \leq r\},$$

$$\theta_f(r_0) = \sup_{r > r_0} \frac{\mathbb{P}(\text{DIS}(\text{B}(f, r)))}{r}.$$

Example: thresholds, uniform D



$\text{B}(f, r)$  is set of thresholds within  $r$  of  $z$

$$\text{DIS}(\text{B}(f, r)) = [z - r, z + r].$$

Since  $\mathbb{P}(\text{DIS}(\text{B}(f, r))) = 2r$ , we have  $\theta_f = 2$ .

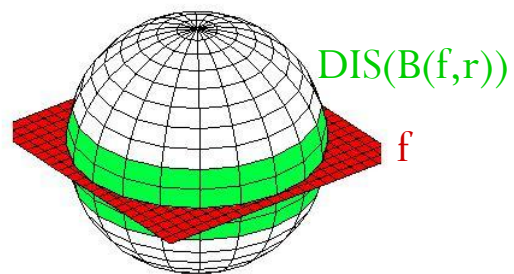
# Disagreement Coefficient

$$\text{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\},$$

$$B(f, r) = \{h \in \mathbb{C} : \mathbb{P}(x : h(x) \neq f(x)) \leq r\},$$

$$\theta_f(r_0) = \sup_{r > r_0} \frac{\mathbf{P}(\text{DIS}(B(f, r)))}{r}.$$

Example: linear separators,  $d$  dimensions, uniform  $D$  on sphere



Classifiers in  $B(f, r)$   
look like this

If we do the calculations, for small  $r$ ,  
 $\mathbb{P}(\text{DIS}(B(f, r))) = \Theta(\sqrt{d}r)$ ,  
 and thus  $\theta_f = \Theta(\sqrt{d})$ .

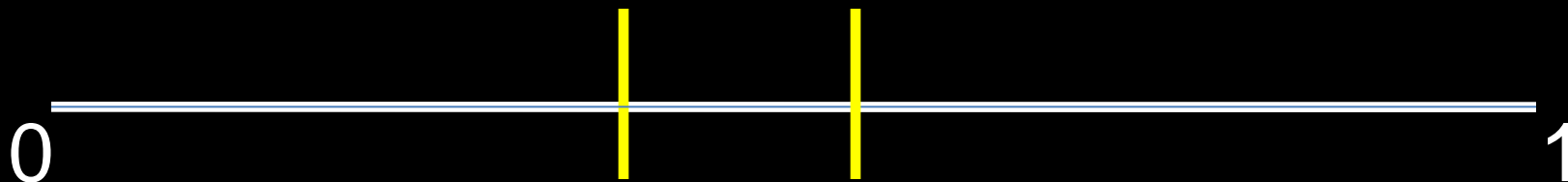
# Disagreement Coefficient

$$\text{DIS}(\mathcal{H}) = \{x \in \mathcal{X} : \exists h, g \in \mathcal{H} \text{ s.t. } h(x) \neq g(x)\},$$

$$\text{B}(f, r) = \{h \in \mathbb{C} : \mathbb{P}(x : h(x) \neq f(x)) \leq r\},$$

$$\theta_f(r_0) = \sup_{r > r_0} \frac{\mathbf{P}(\text{DIS}(\text{B}(f, r)))}{r}.$$

Example: intervals, uniform D



# Label Complexity of CAL

**Theorem:** For a constant  $c \in (0, \infty)$ , for any  $\mathcal{D}_{XY} \in \mathcal{Realizable}(\mathbb{C})$ , if  $f \in \mathbb{C}$  is the target function, CAL achieves label complexity  $\Lambda(\varepsilon, \delta, \mathcal{D}_{XY}) = c \cdot \theta_f(\varepsilon) \left( d \log \theta_f(\varepsilon) + \log \frac{\log(1/\varepsilon)}{\delta} \right) \log \frac{1}{\varepsilon}$ .



# Label Complexity of CAL

Proof Sketch:

Denote by  $m_t$  the  $m$  for the  $t^{\text{th}}$  label request  $Y_{m_t}$ .

$$L = \left\lceil c_1 \cdot 4\theta_f(\varepsilon) \cdot \left( d \log(c_1 2\theta_f(\varepsilon)) + \log \frac{\log_2(2/\varepsilon)}{\delta} \right) \right\rceil.$$

For  $t \leq n - L$ ,  $|\{X_{m_t+1}, \dots, X_{m_t+L}\} \cap \text{DIS}(V_t)| \geq L$ .

First  $L$  of these in  $\text{DIS}(V_t)$  are conditionally (given  $V_t$ ) iid with distrib (given  $V_t$ ) equal to conditional of  $X \sim \mathcal{D}$  given  $X \in \text{DIS}(V_t)$ .

Lemma 1  $\Rightarrow$  with prob  $1 - \delta / \log_2(2/\varepsilon)$ ,  $\forall h \in V_{t+L}$ ,  $\mathbb{P}(h(X) \neq f(X) | X \in \text{DIS}(V_t))$

$$\leq \frac{c_p}{L} \left( d \log(eL/d) + \log \frac{\log_2(2/\varepsilon)}{\delta} \right) \leq \frac{1}{2\theta_f(\varepsilon)}.$$

# Label Complexity of CAL

$$\begin{aligned}
 V_{t+L} \subseteq V_t &\Rightarrow \forall h \in V_{t+L}, \mathbb{P}(h(X) \neq f(X)) \\
 &= \mathbb{P}(h(X) \neq f(X) | X \in \text{DIS}(V_t)) \mathbb{P}(X \in \text{DIS}(V_t)) \\
 &+ \mathbb{P}(h(X) \neq f(X) | X \notin \text{DIS}(V_t)) \mathbb{P}(X \notin \text{DIS}(V_t)) \\
 &\leq \frac{\mathbb{P}(\text{DIS}(V_t))}{2\theta_f(\varepsilon)}. \quad \text{So } V_{t+L} \subseteq \mathcal{B}\left(f, \frac{\mathbb{P}(\text{DIS}(V_t))}{2\theta_f(\varepsilon)}\right).
 \end{aligned}$$

If  $\sup_{h \in V_{t+L}} \mathbb{P}(h(X) \neq f(X)) > \varepsilon$ ,

$$\mathbb{P}(\text{DIS}(V_{t+L})) \leq \mathbb{P}\left(\text{DIS}\left(\mathcal{B}\left(f, \frac{\mathbb{P}(\text{DIS}(V_t))}{2\theta_f(\varepsilon)}\right)\right)\right) \leq \frac{\mathbb{P}(\text{DIS}(V_t))}{2}.$$

Union bound  $\Rightarrow$  this happens with prob  $1 - \delta$  for all  $t = 0, L, 2L, \dots, \lceil \log_2(1/\varepsilon) \rceil \cdot L$ .

$$\sup_{h \in V_n} \mathbb{P}(h(X) \neq f(X)) \leq \mathbb{P}(\text{DIS}(V_n)),$$

so  $n \geq L \cdot \lceil \log_2(1/\varepsilon) \rceil$  suffices to guarantee

$$\sup_{h \in V_n} \text{er}(h) = \sup_{h \in V_n} \mathbb{P}(h(X) \neq f(X)) \leq \varepsilon.$$



# Passive Learning with Noisy Labels



# Where does “noise” come from?

- Stochasticity of nature
- Feature space under-specification
- Multiple labelers, differing interpretations
- Incompetent labelers
- Model mis-specification

# Tsybakov Noise

Definition ( $\varepsilon$ -minimal set): For  $\varepsilon > 0$ , let  $\mathbb{C}(\varepsilon) = \{h \in \mathbb{C} : \text{er}(h) - \nu \leq \varepsilon\}$ .

Definition: For  $\mu, \kappa \in [1, \infty)$ ,  $\mathcal{D}_{XY} \in \mathcal{Tsybakov}(\mu, \kappa, \mathbb{C})$  means  $\forall \varepsilon > 0, \text{diam}(\mathbb{C}(\varepsilon)) \leq \mu \cdot \varepsilon^{\frac{1}{\kappa}}$ .

Example: if  $\mathcal{D}_{XY} \in \mathcal{Realizable}(\mathbb{C})$   $\mathbb{C}(\varepsilon) = B(f, \varepsilon)$ , so  $\text{diam}(\mathbb{C}(\varepsilon)) \leq 2\varepsilon$ .

# Excess Risk Bounds for ERM

For  $\mathcal{H} \subseteq \mathbb{C}$  and  $\alpha \in (0, 1)$ , define

$$\tilde{\mathcal{E}}_m(\mathcal{H}, \alpha) =$$

$$\tilde{c} \left( \sqrt{\frac{\text{diam}(\mathcal{H})(d \log(m) + \log(1/\alpha))}{m}} + \frac{\log(1/\alpha)}{m} \right).$$

# Excess Risk Bounds for ERM

**Lemma 2** [Koltchinskii, 06]: For any  $\alpha \in (0, 1)$ ,  $m \in \mathbb{N}$ , and  $\mathcal{H} \subseteq \mathbb{C}$ , with probability  $1 - \alpha$ , if  $f = \arg \min_{h \in \mathcal{H}} \text{er}(h)$ , then  $\forall h \in \mathcal{H}$ ,

$$\text{er}_m(h) - \min_{g \in \mathcal{H}} \text{er}_m(g) \leq \text{er}(h) - \text{er}(f) + \tilde{\mathcal{E}}_m(\mathcal{H}, \alpha)$$

$$\text{er}(h) - \text{er}(f) \leq \text{er}_m(h) - \text{er}_m(f) + \tilde{\mathcal{E}}_m(\mathcal{H}, \alpha).$$

Corollary: With probability  $1 - \delta$ , if  $\hat{h} = \arg \min_{h \in \mathbb{C}} \text{er}_m(h)$ , then

$$\text{er}(\hat{h}) - \inf_{f \in \mathbb{C}} \text{er}(f) \leq \tilde{\mathcal{E}}_m(\mathbb{C}, \delta).$$

# Excess Risk Bounds for ERM

Corollary: If  $\mathcal{D}_{XY} \in \mathcal{Tsybakov}(\mu, \kappa, \mathbb{C})$ , then with probability  $1 - \delta$ , if  $\hat{h} = \arg \min_{h \in \mathbb{C}} \text{er}_m(h)$ , then

$$\text{er}(\hat{h}) - \inf_{f \in \mathbb{C}} \text{er}(f) \leq c' \cdot \left( \frac{d \log(m) + \log(1/\delta)}{m} \right)^{\frac{\kappa}{2\kappa - 1}}.$$

Corollary: If  $\mathcal{D}_{XY} \in \mathcal{Tsybakov}(\mu, \kappa, \mathbb{C})$  and  $f = \arg \min_{h \in \mathbb{C}} \text{er}(h)$ , then ERM achieves label complexity

$$\begin{aligned} \Lambda(\varepsilon + \text{er}(f), \delta, \mathcal{D}_{XY}) &= c'' \varepsilon^{\frac{1}{\kappa} - 2} \cdot (d \log(d/\varepsilon) + \log(1/\delta)) \\ &= \tilde{O} \left( \varepsilon^{\frac{1}{\kappa} - 2} \right). \end{aligned}$$



# Data-Dependent Bounds

Let  $\xi_1, \xi_2, \dots$  be iid  $\text{Uniform}(\{-1, +1\})$ .

For  $\mathcal{H} \subseteq \mathbb{C}$ , define (*Rademacher process*)

$$\hat{R}_m(\mathcal{H}) = \sup_{g, h \in \mathcal{H}} \frac{1}{2m} \sum_{i=1}^m \xi_i \cdot (g(X_i) - h(X_i)).$$

Define (*empirical diameter*)

$$\hat{D}_m(\mathcal{H}) = \sup_{g, h \in \mathcal{H}} \frac{1}{2m} \sum_{i=1}^m |g(X_i) - h(X_i)|.$$

$$\hat{U}_m(\mathcal{H}, \delta) =$$

$$\hat{K} \cdot \left( \hat{R}_m(\mathcal{H}) + \sqrt{\hat{D}_m(\mathcal{H}) \frac{\log(\log(m)/\delta)}{m}} + \frac{\log(1/\delta)}{m} \right).$$

(can take  $\hat{K} = 752$  for our purposes)

# Data-Dependent Bounds for ERM

**Lemma** [Koltchinskii, 06]: For any  $\alpha \in (0, 1)$ ,  $m \in \mathbb{N}$ , and  $\mathcal{H} \subseteq \mathbb{C}$ , with probability  $1 - \alpha$ , if  $f = \arg \min_{h \in \mathcal{H}} \text{er}(h)$ , then  $\forall h \in \mathcal{H}$ ,

$$\text{er}_m(h) - \min_{g \in \mathcal{H}} \text{er}_m(g) \leq \text{er}(h) - \text{er}(f) + \hat{U}_m(\mathcal{H}, \alpha)$$

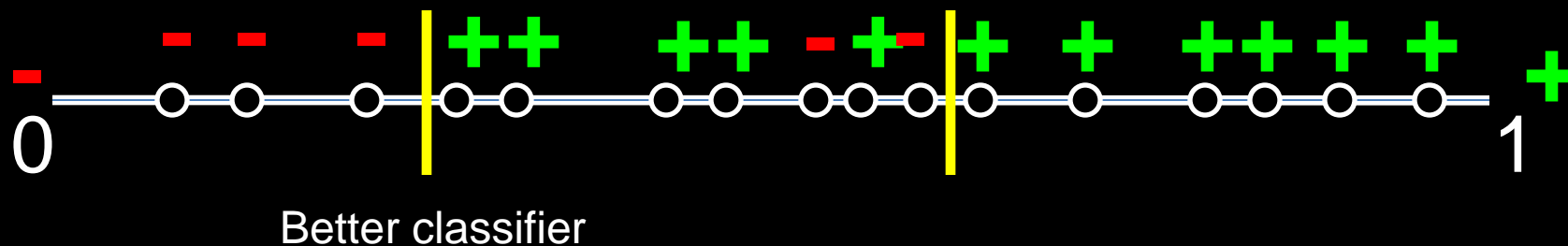
$$\text{er}(h) - \text{er}(f) \leq \text{er}_m(h) - \text{er}_m(f) + \hat{U}_m(\mathcal{H}, \alpha)$$

$$\hat{U}_m(\mathcal{H}, \alpha) \wedge 1 \leq \tilde{\mathcal{E}}_m(\mathcal{H}, \alpha).$$

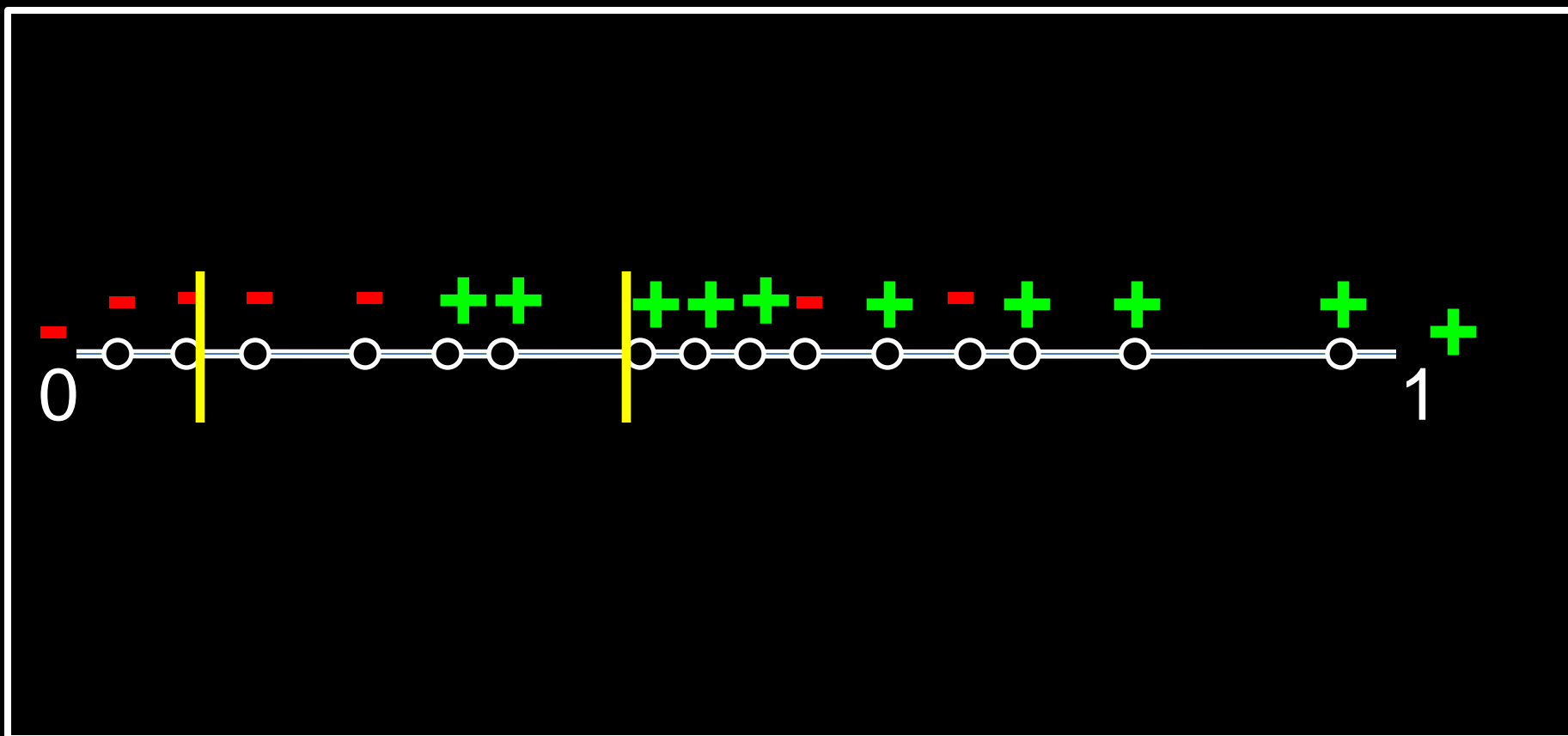
# Active Learning with Noisy Labels



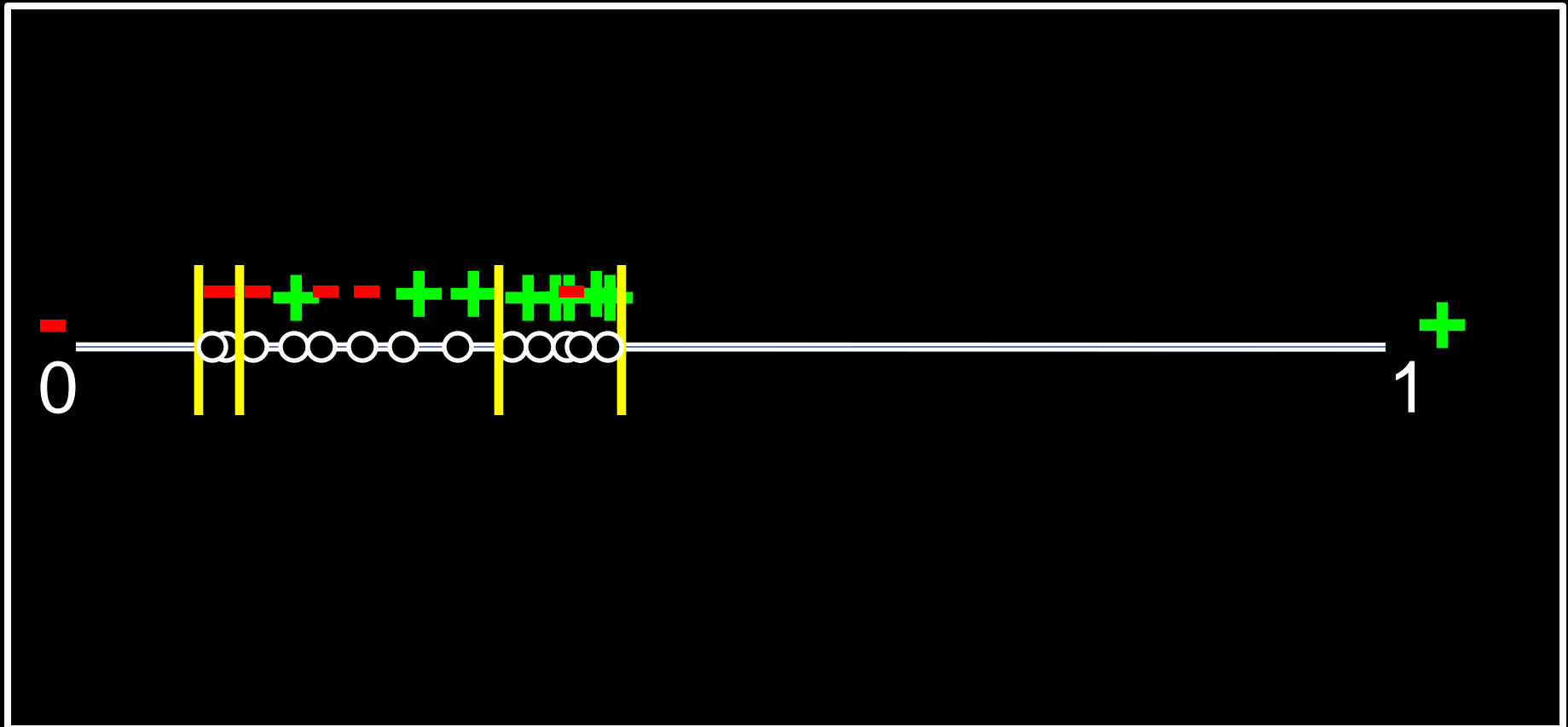
# Agnostic Case Example: Thresholds



# Agnostic Case Example: Thresholds



# Agnostic Case Example: Thresholds



# CAL Revisited

Algorithm: **CAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0$

1. For  $m = 1, 2, \dots$

2.     If  $\exists h_1, h_2 \in V_t$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,

3.         Request  $Y_m$

4.          $t \leftarrow t + 1$

5.          $V_t \leftarrow \{h \in V_{t-1} : h(X_m) = Y_m\}$

6.         If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_n$

# Batch-based CAL

Algorithm: **BatchCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.      $V_{i+1} \leftarrow \{h \in V_i : \text{er}_{\mathcal{L}_{i+1}}(h) = 0\}$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$



# Data-dependent Bounds

Let  $\xi_1, \xi_2, \dots$  be iid  $\text{Uniform}(\{-1, +1\})$ .

For  $i \in \mathbb{N}$ , define

$$\hat{R}_{[i+1]}(V_i) = \sup_{g, h \in V_i} \frac{1}{2^{i+1}} \sum_{m=2^i+1}^{2^{i+1}} \xi_m \cdot (g(X_m) - h(X_m)).$$

$$\hat{D}_{[i+1]}(V_i) = \sup_{g, h \in V_i} \frac{1}{2^{i+1}} \sum_{m=2^i+1}^{2^{i+1}} |g(X_m) - h(X_m)|.$$

$$\hat{U}_{[i+1]}(V_i, \alpha) = \hat{K} \cdot \left( \hat{R}_{[i+1]}(V_i) + \sqrt{\hat{D}_{[i+1]}(V_i) \frac{\log(i/\alpha)}{2^i}} + \frac{\log(1/\alpha)}{2^i} \right).$$

(can take  $\hat{K} = 752$  for our purposes)

# Data-dependent Bounds

Lemma: For  $\alpha \in (0, 1)$ ,  $i \geq 0$ , with prob.  $1 - \alpha$ , letting  $\mathcal{Z}_{i+1} = \{(X_{2^i+1}, Y_{2^i+1}), \dots, (X_{2^{i+1}}, Y_{2^{i+1}})\}$ , if  $f = \arg \min_{h \in V_i} \text{er}(h)$ , then  $\forall h \in V_i$ ,

$$\text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{g \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(g) \leq \text{er}(h) - \text{er}(f) + \hat{U}_{[i+1]}(V_i, \alpha).$$

For each  $i \geq 0$ , take  $\alpha = \delta_i = \delta / (4(i+1)^2)$ .

Then this holds for *every*  $i$ , with probability  $1 - \sum_{i=0}^{\infty} \delta / (4(i+1)^2) \geq 1 - \delta/2$ .

# Batch-based CAL

Algorithm: **BatchCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.      $V_{i+1} \leftarrow \{h \in V_i : \text{er}_{\mathcal{L}_{i+1}}(h) = 0\}$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$

# Robust CAL

Algorithm: **RobustCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}$ ,  $t \leftarrow 0$ ,  $i \leftarrow 0$ ,  $\mathcal{L}_1 \leftarrow \{\}$

1. For  $m = 1, 2, \dots$

2. If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ .

3. Request  $Y_m$

4.  $t \leftarrow t + 1$

5.  $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_i \cup \{(X_m, Y_m)\}$

6. If  $m = 2^{i+1}$

7.  $V_{i+1} \leftarrow \left\{ h \in V_i : \text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{h' \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(h') \leq \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$

8.  $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$

9. If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$

By induction, never removes the  $f$  with  $\min \text{er}(f)$ .

# Data-dependent Bounds

For  $i \in \mathbb{N}$ ,  $h, g \in V_i$ ,

$$\text{er}_{\mathcal{Z}_{i+1}}(h) - \text{er}_{\mathcal{Z}_{i+1}}(g)$$

$$= \frac{1}{2^i} \sum_{(X_m, Y_m) \in \mathcal{Z}_{i+1}} \mathbb{I}[h(X_m) \neq Y_m] - \mathbb{I}[g(X_m) \neq Y_m]$$

$$= \frac{1}{2^i} \sum_{\substack{(X_m, Y_m) \in \mathcal{Z}_{i+1}: \\ X_m \in \text{DIS}(V_i)}} \mathbb{I}[h(X_m) \neq Y_m] - \mathbb{I}[g(X_m) \neq Y_m]$$

$$= \frac{1}{2^i} \sum_{(X_m, Y_m) \in \mathcal{L}_{i+1}} \mathbb{I}[h(X_m) \neq Y_m] - \mathbb{I}[g(X_m) \neq Y_m]$$

$$= \frac{|\mathcal{L}_{i+1}|}{2^i} \left( \text{er}_{\mathcal{L}_{i+1}}(h) - \text{er}_{\mathcal{L}_{i+1}}(g) \right).$$

# Robust CAL

Algorithm: **RobustCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.      $V_{i+1} \leftarrow \left\{ h \in V_i : \text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{h' \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(h') \leq \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$

# Robust CAL

Algorithm: **RobustCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.     
$$V_{i+1} \leftarrow \left\{ h \in V_i : \text{er}_{\mathcal{L}_{i+1}}(h) - \min_{h' \in V_i} \text{er}_{\mathcal{L}_{i+1}}(h') \leq (2^i / |\mathcal{L}_{i+1}|) \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$

# Robust CAL

Algorithm: **RobustCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.      $V_{i+1} \leftarrow \left\{ h \in V_i : \text{er}_{\mathcal{L}_{i+1}}(h) - \min_{h' \in V_i} \text{er}_{\mathcal{L}_{i+1}}(h') \leq (2^i / |\mathcal{L}_{i+1}|) \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$



# Robust CAL

- RobustCAL never eliminates  $f = \operatorname{argmin}_{h \in \mathbb{C}} \operatorname{er}(h)$
- But what is its label complexity?

# Abstract Error Bounds

Lemma: For  $\delta \in (0, 1)$ , with prob.  $1 - \delta/2$ ,  $\forall i \geq 0$ ,  
if  $f = \arg \min_{h \in \mathbb{C}} \text{er}(h)$  and  $\delta_i = \delta / (4(i+1)^2)$ ,  $\forall h \in V_i$ ,

$$\text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{g \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(g) \leq \text{er}(h) - \text{er}(f) + \hat{U}_{[i+1]}(V_i, \delta_i)$$

$$\text{er}(h) - \text{er}(f) \leq \text{er}_{\mathcal{Z}_{i+1}}(h) - \text{er}_{\mathcal{Z}_{i+1}}(f) + \hat{U}_{[i+1]}(V_i, \delta_i)$$

$$\hat{U}_{[i+1]}(V_i, \delta_i) \wedge 1 \leq \tilde{\mathcal{E}}_{2^i}(V_i, \delta_i).$$

$$\text{er}_{\mathcal{Z}_{i+1}}(h) - \text{er}_{\mathcal{Z}_{i+1}}(g) = \frac{|\mathcal{L}_{i+1}|}{2^i} \left( \text{er}_{\mathcal{L}_{i+1}}(h) - \text{er}_{\mathcal{L}_{i+1}}(g) \right).$$

# Abstract Error Bounds

Lemma: For  $\delta \in (0, 1)$ , with prob.  $1 - \delta/2$ ,  $\forall i \geq 0$ ,  
if  $f = \arg \min_{h \in \mathbb{C}} \text{er}(h)$  and  $\delta_i = \delta / (4(i+1)^2)$ ,  $\forall h \in V_i$ ,

$$\text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{g \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(g) \leq \text{er}(h) - \text{er}(f) + \hat{U}_{[i+1]}(V_i, \delta_i)$$

$$\text{er}(h) - \text{er}(f) \leq \text{er}_{\mathcal{Z}_{i+1}}(h) - \text{er}_{\mathcal{Z}_{i+1}}(f) + \hat{U}_{[i+1]}(V_i, \delta_i)$$

$$\hat{U}_{[i+1]}(V_i, \delta_i) \wedge 1 \leq \tilde{\mathcal{E}}_{2^i}(V_i, \delta_i).$$

If  $\text{diam}(V_i) > \log(i/\delta)/2^i$ , then

$$\tilde{\mathcal{E}}_{2^i}(V_i, \delta_i) \leq c \sqrt{\frac{\text{diam}(V_i) (id + \log(1/\delta))}{2^i}}$$

for some constant  $c \in [1, \infty)$ .

# Robust CAL

**Algorithm: RobustCAL**( $\mathbb{C}, n$ )

0.  $V_0 \leftarrow \mathbb{C}, t \leftarrow 0, i \leftarrow 0, \mathcal{L}_1 \leftarrow \{\}$
1. For  $m = 1, 2, \dots$
2.   If  $\exists h_1, h_2 \in V_i$  s.t.  $h_1(X_m) \neq h_2(X_m)$ ,
3.     Request  $Y_m$
4.      $t \leftarrow t + 1$
5.      $\mathcal{L}_{i+1} \leftarrow \mathcal{L}_{i+1} \cup \{(X_m, Y_m)\}$
6.   If  $m = 2^{i+1}$
7.      $V_{i+1} \leftarrow \left\{ h \in V_i : \text{er}_{\mathcal{L}_{i+1}}(h) - \min_{h' \in V_i} \text{er}_{\mathcal{L}_{i+1}}(h') \right.$   
 $\qquad \qquad \qquad \left. \leq (2^i / |\mathcal{L}_{i+1}|) \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$
8.      $i \leftarrow i + 1$ , and  $\mathcal{L}_{i+1} \leftarrow \{\}$
9.   If  $t = n$ , Return an arbitrary classifier  $\hat{h}_n \in V_i$

# Label Complexity of Robust CAL

Suppose  $\mathcal{D}_{XY} \in \mathcal{Tsybakov}(\mu, \kappa, \mathbb{C})$   
and  $f = \arg \min_{h \in \mathbb{C}} \text{er}(h)$ .

**Theorem:** For a  $(\mu, \kappa)$ -dependent constant  $c_1 \in (0, \infty)$ , RobustCAL achieves label complexity

$$\Lambda(\varepsilon + \text{er}(f), \delta, \mathcal{D}_{XY}) = c_1 \cdot \theta_f \left( \varepsilon^{\frac{1}{\kappa}} \right) \varepsilon^{\frac{2}{\kappa} - 2} d \log^2 \left( \frac{d}{\varepsilon \delta} \right).$$

Recall passive learning has  $\tilde{O} \left( \varepsilon^{\frac{1}{\kappa} - 2} \right)$ .

# Label Complexity Analysis: Sketch

Suppose the  $1 - \delta/2$  prob. event (from the lemma).

Claim 1:  $\forall i, f \in V_i$ . (proved already)

Claim 2: For some constant  $a \in [1, \infty)$ ,

$$\forall i, V_i \subseteq \mathbb{C} \left( a \cdot \left( \frac{di + \log(1/\delta)}{2^i} \right)^{\frac{\kappa}{2\kappa - 1}} \right).$$

Claim 3: For any  $n \geq c_1 \theta_f \varepsilon^{\frac{2}{\kappa} - 2} \log^2 \frac{d}{\varepsilon \delta}$ ,

the final value of  $i$  is greater than

$$I = \left\lceil \left(2 - \frac{1}{\kappa}\right) \log_2 \left(\frac{c_2}{\varepsilon}\right) + \log_2 \left(c_3 d \log \left(\frac{c_4 d}{\varepsilon \delta}\right)\right) \right\rceil.$$

Therefore,  $V_i \subseteq V_I \subseteq \mathbb{C}(\varepsilon)$ .

# Label Complexity Analysis: Sketch

Claim 2: For some constant  $a \in [1, \infty)$ ,

$$\forall i, V_i \subseteq \mathbb{C} \left( a \cdot \left( \frac{di + \log(1/\delta)}{2^i} \right)^{\frac{\kappa}{2\kappa-1}} \right).$$

Proof Sketch: (Induction) (base case  $i = 1$ )  
 Suppose true for  $i$ . Then Tsybakov condition

$$\text{diam}(V_i) \leq k_1 \cdot \left( \frac{di + \log(1/\delta)}{2^i} \right)^{\frac{1}{2\kappa-1}}.$$

Then  $V_{i+1} =$

$$\left\{ h \in V_i : \text{er}_{\mathcal{Z}_{i+1}}(h) - \min_{g \in V_i} \text{er}_{\mathcal{Z}_{i+1}}(g) \leq \hat{U}_{[i+1]}(V_i, \delta_i) \right\}$$

$$\subseteq \{ h \in V_i : \text{er}(h) - \text{er}(f) \leq 2\hat{U}_{[i+1]}(V_i, \delta_i) \}$$

$$\subseteq \mathbb{C} \left( 2\tilde{\mathcal{E}}_{2^i}(V_i, \delta_i) \right) \subseteq \mathbb{C} \left( k_2 \sqrt{\frac{\text{diam}(V_i)(di + \log(1/\delta))}{2^i}} \right)$$

$$\subseteq \mathbb{C} \left( k_3 \cdot \left( \frac{d(i+1) + \log(1/\delta)}{2^{i+1}} \right)^{\frac{\kappa}{2\kappa-1}} \right) \text{ Take } a = k_3 \text{ (solve for } a)$$

# Label Complexity Analysis: Sketch

Suppose the  $1 - \delta/2$  prob. event (from the lemma).

Claim 1:  $\forall i, f \in V_i$ . (proved already)

Claim 2: For some constant  $a \in [1, \infty)$ ,

$$\forall i, V_i \subseteq \mathbb{C} \left( a \cdot \left( \frac{di + \log(1/\delta)}{2^i} \right)^{\frac{\kappa}{2\kappa - 1}} \right).$$

Claim 3: For any  $n \geq c_1 \theta_f \varepsilon^{\frac{2}{\kappa} - 2} \log^2 \frac{d}{\varepsilon \delta}$ ,

the final value of  $i$  is greater than

$$I = \left\lceil \left(2 - \frac{1}{\kappa}\right) \log_2 \left(\frac{c_2}{\varepsilon}\right) + \log_2 \left(c_3 d \log \left(\frac{c_4 d}{\varepsilon \delta}\right)\right) \right\rceil.$$

Therefore,  $V_i \subseteq V_I \subseteq \mathbb{C}(\varepsilon)$ .



# Label Complexity Analysis: Sketch

Claim 3: For any  $n \geq c_1 \theta_f \varepsilon^{\frac{2}{\kappa}-2} \log^2 \frac{d}{\varepsilon \delta}$ ,

the final value of  $i$  is greater than

$$I = \left\lceil \left(2 - \frac{1}{\kappa}\right) \log_2 \left(\frac{c_2}{\varepsilon}\right) + \log_2 \left(c_3 d \log \left(\frac{c_4 d}{\varepsilon \delta}\right)\right) \right\rceil.$$

Proof Sketch:

Take  $i < I$ . Note

$$\mathbb{E} \left[ |\mathcal{L}_{i+1}| \middle| V_i \right] = \sum_{m=2^i+1}^{2^{i+1}} \mathbb{P}(X_m \in \text{DIS}(V_i) | V_i)$$

$$= 2^i \mathbb{P}(\text{DIS}(V_i)) \leq 2^i \theta_f \cdot \text{diam}(V_i)$$

$$\leq c_5 \theta_f 2^i \left( \frac{di + \log(1/\delta)}{2^i} \right)^{\frac{1}{2\kappa-1}}$$

$$= c_5 \theta_f 2^{i \frac{2\kappa-2}{2\kappa-1}} \cdot (di + \log(1/\delta))^{\frac{1}{2\kappa-1}}$$

$$\leq c_5 \theta_f 2^{I \frac{2\kappa-2}{2\kappa-1}} \cdot (dI + \log(1/\delta))^{\frac{1}{2\kappa-1}}$$

$$\leq c_6 \theta_f \cdot \varepsilon^{\frac{2}{\kappa}-2} \cdot d \log \frac{d}{\varepsilon \delta}.$$

# Label Complexity Analysis: Sketch

By Chernoff bound, with probability  $1 - \delta/2$ ,  $\forall i < I$ ,  
 $|\mathcal{L}_{i+1}| \leq c_7 \theta_f \cdot \varepsilon^{\frac{2}{\kappa}-2} \cdot d \log \frac{d}{\varepsilon \delta}.$

So  $n \geq c_7 \theta_f \cdot \varepsilon^{\frac{2}{\kappa}-2} \cdot d \log \frac{d}{\varepsilon \delta} \cdot I$   
suffices to guarantee final  $i \geq I$ .

$$c_7 \theta_f \cdot \varepsilon^{\frac{2}{\kappa}-2} \cdot d \log \frac{d}{\varepsilon \delta} \cdot I \leq c_8 \theta_f \cdot \varepsilon^{\frac{2}{\kappa}-2} \cdot d \log^2 \frac{d}{\varepsilon \delta}. \quad \square$$

# Current Hot Directions

- Efficiency [DHM07,BDL09,BHLZ10]
- Improvements over the disagreement-based approach [Das05,BBZ07,BHV10,Han11]

# The End