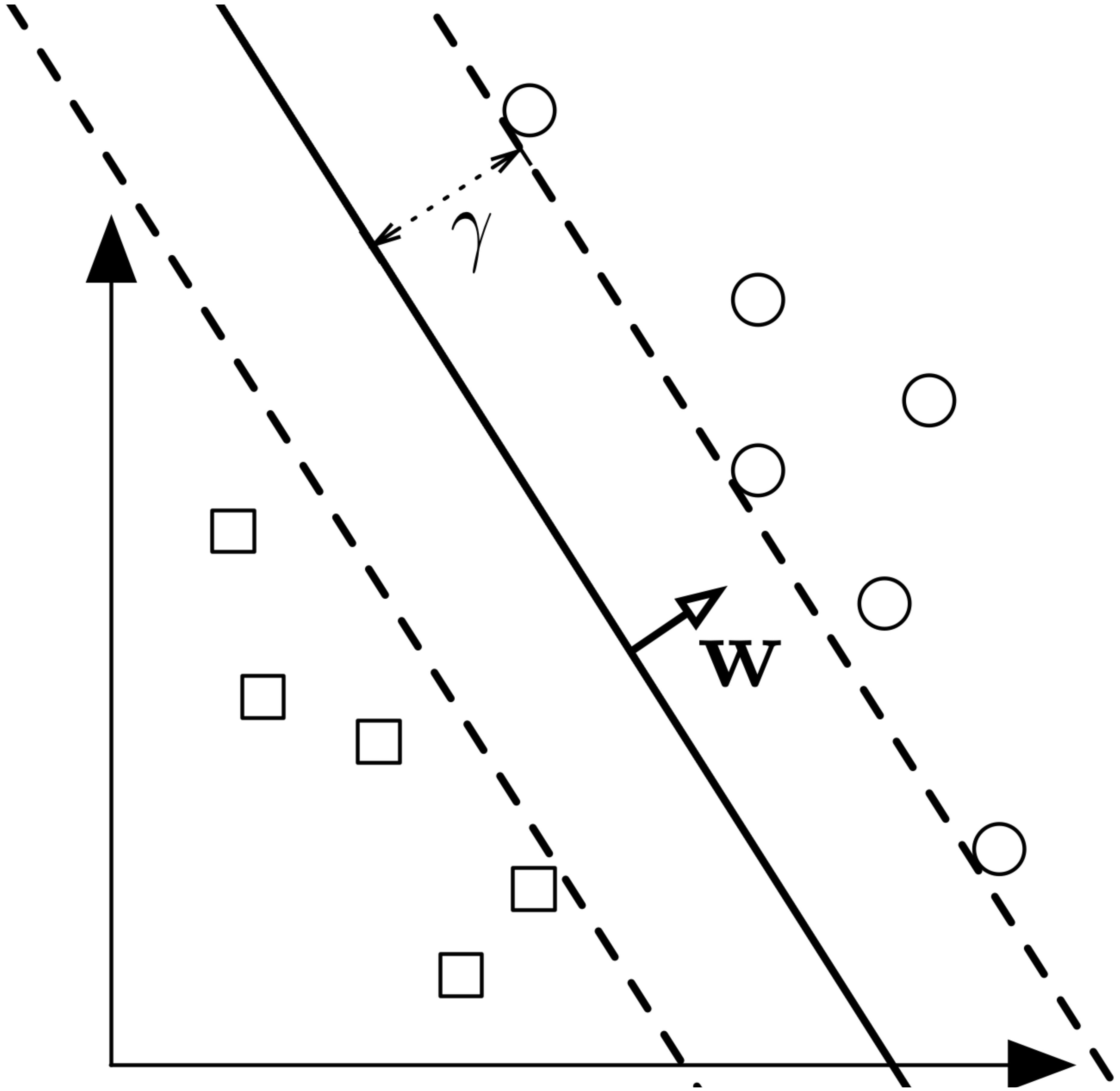
Support Vector Machine (continue)

Announcements

1. Prelim Conflict form is out and due next Tue

2. P4 is going to be out this afternoon (due after prelim)

SVMs



Goal of SVM: find a hyperplane that (1) separates the data, (2) $\gamma(w, b)$ is maximized

The SVM algorithm

Avoids "cheating" (i.e., scale w, b up by large constant)

$$\boxed{\min_{w,b} \|w\|_2^2}$$

$$\boxed{\forall i: \; y_i(w^\top x_i + b) \geq 1}$$

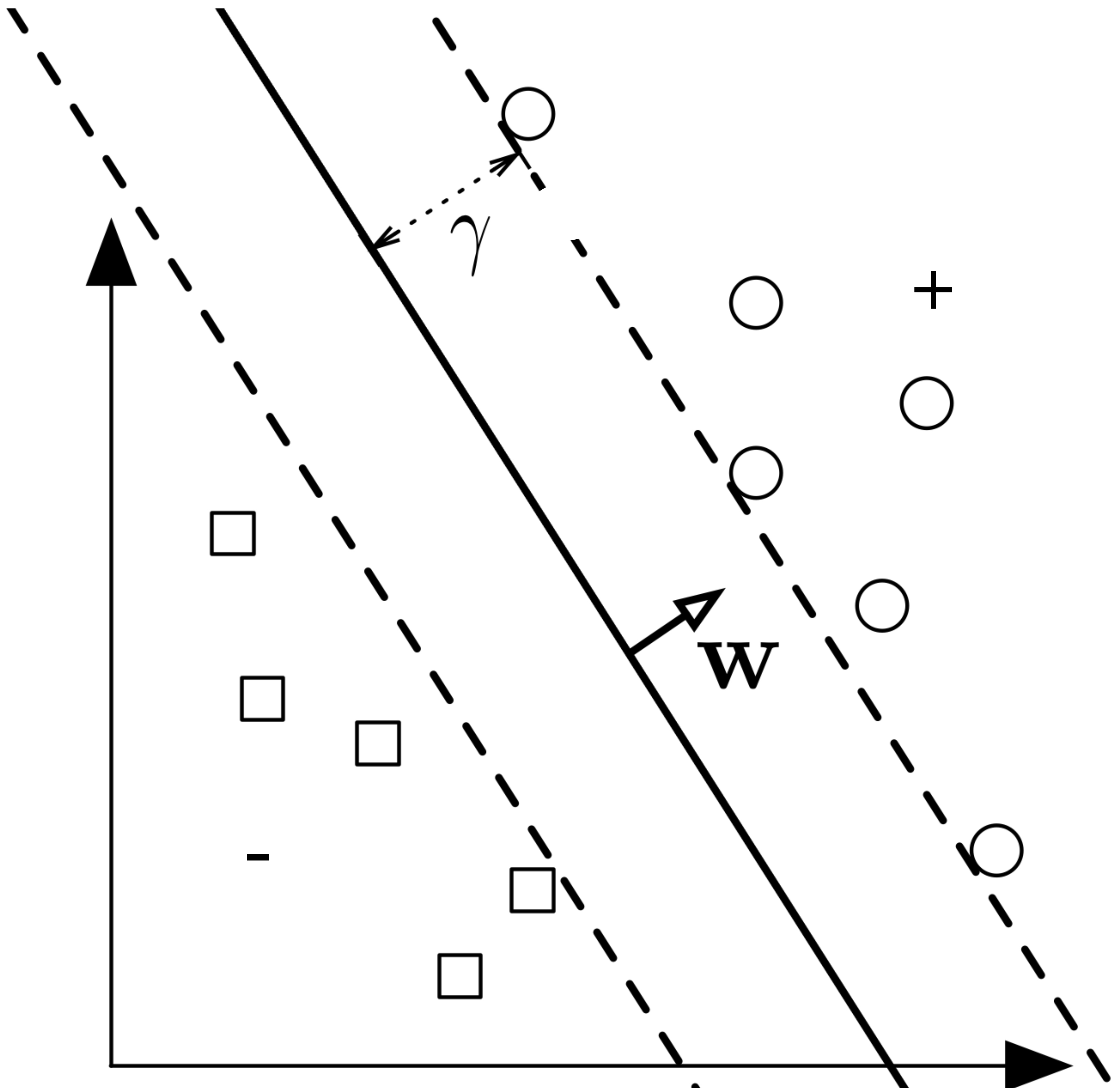
Not only linearly separable, but also has functional margin no less than 1

Denote (w, b) as the optimal solution:

Q: will there be some (x, y), such that $y(w^\top x + b) = 1$?

Support Vectors

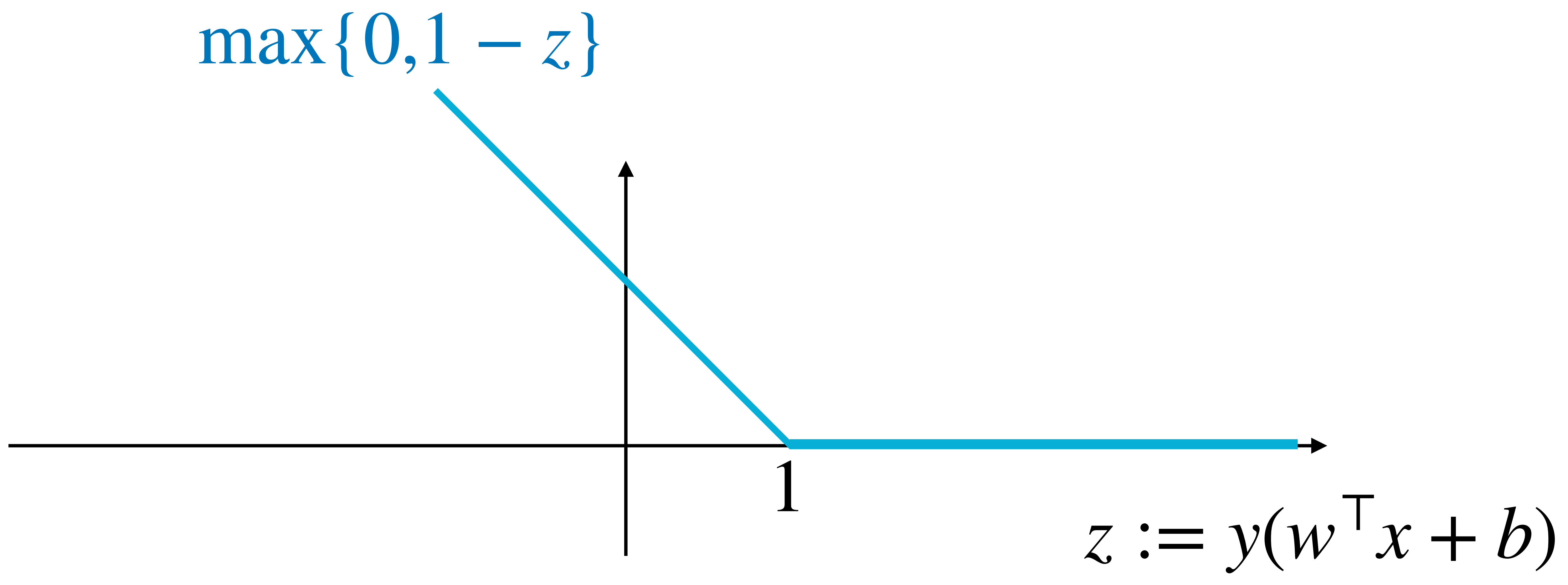
Points x_i such that $y_i(w^\top x_i + b) = 1$ are called **support vectors**



SVM for non-separable data

$$\min_{w,b} \|w\|_2^2 + c\sum_{i=1}^{n} \boxed{\max\left\{0, 1 - y_i(w^\top x_i + b)\right\}}$$

Hinge loss



Hinge loss starts penalizing when functional margin falls below 1

SVM for non-separable data

$$\min_{w,b} \|w\|_2^2 + c \sum_{i=1}^{n} \max\left\{0, 1 - y_i(w^\top x_i + b)\right\}$$

Trades off $\|w\|_2^2$ and functional margins over data

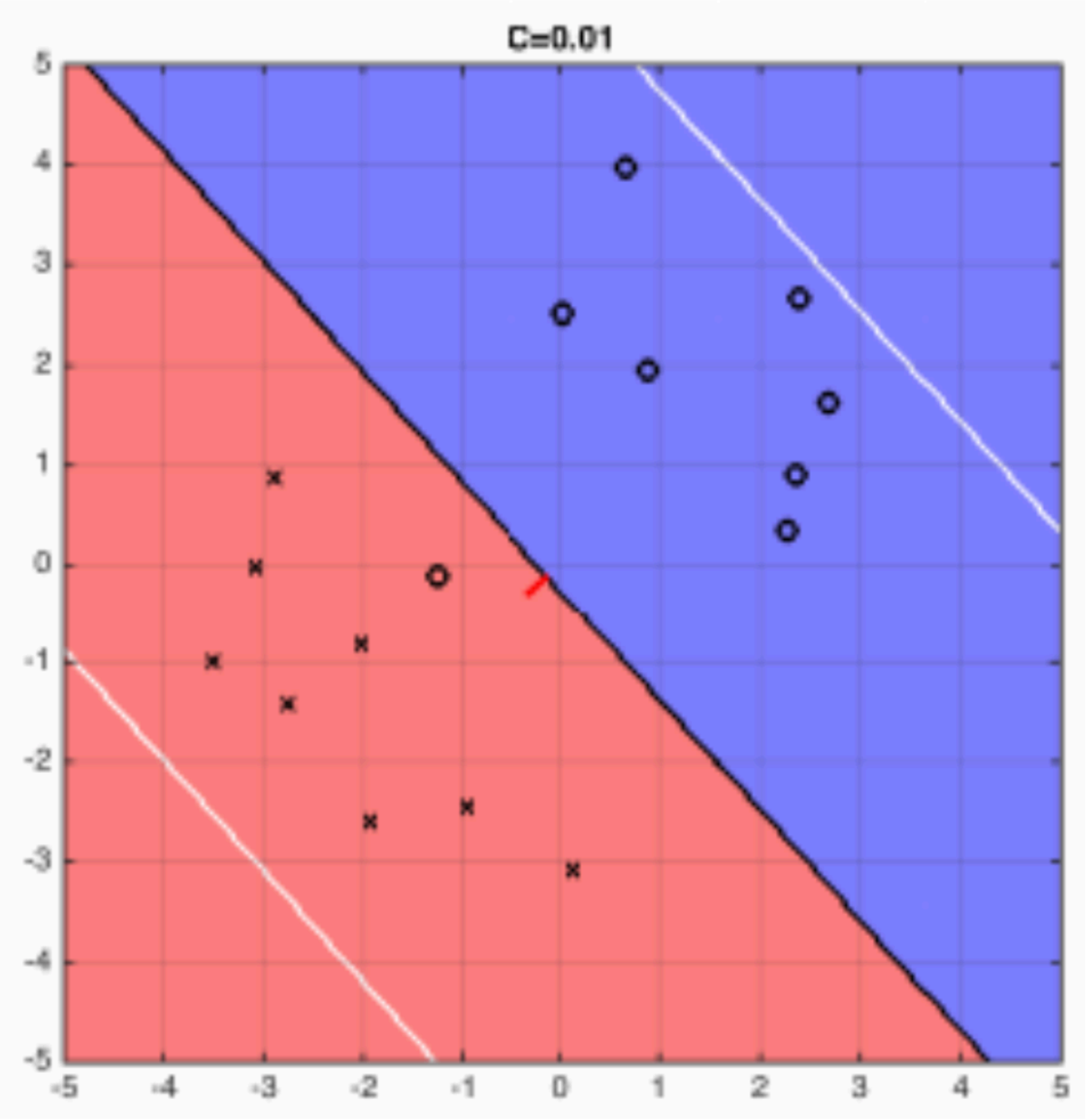
When $c \to +\infty$:

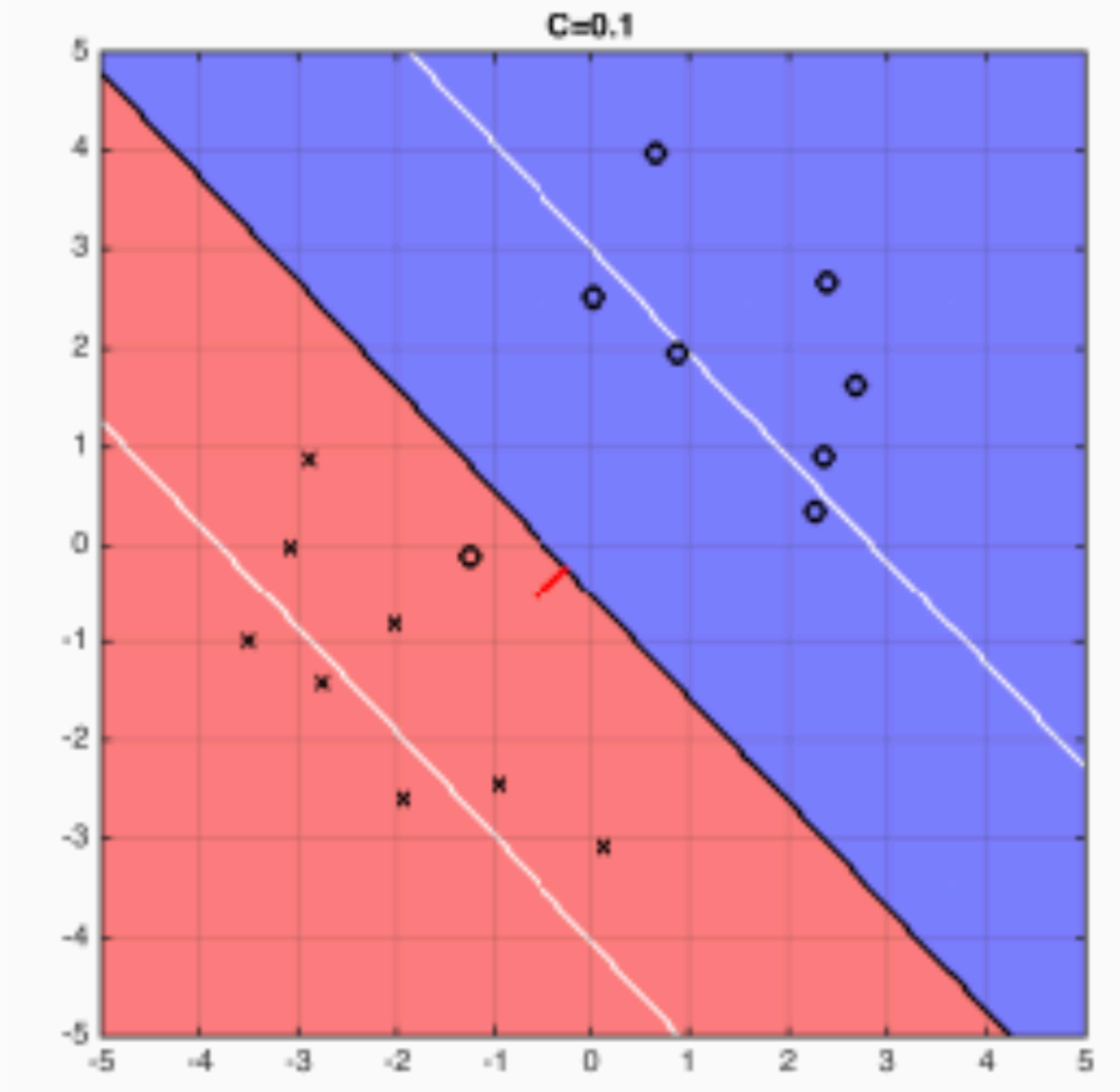
forcing $y_i(w^\top x_i + b) \geq 1$ for as many data points as possible

When $c \to 0^+$:

The solution $w \to \mathbf{0}$ (i.e., we do not care about hinge loss part)

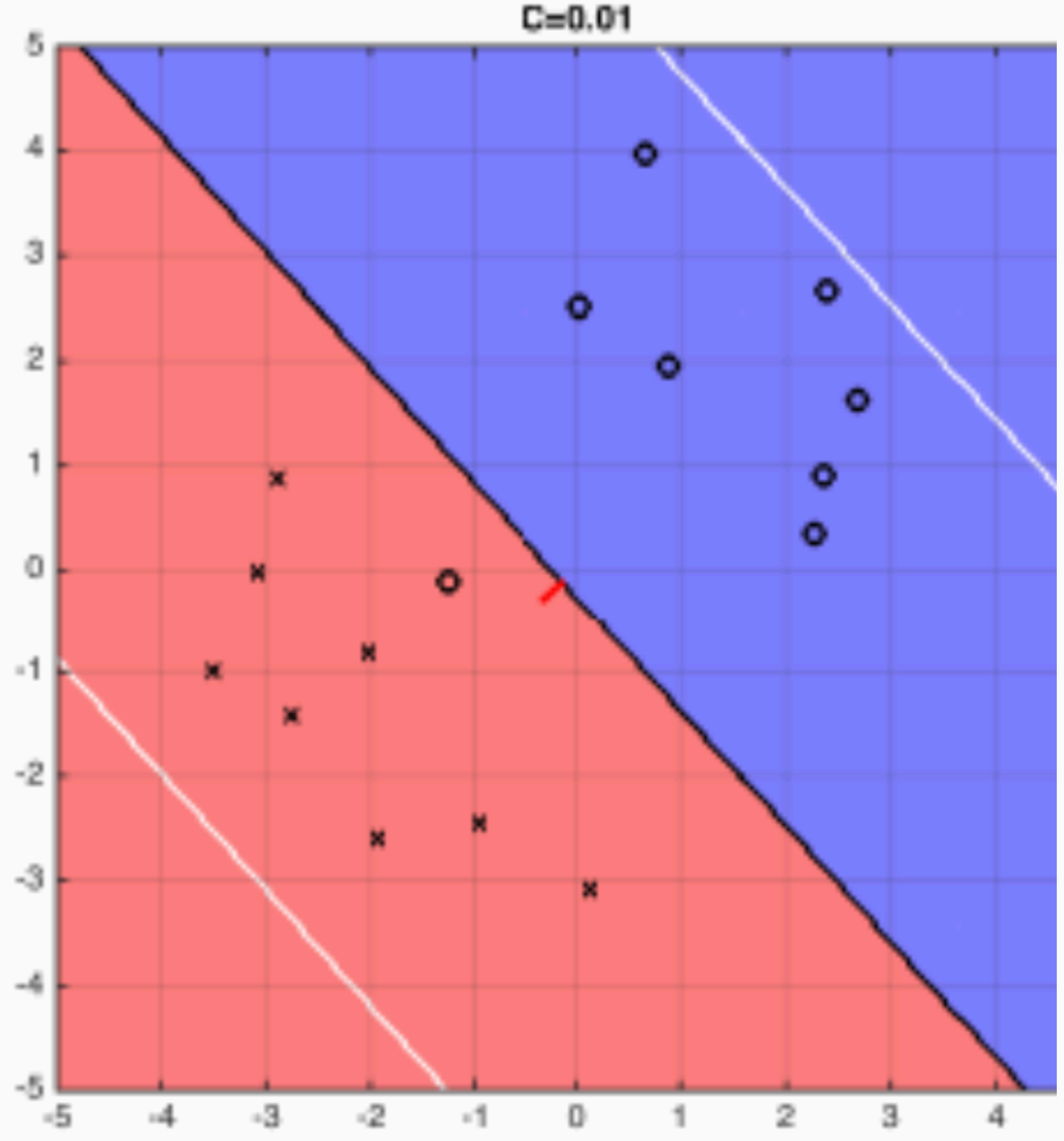
$$\min_{w,b} \|w\|_2^2 + c \sum_{i=1}^{n} \max\left\{0, 1 - y_i(w^\top x_i + b)\right\}$$



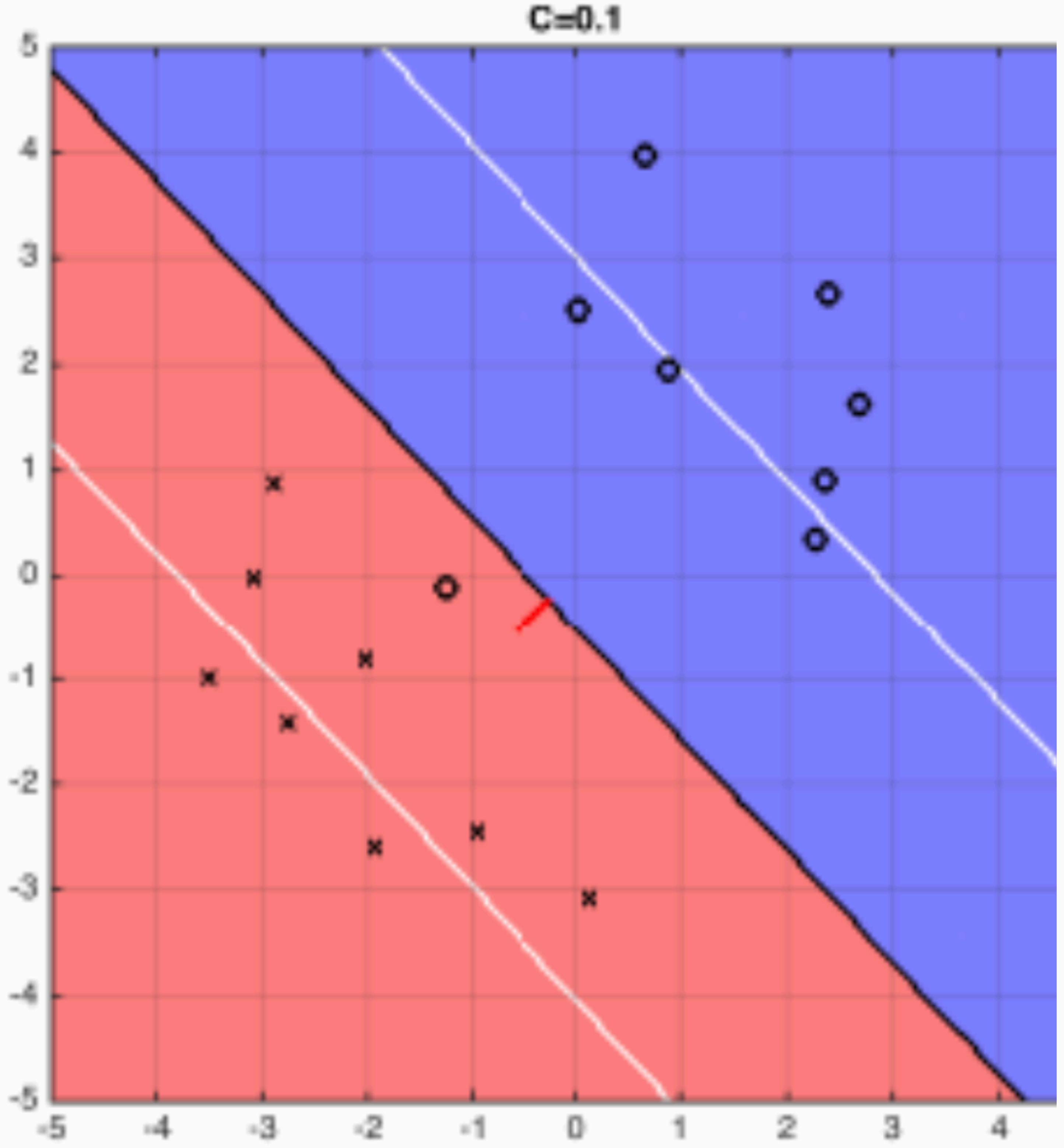


$$\min_{w,b} \|w\|_2^2 + c \sum_{i=1}^{n} \max \left\{0, 1 - y_i(w^\top x_i + b)\right\}$$

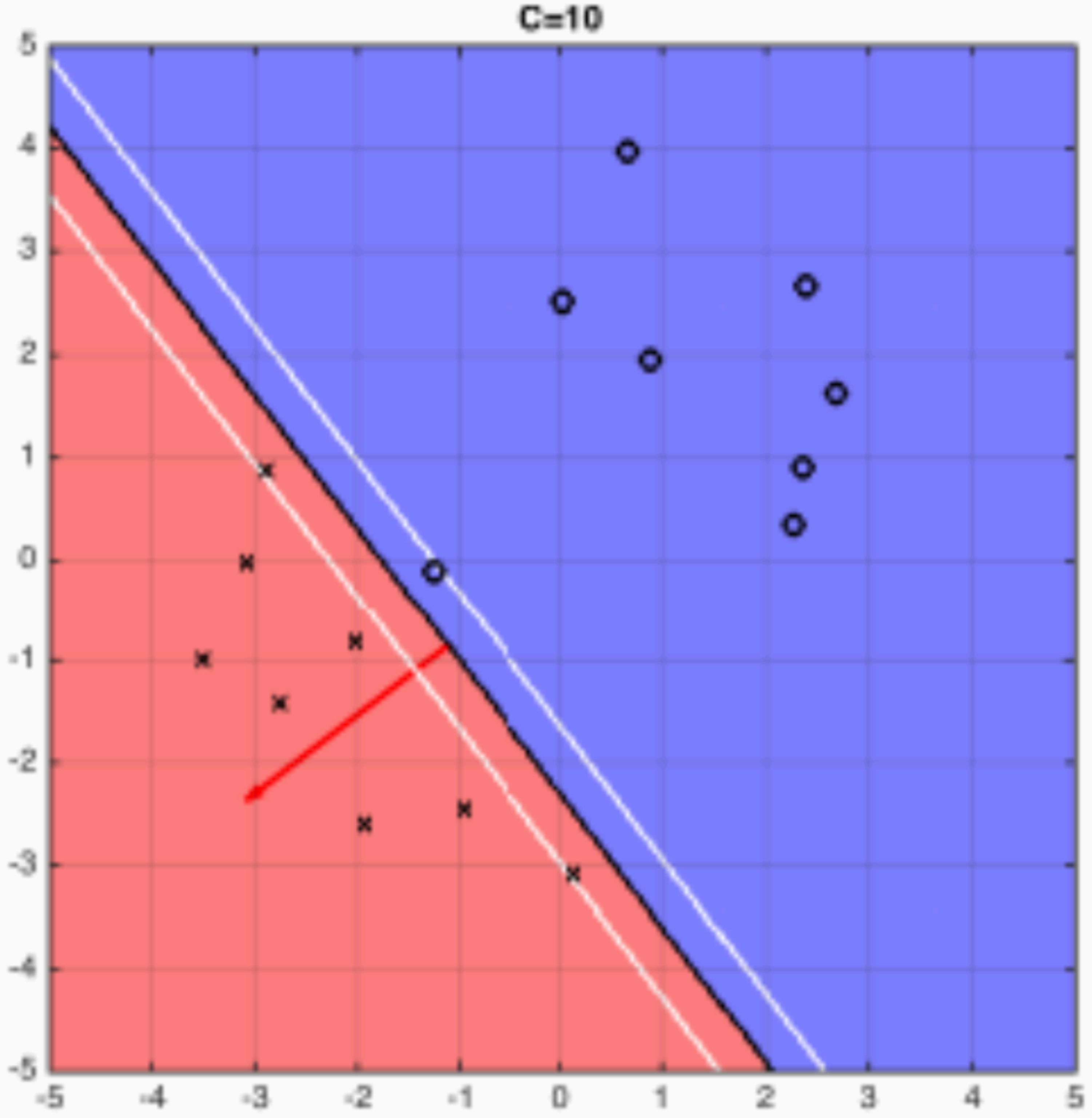
C = 0.01



C = 0.1



C =10

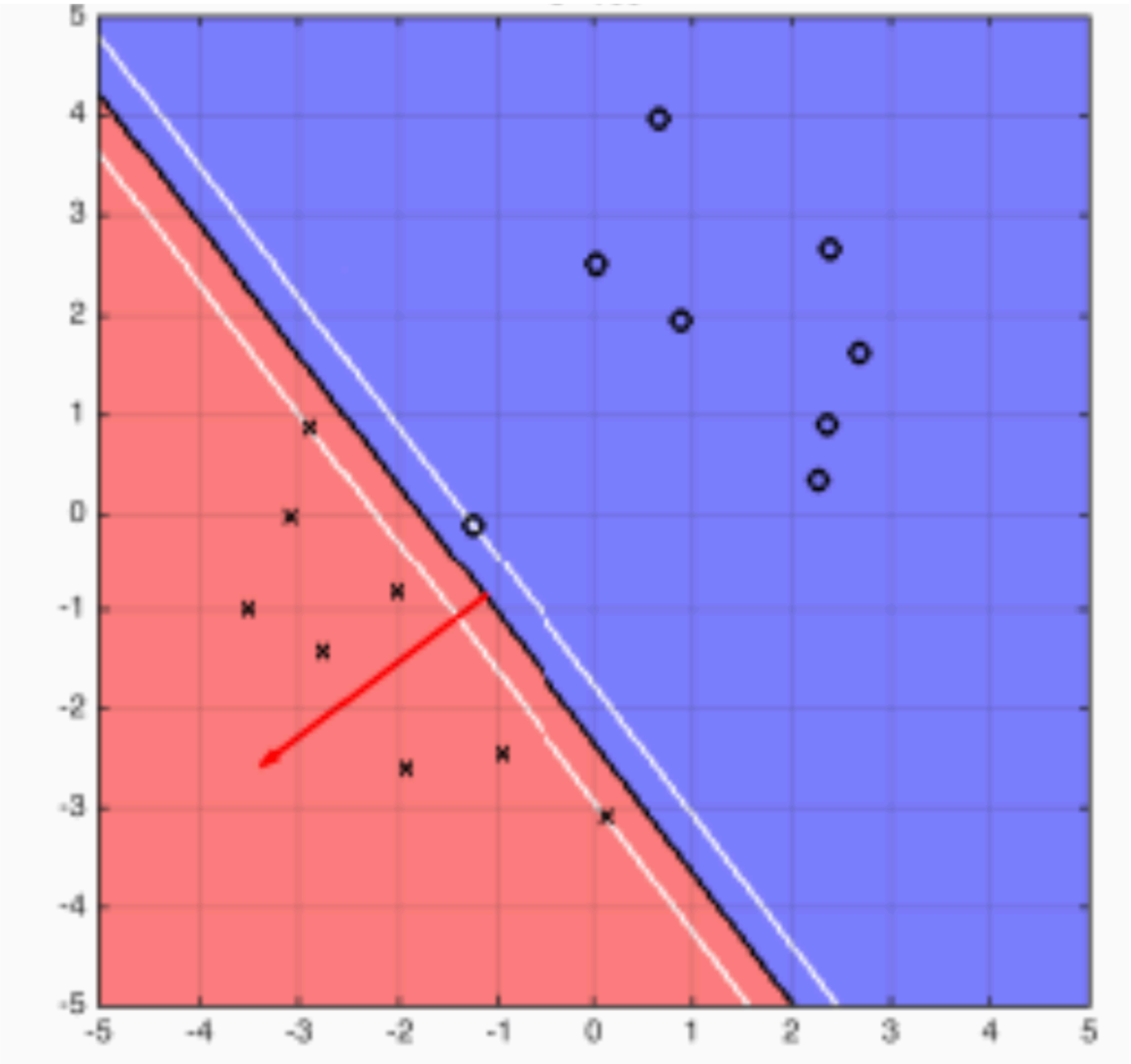


SVM for non-separable data

$$\min_{w,b} \|w\|_2^2 + c \sum_{i=1}^{n} \max\left\{0, 1 - y_i(w^\top x_i + b)\right\}$$

Trades off $\|w\|_2^2$ and functional margins over data

C = 100



all examples have zero Hinge loss, but

w has large norm

Bad geometric margin but good functional margin (achieved by “cheating”)

Potentially overfitting to the noise, not a good classifier in test time maybe

Empirical Risk Minimization

ERM

Recall the general supervised learning setting:

We have some distribution P, dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$

Each data point is i.i.d sampled from P, i.e., $x_i, y_i \sim P$

Hypothesis $h : \mathscr{X} \to \mathbb{R}$, & hypothesis class $\mathscr{H} := \{h\} \subset \mathscr{X} \mapsto \mathbb{R}$

Loss function: $\ell(h(x), y)$

ERM

The ultimate objective function:

$$\arg\min_{h \in \mathcal{H}} \mathbb{E}_{x,y \sim P} [\ell(h(x), y)]$$

Unknown

Instead we have its **empirical** version

$$\arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} [\ell(h(x_i), y_i)]$$

Empirical risk / Empirical error

The generalization error of ERM solution

$$\hat{h}_{ERM} := \arg\min_{h \in \mathscr{H}} \frac{1}{n} \sum_{i=1}^{n} \left[\ell(h(x_i), y_i) \right]$$

We often are interested in the true performance of $\hat{h}_{ERM}$:

$$\mathbb{E}_{\mathscr{D}} \left[\mathbb{E}_{x,y \sim P} \ell(\hat{h}_{ERM}(x), y) \right]$$

Note $\hat{h}_{ERM}$ is a random quantity as
it depends on data $\mathscr{D}$

e.g., In LR: $\hat{w} = (XX^\top)^{-1} XY$.

The generalization error of ERM solution

Ideally, we want the true loss of ERM to be small:

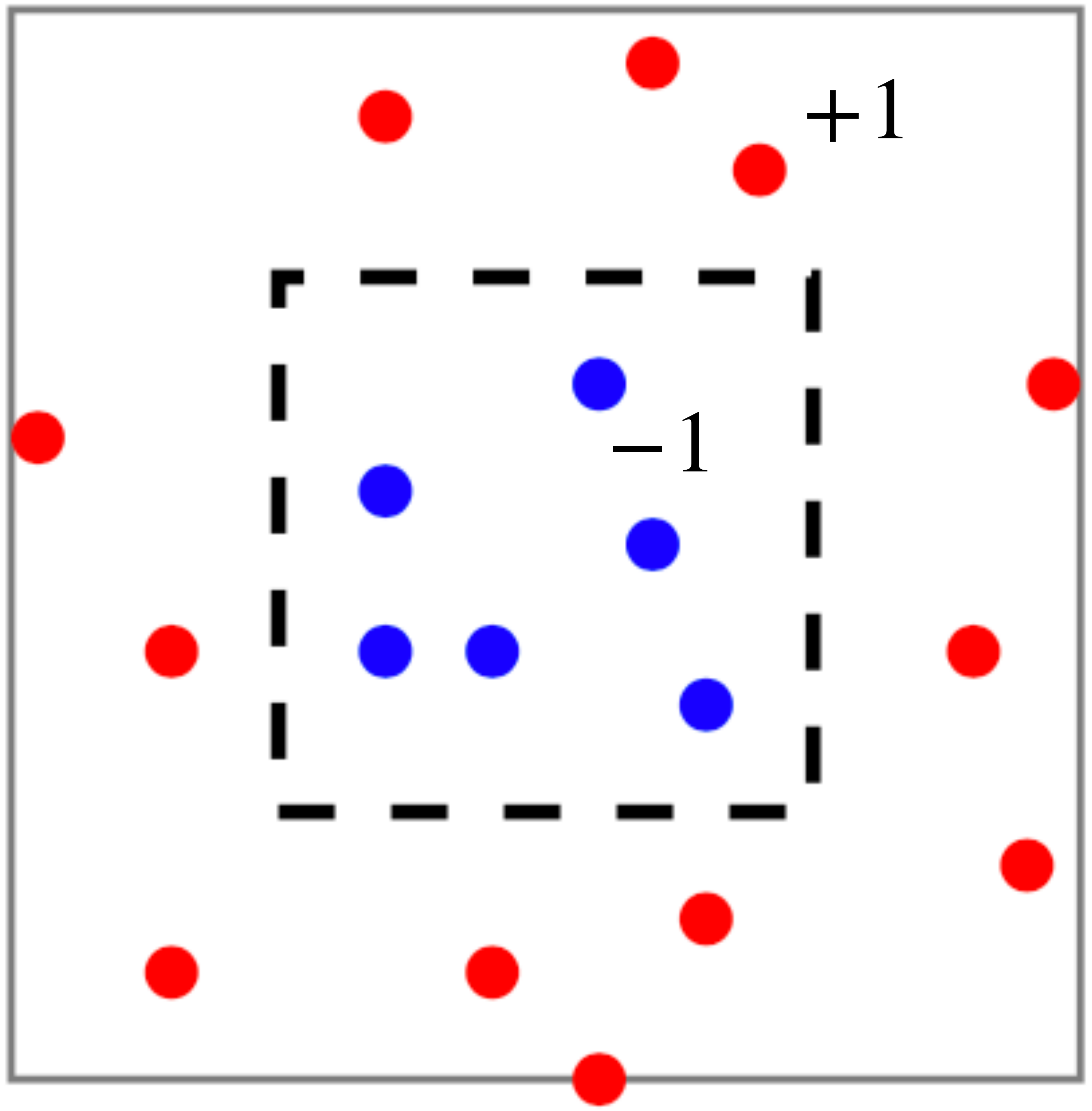
$$\mathbb{E}_{\mathscr{D}}\left[\mathbb{E}_{x,y\sim P}\ell(\hat{h}_{ERM}(x), y)\right] \approx \underbrace{\min_{h\in\mathscr{H}} \mathbb{E}_{x,y\sim P}\ell(h(x), y)}$$

The Minimum expected loss we could get if we knew P

However, this may not hold if we are not careful about designing $\mathscr{H}$

Example:

P: x uniformly distribution over the square;
Label: blue if inside the smaller square, else red



Consider a hypothesis class $\mathscr{H}$ contains ALL mappings from $x \to y$

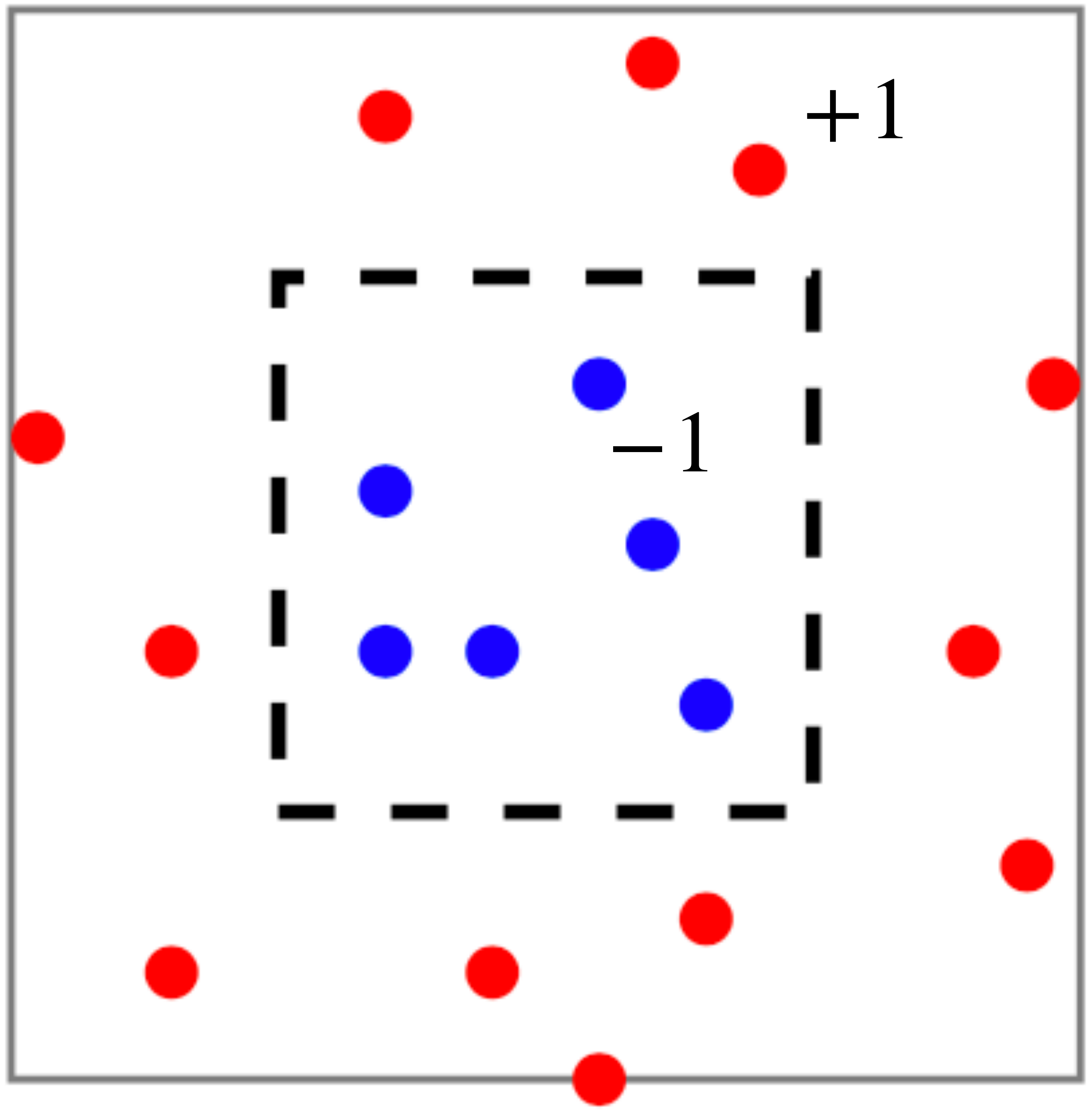
Zero one loss $\ell(h(x), y) = \mathbf{1}(h(x) \neq y)$

Let us consider this solution that memorizes data:

$$\hat{h}(x) = \begin{cases} y_i & \text{if } \exists i, x_i = x \\ +1 & \text{else} \end{cases}$$

Example:

P: x uniformly distribution over the square;
Label: blue if inside the dashed square, else red



$$\hat{h}(x) = \begin{cases} y_i & \text{if } \exists i, x_i = x \\ +1 & \text{else} \end{cases}$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} \ell(\hat{h}(x_i), y_i) = 0$$

Q: But what's the true expected error of this $\hat{h}$?

A: area of smaller box / total area

ERM with inductive bias

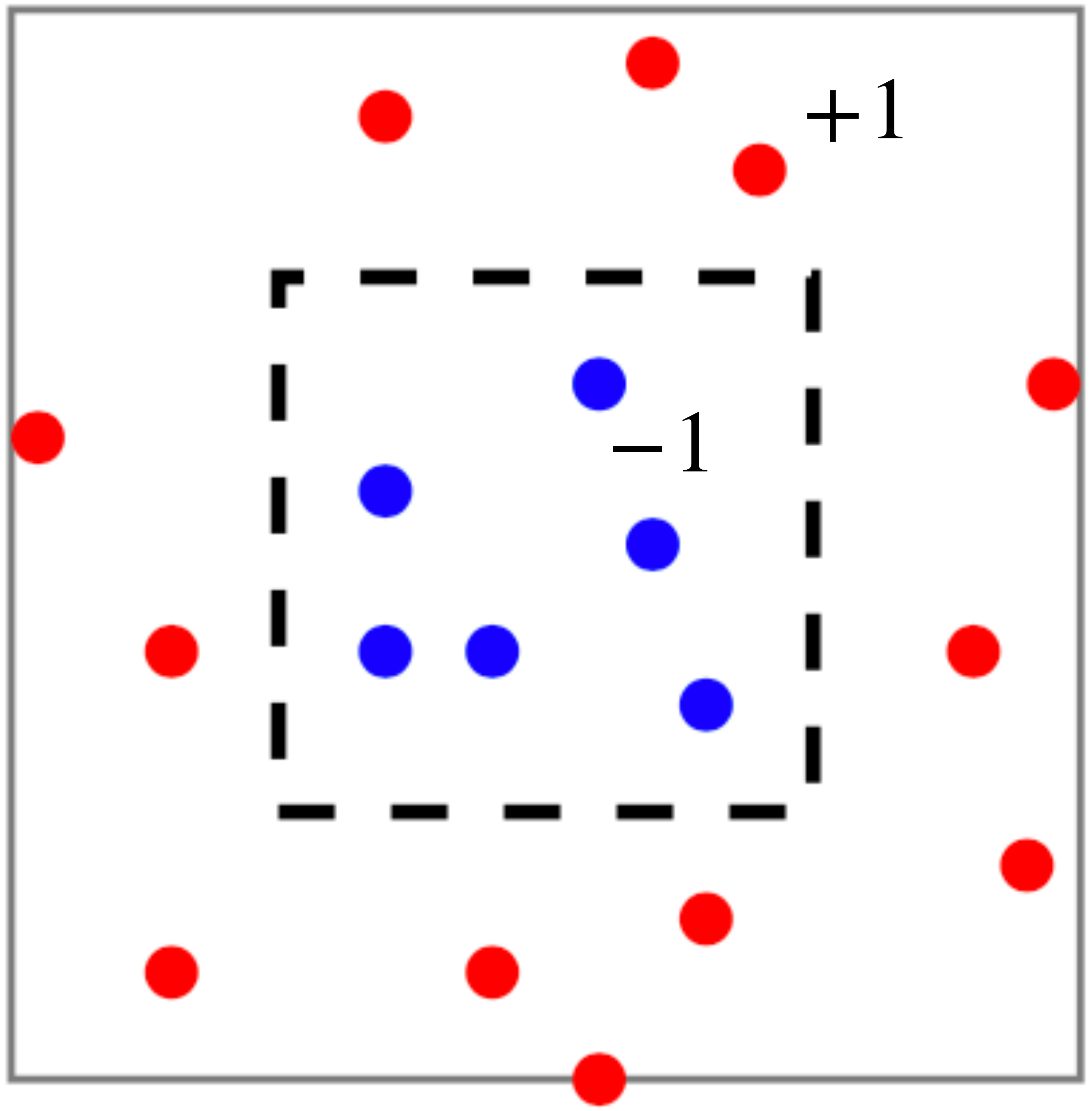
A common solution is to restrict the search space (i.e., hypothesis class)

$$\hat{h}_{ERM} := \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \left[\ell(h(x_i), y_i) \right]$$

By restricting to $\mathcal{H}$, we bias towards solutions from $\mathcal{H}$

Example:

P: x uniformly distribution over the square;
Label: blue if inside the dashed square, else red



Unrestricted hypothesis class did not work;

However, if we restrict $\mathscr{H}$ to contains ALL axis-aligned rectangles,
then ERM will succeed, i.e.,

$$\mathbb{E}_{\mathscr{D}}\left[\mathbb{E}_{x,y\sim P}\ell(\hat{h}_{ERM}(x), y)\right]$$

$$\leq \min_{h\in\mathscr{H}} \mathbb{E}_{x,y\sim P}\ell(h(x), y) + O(1/\sqrt{n})$$

$$\leq O(1/\sqrt{n})$$

(Exact proof out of the scope of this class — see CS 4783/5783)

Summary

ERM with unrestricted hypothesis class could fail (i.e., overfitting)

To guarantee small test error, we need to restrict $\mathscr{H}$

After Prelim

We will continue from ERM:

Examples of loss functions,
ways to restrict the hypothesis classes,
why that really matters in ML (theory and practice)