Applied Machine Learning

Linear Support Vector Machine

Computer Science, Fall 2022

Instructor: Xuhong Zhang

SVM: Introduction

- It is a state-of-the-art classification method introduced in 1992.
- Very widely used in bioinformatics (and other disciplines)
  - High accuracy
  - Ability to deal with high-dimensional data
- It belongs to the general category of kernel methods

Equation of a hyperplane

- In coordinate space $\mathbb{R}^d$ equation

$$\langle w, x \rangle + b = 0$$

$$\sum_{i=1}^{d} w^i x^i + b = 0$$

defines a $(d-1)$ dimensional set of vectors called ***hyperplane***.

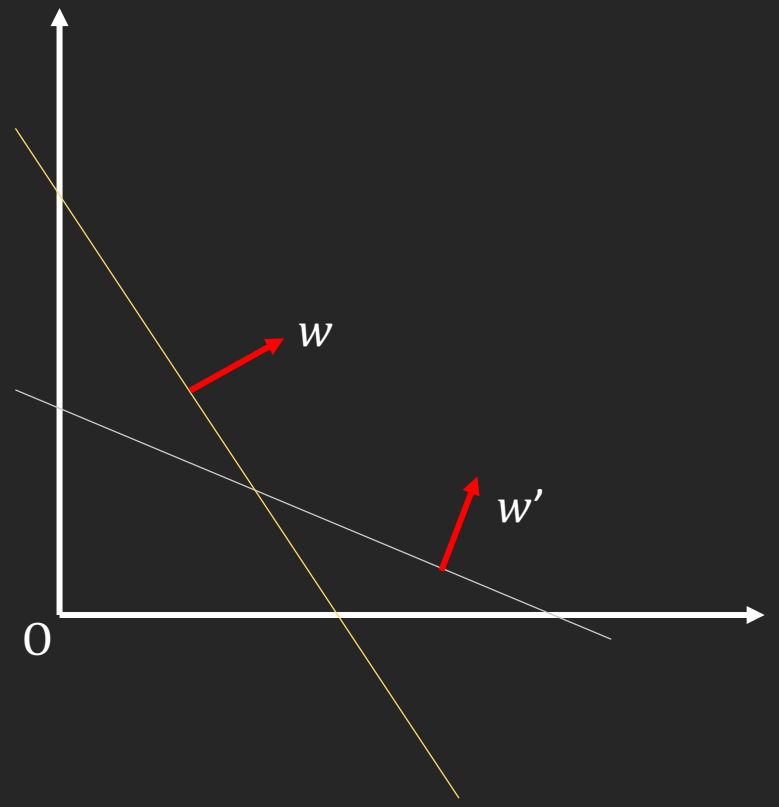
- That is, for a given non-zero vector $\mathrm{w} = (w^1, w^2, \ldots, w^d) \in \mathbb{R}^d$ and a scalar $b \in \mathbb{R}$, the set of all vectors $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^d$ satisfying the above equation forms a hyperplane. Denote the hyperplane by (w, b) and w is the normal vector, and b is the intercept.

Equation of a hyperplane

- Remarks
  - The term "hyperplane" means that the dimensionality of the plane is by one less than the dimensionality of the entire space $\mathbb{R}^d$.
  - a point is a hyperplane in $\mathbb{R}$.
  - a line is a hyperplane in $\mathbb{R}^2$.
  - a plane is a hyperplane in $\mathbb{R}^3$.
  - a three-dimensional space is a hyperplane in $\mathbb{R}^4$.

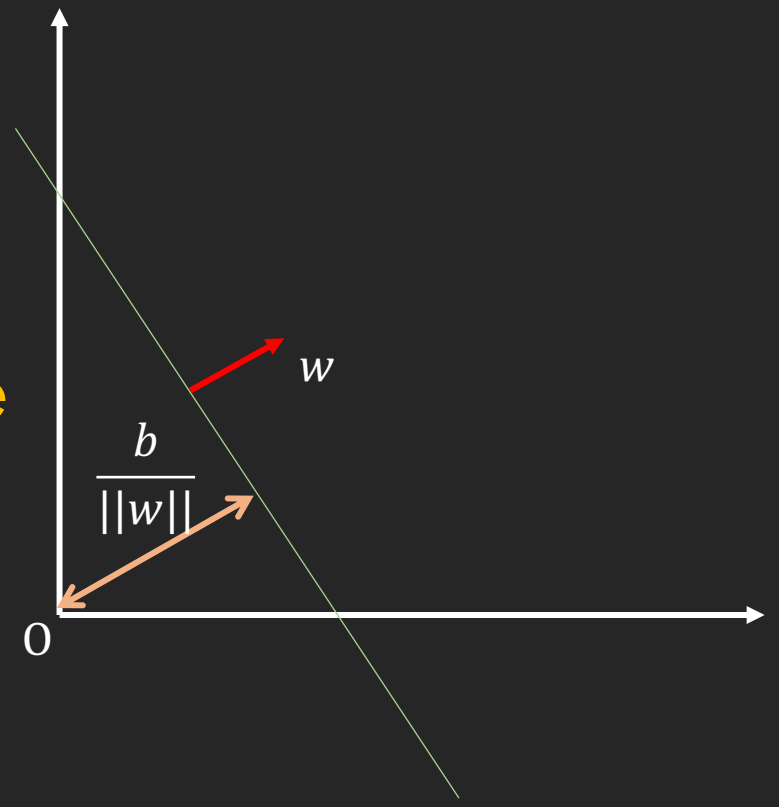
Equation of a hyperplane

- Remarks
  - The normal vector w decides the orientation of the hyperplane



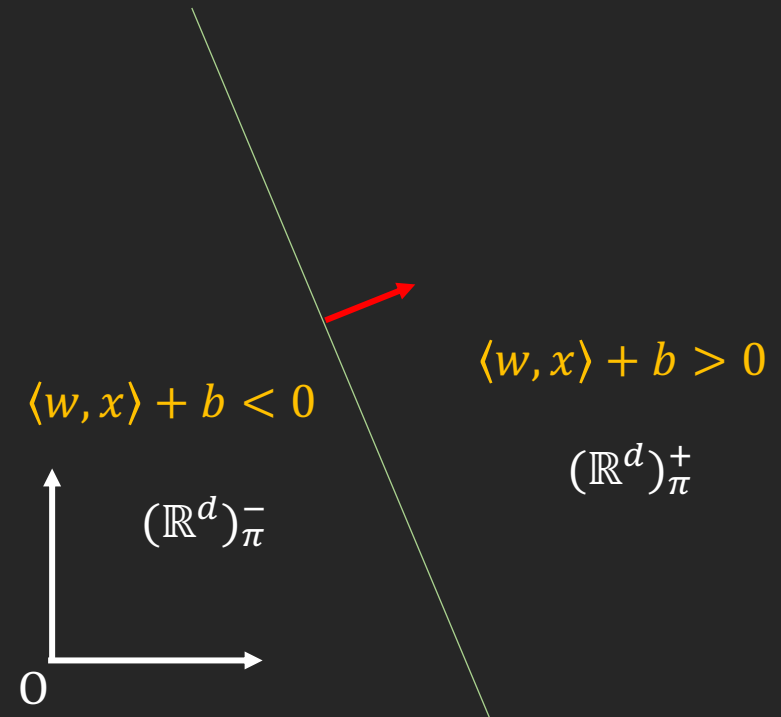
Equation of a hyperplane

- Remarks
  - The normal vector w decides the orientation of the hyperplane
  - The ratio between $||w||$ and b defines the distance between the hyperplane and the origin.



Equation of a hyperplane

- Remarks
  - The hyperplane divides the coordinate space into two parts located sidewise of the hyper-plane.
  - For any vector $x \in (\mathbb{R}^d)^+_\pi$, we have $\langle w, x\rangle + b > 0$; for any $x \in (\mathbb{R}^d)^-_\pi$, we have $\langle w, x\rangle + b < 0$.

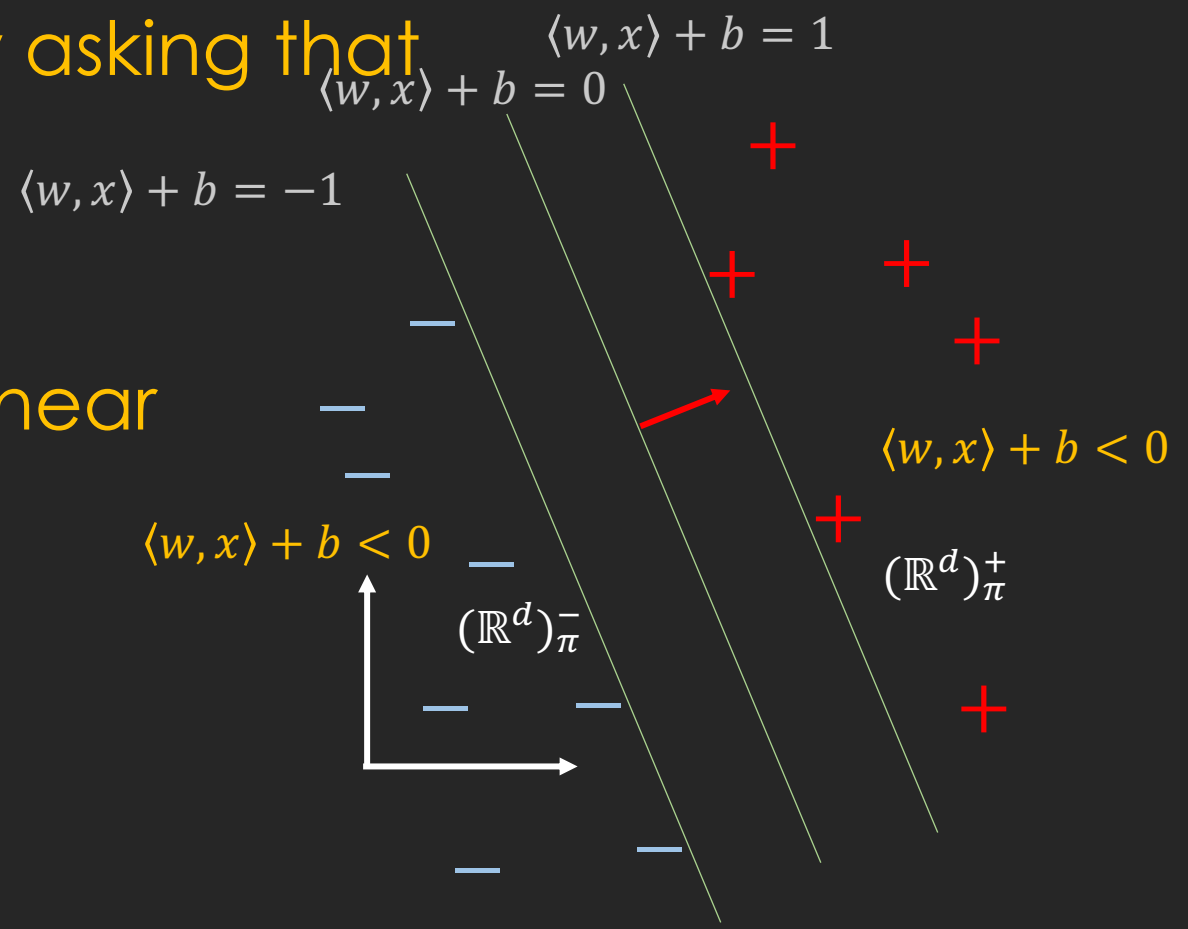


Equation of a hyperplane

- Remarks
  - For arbitrary constant $\alpha \neq 0$, parameters $\alpha w, \alpha b$ define the same hyperplane. If $\alpha < 0$ then the positive and negative half-space will swap around.
    - Are other ways of writing the same dividing line?
    - $\langle w, x \rangle + b = 0$
    - $2\langle w, x \rangle + 2b = 0$
    - $1000\langle w, x \rangle + 1000b = 0$

Equation of a hyperplane

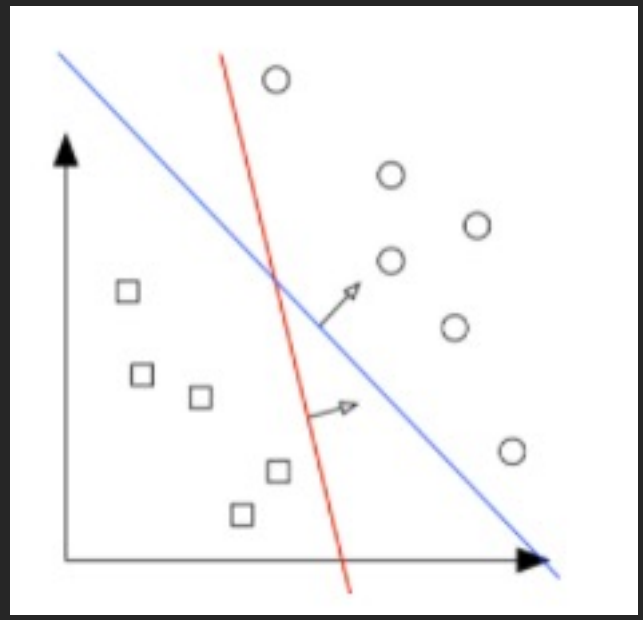
- We then can set the scale by asking that
  - For label $+1$, $\langle w, x \rangle + b \geq 1$
  - For label -1, $\langle w, x \rangle + b \leq 1$
- That is, we want to ask for a linear constraints for all data
  - $y(\langle w, x \rangle + b) \geq 1$



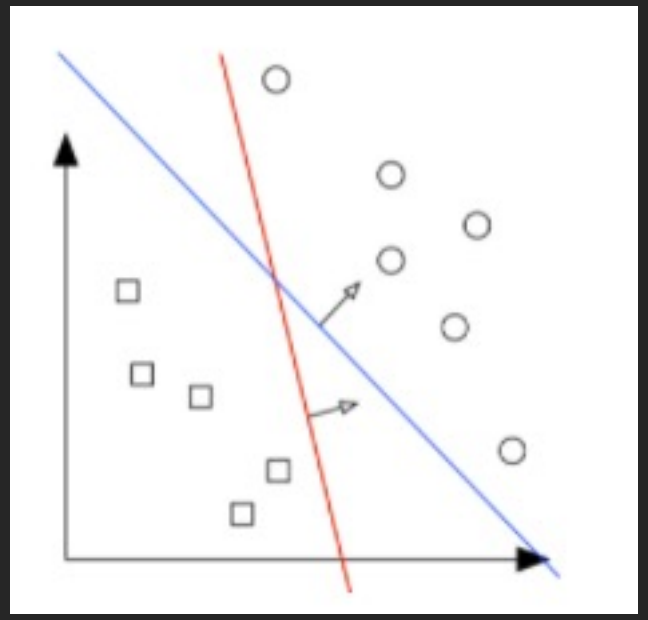
Setting

- We define a linear classifier: $h(\boldsymbol{x}) = \text{sign}(w^T \boldsymbol{x} + b)$ and we assume a binary classification setting with labels $\{+1, -1\}$.

- Typically, if a data set is linearly separable, there are infinitely many separating hyperplanes.



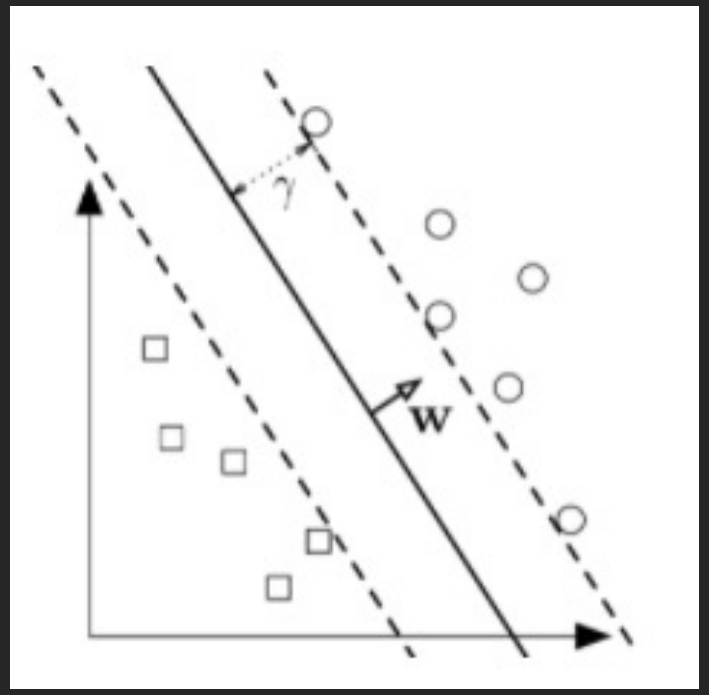
Setting



Question: What is the *best* separating hyperplane?

Two different separating hyperplanes for the same dataset.

Setting



The maximum margin hyperplane.

Answer from SVM: The one that maximizes the distance to the closest data points from both classes. We call it is the hyperplane with ***maximum margin***.

Background and Motivation

- The Support Vector Machine (SVM) is a linear classifier that can be viewed as an extension of the Perceptron developed by Rosenblatt in 1958.

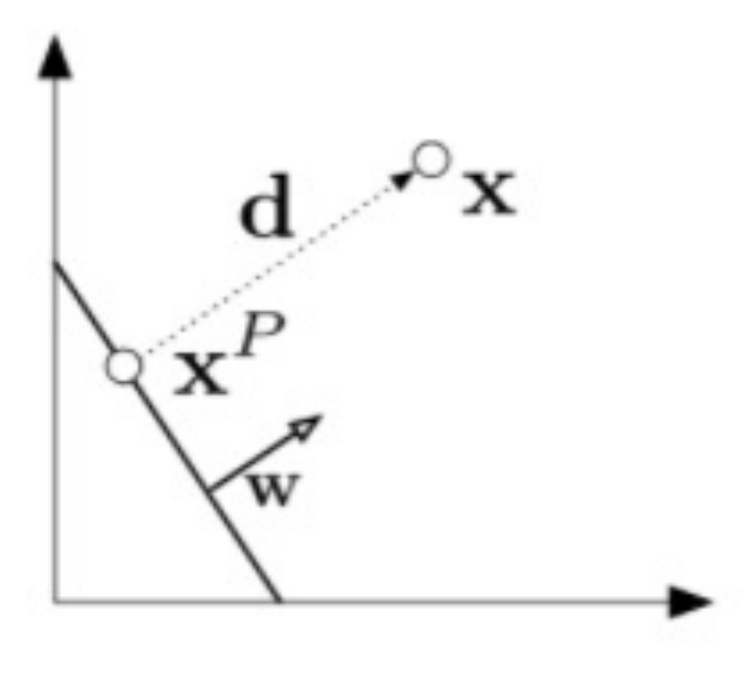
- The Perceptron guaranteed that you find ***a*** hyperplane if it exists. The SVM finds the ***maximum margin*** separating hyperplane.

Margin

- A hyperplane is defined by w, b as a set of points such that $\mathcal{H} = \{x | w^T x + b = 0\}$.
- Let the margin γ be defined as the distance from the hyperplane to the closest point across both classes.
- Q: Is the margin to class $+1$ is the same as the margin to -1?

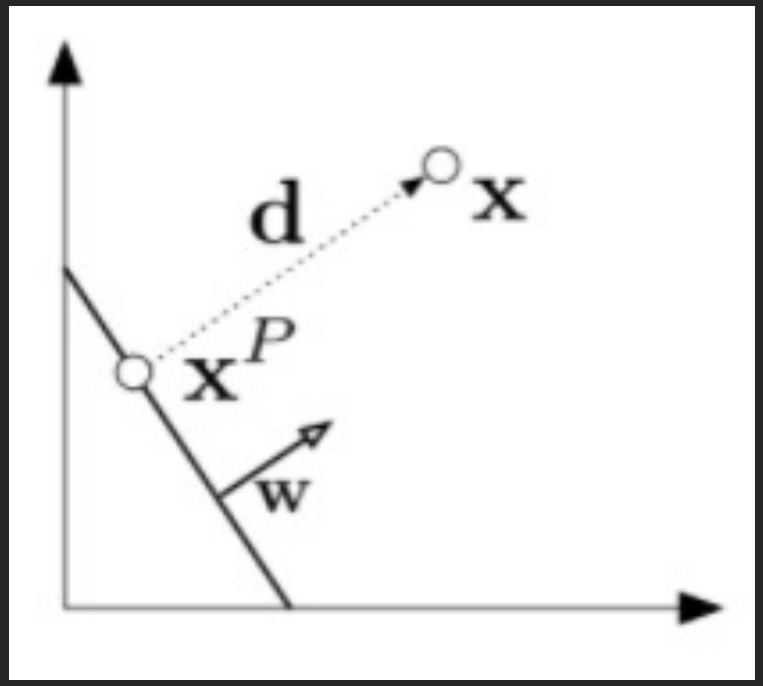
Now the question is, what is the distance of a point **x** to the hyperplane $\mathcal{H}$?

Margin



- A point $\boldsymbol{x}$
- A hyperplane $\mathcal{H}$ which is defined by $\vec{w}$.
- Project x upon the hyperplane, get x^P.
- Let d be the vector from $\mathcal{H}$ to x of minimum Length.

Distance to a hyperplane

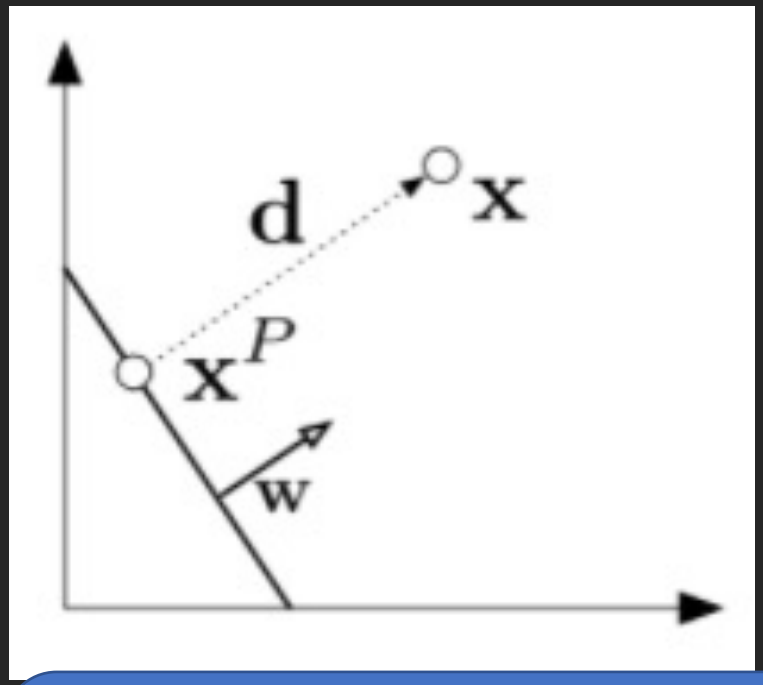


- With all the previous settings:
  - $x^P = x - d$.
  - d is parallel to w, so $d = \alpha w$ for some $\alpha \in \mathbb{R}$.
  - $x^P \in \mathcal{H}$ which implies $w^T x^P + b = 0$

$$\Longrightarrow w^T(x - d) + b = 0$$
$$\Longrightarrow w^T(x - \alpha w) + b = 0$$
$$\Longrightarrow \alpha = \frac{w^T x + b}{w^T w}$$

The length of d: $||d||_2 = \sqrt{d^T d} = \sqrt{\alpha^2 w^T w} = \frac{|w^T x + b|}{\sqrt{w^T w}} = \frac{|w^T x + b|}{||w||_2}$

Distance to a hyperplane



- Margin of $\mathcal{H}$ with respect to D:

$$\gamma(w,b) = min_{x \in D} \frac{|w^T x + b|}{||w||_2}$$

- By definition, the margin and hyperplane are scale invariant:

$$\gamma(\beta w, \beta b) = \gamma(w,b), \forall \beta \neq 0$$

If the hyperplane is such that γ is maximized, it must lie right in the middle of the two classes. In other words, γ must be the distance to the closest point within both classes. (If not, you could move the hyperplane towards data points of the class that is further away and increase γ, which contradicts that γ is maximized).

SVM: Max Margin Classifier

- Now, we can formulate our search for the maximum margin separating hyperplane as a constrained optimization problem.

- The objective is to maximize the margin under the constraints that all data points must lie on the correct side of the hyperplane:

$$\max_{w,b} \gamma(w, b)$$

$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 0$$

Is this enough?

SVM: Max Margin Classifier

- If we plug in the definition for margin γ (the computation for d), we will get:

$$\max_{w,b} \min_{x_i \in D} \frac{1}{||w||_2} |w^T x_i + b|, \ s.t. \forall i \ \ y_i(w^T x_i + b) \geq 0$$

Definition for margin

$$\max_{w,b} \frac{1}{||w||_2} \min_{x_i \in D} |w^T x_i + b|, \ s.t. \forall i \ \ y_i(w^T x_i + b) \geq 0$$

Maximize margin

SVM: Max Margin Classifier

- Remember that the hyperplane is scale invariant, we can fix the scale of w, b anyway we want.
- We just choose w, b to be

$$\min_{x_i \in D} |w^T x_i + b| = 1$$

- We can add this re-scaling as an equality constraint. Then

$$\max_{w,b} \frac{1}{||w||_2} \min_{x_i \in D} |w^T x_i + b| \Rightarrow \max_{w,b} \frac{1}{||w||_2} \cdot 1 \Rightarrow \min_{w,b} ||w||_2 = \min_{w,b} w^T w$$

- Now the new optimization problem becomes

$$\min_{w,b} w^T w$$

$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 0 \;\; , \min_i |w^T x_i + b| = 1$$

SVM: Max Margin Classifier

- The new optimization problem

$$\min_{w,b} w^T w$$

$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 0 \;, \min_i |w^T x_i + b| = 1$$

is equivalent to

$$\min_{w,b} w^T w$$

$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 1$$

Find the simplest hyperplane (where simpler means smaller $w^T w$ such that all inputs lie at least 1 unit away from the hyperplane on the correct side.

- This formulation is a quadratic optimization problem. The objective is *quadratic,* and the constraints are all *linear,* which can be solved by QCQP (quadratically constrained quadratic program). It has a unique solution whenever a separating hyper plane exists.

Support Vectors

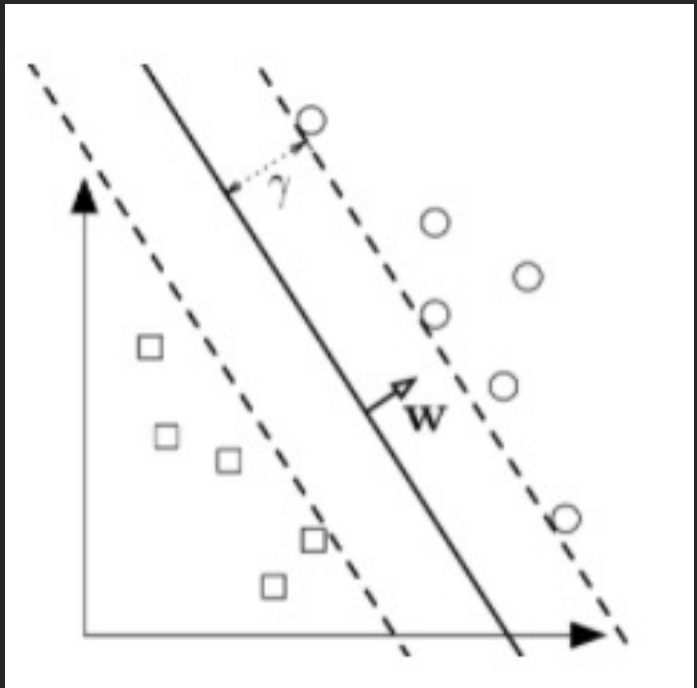
- For the optimal w, b pair, some training points will have tight constraints, i.e.

$$y_i(w^T x_i + b) = 1$$

- This must be the case, because if for all training points we had a strict > inequality, it would be possible to scale down both parameters w, b until the constraints are tight and obtained an even lower objective value.

- We refer to these training points as ***support vectors***.

Support Vectors



- Support vectors are special because they are the training points that define the maximum margin of the hyperplane to the data set and they therefore determine the shape of the hyperplane.
- If you change the support vectors, the resulting hyperplane would change.
- The opposite is the case for non-support vectors (provided you don't move them too much, or they would turn into support vectors themselves.)

SVM with soft constraints

- If the data is low dimensional, it is often the case that there is no separating hyperplane between the two classes.
- In this case, there is no solution to the optimization problems stated above.
- The solution to this case is to allow *the constraints to be violated* ever so slight with the introduction of **slack variables**.

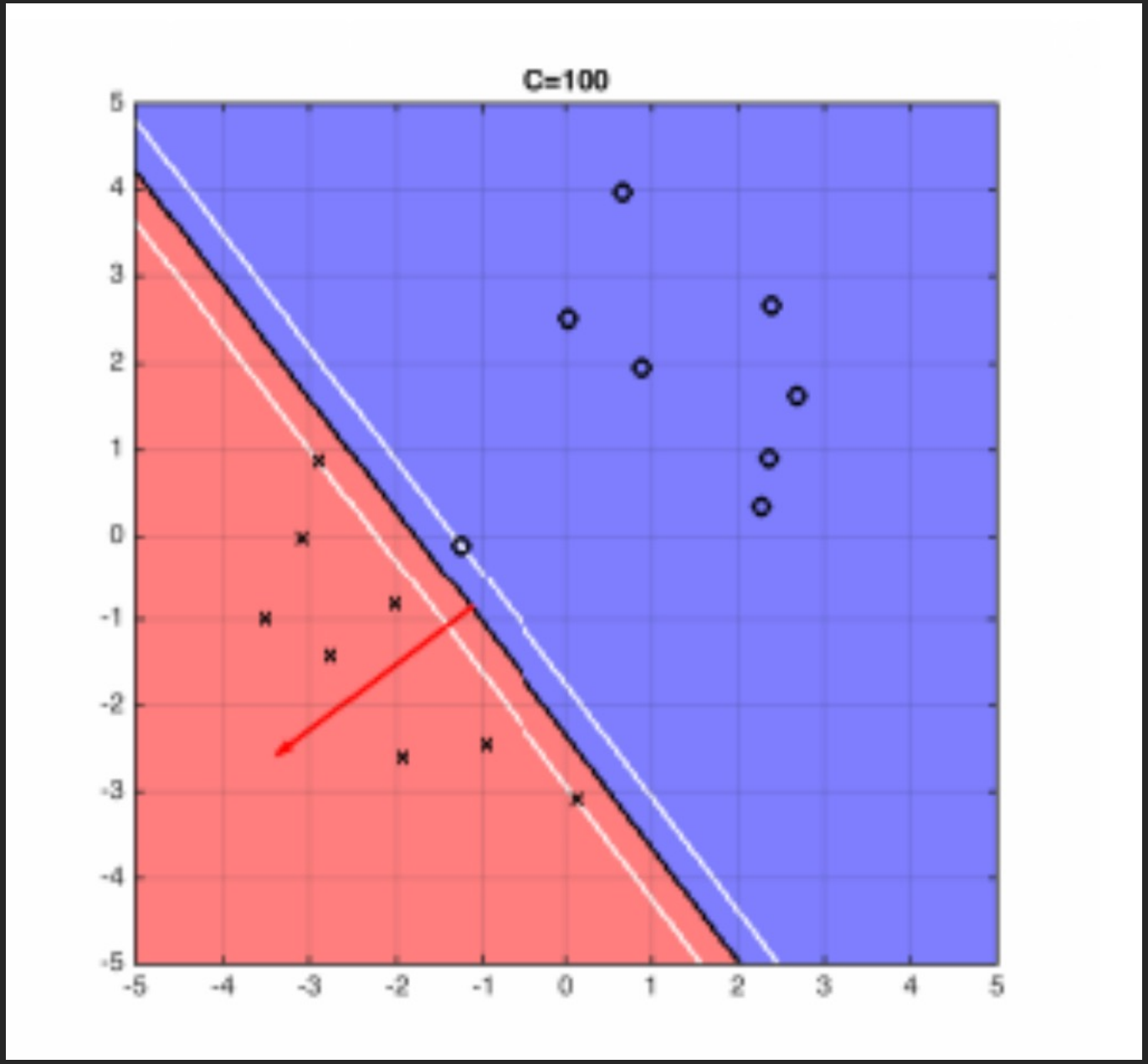
$$\min_{w,b} w^T w + \mathrm{C} \sum_{i=1}^{n} \xi_i$$

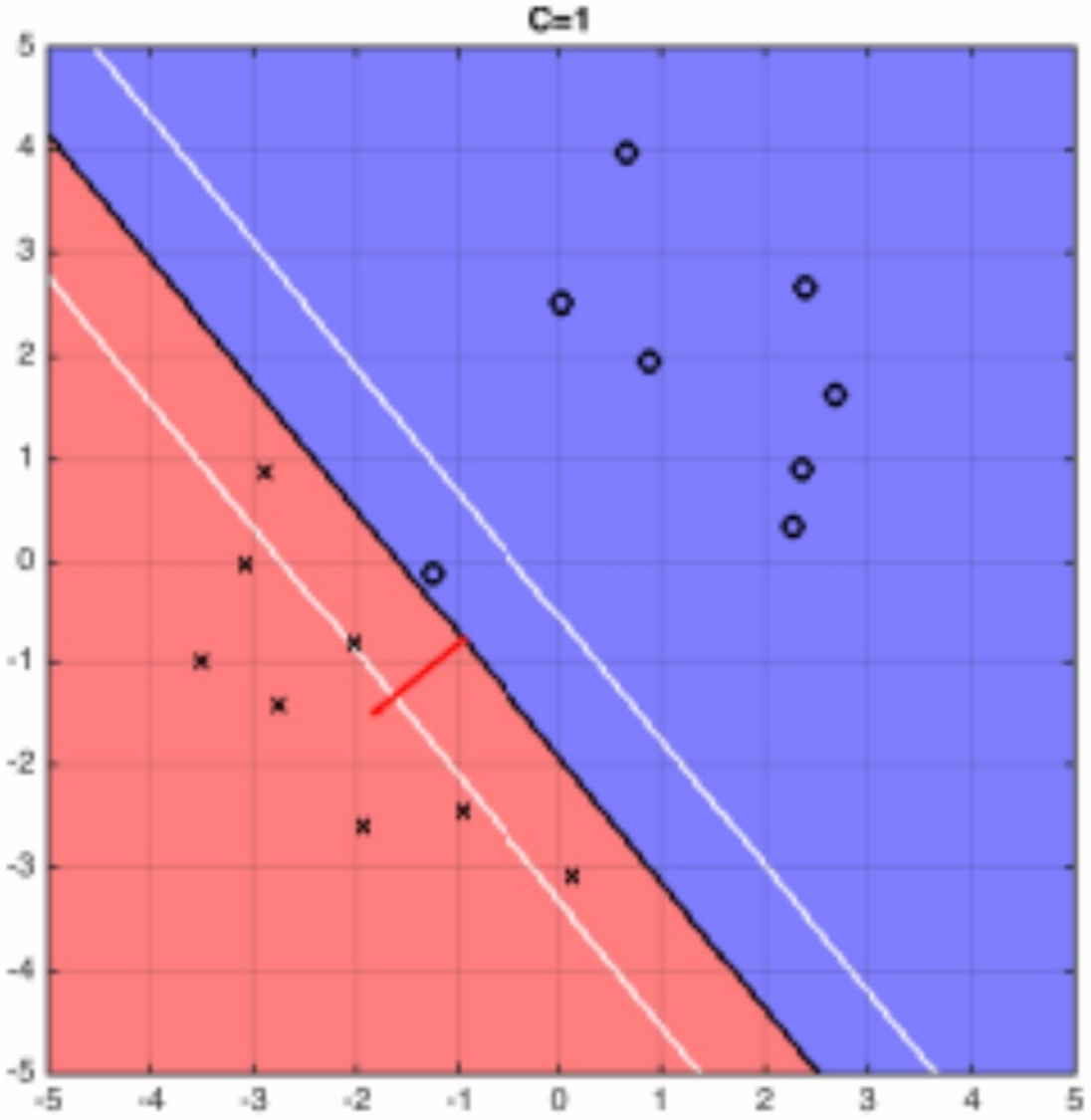
$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 1 - \xi_i \;\; , \quad \forall i \; \xi_i \geq 0$$

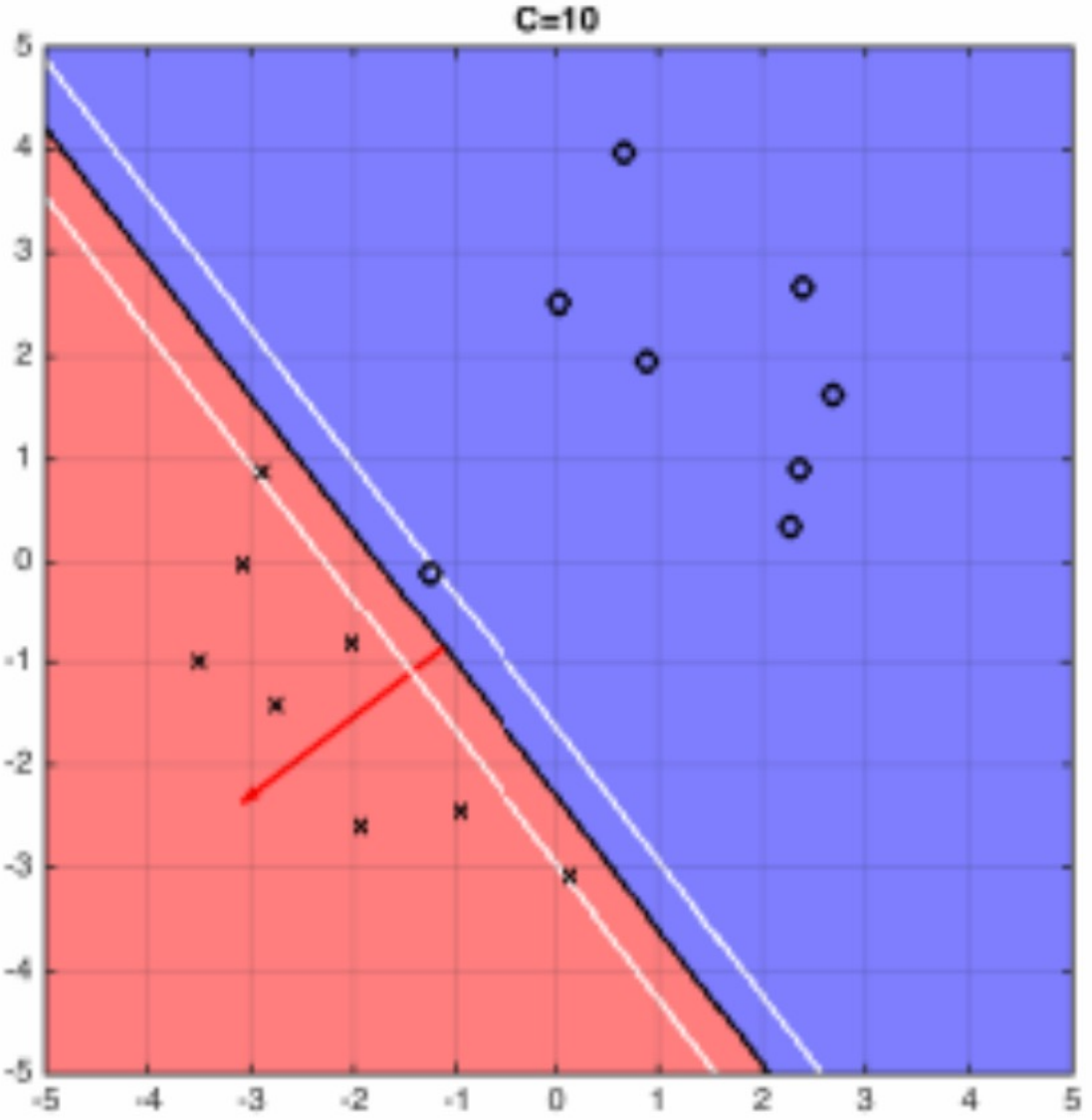
SVM with soft constraints

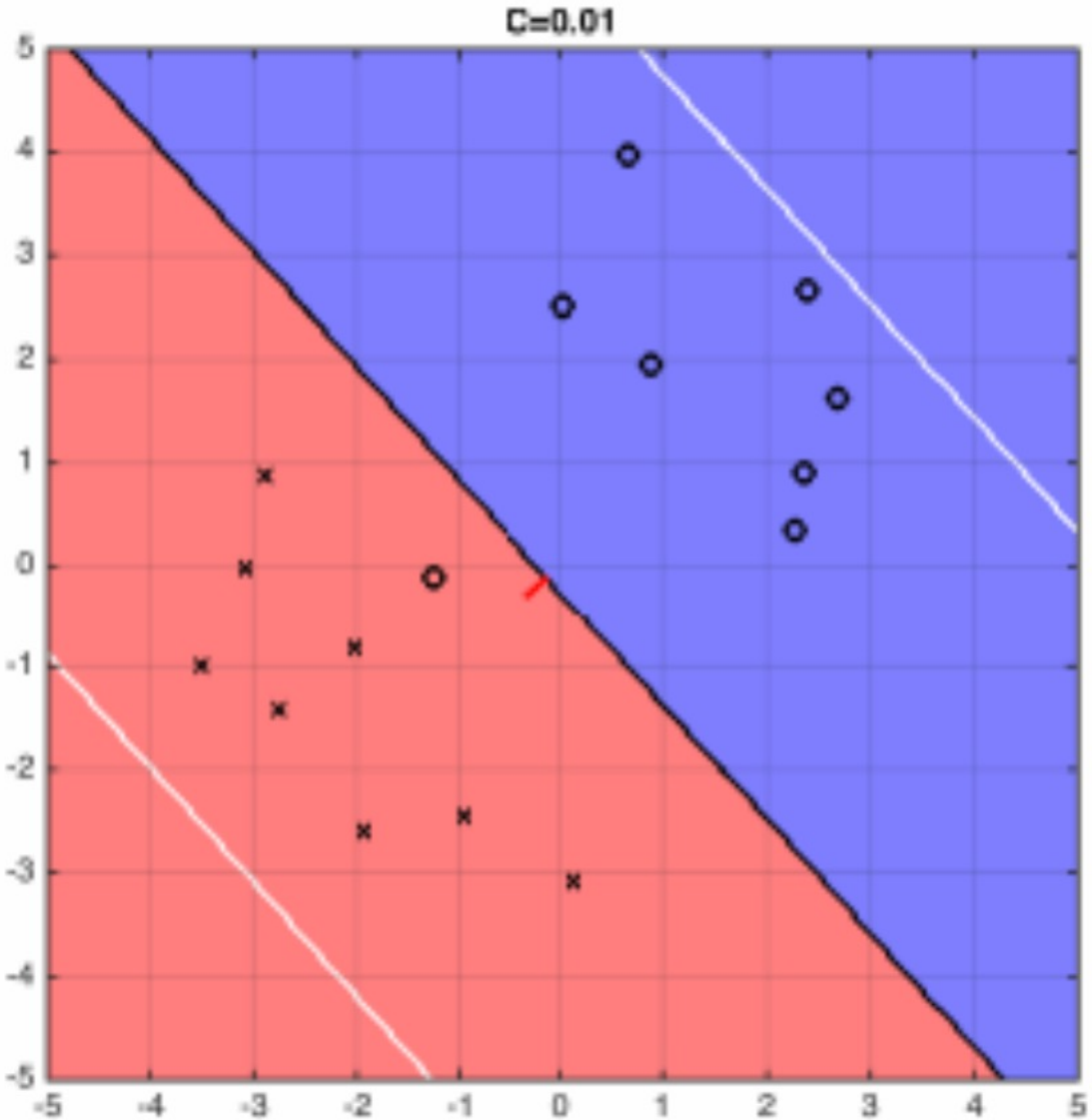
$$\min_{w,b} w^T w + \mathrm{C} \sum_{i=1}^{n} \xi_i$$
$$s.t. \forall i \;\; y_i(w^T x_i + b) \geq 1 - \xi_i \;\; ,$$
$$\forall i \; \xi_i \geq 0$$

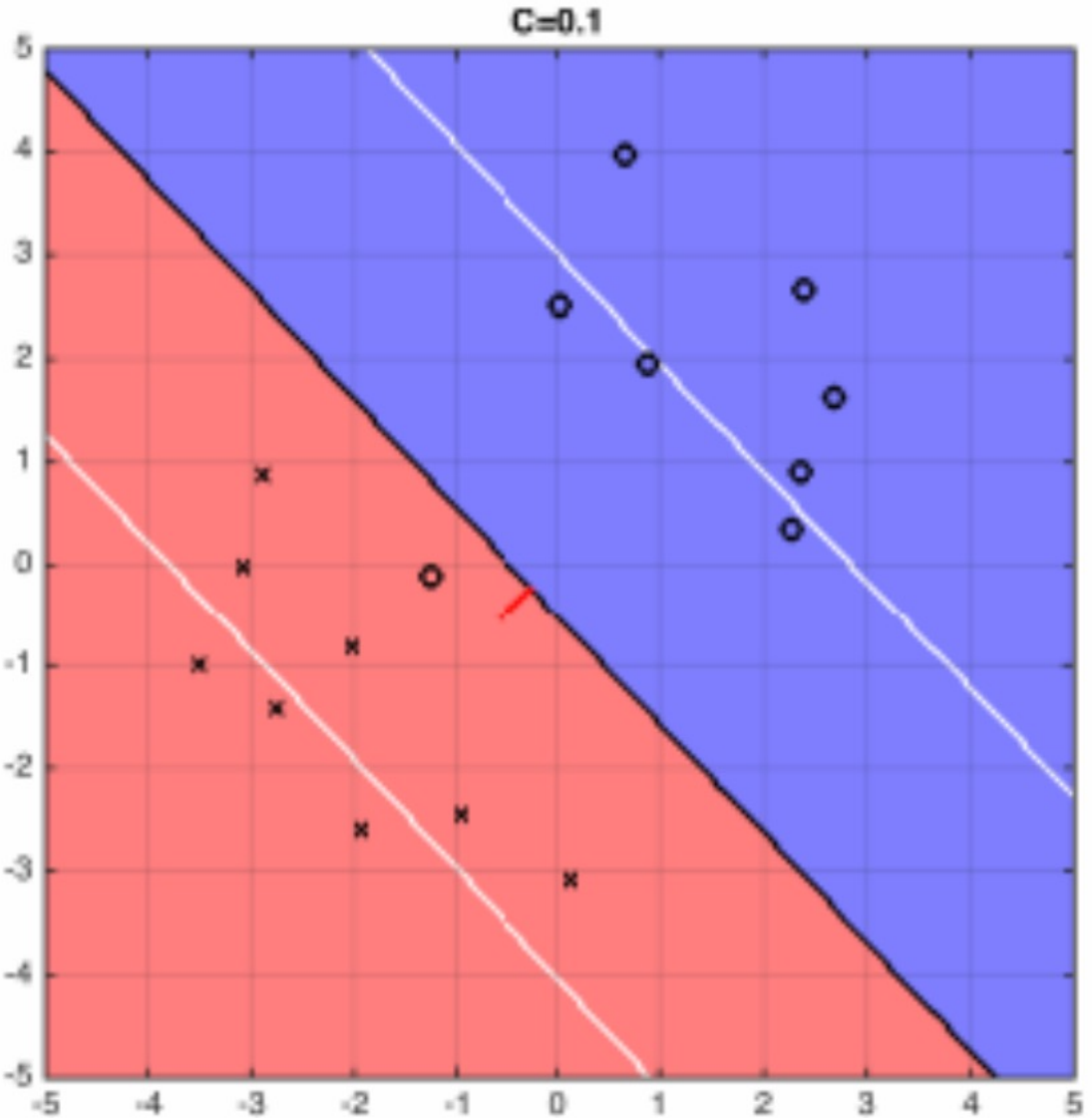
- This slack variable ξ_i allows the input x_i to be closer to the hyperplane (*or even be on the wrong side*), but there is a penalty in the objective function for such "slackness".
  - If C is very large, the SVM becomes very strict and tries to get all points to be on the right side of the hyperplane.
  - If C is very small, the SVM becomes very loose and may "sacrifice" some points to obtain a simpler solution (i.e. lower$||w||_2^2$) solution.


C=100
5
4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3
4
5


C=1


C=10


C=0.01


C=0.1

Unconstrained Formulation

- Consider the value of ξ_i for the case of $C \neq 0$. Because the objective will always try to minimize ξ_i as much as possible, the equation must hold as an equality and we have:

$$\xi_i = \begin{cases} 1 - y_i(w^T x_i + b). & \text{if } y_i(w^T x_i + b) < 1 \\ 0. & \text{if } y_i(w^T x_i + b) \geq 1 \end{cases}$$

This is equivalent to the following closed form:

$$\xi_i = \max(1 - y_i(w^T x_i + b), 0)$$

Unconstrained Formulation

- If we plug this closed form

$$\xi_i = \max(1 - y_i(w^T x_i + b), 0)$$

into the objective of the SVM optimization problem, we obtain the following *unconstrained* version as loss function and regularizer:

$$\min_{w,b} \underbrace{w^T w}_{\text{L2 regularizer}} + C \sum_{i=1}^{n} \underbrace{\max[1 - y_i(w^T x + b), 0]}_{\text{Hinge-loss}}$$

Unconstrained Formulation

- This formulation allows us to optimize the SVM parameters (w, b) just like logistic regression. The only difference is that we have the hinge-loss instead of logistic loss.