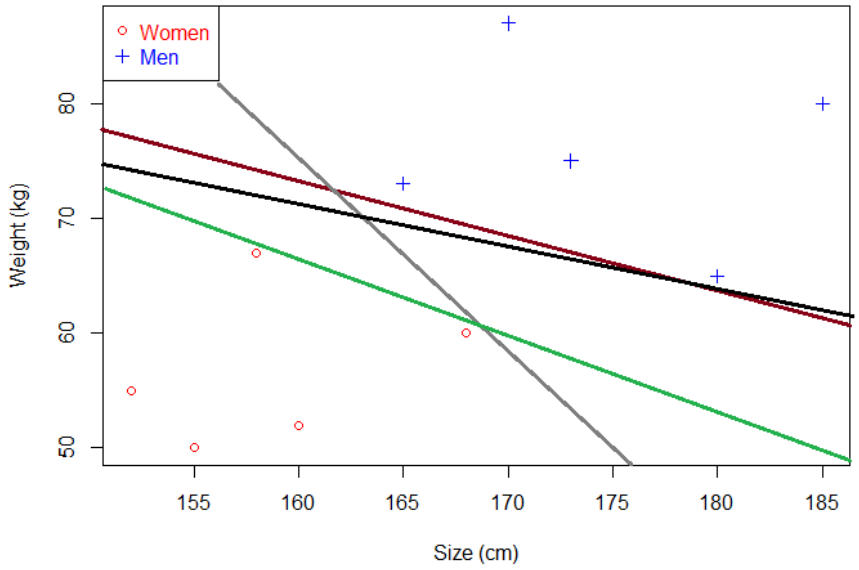
# Support Vector Machine

Department of Computer Languages and Systems

# Introduction

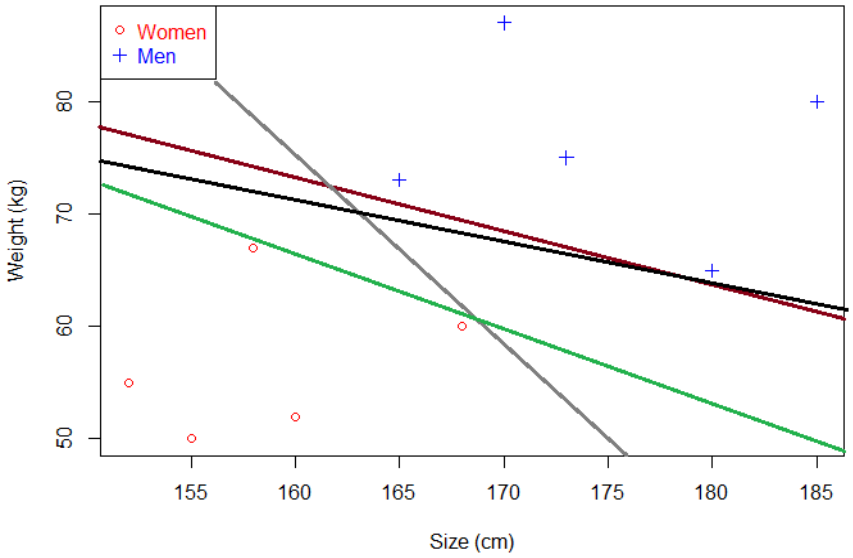
A support vector machine (SVM) tries to find a hyperplane that best separates the two classes …

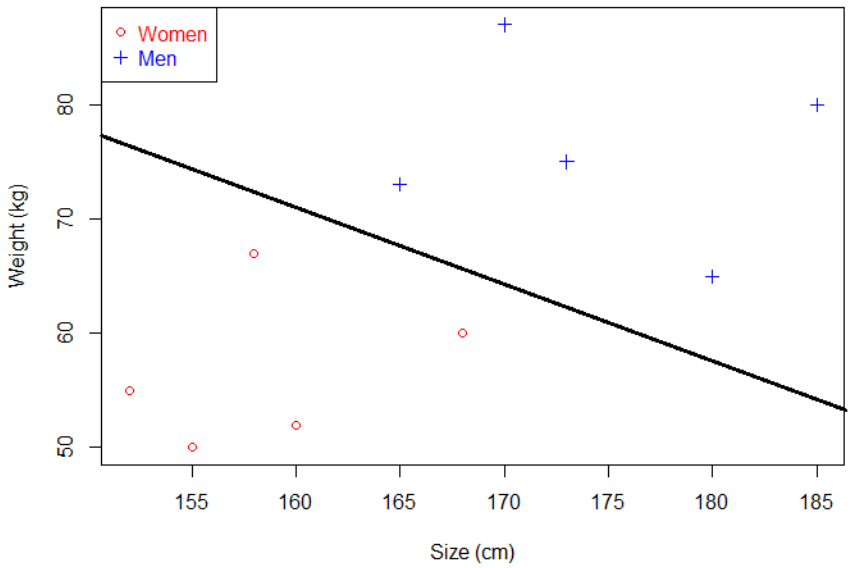
but there are many (infinite) separating hyperplanes, which one does it select?



# Introduction

We will try to select the hyperplane that maximizes the distance (margin) to the closest samples of each class



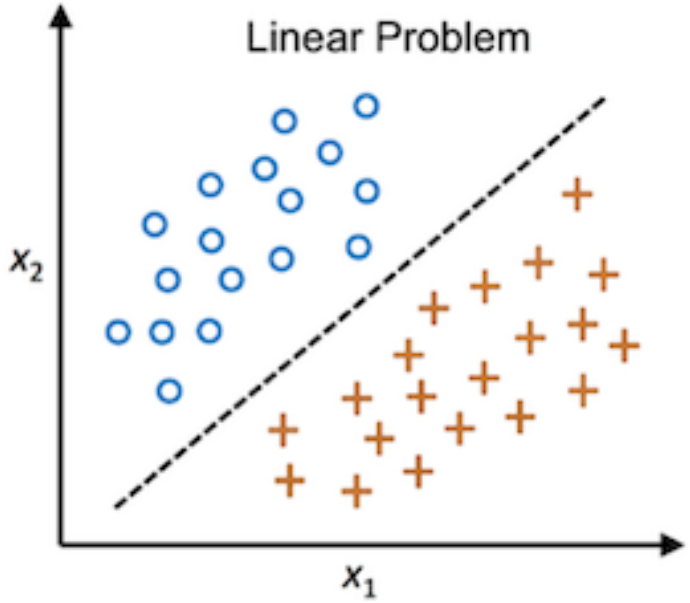




The optimal separation hyperplane is equidistant from the closest example of each class

# Types of SVM algorithms

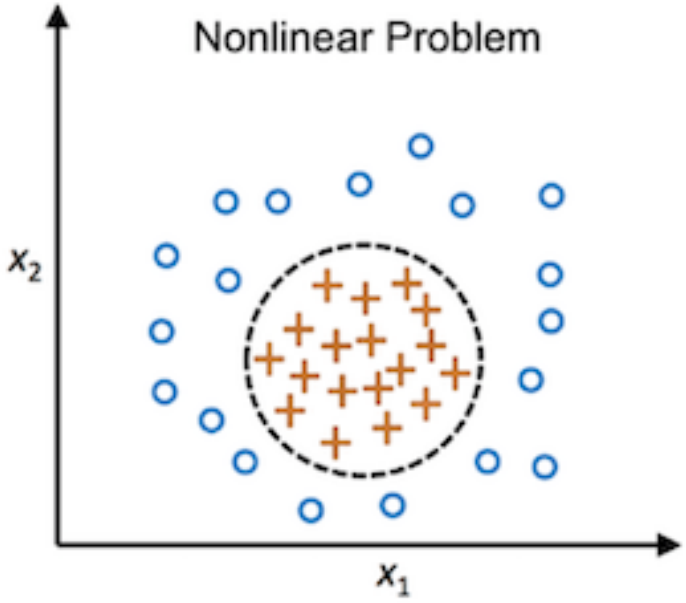
## Linear SVM

When the data is perfectly linearly separable → **hard margin**



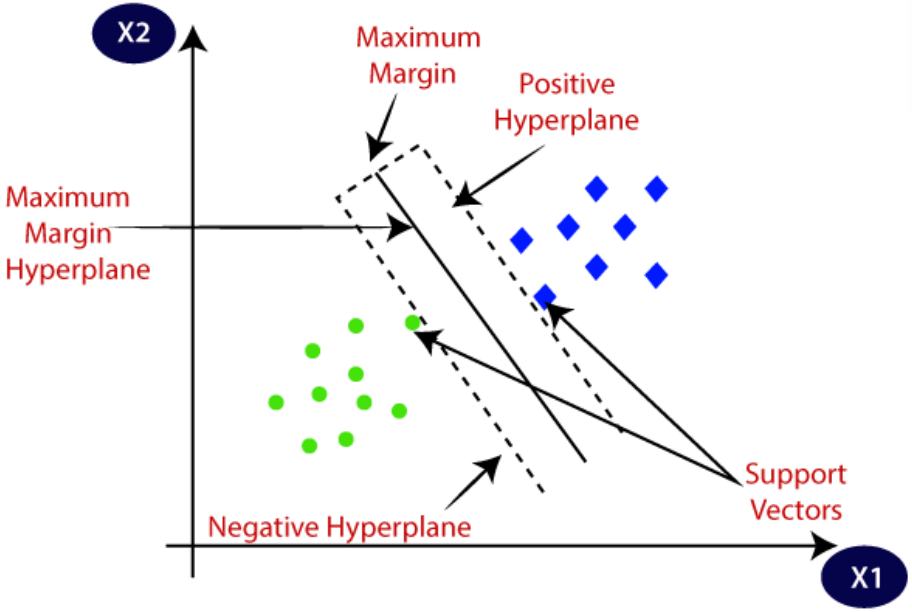

## Non-linear SVM

When the data is not linearly separable, we have to use some kernel → **soft margin**



# What is a margin?

it is the distance between the hyperplane and the samples closest to the hyperplane (support vectors).




the optimal hyperplane will be the one with the largest margin!

maximum margin
=
minimum loss

# How to find the biggest margin?

Given a data set $\mathcal{D}$ with $n$ $d$-dimensional vectors $\mathbf{x}_i$linearly separable, each associated with a value $y_i$ to indicate if $\mathbf{x}_i$ belongs to class -1 or to class +1

$$\mathcal{D} = \left\{(\mathbf{x}_i, y_i) \,\middle|\, \mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1, +1\}\right\}_{i=1}^{n}$$

Steps:

1. Select two hyperplanes that separate the data with no points between them
2. Maximize their distance (the margin)

The region bounded by the two hyperplanes will be the largest possible margin!

# Step 1: Select two hyperplanes …

Given a hyperplane $H_0$ separating the data set and satisfying:

$$H_0: \mathbf{w}^T\mathbf{x} + b = 0$$

– we can select two others hyperplanes $H_1$ and $H_2$ that also separate the data and have the following equations:

$$H_1: \mathbf{w}^T\mathbf{x} + b = \delta \quad \text{and} \quad H_2: \mathbf{w}^T\mathbf{x} + b = -\delta$$

– to simplify the problem, we can set $\delta = 1$:

$$H_1: \mathbf{w}^T\mathbf{x} + b = 1 \quad \text{and} \quad H_2: \mathbf{w}^T\mathbf{x} + b = -1$$

where $\mathbf{w}$ is the weight vector (to define the orientation of the hyperplane) and $b$ is the bias

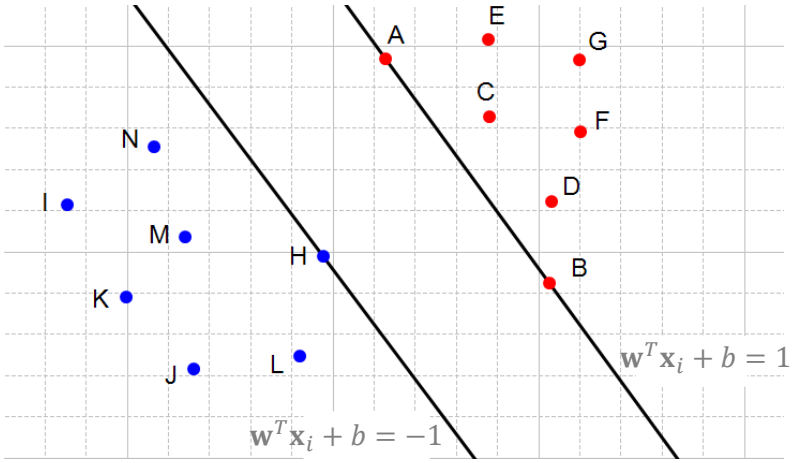
# Step 1 (cont.)

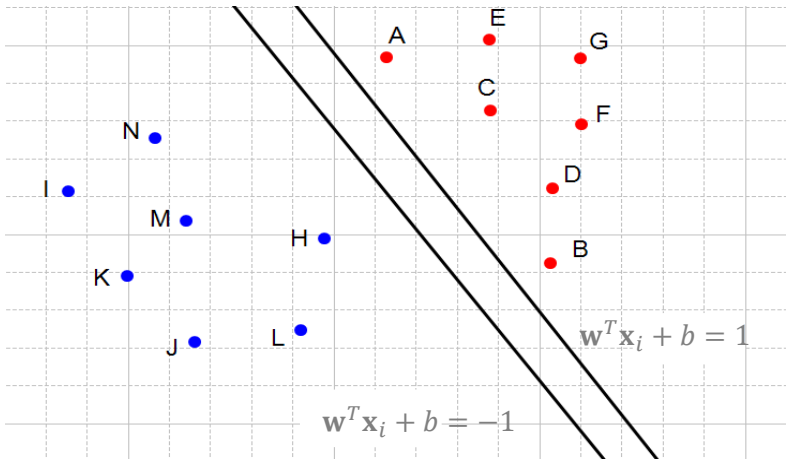
… but not any hyperplane is valid, we will only select those that meet the following two constraints for all $\mathbf{x}_i \in \mathcal{D}$:

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \text{ for } y_i = 1$$

or

$$\mathbf{w}^T\mathbf{x}_i + b \leq 1 \text{ for } y_i = -1$$

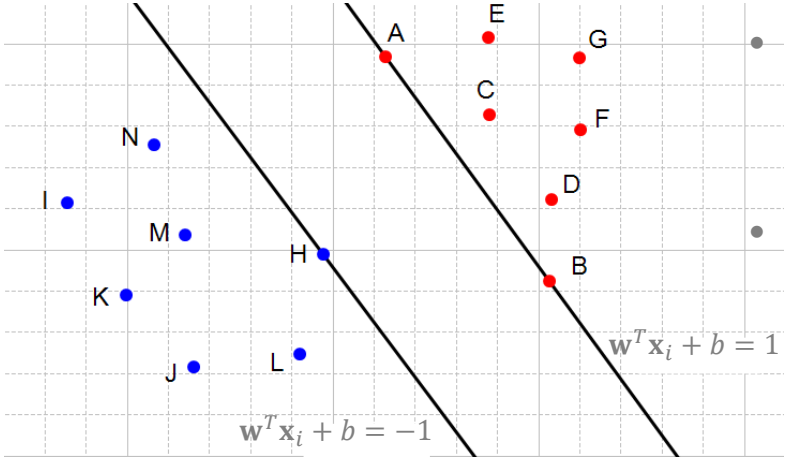


# Step 1 (cont.)

Understanding the constraints: we need to verify the points do not violate the constraints

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \text{ for } y_i = 1 \qquad \mathbf{w}^T\mathbf{x}_i + b \leq 1 \text{ for } y_i = -1$$

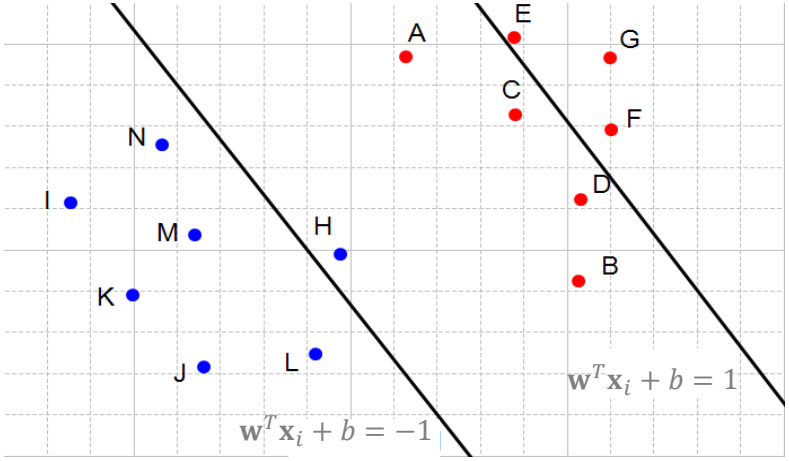



- when $\mathbf{x}_i = A$, we see that the point is on the hyperplane, so $\mathbf{w}^T\mathbf{x}_i + b = 1$ and the constraint is respected
- when $\mathbf{x}_i = C$, we see that the point is above the hyperplane, so $\mathbf{w}^T\mathbf{x}_i + b > 1$ and the constraint is respected

# Step 1 (cont.)

Understanding the constraints: we need to verify the points do not violate the constraints

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \text{ for } y_i = 1 \qquad \mathbf{w}^T\mathbf{x}_i + b \leq 1 \text{ for } y_i = -1$$




when a constraint is not satisfied (i.e. there are points between the two hyperplanes) means that we cannot select these two hyperplanes

# Step 1 (cont.)

Previous equations can be combined into a single constraint:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \quad \forall i = 1, \ldots, n$$

# Step 2: Maximize the margin

Recall that the shortest distance between a point $\mathbf{x}_i$ and a hyperplane $\mathbf{w}^T\mathbf{x}_i + b = 0$ can be calculated as:

$$d_i = \frac{|\mathbf{w}^T\mathbf{x}_i + b|}{\|\mathbf{w}\|}$$

where $\|\mathbf{w}\|$ represents the Euclidean norm of the vector $\mathbf{w}$, which also has the property of being perpendicular to the hyperplane.

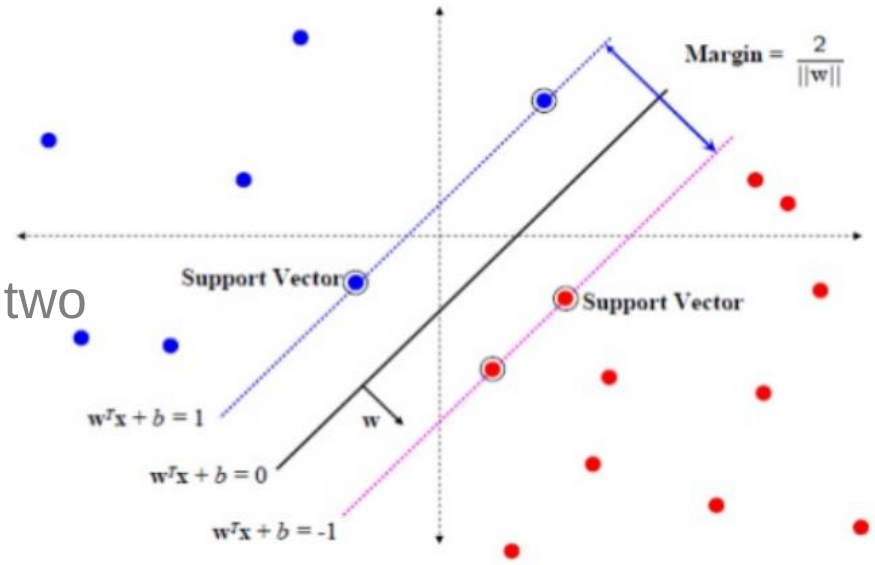
# Step 2 (cont.)

The distance of the closest points (i.e., support vectors) to the optimal hyperplane will be

$$\frac{|\pm 1|}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|}$$

and therefore the margin (the closest distance between the two classes) will be:

$$\rho = \frac{2}{\|\mathbf{w}\|}$$



# Step 2 (cont.)

We want to maximize the margin $\rho$:

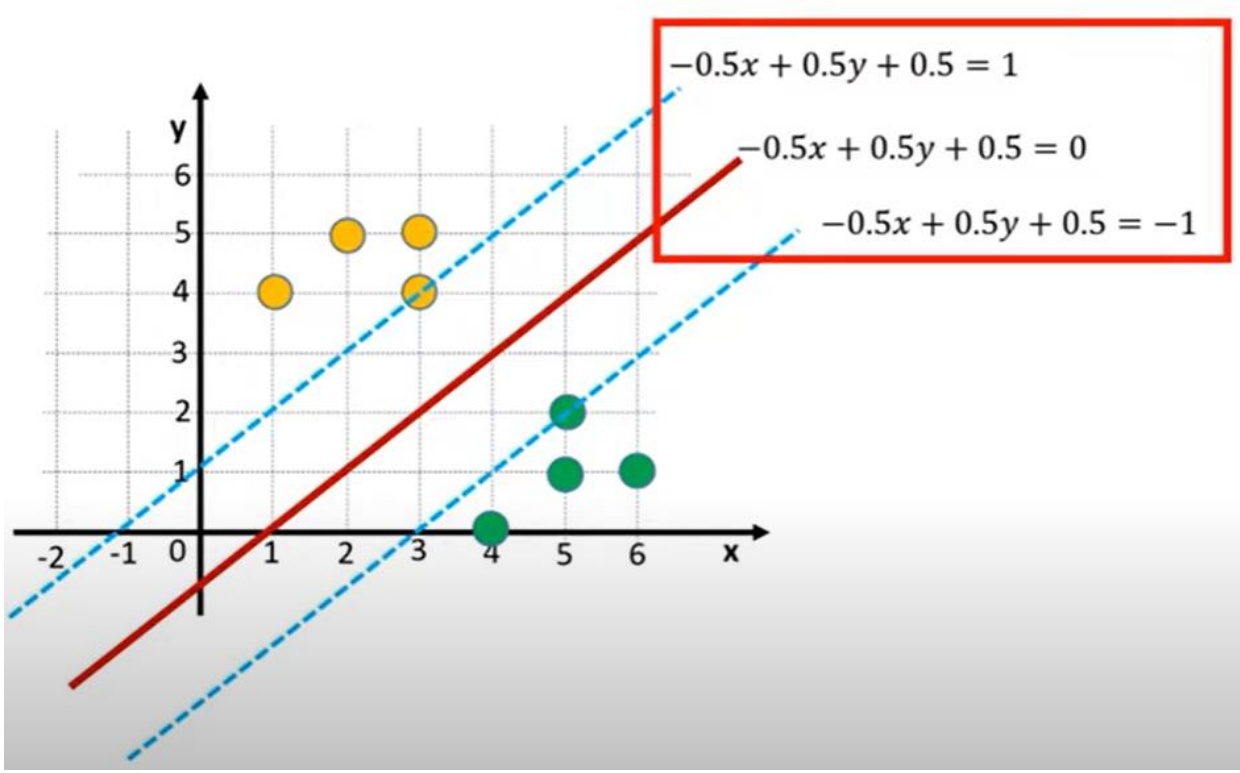
$$\max_{\mathbf{w},b} \frac{\mathbf{2}}{\|\mathbf{w}\|} \equiv \min_{\mathbf{w},b} \|\mathbf{w}\|$$

For computational reasons, this term is usually expressed as

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2$$

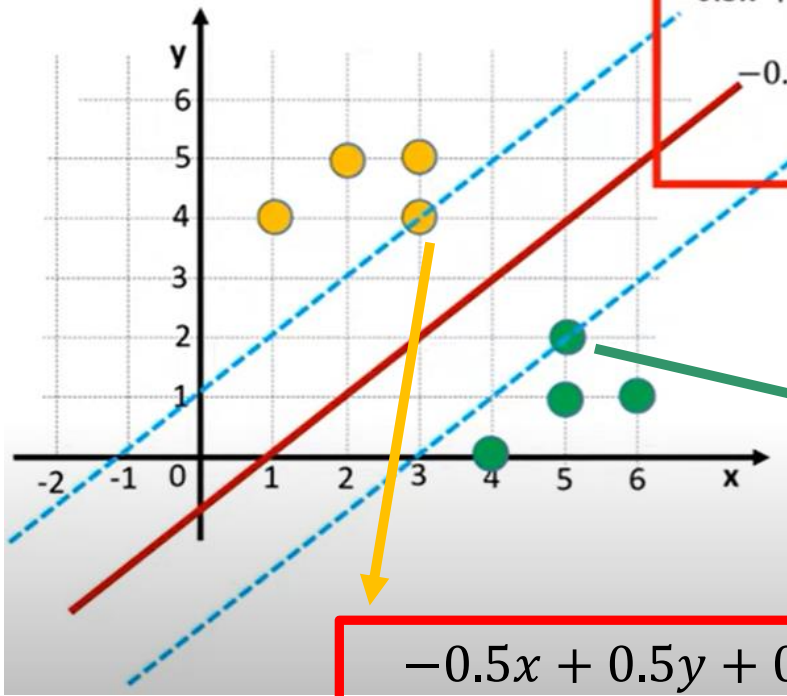
subject to $\quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \quad \forall i = 1, \ldots, n$

# An example (i)




| Group | x | y |
|---|---|---|
| A | 1 | 4 |
| A | 2 | 5 |
| A | 3 | 5 |
| A | 3 | 4 |
| B | 6 | 1 |
| B | 4 | 0 |
| B | 5 | 2 |
| B | 5 | 1 |

# An example (ii)




$$-0.5x + 0.5y + 0.5 = 1$$

$$-0.5x + 0.5y + 0.5 = 0$$

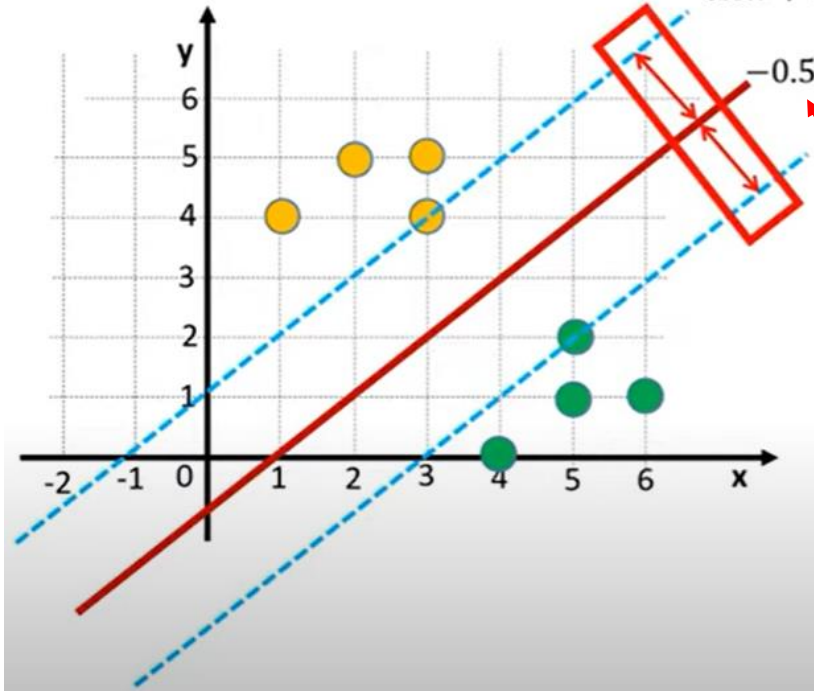
$$-0.5x + 0.5y + 0.5 = -1$$

| Group | x | y |
|---|---|---|
| A | 1 | 4 |
| A | 2 | 5 |
| A | 3 | 5 |
| A | 3 | 4 |
| B | 6 | 1 |
| B | 4 | 0 |
| B | 5 | 2 |
| B | 5 | 1 |

$$-0.5x + 0.5y + 0.5 = -1$$
$$-0.5 \cdot 6 + 0.5 \cdot 3 + 0.5 = -1$$

$$-0.5x + 0.5y + 0.5 = 1$$
$$-0.5 \cdot 3 + 0.5 \cdot 4 + 0.5 = 1$$

# An example (iii)




$-0.5x + 0.5y + b = 1$

$b = 0.5$

$-0.5x + 0.5y + b = 0$

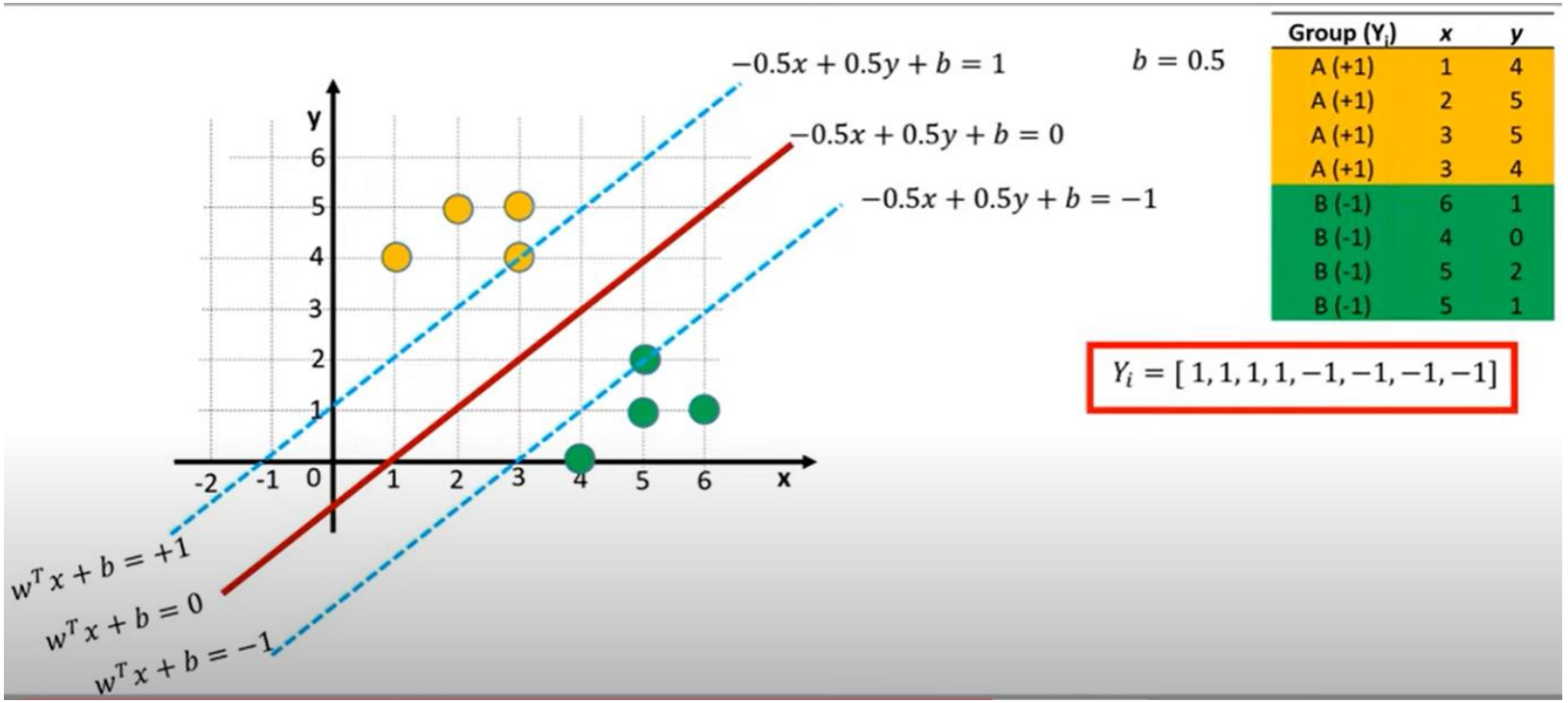
$-0.5x + 0.5y + b = -1$

| Group | x | y |
|---|---|---|
| A | 1 | 4 |
| A | 2 | 5 |
| A | 3 | 5 |
| A | 3 | 4 |
| B | 6 | 1 |
| B | 4 | 0 |
| B | 5 | 2 |
| B | 5 | 1 |

$$\mathbf{w}^T\mathbf{x} + b = 0$$

$$[-0.5 \;\; 0.5] \cdot \begin{bmatrix} x \\ y \end{bmatrix} + b = 0$$

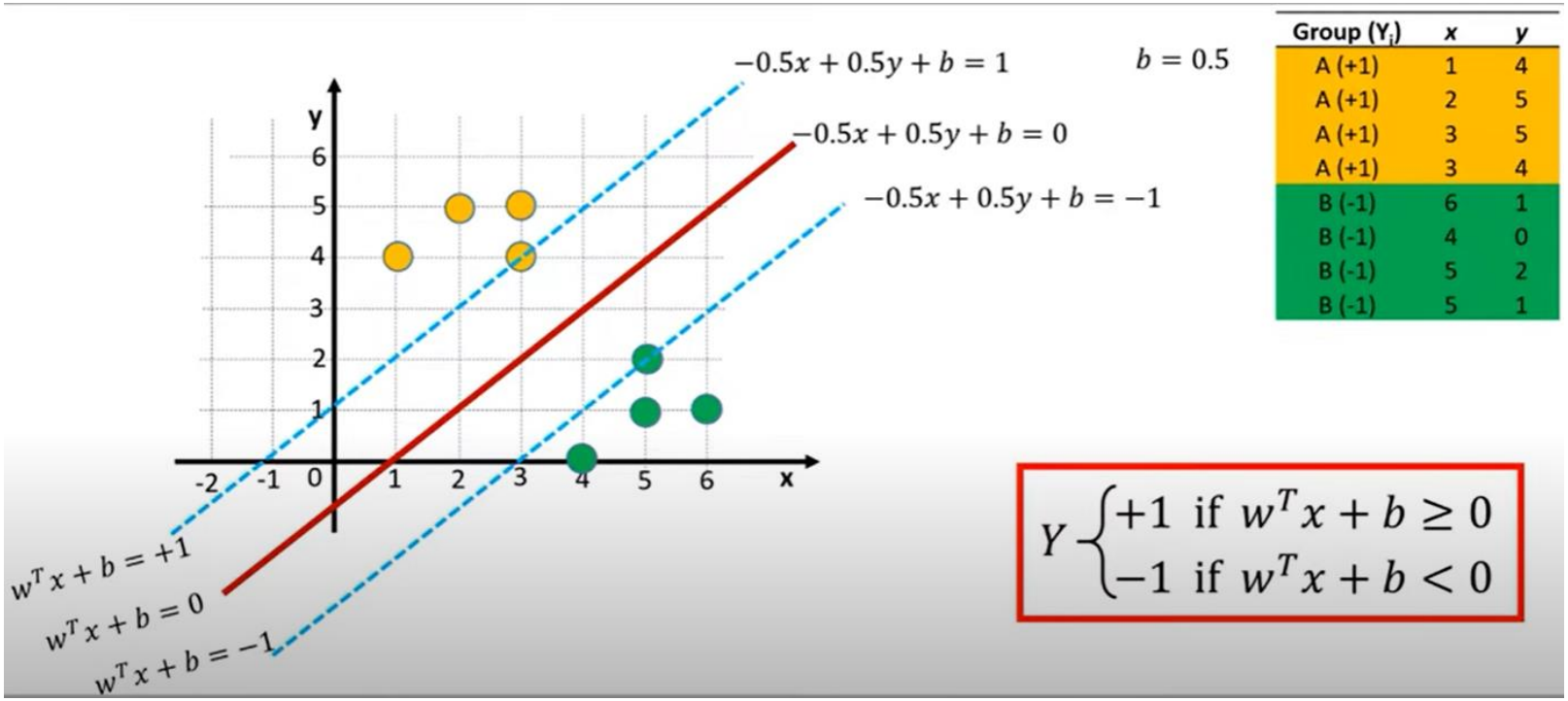
# An example (iv)




| Group ($Y_i$) | $x$ | $y$ |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

$$Y_i = [1, 1, 1, 1, -1, -1, -1, -1]$$

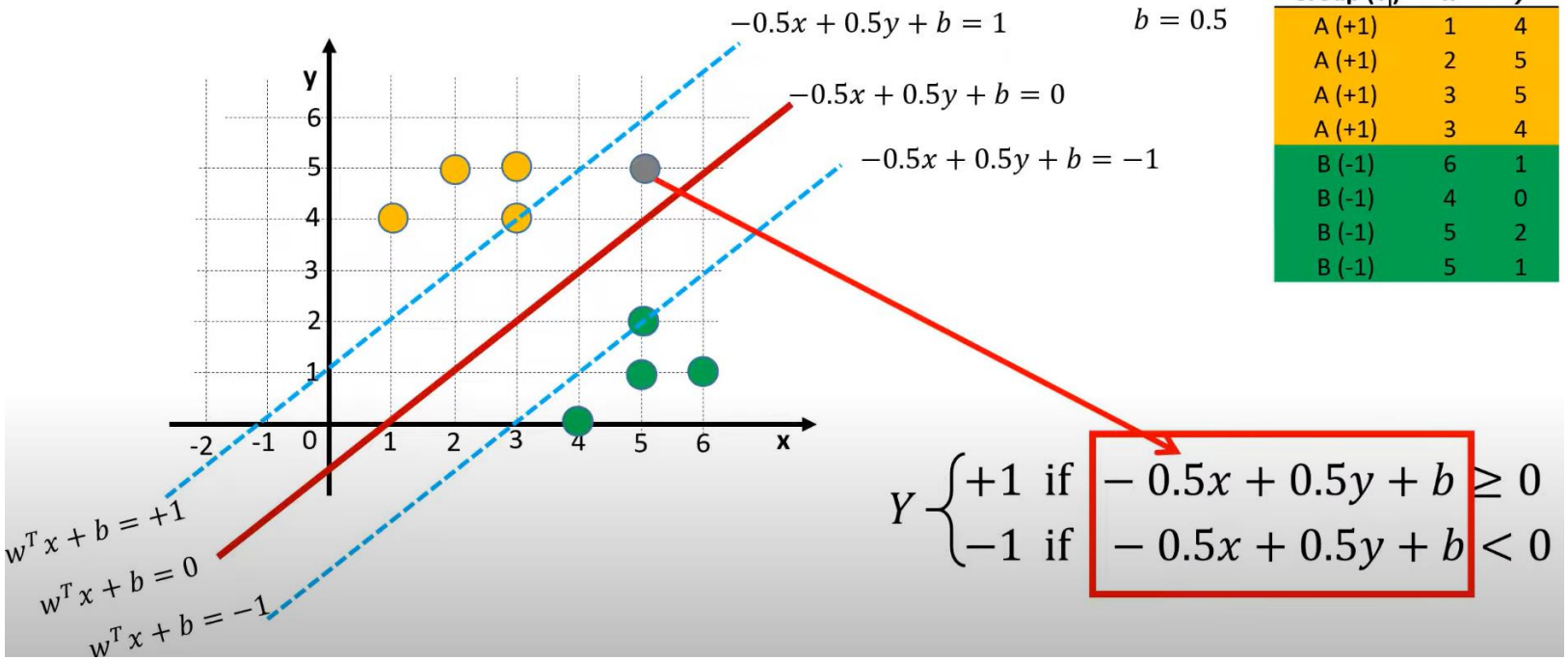
# An example (v)




| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

Prediction for a new sample

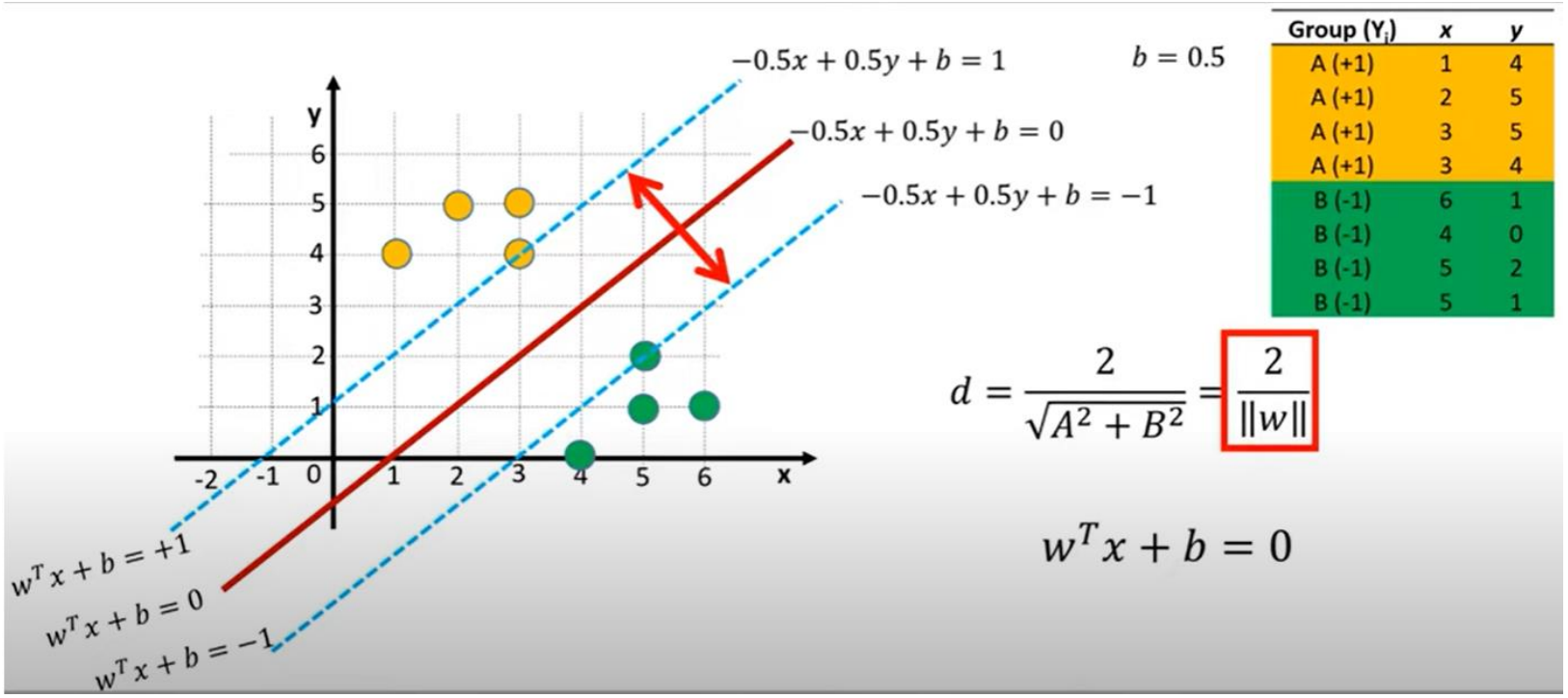
# An example (vi)




| Group ($Y_i$) | $x$ | $y$ |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

Prediction for a new sample

# An example (vii)

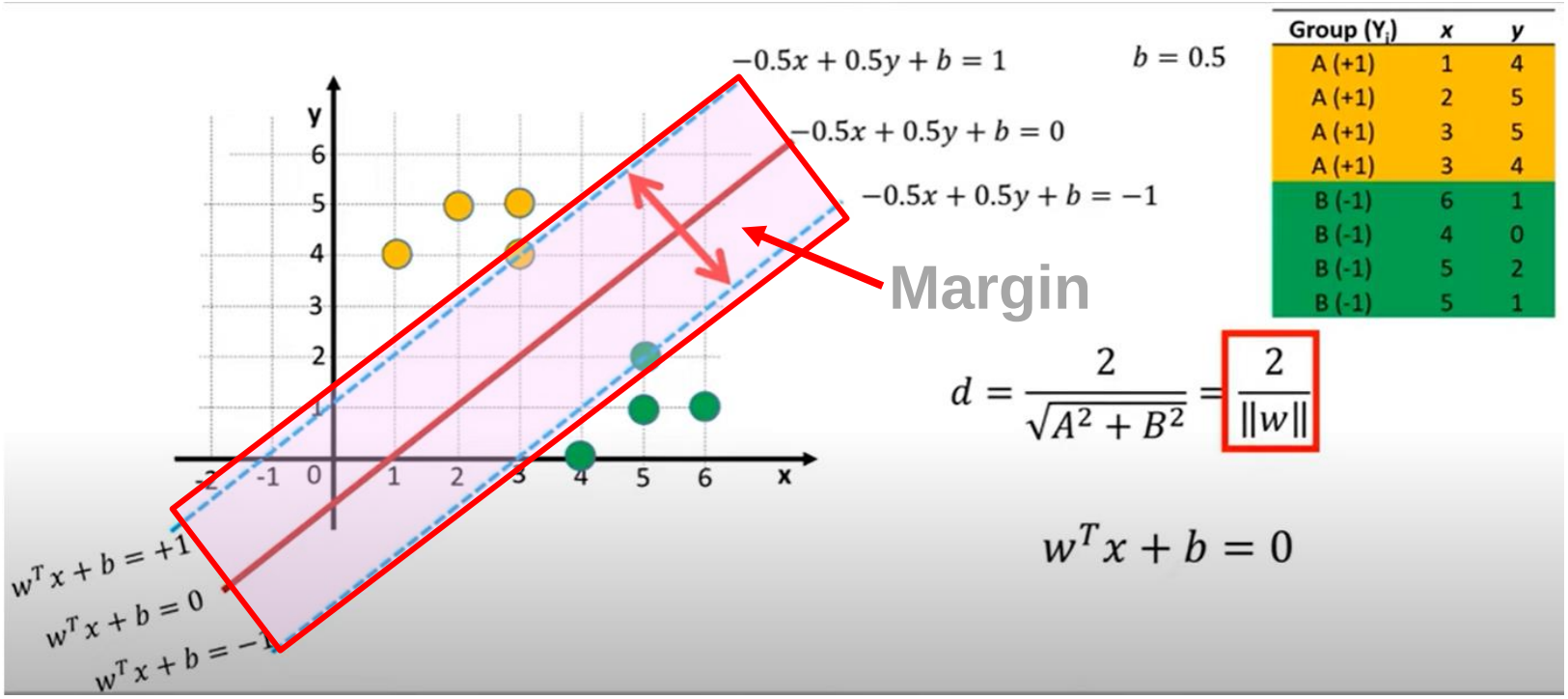



| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

$$d = \frac{2}{\sqrt{A^2 + B^2}} = \frac{2}{\|w\|}$$

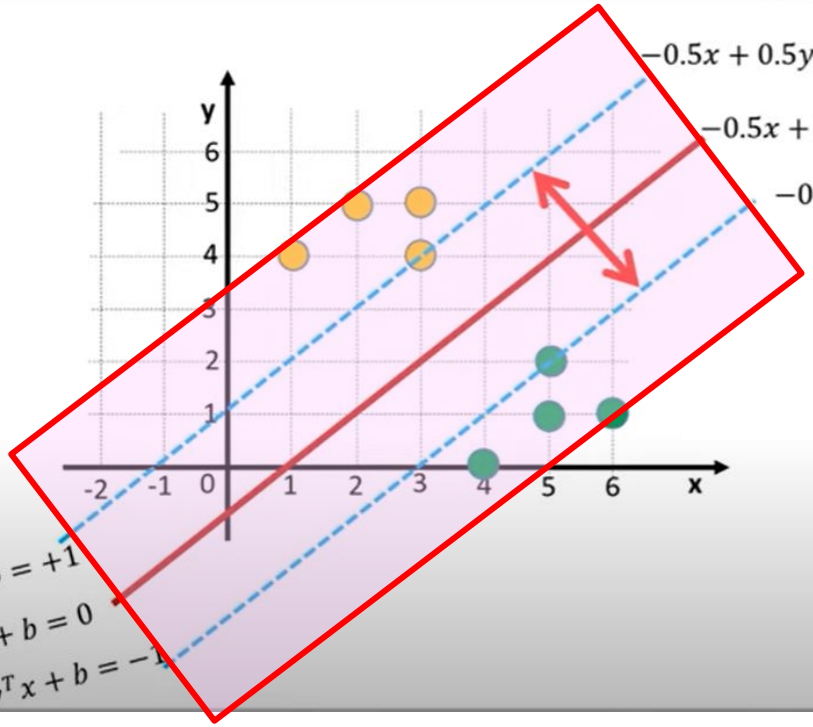
$$w^T x + b = 0$$

# An example (viii)




| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

# An example (ix)



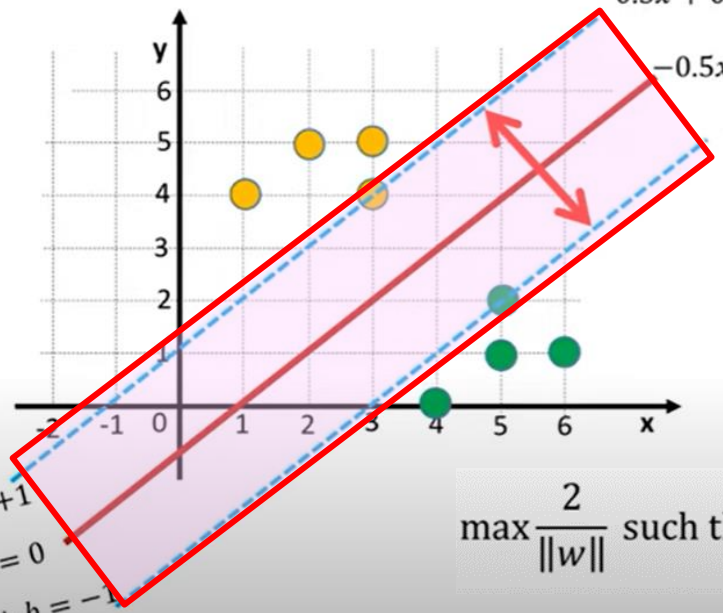

$b = 0.5$

| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

$$d = \frac{2}{\sqrt{A^2 + B^2}} = \frac{2}{\|w\|}$$

$$w^T x + b = 0$$

# An example (x)




$-0.5x + 0.5y + b = 1$

$-0.5x + 0.5y + b = 0$

$-0.5x + 0.5y + b = -1$

$b = 0.5$

| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

$$d = \frac{2}{\sqrt{A^2 + B^2}} = \frac{2}{\|w\|}$$

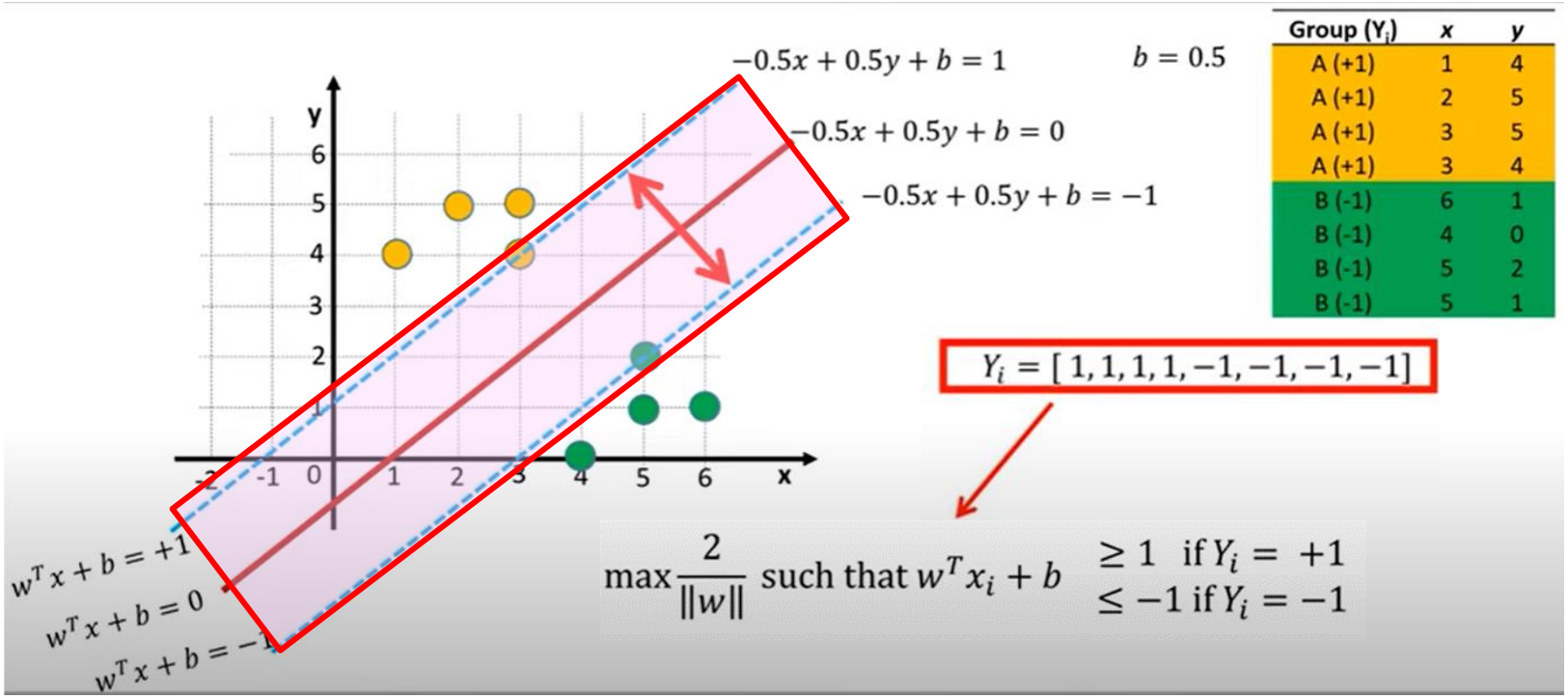
$w^T x + b = +1$

$w^T x + b = 0$

$w^T x + b = -1$

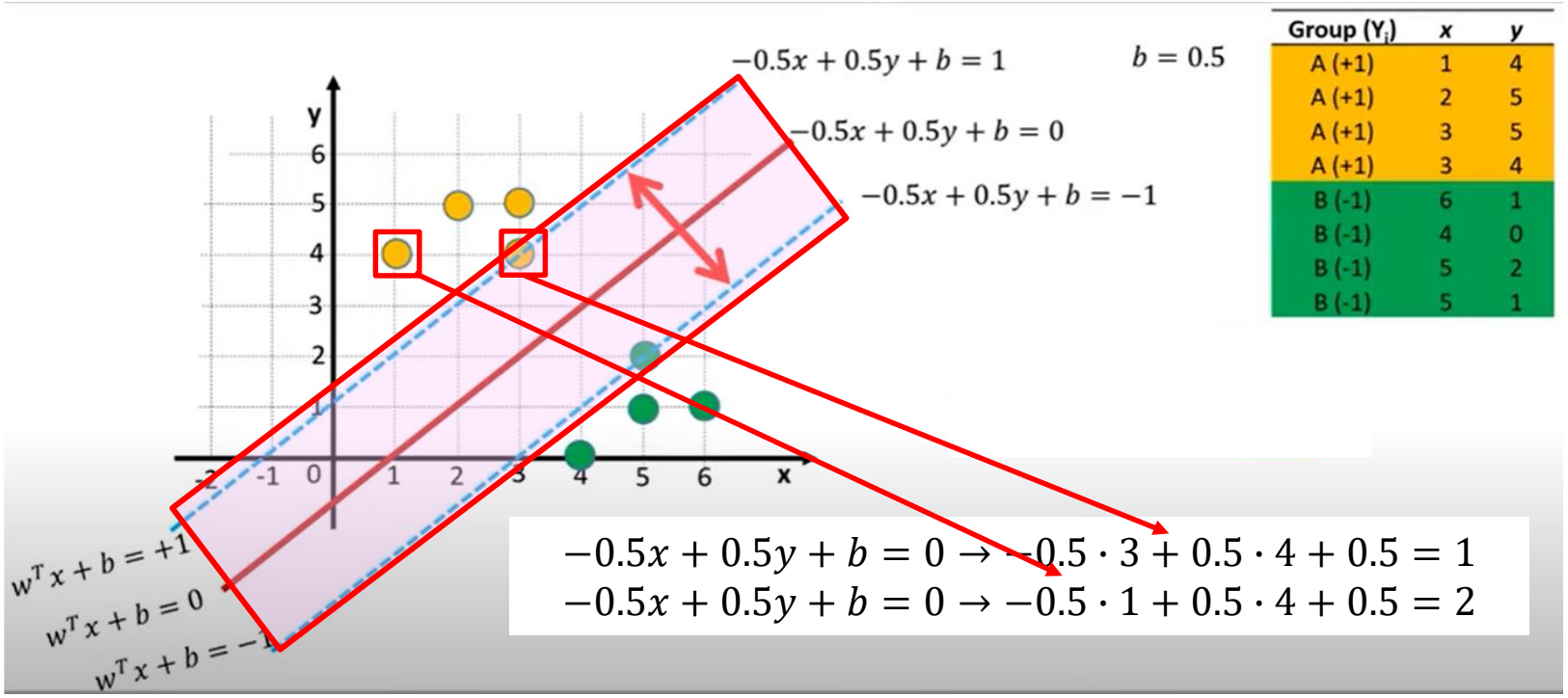
$$\max \frac{2}{\|w\|} \text{ such that } w^T x_i + b \begin{array}{l} \geq 1 \text{ if } Y_i = +1 \\ \leq -1 \text{ if } Y_i = -1 \end{array}$$

# An example (xi)



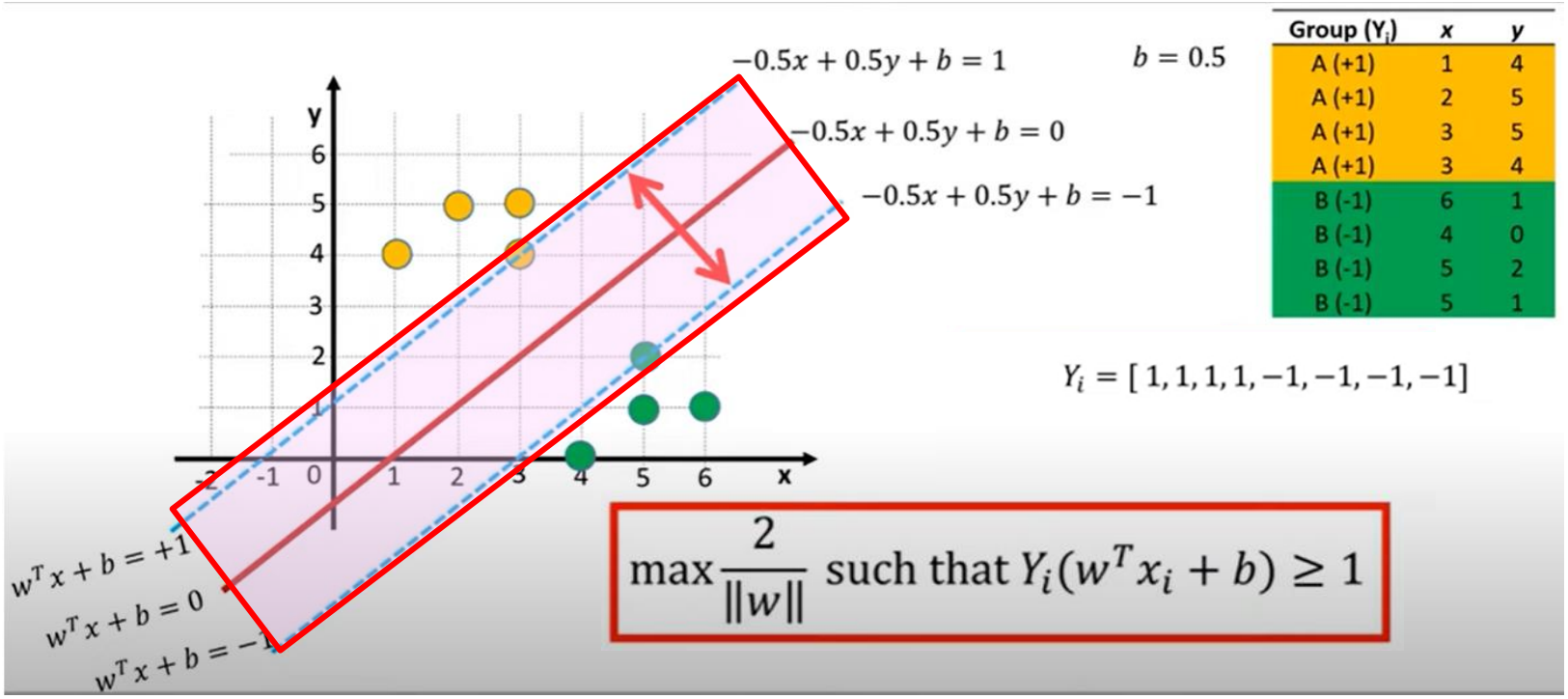

| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

# An example (xii)




| Group ($Y_i$) | $x$ | $y$ |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

# An example (xiii)




| Group ($Y_i$) | x | y |
|---|---|---|
| A (+1) | 1 | 4 |
| A (+1) | 2 | 5 |
| A (+1) | 3 | 5 |
| A (+1) | 3 | 4 |
| B (-1) | 6 | 1 |
| B (-1) | 4 | 0 |
| B (-1) | 5 | 2 |
| B (-1) | 5 | 1 |

# SVM with outliers

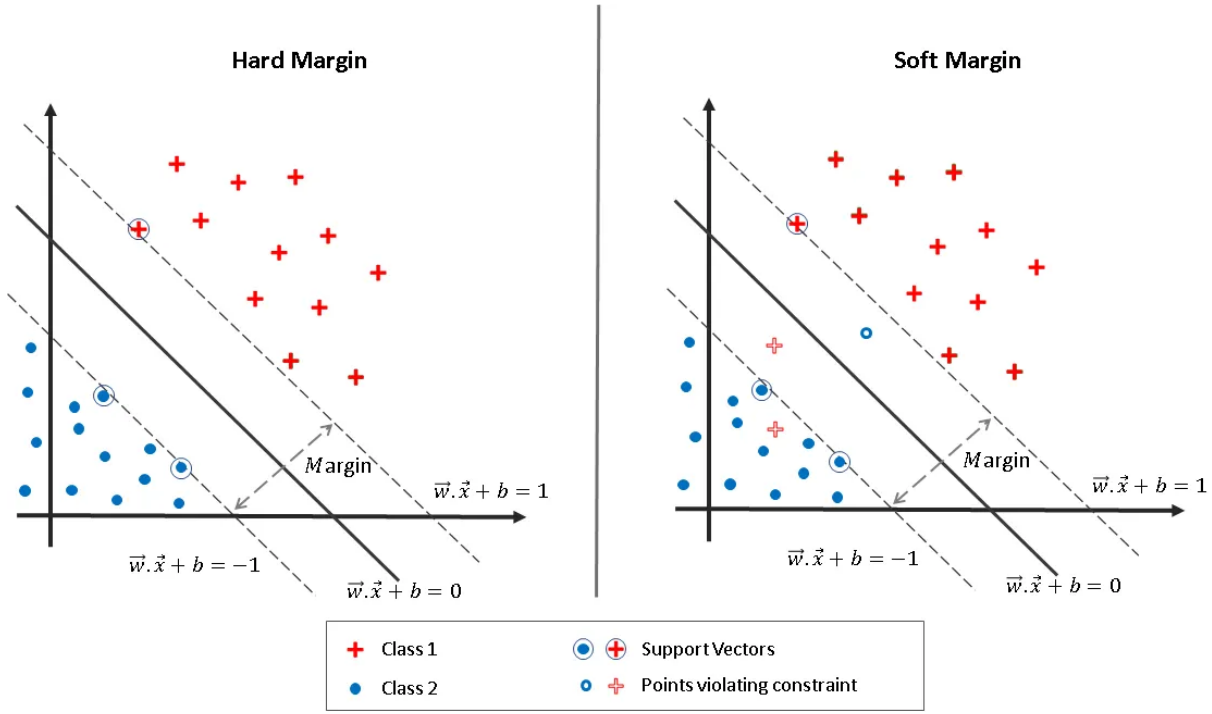
General situation in real-life applications:

- data is almost linearly separable or non-linearly separable
- there are outliers in the data

How to tackle the outliers problem?

to modify the optimization function in such a way that it allows few misclassifications (some points are inside or on the wrong side of the margin) by introducing two hyperparameters: a regularization factor $C$ and a slack variable $\xi_i$ for each data point $\mathbf{x}_i \rightarrow$ **SOFT MARGIN SVM**

# Hard margin vs soft margin



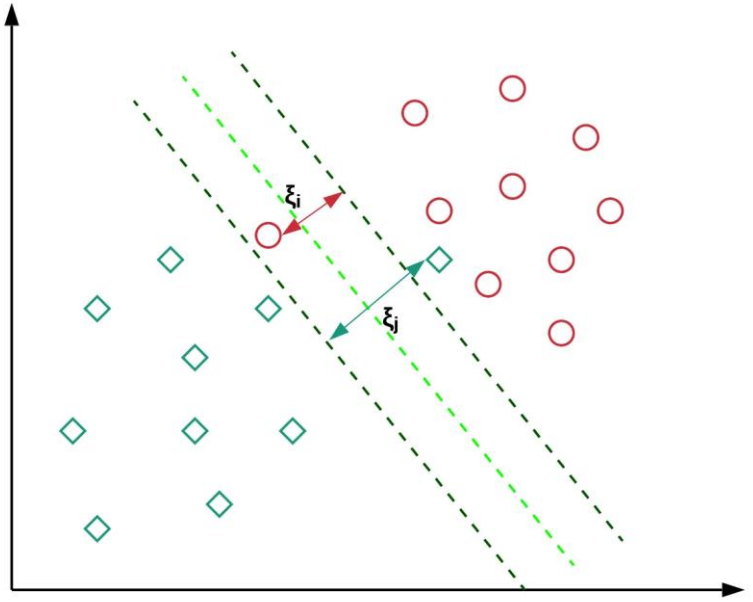

- In hard margin SVM, the points in a class must not lie within the two support hyperplanes:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \quad \forall i = 1, \ldots, n$$

# Slack variables

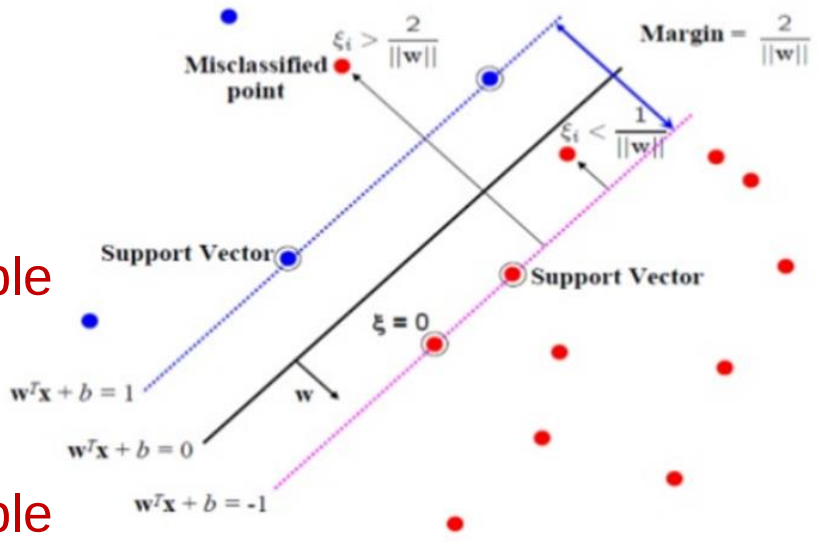
A slack variable $\xi_i \geq 0$ $(i = 1, \ldots, n)$ is the distance of $\mathbf{x}_i$ from its class's margin if $\mathbf{x}_i$ is on the wrong side of the margin; otherwise zero (this corresponds to linearly separable samples)

thus the points that are far away from the margin on the wrong side would get more penalty



# Slack variables (cont.)

- A slack variable $\xi_i = 0$ corresponds to a linearly separable sample
- A slack variable $0 < \xi_i < 1$ corresponds to a non-separable sample
- A slack variable $\xi_i > 1$ corresponds to a non-separable and misclassified sample




The sum of all slack variables $\sum_{i=1}^{n} \xi_i$ allows, in some way, measuring the cost associated with the number of non-separable examples

# Regularization factor

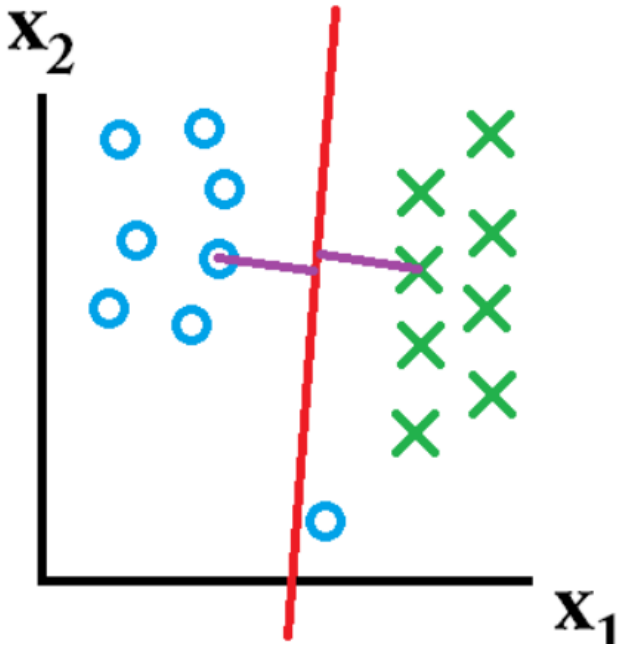
It is a constant, chosen by the user, which allows controlling the cost of non-separable samples in minimization, that is, it tries to set a balance between maximizing the margin and minimizing the misclassifications.

The value of C can be optimized by using some resampling method (e.g., cross-validation).

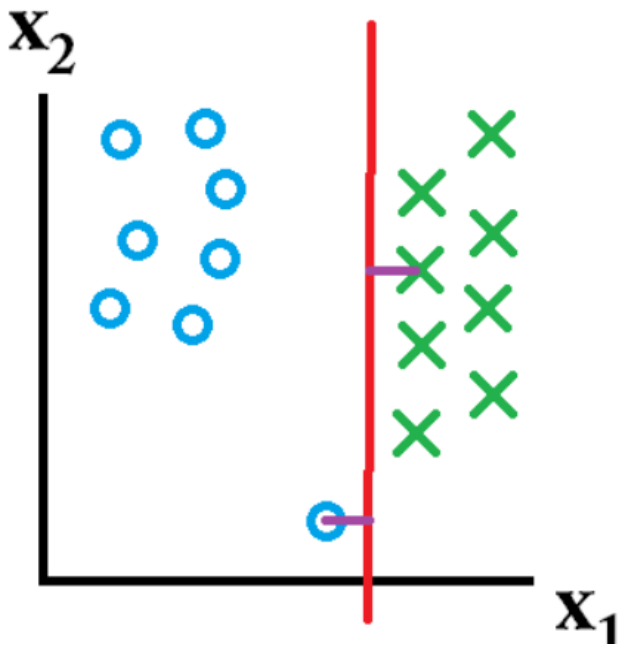
# Regularization factor (cont.)

- When *C* is small, the focus is more on obtaining a wider margin at the expense of allowing more misclassifications:
  - it would allow very large values of $\xi_i$: in the limit ($C \rightarrow 0$), all samples would be allowed to be misclassified ($\xi_i \rightarrow \infty$)
- When *C* is large, the focus is more on avoiding errors at the expense of resulting in a narrower margin:
  - it would allow very small values of $\xi_i$: in the limit ($C \rightarrow \infty$), we would considering the case of perfectly separable samples ($\xi_i \rightarrow 0$)

# Regularization factor (cont.)




**Small *C*** gives a large minimum margin, but a sample will be misclassified




**Large *C*** gives a much smaller margin, but all samples will be classified correctly

# Soft margin SVM: formulation

Now each data point needs to satisfy the following constraint:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i \quad \text{for} \quad i = 1, \ldots, n$$

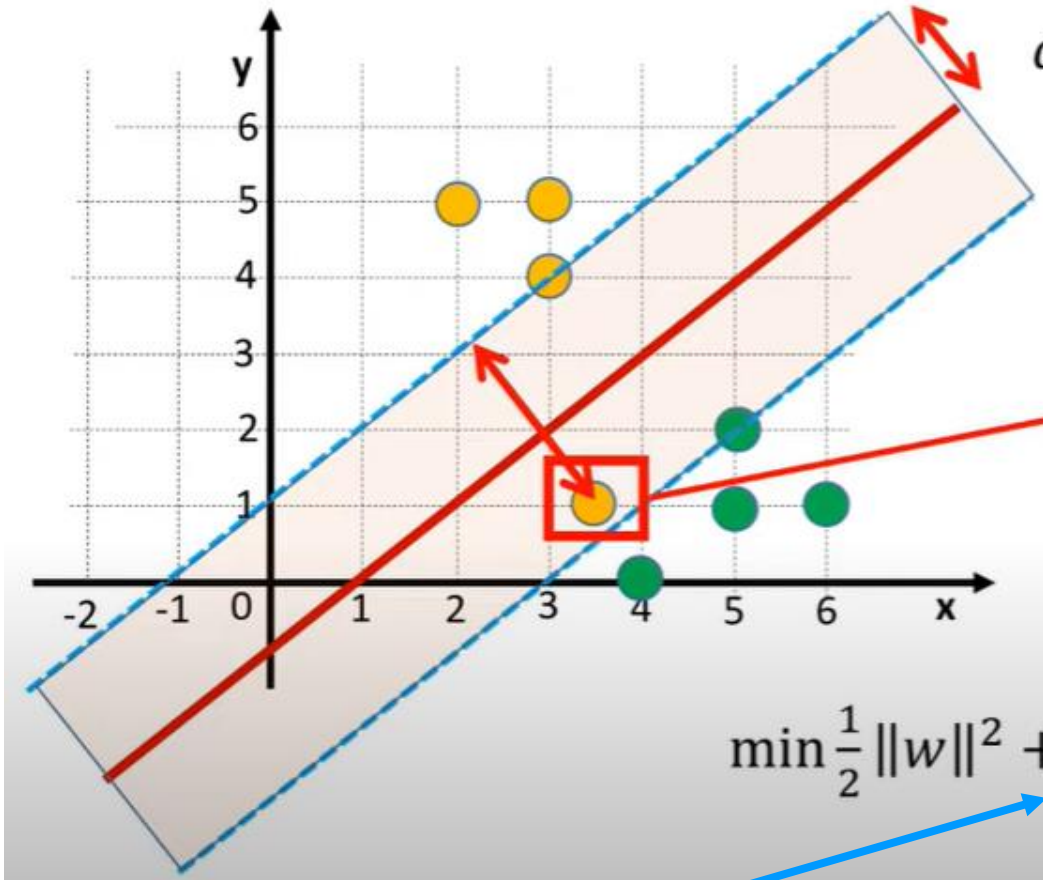
and the objective function to minimize will be:

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i$$

subject to $y_i(\mathbf{w}^T\mathbf{x}_i + b) + \xi_i - 1 \geq 0$

$$\xi_i \geq 0, \ \forall i = 1, \ldots, n$$

# An example (xiv)



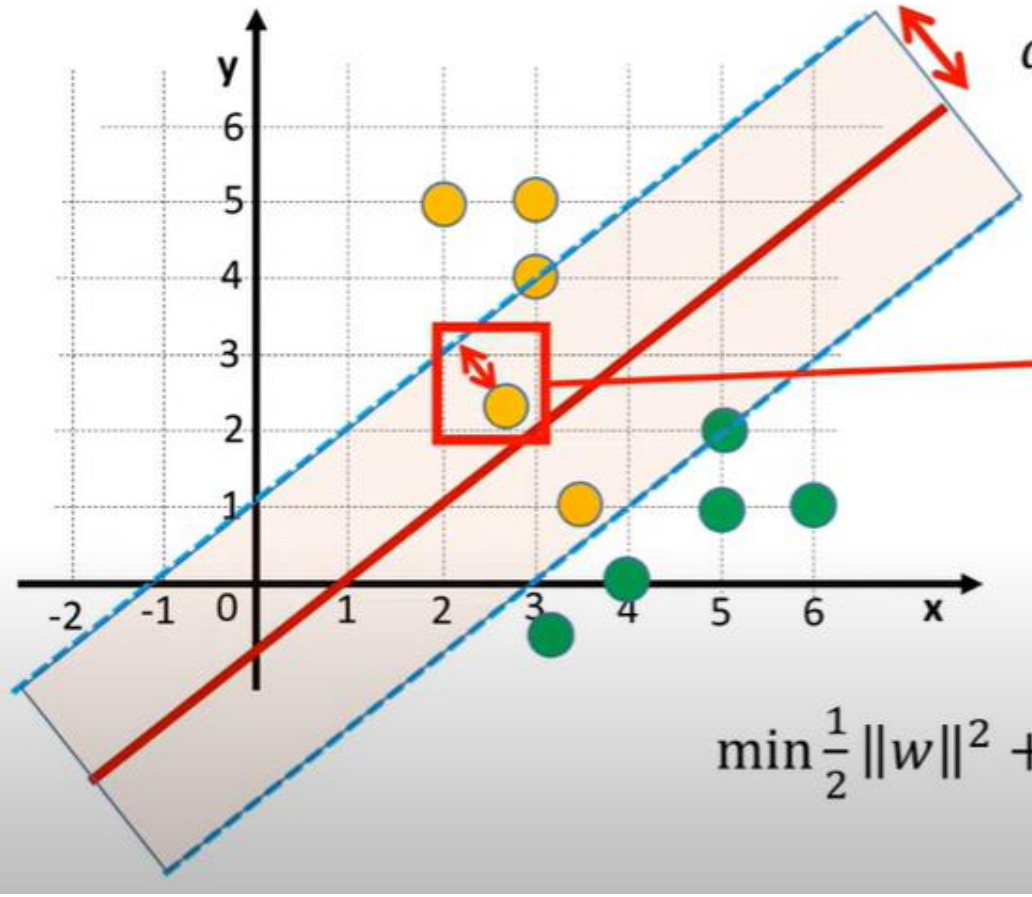

$$d = \frac{1}{\sqrt{A^2 + B^2}} = \frac{1}{\|w\|}$$

$$d = \frac{\varepsilon > 1}{\|w\|}$$

$$\min \frac{1}{2}\|w\|^2 + C\sum_i^N \varepsilon_i$$

This term is to allow misclassifications

# An example (xv)



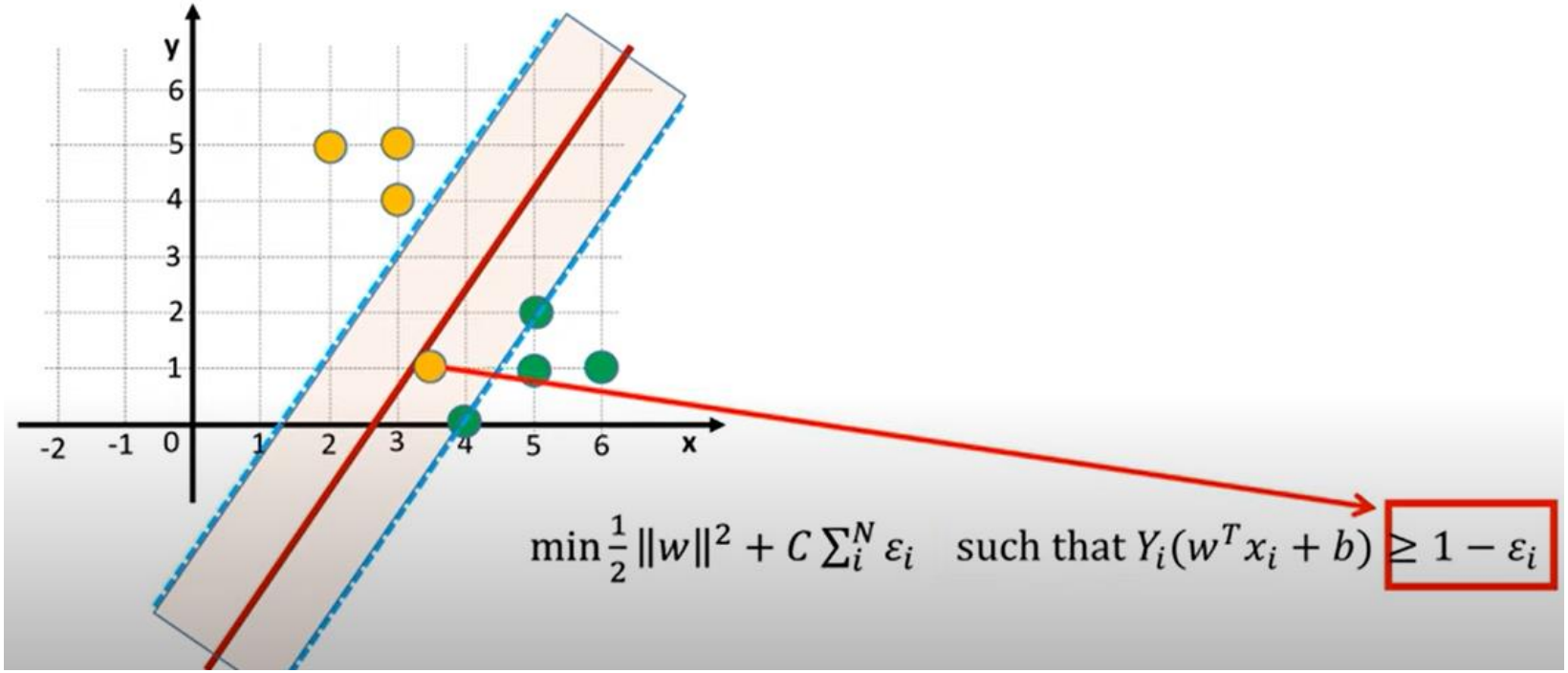

$$d = \frac{1}{\sqrt{A^2 + B^2}} = \frac{1}{\|w\|}$$

$$0 < \varepsilon < 1$$

$$d = \frac{\varepsilon}{\|w\|}$$

$$\min \frac{1}{2}\|w\|^2 + C \sum_i^N \varepsilon_i$$

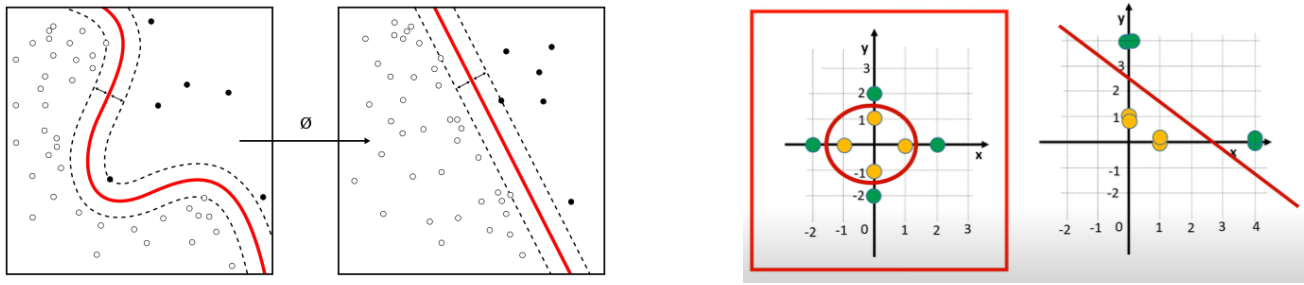
# An example (xvi)



# Non-linear SVM: the kernel trick

Solution to non-linear problems:

1. Map the data onto another feature space to convert the non-linear data to linear data: $\mathbf{x}_i \rightarrow \phi(\mathbf{x}_i)$



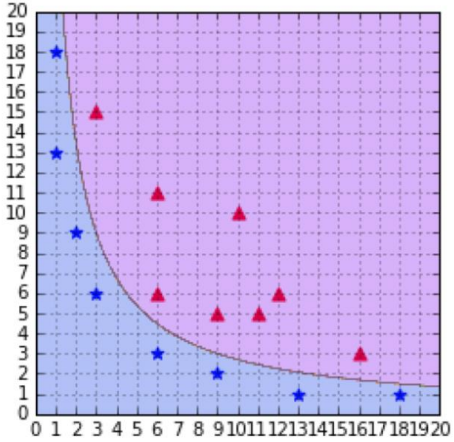

2. Solve a linear SVM in the new feature space

Therefore, the kernel trick is to define a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$

# Non-linear SVM: the kernel trick (ii)

- Polynomial kernel of degree *d*:

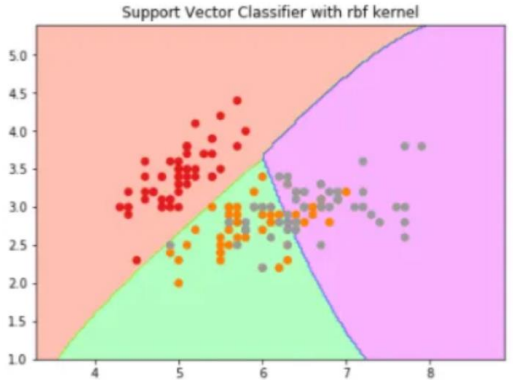
$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \cdot \mathbf{x}_j + 1)^d$$



- RBF kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$$

where $\gamma > 0$, and $\|\mathbf{x}_i - \mathbf{x}_j\|$.



# Non-linear SVM: the kernel trick (iii)

- A special case of RBF kernel is $\gamma = {}^{1}/_{2\sigma^2} \rightarrow$ Gaussian kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$$

where $\sigma$ is the variance.

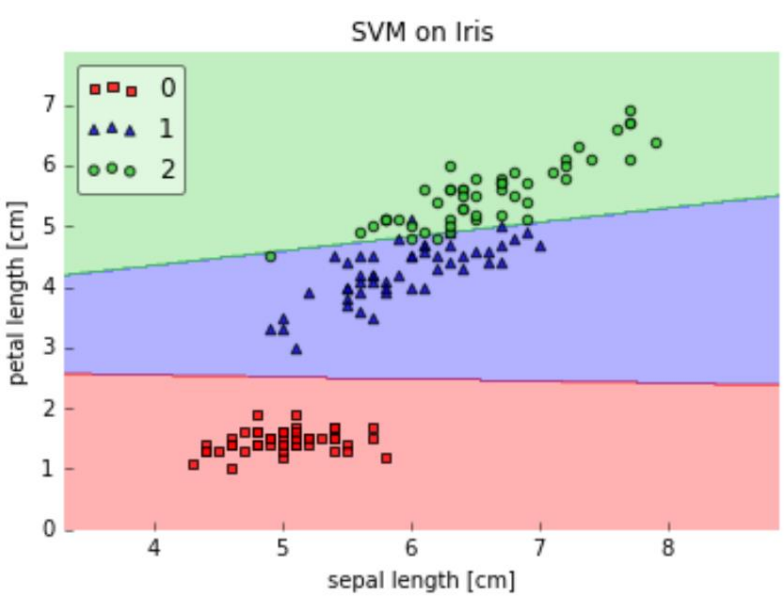
# Non-linear SVM: the kernel trick (iv)

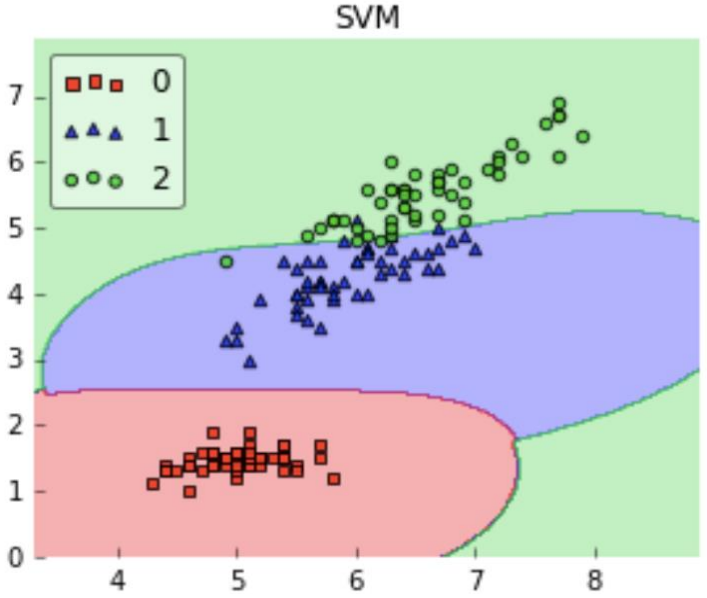
How to choose the right kernel?

- if the data set is linearly separable, then you must opt for a linear function because it is very easy to use and the complexity is much lower compared to other kernel functions
- so, start with a hypothesis that data is linearly separable and choose a linear function. Then use an RBF kernel function (polynomial kernel is rarely used due to poor efficiency)
- if linear and RBF both give approximately similar results, choose the linear SVM. Otherwise, choose the RBF kernel

# Non-linear SVM: the kernel trick (v)

An example with similar results for both linear and RBF functions



# Non-linear SVM: the kernel trick (vi)

An example where a linear SVM gives poor results, while the RBF kernel makes a correct decision boundary