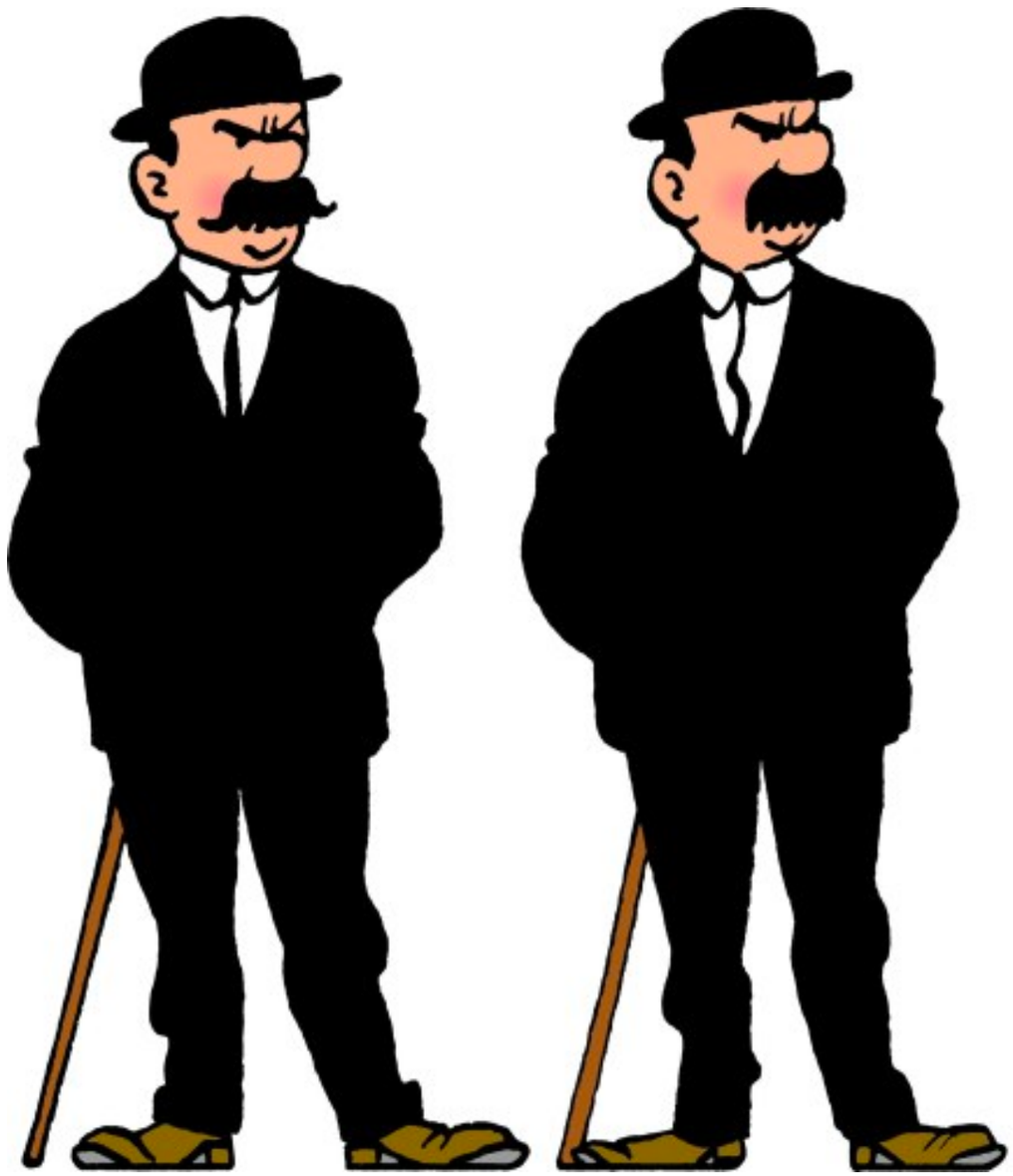
TÉCNICO LISBOA

Planning, Learning and Decision Making

Lecture 14. Learning from examples - Similarity-based approaches



Similarity-based approaches

Approaches so far: DT

- Is it:
  - … a discriminant function?
  - … a discriminative model?
  - … a generative model?

Approaches so far: DT

- How is it trained?
  - Sequence of tests is selected greedily
  - Criterion for selection is **gain of information** on training data

Approaches so far: DT

- How does it work?
  - Each point *x* goes through a sequence of “tests” until class is found
  - **Intuition:** “Tests” allow a discrimination of the classes

Approaches so far: LR

- Is it:
  - … a discriminant function?
  - … a discriminative model?
  - … a generative model?

Approaches so far: LR

- How is it trained?
  - Parameters of the regression selected to maximize log likelihood of the data
  - Optimization usually relies on local search (gradient, Newton, etc.)

Approaches so far: LR

- How does it work?
  - Given a point *x*, …
    - … compute probability of each class
    - … select class that maximizes probability
  - **Intuition:** The estimated probability is close to the real world

Approaches so far: NB

- Is it:
  - … a discriminant function?
  - … a discriminative model?
  - … a generative model?

Approaches so far: NB

- How is it trained?
  - Priors estimated directly from data (class ratios)
  - Likelihood estimated using simple maximum likelihood approaches (e.g., ratios, Gaussian approximations)

Approaches so far: NB

- How does it work?
  - Given a point *x*, …
    - … compute **posterior** probability of each class
    - … select class that maximizes posterior probability
  - **Intuition:** The learned probabilities are close to the real world

Bottom line…

- Models assume specific structure on the data:
  - Attributes are good discriminants

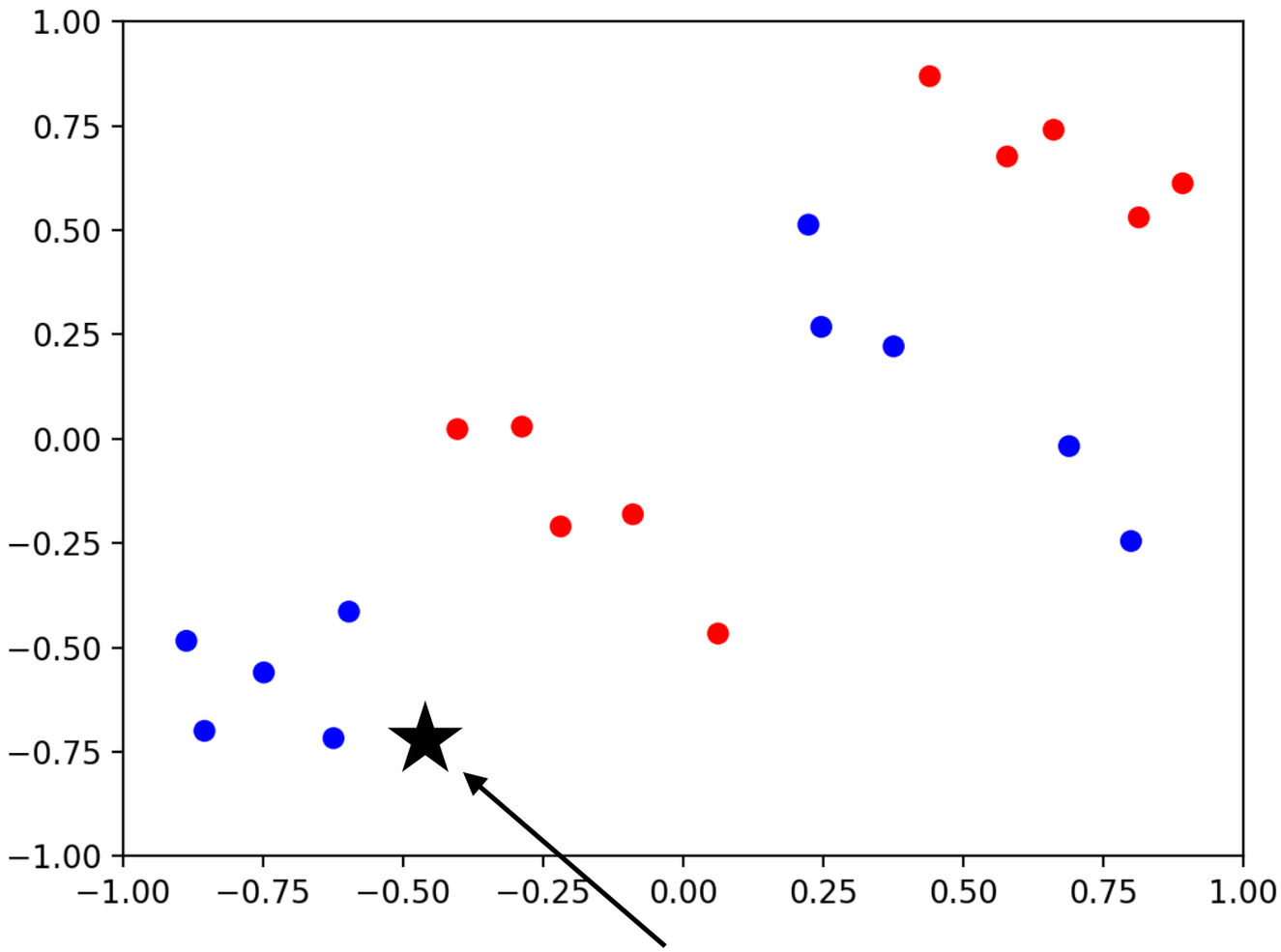
  or
- Inherent classes are probabilistic

However…

- We could make other assumptions…

Example

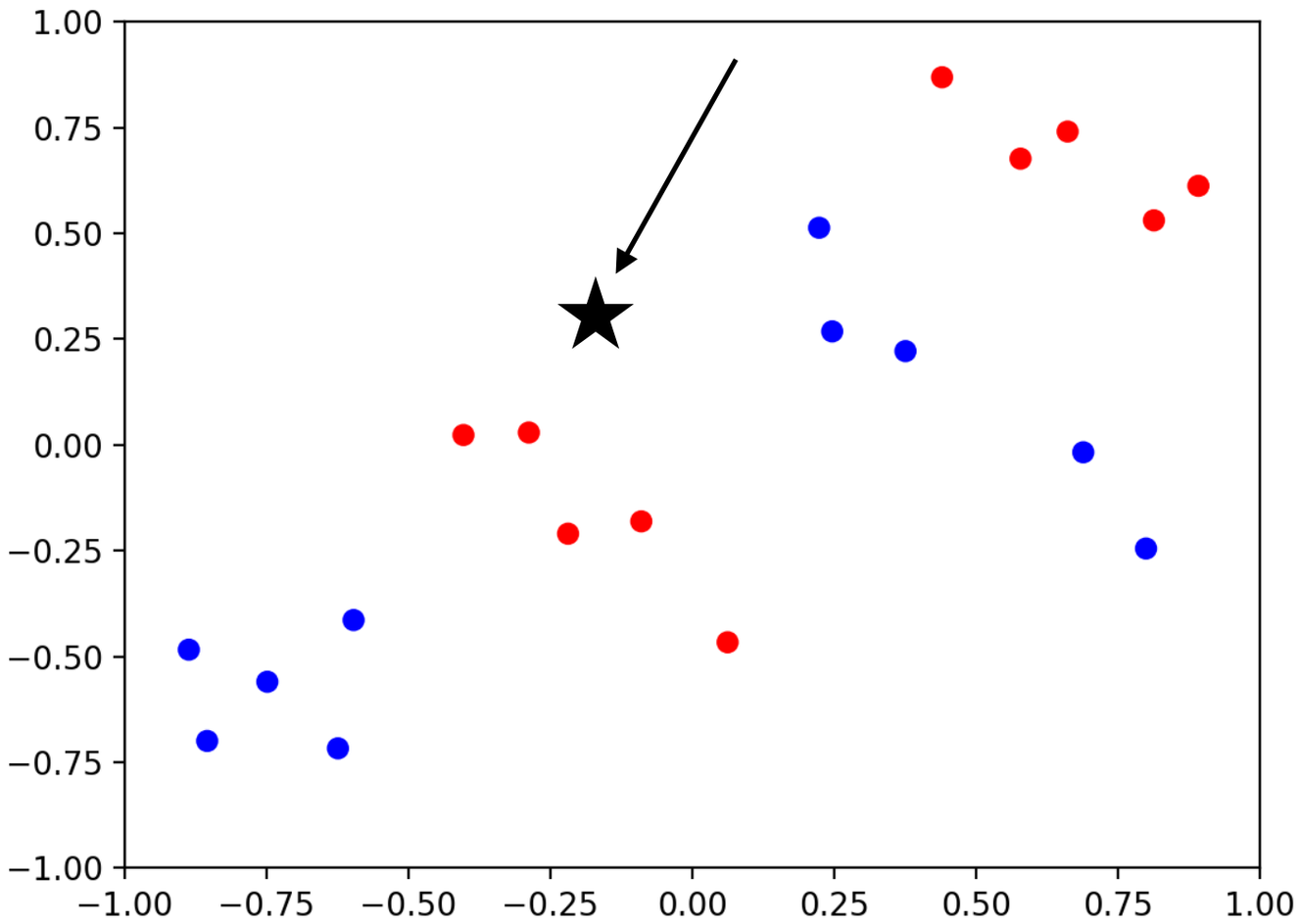
- What class do you think this point is?



Blue •

Example

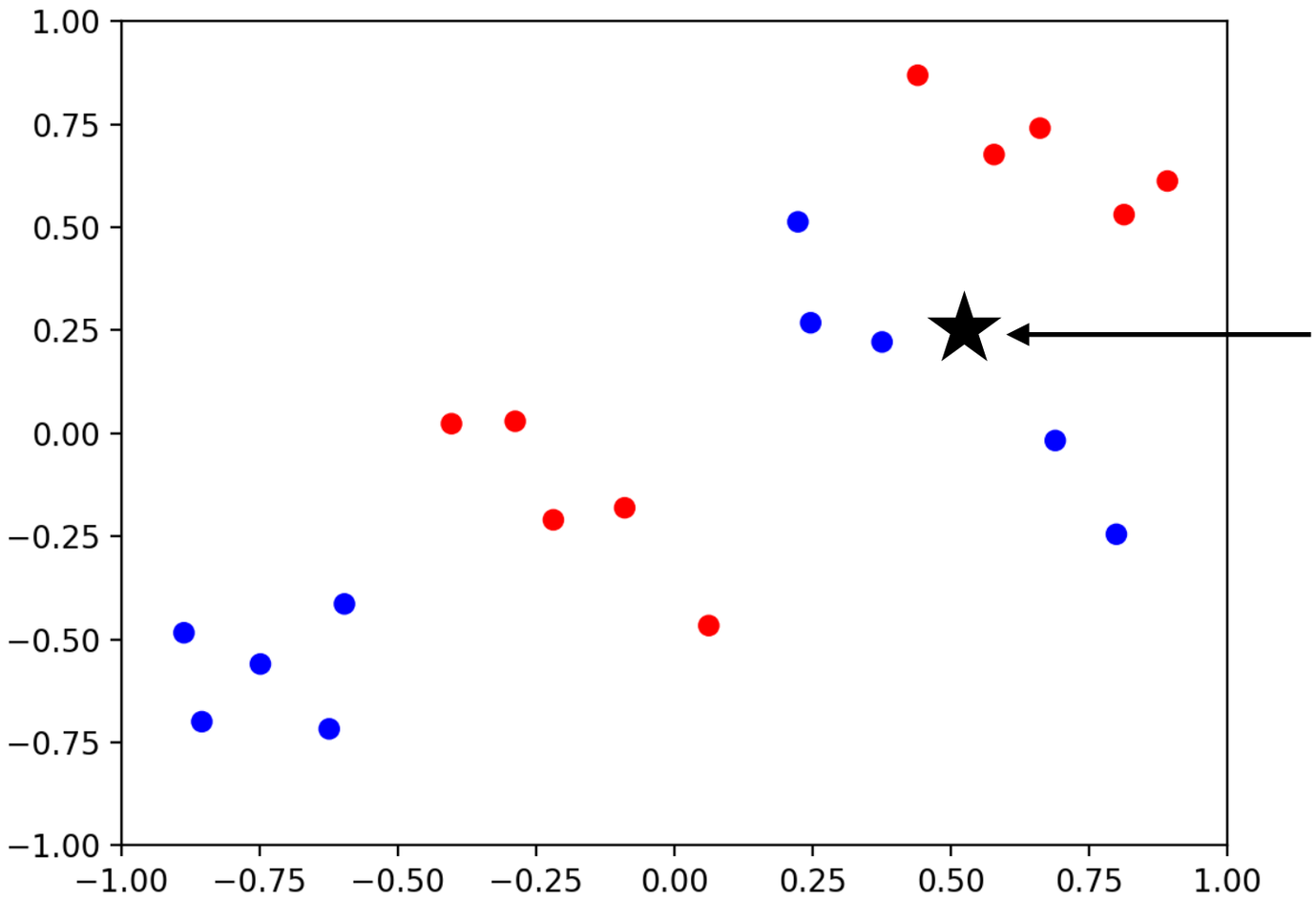
- What about this one?



Red ·

Example

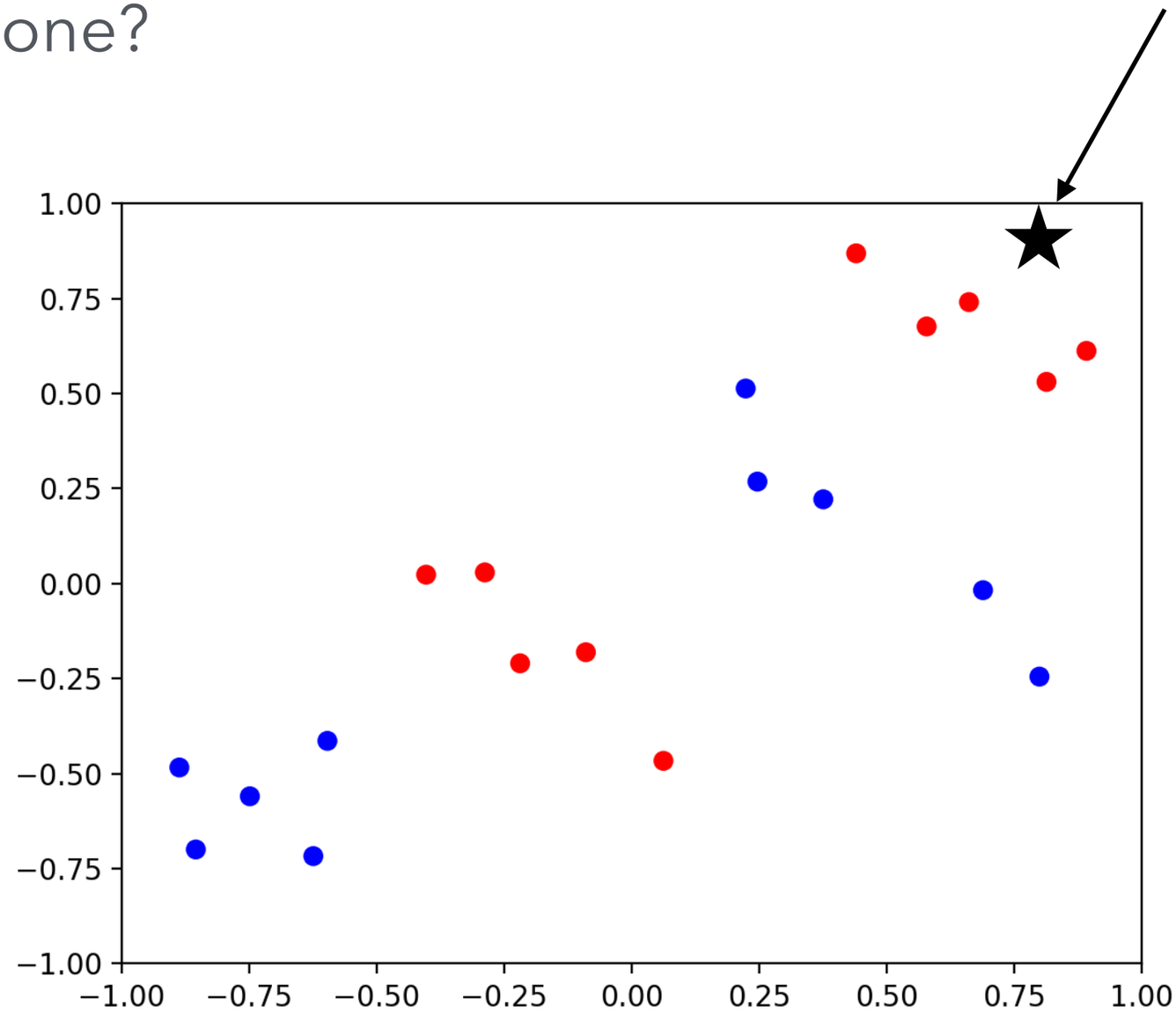
- This one?



Blue •

Example

- And this one?



Red •

Why?

Similarity

- You immediately assume that “geometry matters”:
  - Points close by are alike
  - We can extend that intuition to build a classifier

k Nearest Neighbors

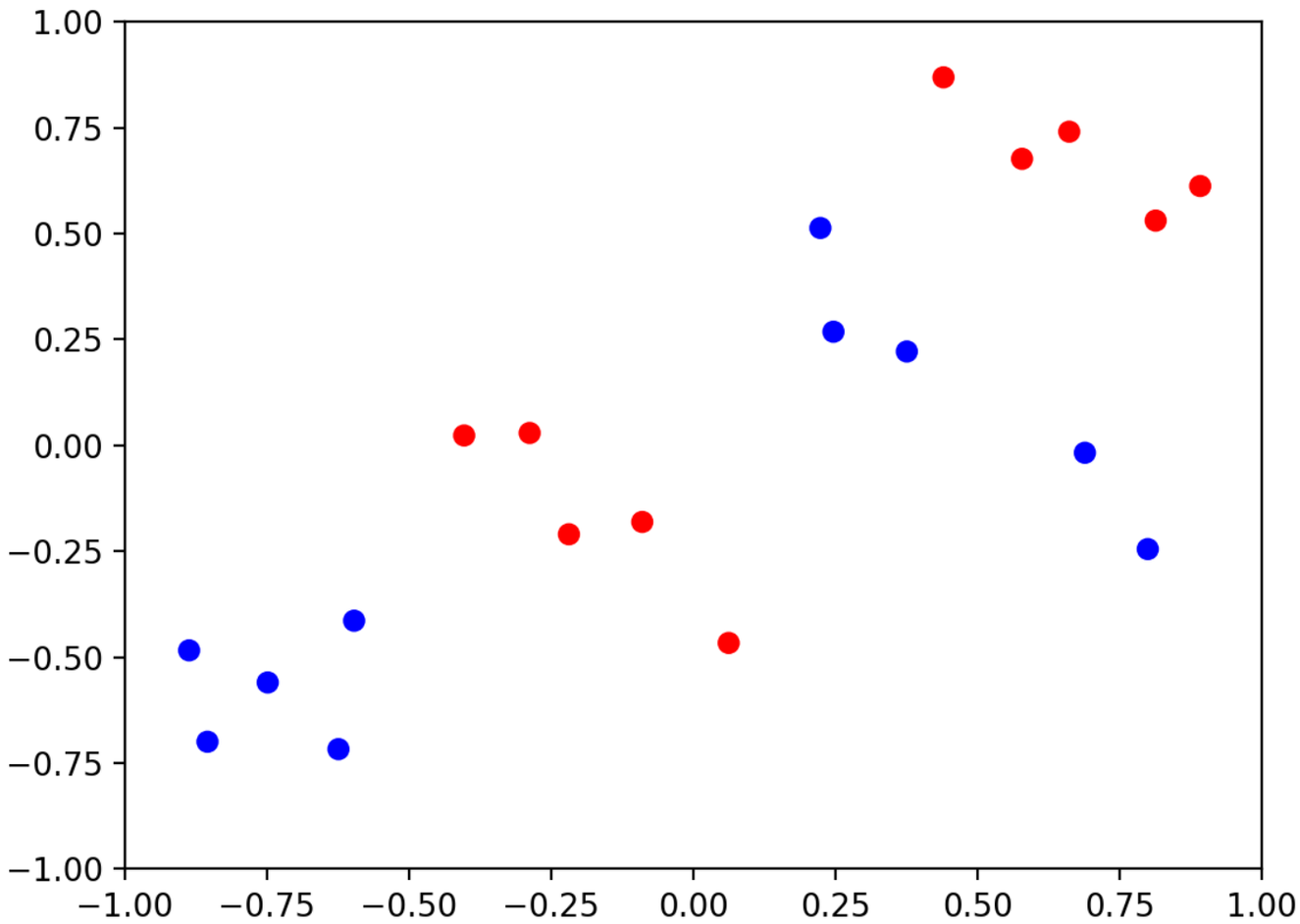
- No “explicit training”
- Dataset is kept in memory

k Nearest Neighbors

- How does it work?
  - When new point arrives, check *k* closest points
  - Select the class of the majority (odd *k*)

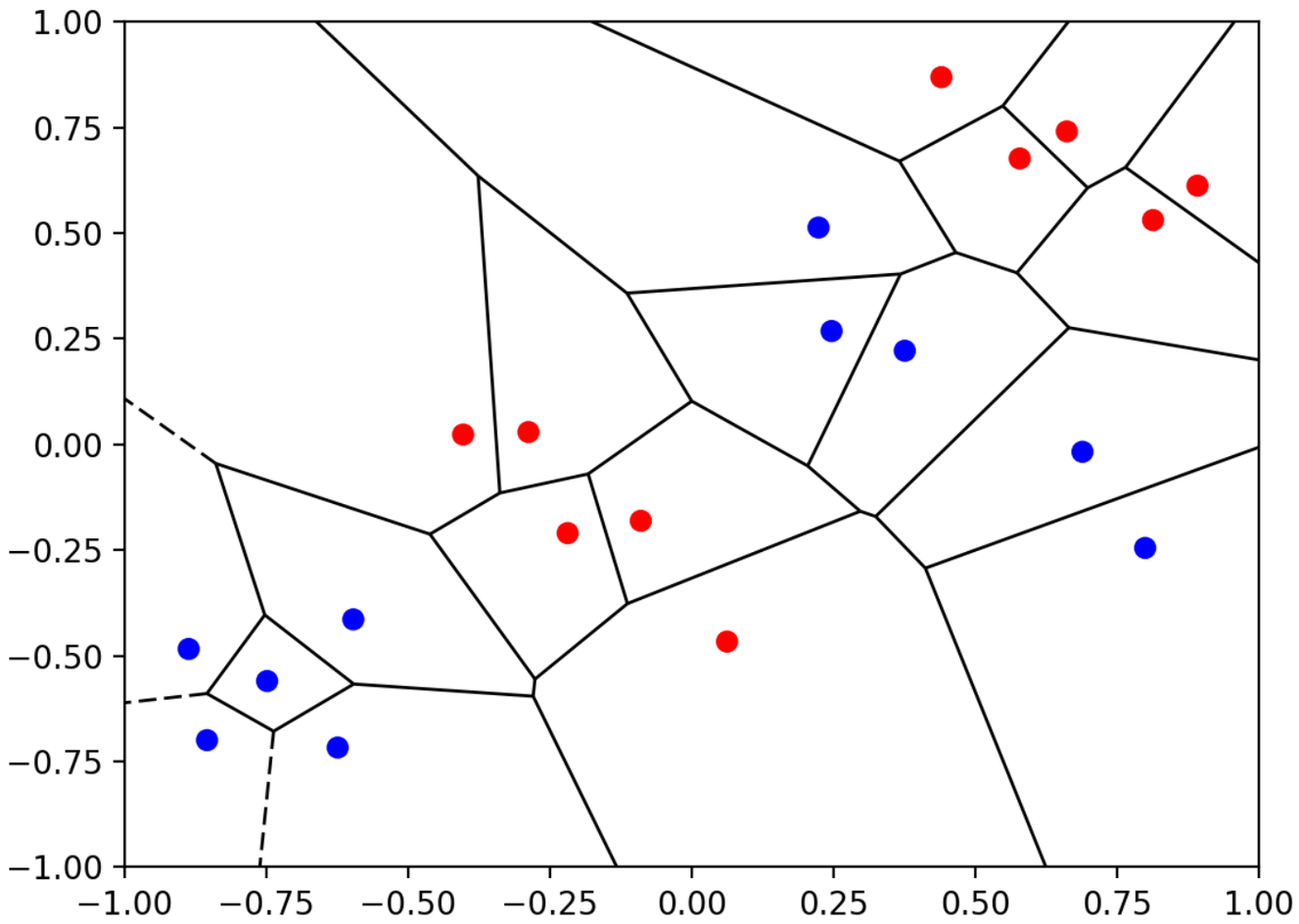
Example

- 1-NN


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

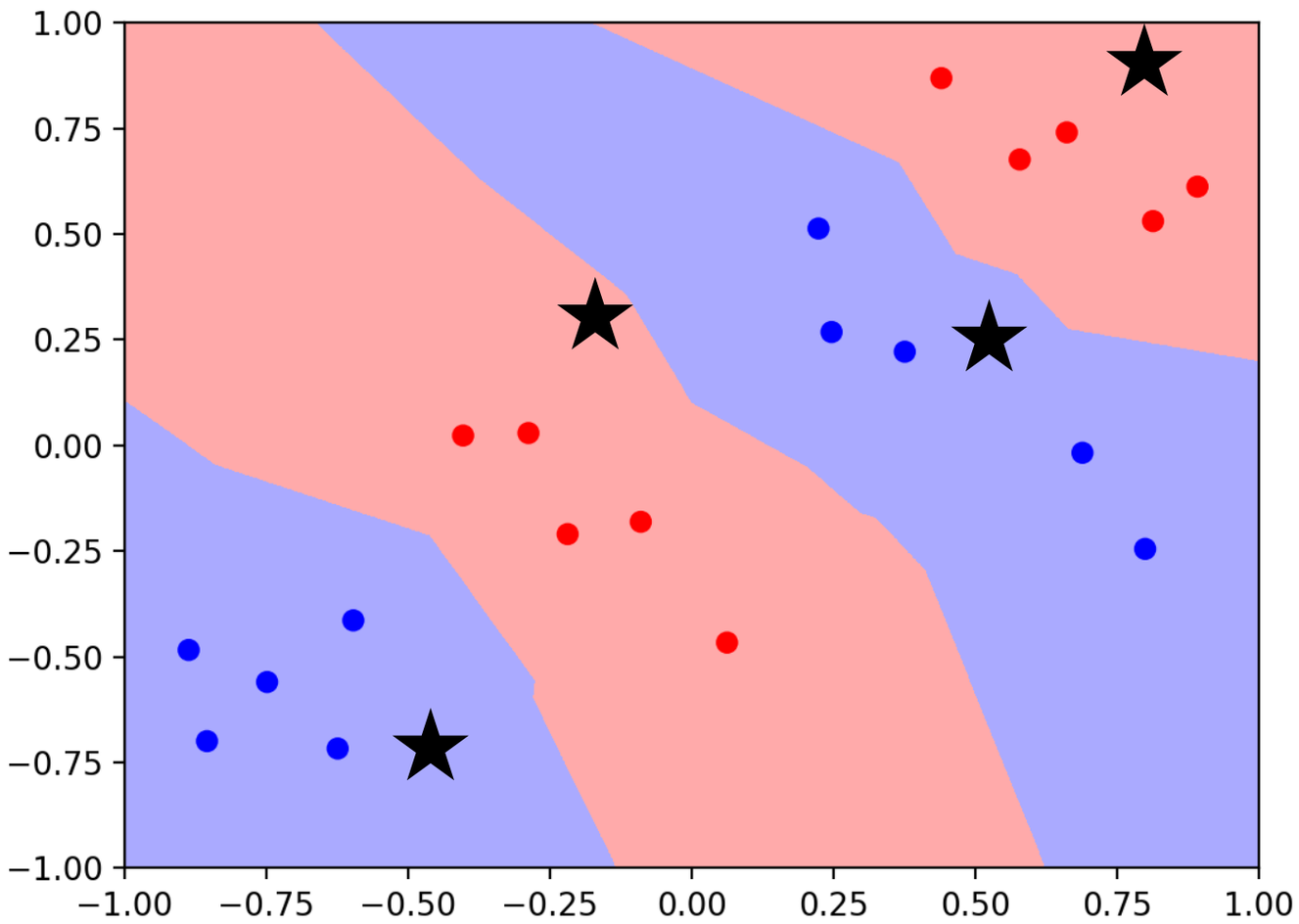
Example

- 1-NN


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

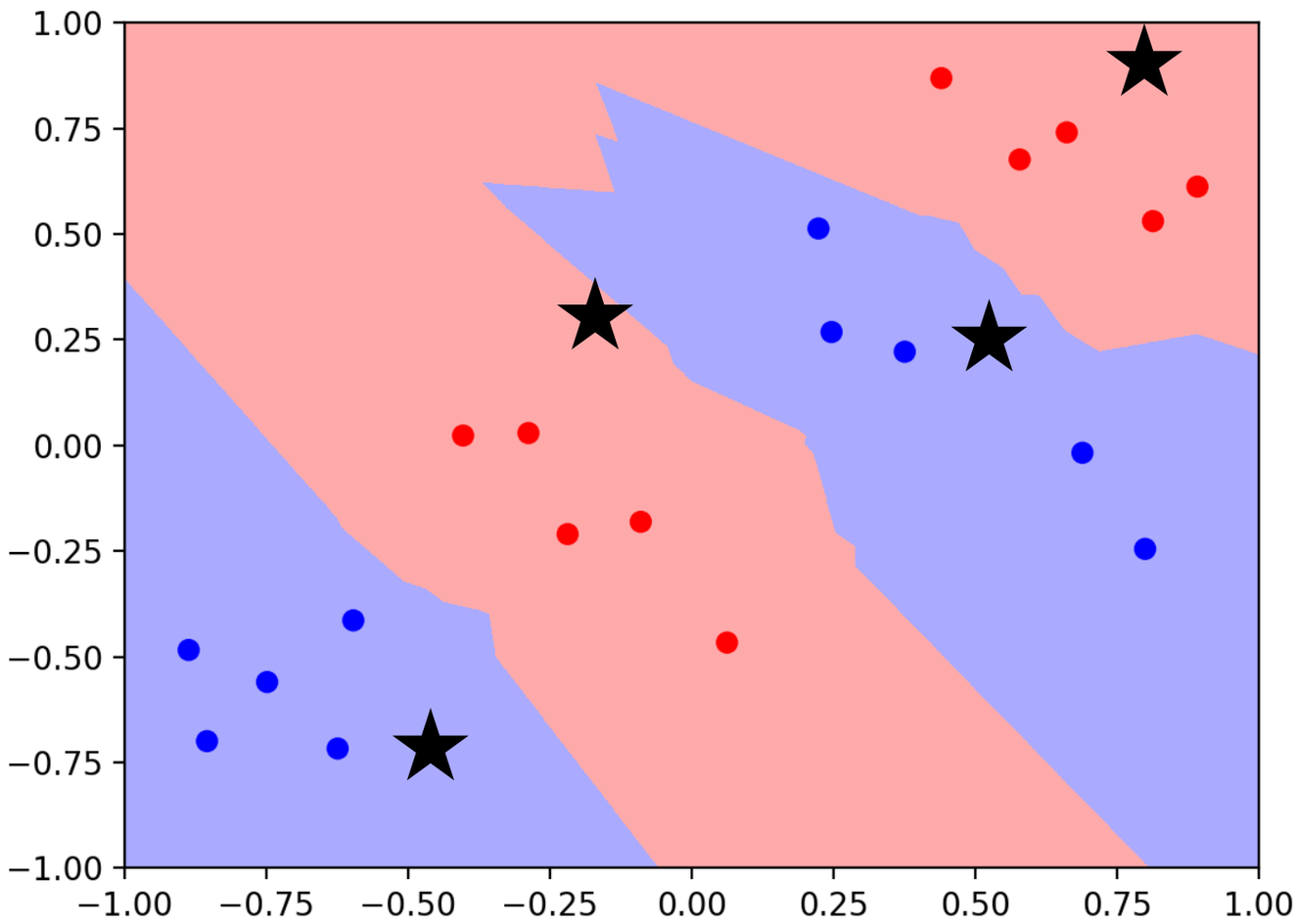
Example

- 1-NN


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

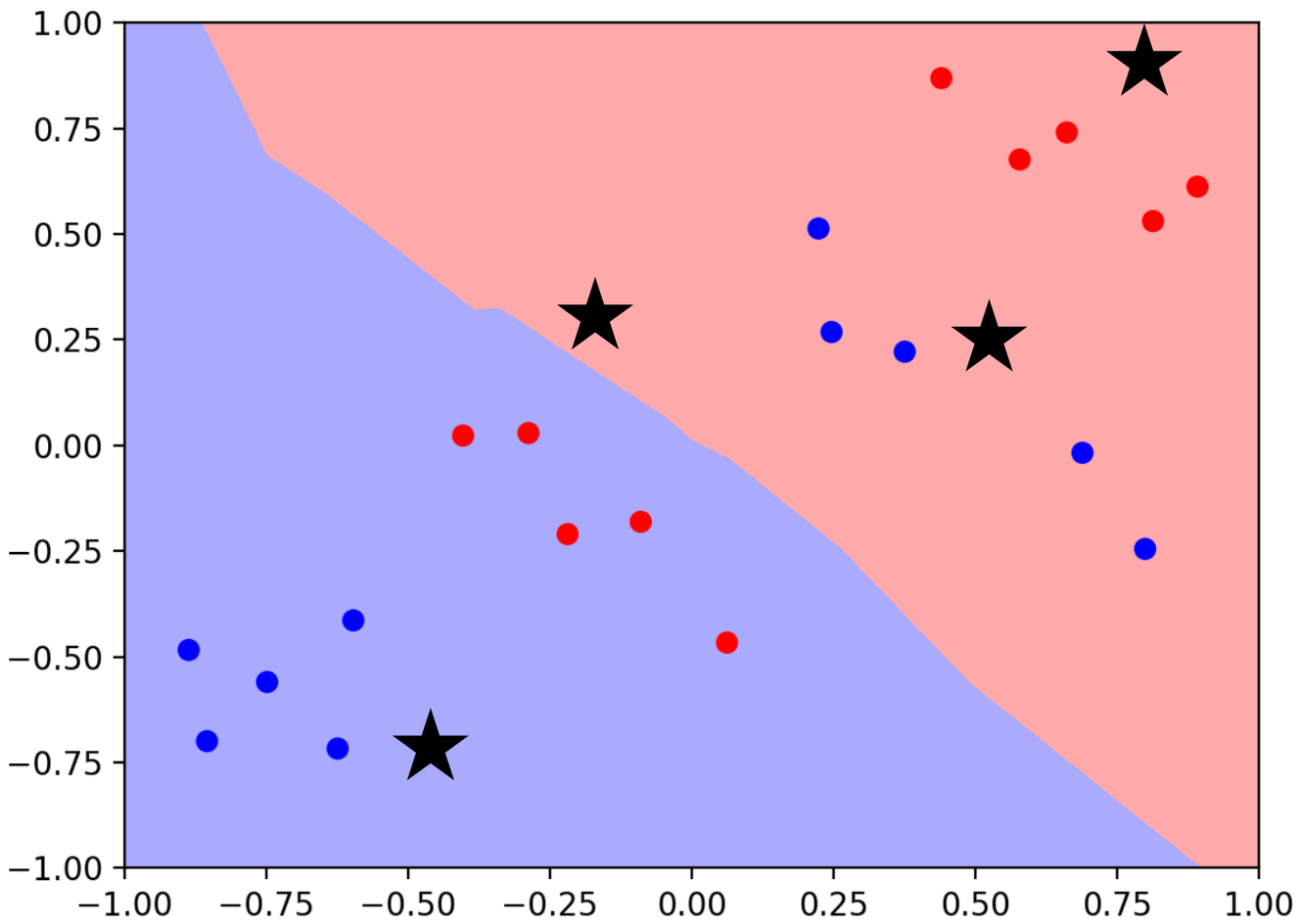
Example

- 5-NN


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

Example

- 15-NN



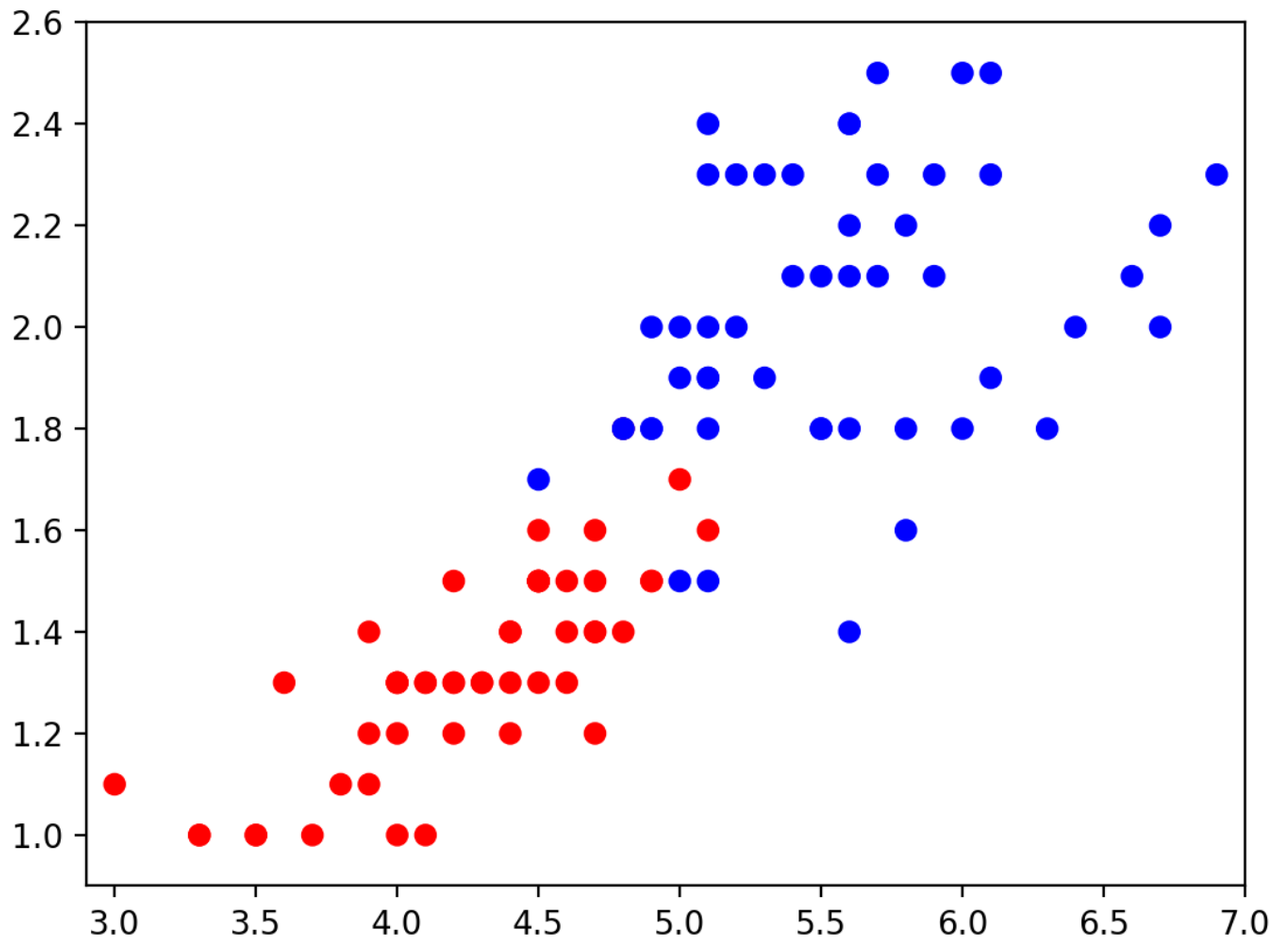
Another example

Iris versicolor

Iris virginica

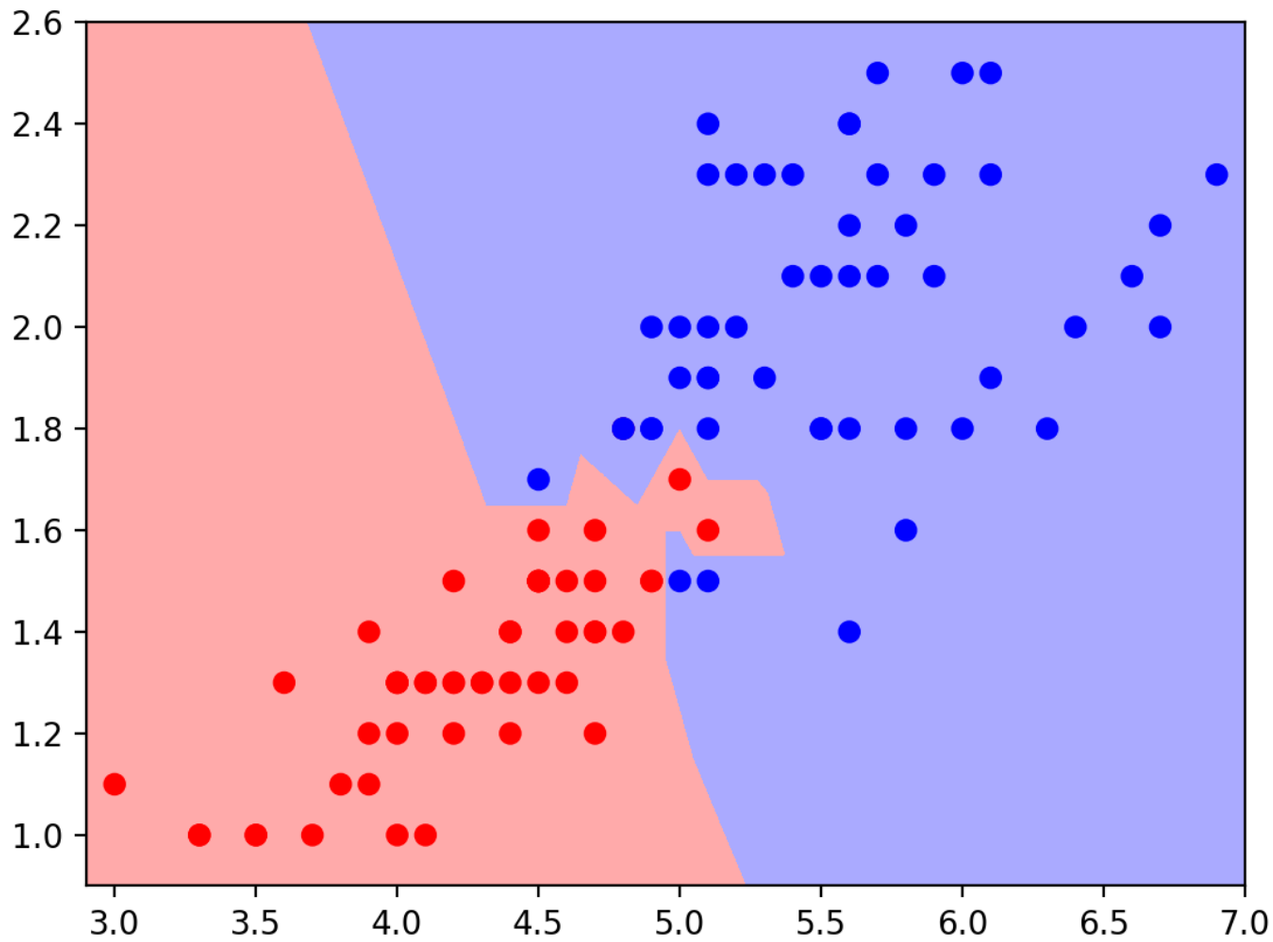
Another example

- 1-NN



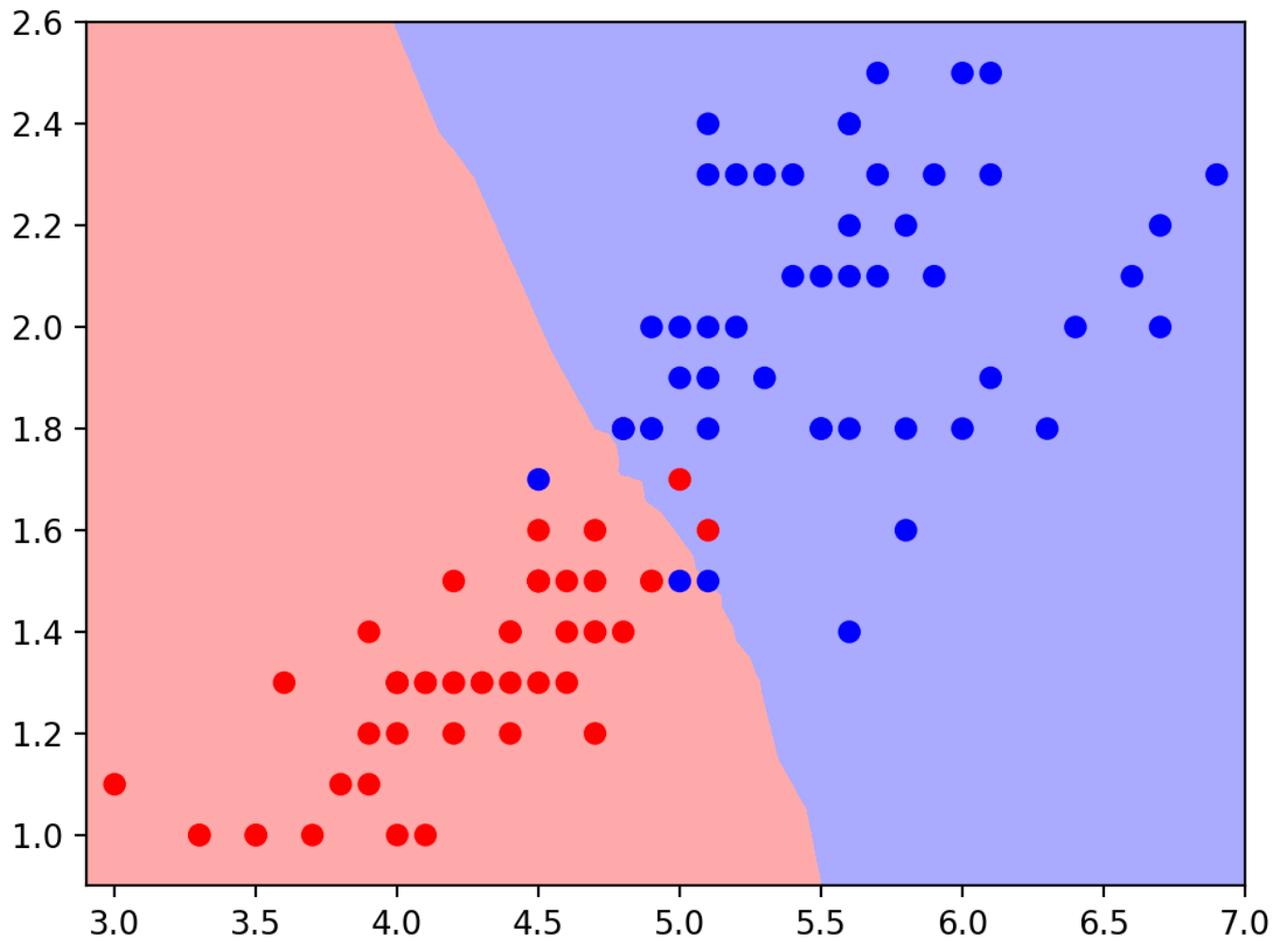
Another example

- 1-NN



Another example

- 15-NN



k Nearest Neighbors 2.0

- An "improved" alternative is to assign more importance to points closer than points away
  - Each point "votes" proportionally to distance

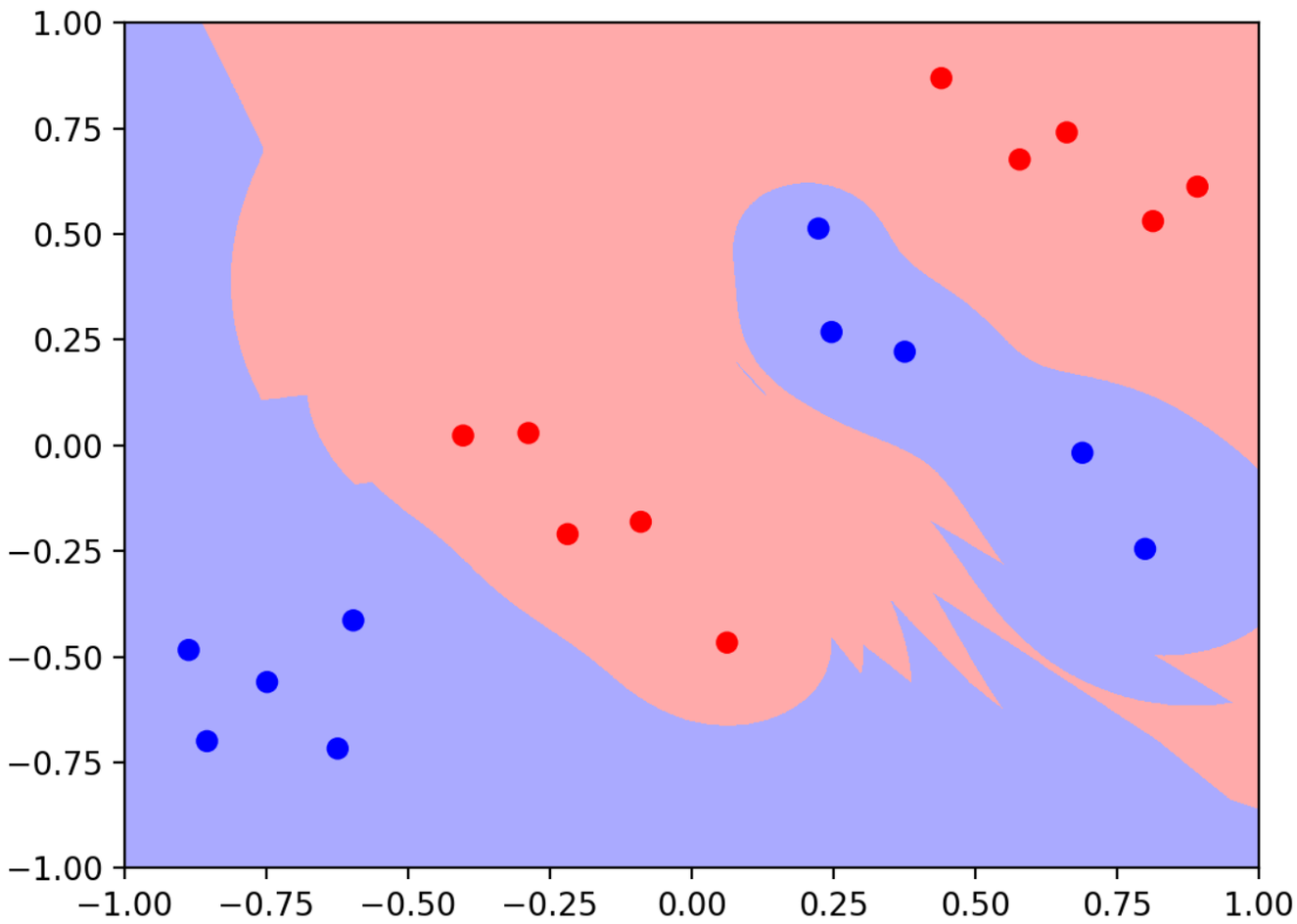
$$y = \operatorname{sgn}\left(\sum_{n=1}^{N} \alpha y_n k(\boldsymbol{x}_n, \boldsymbol{x})\right)$$

+1 or −1 depending on class

Larger for closer $\mathbf{x}_n$

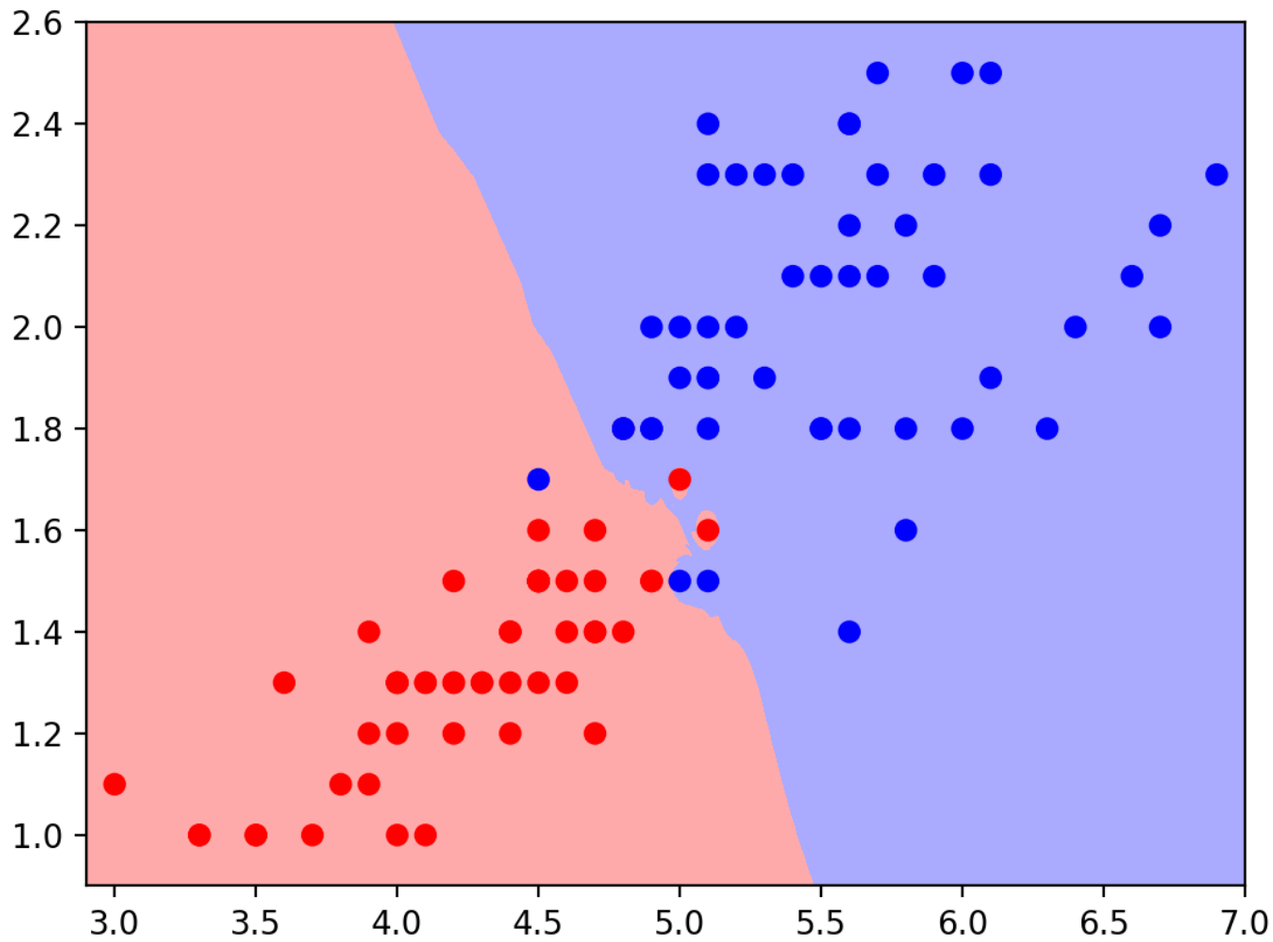
Example 1

- 15-NN



Example 2

- 15-NN



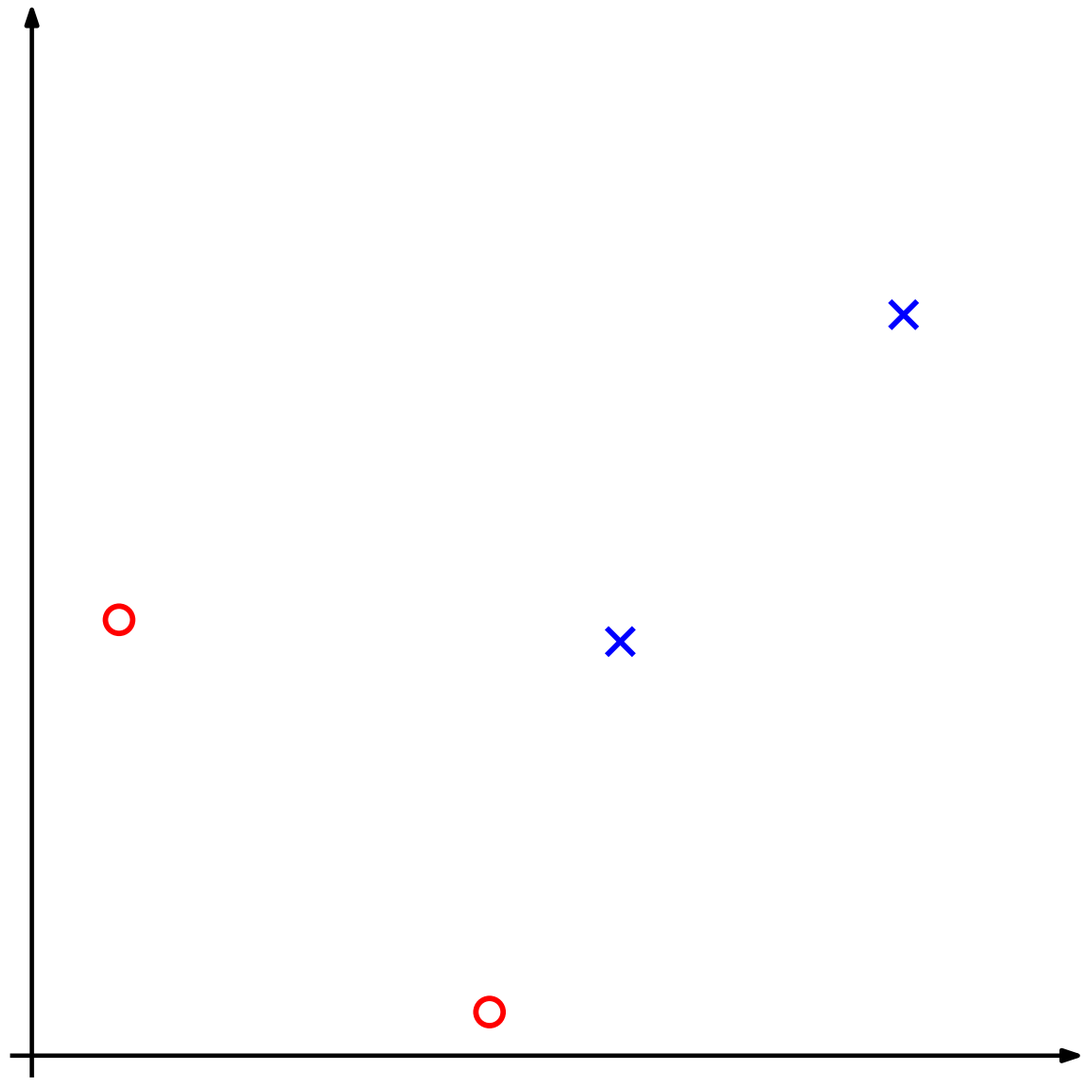
TÉCNICO LISBOA

Ok so far?

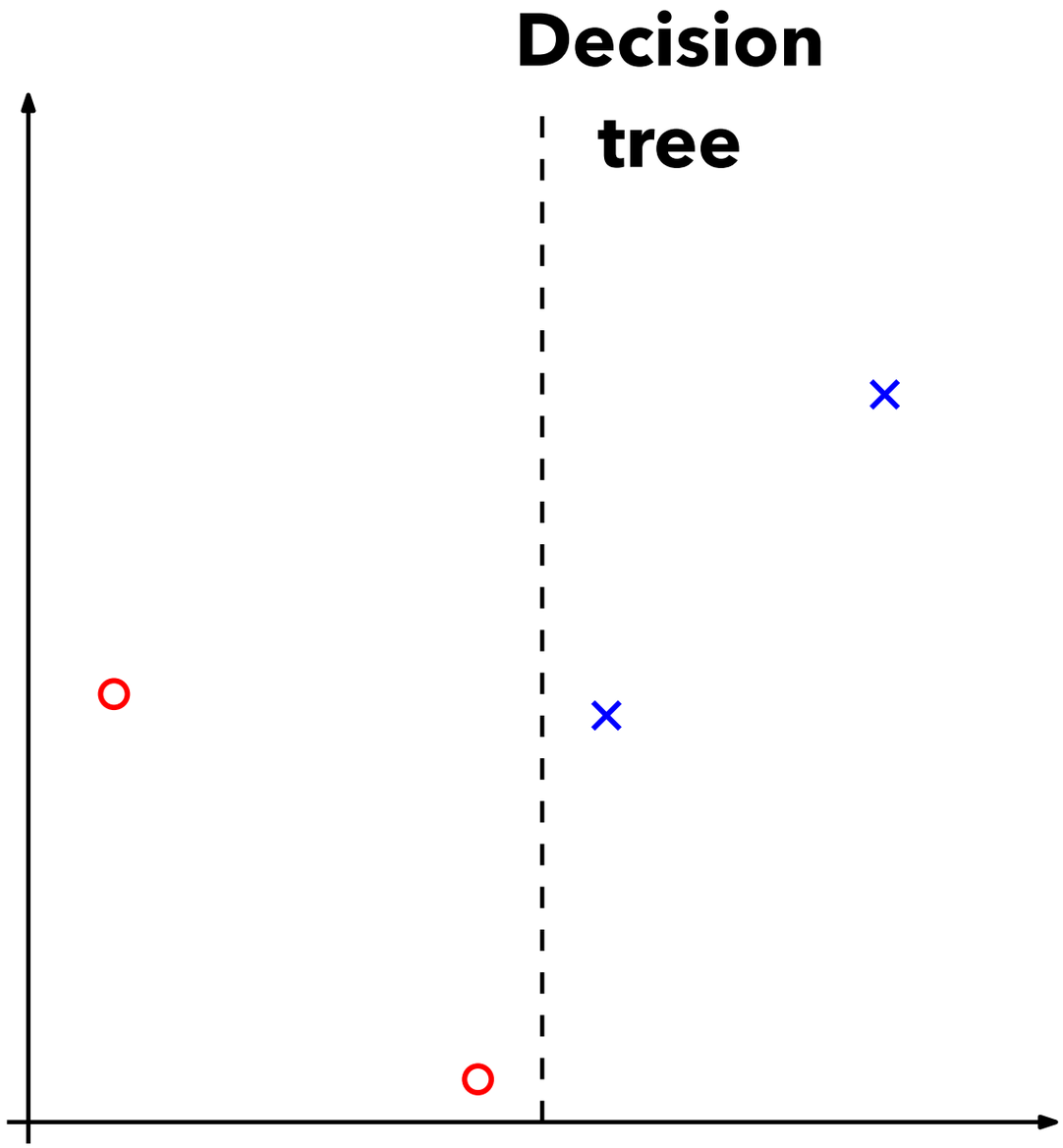

DANGER

Working intuition…

Decision boundary for **Decision tree**?

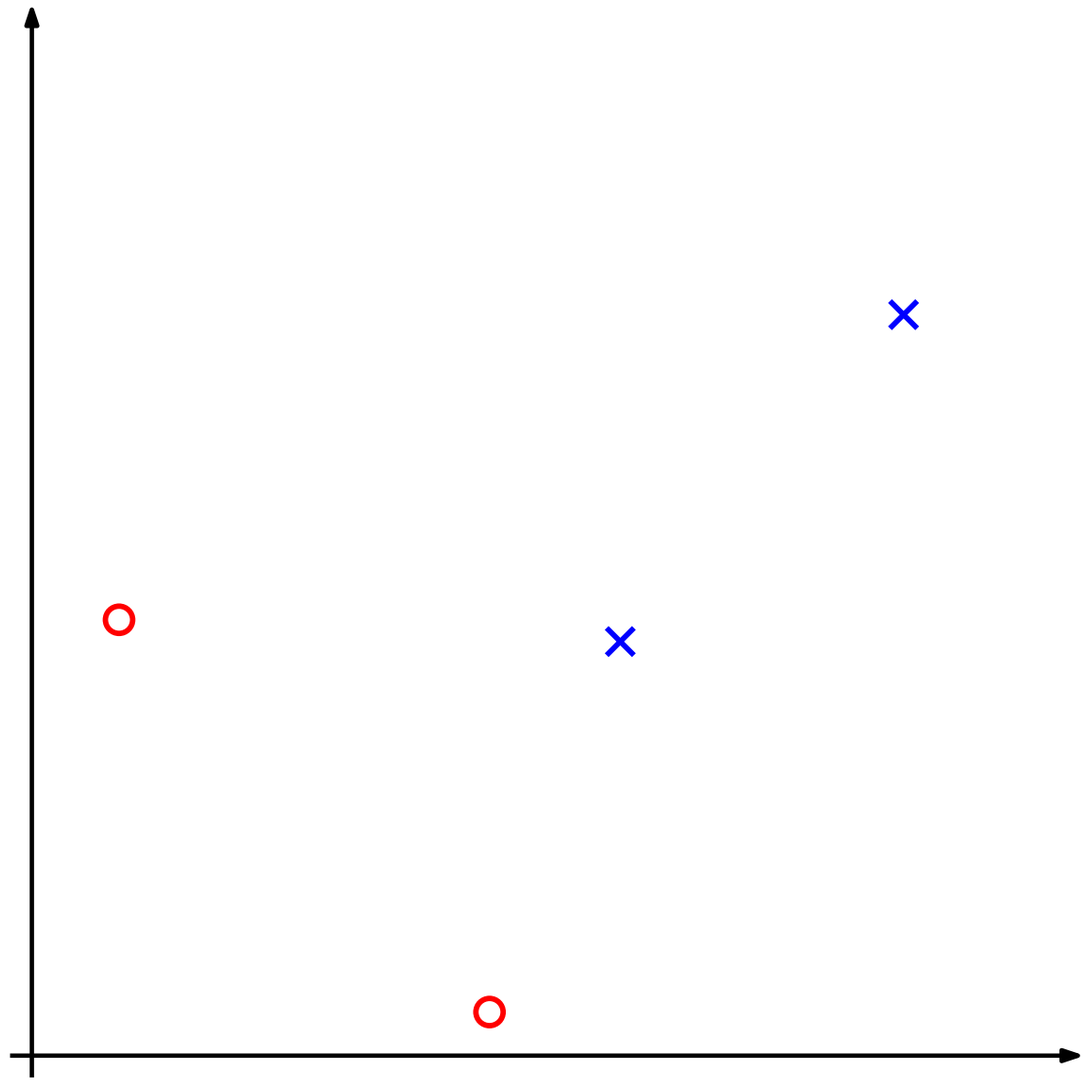


Working intuition…

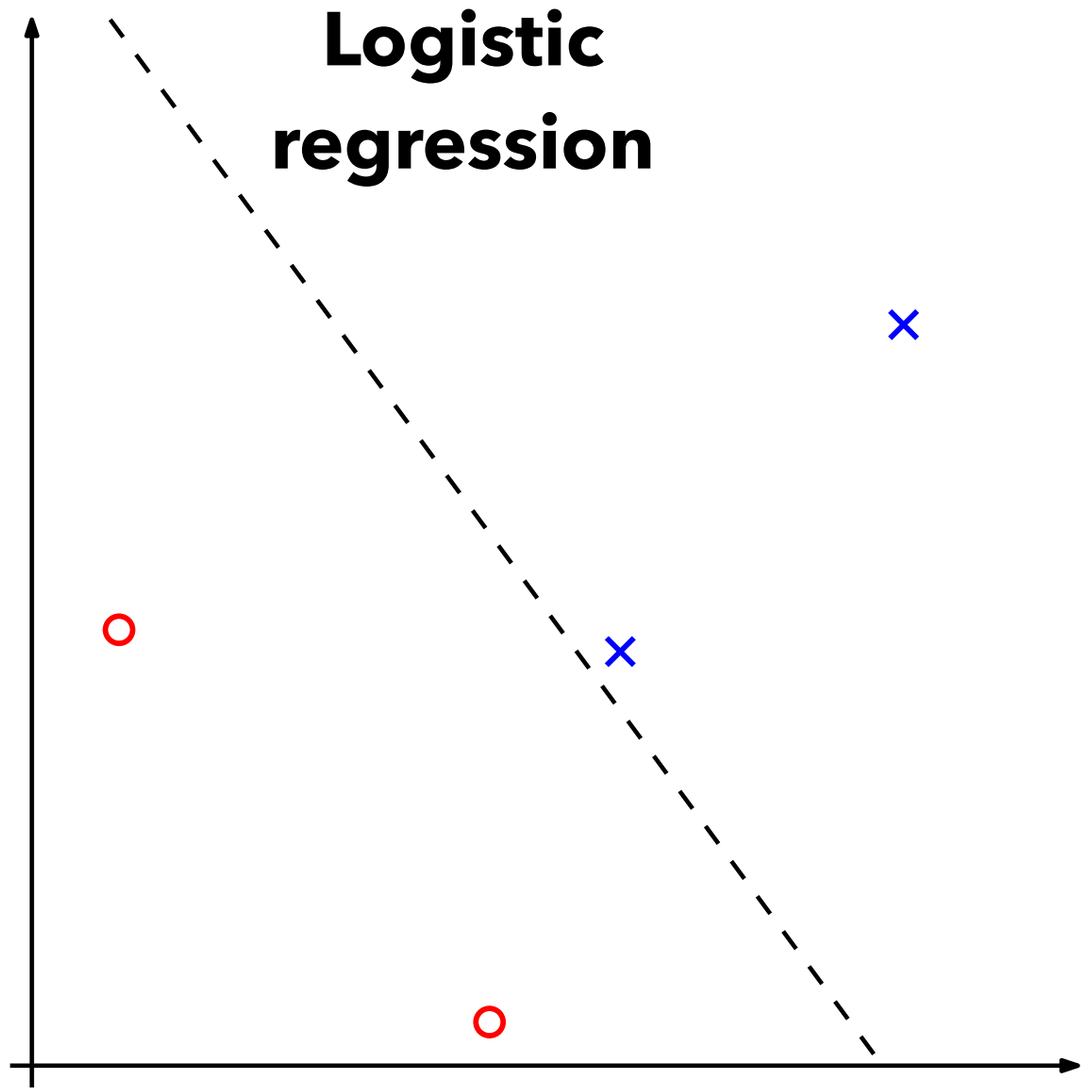


Working intuition…

Decision boundary for **Logistic regression?**

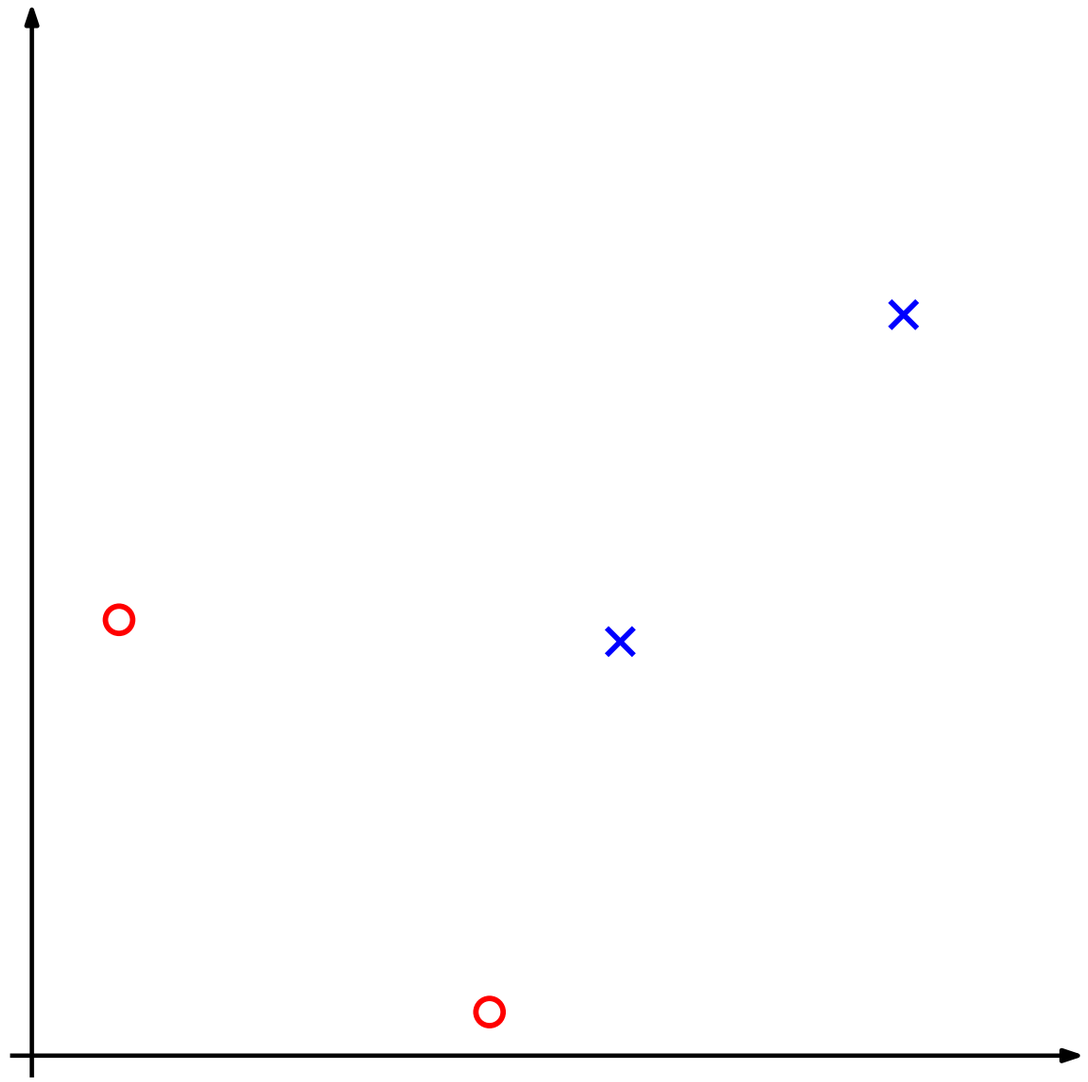


Working intuition…

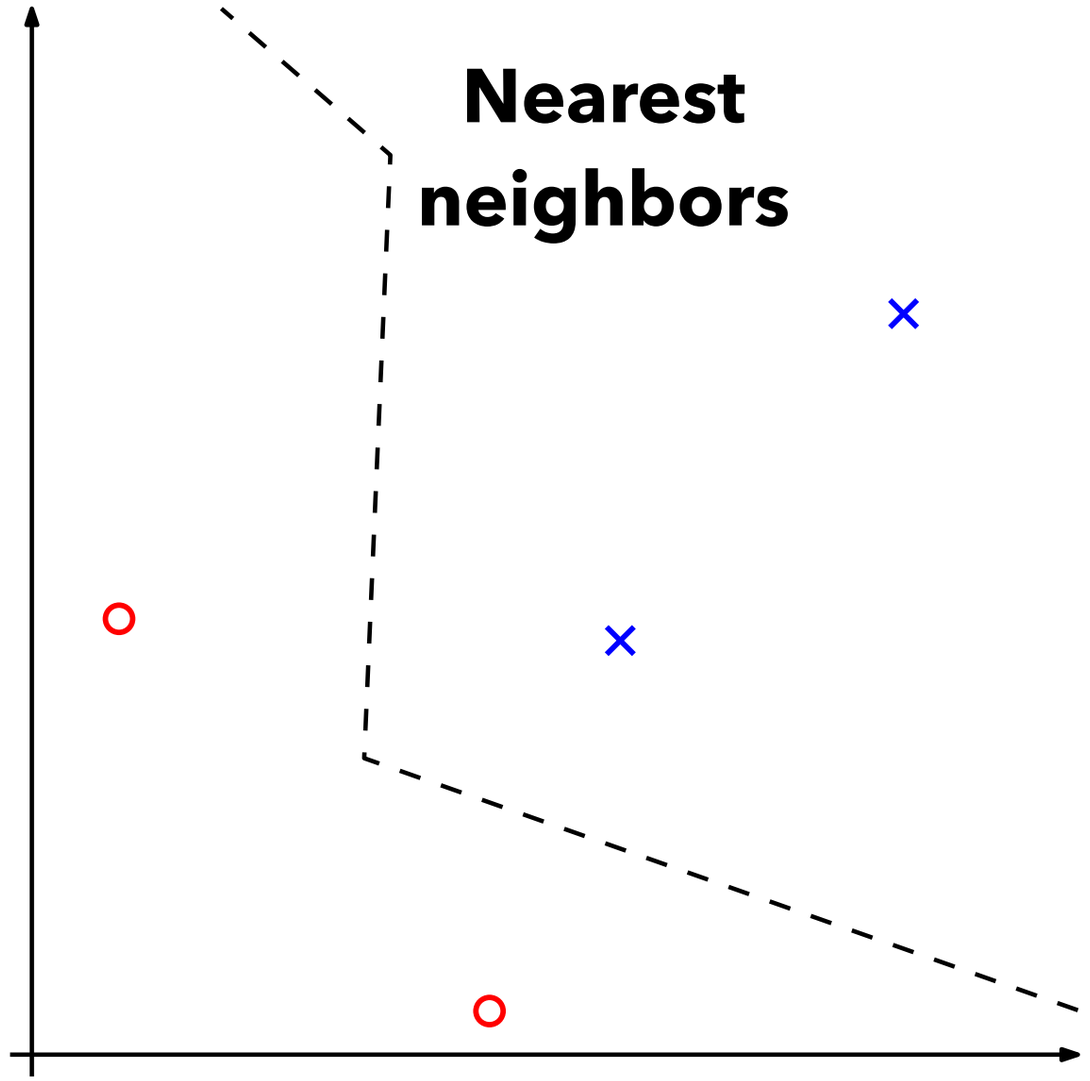


Working intuition…

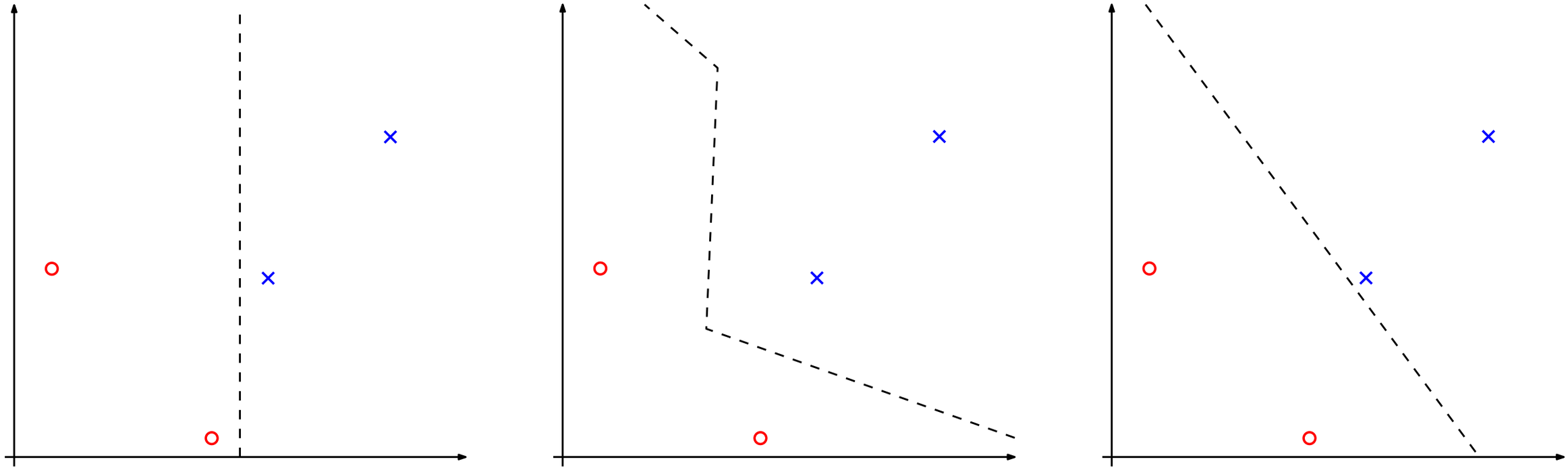
Decision boundary for **Nearest neighbors?**



Working intuition…

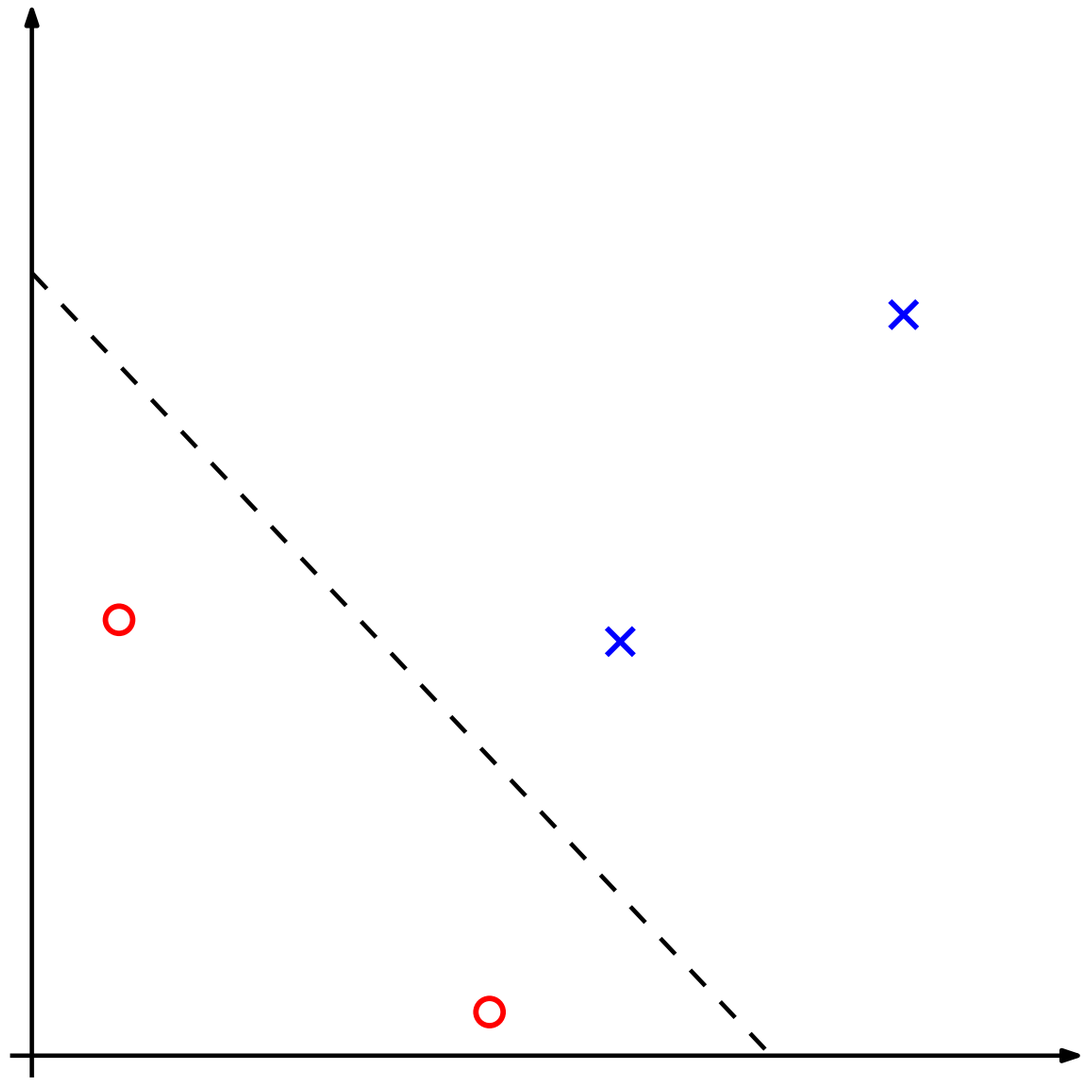


Where would you put it?



Somewhere else?

What about here?

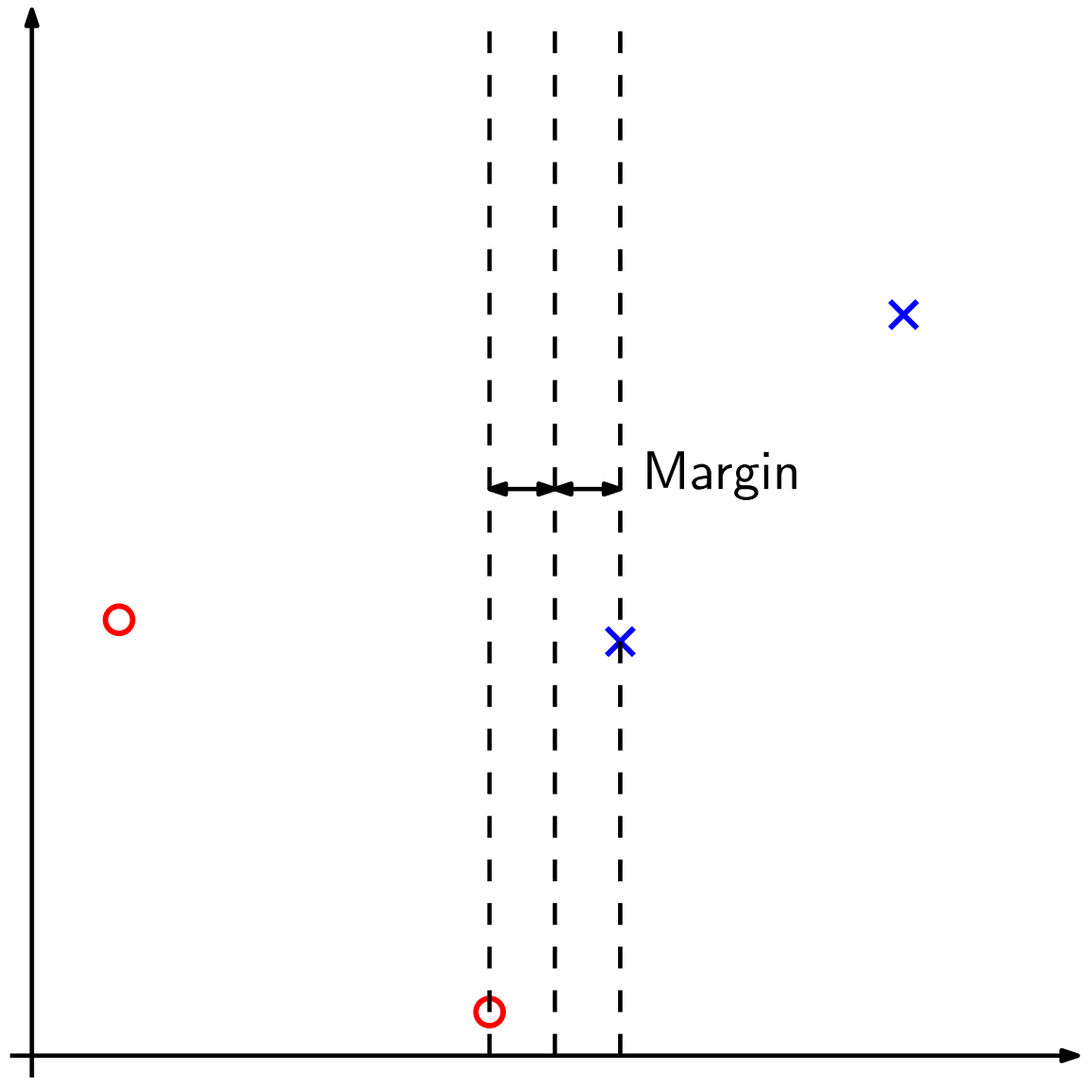


Why?

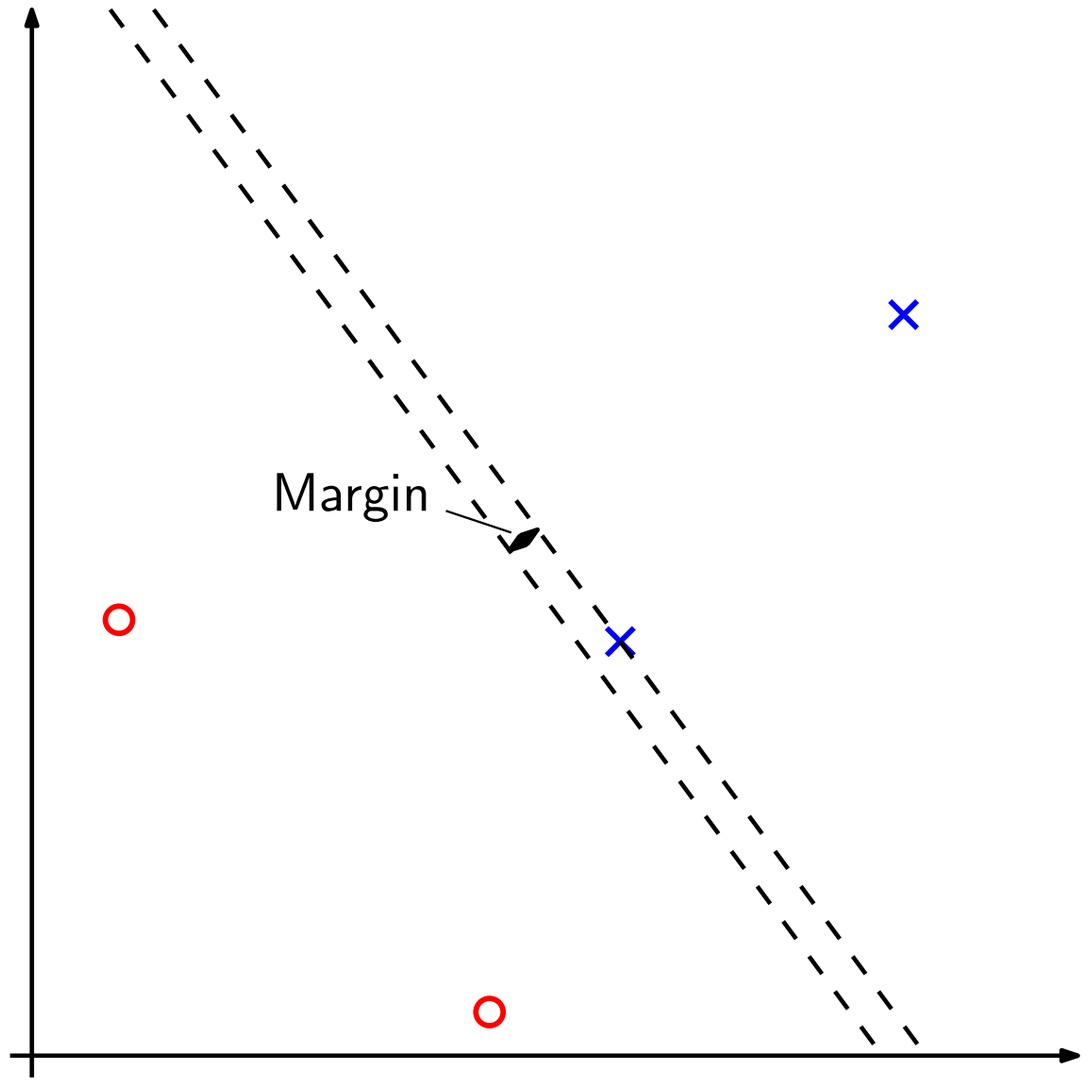
Maximum margin

- It’s the furthest from both classes
  - Maximizes the **margin** between the classes

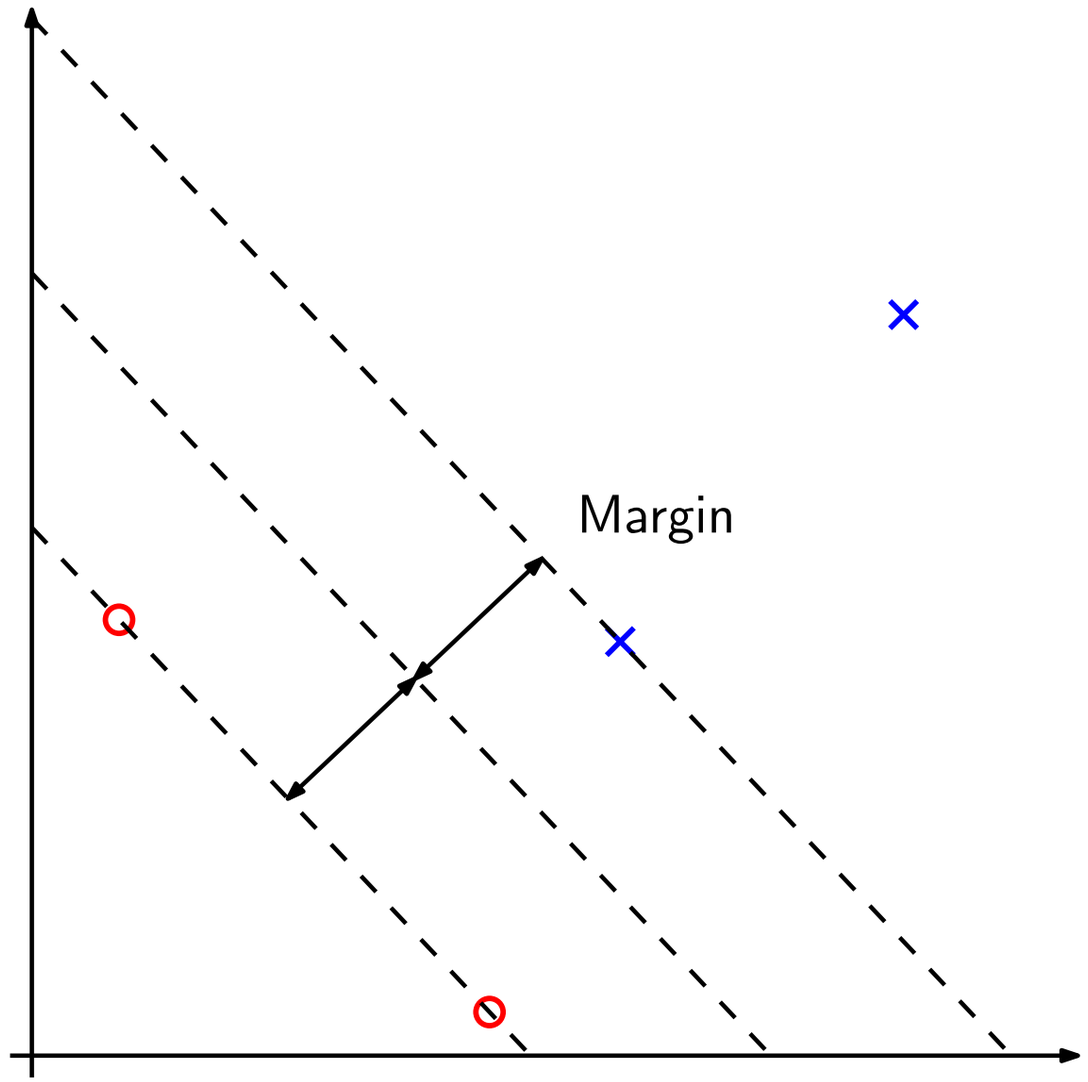
Margin



Margin



Margin



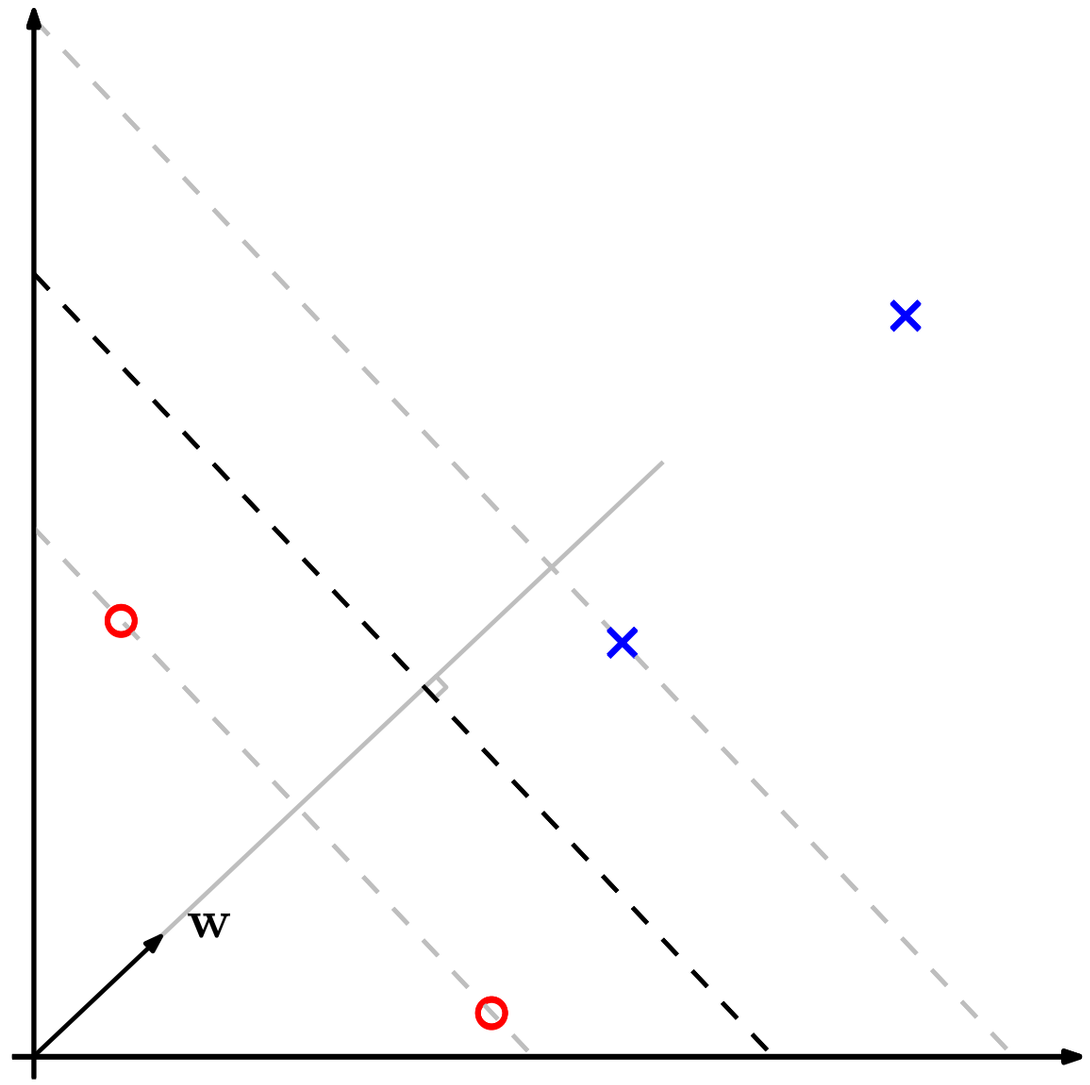
Maximum margin classifier

- How can we compute this classifier?
- We use geometry to compute the **width of the margin**
- **We maximize that width**

- For convenience,
  - Points in the class "×" are denoted with +1
  - Points in the class "o" are denoted with −1

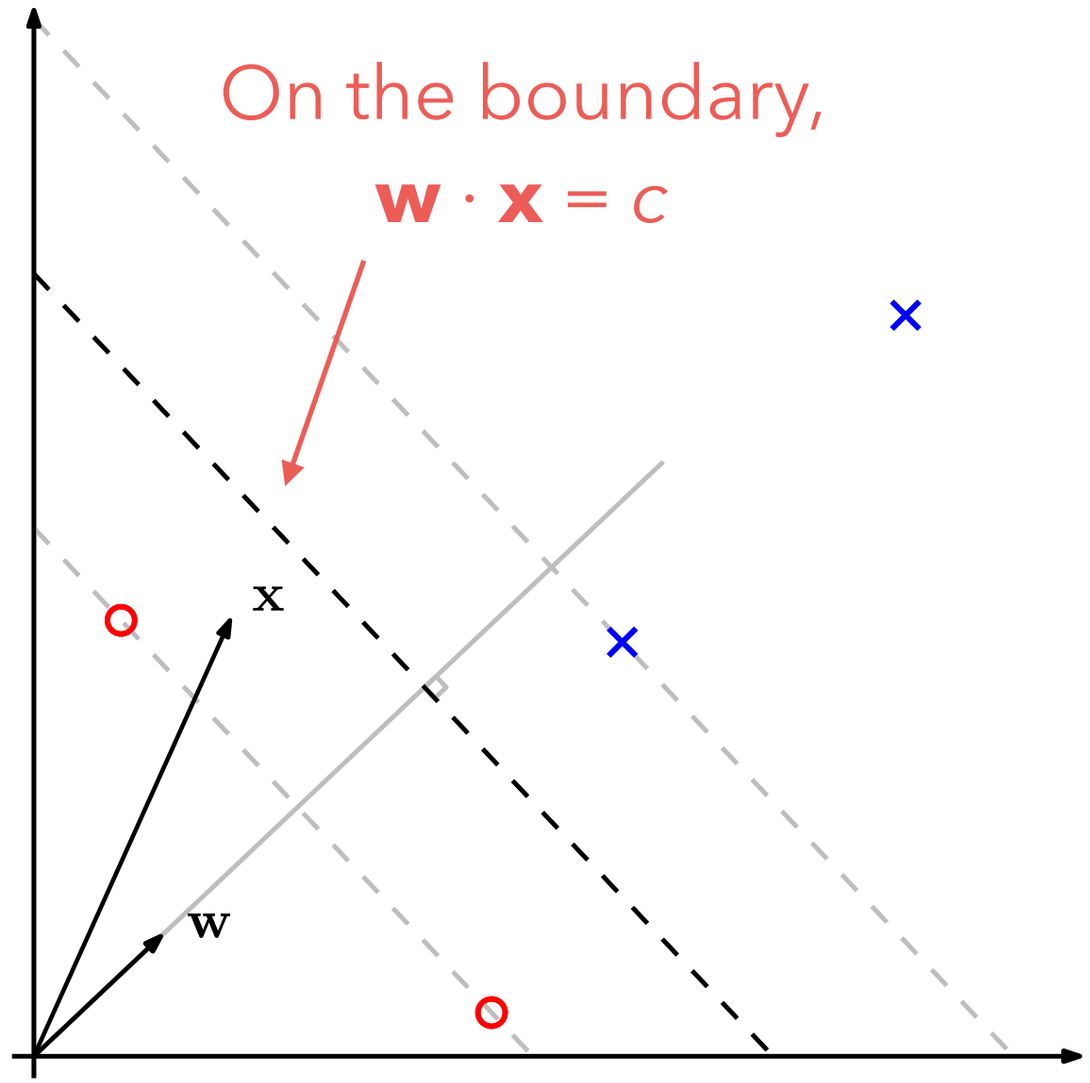
Step 1.

- **w** is a vector orthogonal to the decision boundary



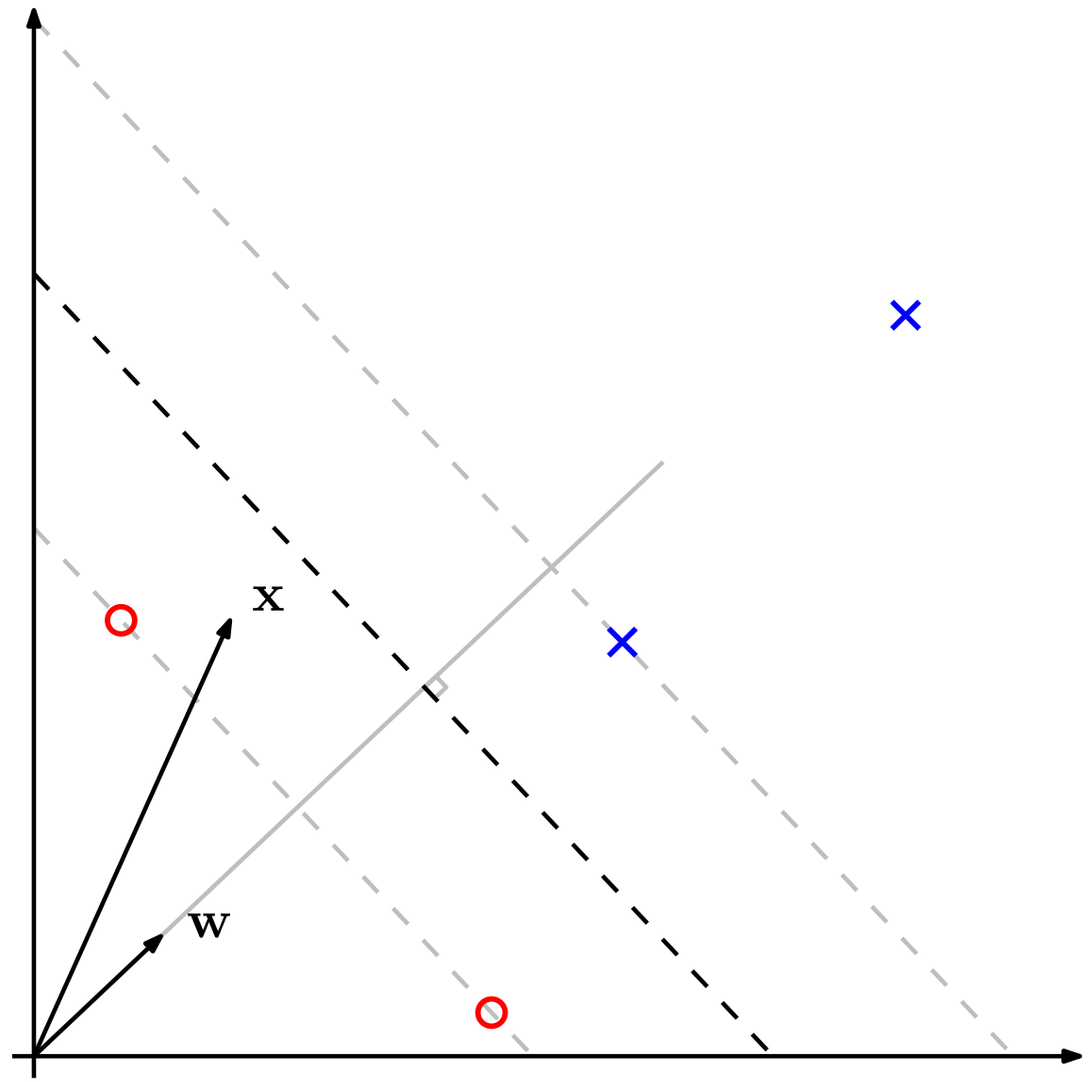
Step 1.

- **w** is a vector orthogonal to the decision boundary
- **x** is an arbitrary point



Step 1.

- **w** is a vector orthogonal to the decision boundary
- **x** is an arbitrary point

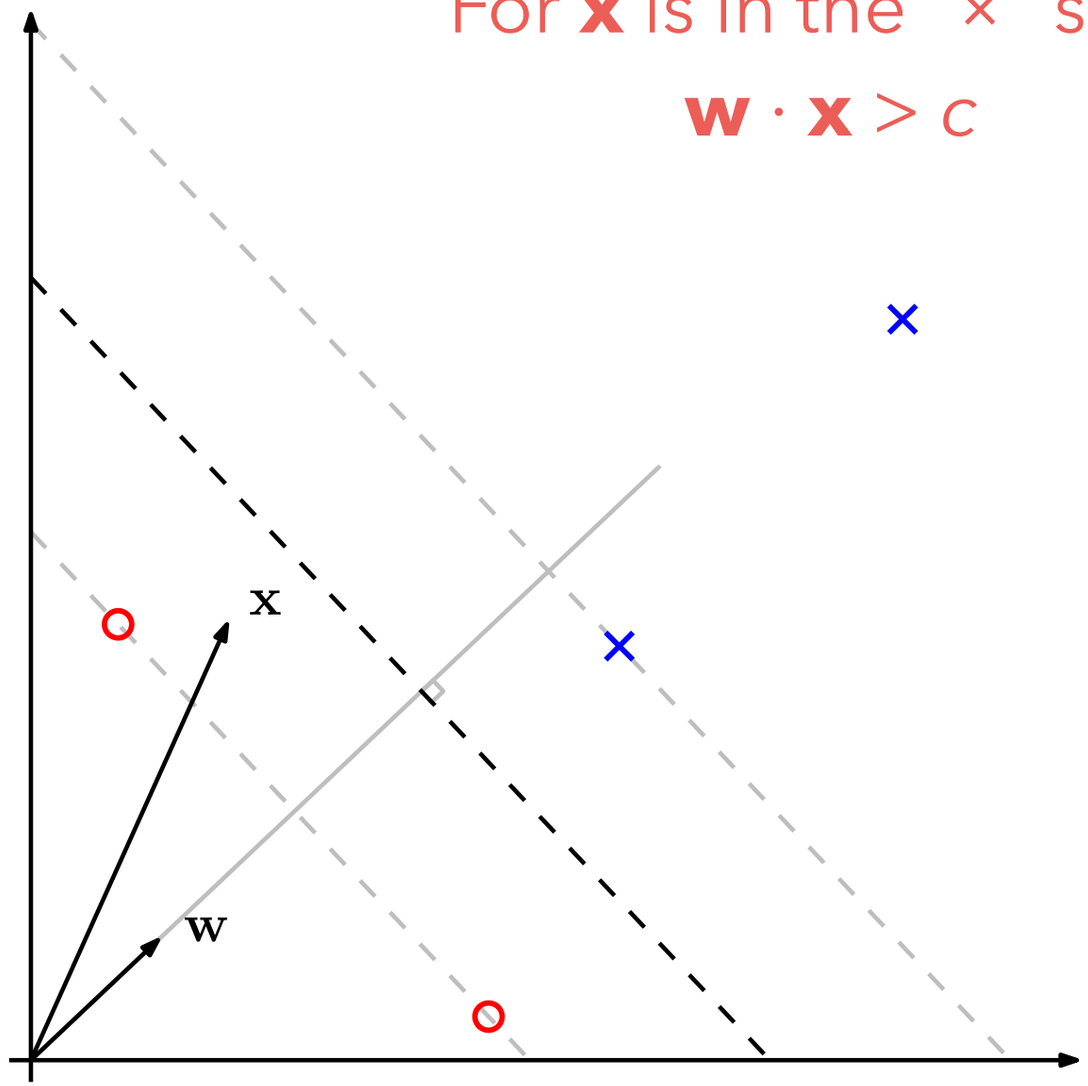


For **x** is in the "o" side,

$\mathbf{w} \cdot \mathbf{x} < c$

Step 1.

- **w** is a vector orthogonal to the decision boundary
- **x** is an arbitrary point



For **x** is in the "×" side,

$\mathbf{w} \cdot \mathbf{x} > c$

Step 1.

- We have that, on the boundary,

$$\boldsymbol{w} \cdot \boldsymbol{x} = c$$

or

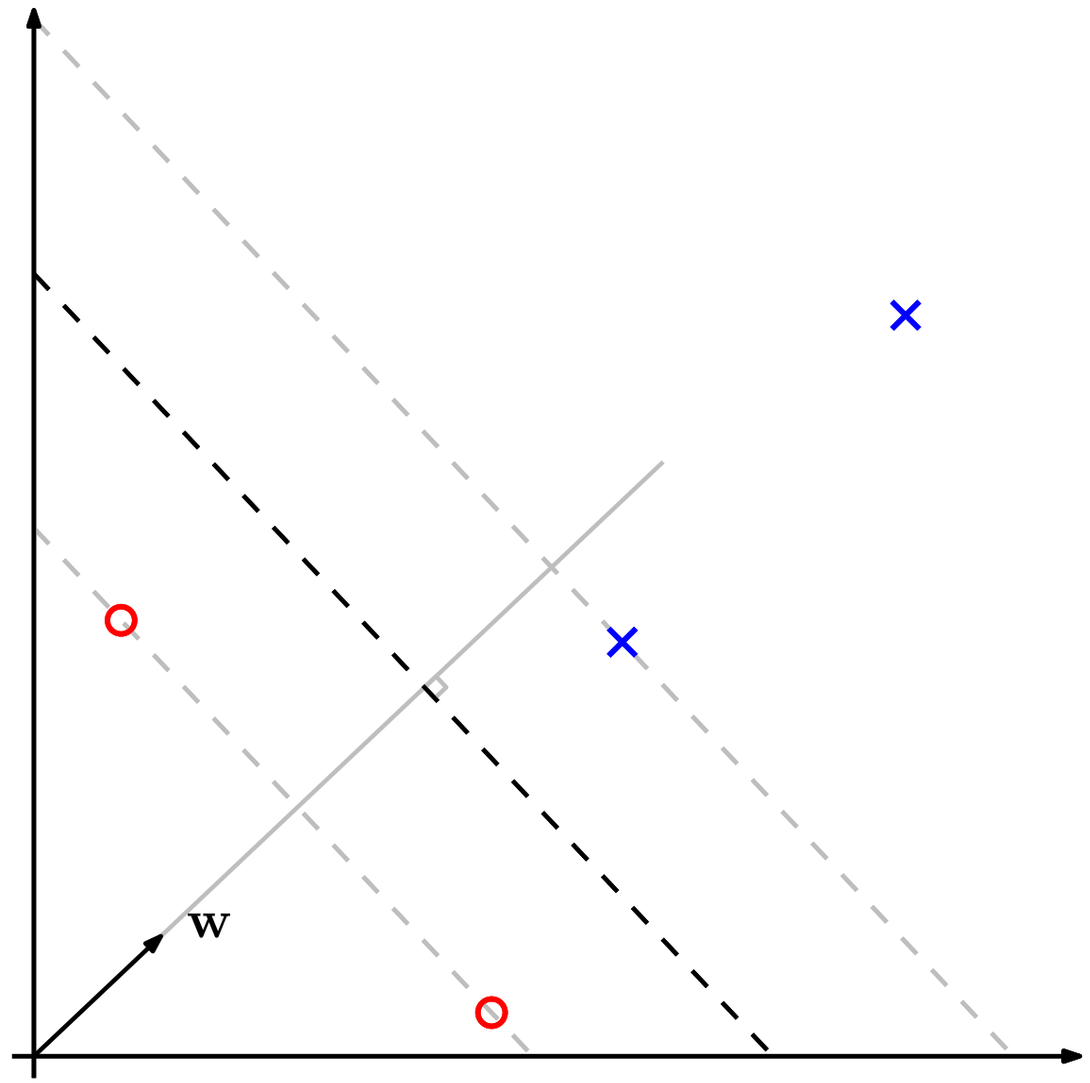
$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- However, we don't know **w** or b…

… nevertheless, let's keep that in mind.

Step 2.

- Let’s look at points in the margin

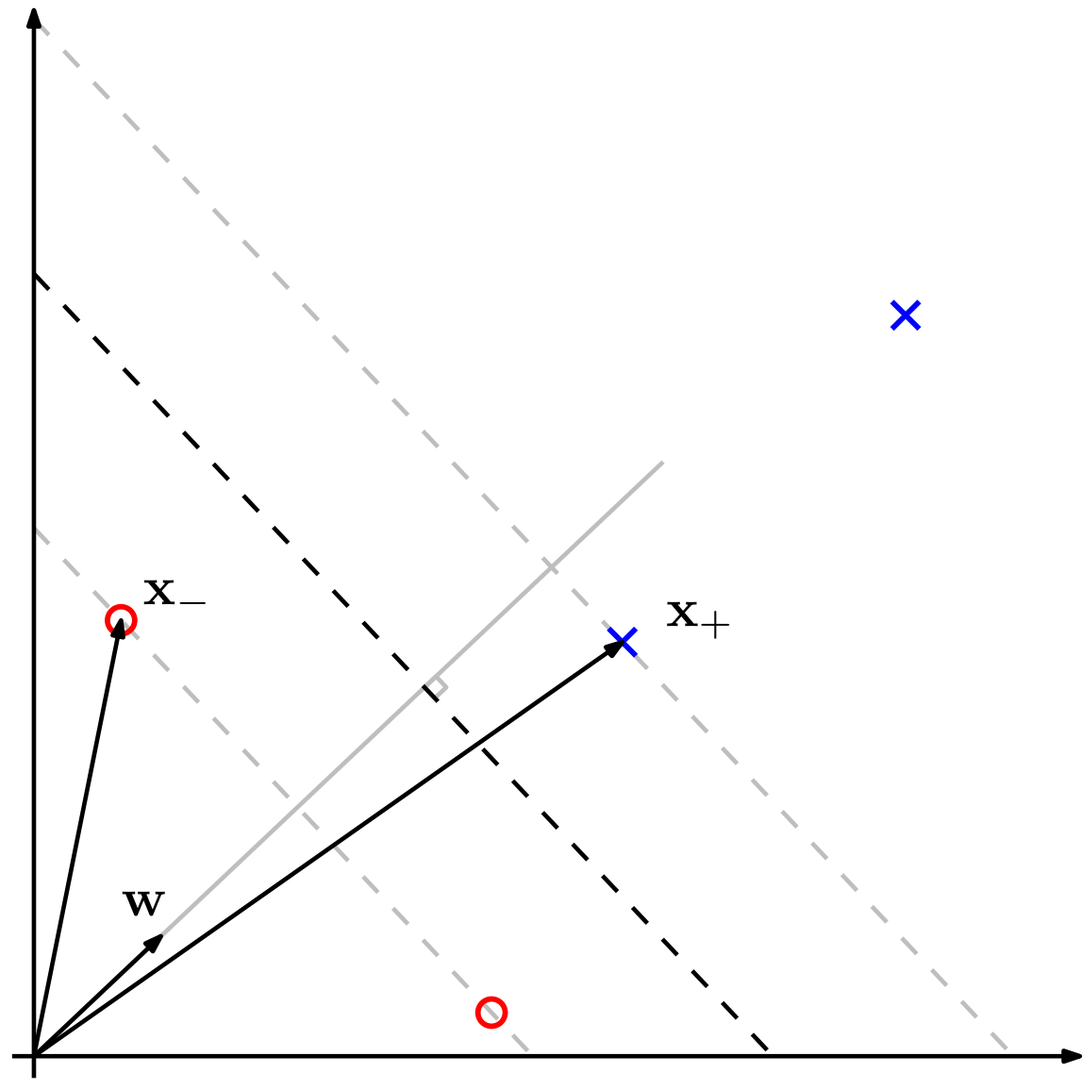


Step 2.

- Let's look at points in the margin

$$\boldsymbol{w} \cdot \boldsymbol{x}_- + b < 0$$

$$\boldsymbol{w} \cdot \boldsymbol{x}_+ + b > 0$$



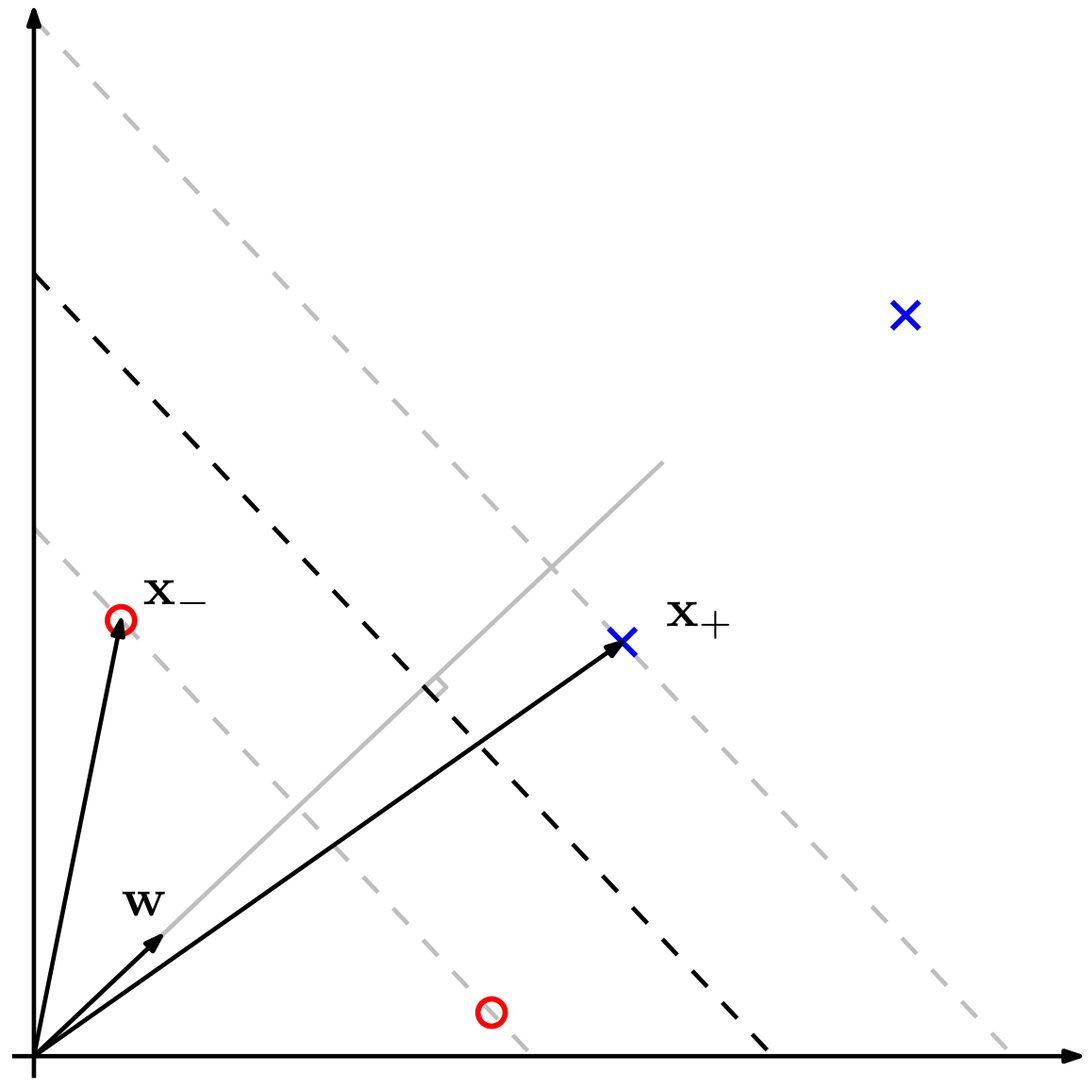
Step 2.

- Let's look at those points in the margin

$$\boldsymbol{w} \cdot \boldsymbol{x}_- + b = \boxed{-1}$$

$$\boldsymbol{w} \cdot \boldsymbol{x}_+ + b = \boxed{1}$$

We convention these values



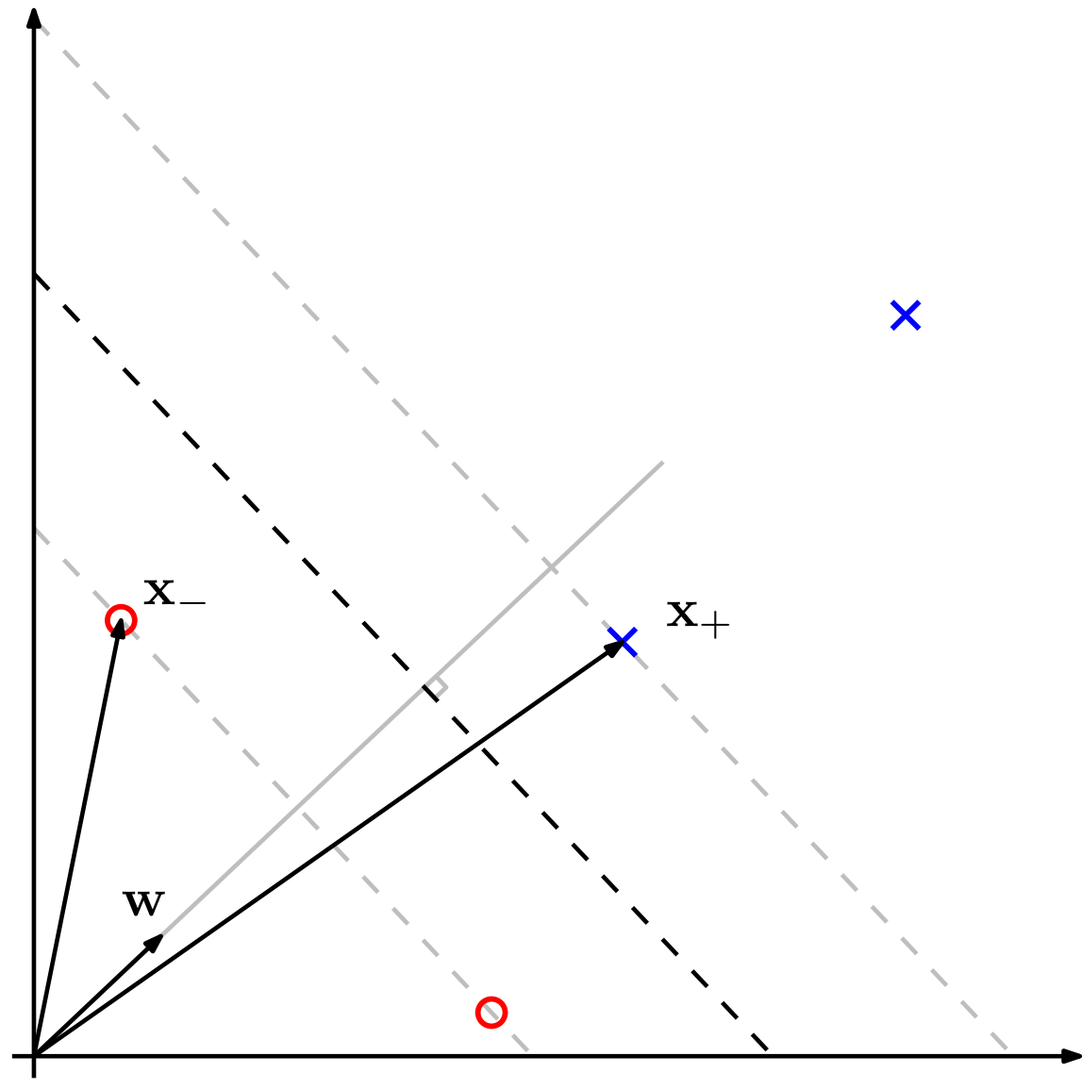
Step 2.

- Let's look at those points in the margin

$$\boldsymbol{w} \cdot \boldsymbol{x}_- + b = -1$$

$$\boldsymbol{w} \cdot \boldsymbol{x}_+ + b = 1$$

What if we multiply by target value (+1)?



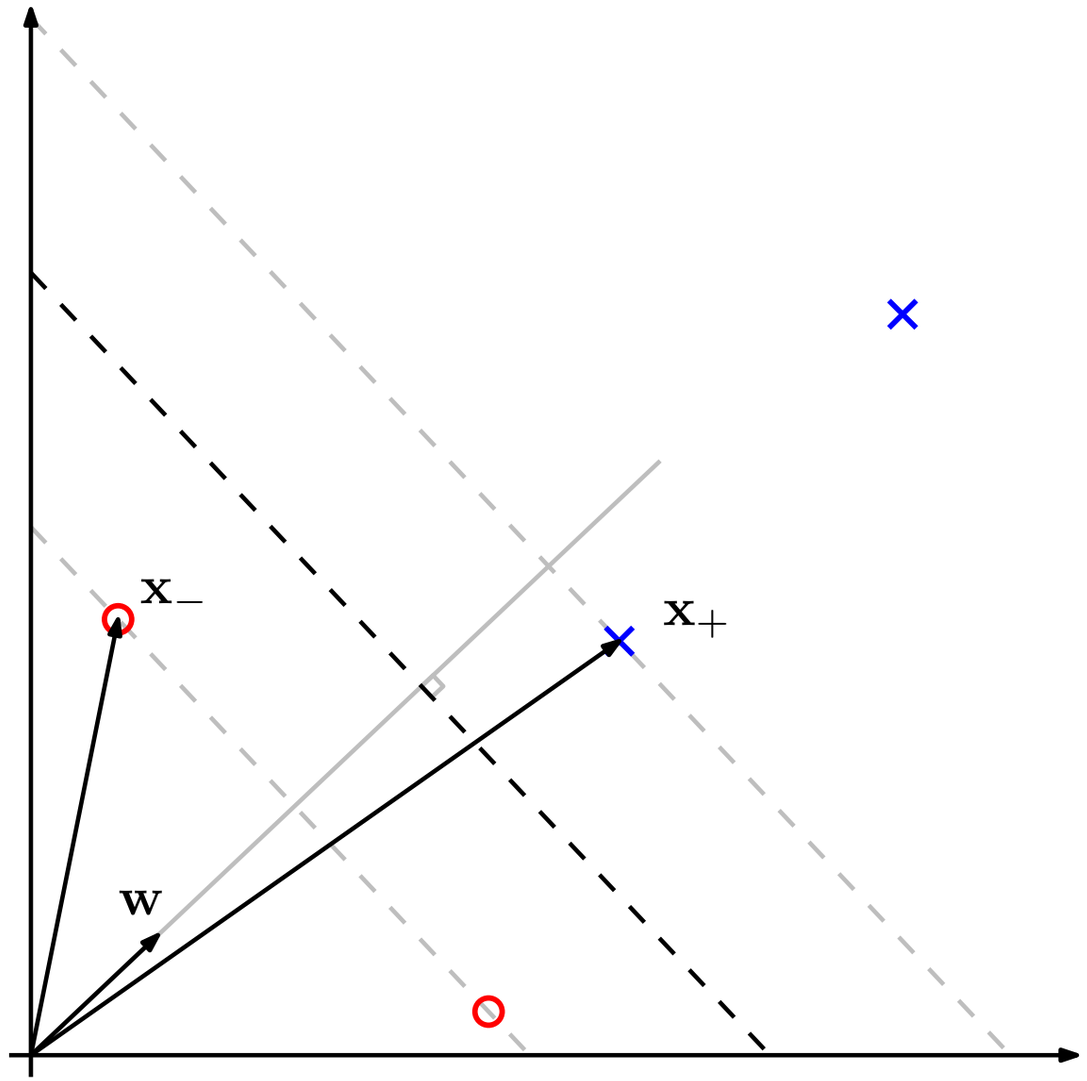
Step 2.

- Let's look at those points in the margin

$$\boldsymbol{w} \cdot \boldsymbol{x}_- + b = -1$$

$$y_+(\boldsymbol{w} \cdot \boldsymbol{x}_+ + b) = 1$$

What if we multiply by target value (−1)?



Step 2.

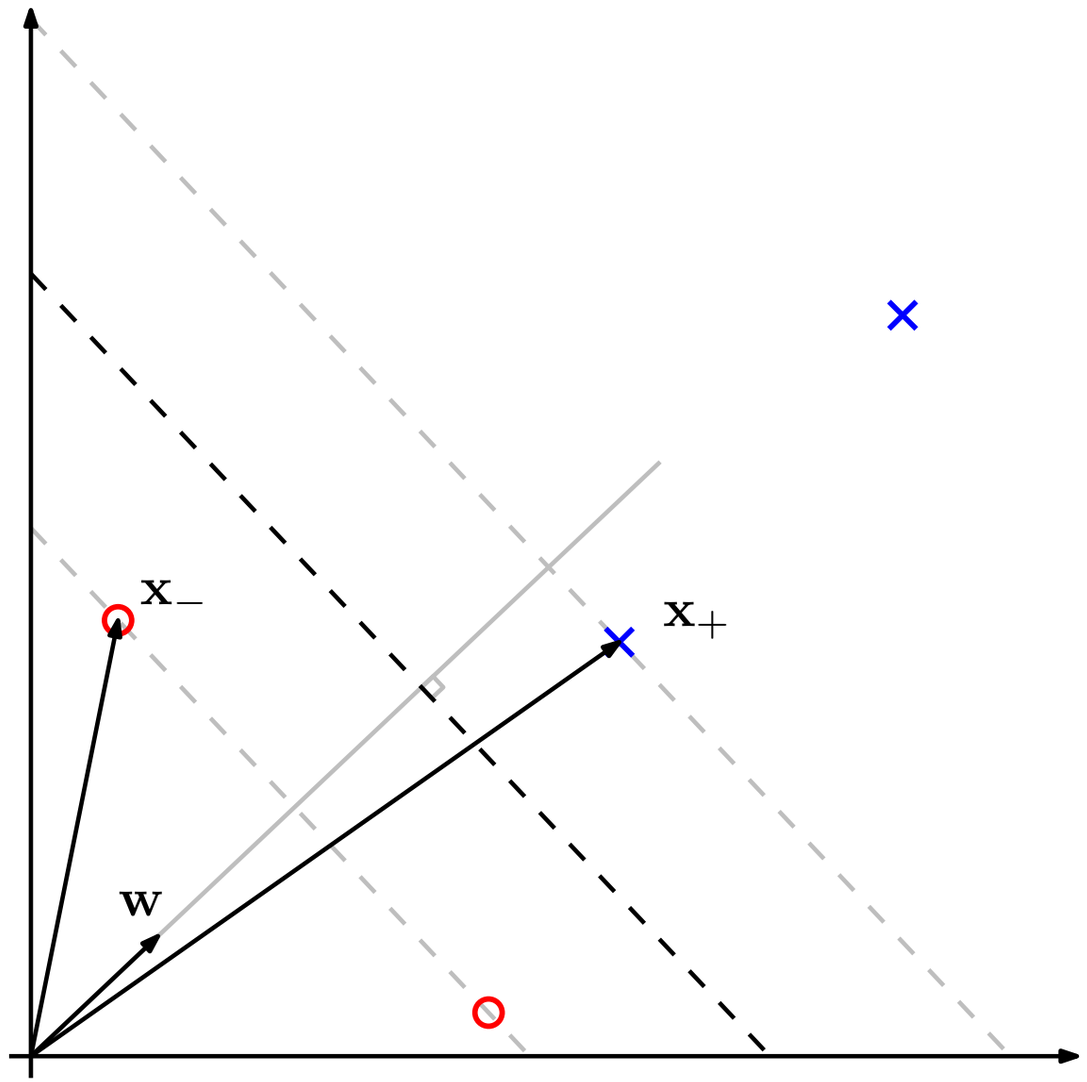
- Let's look at those points in the margin

$$y_-(\boldsymbol{w} \cdot \boldsymbol{x}_- + b) = 1$$

$$y_+(\boldsymbol{w} \cdot \boldsymbol{x}_+ + b) = 1$$

Same equation

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1$$



So far...

- Step 1. Points in the decision boundary:

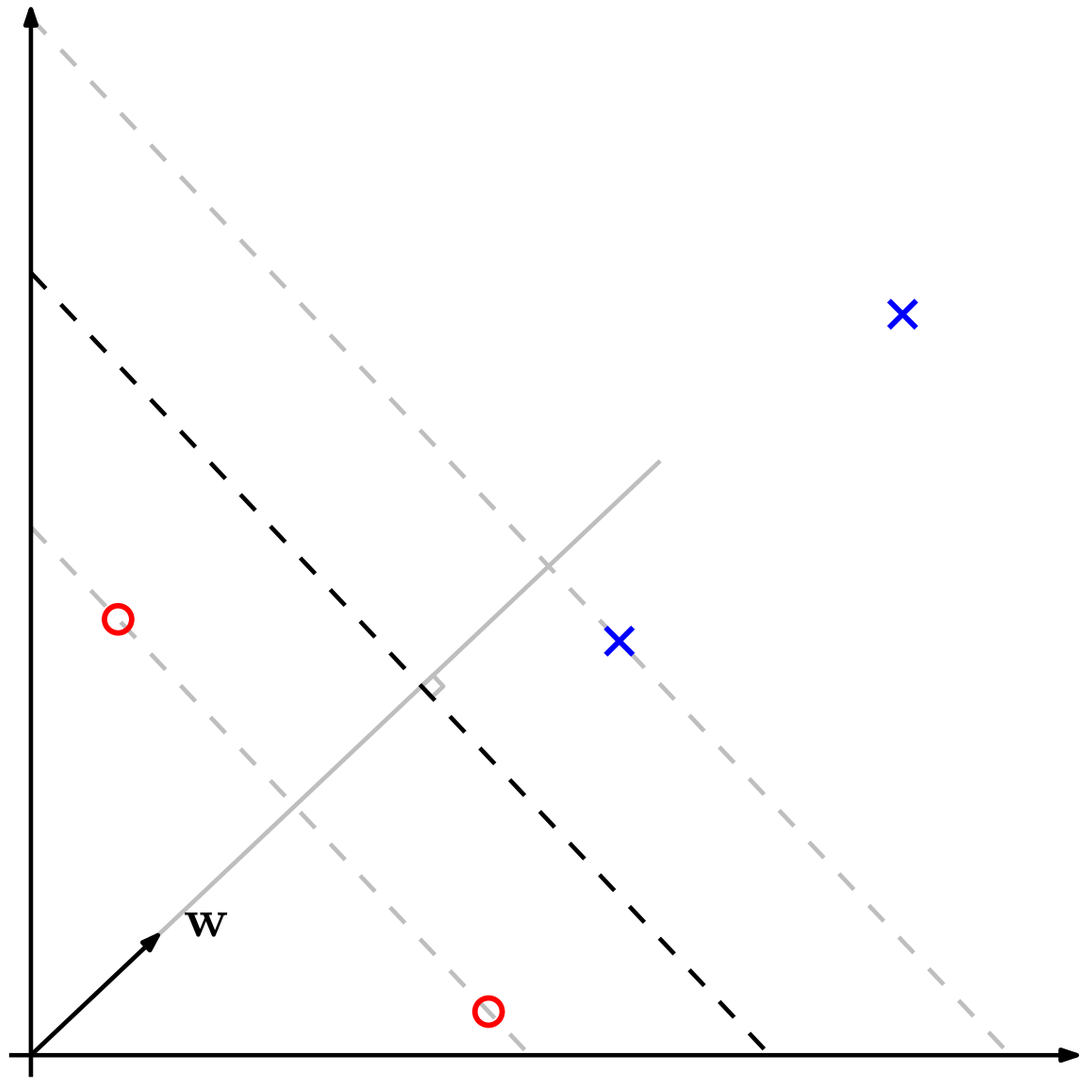
$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1$$

Step 3.

- What about this vector?



Step 3.

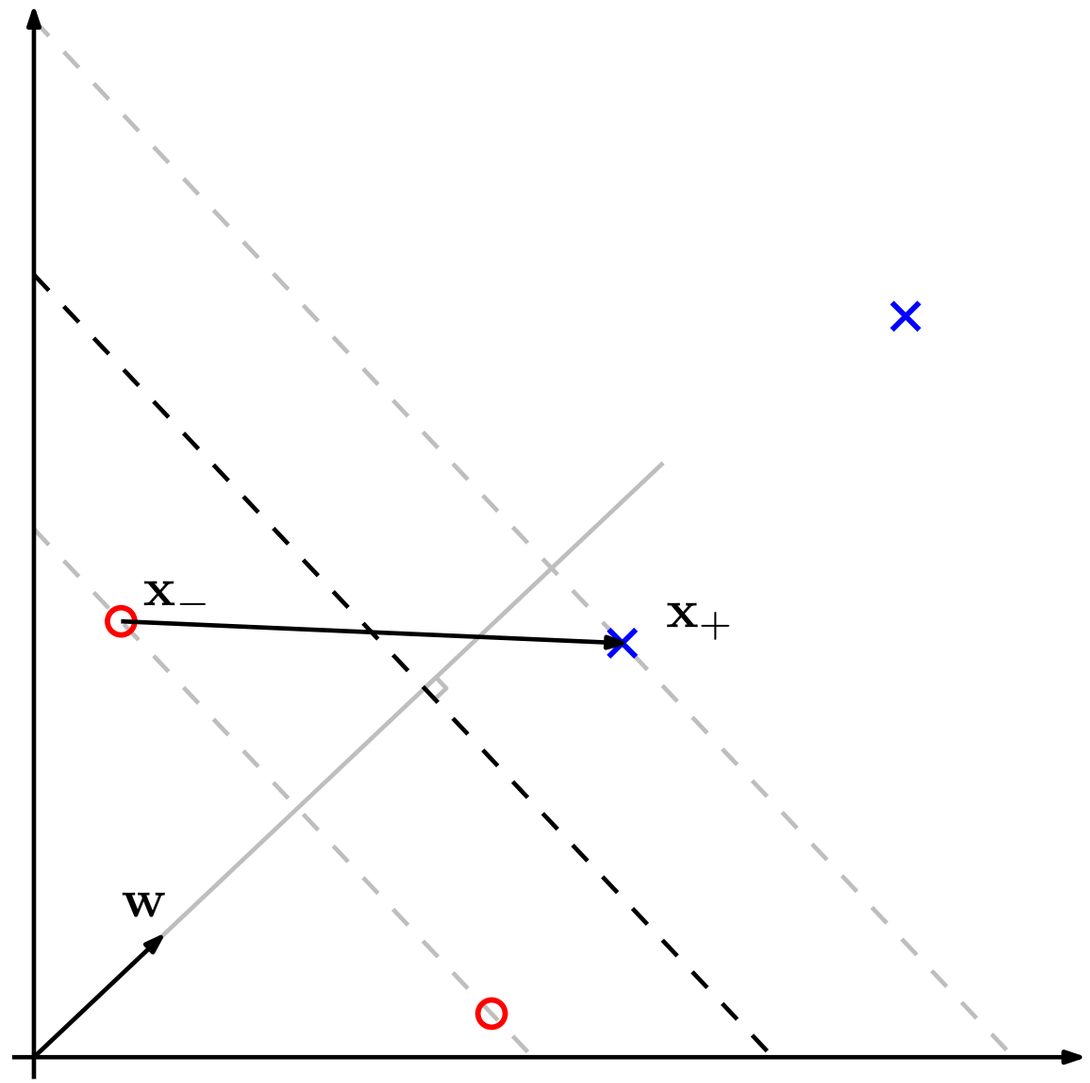
- What about this vector?

$$\boldsymbol{x}_+ - \boldsymbol{x}_-$$

- What is the distance between the gray lines?

$$(\boldsymbol{x}_+ - \boldsymbol{x}_-) \cdot \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$$

We have our "margin"



Finally…

- Step 1. Points in the decision boundary:

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1$$

- Step 3. Width of the (double) margin:

$$(\boldsymbol{x}_+ - \boldsymbol{x}_-) \cdot \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$$

What can we do with this??

Finally…

- Step 1. Points in the decision boundary:

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1 \longrightarrow y_-(\boldsymbol{w} \cdot \boldsymbol{x}_- + b) = 1$$

- Step 3. Width of the (double) margin:

$$(\boldsymbol{x}_+ - \boldsymbol{x}_-) \cdot \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$$

Finally…

- Step 1. Points in the decision boundary:

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1 \longrightarrow y_+(\boldsymbol{w} \cdot \boldsymbol{x}_+ + b) = 1$$

- Step 3. Width of the (double) margin:

$$\frac{\boldsymbol{x}_+ \leftrightarrow \boldsymbol{w} - (-b - 1)}{\|\boldsymbol{w}\|}$$

Finally…

- Step 1. Points in the decision boundary:

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1$$

- Step 3. Width of the (double) margin:

$$\frac{1 - b + b + 1)}{\|\boldsymbol{w}\|}$$

Finally…

- Step 1. Points in the decision boundary:

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

- Step 2. Points in the margin:

$$y(\boldsymbol{w} \cdot \boldsymbol{x} + b) = 1$$

- Step 3. Width of the (double) margin:

$$\frac{2}{\|\boldsymbol{w}\|}$$

And we're done!

- We want to solve the following optimization problem:

$$\begin{aligned} \max \quad & \frac{1}{\|\boldsymbol{w}\|} \\ \text{s.t.} \quad & y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) \geq 1 \end{aligned}$$

And we're done!

- We want to solve the following optimization problem:

$$\begin{aligned} \min \quad & \|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) \geq 1 \end{aligned}$$


Huh?

Lagrangian

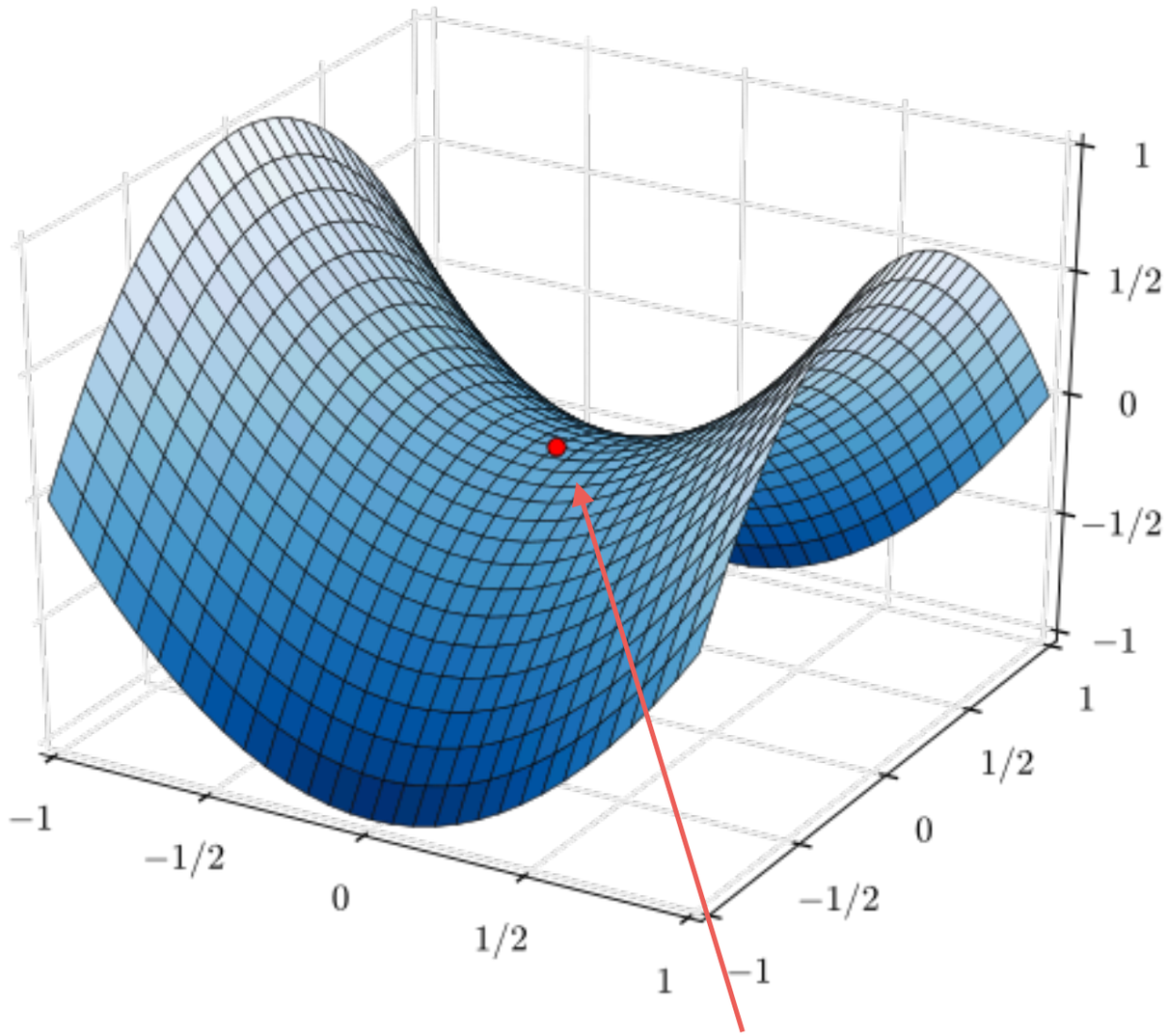
- The optimization problem:

$$\begin{aligned} \min \quad & \|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) \geq 1 \end{aligned}$$

- Can be turned into:

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

Optimization detour



How can we know this is a minimum?

Optimization detour

- We need additional conditions:
  - E.g., information about second derivative
- For the Lagrangian, we need more complicated conditions:
  - Karush-Kuhn Tucker (KKT) conditions

Lagrangian

- Constrained optimization problem:

$$\begin{aligned} \min \quad & \|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) \geq 1 \end{aligned}$$

- Lagrangian: $G_n(\boldsymbol{w}) = 1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b)$

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

KKT conditions

- If (**w***, **α***) are solutions to the primal and dual, then
  - $\nabla_{\mathrm{w}} L(\mathbf{w}^*, \boldsymbol{\alpha}^*) = 0$ **w minimizes the Lagrangian**
  - $G_n(\mathbf{w}^*) \leq 0, \text{for } n = 1, \ldots, N$
  - $\alpha_n^* \geq 0, \text{for } n = 1, \ldots, N$

Both solutions verify constraints

  - $\alpha_n^* G_n(\boldsymbol{w}^*) = 0, \text{for } n = 1, \ldots, N$

Verified constraints don't add to the primal objective

What comes out of this?

- The Lagrangian is:

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

- Condition 1: **w** minimizes the Lagrangian

$$\nabla_{\mathrm{w}} L(\mathbf{w}^*, \boldsymbol{\alpha}^*) = 0$$

What comes out of this?

- The Lagrangian is:

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

- Condition 1: **x** minimizes the Lagrangian

$$\boldsymbol{w} = \sum_{n=1}^{N} \alpha_n \boldsymbol{x}_n y_n$$

Some constants

What comes out of this?

- Replacing **w**, we get

$$y = \text{sgn}(\boldsymbol{w} \cdot \boldsymbol{x} + b)$$

What comes out of this?

- Replacing **w**, we get

$$y = \text{sgn}\left(\sum_{n=1}^{N} \alpha_n y_n (\boldsymbol{x}_n \cdot \boldsymbol{x}) + b\right)$$

- The output is a combination of the observed outputs y_n weighted by how similar **x** is to each $\mathbf{x}_n$
- Only depends on the **dot product** (similarity) between **x** and the $\mathbf{x}_n$

$$y = \text{sgn}\left(\sum_{n=1}^{N} \alpha_n y_n k(\boldsymbol{x}_n, \boldsymbol{x}) + b\right)$$

What comes out of this?

- The Lagrangian is:

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

- Condition 3: Active constraints don't affect objective

$$\alpha_n^* G_n(\boldsymbol{w}^*) = 0, \text{for } n = 1, \ldots, N$$

Finally…

- The Lagrangian is:

$$L(\boldsymbol{w}, \boldsymbol{\alpha}) = \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b))$$

- Condition 3: Active constraints don't affect objective

$$\alpha_n (1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b)) = 0,$$

Finally…

- Let us consider the last condition:

$$\alpha_n(1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b)) = 0,$$

- Only the points $\mathbf{x}_n$ that fall in the margin have positive α_n
- All other have null weights

Summarizing…

In an SVM, the output is a **combination** of the labels of the **support vectors** (i.e., the data points that fall in the margin)

SVMs in one slide