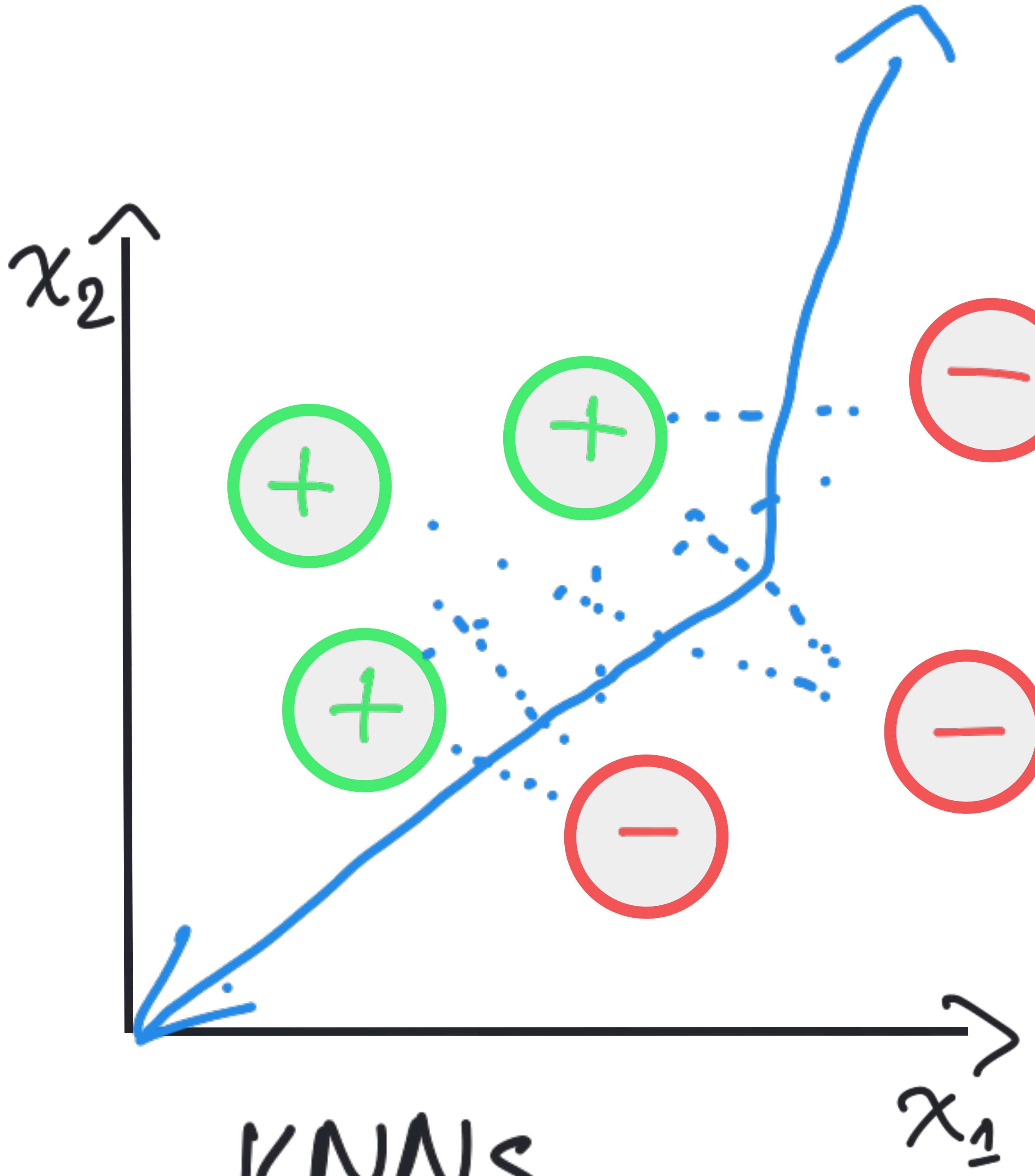
6.034 FALL 2020

RECITATION 9: SVMs

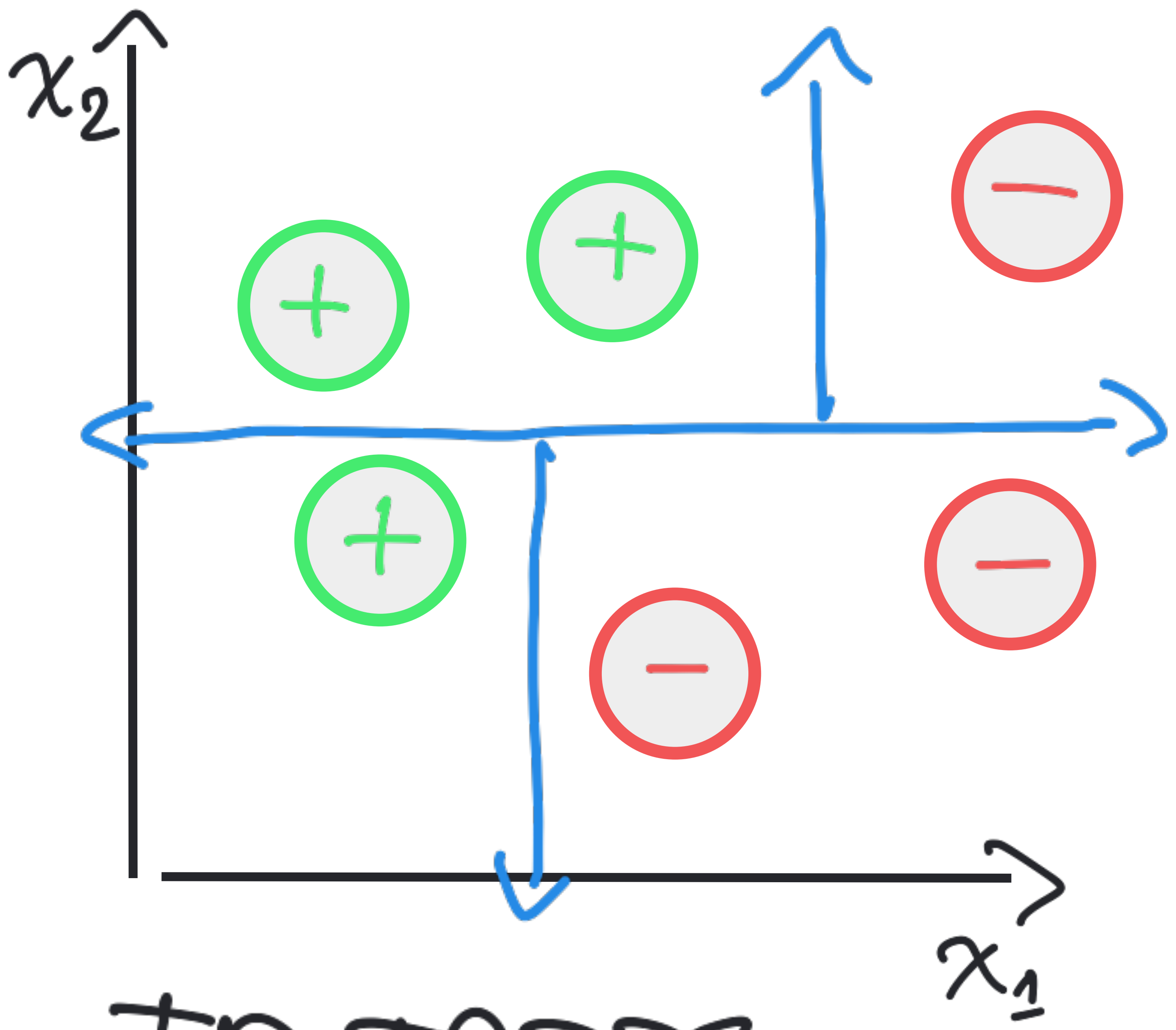
- [ ] DRAW SVM BOUNDARY & GUTTER
- [ ] CALCULATING $\vec{w}$, b & α VALUES
- [ ] KERNEL TRICK

SUPERVISED MACHINE LEARNING

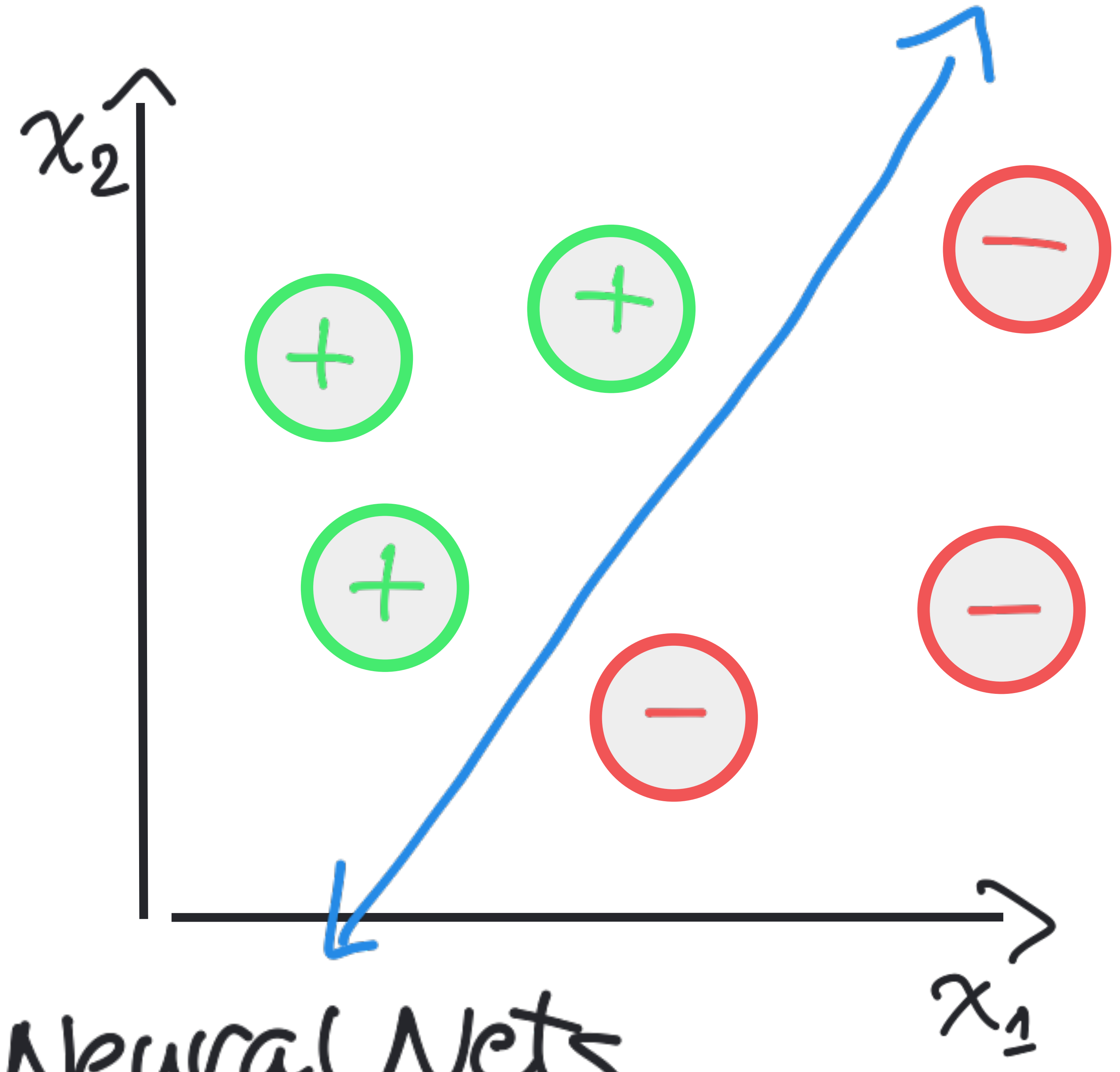
Goal*: Draw boundaries



KNNs
($\perp$ bisector)



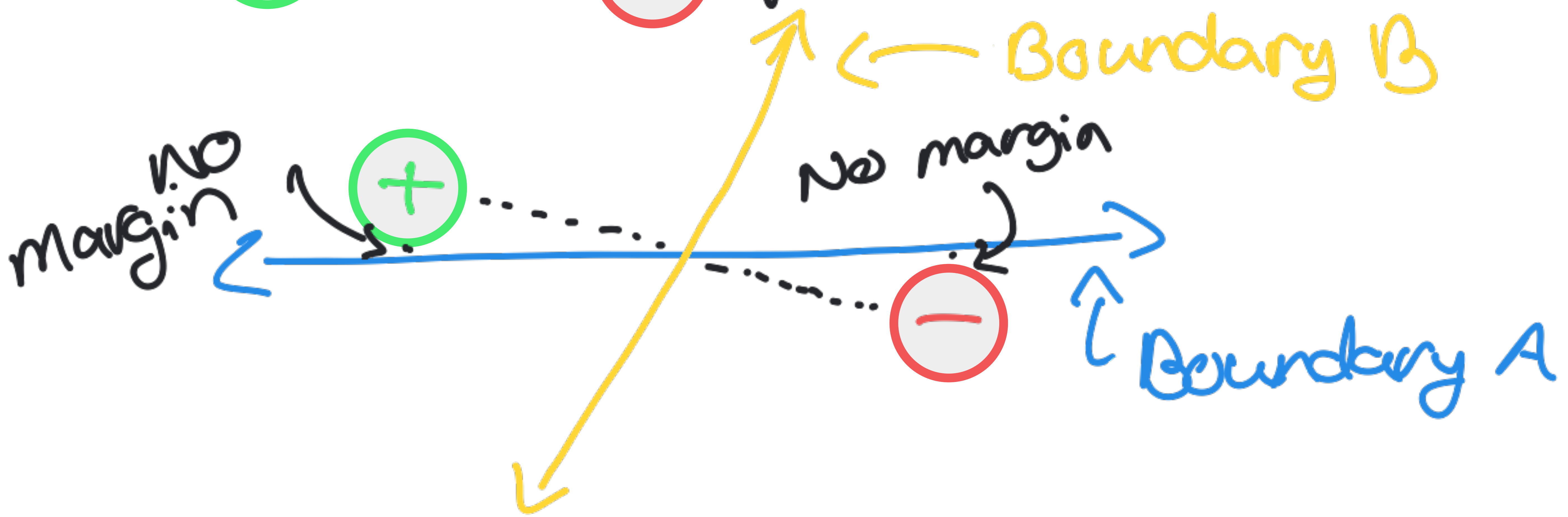
ID TREES
(x & y test)



Neural Nets

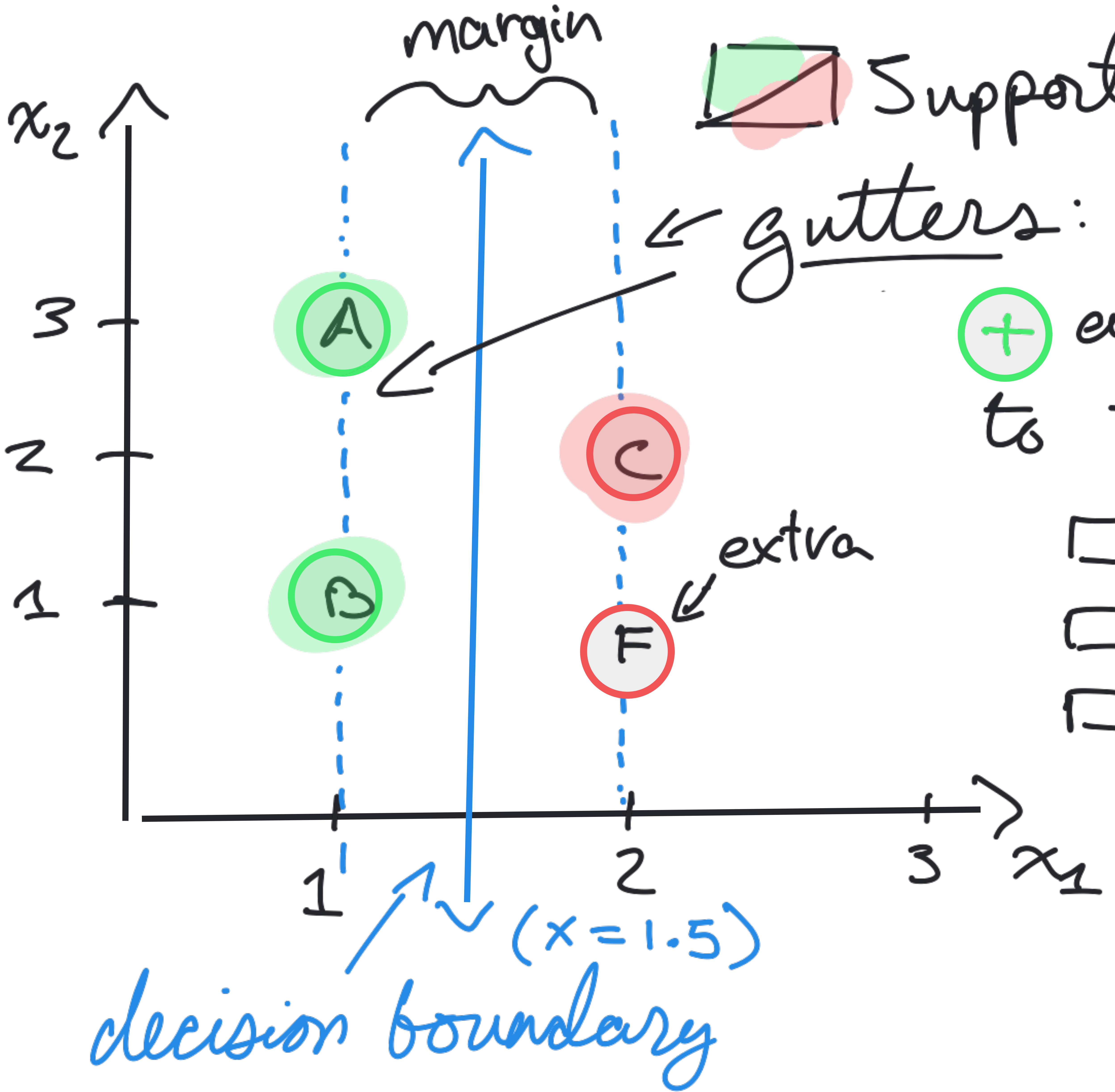
Support Vector Machines

- ☐ Binary (+/-) classifier
- ☐ Numerical classifier (NOT symbolic)
- ☐ Draws a single decision boundary to maximize margin width between + and − points



Prefer B to A due to the larger margins.

Support Vector Machines

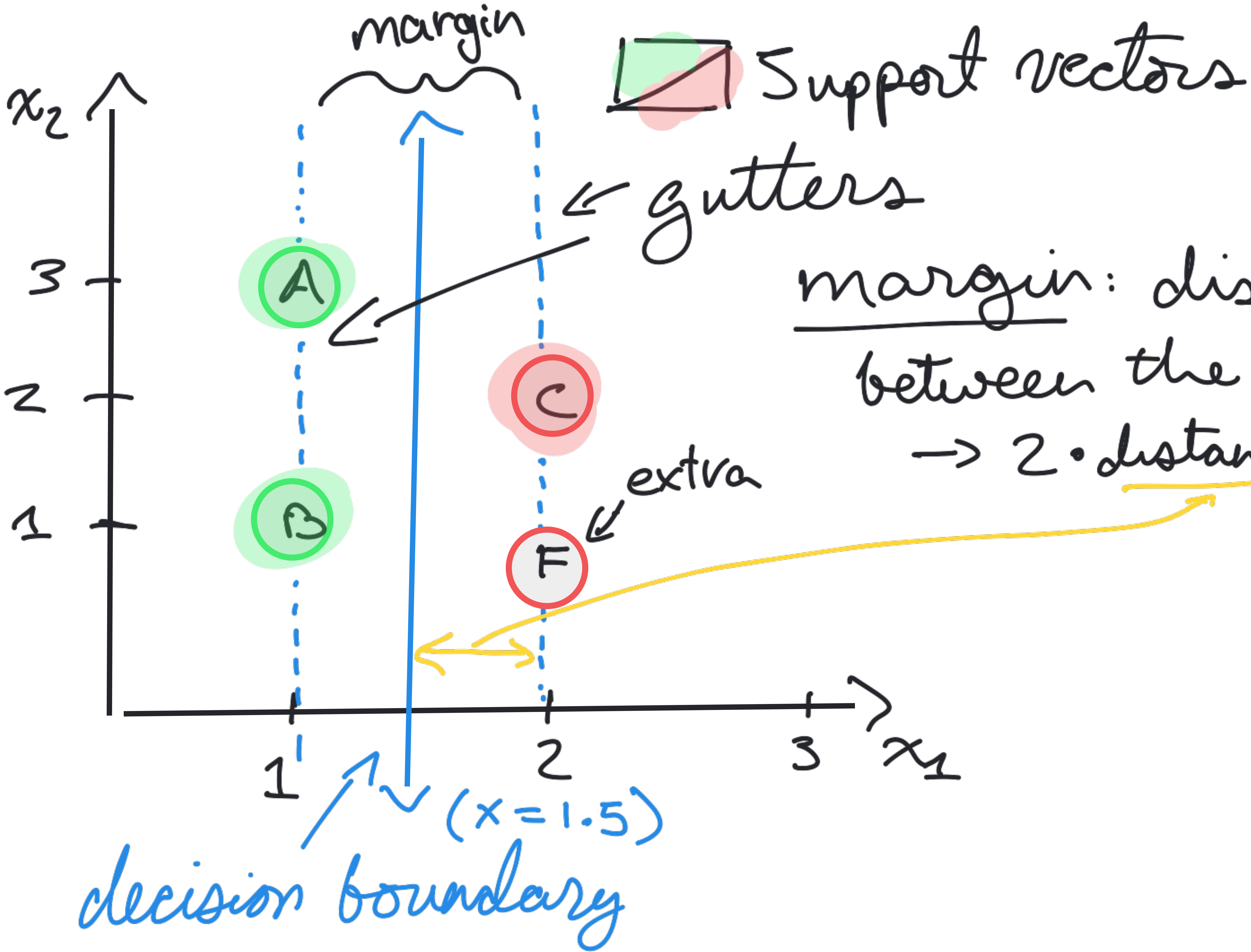


Support vectors

gutters: go through the + and − points closest to the decision boundary

- [ ] 4 training points
- [ ] 2 classes
- [ ] 3 support vectors

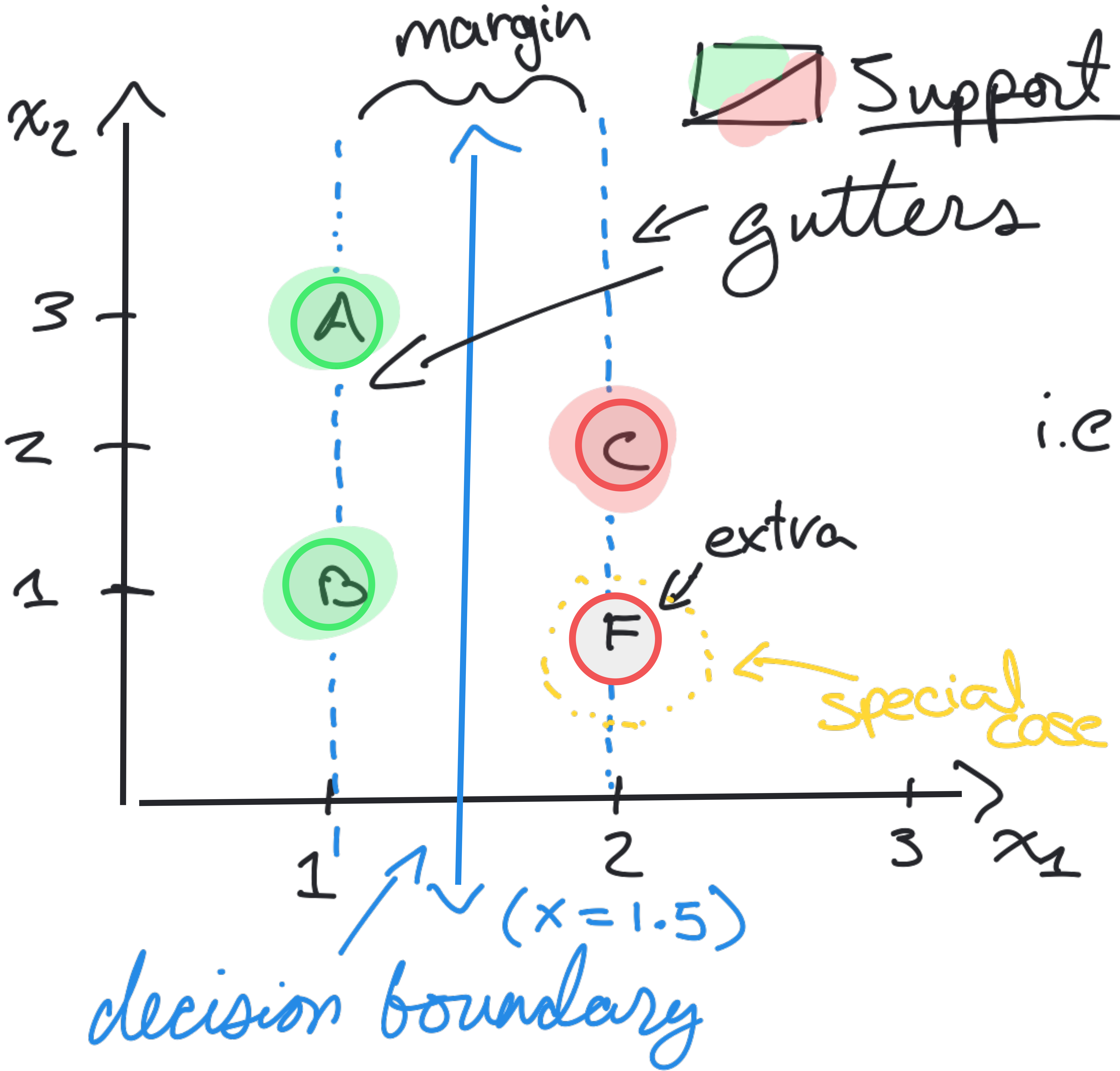
Support Vector Machines



<u>margin</u>: distance between the gutters

$\rightarrow 2 \cdot \text{distance}(\text{boundary}, \text{gutter})$

Support Vector Machines



gutters

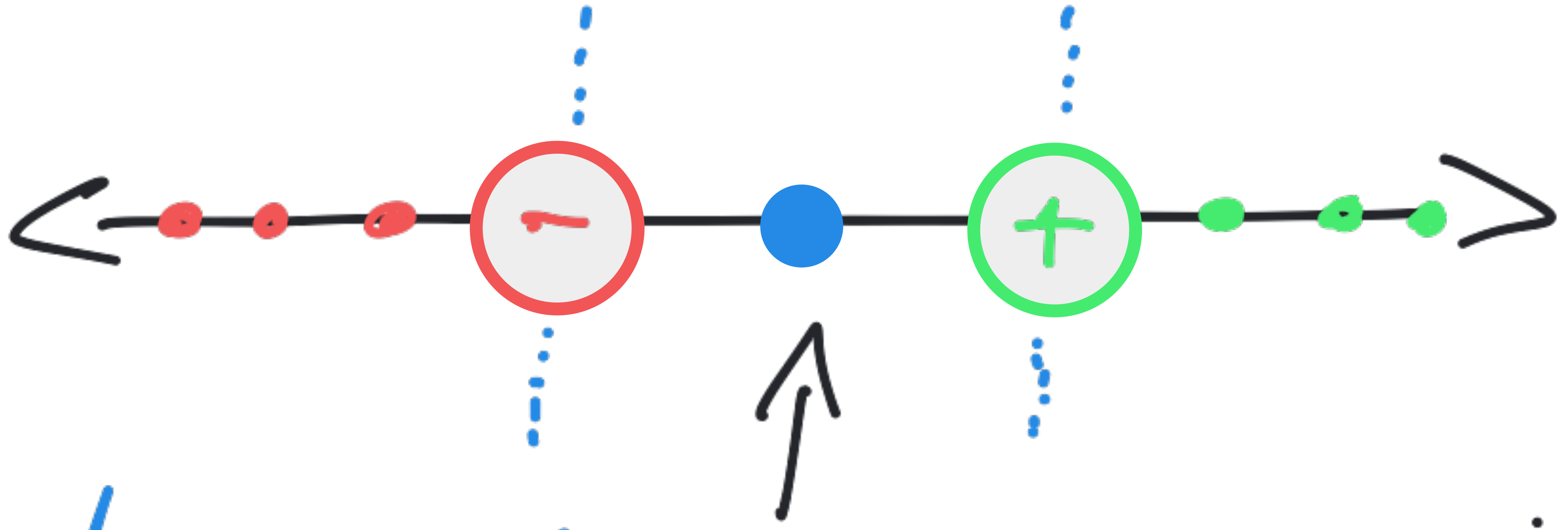
Support vectors: any point that defines the decision boundary

i.e. if a support vector is removed, the boundary will change

Support Vector Machines

Cases of Support Vectors :

1D :

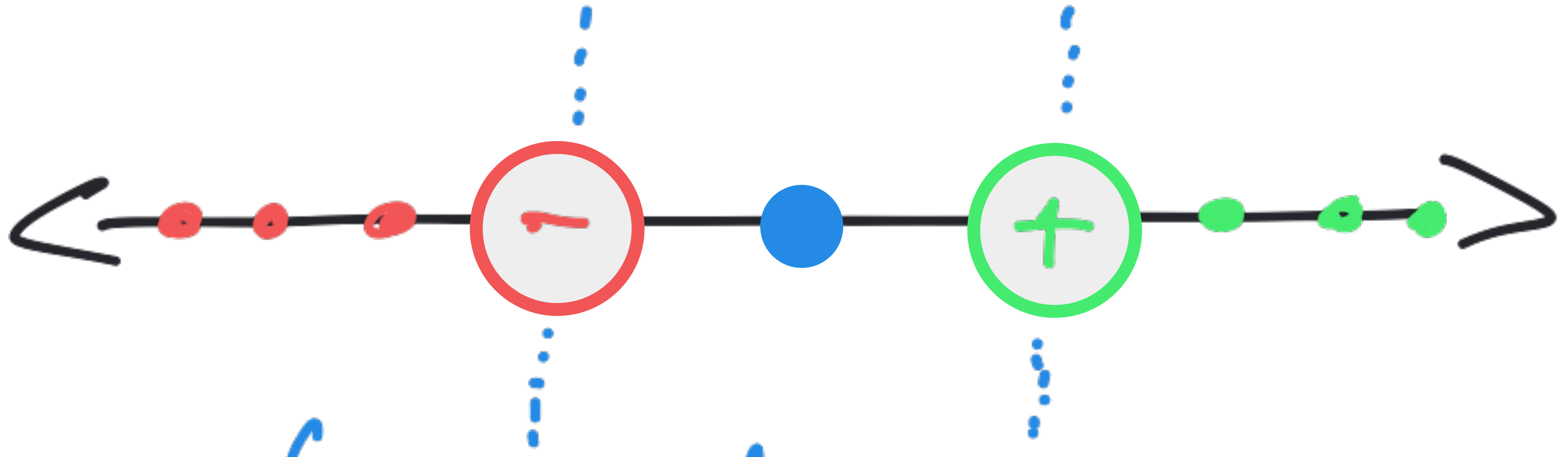


- ☐ Decision boundary is a point
- ☐ Only 2 support vectors

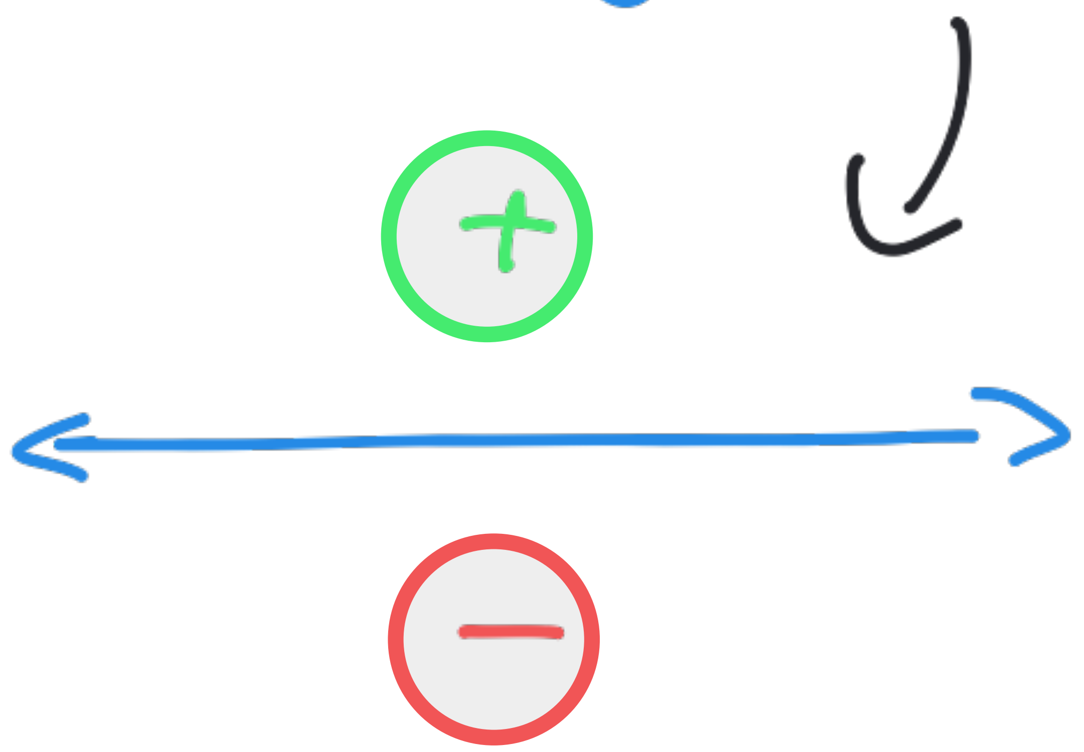
Support Vector Machines

Cases of Support Vectors :

1D :



2D : Decision Boundary is a line

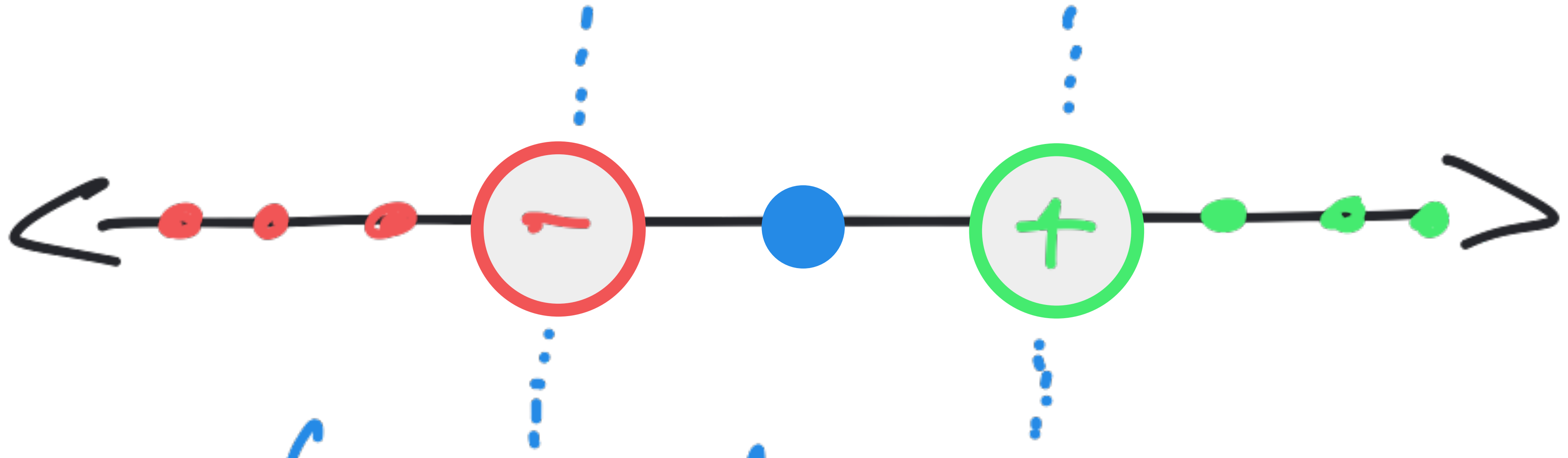


(typically) 2-3 support vectors *

Support Vector Machines

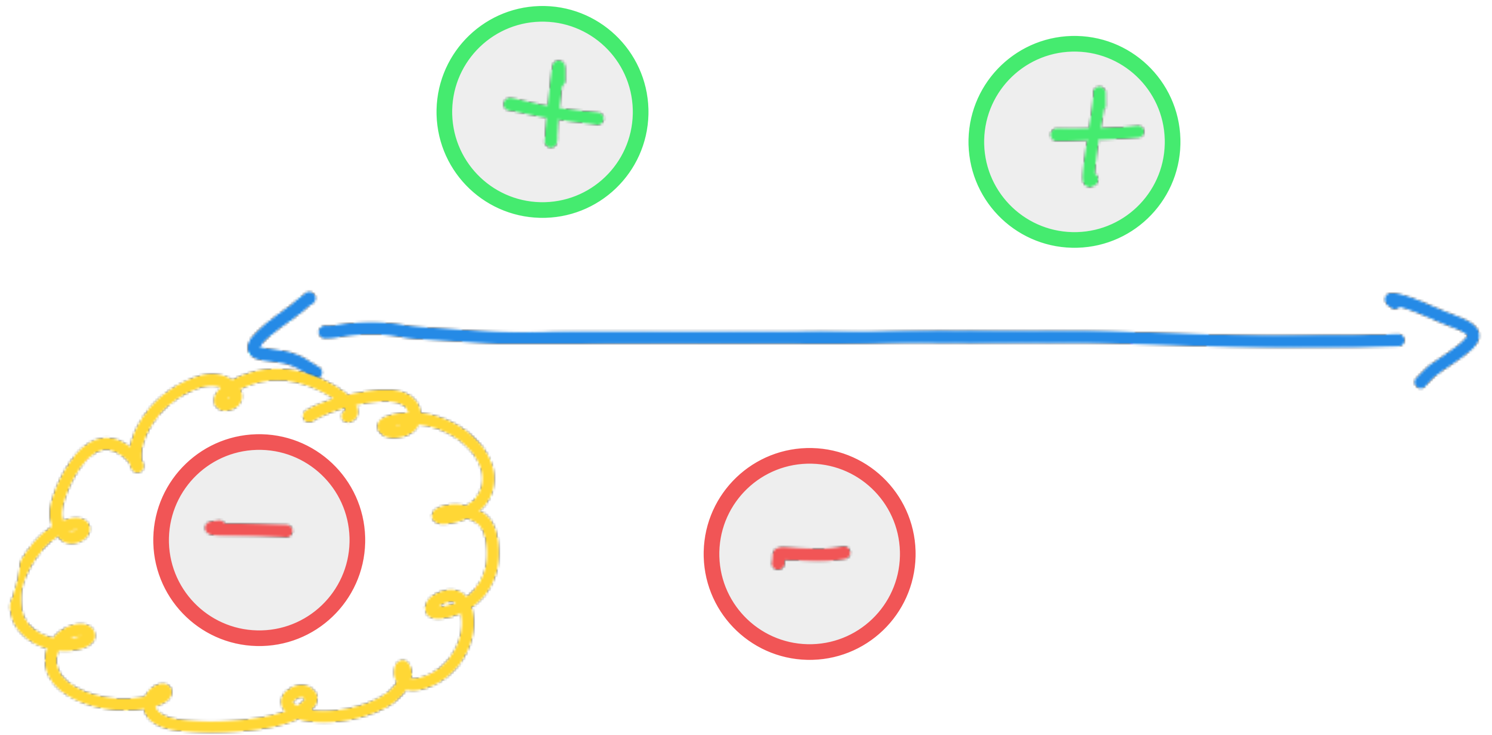
Cases of Support Vectors:

1D:



2D: Decision Boundary is a line

Suppose I add the following point...

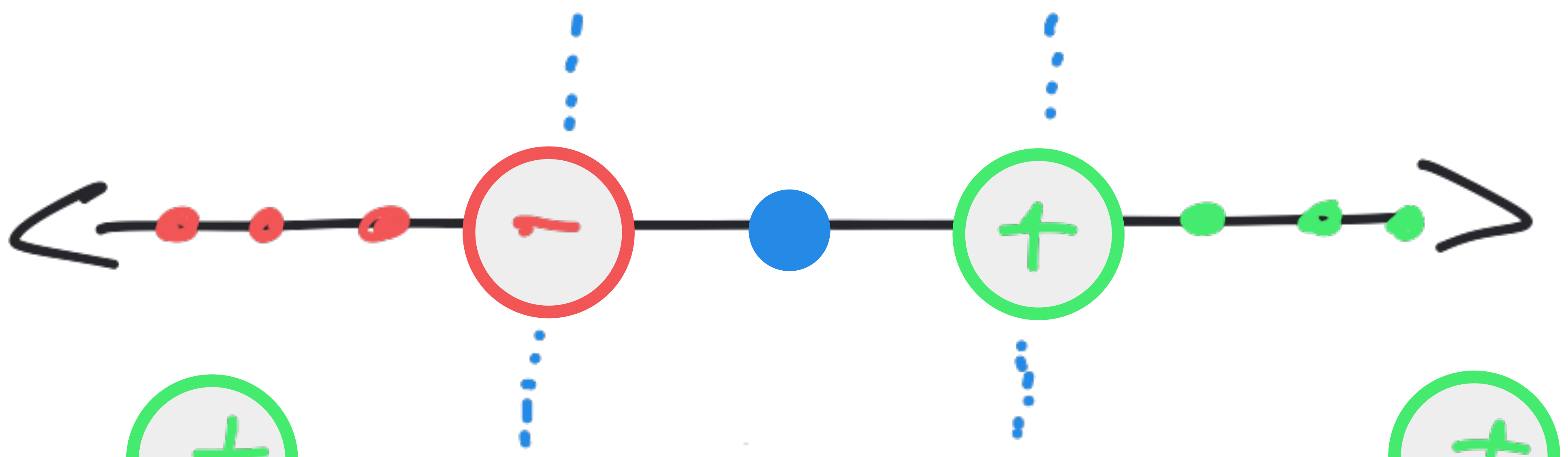


The notion of support vectors becomes more ambiguous

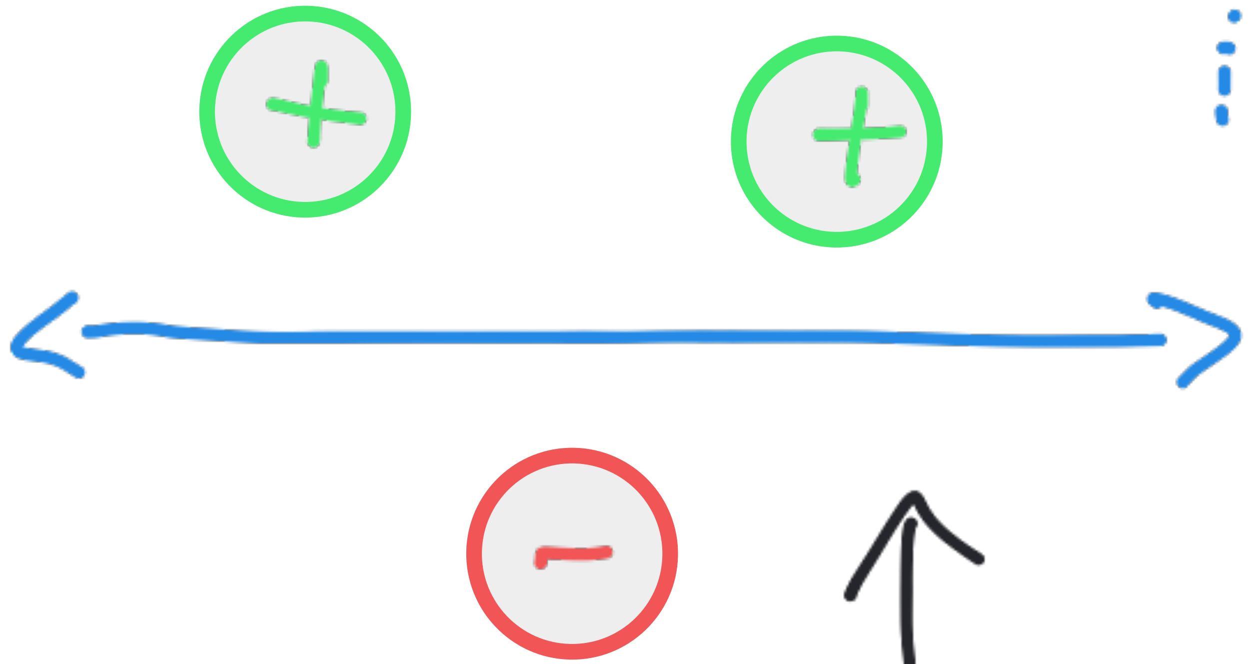
Support Vector Machines

Cases of Support Vectors:

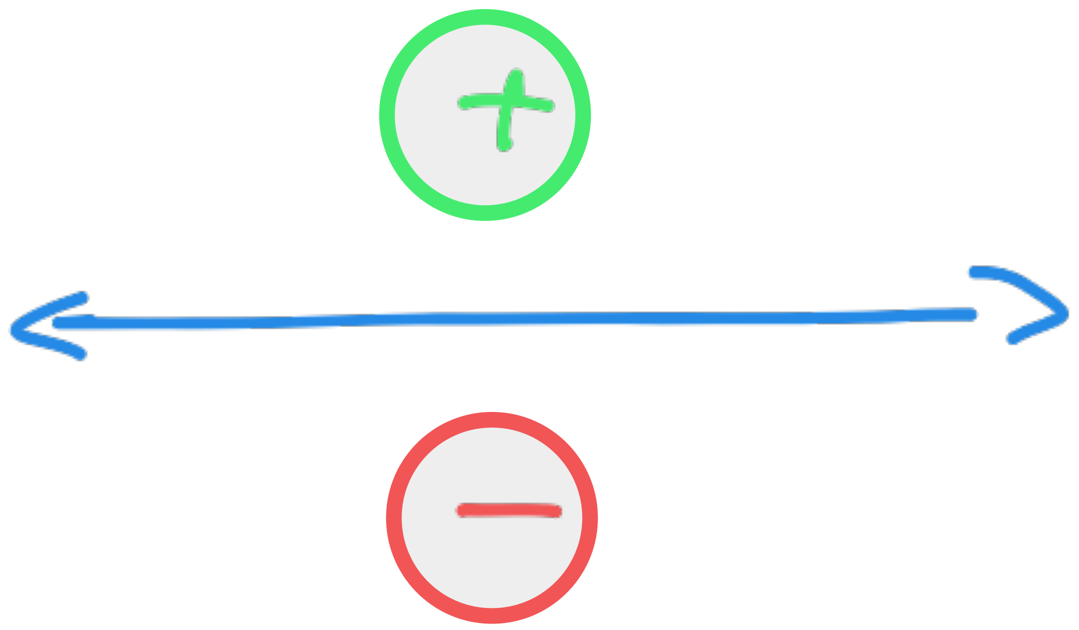
1D:



2D:

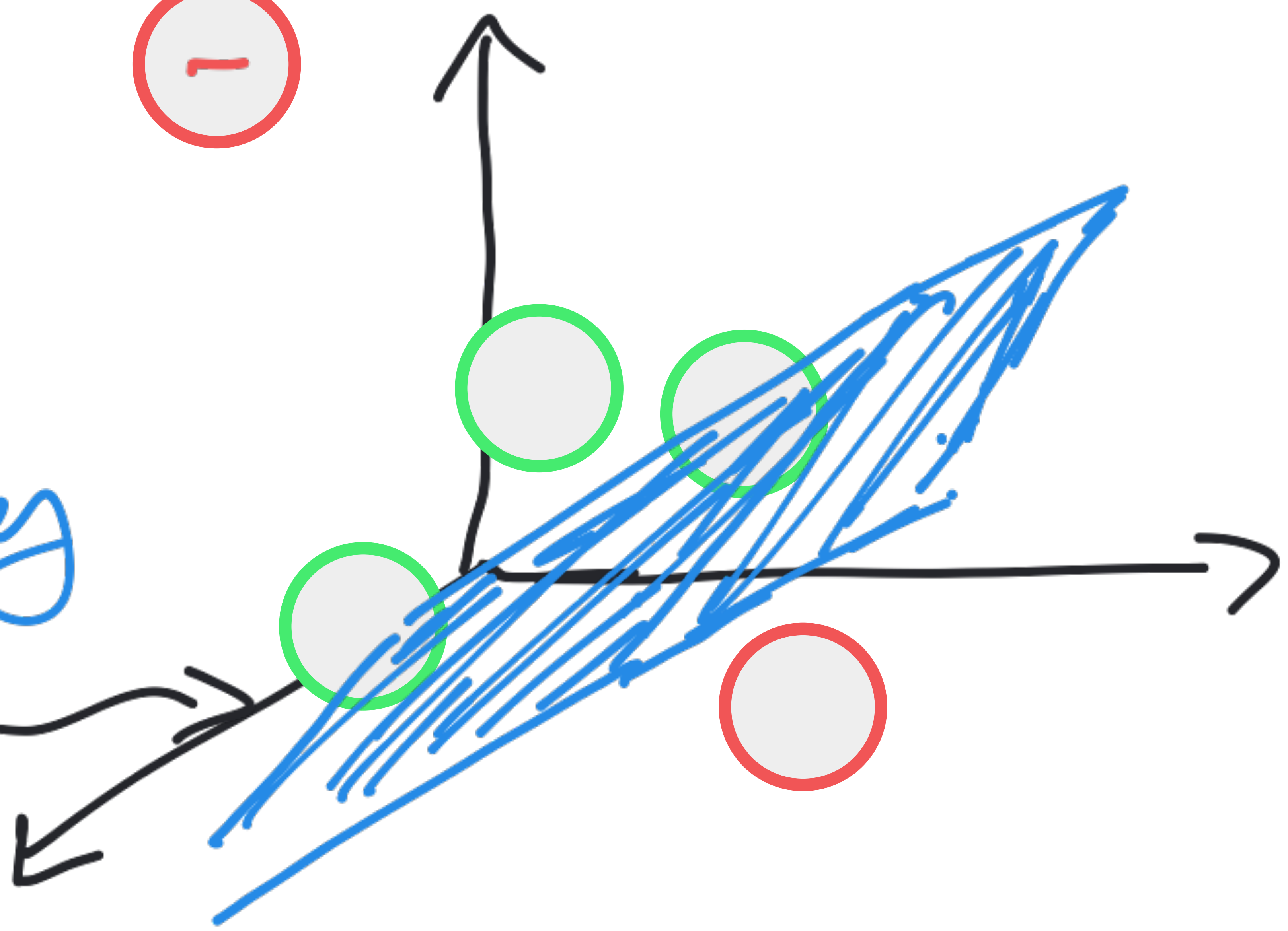


or



3D:

Decision boundary is a surface



Convex Hull Method

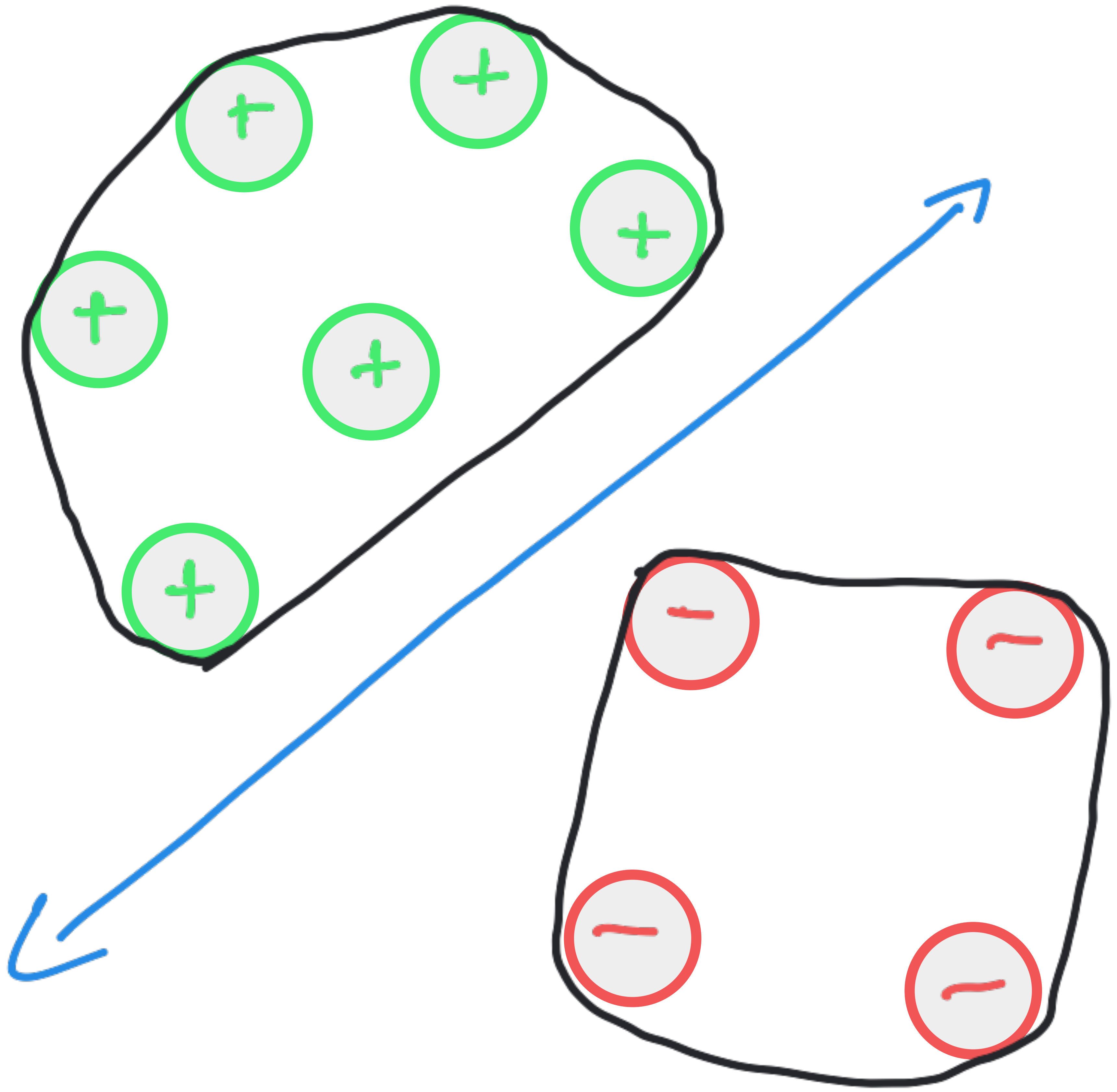
- [ ] A way of visualizing the decision boundary when it may be hard to see initially

- [ ] Imagine you stretch a rubber band around +s and −s

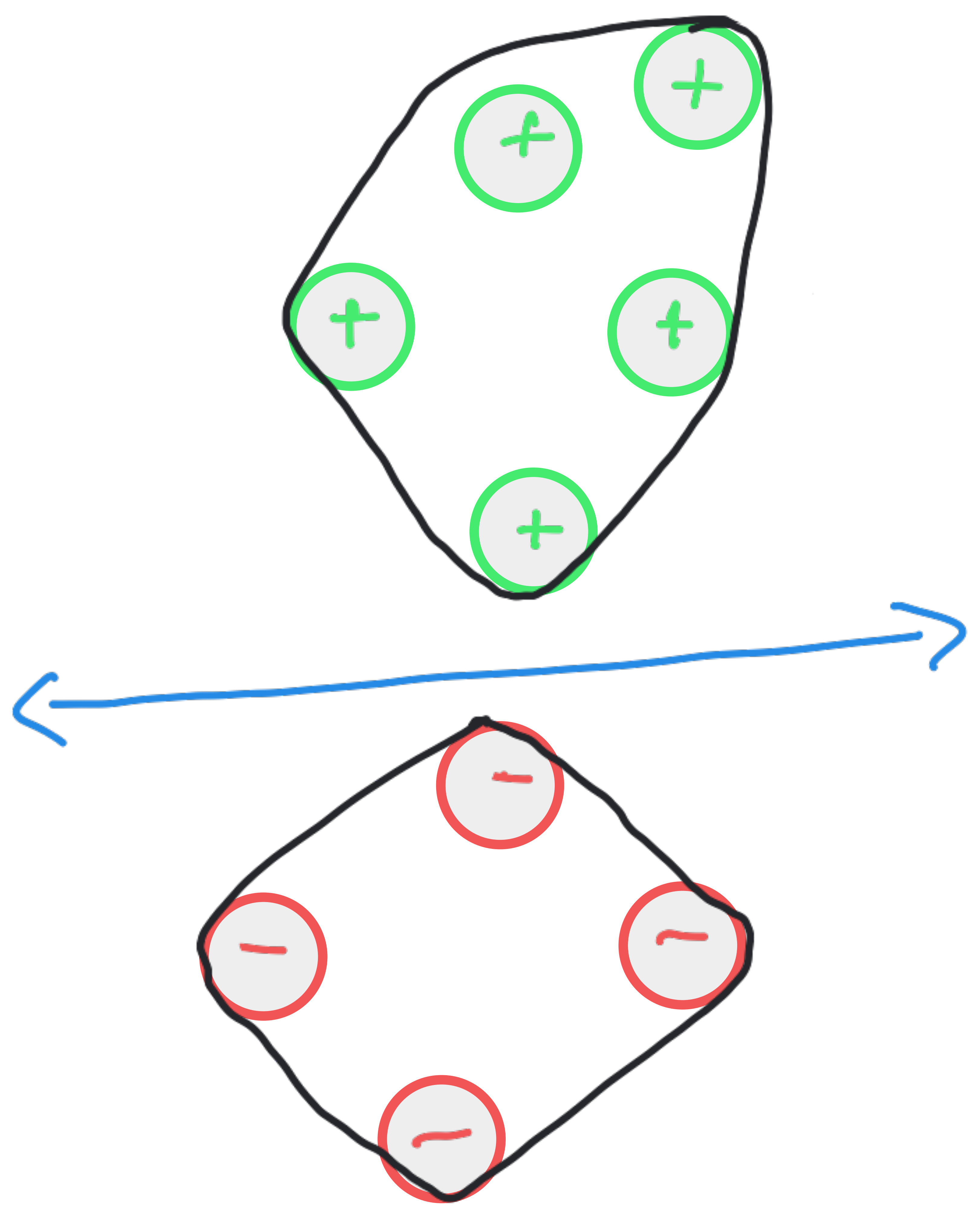
Convex Hull Method

Case 1:

- [ ] line & point
- [ ] decision boundary is // to the line
- [ ] 2/3 support vectors



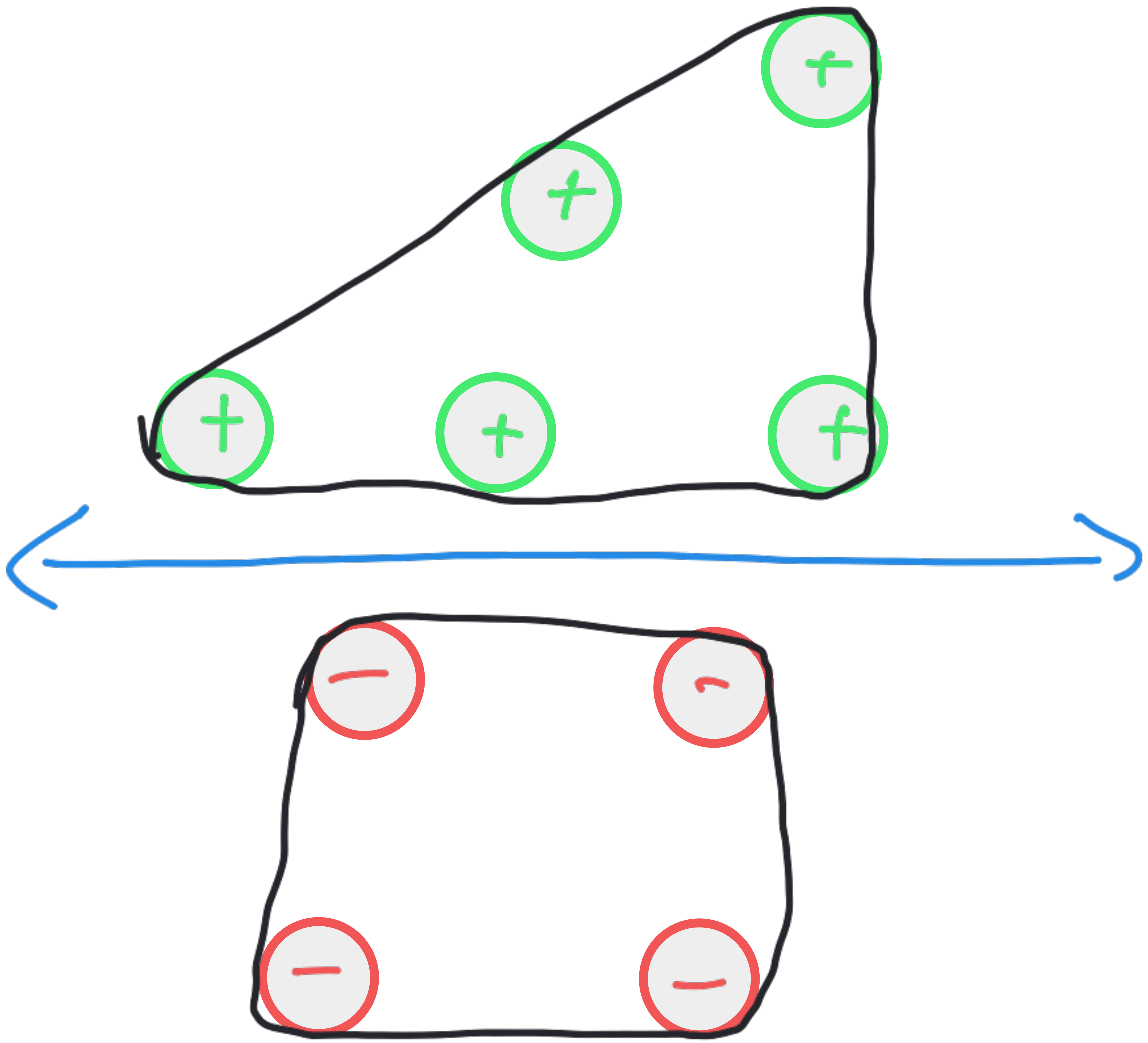
Convex Hull Method



Case 2:

- [ ] 2 points
- [ ] decision boundary is $\perp$ to the line connecting the points (KNNs)
- [ ] 2 support vectors

Convex Hull Method



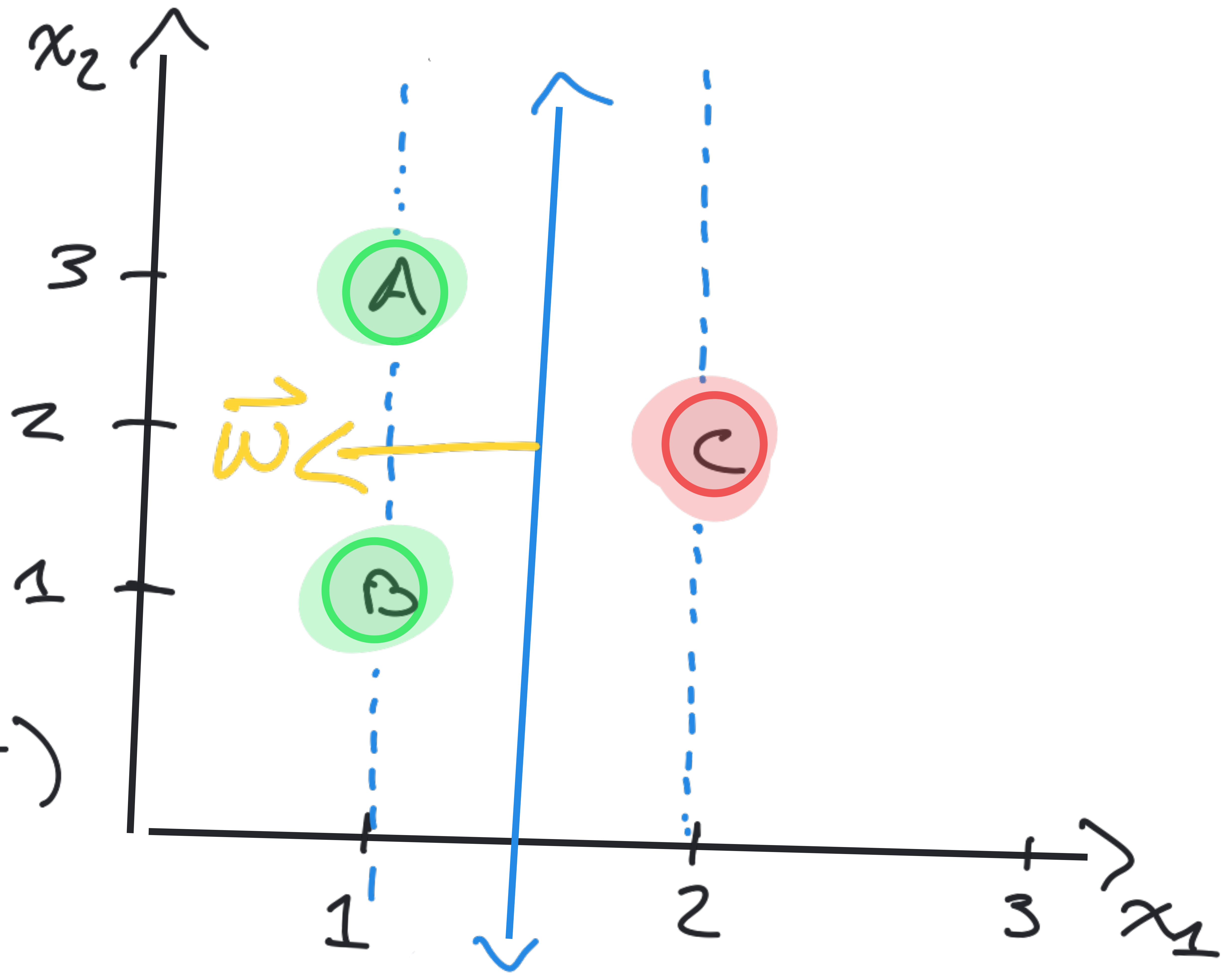
Case 3:

- [ ] 2 lines
- [ ] Decision boundary is halfway between the 2 lines
- [ ] 2/3 support vectors*

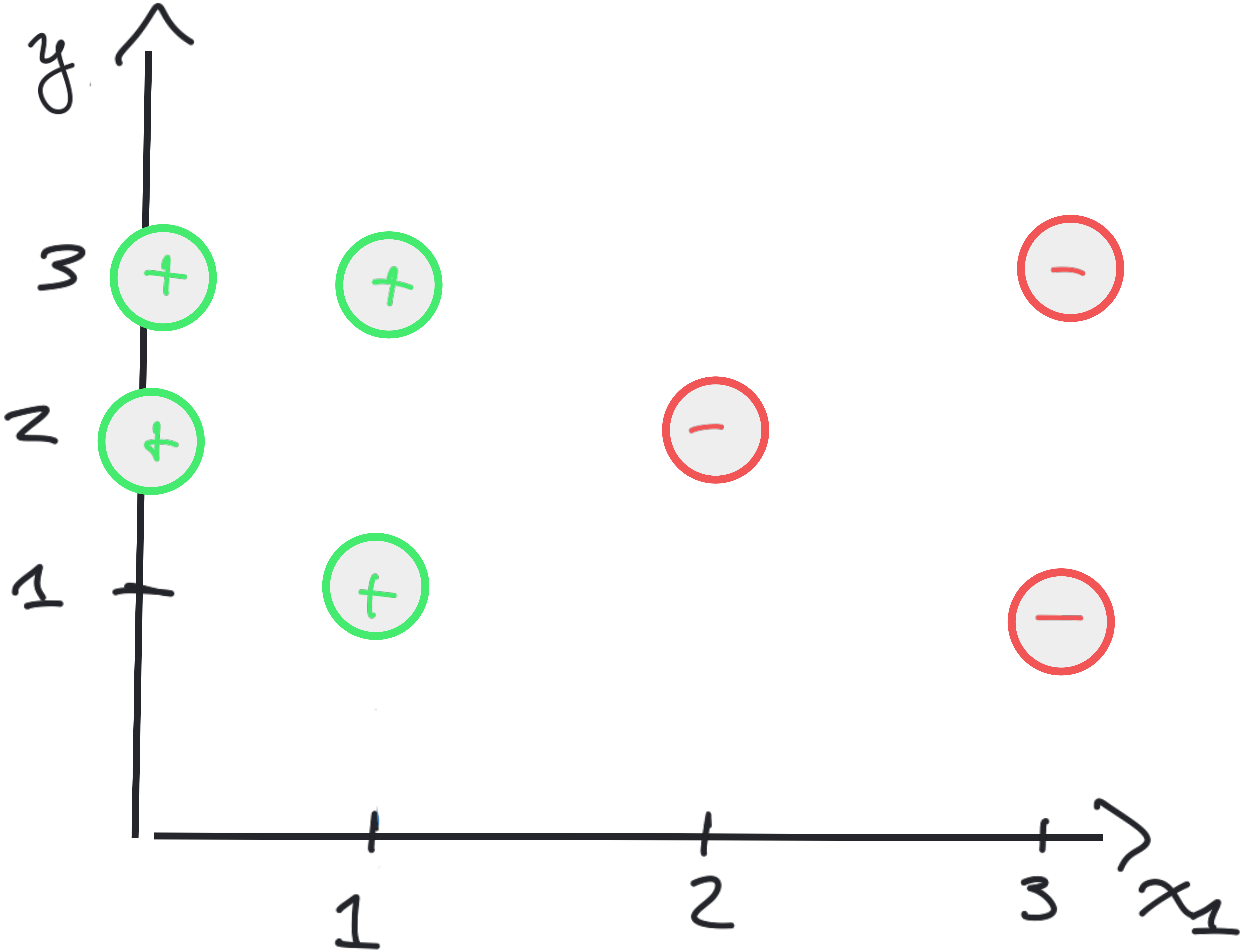
HOW TO DO IT

$\vec{w}$: normal vector $\perp$ to the decision boundary and points toward the $+$

b: offset (a.k.a. y-intercept)

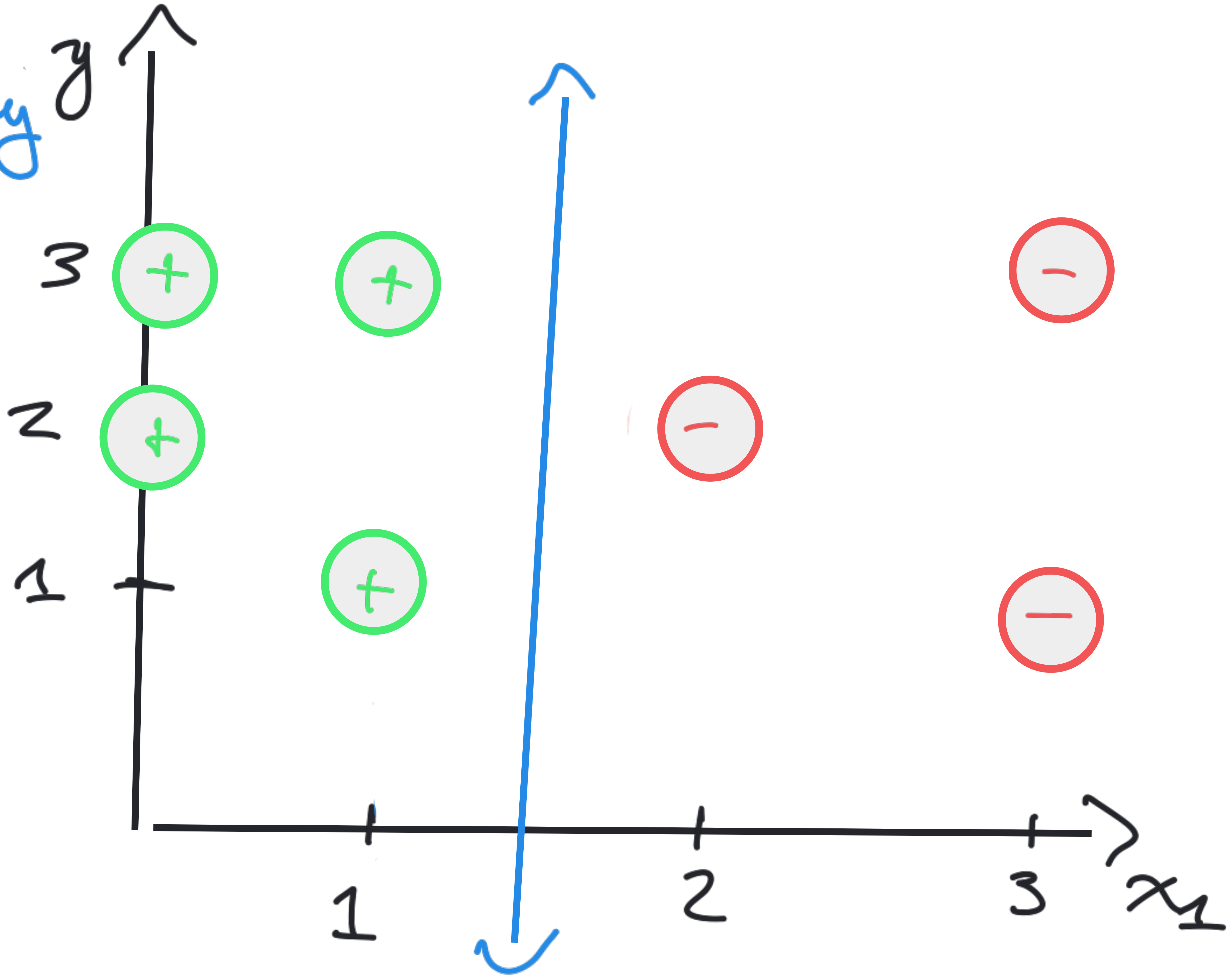


HOW TO DO IT



HOW TO DO IT

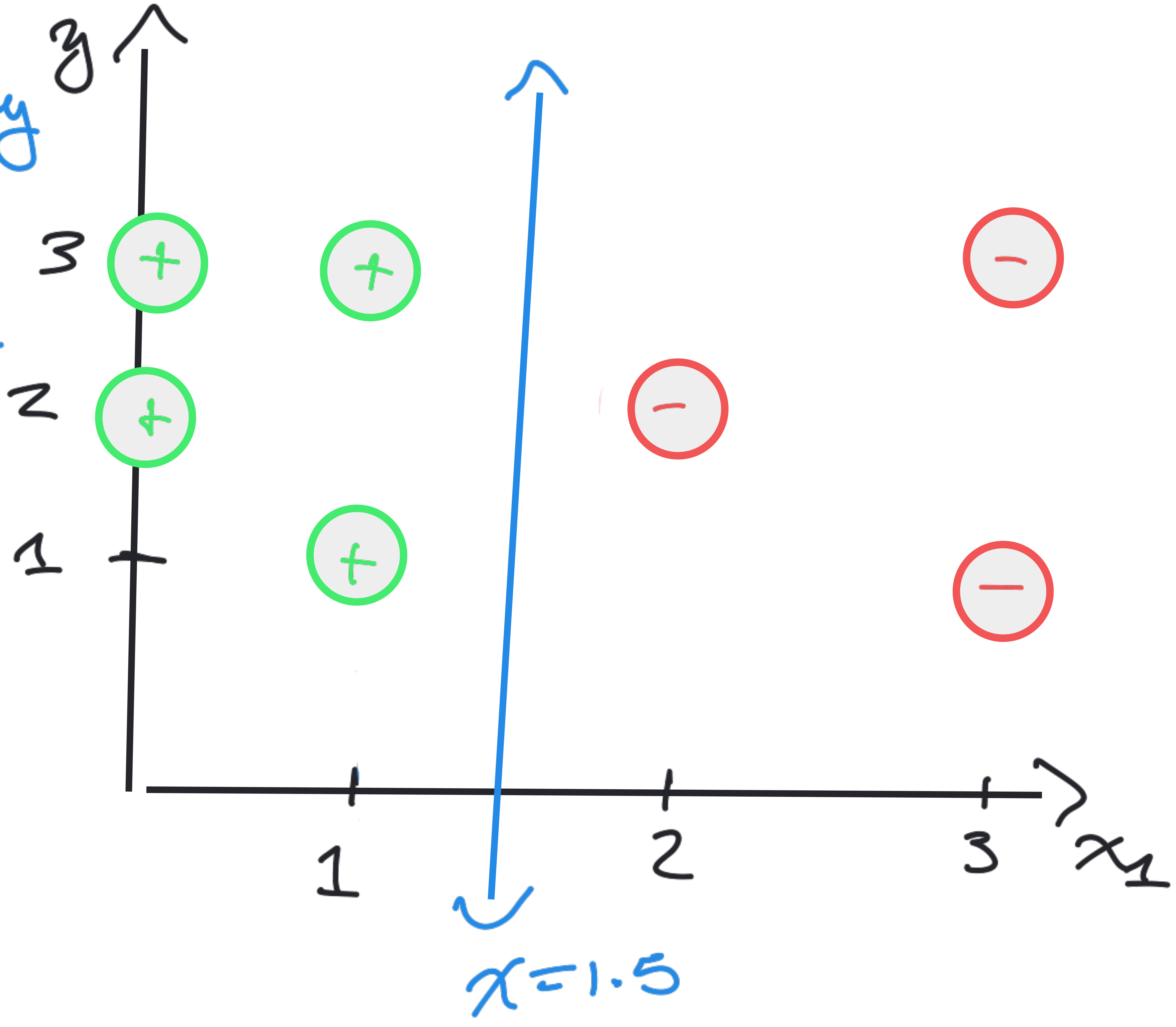
1) Draw decision boundary



HOW TO DO IT

1) Draw decision boundary
2) Write the equation for the boundary line

$x = 1.5$



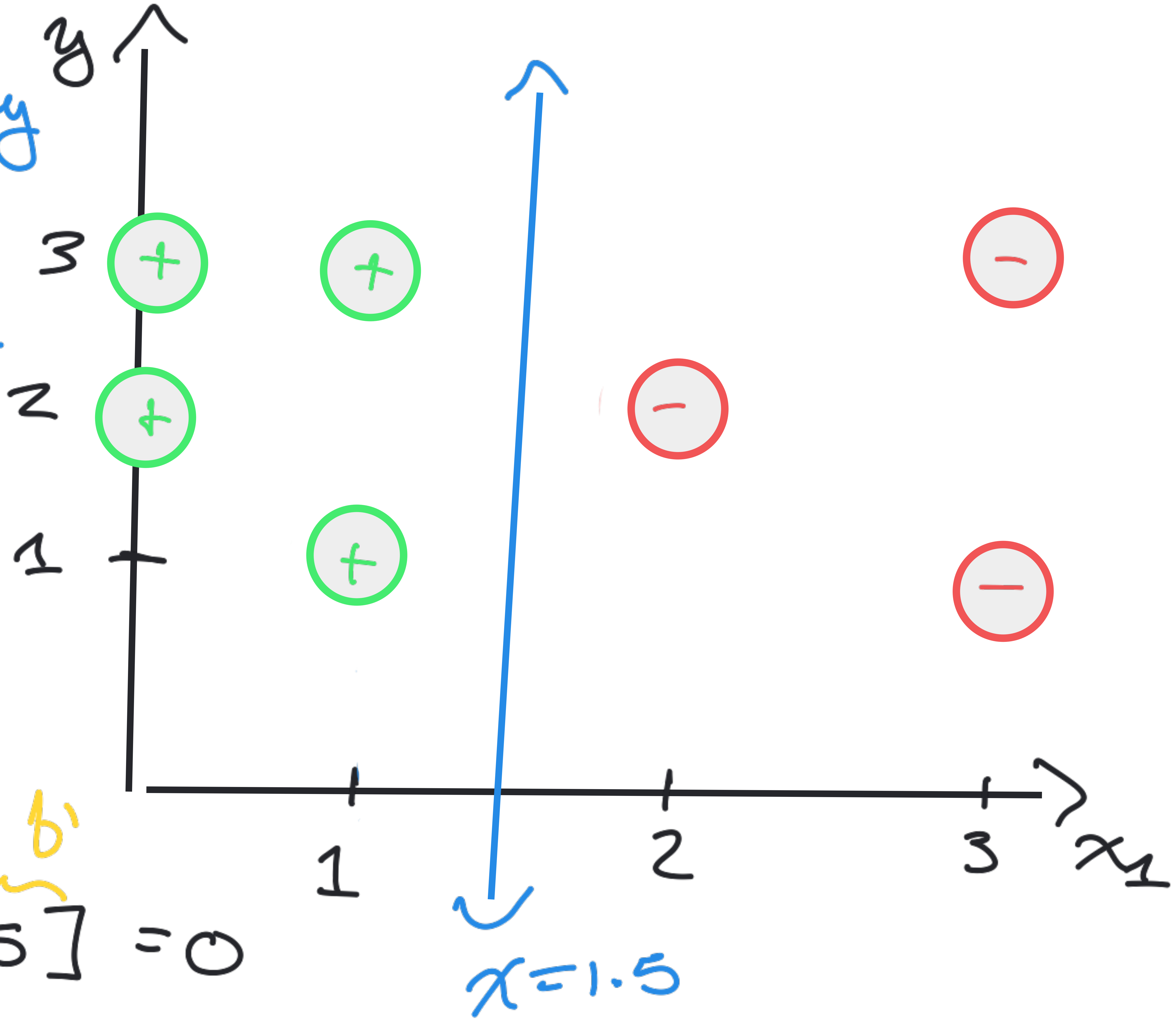
HOW TO DO IT

1) Draw decision boundary
2) Write the equation for the boundary line
3) Rewrite equation in the form $\vec{\omega}\vec{x}+b=0$

$$(1)x+(0)y=1.5$$

$$(1)x+(0)y-1.5=0$$

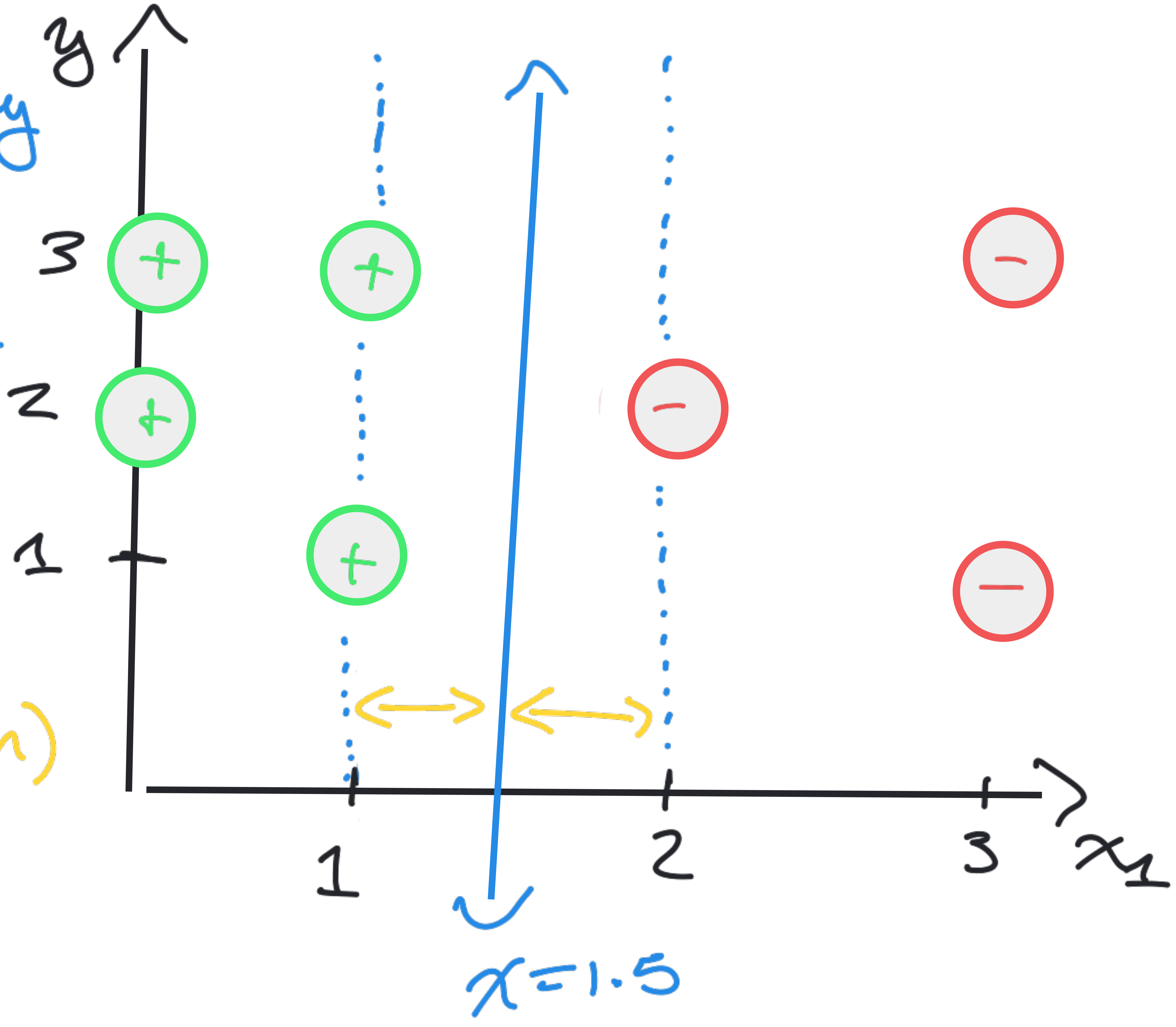
$$\underbrace{\begin{bmatrix}1 & 0\end{bmatrix}}_{\vec{\omega}'}\underbrace{\begin{bmatrix}x\\y\end{bmatrix}}_{\vec{x}}+\underbrace{\begin{bmatrix}-1.5\end{bmatrix}}_{b'}=0$$



HOW TO DO IT

1) Draw decision boundary
2) Write the equation for the boundary line
3) Rewrite equation in the form $\vec{\omega}\vec{x}+b=0$
4) Scale $\vec{\omega}\cdot\vec{x}+b=0$ so the marginwidth (m)

$$m = \frac{2}{\|\vec{\omega}\|}$$



HOW TO DO IT

- [ ] Equation 1: $\vec{w}\vec{x} + b = 0$

$\rightarrow [1\ 0]\begin{bmatrix} x \\ y \end{bmatrix} - 1.5 = 0 \Rightarrow \vec{w}' = [1\ 0]$

- [ ] Equation 2: $m = \dfrac{2}{\|\vec{w}\|}$

$\Rightarrow (1) = \dfrac{2}{\|\vec{w}\|}$

$\|\vec{w}\| = 2$ ☆

Now we know $\|\vec{w}\|$ and that it should point toward

HOW TO DO IT

- ☐ Equation 1: $\vec{w}\vec{x}+b=0$

  $\rightarrow [1\ 0]\begin{bmatrix} x \\ y \end{bmatrix} - 1.5 = 0$ $\Rightarrow \vec{w}' = [1\ 0]$

  $\Rightarrow \|\vec{w}\| = 2$

- ☐ Equation 2: $m = \dfrac{2}{\|\vec{w}\|}$

- ☐ Equation 3: gutter constraint
  - $\forall$ points : $\text{sign}(\vec{w}\cdot\vec{x}+b) = y_i$ $y_i \rightarrow$ classification of point x_i
  - $\forall$ points on the gutter:

    $$\vec{w}\cdot\vec{x}+b = \pm 1$$

HOW TO DO IT

Apply equation $\boxed{3}$ onto point C :

$$[1\ 0]\begin{bmatrix}2\\2\end{bmatrix} - 1.5 = -1$$

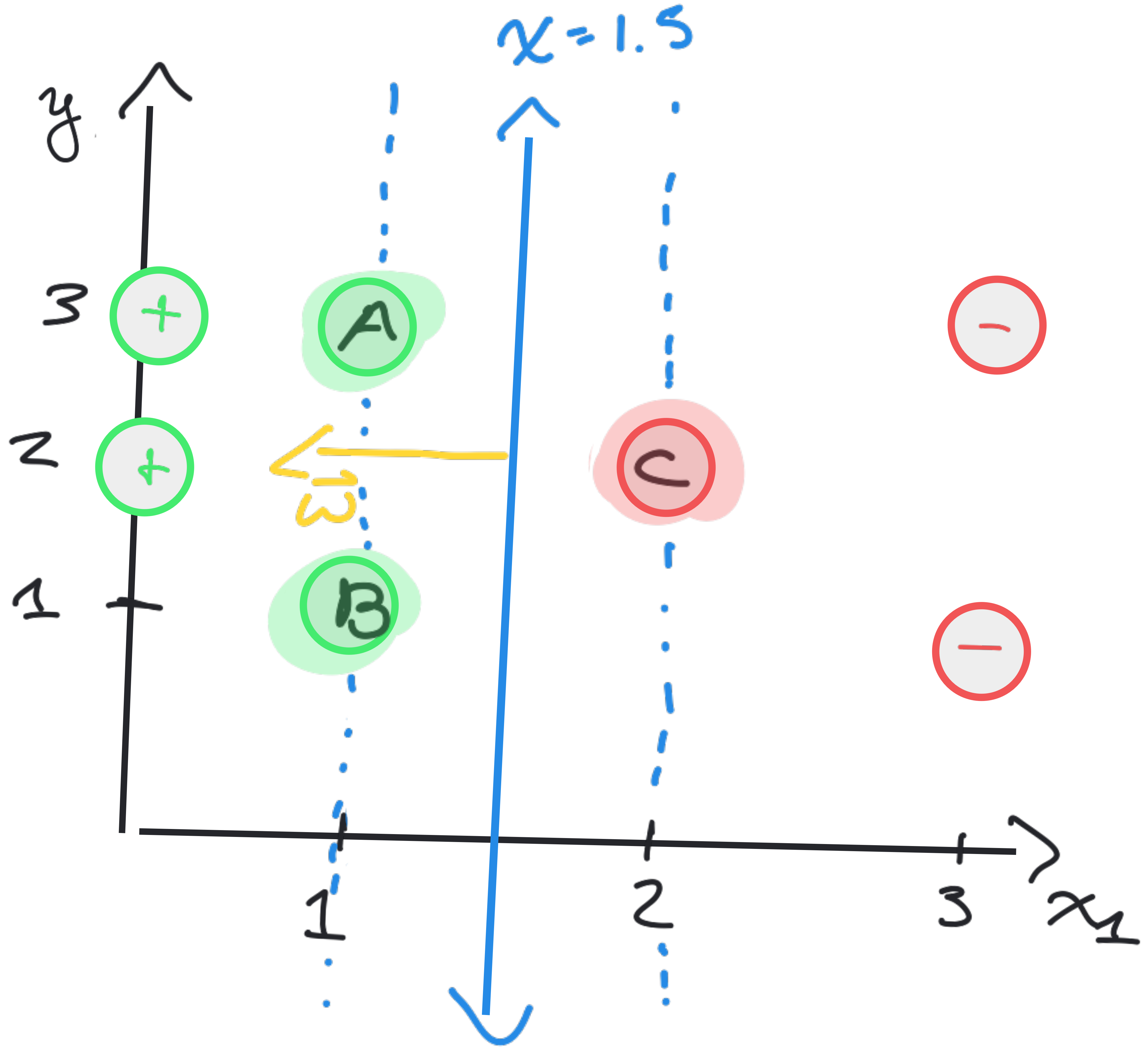
$$= 0.5 \neq -1$$

So we need to scale $\vec{w}'$ and b'

$$c \cdot \left([1\ 0]\begin{bmatrix}2\\2\end{bmatrix} - 1.5\right) = -1$$

$$c \cdot (0.5) = -1$$

$$c = -2$$



HOW TO DO IT

Apply equation $\boxed{3}$ onto point C :

$$-2\left(\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}2\\2\end{bmatrix}-1.5\right)=-1$$

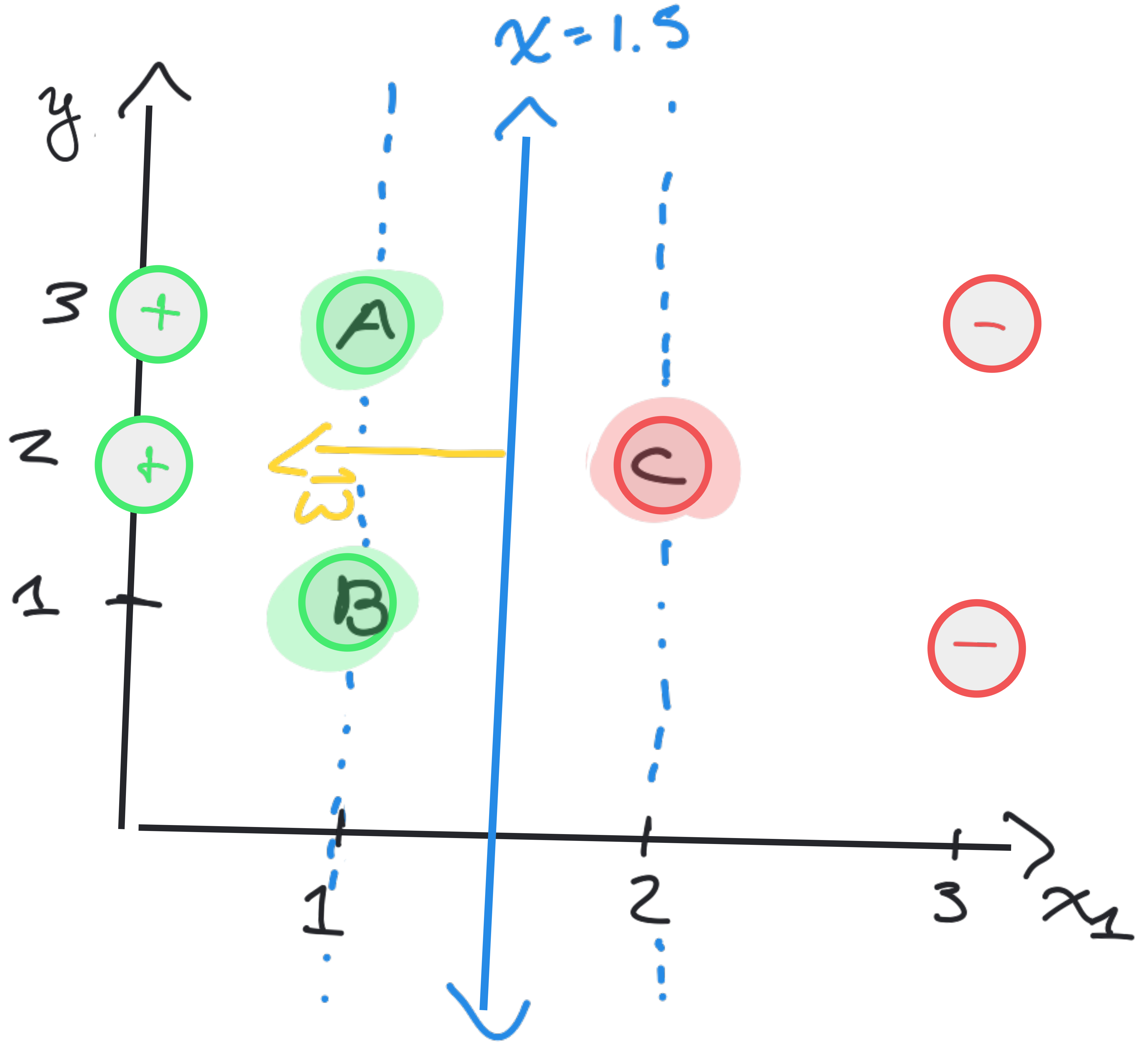
$$=\begin{bmatrix}-2 & 0\end{bmatrix}\begin{bmatrix}2\\2\end{bmatrix}+3 \quad =-1$$

$$=-1=-1 \checkmark$$

$$\Rightarrow \begin{bmatrix}-2 & 0\end{bmatrix}\cdot\vec{x}_i+\begin{bmatrix}3\end{bmatrix}=y_i$$

$$\vec{w}=[-2,0] \leftarrow$$

$$b=[3]$$



★ This is consistent with our findings from equation $\boxed{2}$

HOW TO DO IT

Calculating α values (lagrange multipliers)

- ☐ α: how much a point supports the boundary
- ☐ via sequential optimization of the margin
- ☐ iteratively adjust α for each point to maximize the margin

HOW TO DO IT

Calculating α values (lagrange multipliers)

1) For all non-support vectors: $\alpha = 0$
otherwise, point is support vector and lies on the gutter: $\alpha > 0$

Equation 4

2) $\sum_{\text{support vector } i} \alpha_i y_i = 0 \iff \sum_{+\text{ support vectors}} \alpha_+ = \sum_{-\text{ support vectors}} \alpha_-$

Equation 5

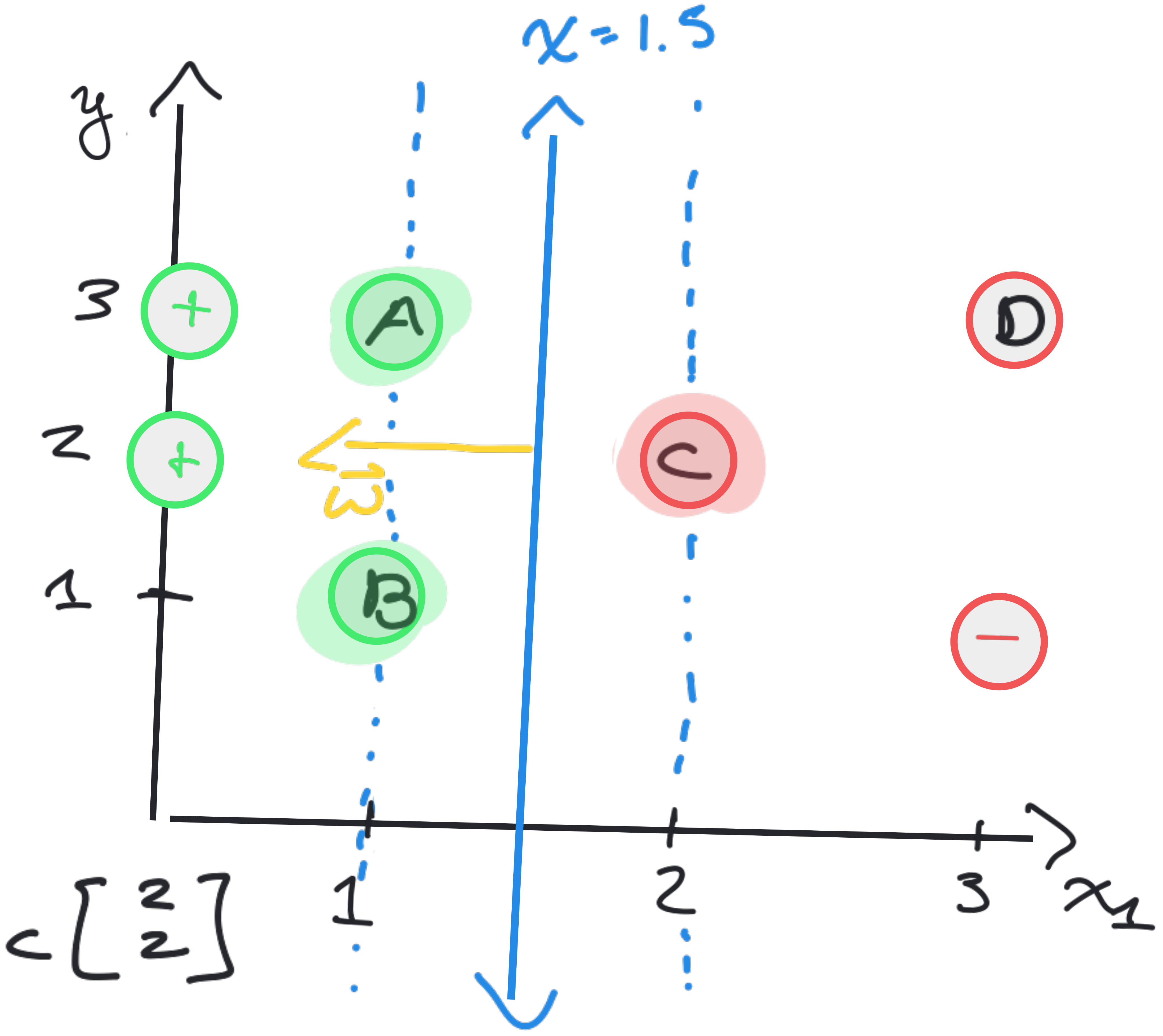
3) $\sum_{\text{support vectors } i} \alpha_i y_i \vec{x}_i = \vec{\omega} \iff \vec{\omega} = \sum_{+\text{ support vectors}} \alpha_+ \vec{x}_+ - \sum_{-\text{ support vectors}} \alpha_- \vec{x}_-$

HOW TO DO IT

Apply equation $\boxed{4}$ onto points A, B, C, D :

$$\alpha_A + \alpha_B - \alpha_C - \alpha_D = 0$$

$$\Rightarrow \alpha_A + \alpha_B = \alpha_C + \cancel{\alpha_D}^{\,0}$$



Apply equation $\boxed{5}$

$$\begin{bmatrix} -2 \\ 0 \end{bmatrix} = \alpha_A \begin{bmatrix} 1 \\ 3 \end{bmatrix} + \alpha_B \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \alpha_C \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

$\hookrightarrow -2 = \alpha_A + \alpha_B - 2\alpha_C$

$\hookrightarrow 0 = 3\alpha_A + \alpha_B - 2\alpha_C$

HOW TO DO IT

Now there is a system of . . .

- [ ] 3 equations
- [ ] 3 unknowns

$$0 = \alpha_A + \alpha_B - \alpha_C$$

$$-2 = \alpha_A + \alpha_B - 2\alpha_C$$

$$0 = 3\alpha_A + \alpha_B - 2\alpha_C$$

$\alpha_A = \alpha_B = 1$ ← $\alpha_A = \alpha_B$ because A & B are symmetric around C

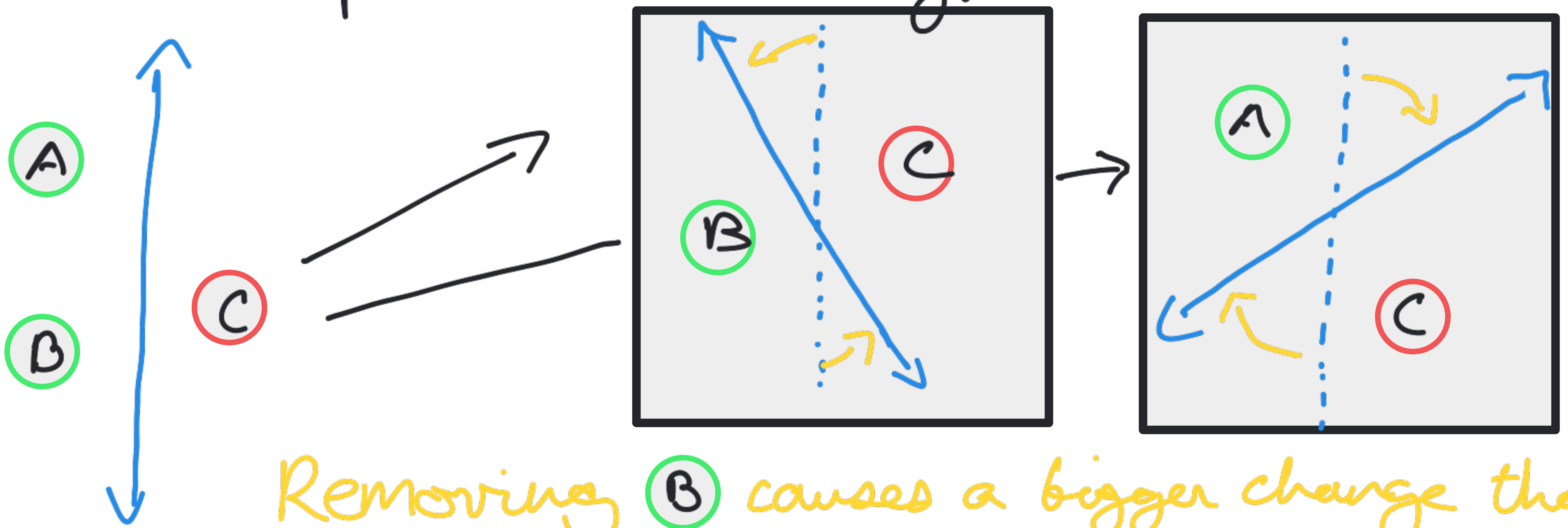
$\alpha_C = 2$ ← α_C is larger because it supports the boundary by itself

HOW TO DO IT

☐ What happens to α_s when margins increase?

margin width $\uparrow \rightarrow m\uparrow = \frac{2}{\|\vec{w}\|}\uparrow \Rightarrow \|\vec{w}\|\downarrow = \alpha_s\downarrow$

☐ What if points are not symmetric?



Removing B causes a bigger change than removing A $\Rightarrow \alpha_B > \alpha_A$

KERNEL TRICK

- ☐ What if the data is not linearly separable?
  - ☐ Apply transformations (e.g. polar)
  - ☐ Replace the dot product with another kernel - another function that transforms the space

  - ☐ current kernel: $K(\vec{u}, \vec{v}) = \vec{u} \cdot \vec{v}$

KERNEL TRICK

- Current kernel : $K(\vec{u}, \vec{v}) = \vec{u} \cdot \vec{v}$

$$\text{class}(\vec{x}_i) = \text{sign}(\vec{w} \cdot \vec{x}_i + b)$$
$$= \text{sign}\left(\left[\sum_j \alpha_j y_j \vec{x}_j\right] \cdot \vec{x}_i + b\right)$$
$$= \text{sign}\left(\sum_j \alpha_j y_j (\vec{x}_j \cdot \vec{x}_i) + b\right)$$

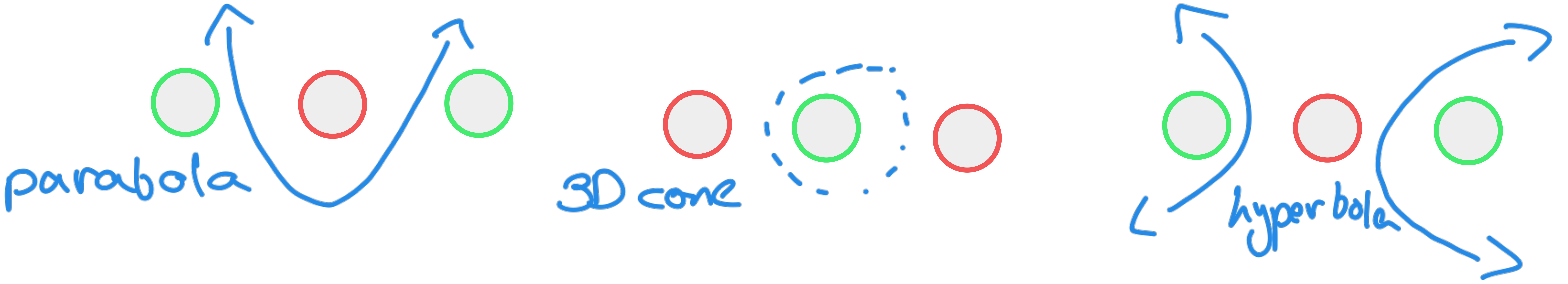
$$\text{class}(\vec{x}_i) = \text{sign}\left(\sum_j \alpha_j y_j \underline{K(\vec{x}_j, \vec{x}_i)} + b\right)$$

- use kernel to determine whether $\vec{x}_i$ is more similar to $+$ or $-$

- A kernel function is a similarity measure

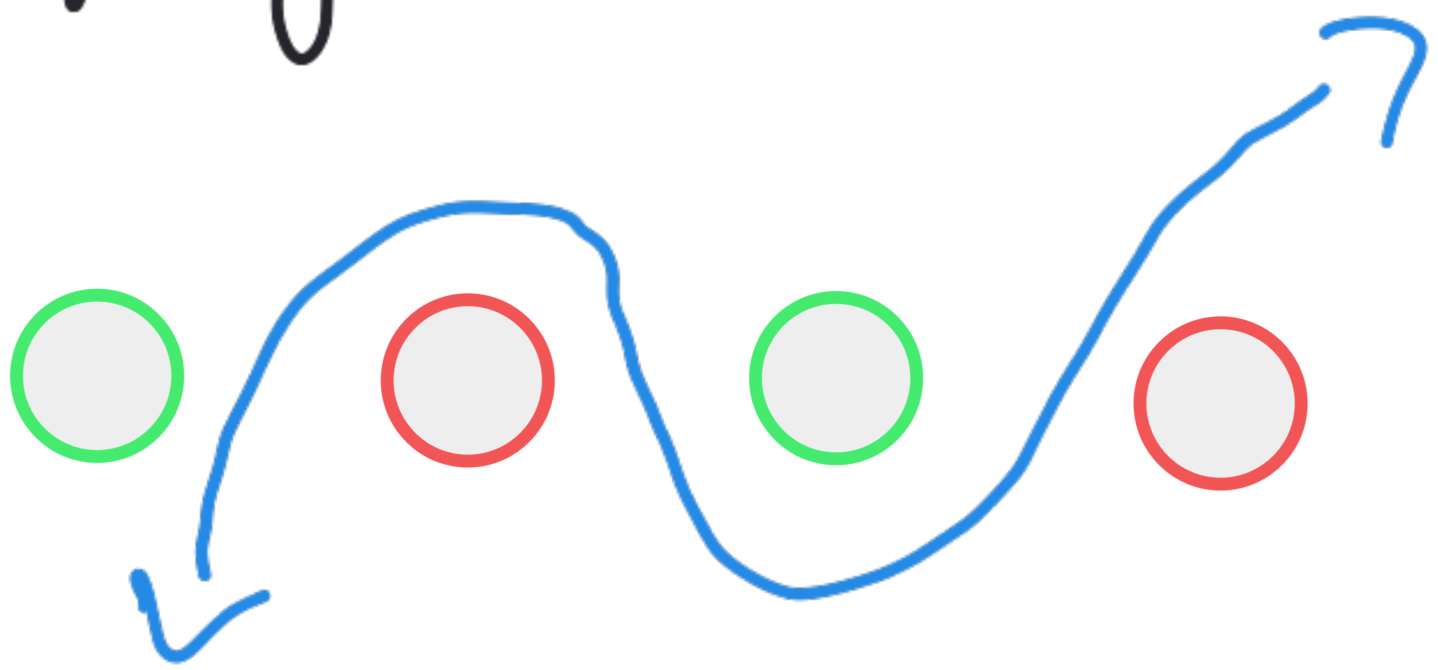
KERNEL TRICK

- Common kernels
  - linear: $K(\vec{u},\vec{v}) = \vec{u}\cdot\vec{v} + b$ (line)
  - quadratic: $K(\vec{u},\vec{v}) = (\vec{u}\cdot\vec{v} + b)^2$ (conic section)



KERNEL TRICK

- Common Kernels
  - linear: $K(\vec{u},\vec{v}) = \vec{u}\cdot\vec{v} + b$ (line)
  - quadratic: $K(\vec{u},\vec{v}) = (\vec{u}\cdot\vec{v} + b)^2$ (conic section)
  - polynomial: $K(\vec{u},\vec{v}) = (\vec{u}\cdot\vec{v} + b)^p$

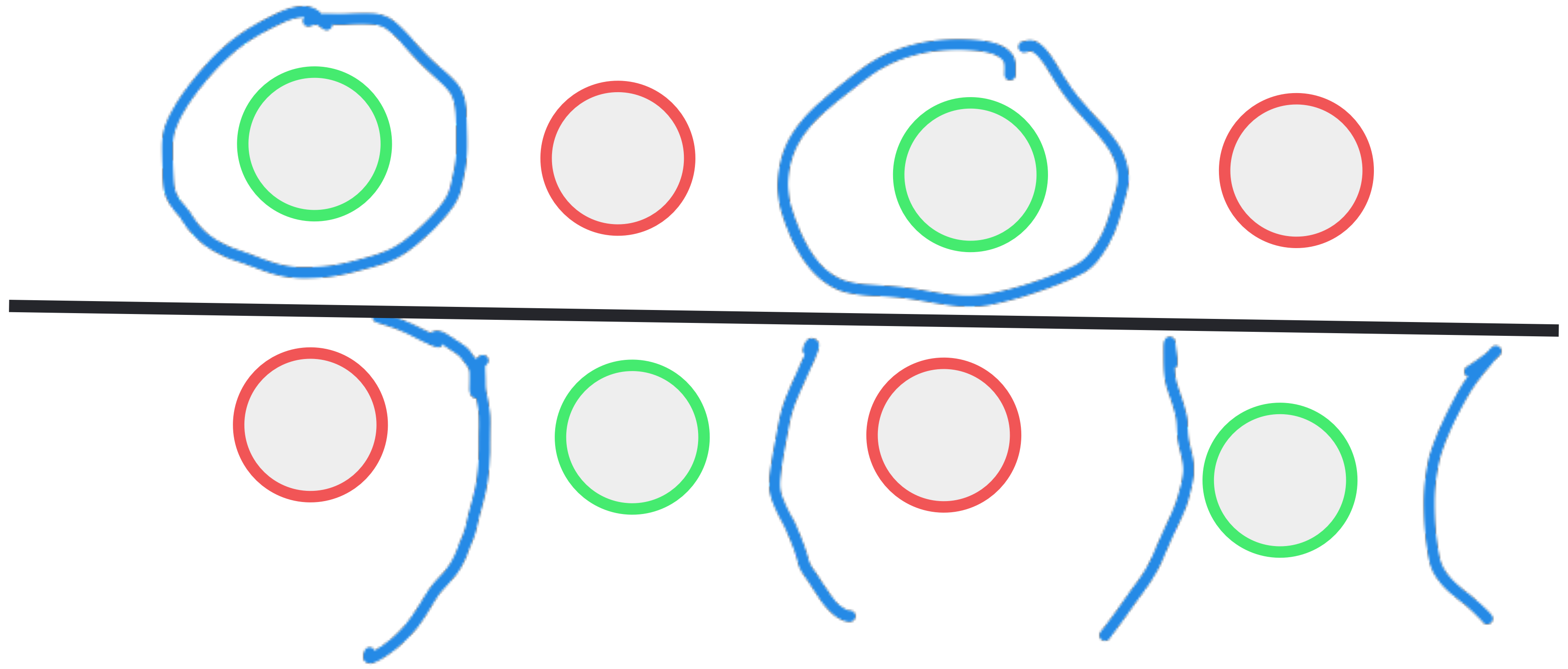


curvedness is indicative of the degree of the polynomial (p)

KERNEL TRICK

- Common Kernels
  - linear: $K(\vec{u},\vec{v}) = \vec{u}\cdot\vec{v} + b$ (line)
  - quadratic: $K(\vec{u},\vec{v}) = (\vec{u}\cdot\vec{v} + b)^2$ (conic section)
  - polynomial: $K(\vec{u},\vec{v}) = (\vec{u}\cdot\vec{v} + b)^p$
  - radial basis function (RBF): $K(\vec{u},\vec{v}) = \exp\left(-\frac{1}{\sigma^2}\|\vec{u}-\vec{v}\|^2\right)$

Draws circles around points



can classify any dataset

SVM SUMMARY

1) For points on the boundary: $\vec{w} \cdot \vec{x} + b = 0$

2) Margin width (m): $m = \dfrac{2}{\|\vec{w}\|}$

3) Gutter constraint: $\vec{w} \cdot \vec{x} + b = \pm 1$

Points not on gutter: $\text{sign}(\vec{w} \cdot \vec{x} + b) = y$

4) $\displaystyle\sum_{\text{support vector } i} \alpha_i \cdot y_i = 0 \iff \sum_{+\text{ support vector}} \alpha_+ = \sum_{-\text{ support vector}} \alpha_-$

5) $\displaystyle\vec{w} = \sum_{\text{support vector } i} \alpha_i y_i \vec{x}_i \iff \vec{w} = \sum_{+\text{ support vector}} \alpha_+ \vec{x}_+ - \sum_{-\text{ support vector}} \alpha_- \vec{x}_-$