

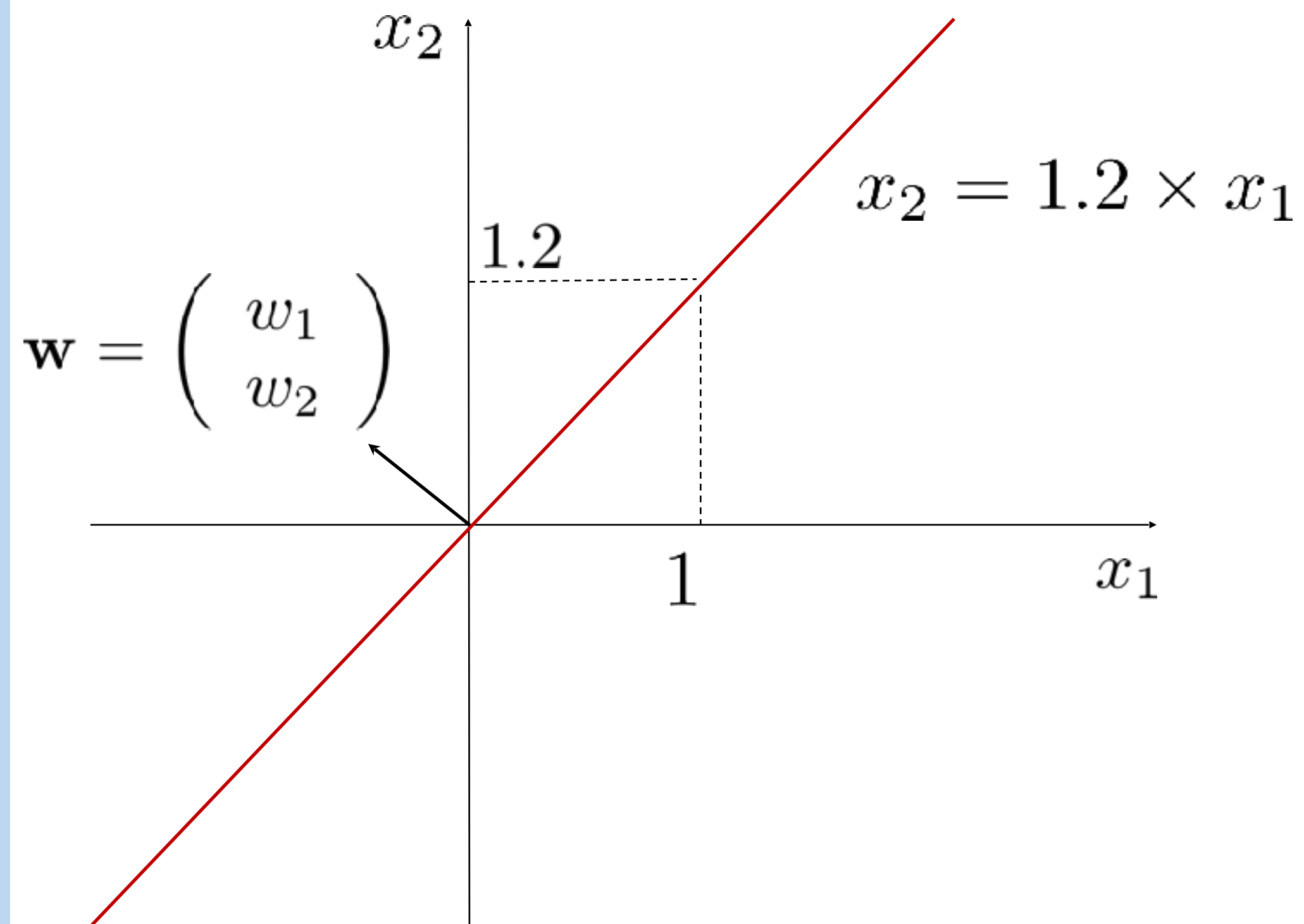
Lecture 9 pre-video

Decision boundaries

Orienting yourself



Line and vector
an example:

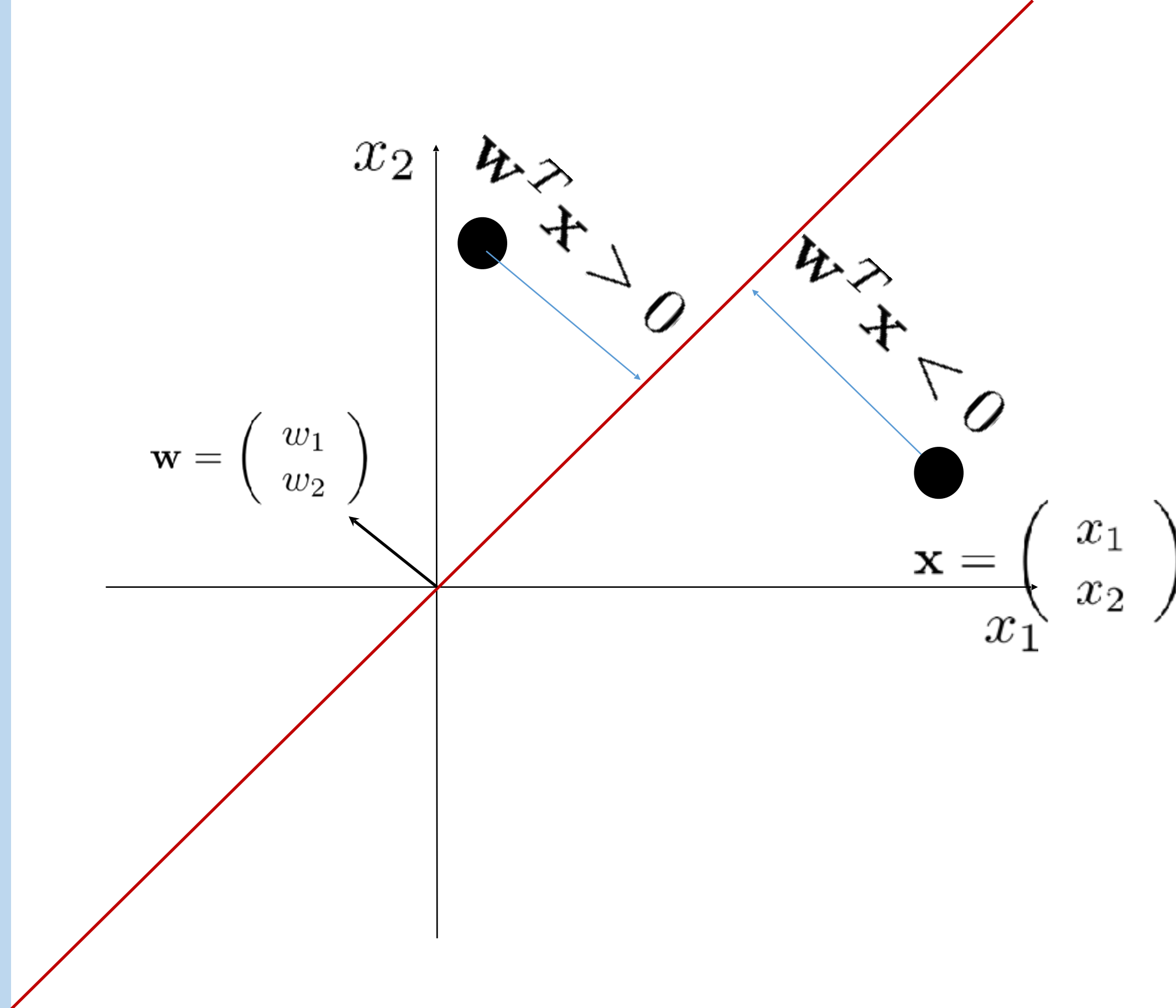


\mathbf{w} is the **normal** direction of the line

Often: $\|\mathbf{w}\|_2 = 1$: a unit vector

$$\mathbf{w} = \begin{bmatrix} \frac{-1}{\sqrt{2.44}} \\ \frac{1.2}{\sqrt{2.44}} \end{bmatrix}$$

Distance to the decision boundary



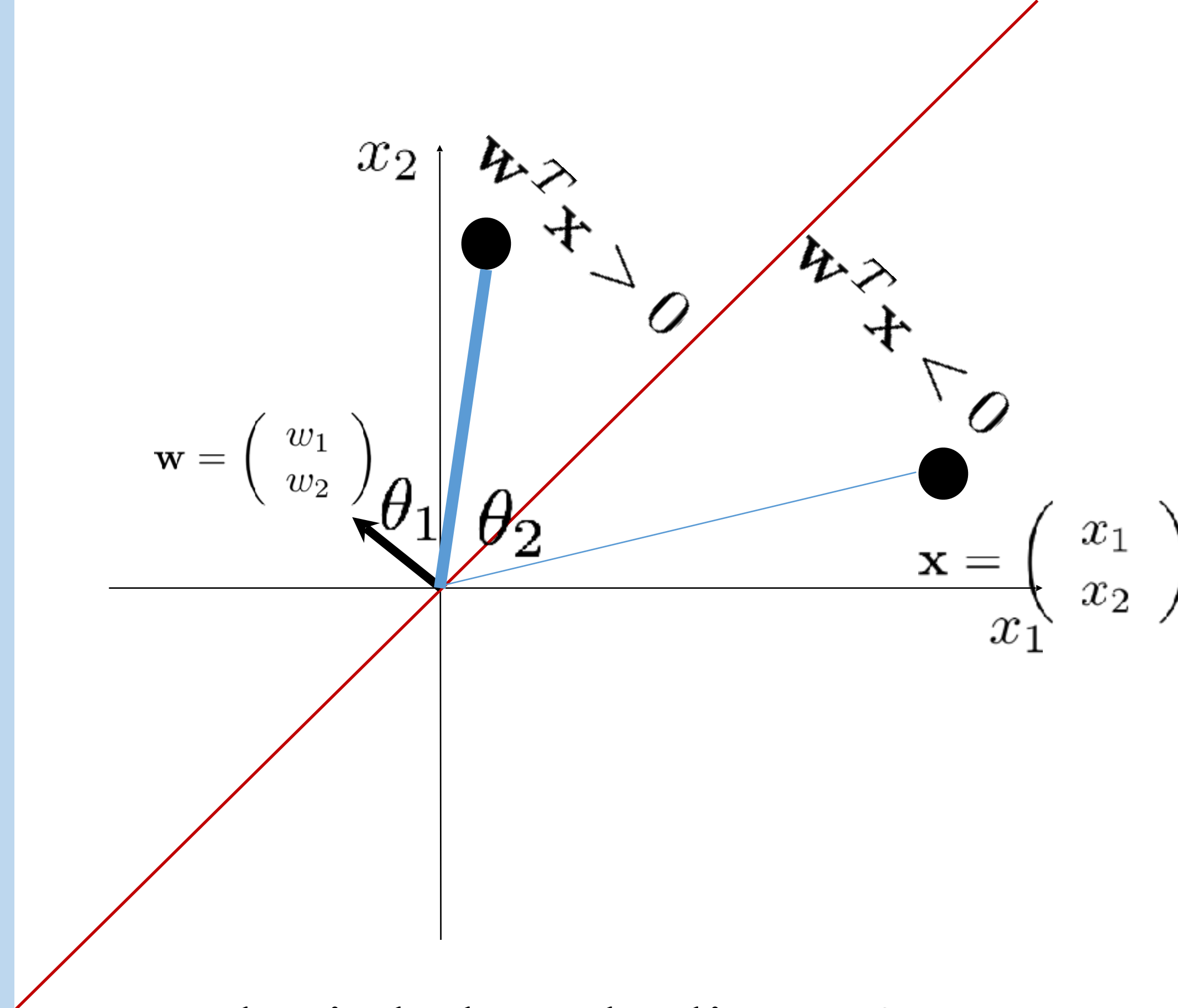
The distance (signed) of any point \mathbf{x} to the line is:

$$\mathbf{w}^T \mathbf{x} \equiv \langle \mathbf{w}, \mathbf{x} \rangle$$

$\mathbf{w}^T \mathbf{x} > 0$: above the line

$\mathbf{w}^T \mathbf{x} < 0$: below the line

Distance to the decision boundary

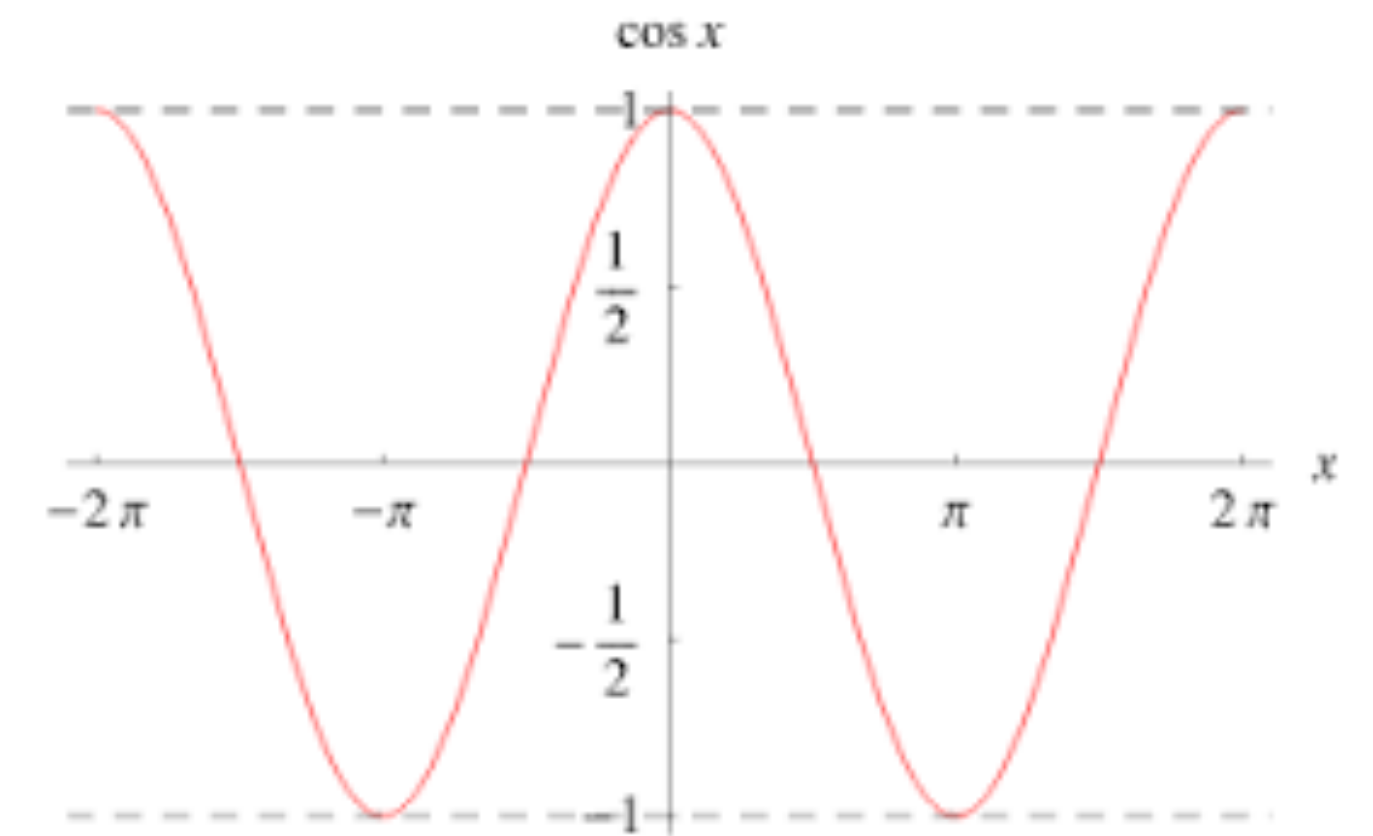


Why is below the line < 0
while above the line is > 0 ?

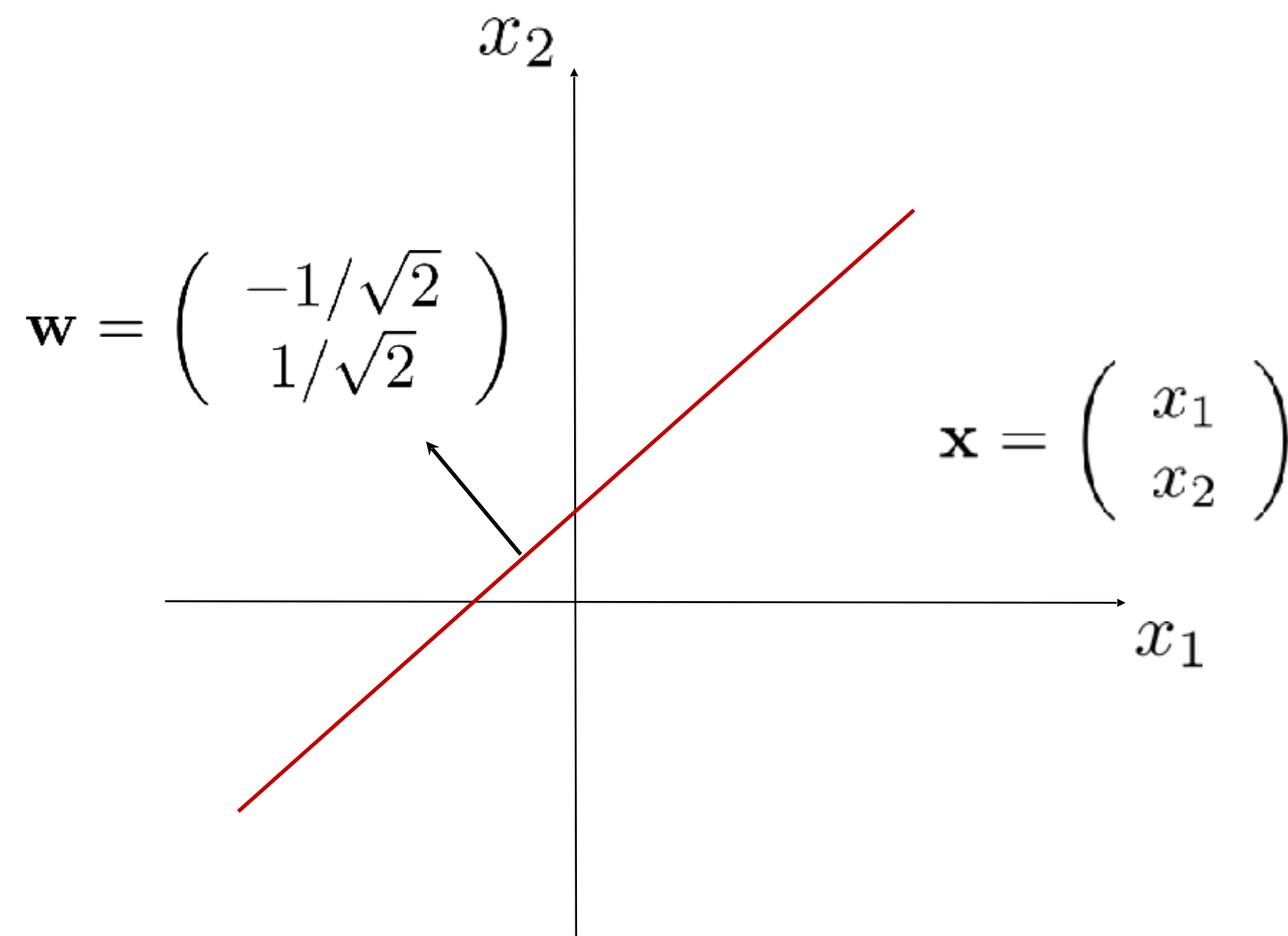
$$\mathbf{w}^T \mathbf{x} \equiv \langle \mathbf{w}, \mathbf{x} \rangle$$

$$\mathbf{w}^T \mathbf{x} = \|\mathbf{w}\| \cdot \|\mathbf{x}\| \cdot \cos \theta$$

It depends on the angle
formed by \mathbf{w} and \mathbf{x}



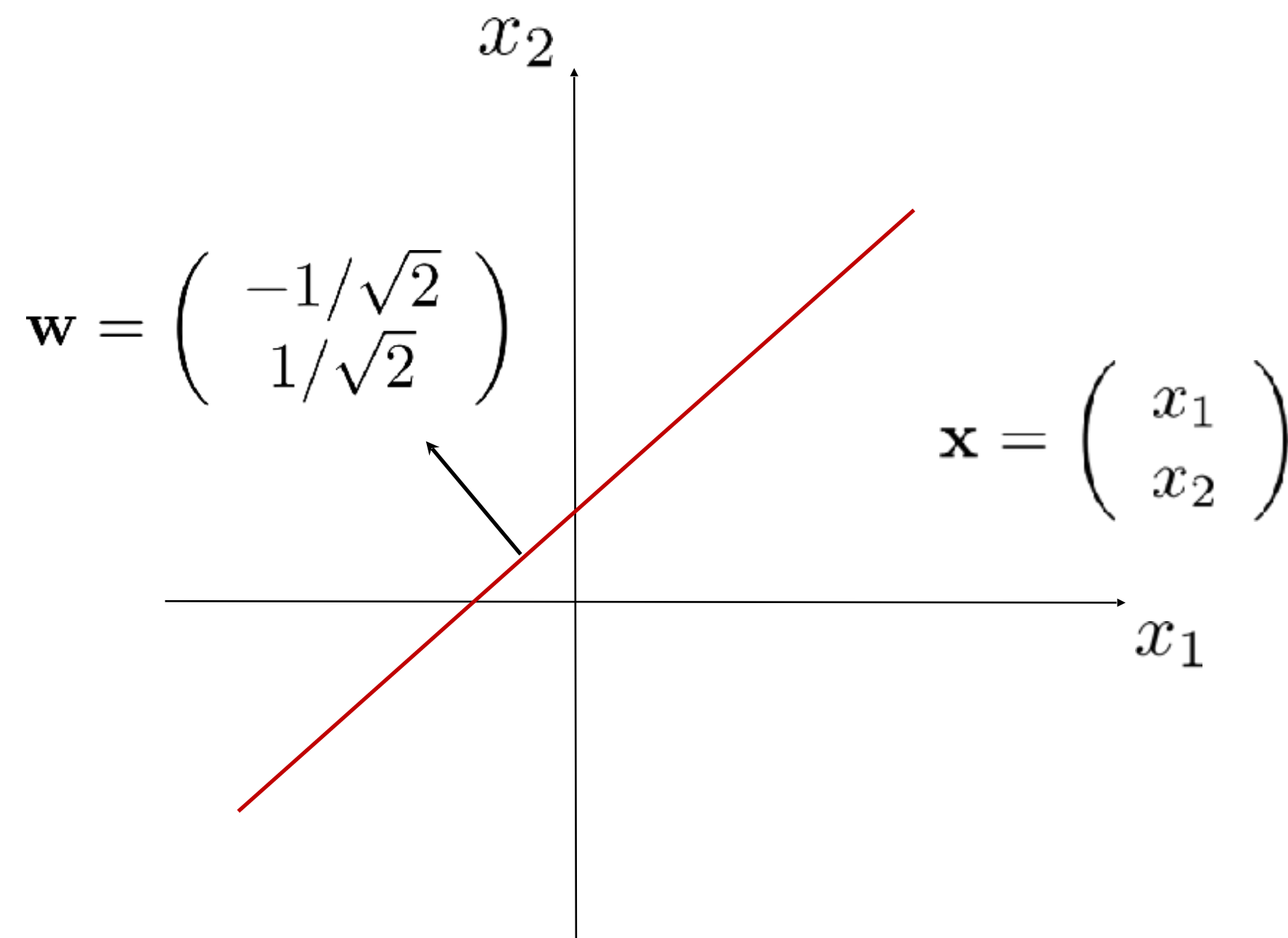
Can the line in red be the decision boundary of the classifier \mathbf{w} shown to the right?



$$\mathbf{w} = \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} \quad y = \begin{cases} +1 & \text{if } \mathbf{w}^T \mathbf{x} > 0 \\ -1 & \text{if } \mathbf{w}^T \mathbf{x} < 0 \end{cases}$$

- A. Yes
- B. No
- C. It depends

Can the line in red be the decision boundary of the classifier w shown to the right?



$$\mathbf{w} = \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} \quad y = \begin{cases} +1 & \text{if } \mathbf{w}^T \mathbf{x} > 0 \\ -1 & \text{if } \mathbf{w}^T \mathbf{x} < 0 \end{cases}$$

A. Yes



B. No

C. It depends

Decision boundary

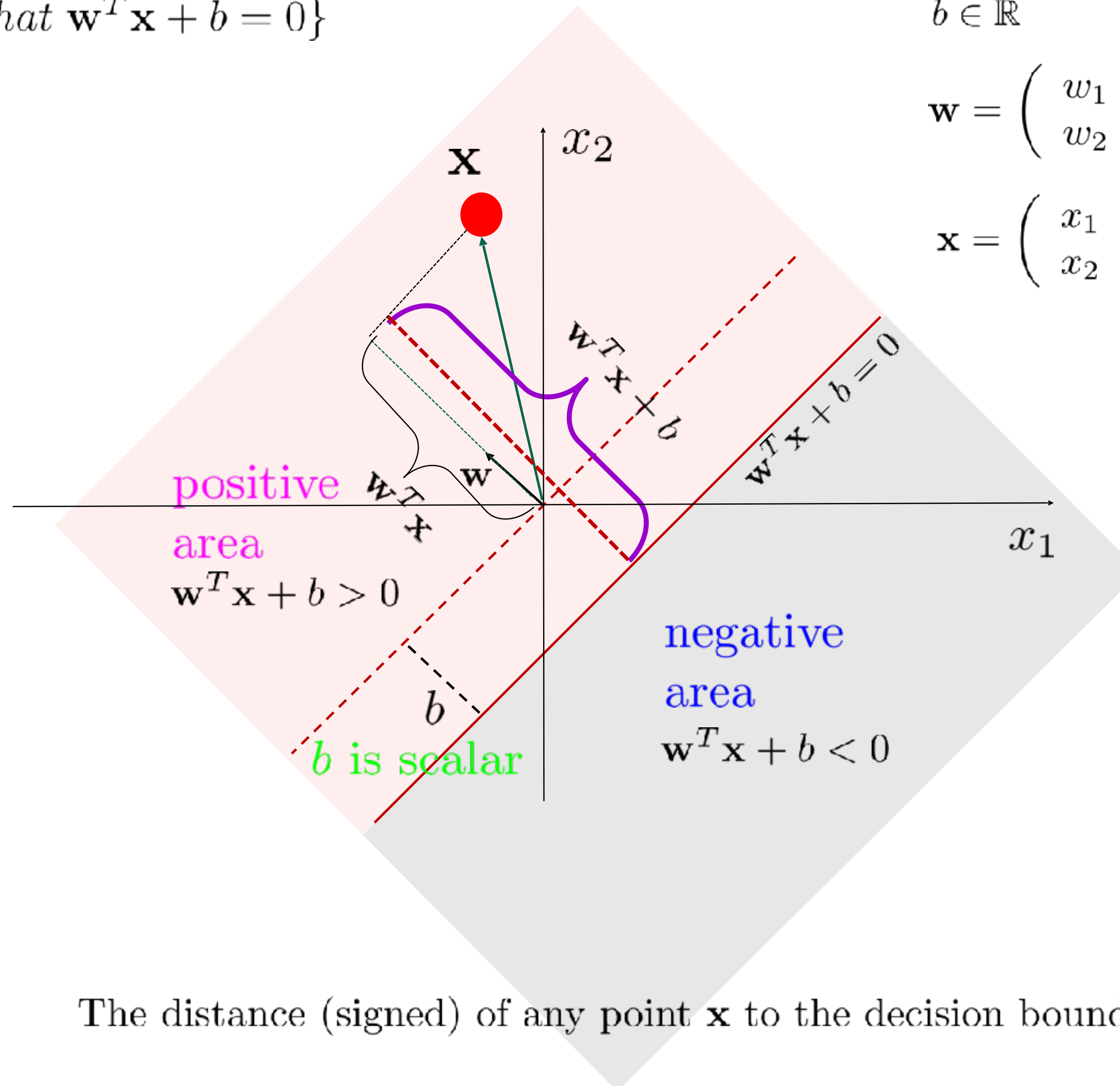
Decision boundary:

$$\{\mathbf{x}; \forall \mathbf{x} \text{ such that } \mathbf{w}^T \mathbf{x} + b = 0\}$$

$$b \in \mathbb{R}$$

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$



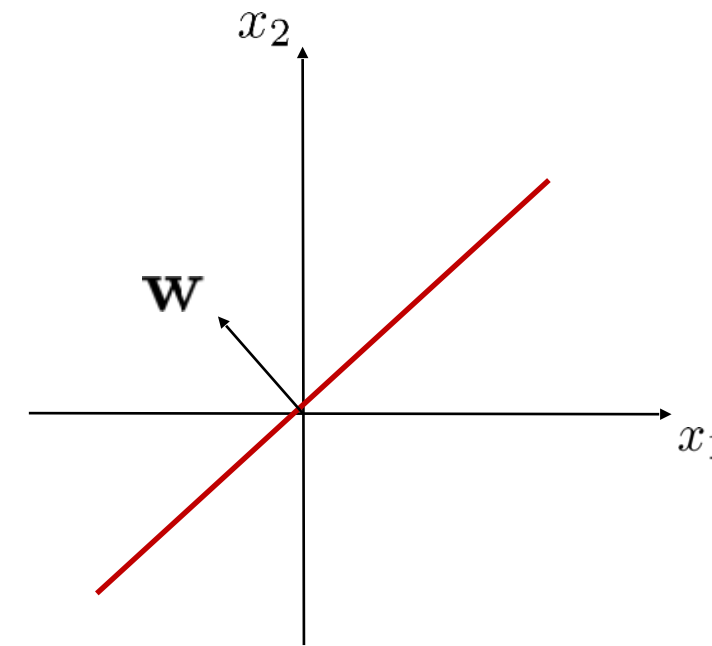
The distance (signed) of any point \mathbf{x} to the decision boundary is: $\mathbf{w}^T \mathbf{x} + b$

Decision boundary

When $b=0$:

$$\{\mathbf{x}; \forall \mathbf{x} \text{ such that } \mathbf{w}^T \mathbf{x} = 0\}$$

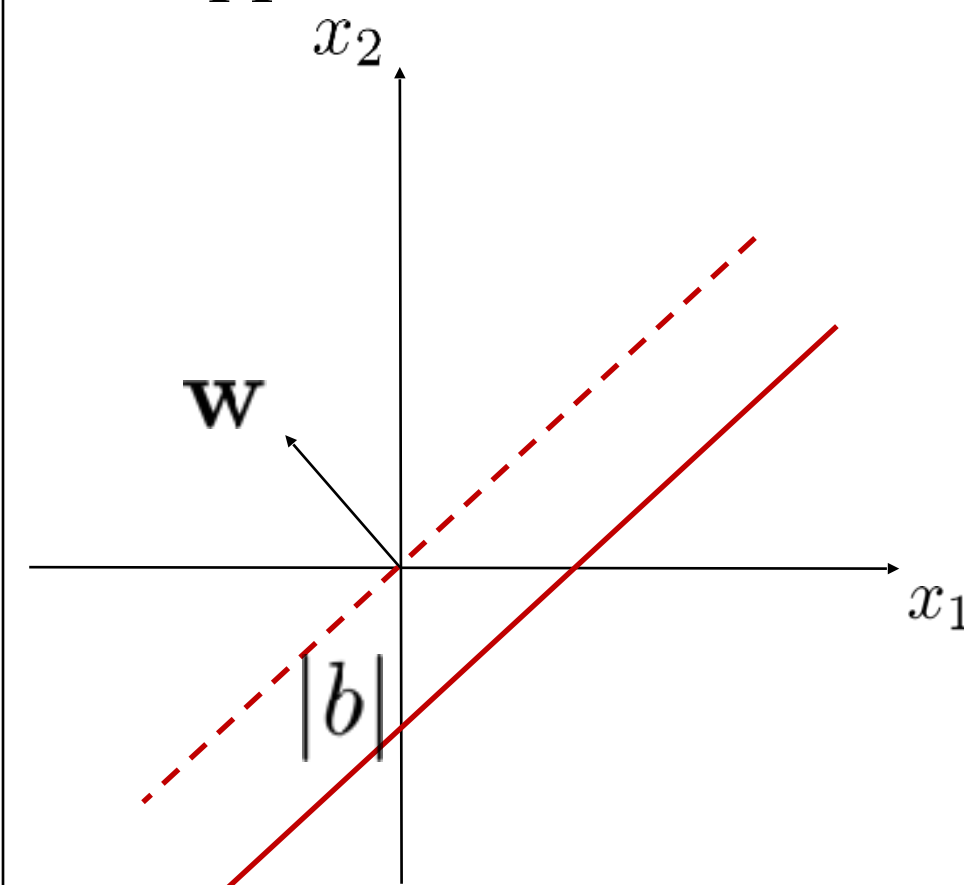
The decision boundary always goes through the origin.



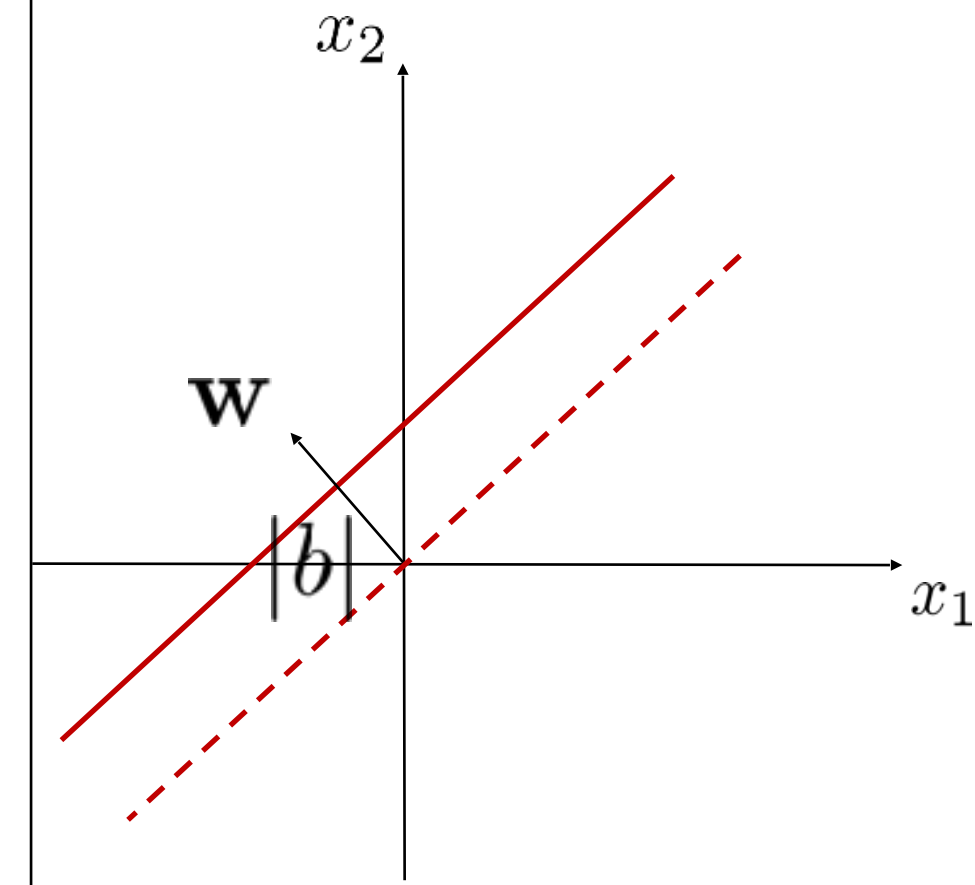
Decision boundary:

$$\{\mathbf{x}; \forall \mathbf{x} \text{ such that } \mathbf{w}^T \mathbf{x} + b = 0\}$$

When $b > 0$ the decision boundary is moved along the opposite direction of w .



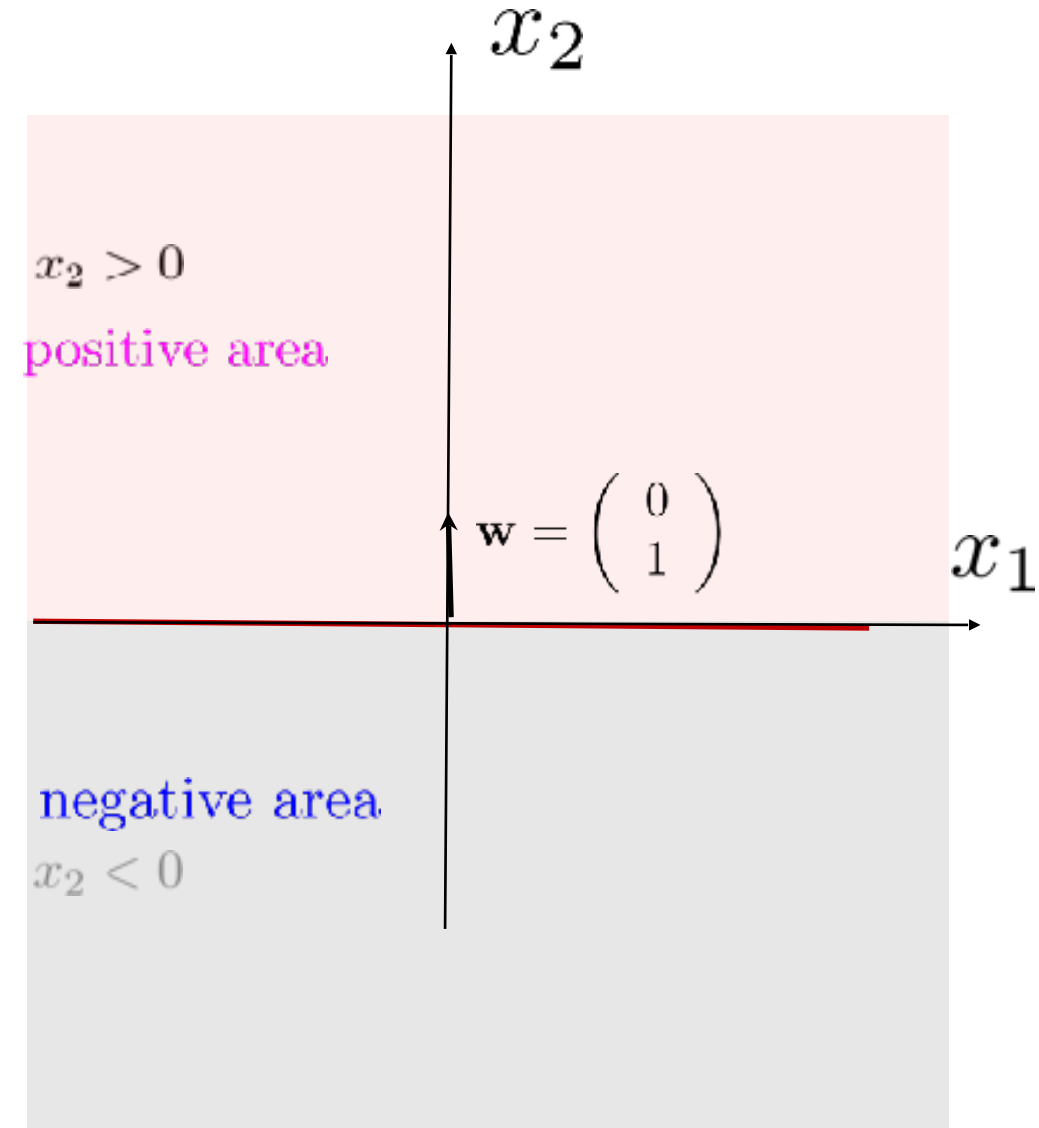
When $b < 0$ the decision boundary is moved along the same direction of w .



Some typical examples

Assuming \mathbf{w} being normalized: $\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2} = 1$.

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \quad b \in \mathbb{R} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$



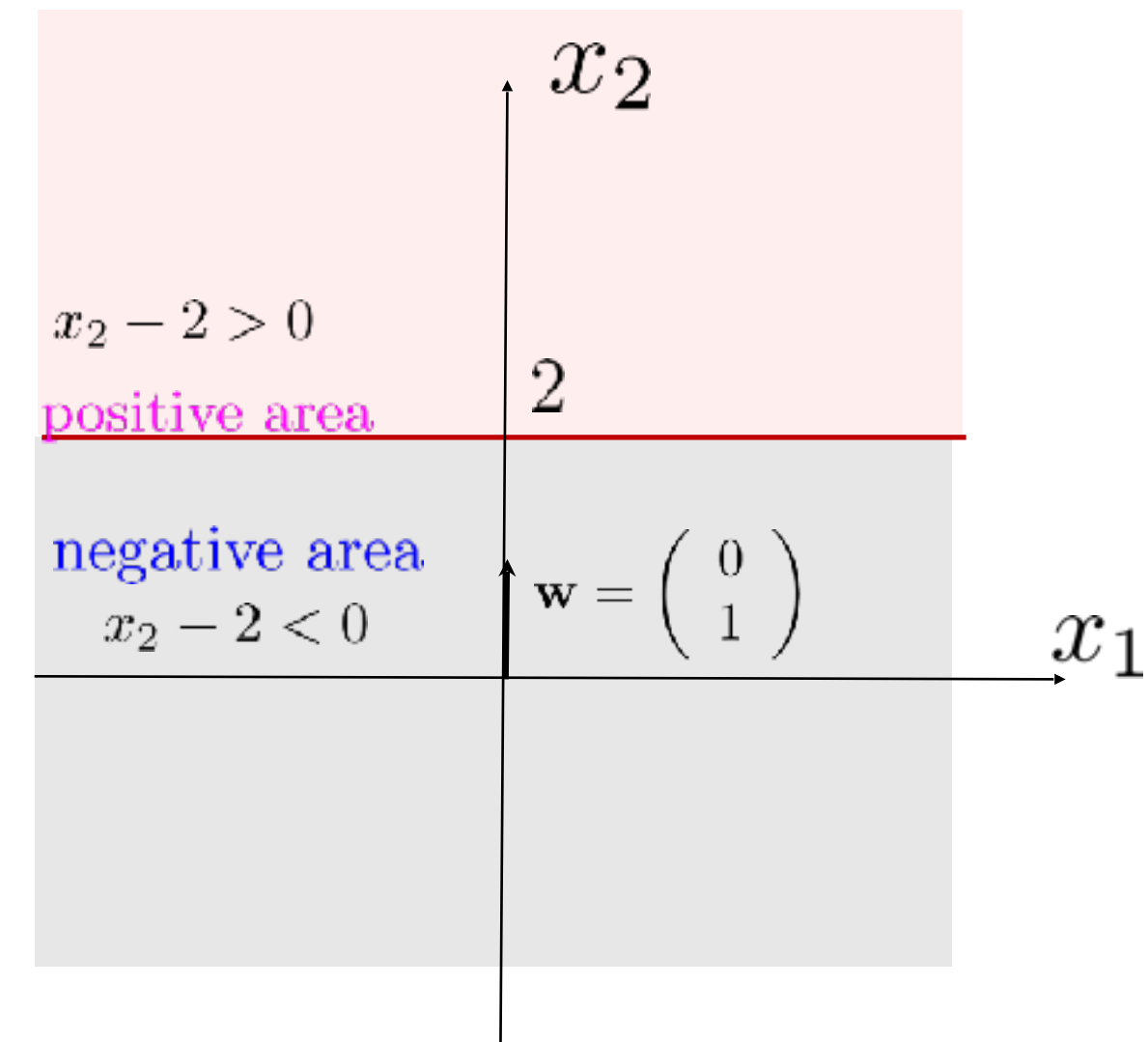
$$\mathbf{w} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$b = 0$$

Decision boundary:

$$0 \times x_1 + 1 \times x_2 = 0$$

$$\Downarrow \\ x_2 = 0$$



$$\mathbf{w} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$b = -2$$

Decision boundary:

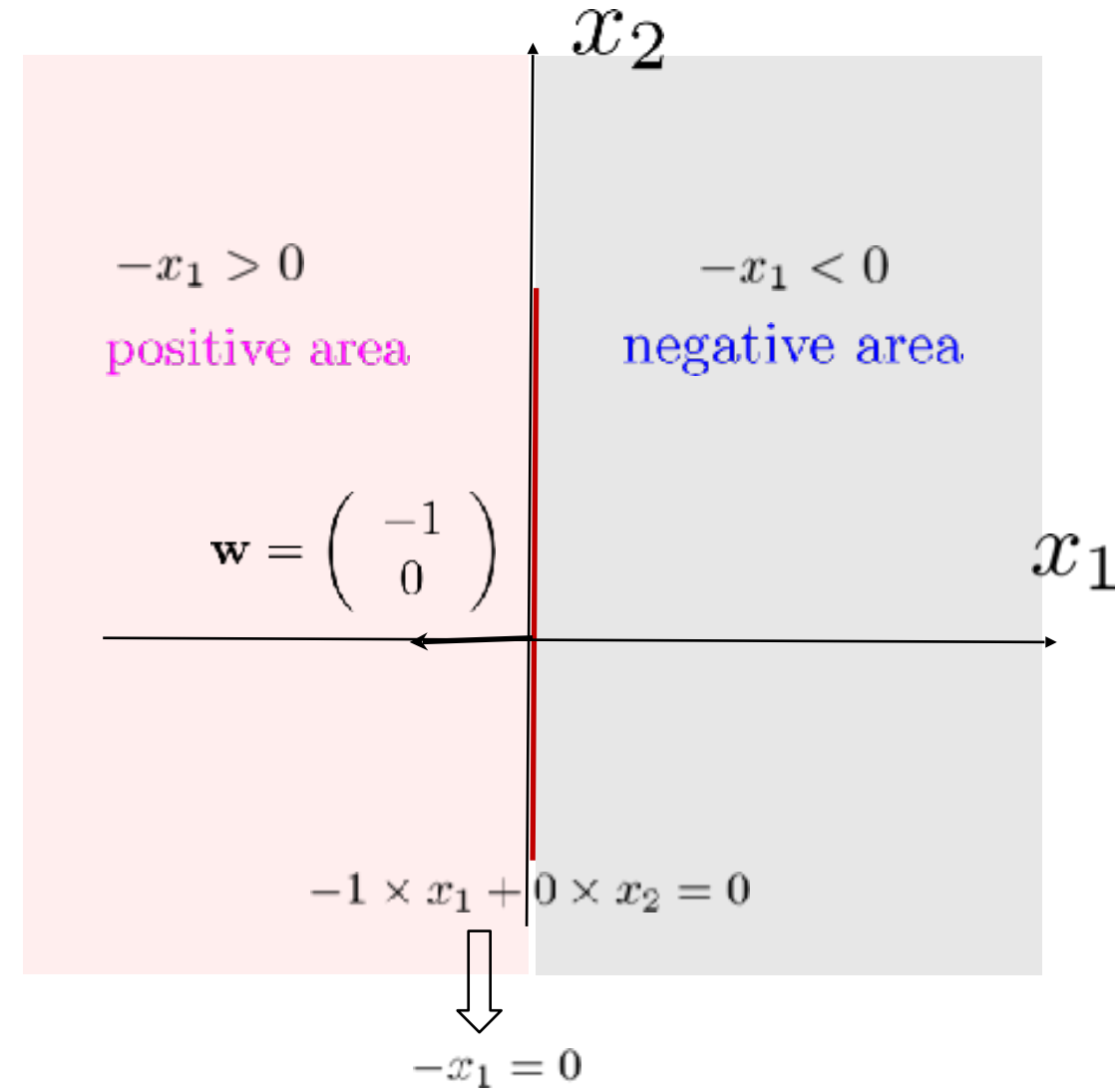
$$0 \times x_1 + 1 \times x_2 - 2 = 0$$

$$\Downarrow \\ x_2 - 2 = 0$$

Some typical examples

Assuming \mathbf{w} being normalized: $\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2} = 1$.

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \quad b \in \mathbb{R} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$



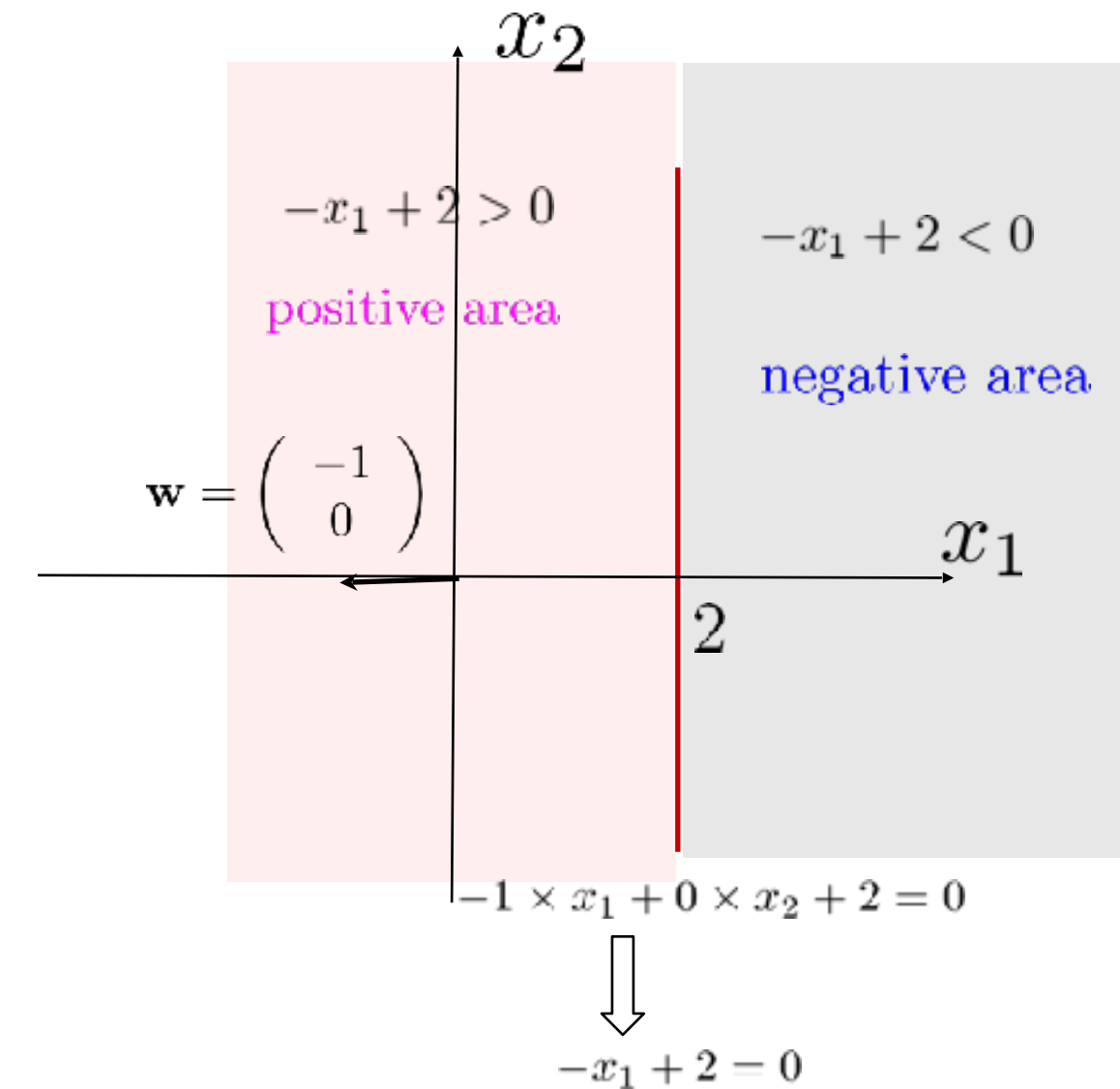
$$\mathbf{w} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$b = 0$$

Decision boundary:

$$-1 \times x_1 + 0 \times x_2 = 0$$

$$\Downarrow$$
$$-x_1 = 0$$



$$\mathbf{w} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$b = 2$$

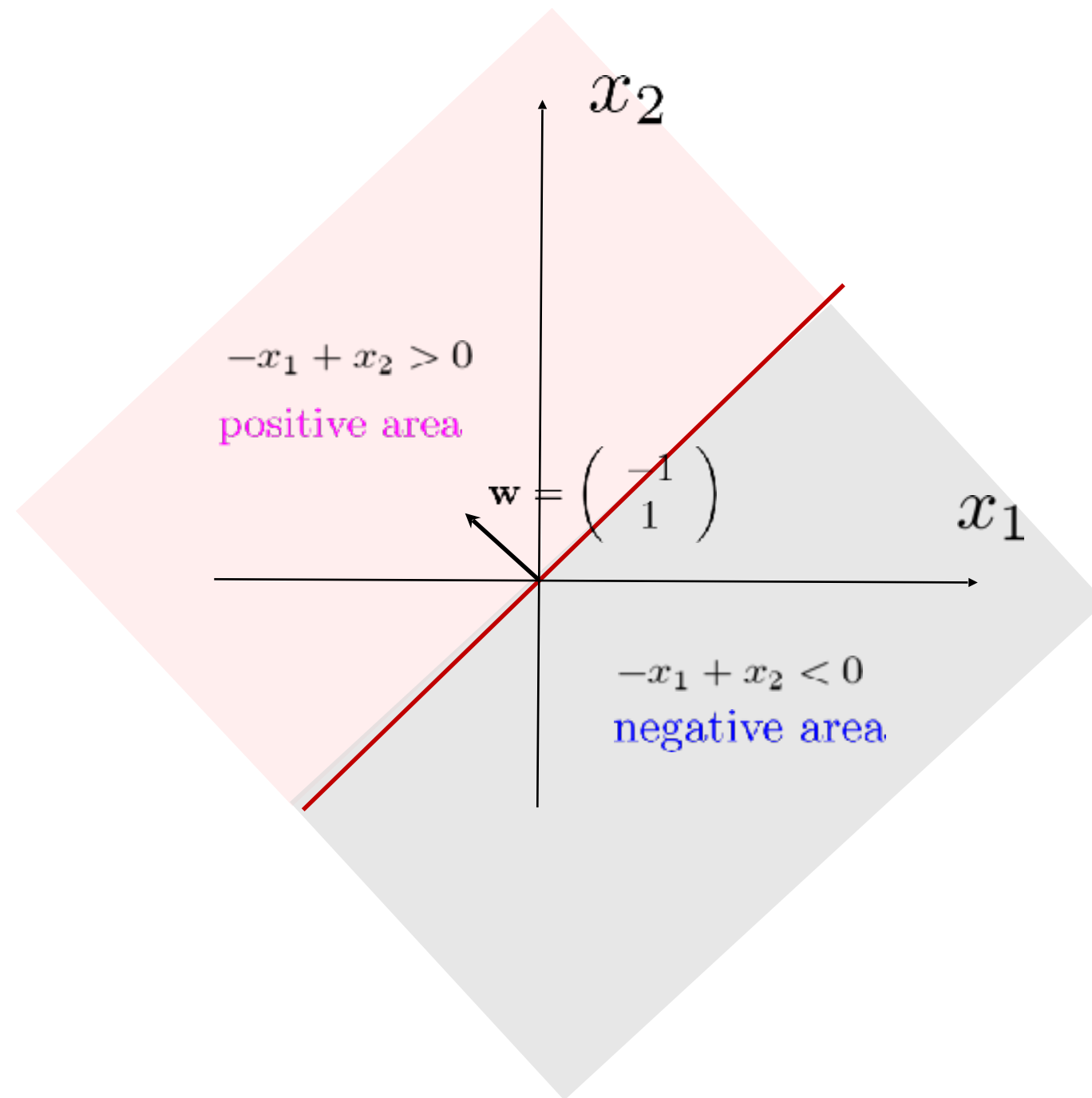
Decision boundary:

$$-1 \times x_1 + 0 \times x_2 + 2 = 0$$

$$\Downarrow$$
$$-x_1 + 2 = 0$$

Some typical examples

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \quad b \in \mathbb{R} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

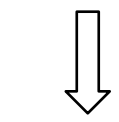


$$\mathbf{w} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

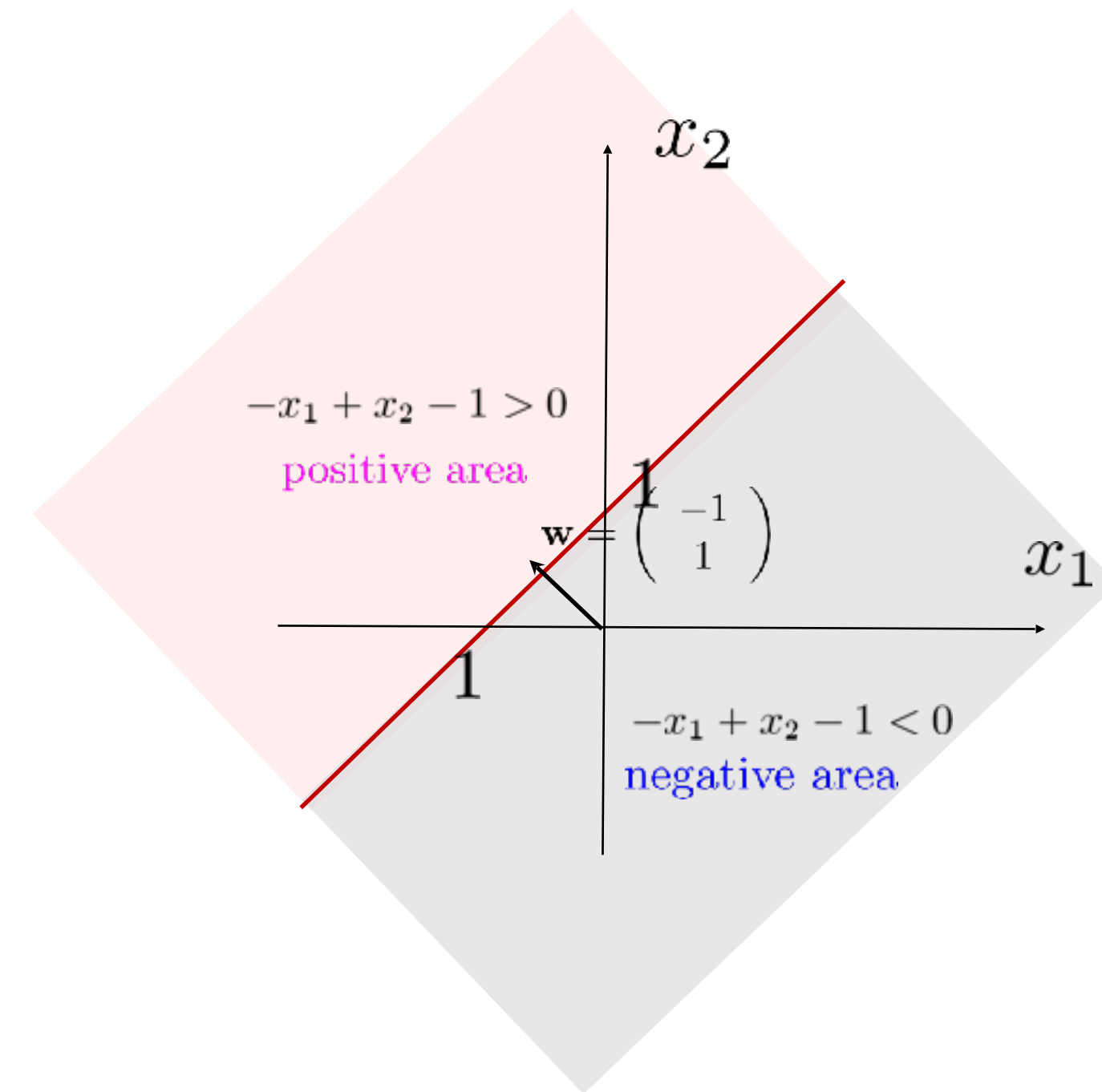
$$b = 0$$

Decision boundary:

$$-1 \times x_1 + 1 \times x_2 = 0$$



$$-x_1 + x_2 = 0$$



$$\mathbf{w} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$b = -1$$

Decision boundary:

$$-1 \times x_1 + 1 \times x_2 - 1 = 0$$



$$-x_1 + x_2 - 1 = 0$$

Take home message

Any data sample (point) lying on the decision boundary receives a classification decision that is **equally positive and negative**.

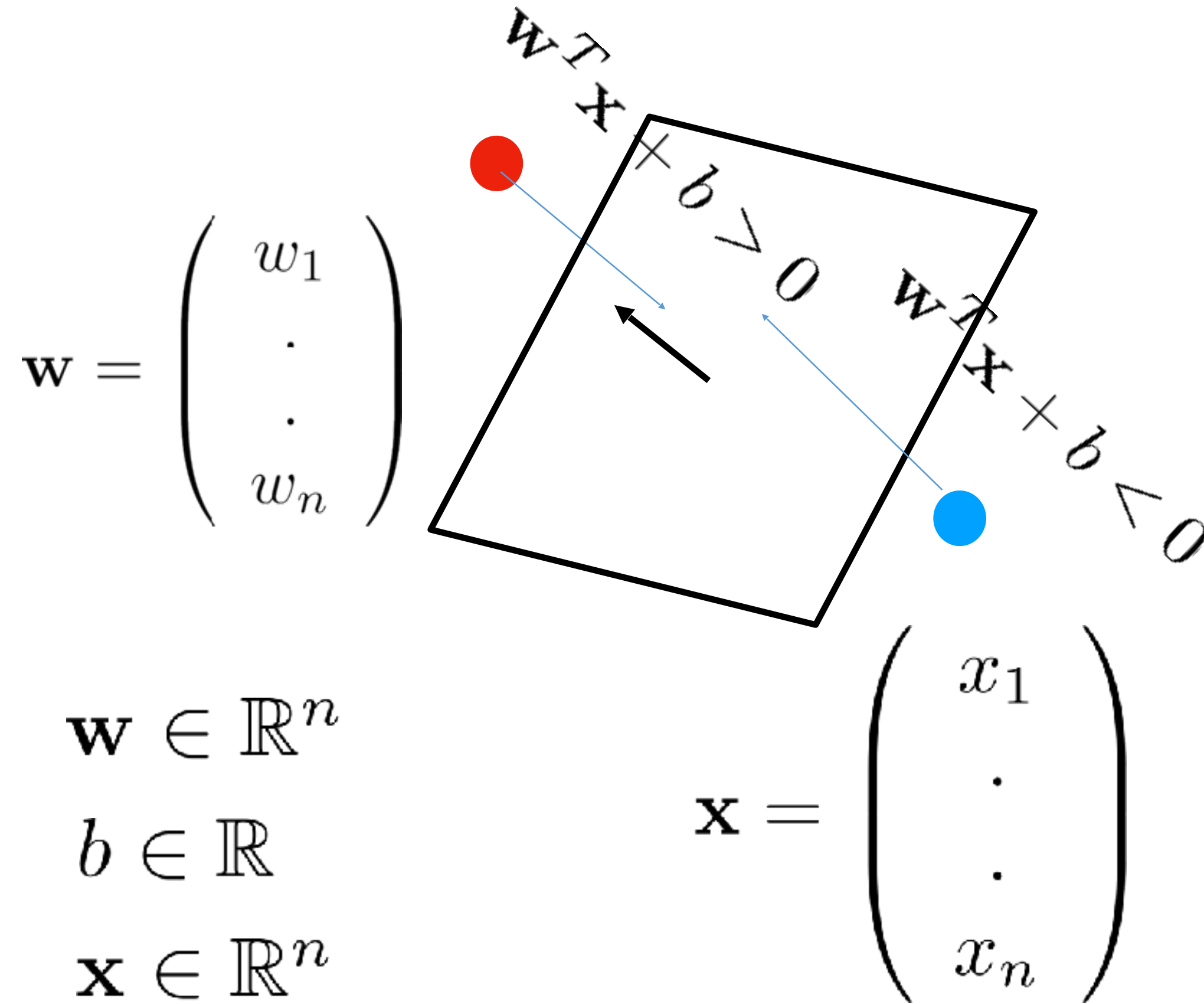
The decision boundary of a linear classifier is a **hyper-plane**.

The **model parameter** \mathbf{w} is along the **normal** direction of the decision boundary, pointing to the **positive** samples.

The bias terms, b (scalar), refers to as the **translation** (shift) of the decision boundary.

Distance to the decision boundary

in an arbitrary vector space

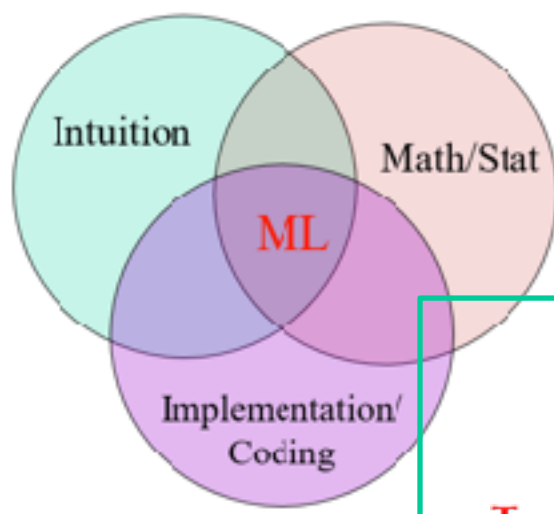


The distance (signed) of any point \mathbf{x} to the hyper-plane is:

$$\mathbf{w}^T \mathbf{x} + b \equiv \langle \mathbf{w}, \mathbf{x} \rangle + b \equiv \mathbf{w} \cdot \mathbf{x} + b$$

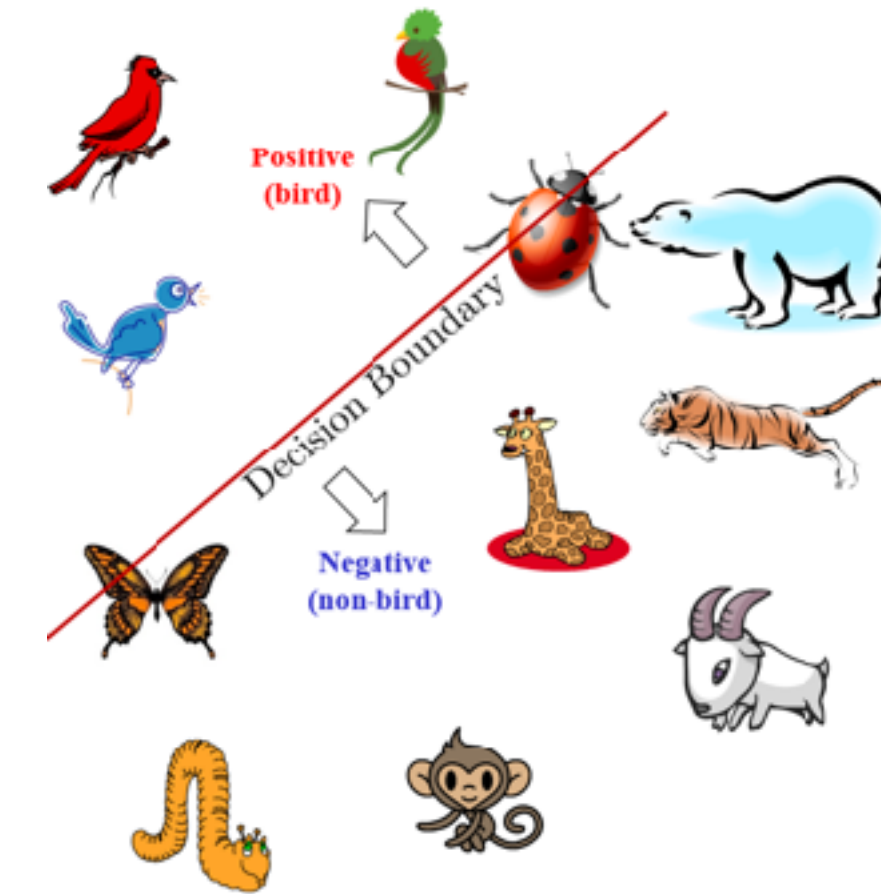
$\mathbf{w}^T \mathbf{x} + b < 0$: below the hyper-plane

$\mathbf{w}^T \mathbf{x} + b > 0$: above the hyper-plane



Recap: Decision Boundary

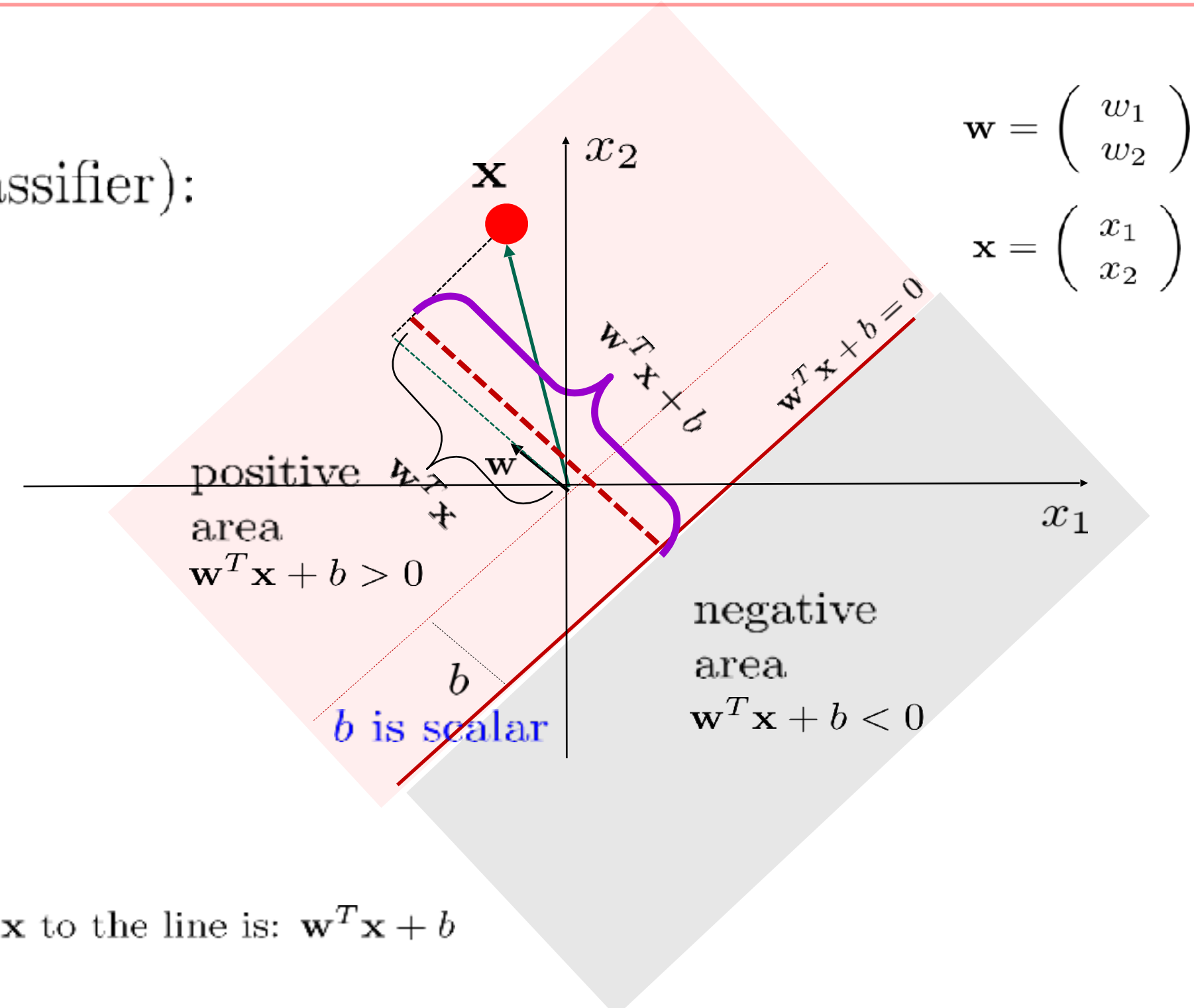
Intuition: Decision boundary of a discriminative classifier is a **set** that consists of possible samples that are on the **border** (typically 50% – 50%) of the separation between the positive and negative areas (for two-class classification).



Math:

Decision boundary (for a linear classifier):

$$\{\mathbf{x}; \forall \mathbf{x} \text{ such that } \mathbf{w}^T \mathbf{x} + b = 0\}$$



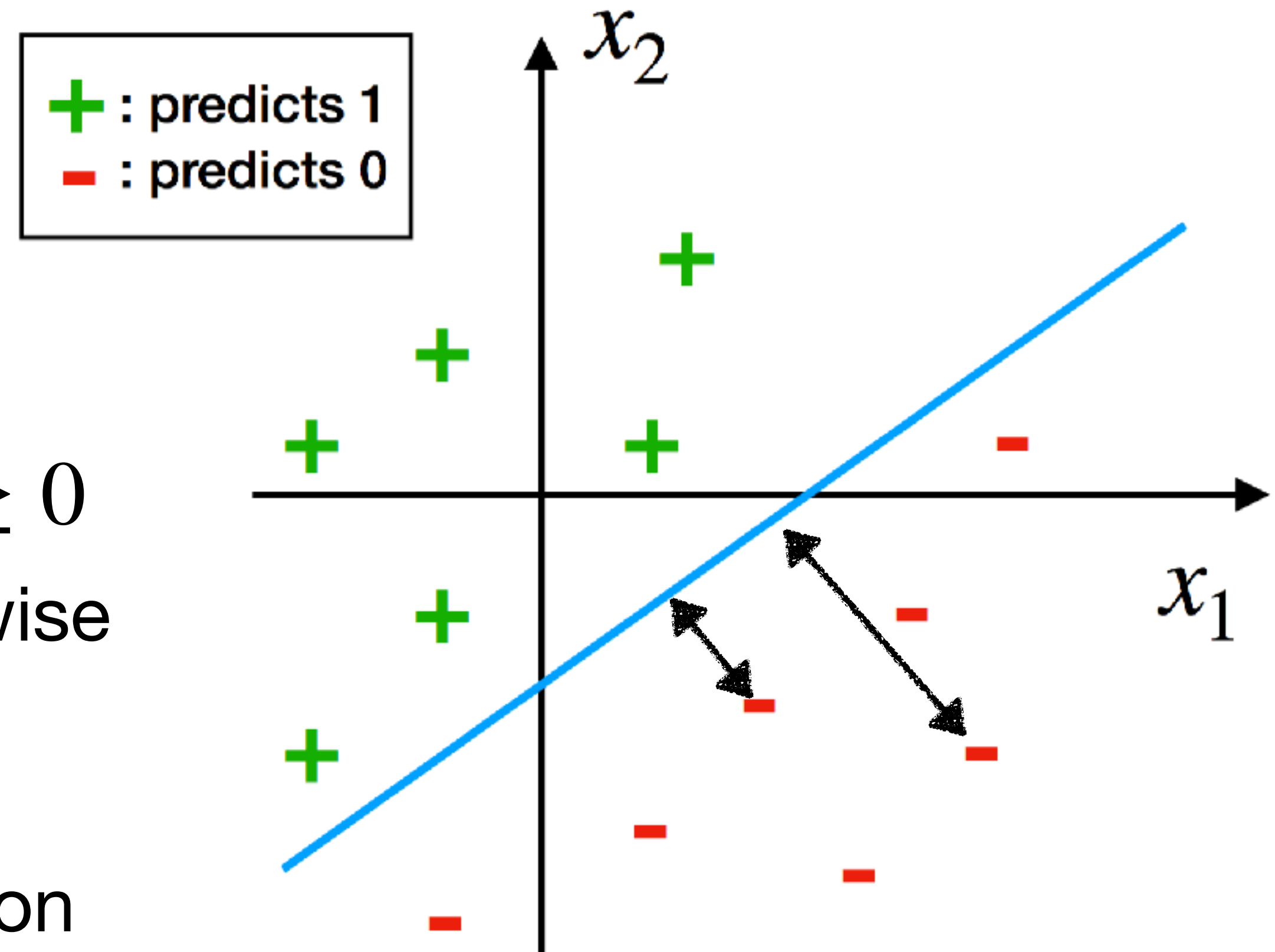
The distance (signed) of any point \mathbf{x} to the line is: $\mathbf{w}^T \mathbf{x} + b$

Lecture 9 pre-video

Logit + its derivatives

Distance as Probability

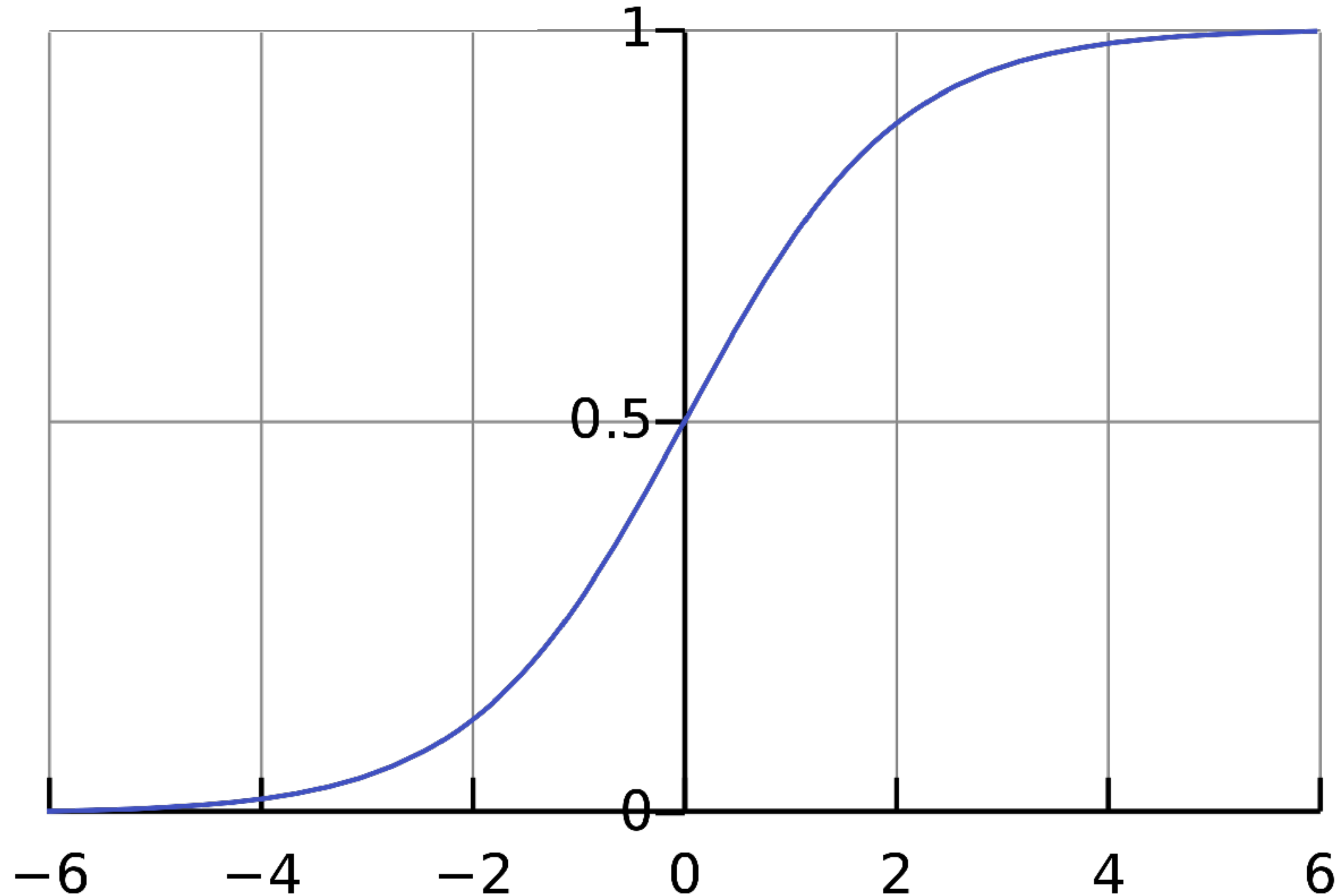
- Classification $f(x; w) = \begin{cases} 1 & \text{if } w^T x \geq 0 \\ -1 & \text{otherwise} \end{cases}$
- Our predictions are only +1 or -1
- What if we want to make our prediction a probability?
- $f(x; w) = p(y = +1 | x; w)$
or equivalently
 $f(x; w) = -p(y = -1 | x; w)$



Logistic function a.k.a. logit

$$f(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

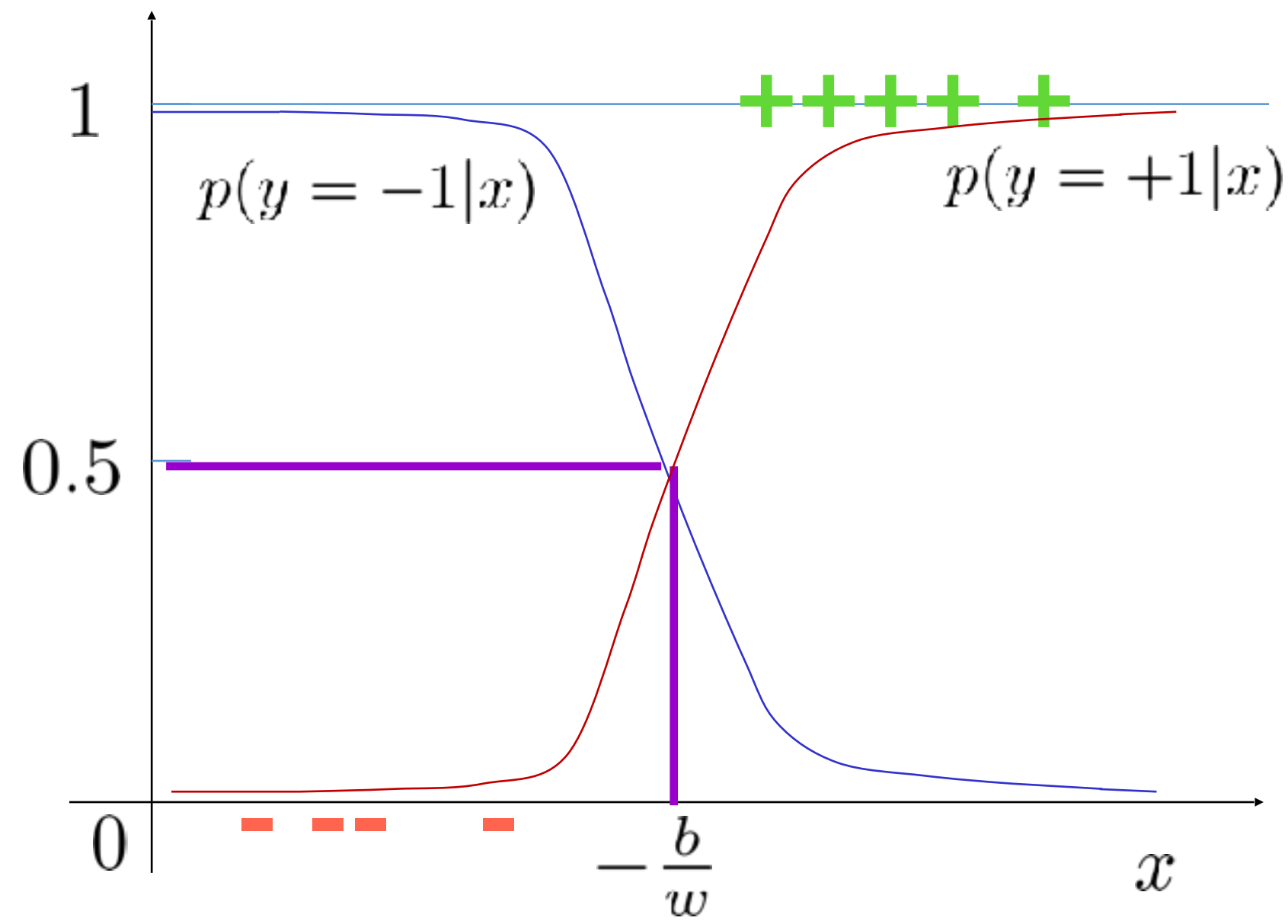
Always in range 0 - 1 => probability



Logistic regression classifier

$x, w, b \in \mathbb{R}$

Let's look at the simplest case where x is a scalar:



We have:
$$f(x; w, b) = \begin{cases} +1 & \text{if } w \times x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}.$$

Probability of a sample being a positive case

$$p(y = +1|x) = \frac{e^{w \times x + b}}{1 + e^{w \times x + b}}$$

Probability of a sample being a negative case

$$p(y = -1|x) = \frac{1}{1 + e^{w \times x + b}}$$

$$\begin{aligned} p(y = -1|x) &= 1 - p(y = +1|x) \\ &= 1 - \frac{e^{w \times x + b}}{1 + e^{w \times x + b}} \\ &= \frac{1 + e^{w \times x + b}}{1 + e^{w \times x + b}} - \frac{e^{w \times x + b}}{1 + e^{w \times x + b}} \end{aligned}$$

The main mathematical
convenience of the logistic
regression function!

Logistic regression function

$$\begin{aligned} p(y=+1|\mathbf{x}) &= \frac{e^{\mathbf{w}\mathbf{x}+b}}{1+e^{\mathbf{w}\mathbf{x}+b}} \\ &= \frac{e^{\mathbf{w}\mathbf{x}+b}}{1+e^{\mathbf{w}\mathbf{x}+b}} \cdot \frac{e^{-(\mathbf{w}\mathbf{x}+b)}}{e^{-(\mathbf{w}\mathbf{x}+b)}} \\ &= \frac{1}{e^{-(\mathbf{w}\mathbf{x}+b)} + 1} \end{aligned}$$

$$p(y = +1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x}+b)}}$$

$$p(y = -1|\mathbf{x}) = \frac{1}{1+e^{(\mathbf{w}^T \mathbf{x}+b)}} \quad y \in \{-1, +1\}$$

$$p(y|\mathbf{x}) = \frac{1}{1+e^{-y(\mathbf{w}^T \mathbf{x}+b)}}$$

A general form, independent of the value of y!

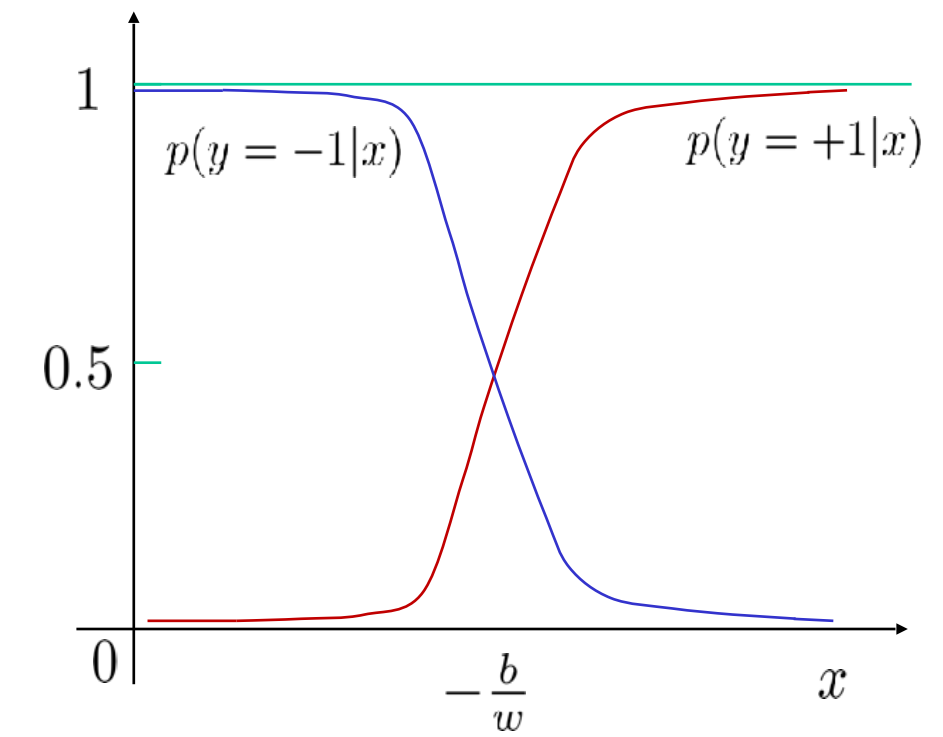
Training a logistic regression classifier

$$S_{training} = \{(-1.1, -1), (3.2, +1), (2.5, -1), (5.0, +1), (4.3, +1)\}$$

$$p(y = +1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x}+b)}}$$

$$p(y = -1|\mathbf{x}) = \frac{1}{1+e^{(\mathbf{w}^T \mathbf{x}+b)}}$$

$$p(y_i|\mathbf{x}_i) = \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i+b)}}$$



$$\text{Train a logistic regression classifier } f(\mathbf{x}) = \begin{cases} +1 & \text{if } \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x}+b)}} \geq 0.5 \\ -1 & \text{otherwise} \end{cases} :$$

Intuition: find the best parameters $(\mathbf{w}, b)^*$ to maximize the probabilities of fitting the ground-truth label y_i for each \mathbf{x}_i .

$$\text{Math: } (\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i+b)}}$$

Training a logistic regression classifier

Intuition: find the best parameters $(\mathbf{w}, b)^*$ to maximize the probabilities of fitting the ground-truth label y_i for each \mathbf{x}_i .

Math: $(\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$

$$\begin{aligned}(\mathbf{w}, b)^* &= \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \\&= \arg \max_{(\mathbf{w}, b)} \ln\left(\prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}\right) \\&= \arg \min_{(\mathbf{w}, b)} \sum_{i=1}^n -\ln\left(\frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}\right)\end{aligned}$$

Logistic regression

Jason G. Fleischer, Ph.D.

Asst. Teaching Professor

Department of Cognitive Science, UC San Diego

jfleischer@ucsd.edu



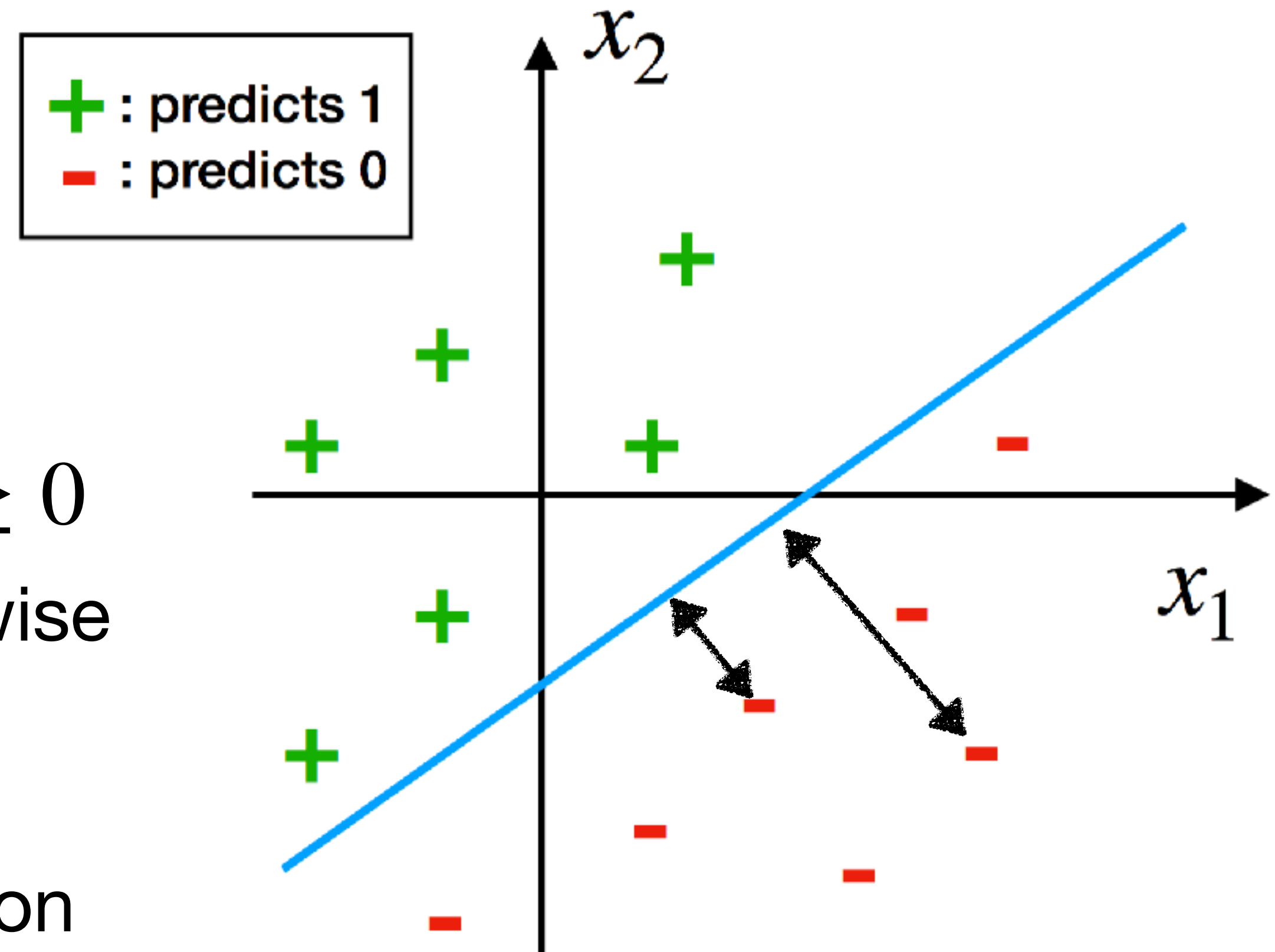
@jasongfleischer

<https://jgfleischer.com>

Slides in this presentation are from material kindly provided by
Zhuowen Tu and others credited at those slides

Distance as Probability

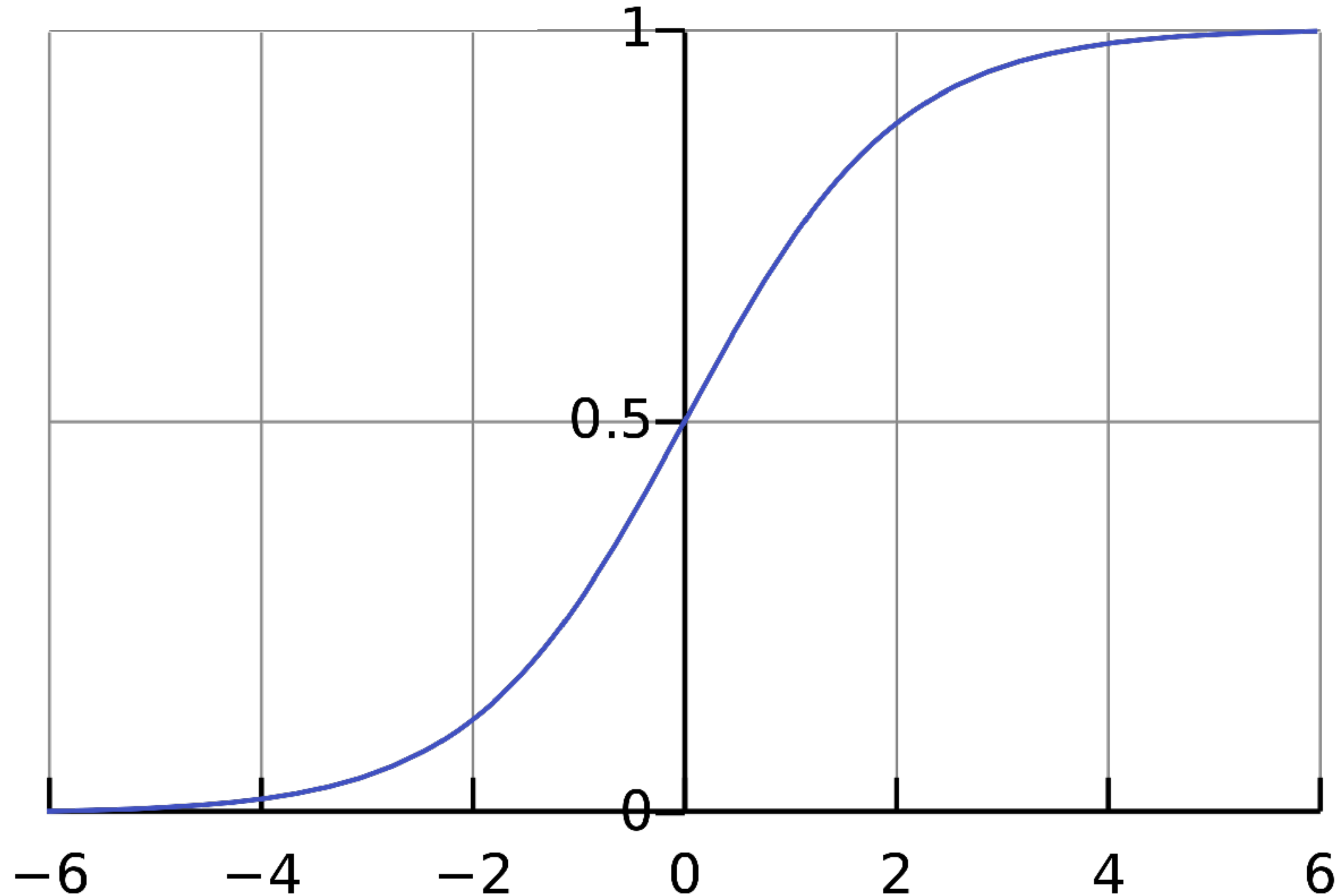
- Classification $f(x; w) = \begin{cases} 1 & \text{if } w^T x \geq 0 \\ -1 & \text{otherwise} \end{cases}$
- Our predictions are only +1 or -1
- What if we want to make our prediction a probability?
- $f(x; w) = p(y = +1 | x; w)$
or equivalently
 $f(x; w) = -p(y = -1 | x; w)$



Logistic function a.k.a. logit

$$f(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

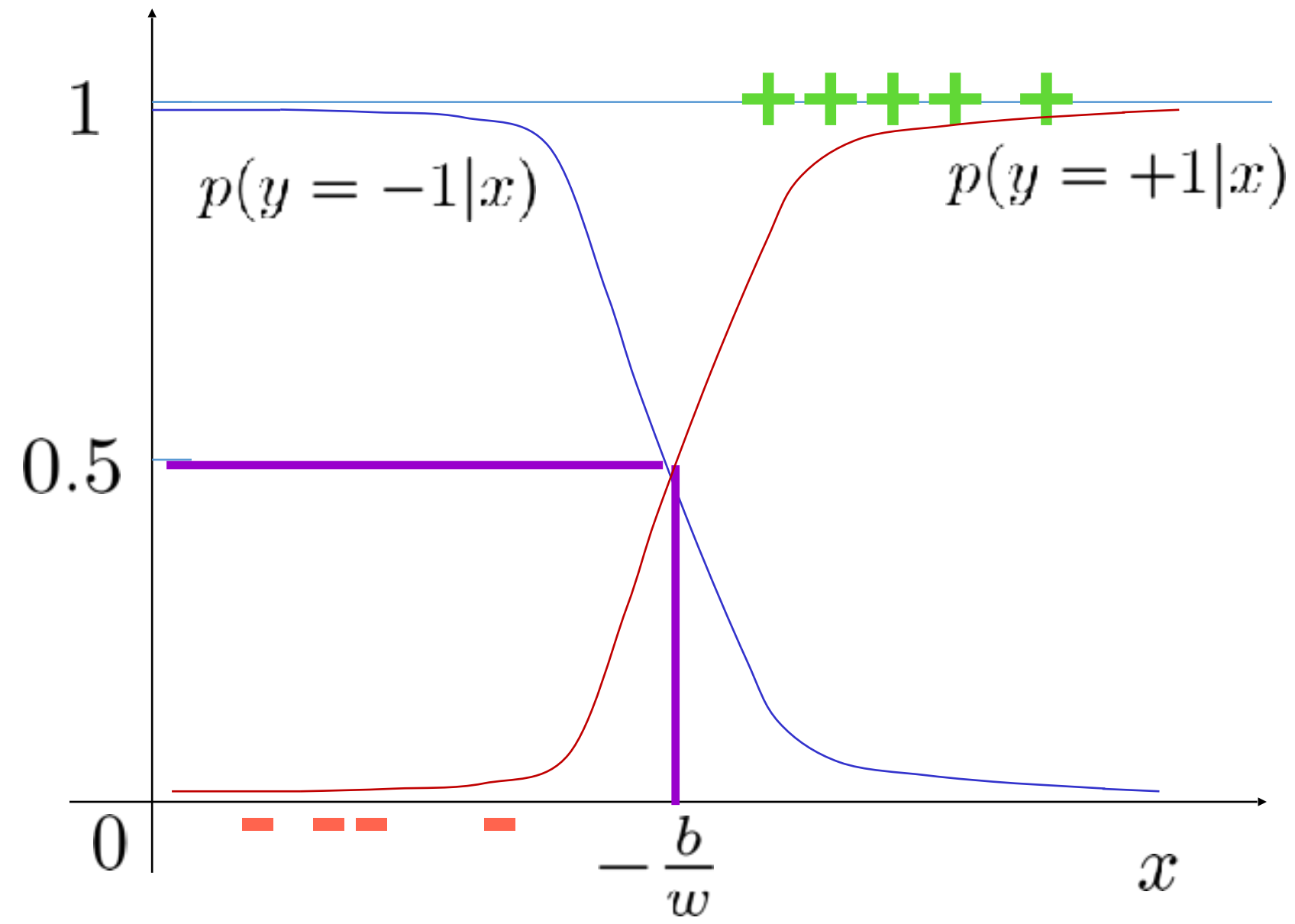
Always in range 0 - 1 => probability



Logistic regression classifier

$x, w, b \in \mathbb{R}$

Let's look at the simplest case where x is a scalar:



We have:
$$f(x; w, b) = \begin{cases} +1 & \text{if } w \times x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}.$$

Probability of a sample being a positive case

$$p(y = +1|x) = \frac{e^{w \times x + b}}{1 + e^{w \times x + b}}$$

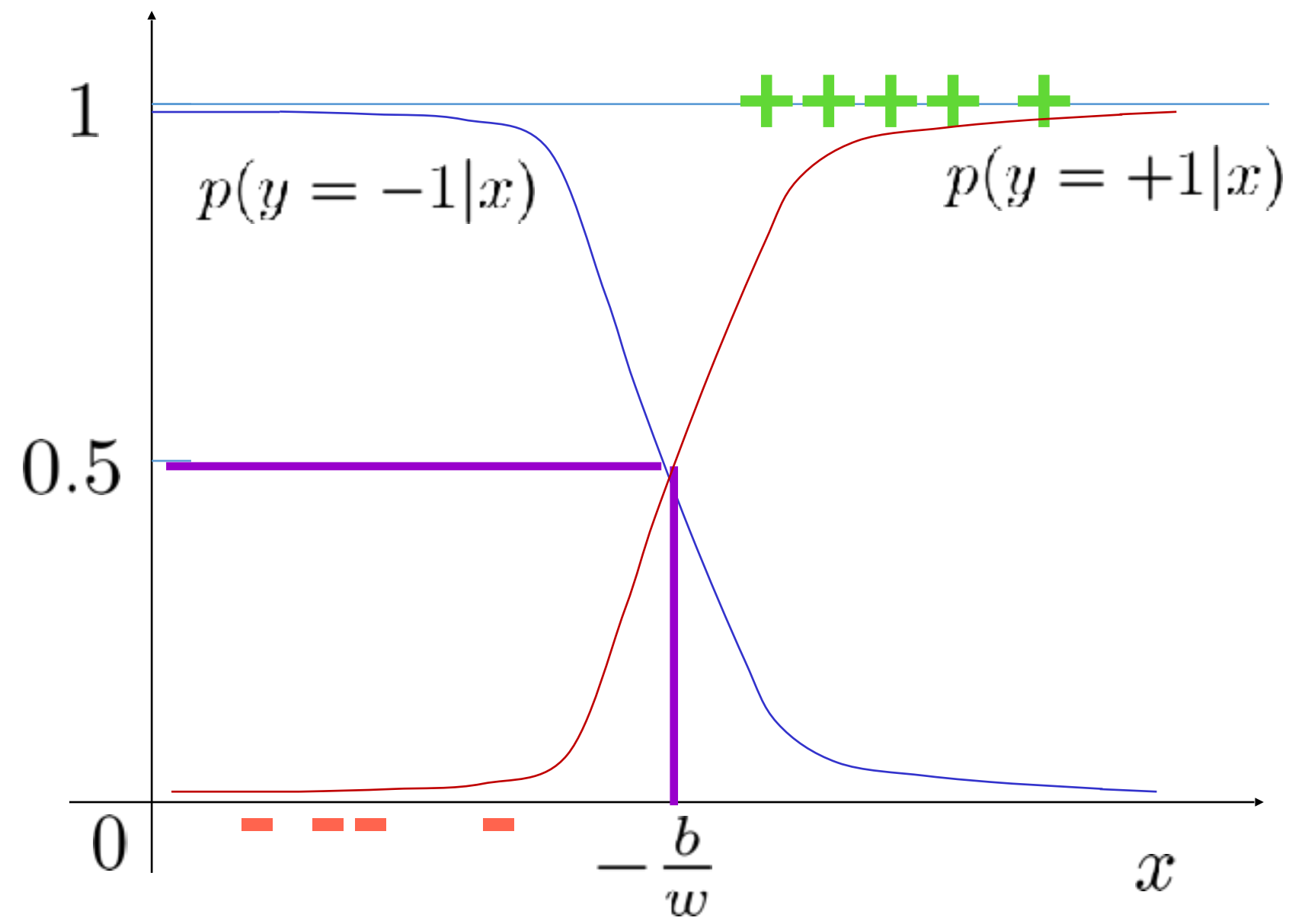
Probability of a sample being a negative case

$$p(y = -1|x) = \frac{1}{1 + e^{w \times x + b}}$$

Logistic regression classifier

$x, w, b \in \mathbb{R}$

Let's look at the simplest case where x is a scalar:



We have:
$$f(x; w, b) = \begin{cases} +1 & \text{if } w \times x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}.$$

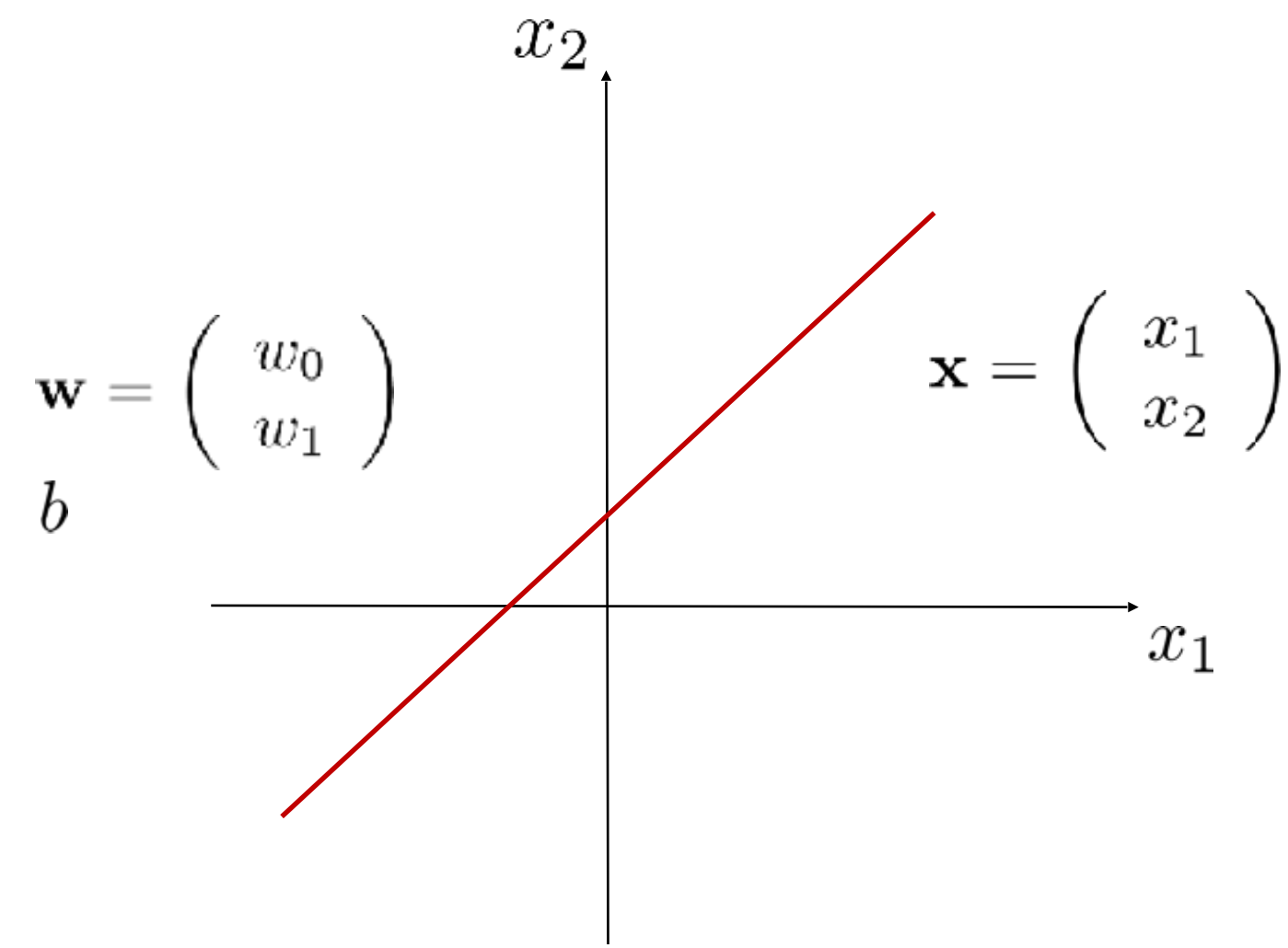
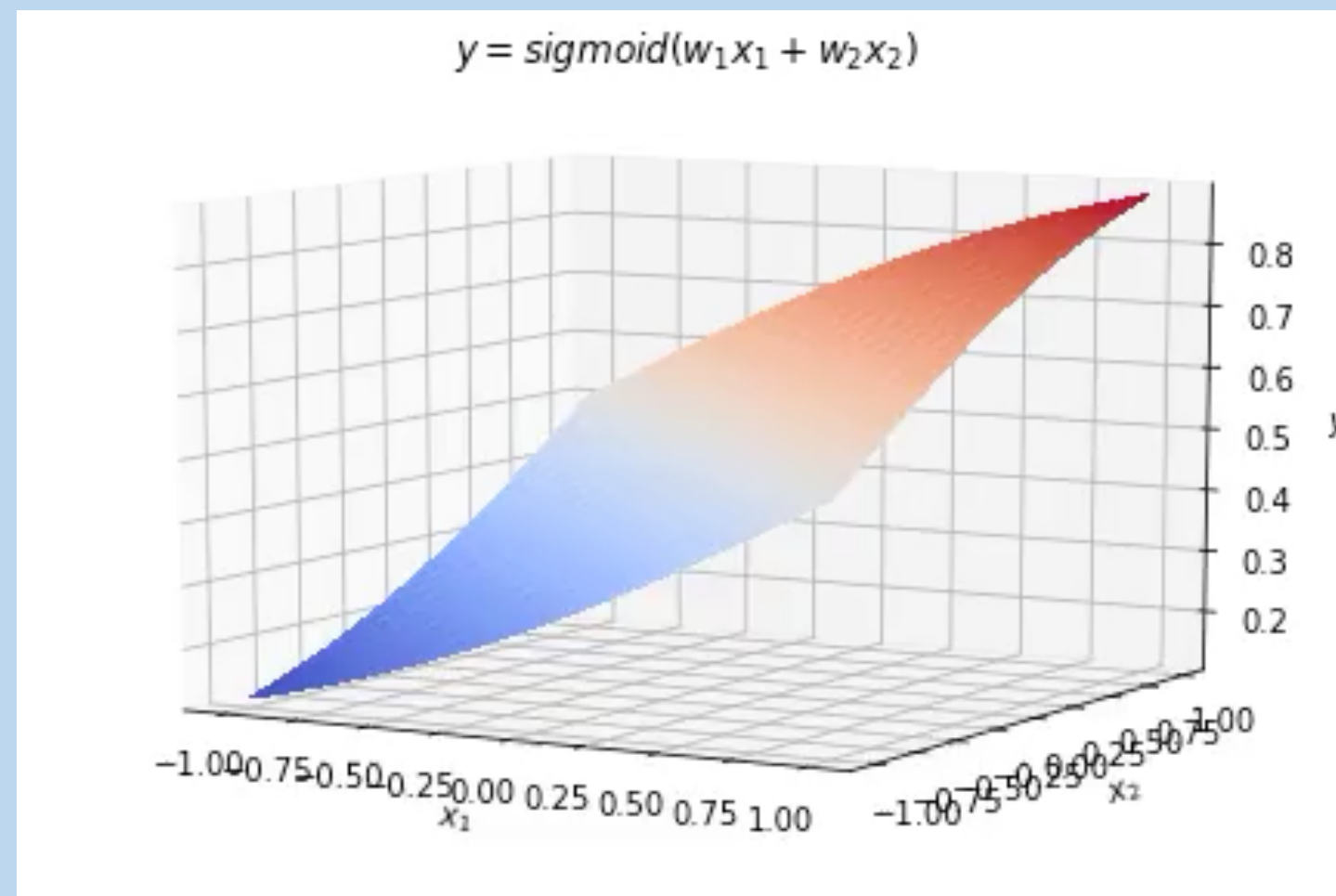
Probability of a sample being a positive case

$$p(y = +1|x) = \frac{e^{w \times x + b}}{1 + e^{w \times x + b}}$$

Probability of a sample being a negative case

$$p(y = -1|x) = \frac{1}{1 + e^{w \times x + b}}$$

Logistic regression classifier (2D case)



We have:
$$f(\mathbf{x}; \mathbf{w}, b) = \begin{cases} +1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b \geq 0 \\ -1 & \text{otherwise} \end{cases}.$$

sigmoid function:
$$\sigma(v) = \frac{1}{1 + e^{(-v)}}.$$

$$p(y = +1 | \mathbf{x}) = \sigma(\mathbf{w} \cdot \mathbf{x} + b)$$

$$p(y = -1 | \mathbf{x}) = \sigma(-(\mathbf{w} \cdot \mathbf{x} + b))$$

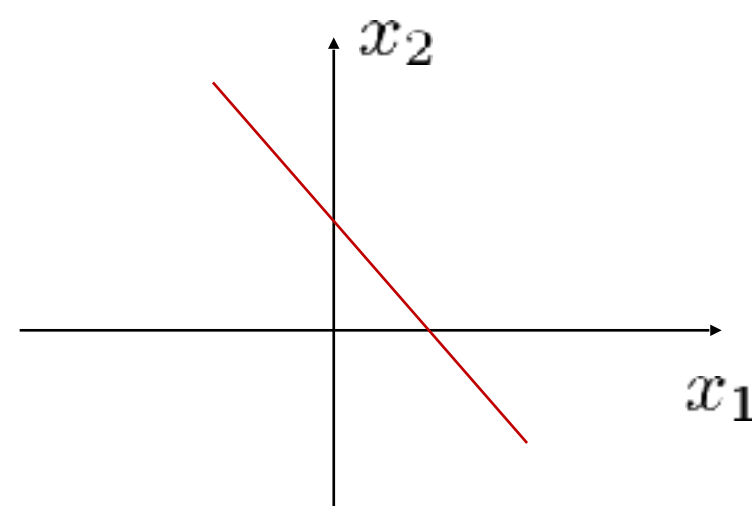
Decision boundary for a logistic regression classifier?

$$\mathbf{w} = \begin{pmatrix} w_0 \\ w_1 \end{pmatrix} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

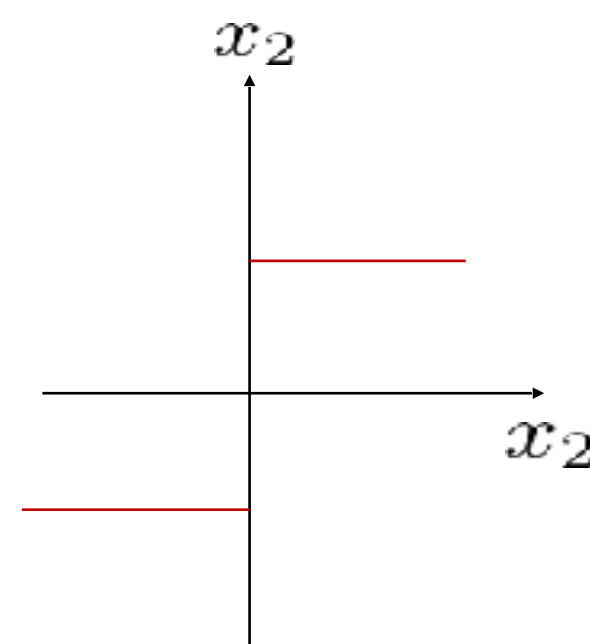
$$f(\mathbf{x}; \mathbf{w}; b) = \begin{cases} +1 & \text{if } \frac{1}{1+e^{-(\mathbf{x} \cdot \mathbf{w} + b)}} \geq 0.5 \\ -1 & \text{otherwise} \end{cases}.$$



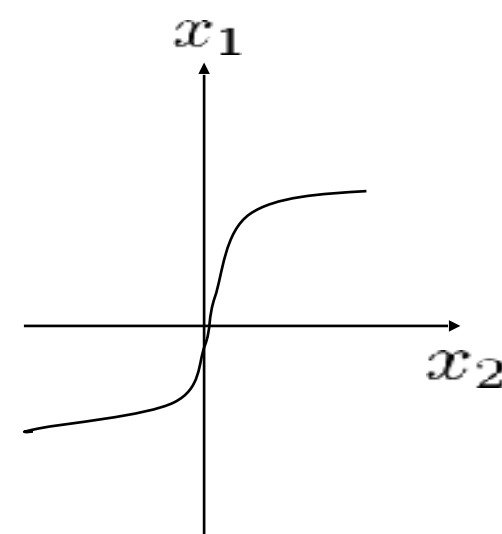
A.



B.



C.

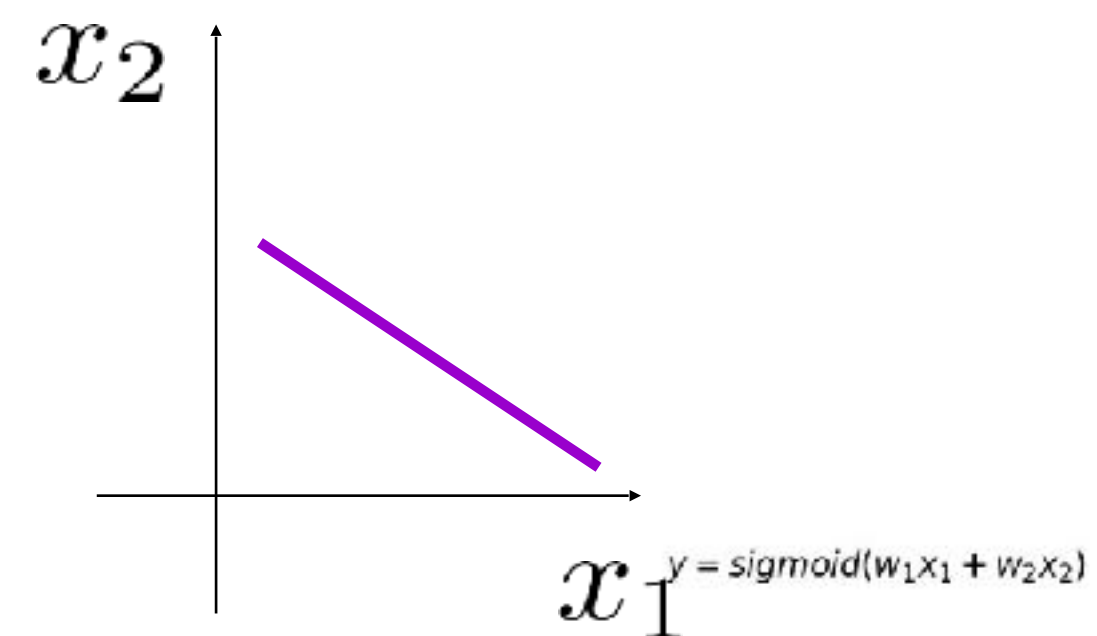


D. None of above.

Logistic regression function

$$p(y = +1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w} \cdot \mathbf{x} + b)}}$$

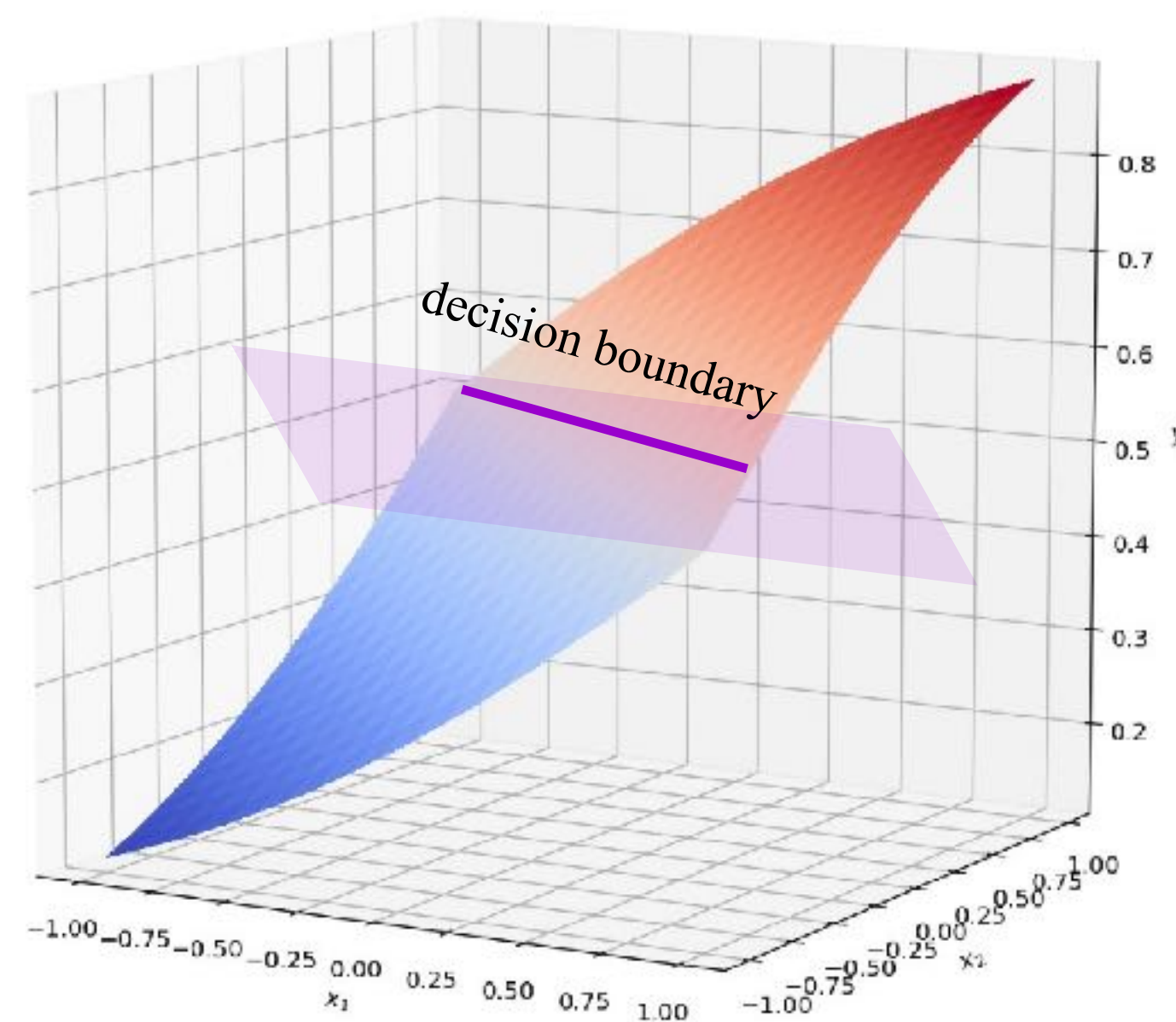
$$p(y = -1|\mathbf{x}) = \frac{1}{1+e^{(\mathbf{w} \cdot \mathbf{x} + b)}}$$



$$\mathbf{x}, \mathbf{w} \in \mathbb{R}^m$$

$$b \in \mathbb{R}$$

$$y \in \{-1, +1\}$$



Training a logistic regression classifier

$$S_{training} = \{(\mathbf{x}_i, y_i), i = 1..n\}$$

$$\mathbf{x}_i \in \mathbb{R}^m, i = 1..n \quad y_i \in \{-1, +1\}, i = 1..n$$

$$p(y_i | \mathbf{x}_i) = \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$$

Model parameters:

$$\mathbf{w} \in \mathbb{R}^m$$

$$b \in \mathbb{R}$$

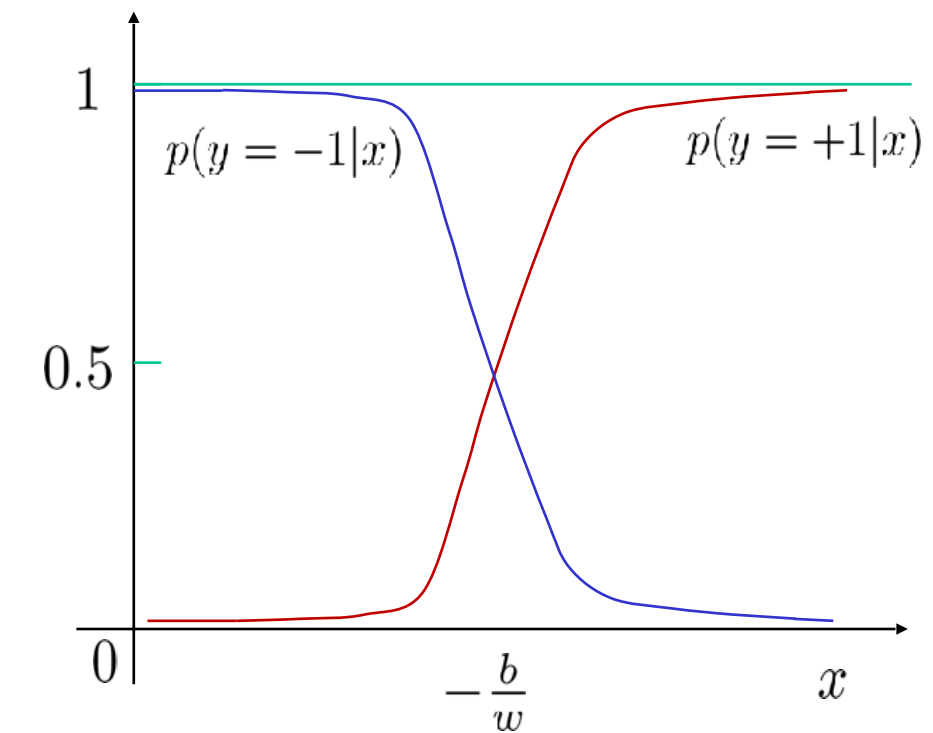
Training a logistic regression classifier

$$S_{training} = \{(-1.1, -1), (3.2, +1), (2.5, -1), (5.0, +1), (4.3, +1)\}$$

$$p(y = +1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x}+b)}}$$

$$p(y = -1|\mathbf{x}) = \frac{1}{1+e^{(\mathbf{w}^T \mathbf{x}+b)}}$$

$$p(y_i|\mathbf{x}_i) = \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i+b)}}$$



Train a logistic regression classifier $f(\mathbf{x}) = \begin{cases} +1 & \text{if } \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x}+b)}} \geq 0.5 \\ -1 & \text{otherwise} \end{cases}$:

Intuition: find the best parameters $(\mathbf{w}, b)^*$ to maximize the probabilities of fitting the ground-truth label y_i for each \mathbf{x}_i .

Math: $(\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i+b)}}$

Training a logistic regression classifier

Intuition: find the best parameters $(\mathbf{w}, b)^*$ to maximize the probabilities of fitting the ground-truth label y_i for each \mathbf{x}_i .

Math: $(\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$

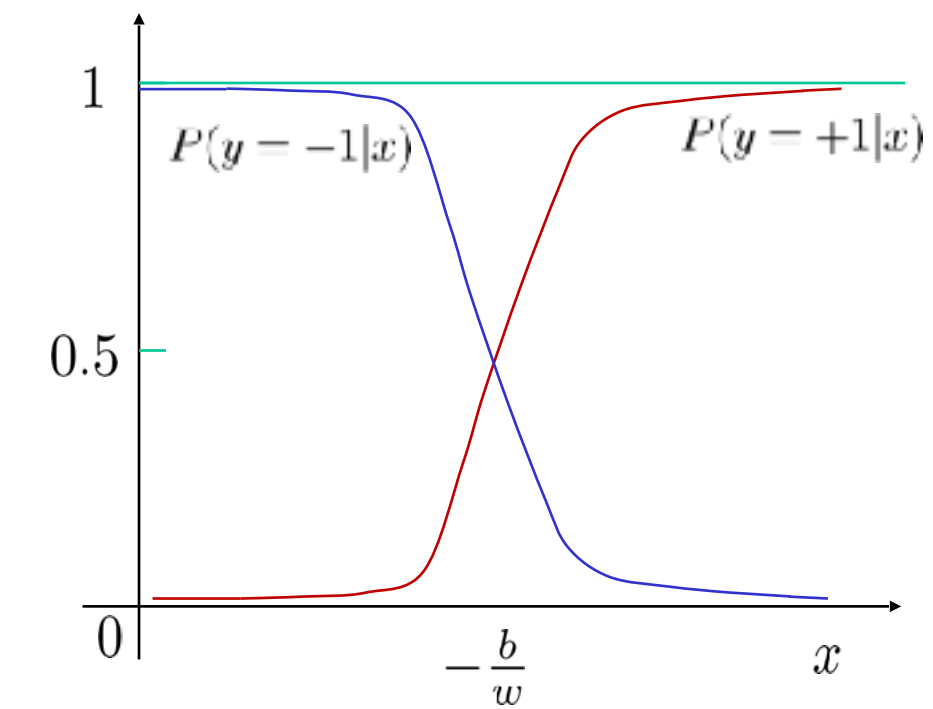
$$\begin{aligned}(\mathbf{w}, b)^* &= \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \\&= \arg \max_{(\mathbf{w}, b)} \ln \left(\prod_{i=1}^n \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \right) \\&= \arg \min_{(\mathbf{w}, b)} \sum_{i=1}^n -\ln \left(\frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \right)\end{aligned}$$

Training a logistic regression classifier

$$S_{training} = \{(-1.1, -1), (3.2, +1), (2.5, -1), (5.0, +1), (4.3, +1)\} \quad y_i \in \{-1, +1\}, i = 1..n$$

$$p(y_i | \mathbf{x}_i) = \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$$

$$(\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n [p(y_i | \mathbf{x}_i)]$$



Minus sign moves inside the sum and ln!

$$(w, b)^* = \arg \min_{(w, b)} - \sum_{i=1}^n \ln\left(\frac{1}{1 + e^{-y_i \times (w \times x_i + b)}}\right) = \arg \min_{(w, b)} \sum_{i=1}^n \ln(1 + e^{-y_i \times (w \times x_i + b)})$$

$$(w, b)^* = \arg \min_{(w, b)} [\ln(1 + e^{(-1.1w+b)}) + \ln(1 + e^{-(3.2w+b)}) + \\ \ln(1 + e^{(2.5w+b)}) + \ln(1 + e^{-(5.0w+b)}) + \ln(1 + e^{-(4.3w+b)})]$$

Training a MULTIVARIATE logistic regression classifier

$$\mathbf{x}_i \in \mathbb{R}^m, i = 1..n \quad y_i \in \{-1, +1\}, i = 1..n$$

Model parameters: $\mathbf{w} \in \mathbb{R}^m$ and $b \in \mathbb{R}$

$$p(y_i|\mathbf{x}_i) = \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$$

Intuition: find the best parameters $(\mathbf{w}, b)^*$ to maximize the probabilities of fitting the ground-truth label y_i for each x_i .

Math: $(\mathbf{w}, b)^* = \arg \max_{(\mathbf{w}, b)} \prod_{i=1}^n \frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$

$$\begin{aligned} (\mathbf{w}, b)^* &= \arg \min_{(\mathbf{w}, b)} \sum_{i=1}^n -\ln\left(\frac{1}{1+e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}\right) \\ &= \arg \min_{\mathbf{w}, b} \mathcal{L}(\mathbf{w}, b) \end{aligned}$$

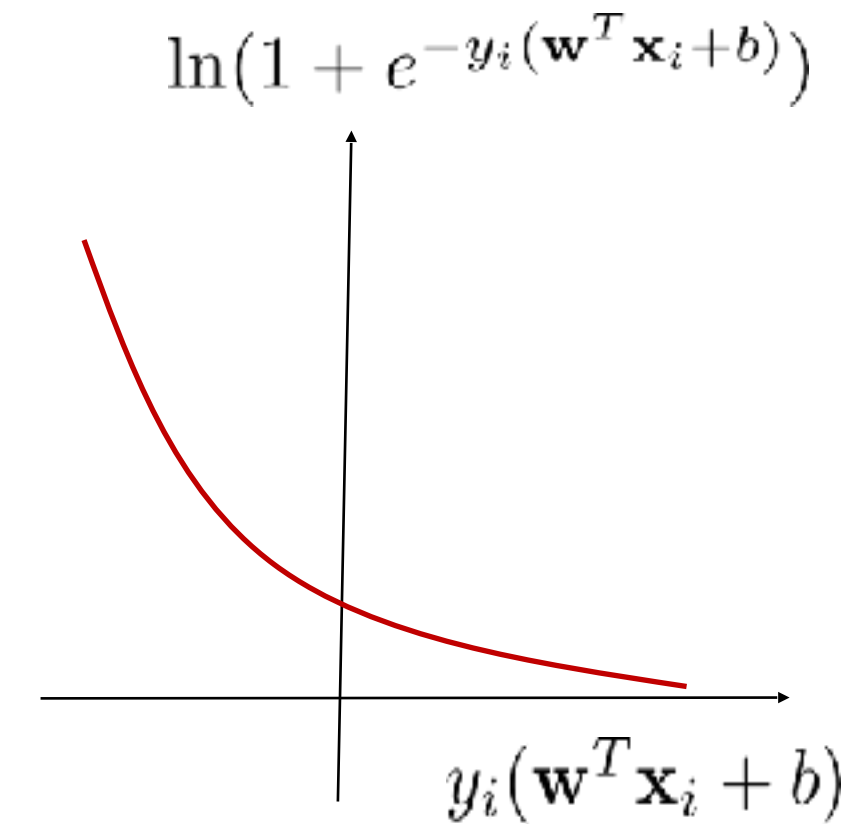
Derivative for the logistic regression classifier

$$\mathcal{L}(\mathbf{w}, b) = \sum_{i=1}^n \ln(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}) \quad p(y_i | \mathbf{x}_i) = \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}$$

$$\begin{aligned} \frac{\partial \mathcal{L}(\mathbf{w}, b)}{\partial \mathbf{w}} &= \sum_{i=1}^n \frac{\partial \ln(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)})}{\partial \mathbf{w}} \\ &= \sum_{i=1}^n \frac{\frac{\partial(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)})}{\partial \mathbf{w}}}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \quad \text{Ln}' = 1/x; \text{ chain rule} \\ &= \sum_{i=1}^n \frac{e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}(-y_i \mathbf{x}_i)}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \quad \text{Exp}' = \text{exp}; \text{ chain rule} \\ &= \sum_{i=1}^n \frac{(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)} - 1)(-y_i \mathbf{x}_i)}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} \quad +1 - 1 \text{ to allow factoring below} \\ &= \sum_{i=1}^n \left(1 - \frac{1}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}\right)(-y_i \mathbf{x}_i) \\ &= \sum_i -y_i \mathbf{x}_i (1 - p(y_i | \mathbf{x}_i)) \end{aligned}$$

Multivariate input

$$\mathcal{L}(\mathbf{w}, b) = \sum_{i=1}^n \ln(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)})$$



$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}, b) = \sum_i \frac{-y_i \mathbf{x}_i e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} = \sum_i -y_i \mathbf{x}_i (1 - p(y_i | \mathbf{x}_i))$$

$$\nabla_b \mathcal{L}(\mathbf{w}, b) = \sum_i \frac{-y_i e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}}{1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)}} = \sum_i -y_i (1 - p(y_i | \mathbf{x}_i))$$

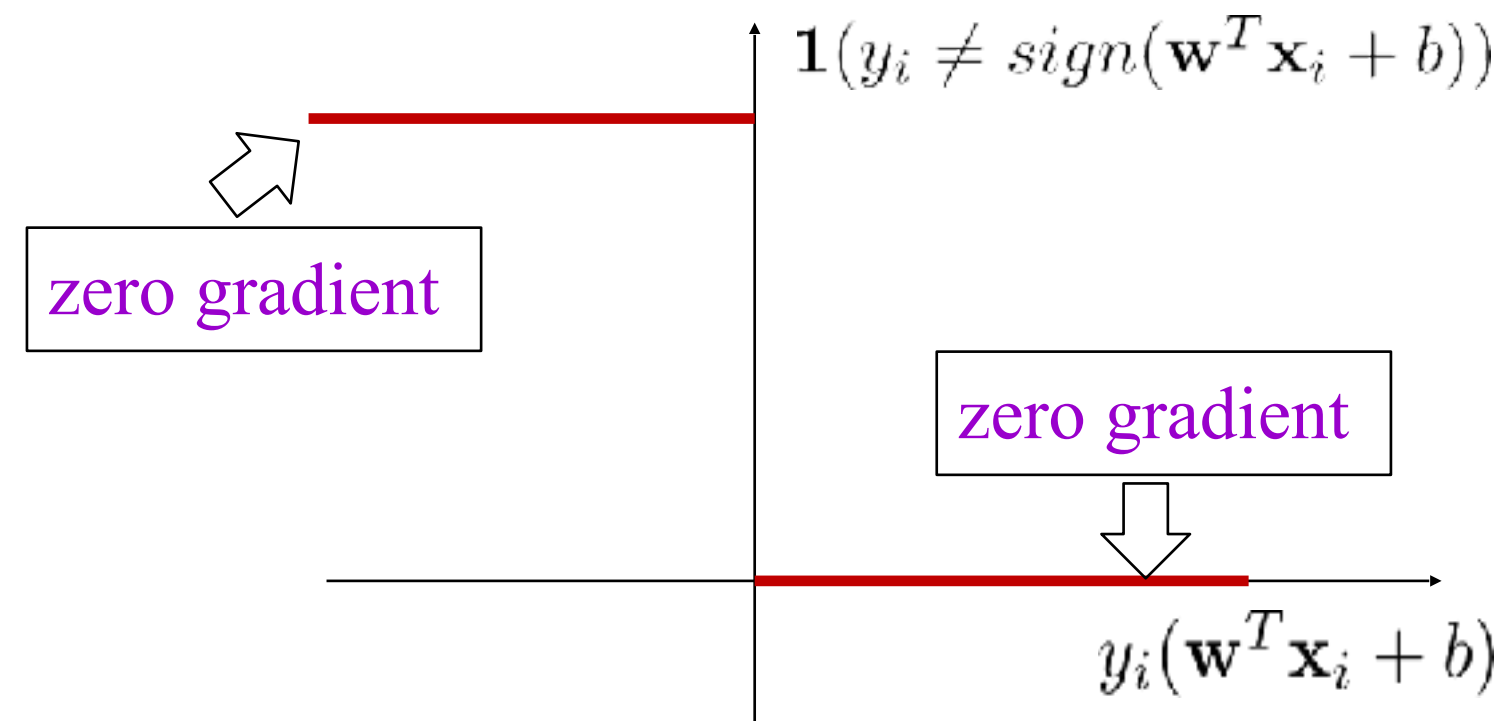
$$\mathbf{w}_{t+1} = \mathbf{w}_t - \lambda_t \times \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}_t, b_t)$$

$$b_{t+1} = b_t - \lambda_t \times \nabla_b \mathcal{L}(\mathbf{w}_t, b_t)$$

Standard loss (error) function

Standard 0/1 loss (gradient 0 nearly everywhere,
no gradient feedback):

Training: Minimize $\mathcal{L}(\mathbf{w}, b) = \sum_i \mathbf{1}(y_i \neq \text{sign}(\mathbf{w}^T \mathbf{x}_i + b))$



Main motivation

Hard->Half-hard->Soft

Error

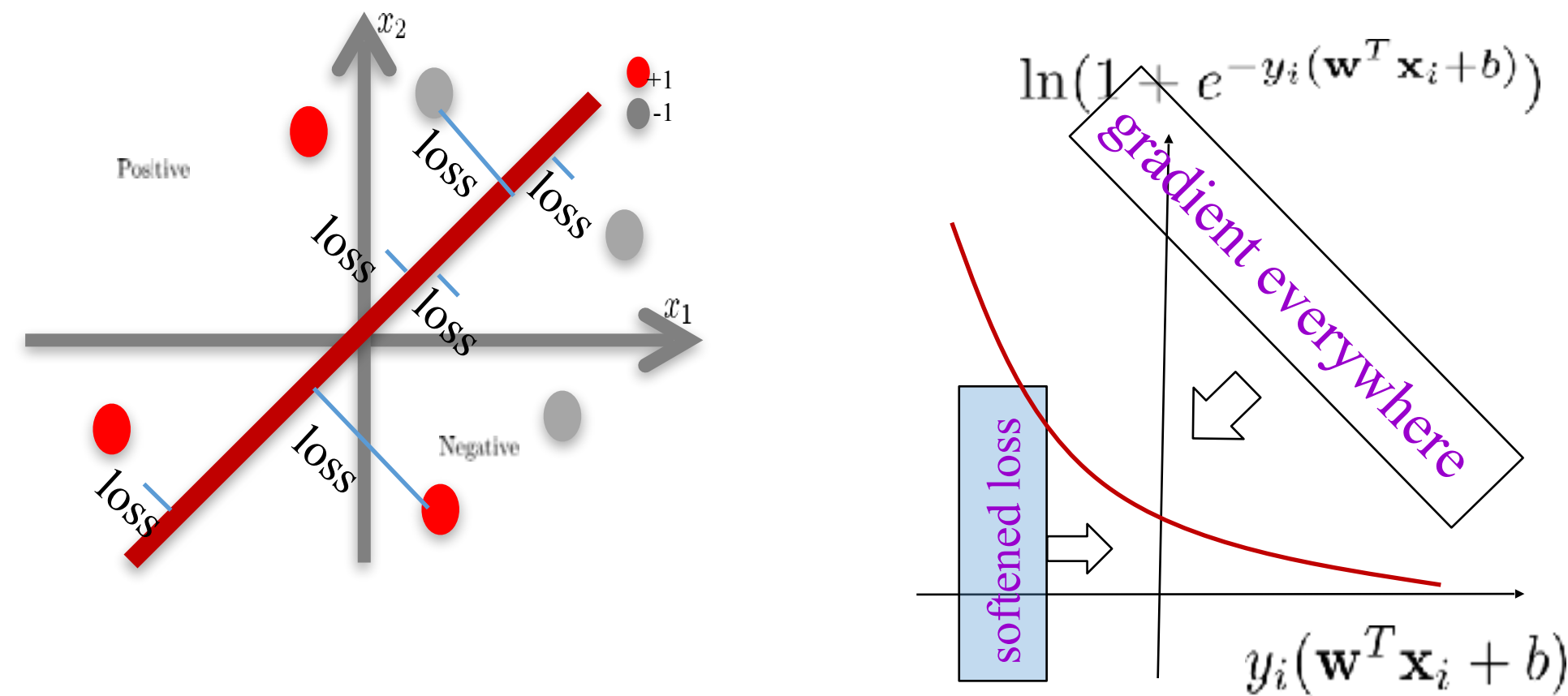
It is the most **directly** loss, but is
also the **hardest** to minimize.

Zero gradient everywhere!

Soft loss (error) function

Loss used in logistic regression.

Training: minimize $\mathcal{L}(\mathbf{w}, b) = \sum_{i=1}^n \ln(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)})$



Main motivation

Hard->Half-hard->**Soft**

Error

Every data point receives a loss (gradient everywhere).

A loss based on the **distance to the decision boundary** for wrong classification (has a gradient).

Used in **logistic regression** classifier.

Logistic regression classifier

$$p(y_i|\mathbf{x}_i) = \frac{1}{1+e^{-y_i(\mathbf{w}\cdot\mathbf{x}_i+b)}}$$

$$\mathbf{x} \in \mathbb{R}^m$$

$$y \in \{-1, +1\}$$

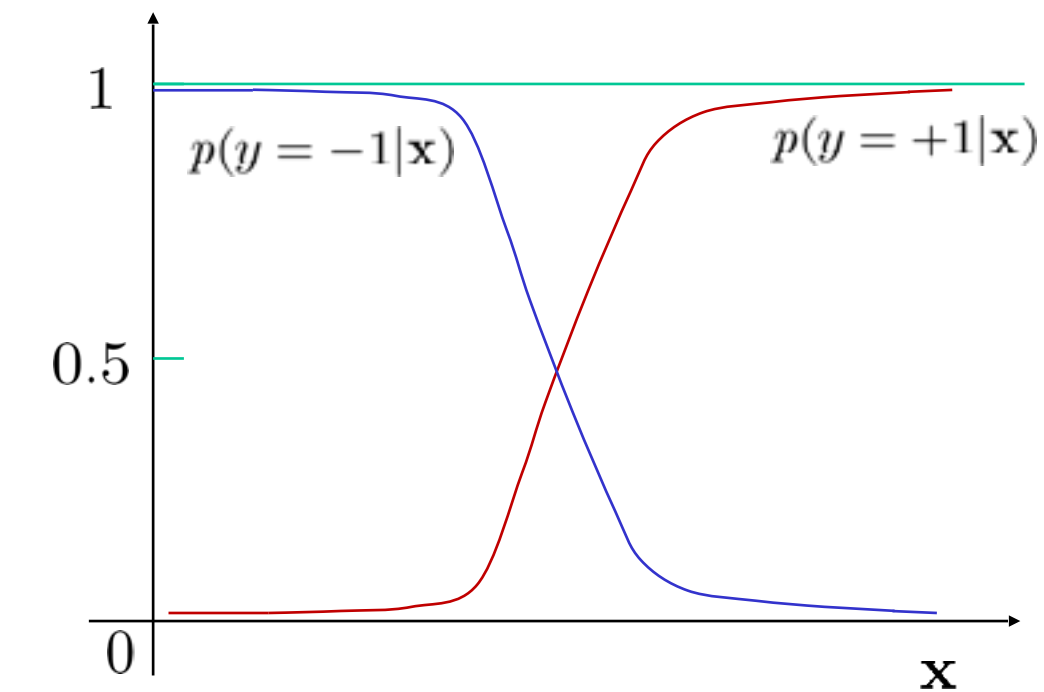
$$f(\mathbf{x}) = \begin{cases} +1 & \text{if } \frac{1}{1+e^{-(\mathbf{w}\cdot\mathbf{x}+b)}} \geq 0.5 \\ -1 & \text{otherwise} \end{cases}$$

Pros:

1. It is well-normalized.
2. Easy to turn into probability.
3. Easy to implement.

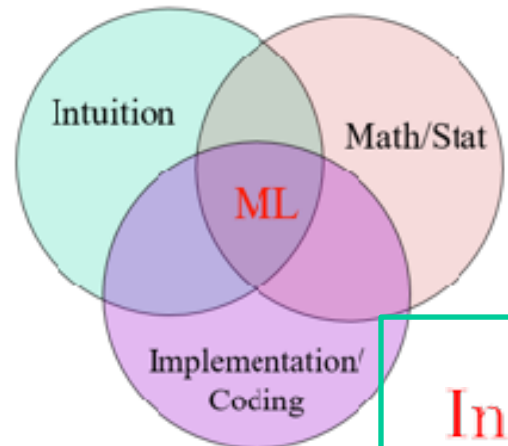
Cons:

1. Indirect loss function.
2. Dependent on good feature set.
3. Weak on feature selection.



Take home message

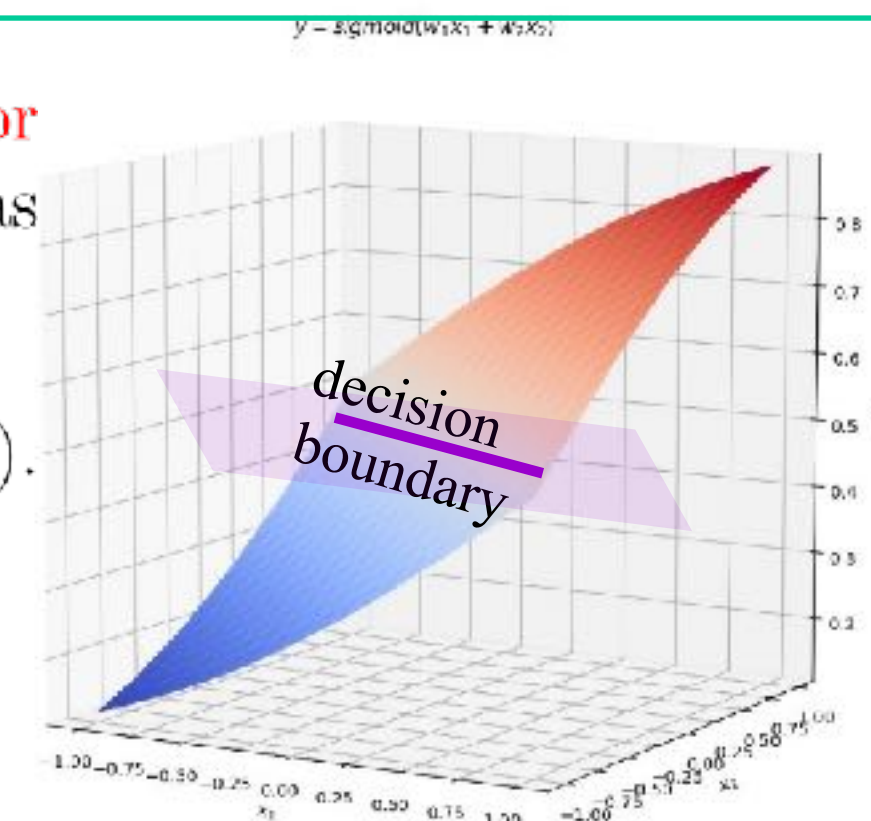
- Logistic regression classifier is still a **linear** classifier but with a **probability** output.
- It can be trained using a **gradient** descent algorithm.
- The “**regression**” refers to fitting the **discriminative probabilities**: $p(y|\mathbf{x})$
- It has been widely adopted in practice, especially in the modern **deep learning** era.



Recap: Logistic Regression Classifier

Intuition: Logistic regression classifier nicely turns a **hard classification error** (0 or 1) into a **soft measure** using the sigmoid function $\sigma(v) = \frac{1}{1+e^{-v}}$ which has three particularly appealing properties:

- A soft measure that maps any value $v \in (-\infty, \infty)$ to a normalized $\rightarrow (0, 1)$.
- Nice gradient form.
- Convex function for the objective function in training.



Math:

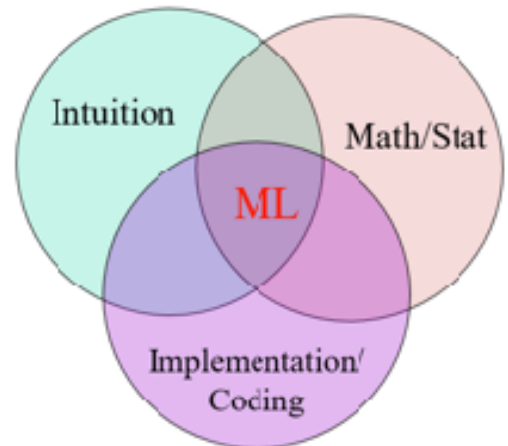
$$p(y|\mathbf{x}) = \frac{1}{1+e^{-y(\mathbf{w}^T \mathbf{x} + b)}}$$

Training :

$$(\mathbf{w}, b)^* = \arg \min_{(\mathbf{w}, b)} \mathcal{L}(\mathbf{w}, b) = \arg \min_{(\mathbf{w}, b)} \sum_{i=1}^n \ln(1 + e^{-y_i(\mathbf{w}^T \mathbf{x}_i + b)})$$

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}, b) = \sum_i -y_i \times \mathbf{x}_i (1 - p(y_i|\mathbf{x}_i))$$

$$\nabla_b \mathcal{L}(\mathbf{w}, b) = \sum_i -y_i \times (1 - p(y_i|\mathbf{x}_i))$$



Recap: Logistic Regression Classifier

Implementation:

Gradient Descent Direction

- (a) Pick a direction $\nabla \mathcal{L}(\mathbf{w}_t, b_t)$
- (b) Pick a step size λ_t
- (c) $\mathbf{w}_{t+1} = \mathbf{w}_t - \lambda_t \times \nabla \mathcal{L}_{\mathbf{w}_t}(\mathbf{w}_t, b_t)$ such that function decreases;
 $b_{t+1} = b_t - \lambda_t \times \nabla \mathcal{L}_{b_t}(\mathbf{w}_t, b_t)$
- (d) Repeat

