

California State University, Dominguez Hills

Department of Computer Science

CSC 595

Professor: Dr. Benyamin Ahmadnia
bahmadniayebosari@csudh.edu

Spring 2025

Copyright Notice

These slides are distributed under the Creative Commons License.

[DeepLearning.AI](#) makes these slides available for educational purposes. You may not use or distribute these slides for commercial purposes. You may make copies of these slides and use or distribute them for educational purposes as long as you cite [DeepLearning.AI](#) as the source of the slides.

For the rest of the details of the license, see <https://creativecommons.org/licenses/by-sa/2.0/legalcode>

Table of Contents

- Cost Function
- Gradient Descent
- Cost Function for Logistic Regression
- Gradient Descent for Logistic Regression
- Classification with Logistic Regression

Gradient Descent

Gradient Descent

Have some function $J(w, b)$ *for linear regression
or any function*

Want $\min_{w, b} J(w, b)$ $\min_{w_1, \dots, w_n, b} \underline{J(w_1, w_2, \dots, w_n, b)}$

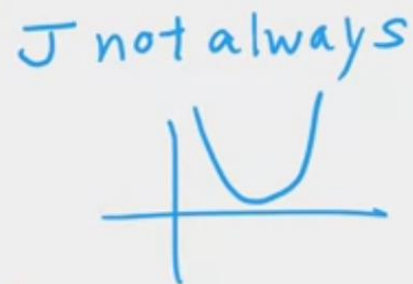
Outline:

Start with some w, b *(set $w=0, b=0$)*

Keep changing w, b to reduce $J(w, b)$

Until we settle at or near a minimum

may have >1 minimum



Implementing Gradient Descent

Gradient descent algorithm

$$w = w - \alpha \frac{d}{dw} J(w, b)$$

Implementing Gradient Descent

Gradient descent algorithm

$$w \leftarrow w - \alpha \frac{d}{dw} J(w, b)$$

Assignment

$$a = C$$



$$a = a + 1$$

Code

Implementing Gradient Descent

Gradient descent algorithm

$$w \leftarrow w - \alpha \frac{d}{dw} J(w, b)$$

Learning rate

Assignment

$$a = c$$

$$a = a + 1$$

Code

Truth assertion

$$a = c$$

$$a = a + 1$$

Math

$$a == c$$

Implementing Gradient Descent

Gradient descent algorithm

$$w \leftarrow w - \alpha \frac{d}{dw} J(w, b)$$

Learning rate
Derivative

Assignment

$$a = c$$

$$a = a + 1$$

Code

Truth assertion

$$a = c$$

$$a = a + 1$$

Math

$$a == c$$

Implementing Gradient Descent

Gradient descent algorithm

Repeat until convergence

$$w \leftarrow w - \alpha \frac{d}{dw} J(w, b)$$

$$b \leftarrow b - \alpha \frac{d}{db} J(w, b)$$

Learning rate
Derivative

Assignment

$$a = c$$

$$a = a + 1$$

Code

Truth assertion

$$a = c$$

$$a = a + 1$$

Math

$$a == c$$

Implementing Gradient Descent

Gradient descent algorithm

Repeat until convergence

$$\begin{cases} \underline{w} = w - \alpha \frac{d}{dw} J(w, b) \\ \underline{b} = b - \alpha \frac{d}{db} J(w, b) \end{cases}$$

Learning rate
Derivative

Simultaneously
update w and b

Assignment

$a = c$
 $a = a + 1$
Code

Truth assertion

$a = c$
 $a = a + 1$
~~X~~
Math
 $a == c$

Implementing Gradient Descent

Gradient descent algorithm

Repeat until convergence

$$\left\{ \begin{array}{l} \underline{w} = w - \alpha \frac{\partial}{\partial w} J(w, b) \\ \underline{b} = b - \alpha \frac{\partial}{\partial b} J(w, b) \end{array} \right.$$

Learning rate
Derivative

Simultaneously
update w and b

Assignment

$$a = c$$
$$a = a + 1$$

Code

Truth assertion

$$a = c$$
$$a = a + 1$$

Math
 $a == c$

Correct: Simultaneous update

$$tmp_w = w - \alpha \frac{\partial}{\partial w} J(w, b)$$

$$tmp_b = b - \alpha \frac{\partial}{\partial b} J(w, b)$$

$$w = tmp_w$$

$$b = tmp_b$$

Implementing Gradient Descent

Gradient descent algorithm

Repeat until convergence

$$\left\{ \begin{array}{l} \underline{w} = w - \alpha \frac{\partial}{\partial w} J(w, b) \\ \underline{b} = b - \alpha \frac{\partial}{\partial b} J(w, b) \end{array} \right.$$

Learning rate
Derivative

Simultaneously
update w and b

Assignment

$$a = c$$

$$a = a + 1$$

Code

Truth assertion

$$a = c$$

$$a = a + 1$$

Math

$$a == c$$

Correct: Simultaneous update

$$tmp_w = w - \alpha \frac{\partial}{\partial w} J(w, b)$$

$$tmp_b = b - \alpha \frac{\partial}{\partial b} J(w, b)$$

$$w = tmp_w$$

$$b = tmp_b$$

Gradient Descent Intuition

Gradient descent algorithm

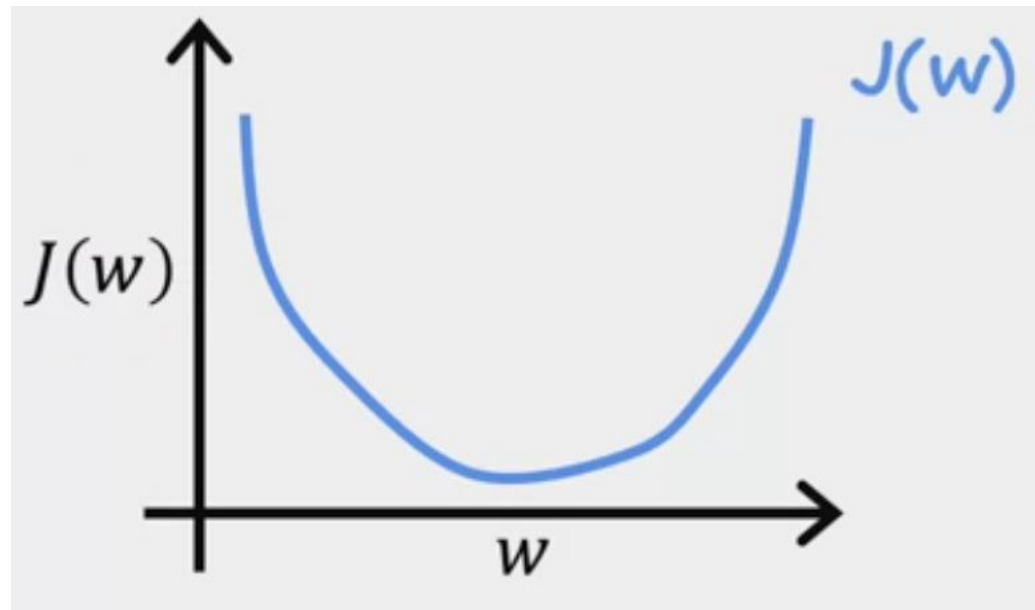
repeat until convergence {

learning rate α derivative

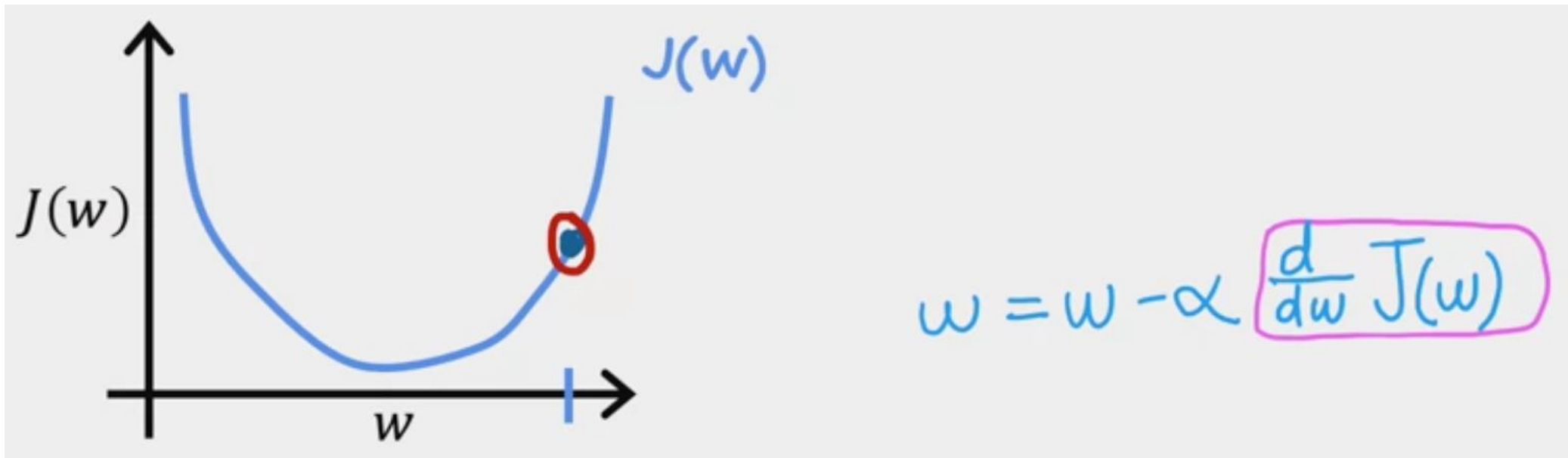
$$\begin{cases} \underline{w} = w - \alpha \frac{\partial}{\partial w} J(w, b) \\ \underline{b} = b - \alpha \frac{\partial}{\partial b} J(w, b) \end{cases}$$

$$\begin{aligned} &J(w) \\ &w = w - \alpha \frac{\partial}{\partial w} J(w) \\ &\underline{\min}_w J(w) \end{aligned}$$

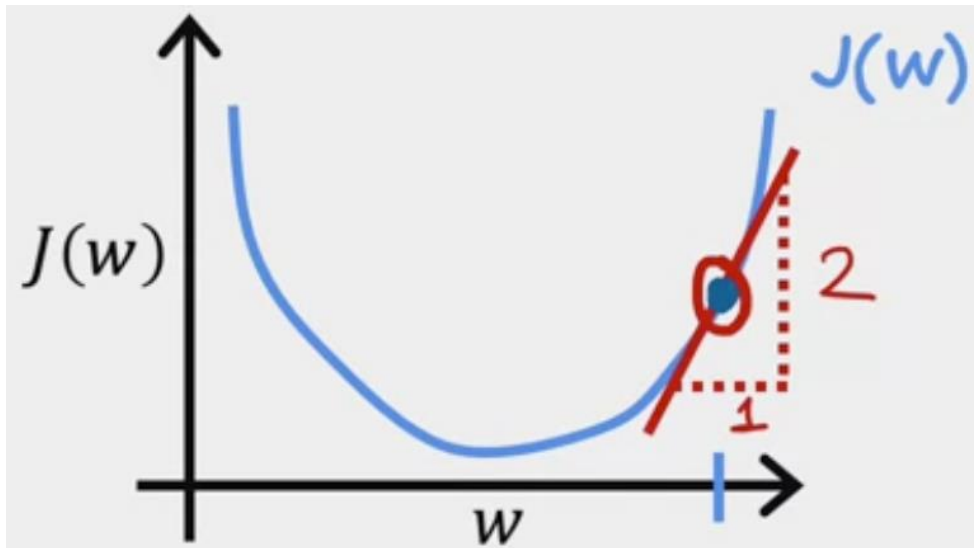
Gradient Descent Intuition



Gradient Descent Intuition



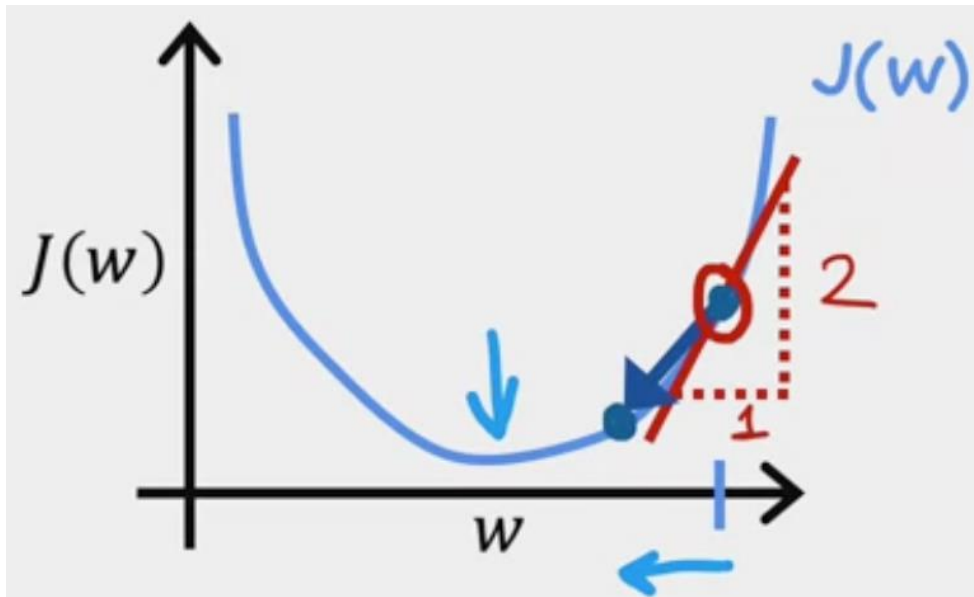
Gradient Descent Intuition



$$w = w - \alpha \underbrace{\frac{d}{dw} J(w)}_{>0}$$

$$w = w - \underline{\alpha} \cdot (\text{positive number})$$

Gradient Descent Intuition

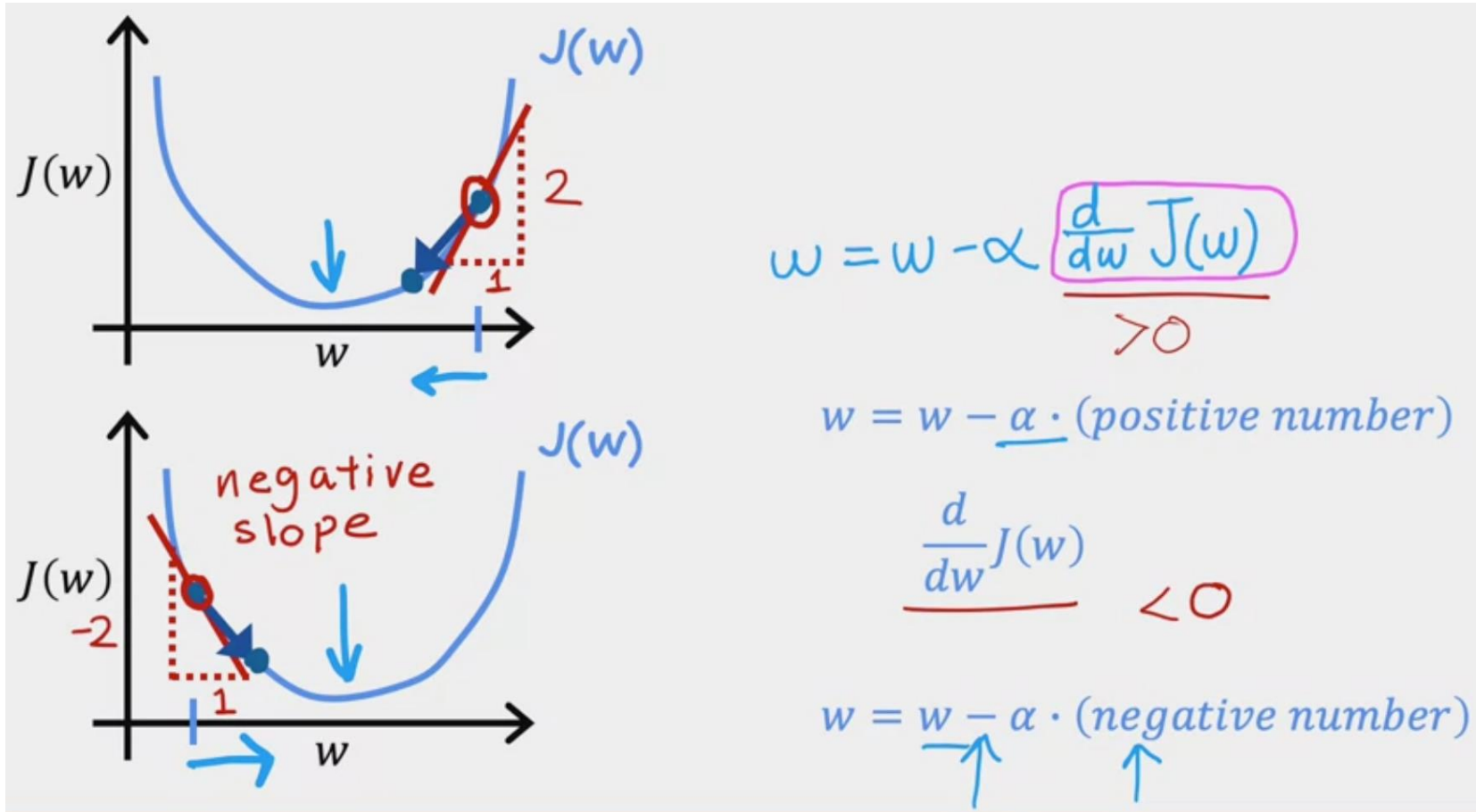


$$w = w - \alpha \frac{d}{dw} J(w)$$

> 0

$$w = w - \underline{\alpha} \cdot (\text{positive number})$$

Gradient Descent Intuition



Classification with Logistic Regression

Classification – Motivation

| Question | Answer " <i>y</i> " |
|--|---------------------|
| Is this email <u>spam</u> ? | no yes |
| Is the transaction <u>fraudulent</u> ? | no yes |
| Is the tumor <u>malignant</u> ? | no yes |

y can only be one of **two** values

"**binary** classification"

class = category

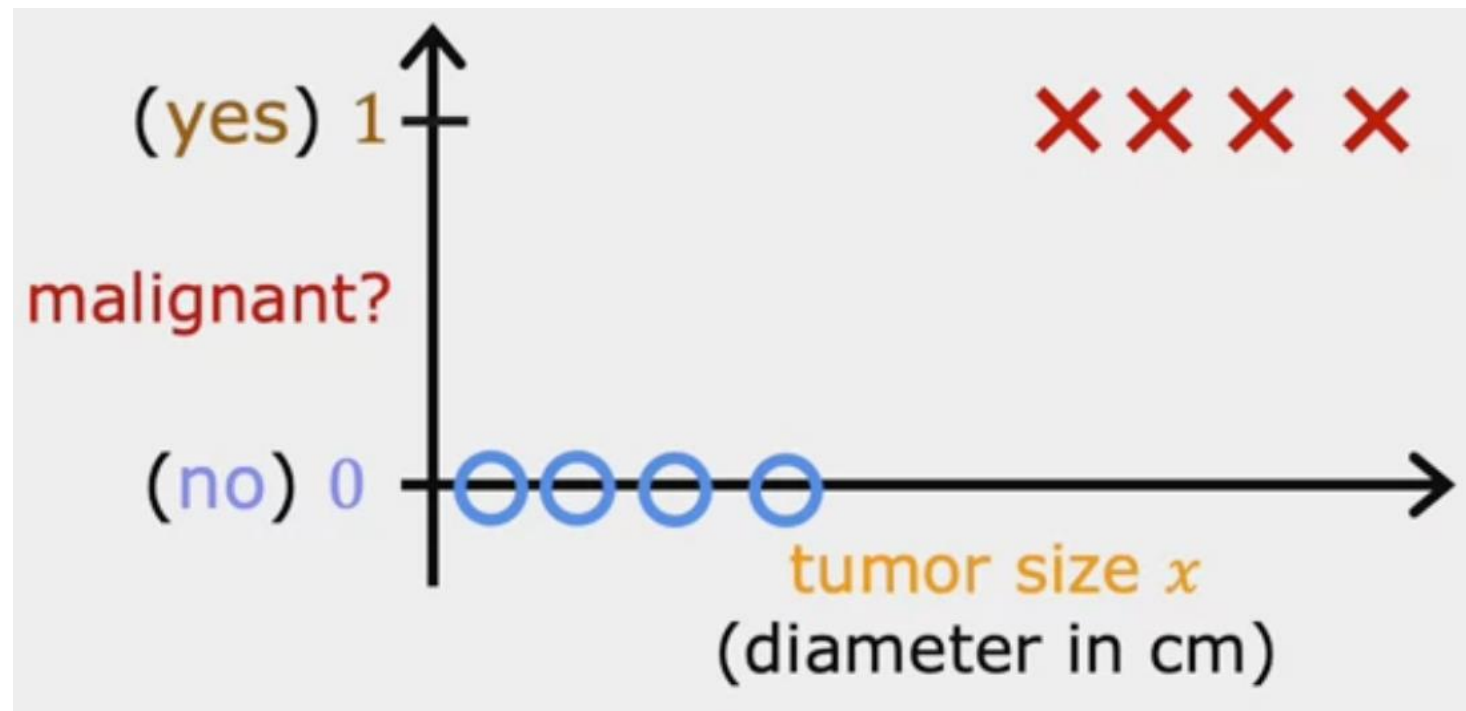
0 1

negative class
≠ "bad"
absence

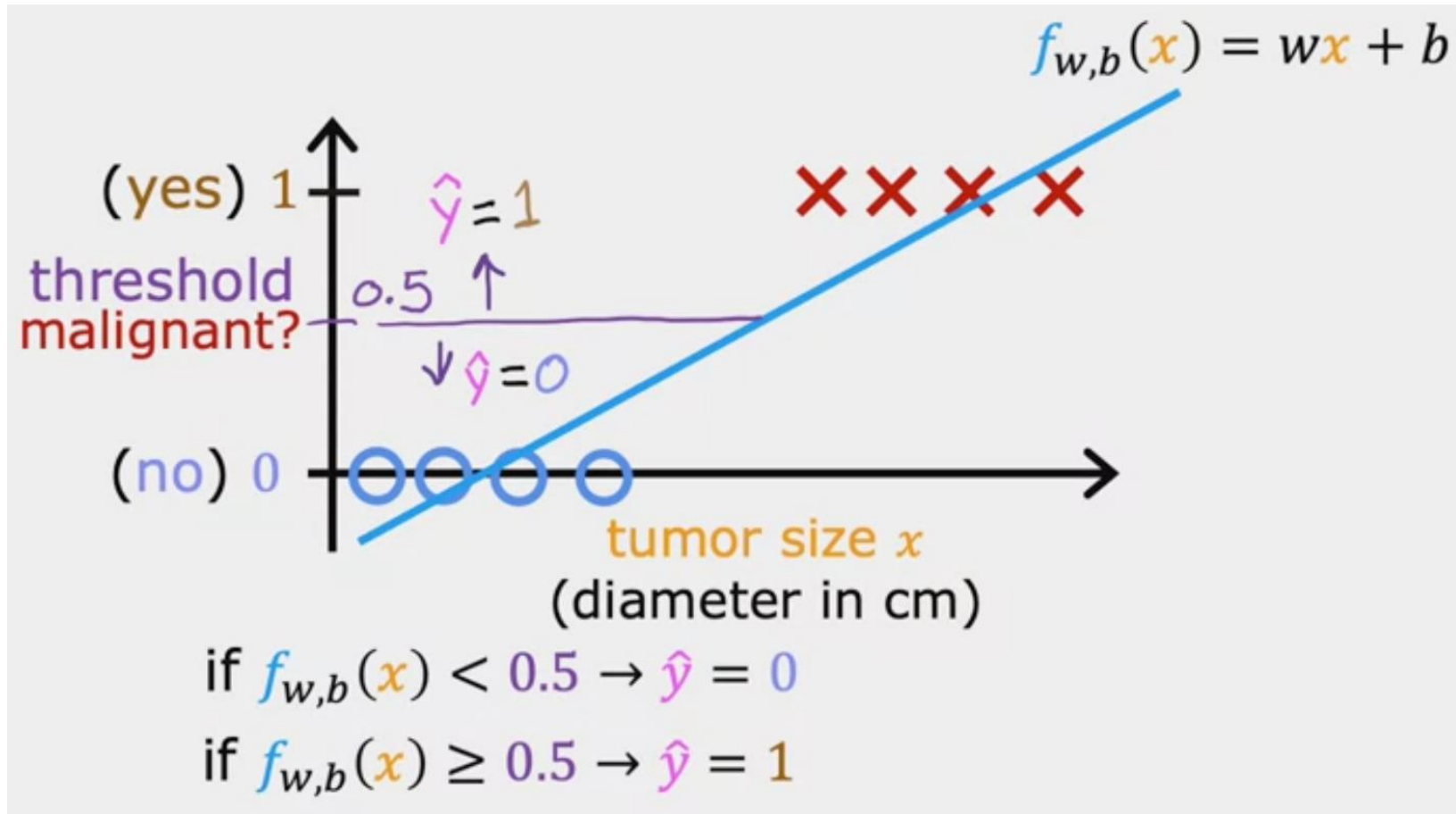
positive class
≠ "good"
presence

useful for classification

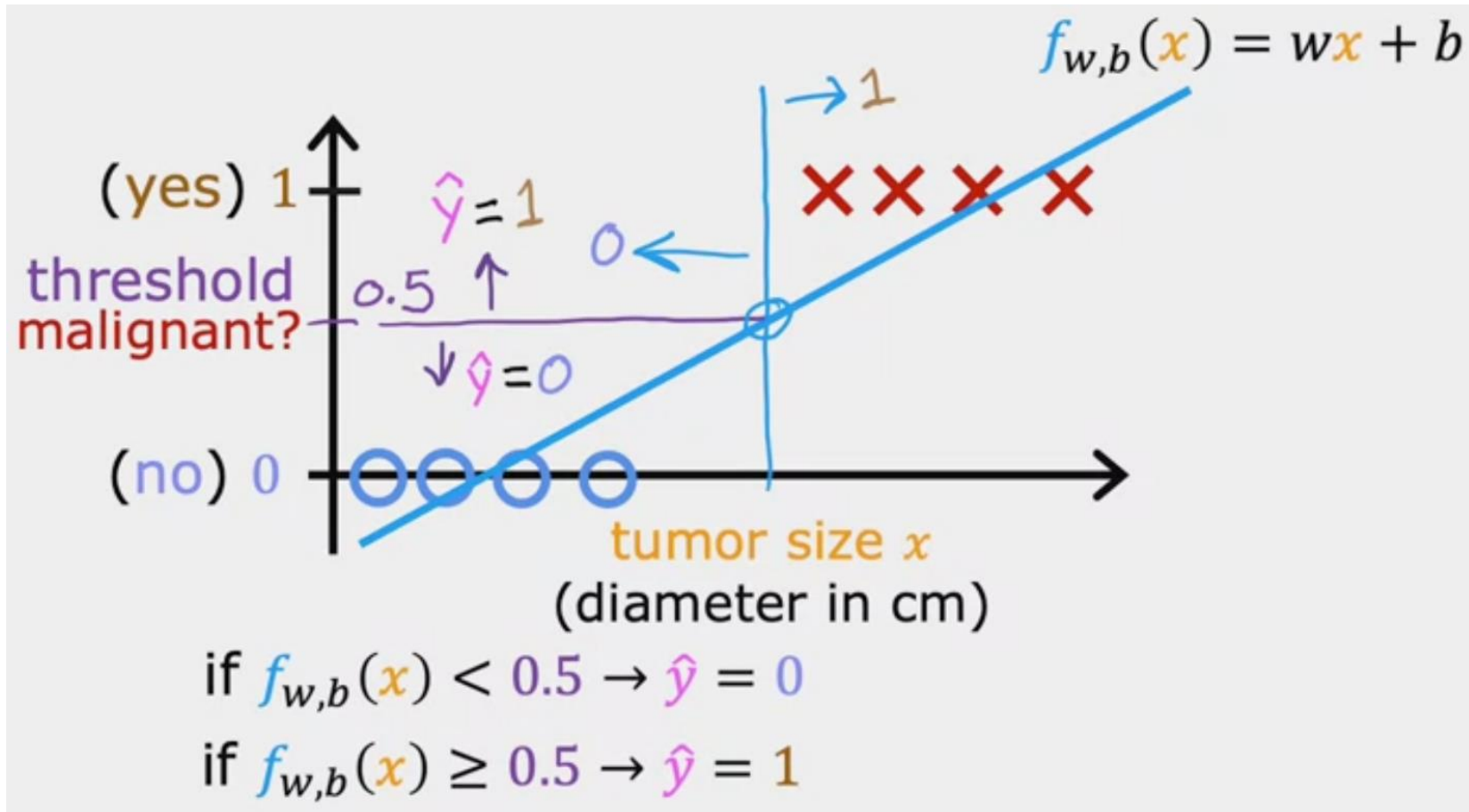
Classification – Motivation



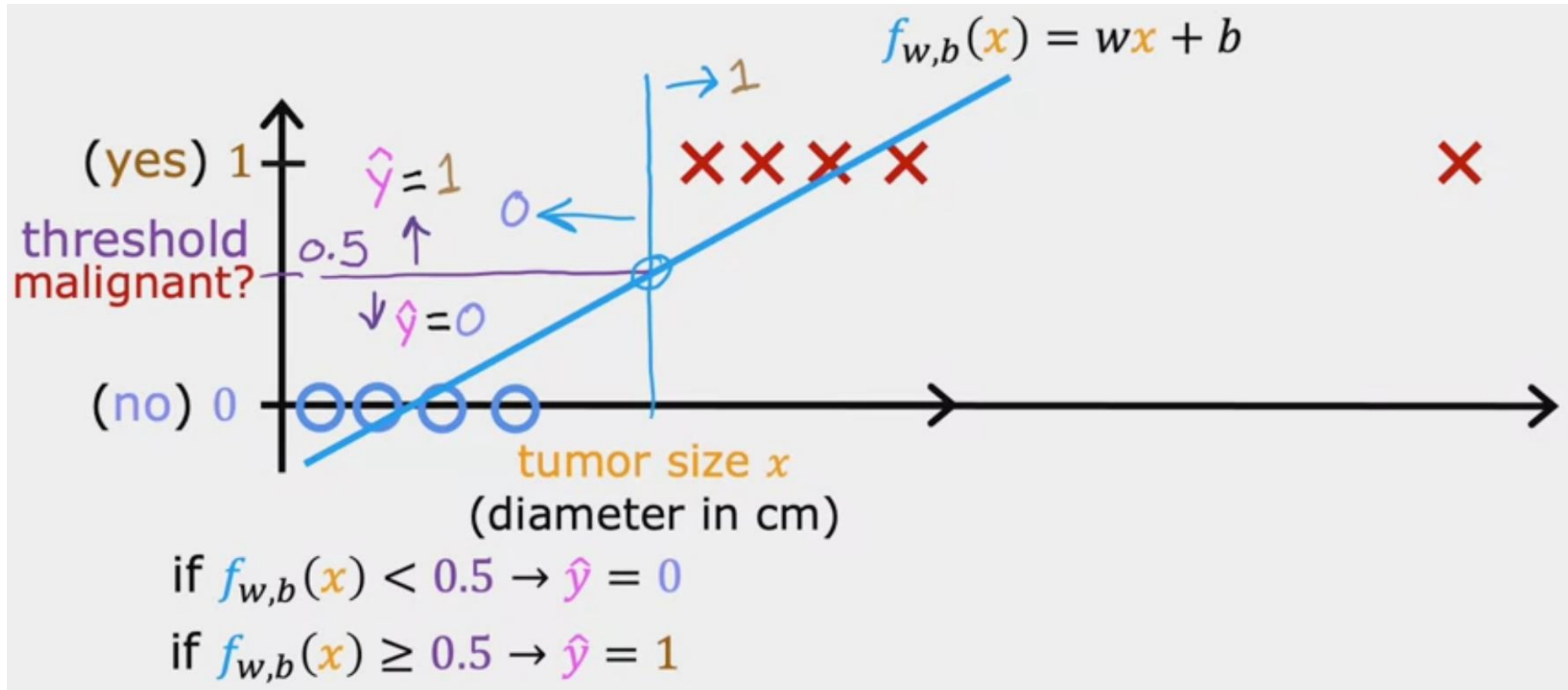
Classification – Motivation



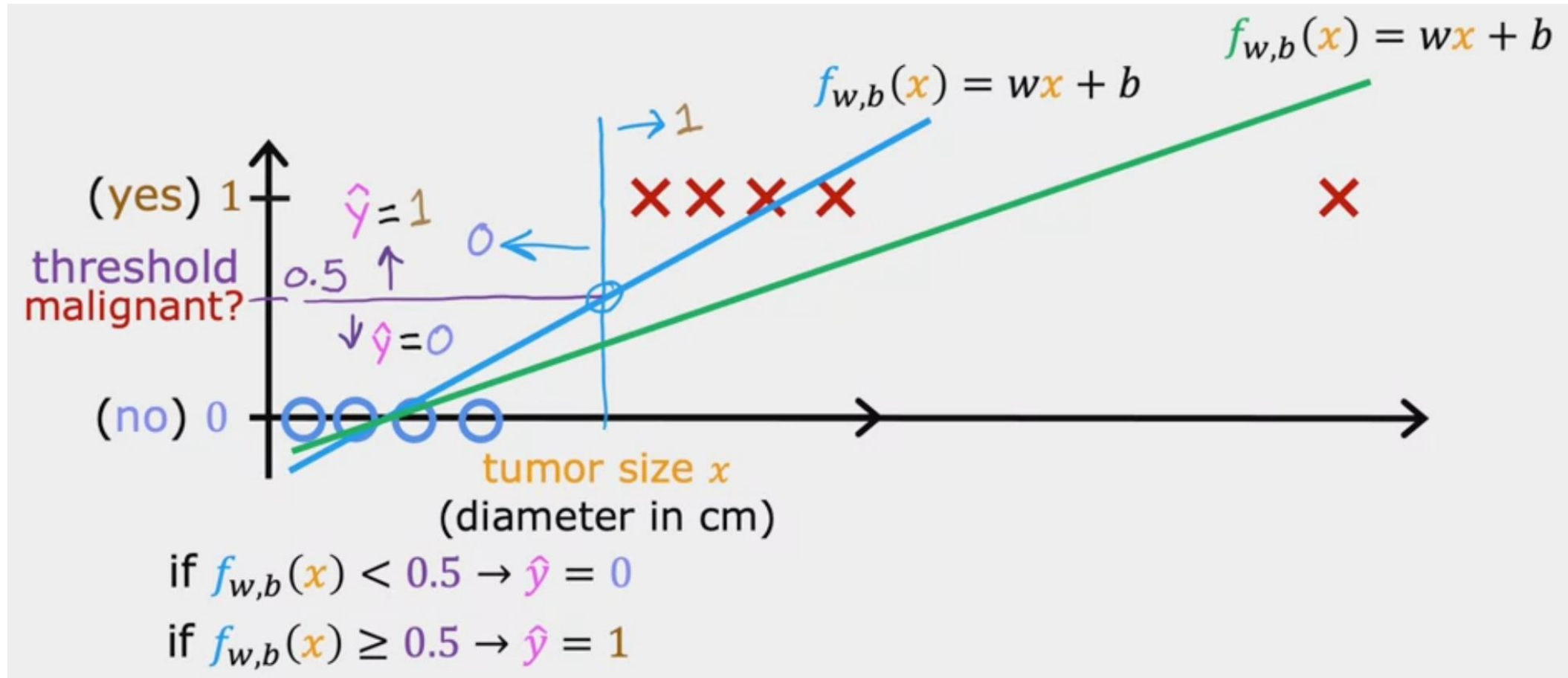
Classification – Motivation



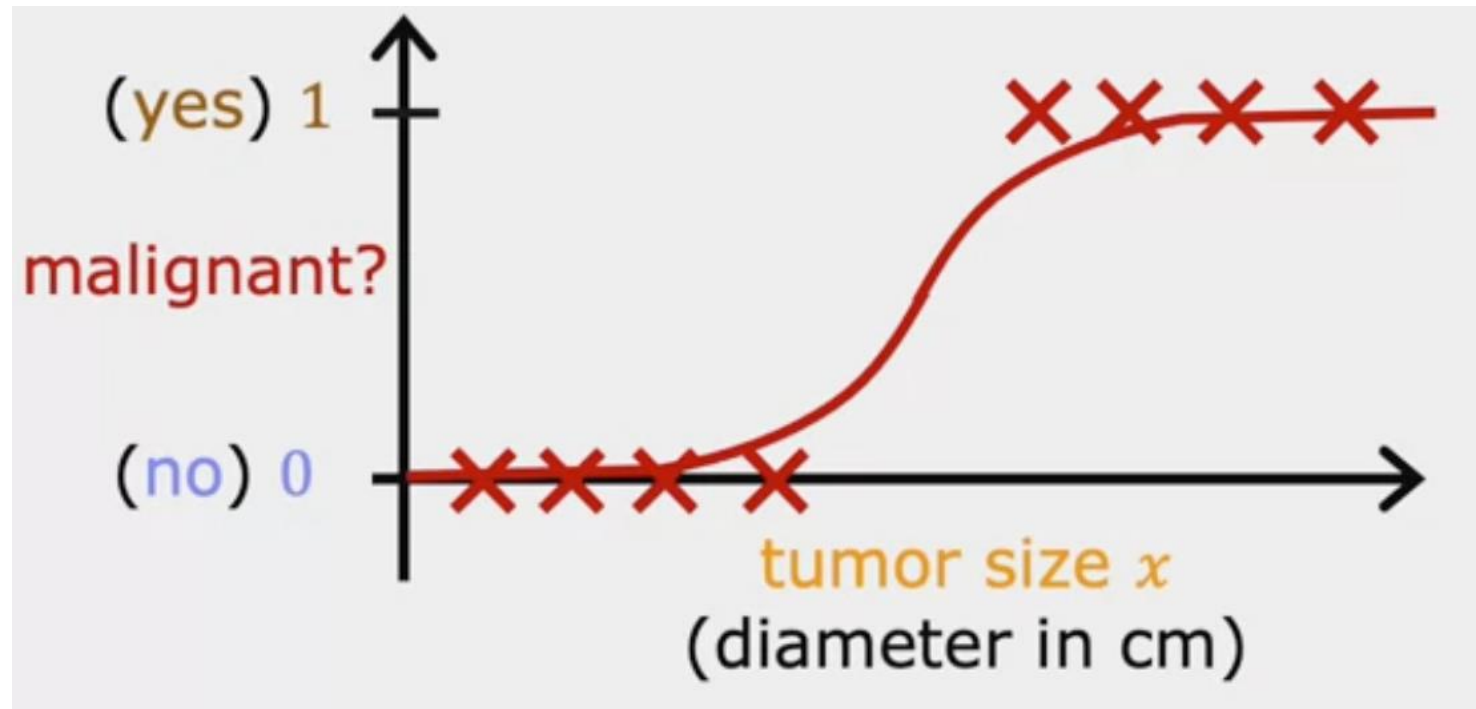
Classification – Motivation



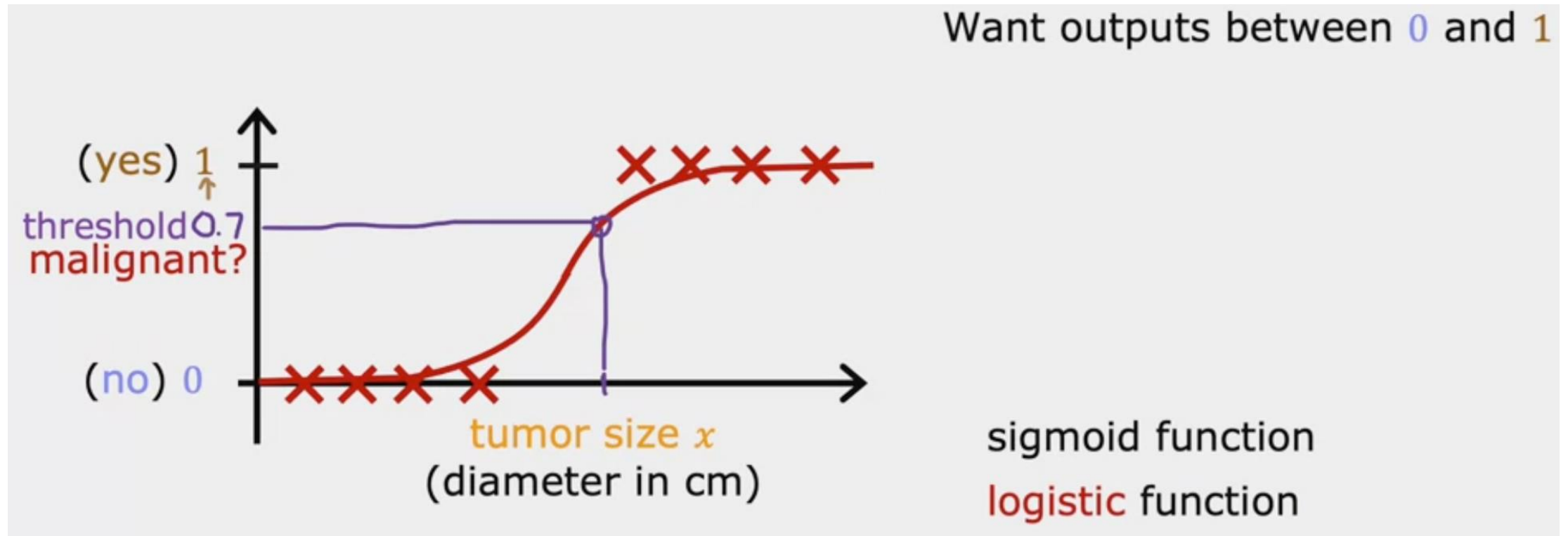
Classification – Motivation



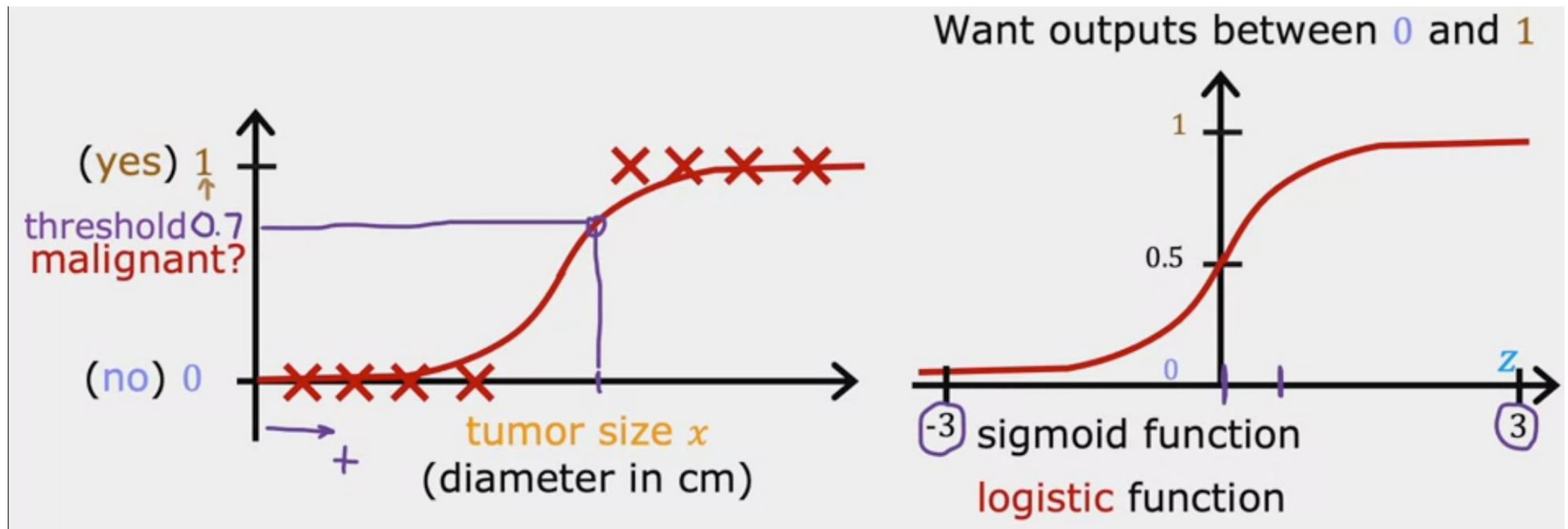
Classification – Logistic Regression



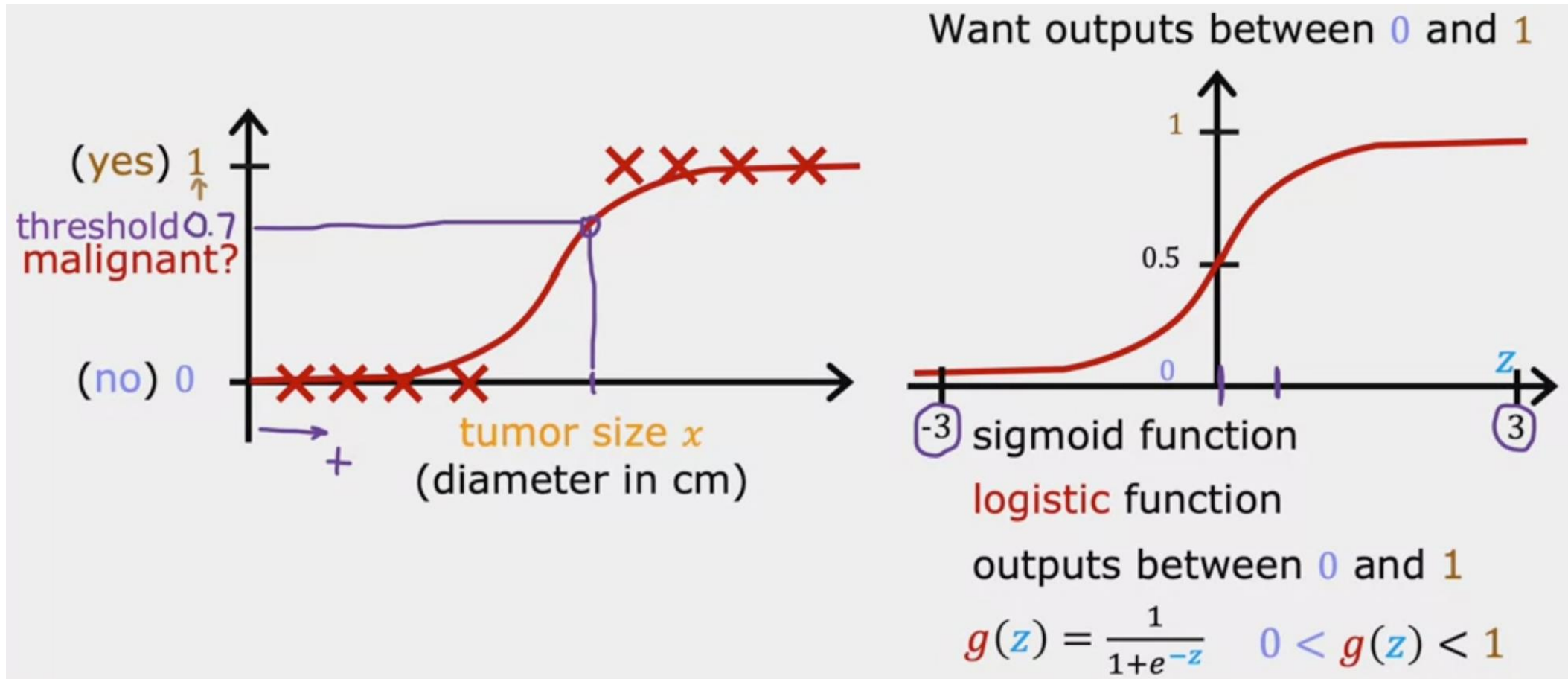
Classification – Logistic Regression



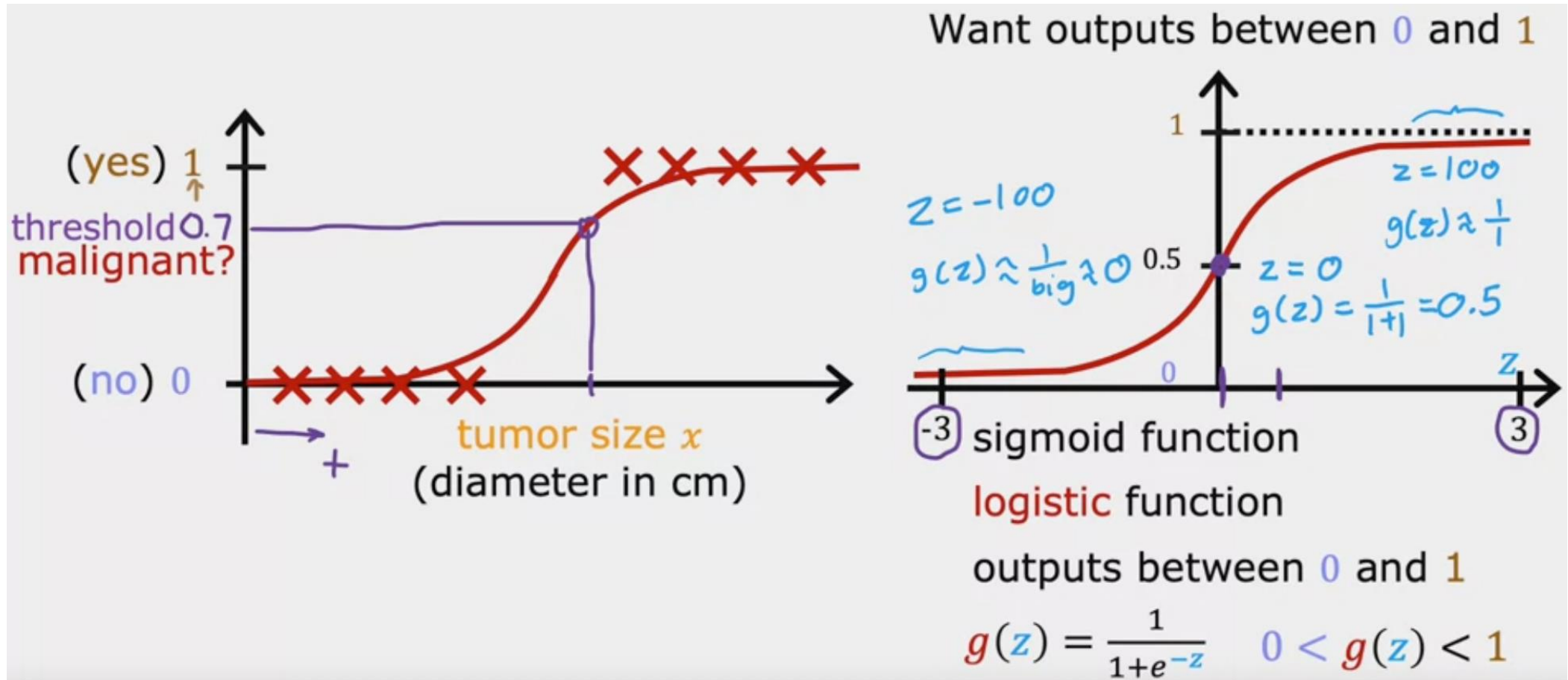
Classification – Logistic Regression



Classification – Logistic Regression

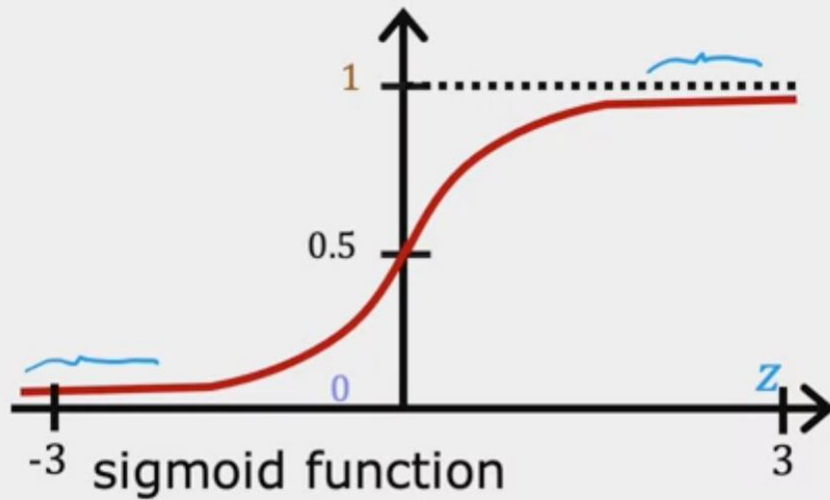


Classification – Logistic Regression



Classification – Logistic Regression

Want outputs between 0 and 1



logistic function

outputs between 0 and 1

$$g(z) = \frac{1}{1+e^{-z}} \quad 0 < g(z) < 1$$

$f_{\vec{w},b}(\vec{x})$

$$z = \vec{w} \cdot \vec{x} + b$$

\downarrow
 z
 \downarrow

$$g(z) = \frac{1}{1+e^{-z}}$$

$$f_{\vec{w},b}(\vec{x}) = g(\underbrace{\vec{w} \cdot \vec{x} + b}_z) = \frac{1}{1 + e^{-(\vec{w} \cdot \vec{x} + b)}}$$

"logistic regression"

Classification – Logistic Regression

Interpretation of logistic regression output

$$f_{\vec{w},b}(\vec{x}) = \frac{1}{1 + e^{-(\vec{w} \cdot \vec{x} + b)}}$$

“probability” that class is 1

Example:

x is “tumor size”

y is 0 (not malignant)
or 1 (malignant)

$$f_{\vec{w},b}(\vec{x}) = 0.7$$

70% chance that y is 1

$$f_{\vec{w},b}(\vec{x}) = P(y = 1 | \vec{x}; \vec{w}, b)$$

Probability that y is 1,
given input \vec{x} , parameters \vec{w}, b

$$P(y = 0) + P(y = 1) = 1$$