Day-85, Feb 23, 2025 (Falgun 11, 2081 B.S.)

1. Regression with a perceptron
2. Regression with a perceptron - loss function
3. Regression with a perceptron - Gradient Descent
4. Classification with Perceptron
5. Classification with perceptron - the Sigmoid function
6. Classification with perceptron - Gradient Descent
7. Classification with " → Calculating the derivatives

Regression with a perceptron

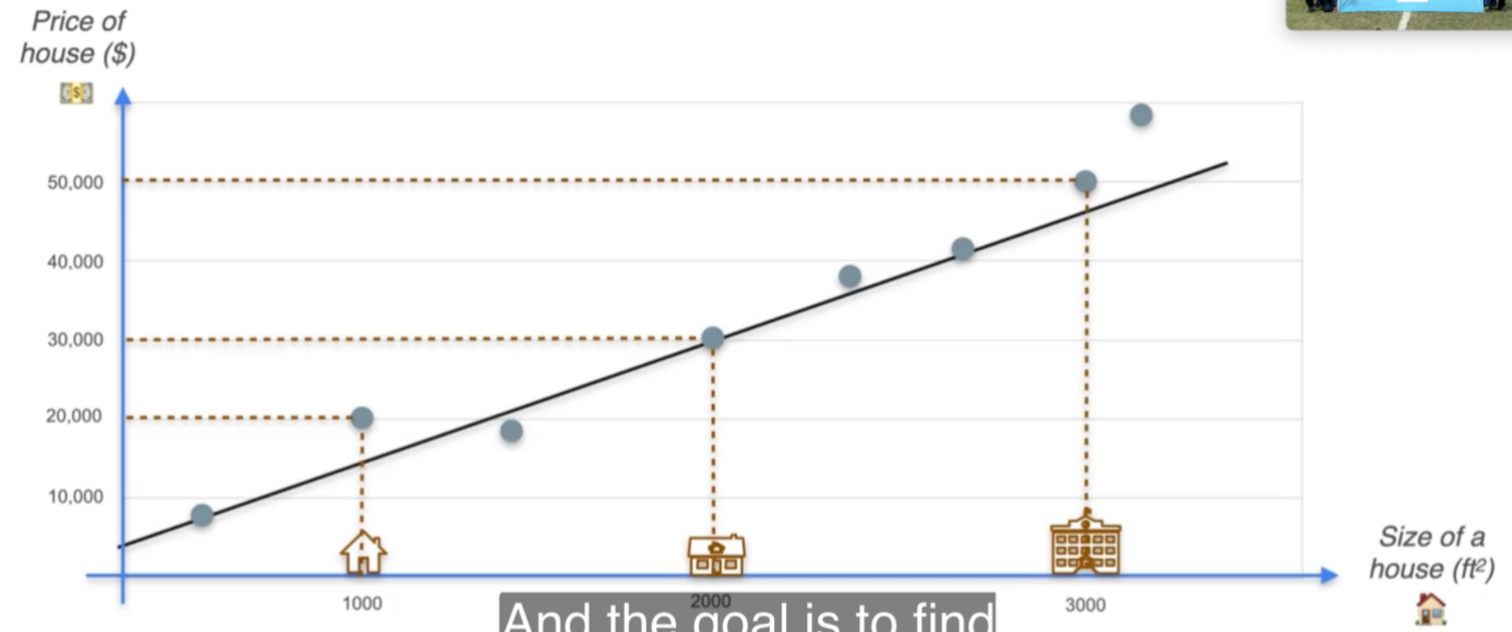
Regression With a Perceptron

	Size of a house (ft^2)	Number of rooms	Price of house ($)
	$1000ft^2$	2	$20,000
	$2000ft^2$	4	$30,000
	$3000ft^2$	7	$50,000

DeepLearning.AI

Well here's where the perception comes in.

Regression Problem Motivation



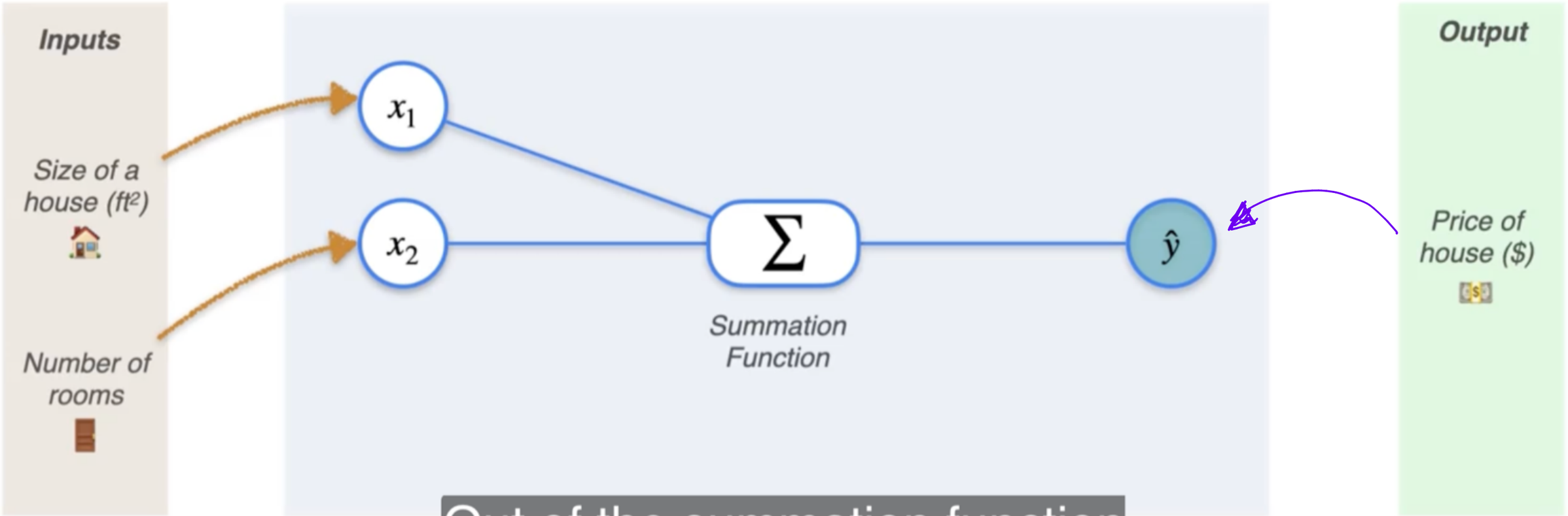
And the goal is to find the best possible line.

DeepLearning.AI

Main goal:
find weights and bias that will optimize the predictions.
i.e. Reduce the errors in the predictions using loss function.

Regression With a Perceptron

Single Layer Neural Network Perceptron

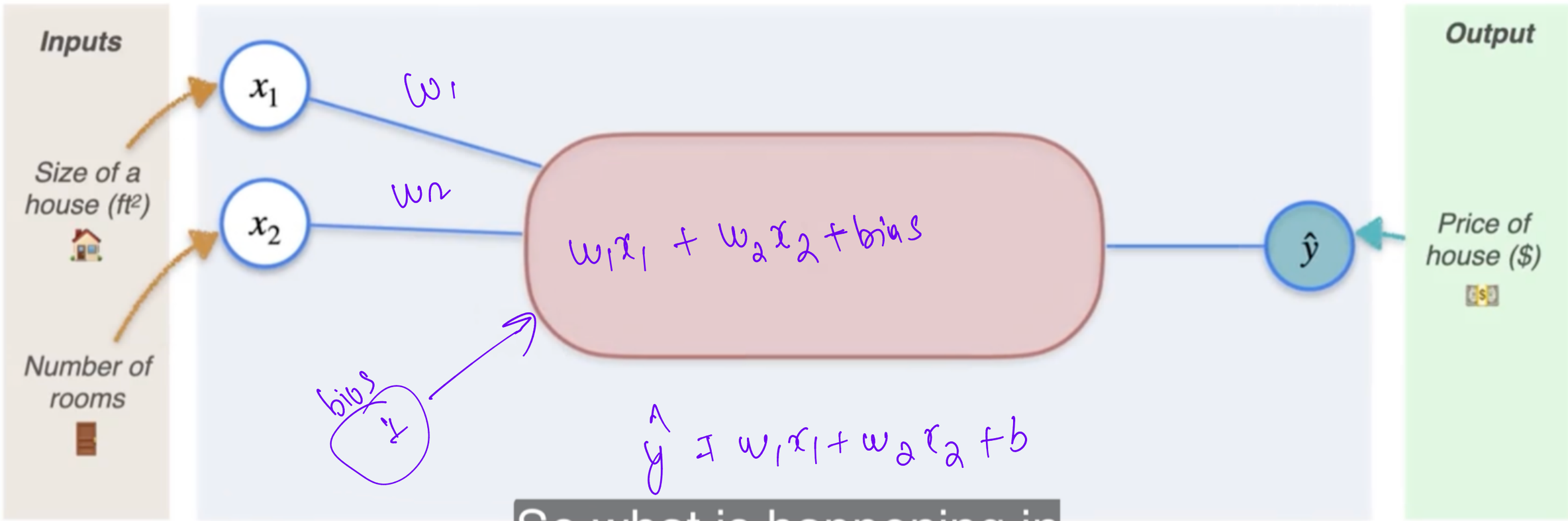


Out of the summation function comes the output y hat and

DeepLearning.AI

Regression With a Perceptron

Single Layer Neural Network Perceptron



So what is happening in this summation step?

DeepLearning.AI

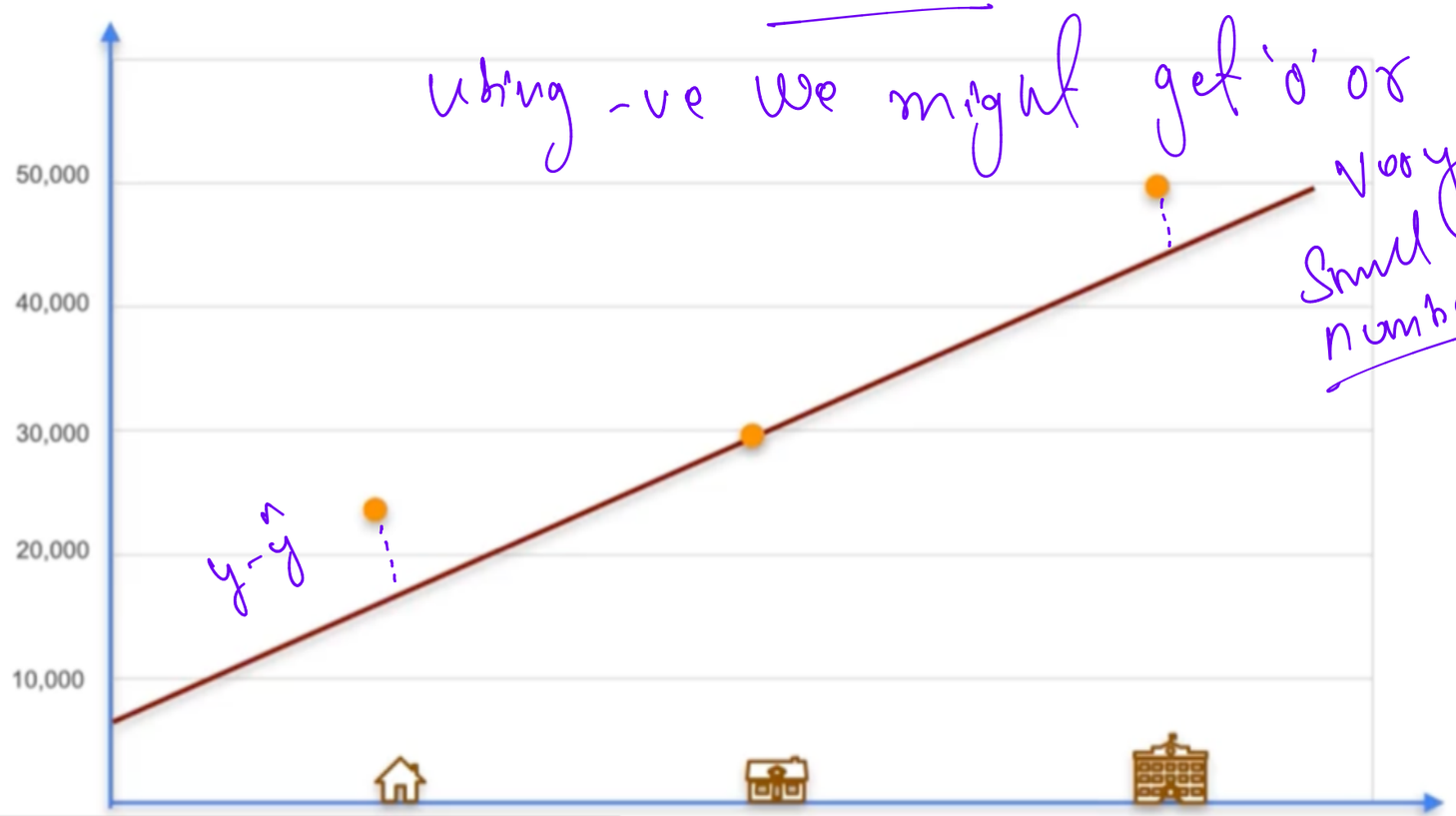
problem with MSE is if

+ve ↑ ↑

↓ -ve.

Mean Squared Error

	y	$\hat{y}$	Error $y - \hat{y}$
	$20,000	$15,000	5000
	$30,000	$30,000	0
	$50,000	$45,000	5000



to avoid -ve we do squaring.

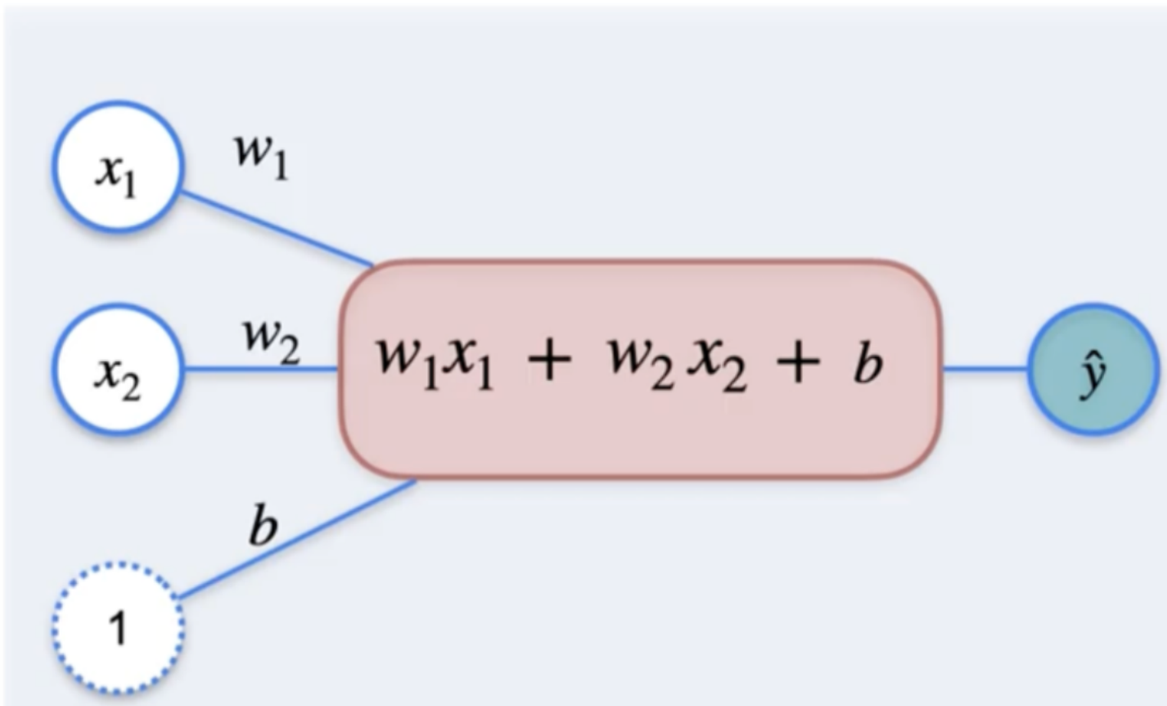
using -ve we might get '0' or very small number.

the first house is at height 15,000 and the

DeepLearning.AI

Regression With a Perceptron

Single Layer Neural Network Perceptron



Prediction Function:

$$\hat{y} = w_1x_1 + w_2x_2 + b$$

loss function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

Main Goal:

find w_1, w_2, b such that we get $\hat{y}$ with the least error.

outputs some prediction y and is given by this formula.

DeepLearning.AI

Best Model that makes Smallest mistakes!

Regression With a Perceptron

Prediction Function:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b$$

Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

Main Goal:

Find w_1 , w_2 , b that give $\hat{y}$ with the least error

To find optimal values for:

w_1, w_2, b

You need gradient descent

$$w_1 \to w_1 - \alpha \cdot \frac{\partial L}{\partial w_1}$$

$$w_2 \to w_2 - \alpha \cdot \frac{\partial L}{\partial w_2}$$

$$b \to b - \alpha \frac{\partial L}{\partial b}$$

Now we know what the problem is.

DeepLearning.AI

Regression With a Perceptron

Prediction Function:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b$$

Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

Main Goal:

Find w_1 , w_2 , b that give $\hat{y}$ with the least error

To find optimal values for:

w_1 , w_2 , b

You need gradient descent

Some initial starting values

$$w_1 \rightarrow w_1 - \alpha \frac{\partial L}{\partial w_1}$$

$$w_2 \rightarrow w_2 - \alpha \frac{\partial L}{\partial w_2}$$

$$b \rightarrow b - \alpha \frac{\partial L}{\partial}$$

SUB-TASK

Find the following partial derivatives

a very low loss function and that implies a good model.

DeepLearning.AI

Regression With Perceptron

Prediction Function:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b$$

Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

depends on

Using chain rule:

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$

$$\frac{\partial L}{\partial w_1} \quad \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1}$$

$$\frac{\partial L}{\partial w_2} \quad \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$

First of all, what is dL over db?

DeepLearning.AI

Regression With Perceptron

Prediction Function:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b$$

Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

Using chain rule:

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2}$$

actually plugging in the whole thing.

DeepLearning.AI

Let's find the $\frac{\partial L}{\partial \hat{y}}$, $\frac{\partial \hat{y}}{\partial w_1}$ and $\frac{\partial \hat{y}}{\partial w_2}$ then we get,

$$\frac{\partial L}{\partial \hat{y}} = -(y - \hat{y})$$

$$\frac{d}{d\hat{y}}(y - \hat{y})^2 = 2(y - \hat{y}) \cdot \frac{d}{d\hat{y}}(y - \hat{y})$$

$$\Rightarrow \frac{1}{2} \cdot 2(y - \hat{y}) \cdot (-1)$$

100DaysOfMaths_@dilli_hangrae

Regression With Perceptron

Prediction Function:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b$$

Loss Function:

$$L(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$$

$\frac{\partial L}{\partial \hat{y}}$	$= -(y - \hat{y})$
$\frac{\partial \hat{y}}{\partial b}$	$= 1$
$\frac{\partial \hat{y}}{\partial w_1}$	$= x_1$
$\frac{\partial \hat{y}}{\partial w_2}$	$= x_2$

the constant accompany w_2.

Regression With Perceptron

$$\frac{\partial L}{\partial \hat{y}} = -(y - \hat{y})$$

$$\frac{\partial \hat{y}}{\partial b} = 1$$

$$\frac{\partial \hat{y}}{\partial w_1} = x_1$$

$$\frac{\partial \hat{y}}{\partial w_2} = x_2$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b} \qquad -(y-\hat{y}) \cdot (1)$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} \qquad -(y-\hat{y}) \cdot x_1$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} \qquad -(y-\hat{y}) \cdot x_2$$

DeepLearning.AI

dL over db is dL over dy hat,

Regression With a Perceptron

Main Goal:

Find w_1 , w_2 , b that give $\hat{y}$ with the least error

ie. optimal values for:
w_1 , w_2 , b

Perform Gradient Descent ?

$$w_1 \to w_1 - \alpha \frac{\partial L}{\partial w_1}$$

$$\Rightarrow w_1 - \alpha \left(-x_1 (y - \hat{y}) \right)$$

$$w_2 \to w_2 - \alpha \left(-x_2 (y - \hat{y}) \right)$$

$$b \to b - \alpha \left(-(y - \hat{y}) \right)$$

the predictions y had with
the smallest possible error.

DeepLearning.AI

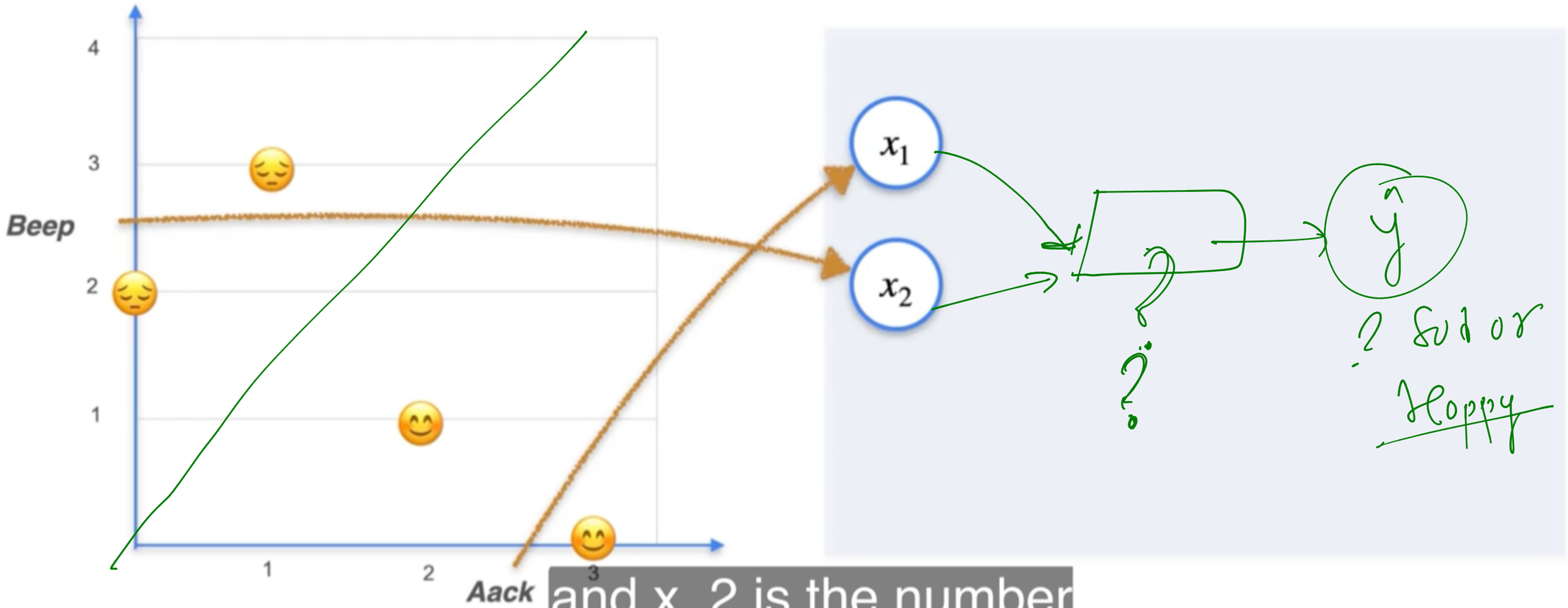
Classification Problem Motivation

Sentence	Aack	Beep	Mood
Aack aack aack!	3	0	Happy
Beep beep!	0	2	Sad
Aack beep beep beep!	1	3	Sad
Aack beep aack!	2	1	Happy

to tell the mood of
the each sentence.

DeepLearning.AI

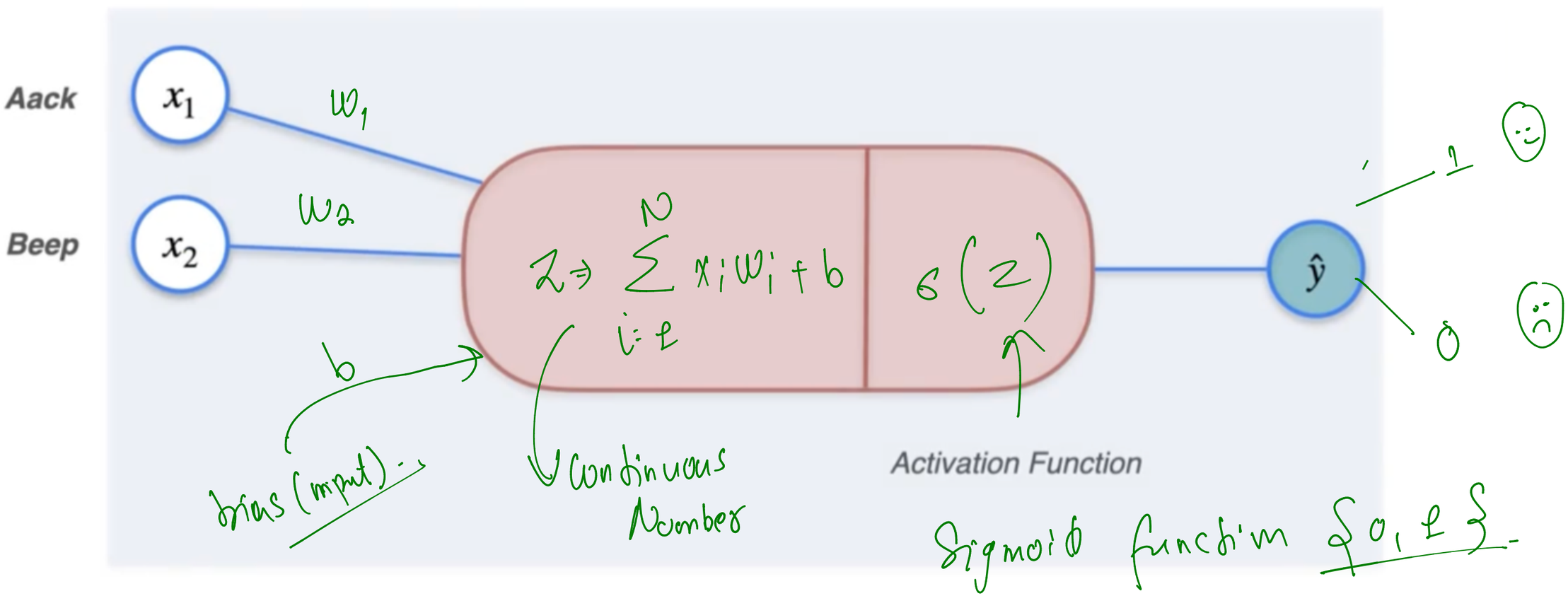
Classification Problem Motivation



and x_2 is the number of times beep appears.

DeepLearning.AI

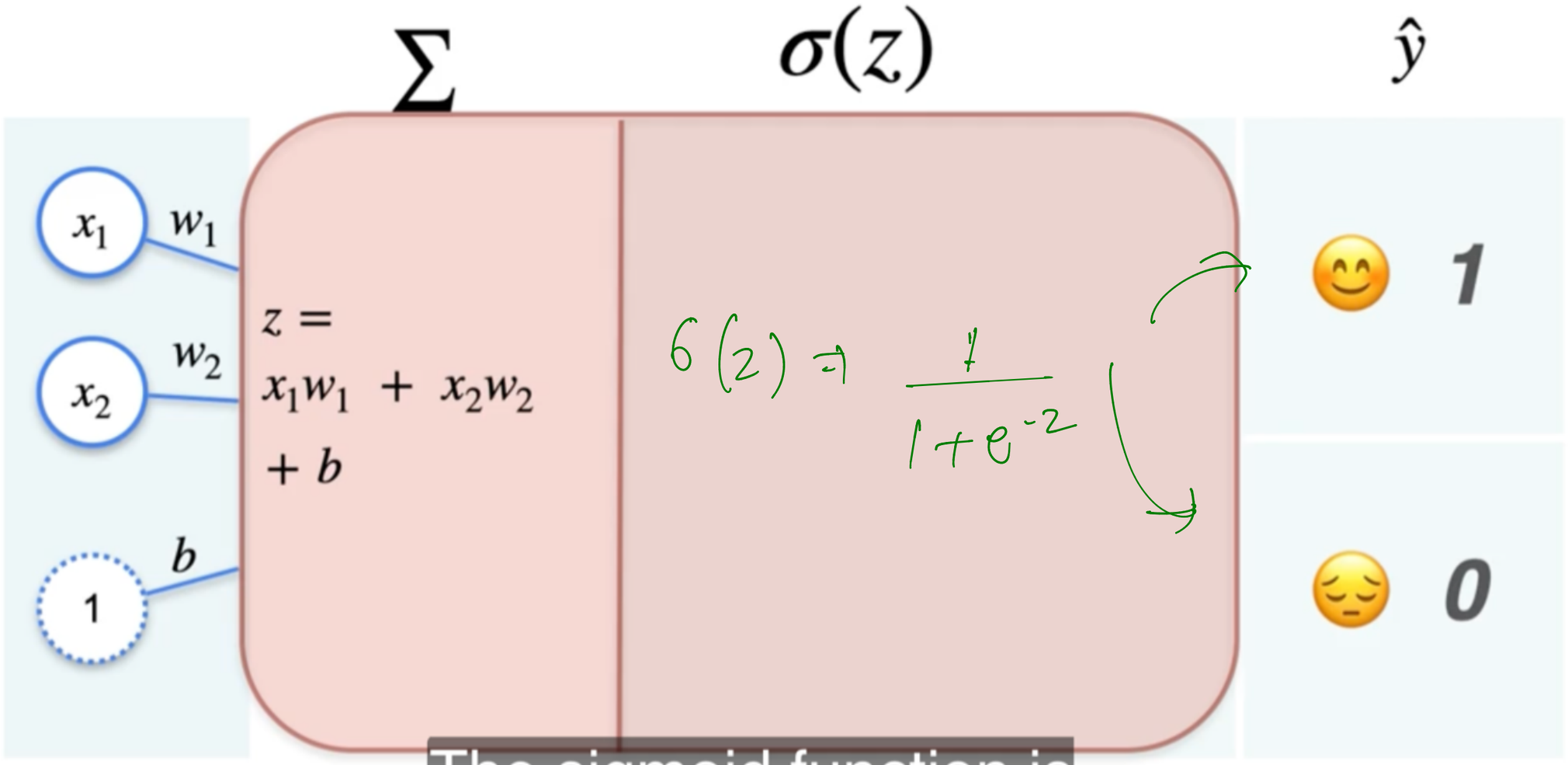
Classification With a Perceptron



it's going to be very similar to

DeepLearning.AI

Sigmoid Function



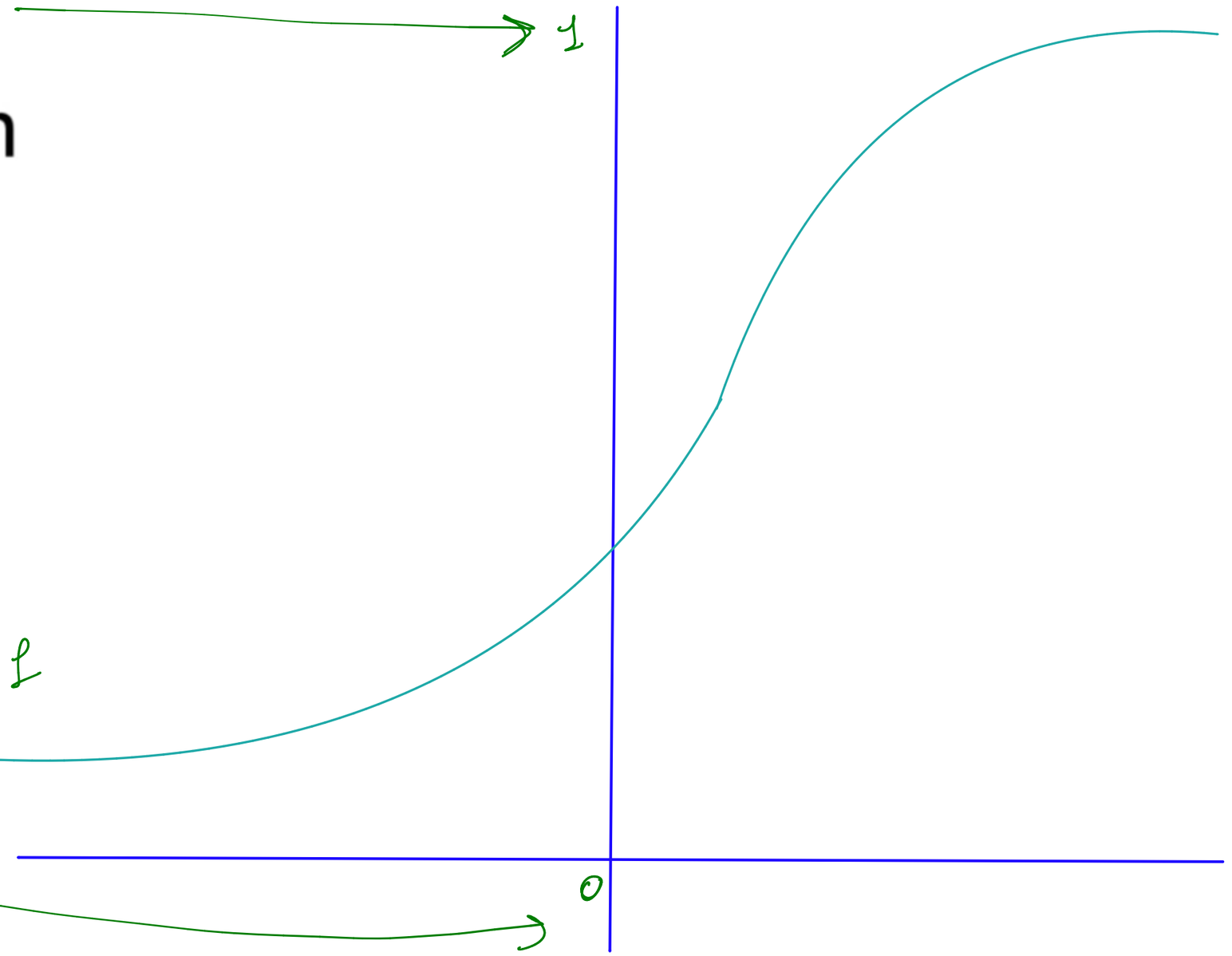
The sigmoid function is given by the formula

Sigmoid Function

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

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

$$\frac{d}{dz}\sigma(z) = \frac{d}{dz}(1 + e^{-z})^{-1}$$



the entire number line and

$$\frac{d}{dz}\,\sigma(z) = -1(1+e^{-z})^{-1-1}\left(\frac{d}{dz}(1+e^{-z})\right)$$

$$\Rightarrow -1(1+e^{-z})^{-2}\left(\frac{d}{dz}(1)+\frac{d}{dz}(e^{-z})\right)$$

$$\Rightarrow -1(1+e^{-z})^{-2}\left(0+e^{-z}\left(\frac{d}{dz}(-z)\right)\right)$$

$$\left[\frac{d}{dz}\,\sigma(z) \Rightarrow -1(1+e^{-z})^{-2}(e^{-z})(-1)\right]$$

$$\Rightarrow (1+e^{-z})^{-2}(e^{-z})$$

$$\Rightarrow \frac{1}{(1+e^{-z})^{2}}(e^{-z}) \Rightarrow \frac{e^{-z}}{(1+e^{-z})^{2}}$$

$$\frac{d}{dz}\sigma(z) \Rightarrow \frac{e^{-z} + 1 - 1}{(1+e^{-z})^2}$$

$$\Rightarrow \frac{1 + e^{-z} - 1}{(1+e^{-z})^2}$$

$$\Rightarrow \frac{1+e^{-z}}{(1+e^{-z})^2} - \frac{1}{(1+e^{-z})^2}$$

$$\Rightarrow \frac{1}{(1+e^{-z})} - \left(\frac{1}{1+e^{-z}}\right)\left(\frac{1}{1+e^{-z}}\right)$$

$$\Rightarrow \frac{1}{1+e^{-z}}\left(1 - \frac{1}{1+e^{-z}}\right)$$

So, $\sigma(z) = \frac{1}{1+e^{-z}}$

$\frac{d}{dz}\sigma(z) = \sigma(z)\,(1-\sigma(z))$

ative of a Sigmoid Function

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

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

$$\frac{d}{dz}\sigma(z) = \frac{d}{dz}(1+e^{-z})^{-1}$$

$$\frac{d}{dz}\sigma(z) = -1\ (1+e^{-z})^{-1-1}\ (\frac{d}{dz}(1+e^{-z}))$$

$$= -1\ (1+e^{-z})^{-2}\ (\frac{d}{dz}(1)\ + \frac{d}{dz}(e^{-z}))$$

$$= -1\ (1+e^{-z})^{-2}\ (0\ + e^{-z}(\frac{d}{dz}(-z)))$$

$$= -1\ (1+e^{-z})^{-2}\ (e^{-z})\ (-1)$$

We're about halfway there.

ative of a Sigmoid Function

$$\frac{d}{dz}\sigma(z) = -1\ (1+e^{-z})^{-2}\ (e^{-z})\ (-1)$$

$$= (1+e^{-z})^{-2}\ (e^{-z})$$

$$= \frac{1}{(1+e^{-z})^2}\ (e^{-z})$$

$$= \frac{e^{-z}}{(1+e^{-z})^2}$$

1 plus e^negative z squared.

ative of a Sigmoid Function

$$\frac{d}{dz}\sigma(z) = \frac{e^{-z} + 1 - 1}{(1+e^{-z})^2}$$

$$= \frac{1+e^{-z}-1}{(1+e^{-z})^2}$$

$$= \frac{1+e^{-z}}{(1+e^{-z})^2} - \frac{1}{(1+e^{-z})^2}$$

$$= \frac{1}{(1+e^{-z})} - \frac{1}{(1+e^{-z})^2}$$

$$\frac{d}{dz}\sigma(z) = \frac{1}{(1+e^{-z})} - \left(\frac{1}{(1+e^{-z})}\right)\left(\frac{1}{(1+e^{-z})}\right)$$

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

Recall that: $\sigma(z) = \frac{1}{1+e^{-z}}$

$$\frac{d}{dz}\sigma(z) = \sigma(z)\,(1-\sigma(z))$$

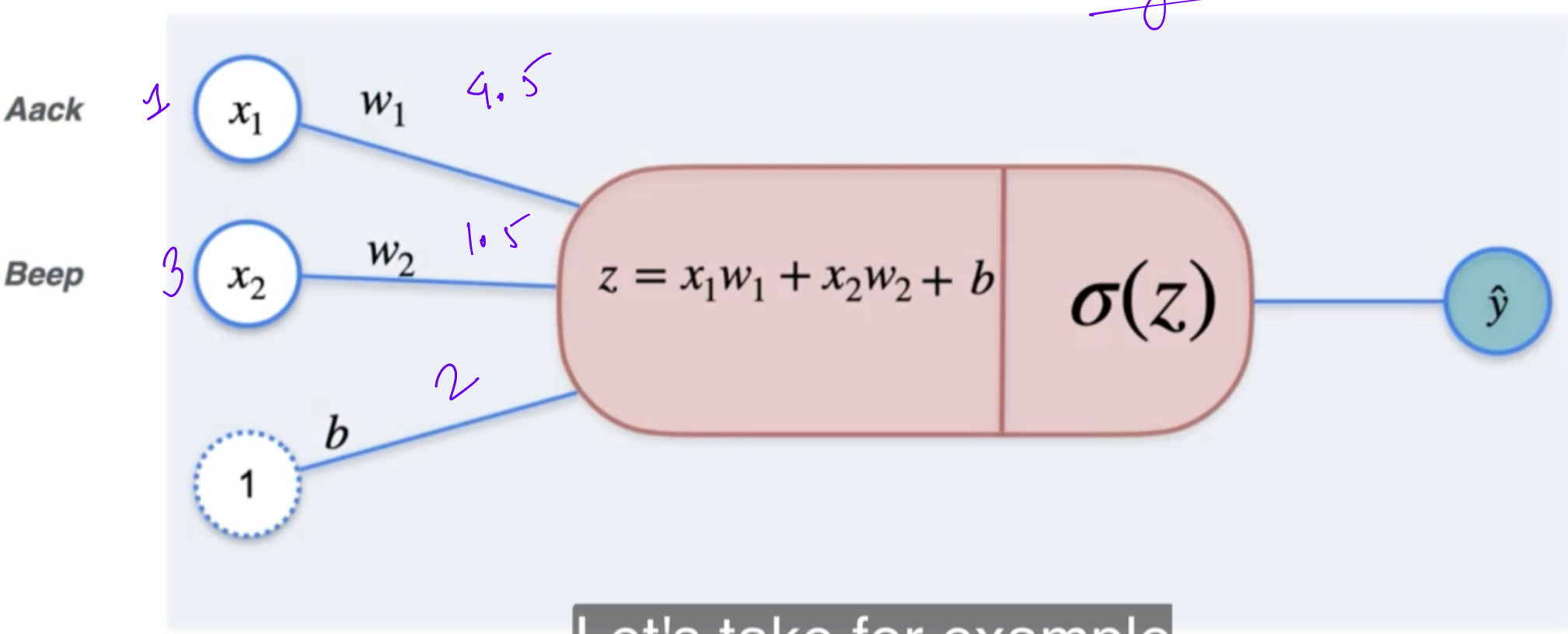
The derivative of sigmoid is

DeepLearning.AI

Classification With a Perceptron

Aack beep beep beep

Logloss

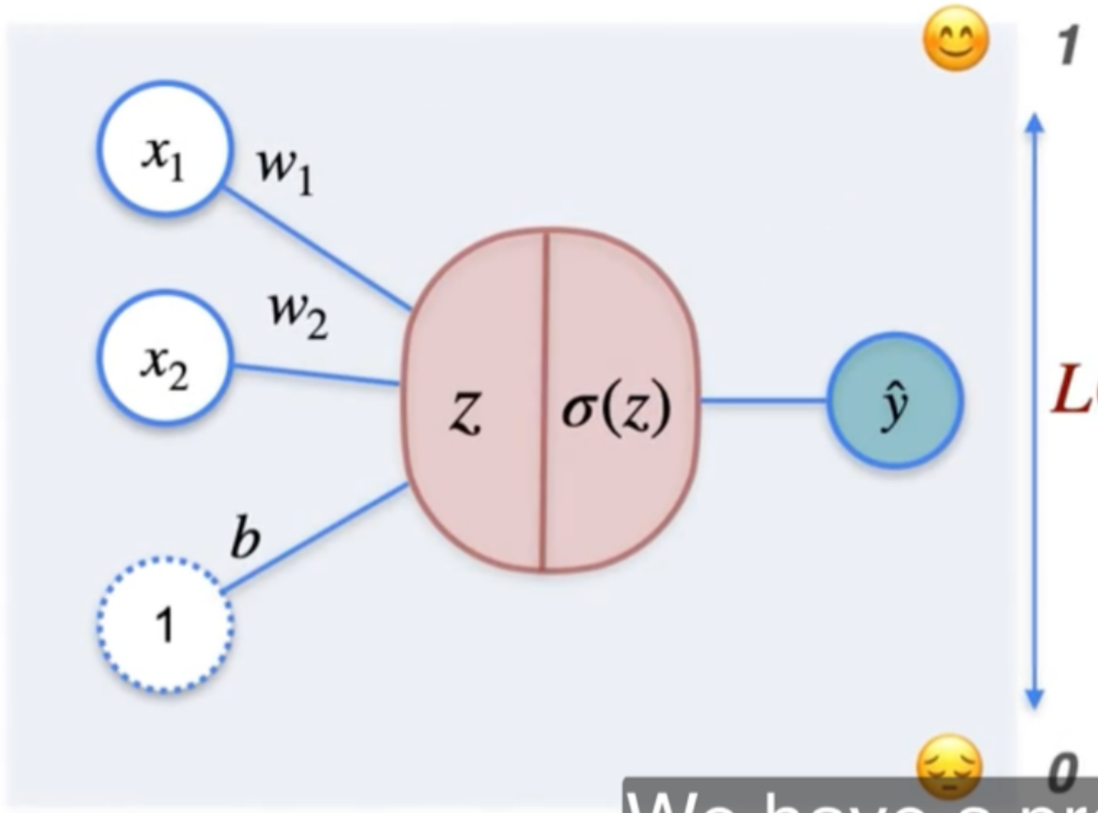


Let's take for example
the sentence Aack,

1:51

DeepLearning.AI

Classification With a Perceptron



Prediction function:

$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$$

Loss function:

$$\mathcal{L}(y, \hat{y}) \Rightarrow -y \ln(\hat{y}) - (1-y)\ln(1-\hat{y})$$

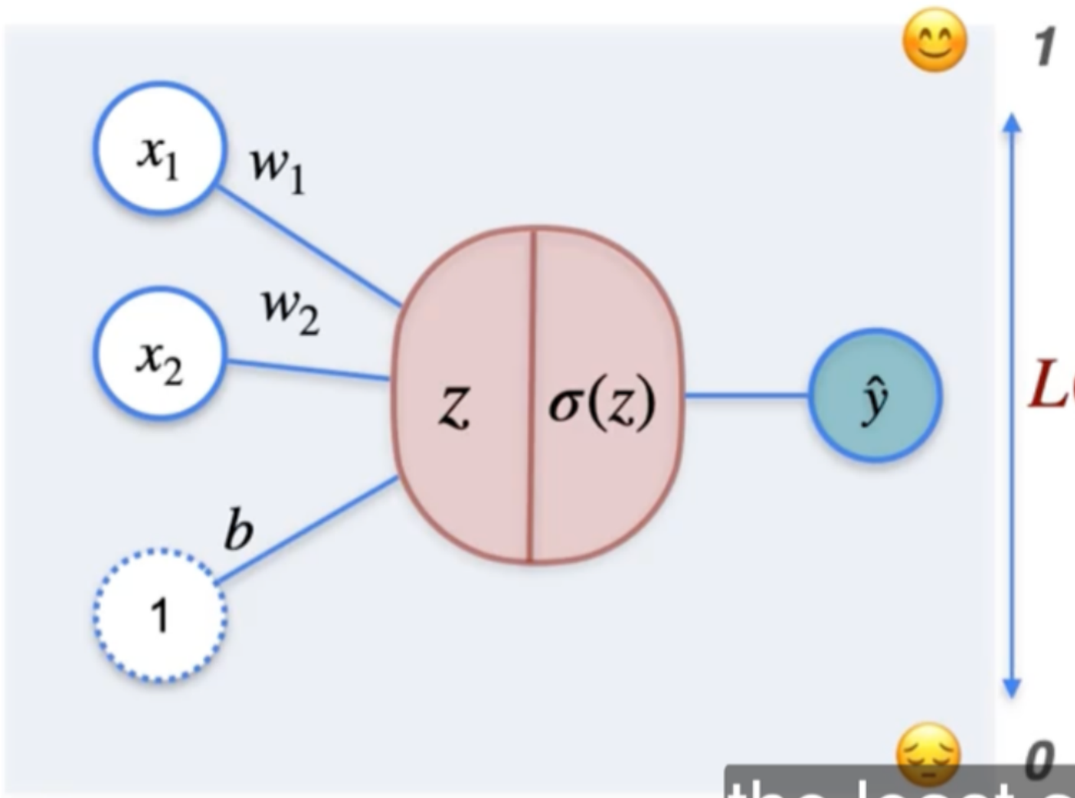
$\Rightarrow$ Main Goal find w_1, w_2 & b to minimize Loss function.

Probabilistic Nature: Of classification

We have a prediction function y-hat with an activation of

DeepLearning.AI

Classification With a Perceptron



Prediction Function:

$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$$

Loss Function:

$L(y, \hat{y})$ $L(y, \hat{y}) = -y \ln(\hat{y}) - (1 - y)\ln(1 - \hat{y})$

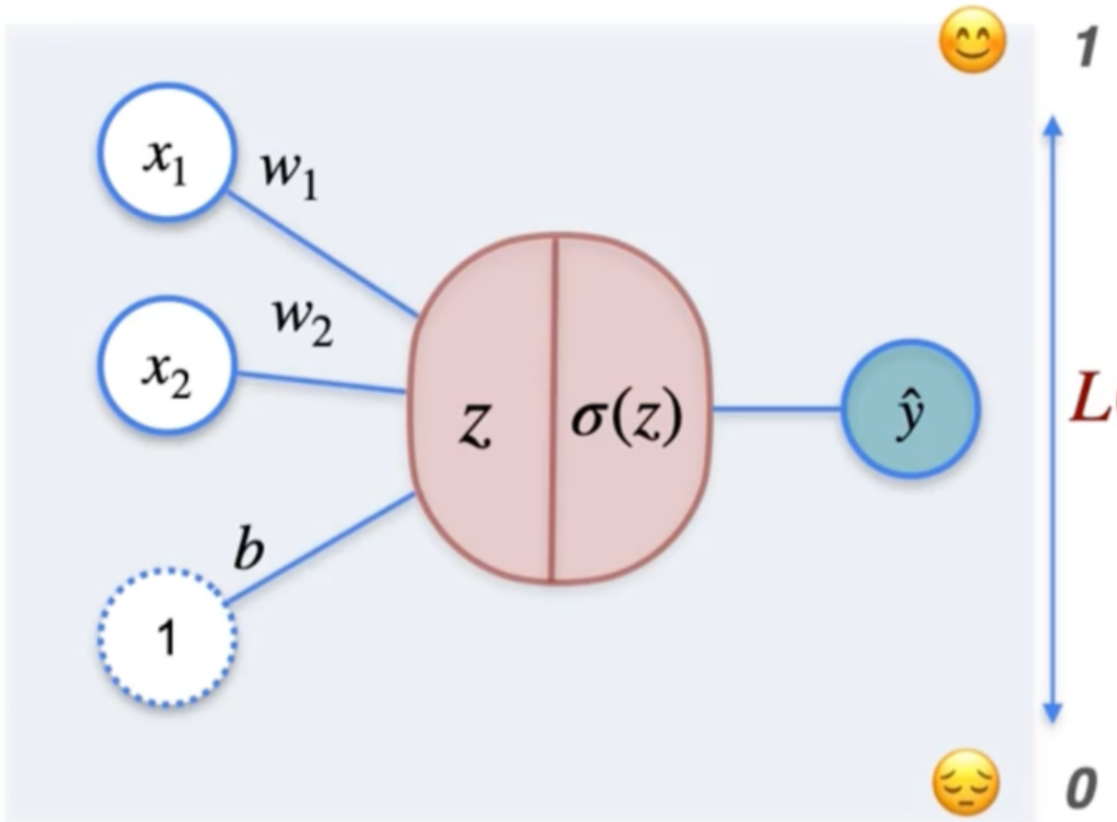
Main Goal:

Find w_1 , w_2 , b that give $\hat{y}$ with the least error

the least amount of log loss L(y, y-hat) error.

DeepLearning.AI

Classification With a Perceptron



To find optimal values for:

w_1, w_2, b

We need gradient descent:

$$w_1 \rightarrow w_1 - \alpha \frac{\partial L}{\partial w_1}$$

$$w_2 \rightarrow w_2 - \alpha \cdot \frac{\partial L}{\partial w_2}$$

$$b \rightarrow b - \alpha \cdot \frac{\partial L}{\partial b}$$

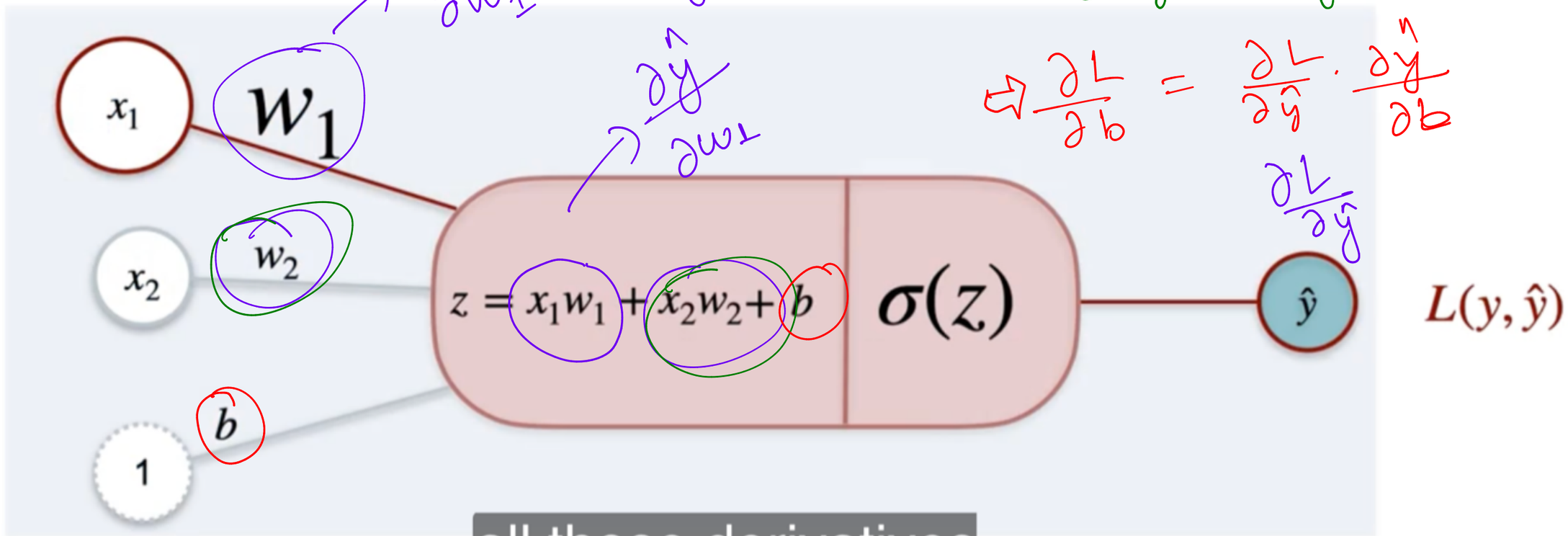
In order to be able to do this,

Classification with a Perceptron.

Classification With a Perceptron

$$\Rightarrow \frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} \qquad \Leftrightarrow \frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2}$$

$$\Leftrightarrow \frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$



all those derivatives
and be able to do

Classification With a Perceptron

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2}$$

$\frac{\partial L}{\partial \hat{y}} =$

$\frac{\partial \hat{y}}{\partial b} =$

$\frac{\partial \hat{y}}{\partial w_1} = ?$

$\frac{\partial \hat{y}}{\partial w_2} =$

$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$

$L(y, \hat{y}) = -y \ln(\hat{y}) - (1-y) \ln(1-\hat{y})$

d y Hat is there many times. What are these ones?

DeepLearning.AI

Classification With a Perceptron

$$\frac{\partial L}{\partial \hat{y}}$$

$$L(y, \hat{y}) = -y \ln(\hat{y}) - (1 - y)\ln(1 - \hat{y})$$

$$\frac{\partial L}{\partial \hat{y}} = -\frac{y}{\hat{y}} + \frac{1 - y}{1 - \hat{y}}$$

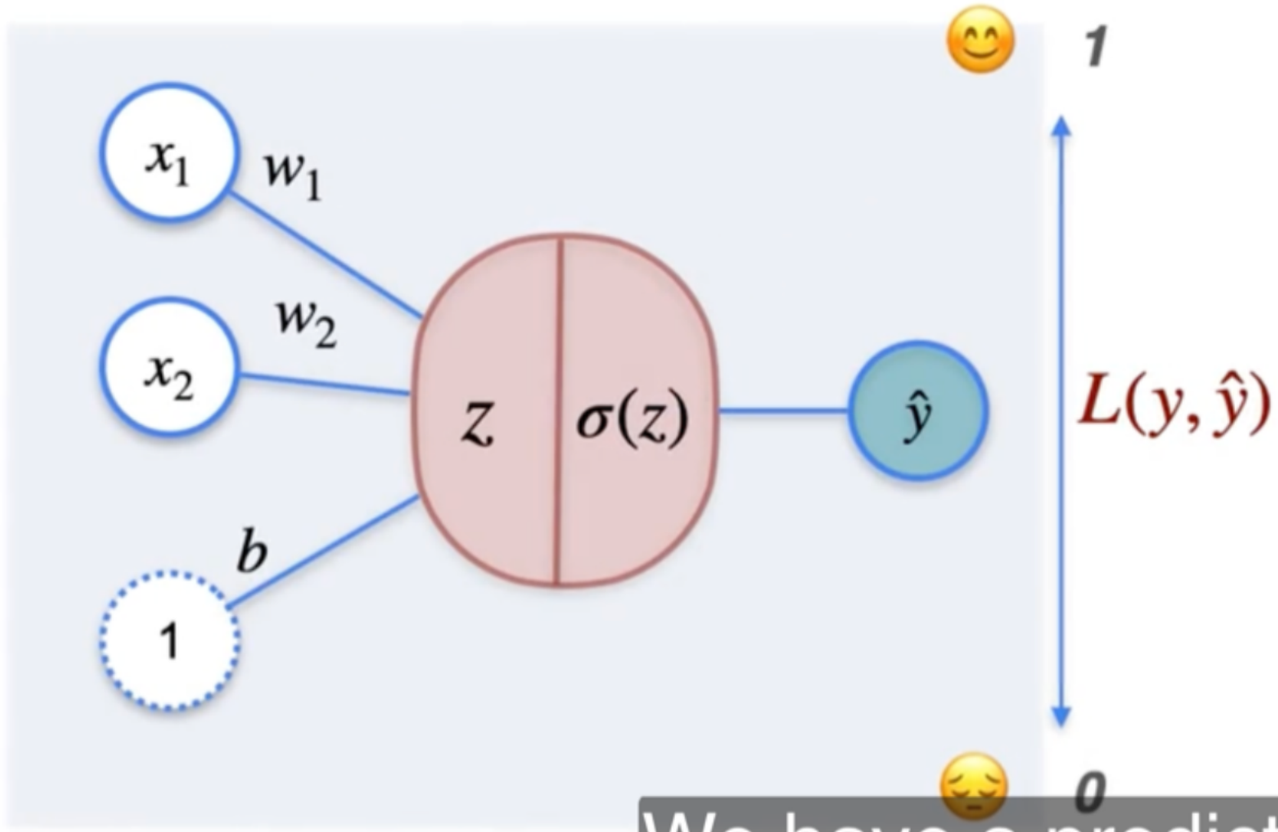
$$= \frac{-y + y\hat{y} + \hat{y} - y\hat{y}}{\hat{y}(1 - \hat{y})}$$

$$= -y(y - \hat{y})$$

Notice that the logarithm of

DeepLearning.AI

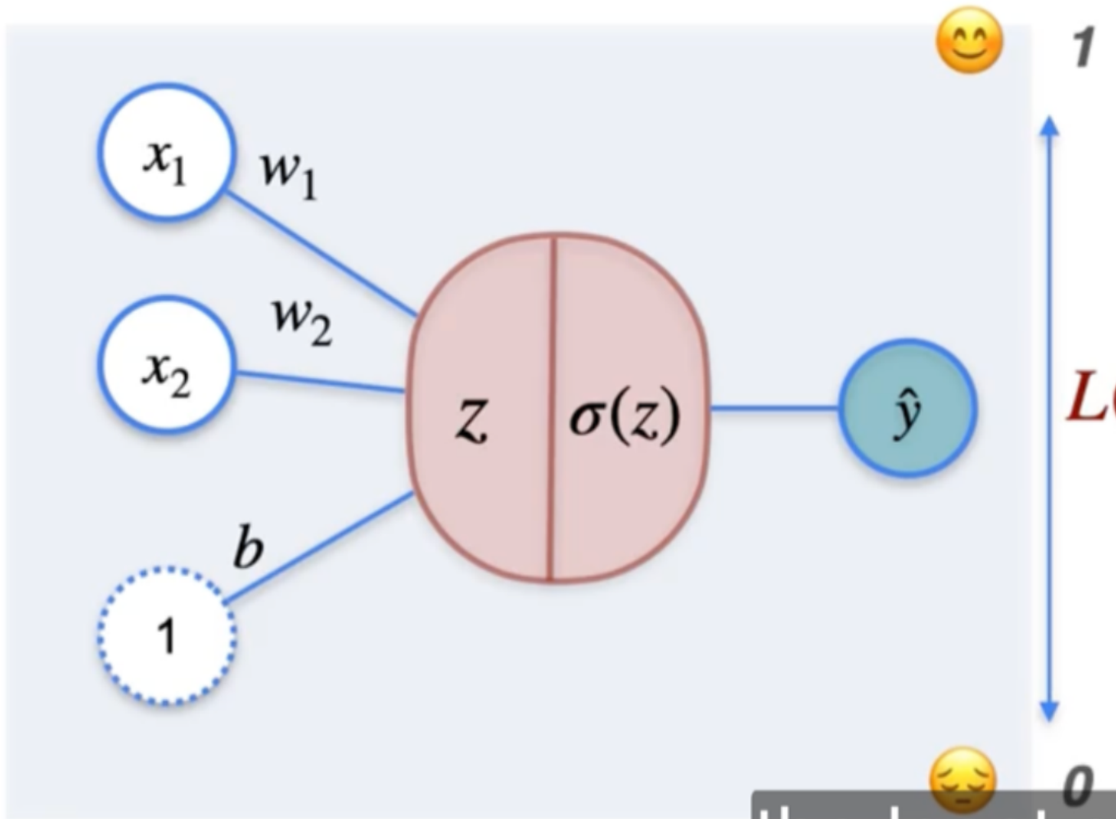
Classification With a Perceptron



We have a prediction function
y-hat with an activation of

DeepLearning.AI

Classification With a Perceptron



Prediction Function:

$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$$

Loss Function:

$L(y, \hat{y})$ $$L(y, \hat{y}) = -y \ln(\hat{y}) - (1 - y)\ln(1 - \hat{y})$$

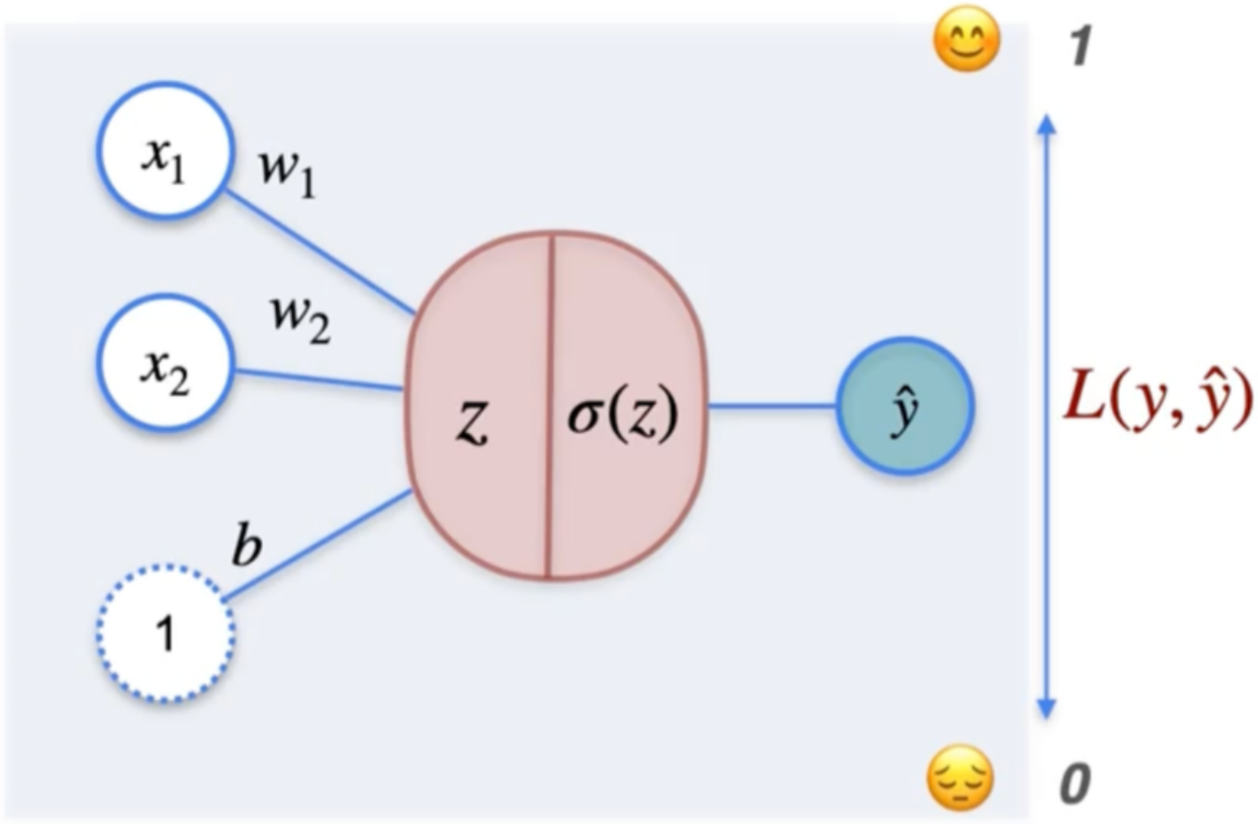
Main Goal:

Find w_1 , w_2 , b that give $\hat{y}$ with the least error

the least amount of log loss L(y, y-hat) error.

DeepLearning.AI

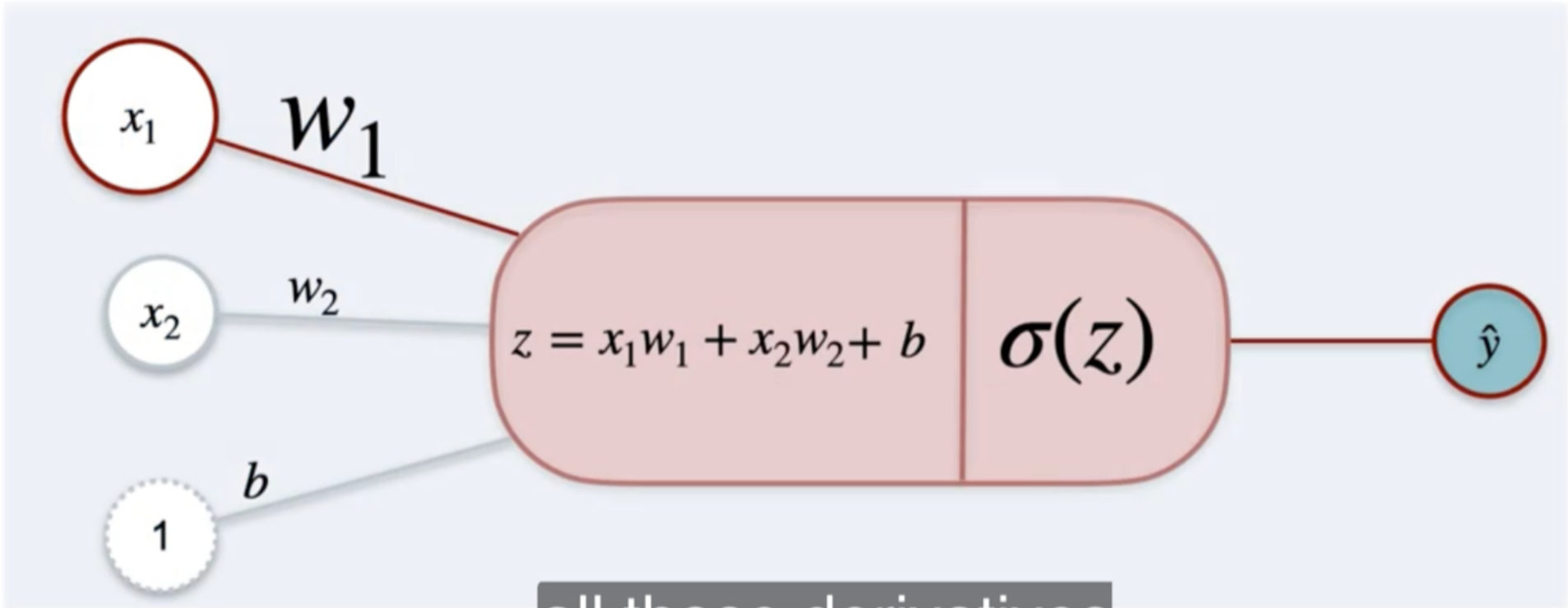
Classification With a Perceptron



In order to be able to do this,

DeepLearning.AI

Classification With a Perceptron



all those derivatives
and be able to do

DeepLearning.AI

Classification With a Perceptron

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b}$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2}$$

d y Hat is there many
times. What are these ones?

DeepLearning.AI

Classification With a Perceptron

$$\hat{y} = \sigma(w_1 x_1 + w_2 x_2 + b)$$

$\frac{\partial \hat{y}}{\partial w_1}$	$= \hat{y}(1-\hat{y})x_1$
$\frac{\partial \hat{y}}{\partial w_2}$	$= \hat{y}(1-\hat{y})x_2$
$\frac{\partial \hat{y}}{\partial b}$	$= \hat{y}(1-\hat{y})$

of something is sigmoid
times one minus sigmoid.

DeepLearning.AI

Classification With a Perceptron

$$\frac{\partial L}{\partial \hat{y}} = \frac{-(y - \hat{y})}{\hat{y}(1-\hat{y})}$$

$$\frac{\partial \hat{y}}{\partial b} = \hat{y}(1-\hat{y})$$

$$\frac{\partial \hat{y}}{\partial w_1} = \hat{y}(1-\hat{y})x_1$$

$$\frac{\partial \hat{y}}{\partial w_2} = \hat{y}(1-\hat{y})x_2$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b} = \frac{-(y - \hat{y})}{\hat{y}(1-\hat{y})}\,\hat{y}(1-\hat{y})$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} = \frac{-(y - \hat{y})}{\hat{y}(1-\hat{y})}\,\hat{y}(1-\hat{y})x_1$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = \frac{-(y - \hat{y})}{\hat{y}(1-\hat{y})}\,\hat{y}(1-\hat{y})x_2$$

Now we need to cancel out a bunch of things.

DeepLearning.AI

Classification With a Perceptron

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b} = \frac{-(y - \hat{y})}{\hat{y}(1 - \hat{y})} \hat{y}(1 - \hat{y})$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_1} = \frac{-(y - \hat{y})}{\hat{y}(1 - \hat{y})} \hat{y}(1 - \hat{y}) x_1$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = \frac{-(y - \hat{y})}{\hat{y}(1 - \hat{y})} \hat{y}(1 - \hat{y}) x_2$$

cancel out on the denominator
and the numerator.

DeepLearning.AI

Classification With a Perceptron

$$\frac{\partial L}{\partial b} = -(y - \hat{y})$$

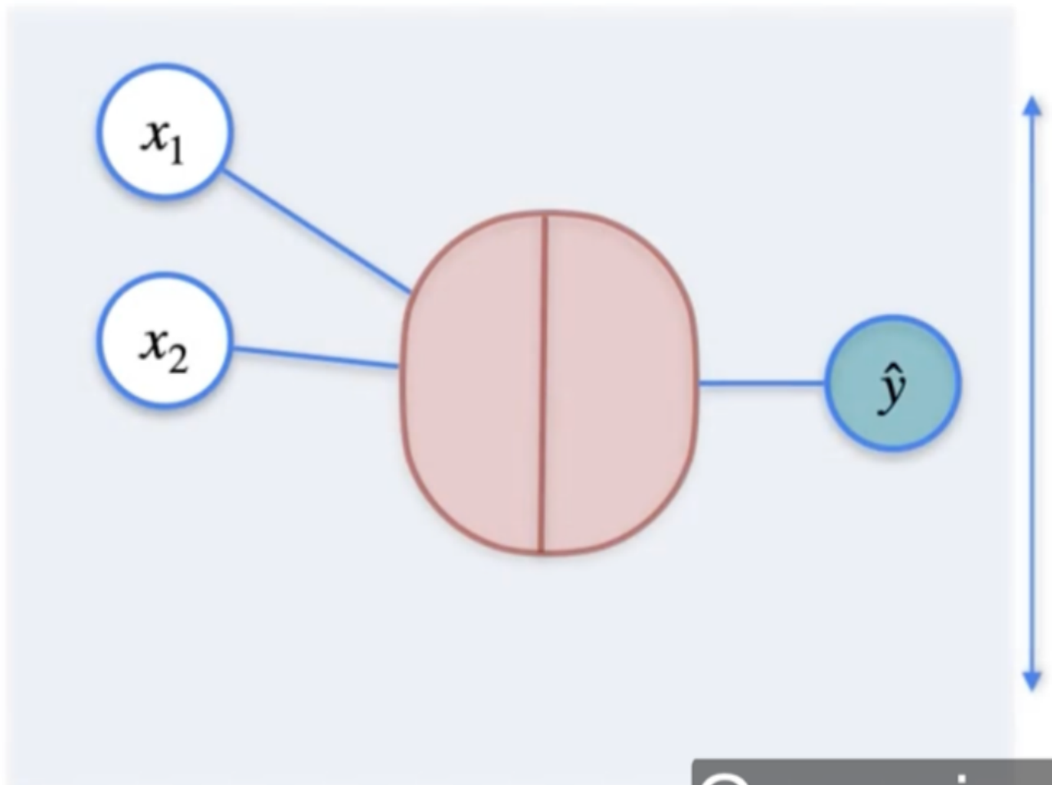
$$\frac{\partial L}{\partial w_1} = -(y - \hat{y})x_1$$

$$\frac{\partial L}{\partial w_2} = -(y - \hat{y})x_2$$

Therefore, you need a large

DeepLearning.AI

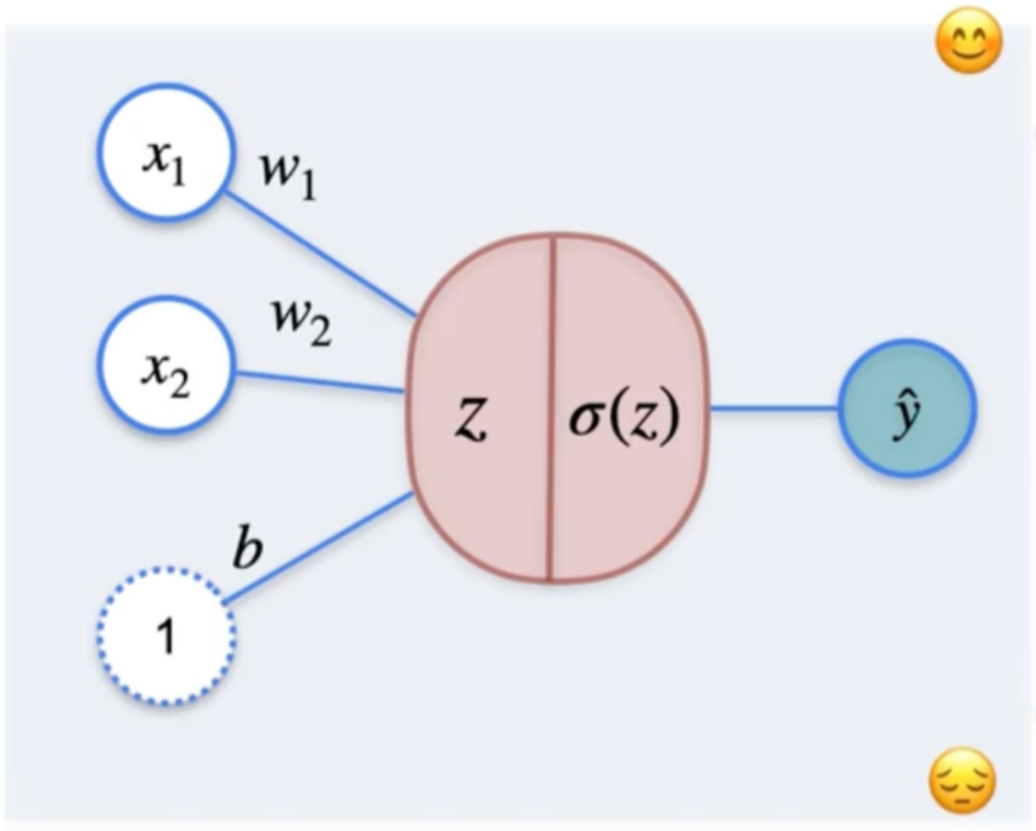
Classification With a Perceptron



Our main goal was to find
the optimal values for W1,

DeepLearning.AI

Classification With a Perceptron



To find optimal values for:

w_1 , w_2 , b

You need gradient descent

$$w_1 \rightarrow w_1 - \alpha(-x_1(y - \hat{y}))$$

$$w_2 \rightarrow w_2 - \alpha(-x_2(y - \hat{y}))$$

$$b \rightarrow b - \alpha(-(y - \hat{y}))$$

You start with some W1,

DeepLearning.AI