SIGMOID – LEARNING ALGORITHM

SHARMILA BANU K PHD
VIT, VELLORE
SCHOOL OF COMPUTER SCIENCE & ENGINEERING

LEARNING - OBJECTIVE
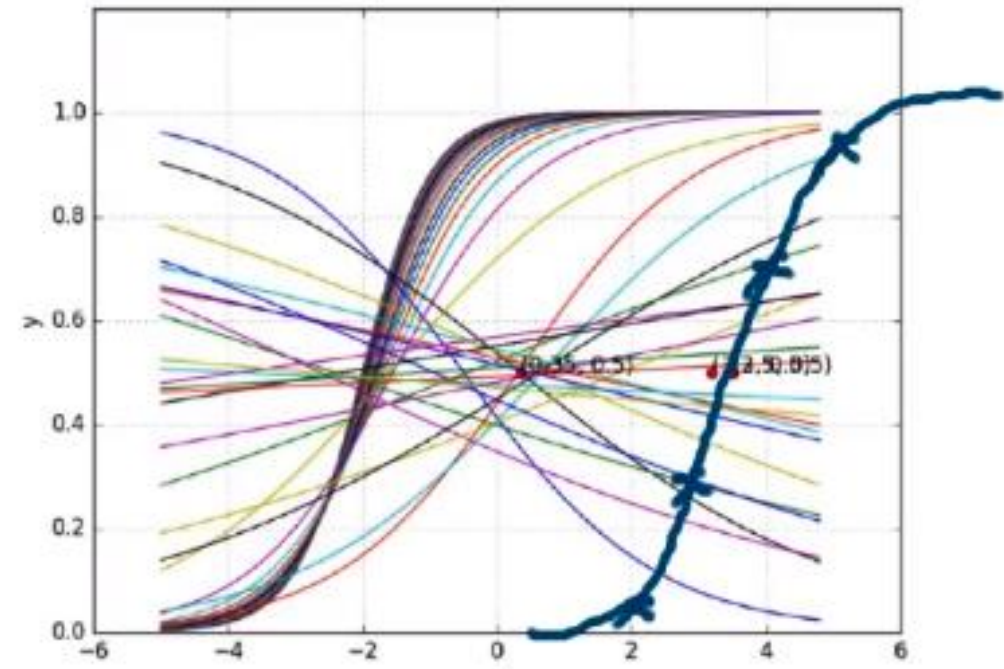
$$h = \frac{1}{1+e^{-(w*x+b)}}$$

I/P	O/P
2	0.047
3	0.268
4	0.73
5	0.952
8	0.999

Initialise

Iterate over data:

till satisfied

LEARNING BY GUESSING

Initialise w, b

Iterate over data:

$guess\_and\_update(x_i)$

till satisfied

I/P	O/P
2	0.047
3	0.268
4	0.73
5	0.952
8	0.999
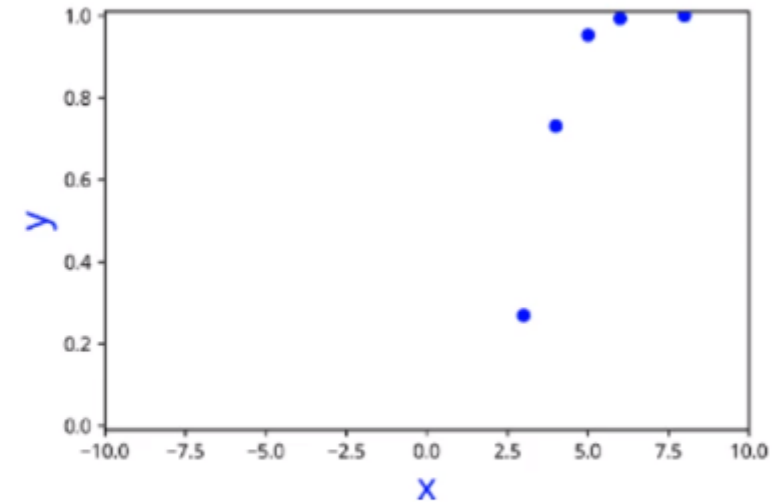
$$h = \frac{1}{1+e^{-(w*x+b)}}$$

w	b

FIRST GUESS

w	b
0	0

Initialise w, b

Iterate over data:

$guess\_and\_update(x_i)$

till satisfied

I/P	O/P
2	0.047
3	0.268
4	0.73
5	0.952
8	0.999

$$h = \frac{1}{1+e^{-(w*x+b)}}$$

w = 0
b = 0
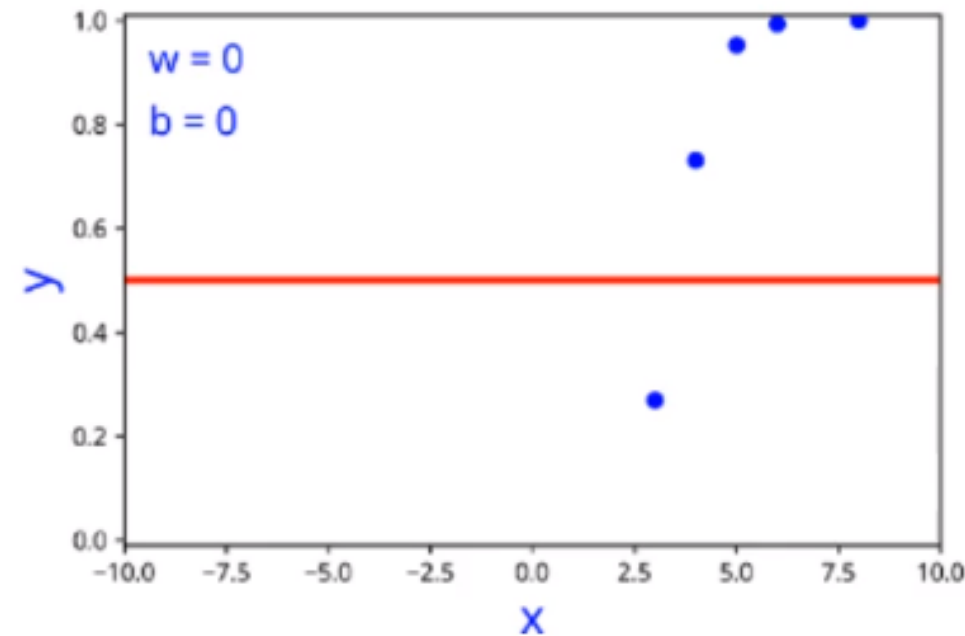
SECOND GUESS

w	b
1	0

Initialise w, b

Iterate over data:

$guess\_and\_update(x_i)$

till satisfied

I/P	O/P
2	0.047
3	0.268
4	0.73
5	0.952
8	0.999

$$h = \frac{1}{1 + e^{-(w*x+b)}}$$

w = 1
b = 0
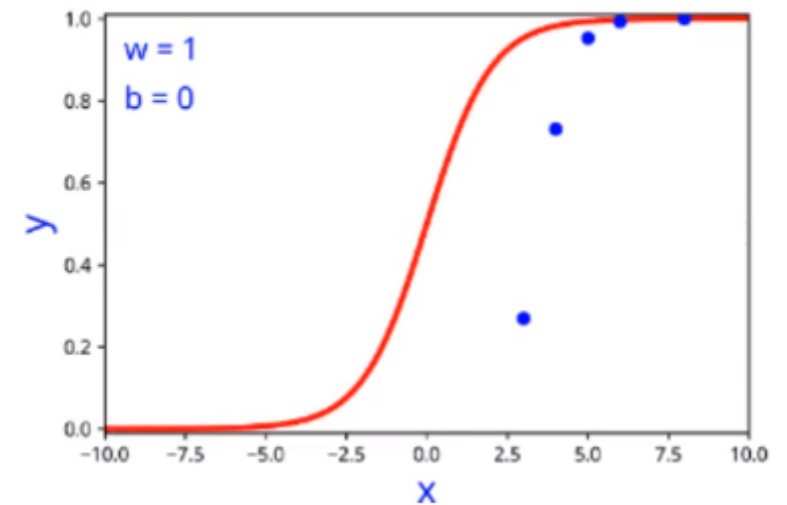
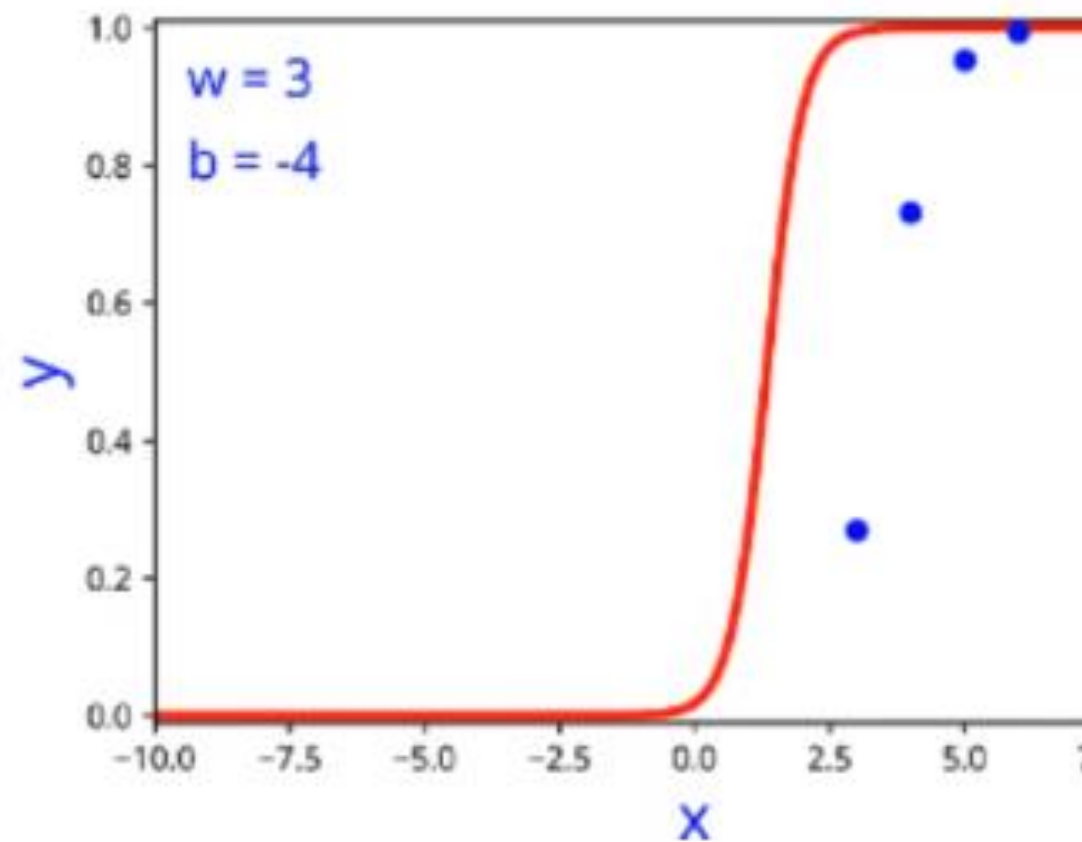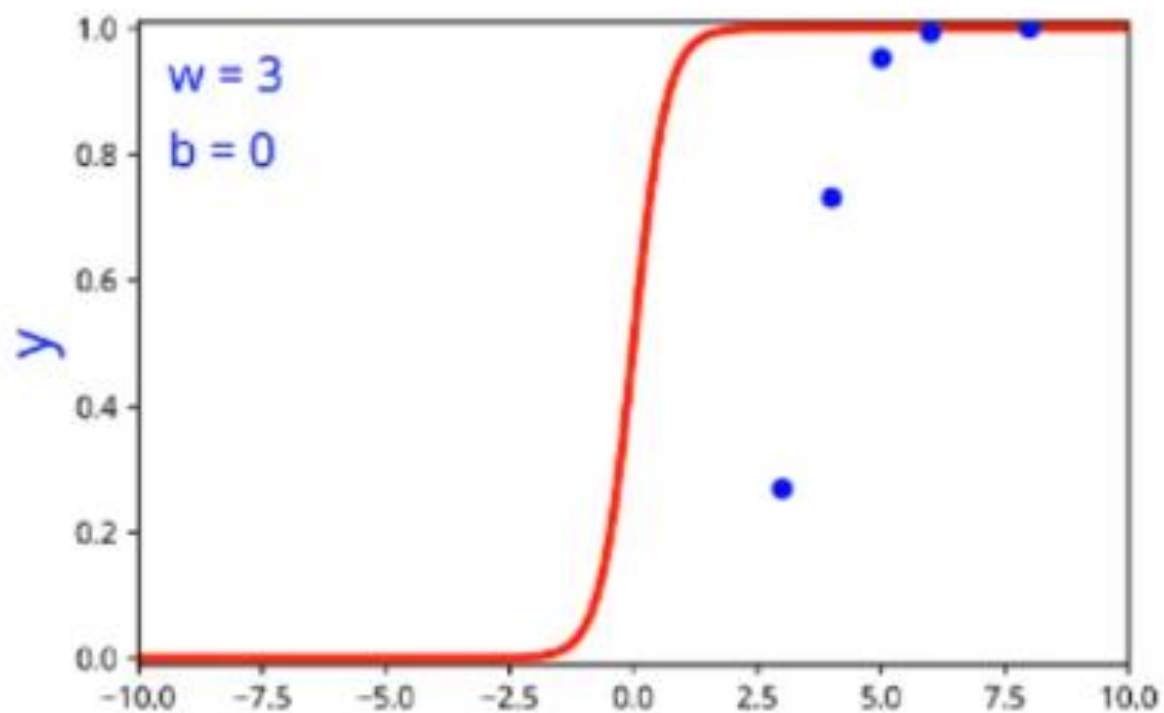# THIRD GUESS BASED ON PROGRESS IN SECOND GUESS

w	b
3	0

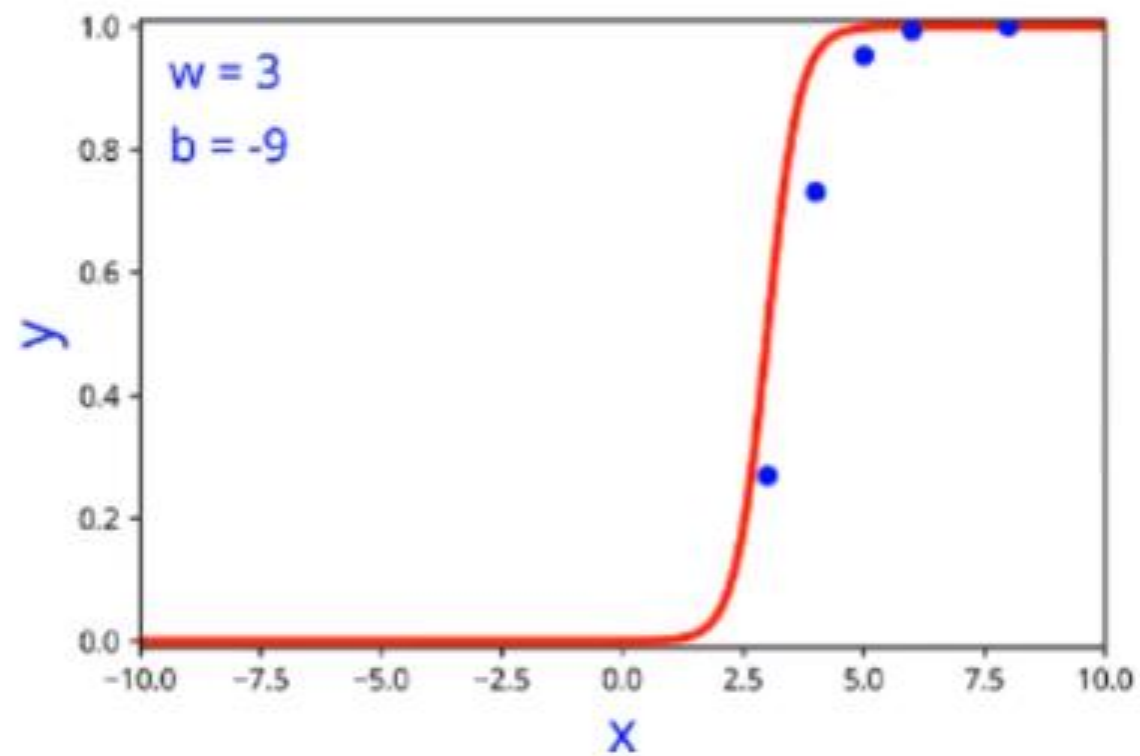- To get closer to intended sigmoid, we have to move the sigmoid along x
- It can be done by decreasing b

w = 3
b = 0
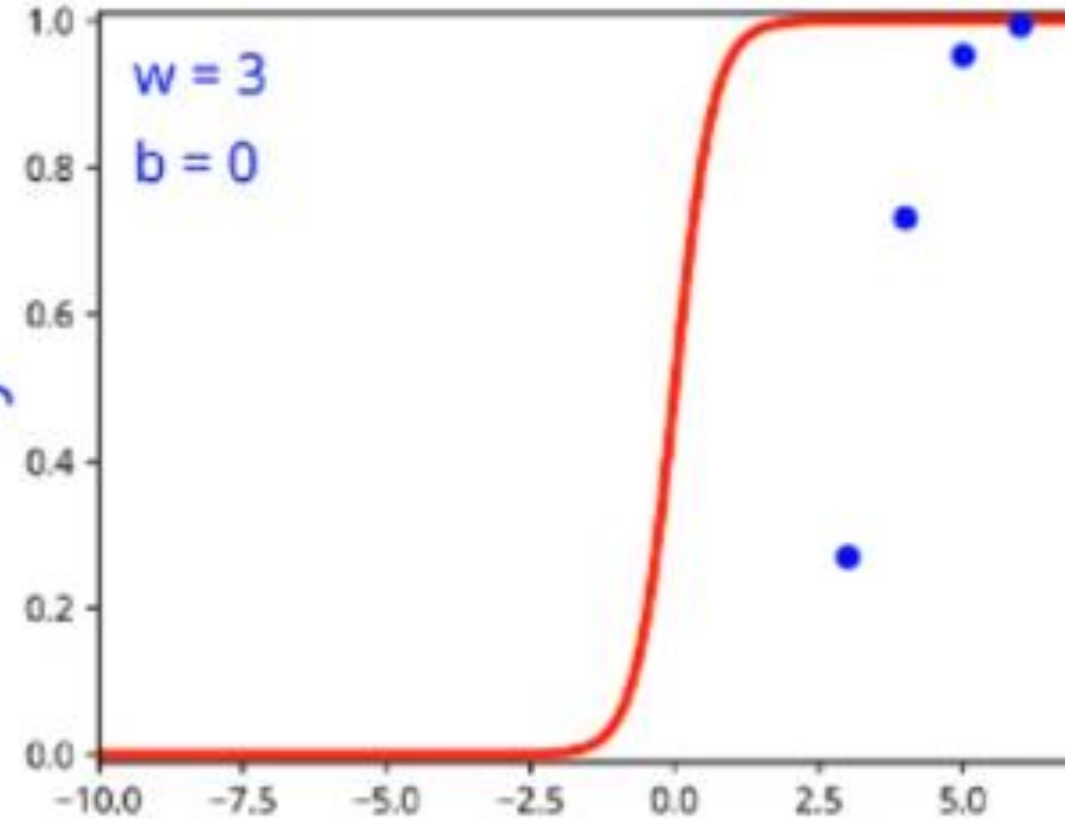
NEXT GUESS IS

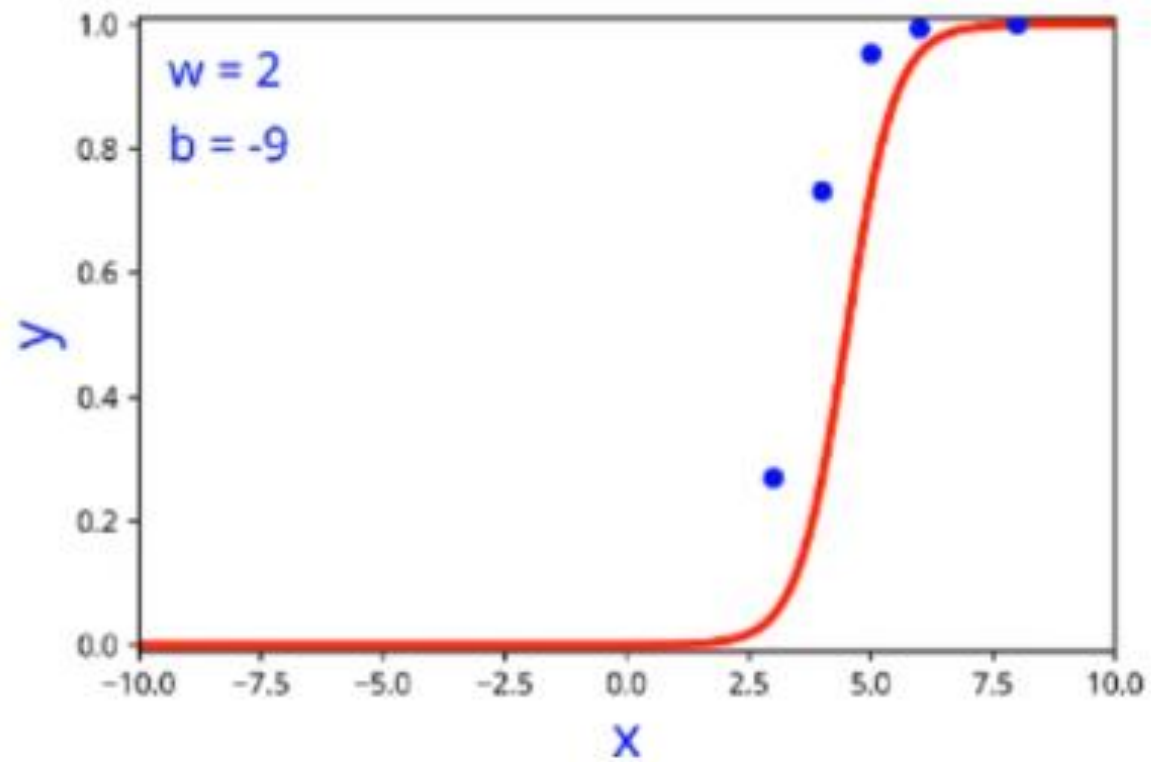| w	b
3	-4

MORE GUESSING
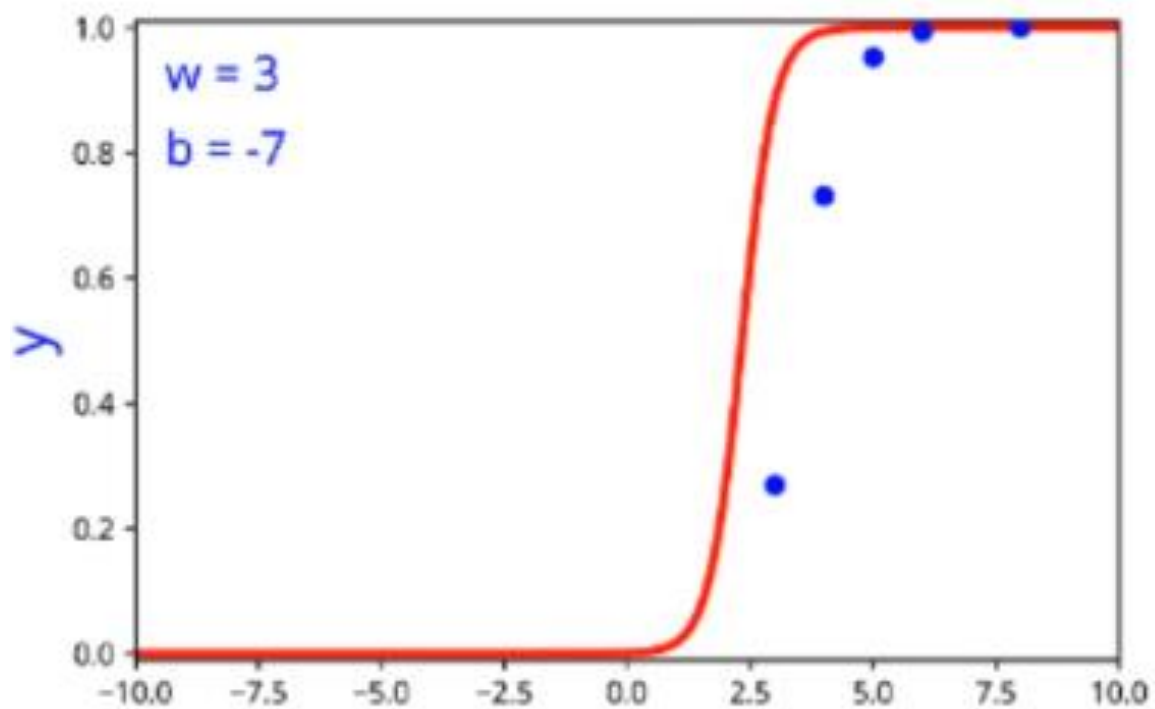
OVERSHOT!

FINAL ADJUSTMENT

I/P	O/P
2	0.047
3	0.268
4	0.73
5	0.952
8	0.999

- Exact fit
- Guessing worked for 1D

w = 2
b = -7
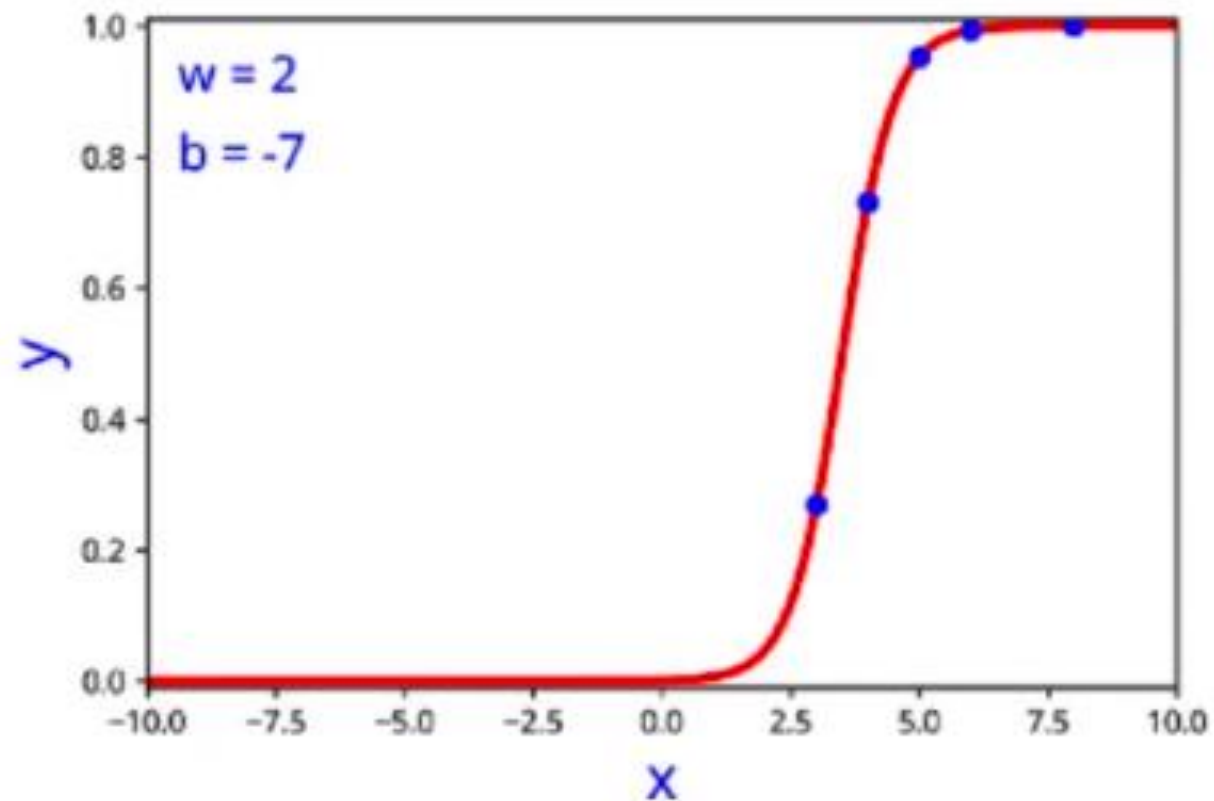
LEARNING ALGORITHM

Initialise w, b

Iterate over data:

$$w = w + \Delta w$$
$$b = b + \Delta b$$

till satisfied

Initialise w, b

Iterate over data:
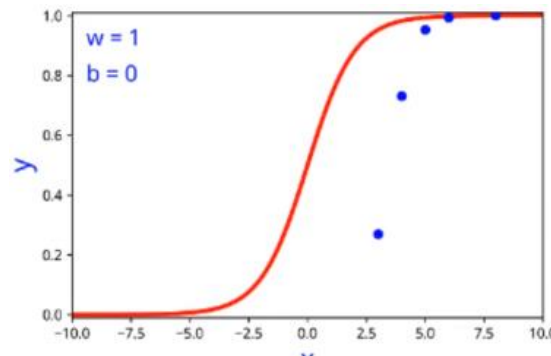
w = 0

b = 0

till satisfied

Initialise w, b

Iterate over data:

1 = 0 + Δ1

0 = 0 + Δ0

till satisfied

Initialise w, b

Iterate over data:

2 = 1 + Δ1

0 = 0 + Δ0

till satisfied

LEARNING ALGORITHM —CTD...

Initialise w, b

Iterate over data:

$3 = 2 + \Delta1$

$0 = 0 + \Delta0$

till satisfied

Initialise w, b

Iterate over data:

$3 = 3 + \Delta0$

$-2 = 0 + \Delta-2$

till satisfied

Initialise w, b

Iterate over data:

$3 = 3 + \Delta0$

$-4 = -2 + \Delta-2$
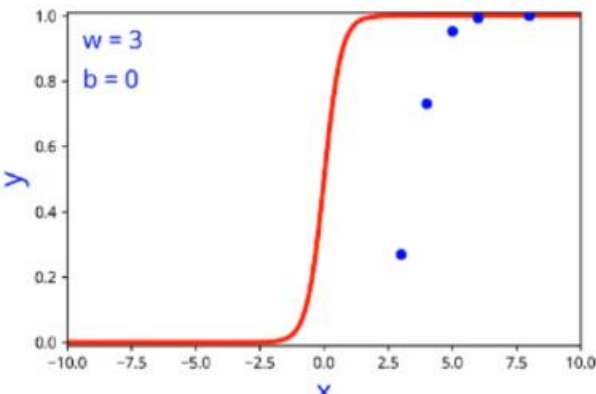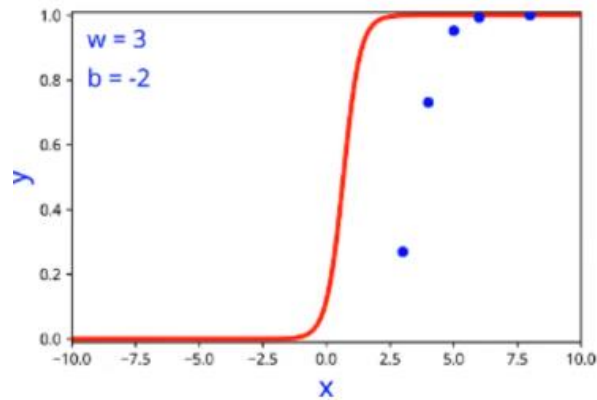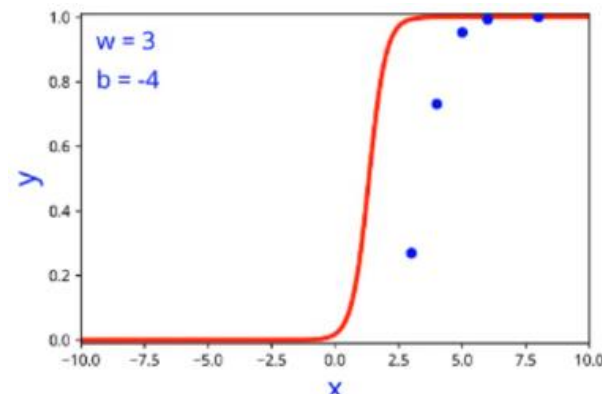
till satisfied

WHY GUESSING IS NOT GOOD?

Initialise w, b

Iterate over data:
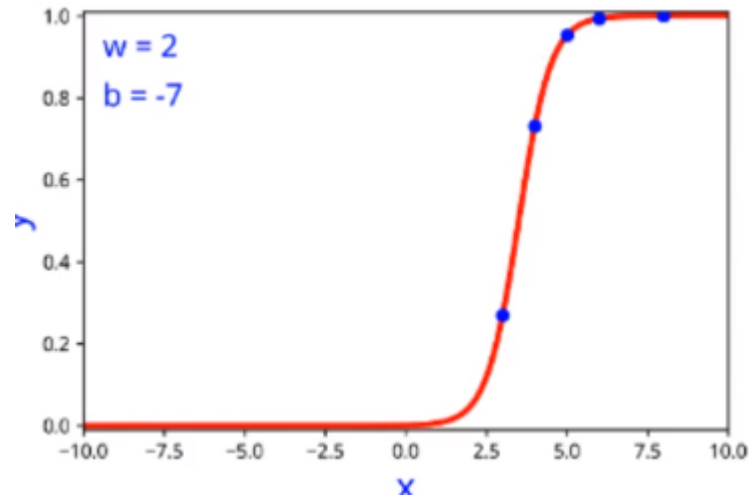
$$w = w + \Delta w$$
$$b = b + \Delta b$$

till satisfied

$$\Delta w = some\_guess$$
$$\Delta b = some\_guess$$

w = 2
b = -7

INFERENCE

Guessing is not feasible in real-life situations

It will take a long time to converge to a solution

It is good to follow a principled approach based on Loss Function

LEARNING W.R.T LOSS FUNCTION (MSE)

Initialise w, b
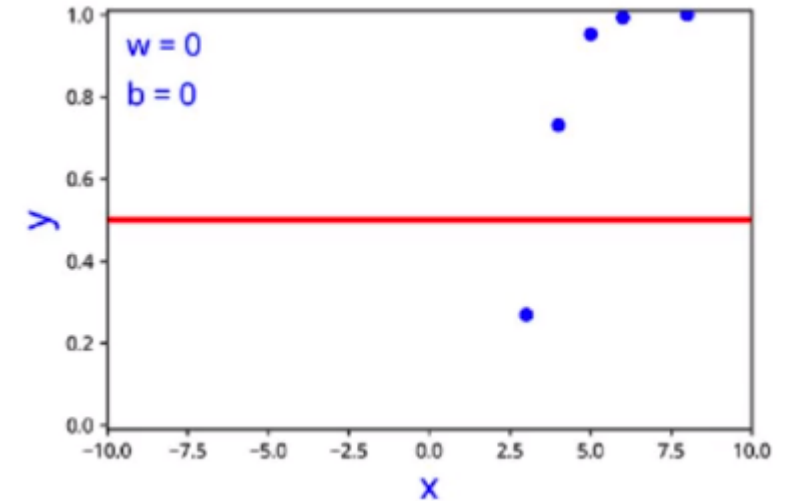
Iterate over data:

    w = 0

    b = 0

till satisfied

I/P	O/P
3	0.268
4	0.73
5	0.952
6	0.994
8	0.999

w	b	Loss
0	0	0.1609

LOSS FOR THE SECOND GUESS

Initialise w, b

I/P	O/P
3	0.268
4	0.73
5	0.952
6	0.994
8	0.999

Iterate over data:

$$1 = 0 + \Delta 1$$

$$0 = 0 + \Delta 0$$

till satisfied

$w = 1$
$b = 0$
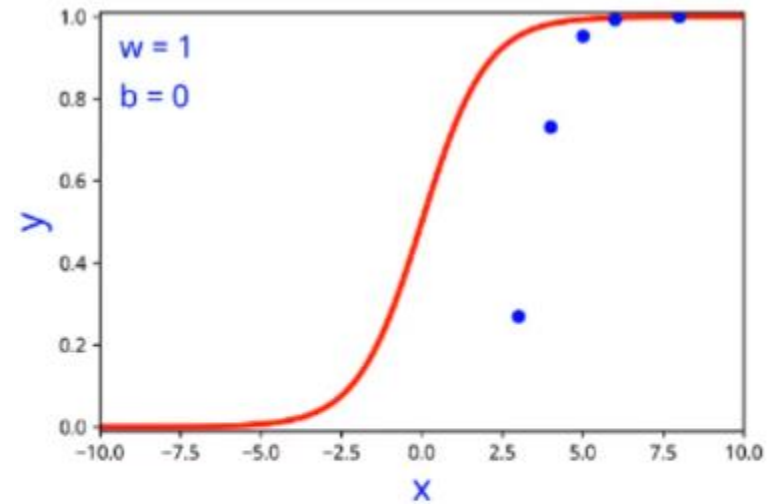
w	b	Loss
1	0	0.1064

HAS THE LOSS COME DOWN WITH NEXT GUESS?

Initialise w, b

Iterate over data:
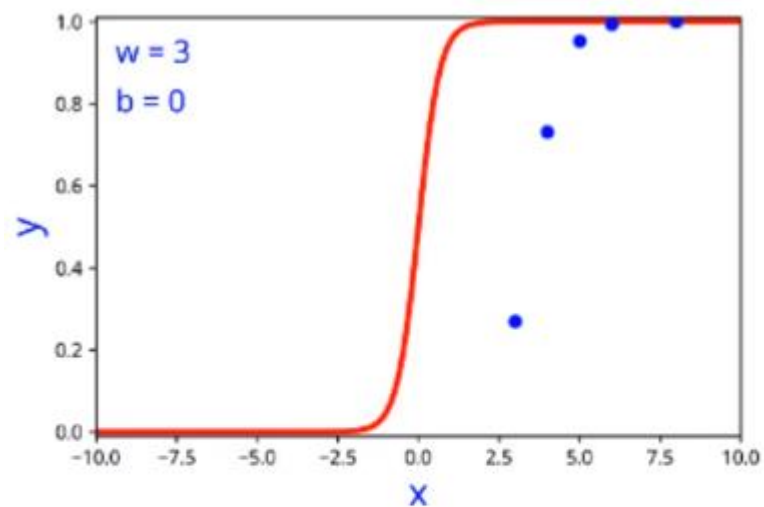
$$2 = 1 + \Delta1$$

$$0 = 0 + \Delta0$$

till satisfied

I/P	O/P
3	0.268
4	0.73
5	0.952
6	0.994
8	0.999

$w = 2$
$b = 0$
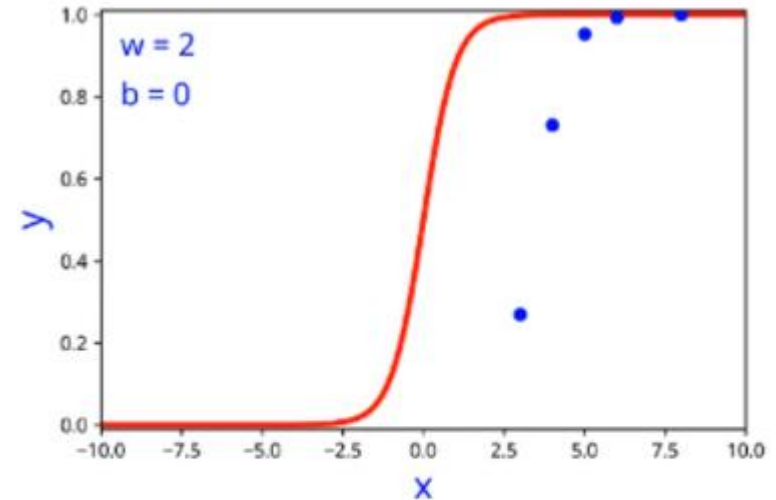
w	b	Loss
2	0	0.1210

$3 = 2 + \Delta 1$

$0 = 0 + \Delta 0$

$3 = 3 + \Delta 0$

$-2 = 0 + \Delta -2$

$2 = 1 + \Delta 1$

$0 = 0 + \Delta 0$

w	b	Loss
3	0	0.1217
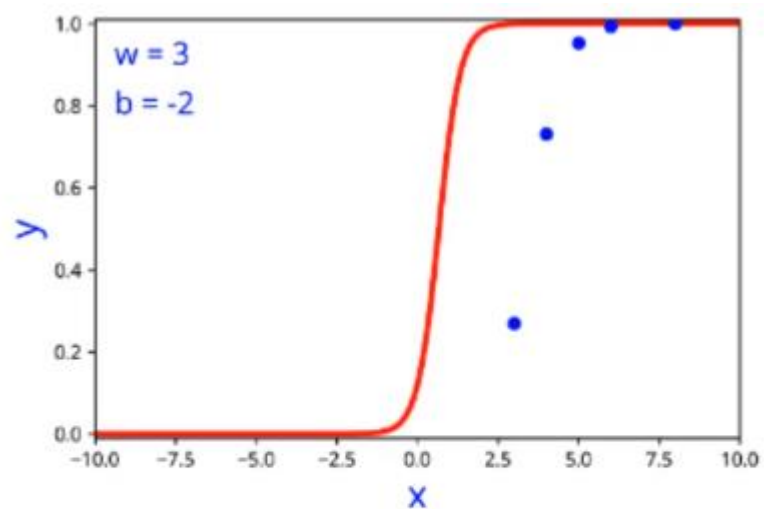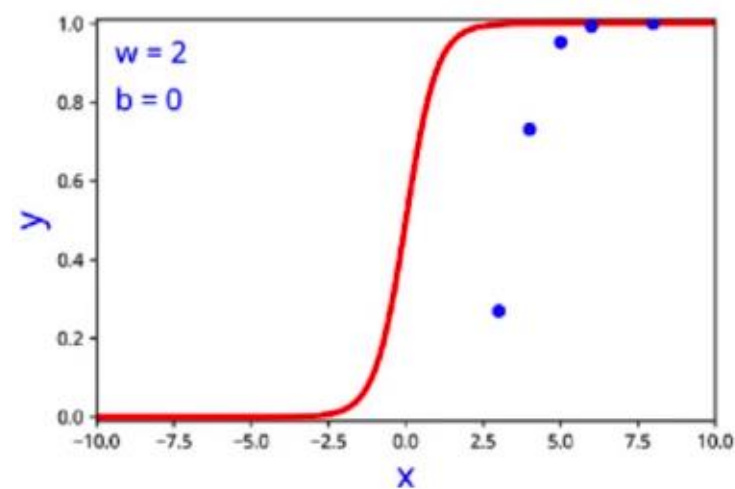
w	b	Loss
3	-2	0.1215

w	b	Loss
2	0	0.1210

$3 = 3 + \Delta 0$
$-4 = -2 + \Delta -2$

$3 = 3 + \Delta 0$
$-7 = -4 + \Delta -3$

$3 = 3 + \Delta 0$
$-9 = -7 + \Delta -2$

$2 = 3 + \Delta -1$
$-9 = -7 + \Delta -2$
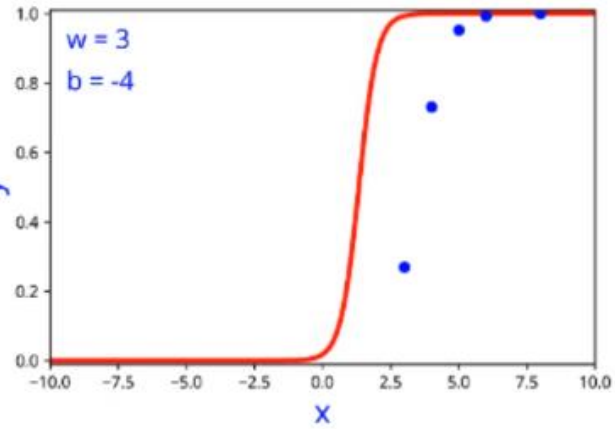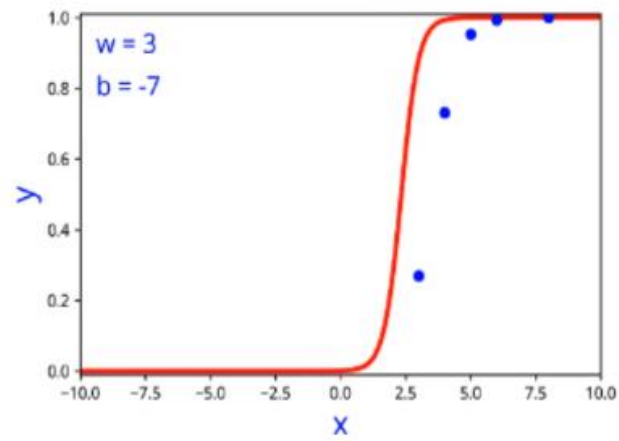
w	b	Loss
3	-4	0.1198
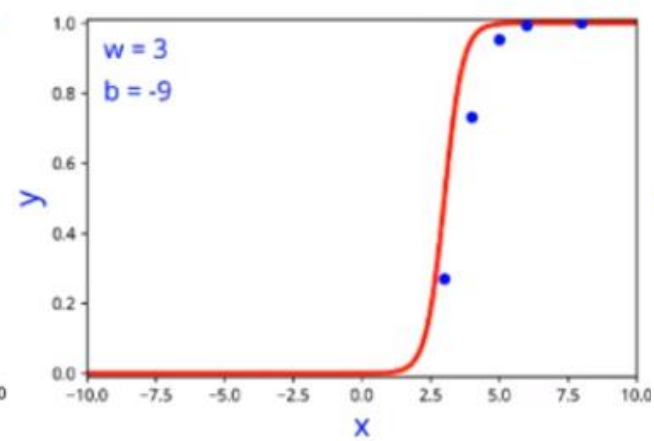
w	b	Loss
3	-7	0.1081
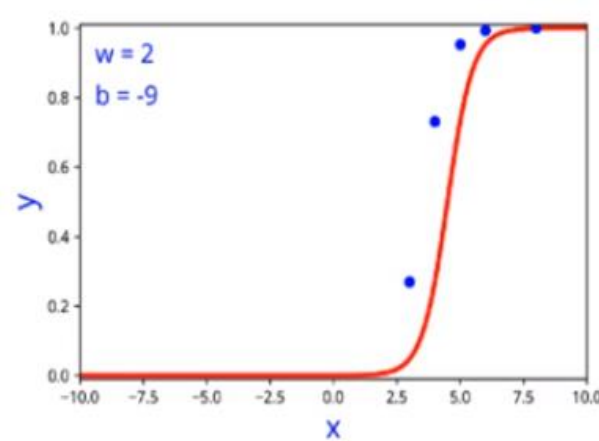
w	b	Loss
3	-9	0.0209

w	b	Loss
2	-9	0.0636

$2 = 3 + \Delta -1$
$-7 = -9 + \Delta 2$
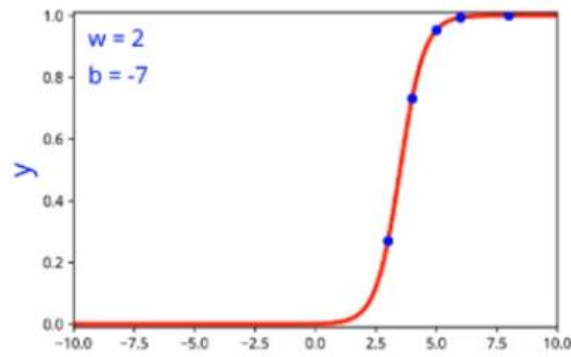
w	b	Loss
2	-7	0.000

INFERENCE

$$w = w + \Delta w$$
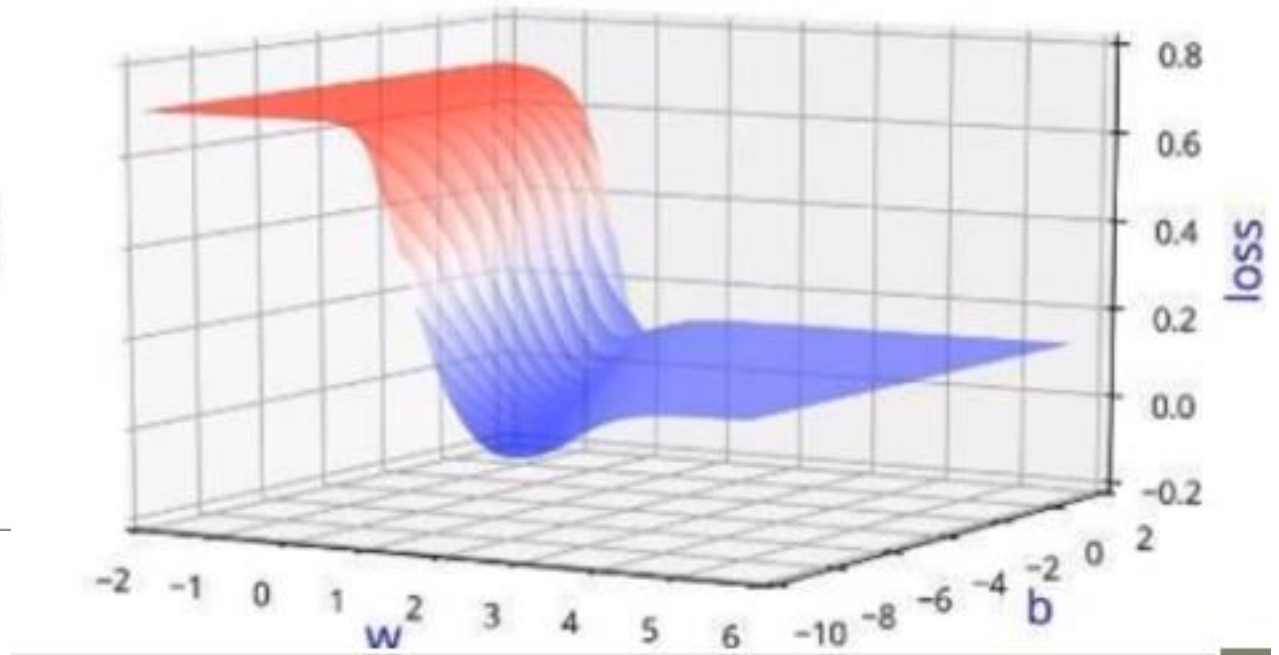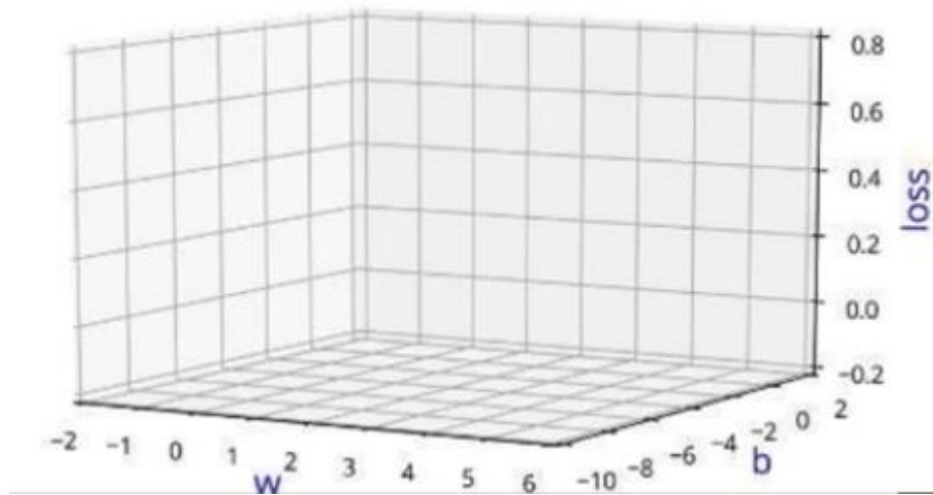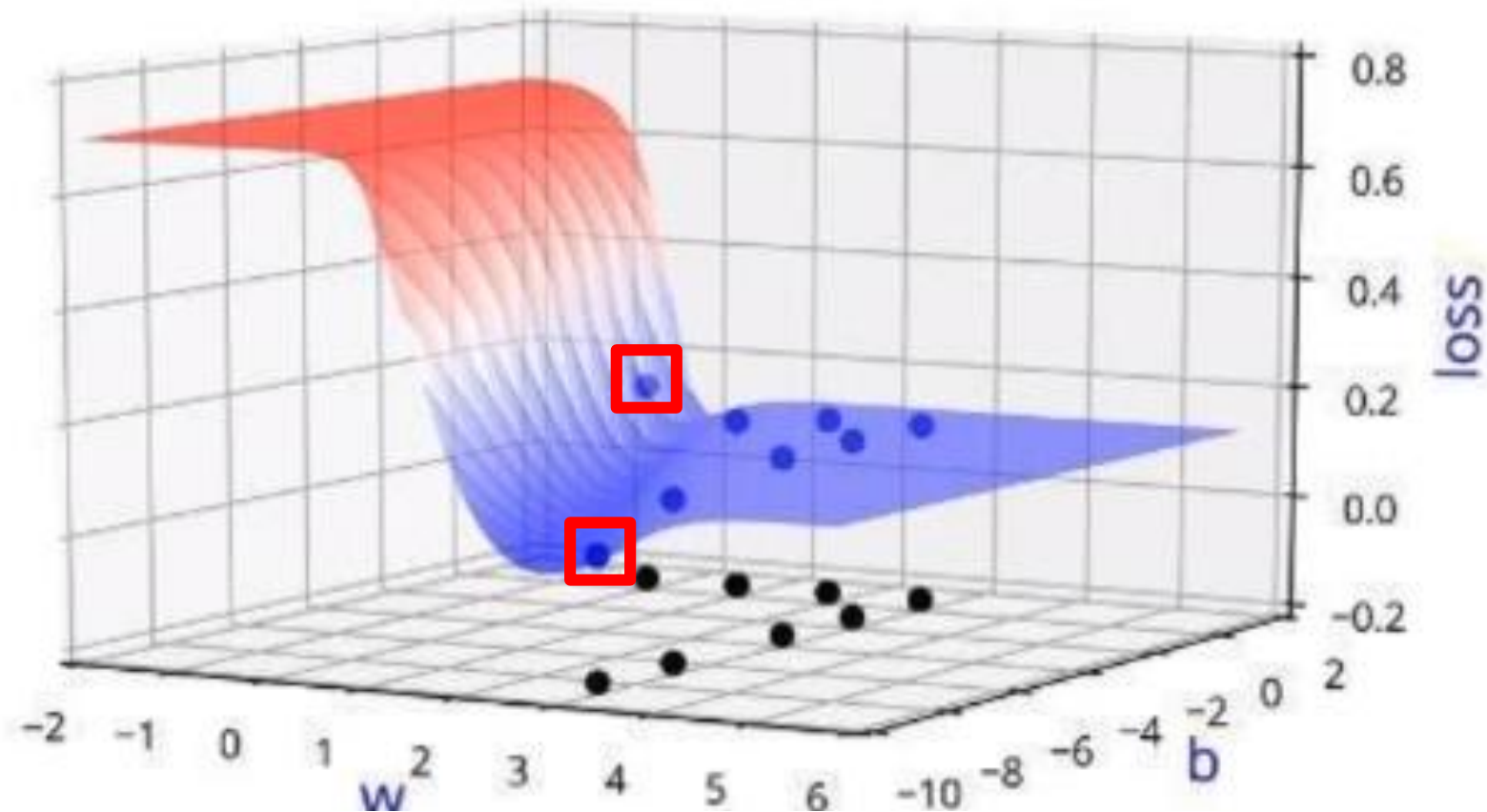$$b = b + \Delta b$$

- NO CONSTANT INCREASE OR **_DECREASE_** IN LOSS FUNCTION
- IT IS FLUCTUATING WITH THE GUESSES MADE

VISUALISING THE ERROR SURFACE

LOSS VS W,B
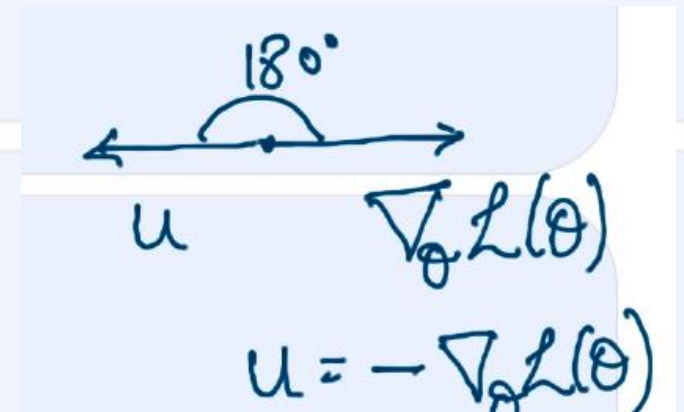# (LOOK OUT FOR ABSOLUTE MINIMUM AND IF THERE IS A STEADY DECREASE FROM INITIAL GUESS)

AN INSIGHT ON MATHS

DERIVING GRADIENT DESCENT RULE

Gradient Descent Rule,

- The direction u that we intend to move in should be at 180° w.r.t. the gradient.

- In other words, move in a direction opposite to the gradient.

$$180°$$

$$u \qquad \nabla_\theta \mathscr{L}(\theta)$$

$$u = - \nabla_\theta \mathscr{L}(\theta)$$

Parameter Update Rule

$$w_{t+1} = w_t - \eta \Delta w_t$$
$$b_{t+1} = b_t - \eta \Delta b_t$$

$$\text{where } \Delta w_t = \frac{\partial \mathscr{L}(w,b)}{\partial w} \text{ at } w=w_t, b=b_t, \quad \Delta b_t = \frac{\partial \mathscr{L}(w,b)}{\partial b} \text{ at } w=w_t, b=b_t$$

LEARNING ALGORITHM - REDEFINED

ComputeGradient(L, θ)

Initialise w, b

Iterate over data:

    $compute\ \hat{y}$

    $compute\ \mathscr{L}(w,b)$

    $w_{t+1} = w_t - \eta \Delta w_t$

    $b_{t+1} = b_t - \eta \Delta b_t$

till satisfied

I/P	O/P
3	0.268
4	0.73
5	0.952
6	0.994
8	0.999

$$\hat{y} = \frac{1}{1 + e^{-(wx+b)}}$$

$$Loss\,\mathscr{L}(w,b) = \sum_{i=1}^{5} (y_i - \hat{y}_i)^2$$