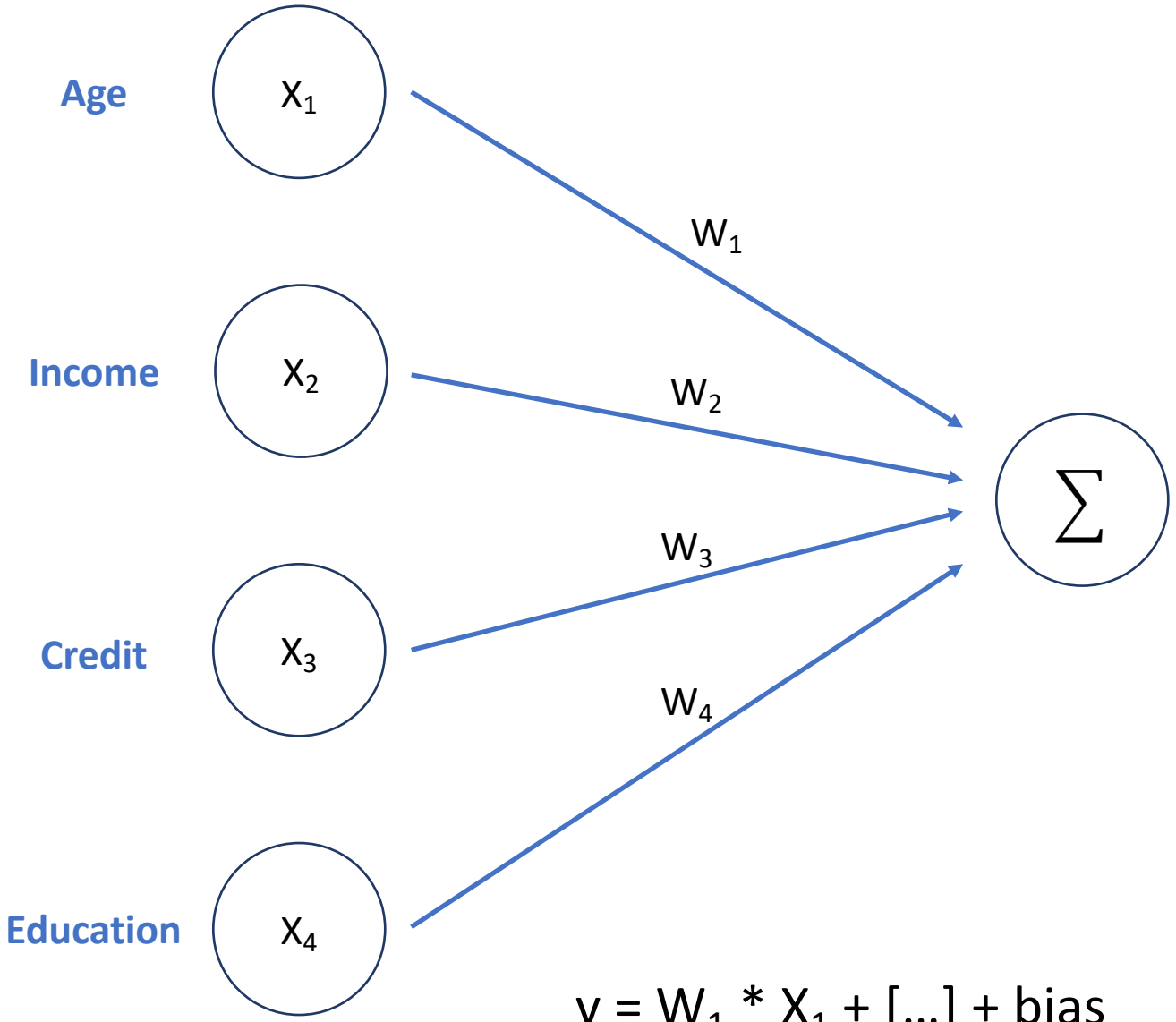
Activation Function

- Activation functions are used to get the output of node.
- Activation functions map the values between 0 to 1 or -1 to 1 etc, depending on the function.

$$(PAC, SHO, PAS, DRI, DEF, PHY) * \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix} = (w_1*PAC + w_2*SHO + w_3*PAS + w_4*DRI + w_5*DEF + w_6*PHY)$$

89

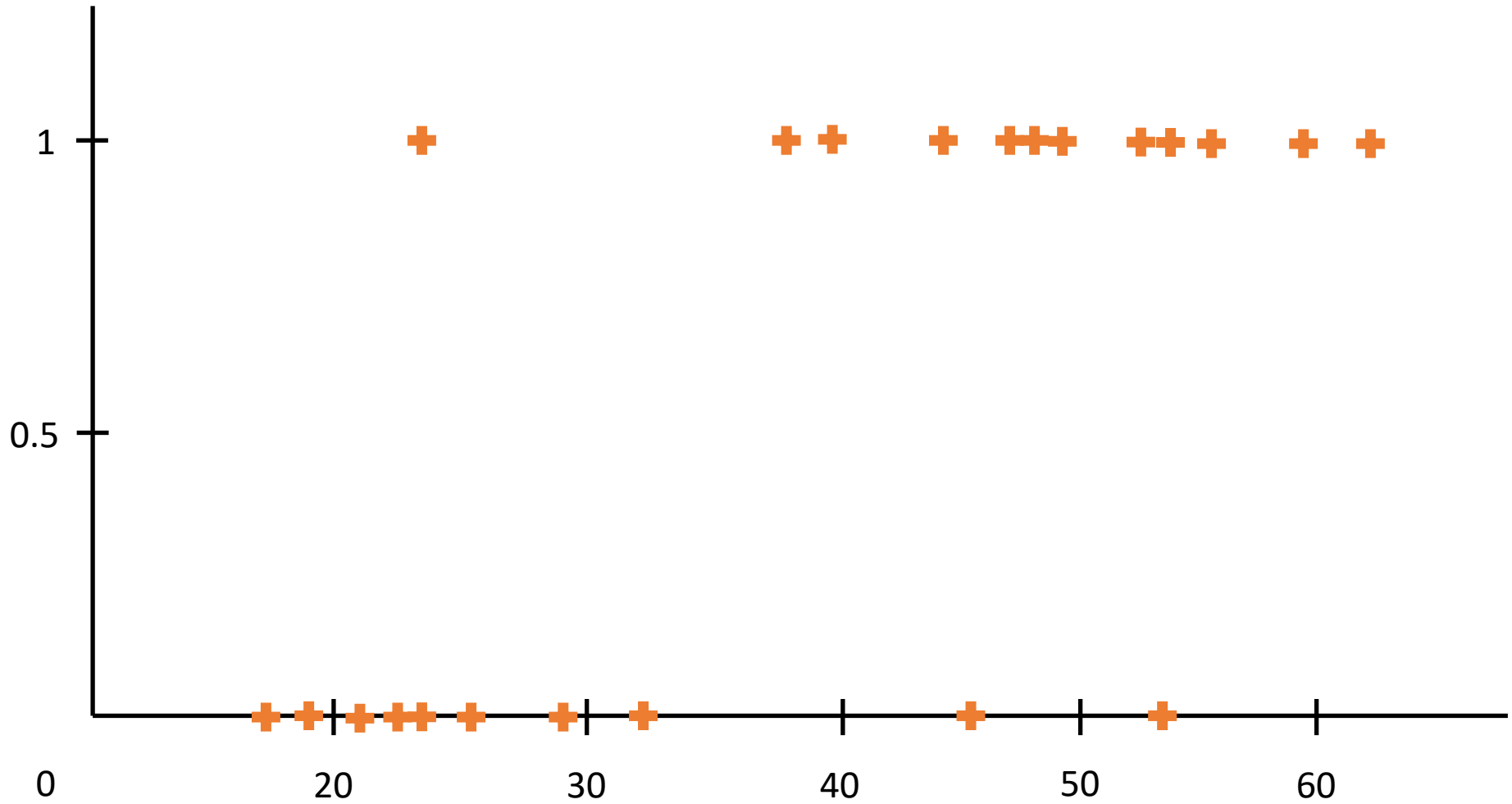


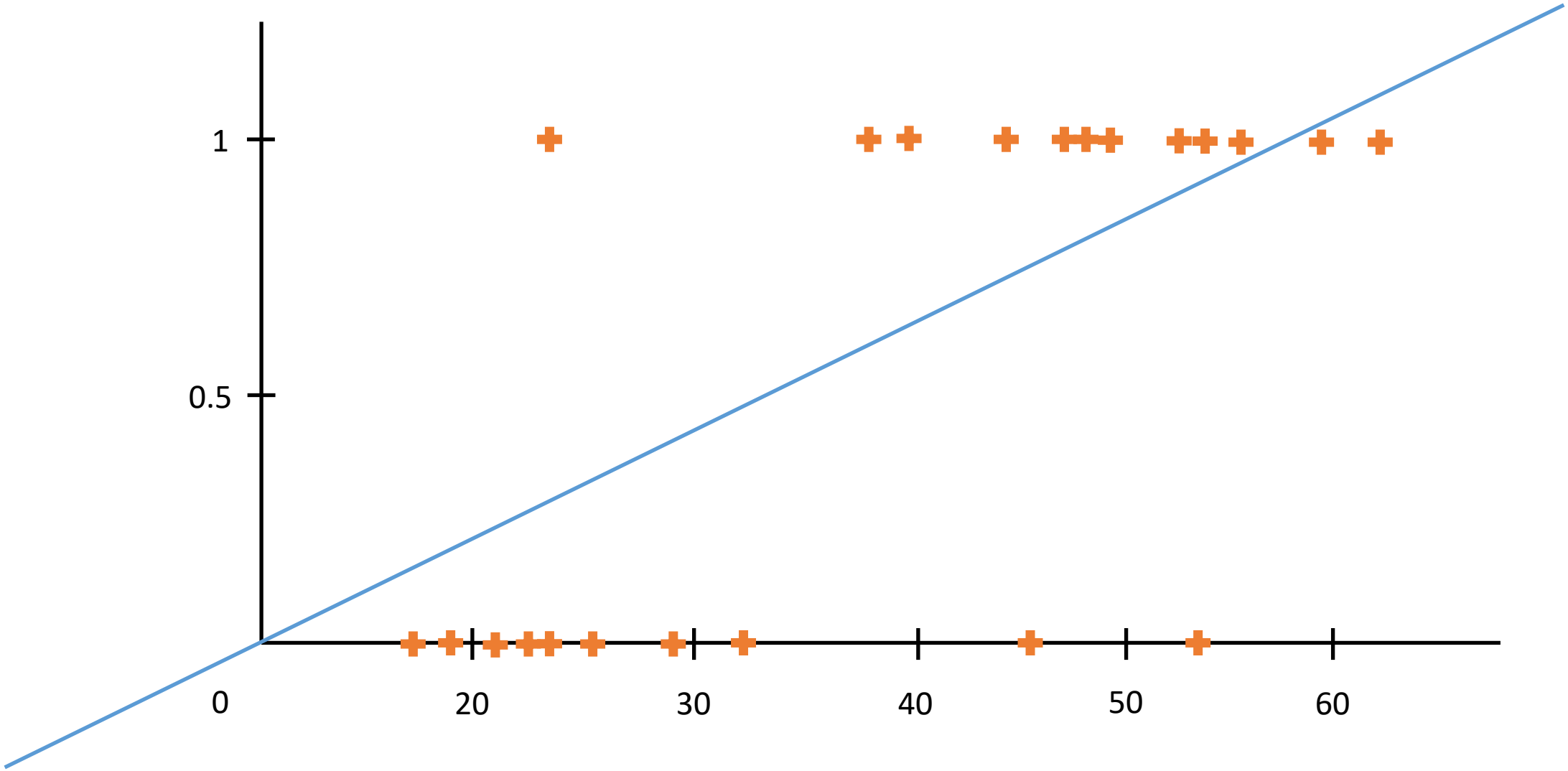
$$y = W_1 * X_1 + [\ldots] + \text{bias}$$

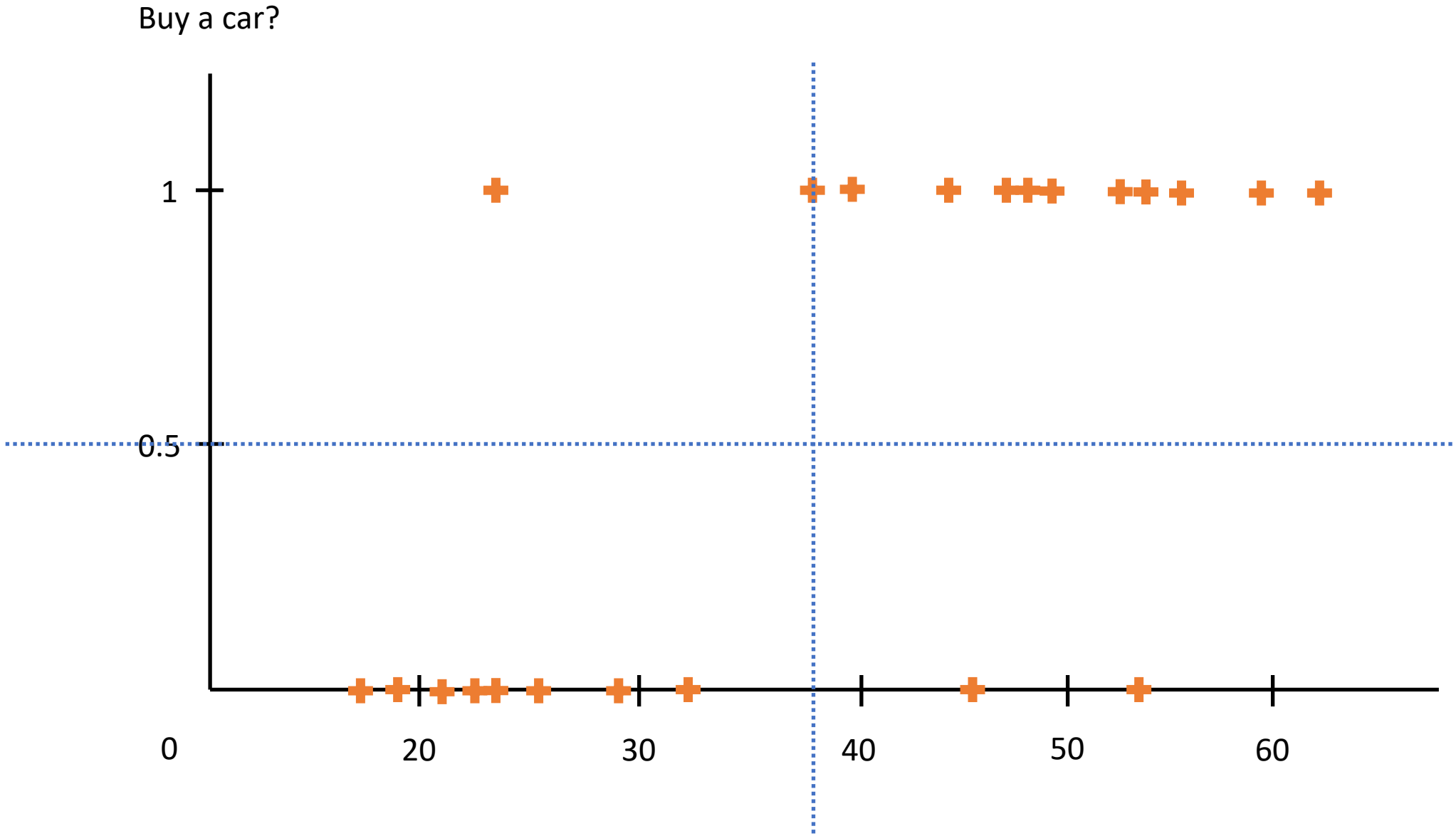
$$y = 0.05 * X_1 + 0.25 * X_2 + 0.15 * X_3 + 0.012 * X_4 - 1.22$$

Binary Classification

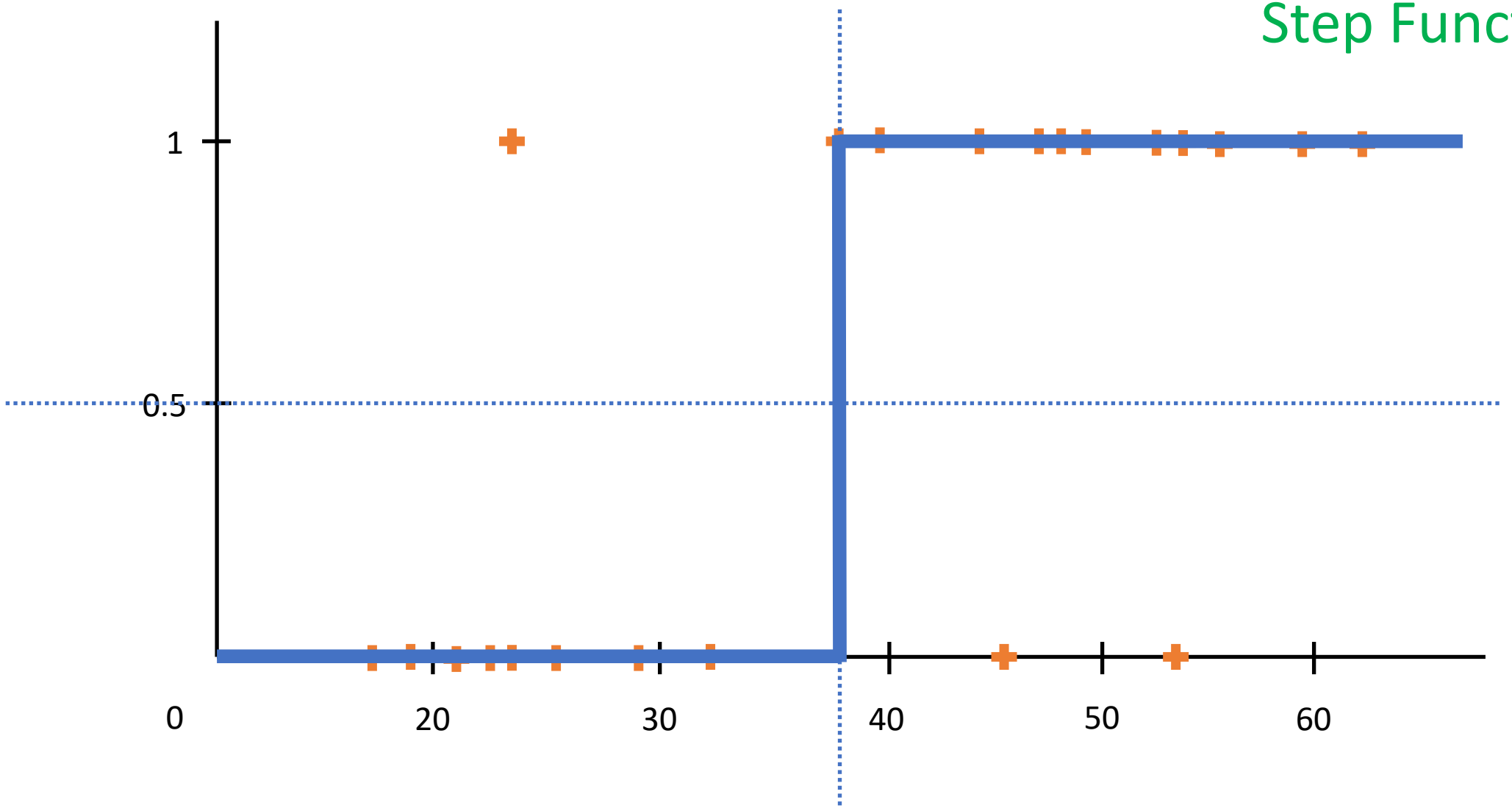
Age	Buy_Car
21	0
48	1
28	0
19	0
56	1
65	1
32	1
24	0
43	1
22	0
53	1

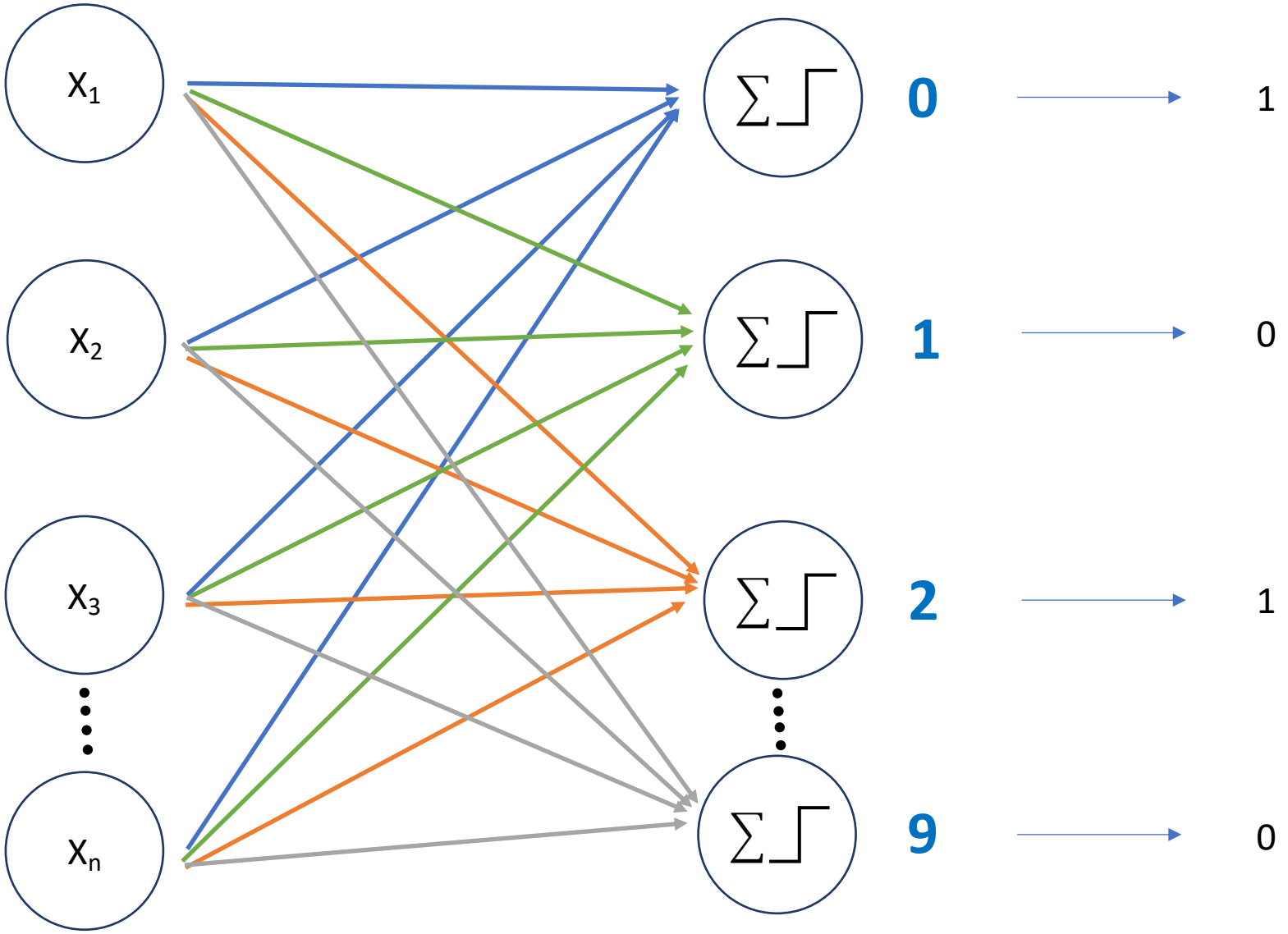

1
0.5
0
20
30
40
50
60


1
0.5
0
20
30
40
50
60


Buy a car?
1
0.5
0
20
30
40
50
60

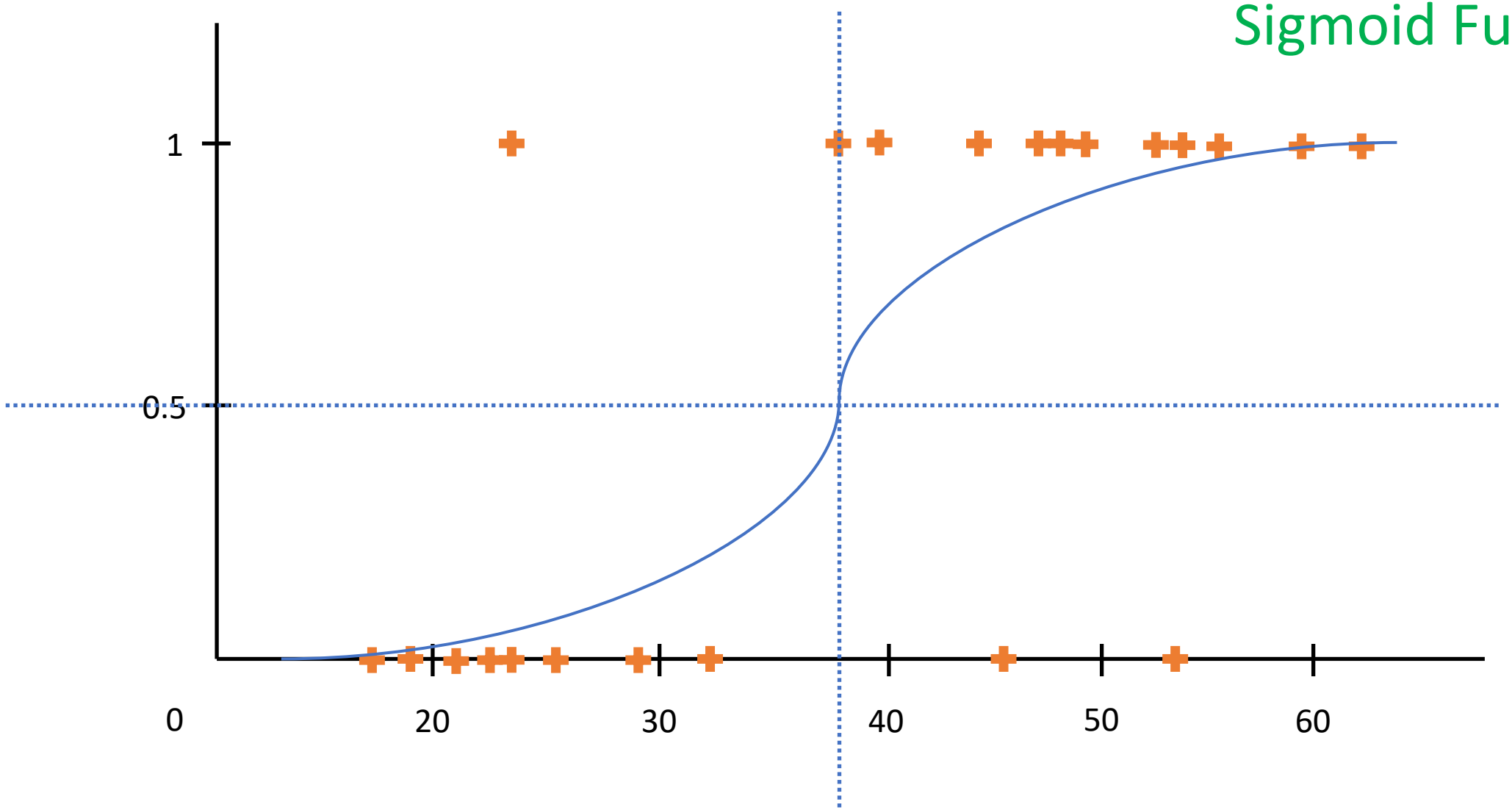
Step Function

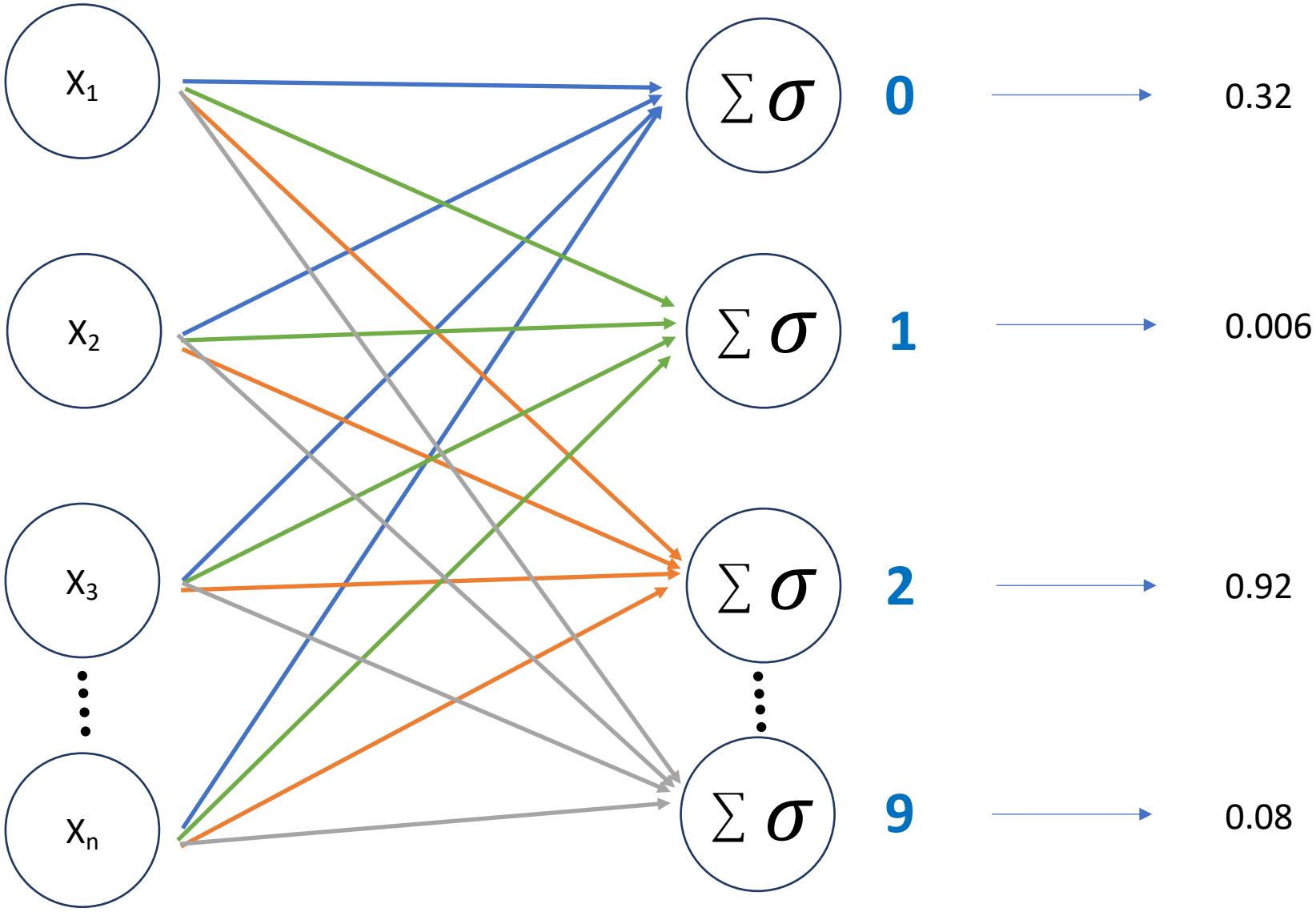




$$y = 0.05 * X_1 + 0.25 * X_2 + 0.15 * X_3 + 0.012 * X_4 - 1.22$$

Sigmoid Function

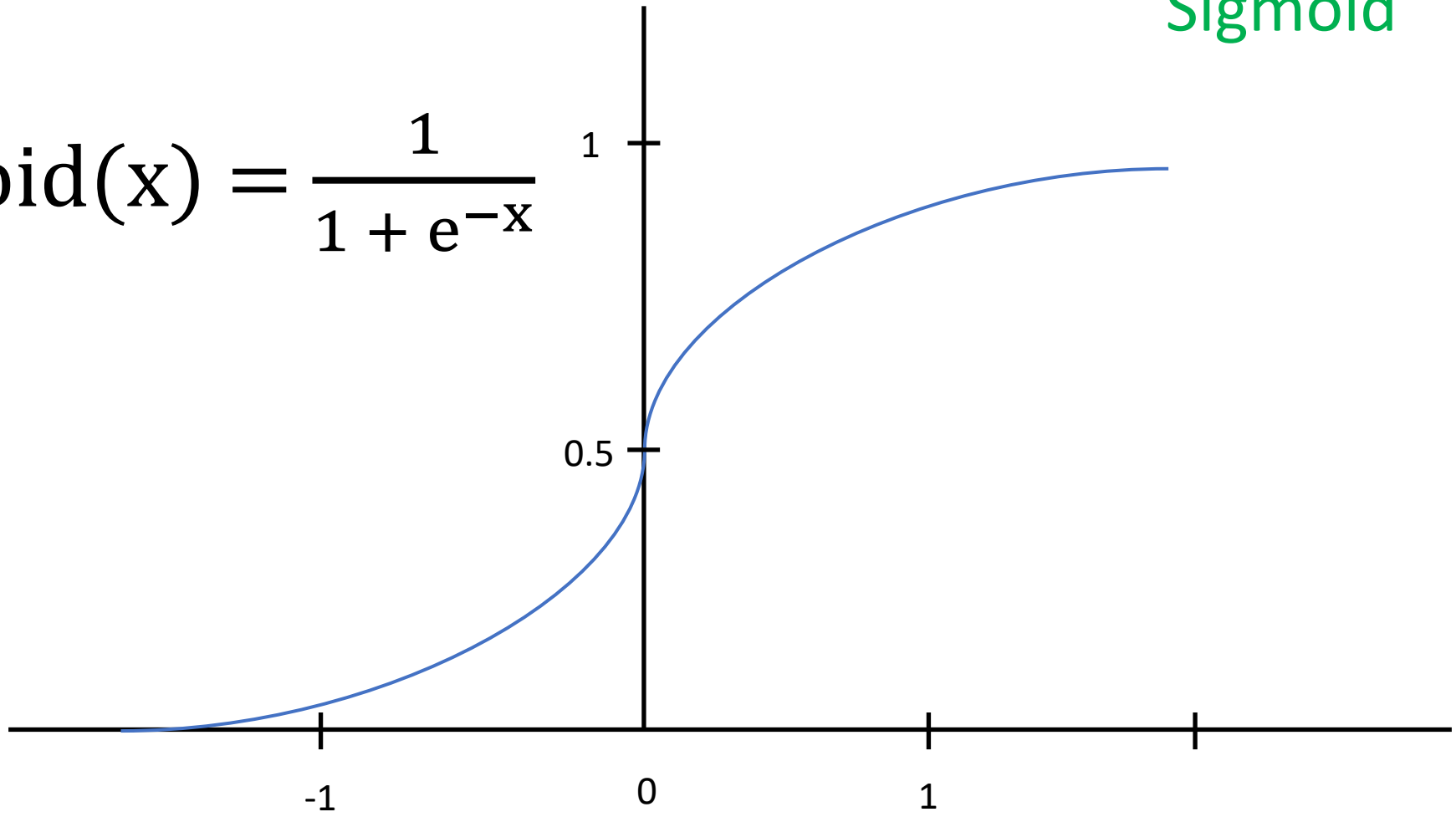




$$y = 0.05 * X_1 + 0.25 * X_2 + 0.15 * X_3 + 0.012 * X_4 - 1.22$$

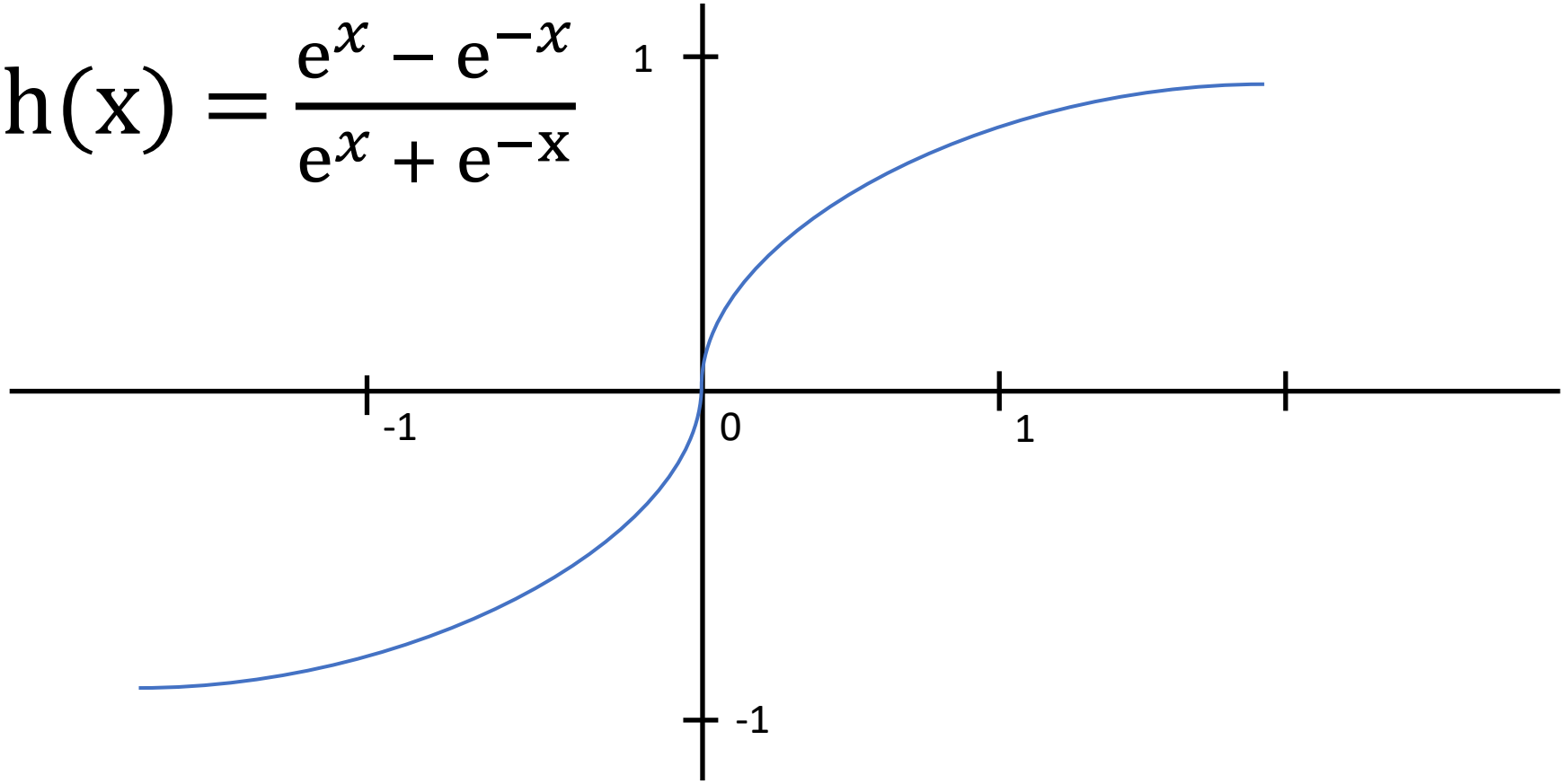
Sigmoid

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



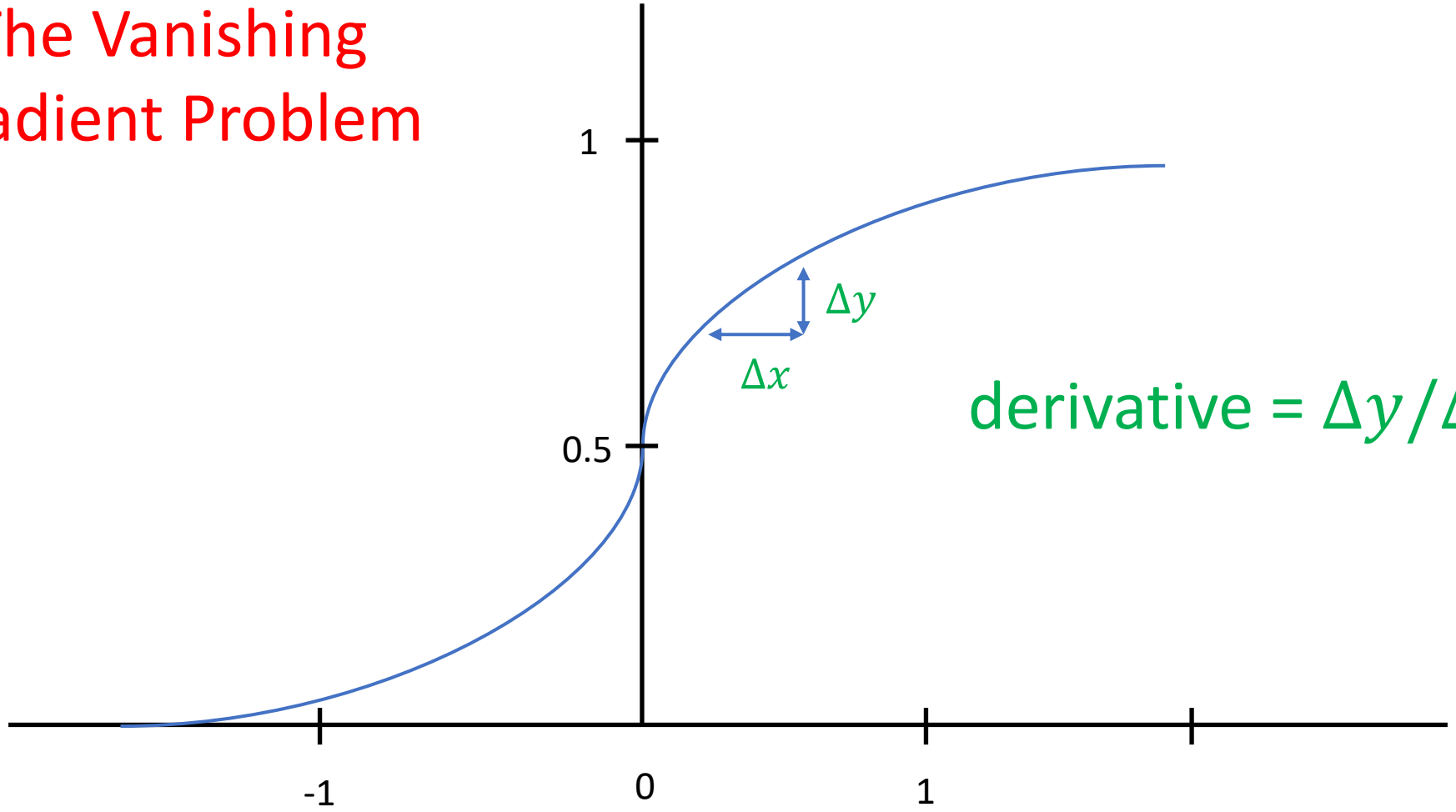
tanh

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



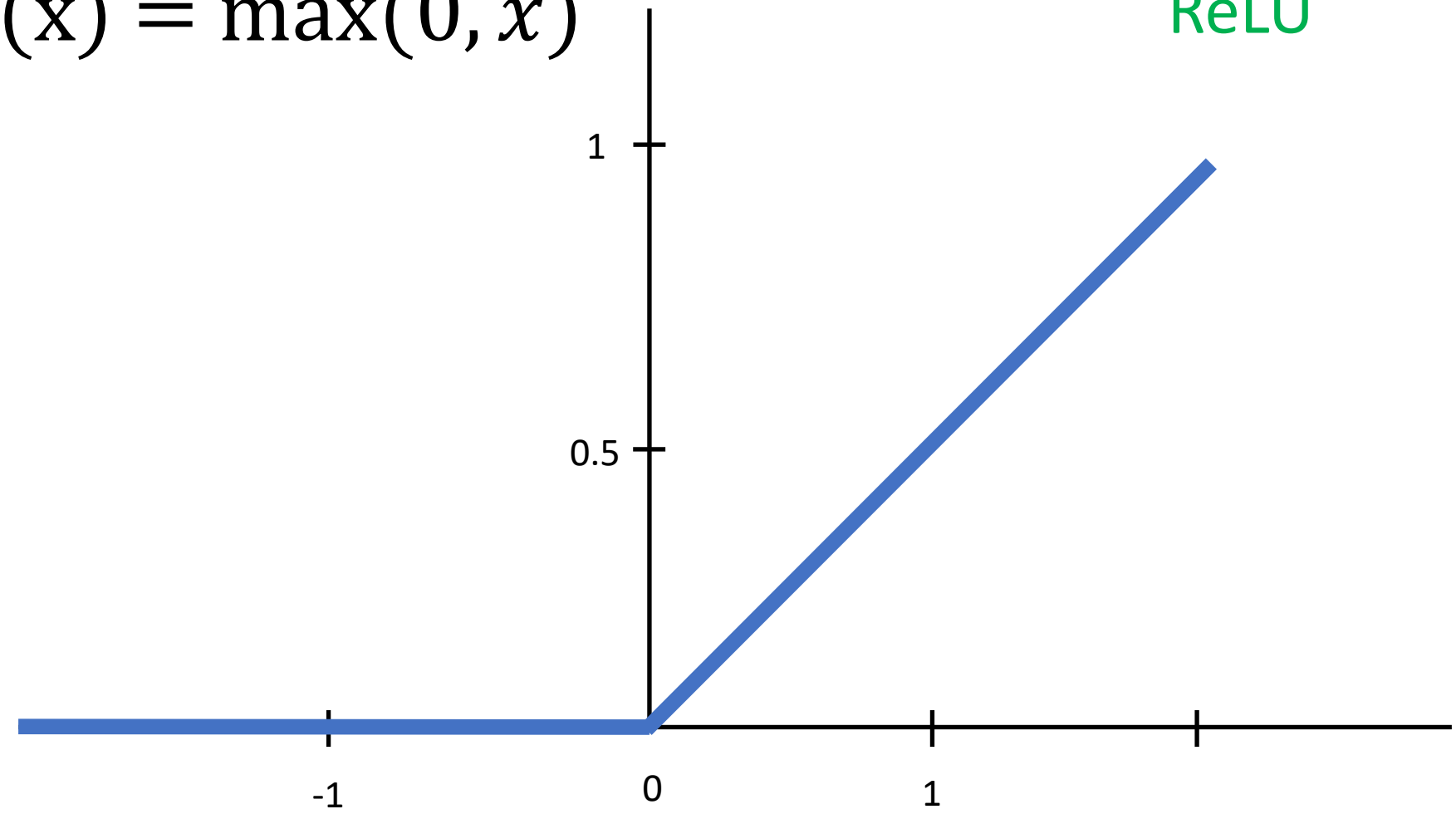
Issues with sigmoid and tanh

The Vanishing Gradient Problem

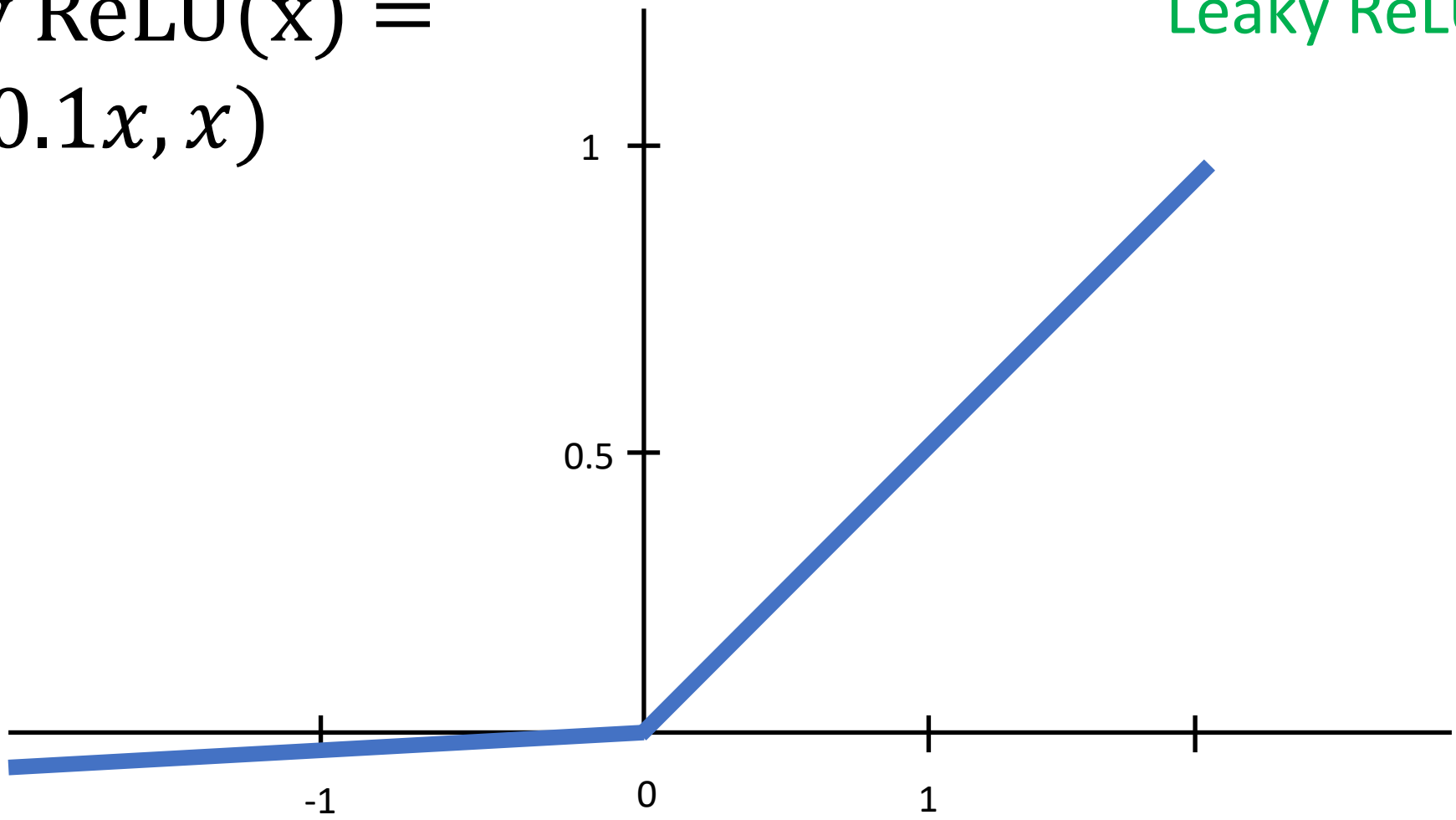


derivative = $\Delta y / \Delta x$

$$\mathrm{ReLU}(x) = \max(0, x)$$



$$\text{Leaky ReLU}(x) = \max(0.1x, x)$$



Activation Function

- PyTorch: torch.nn.ReLU() / torch.nn.LeakyReLU() …

(https://pytorch.org/docs/stable/nn.html)

- TensorFlow-Keras: tf.keras.activations.relu()

(https://www.tensorflow.org/api_docs/python/tf/keras/activations)