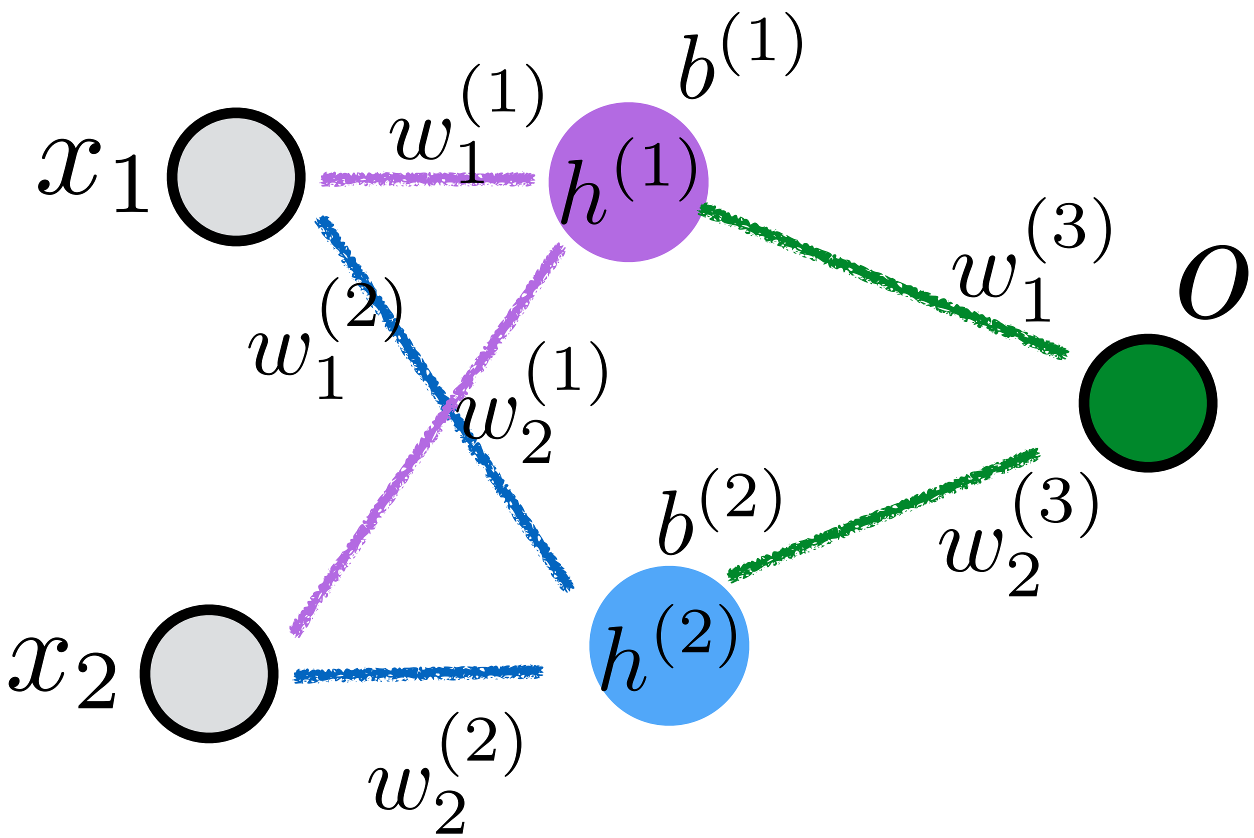
Short review

Basic construction of neural networks

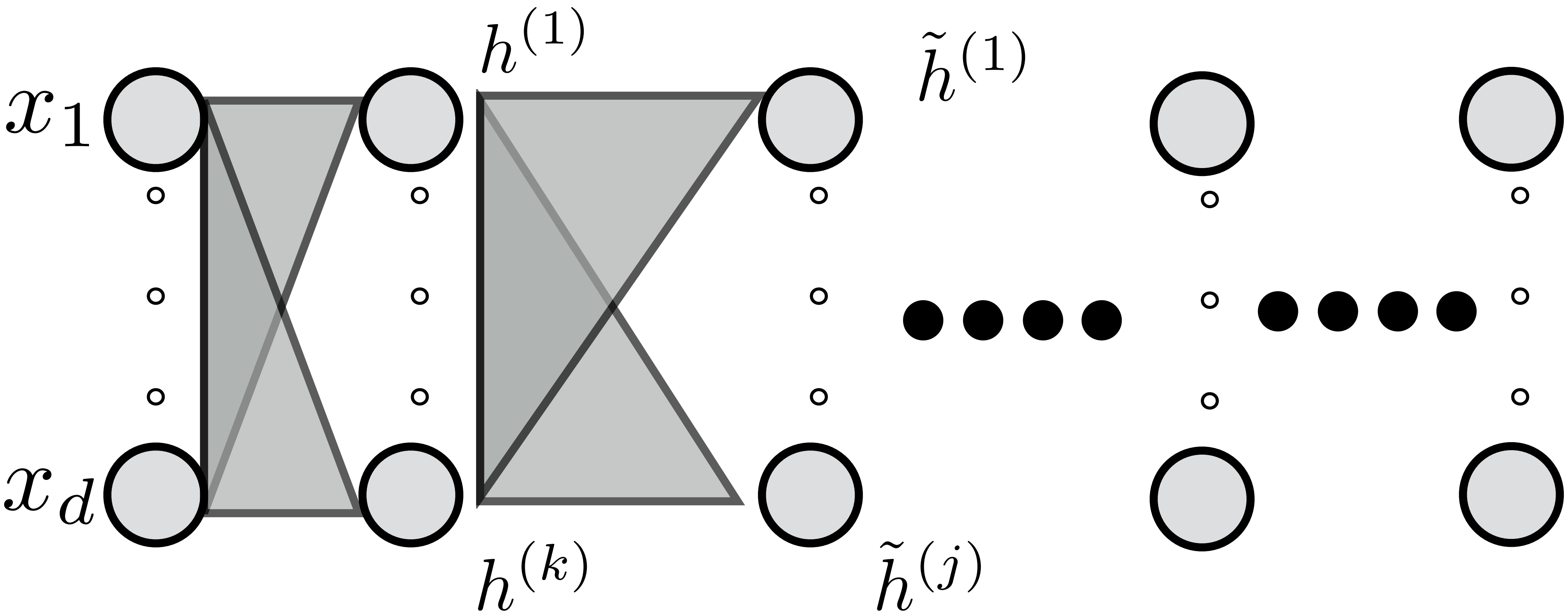


$$h^{(1)} = \sigma(w_1^{(1)} x_1 + w_2^{(1)} x_2 + b^{(1)})$$

$$h^{(2)} = \sigma(w_1^{(2)} x_1 + w_2^{(2)} x_2 + b^{(2)})$$

$$o = \sigma(w_1^{(3)} h^{(1)} + w_2^{(3)} h^{(2)} + b^{(3)})$$

Basic construction of neural networks

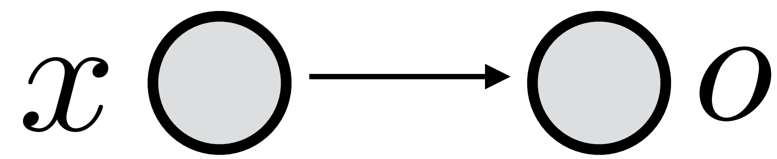


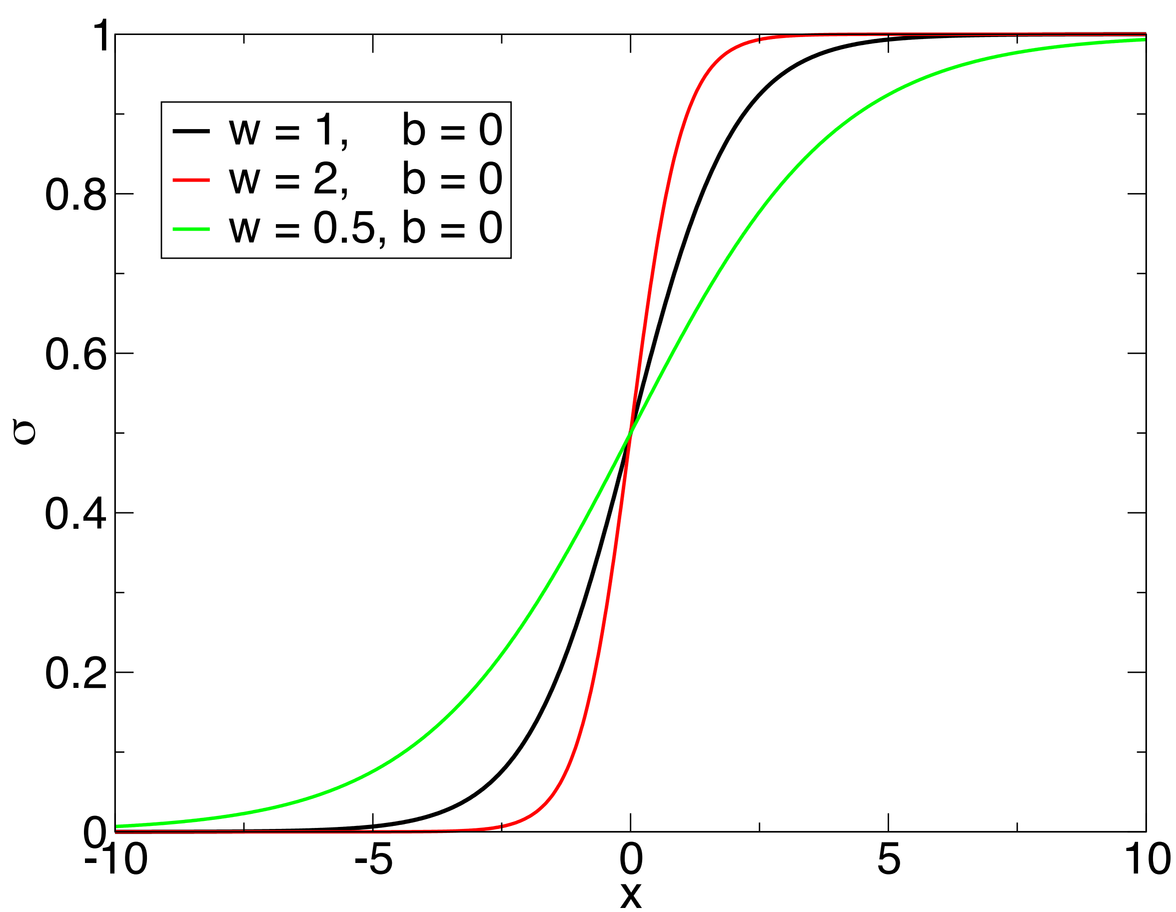
$$h^{(1)} = \sigma(\vec{w}_1 \cdot \vec{x} + b_1)$$
$$h^{(k)} = \sigma(\vec{w}_k \cdot \vec{x} + b_k)$$
$$\tilde{h}^{(j)} = \sigma(\vec{w}_j \cdot \vec{h} + b_j)$$

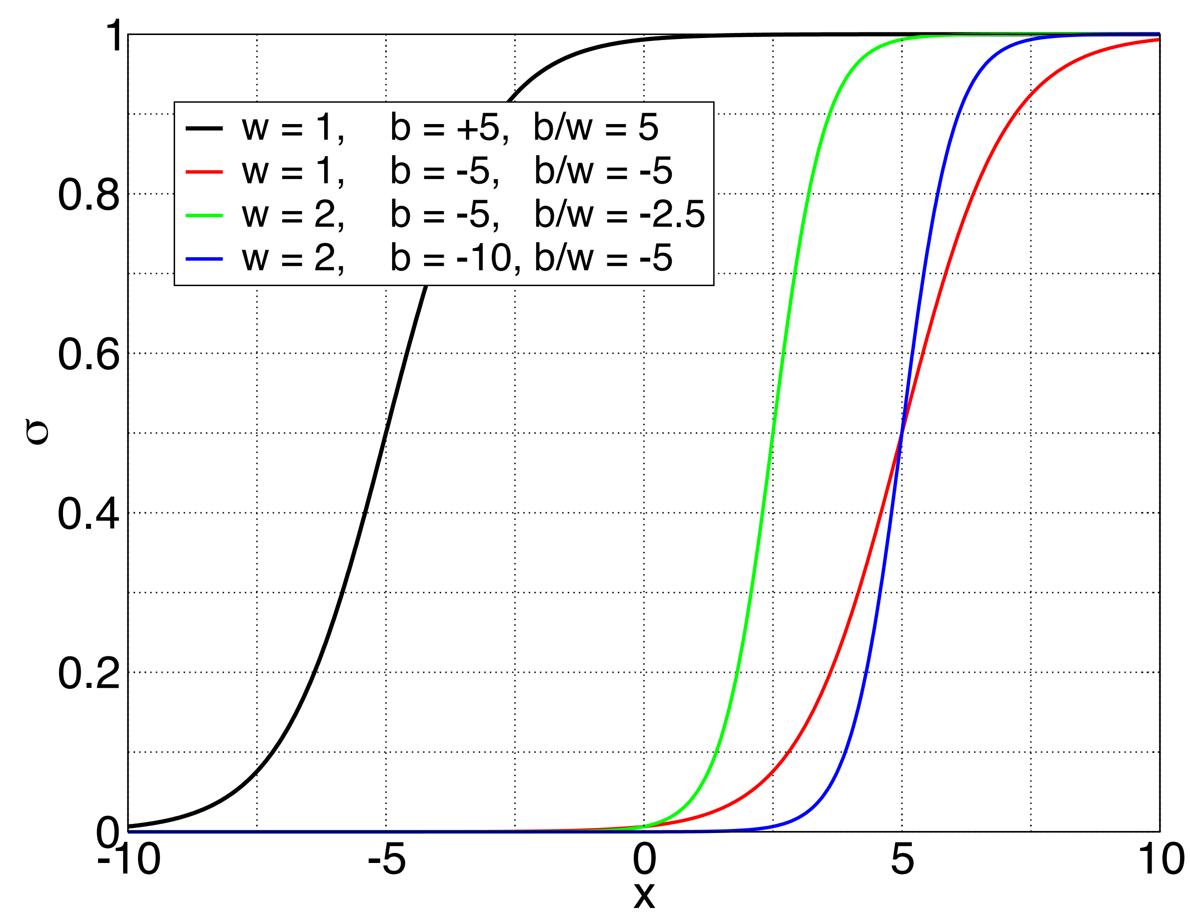
Role of parameters w and b

$x \in \mathbb{R}$

$$o = \frac{1}{1 + \exp(-wx - b)}$$



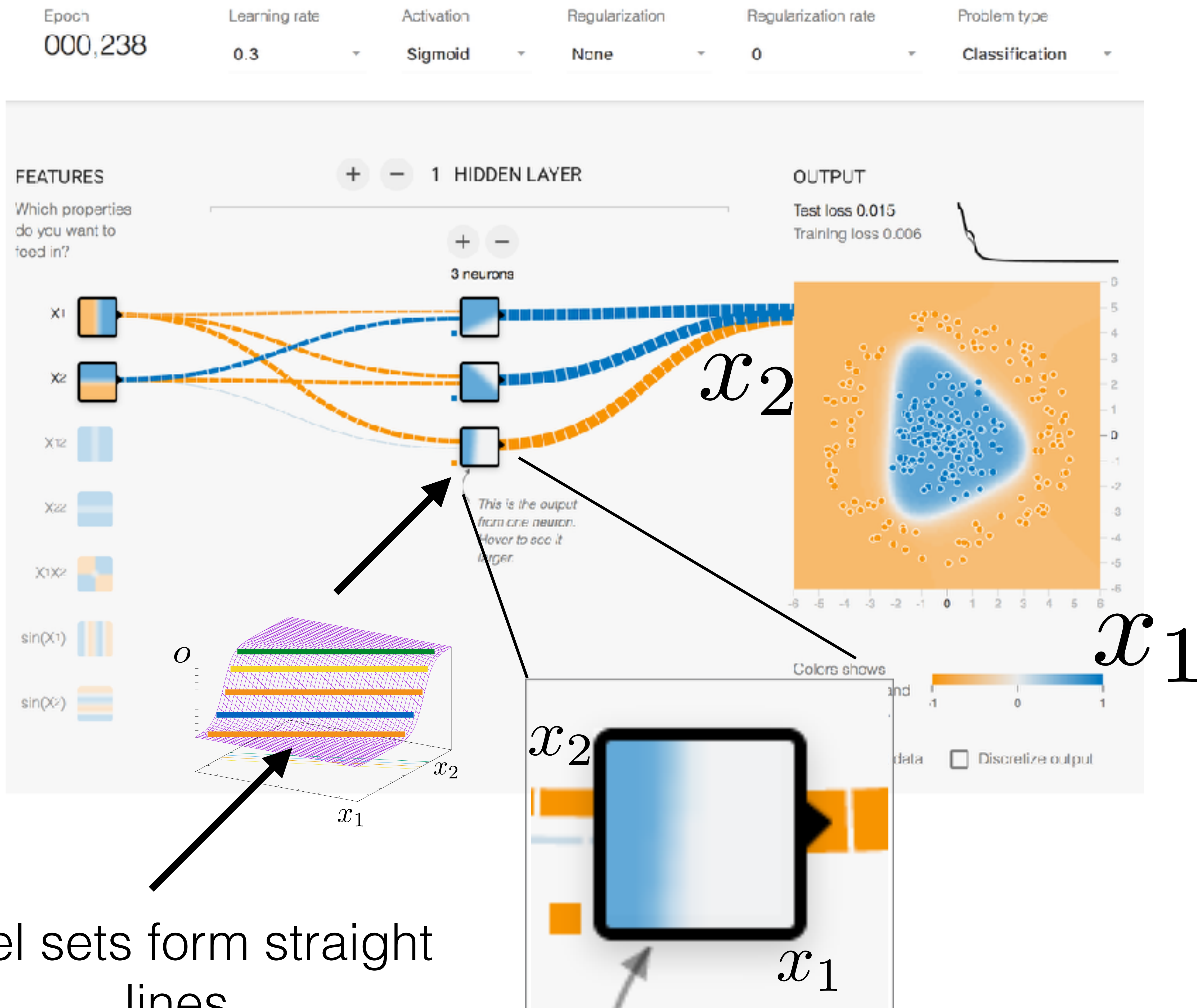




Role of parameters w and b

Do it yourself : plot ReLU graph for various w b

Function picture of network



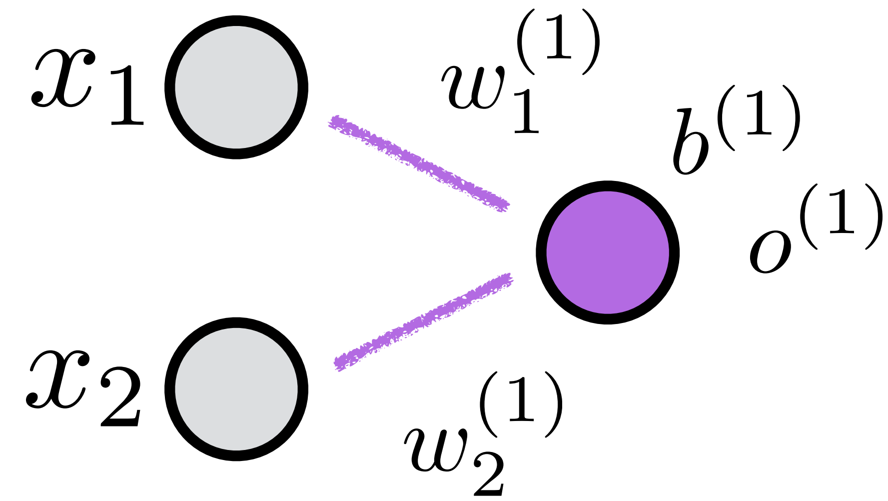
level sets form straight lines

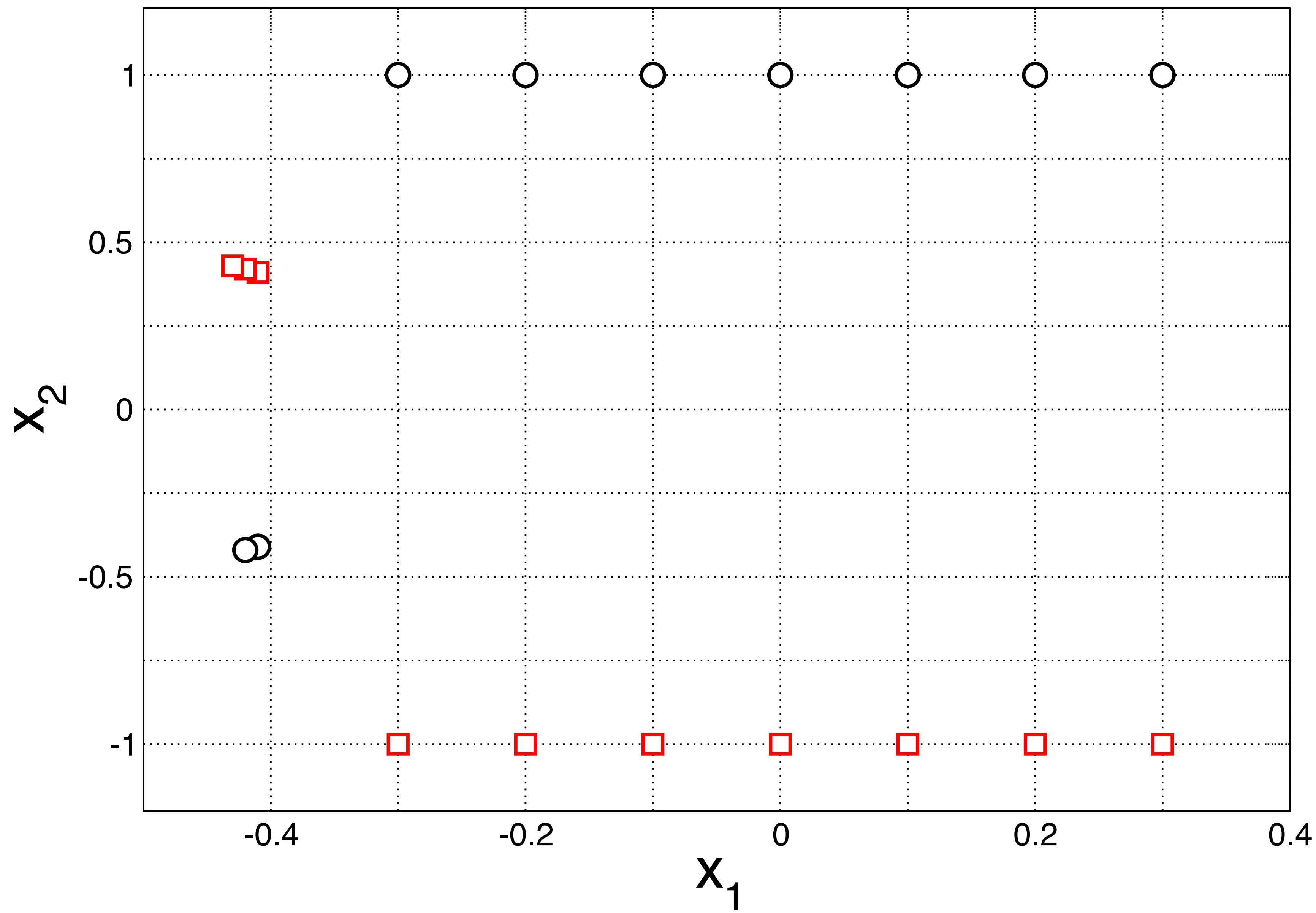
Local minimum problem

$$o_i = \sigma(w_1 x_1 + w_2 x_2) \text{ with } w_2 = 1$$

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

$$C(w_1) = \frac{1}{n} \sum_i (y_i - o_i)^2$$

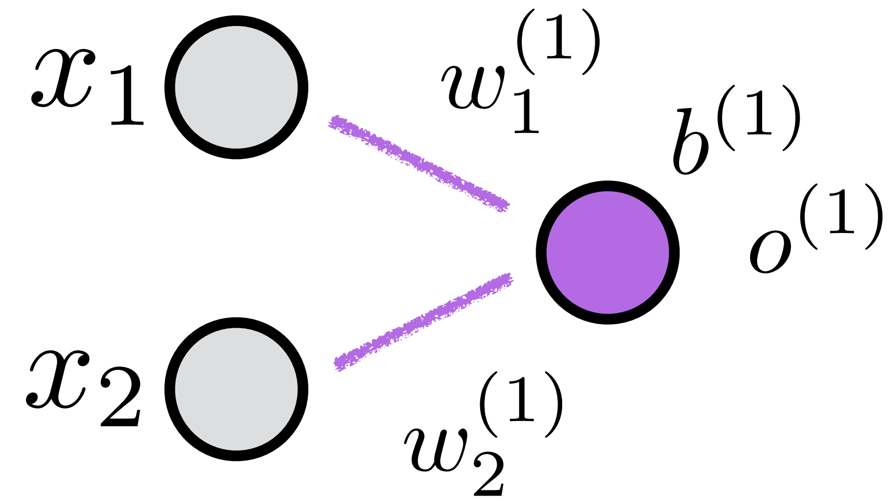


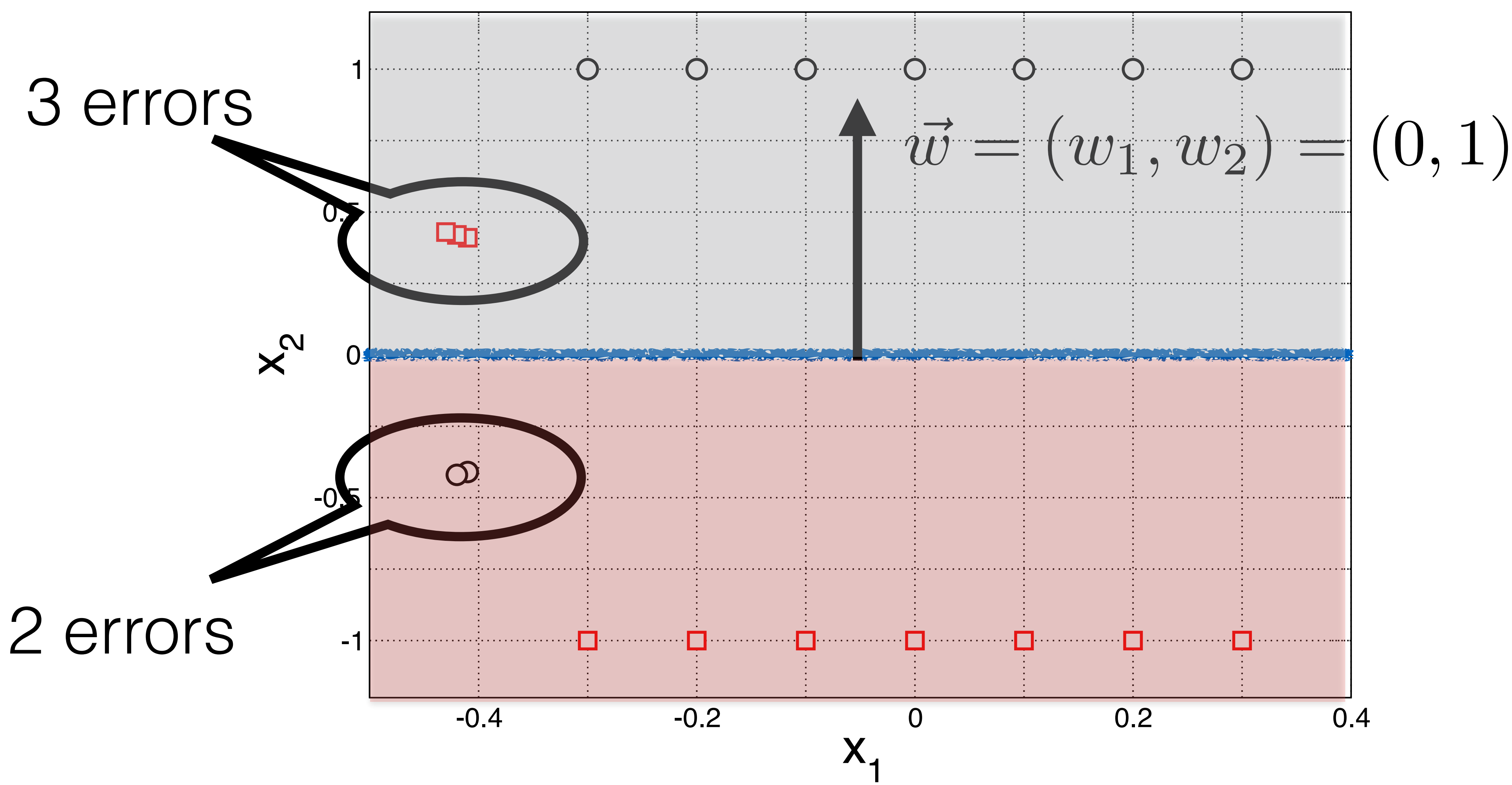


$$o_i = \sigma(w_1 x_1 + w_2 x_2) \text{ with } w_2 = 1$$

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

$$C(w_1) = \frac{1}{n} \sum_i (y_i - o_i)^2$$

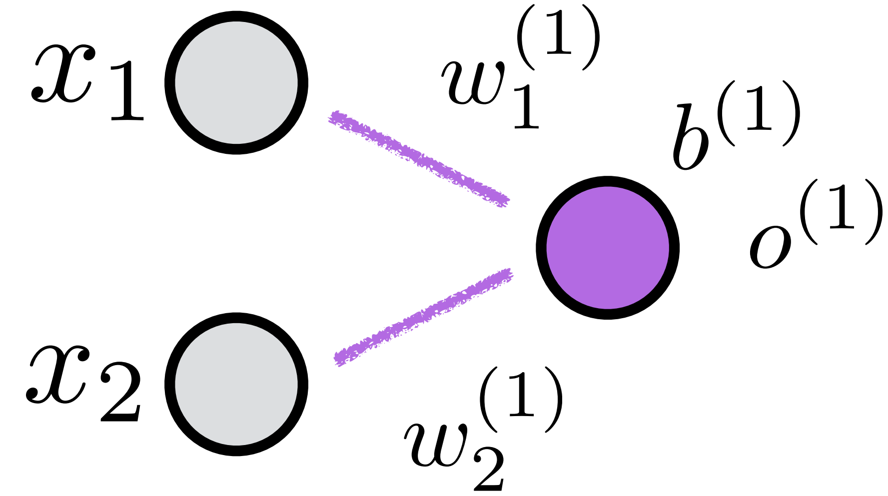


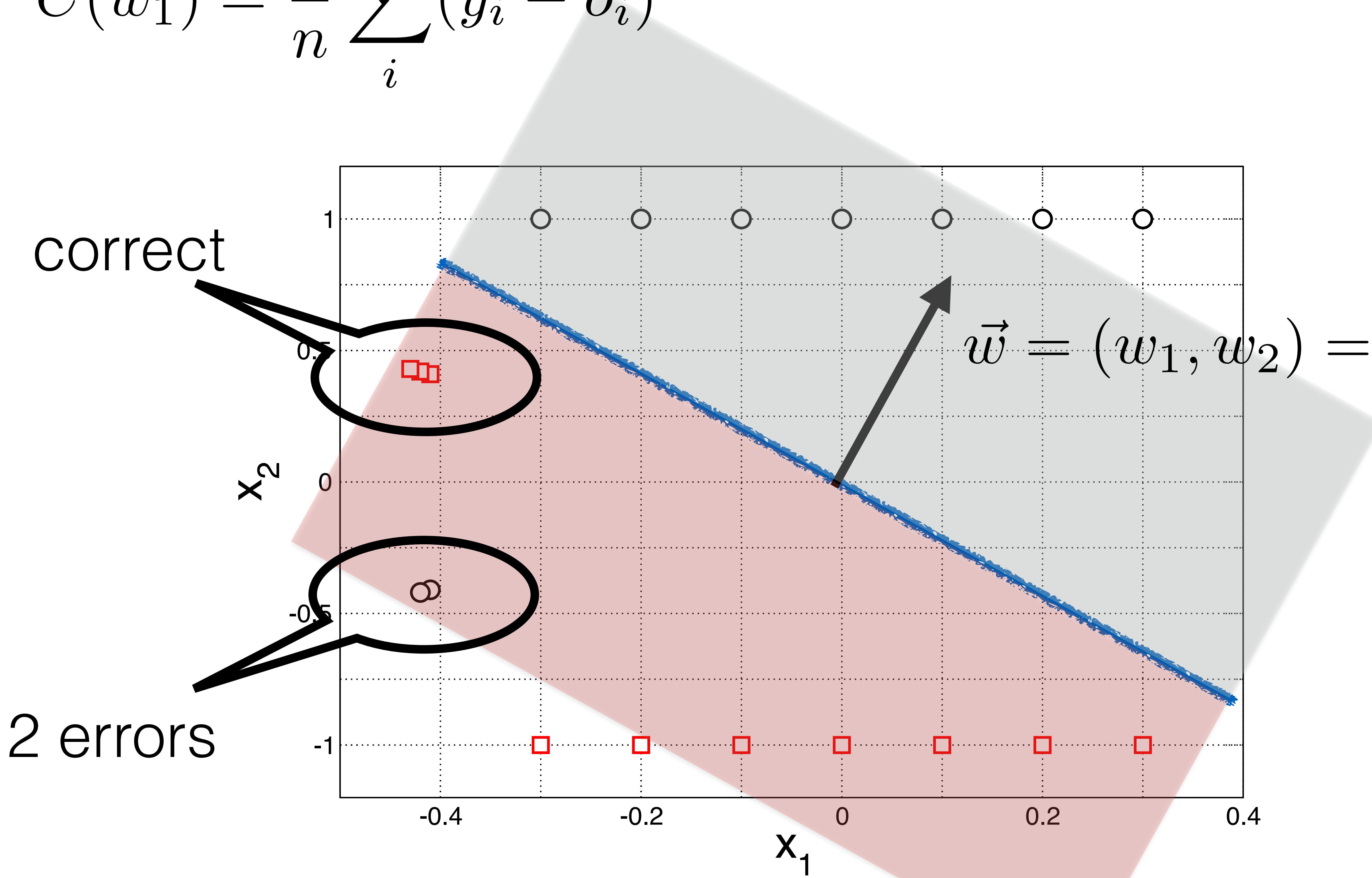


$$o_i = \sigma(w_1 x_1 + w_2 x_2) \text{ with } w_2 = 1$$

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

$$C(w_1) = \frac{1}{n} \sum_i (y_i - o_i)^2$$

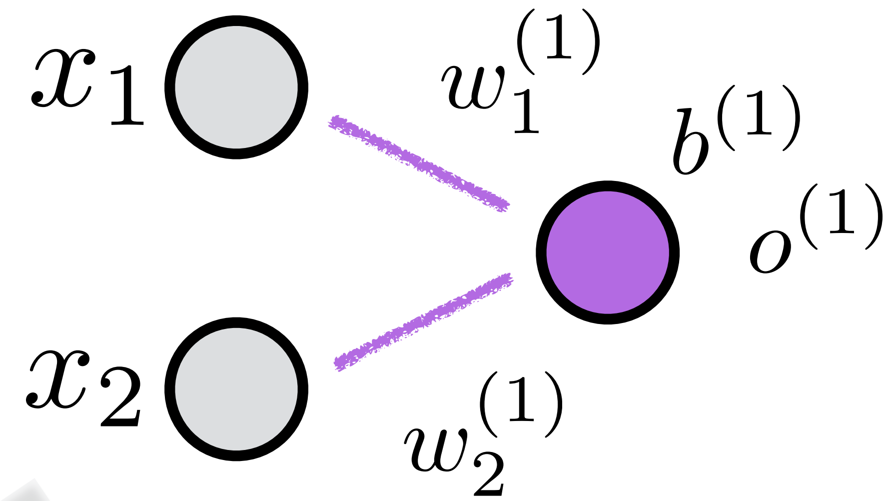


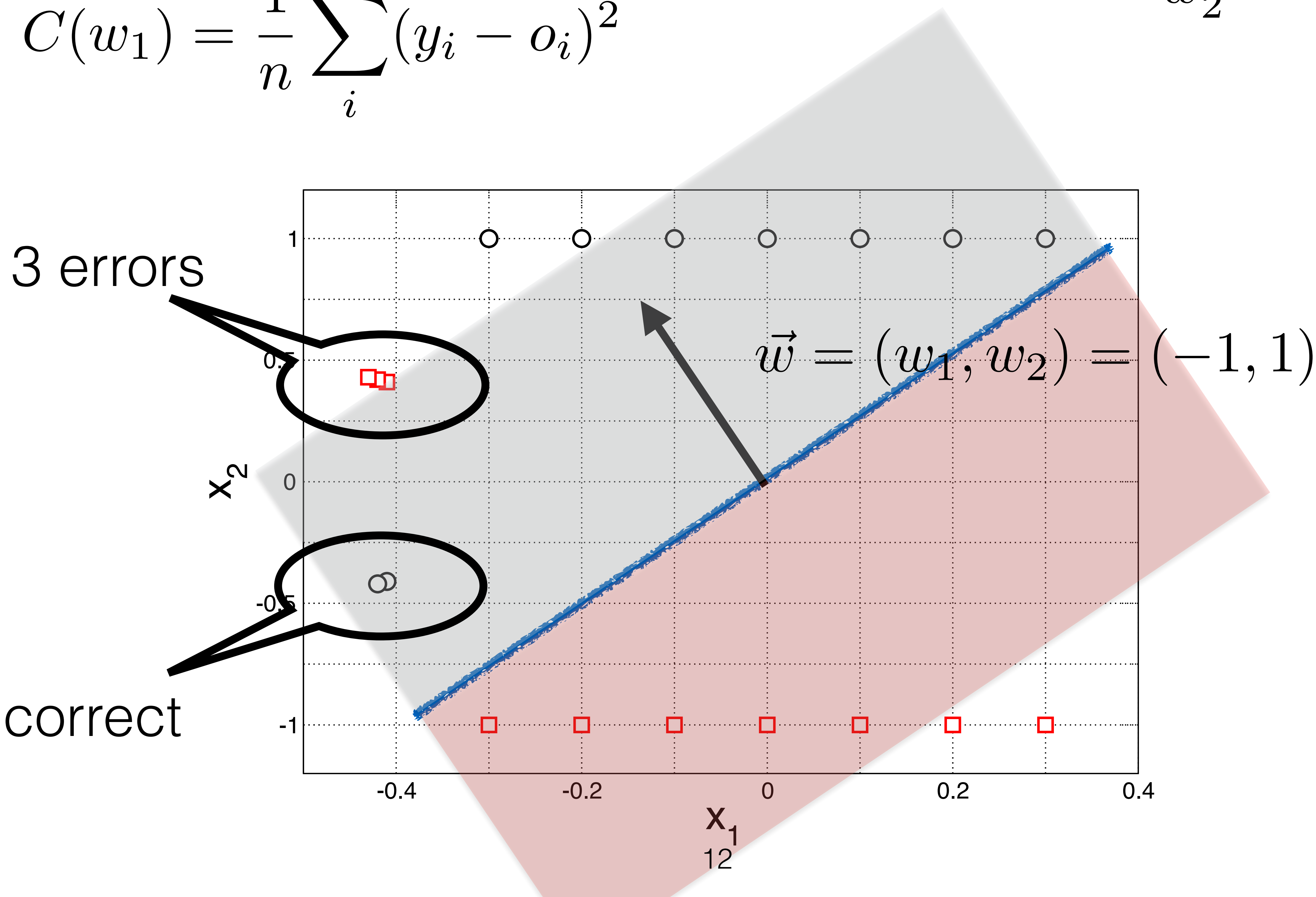


$$o_i = \sigma(w_1 x_1 + w_2 x_2) \text{ with } w_2 = 1$$

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

$$C(w_1) = \frac{1}{n} \sum_i (y_i - o_i)^2$$

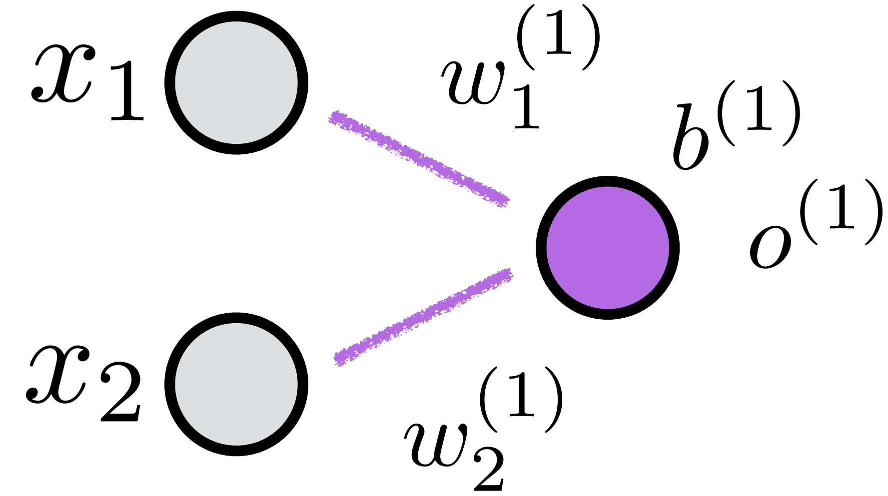


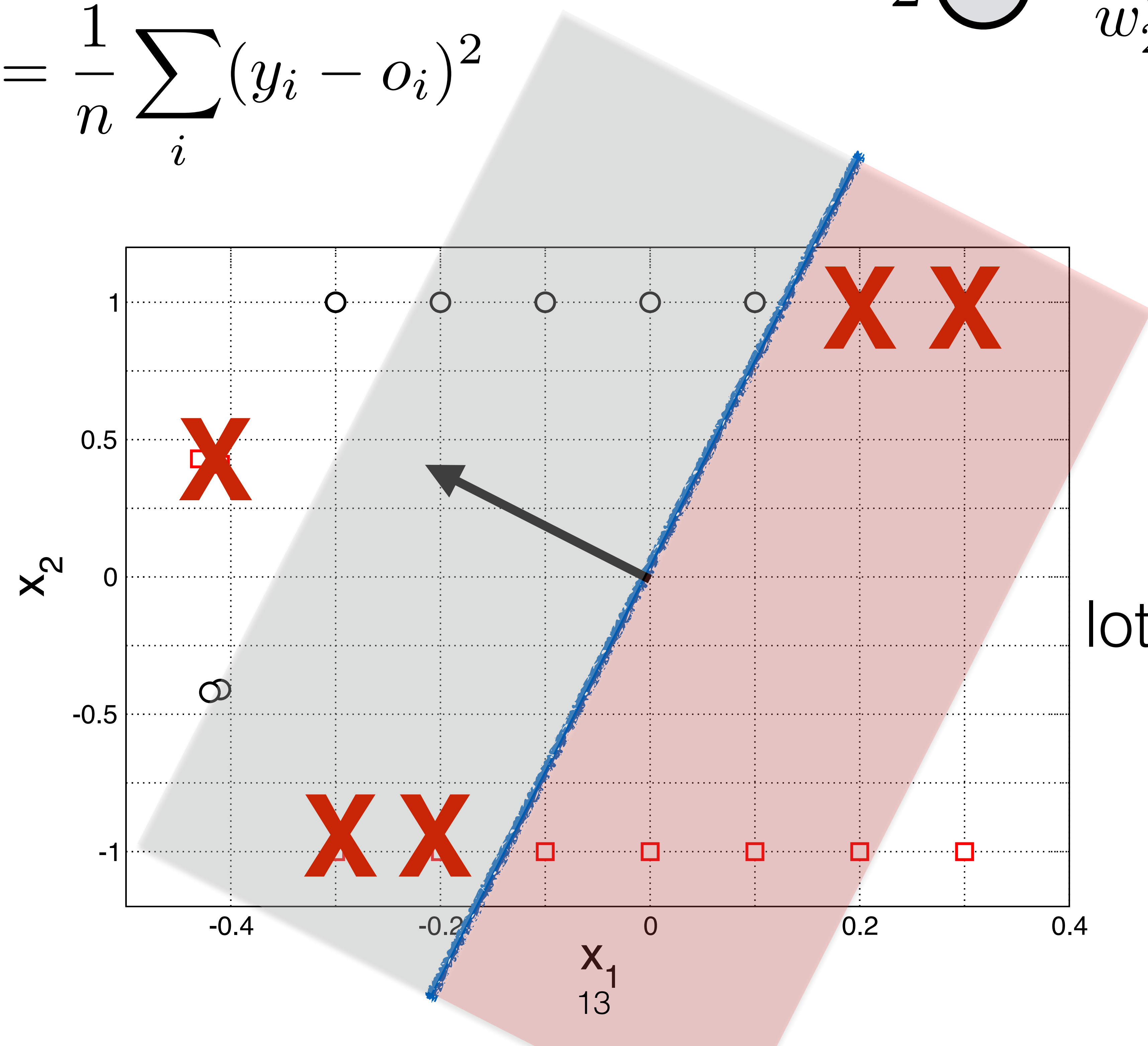


$$o_i = \sigma(w_1 x_1 + w_2 x_2) \text{ with } w_2 = 1$$

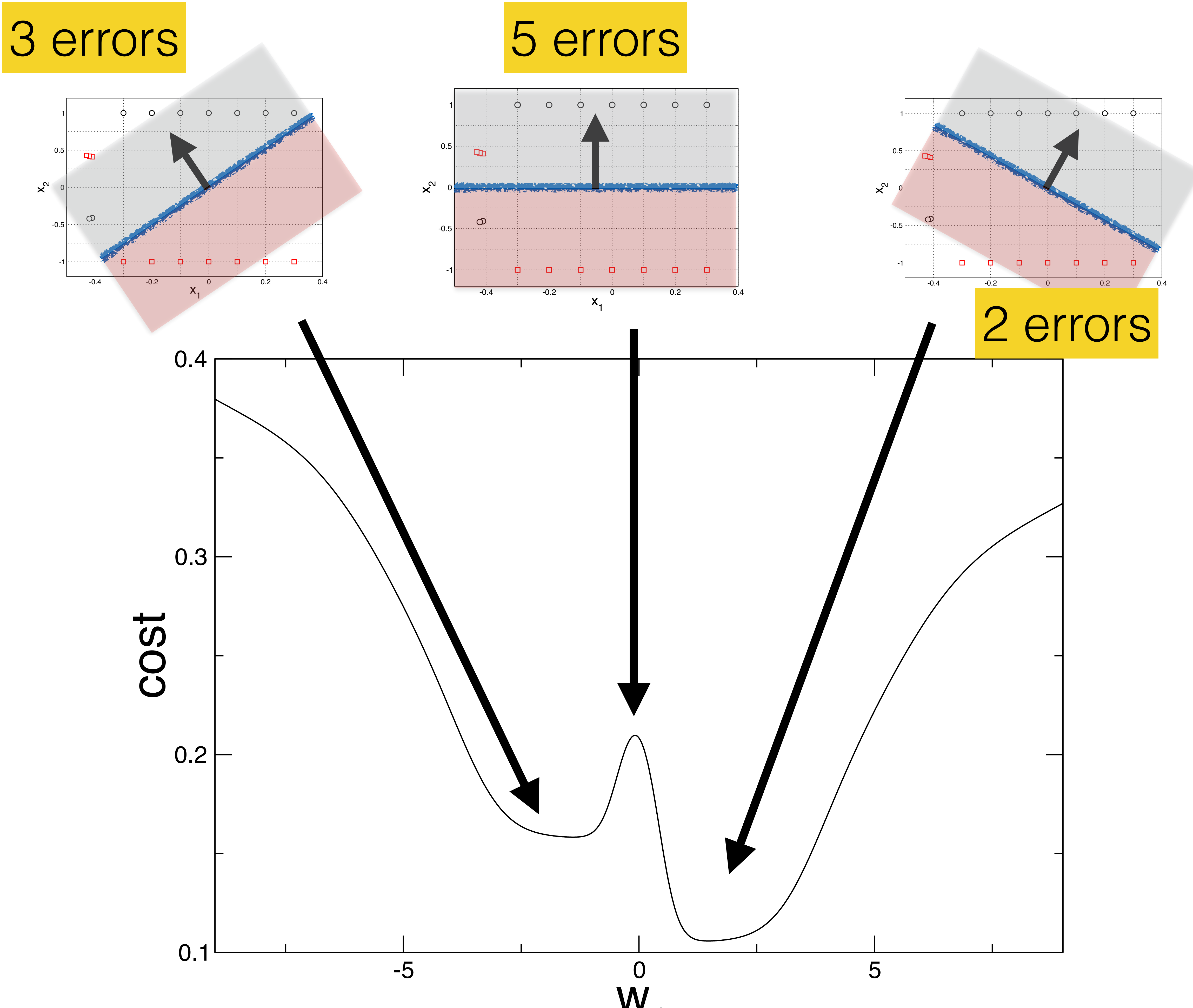
$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$

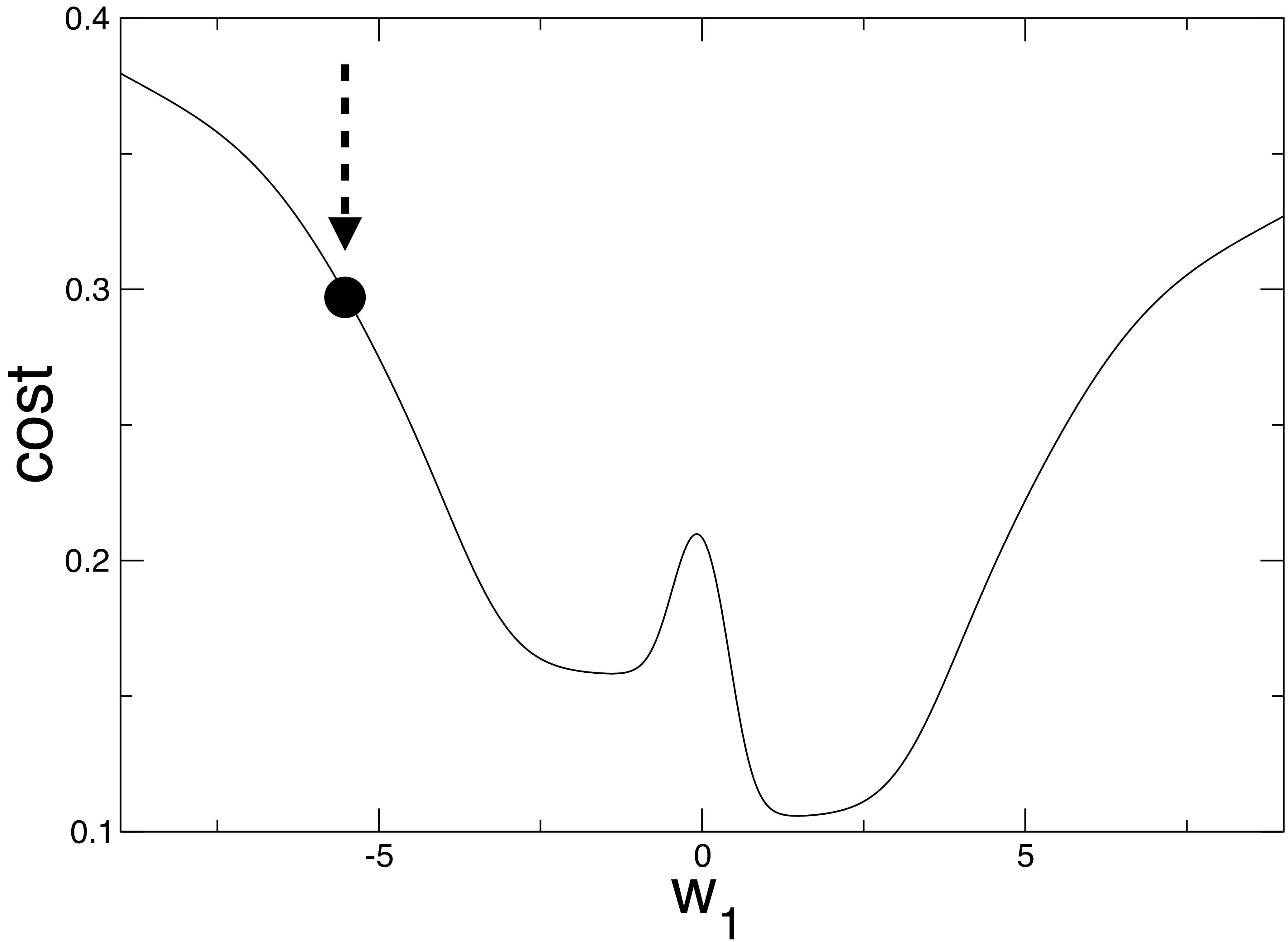
$$C(w_1) = \frac{1}{n} \sum_i (y_i - o_i)^2$$



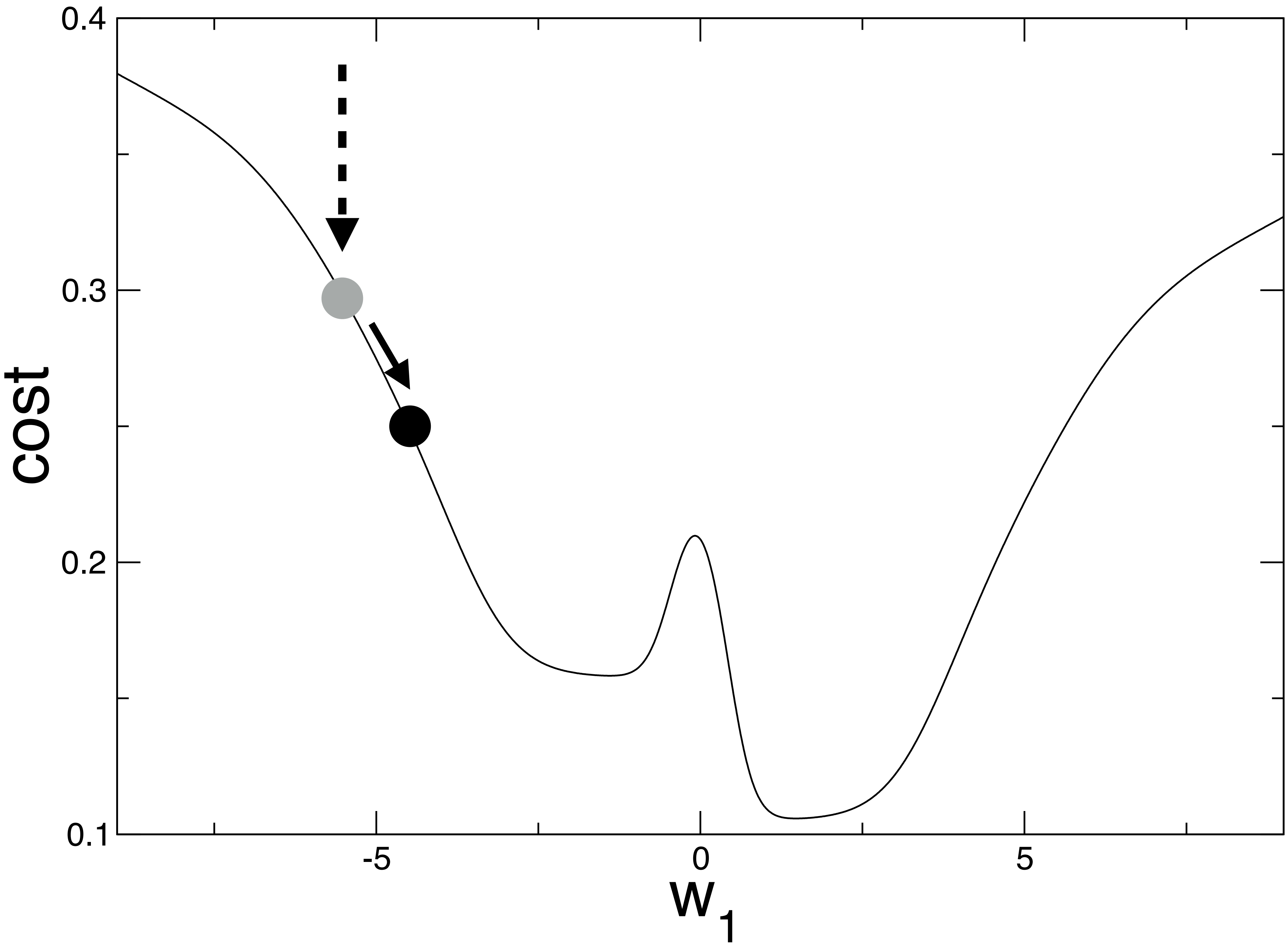


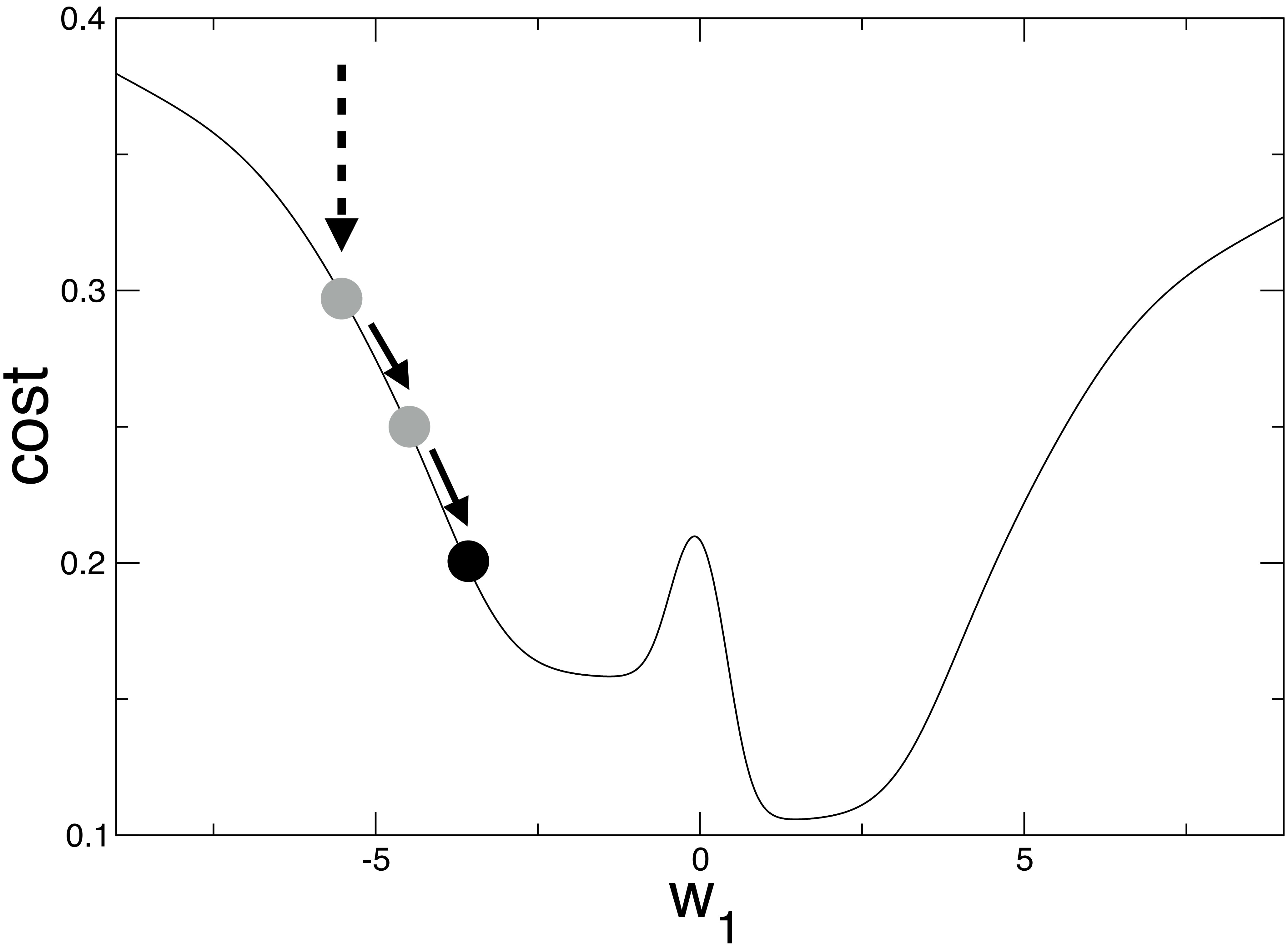
lots of errors!


3 errors
5 errors
2 errors
x_1
x_2
cost
w_1
0.4
0.3
0.2
0.1
-5
0
5

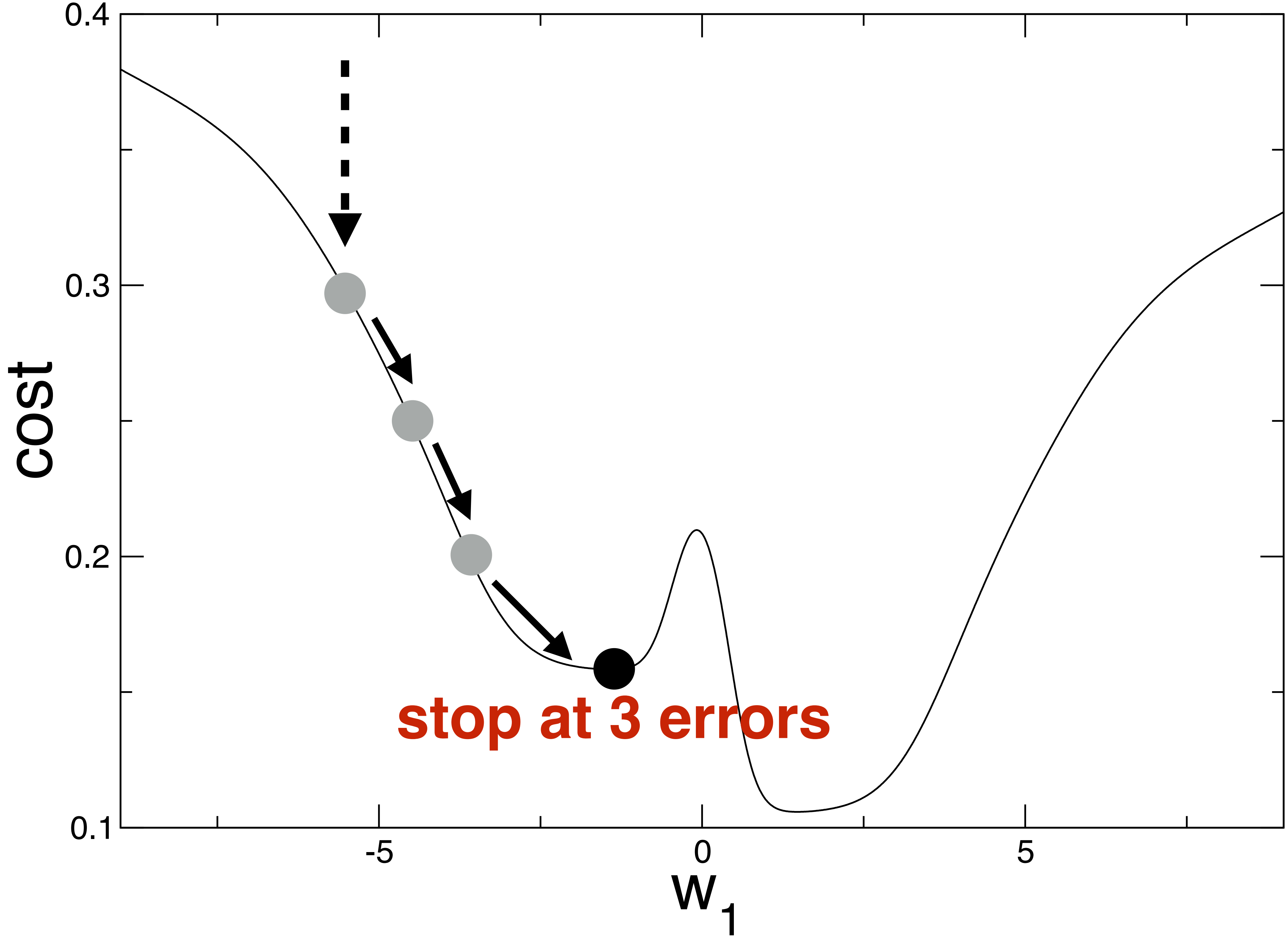


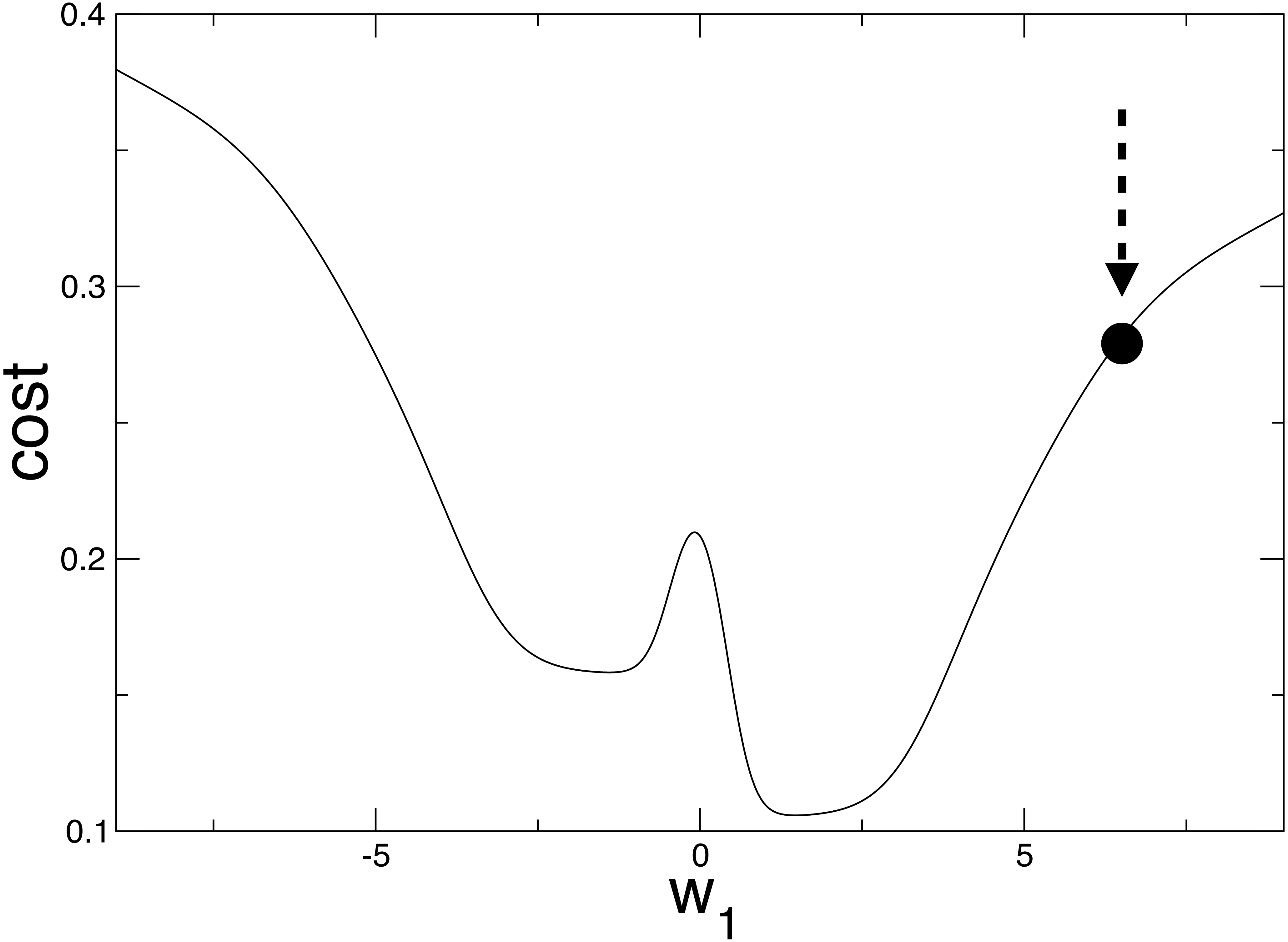
0.4
0.3
0.2
0.1
cost
-5
0
5
w1

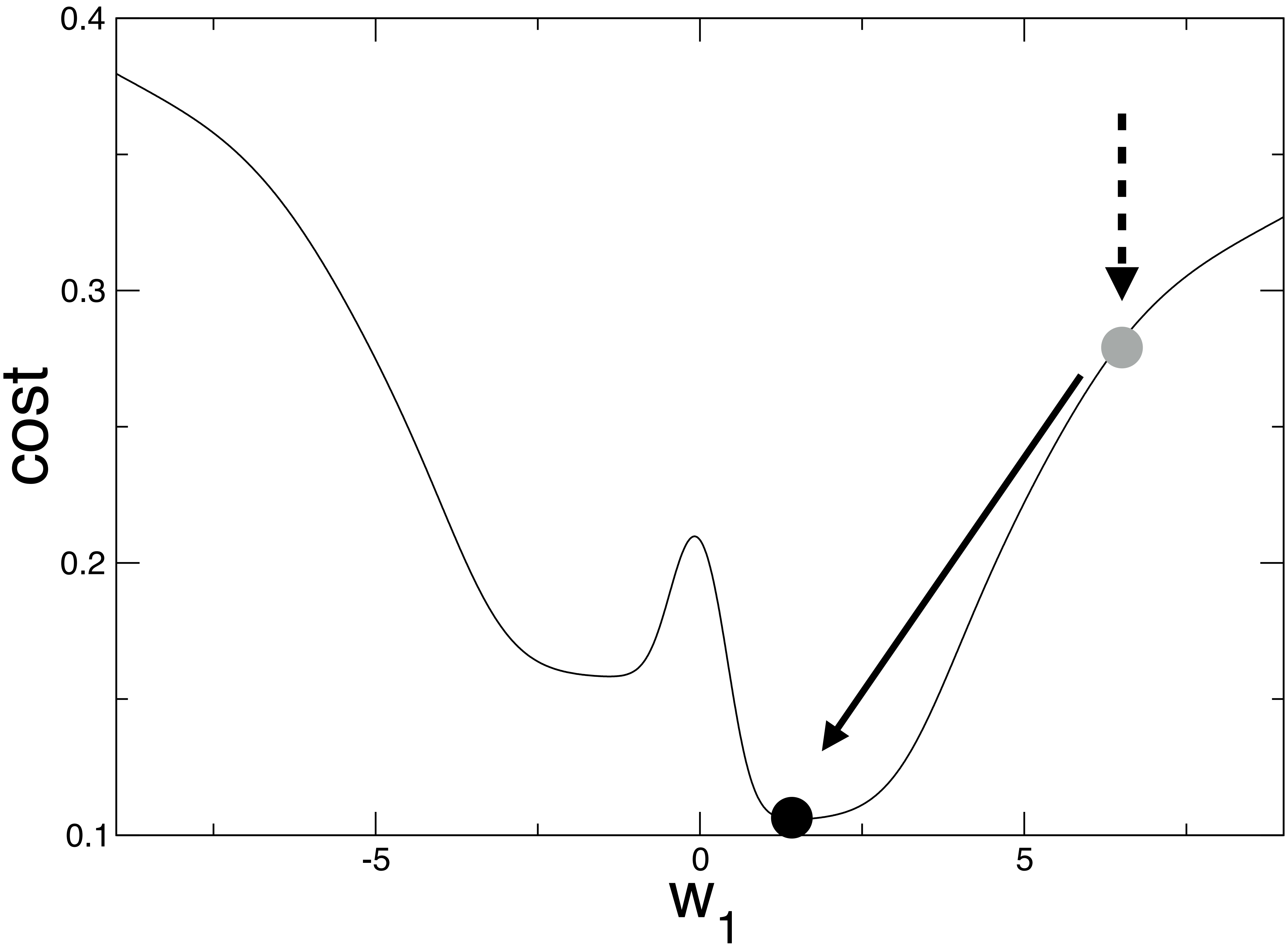

0.4
0.3
0.2
0.1
cost
-5
0
5
w_1



0.4
0.3
0.2
0.1
cost
-5
0
5
w_1


0.4
0.3
0.2
0.1
cost
-5
0
5
w1
stop at 3 errors

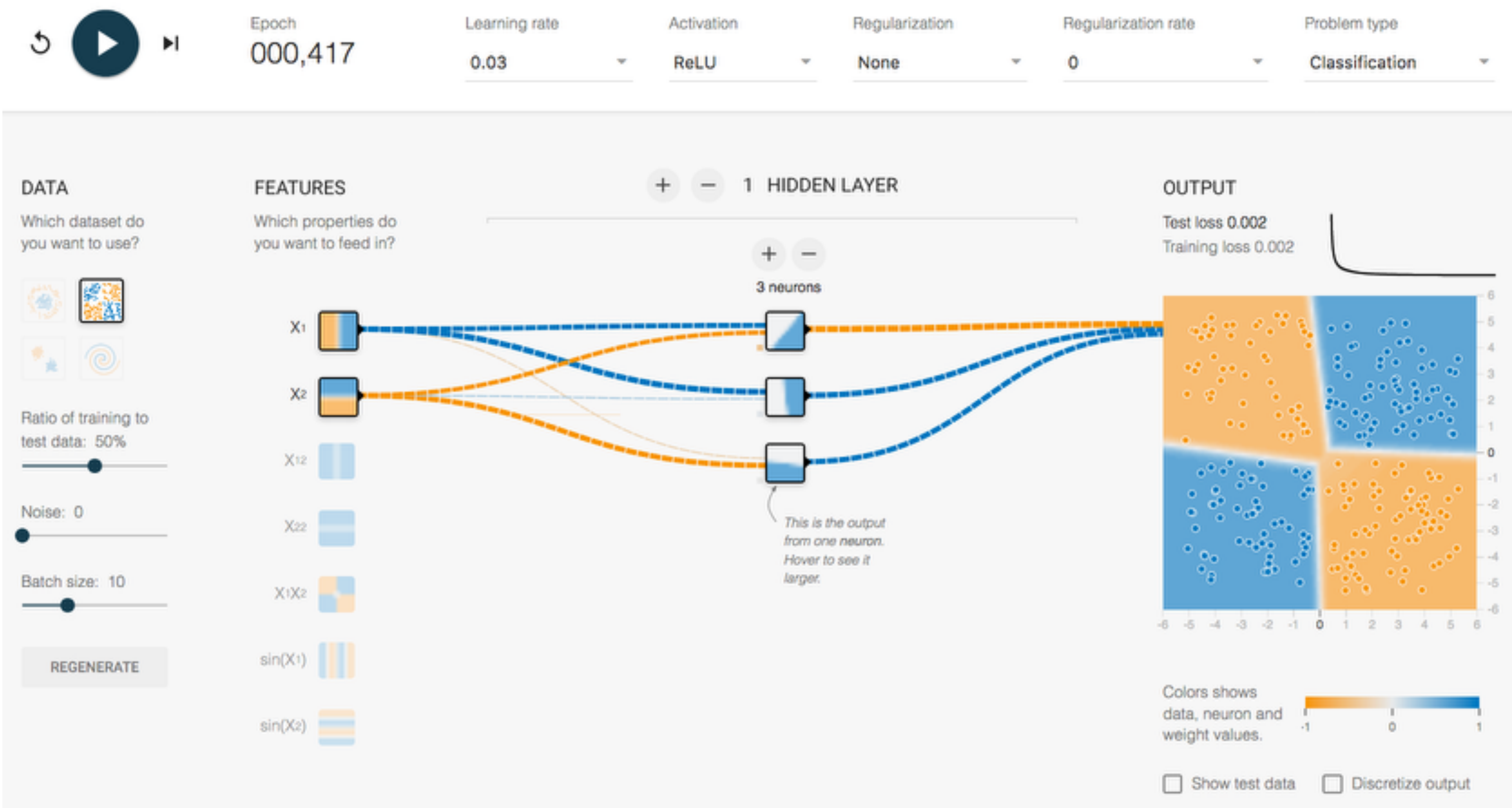

0.4
0.3
0.2
0.1
-5
0
5
cost
w1



0.4
0.3
0.2
0.1
cost
-5
0
5
w1

show playground XOR example

Good solution example

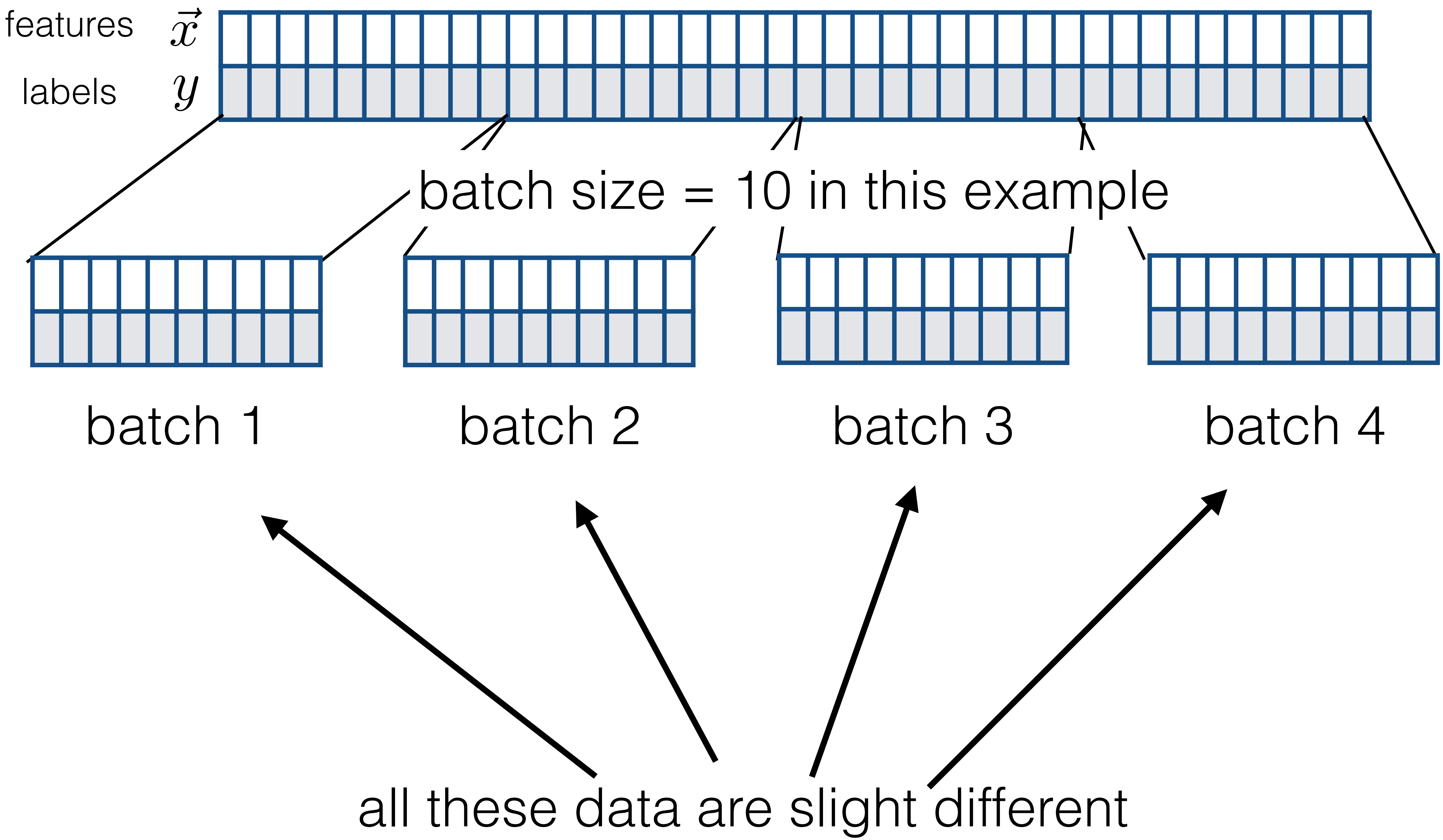


Local minimum examples

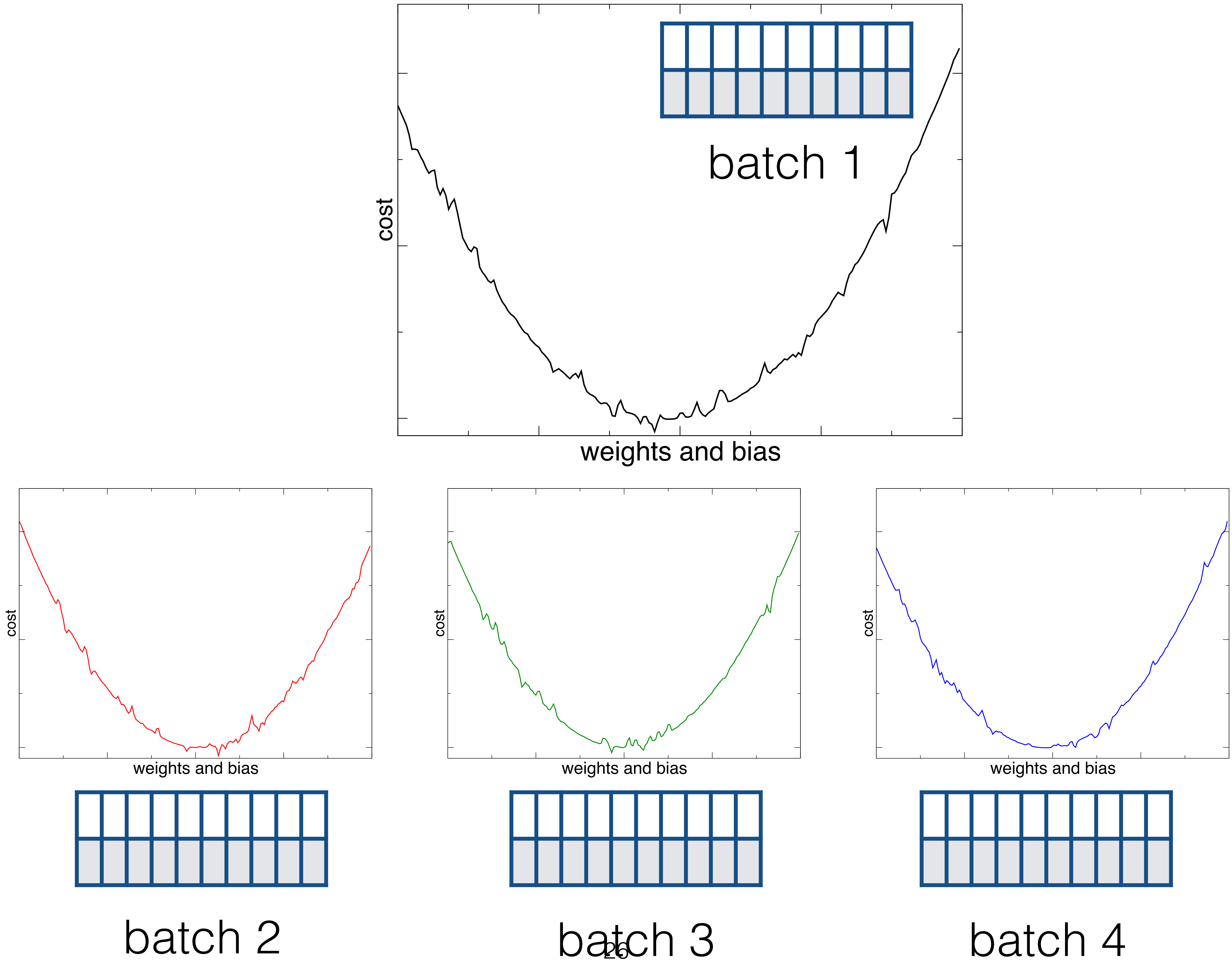
Strategies for overcoming local minimum problem

1. Stochastic gradient descend
2. Adam method, momentum

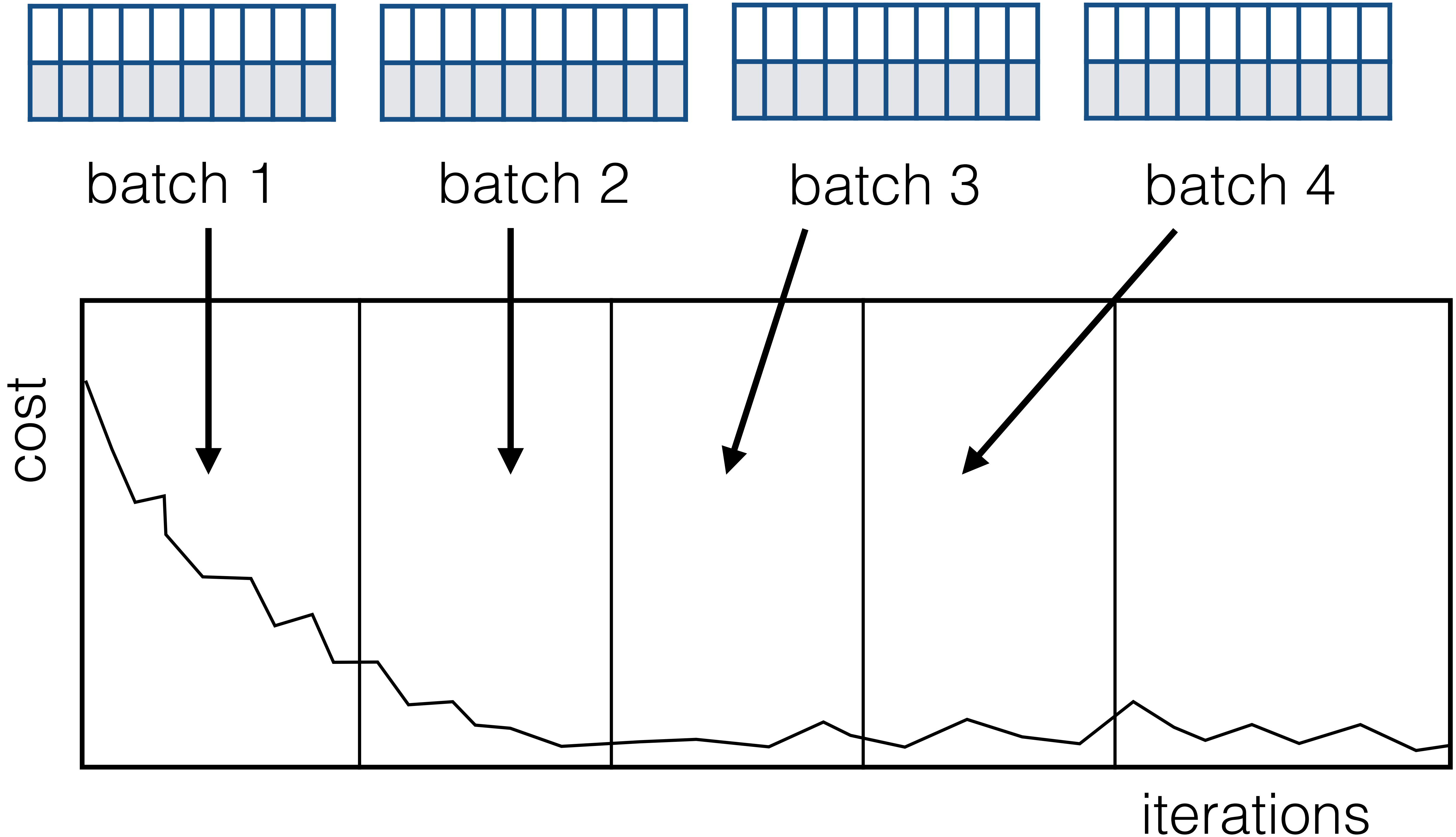
Minibatch gradient descend

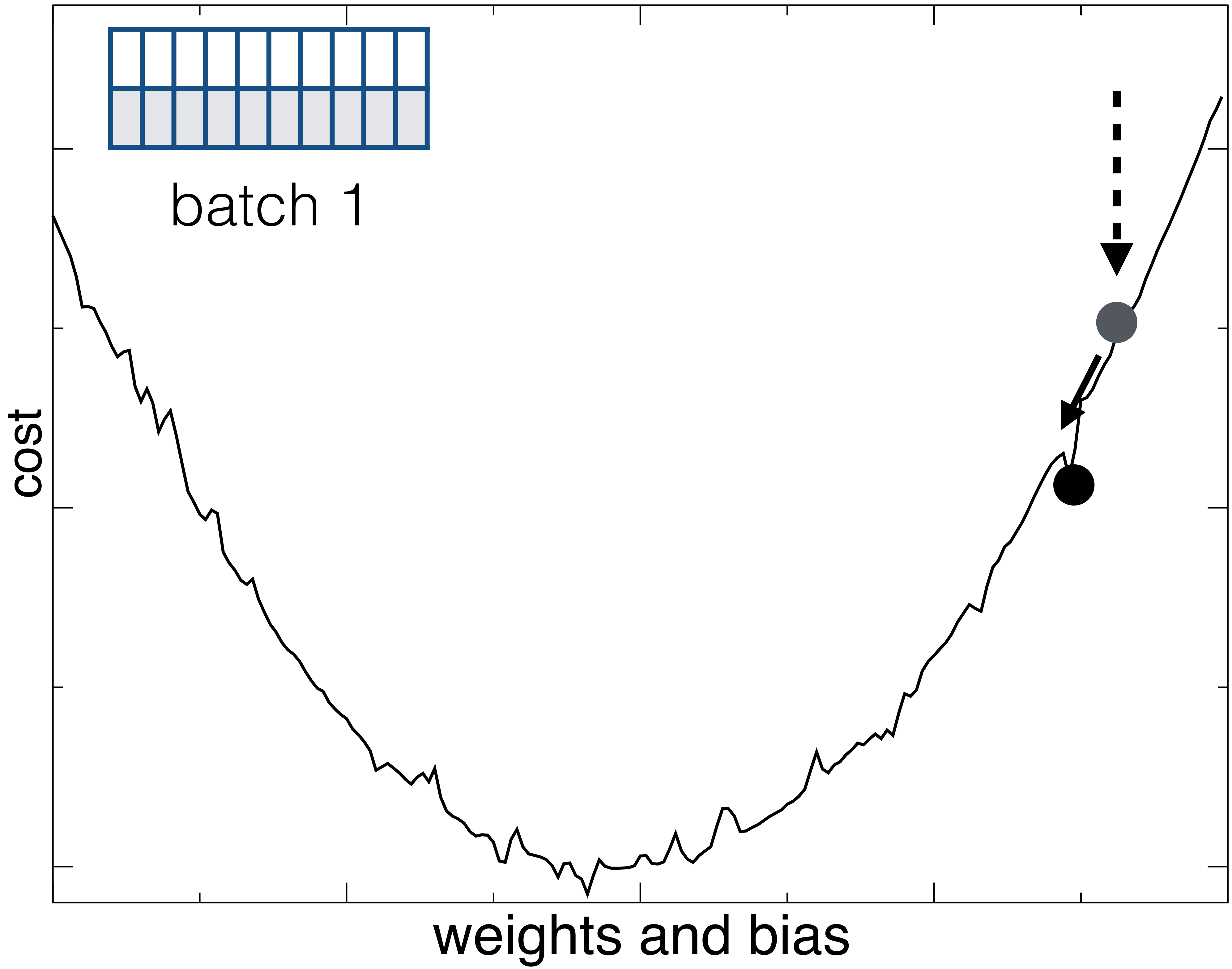


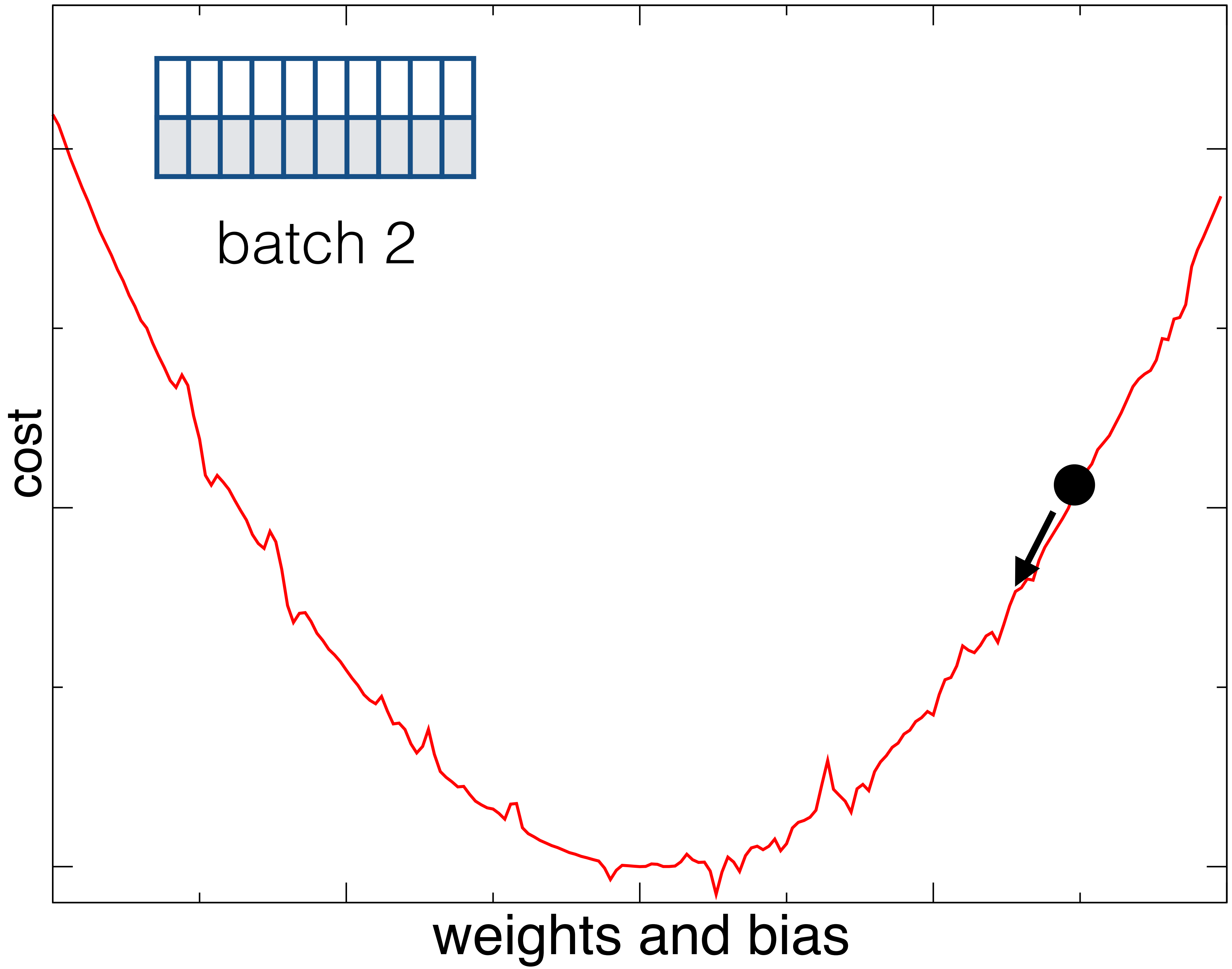
cost surfaces are different for different data sets

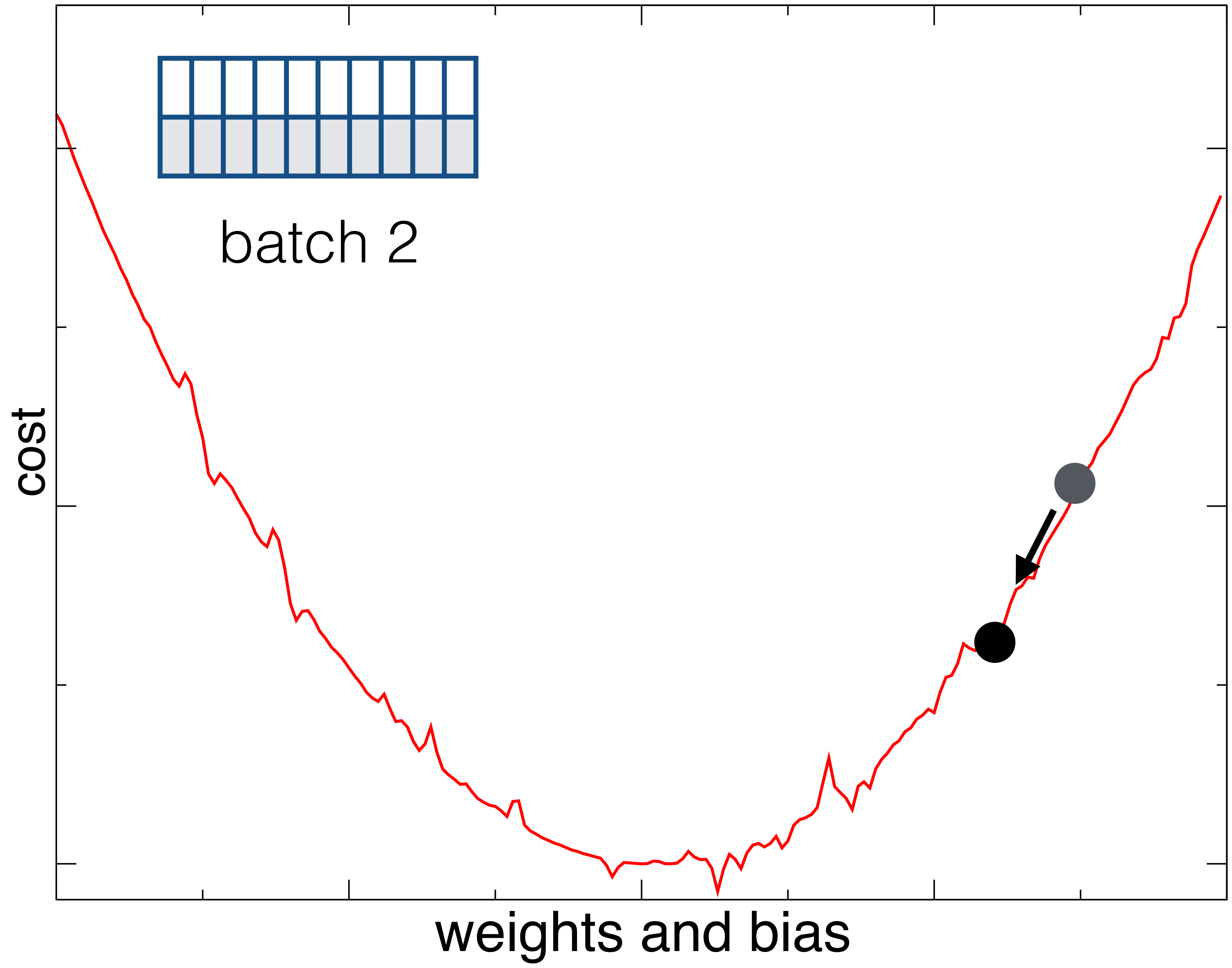

batch 1
cost
weights and bias
cost
weights and bias
cost
weights and bias
cost
weights and bias
batch 2
batch 3
batch 4

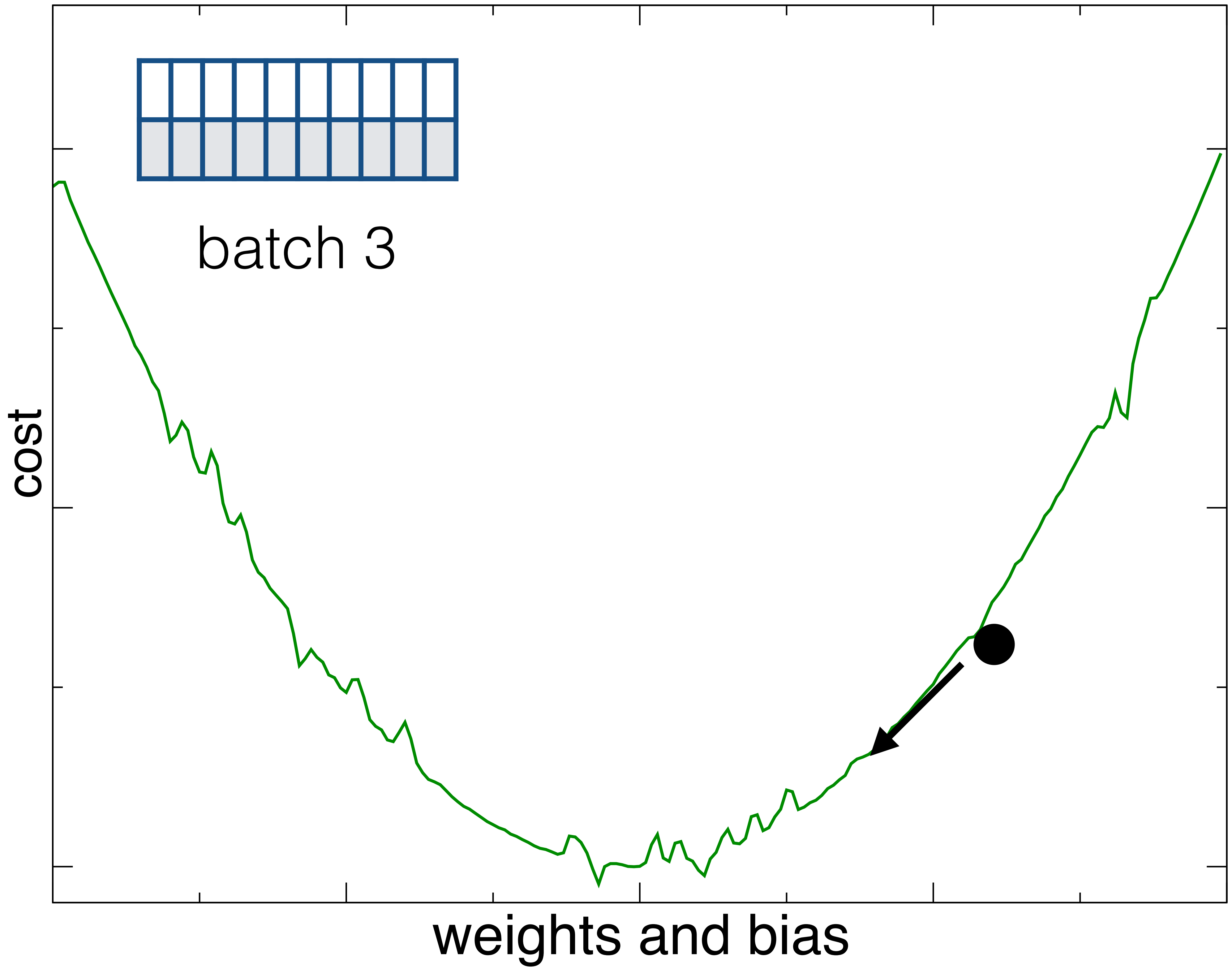
Always remember to shuffle the data

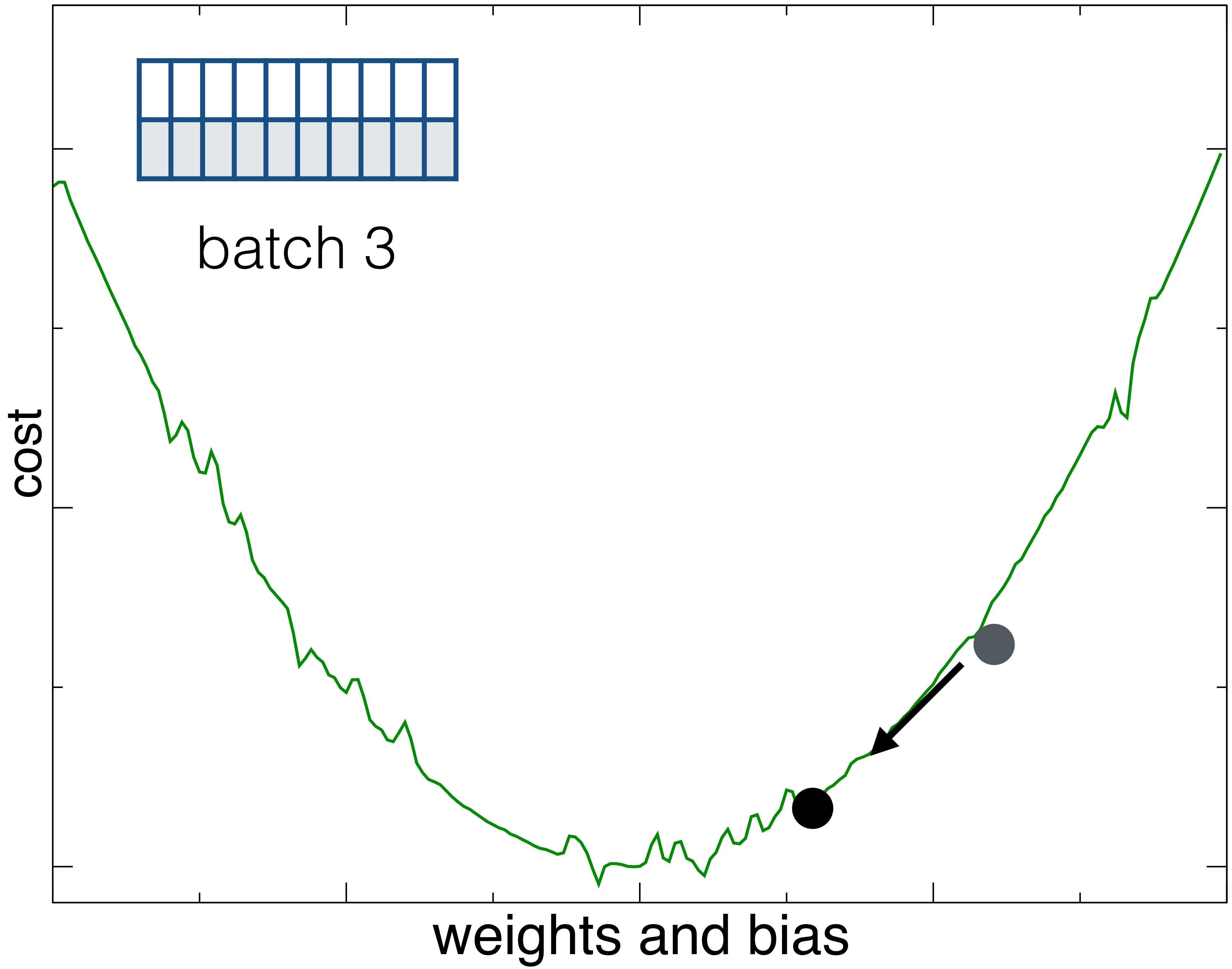


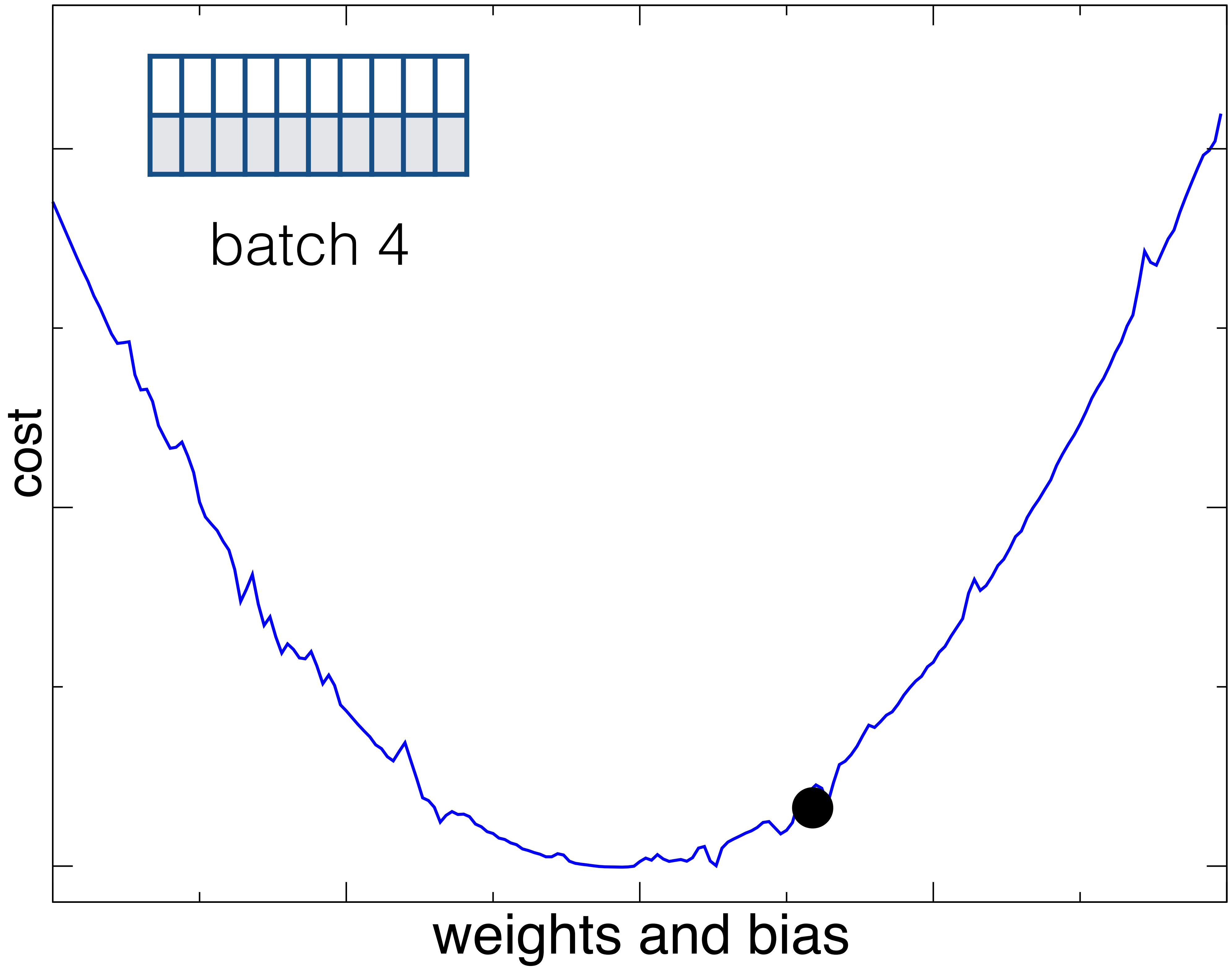

batch 1
cost
weights and bias


batch 2
cost
weights and bias

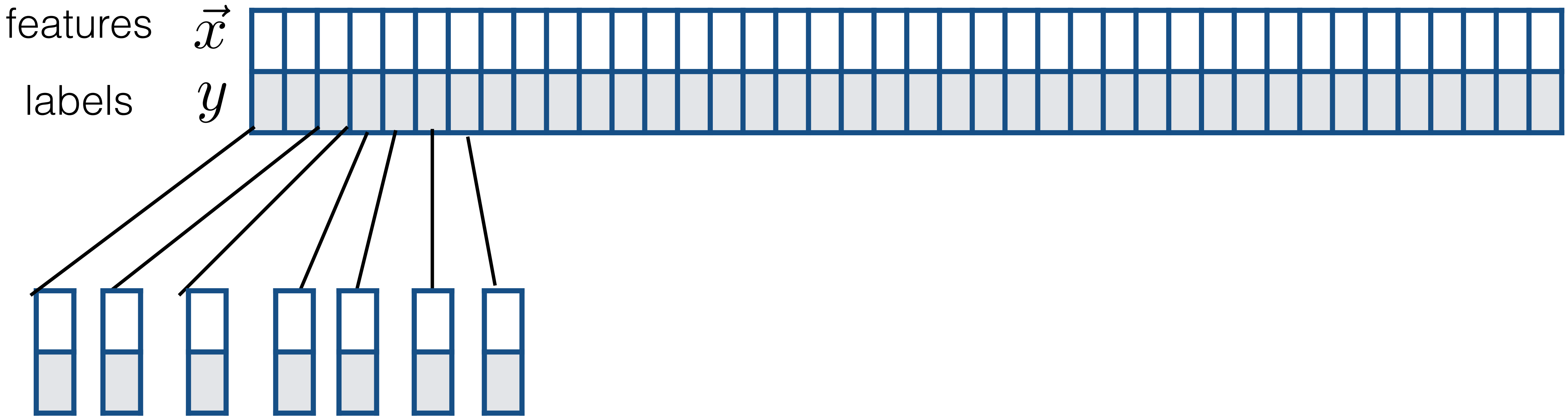

batch 2
cost
weights and bias


batch 3
cost
weights and bias


batch 3
cost
weights and bias


batch 4
cost
weights and bias

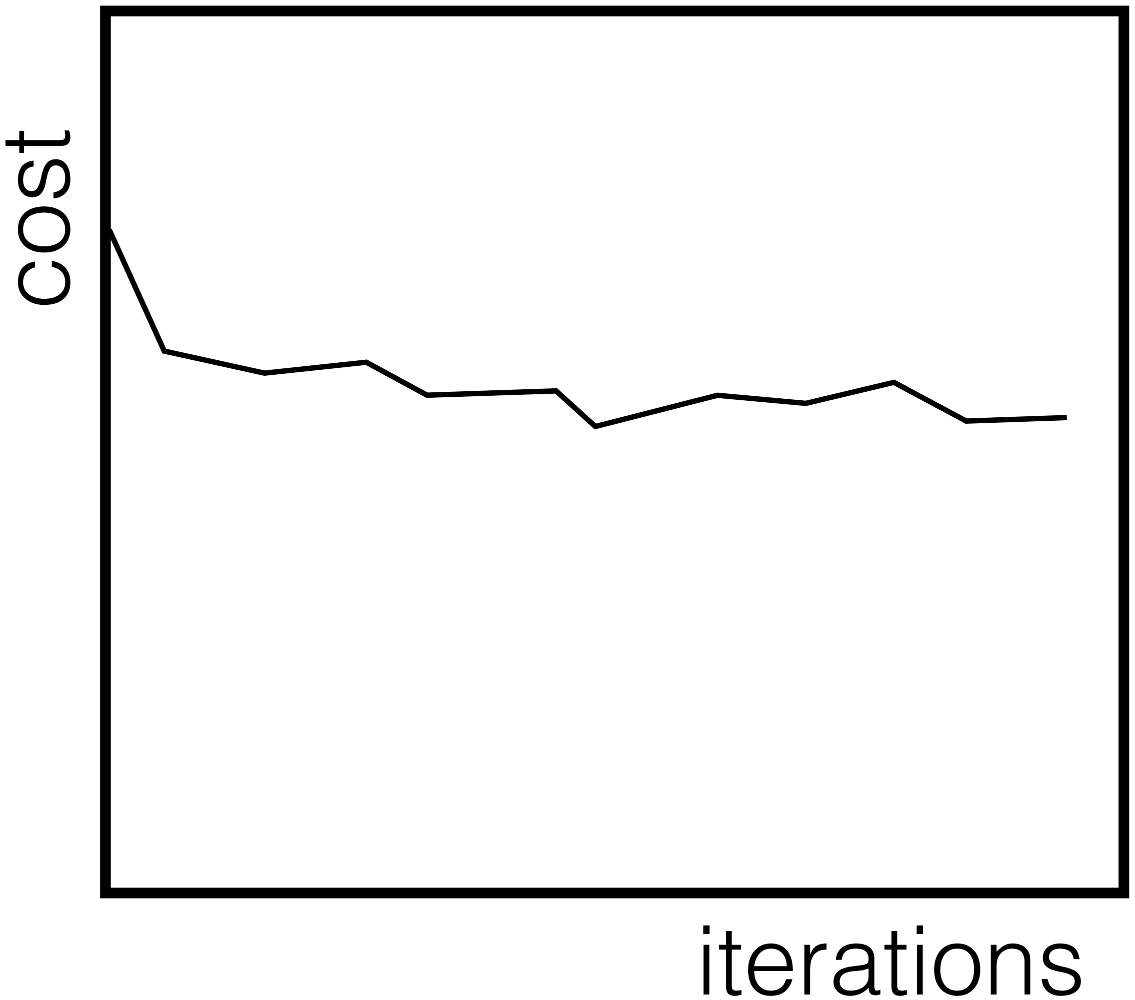
Stochastic gradient descend

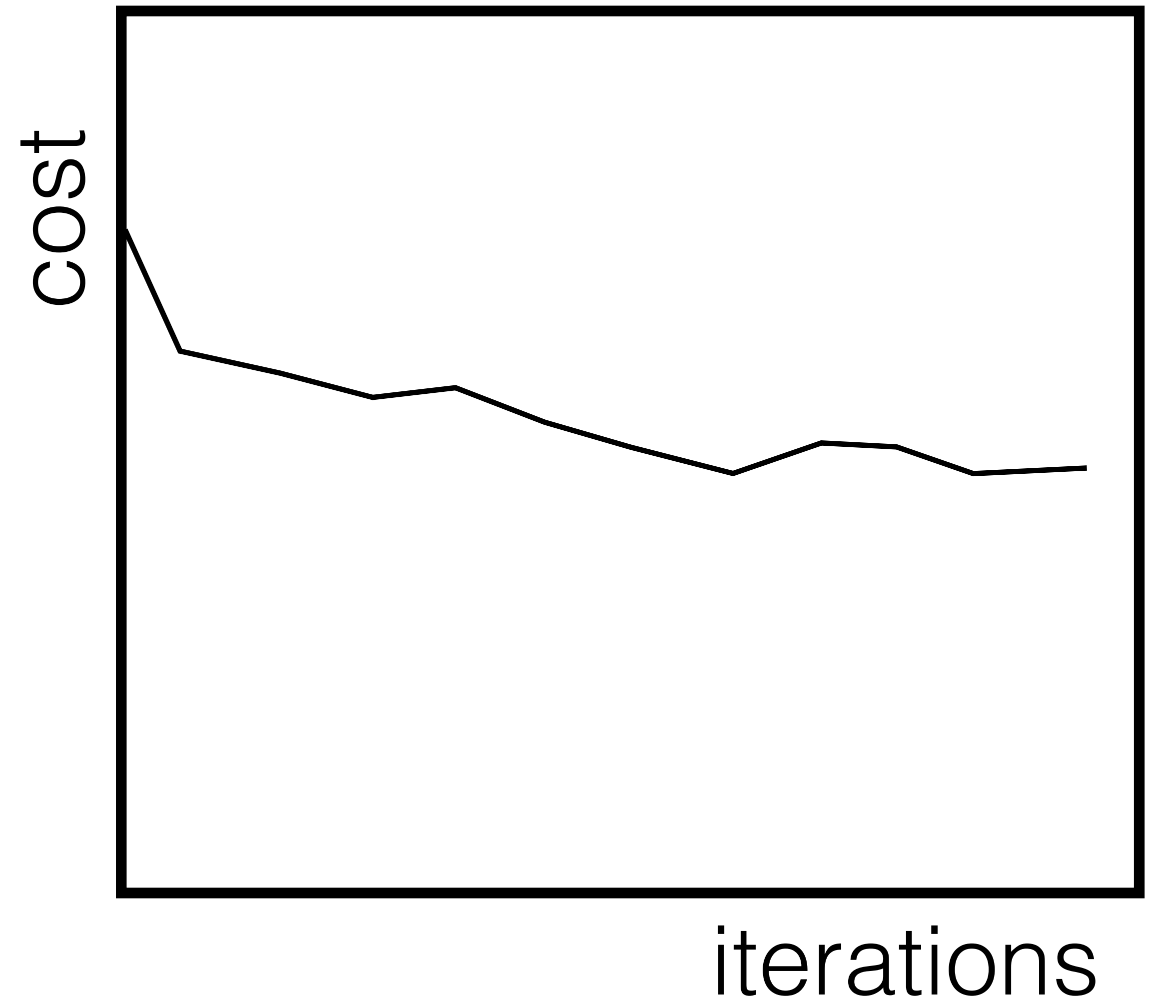


use batch size = 1 for stochastic gradient descend

Signs of trouble

always look at cost versus iterations plots



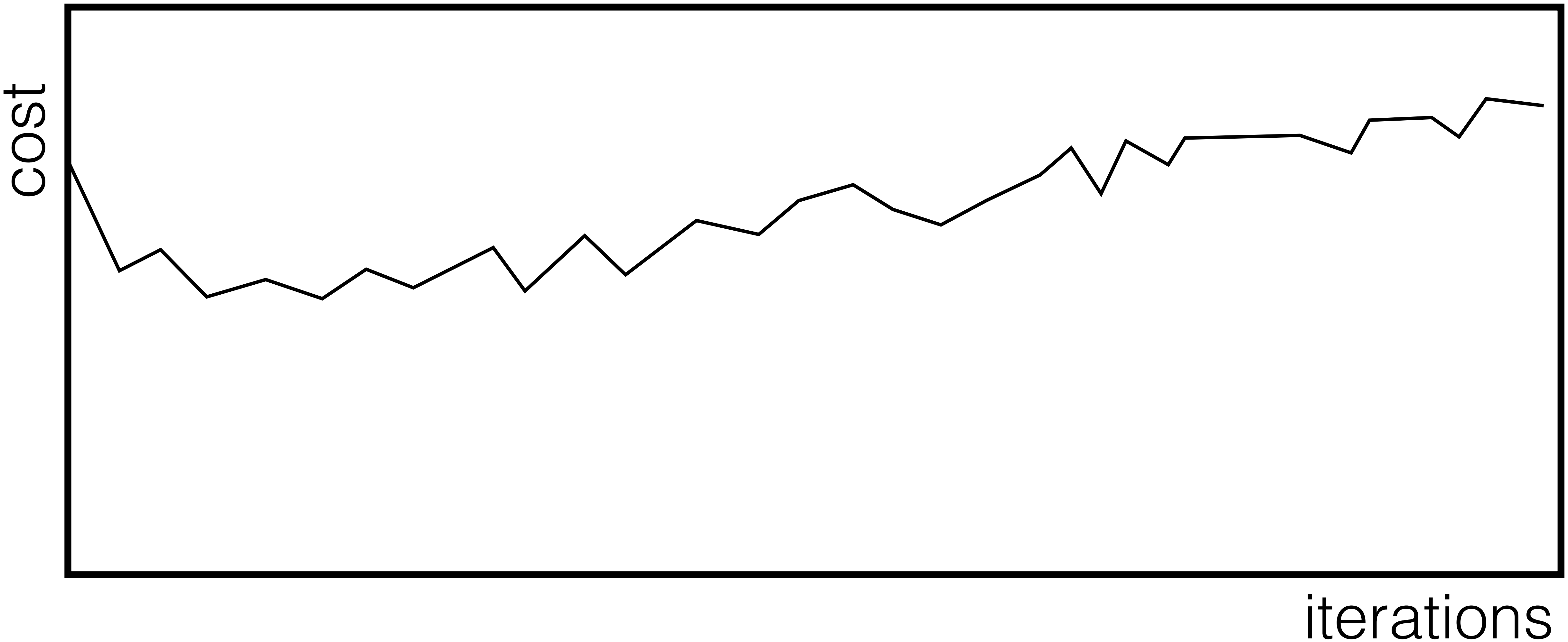


Cost not decreasing
looks like a local minimum

Cost decrease over slowly
looks like at very flat region
of cost surface

Signs of trouble

always look at cost versus iterations plots



Cost actually increasing

Please check for a **bug** in your code!!