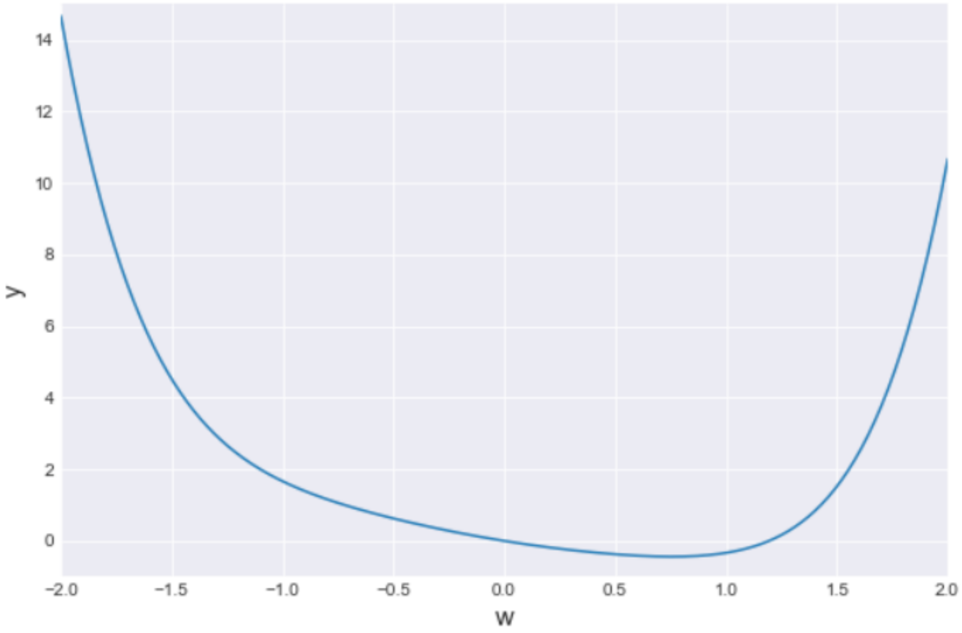
Derivative



$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Minima/
Maxima

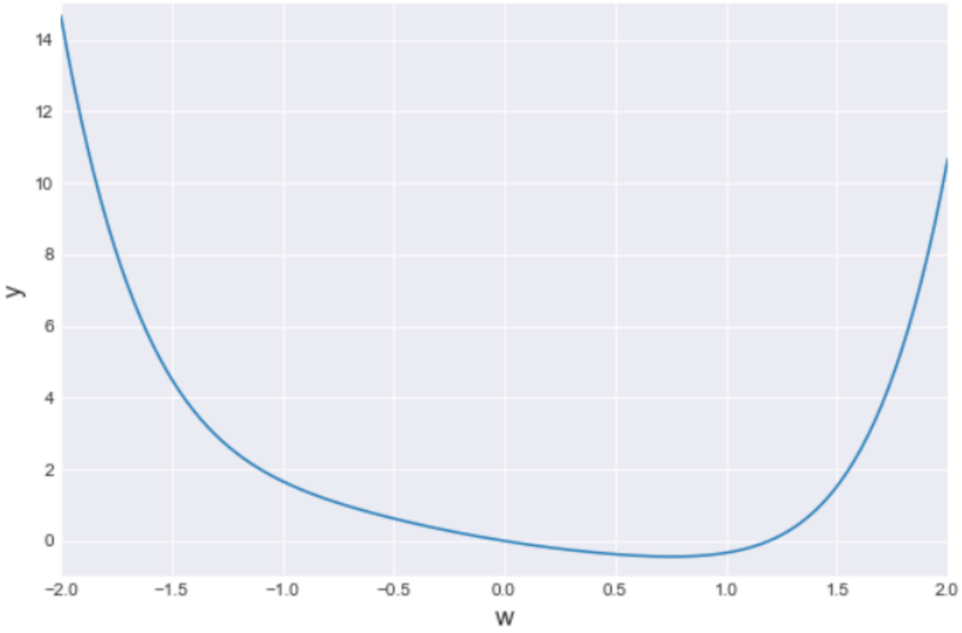
$$w^5 + w - 1 = 0$$

$$w^5 + w = 1$$

$$w(w^4 + 1) = 1$$

?

Derivative



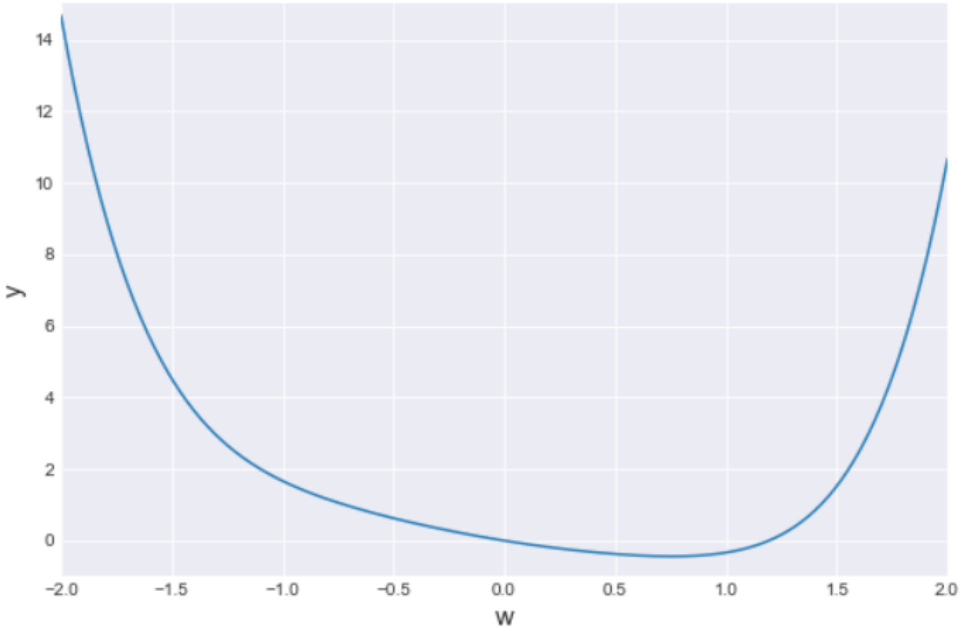
w = … :

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w$$

y(…) = ?

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Derivative

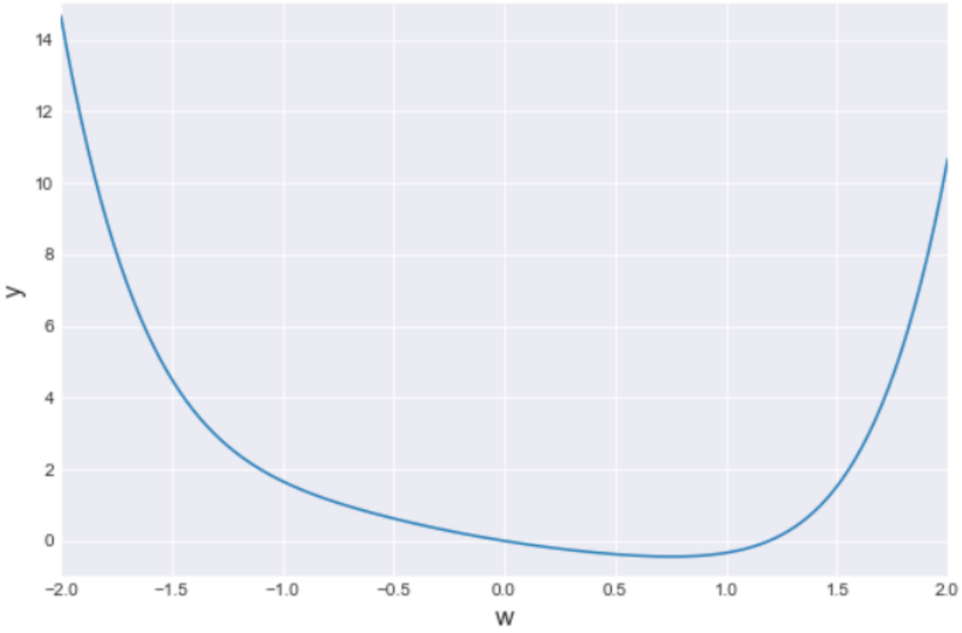


$w \in \{-2, -1.99, \ldots, 1.99, 2\}$:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Derivative



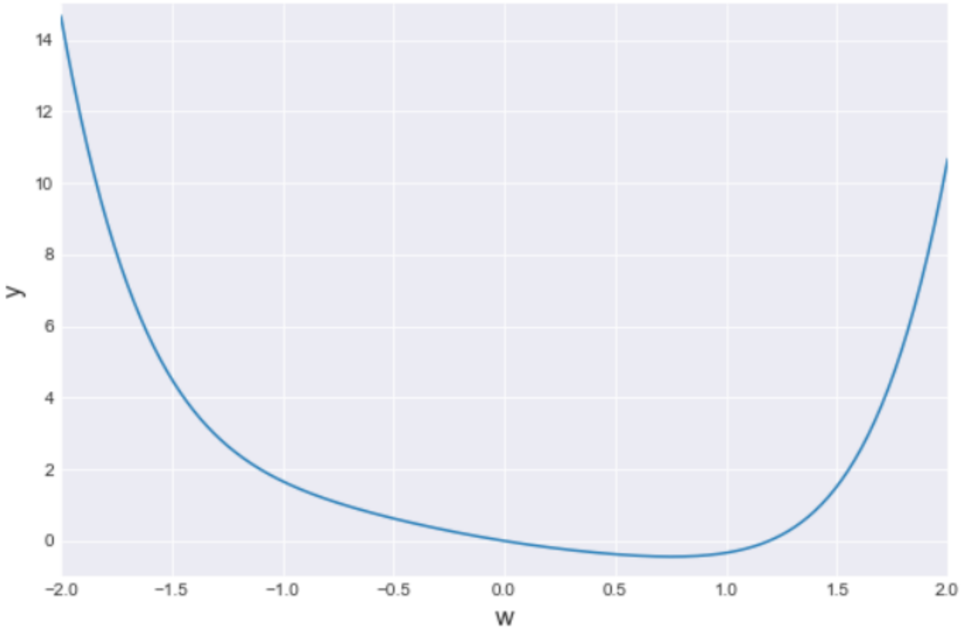
$w \in \{-2, -1.99, \ldots, 1.99, 2\}$:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w$$

$$y(0.75) = \underline{\underline{-0.439087}}$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Derivative

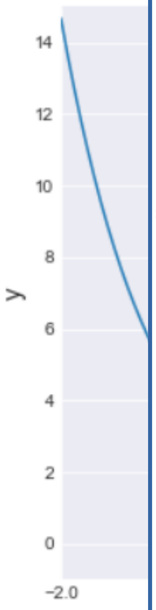


$w \in \{-2, -1.999, \ldots, 1.999, 2\}$:

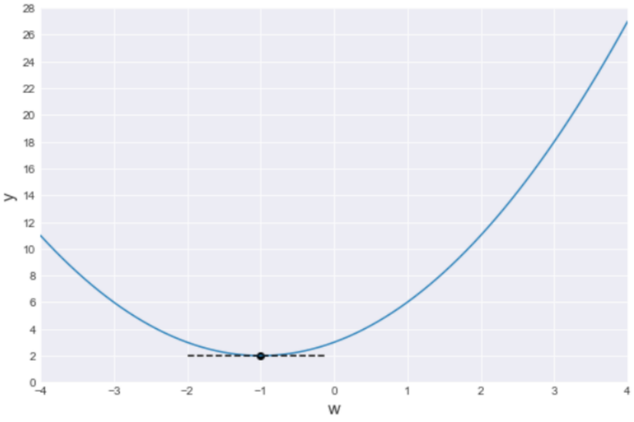
$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w \qquad y(0.755) = -0.439118$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Derivative



Derivative



$$y = f(w) = w^2 + 2w + 3$$

$$\frac{dy}{dw} = f'(w) = 2w + 2$$

Minima/ Maxima

$$2w + 2 = 0$$

$$2w = -2$$

$$\underline{\underline{w = -1}}$$

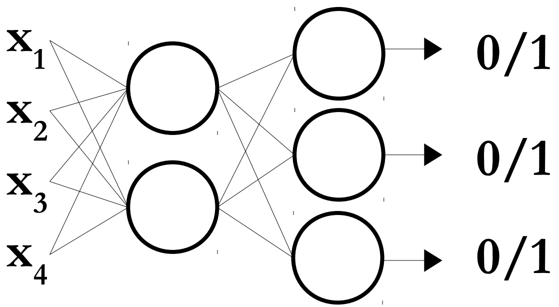
$w \in \{-2, -1.999, \ldots, 1.999, 2\}$:

– w $\quad y(0.755) = \underline{\underline{-\ 0.439118}}$

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

Cost function/Loss function:

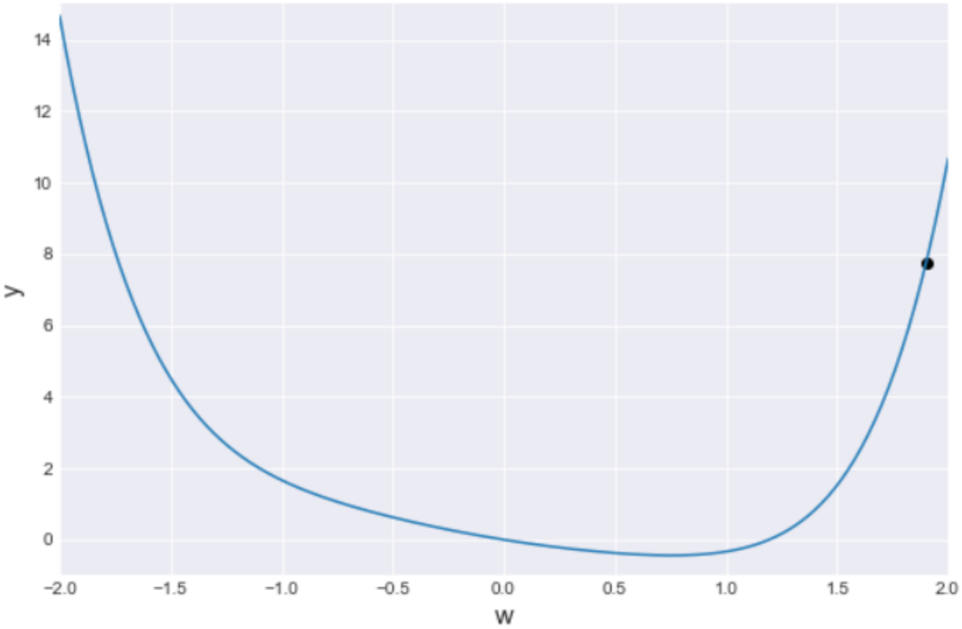
How do you determine the right parameters for the algorithm?

$$\text{Mean Squared Error} = \frac{1}{N}\sum_{e}\sum_{n}(O_{out\,e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N}\sum_{e}\sum_{n}(step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

N: e * n

Derivative



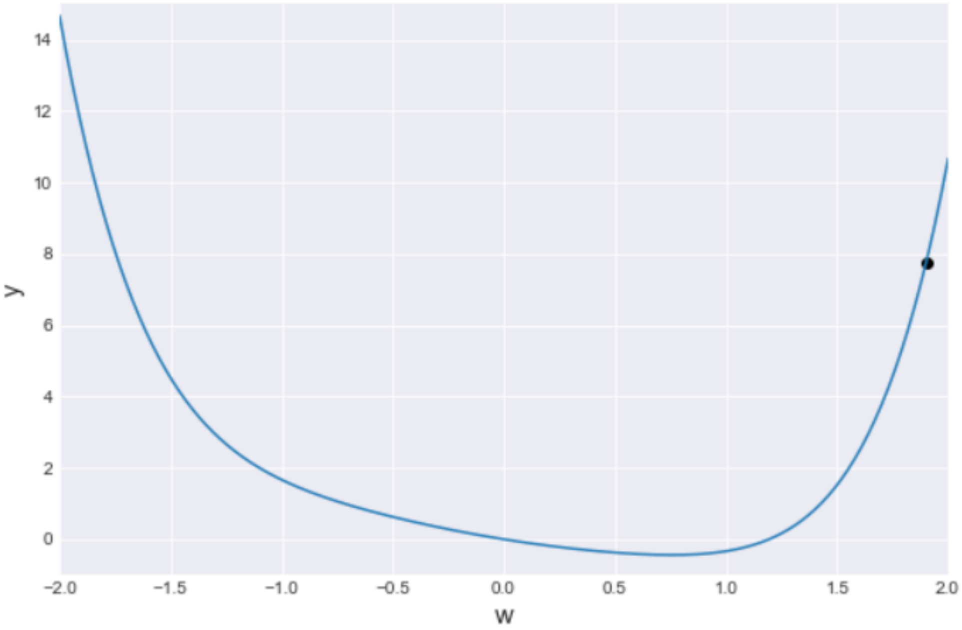
w = 1.9:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w$$

$$y(1.9) = 7.75$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

Derivative



w = 1.9:

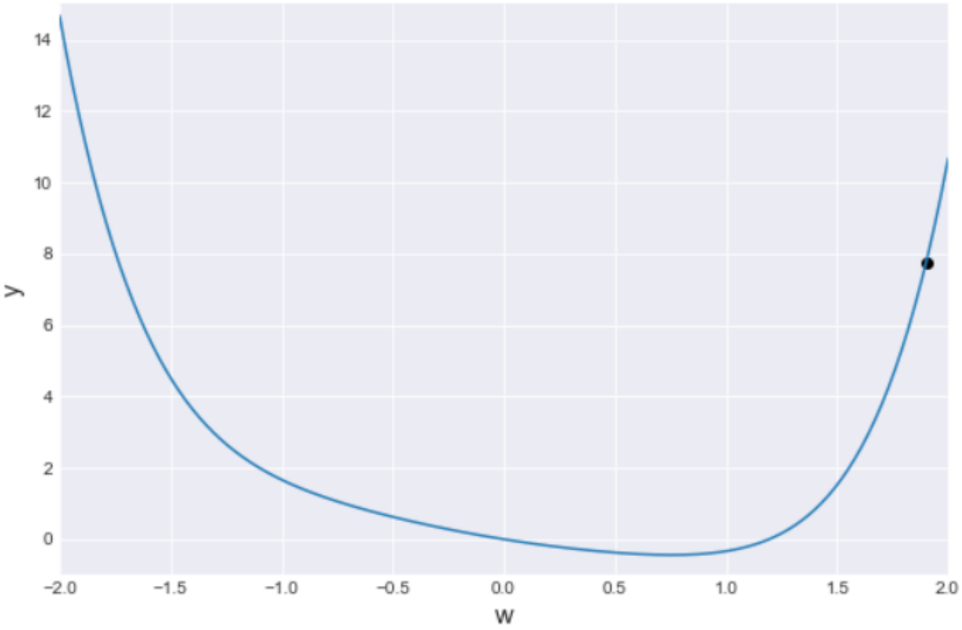
$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w$$

$$y(1.9) = 7.75$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

$$\frac{dy}{dw}(1.9) = 25.66$$

Derivative



w = 1.9:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w$$

$$y(1.9) = 7.75$$

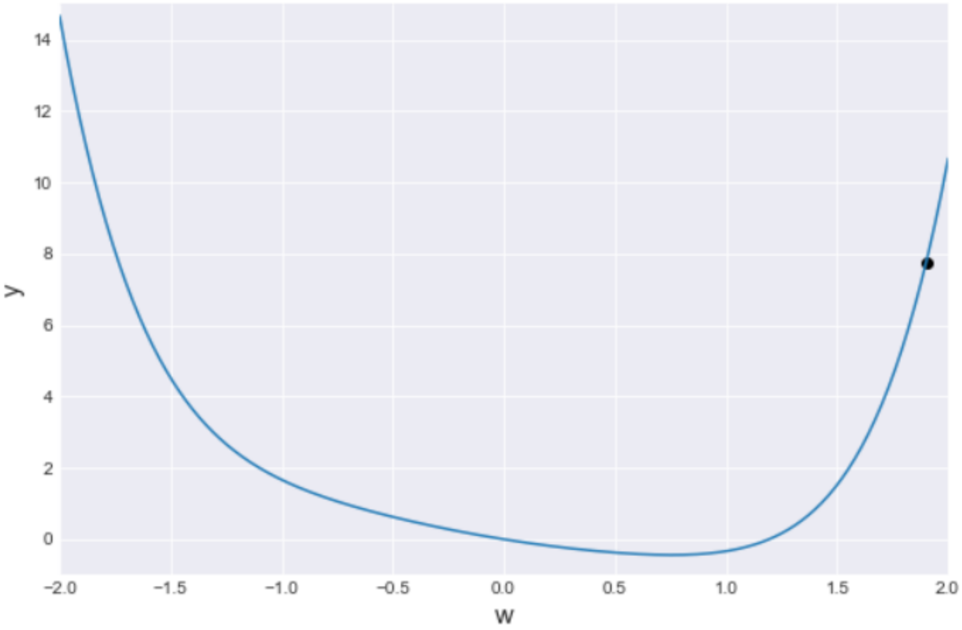
$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

$$\frac{dy}{dw}(1.9) = 25.66$$

Update rule:

$$w := w - \frac{dy}{dw}(w)$$

Derivative



w = 1.9:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w \qquad y(1.9) = 7.75$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(1.9) = 25.66$$

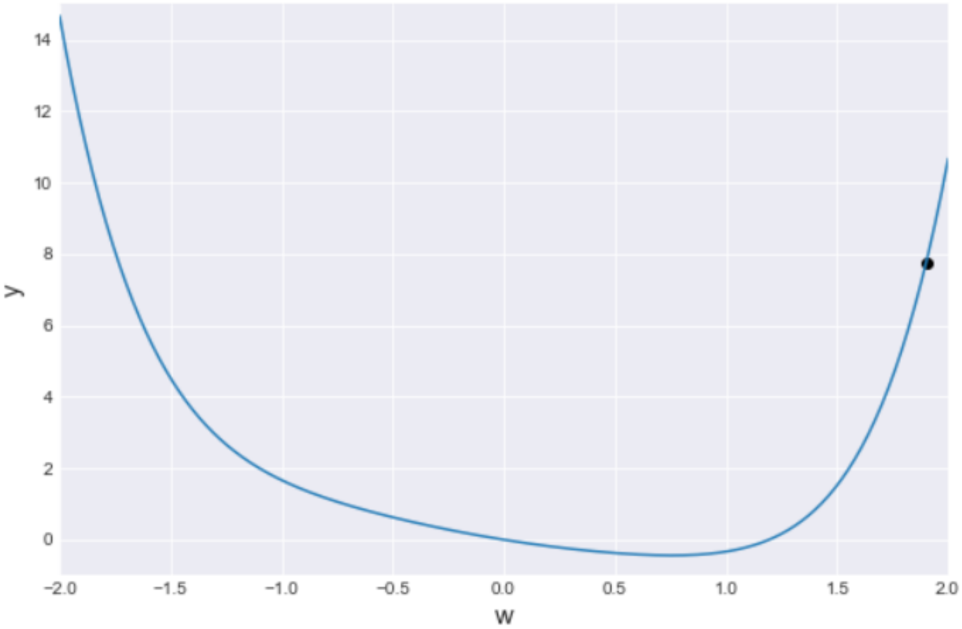
Update rule:

$$w := w - \frac{dy}{dw}(w)$$

$$w := 1.9 - 25.66$$

$$w := -23.76$$

Derivative



w = -23.76:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w \qquad y(-23.76) = 29{,}986{,}915$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(-23.76) = -7{,}572{,}400$$

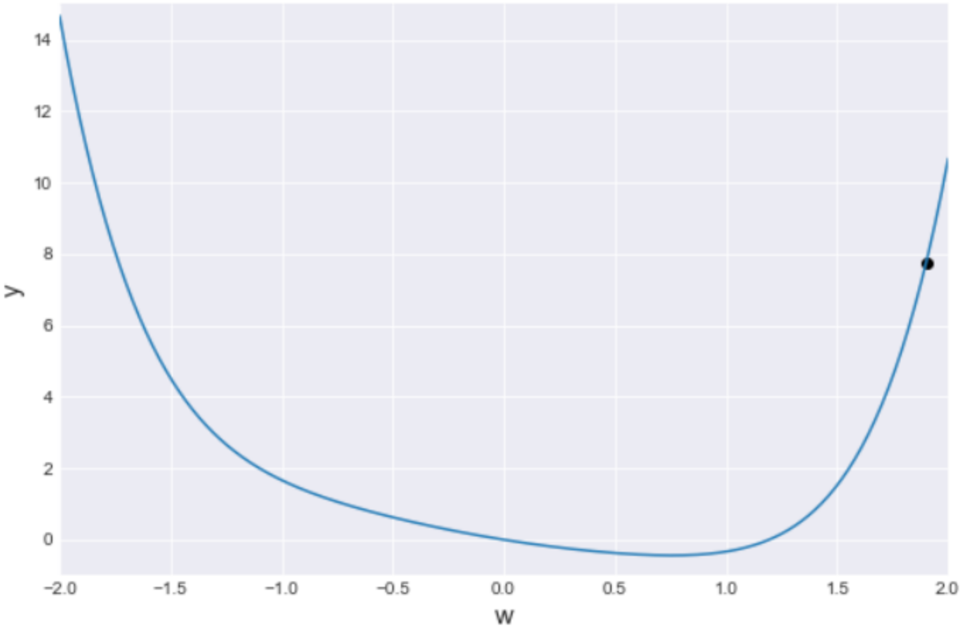
Update rule:

$$w := w - \frac{dy}{dw}(w)$$

$$w := 1.9 - 25.66$$

$$w := -23.76$$

Derivative



w = -23.76:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w \qquad y(-23.76) = 29{,}986{,}915$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(-23.76) = -7{,}572{,}400$$

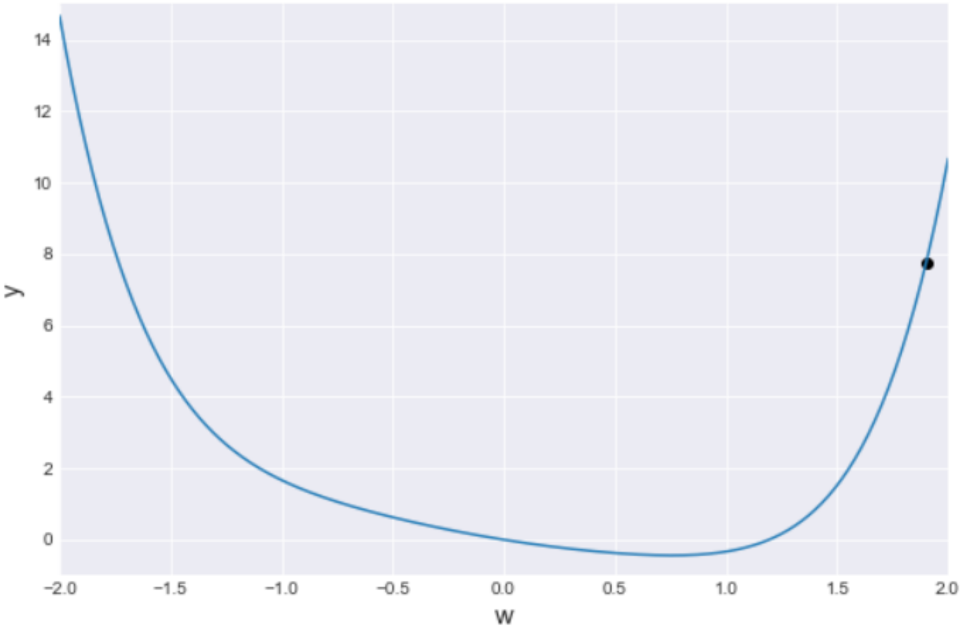
Update rule:

$$w := w - \frac{dy}{dw}(w)$$

$$w := -23.76 - (-7{,}572{,}400)$$

w := 7,572,376.24

Derivative



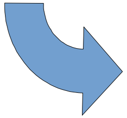
w = 1.9:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w$$

$$y(1.9) = 7.75$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

$$\frac{dy}{dw}(1.9) = 25.66$$

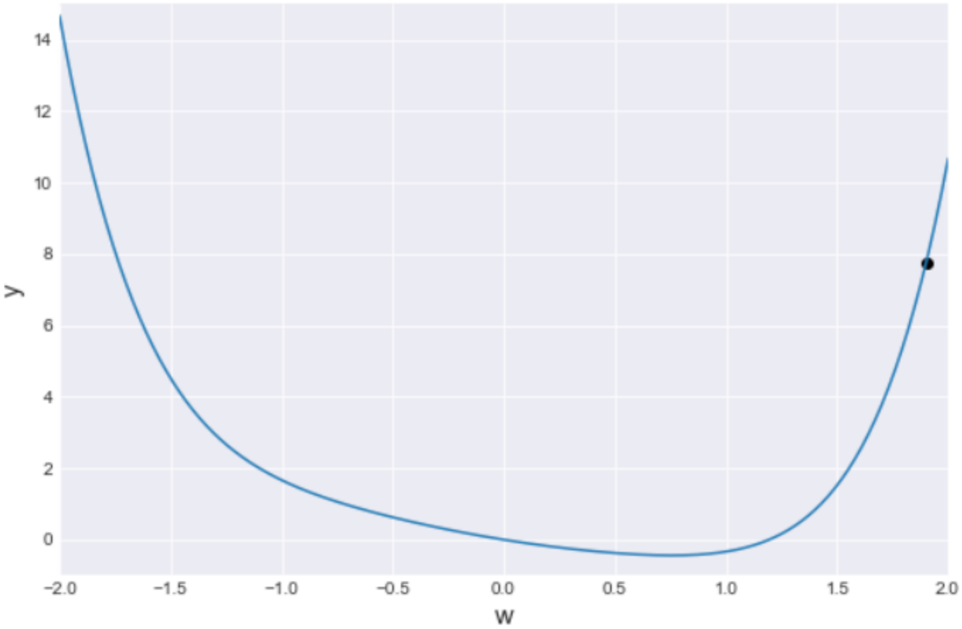


Update rule:

$$w := w - \alpha * \frac{dy}{dw}(w)$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



w = 1.9:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w \qquad y(1.9) = 7.75$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(1.9) = 25.66$$

Update rule:

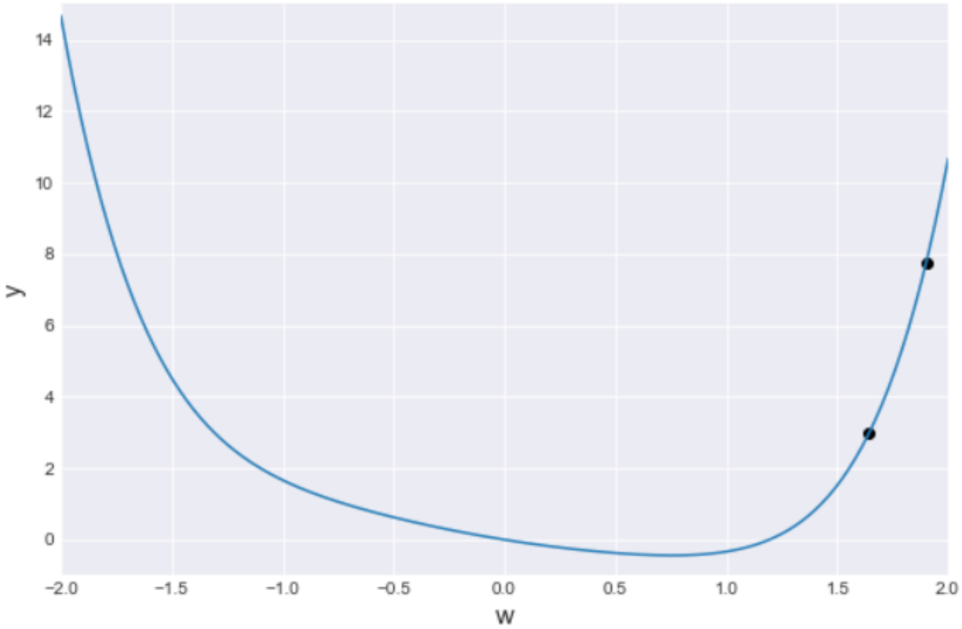
$$w := w - \alpha * \frac{dy}{dw}(w)$$

$$w := 1.9 - 0.01 * 25.66$$

$$\underline{w := 1.64}$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



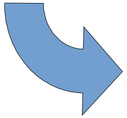
w = 1.64:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w$$

$$y(1.64) = 2.95$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

$$\frac{dy}{dw}(1.64) = 12.50$$

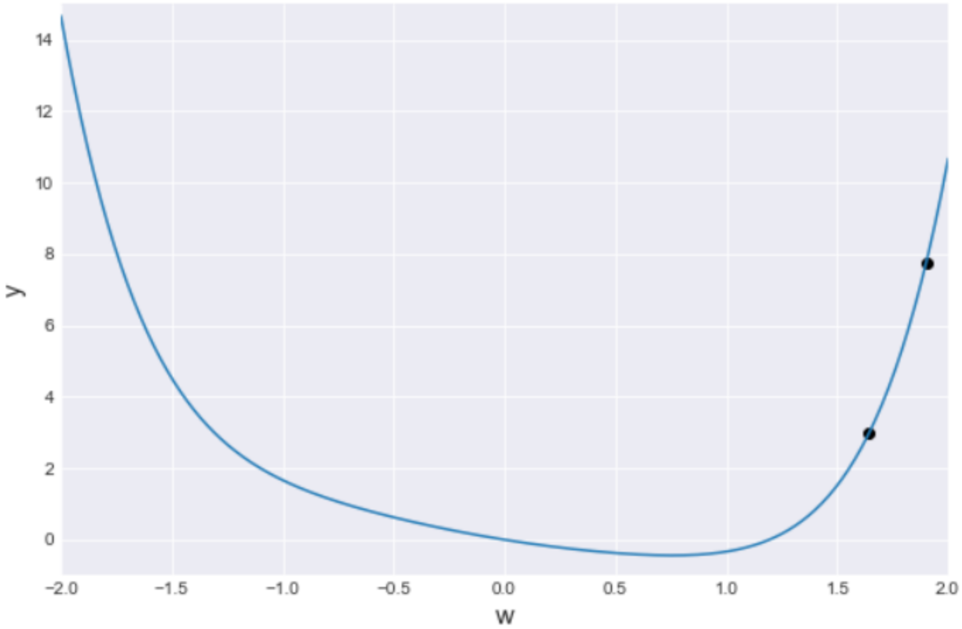


Update rule:

$$w := w - \alpha * \frac{dy}{dw}(w)$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



w = 1.64:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w \qquad y(1.64) = 2.95$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(1.64) = 12.50$$

Update rule:

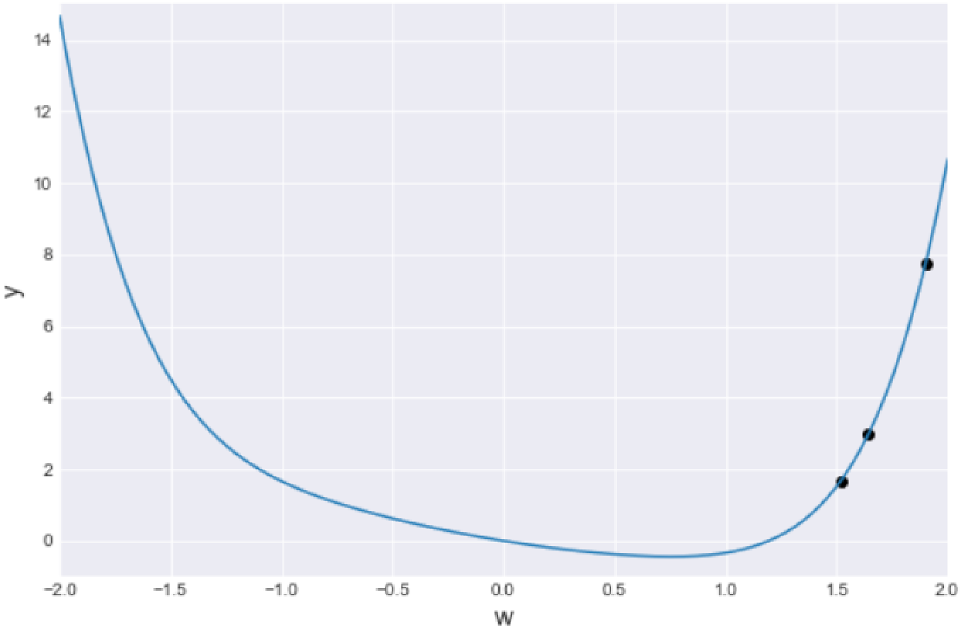
$$w := w - \alpha * \frac{dy}{dw}(w)$$

$$w := 1.64 - 0.01 * 12.50$$

$$w := 1.52$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



w = 1.52:

$$y = f(w) = \frac{1}{6} w^6 + \frac{1}{2} w^2 - w \qquad y(1.52) = 1.69$$

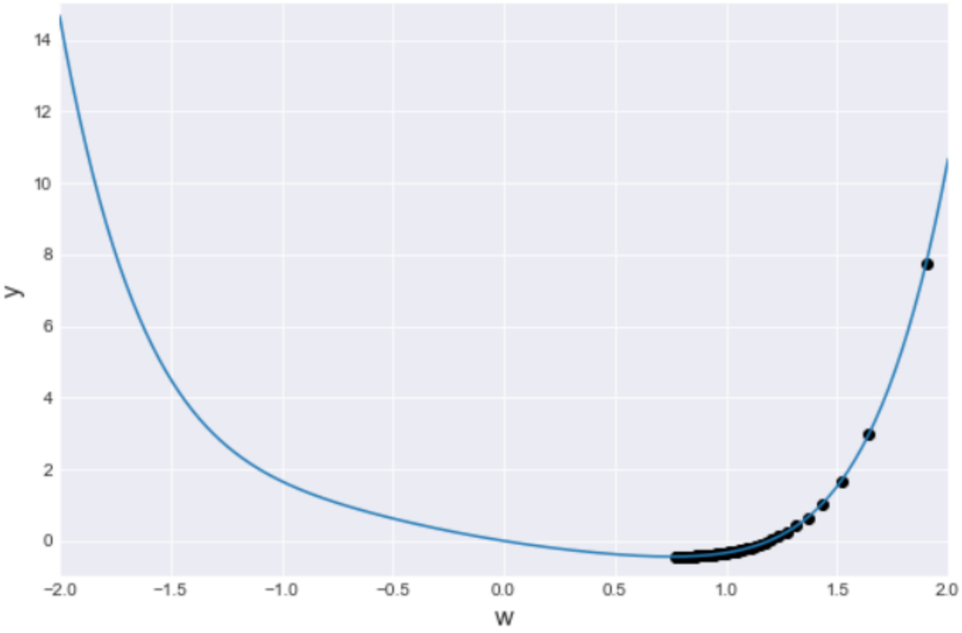
$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(1.52) = 8.63$$

Update rule:

$$w := w - \alpha * \frac{dy}{dw}(w)$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



w = 0.76:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w \qquad y(0.76) = -0.44$$

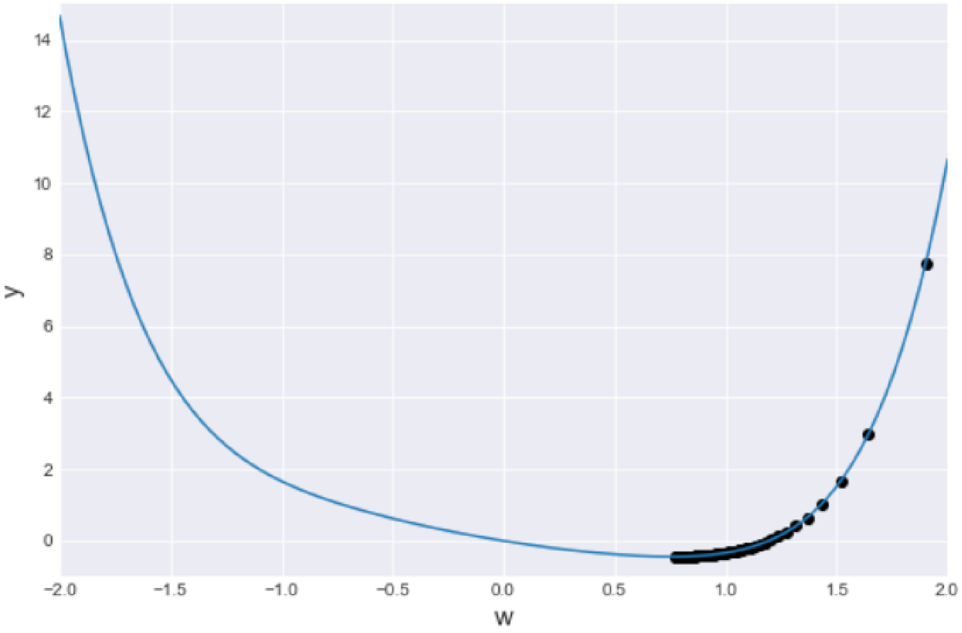
$$\frac{dy}{dw} = f'(w) = w^5 + w - 1 \qquad \frac{dy}{dw}(0.76) = 0.01$$

Update rule:

$$w := w - \alpha * \frac{dy}{dw}(w)$$

α – learning rate: 0.1, 0.01 or 0.001

Derivative



w = 0.76:

$$y = f(w) = \frac{1}{6}w^6 + \frac{1}{2}w^2 - w$$

$$y(0.76) = -0.44$$

$$\frac{dy}{dw} = f'(w) = w^5 + w - 1$$

$$\frac{dy}{dw}(0.76) = 0.01$$

Gradient Descent:

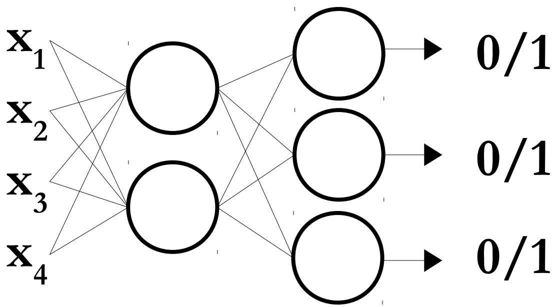
$$w := w - \alpha * \frac{dy}{dw}(w)$$

α – learning rate: 0.1, 0.01 or 0.001

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

Cost function/Loss function:

How do you determine the right parameters for the algorithm?

$$\text{Mean Squared Error} = \frac{1}{N}\sum_{e}\sum_{n}(O_{out\ e,n} - Y_{e,n})^2$$

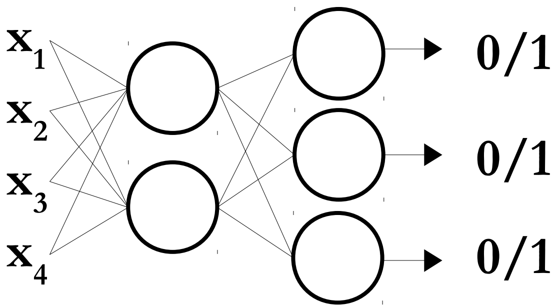
$$\text{MSE} = \frac{1}{N}\sum_{e}\sum_{n}(step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

N: e * n

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N}\sum_e\sum_n (O_{out\ e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N}\sum_e\sum_n (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

How do you determine the right parameters for the algorithm?

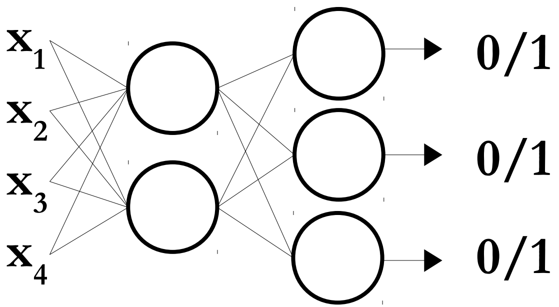
Gradient Descent:

$$w := w - \alpha \cdot \frac{d\ MSE}{d\ w}(w)$$

N: e * n

Deep Learning

Feedforward



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How does the algorithm make a decision?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N}\sum_{e}\sum_{n}(O_{out\ e,n} - Y_{e,n})^2$$

$$\textbf{MSE} = \frac{1}{N}\sum_{e}\sum_{n}(step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

$$w_i := w_i - \alpha \cdot \frac{d\ MSE}{d\ w_i}(w_i)$$

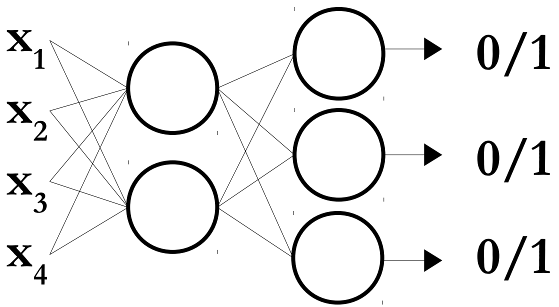
How do you determine the right parameters for the algorithm?

N: e * n

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How do you determine the right parameters for the algorithm?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N}\sum_e \sum_n (O_{out\,e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N}\sum_e \sum_n (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

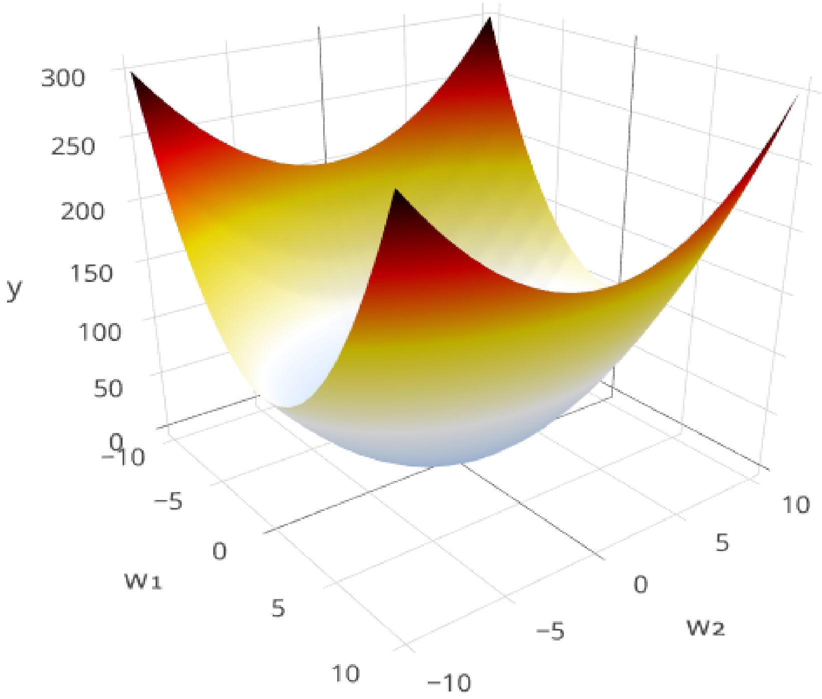
$$w_i := w_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, w_i}(w_i)$$

N: e * n

Partial Derivatives

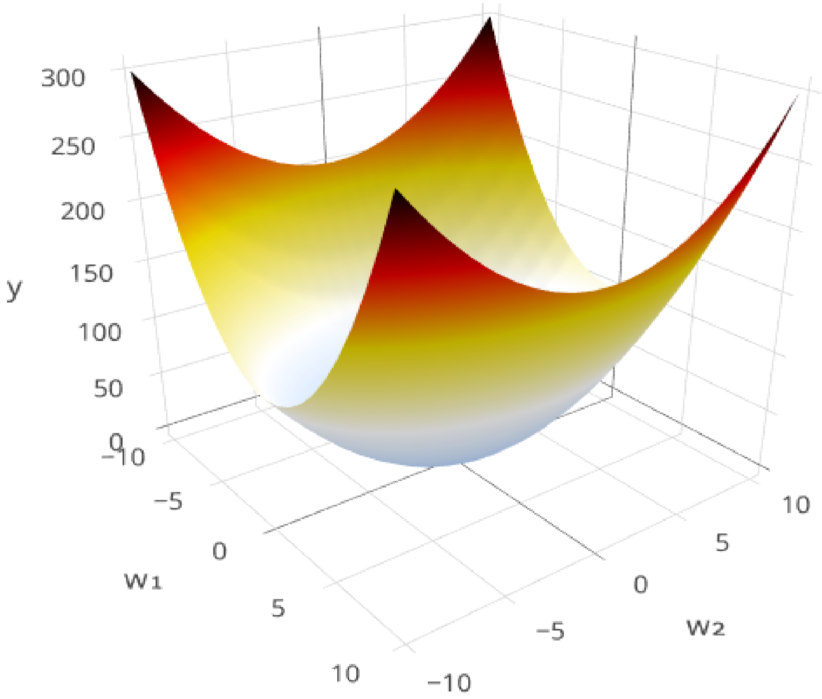
$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

Partial Derivatives



$y = f(w_1, w_2) = 2w_1^2 + w_2^2$

Partial Derivatives

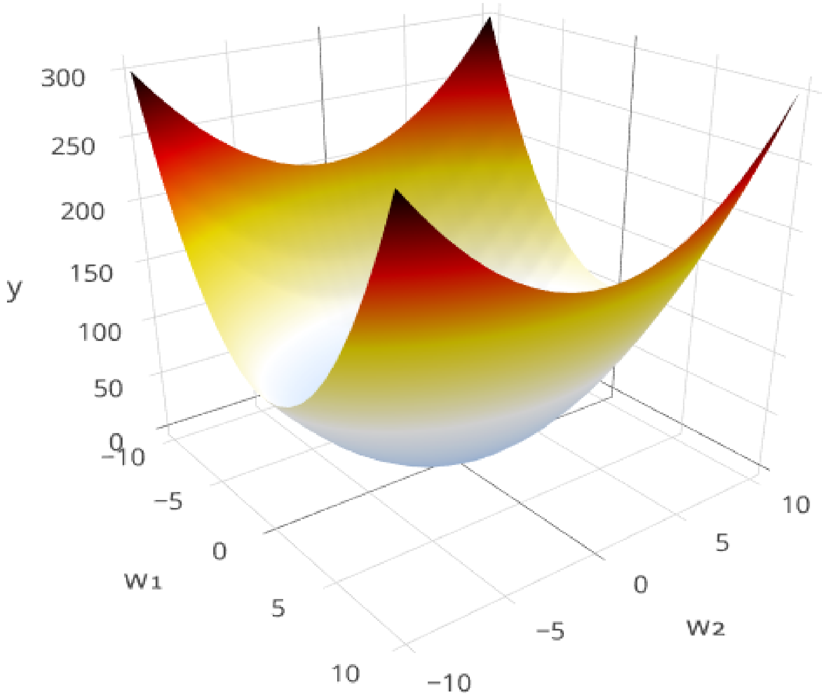


$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

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

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

Partial Derivatives

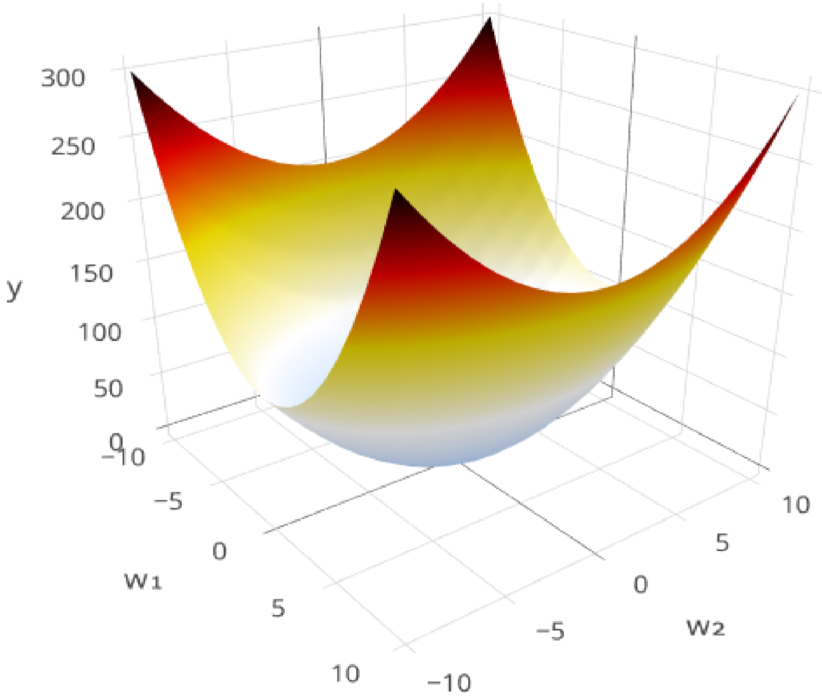


$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

$$\frac{\partial y}{\partial w_1} = 4w_1$$

$$\frac{\partial y}{\partial w_2} = 2w_2$$

Partial Derivatives



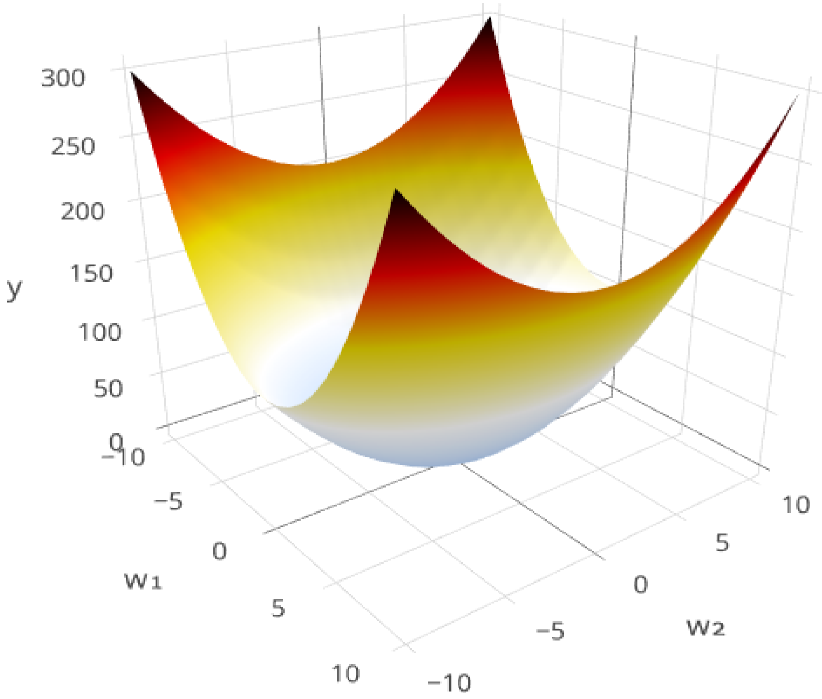
$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

$$\frac{\partial y}{\partial w_1} = 4w_1$$

$$\frac{\partial y}{\partial w_2} = 2w_2$$

<u>$w_1 = -8; w_2 = 7$:</u>

Partial Derivatives



$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

$$\frac{\partial y}{\partial w_1} = 4w_1$$

$$\frac{\partial y}{\partial w_2} = 2w_2$$

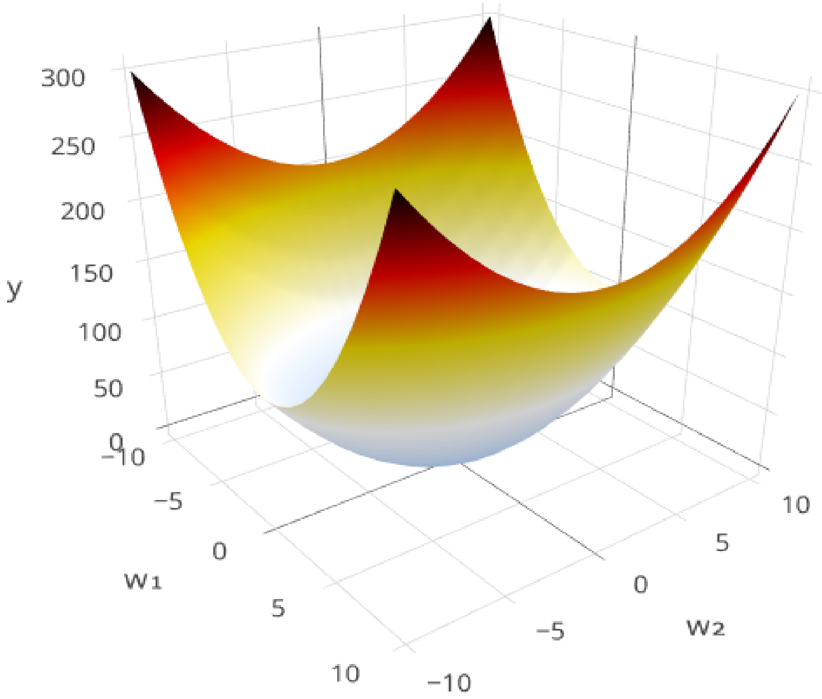
<u>$w_1 = -8; w_2 = 7$:</u>

$$y(-8, 7) = 177$$

$$\frac{\partial y}{\partial w_1}(-8) = -32$$

$$\frac{\partial y}{\partial w_2}(7) = 14$$

Partial Derivatives



$w_1 = -8; w_2 = 7:$

$$y = f(w_1, w_2) = 2w_1^2 + w_2^2 \qquad y(-8, 7) = 177$$

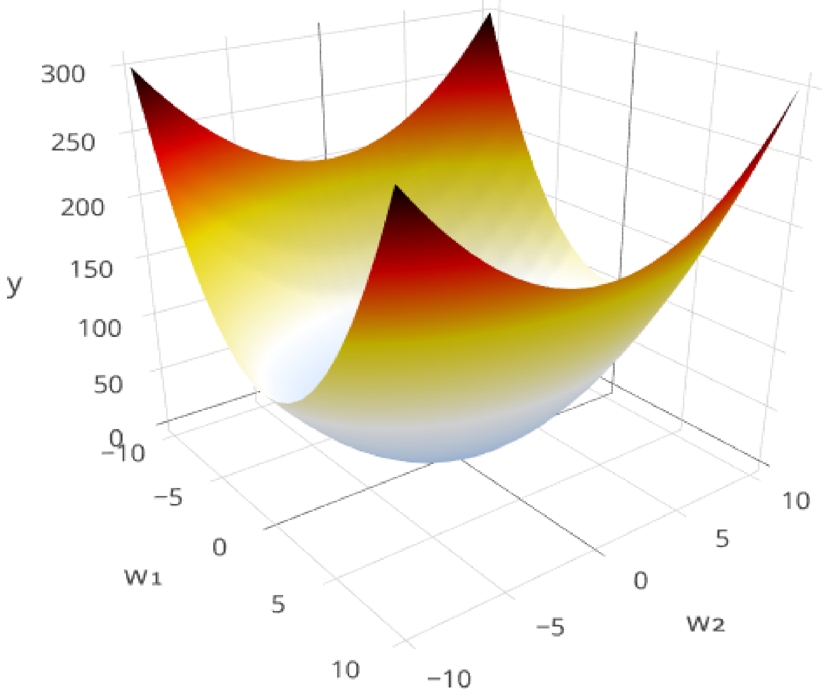
$$\frac{\partial y}{\partial w_1} = 4w_1 \qquad \frac{\partial y}{\partial w_1}(-8) = -32$$

$$\frac{\partial y}{\partial w_2} = 2w_2 \qquad \frac{\partial y}{\partial w_2}(7) = 14$$

Gradient Descent:

$$w_1 := w_1 - \alpha * \frac{\partial y}{\partial w_1}(w_1) \qquad w_2 := w_2 - \alpha * \frac{\partial y}{\partial w_2}(w_2)$$

Partial Derivatives



$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

$$\frac{\partial y}{\partial w_1} = 4w_1$$

$$\frac{\partial y}{\partial w_2} = 2w_2$$

Minima/
Maxima

$$4w_1 = 0 \qquad 2w_2 = 0$$

$$\underline{\underline{w_1 = 0}} \qquad \underline{\underline{w_2 = 0}}$$

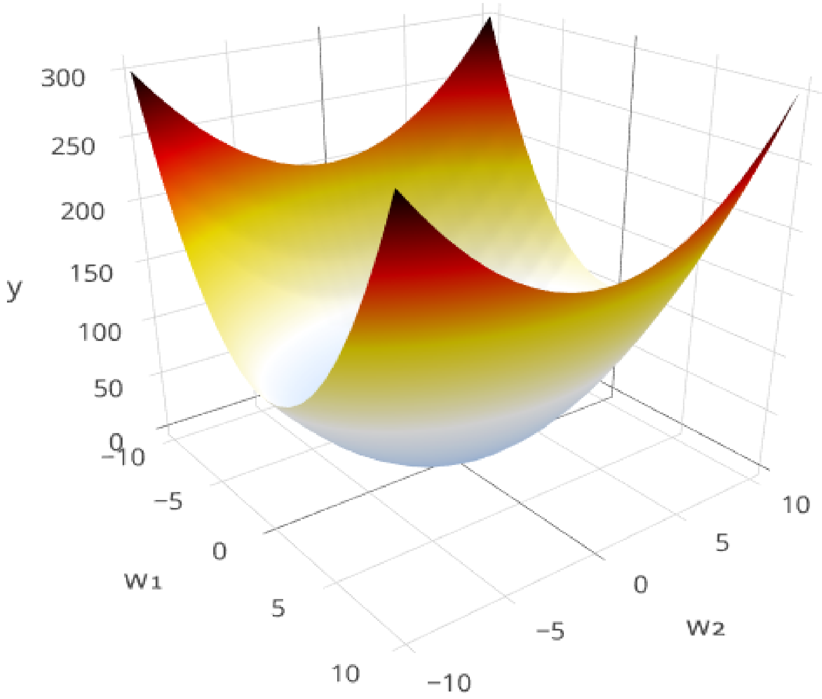
$\underline{w_1 = -8;\ w_2 = 7:}$

$$y(-8, 7) = 177$$

$$\frac{\partial y}{\partial w_1}(-8) = -32$$

$$\frac{\partial y}{\partial w_2}(7) = 14$$

Partial Derivatives



$$y = f(w_1, w_2) = 2w_1^2 + w_2^2$$

$$\frac{\partial y}{\partial w_1} = 4w_1$$

$$\frac{\partial y}{\partial w_2} = 2w_2$$

$w_1 = -8; w_2 = 7$:

$$y(-8, 7) = 177$$

$$\frac{\partial y}{\partial w_1}(-8) = -32$$

$$\frac{\partial y}{\partial w_2}(7) = 14$$

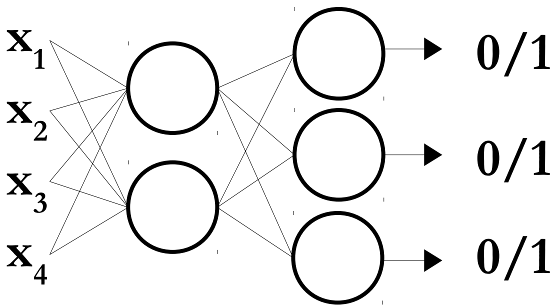
Gradient Descent:

$$w_1 := w_1 - \alpha * \frac{\partial y}{\partial w_1}(w_1) \qquad w_2 := w_2 - \alpha * \frac{\partial y}{\partial w_2}(w_2)$$

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

Cost function/Loss function:

How do you determine the right parameters for the algorithm?

$$\text{Mean Squared Error} = \frac{1}{N}\sum_e\sum_n (O_{out\ e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N}\sum_e\sum_n (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

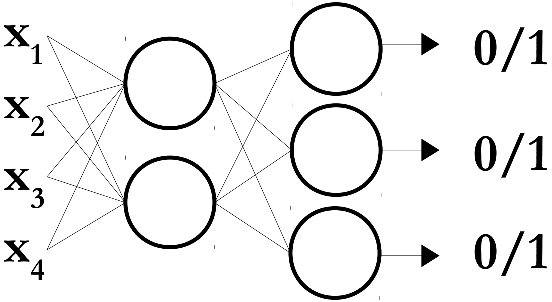
$$w_i := w_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, w_i}(w_i)$$

N: e * n

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How do you determine the right parameters for the algorithm?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N} \sum_{e} \sum_{n} (O_{out\ e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N} \sum_{e} \sum_{n} (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

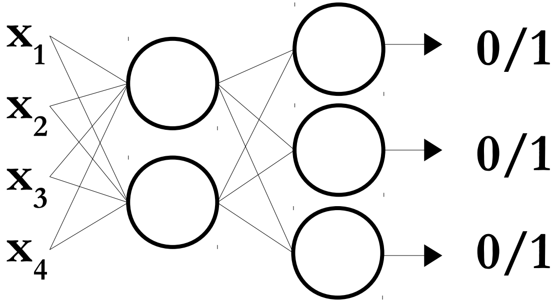
$$W_i := W_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, W_i}(W_i)$$

N: e * n

Deep Learning

How does the algorithm make a decision?

Feedforward



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How do you determine the right parameters for the algorithm?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N} \sum_{e} \sum_{n} (O_{out\ e,n} - Y_{e,n})^2$$

$$\textbf{MSE} = \frac{1}{N} \sum_{e} \sum_{n} (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

$$W_i := W_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, W_i}(W_i)$$

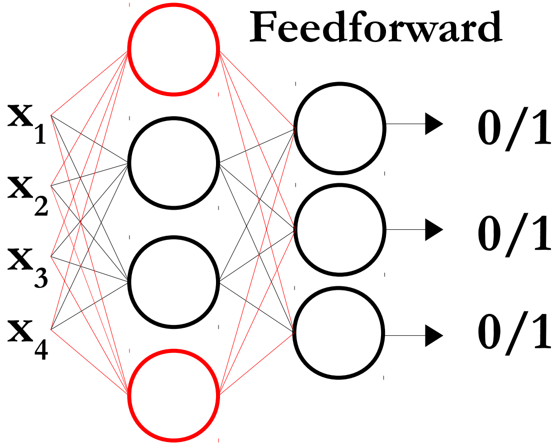
$$\frac{\partial\, MSE}{\partial\, W_1}(W_1) =$$

$$\frac{\partial\, MSE}{\partial\, W_2}(W_2) =$$

N: e * n

Deep Learning

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How do you determine the right parameters for the algorithm?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N}\sum_e\sum_n (O_{out\ e,n} - Y_{e,n})^2$$

$$\mathbf{MSE} = \frac{1}{N}\sum_e\sum_n (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

$$W_i := W_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, W_i}(W_i)$$

$$\frac{\partial\, MSE}{\partial\, W_1}(W_1) =$$

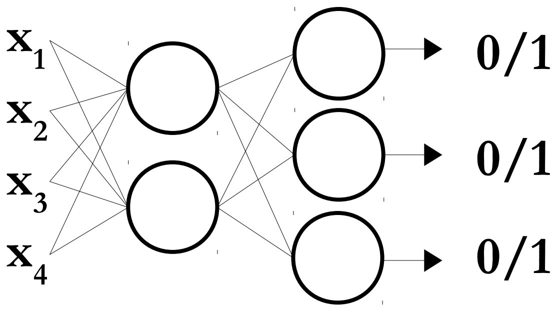
$$\frac{\partial\, MSE}{\partial\, W_2}(W_2) =$$

N: e * n

Deep Learning

Feedforward

How does the algorithm make a decision?



$$H_{in} = XW_1$$

$$H_{out} = step(H_{in})$$

$$O_{in} = H_{out}W_2$$

$$O_{out} = step(O_{in})$$

How do you determine the right parameters for the algorithm?

Cost function/Loss function:

$$\text{Mean Squared Error} = \frac{1}{N}\sum_e\sum_n (O_{out\ e,n} - Y_{e,n})^2$$

$$\text{MSE} = \frac{1}{N}\sum_e\sum_n (step(step(XW_1)W_2)_{e,n} - Y_{e,n})^2$$

Gradient Descent:

$$W_i := W_i - \alpha \cdot \frac{\partial\, MSE}{\partial\, W_i}(W_i)$$

$$\frac{\partial\, MSE}{\partial\, W_1}(W_1) =$$

$$\frac{\partial\, MSE}{\partial\, W_2}(W_2) =$$

N: e * n