

Глубинное обучение

Функции активации, Инициализация весов,.

Михаил Лазарев

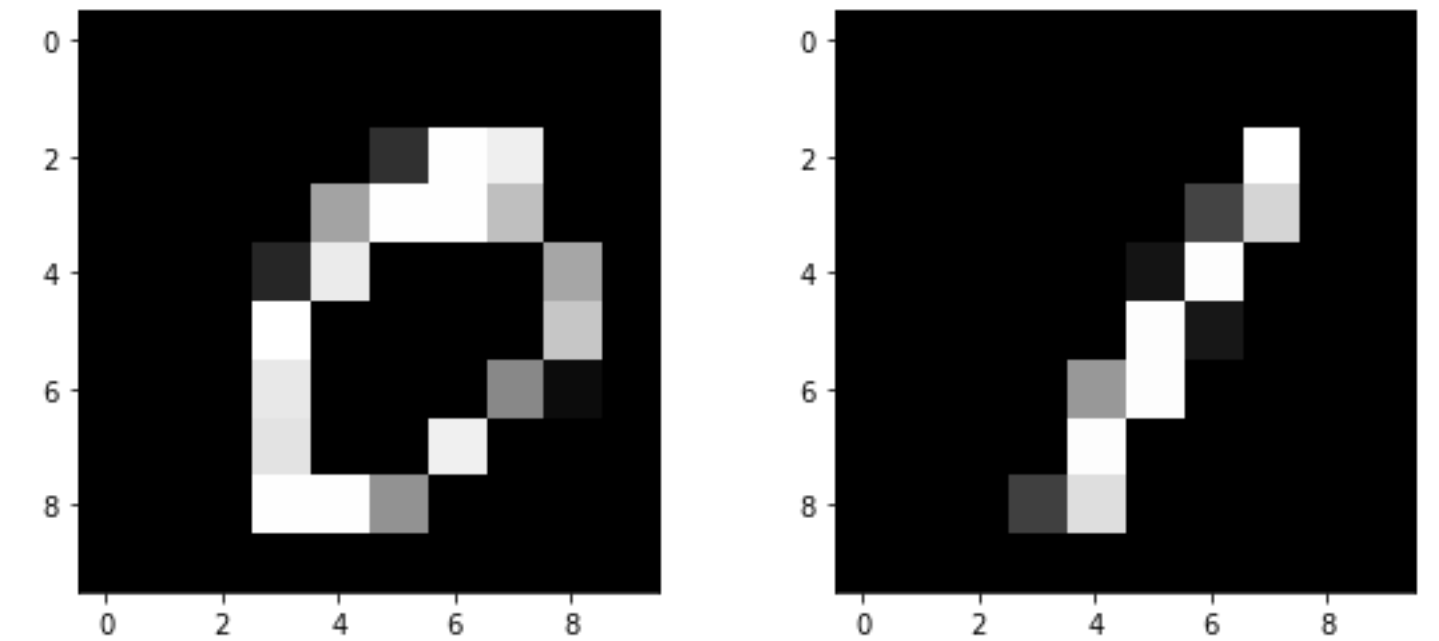
Recap

Бинарная классификация

Объекты x_1, \dots, x_n Ответы $y_1, \dots, y_n \in [0,1]$

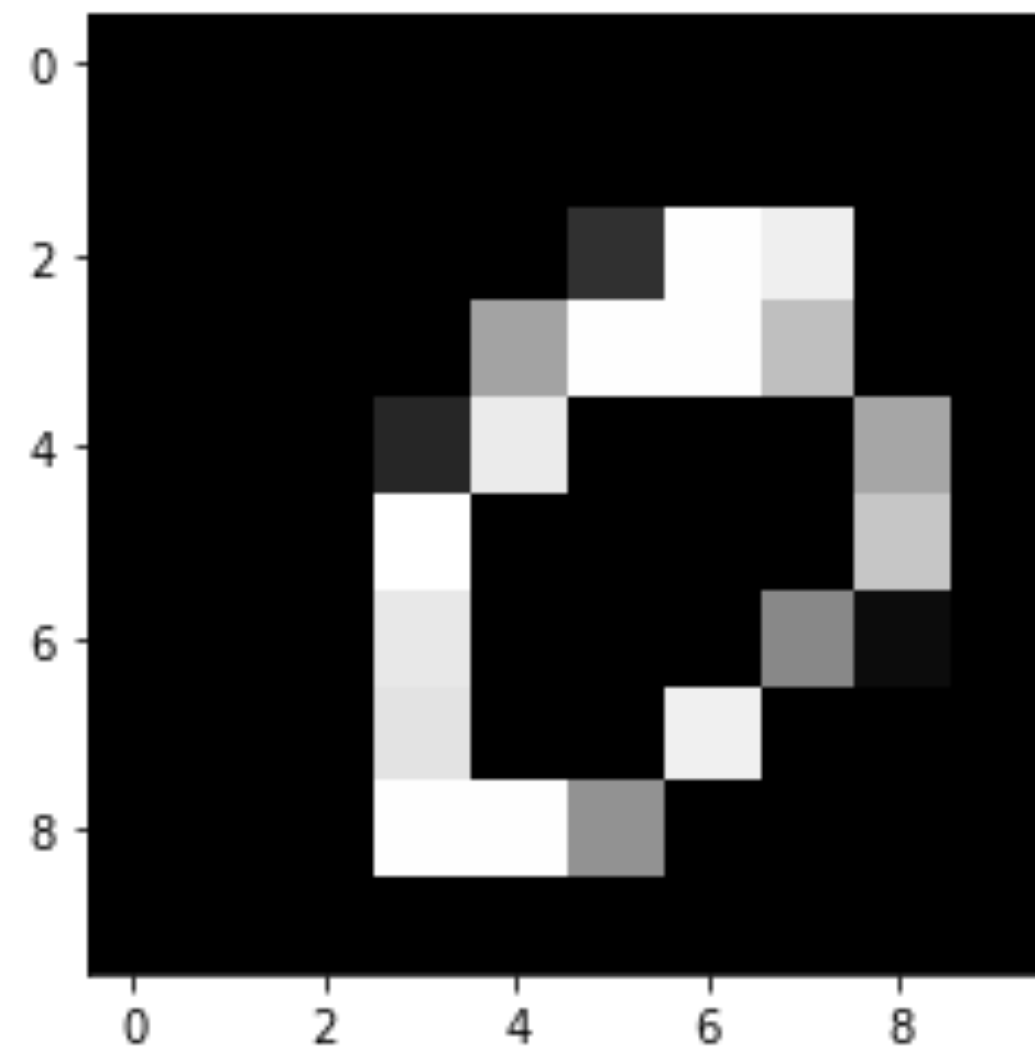
Алгоритм предсказания $f(x, \theta) = P(y = 1 | x, \theta) > \frac{1}{2}$

нейросеть из двух блоков вида Linear \rightarrow Sigmoid



Бинарная классификация

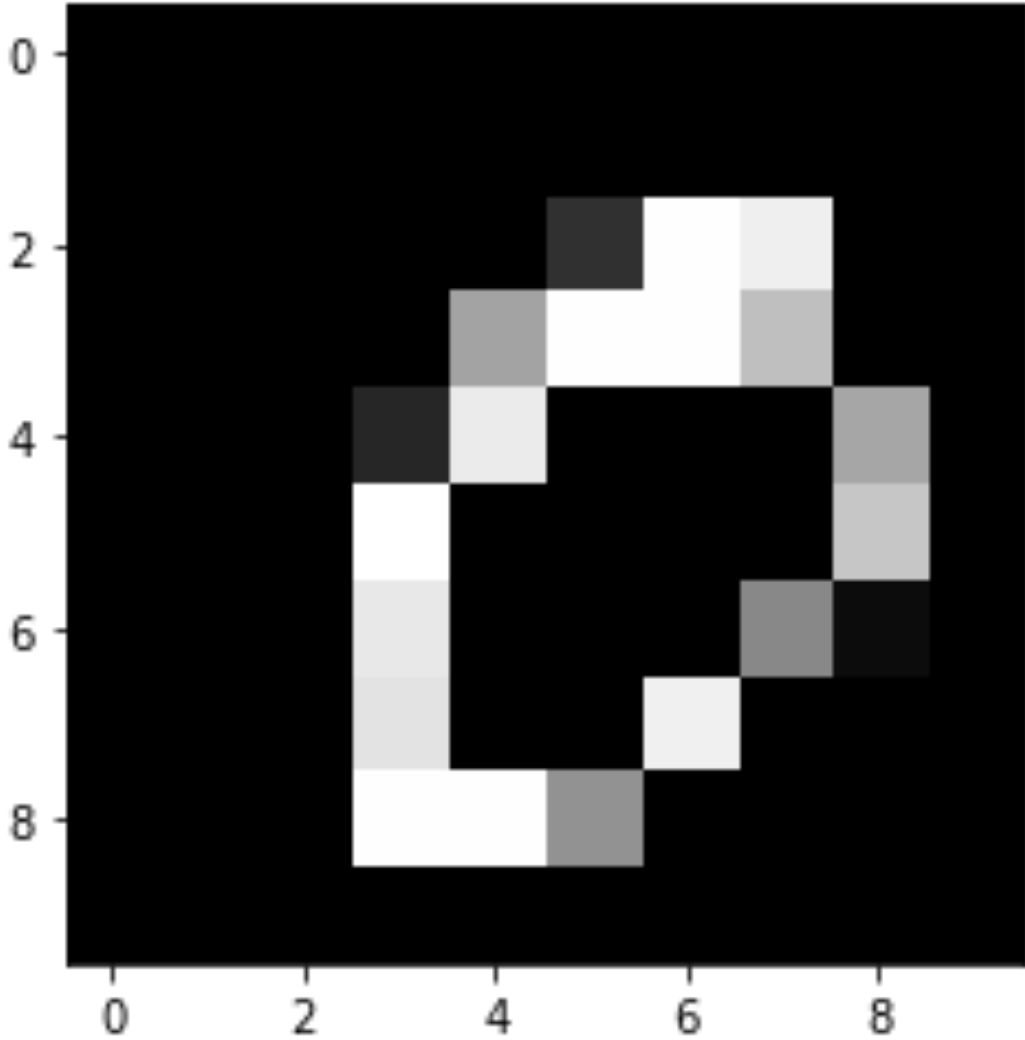
Input



\mathcal{X} - ч/б картинка

Бинарная классификация

Input



X - ч/б картинка

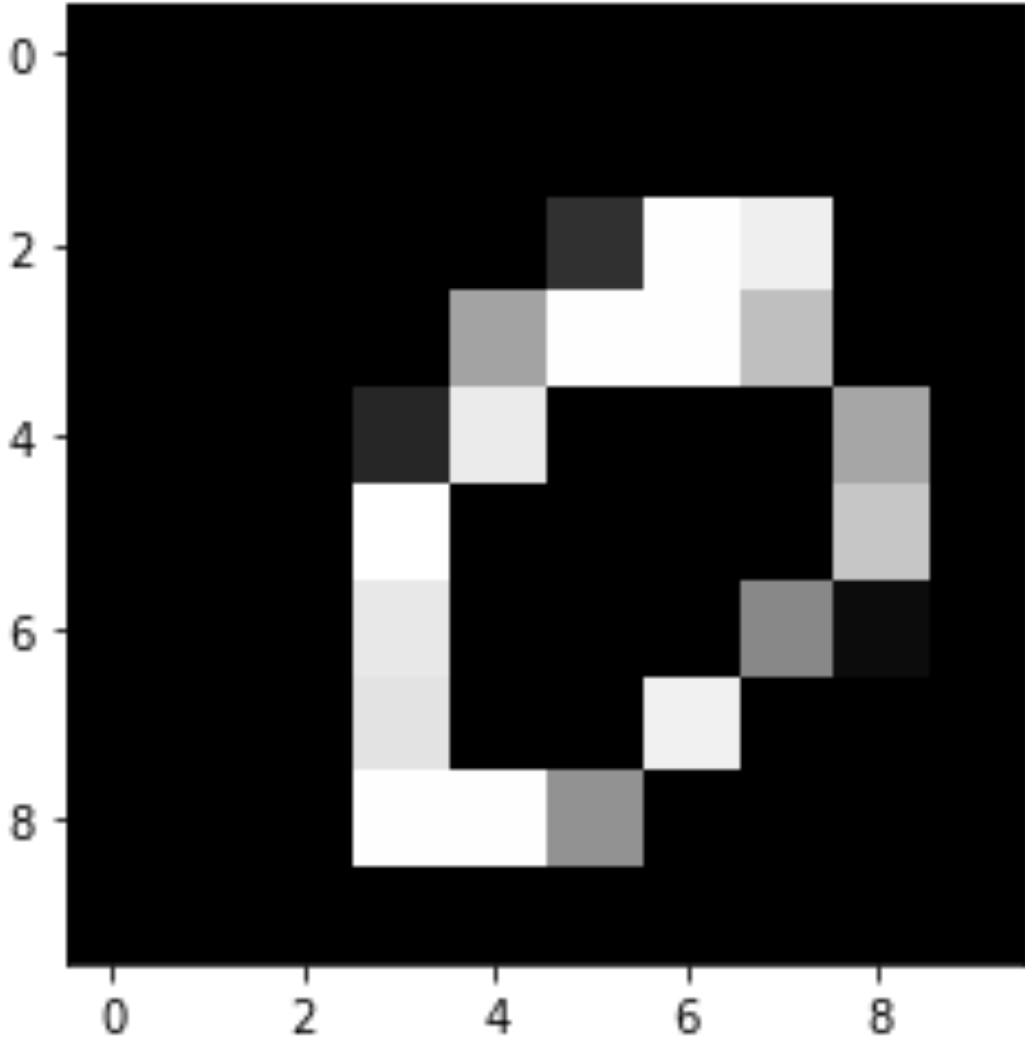


```
[[0.      , 0.      , 0.      , 0.      , 0.      ,  
 0.      , 0.      , 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.      , 0.      ,  
 0.      , 0.      , 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.      , 0.      ,  
 0.1875   , 0.984375 , 0.92578125, 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.      , 0.63671875 ,  
 0.984375 , 0.984375 , 0.73828125, 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.1484375 , 0.91015625 ,  
 0.      , 0.      , 0.      , 0.64453125, 0.      ],  
 [0.      , 0.      , 0.      , 0.98828125, 0.      ,  
 0.      , 0.      , 0.      , 0.765625  , 0.      ],  
 [0.      , 0.      , 0.      , 0.8984375 , 0.      ,  
 0.      , 0.      , 0.52734375, 0.046875 , 0.      ],  
 [0.      , 0.      , 0.      , 0.87890625, 0.      ,  
 0.      , 0.9296875 , 0.      , 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.984375  , 0.984375 ,  
 0.56640625, 0.      , 0.      , 0.      , 0.      ],  
 [0.      , 0.      , 0.      , 0.      , 0.      ,  
 0.      , 0.      , 0.      , 0.      , 0.      ]],
```

массив с элементами от 0 до 1
0 - черный, 1 - белый

Бинарная классификация

Input



X - ч/б картинка



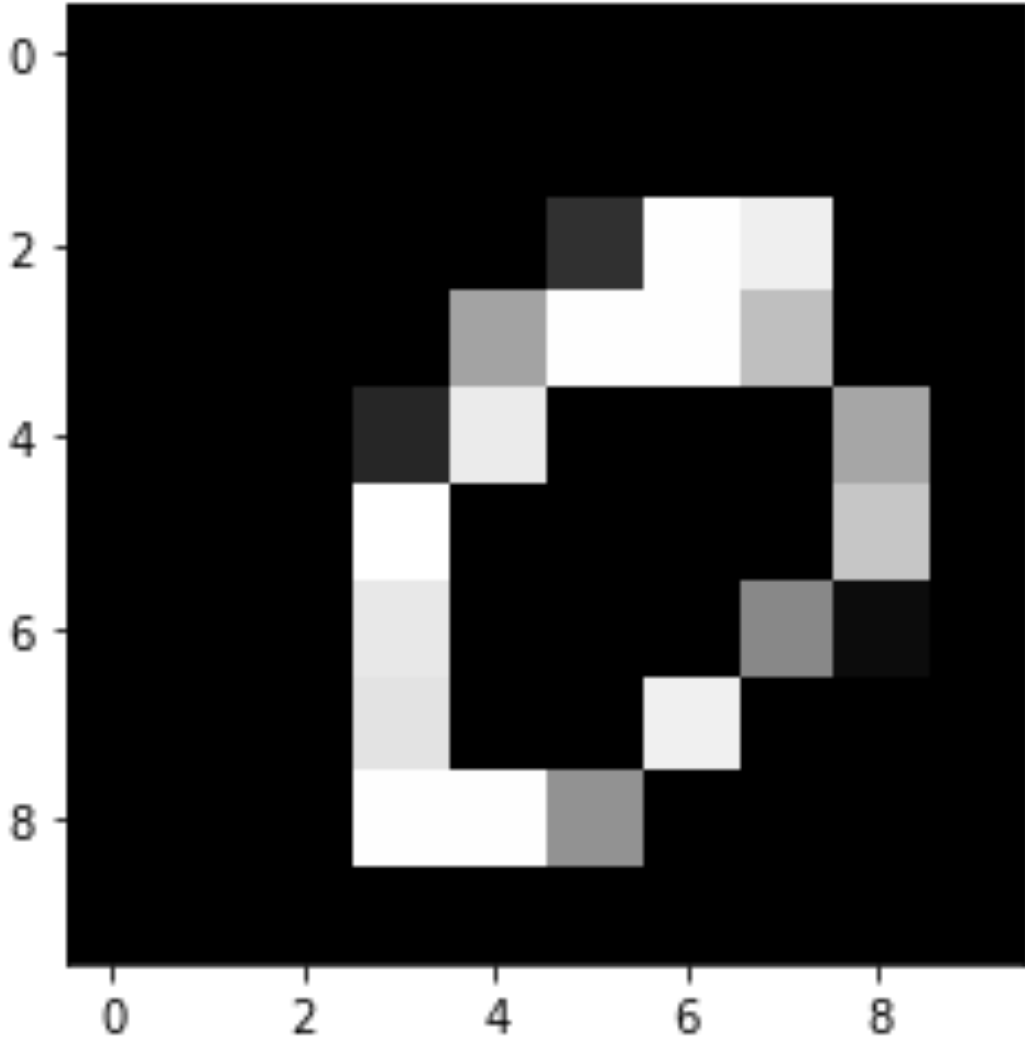
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.1875	,	0.984375	,	0.92578125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.63671875	,	
	0.984375	,	0.984375	,	0.73828125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.1484375	,	0.91015625	,	
	0.	,	0.	,	0.	,	0.64453125	,	0.	,]
[0.	,	0.	,	0.	,	0.98828125	,	0.	,	
	0.	,	0.	,	0.	,	0.765625	,	0.	,]
[0.	,	0.	,	0.	,	0.8984375	,	0.	,	
	0.	,	0.	,	0.52734375	,	0.046875	,	0.	,]
[0.	,	0.	,	0.	,	0.87890625	,	0.	,	
	0.	,	0.9296875	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.984375	,	0.984375	,	
	0.56640625	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]]

массив с элементами от 0 до 1
0 - черный, 1 - белый

Какая размерность?

Бинарная классификация

Input



[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.1875	,	0.984375	,	0.92578125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.63671875	,	
	0.984375	,	0.984375	,	0.73828125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.1484375	,	0.91015625	,	
	0.	,	0.	,	0.	,	0.64453125	,	0.	,]
[0.	,	0.	,	0.	,	0.98828125	,	0.	,	
	0.	,	0.	,	0.	,	0.765625	,	0.	,]
[0.	,	0.	,	0.	,	0.8984375	,	0.	,	
	0.	,	0.	,	0.52734375	,	0.046875	,	0.	,]
[0.	,	0.	,	0.	,	0.87890625	,	0.	,	
	0.	,	0.9296875	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.984375	,	0.984375	,	
	0.56640625	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]

X - ч/б картинка

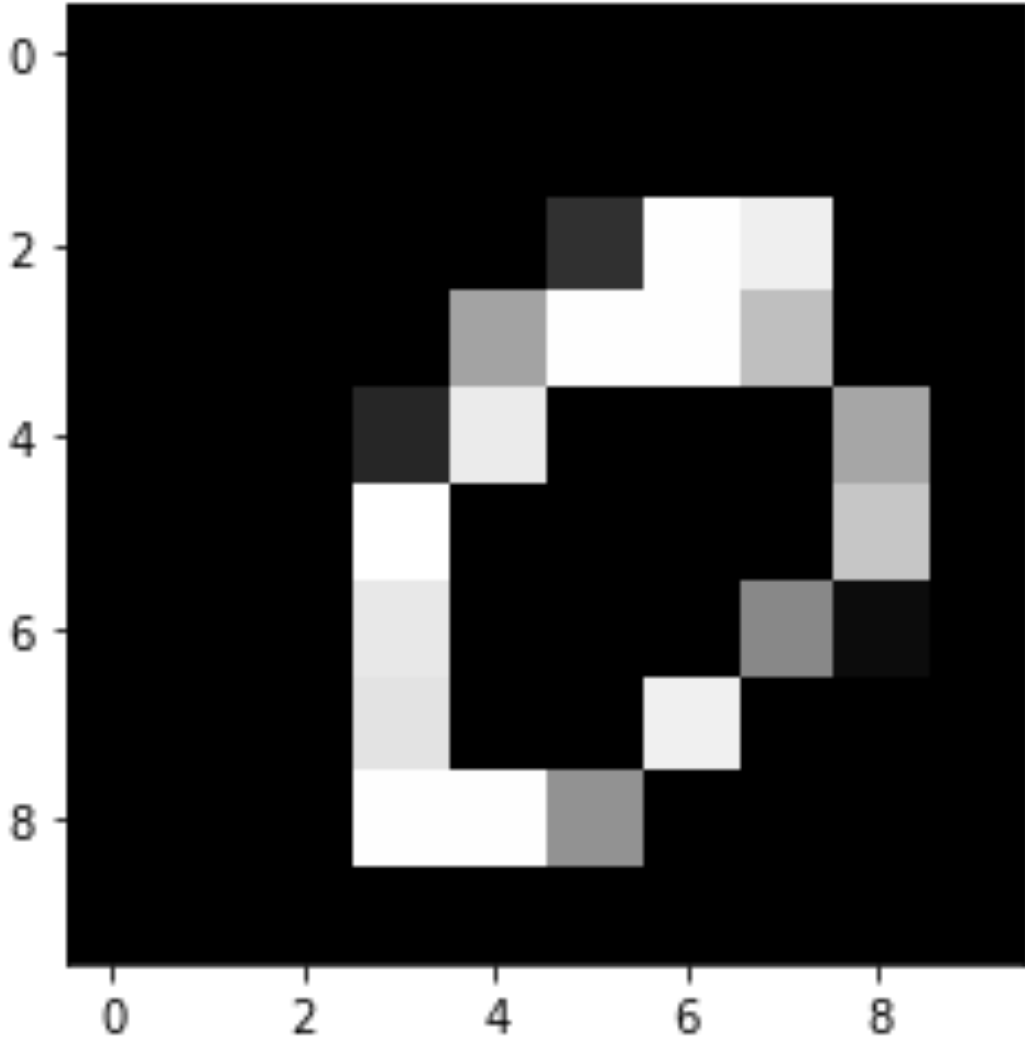
=

массив с элементами от 0 до 1
0 - черный, 1 - белый

$$H \times W \times C$$

Бинарная классификация

Input



[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.1875	,	0.984375	,	0.92578125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.63671875	,	
	0.984375	,	0.984375	,	0.73828125	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.1484375	,	0.91015625	,	
	0.	,	0.	,	0.	,	0.64453125	,	0.	,]
[0.	,	0.	,	0.	,	0.98828125	,	0.	,	
	0.	,	0.	,	0.	,	0.765625	,	0.	,]
[0.	,	0.	,	0.	,	0.8984375	,	0.	,	
	0.	,	0.	,	0.52734375	,	0.046875	,	0.	,]
[0.	,	0.	,	0.	,	0.87890625	,	0.	,	
	0.	,	0.9296875	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.984375	,	0.984375	,	
	0.56640625	,	0.	,	0.	,	0.	,	0.	,]
[0.	,	0.	,	0.	,	0.	,	0.	,	
	0.	,	0.	,	0.	,	0.	,	0.	,]

X - ч/б картинка

=

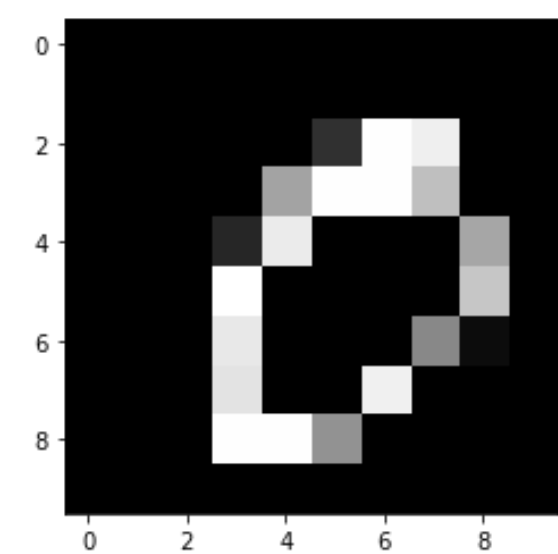
массив с элементами от 0 до 1
0 - черный, 1 - белый

$$H \times W \times C = 10 \times 10 \times 1$$

Бинарная классификация: forward pass

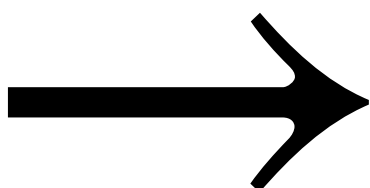
Input

x



=

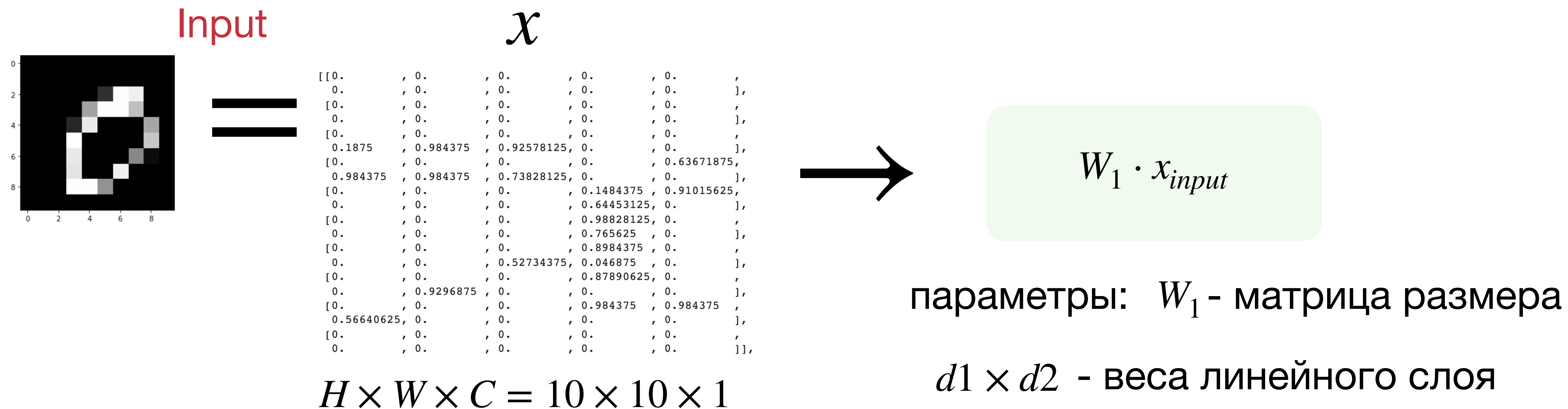
```
[ [0.      , 0.      , 0.      , 0.      , 0.      ,  
  0.      , 0.      , 0.      , 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.      , 0.      ,  
  0.      , 0.      , 0.      , 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.      , 0.      ,  
  0.1875   , 0.984375 , 0.92578125, 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.      , 0.63671875,  
  0.984375  , 0.984375 , 0.73828125, 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.1484375 , 0.91015625,  
  0.      , 0.      , 0.      , 0.64453125, 0.      ],  
  [0.      , 0.      , 0.      , 0.98828125, 0.      ,  
  0.      , 0.      , 0.      , 0.765625  , 0.      ],  
  [0.      , 0.      , 0.      , 0.8984375 , 0.      ,  
  0.      , 0.      , 0.52734375, 0.046875 , 0.      ],  
  [0.      , 0.      , 0.      , 0.87890625, 0.      ,  
  0.      , 0.9296875 , 0.      , 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.984375  , 0.984375 ,  
  0.56640625, 0.      , 0.      , 0.      , 0.      ],  
  [0.      , 0.      , 0.      , 0.      , 0.      ,  
  0.      , 0.      , 0.      , 0.      , 0.      ]],
```



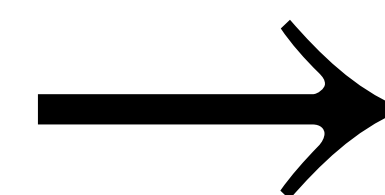
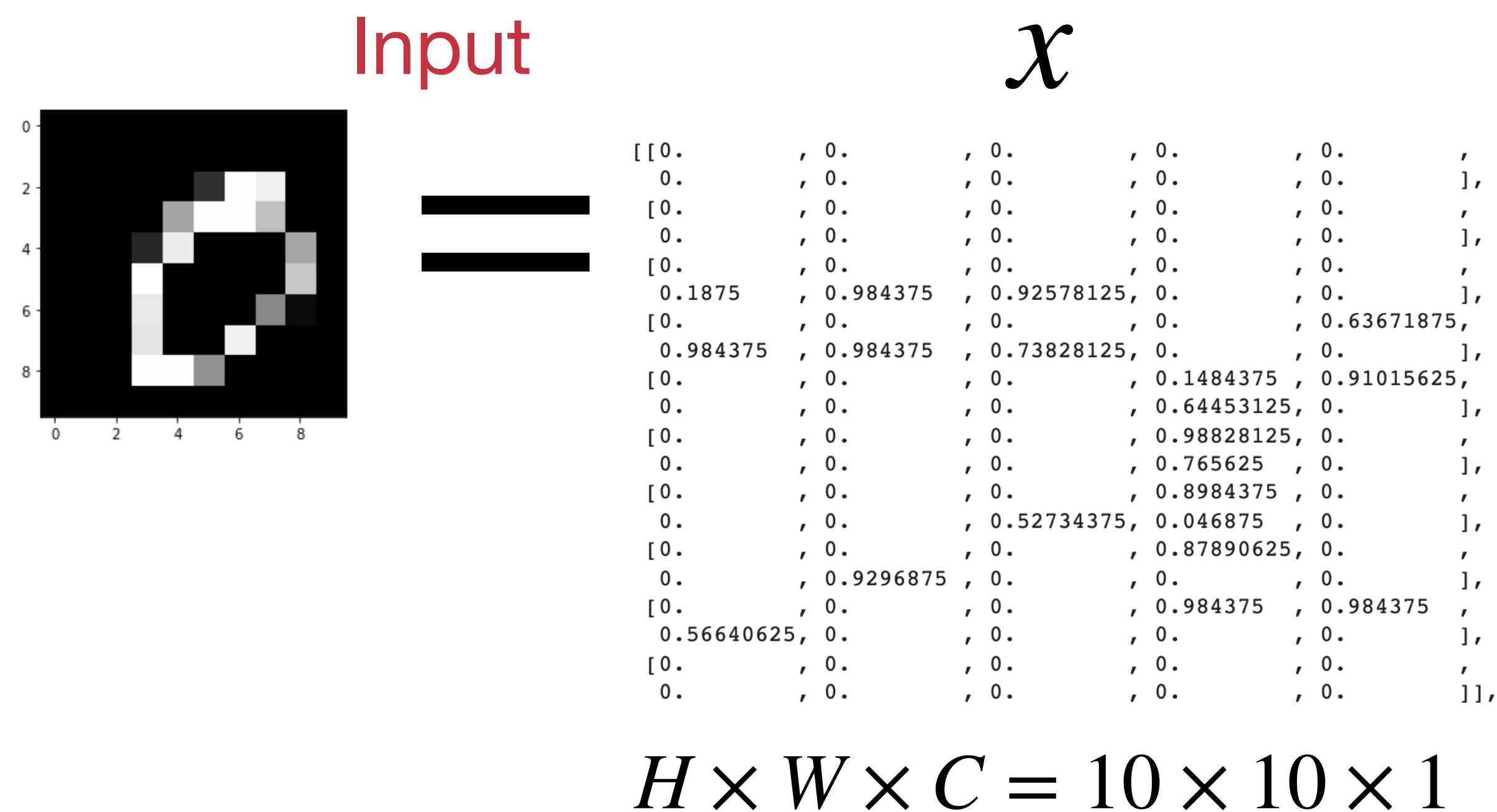
Linear

$$H \times W \times C = 10 \times 10 \times 1$$

Бинарная классификация: forward pass



Бинарная классификация: forward pass

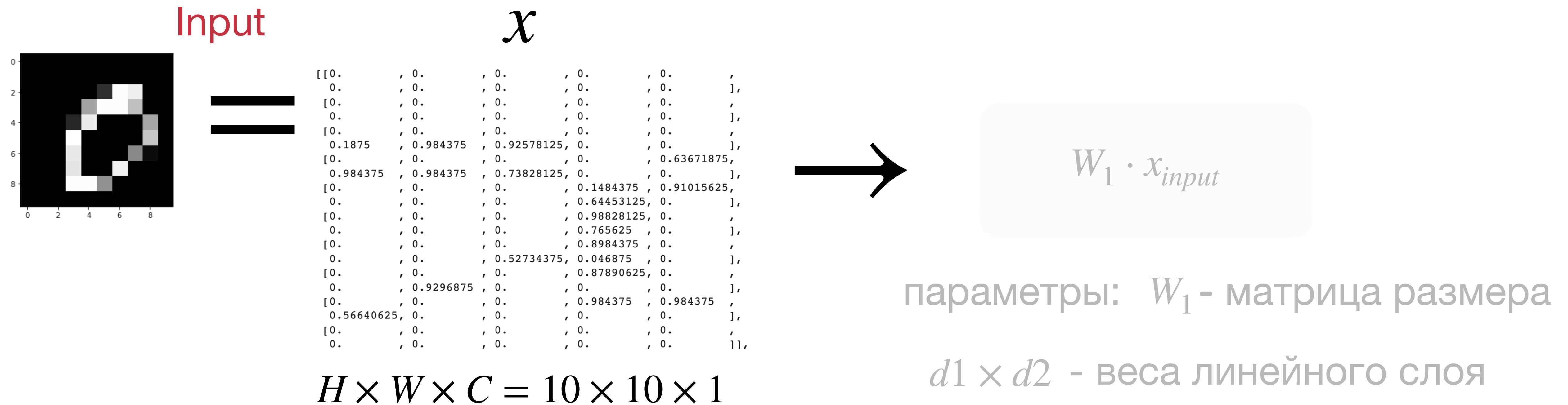


$$W_1 \cdot x_{input}$$

параметры: W_1 - матрица размера $d1 \times d2$ - веса линейного слоя

Какие параметры выбираем?

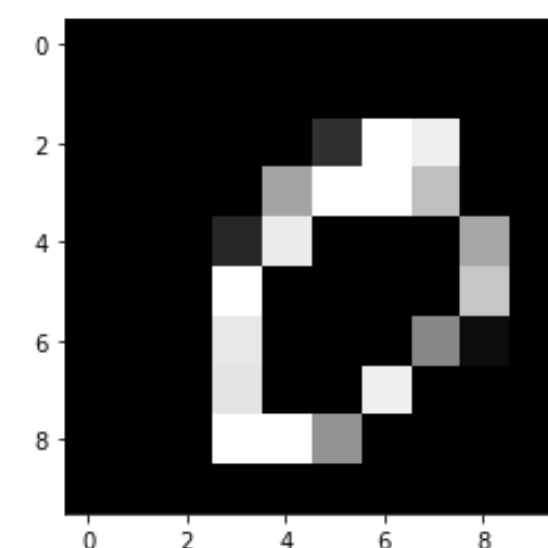
Бинарная классификация: forward pass



Нужно сначала представить картинку в виде вектора
размера $H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$

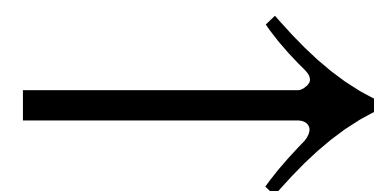
Бинарная классификация: forward pass

Input



 χ

```
[ [0. , 0. , 0. , 0. , 0. ,
  0. , 0. , 0. , 0. , 0. ],
 [0. , 0. , 0. , 0. , 0. ,
  0. , 0. , 0. , 0. , 0. ],
 [0. , 0. , 0. , 0. , 0. ,
  0.1875 , 0.984375 , 0.92578125 , 0. , 0. ],
 [0. , 0. , 0. , 0. , 0. ,
  0.984375 , 0.984375 , 0.73828125 , 0. , 0.63671875 ],
 [0. , 0. , 0. , 0.1484375 , 0.91015625 ,
  0. , 0. , 0.64453125 , 0. , 0. ],
 [0. , 0. , 0. , 0.98828125 , 0. ,
  0. , 0. , 0.765625 , 0. , 0. ],
 [0. , 0. , 0. , 0.8984375 , 0. ,
  0. , 0. , 0.52734375 , 0.046875 , 0. ],
 [0. , 0. , 0. , 0.87890625 , 0. ,
  0. , 0.9296875 , 0. , 0. , 0. ],
 [0. , 0. , 0. , 0.984375 , 0.984375 ,
  0.56640625 , 0. , 0. , 0. , 0. ],
 [0. , 0. , 0. , 0. , 0. ,
  0. , 0. , 0. , 0. , 0. ],
 [0. , 0. , 0. , 0. , 0. ]]
```

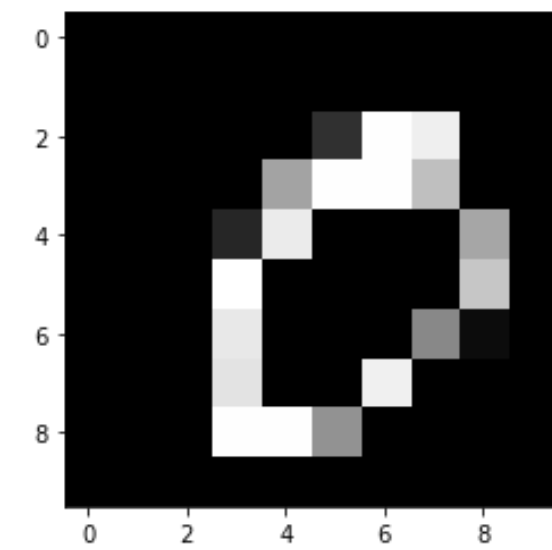
$$H \times W \times C = 10 \times 10 \times 1$$
[illegible]

$$H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$$

Операция flatten

Бинарная классификация: forward pass

Input

 χ [illegible]

$$H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$$

$$W_1 \cdot x_{input}$$

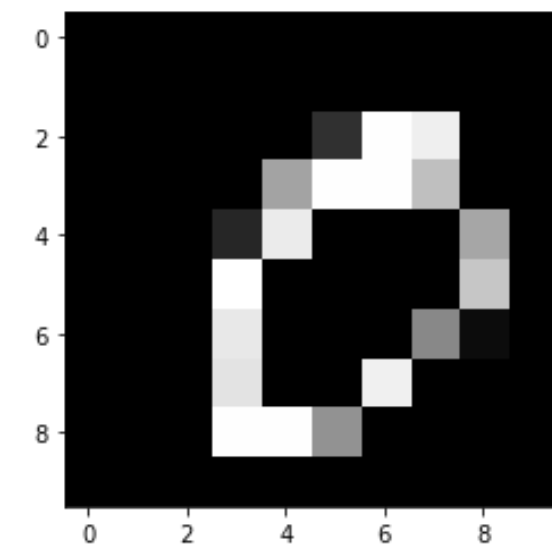
параметры: W_1 - матрица размера

$d1 \times d2$ - веса линейного слоя

Какие параметры выбираем?

Бинарная классификация: forward pass

Input

 χ [illegible]

$$H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$$

$$W_1 \cdot x_{input}$$

параметры: W_1 - матрица размера 128×100 - веса линейного слоя

Бинарная классификация: forward pass

Input

 χ [illegible]

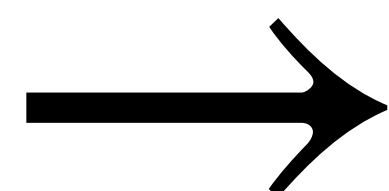
- ВЕКТОР размера

$$H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$$

Output текущего слоя

 x_{output_1}

– вектор
размера 128



$$W_1 \cdot x_{input}$$

параметры: W_1 - матрица размера

128 × 100 - веса линейного слоя



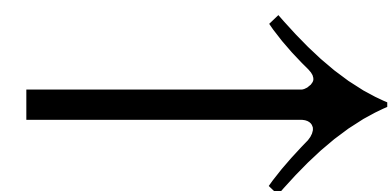
Бинарная классификация: forward pass

Input

 χ [illegible]

- ВЕКТОР размера

$$H \cdot W \cdot C = 10 \cdot 10 \cdot 1 = 100$$



$$W_1 \cdot x_{input}$$

параметры: W_1 - матрица размера

128 × 100 - веса линейного слоя



Output текущего слоя

 x_{output_1}

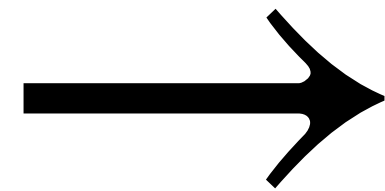
– Вектор
размера 128

Передаем дальше

Бинарная классификация: forward pass

Input текущего слоя

x_{output_1}
– вектор
размера 128

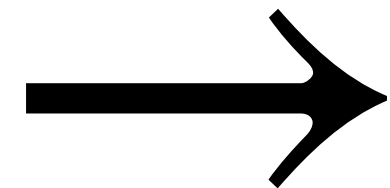


Бинарная классификация: forward pass

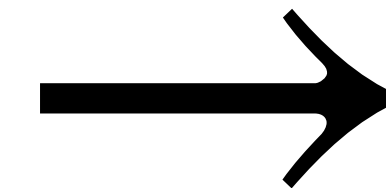
Input текущего слоя

Output текущего слоя

x_{output_1}
- вектор
размера 128



Sigmoid



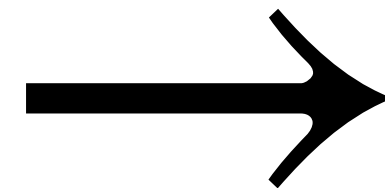
x_{output_2}

Бинарная классификация: forward pass

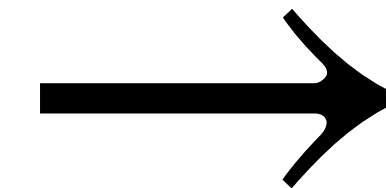
Input текущего слоя

Output текущего слоя

x_{output_1}
- вектор
размера 128



$$\frac{1}{1 + e^{-x_{output_1}}}$$

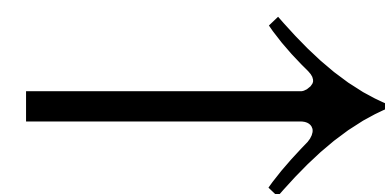


x_{output_2}

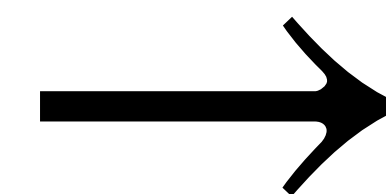
Бинарная классификация: forward pass

Input текущего слоя

x_{output_1}
- вектор
размера 128



$$\frac{1}{1 + e^{-x_{output_1}}}$$



Output текущего слоя

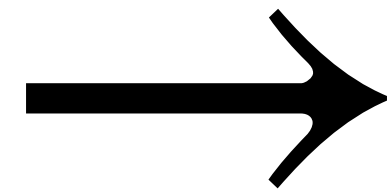
x_{output_2}
- вектор
размера 128

Бинарная классификация: forward pass

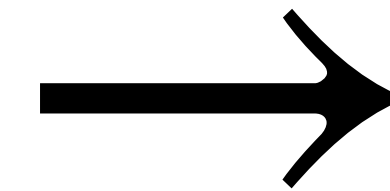
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



Linear

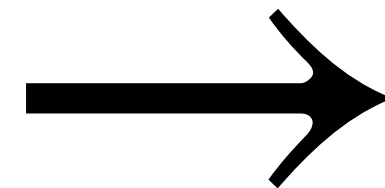


Бинарная классификация: forward pass

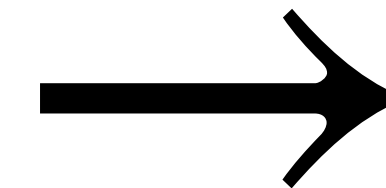
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



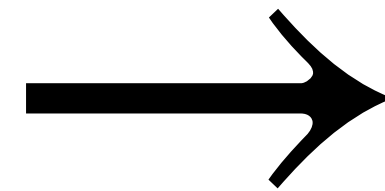
Какой размер W_2 выбираем?

Бинарная классификация: forward pass

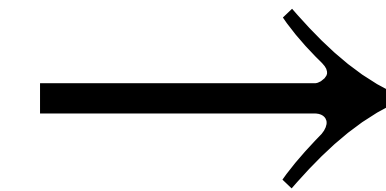
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



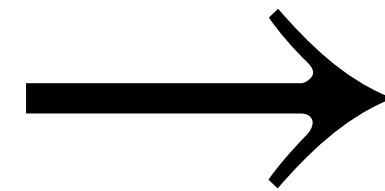
W_2 размера 1×128

Бинарная классификация: forward pass

Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1

W_2 размера 1×128

Хотим оценить
 $P(y = 1 | x, \theta)$

Бинарная классификация: forward pass

Input текущего слоя

Output текущего слоя

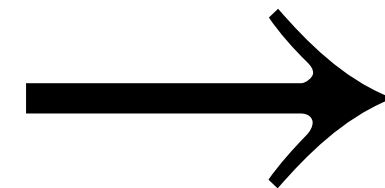


Бинарная классификация: forward pass

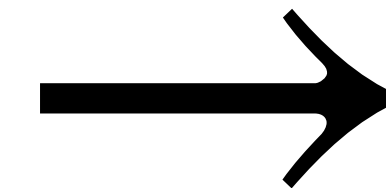
Input текущего слоя

Output текущего слоя

x_{output_3}
- число,
размер 1



$$\frac{1}{1 + e^{-x_{output_3}}}$$

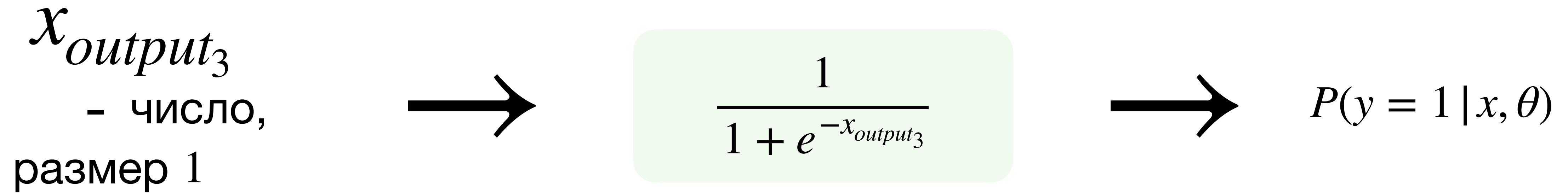


$$P(y = 1 | x, \theta)$$

Бинарная классификация: forward pass

Input текущего слоя

Output текущего слоя



Функция потерь

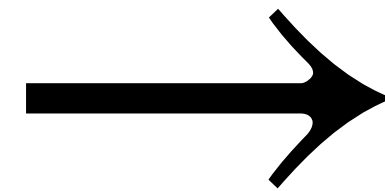
$$L(y, \hat{y}) = - [y \log P(y = 1 | x, \theta) + (1 - y) \log(1 - P(y = 1 | x, \theta))]$$

Бинарная классификация: backward pass

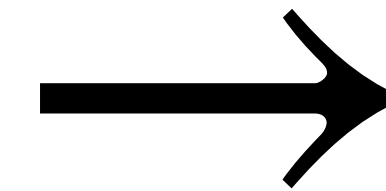
Input текущего слоя

Output текущего слоя

x_{output_3}
- число,
размер 1



$$\frac{1}{1 + e^{-x_{output_3}}}$$



$$P(y = 1 | x, \theta)$$

Хотим посчитать $\frac{dL}{dW_1}, \frac{dL}{dW_2}$

и сделать шаг градиентного спуска

Функция потерь

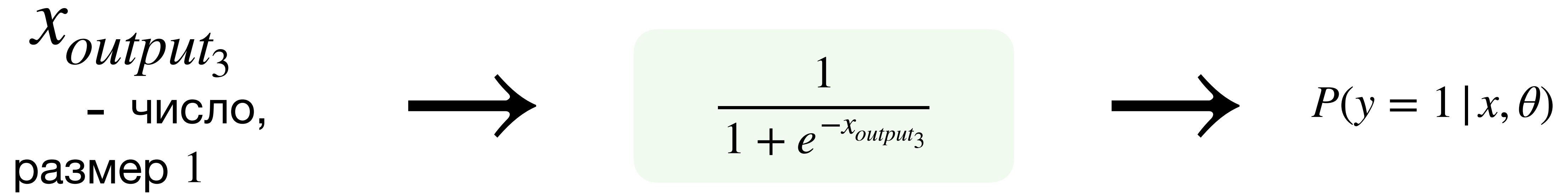
$$L(y, \hat{y}) = - [y \log P(y = 1 | x, \theta) + (1 - y) \log(1 - P(y = 1 | x, \theta))]$$

$$W_{new_i} = W_i - \alpha \frac{dL}{dW_i}$$

Бинарная классификация: backward pass

Input текущего слоя

Output текущего слоя



Функция потерь

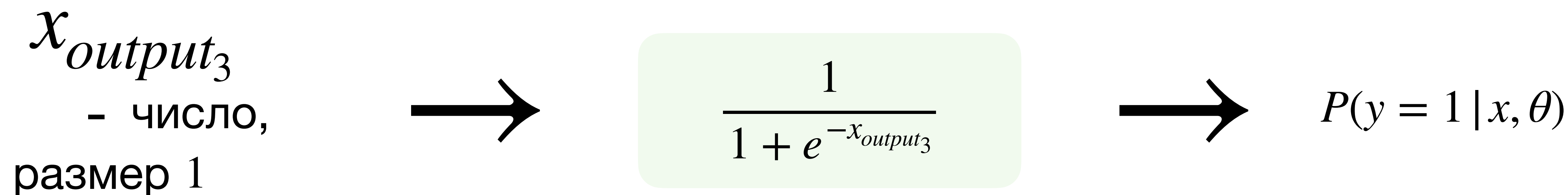
$$L(y, \hat{y}) = - [y \log P(y = 1 | x, \theta) + (1 - y) \log(1 - P(y = 1 | x, \theta))]$$

$$\frac{dL}{dL} = 1$$

Бинарная классификация: backward pass

Input текущего слоя

Output текущего слоя



Функция потерь

$$L(y, \hat{y}) = - [y \log P(y = 1 | x, \theta) + (1 - y) \log(1 - P(y = 1 | x, \theta))]$$

$$\frac{dL}{dL} = 1$$

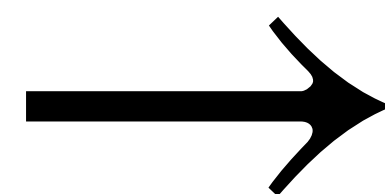
$$\frac{dL}{dP} = -\frac{y}{P} + \frac{1 - y}{1 - P}$$

Бинарная классификация: backward pass

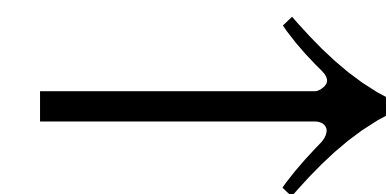
Input текущего слоя

Output текущего слоя

x_{output_3}
- число,
размер 1



$$\frac{1}{1 + e^{-x_{output_3}}}$$



$P(y = 1 | x, \theta)$



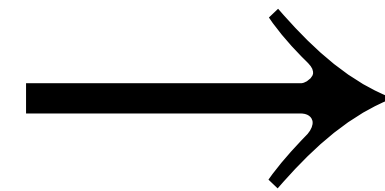
$$\frac{dL}{dP} = -\frac{y}{P} + \frac{1 - y}{1 - P}$$

Бинарная классификация: backward pass

Input текущего слоя

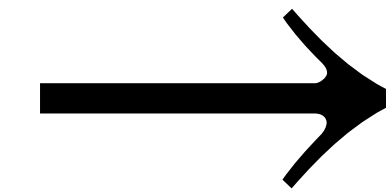
Output текущего слоя

x_{output_3}
- число,
размер 1

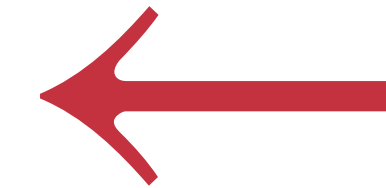


$$\frac{1}{1 + e^{-x_{output_3}}}$$

$$\frac{dP}{dx_{output_3}} = P(1 - P)$$



$$P(y = 1 | x, \theta)$$



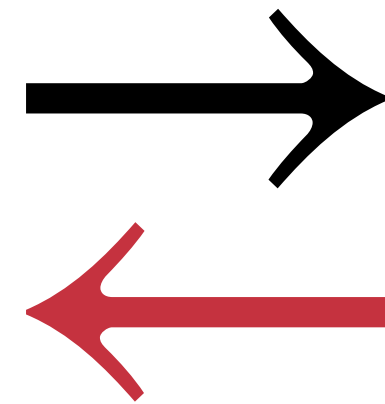
$$\frac{dL}{dP} = -\frac{y}{P} + \frac{1 - y}{1 - P}$$

Бинарная классификация: backward pass

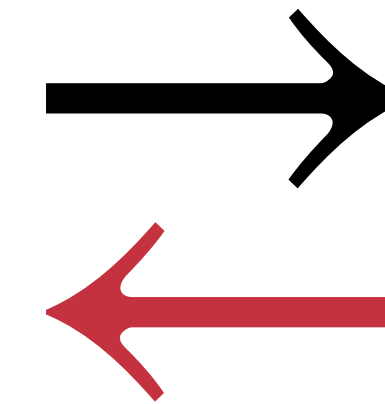
Input текущего слоя

Output текущего слоя

x_{output_3}
- число,
размер 1



$$\frac{1}{1 + e^{-x_{output_3}}}$$



$P(y = 1 | x, \theta)$

$$\frac{dL}{dx_{output_3}} = \frac{dL}{dP} \frac{dP}{dx_{output_3}}$$

$$\frac{dP}{dx_{output_3}} = P(1 - P)$$

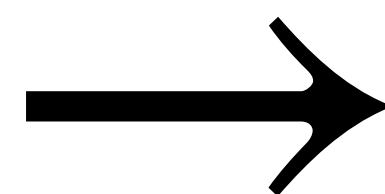
$$\frac{dL}{dP} = -\frac{y}{P} + \frac{1 - y}{1 - P}$$

Бинарная классификация: backward pass

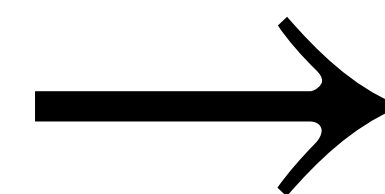
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1

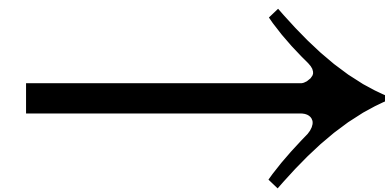
$$\frac{dL}{dx_{output_3}} = \frac{dL}{dP} \frac{dP}{dx_{output_3}}$$

Бинарная классификация: backward pass

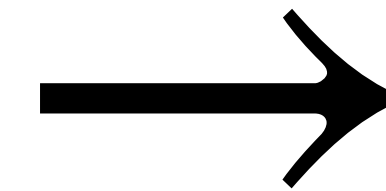
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1



$$\frac{dx_{output_3}}{dW_2} = x_{output_2}^T$$

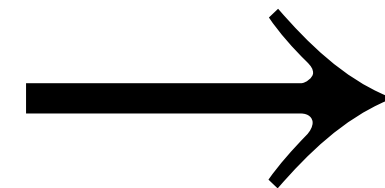
$$\frac{dL}{dx_{output_3}}$$

Бинарная классификация: backward pass

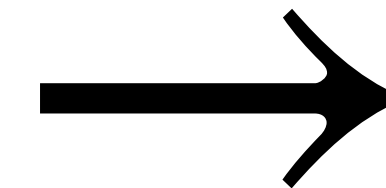
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1



$$\frac{dx_{output_3}}{dW_2} = x_{output_2}^T$$

$$\frac{dL}{dx_{output_3}}$$

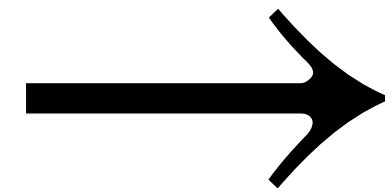
$$\frac{dL}{dW_2} = \frac{dL}{dx_{output_3}} \frac{dx_{output_3}}{dW_2} = \frac{dL}{dx_{output_3}} x_{output_2}^T$$

Бинарная классификация: backward pass

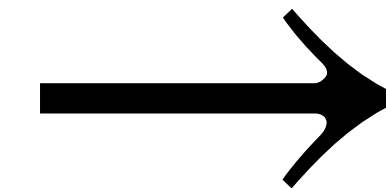
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1



$$\frac{dx_{output_3}}{dW_2} = x_{output_2}^T$$

$$\frac{dL}{dx_{output_3}}$$

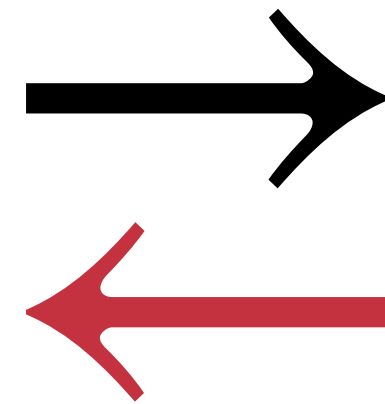
$$\frac{dL}{dW_2} = \frac{dL}{dx_{output_3}} \frac{dx_{output_3}}{dW_2} = \frac{dL}{dx_{output_3}} x_{output_2}^T \Rightarrow W_{new_2} = W_2 - \alpha \frac{dL}{dW_2}$$

Бинарная классификация: backward pass

Input текущего слоя

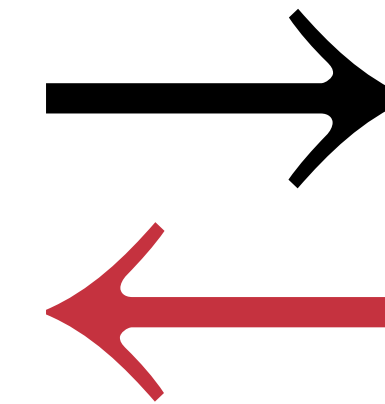
Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$

$$\frac{dx_{output_3}}{dx_{output_2}} = W_2^T$$



x_{output_3}
- число,
размер 1

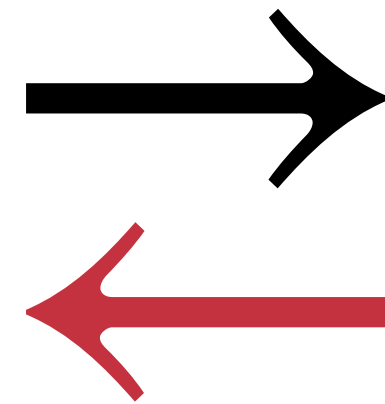
$$\frac{dL}{dx_{output_3}}$$

Бинарная классификация: backward pass

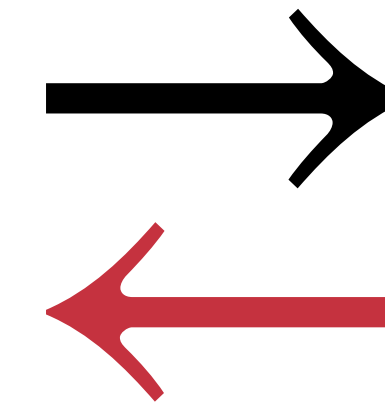
Input текущего слоя

Output текущего слоя

x_{output_2}
- вектор
размера 128



$$W_2 \cdot x_{output_2}$$



x_{output_3}
- число,
размер 1

$$\frac{dL}{dx_{output_2}} = \frac{dL}{dx_{output_3}} \frac{dx_{output_3}}{dx_{output_2}}$$

$$\frac{dx_{output_3}}{dx_{output_2}} = W_2^T$$

$$\frac{dL}{dx_{output_3}}$$

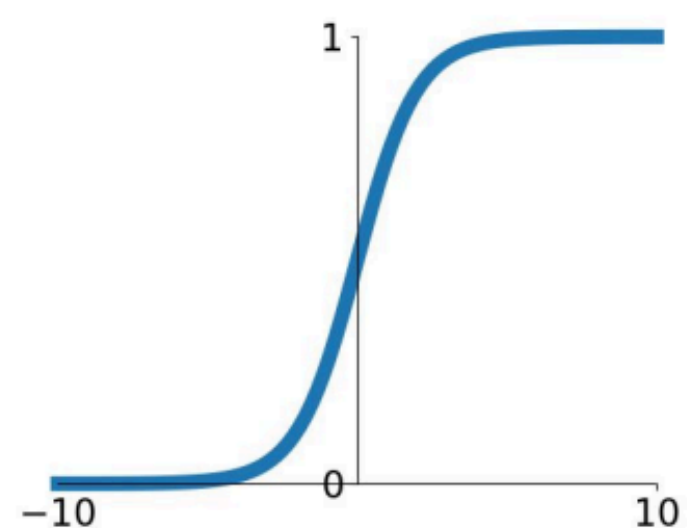
Train loop

```
for epoch in range(epochs):           # эпоха - проход по датасету
    model.train()                       # переключаем все в режим тренировки (DO/BN/...)
    for x, gt in tqdm(train_loader):    # датасет разбит на (мини)батчи
        logits = network.forward(x)    # предсказания сети
        loss = loss_fn(logits, gt)     # подсчет ошибки
        accuracy = accuracy(logits, gt) # подсчет метрик
        loss.backward()                # подсчет градиентов
        network.apply_updates()        # обновление весов
    model.eval()                        # переключаем все в режим валидации (DO/BN/...)
    for x, gt in tqdm(val_loader):      # валидация
        logits = network.forward(x)
        loss = loss_fn(logits, gt)     # подсчет ошибок
        accuracy = accuracy(logits, gt) # подсчет метрик
```

Функции активации

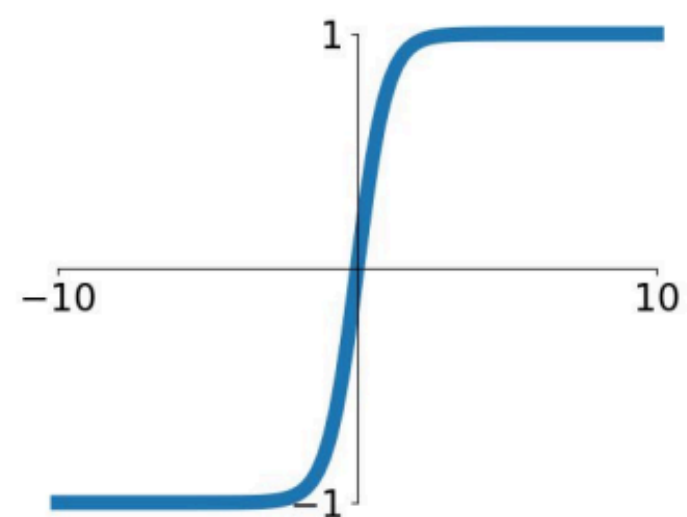
Sigmoid

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



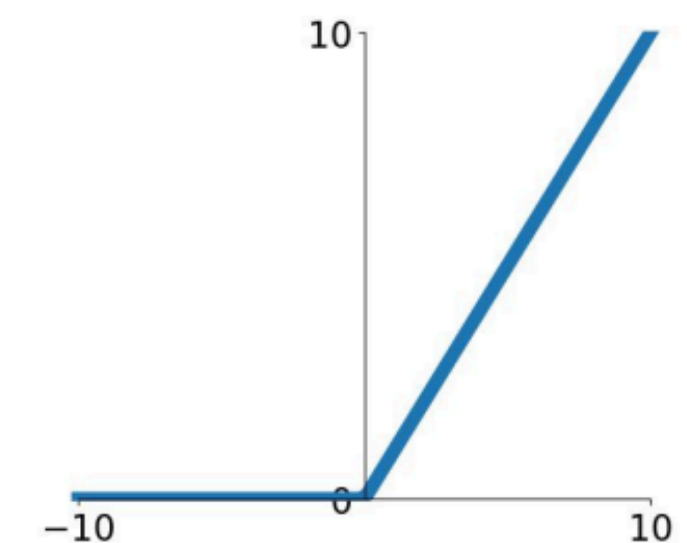
tanh

$$\tanh(x)$$



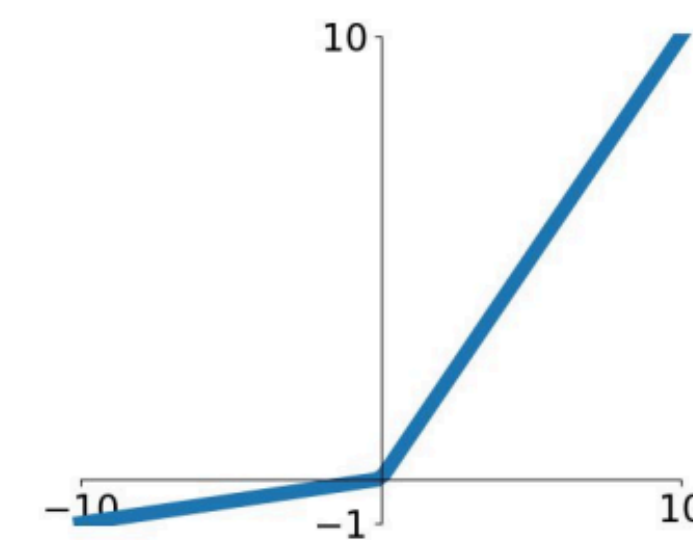
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

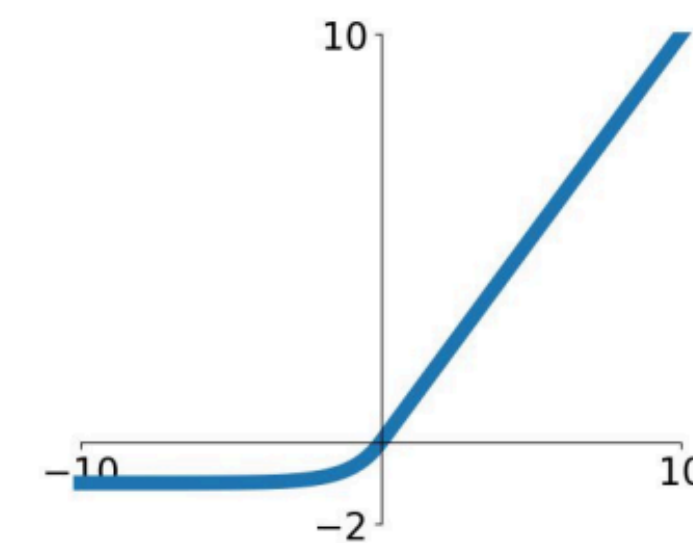


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

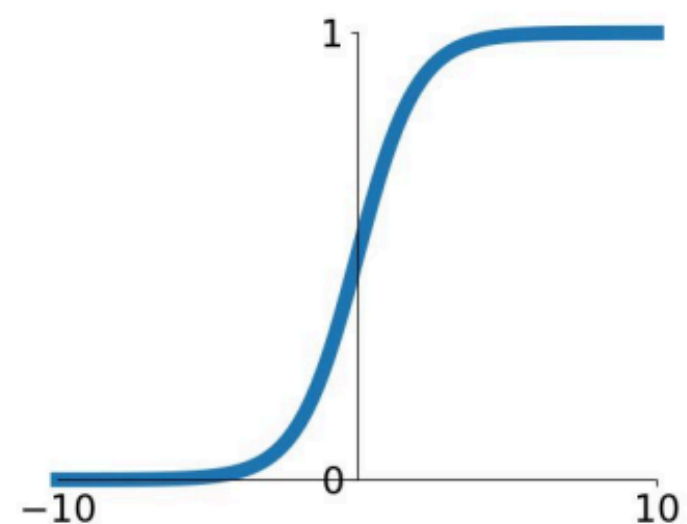
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Функции активации: Sigmoid

Sigmoid

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



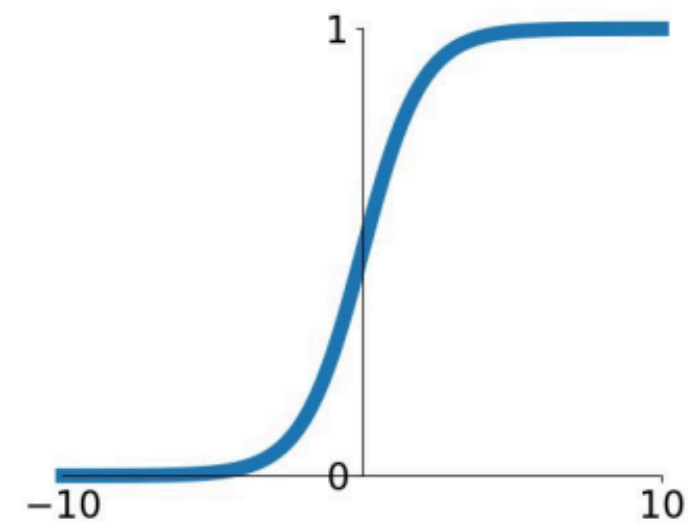
Выход в диапазоне от 0 до 1

Проблемы?

Функции активации: Sigmoid

Sigmoid

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

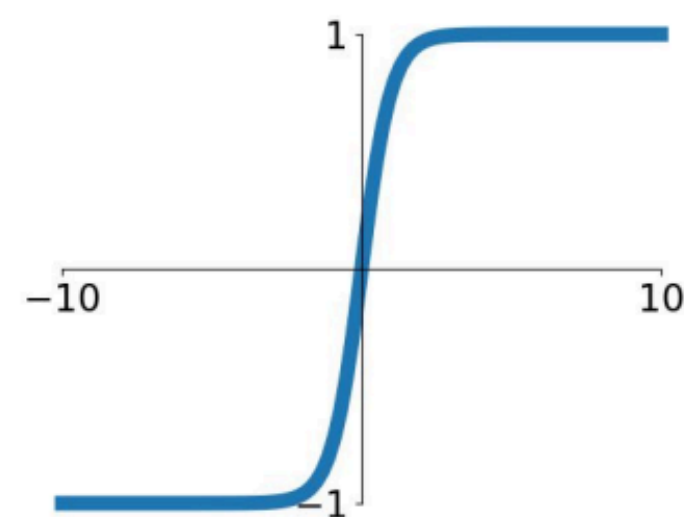


Выходы в диапазоне от 0 до 1

- На краях одинаково работает
- Local grad на краях маленький
- Выходы не центрированы

Функции активации: Tanh

tanh
 $\tanh(x)$

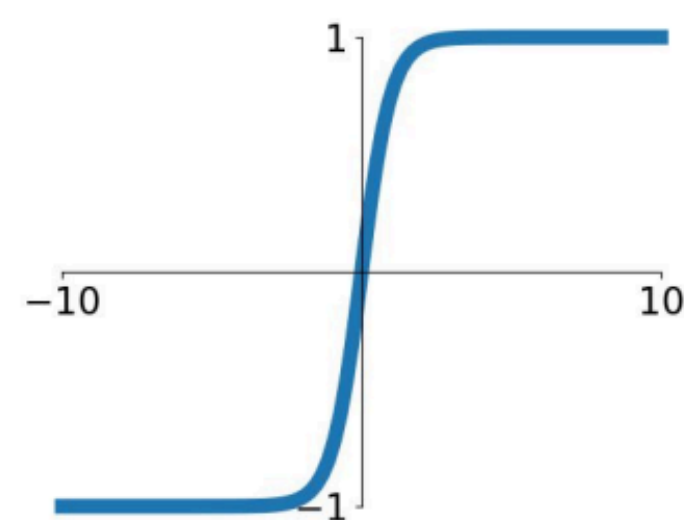


Выходы в диапазоне от -1 до 1

- Выходы центрированы

Функции активации: Tanh

tanh
 $\tanh(x)$

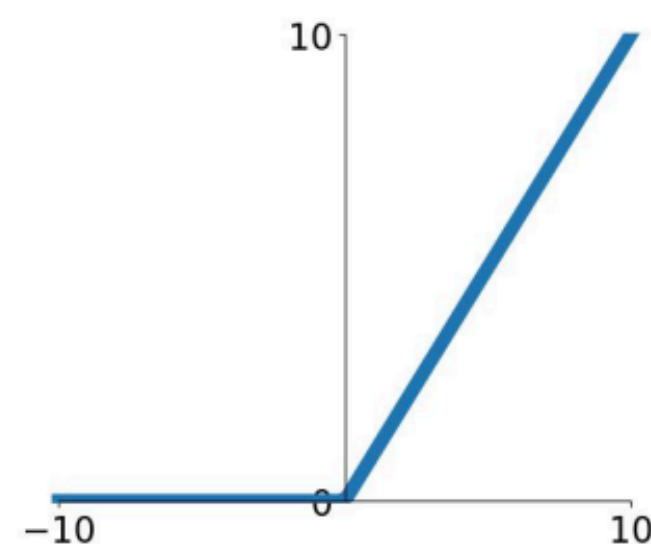


Выходы в диапазоне от -1 до 1

- На краях одинаково работает
- Local grad на краях маленький
- Выходы центрированы

Функции активации: ReLU

ReLU
 $\max(0, x)$

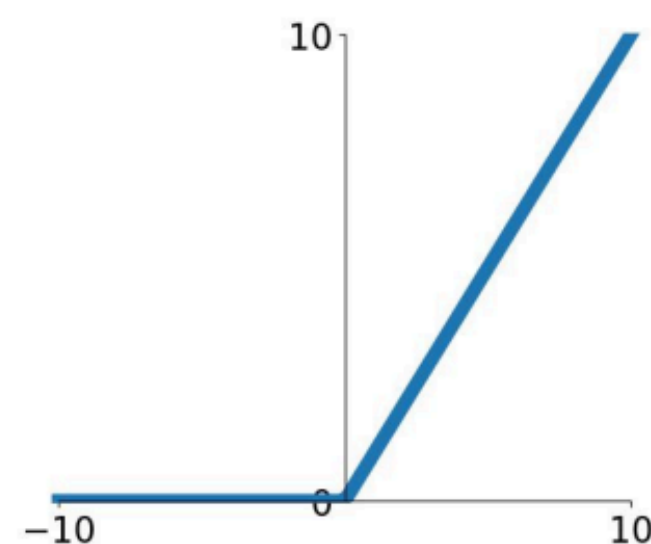


Выходы в диапазоне от 0 до $+\infty$

- На краях работает по-разному
- Нет вычислений exp
- Быстрая сходимость (x6)

Функции активации: ReLU

ReLU
 $\max(0, x)$

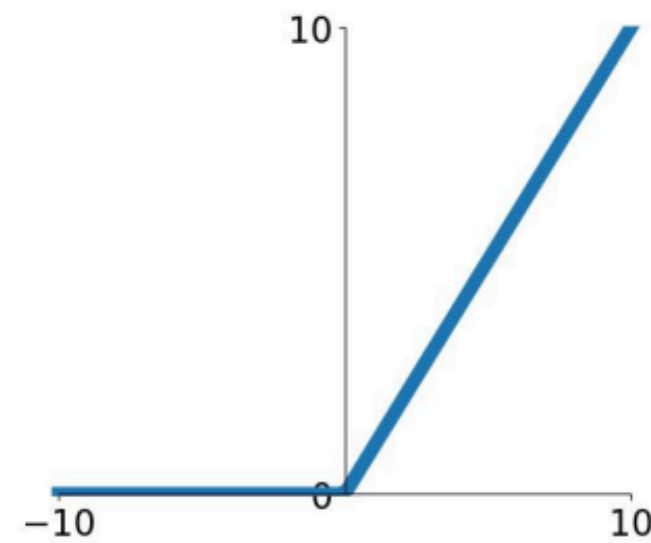


Выходы в диапазоне от 0 до $+\infty$

- На краях работает по-разному
- Нет вычислений \exp
- Быстрая сходимость (x6)
- Выходы не центрированы
- Local grad для $x < 0$

Функции активации: ReLU

ReLU
 $\max(0, x)$



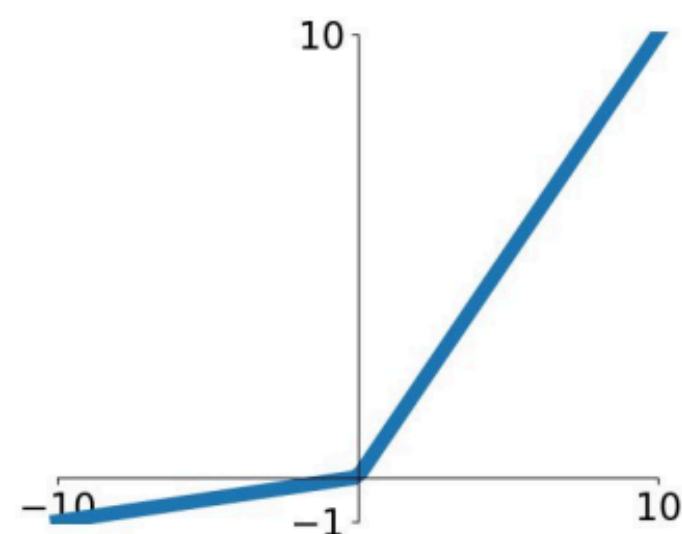
Выходы в диапазоне от 0 до $+\infty$

- На краях работает по-разному
- Нет вычислений exp
- Быстрая сходимость (x6)
- Выходы не центрированы
- Local grad для $x < 0$

Хороший выбор, но learning rate не должен быть большим

Функции активации: Leaky ReLU

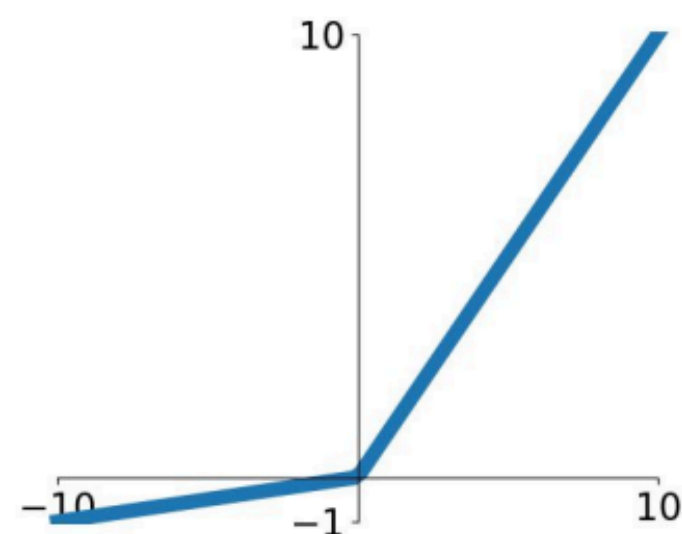
Leaky ReLU
 $\max(0.1x, x)$



Выходы в диапазоне от $-\infty$ до $+\infty$

Функции активации: Leaky ReLU

Leaky ReLU
 $\max(0.1x, x)$



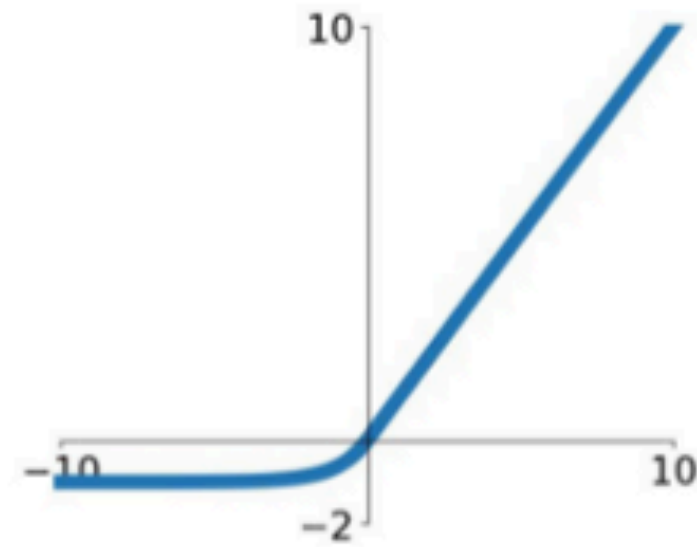
Выходы в диапазоне от $-\infty$ до $+\infty$

- На краях работает по-разному
- Нет вычислений \exp
- Быстрая сходимость

Функции активации: ELU

ELU

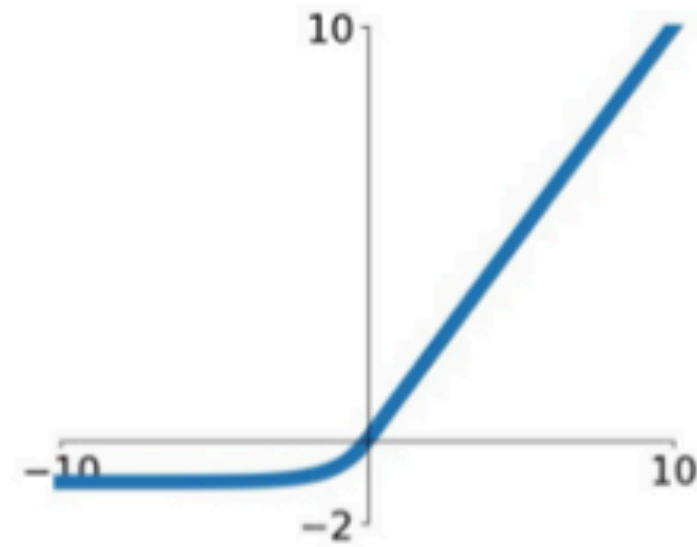
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Функции активации: ELU

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Выходы в диапазоне от $-\infty$ до $+\infty$

- На краях работает по-разному
- Быстрая сходимость
- Вычисляем \exp

Функции активации: ВЫВОД

ReLU - хороший базовый выбор

Можно пробовать **LeakyReLU**, **ELU**, **GELU**, etc.

Избегать **Sigmoid**

Функции активации: ВЫВОД

ReLU - хороший базовый выбор

Можно пробовать **LeakyReLU**, **ELU**, **GELU**, etc.

Избегать **Sigmoid**

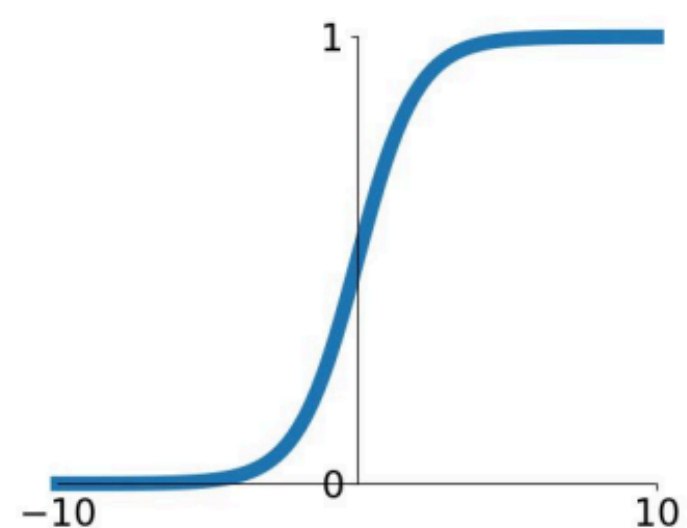
Важно - подбирать lr , инициализации весов...

Виды слоев в нейросетях

Функции активации

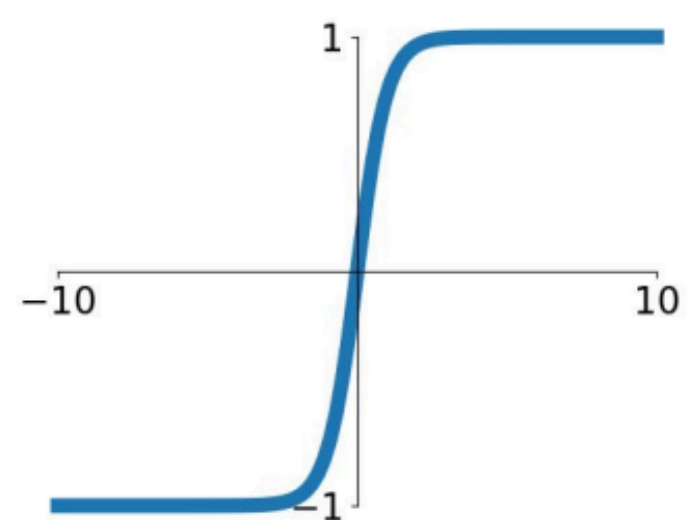
Sigmoid

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



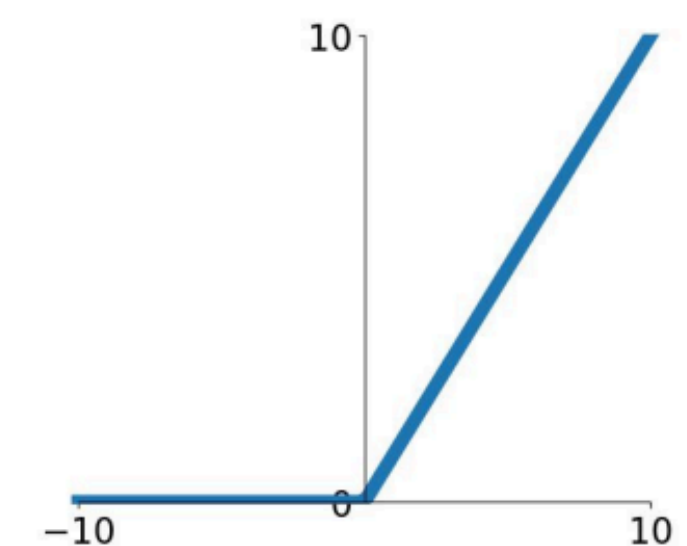
tanh

$$\tanh(x)$$



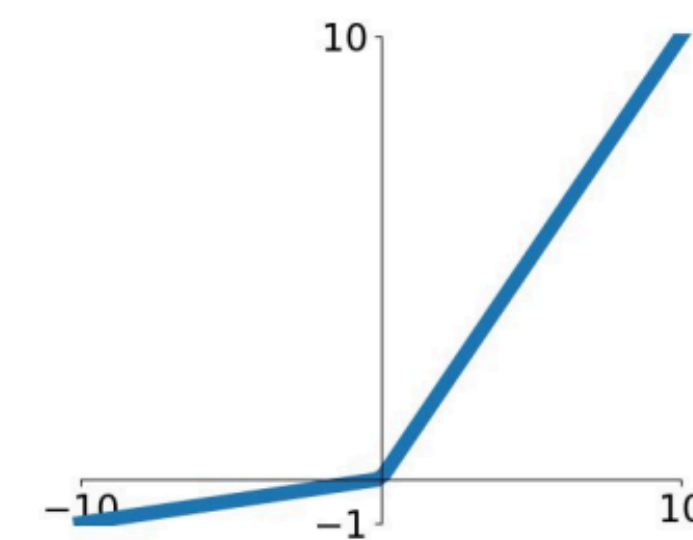
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

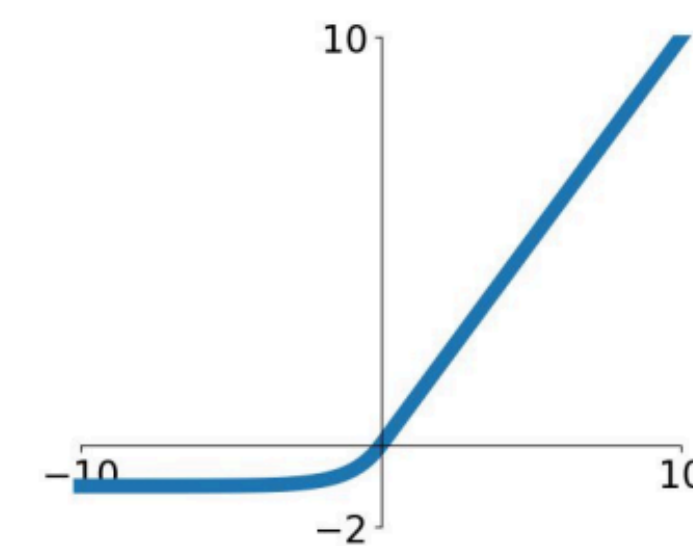


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Функции активации: ВЫВОД

ReLU - хороший базовый выбор

Можно пробовать **LeakyReLU**, **ELU**, **GELU**, etc.

Избегать **Sigmoid**

Важно - подбирать lr , инициализации весов...

Инициализация

Инициализация

Какие значения выбрать при построении сети для весов?

Инициализация

Инициализация нулями?

Инициализация

Инициализация нулями?

Градиентный спуск: $\theta_{t+1} = \theta_t - \alpha \frac{dL}{d\theta}$

Веса будут меняться одинаково!

Инициализация

Инициализация случайными значениями

А есть значения слишком большие?

Инициализация

Инициализация случайными значениями

А есть значения слишком большие?

Рассмотрим MLP с L слоями, без активация (identity активации)

$$W_1 = W_2 = \dots = W_L = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix} = 1.5 \cdot I$$

$$y_L = W_L \cdot \dots \cdot W_1 \cdot x = 1.5^L x$$

Инициализация

Инициализация случайными значениями

А есть значения слишком большие?

Рассмотрим MLP с L слоями, без активация (identity активации)

$$W_1 = W_2 = \dots = W_L = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix} = 1.5 \cdot I$$

$$y_L = W_L \cdot \dots \cdot W_1 \cdot x = 1.5^L x$$

Backward pass: exploding gradients

Инициализация

Инициализация **небольшими** случайными значениями

Инициализация

Инициализация **небольшими** случайными значениями

Рассмотрим MLP с L слоями, без активация (identity активации)

$$W_1 = W_2 = \dots = W_L = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} = 0.5 \cdot I$$

$$y_L = W_L \cdot \dots \cdot W_1 \cdot x = 0.5^L x$$

Инициализация

Инициализация **небольшими** случайными значениями

Рассмотрим MLP с L слоями, без активация (identity активации)

$$W_1 = W_2 = \dots = W_L = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} = 0.5 \cdot I$$

$$y_L = W_L \cdot \dots \cdot W_1 \cdot x = 0.5^L x$$

Backward pass: vanishing gradients

Инициализация

Инициализация **небольшими** случайными значениями

Поможет калиброванная инициализация: Xavier/Glorot init, He init

Инициализация

Инициализация **небольшими** случайными значениями

Поможет калиброванная инициализация: Xavier/Glorot init, He init

Идея:

- Mean выходов слоев должны быть 0 $E y_{L-1} = E y_L = 0$
- Variance выходов слоев должны быть одинаковыми $Var y_{L-1} = Var y_L$

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию

$$\text{Var}[y_i] = \text{Var}[w_i x_i] = \mathbb{E}[w_i^2 x_i^2] - (\mathbb{E}[w_i x_i])^2 =$$

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию

$$\begin{aligned}\text{Var}[y_i] &= \text{Var}[w_i x_i] = \mathbb{E}[w_i^2 x_i^2] - (\mathbb{E}[w_i x_i])^2 = \\ &= \mathbb{E}[x_i]^2 \text{Var}[w_i] + \mathbb{E}[w_i]^2 \text{Var}[x_i] + \text{Var}[w_i] \text{Var}[x_i]\end{aligned}$$

Формула для дисперсии произведения независимых с.в.

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию

$$\begin{aligned}\text{Var}[y_i] &= \text{Var}[w_i x_i] = \mathbb{E}[w_i^2 x_i^2] - (\mathbb{E}[w_i x_i])^2 = \\ &= \cancel{\mathbb{E}[x_i]^2} \text{Var}[w_i] + \cancel{\mathbb{E}[w_i]^2} \text{Var}[x_i] + \text{Var}[w_i] \text{Var}[x_i]\end{aligned}$$

Потребуем, чтобы мат.ожидания были 0

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона:

$$\text{Var}[y] = \text{Var}\left[\sum_{i=1}^{n_{\text{out}}} y_i\right] = \sum_{i=1}^{n_{\text{out}}} \text{Var}[w_i x_i] = n_{\text{out}} \text{Var}[w_i] \text{Var}[x_i]$$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона:

$$\text{Var}[y] = \text{Var}\left[\sum_{i=1}^{n_{\text{out}}} y_i\right] = \sum_{i=1}^{n_{\text{out}}} \text{Var}[w_i x_i] = n_{\text{out}} \text{Var}[w_i] \text{Var}[x_i]$$

Дисперсия выхода

Дисперсия входа

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона:

$$\text{Var}[y] = \text{Var}\left[\sum_{i=1}^{n_{\text{out}}} y_i\right] = \sum_{i=1}^{n_{\text{out}}} \text{Var}[w_i x_i] = n_{\text{out}} \text{Var}[w_i] \text{Var}[x_i]$$

Дисперсия выхода = Дисперсия входа

ХОТИМ

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона:

для всего нейрона:

$$\text{Var}[y] = \text{Var}\left[\sum_{i=1}^{n_{\text{out}}} y_i\right] = \sum_{i=1}^{n_{\text{out}}} \text{Var}[w_i x_i] = n_{\text{out}} \text{Var}[w_i] \text{Var}[x_i]$$

Дисперсия выхода

ХОТИМ

Дисперсия входа

должно быть = 1

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона: $n_{\text{out}} \text{Var}[w_i] = 1$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона: $n_{\text{out}} \text{Var}[w_i] = 1$

$$\text{Var}[w_i] = \frac{1}{n_{\text{out}}}$$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Считаем дисперсию $\text{Var}[y_i] = \text{Var}[w_i] \text{Var}[x_i]$

Для всего нейрона: $n_{\text{out}} \text{Var}[w_i] = 1$

$$\text{Var}[w_i] = \frac{1}{n_{\text{out}}}$$

А есть тоже самое для backward pass?

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Forward pass: $n_{\text{out}} \text{Var}[w_i] = 1$

$$\text{Var}[w_i] = \frac{1}{n_{\text{out}}}$$

Backward pass: $n_{\text{in}} \text{Var}[w_i] = 1$

$$\text{Var}[w_i] = \frac{1}{n_{\text{in}}}$$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Forward pass:

$$n_{\text{out}} \text{Var}[w_i] = 1$$

$$\text{Var}[w_i] = \frac{1}{n_{\text{out}}}$$

Возьмем среднее

Backward pass:

$$n_{\text{in}} \text{Var}[w_i] = 1$$

$$\text{Var}[w_i] = \frac{1}{n_{\text{in}}}$$

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

$$\text{Var}[w_i] = \frac{2}{n_{\text{in}} + n_{\text{out}}}$$

Инициализация: Xavier/Glorot

Рассмотрим **нейрон** $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев должны быть 0 $\rightarrow E w_L = 0$
- Variance выходов слоев должны быть одинаковыми $\rightarrow \text{Var } w_L = \frac{2}{n_{in} + n_{out}}$

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев должны быть 0 $\rightarrow E w_L = 0$
- Variance выходов слоев должны быть одинаковыми $\rightarrow \text{Var } w_L = \frac{2}{n_{in} + n_{out}}$

Какое распределение подходит?

Инициализация: Xavier/Glorot

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев должны быть 0 $\rightarrow E w_L = 0$
- Variance выходов слоев должны быть одинаковыми $\rightarrow \text{Var } w_L = \frac{2}{n_{in} + n_{out}}$

Какое распределение подходит? $w_i \sim U \left[-\frac{\sqrt{6}}{\sqrt{n_{in} + n_{out}}}, \frac{\sqrt{6}}{\sqrt{n_{in} + n_{out}}} \right]$

Инициализация: Хе

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев может быть не 0 (например, с ReLU активациями)
- Variance выходов слоев должны быть одинаковыми

Логика вывода похожая

Инициализация: Хе

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев может быть не 0 (например, с ReLU активациями)
- Variance выходов слоев должны быть одинаковыми $\rightarrow \text{Var } w_L = \frac{2}{n_{in}}$

Какое распределение подходит?

Инициализация: Хе

Рассмотрим нейрон $y = w^T x = \sum_i w_i x_i$

Идея:

- Mean выходов слоев может быть не 0 (например, с ReLU активациями)
- Variance выходов слоев должны быть одинаковыми $\rightarrow \text{Var } w_L = \frac{2}{n_{in}}$

Какое распределение подходит? $w_i \sim N(0, \sqrt{2/n_{in}^{(l)}})$

Пара полезных ссылок

<https://towardsdatascience.com/>

<https://www.deeplearning.ai/ai-notes/initialization/index.html>