Deep Learning School

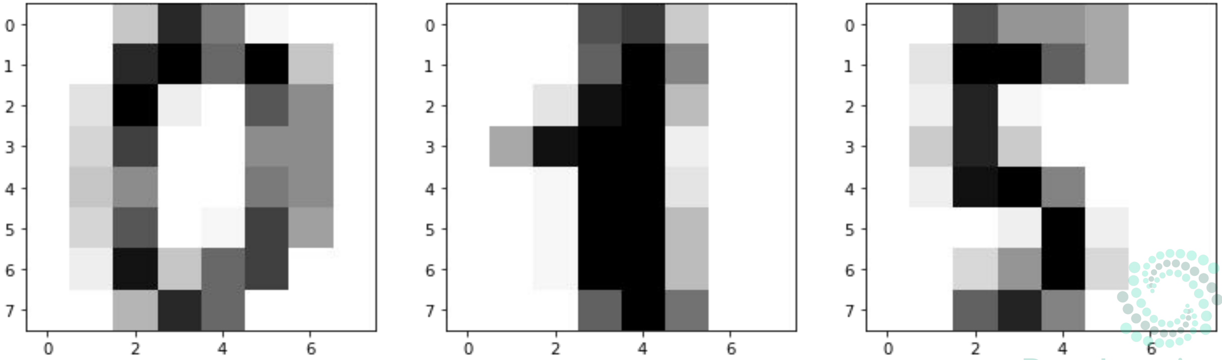
Лекция
Полносвязная нейронная сеть

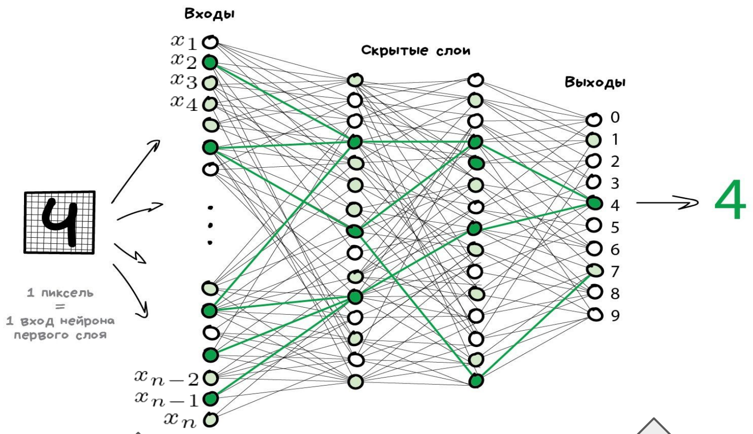
План лекции

- Модель нейрона
- Полносвязная нейронная сеть для классификации
  - Скрытые слои
  - Последний слой
- Обучение нейронных сетей
  - Loss function
  - Back Propagation
- Разбор BackProp для полносвязного слоя

Задача: распознавание рукописных цифр

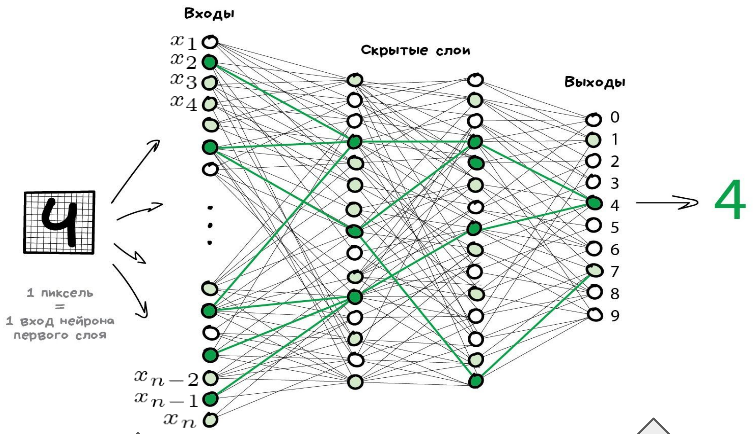
- **Дано**: чёрно-белые изображения 8x8
- **Определить:** какая из 10 цифр нарисована (10 классов)
- **Имеется**: обучающая выборка *размеченных* изображений
    - Несколько тысяч изображений с известными классами





Описание модели

Обучение модели

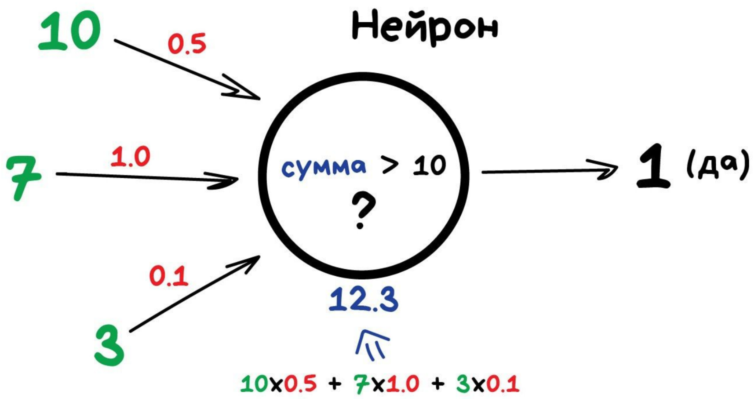


Описание модели

Обучение модели

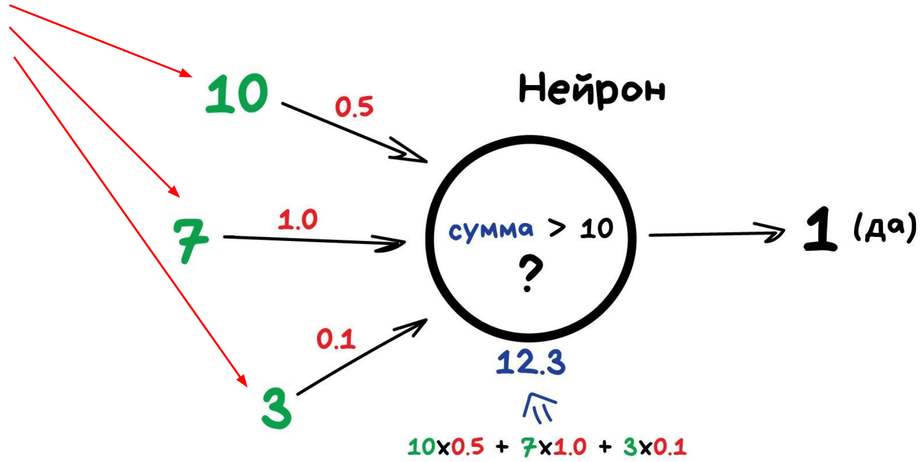
Один нейрон

Модель нейрона



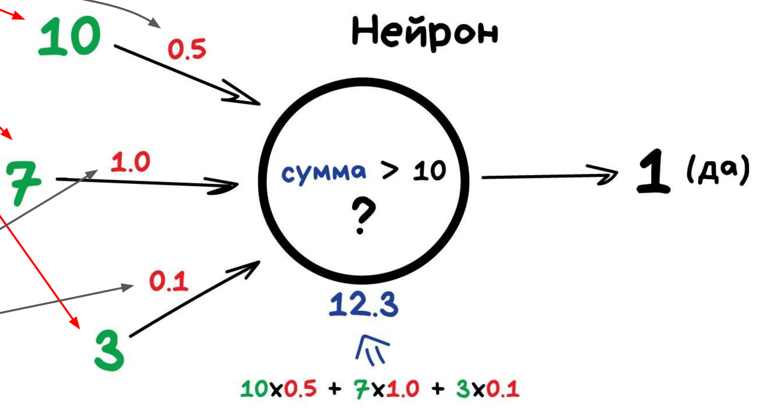
Модель нейрона

входы нейрона:
$x = (x_1, x_2, \ldots, x_n)$



Модель нейрона

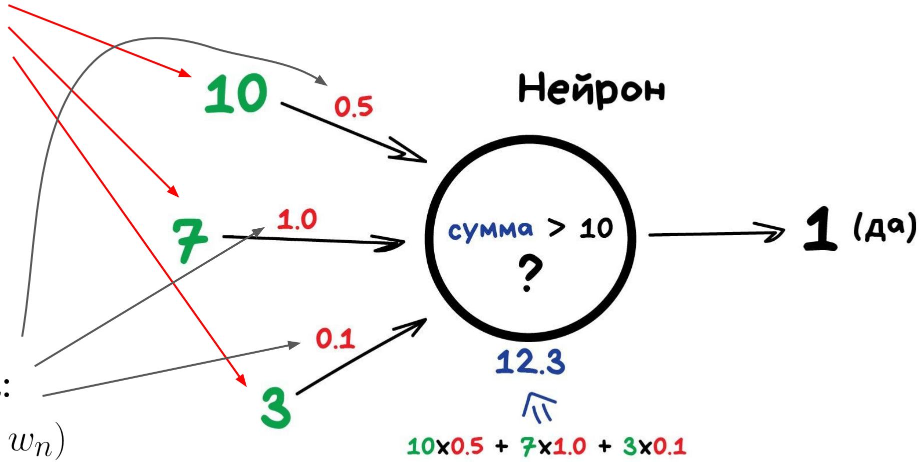
входы нейрона:
$x = (x_1, x_2, \ldots, x_n)$



веса нейрона:
$w = (w_1, w_2, \ldots, w_n)$

Модель нейрона

входы нейрона:
$x = (x_1, x_2, \ldots, x_n)$



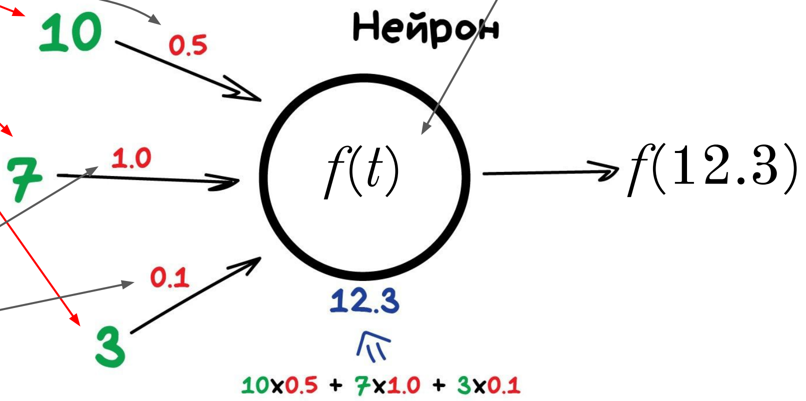
веса нейрона:
$w = (w_1, w_2, \ldots, w_n)$

скалярное произведение векторов x, w:
$$w_1 \cdot x_1 + w_2 \cdot x_2 + \ldots + w_n \cdot x_n = \langle w, x \rangle$$

Модель нейрона

функция активации

входы нейрона:
$x = (x_1, x_2, \ldots, x_n)$

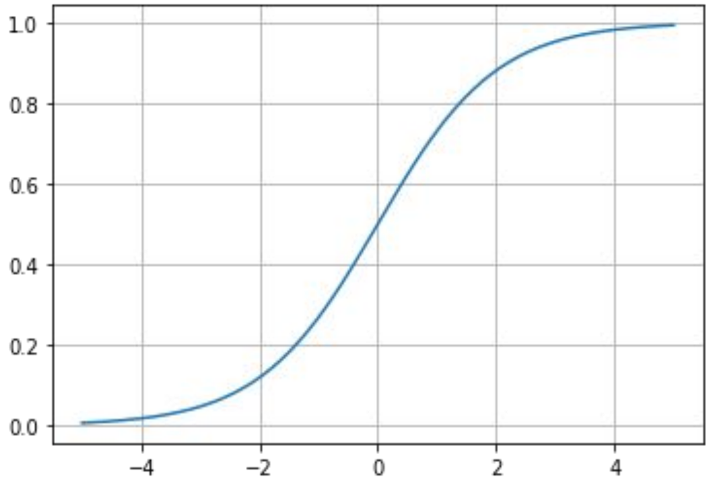


веса нейрона:
$w = (w_1, w_2, \ldots, w_n)$

скалярное произведение векторов x, w:
$$w_1 \cdot x_1 + w_2 \cdot x_2 + \ldots + w_n \cdot x_n = \langle w, x \rangle$$

Функция сигмоиды

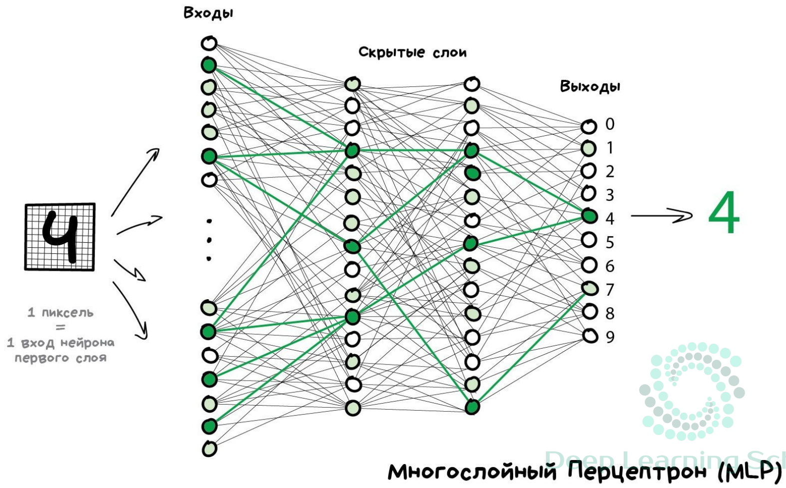
$$\sigma(t) = \frac{1}{1 + e^{-t}}$$



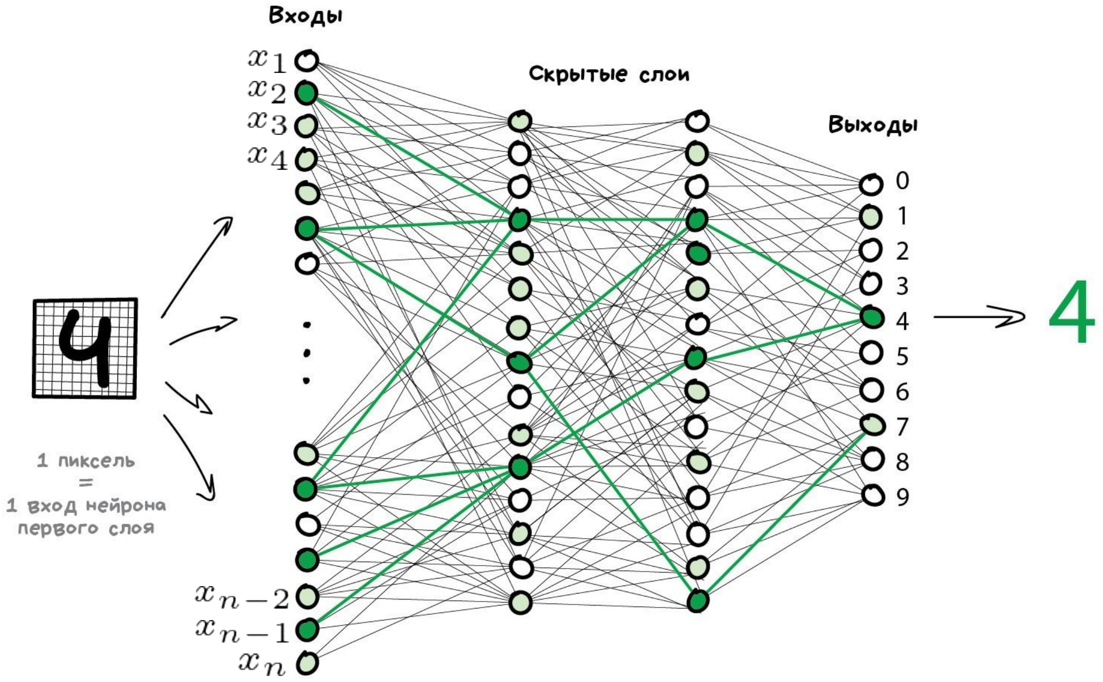
Многослойный перцептрон

Многослойный перцептрон

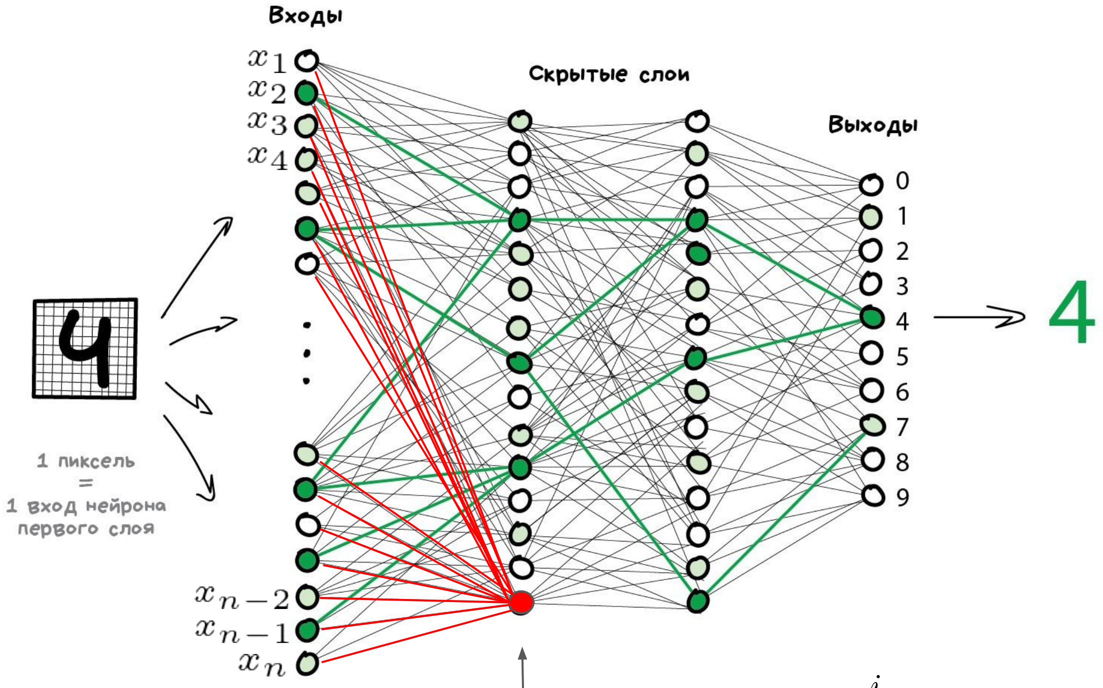
- Многослойный перцептрон — простейшая архитектура нейронной сети
- Каждый слой нейронов связан со всем нейронами с предыдущего слоя
- Десять выходных нейронов соответствуют классам изображений



Многослойный перцептрон



Многослойный перцептрон

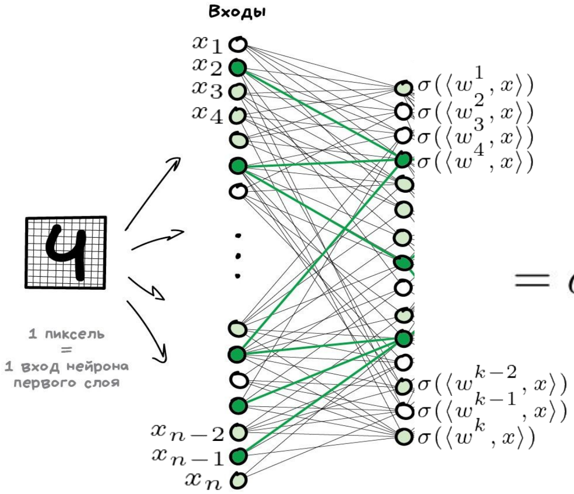


$$y_i = \sigma(\langle w^i, x \rangle + b_i)$$

w^i — вектор весов i-ого нейрона

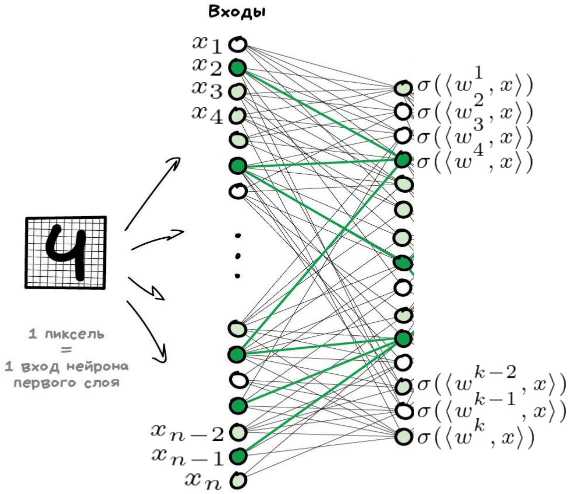
b^i — свободный член

Преобразование вектора в перцептроне



$$= \sigma \begin{pmatrix} \langle w^1, x\rangle + b_1 \\ \langle w^2, x\rangle + b_2 \\ \dots \\ \langle w^k, x\rangle + b_k \end{pmatrix} = \sigma(Wx + b)$$

Преобразование вектора в перцептроне



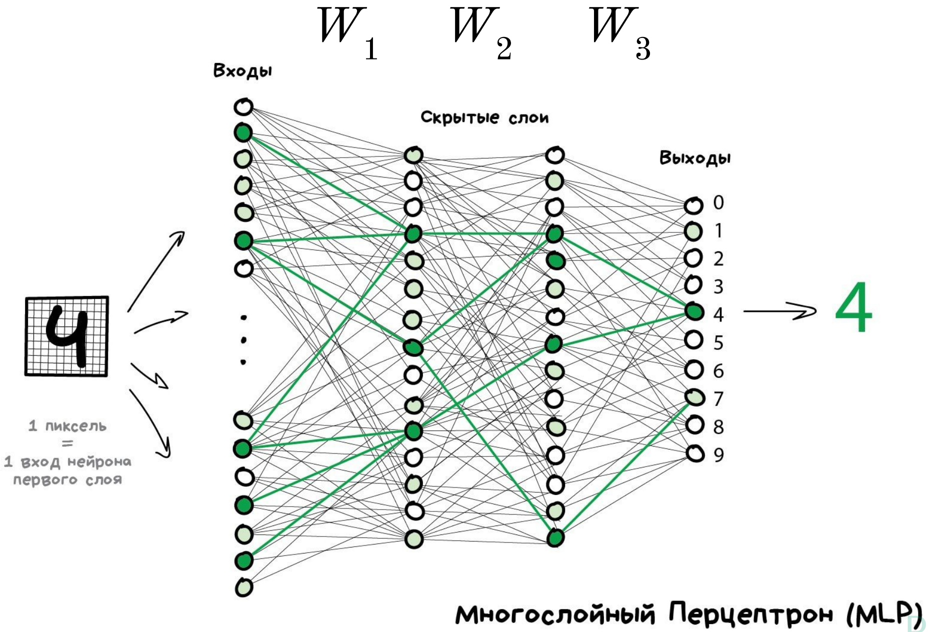
$$= \sigma(Wx + b)$$

линейное преобразование вектора x

$$W = \begin{pmatrix} w_1^1 & w_2^1 & \dots & w_n^1 \\ w_1^2 & w_2^2 & \dots & w_n^2 \\ \vdots & \ddots & \ddots & \vdots \\ w_1^k & w_2^k & \dots & w_n^k \end{pmatrix} = \begin{pmatrix} w^1 \\ w^2 \\ \dots \\ w^k \end{pmatrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \qquad b = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}$$

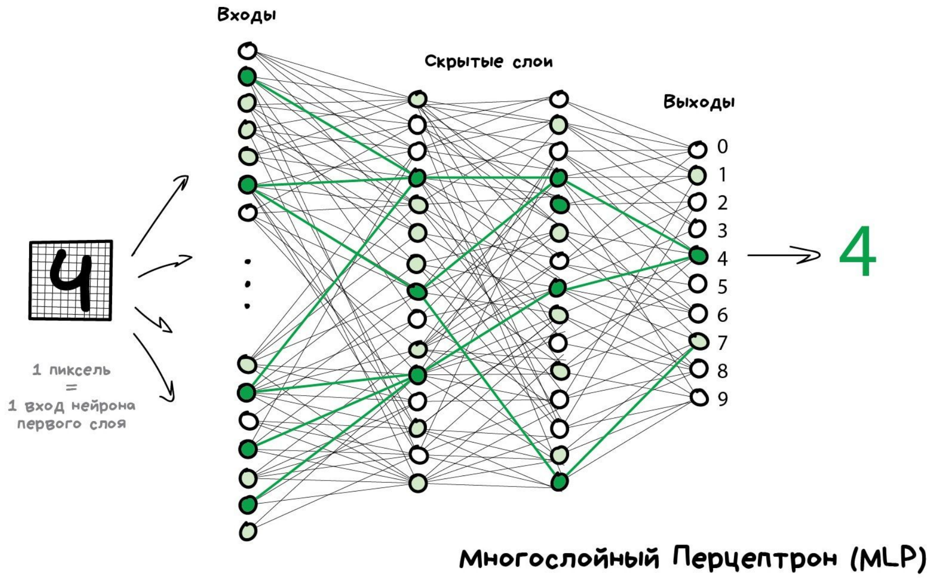
Параметры нейронной сети

- $(W_1, W_2, W_3, b_1, b_2, b_3)$ — совокупность параметров нейронной сети



Последний слой в задаче классификации

- Что происходит на выходном слое перцептрона?
- Как выходы нейронов преобразуются в вероятности классов?



Последний слой в задаче классификации

- Решаем задачу классификации на 10 классов
- Предпоследний слой содержит 10 линейных нейронов — меры принадлежности объекта каждому классу
- Пусть $y_0, y_1, \ldots, y_9$ — выходы предпоследнего слоя

Последний слой в задаче классификации

- Решаем задачу классификации на 10 классов
- Предпоследний слой содержит 10 линейных нейронов — меры принадлежности объекта каждому классу
- Пусть $y_0, y_1, \ldots, y_9$ — выходы предпоследнего слоя
- На последнем слое применяется softmax-преобразование:

Deep Learning School

Последний слой в задаче классификации

- Решаем задачу классификации на 10 классов
- Предпоследний слой содержит 10 линейных нейронов — меры принадлежности объекта каждому классу
- Пусть $y_0, y_1, \dots, y_9$ — выходы предпоследнего слоя
- На последнем слое применяется softmax-преобразование:

$$y_0, y_1, \dots, y_9$$

Последний слой в задаче классификации

- Решаем задачу классификации на 10 классов
- Предпоследний слой содержит 10 линейных нейронов — меры принадлежности объекта каждому классу
- Пусть $y_0, y_1, \ldots, y_9$ — выходы предпоследнего слоя
- На последнем слое применяется softmax-преобразование:

$$y_0, y_1, \ldots, y_9$$
$$\Downarrow$$
$$e^{y_0}, e^{y_1}, \ldots, e^{y_9}$$

Последний слой в задаче классификации

- Решаем задачу классификации на 10 классов
- Предпоследний слой содержит 10 линейных нейронов — меры принадлежности объекта каждому классу
- Пусть $y_0, y_1, \ldots, y_9$ — выходы предпоследнего слоя
- На последнем слое применяется softmax-преобразование:

$$y_0, y_1, \ldots, y_9$$

$$\Downarrow$$

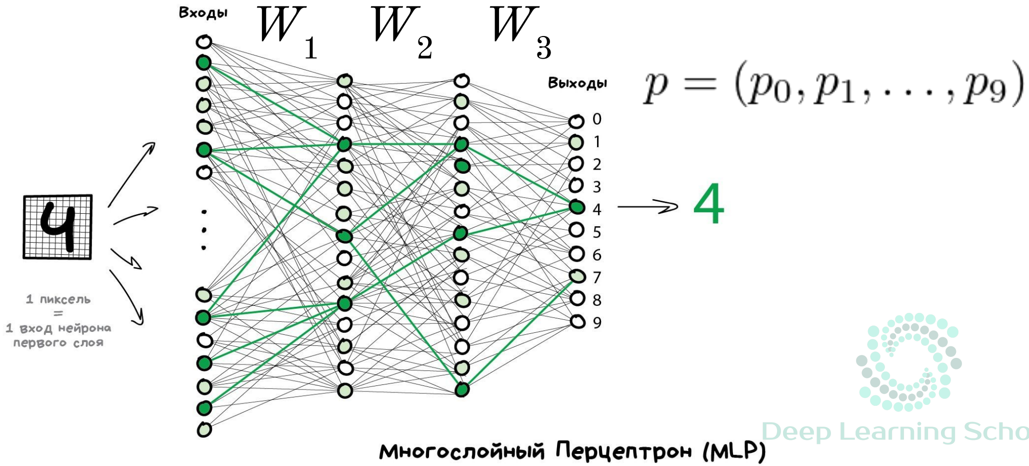
$$e^{y_0}, e^{y_1}, \ldots, e^{y_9}$$

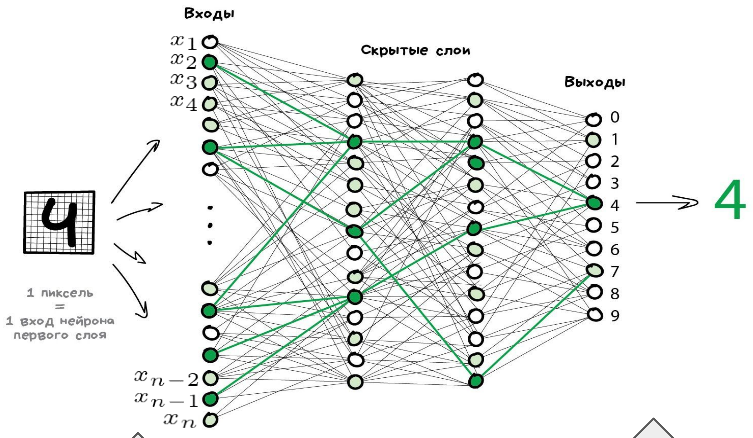
$$\Downarrow$$

$$\frac{e^{y_0}}{e^{y_0}+e^{y_1}+\ldots+e^{y_9}}, \frac{e^{y_1}}{e^{y_0}+e^{y_1}+\ldots+e^{y_9}}, \ldots, \frac{e^{y_9}}{e^{y_0}+e^{y_1}+\ldots+e^{y_9}}$$

Последний слой в задаче классификации

- Числа $\frac{e^{y_0}}{e^{y_0}+e^{y_1}+...+e^{y_9}}, \frac{e^{y_1}}{e^{y_0}+e^{y_1}+...+e^{y_9}}, \dots, \frac{e^{y_9}}{e^{y_0}+e^{y_1}+...+e^{y_9}}$:
  - Положительны
  - В сумме дают 1
- Их можно интерпретировать как *вероятности принадлежности соответствующим классам*





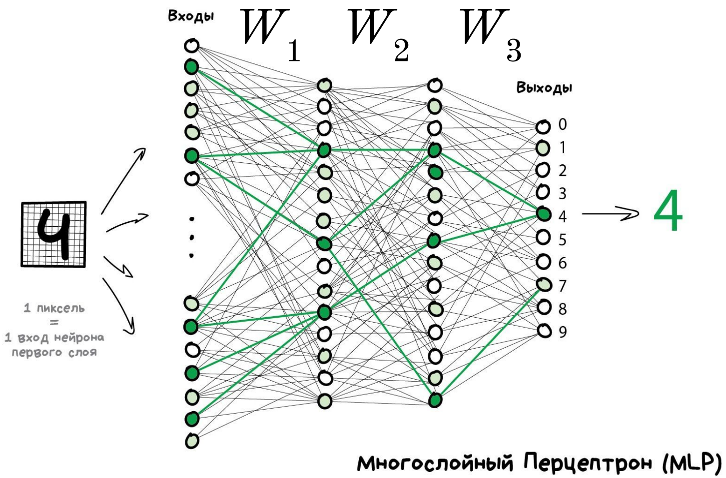
Описание модели

Обучение модели

Обучение MLP (Multilayer Perceptron) для задачи классификации

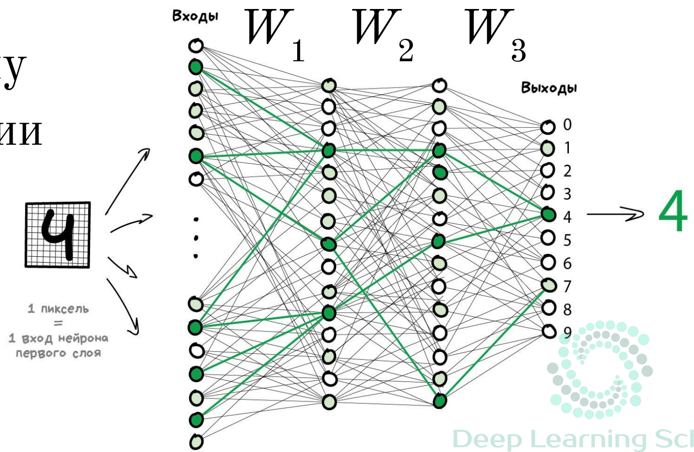
Обучение перцептрона

- Проходимся по элементам обучающей выборки
- Имеется *размеченное* изображение цифры “4”
- На последнем слое нейронной сети сформировались вероятности $p = (p_0, p_1, \ldots, p_9)$



Обучение перцептрона

- Хотим сделать p_4 *как можно больше*
- Аналогично для всех картинок
- **Цель:** подобрать матрицы W_1, W_2, W_3 так, чтобы *максимизировать произведение вероятностей правильной классификации по всем элементам обучающей выборки*
- Имеем сложную задачу численной оптимизации



Многослойный Перцептрон (MLP)

Deep Learning School

Оптимизация функции потерь

- Пусть (x, y) — элемент обучающей выборки, θ — вектор параметров нейросети

Оптимизация функции потерь

- Пусть (x, y) — элемент обучающей выборки, θ — вектор параметров нейросети
- На последнем слое нейронной сети — вероятности классов

$$p = (p_0, p_1, \dots, p_9)^\top$$

Оптимизация функции потерь

- Пусть (x, y) — элемент обучающей выборки, θ — вектор параметров нейросети
- На последнем слое нейронной сети — вероятности классов

$$p = (p_0, p_1, \ldots, p_9)^\top$$

- Как и в случае логистической регрессии, оптимизируем $\mathrm{logloss}(y, p)$:

$$\log p_y(x) \to \max$$

$$-\log p_y(x) \to \min$$

Оптимизация функции потерь

- Пусть (x, y) — элемент обучающей выборки, θ — вектор параметров нейросети
- На последнем слое нейронной сети — вероятности классов

$$p = (p_0, p_1, \ldots, p_9)^\top$$

- Как и в случае логистической регрессии, оптимизируем $\mathrm{logloss}(y, p)$:

$$\log p_y(x) \to \max$$

$$-\log p_y(x) \to \min$$

- Итоговая задача оптимизации:

$$\frac{1}{\ell} \sum_{(x,y) \in X_{train}} logloss(y, p(x, \theta)) \to \min_{\theta}$$

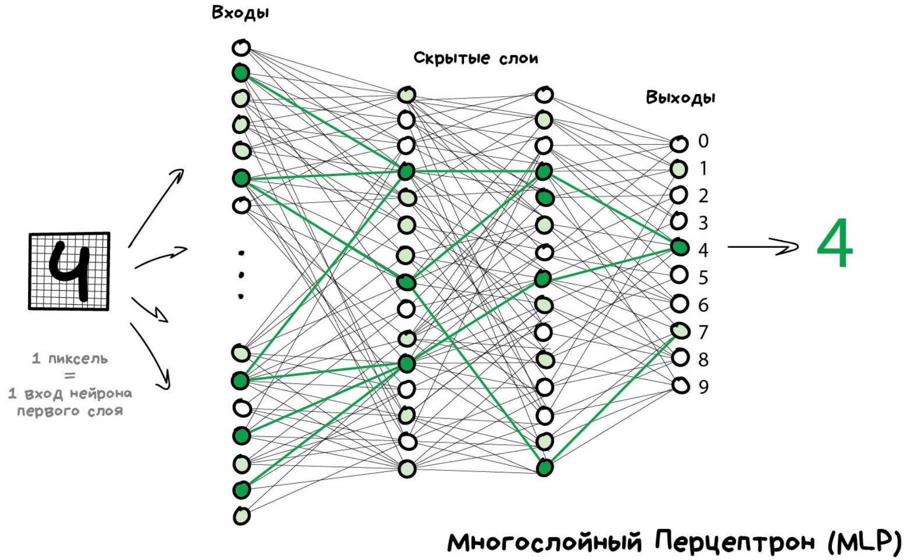
Оптимизация в общем случае

Можно оптимизировать и другие функции потерь.

Например, Mean Squared Error в случае задачи регрессии

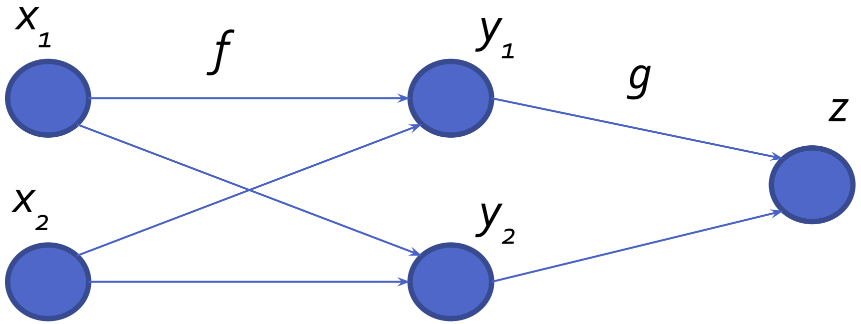
Стохастический градиентный спуск

- Выбираем батч примеров из обучающей выборки
- Вычисляем производную функции потерь по всем весам нейросети
- Обновляем веса в направлении антиградиента



Алгоритм обратного распространения ошибки (BackProp)

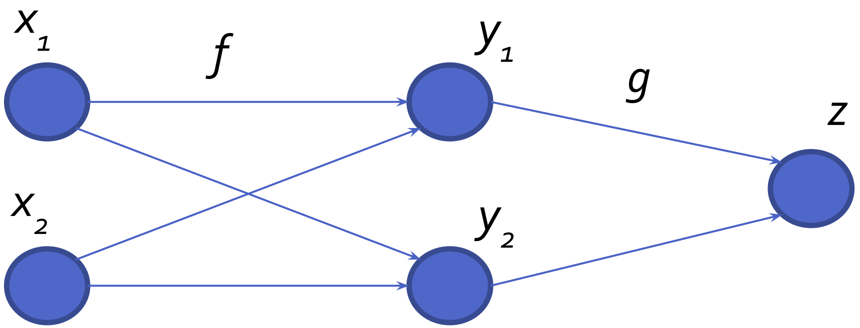
Производная композиции



$$\frac{dz}{dy} = \begin{bmatrix} \frac{\partial z}{\partial y_1}(y_1, y_2) \\ \\ \frac{\partial z}{\partial y_2}(y_1, y_2) \end{bmatrix}$$

$$\frac{\partial z}{\partial x_1} = \frac{\partial z}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_1} + \frac{\partial z}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_1}$$

Производная композиции



$$\frac{\partial z}{\partial x_1} = \frac{\partial z}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_1} + \frac{\partial z}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_1}$$

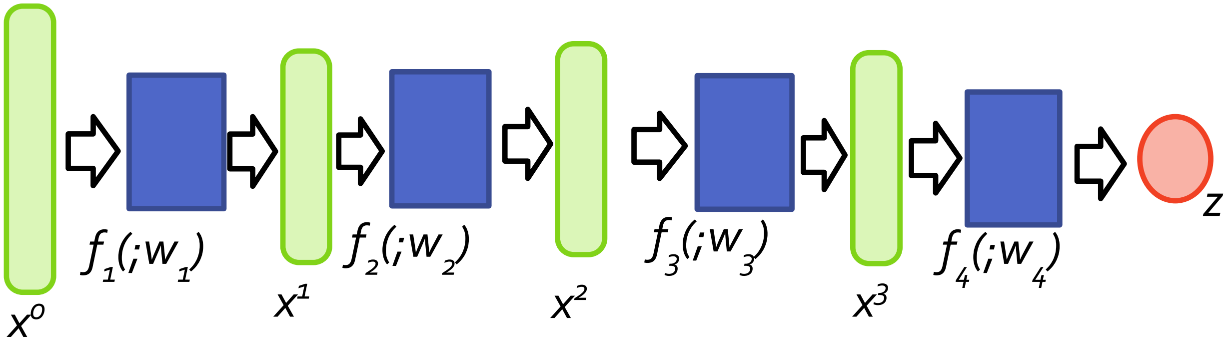
$$\frac{\partial z}{\partial x_2} = \frac{\partial z}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_2} + \frac{\partial z}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_2}$$

$$\frac{dz}{dx} = \begin{bmatrix} \frac{\partial z}{\partial x_1} \\ \\ \frac{\partial z}{\partial x_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} \\ \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial z}{\partial y_1} \\ \\ \frac{\partial z}{\partial y_2} \end{bmatrix}$$

$$\frac{dz}{dx} = \left(\frac{dy}{dx}\right)^T \frac{dz}{dy}$$

слайд В. Лемпицкого

Вычисление глубоких производных



$$z = f_4(f_3(f_2(f_1(x; w_1); w_2); w_3); w_4)$$

слайд В. Лемпицкого

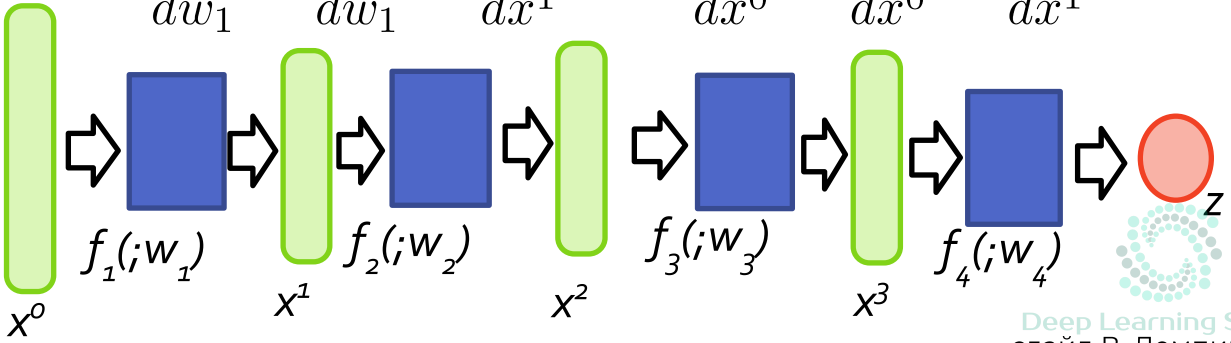
Вычисление глубоких производных

$\frac{dz}{dx^3}, \frac{dz}{dw_4}$ можно вычислить

$$\frac{dz}{dw_3} = \frac{dx^3}{dw_3}^T \cdot \frac{dz}{dx^3} \qquad \frac{dz}{dx^2} = \frac{dx^3}{dx^2}^T \cdot \frac{dz}{dx^3}$$

$$\frac{dz}{dw_2} = \frac{dx^2}{dw_2}^T \cdot \frac{dz}{dx^2} \qquad \frac{dz}{dx^1} = \frac{dx^2}{dx^1}^T \cdot \frac{dz}{dx^2}$$

$$\frac{dz}{dw_1} = \frac{dx^1}{dw_1}^T \cdot \frac{dz}{dx^1} \qquad \frac{dz}{dx^0} = \frac{dx^1}{dx^0}^T \cdot \frac{dz}{dx^1}$$



слайд В. Лемпицкого

Слой нейронной сети

Чтобы определить слой, необходимо задать:

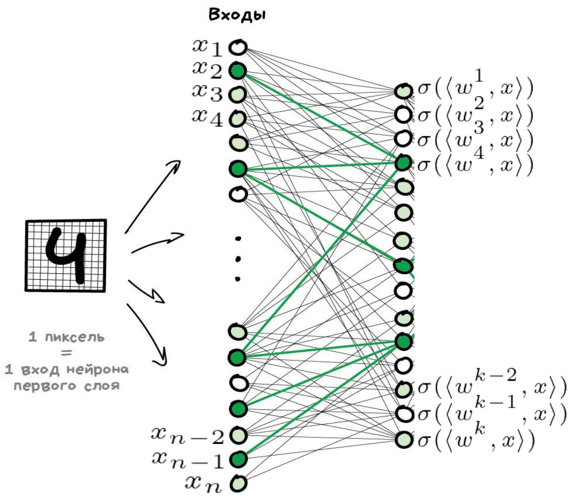
- forward performance: $y = f(x; w)$
- backward performance: $z(x) = z(f(x;\ w))$

В случае, если слой реализует простую функцию, то для backward пользуемся правилом

$$\frac{dz}{dx} = \left(\frac{dy}{dx}\right)^T \frac{dz}{dy}$$

Back propagation через линейный слой

Back propagation через линейный слой



линейный слой + поэлементная сигмоида

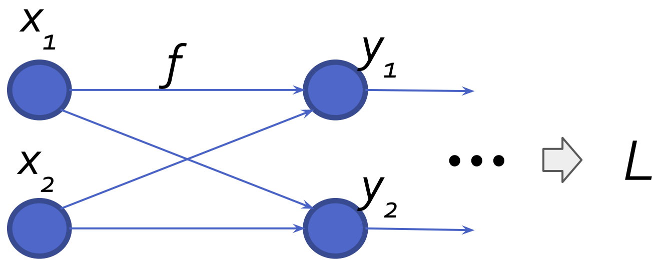
Back propagation через линейный слой

$$y = f_W(x) = Wx$$

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

$$y_1 = w_{11}x_1 + w_{12}x_2$$

$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$

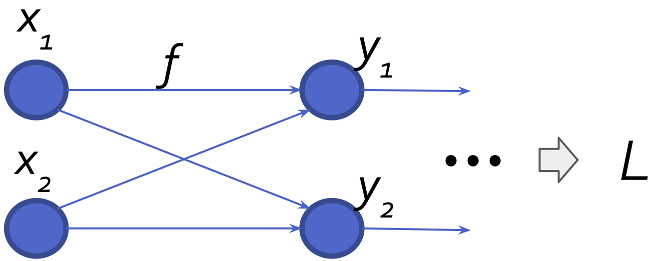
Back propagation через линейный слой

$$y = f_W(x) = Wx$$

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

$$y_1 = w_{11}x_1 + w_{12}x_2$$

$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

Deep Learning School

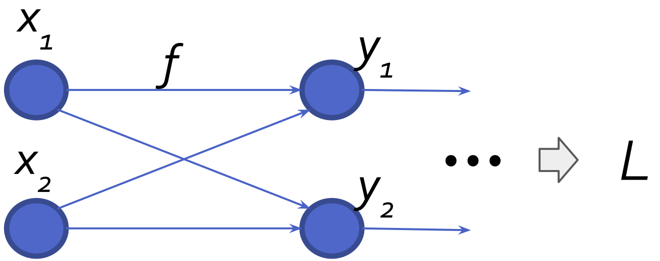
Back propagation через линейный слой

$$y = f_W(x) = Wx$$

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

$$y_1 = w_{11}x_1 + w_{12}x_2$$

$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

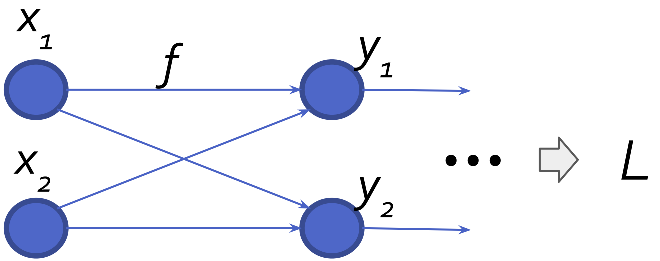
Считаем промежуточные производные:

$$\frac{\partial y_1}{\partial x_1} = w_{11}, \quad \frac{\partial y_2}{\partial x_1} = w_{21}$$

$$\frac{\partial y_1}{\partial x_2} = w_{12}, \quad \frac{\partial y_2}{\partial x_2} = w_{22}$$

Back propagation через линейный слой

$$y = f_W(x) = Wx$$
$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$
$$y_1 = w_{11}x_1 + w_{12}x_2$$
$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

Считаем промежуточные производные:

$$\frac{\partial y_1}{\partial x_1} = w_{11}, \quad \frac{\partial y_2}{\partial x_1} = w_{21}$$
$$\frac{\partial y_1}{\partial x_2} = w_{12}, \quad \frac{\partial y_2}{\partial x_2} = w_{22}$$

Записываем правило производной композиции:

$$\frac{\partial L}{\partial x_1} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_1} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_1} = \frac{\partial L}{\partial y_1} w_{11} + \frac{\partial L}{\partial y_2} w_{21}$$
$$\frac{\partial L}{\partial x_2} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_2} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_2} = \frac{\partial L}{\partial y_1} w_{12} + \frac{\partial L}{\partial y_2} w_{22}$$

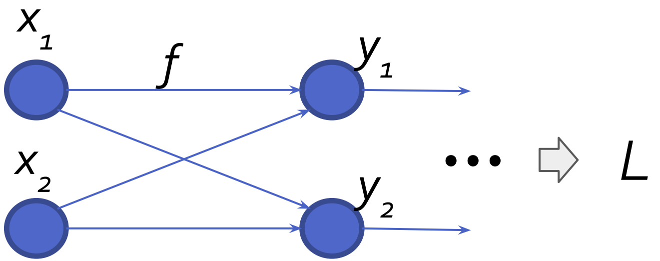
Back propagation через линейный слой

$$y = f_W(x) = Wx$$

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

$$y_1 = w_{11}x_1 + w_{12}x_2$$

$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

Считаем промежуточные производные:

$$\frac{\partial y_1}{\partial x_1} = w_{11}, \quad \frac{\partial y_2}{\partial x_1} = w_{21}$$

$$\frac{\partial y_1}{\partial x_2} = w_{12}, \quad \frac{\partial y_2}{\partial x_2} = w_{22}$$

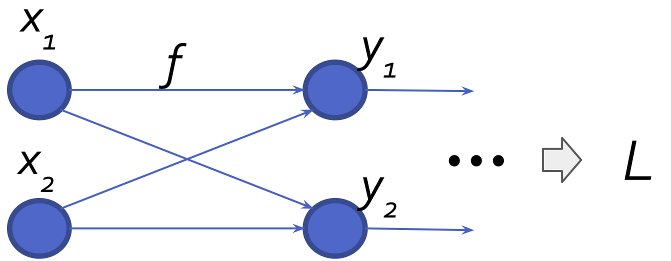
Записываем правило производной композиции:

$$\frac{\partial L}{\partial x} \begin{cases} \boxed{\frac{\partial L}{\partial x_1}} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_1} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_1} = \frac{\partial L}{\partial y_1} w_{11} + \frac{\partial L}{\partial y_2} w_{21} \\ \boxed{\frac{\partial L}{\partial x_2}} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_2} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_2} = \frac{\partial L}{\partial y_1} w_{12} + \frac{\partial L}{\partial y_2} w_{22} \end{cases}$$

Deep Learning School

Back propagation через линейный слой

$$y = f_W(x) = Wx$$
$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$
$$y_1 = w_{11}x_1 + w_{12}x_2$$
$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

Считаем промежуточные производные:

$$\frac{\partial y_1}{\partial x_1} = w_{11}, \ \frac{\partial y_2}{\partial x_1} = w_{21}$$
$$\frac{\partial y_1}{\partial x_2} = w_{12}, \ \frac{\partial y_2}{\partial x_2} = w_{22}$$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

Записываем правило производной композиции:

$$\frac{\partial L}{\partial x} \begin{cases} \boxed{\frac{\partial L}{\partial x_1}} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_1} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_1} = \frac{\partial L}{\partial y_1} w_{11} + \frac{\partial L}{\partial y_2} w_{21} \\ \boxed{\frac{\partial L}{\partial x_2}} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial x_2} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial x_2} = \frac{\partial L}{\partial y_1} w_{12} + \frac{\partial L}{\partial y_2} w_{22} \end{cases} = \begin{pmatrix} w_{11} & w_{22} \\ w_{12} & w_{22} \end{pmatrix} \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix} = W^\top \frac{dL}{dy}$$

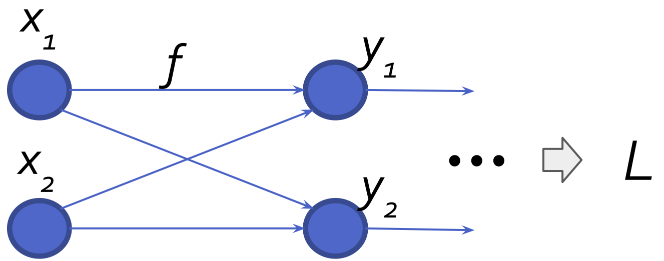
Back propagation через линейный слой

$$y = f_W(x) = Wx$$

$$W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

$$y_1 = w_{11}x_1 + w_{12}x_2$$

$$y_2 = w_{21}x_1 + w_{22}x_2$$



Знаем: $\frac{dL}{dy} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix}$ узнать: $\frac{dL}{dx} = \begin{pmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \end{pmatrix}$

Считаем промежуточные производные:

$$\frac{\partial y_1}{\partial x_1} = w_{11}, \quad \frac{\partial y_2}{\partial x_1} = w_{21}$$

$$\frac{\partial y_1}{\partial x_2} = w_{12}, \quad \frac{\partial y_2}{\partial x_2} = w_{22}$$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial w_{11}} & \frac{\partial L}{\partial w_{12}} \\ \frac{\partial L}{\partial w_{21}} & \frac{\partial L}{\partial w_{22}} \end{pmatrix}$$

Наконец, считаем производную по матрице W

$$\frac{\partial L}{\partial w_{ij}} = \frac{\partial L}{\partial y_1} \cdot \frac{\partial y_1}{\partial w_{ij}} + \frac{\partial L}{\partial y_2} \cdot \frac{\partial y_2}{\partial w_{ij}} = \frac{\partial L}{\partial y_i} \cdot x_j$$

$$\frac{dL}{dW} = \begin{pmatrix} \frac{\partial L}{\partial y_1} \\ \frac{\partial L}{\partial y_2} \end{pmatrix} \cdot (x_1, x_2) = \frac{dL}{dy} \cdot (x_1, x_2)$$

Реализация полносвязного слоя

```
class Linear(Module):
    """
    A module which applies a linear transformation
    A common name is fully-connected layer

    The module should work with 2D input of shape (n_samples, n_feature).
    """

    def forward(self, input):
        self.output = np.dot(input, self.W.T) + self.b
        return self.output

    def updateGradInput(self, input, gradOutput):
        self.gradInput = np.dot(gradOutput, self.W)

        self.gradW = np.dot(gradOutput.T, input)
        self.gradb = np.sum(gradOutput, axis=0)

        return self.gradInput
```

Вопрос. Как устроен back propagation через слой сигмоиды?

$$\frac{dz}{dx} = \left(\frac{dy}{dx}\right)^T \frac{dz}{dy}$$

Recap

- Модель нейрона
- Полносвязная нейронная сеть для классификации
  - Скрытые слои
  - Последний слой
- Обучение нейронных сетей
  - Loss function
  - Back Propagation
- Разбор BackProp для полносвязного слоя