

Глубокие нейронные сети

# Градиентный спуск (Gradient descent)

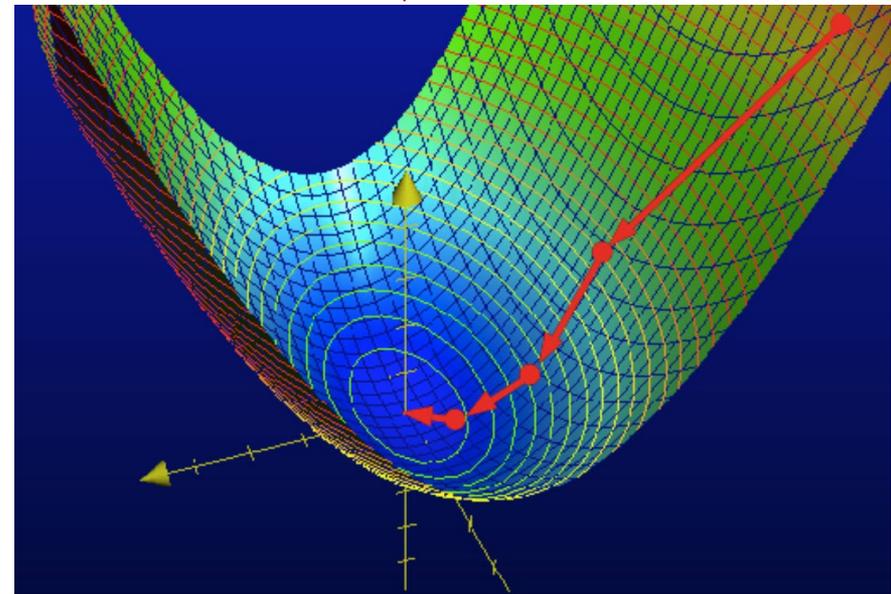
Лекция 3

# Метод градиентного спуска (двумерный случай)

- Найти минимум функции  $f(x_1, x_2): \mathbb{R}^2 \rightarrow \mathbb{R}$
- Градиент:

$$\nabla f = \left( \frac{\partial f(x_1, x_2)}{\partial x_1}; \frac{\partial f(x_1, x_2)}{\partial x_2} \right)$$

- $\Delta x := -\eta \nabla f$ ,  
где  $\eta > 0$  – малый  
параметр (*скорость  
обучения*)



---

$\nabla$  – СИМВОЛ «НАБЛА»

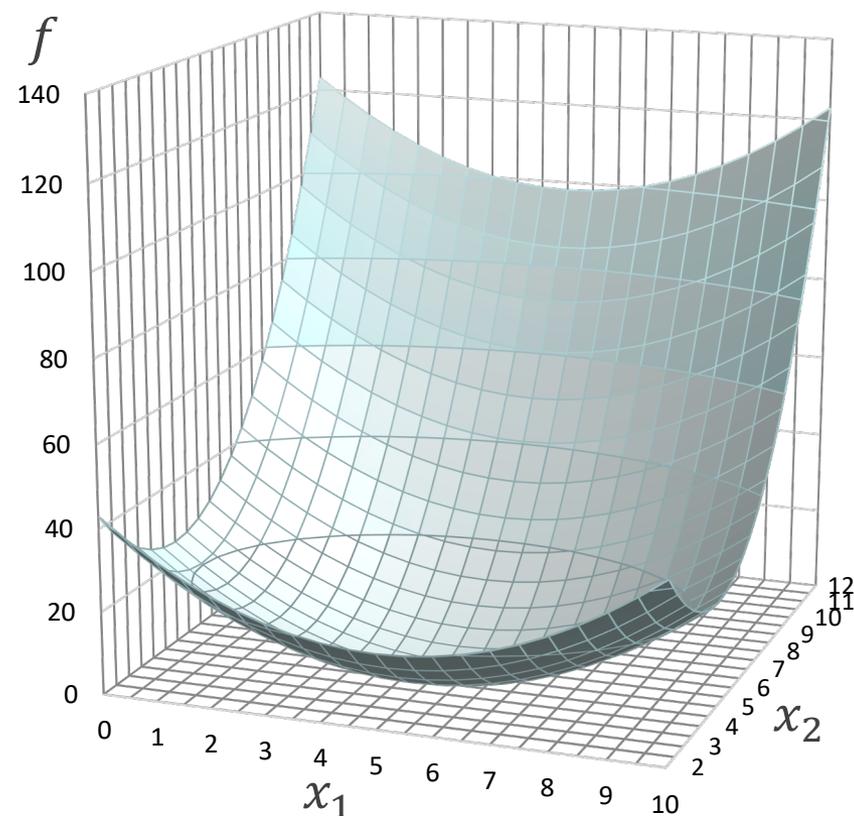
# Пример вычисления градиента

$$f(x_1, x_2) = x_1^2 - 10x_1 + 2x_2^2 - 20x_2 + 75$$

$$\frac{\partial f(x_1, x_2)}{\partial x_1} = ?$$

$$\frac{\partial f(x_1, x_2)}{\partial x_2} = ?$$

$$\nabla f = ?$$



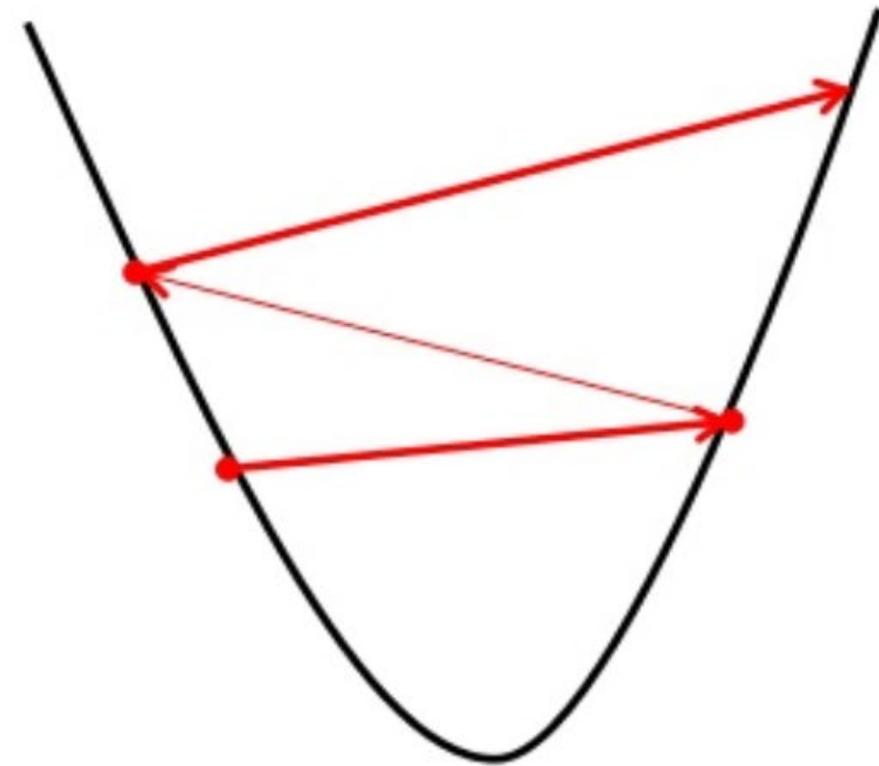
# Нахождение минимума функции $f$ методом градиентного спуска

$$\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}, \mathbf{x}^{(n+1)} \dots$$

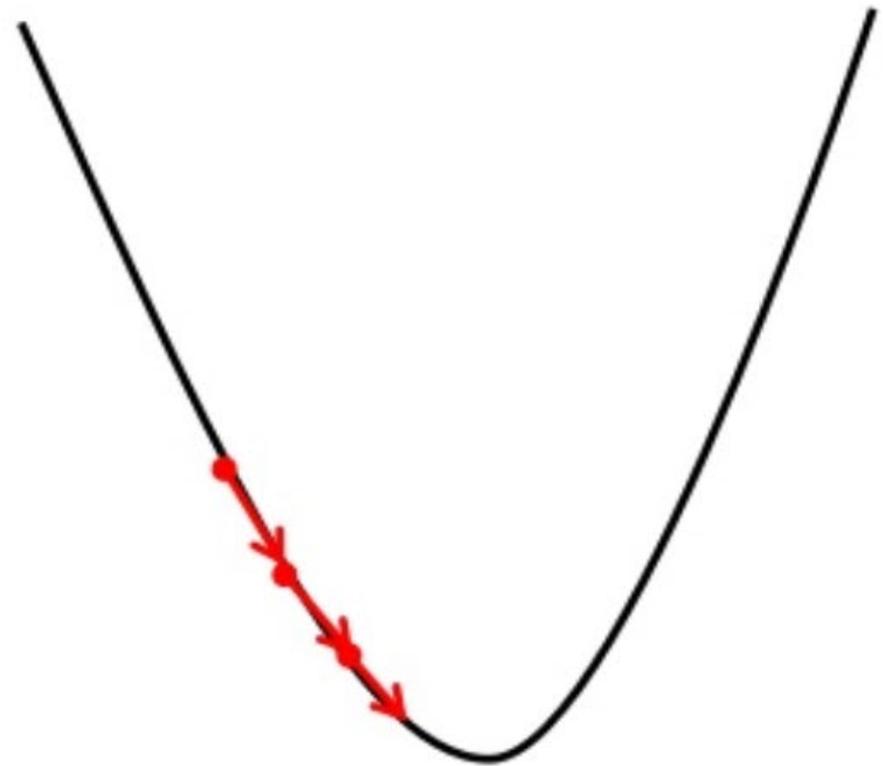
$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \eta \nabla f(\mathbf{x}^{(n)})$$

$\eta > 0$  – *скорость обучения*

# Выбор скорости обучения $\eta$



Большая скорость обучения



Малая скорость обучения

# Высокая скорость обучения предотвращает остановку в локальном минимуме



# Обучение нейронной сети методом градиентного спуска

Необходимо минимизировать  
среднеквадратичную ошибку:

$$\mathbb{C} = \frac{1}{|V|} \sum_{(x,y) \in V} \frac{\|\alpha(x) - y\|^2}{2}$$

$$\mathbf{w} := \mathbf{w} - \eta \nabla_{\mathbf{w}} \mathbb{C}$$

$$\mathbf{b} := \mathbf{b} - \eta \nabla_{\mathbf{b}} \mathbb{C}$$

© Соколинский Л.Б. Глубокие нейронные сети 05.11.2022

## Градиенты

$$\nabla_{\mathbf{w}} \mathbb{C} = \left( \frac{\partial \mathbb{C}}{\partial w_1}, \dots, \frac{\partial \mathbb{C}}{\partial w_Q} \right)$$

$$\nabla_{\mathbf{b}} \mathbb{C} = \left( \frac{\partial \mathbb{C}}{\partial b_1}, \dots, \frac{\partial \mathbb{C}}{\partial b_P} \right)$$

3. Градиентный спуск 22

© Соколинский Л.Б. Глубокие нейронные сети 24.09.2022

## Модель обучения

- $\mathbf{x} \in \mathbb{R}^{28 \times 28}$  – изображение  $28 \times 28$ , подаваемое на вход нейронной сети
- $\mathbf{y} \in \mathbb{R}^{10}$  – правильный ответ
- $\alpha(\mathbf{x}) \in \mathbb{R}^{10}$  – ответ, выдаваемый нейронной сетью для входного изображения  $\mathbf{x}$
- $V = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \mid 1 \leq i \leq K\}$  – обучающая выборка,  $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$  – прецедент
- $\mathbf{w} \in \mathbb{R}^Q$  – вектор, содержащий все синаптические веса сети
- $\mathbf{b} \in \mathbb{R}^P$  – вектор, содержащий все смещения
- $\mathbb{C} = \frac{1}{|V|} \sum_{(x,y) \in V} \frac{\|\alpha(x) - y\|^2}{2}$  – функция потерь: среднеквадратическая ошибка
- Необходимо минимизировать среднеквадратическую ошибку путем подбора значений  $\mathbf{w}$  и  $\mathbf{b}$

2. Модель нейронной сети 34

# Проблема

Высокая вычислительная сложность процесса обучения

The diagram illustrates the computational complexity of the training process. It features three callout boxes with dashed blue borders:

- Top-left: 60 000 прецедентов (precedents)
- Top-right: 10 000 000 ВЕСОВ (weights)
- Bottom-right: 100 000 смещений (biases)

The cost function is defined as:
$$\mathbb{C} = \frac{1}{|V|} \sum_{(x,y) \in V} \frac{\|\alpha(x) - y\|^2}{2}$$

The weight update rule is:
$$\mathbf{w} := \mathbf{w} - \eta \nabla_{\mathbf{w}} \mathbb{C}$$

The bias update rule is:
$$\mathbf{b} := \mathbf{b} - \eta \nabla_{\mathbf{b}} \mathbb{C}$$

Dashed blue arrows point from the callout boxes to the corresponding terms in the equations: from the 60,000 precedents box to the denominator |V|, from the 10,000,000 weights box to the summation index (x,y) in V, and from the 100,000 biases box to the bias update equation.

# Переход к векторно-матричным операциям

$m$  – количество нейронов в слое  $l$

$n$  – количество нейронов в слое  $l-1$

$$W^{(l)} = \begin{bmatrix} w_{11}^{(l)} & \cdots & w_{1m}^{(l)} \\ \vdots & \ddots & \vdots \\ w_{n1}^{(l)} & \cdots & w_{nm}^{(l)} \end{bmatrix}$$

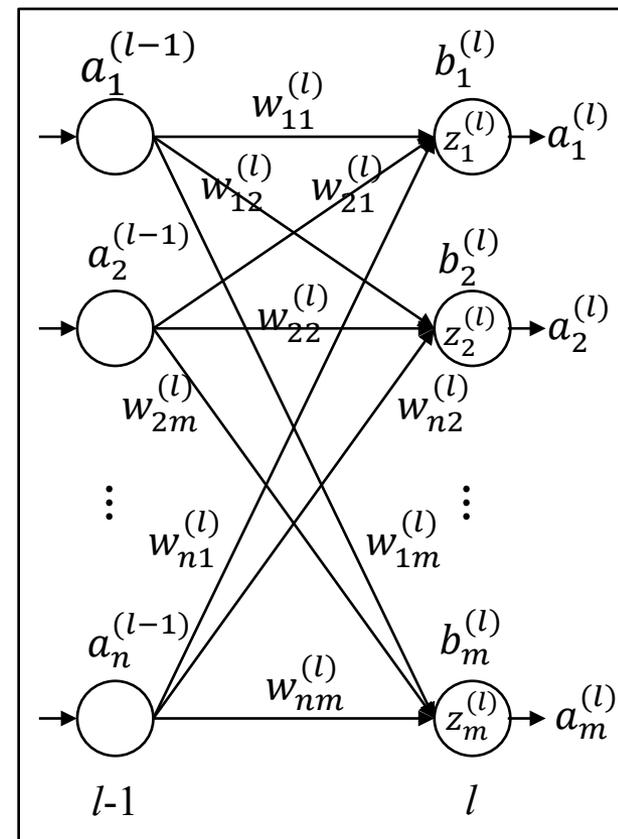
$$\mathbf{b}^{(l)} = [b_1^{(l)} \quad \cdots \quad b_m^{(l)}]^\top \quad \mathbf{z}^{(l)} = [z_1^{(l)} \quad \cdots \quad z_m^{(l)}]^\top$$

$$\mathbf{a}^{(l-1)} = [a_1^{(l-1)} \quad \cdots \quad a_n^{(l-1)}]^\top \quad \mathbf{a}^{(l)} = [a_1^{(l)} \quad \cdots \quad a_m^{(l)}]^\top$$

$$\boldsymbol{\sigma}(\mathbf{z}^{(l)}) = \left[ \sigma(z_1^{(l)}) \quad \cdots \quad \sigma(z_m^{(l)}) \right]^\top$$

$$\mathbf{z}^{(l)} = W^{(l)\top} \mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}$$

$$\mathbf{a}^{(l)} = \boldsymbol{\sigma}(\mathbf{z}^{(l)})$$



# Преобразование формулы среднеквадратичной ошибки

Имеем

$$\mathbb{C} = \frac{1}{|V|} \sum_{(x,y) \in V} \frac{\|\alpha(x) - y\|^2}{2}$$

Обозначим

$$C_{(x,y)} = \frac{\|\alpha(x) - y\|^2}{2}$$

Тогда

$$\mathbb{C} = \frac{1}{|V|} \sum_{(x,y) \in V} C_{(x,y)}$$

Следовательно

$$\begin{aligned} \nabla_w \mathbb{C} &= \frac{1}{|V|} \sum_{(x,y) \in V} \nabla_w C_{(x,y)} \\ \nabla_b \mathbb{C} &= \frac{1}{|V|} \sum_{(x,y) \in V} \nabla_b C_{(x,y)} \end{aligned}$$

# Стохастический градиентный спуск (SGD)

1.  $\mathbf{w} := rnd; \mathbf{b} := rnd$  // Присваиваем случайные значения
2. **for**  $epoch = 1 \dots 10$  **do** -----
3.  $V \rightarrow V_1, \dots, V_M$  // Последовательно разбиваем  $V$  на подвыборки
4. **for**  $i = 1 \dots M$  **do** -----
5. 
$$\nabla_{\mathbf{w}} \mathbb{C}_{V_i} := \frac{1}{|V_i|} \sum_{(x,y) \in V_i} \nabla_{\mathbf{w}} \mathbb{C}(x,y)$$
6. 
$$\nabla_{\mathbf{b}} \mathbb{C}_{V_i} := \frac{1}{|V_i|} \sum_{(x,y) \in V_i} \nabla_{\mathbf{b}} \mathbb{C}(x,y)$$
7. 
$$\mathbf{w} := \mathbf{w} - \eta \nabla_{\mathbf{w}} \mathbb{C}_{V_i}$$
8. 
$$\mathbf{b} := \mathbf{b} - \eta \nabla_{\mathbf{b}} \mathbb{C}_{V_i}$$
9. **end for** -----
10.  $shuffle(V)$  // Перемешиваем  $V$
11. **end for** -----

Цикл по подвыборкам

Цикл по эпохам обучения

# Градиенты

$$\nabla_{\mathbf{w}} \mathbb{C} = \left( \frac{\partial \mathbb{C}}{\partial w_1}, \dots, \frac{\partial \mathbb{C}}{\partial w_Q} \right)$$

$$\nabla_{\mathbf{b}} \mathbb{C} = \left( \frac{\partial \mathbb{C}}{\partial b_1}, \dots, \frac{\partial \mathbb{C}}{\partial b_P} \right)$$

# Проблема

- Необходим эффективный алгоритм вычисления градиентов

$$\nabla_w C(x, y)$$

$$\nabla_b C(x, y)$$

# Конец лекции 3