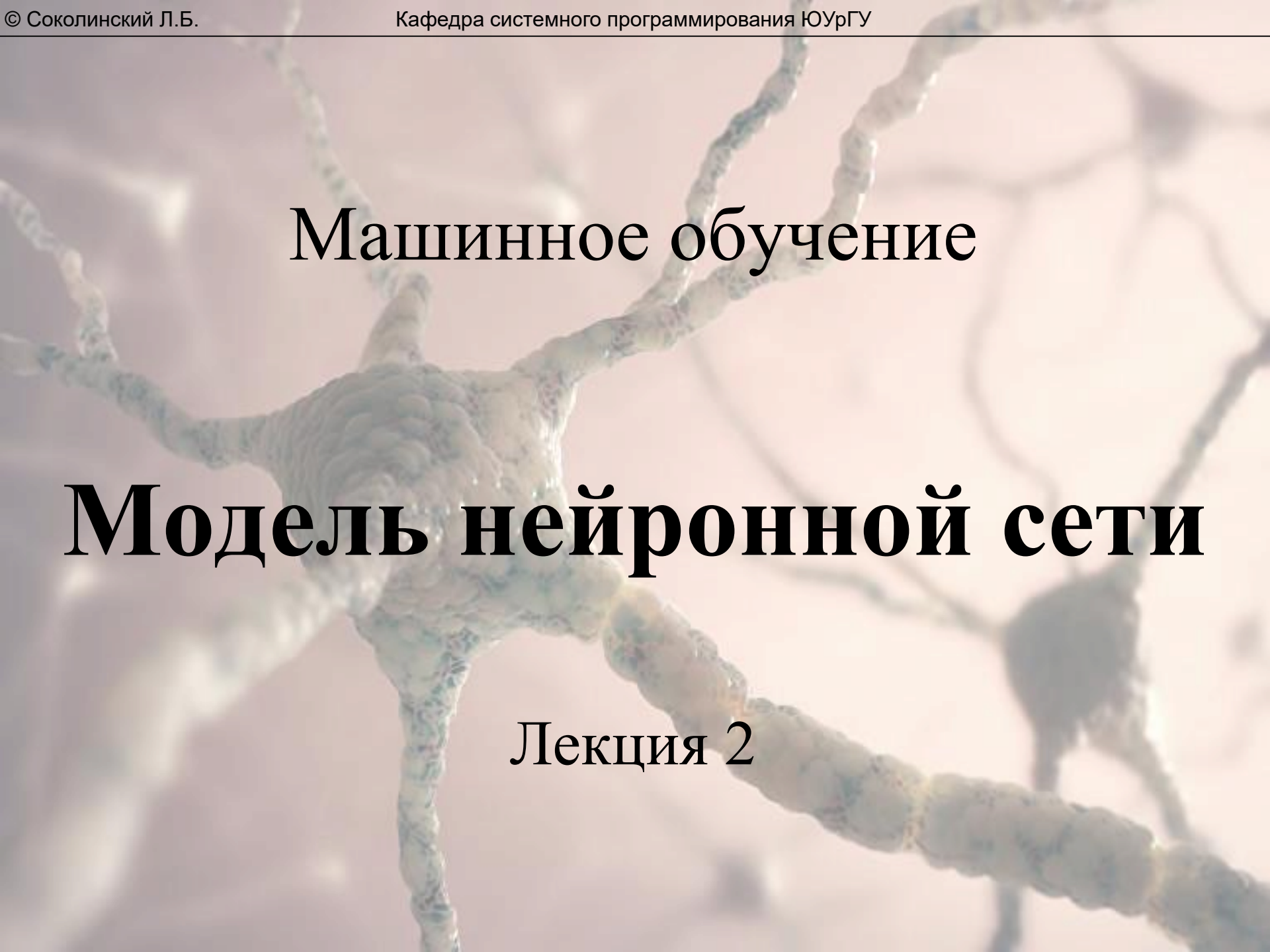


Новый спецкурс по машинному обучению для магистрантов каф. системного программирования

- Мотивация
 - Искусственные нейронные сети: ведущий тренд в ИТ на ближайшие 10-20 лет
 - Искусственные нейронные сети - основная платформа для систем искусственного интеллекта
 - Наличие мощного теоретического фундамента
 - Появление высокопроизводительных аппаратных платформ с GPU
 - Индустрия 4.0 базируется на кибер-физических системах с искусственным интеллектом
 - Интеллектуальные системы в вооруженных силах
 - Проект с ММК по машинному зрению
- Специфика
 - Наличие большого количество свободно распространяемых библиотек для создания и обучения искусственных нейронных сетей
 - Нетривиальный математический аппарат
- Структура
 - Лекции 36 ч.
 - Лабораторные 18 ч.

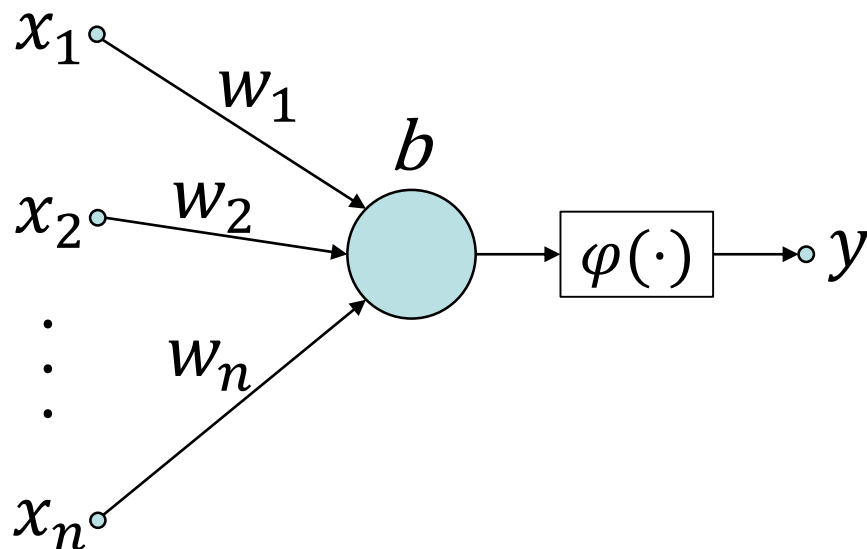
A detailed 3D rendering of a neuron, showing a central cell body (soma) with a textured, bumpy surface. Several long, thin, branching processes (dendrites and an axon) extend from the cell body. The processes have a segmented, beaded appearance. The background is a soft, out-of-focus light pinkish-white.

Машинное обучение

Модель нейронной сети

Лекция 2

Модель нейрона Мак-Каллока–Питса



$$y = \begin{cases} 0, & \text{если } \varphi(b + \langle \vec{w}, \vec{x} \rangle) \leq 0 \\ 1, & \text{если } \varphi(b + \langle \vec{w}, \vec{x} \rangle) > 0 \end{cases}$$

$$\langle \vec{w}, \vec{x} \rangle = \sum_{i=1}^n w_i x_i$$

$\vec{x} = (x_1, x_2, \dots, x_n)$ – входные сигналы: $x_i \in \{0, 1\}$

$\vec{w} = (w_1, w_2, \dots, w_n)$ – синаптические веса: $w_i \in \mathbb{R}$

b – порог: $b \in \mathbb{R}$

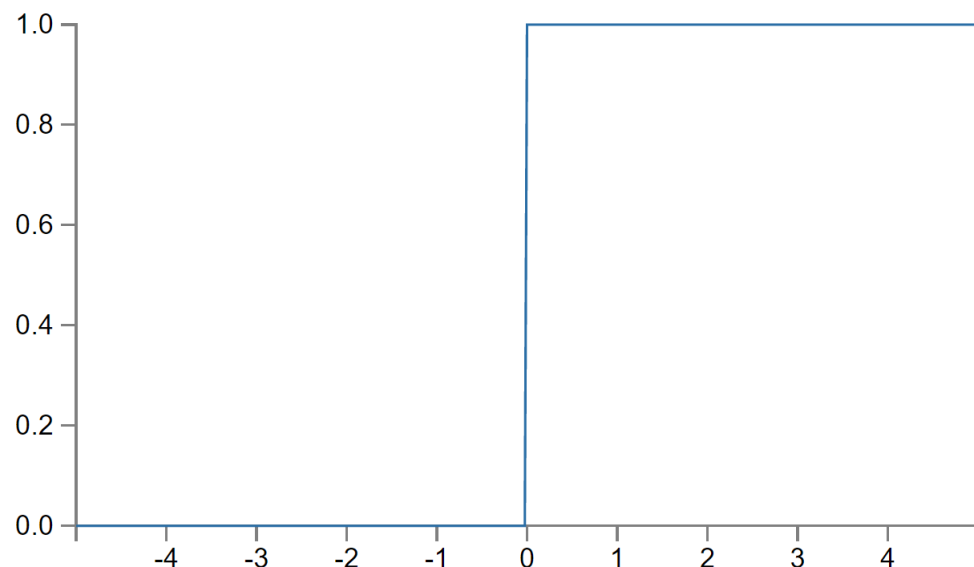
$\varphi: \mathbb{R} \rightarrow \{0, 1\}$ – функция активации

y – выходной сигнал: $y \in \{0, 1\}$

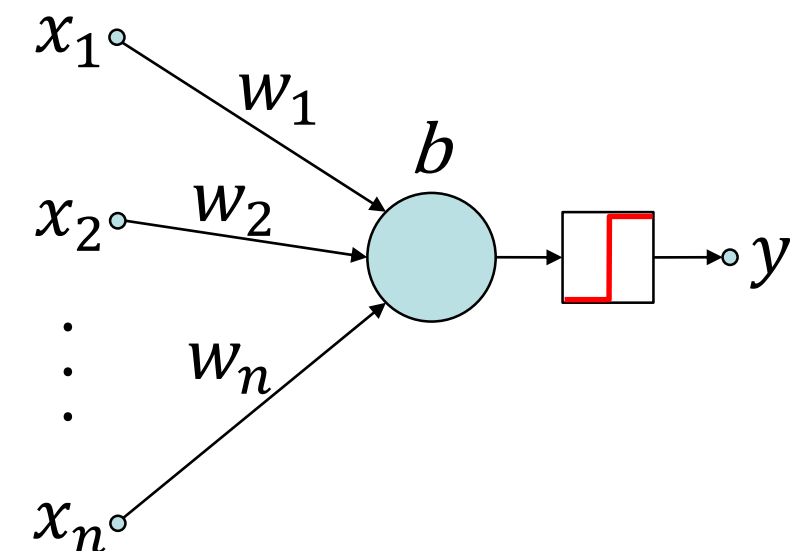
Персептрон

- Частный случай модели Мак-Каллока–Питса
- Функция активации является пороговой функцией

$$\varphi(v) = \begin{cases} 0, & \text{если } v \leq 0 \\ 1, & \text{если } v > 0 \end{cases}$$



Персептрон



$$y = \begin{cases} 0, & \text{если } b + \langle \vec{w}, \vec{x} \rangle \leq 0 \\ 1, & \text{если } b + \langle \vec{w}, \vec{x} \rangle > 0 \end{cases}$$

$$\langle \vec{w}, \vec{x} \rangle = \sum_{i=1}^n w_i x_i$$

$\vec{x} = (x_1, x_2, \dots, x_n)$ – входные сигналы: $x_i \in \{0, 1\}$

$\vec{w} = (w_1, w_2, \dots, w_n)$ – синаптические веса: $w_i \in \mathbb{R}$

b – порог: $b \in \mathbb{R}$

y – выходной сигнал: $y \in \{0, 1\}$

Простой пример с погодой

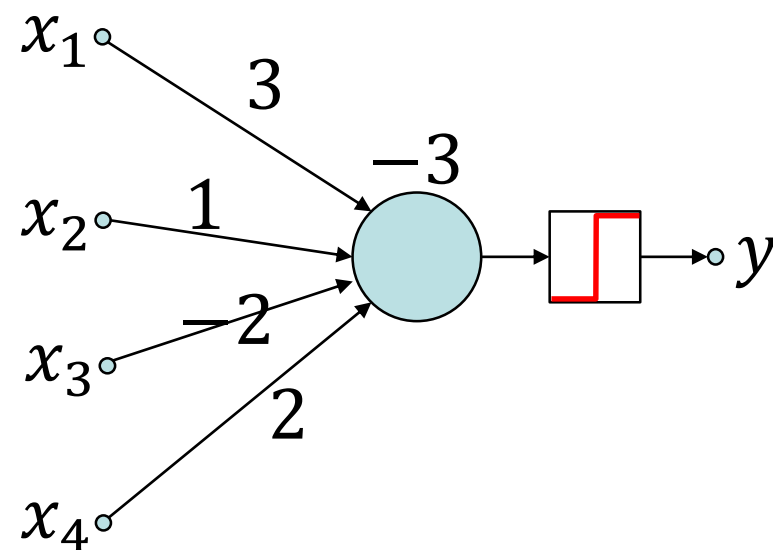
x_1 – температура больше $+15^{\circ}\text{C}$ и меньше $+30^{\circ}\text{C}$

x_2 – ветер меньше 3 м/сек.

x_3 – идет дождь

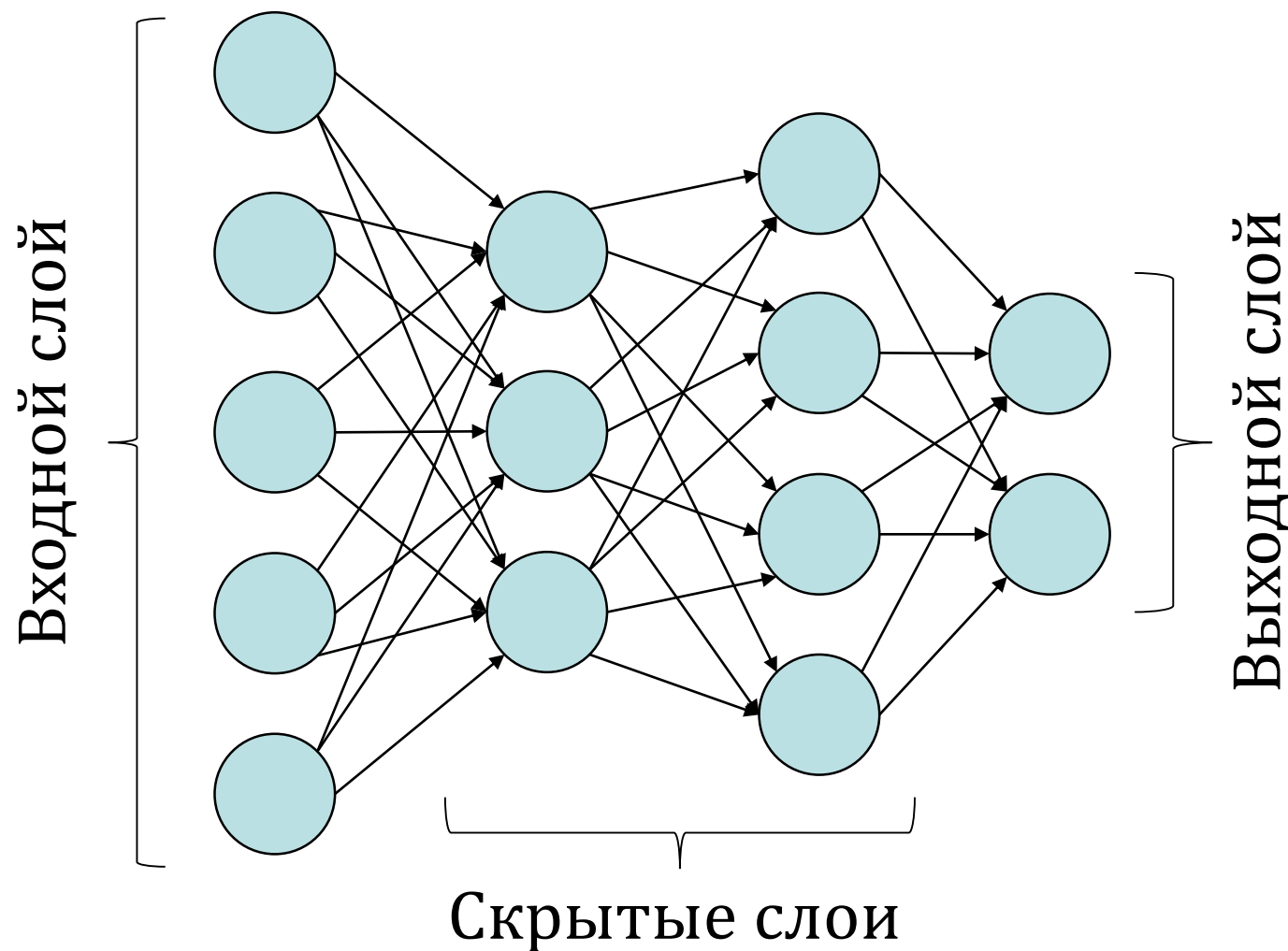
x_4 – светит солнце

y – можно идти гулять

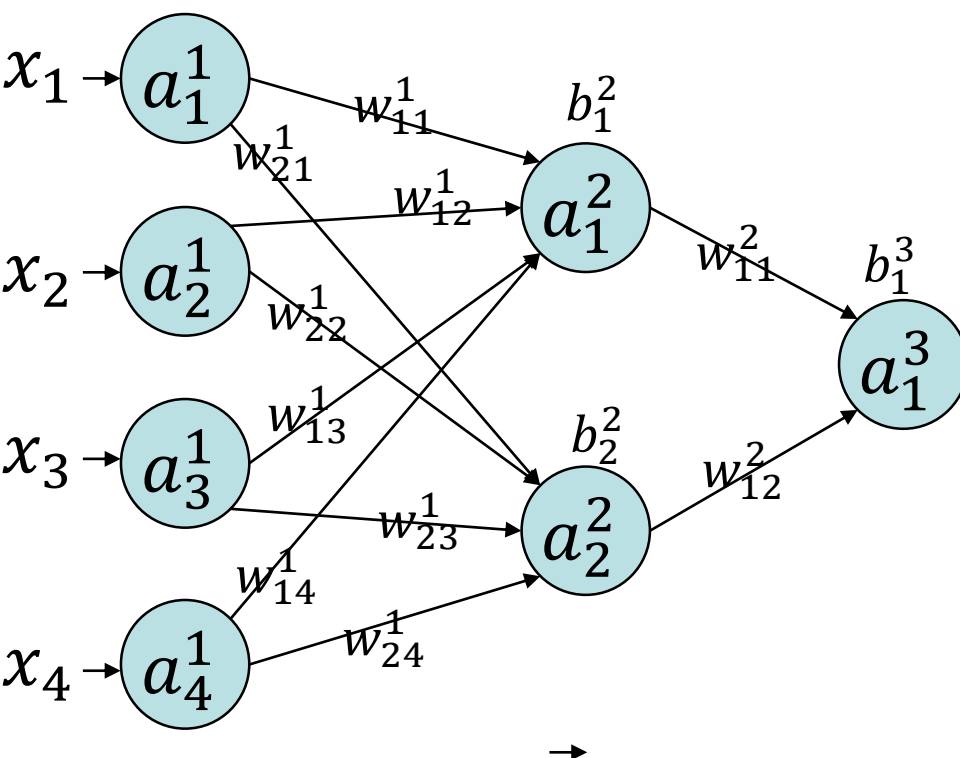


$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \rightarrow 1 \quad \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \rightarrow 0 \quad \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \rightarrow 1$$

Глубокая нейронная сеть прямого распространения



Вычисление выходного сигнала



$$a_j^1 = x_i \quad (j = 1, \dots, 4)$$

$$y_1^2 = w_{11}^1 a_1^1 + w_{12}^1 a_2^1 + w_{13}^1 a_3^1 + w_{14}^1 a_4^1$$

$$a_1^2 = \begin{cases} 0, & \text{если } b_1^2 + y_1^2 \leq 0 \\ 1, & \text{если } b_1^2 + y_1^2 > 0 \end{cases}$$

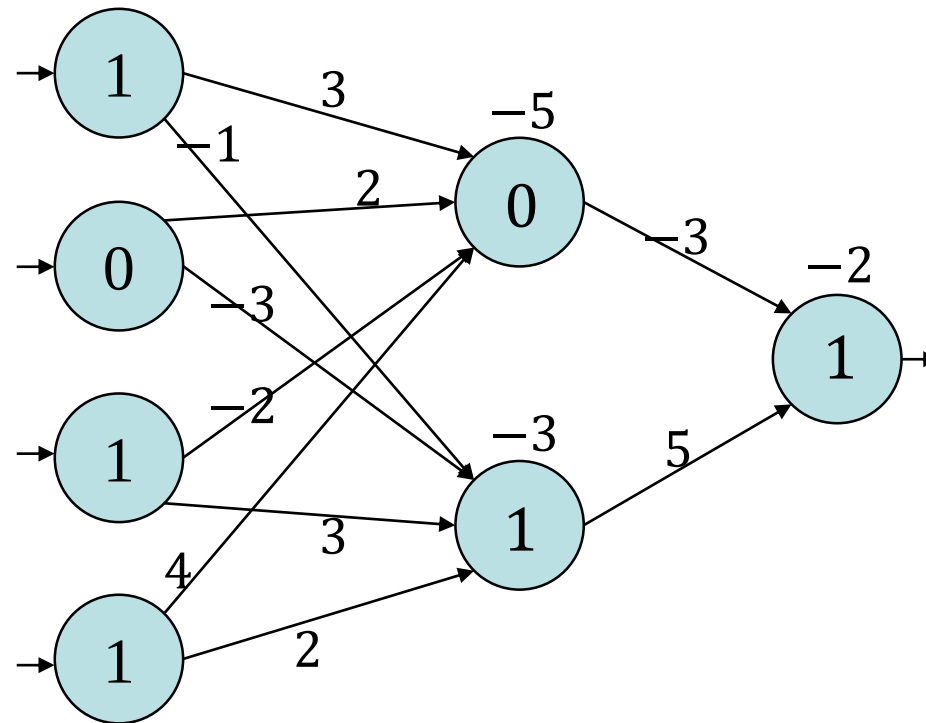
$$y_2^2 = w_{21}^1 a_1^1 + w_{22}^1 a_2^1 + w_{23}^1 a_3^1 + w_{24}^1 a_4^1$$

$$a_2^2 = \begin{cases} 0, & \text{если } b_2^2 + y_2^2 \leq 0 \\ 1, & \text{если } b_2^2 + y_2^2 > 0 \end{cases}$$

$$y_1^3 = w_{11}^2 a_1^2 + w_{21}^2 a_2^2$$

$$a_1^3 = \begin{cases} 0, & \text{если } b_1^3 + y_1^3 \leq 0 \\ 1, & \text{если } b_1^3 + y_1^3 > 0 \end{cases}$$

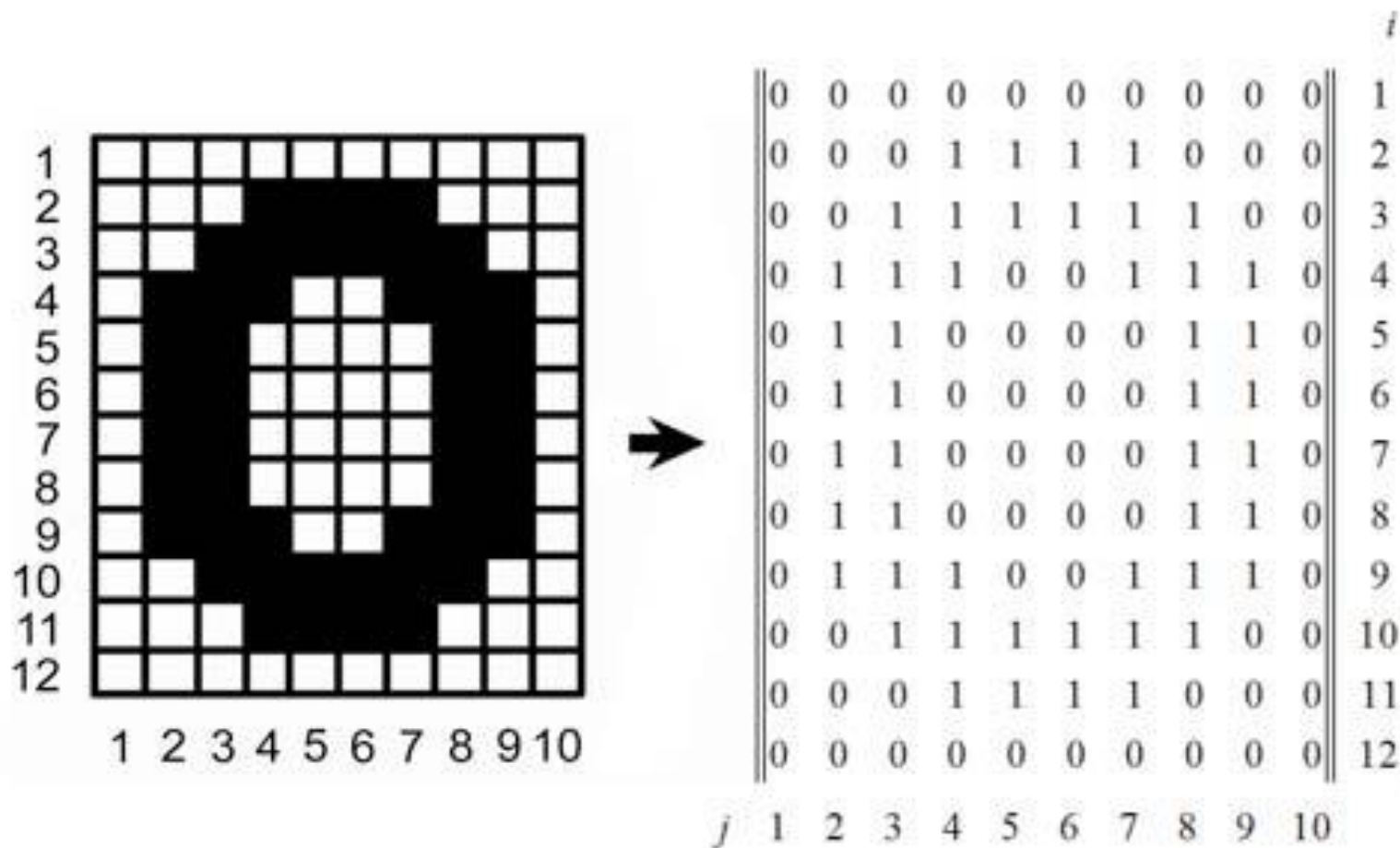
Пример вычисления выходного сигнала



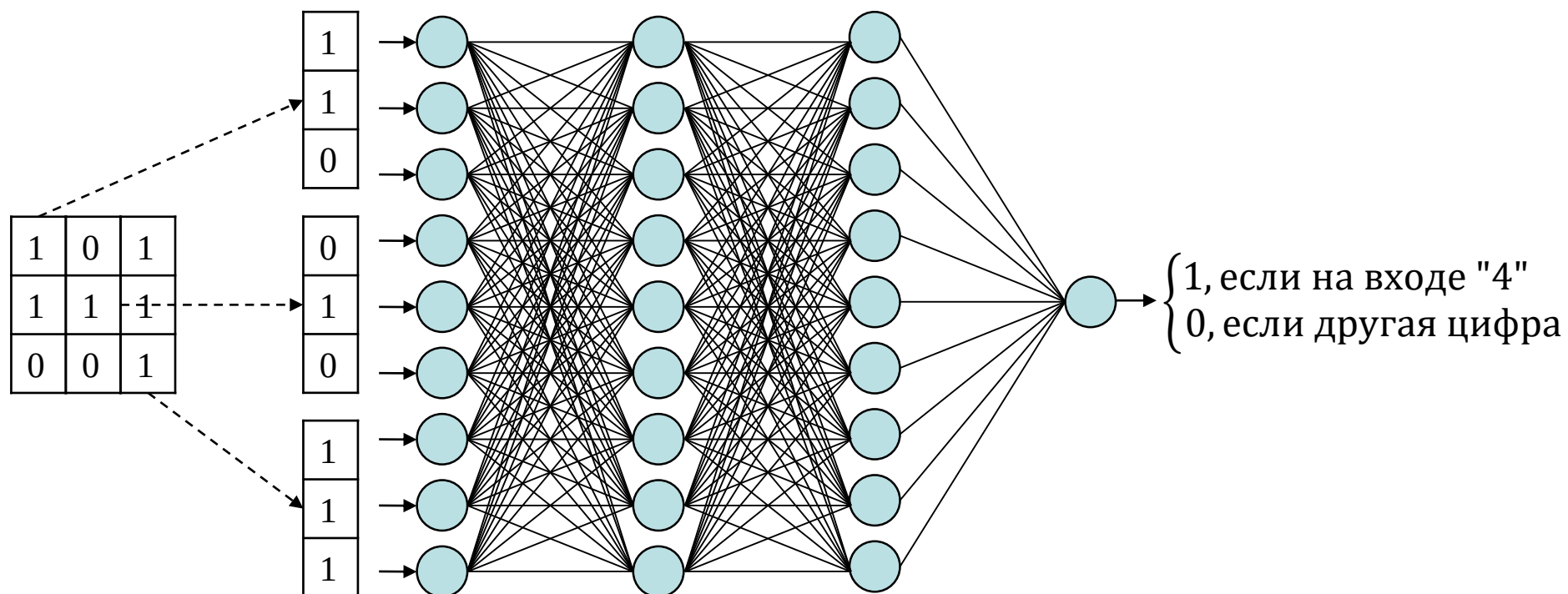
Задача распознавания рукописных цифр



Цифровой образ изображения

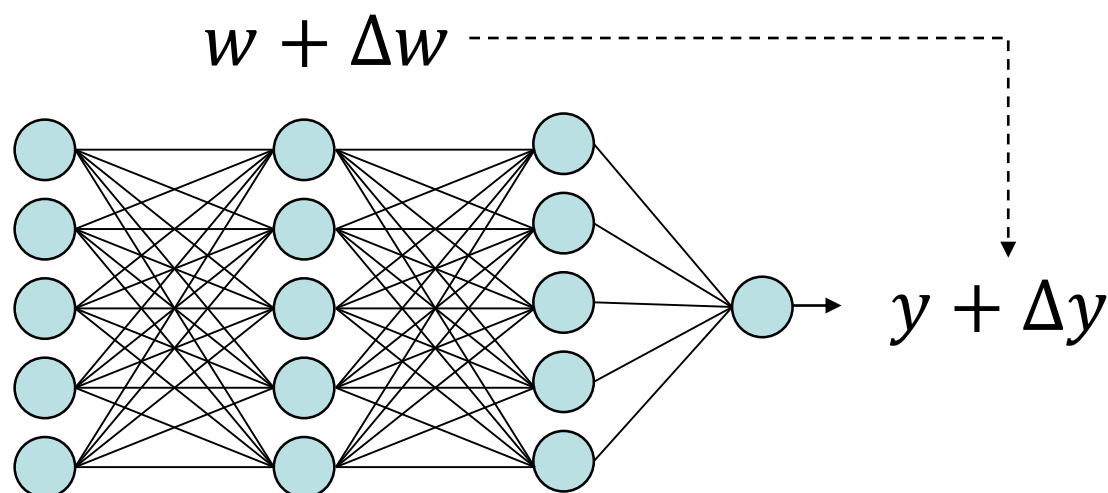


Распознавание цифры «4»



Возможность обучения нейронной сети

- Обучение – подбор синаптических весов и пороговых значений
- Это возможно, если малое изменение любого синаптического веса (или порогового значения) приводит к малому изменению выходного сигнала y



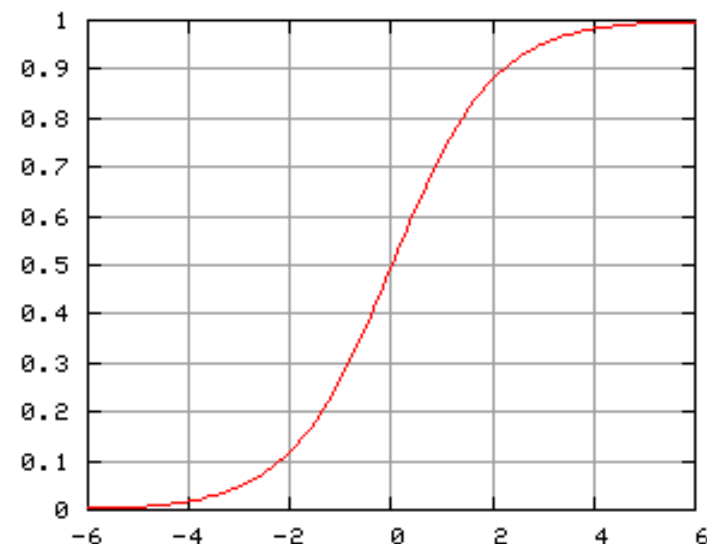
Трудности с обучением персептронных сетей

Сеть с персептроном трудно обучать, так как малое изменение синаптических весов может привести к скачкообразному изменению сигнала от 0 до 1.

Необходима другая модель нейрона!

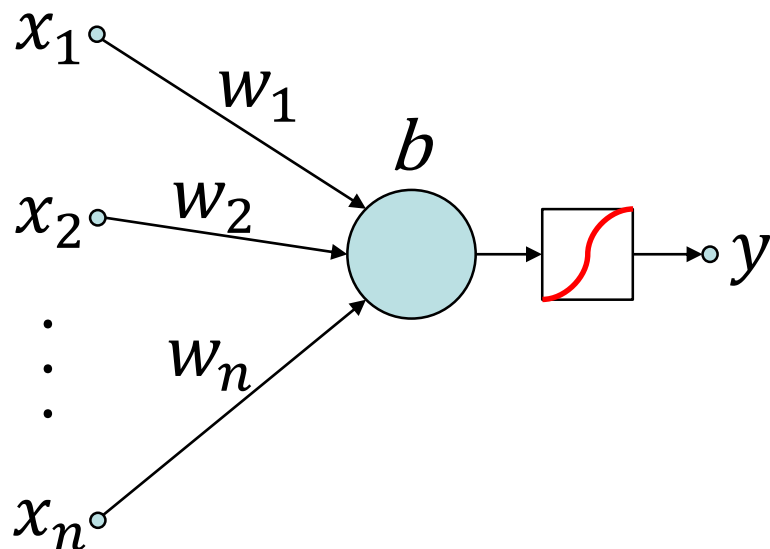
Сигмоидная (логистическая) функция

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



- Область определения: $\mathbb{R}$
- Множество значений: $(0;1)$

Сигмоид (сигмоидный нейрон)



$$y = \sigma(b + \langle \vec{w}, \vec{x} \rangle)$$

$$\langle \vec{w}, \vec{x} \rangle = \sum_{i=1}^n w_i x_i$$

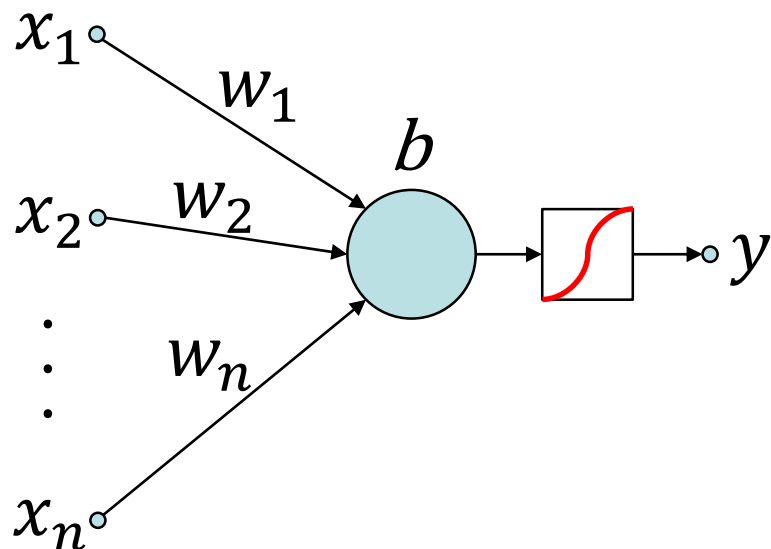
$\vec{x} = (x_1, x_2, \dots, x_n)$ – входные сигналы: $0 < x_i < 1$

$\vec{w} = (w_1, w_2, \dots, w_n)$ – синаптические веса: $w_i \in \mathbb{R}$

b – порог: $b \in \mathbb{R}$

y – выходной сигнал: $0 < y < 1$

Обучаемость Сигмоида

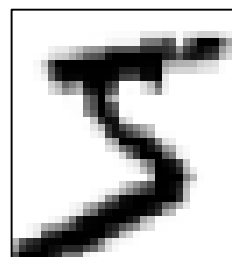


$$y = \frac{1}{1 + e^{-b - \sum_{i=1}^n w_i x_i}}$$

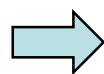
$$\Delta y \approx \frac{\partial y}{\partial b} \Delta b + \sum_{i=1}^n \frac{\partial y}{\partial w_i} \Delta w_i$$

Пример нейронной сети для распознавания рукописных цифр

5 0 4 1 9 2



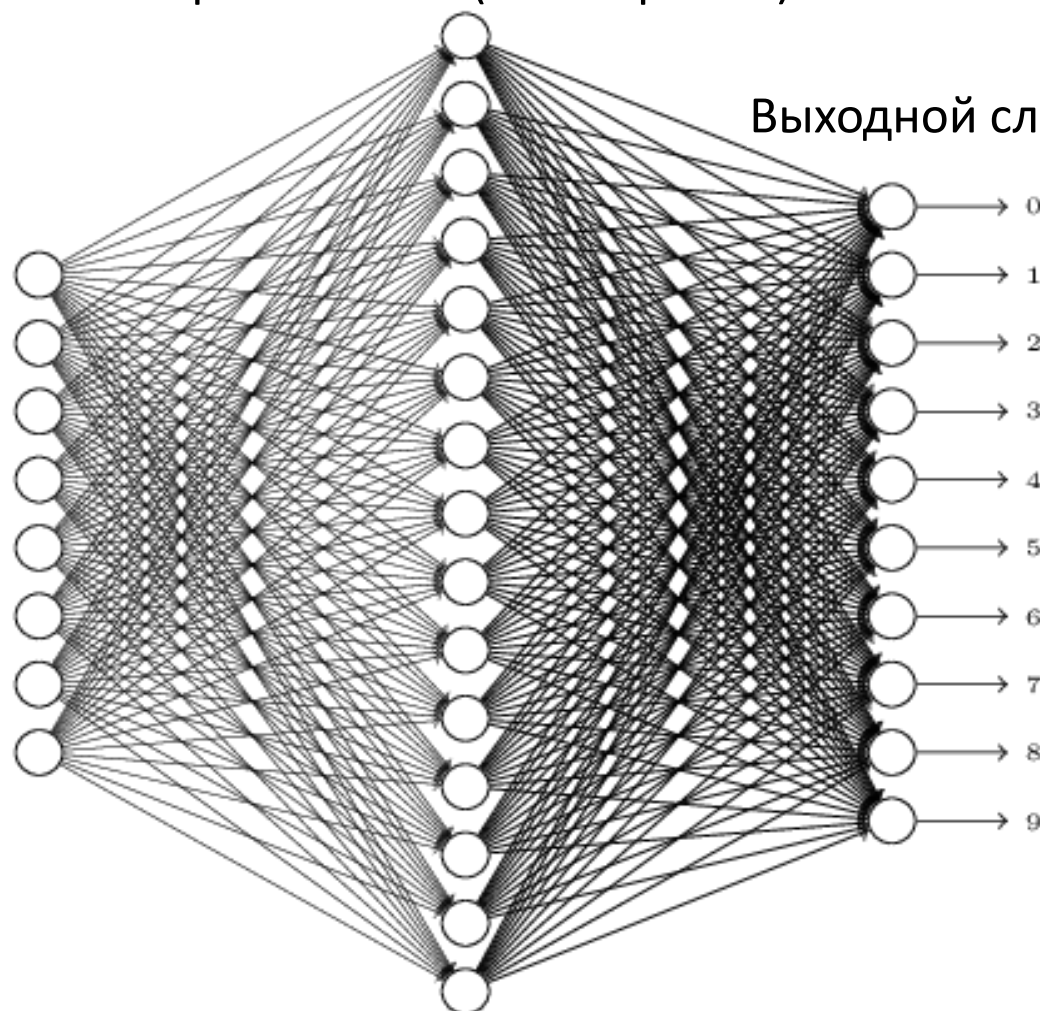
28×28



Входной слой
(784 нейрона)

Скрытый слой (15 нейронов)

Выходной слой



Модель обучения

- $\vec{x} \in \mathbb{R}^{28 \times 28}$ – изображение 28×28, подаваемое на вход нейронной сети
- $\vec{a} \in \mathbb{R}^{10}$ – правильный ответ
- $\vec{y}(\vec{x}) \in \mathbb{R}^{10}$ – ответ, выдаваемый нейронной сетью для входного изображения $\vec{x}$
- $X = \{(\vec{x}_i, \vec{a}_i) | 1 \leq i \leq m\}$ – обучающая выборка
- $\vec{w} \in \mathbb{R}^K$ – вектор, содержащий все синаптические веса сети
- $\vec{b} \in \mathbb{R}^L$ – вектор, содержащий все пороговые значения
- $$C(\vec{w}, \vec{b}) = \frac{1}{m} \sum_{(\vec{x}, \vec{a}) \in X} \frac{\|\vec{y}(\vec{x}) - \vec{a}\|^2}{2}$$
 – стоимостная функция (среднеквадратичная ошибка)
- Необходимо минимизировать среднеквадратичную ошибку путем подбора значений $\vec{w}$ и $\vec{b}$

Вопросы

- Где взять обучающую выборку?
- Как минимизировать среднеквадратичную ошибку?

База данных MNIST

<http://yann.lecun.com/exdb/mnist/>

- База данных образцов рукописного написания цифр
- Все изображения имеют размер 28×28 пикселей в градациях серого
- 60000 изображений для обучения нейронной сети
- 10000 изображений для тестирования нейронной сети
- Каждое изображение в базе данных промаркировано правильной цифрой



Метод градиентного спуска (двумерный случай)

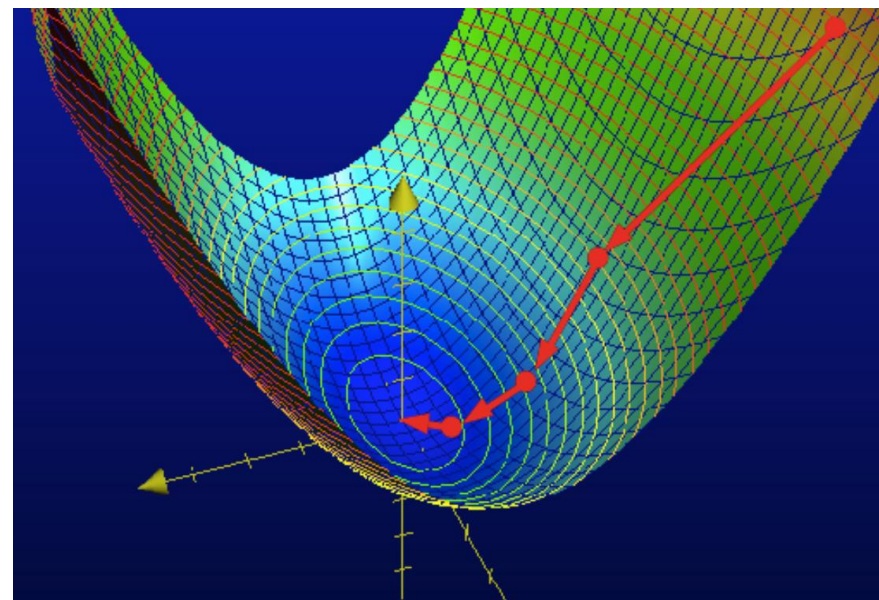
- Найти минимум функции $f(x_1, x_2): \mathbb{R}^2 \rightarrow \mathbb{R}$

- Градиент:

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

- $\Delta \vec{x} := -\eta \nabla f$,

где $\eta > 0$ – малый
параметр (*скорость
обучения*)



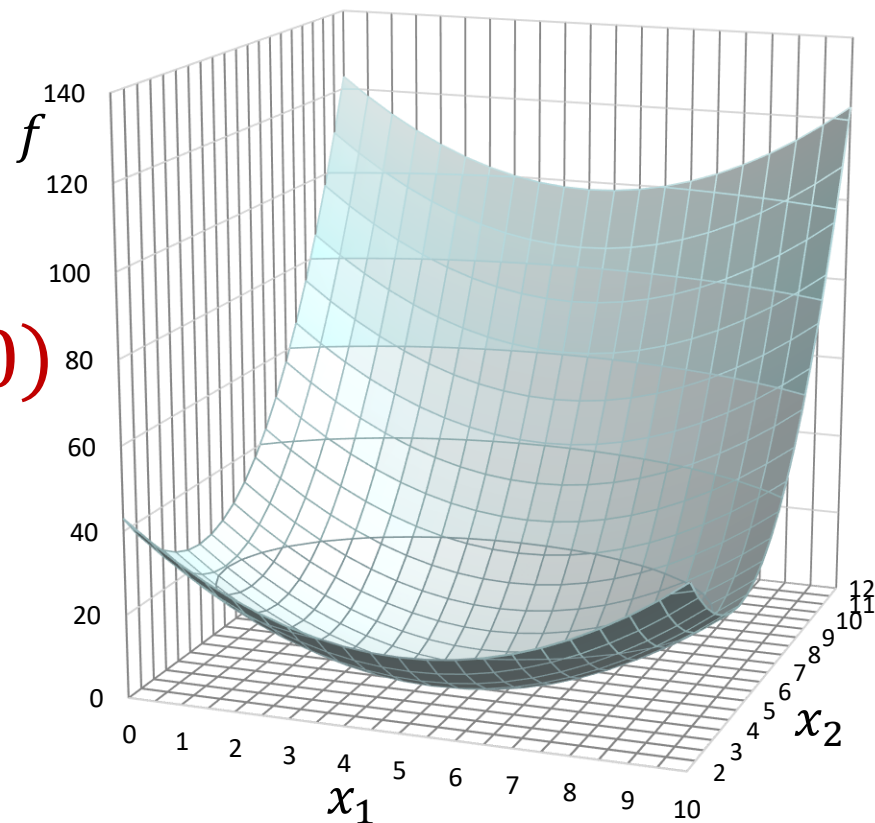
Пример вычисления градиента

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

$$\frac{\partial f(x_1, x_2)}{\partial x_1} = 2x_1 - 10$$

$$\frac{\partial f(x_1, x_2)}{\partial x_2} = 4x_2 - 20$$

$$\nabla f = (2x_1 - 10; 4x_2 - 20)$$



Пример градиентного спуска

$$\nabla f(x_1, x_2) = (2x_1 - 10; 4x_2 - 20)$$

$$\eta = 0.1$$

$$\Delta \vec{x} := -\eta \nabla f(x_1, x_2)$$

$$1. \quad \vec{x} := (1, 11)$$

$$2. \quad \nabla f(1, 11) = (-8; 24)$$

$$3. \quad \Delta \vec{x} := (0.8; -2.4)$$

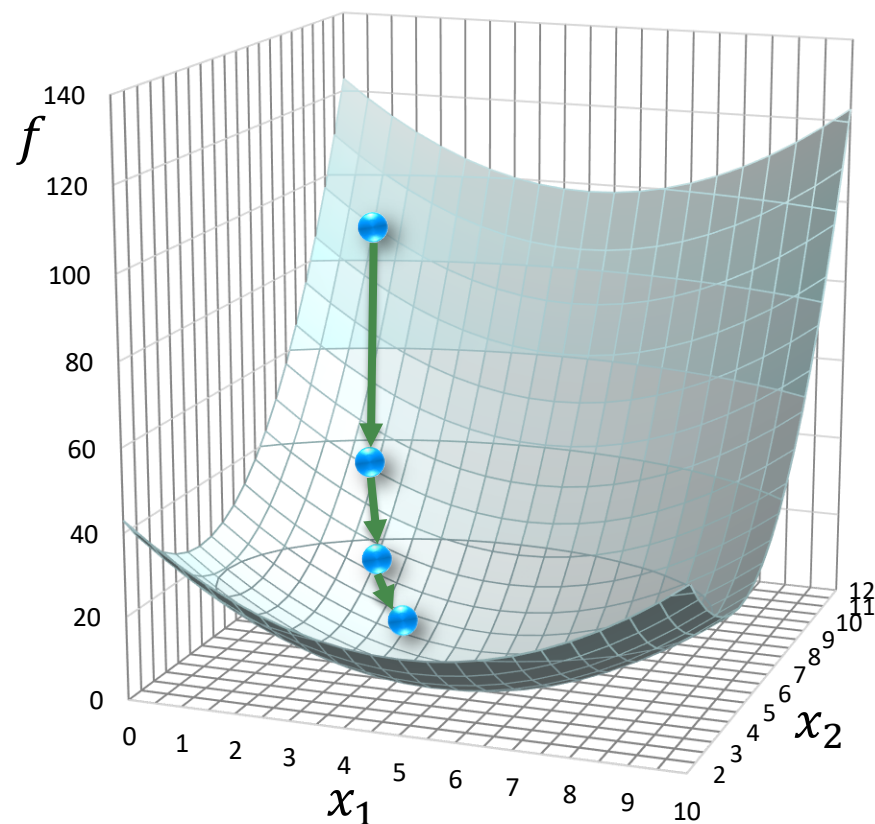
$$4. \quad \vec{x} := \vec{x} + \Delta \vec{x} = (1.8; 8.6)$$

$$5. \quad \nabla f(1.8, 8.6) = (-6.4; 14.4)$$

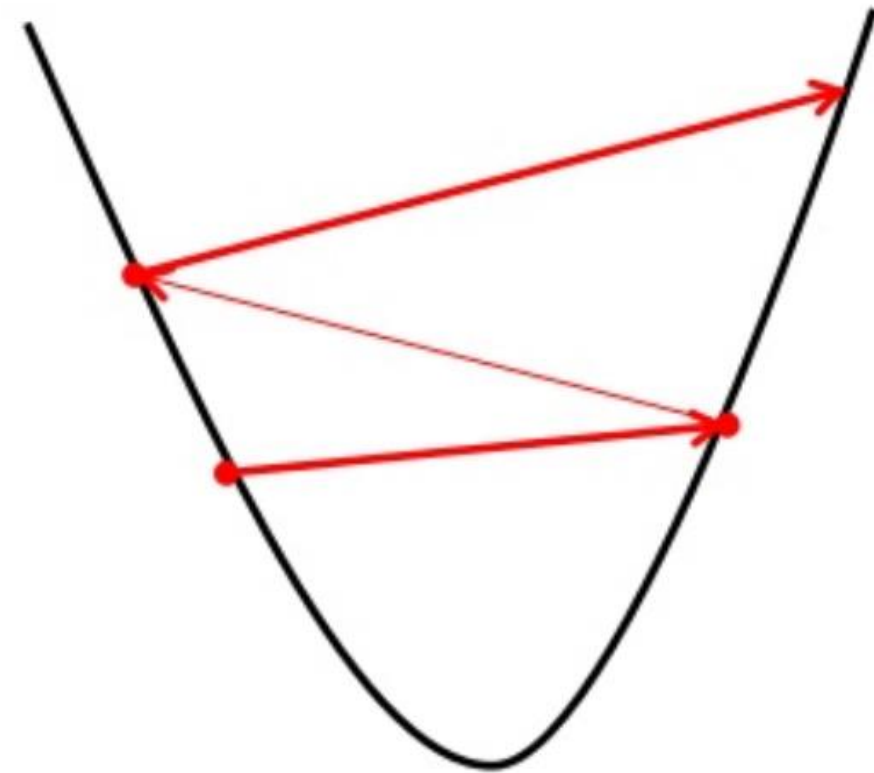
$$6. \quad \Delta \vec{x} := (0.64; -1.44)$$

$$7. \quad \vec{x} := \vec{x} + \Delta \vec{x} = (2.44; 7.16)$$

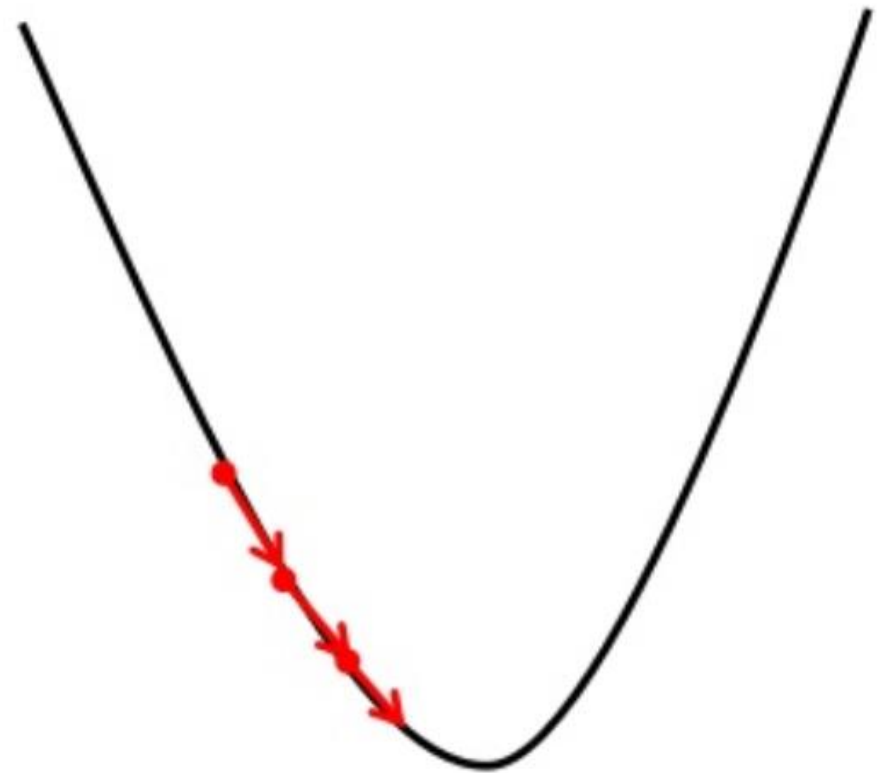
... ..



Выбор скорости обучения η



Большая скорость обучения



Малая скорость обучения

Применение к обучению нейронной сети

Необходимо минимизировать
среднеквадратичную ошибку:

$$C(\vec{w}, \vec{b}) = \frac{1}{m} \sum_{(\vec{x}, \vec{a}) \in X} \frac{\|\vec{y}(\vec{x}) - \vec{a}\|^2}{2}$$

$$w_k := w_k - \eta \frac{\partial C(\vec{w}, \vec{b})}{\partial w_k}$$

$$b_l := b_l - \eta \frac{\partial C(\vec{w}, \vec{b})}{\partial b_l}$$

© Соколинский Л.Б. 2. Модель нейронной сети 18.10.2017		
Модель обучения		
• $\vec{x} \in \mathbb{R}^{28 \times 28}$ – изображение 28×28, подаваемое на вход нейронной сети • $\vec{a} \in \mathbb{R}^{10}$ – правильный ответ • $\vec{y}(\vec{x}) \in \mathbb{R}^{10}$ – ответ, выдаваемый нейронной сетью для входного изображения $\vec{x}$ • $X = \{(\vec{x}_i, \vec{a}_i) 1 \leq i \leq m\}$ – обучающая выборка • $\vec{w} \in \mathbb{R}^K$ – вектор, содержащий все синаптические веса сети • $\vec{b} \in \mathbb{R}^L$ – вектор, содержащий все пороговые значения • $c(\vec{w}, \vec{b}) = \frac{1}{m} \sum_{(\vec{x}, \vec{a}) \in X} \frac{\ \vec{y}(\vec{x}) - \vec{a}\ ^2}{2}$ – стоимостная функция (среднеквадратичная ошибка) • Необходимо минимизировать среднеквадратичную ошибку путем подбора значений $\vec{w}$ и $\vec{b}$		
Машинное обучение 19/30		

Проблема

- Высокая вычислительная сложность процесса обучения

Стохастический градиентный спуск

$$C_{x,a}(\vec{w}, \vec{b}) = \frac{\|y(x) - a\|^2}{2}$$

Случайная разбивка обучающей выборки X на непересекающиеся подвыборки:

$$X = \bigcup_{i=1}^M X_i$$

1. $i := 0$
2. $i := i + 1$
3. $C_{X_i}(\vec{w}, \vec{b}) \stackrel{\text{def}}{=} \frac{1}{|X_i|} \sum_{(x,a) \in X_i} \frac{\|y(x) - a\|^2}{2}$
4. $\forall k \in \{1, \dots, K\} \left(w'_k := w_k - \eta \frac{\partial C_{X_i}}{\partial w_k} \right)$
5. $\forall l \in \{1, \dots, L\} \left(b'_l := b_l - \eta \frac{\partial C_{X_i}}{\partial b_l} \right)$
6. **if** $\|\vec{w}' - \vec{w}\| + \|\vec{b}' - \vec{b}\| \leq \varepsilon$ **goto** 10
7. $\vec{w} := \vec{w}'$
8. $\vec{b} := \vec{b}'$
9. **goto** 4
10. **if** $i < M$ **goto** 2

Градиентный спуск

Эпоха обучения

Проблемы

- Необходим эффективный алгоритм вычисления выходного сигнала

$$\vec{y}(\vec{x})$$

- Необходим эффективный алгоритм вычисления градиента стоимостной функции

$$C(\vec{w}, \vec{b})$$

Источники

1. Nielsen M.A. Neural Networks and Deep Learning [Electronic resource]. Determination Press, 2015. URL:
<http://neuralnetworksanddeeplearning.com>
2. Haykin S. Neural networks and learning machines. 3rd ed. Prentice Hall, 2008. 906 p.