

Машинное обучение

Введение в глубинное обучение.

Векторное представление текстов.

Михаил Гущин

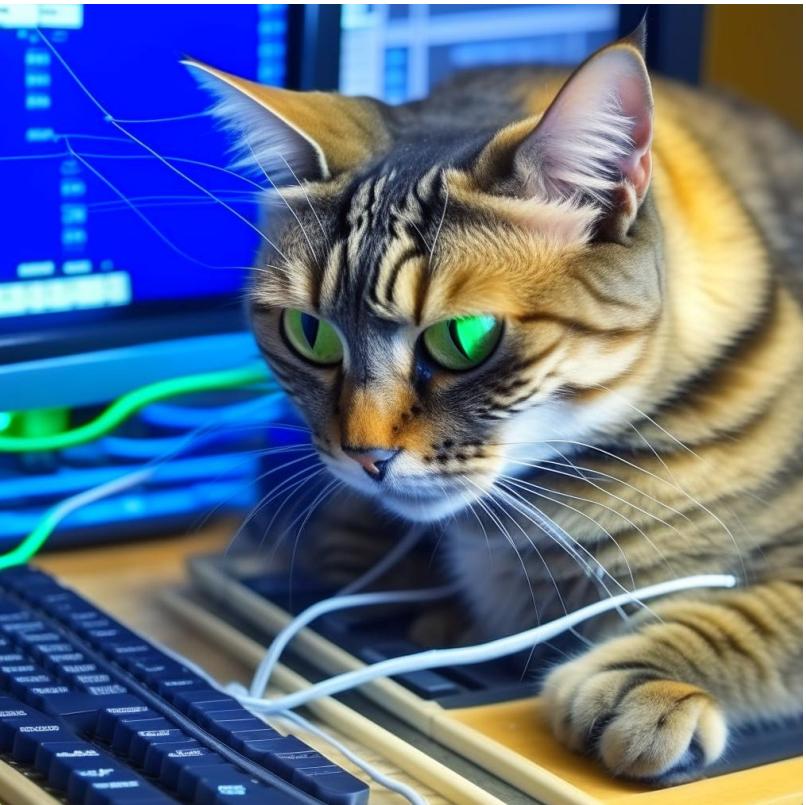
mhushchyn@hse.ru

НИУ ВШЭ, 2024



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Генерация изображения по описанию



Кот фиксит баг в обучении
нейронной сети



Розовый фламинго стоит на
одной ноге в воде

Ссылка: <https://www.sberbank.com/promo/kandinsky>

Deepfake



John Wick 4: Dangerous Cleanup
7,4 млн просмотров



This is you right now 😂
1,2 млн просмотров



Katana wins
18 млн просмотров



Thank you, my followers ❤️
1,9 млн просмотров



10 minutes of eternity...
1,9 млн просмотров



Keanu Reeves lives with his girlfriend
19 млн просмотров



Did you recognize everyone?
2,1 млн просмотров



When I came back from filming
1,2 млн просмотров



You spin USB right 'round, baby, right 'round... #keanu...
1,5 млн просмотров



Who is the best dancer? 😎
#keanu #reeves #dance #fyp
7,9 млн просмотров



Mirror trick 😊 #keanu #reeves #mirror #tenet
1,1 млн просмотров

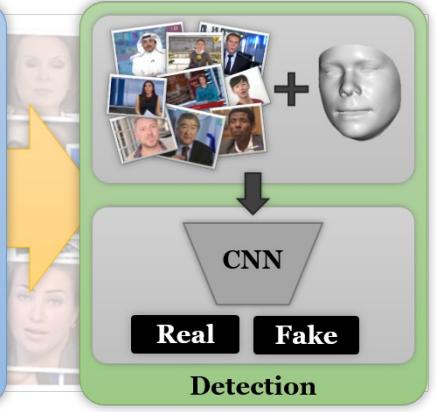
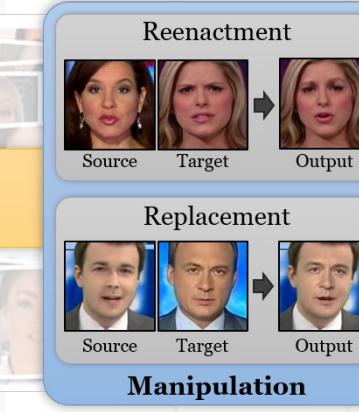
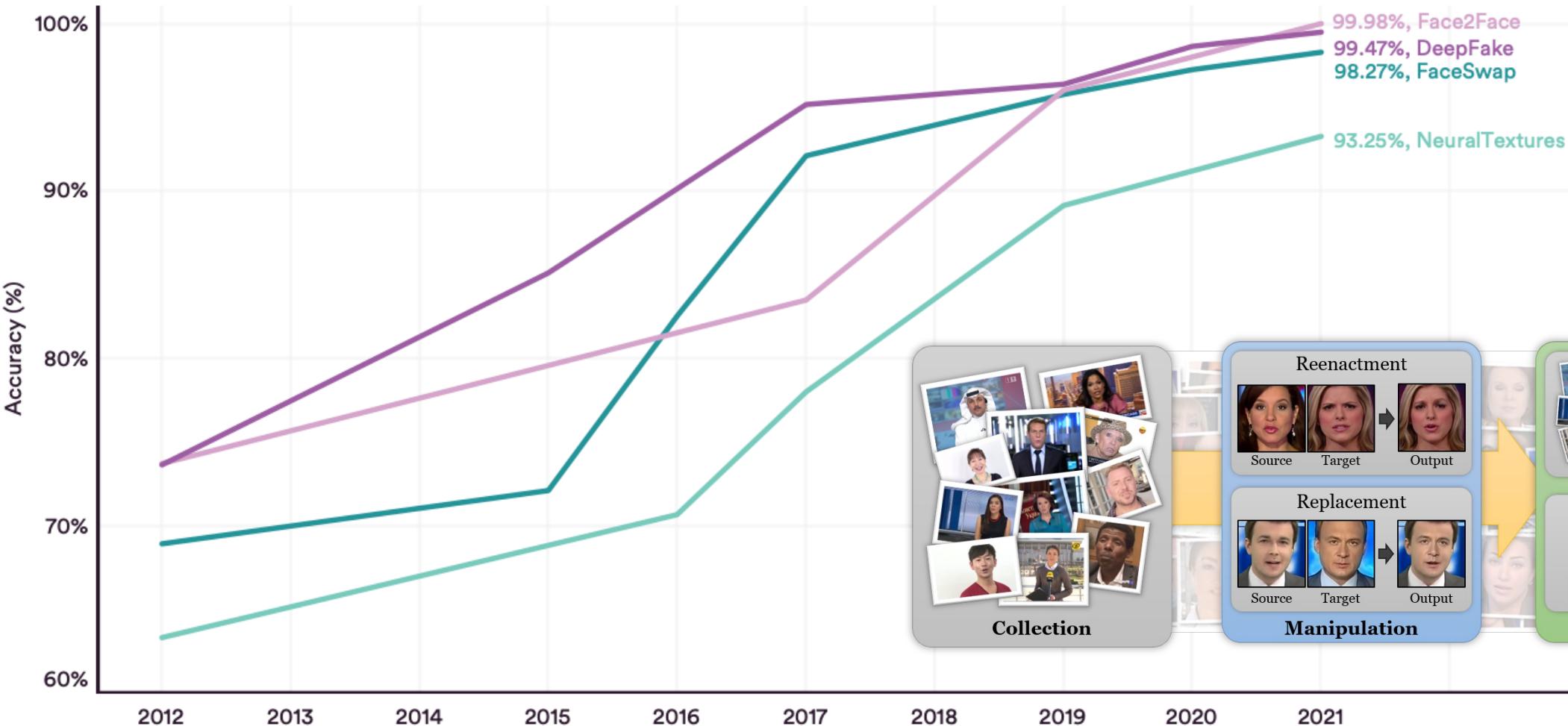


Dressing up like a cool guy.
#reeves #keanu #dressup
1,1 млн просмотров

Deepfake detection

FACEFORENSICS++: ACCURACY

Source: arXiv, 2021 | Chart: 2022 AI Index Report



Как отличить картинки



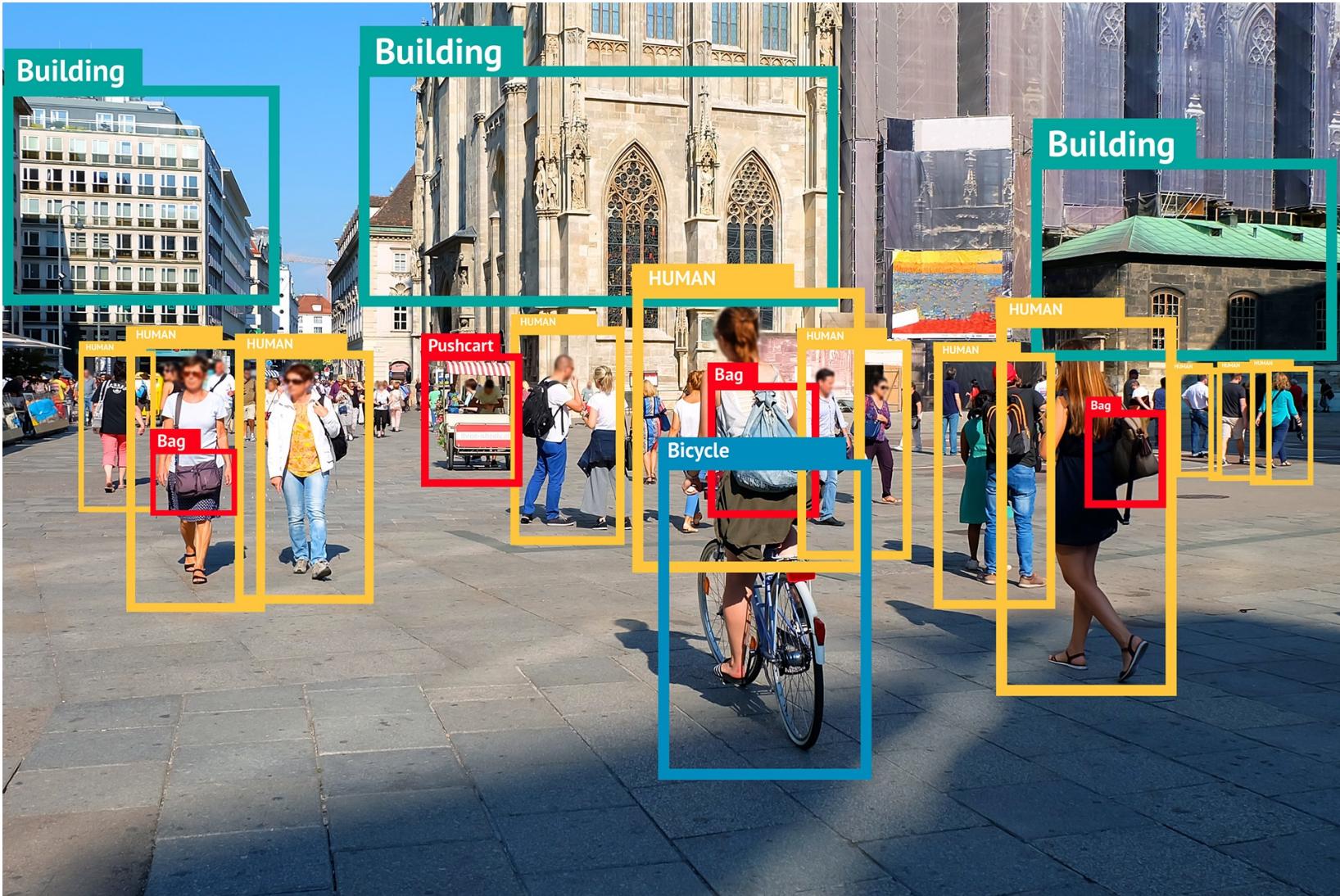
Erkhyan
@erkhyan@yiff.life

◀ prg_memes

Что забавно: если хочешь понять,
сгенерирован ли рисунок нейросеткой,
следуй правилам, по которым в старых
сказках распознавали нечистую силу.

«Посчитай пальцы, посчитай костяшки,
посчитай зубы, проверь тень...»
...и НИ ПРИ КАКИХ обстоятельствах не
соглашайся на сделку с этим отродьем.

Детектирование объектов



Генерация текста

A screenshot of a text generation interface. At the top, a user icon shows a person's face, followed by the text: "give some examples of the types of writing AI can do, and why that will transform businesses in a paragraph." To the right of this text is a small edit icon. Below this, the AI's response is shown, preceded by an AI icon (a green square with a white swirl). The AI text reads: "AI can be used to write a wide range of business-related documents, including reports, proposals, and marketing materials. By utilizing AI's ability to generate ideas and provide information, businesses can quickly and easily create high-quality written materials. This can save time and resources, allowing businesses to focus on other important tasks. Additionally, AI can help ensure accuracy and consistency in writing, improving the overall professionalism and effectiveness of business documents. The use of AI in writing will transform businesses by streamlining the writing process and improving the quality and accuracy of written materials." To the right of the AI text are two small icons: a thumbs up and a thumbs down.

Машинный перевод

Яндекс Браузер

Яндекс.Браузер обновился. Версия 21.8.2

1

The screenshot shows the Yandex Browser homepage. At the top left is the browser logo and name. At the top right are update notifications for 'Яндекс.Браузер обновился. Версия 21.8.2' and like/dislike buttons. The main content area features a video player with a play button. A speech bubble from a character named David says: 'Hi! I'm David and I lead the NLP team at Yandex'. Below the video, there's a large heading 'Закадровый перевод видео с английского' (Subtitles for video from English) and a text block explaining the feature: 'Нейросети Яндекса научились сами переводить и озвучивать видео на английском языке. Пока — не везде, но уже скоро любой ролик на английском можно будет смотреть на русском.' It also includes a link to try it out. To the left of the video player is a small thumbnail of the video frame. Below the video player are social sharing links for Dzen, VKontakte, Twitter, and Telegram, along with a note about DRM restrictions.

Hi! I'm David and I lead the NLP team at Yandex

Закадровый перевод видео с английского

Нейросети Яндекса научились сами переводить и озвучивать видео на английском языке. Пока — не везде, но уже скоро любой ролик на английском можно будет смотреть на русском.

Сразу попробовать новую функцию можно [по ссылке](#).

Как включить перевод видео?

Подписывайтесь на новости Яндекс.Браузера:

Дзен ВКонтакте Твиттер Телеграм

Перевод не доступен для видео, у которых есть технические средства защиты авторских прав (DRM).

Голосовые помощники

The screenshot shows the Yandex.Alice app interface. At the top center is the Alice logo (a stylized white egg inside a purple circle). Below it is a large white speech bubble containing the text "Привет, я Алиса!". The background is a solid purple color. In the center, the text "Я готова помочь" is displayed. Below this, there are six items, each with an icon and text: "Определить песню" (Song recognition), "Узнать, что на фото" (Identify what's in the photo), "Включить сказку" (Play a story), "Одеться по погоде" (Dress according to the weather), "Поиграть" (Play), and "Построить маршрут" (Build a route). There are also links for "Вызвать такси" (Call a taxi) and "Найти нужное место" (Find the right place), along with "Купить на Беру" (Buy on Beru). Each item is preceded by a right-pointing arrow.

- Определить песню >
- Узнать, что на фото >
- Включить сказку >
- Одеться по погоде >
- Поиграть >
- Построить маршрут >
- Вызвать такси >
- Найти нужное место >
- Купить на Беру >

Эволюция нейронных сетей

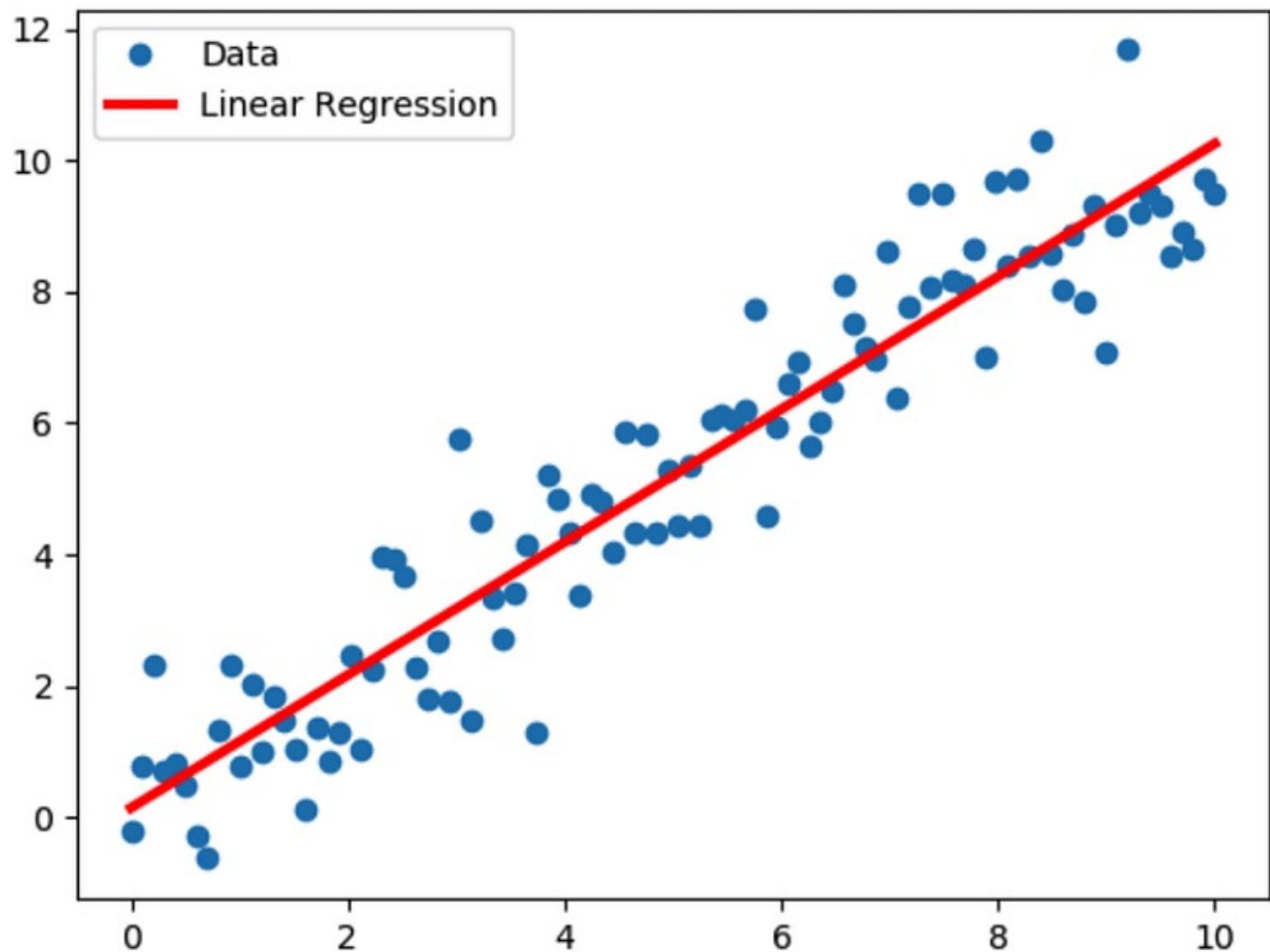


Линейная
регрессия

Логистическая
регрессия

Нейронная сеть

Линейная регрессия



Линейная регрессия

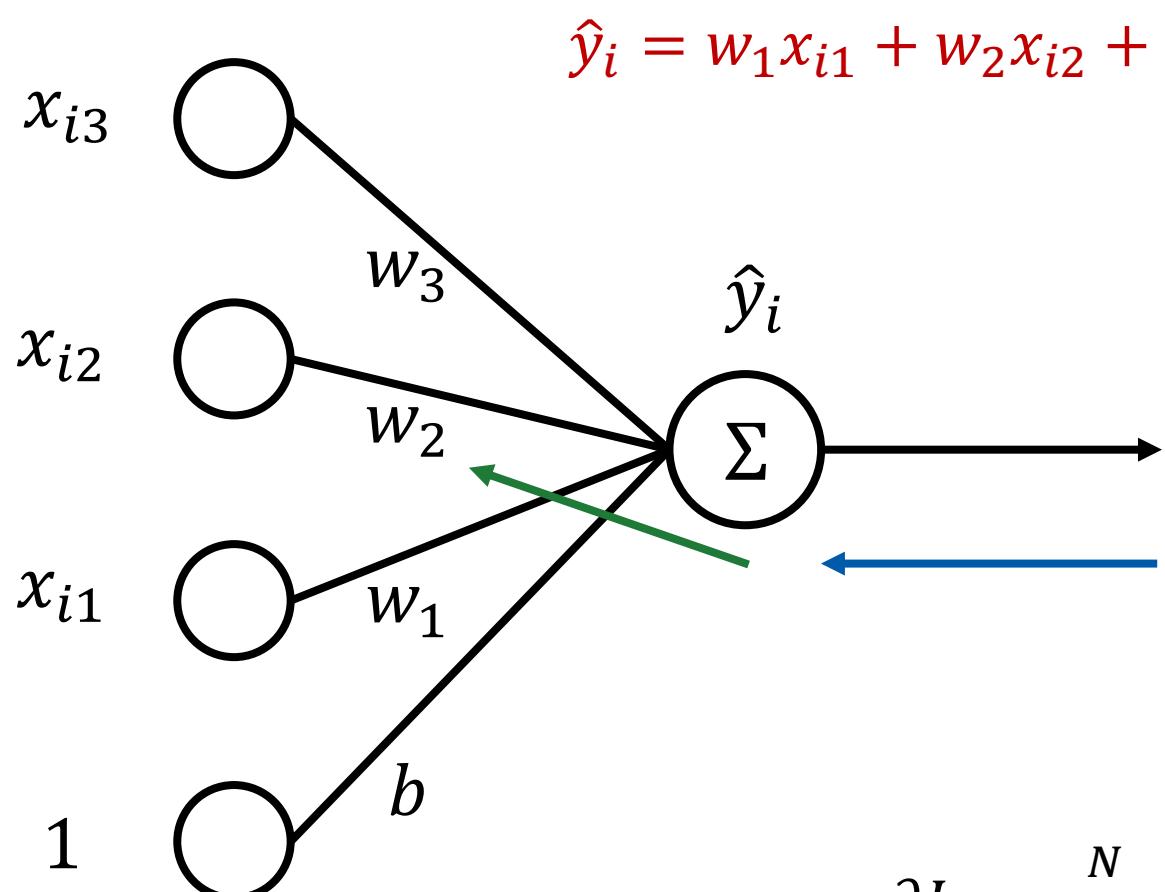
- ▶ Пусть дан набор наблюдений $\{x_i, y_i\}_{i=1}^N$, где $x_i \in R^3$, $y_i \in R$
- ▶ Модель линейной регрессии:

$$\hat{y}_i = w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i3} + b = x_i^T w + b$$

- ▶ Функция потерь:

$$L = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2 \rightarrow \min_{b, w_1, w_2, w_3}$$

Градиентный спуск



$$\frac{\partial L}{\partial w_2} = \sum_{i=1}^N \frac{\partial L_i}{\partial w_2}$$

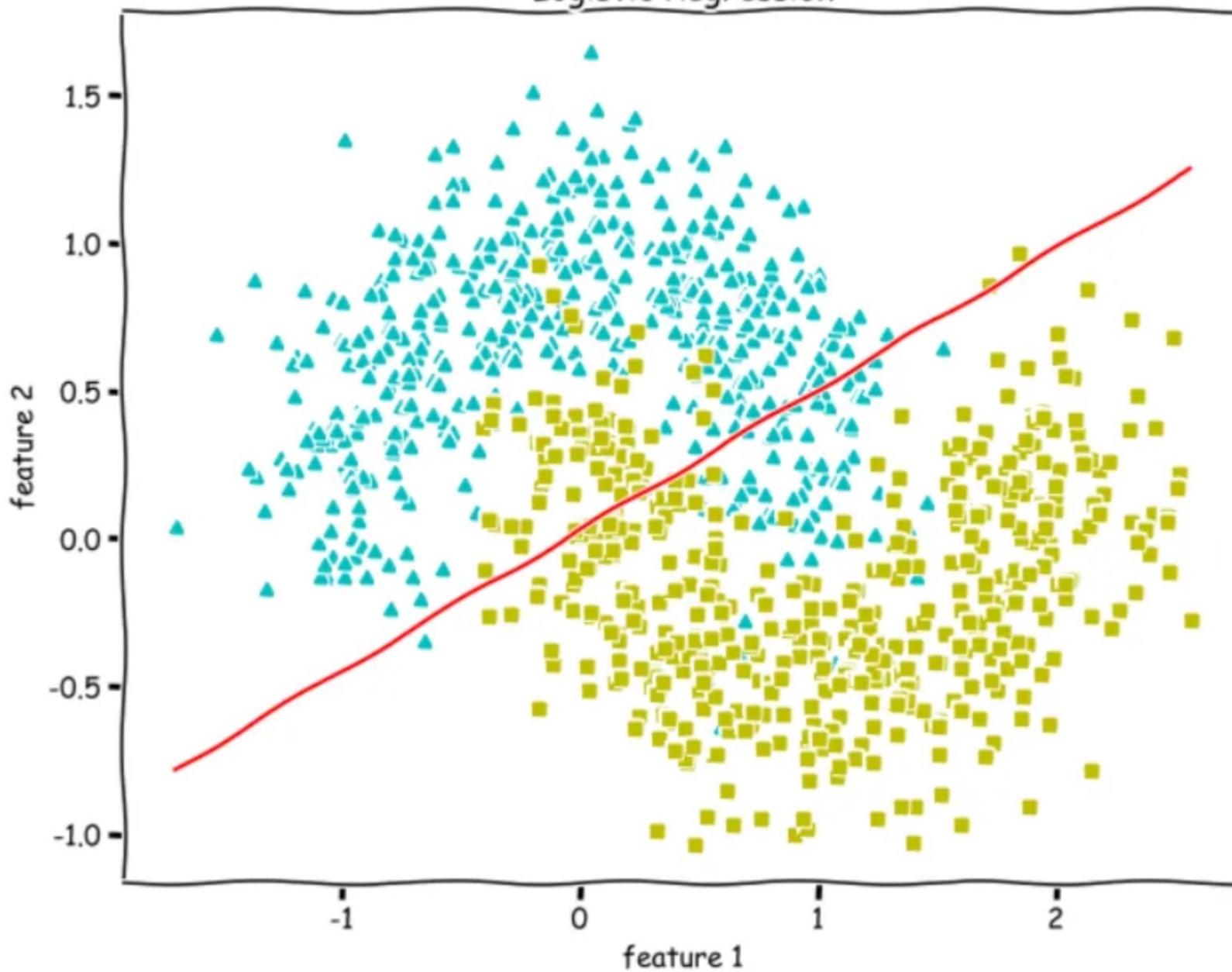
$$L_i = \frac{1}{N} (\hat{y}_i - y_i)^2$$

$$\boxed{\frac{\partial L_i}{\partial w_2} = \frac{\partial L_i}{\partial \hat{y}_i} \frac{\partial \hat{y}_i}{\partial w_2}}$$

$$w_2^{(t+1)} = w_2^{(t)} - \alpha \frac{\partial L}{\partial w_2}$$

Логистическая регрессия

Logistic Regression



Логистическая регрессия

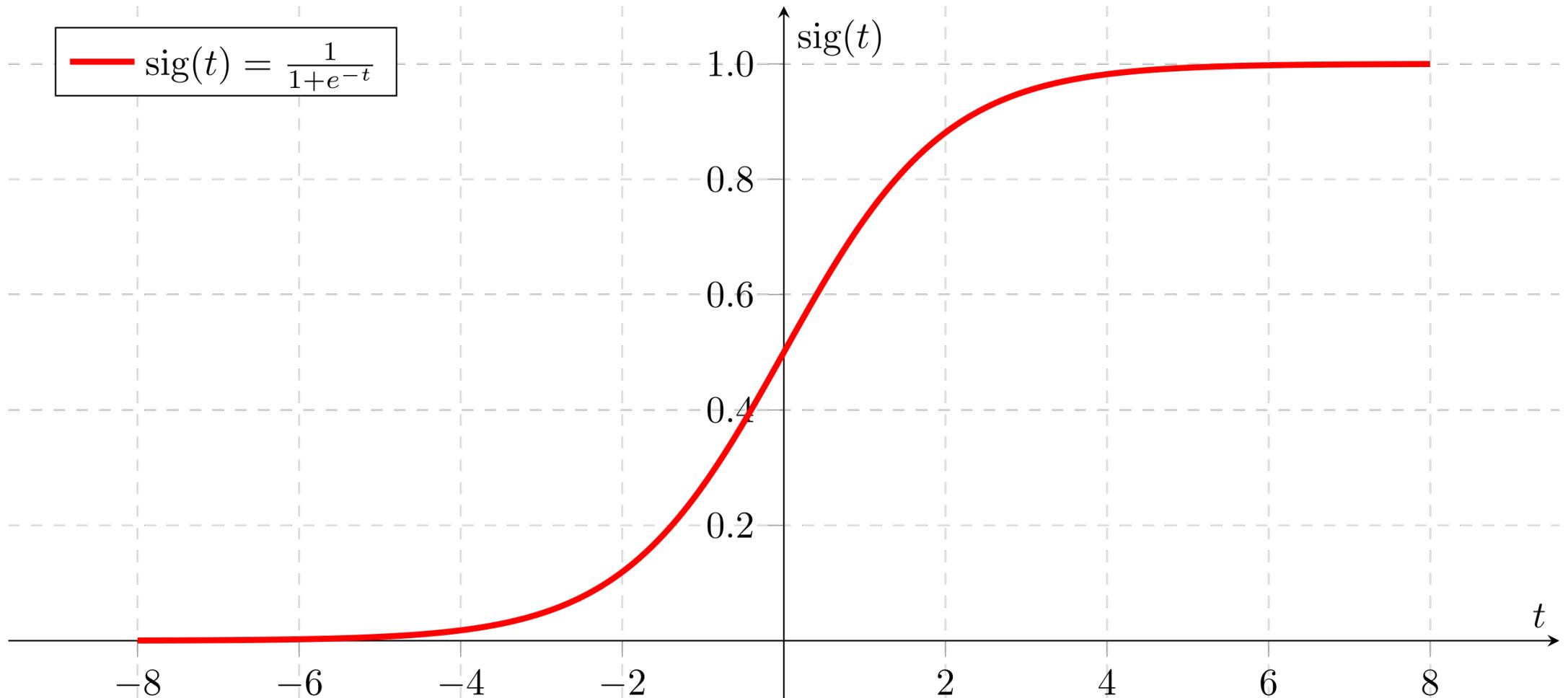
- ▶ Пусть дан набор наблюдений $\{x_i, y_i\}_{i=1}^N$, где $x_i \in R^3$, $y_i \in \{0, 1\}$
- ▶ Модель логистической регрессии:

$$\hat{y}_i = \sigma(w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i3} + b) = \sigma(x_i^T w + b)$$

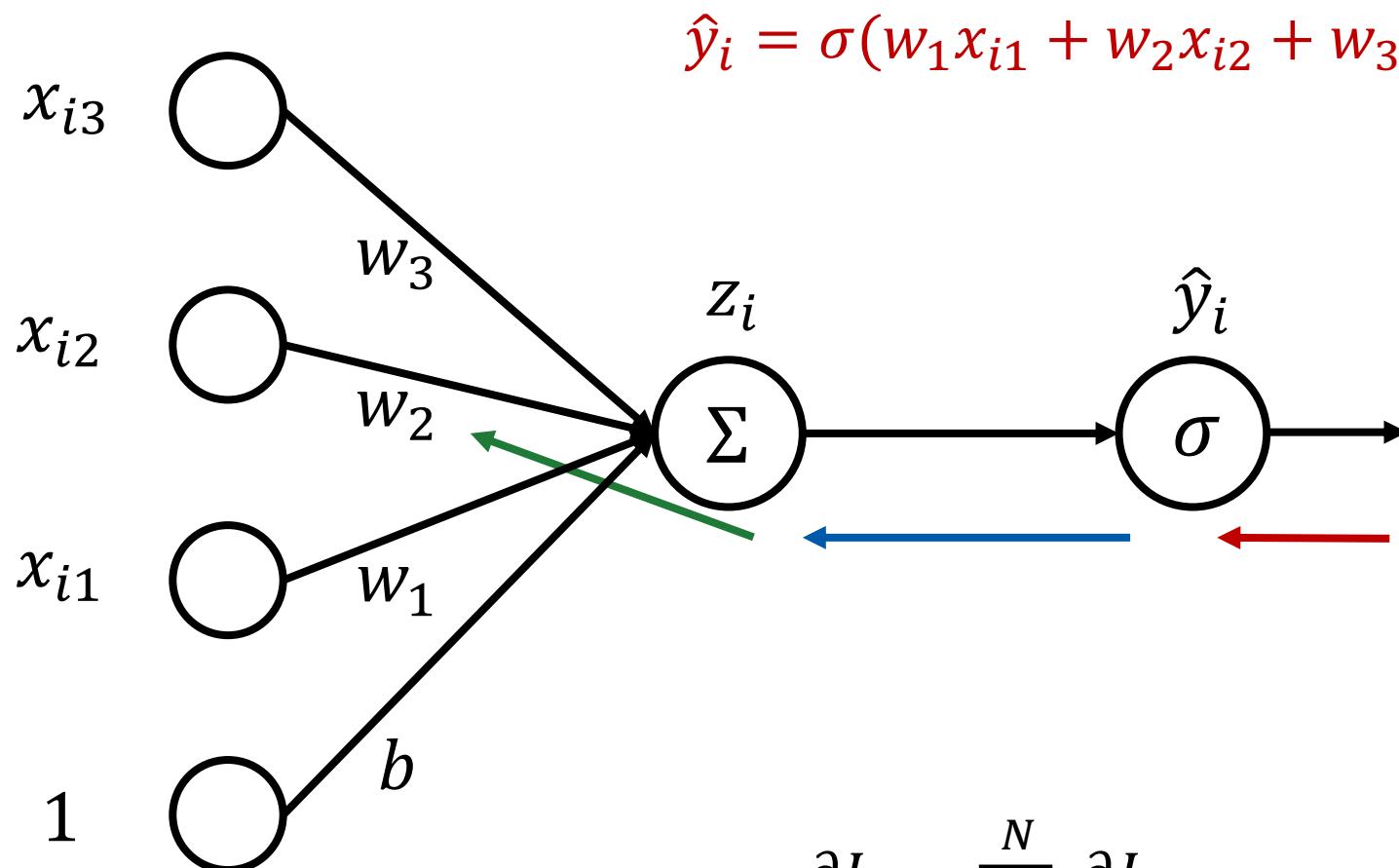
- ▶ Функция потерь:

$$L = -\frac{1}{N} \sum_{i=1}^N y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \rightarrow \min_{b, w_1, w_2, w_3}$$

Сигмоида



Градиентный спуск



$$L_i = -y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i)$$

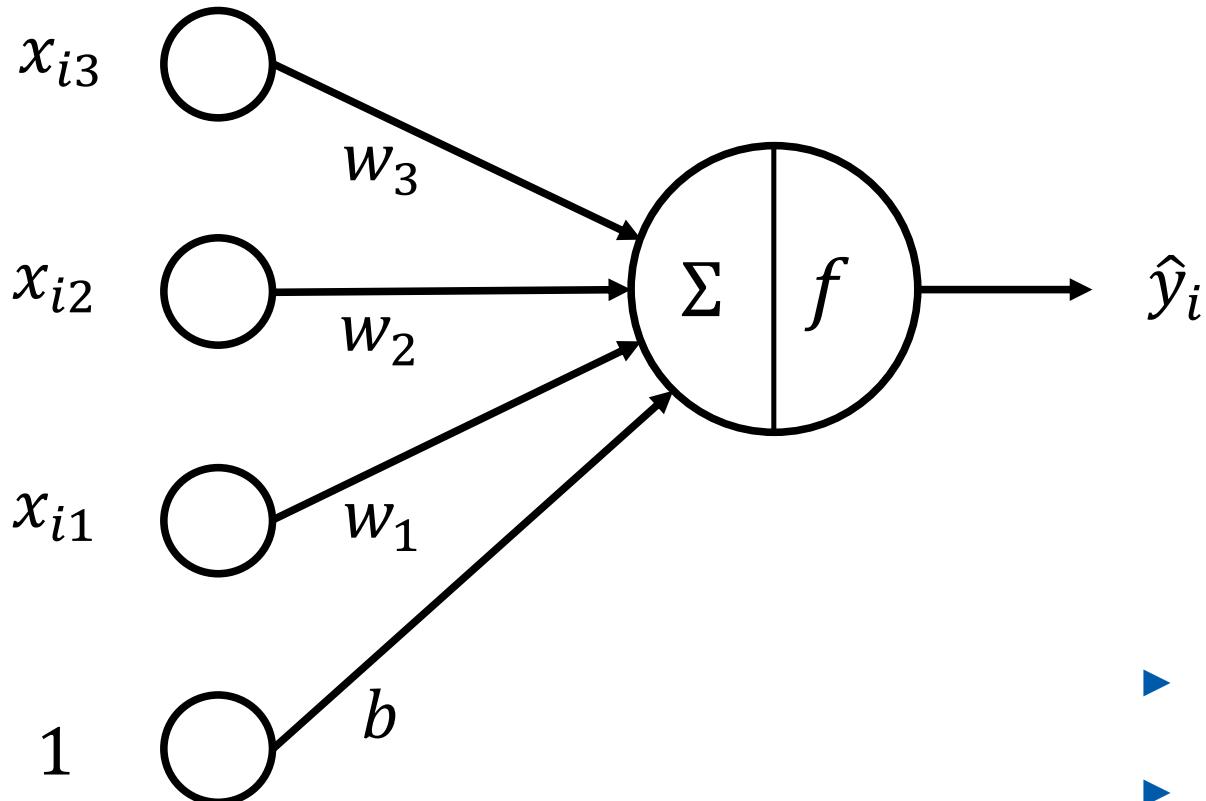
$$\boxed{\frac{\partial L_i}{\partial w_2} = \frac{\partial L_i}{\partial \hat{y}_i} \frac{\partial \hat{y}_i}{\partial z_i} \frac{\partial z_i}{\partial w_2}}$$

$$\frac{\partial L}{\partial w_2} = \sum_{i=1}^N \frac{\partial L_i}{\partial w_2}$$

$$w_2^{(t+1)} = w_2^{(t)} - \alpha \frac{\partial L}{\partial w_2}$$

Линейная модель в общем виде

$$\hat{y}_i = f(x_i^T w + b)$$

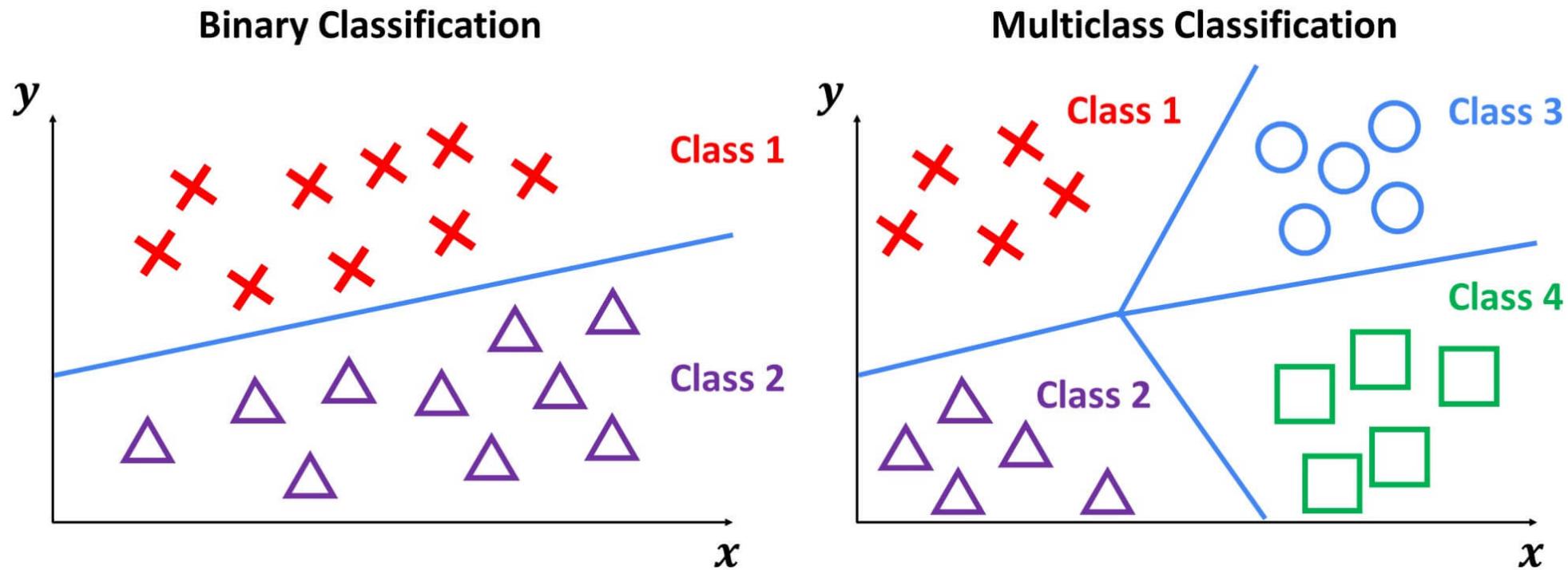


- ▶ $f(z) = z$ – линейная регрессия
- ▶ $f(z) = \sigma(z)$ – логистическая регрессия

Что дальше?

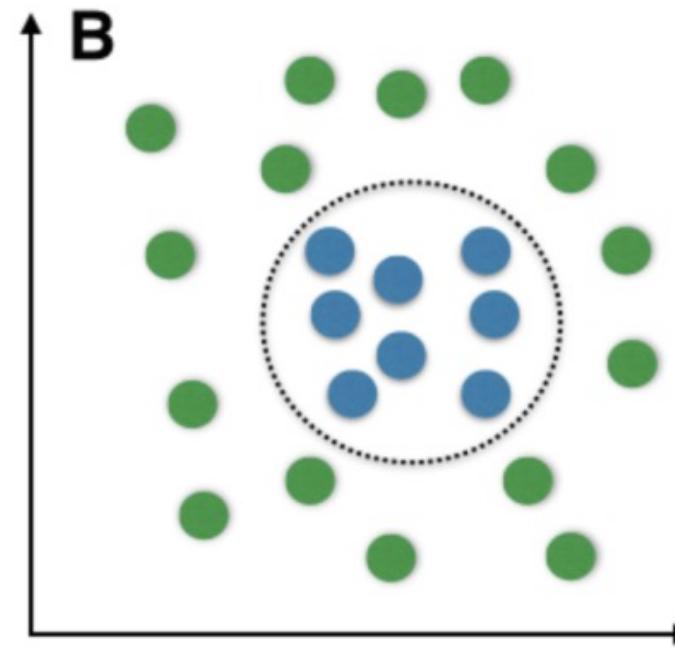
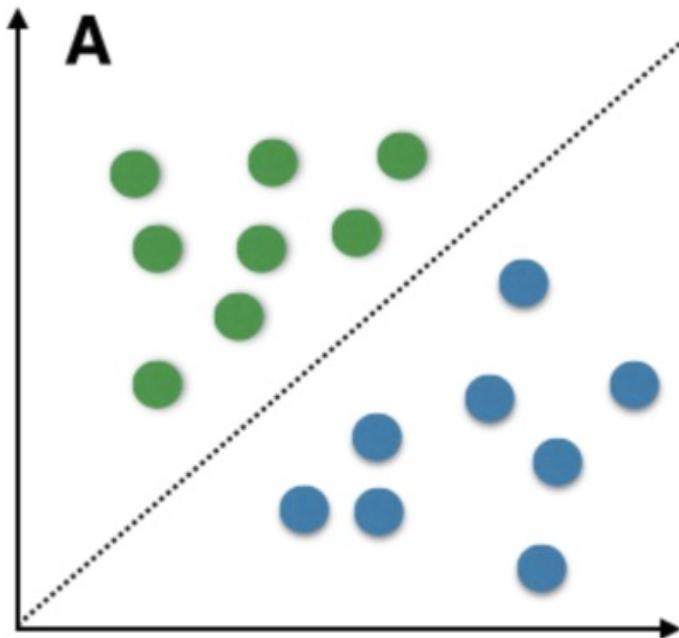
А что дальше?

- ▶ Что делать, если у меня больше, чем два класса в данных?



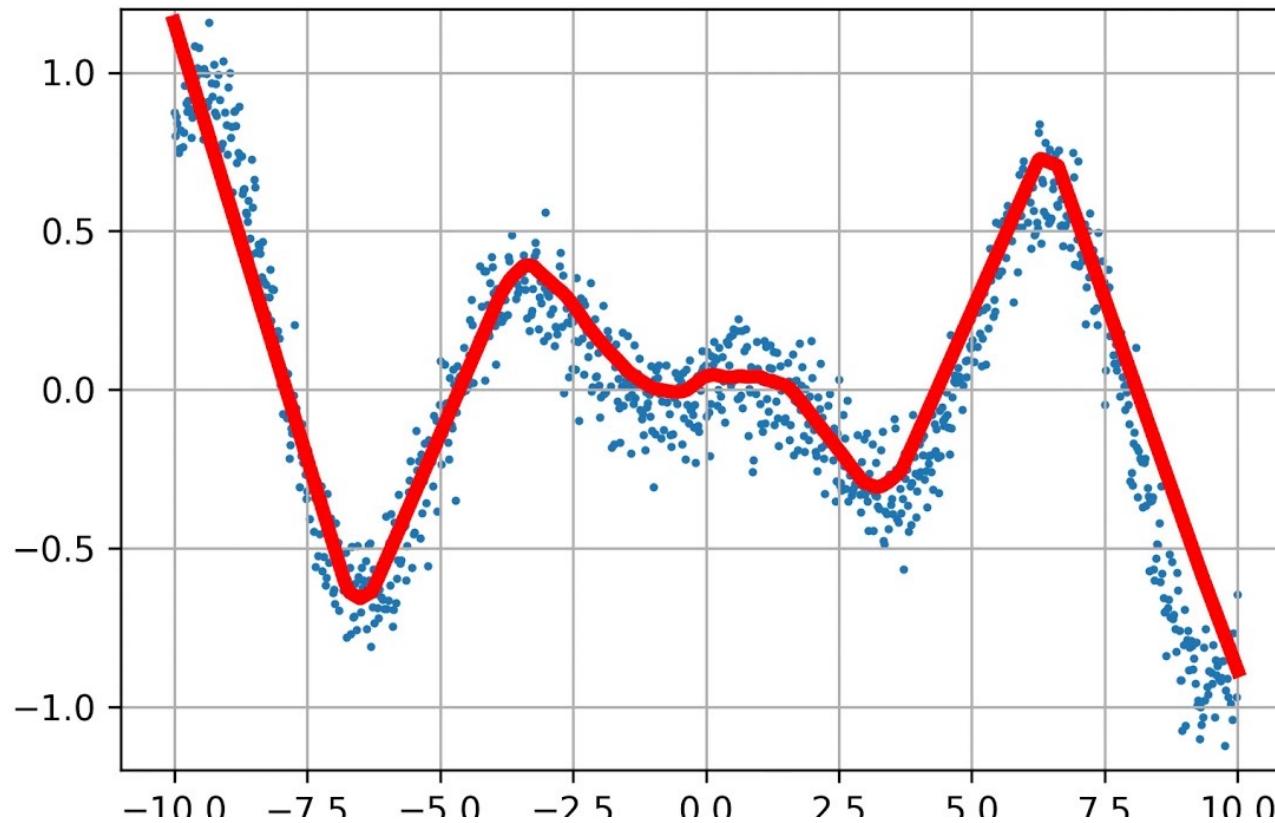
А что дальше?

- ▶ Что делать, если классы не разделяются линейной функцией?



А что дальше?

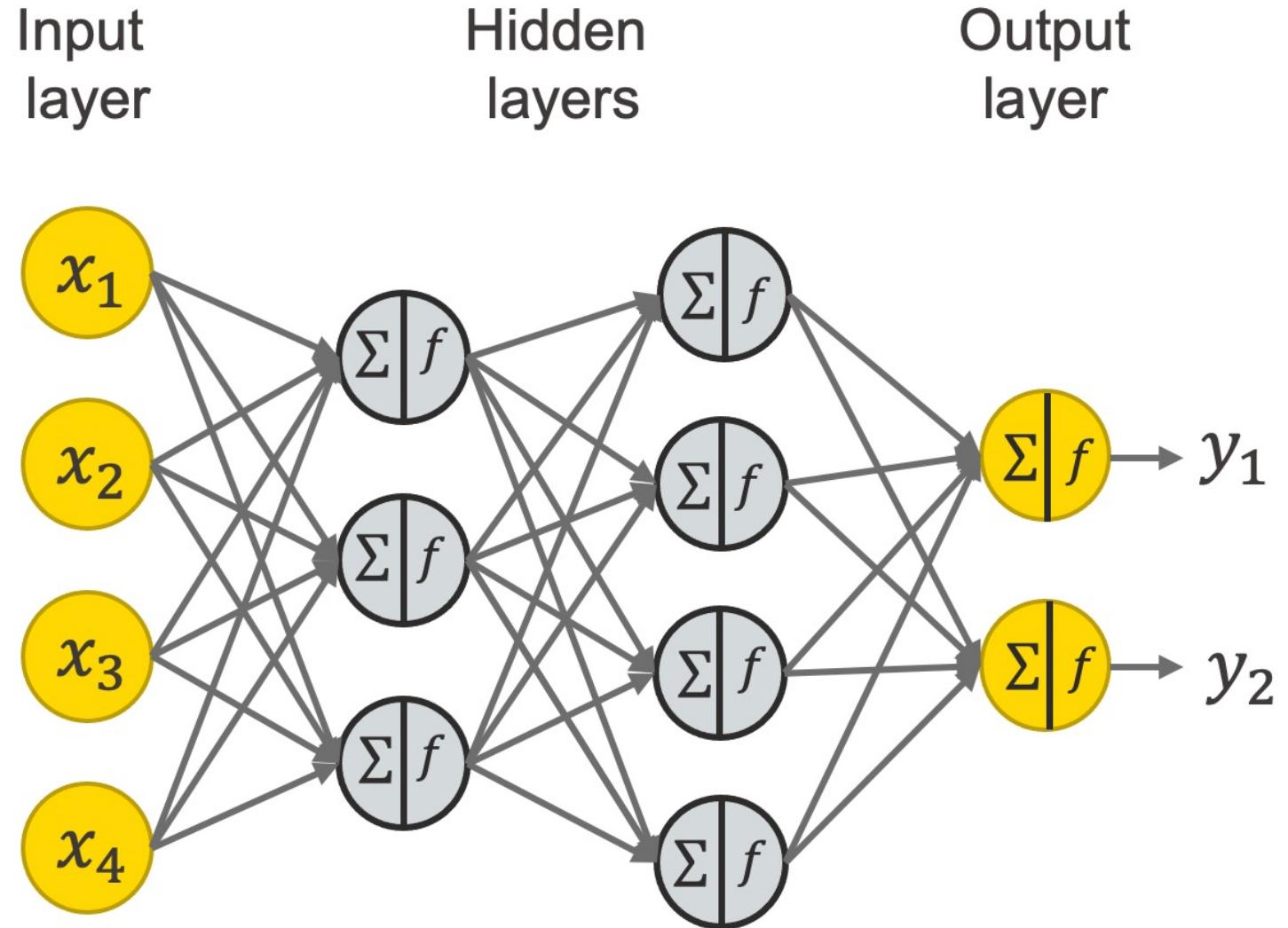
- ▶ Что делать, если у меня нелинейная регрессия?



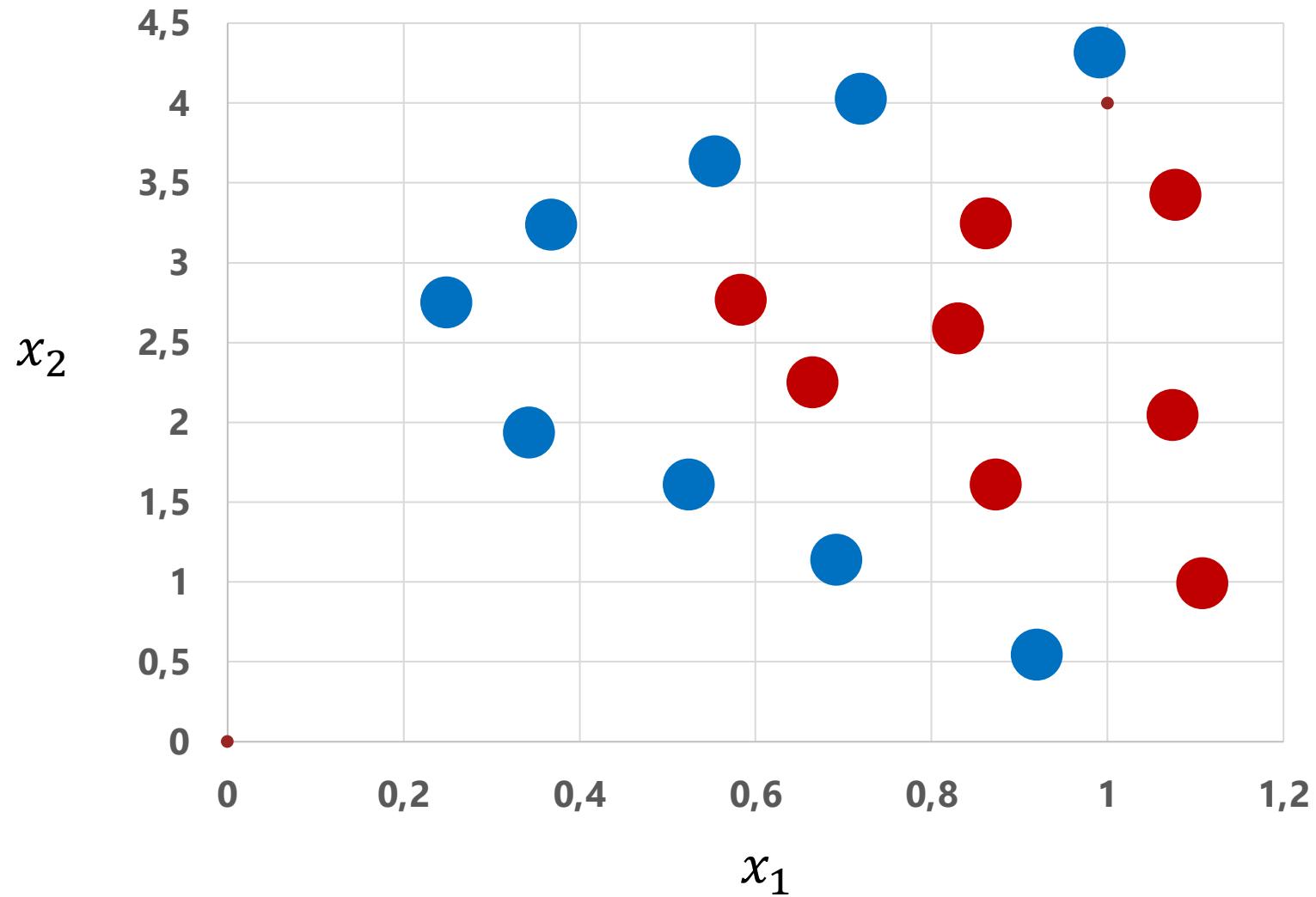
Нейронные сети



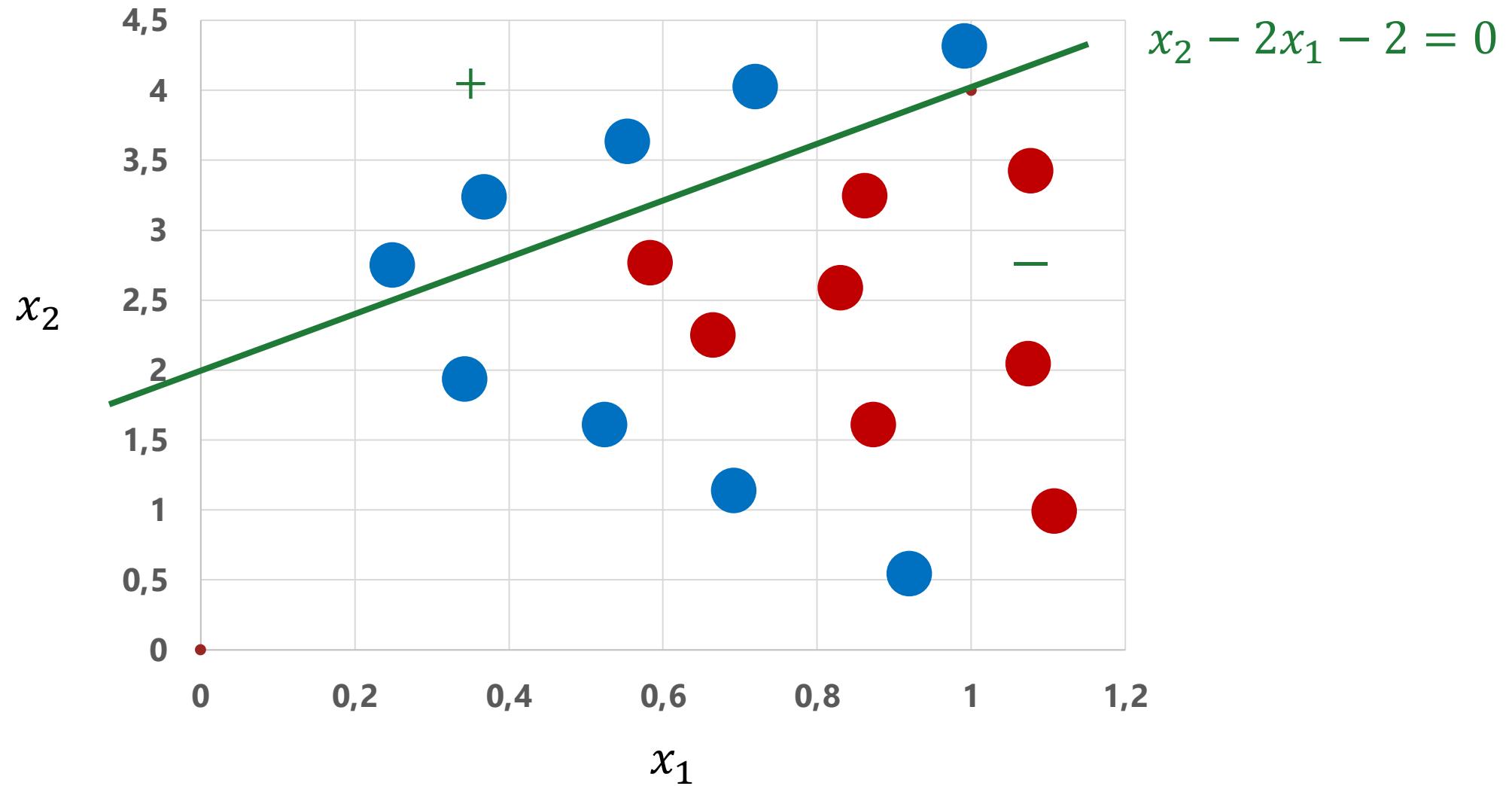
Нейронная сеть



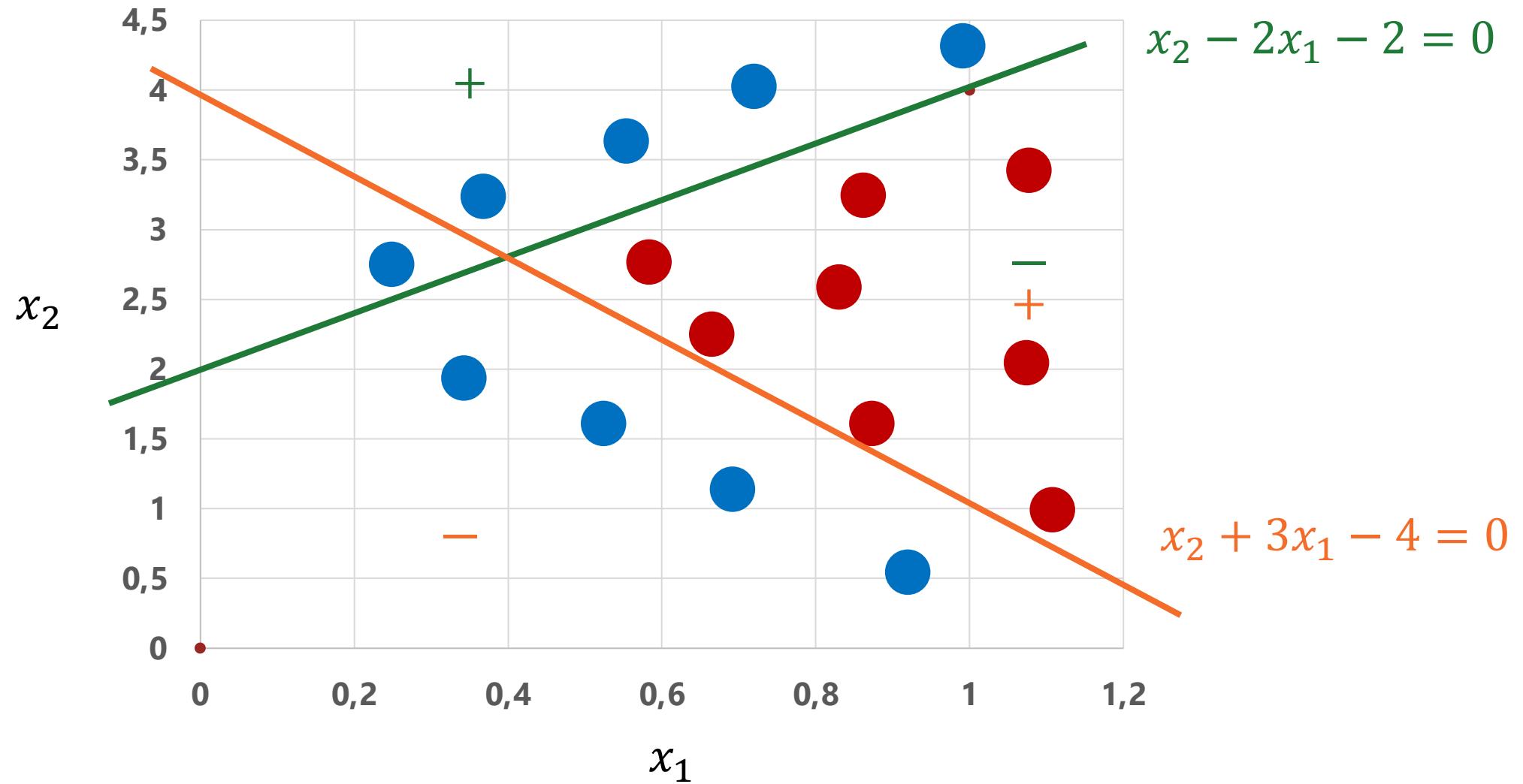
Пример



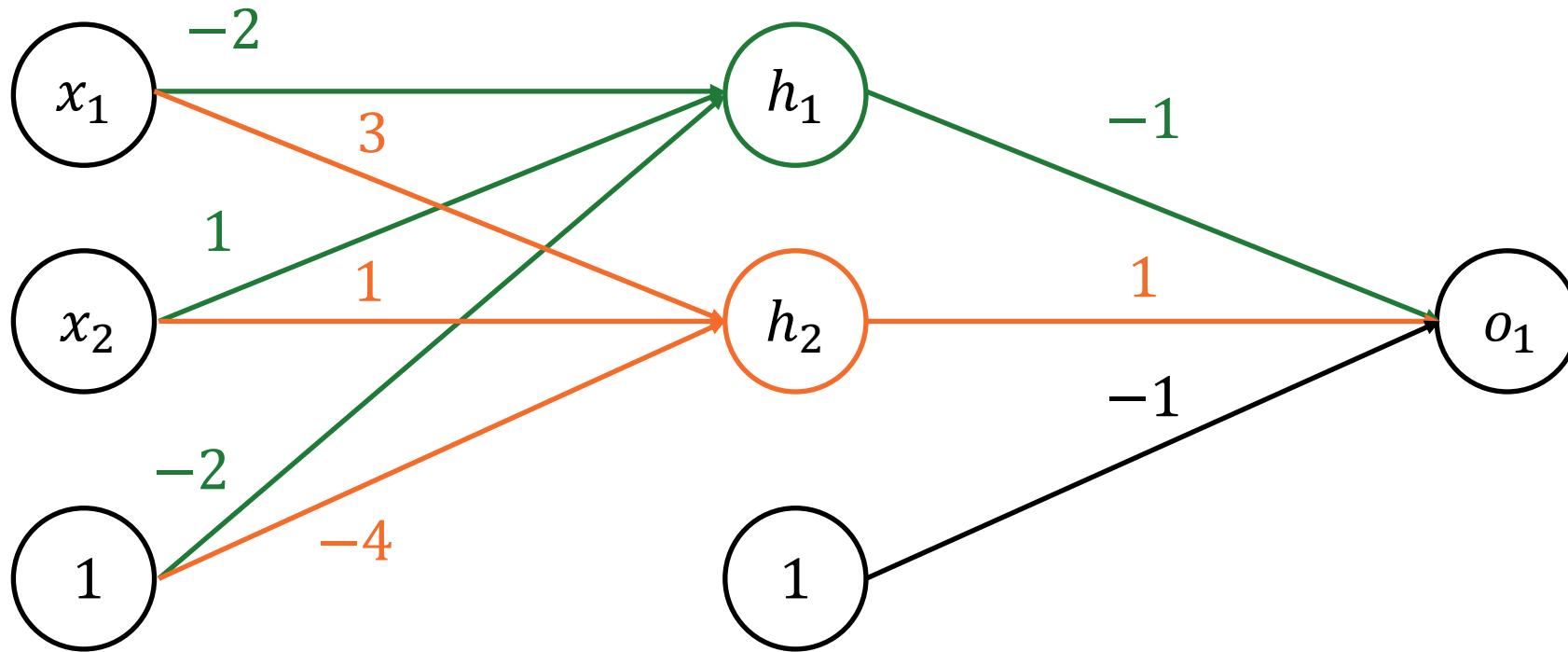
Пример



Пример



Нейронная сеть для примера



$$h_1 = \text{sign}(x_2 - 2x_1 - 2)$$

$$h_2 = \text{sign}(x_2 + 3x_1 - 4)$$

$$o_1 = \text{sign}(h_2 - h_1 - 1)$$

Векторная запись нейронной сети

$$h = f_1(x^T W^{(1)} + b^{(1)})$$

$$o = f_2(h^T W^{(2)} + b^{(2)})$$

$$W^{(1)} = \begin{pmatrix} -2 & 3 \\ 1 & 1 \end{pmatrix}, \quad b^{(1)} = \begin{pmatrix} -2 \\ -4 \end{pmatrix}, \quad f_1(z) = sign(z)$$

$$W^{(2)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \quad b^{(2)} = (-1), \quad f_1(z) = sign(z)$$

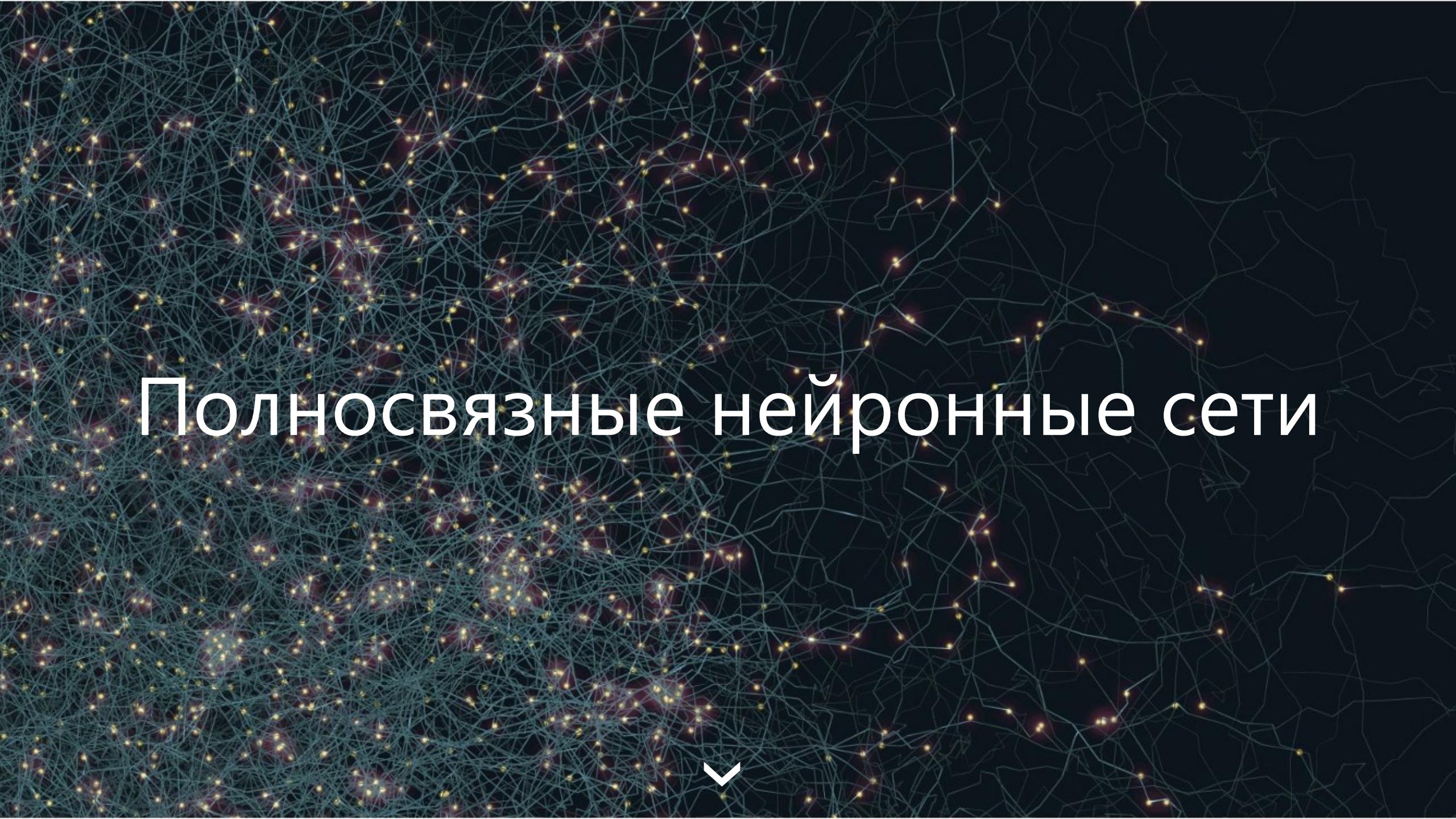
Матричная запись нейронной сети

$$H = f_1(XW^{(1)} + b^{(1)})$$

$$O = f_2(HW^{(2)} + b^{(2)})$$

$$X = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{n1} & x_{n2} \end{pmatrix}, W^{(1)} = \begin{pmatrix} -2 & 3 \\ 1 & 1 \end{pmatrix}, b^{(1)} = (-2 \quad -4), f_1(z) = sign(z)$$

$$H = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{n1} & h_{n2} \end{pmatrix}, W^{(2)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, b^{(2)} = (-1), f_1(z) = sign(z)$$

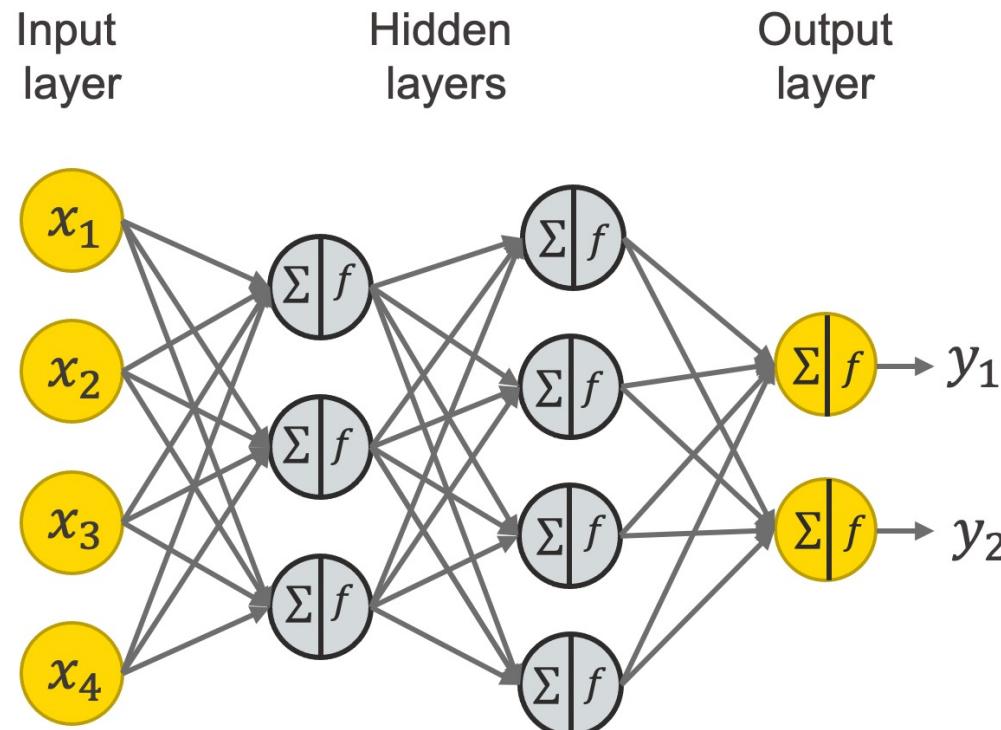


Полносвязные нейронные сети



Нейронная сеть

- ▶ Пусть дан набор наблюдений $\{x_i, y_i\}_{i=1}^n$, где $x_i \in R^d$, $y_i \in R^q$
- ▶ Построим нейронную сеть



Нейронная сеть в матричной форме

- ▶ Матрица наблюдений $X \in R^{n \times d}$ из n объектов, каждый из которых имеет d входных признаков:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

- ▶ Число строк – число наблюдений (объектов)
- ▶ Число столбцов – число входных признаков

Нейронная сеть в матричной форме

$$H^{(1)} = f_1(XW^{(1)} + b^{(1)})$$

$$H^{(2)} = f_2(H^{(1)}W^{(2)} + b^{(2)})$$

$$O = f_3(H^{(2)}W^{(3)} + b^{(3)})$$

Размеры матриц:

- ▶ $X \in R^{n \times d}$, $W^{(1)} \in R^{d \times h}$, $b^{(1)} \in R^{1 \times h}$, h - число нейронов в первом слое
- ▶ $H^{(1)} \in R^{n \times h}$, $W^{(2)} \in R^{h \times m}$, $b^{(2)} \in R^{1 \times m}$, m - число нейронов во втором слое
- ▶ $H^{(2)} \in R^{n \times m}$, $W^{(3)} \in R^{m \times q}$, $b^{(3)} \in R^{1 \times q}$, q - число выходов сети
- ▶ $O \in R^{n \times q}$ - матрица прогнозов сети

Полносвязный слой

$$H^{(1)} = f_1(XW^{(1)} + b^{(1)})$$

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}, \quad W^{(1)} = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1h} \\ w_{21} & w_{22} & \cdots & w_{2h} \\ \vdots & \vdots & \ddots & \vdots \\ w_{d1} & w_{d2} & \cdots & w_{dh} \end{pmatrix}, \quad H^{(1)} = \begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1h} \\ h_{21} & h_{22} & \cdots & h_{2h} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \cdots & h_{nh} \end{pmatrix}$$

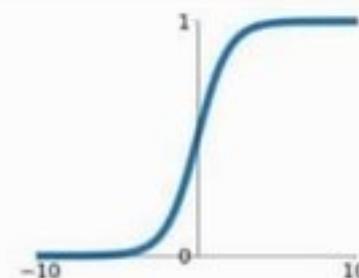
Размеры матриц:

- $X \in R^{n \times d}, W^{(1)} \in R^{d \times h}, b^{(1)} \in R^{1 \times h}, h$ - число нейронов в первом слое
- $H^{(1)} \in R^{n \times h}$ - выход слоя
- $f_1(z)$ - **функция активации**

ФУНКЦИИ АКТИВАЦИИ f

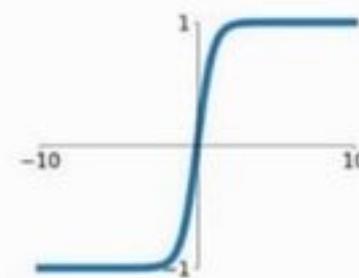
Sigmoid

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



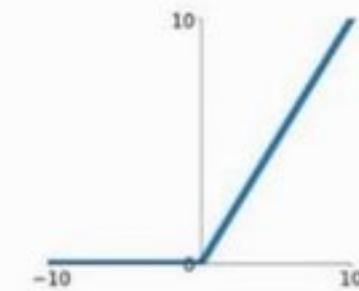
tanh

$$\tanh(x)$$



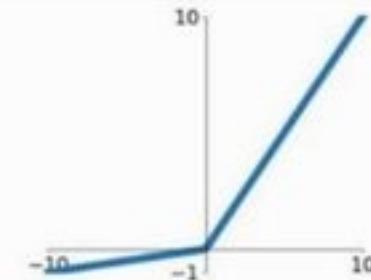
ReLU

$$\max(0, x)$$



Leaky ReLU

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

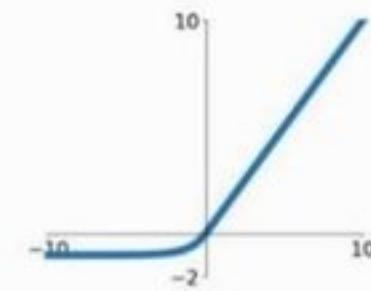


Maxout

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

ELU

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



Функции активации f

- ▶ **ReLU** (Rectified Linear Unit) – наиболее популярная функция активации в современных сетях. Походит для глубоких сетей – сетей с большим числом слоев.
- ▶ **Sigmoid** – использует, когда в сети 1-2 слоя. Также используется в выходном слое в задаче бинарной классификации
- ▶ **Tanh** – хорошая альтернатива sigmoid. Используется в скрытых слоях.

Вопрос

- ▶ Зачем использовать функции активации?
- ▶ Что будет, если их не использовать?

Ответ

- ▶ Пусть дана нейронная сеть без функций активации:

$$H^{(1)} = XW^{(1)} + b^{(1)}$$

$$O = H^{(1)}W^{(2)} + b^{(2)}$$

- ▶ Перепишем:

$$O = (XW^{(1)} + b^{(1)})W^{(2)} + b^{(2)} = X\textcolor{blue}{W}^{(1)}W^{(2)} + (\textcolor{green}{b}^{(1)}W^{(2)} + b^{(2)})$$

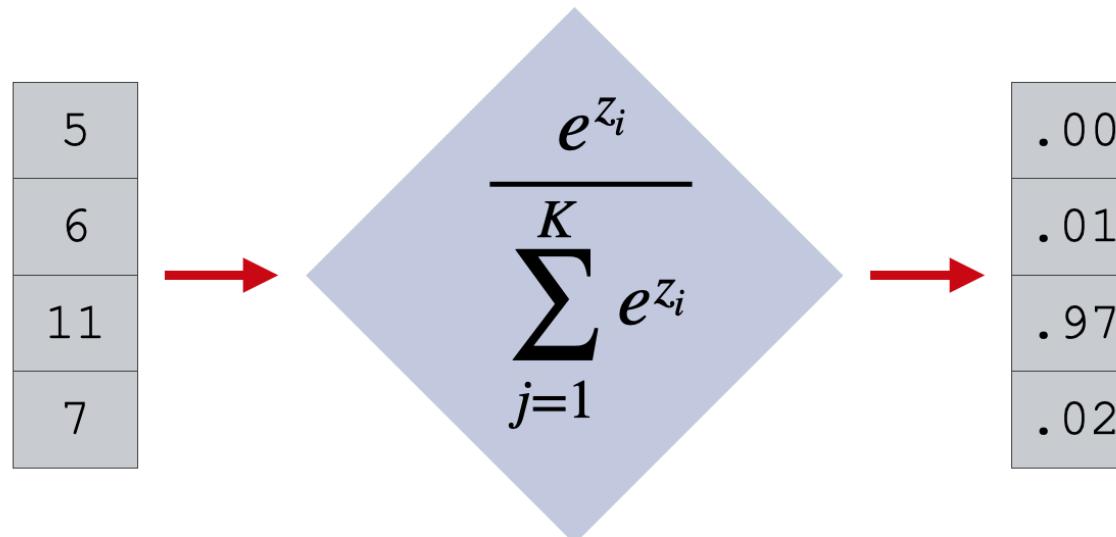
$$O = X\textcolor{blue}{W} + \textcolor{green}{b}$$

- ▶ В итоге получаем линейную зависимость

Функции активации softmax

- ▶ Дан вектор z
- ▶ Хотим получить вектор вероятностей p , чтобы сумма равнялась 1.
- ▶ Используем функцию softmax:

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

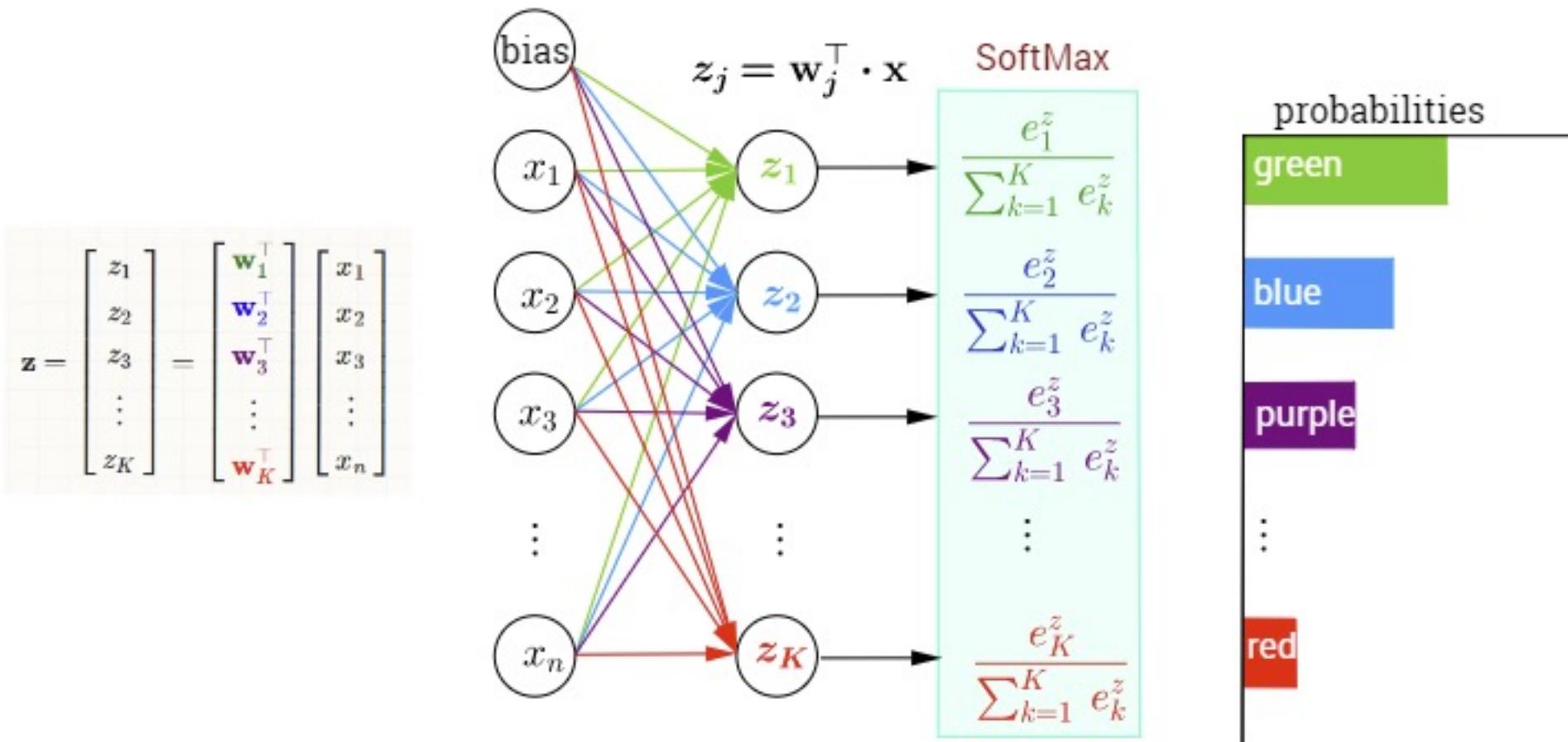


Функции активации softmax

- ▶ **Softmax** используется в выходном слое нейронной сети для решения задач многоклассовой классификации
- ▶ Выход функции можно интерпретировать как «вероятность» класса

ФУНКЦИИ АКТИВАЦИИ softmax

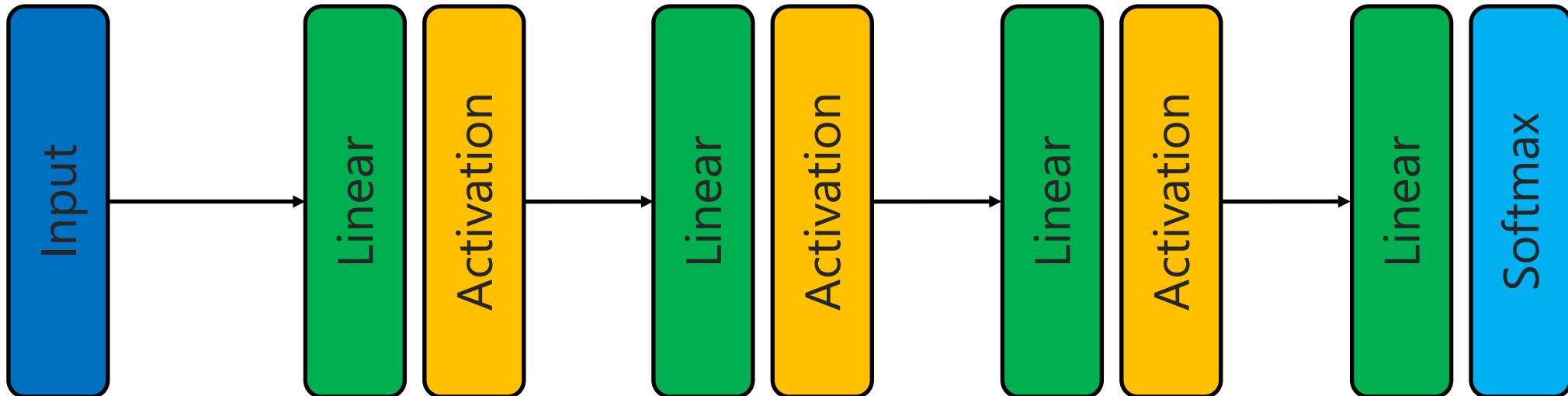
Multi-Class Classification with NN and SoftMax Function



Источник: <https://rinterested.github.io/statistics/softmax.html>

Архитектура нейронной сети

- ▶ Последовательность линейных слоев и функций активации
- ▶ Обычно, во всех скрытых слоях используется одна функция активации
- ▶ На выходе softmax или sigmoid для классификации, линейная функция для регрессии



Теорема Цыбенко

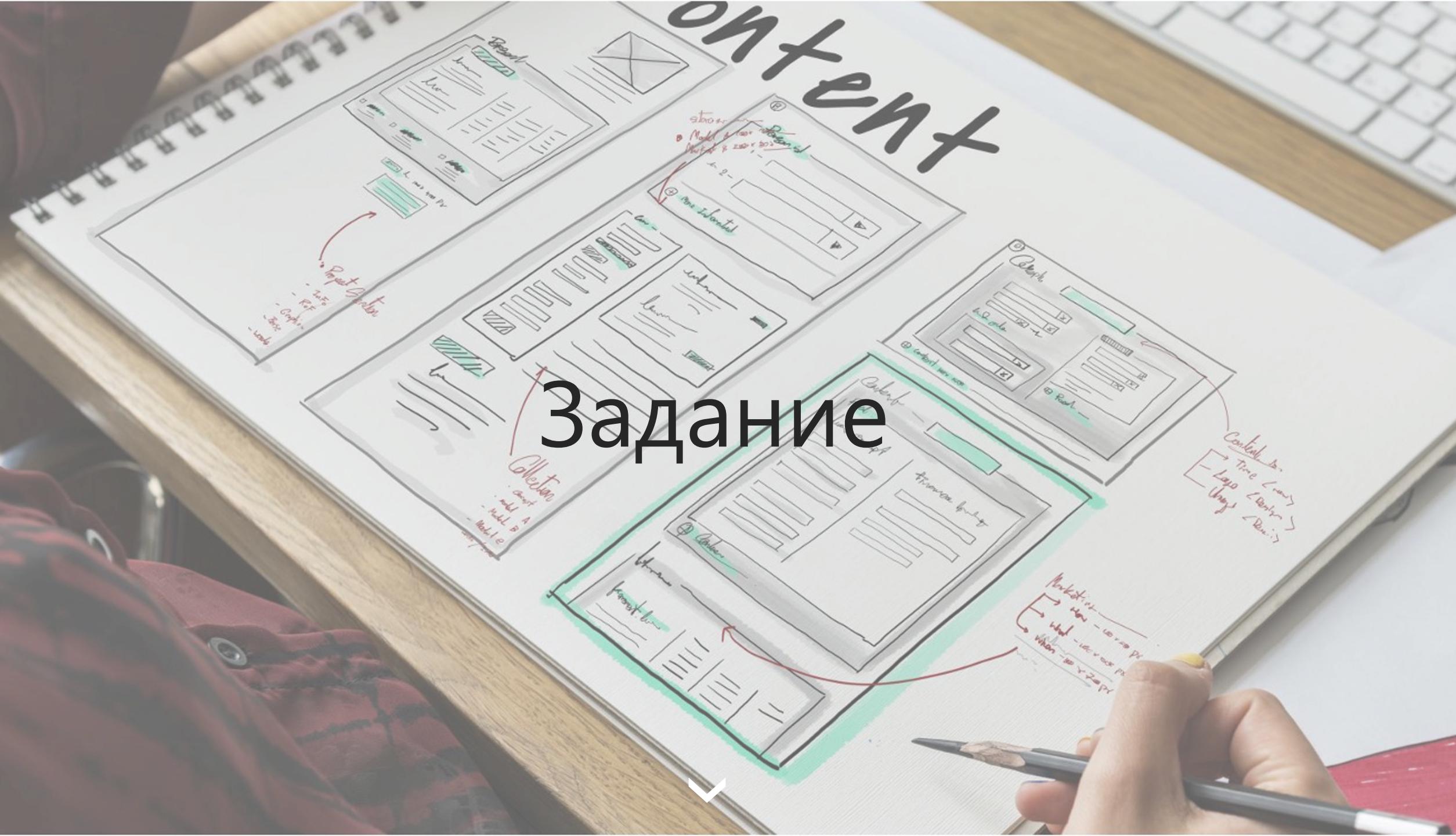
- ▶ Пусть дана нейронная сеть с одним скрытым слоем
- ▶ В слое достаточно много нейронов
- ▶ Веса нейронов подобраны оптимально

Теорема Цыбенко утверждает, что такой сети достаточно для моделирования **любой функции**

Правда, обучить такую сеть может быть тяжело 😊

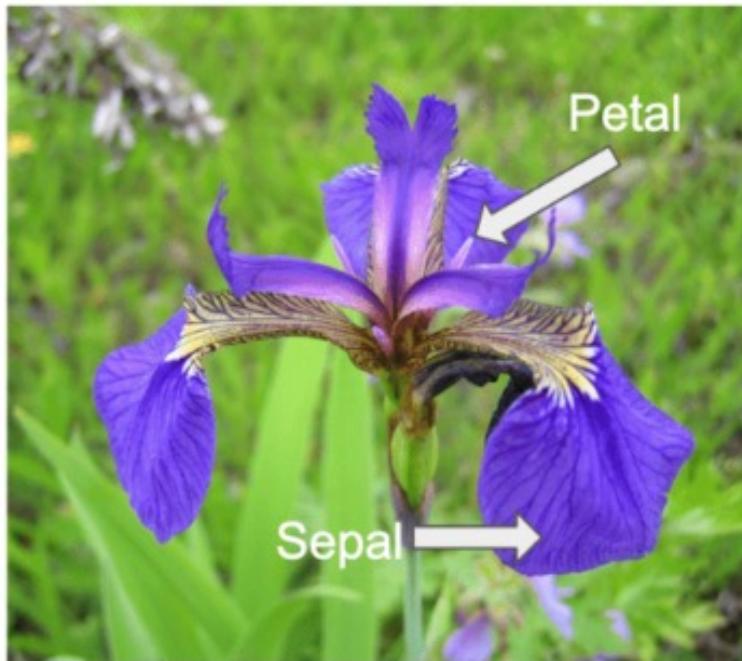
content

Задание

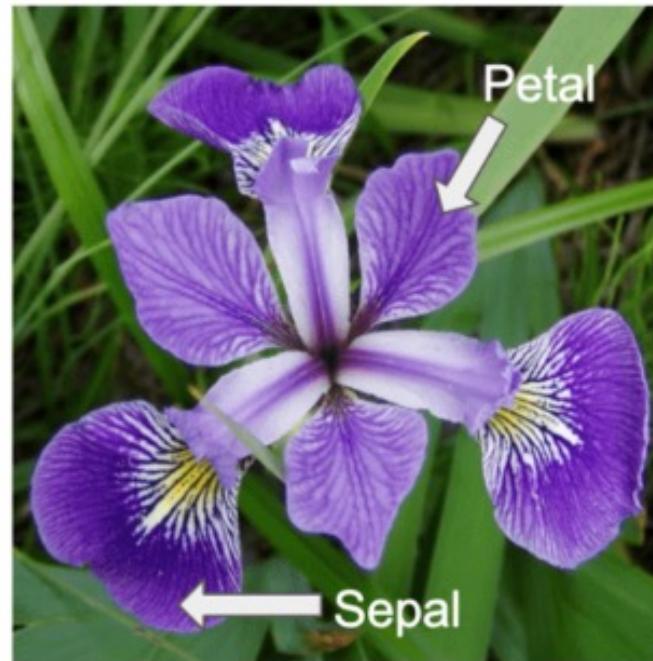


Классификация ирисов

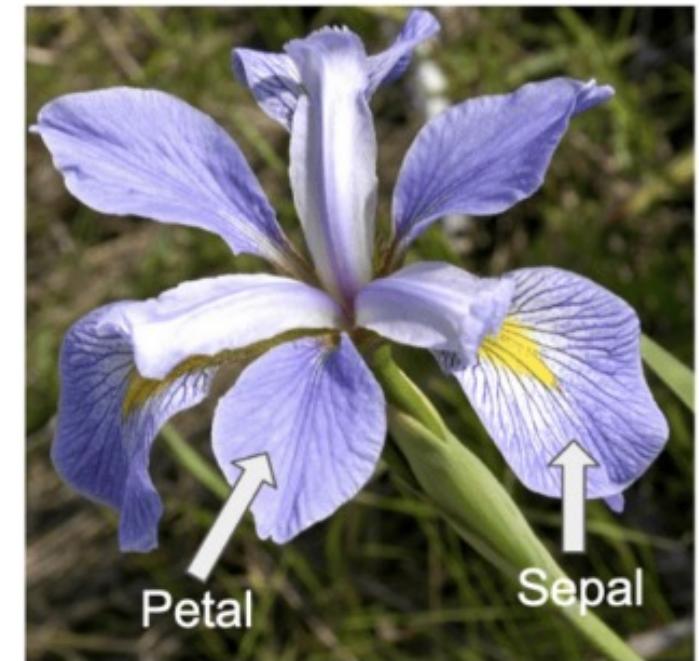
Iris setosa



Iris versicolor



Iris virginica

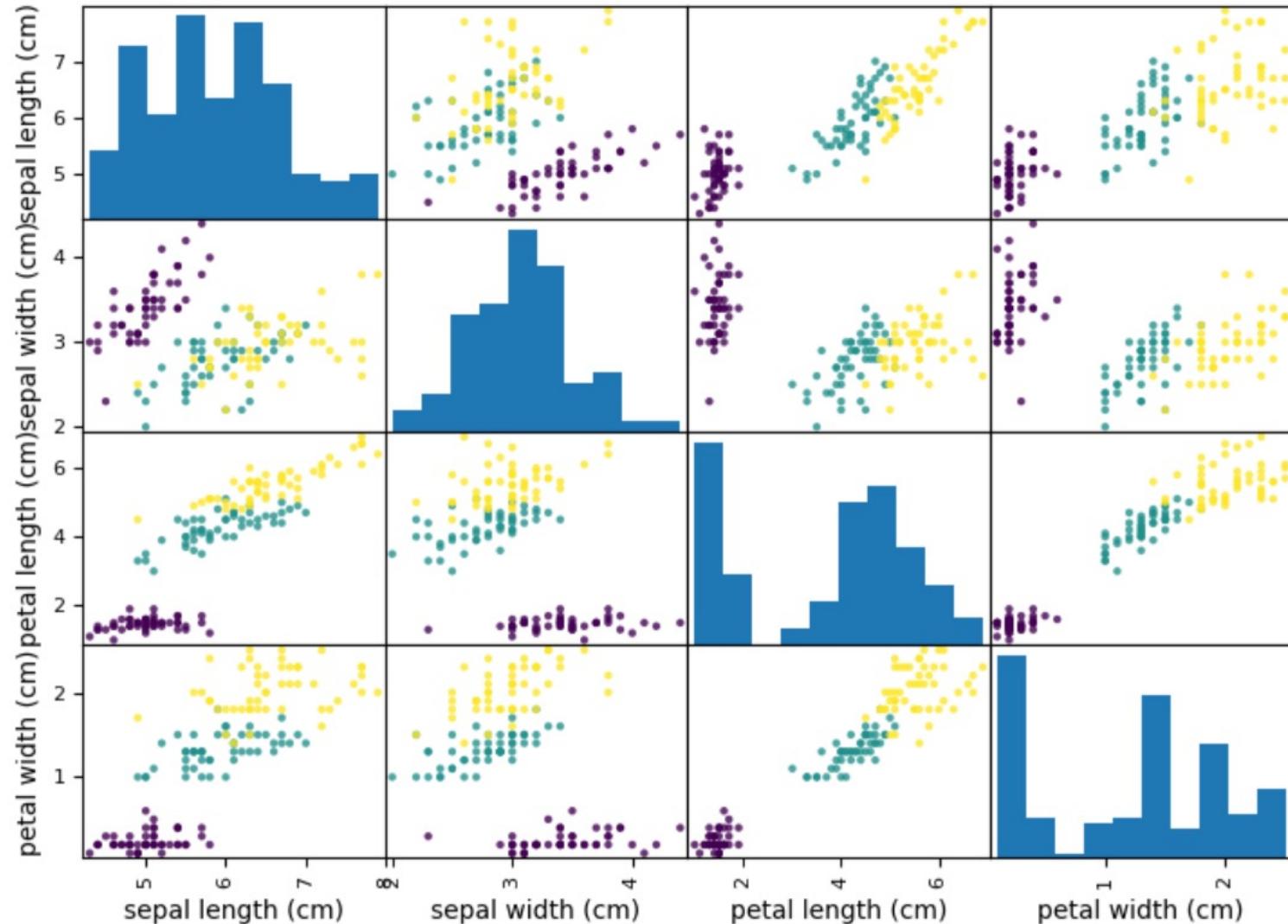


Классификация ирисов

Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa



Классификация ирисов

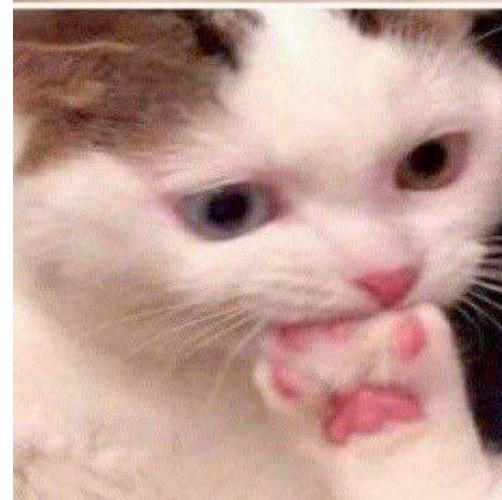


Задание

- ▶ Задача: классификация ирисов на три класса
- ▶ Запишите нейронную сеть для решения этой задачи. В сети должно быть три скрытых слоя по 200, 100, 50 нейронов соответственно.
- ▶ Какие функции активации лучше использовать?

Данные: существуют

Аналитики:



Решение

$$H^{(1)} = f_1(XW^{(1)} + b^{(1)})$$

$$H^{(2)} = f_2(H^{(1)}W^{(2)} + b^{(2)})$$

$$H^{(3)} = f_3(H^{(2)}W^{(3)} + b^{(3)})$$

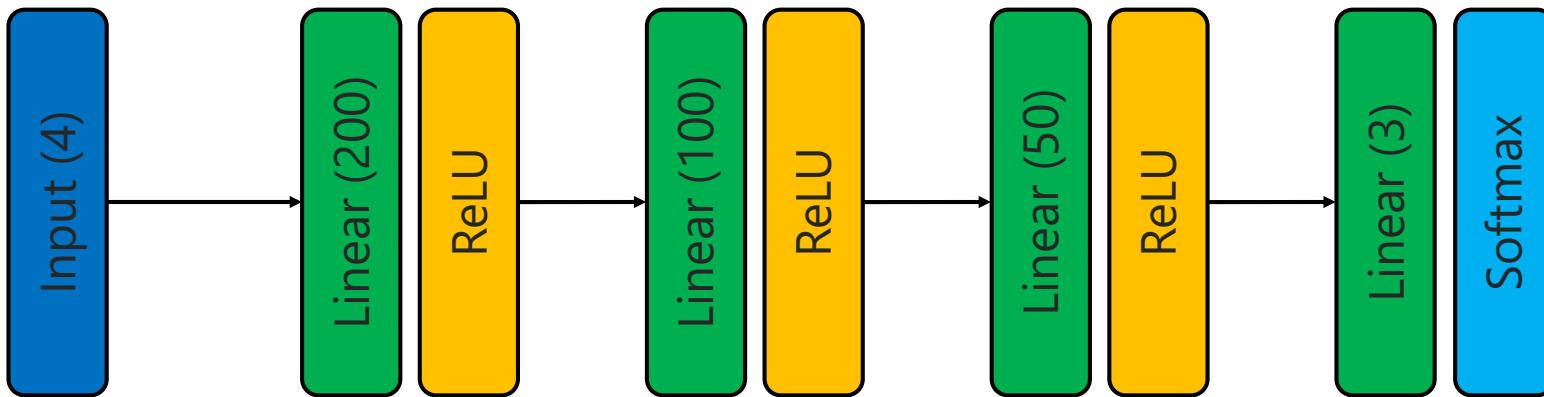
$$O = f_4(H^{(3)}W^{(4)} + b^{(4)})$$

Размеры матриц:

- ▶ $X \in R^{150 \times 4}, W^{(1)} \in R^{4 \times 200}, b^{(1)} \in R^{1 \times 200}, f_1 - \text{ReLU}()$
- ▶ $H^{(1)} \in R^{150 \times 200}, W^{(2)} \in R^{200 \times 100}, b^{(2)} \in R^{1 \times 100}, f_2 - \text{ReLU}()$
- ▶ $H^{(2)} \in R^{150 \times 100}, W^{(3)} \in R^{100 \times 50}, b^{(3)} \in R^{1 \times 50}, f_3 - \text{ReLU}()$
- ▶ $H^{(3)} \in R^{150 \times 50}, W^{(4)} \in R^{50 \times 3}, b^{(4)} \in R^{1 \times 3}, f_4 - \text{Softmax}()$
- ▶ $O \in R^{150 \times 3}$

Архитектура нейронной сети

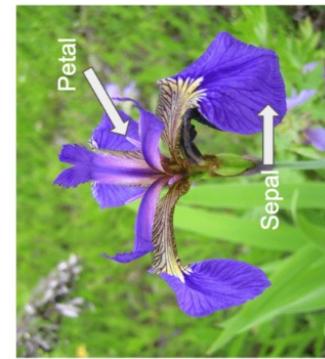
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
5.1	3.5	1.4	0.2	
4.9	3	1.4	0.2	
4.7	3.2	1.3	0.2	
4.6	3.1	1.5	0.2	
5	3.6	1.4	0.2	
5.4	3.9	1.7	0.4	
4.6	3.4	1.4	0.3	
5	3.4	1.5	0.2	
4.4	2.9	1.4	0.2	
4.9	3.1	1.5	0.1	
5.4	3.7	1.5	0.2	
4.8	3.4	1.6	0.2	



Iris virginica

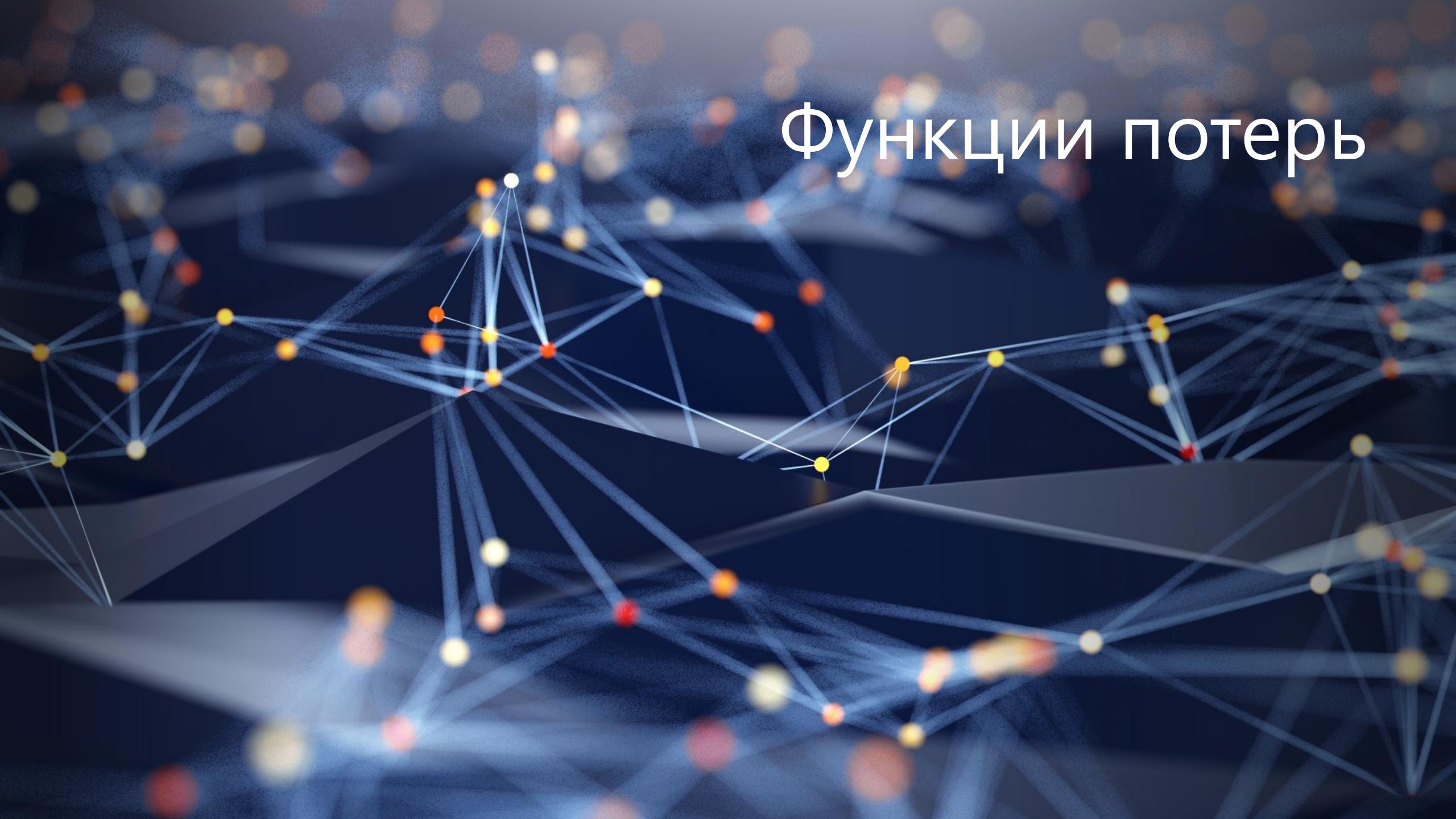


Iris versicolor



Iris setosa

Функции потерь



Функция потерь для классификации

- ▶ Функция потерь для 2 классов, где $y_i \in \{0, 1\}$:

$$L = -\frac{1}{N} \sum_{i=1}^N y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

- ▶ Функция потерь для K классов, где $y_i \in \{0, 1, 2, \dots, K\}$:

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K [y_i == k] \log \hat{y}_{ik}$$

\hat{y}_{ik} - прогноз вероятности для класса k

Функция потерь для классификации

Предсказания нейронной сети			
	пес	кот	жук
1	0.7	0.1	0.2
2	0.8	0.15	0.05
3	0.04	0.9	0.06
4	0.6	0.1	0.3
5	0.75	0.2	0.05

Правильные ответы Y	
	Y
1	кот
2	жук
3	кот
4	пес
5	кот

Кросс-энтропийный функционал

$$Q(w) = - \sum_{n=1}^N \log P(y_n | f(x_n, w))$$

$$\begin{aligned} & -\log 0.1 - \log 0.05 - \log \\ & 0.9 - \log 0.6 - \log 0.2 \end{aligned}$$

ФУНКЦИИ ПОТЕРЬ

- ▶ Классификация на 2 класса:

$$L = -\frac{1}{N} \sum_{i=1}^N y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

- ▶ Классификация на K классов, где $y_i \in \{0, 1, 2, \dots, K\}$:

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K [y_i == k] \log \hat{y}_{ik}$$

- ▶ Регрессия для y_i любой размерности:

$$L = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K (\hat{y}_{ik} - y_{ik})^2$$

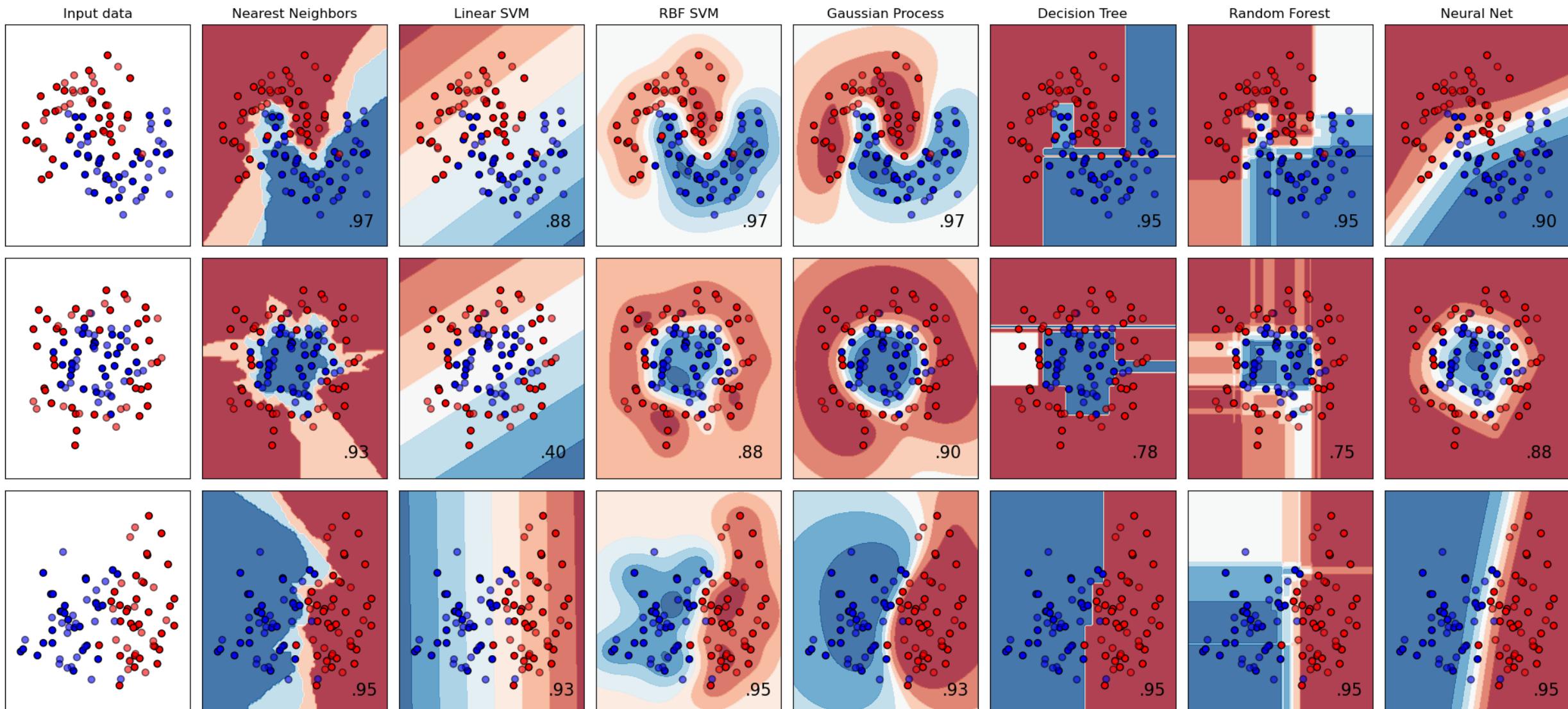
Градиентный спуск

- ▶ Нейронные сети обучаем с помощью градиентного спуска

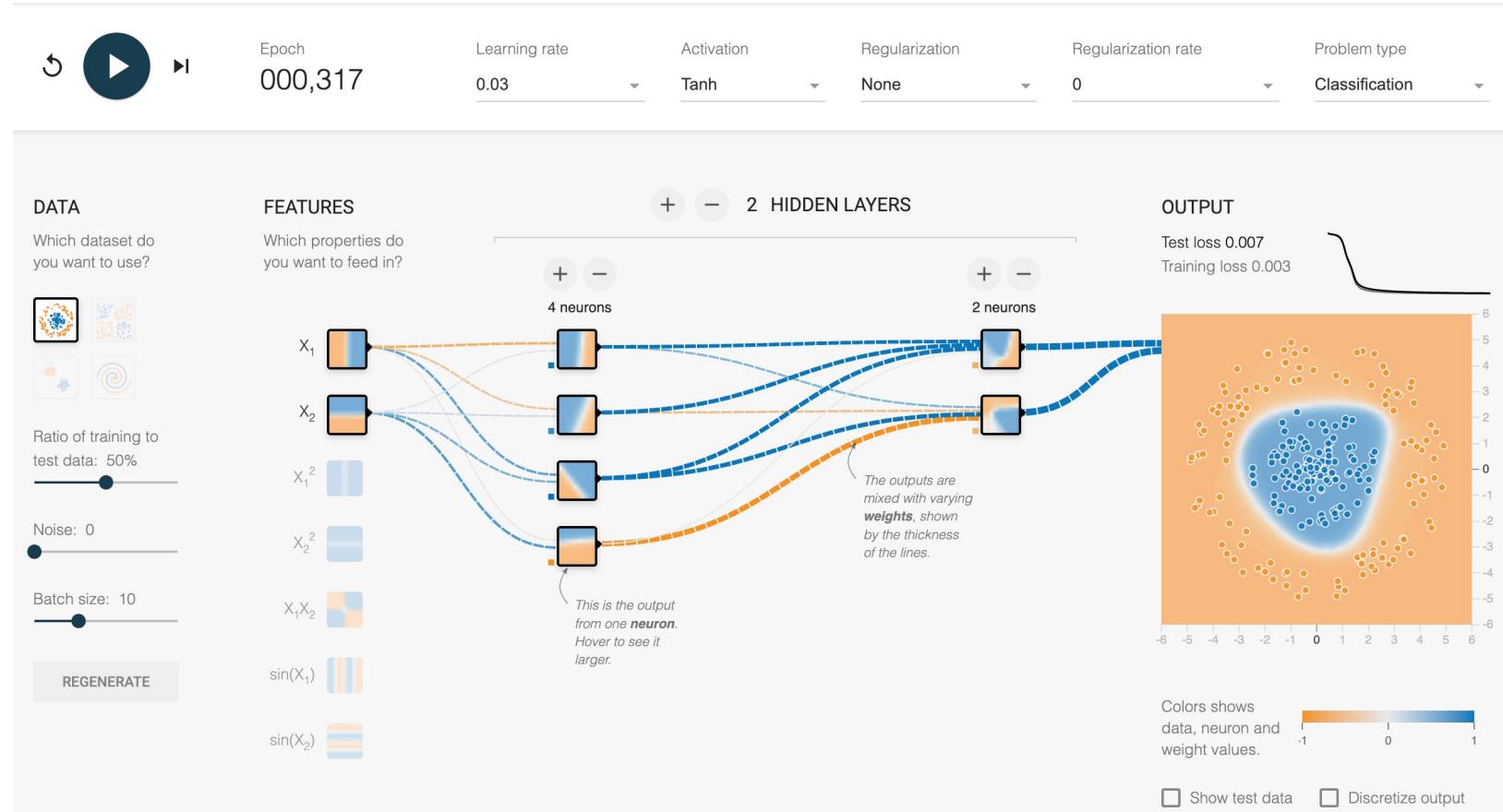
$$w = w - \alpha \frac{\partial L}{\partial w}$$

- ▶ Как посчитать градиент функции потерь по нужному весу сети?
- ▶ Про это поговорим на следующей лекции

Примеры



Демонстрация



<https://playground.tensorflow.org>

Векторизация текстов

Задача классификации текстов

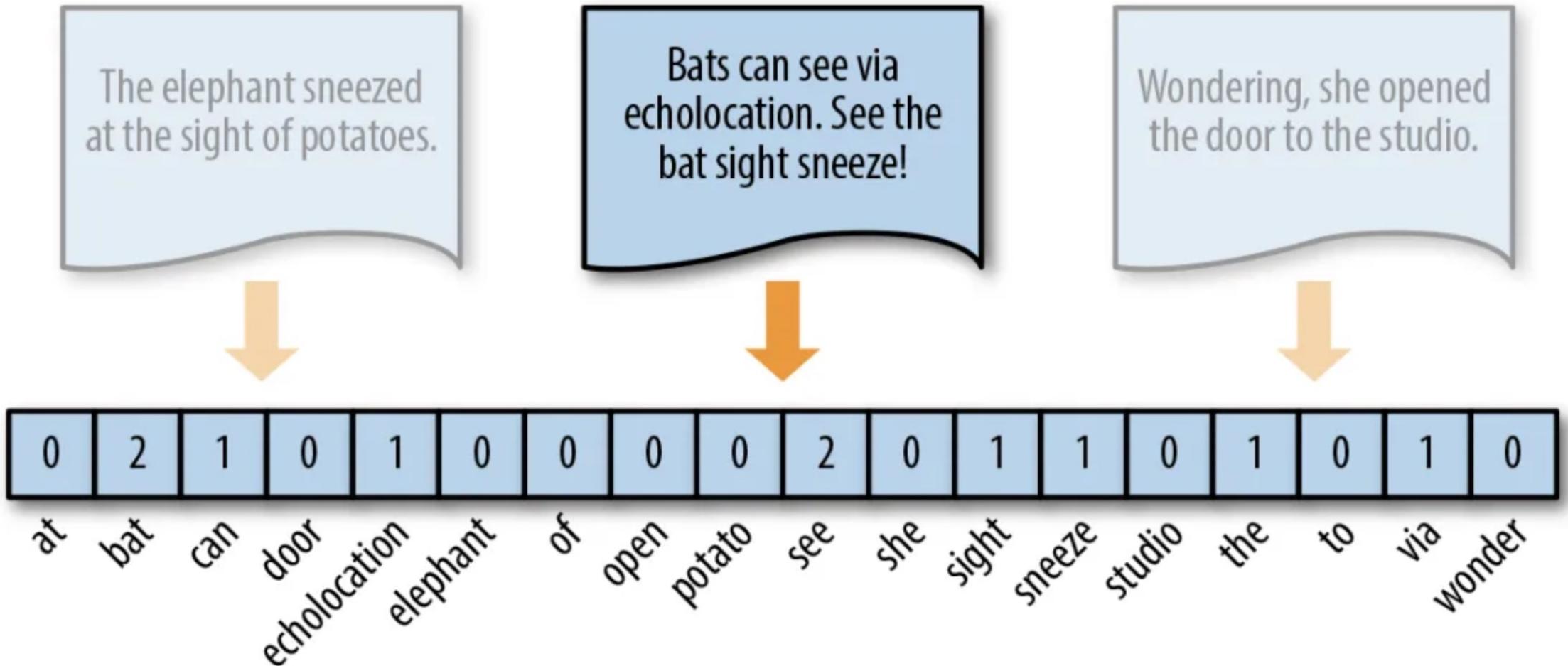
"text": "Двое налетчиков совершили нападение на охранника банка \"ЦентрКредит\" в Алматы и завладели его оружием, сообщает пресс-служба ДВД Алматы.\n\"Сегодня, в 08.10 в Центр оперативного управления ДВД города Алматы поступило сообщение о нападении на охранника одного из отделений банка \"Центркредит\" в Алмалинском районе Алматы, в результате чего двое неустановленных лиц завладели его оружием\", — говорится в сообщении.\nПо неподтвержденным данным, чрезвычайное происшествие случилось в отделении, которое находится на улице Маметовой, между улицами Сейфуллина и Дзержинского.\nКак информирует ведомство, в настоящее время по городу объявлен спецплан \"Сирена\". Отрабатывается комплекс оперативных мероприятий, которые направлены на розыск и задержание преступников.\n",
"sentiment": "negative»

"text": "АСТАНА. КАЗИНФОРМ - Карим Масимов провел совещание по вопросам деятельности Актауского международного морского торгового порта. Read on the original site", "id": 2085,
"sentiment": "positive"

Мешок слов (Bag of Words)

	the	red	dog	cat	eats	food
1. the red dog →	1	1	1	0	0	0
2. cat eats dog →	0	0	1	1	1	0
3. dog eats food →	0	0	1	0	1	1
4. red cat eats →	0	1	0	1	1	0

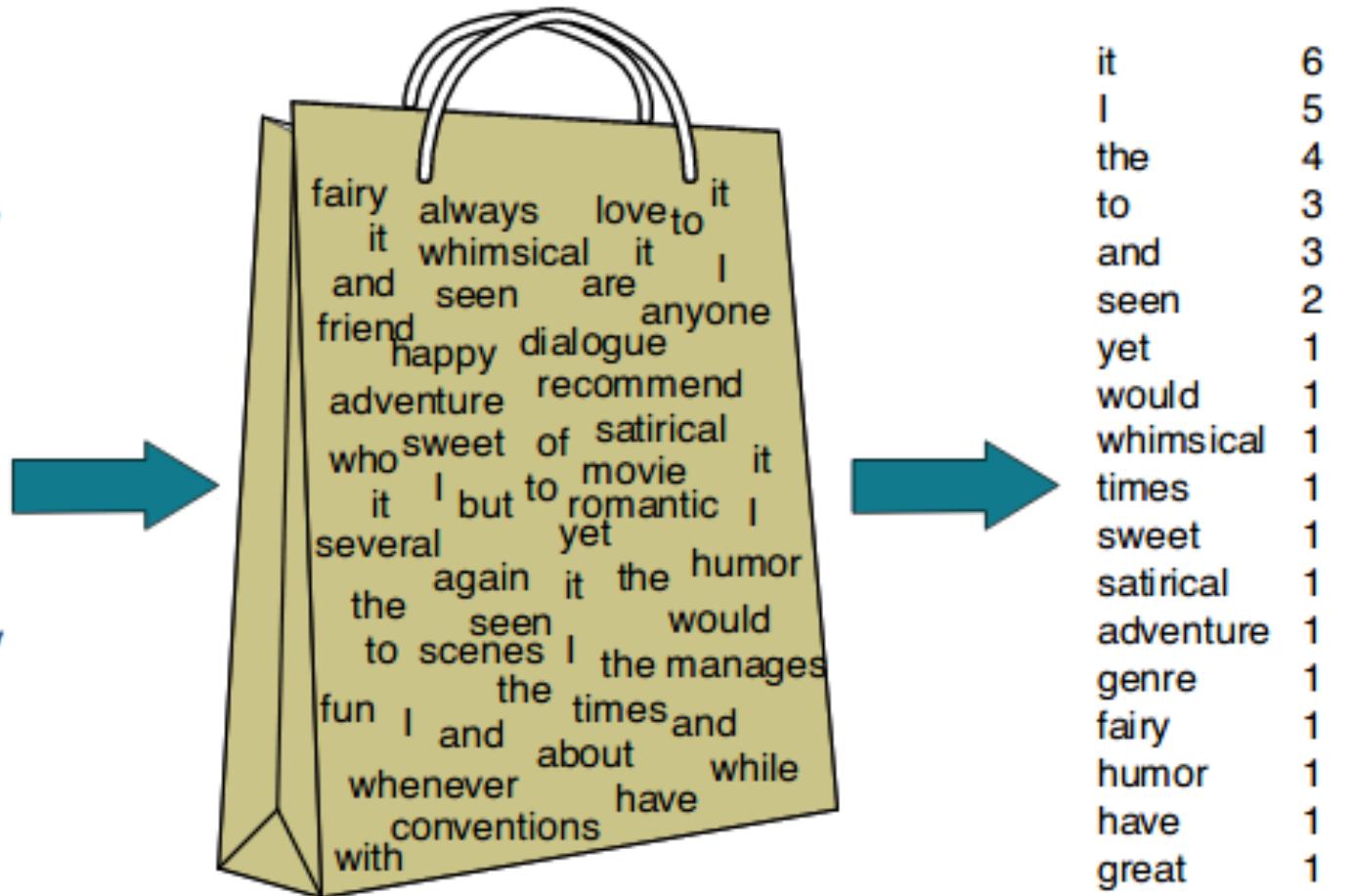
Мешок слов (Bag of Words)



Источник: <https://towardsdatascience.com/from-word-embeddings-to-pretrained-language-models-a-new-age-in-nlp-part-1-7ed0c7f3dfc5>

Мешок слов (Bag of Words)

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



Источник: <https://dudeperf3ct.github.io/lstm/gru/nlp/2019/01/28/Force-of-LSTM-and-GRU/>

Алгоритм решения

- ▶ Пусть есть набор текстов
- ▶ Делим каждый текст на токены и приводим слова к нормальной форме
- ▶ Используем Bag Of Words или TF-IDF, чтобы получить векторное представление каждого текста
- ▶ Используем эти вектора как вектора признаков
- ▶ Используем любые модели классификации и регрессии для анализа текстов

Мешок слов (Bag of Words)

- ▶ Слишком много признаков
- ▶ Похожие по смыслу тексты могут иметь разные представления
- ▶ Можно ли иначе получить векторные представления слов?

The background of the image features a complex, abstract pattern resembling a topographic map or a microscopic view of a material. It is dominated by dark purple and blue colors, with numerous thin, light-colored lines forming intricate, winding paths and ridges across the entire surface.

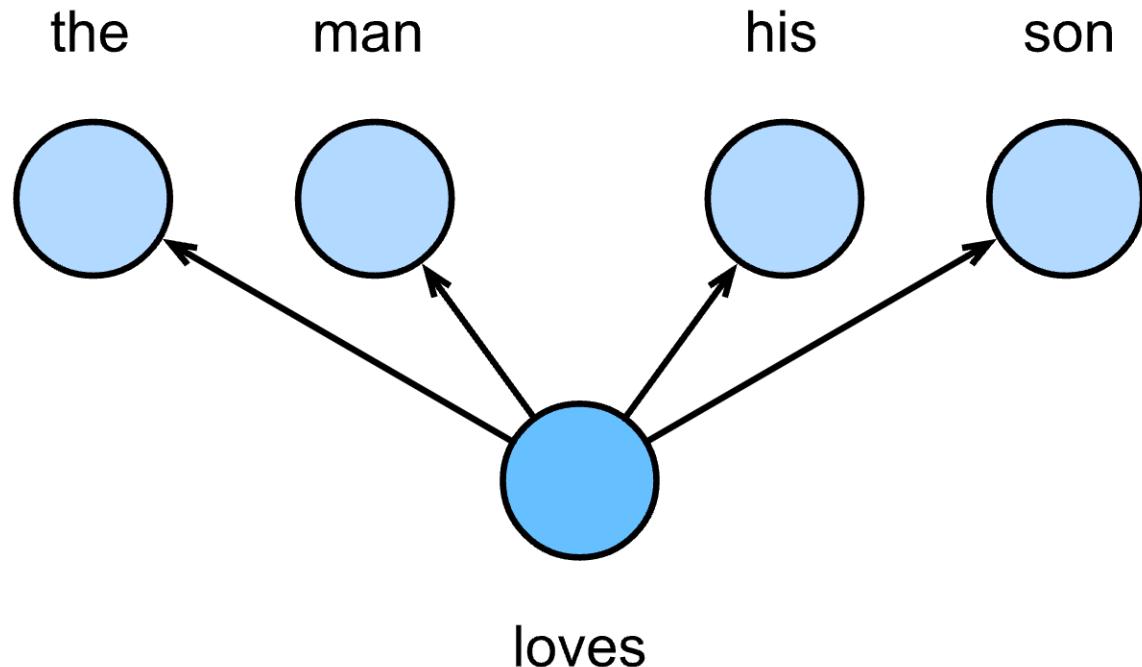
word2vec

Пример

The man **loves** his son

- ▶ Назовем "loves" – центральным словом
- ▶ "the", "man", "his", "son" – контекстом
- ▶ Word2vec предполагает, что вектора центрального слова и контекстных близки

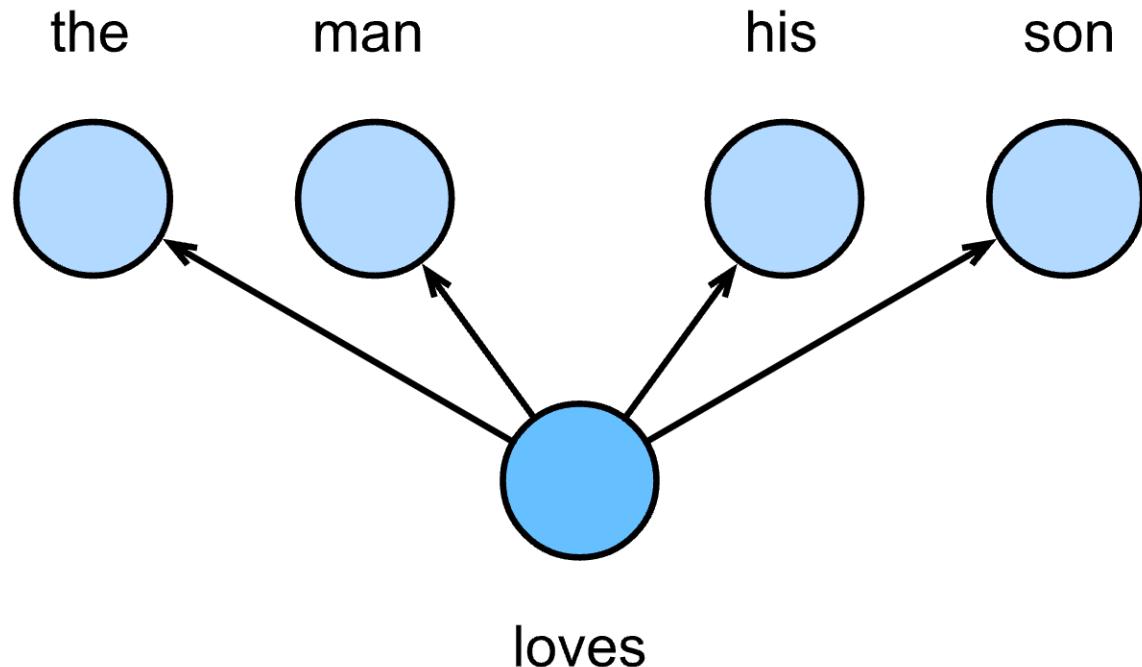
Модель skip-gram



- ▶ Skip-gram модель предполагает, что контекст зависит от центрального слова:

$$P(\text{"the"}, \text{"man"}, \text{"his"}, \text{"son"} \mid \text{"loves"})$$

Модель skip-gram



- ▶ Предположим, что распределение условно независимое:

$$\begin{aligned} P(\text{"the"}, \text{"man"}, \text{"his"}, \text{"son"} \mid \text{"loves"}) &= \\ P(\text{"the"} \mid \text{"loves"}) \times P(\text{"man"} \mid \text{"loves"}) \times P(\text{"his"} \mid \text{"loves"}) \times P(\text{"son"} \mid \text{"loves"}) \end{aligned}$$

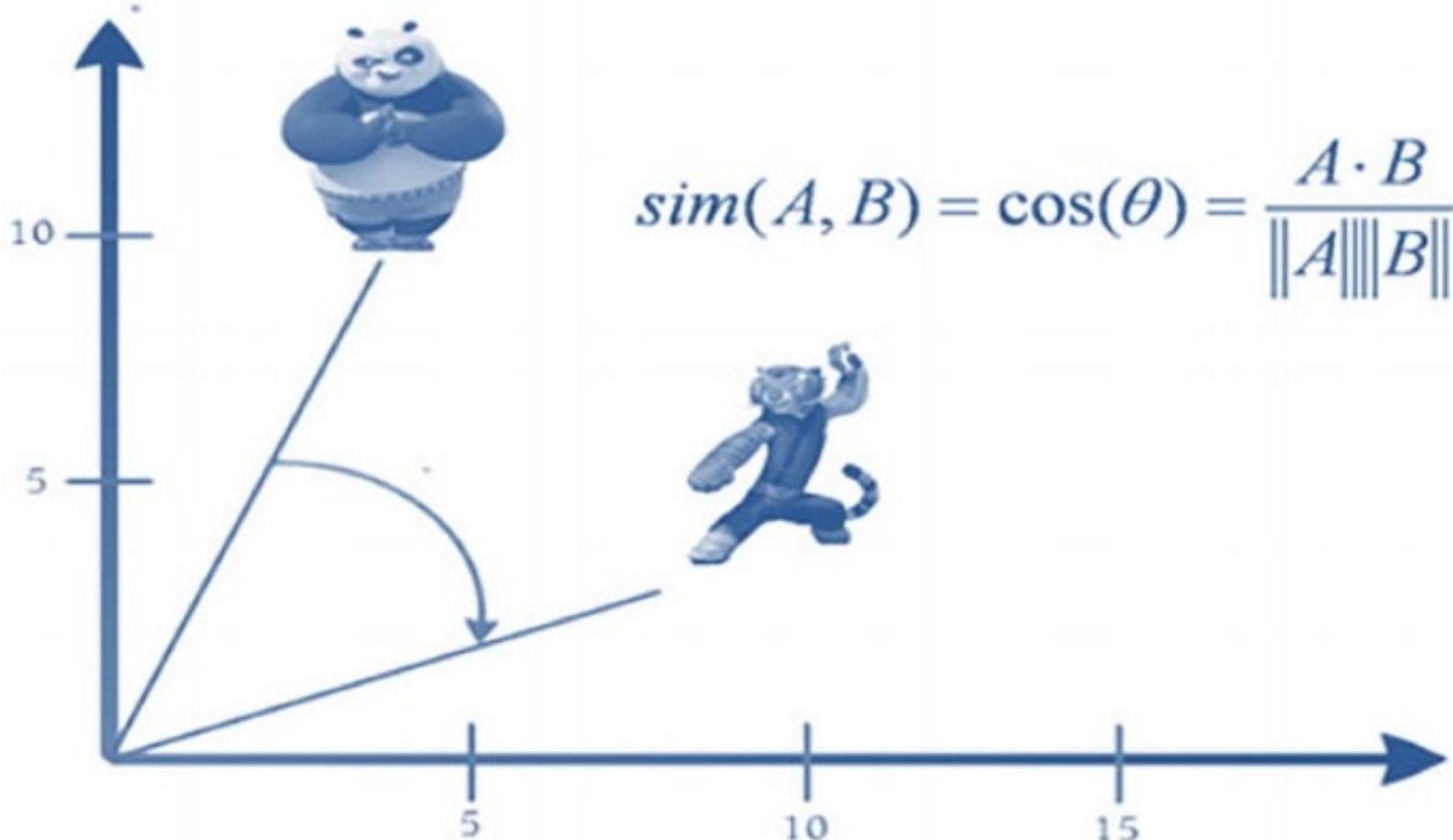
Модель skip-gram

- ▶ Обозначим центральное слово ("loves") как w_c
- ▶ Обозначим контекстное слово ("man") как w_o
- ▶ Для слов будем искать векторные представления: $u_o, v_c \in R^d$
- ▶ Тогда опишем вероятности как:

$$P(w_0|w_c) = \frac{\exp(u_o^T v_c)}{\sum_{i \in V} \exp(u_i^T v_c)}$$

Где V – текст длиной T слов

Почему uv?



Модель skip-gram

- ▶ Тогда функция потерь word2vec модели запишем как:

$$L = - \sum_{t=1}^T \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \log P(w_{t+j} | w_t) \rightarrow \min$$

- ▶ Т.е. сумма логарифмов вероятностей для всех последовательностей слов длиной $2m$
- ▶ Есть связь с функцией потерь для классификации? ☺

Пример текста

Text Corpus

Window Size = 2

The	quick	brown
-----	-------	-------

fox jumps over the red dog

The	quick	brown	fox
-----	-------	-------	-----

jumps over the red dog

The	quick	brown	fox	jumps
-----	-------	-------	-----	-------

over the red dog

The	quick	brown	fox	jumps	over
-----	-------	-------	-----	-------	------

the red dog

Training Samples

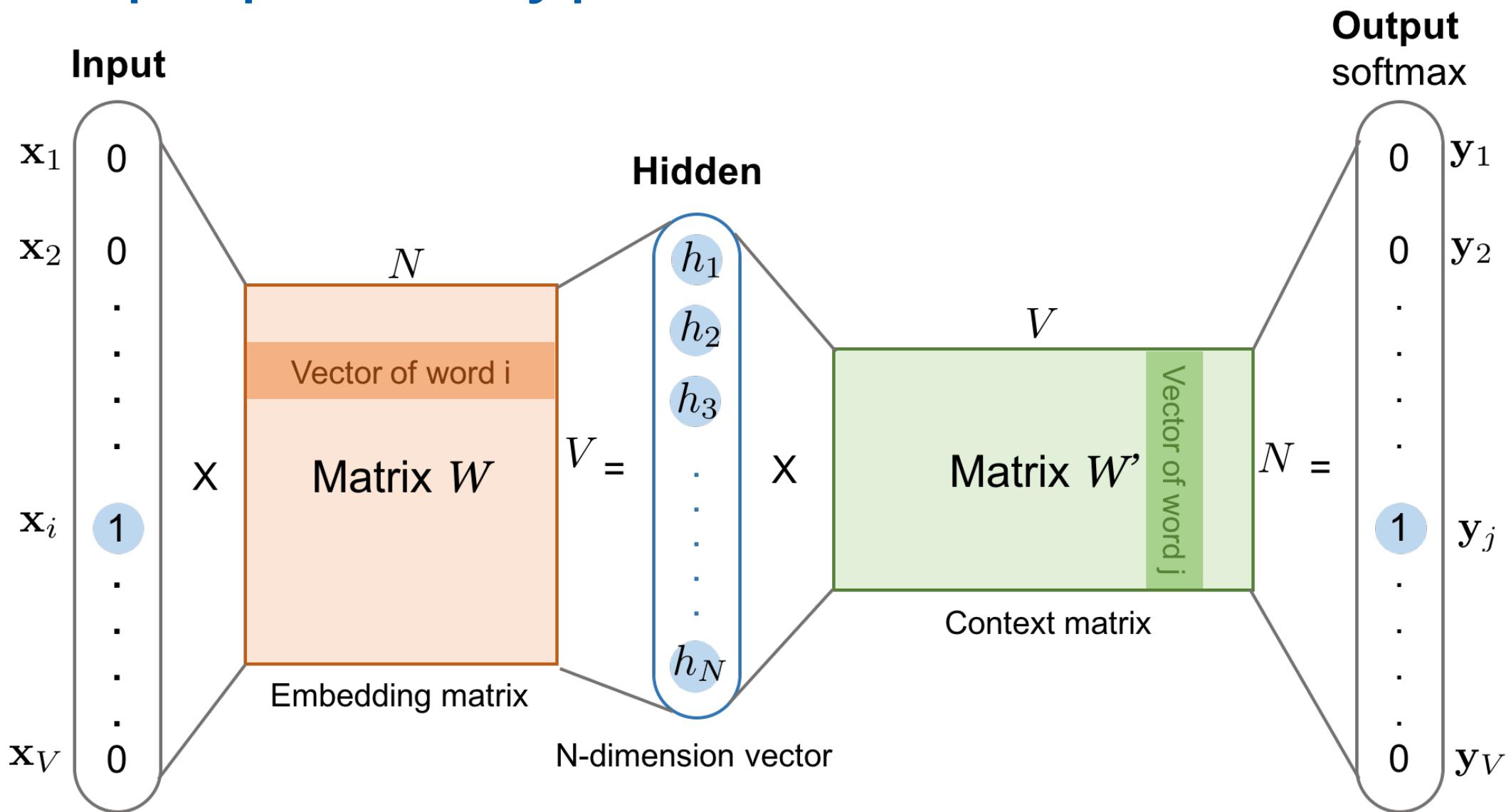
(The , quick)
(The , brown)

(quick, the)
(quick , brown)
(quick, fox)

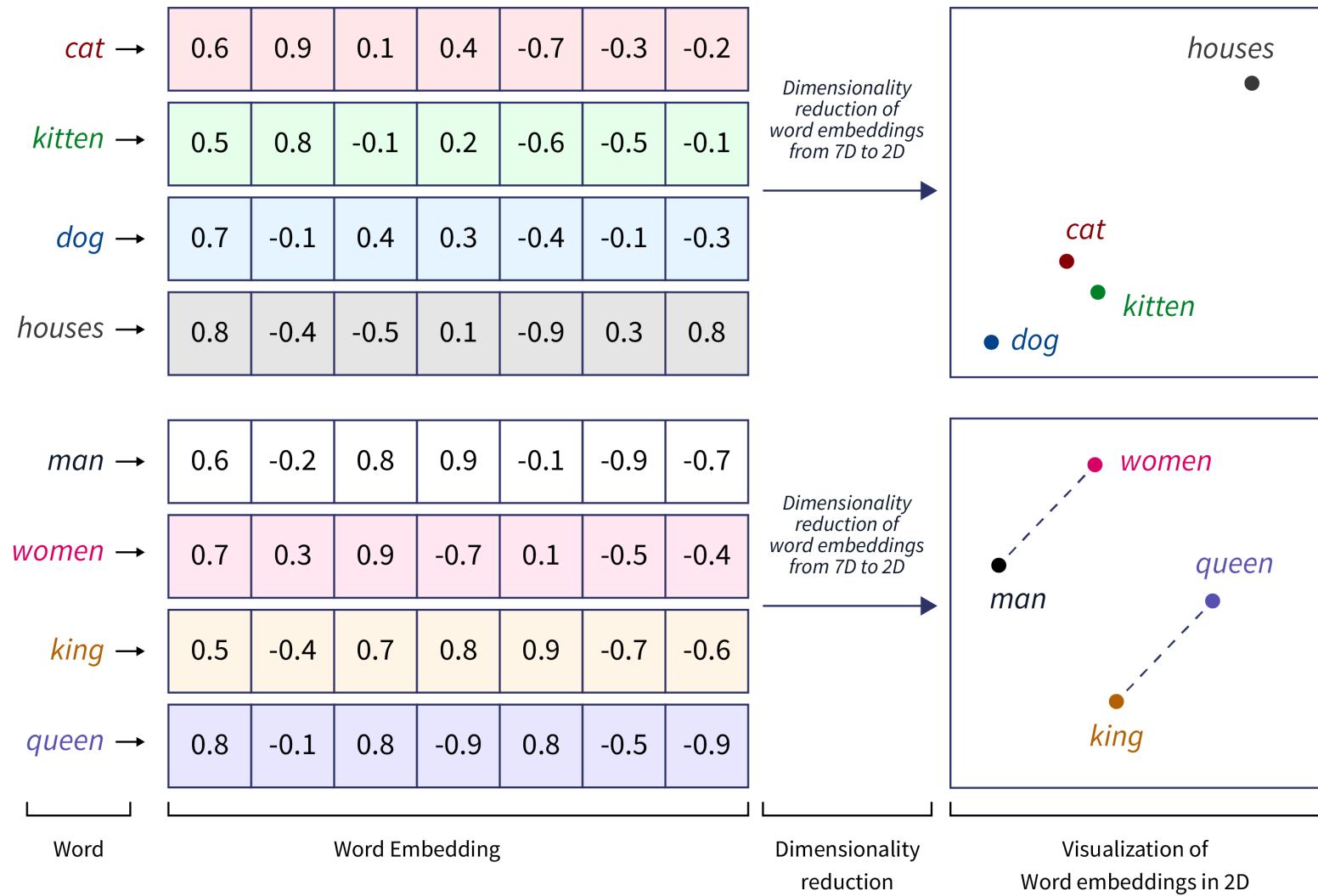
(brown , the)
(brown , quick)
(brown , fox)
(brown , jumps)

(fox , quick)
(fox , brown)
(fox , jumps)
(fox , over)

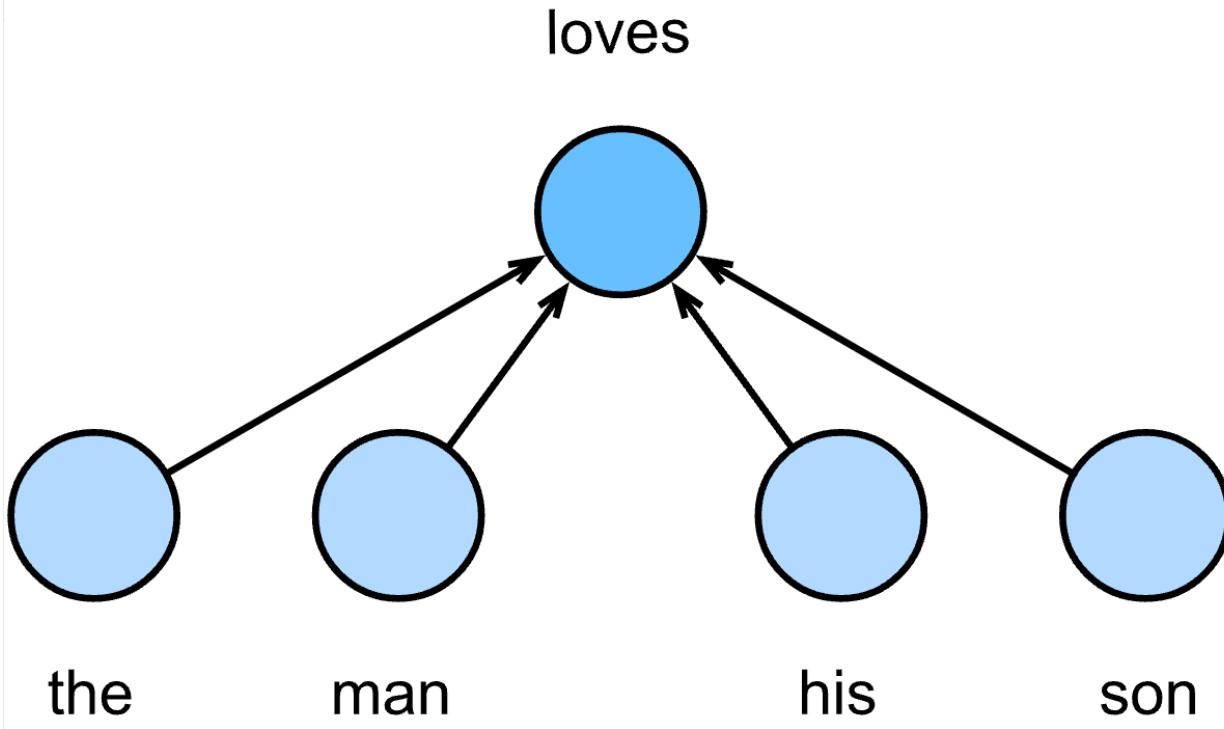
Пример архитектуры



Пример эмбеддингов (векторов)



Модель continuous bag of words (CBOW)



- ▶ СВОУ модель предполагает, что центральное слово зависит от контекста:

$$P(\text{"loves"} \mid \text{"the"}, \text{"man"}, \text{"his"}, \text{"son"})$$

Модель CBOW

- ▶ Обозначим центральное слово ("loves") как w_c
- ▶ Обозначим контекстное слово ("man") как w_{oi}
- ▶ Для слов будем искать векторные представления: $u_{oi}, v_c \in R^d$
- ▶ Тогда опишем вероятности как:

$$P(w_c | w_{o1}, w_{o2}, \dots, w_{o2m}) = \frac{\exp(u_c^T \bar{v}_o)}{\sum_{i \in V} \exp(u_i^T \bar{v}_o)}$$

$$\bar{v}_o = \frac{1}{2m} (v_{o1} + v_{o2} + \dots + v_{o2m})$$

Где V – текст длиной T слов

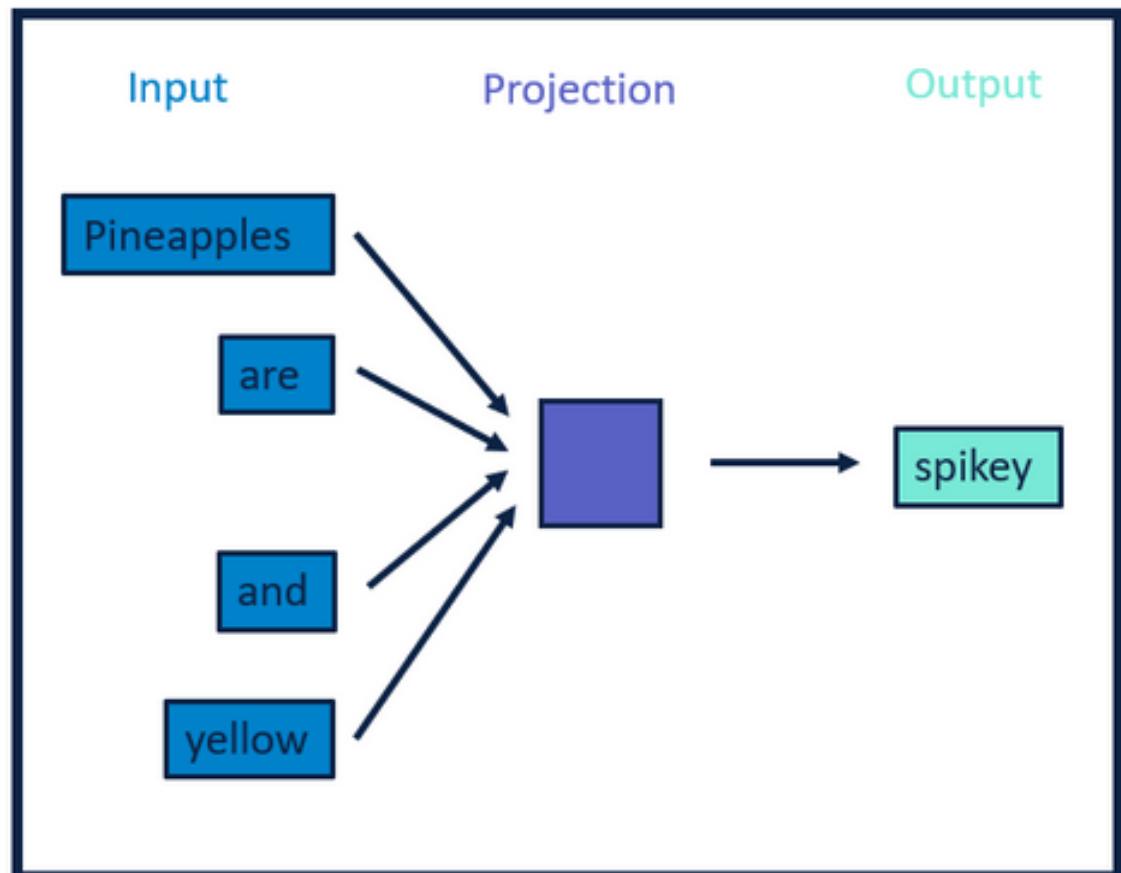
Модель CBOW

- ▶ Тогда функция потерь word2vec модели запишем как:

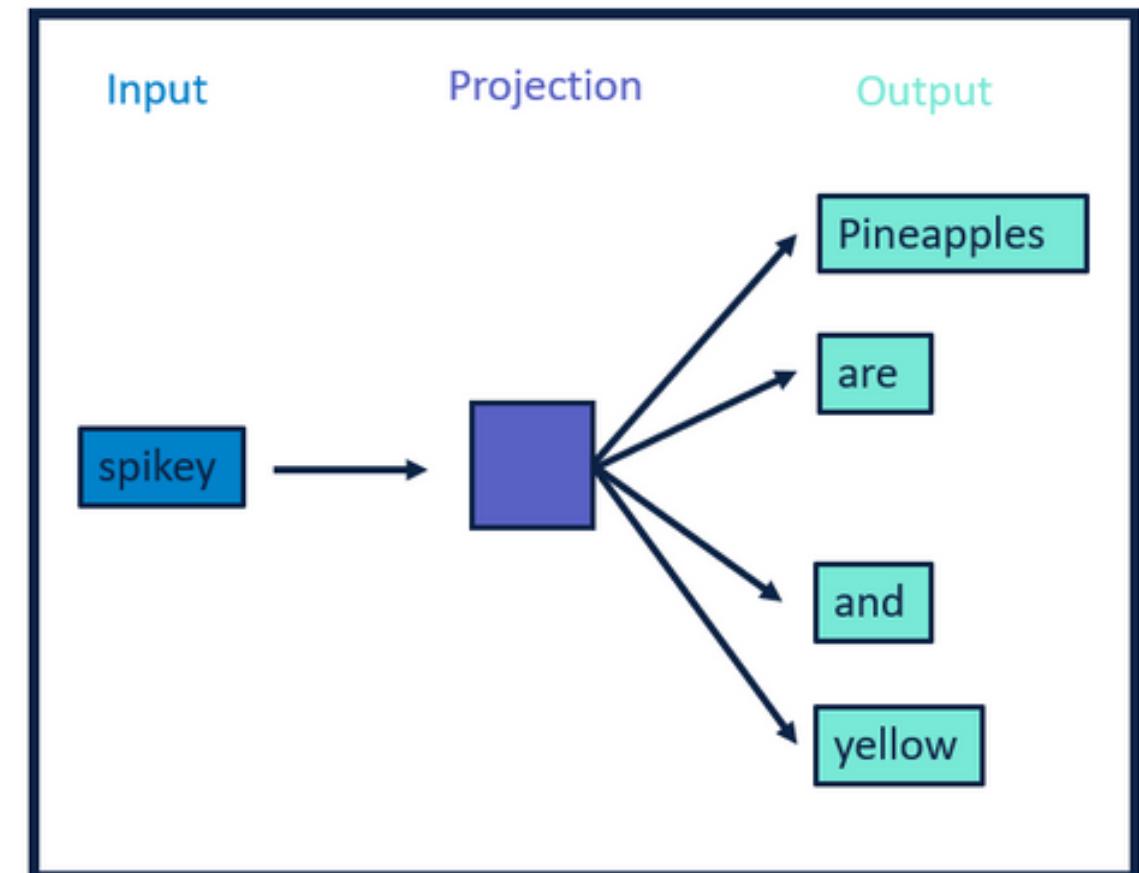
$$L = - \sum_{t=1}^T \log P(w_c | w_{o1}, w_{o2}, \dots, w_{o2m}) \rightarrow \min$$

- ▶ Т.е. сумма логарифмов вероятностей для всех последовательностей слов длиной $2m$

Word2vec



CBOW



Skip-gram

Проблемы word2vec

- ▶ Не учитывает структуру слов (окончания, суффиксы, приставки)
- ▶ Не использует никакой априорной информации о разных формах одного слова

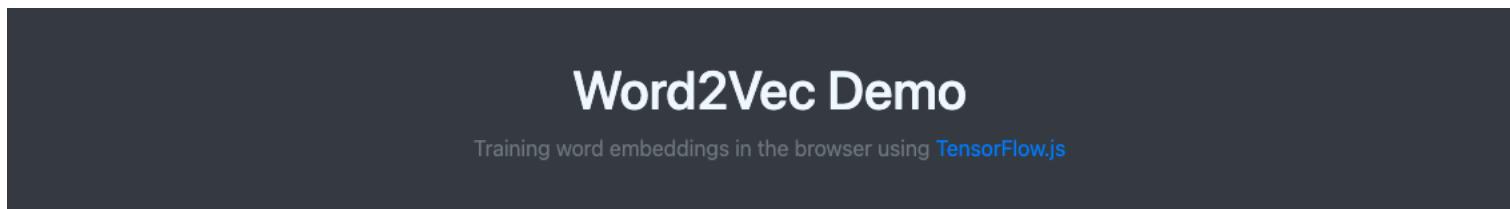
Модель FastText

- ▶ Заменим каждое слово на n токенов
 - «руслан» -> (<руслан>, <ру, рус, усл, сла, лан, ан>)
- ▶ Обучаем векторы токенов: u_1, u_2, \dots, u_n
- ▶ Вектор первоначального слова $z_w = \sum_{i=1}^n u_i$
- ▶ Архитектура и обучение как в word2vec

Пример



Demo



<https://remykarem.github.io/word2vec-demo/>