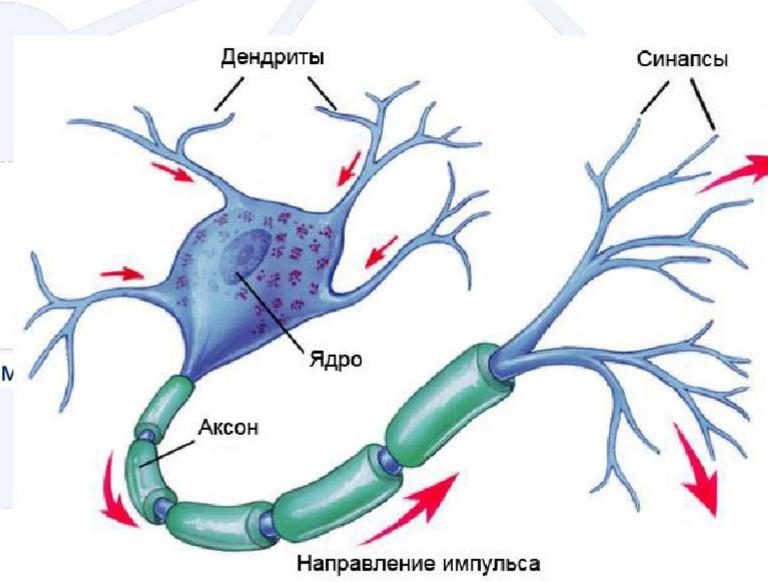
Введение в нейронные сети

Сферы применения технологий ИИ



виологический неирон человеческого мозга

Дендриты получают входные сигналы от других нейронов **Тело нейрона** суммирует входные сигналы, полученные от других нейронов, и в случае, когда сумма превышает определенный порог, генерирует выходной сигнал **Аксон** служит для передачи выходного сигнала другим нейронам **Синапсы** соединяют аксон с дендритами других нейронов; величина сигнала зависит от силы (синаптического веса) связи

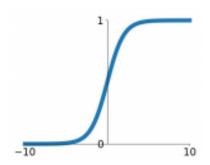


Входные Блок нелинейного Блок Выходной Синаптические преобразования сигналы суммир ования веса сигнап W_1 \mathbf{Y} W_2 W_{n}

Функции активации

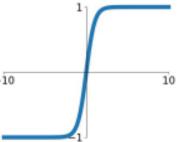
Sigmoid

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



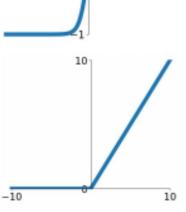
tanh

tanh(x)



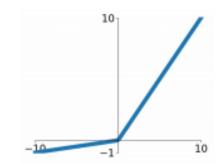
ReLU

 $\max(0, x)$



Leaky ReLU

 $\max(0.1x, x)$

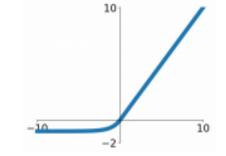


Maxout

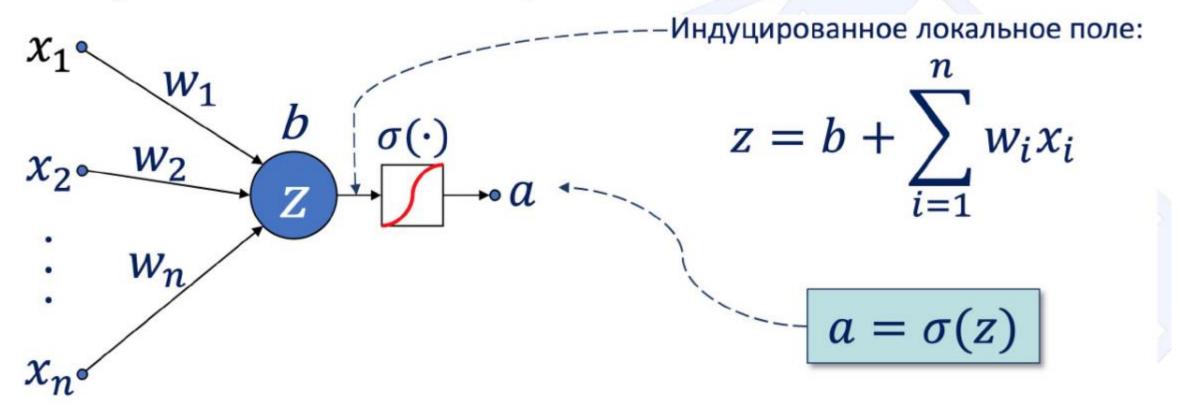
 $\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

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



Искусственный нейрон



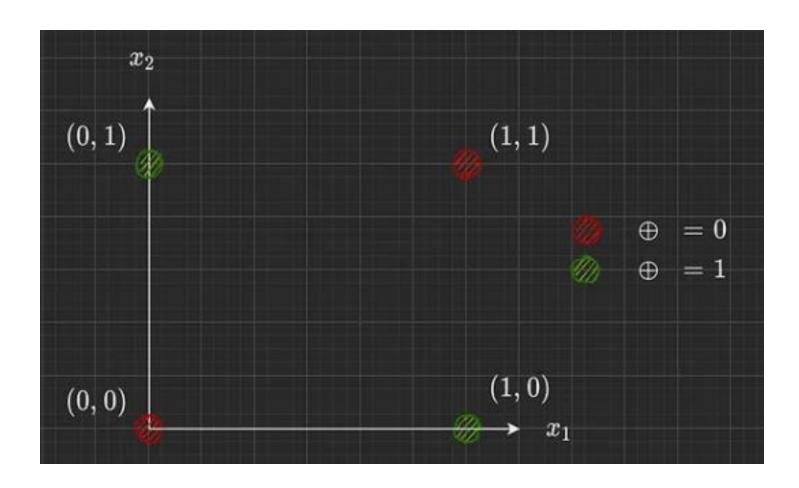
$$\vec{x} = (x_1, x_2, ..., x_n)$$
 – входные сигналы: $x_i \in \mathbb{R}$

$$\overrightarrow{w} = (w_1, w_2, ..., w_n)$$
 – синапсические веса: $w_i \in \mathbb{R}$

b – смещение: $b \in \mathbb{R}$

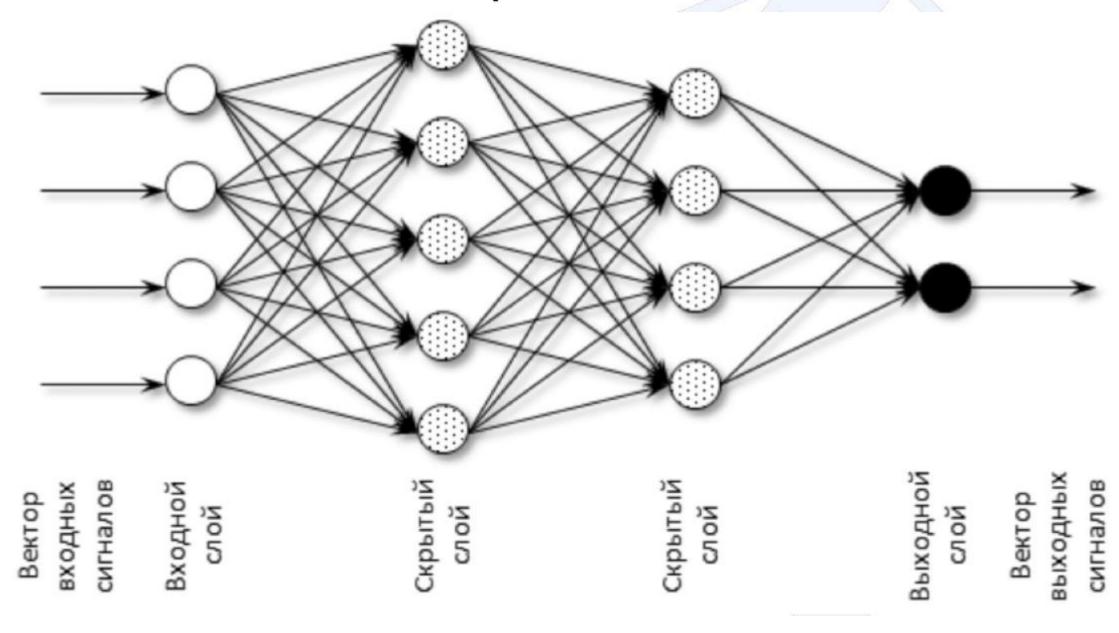
a – выходной сигнал: 0 < a < 1

Проблема



Проблема в том, что линейная модель не может решить на первый взгляд простейшую задачу – имитацию поведения операции XOR (исключающее или)

Полносвязная нейронная сеть



Как обучать такую сеть?

Терминология.

Я предлагаю сразу определиться с терминами, которые будут звучать в дальнейшем, чтобы никто не путался.

- 1. Скрытый слой промежуточный слой в нейронной сети. Hidden layer
- 2. Прямой проход прохождение сигнала по сети для предсказания (слева направо)
- 3. Обратный проход (back propagation) обновление параметров модели.

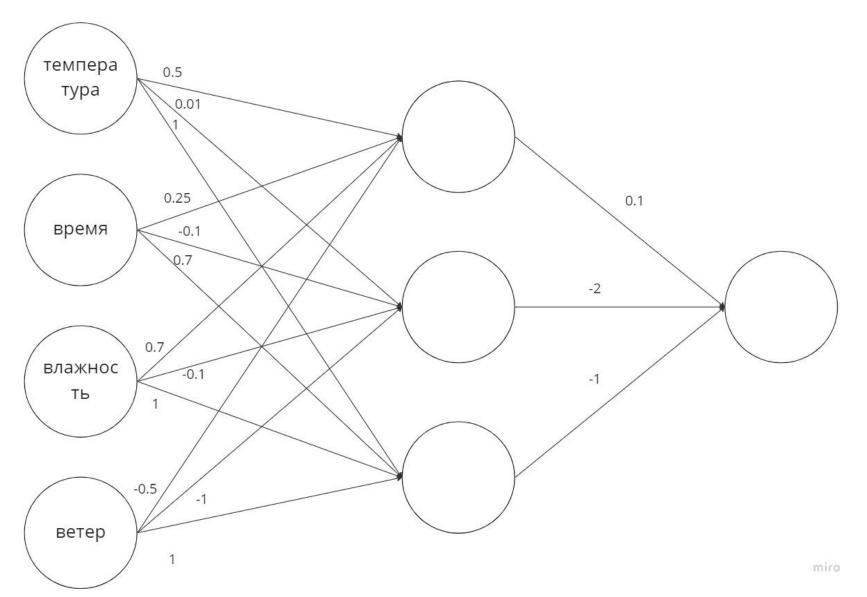
Обратное распространение ошибки

Прямое распространение ошибки (предсказание)	Обратное распространение ошибки.
Нужно посчитать выход каждого нейрона. $z^i = W^{i-1}x + b^{i-1}$ $a^i = f(z^i)$	Сначала нужно найти ошибку на последнем слое нейронной сети. $E = cost(predict - y)$ Далее для каждого нейрона считаем его «его долю в финальной ошибке» $\frac{dE}{dW^i} = \frac{dE}{da^{i+1}} * \frac{da^{i+1}}{dz^{i+1}} * \frac{dz^{i+1}}{dW^i}$

Формулы выглядят сурово, но давайте попробуем разобраться, что требуется, чтобы обучить сеть:

- 1. Нужно сделать предсказание (прямой проход)
- 2. Сохранить результаты промежуточных слоев, так как они потом участвуют в расчете ошибки

В качестве примера рассмотрим задачу. Идти ли на улицу?



Прямой проход.

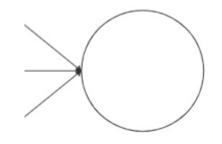
• Посчитайте выход каждого нейрона.

Обратный проход.

Для подсчета ошибки будем использовать MSE.

Сначала нужно понять насколько наше предсказание неверно:

Напомню, что предсказание
– это выход последнего
нейрона



Ошибка
$$E = \frac{1}{2}(y - pred)^2$$

Теперь мы знаем насколько ошиблись и по сложной формуле

$$\frac{dE}{dW^{i}} = \frac{dE}{da^{i+1}} * \frac{da^{i+1}}{dz^{i+1}} * \frac{dz^{i+1}}{dW^{i}}$$
 можем обновить веса.

Чтобы найти $\frac{dE}{dW^i}$ или влияние веса W^i на ошибку нужно знать:

$$\frac{dE}{da^{i+1}}$$
 — влияние функции активации на ошибку

$$\frac{da^{i+1}}{dz^{i+1}}$$
 — влияние функции активации на выход нейрона

$$\frac{dz^{i+1}}{dW^i}$$
 — влияние веса W^i на выход нейрона

Посчитаем для $W_{1,1}^2$ вместе:

$$\frac{dE}{da^2} - y - pred$$

$$rac{da^2}{dz^2}$$
 — на выходном слое нет активации $ightarrow 1$

$$\frac{dz^2}{dW^2}$$
 — z зависит от $W^2x \to x$

Нюанс в обновлении весов!

Есть небольшая хитрость в том, как обновлять веса предыдущего слоя! Посчитаем для $W_{1,1}^1$ вместе:

$$\frac{dE}{dW_{1,1}^1} = \frac{dE}{da_2} * \frac{da_2}{dz_2} * \frac{dz_2}{da_1} * \frac{da_1}{dz_1} * \frac{dz_1}{dW_{1,1}^1},$$
 как вы могли заметить $\frac{dE}{da_2} * \frac{da_2}{dz_2}$ мы уже вычисляли, что сильно упрощает нам жизнь!

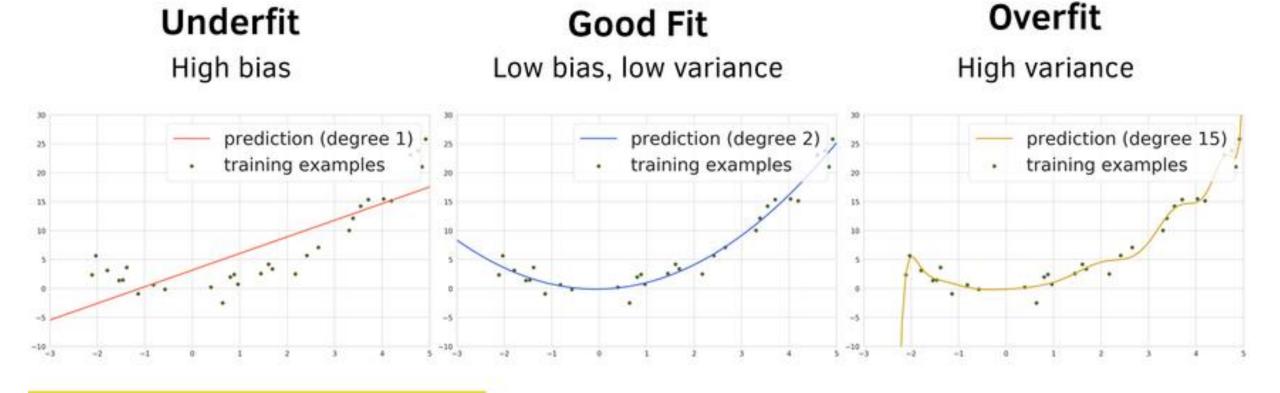
Для чего!?

Да, современные библиотеки вычисляют все градиенты и обновляют веса автоматически, но вы должны были попробовать себя в роли нейронной сети, чтобы понять несколько важных правил о проектировании нейронных сетей:

- 1. Чем глубже нейронная сеть, тем дольше она обучается (почему?)
- 2. Без функции активации нет смысла строить глубокие нейронные сети (почему)
- 3. На этапе обучения для тренировки сети требуется дополнительная память (почему?)

Что может пойти не так?

Types of Model Fit

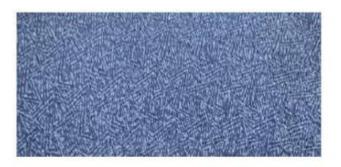


Применении ИИ:

Раскраска изображений



Примерка интерьера







Стилизация изображений







P.S.

У кого возникают трудности в понимании Python, посмотрите в сторону <u>академии Яндекса</u>.