

# Глубинное обучение

## Введение в нейронные сети

Даниил Водолазский

ВШЭ

23 июня 2021 г.



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ

# ВНИМАНИЕ!

ДАННЫЙ КУРС СОДЕРЖИТ БОЛЬШОЕ КОЛИЧЕСТВО РАЗНООБРАЗНОГО КОДА И ЗАДАНИЙ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ.

НА ПЕРВЫЙ ВЗГЛЯД ОН МОЖЕТ ПОКАЗАТЬСЯ СЛОЖНЫМ И ТРАВМИРОВАТЬ НЕПОДГОТОВЛЕННУЮ ПСИХИКУ. ТАКЖЕ ОН СОДЕРЖИТ БОЛЬШОЕ КОЛИЧЕСТВО НЕУДАЧНЫХ ШУТОК И НЕУМЕСТНЫХ ОТСЫЛОК.

В СВЯЗИ С ЭТИМ КУРС НЕ РЕКОМЕНДУЕТСЯ ПРОСЛУШИВАТЬ... НИКОМУ.

18+

# Содержание

## 1 Структура курса, материалы

Как проходят занятия

Темы занятий

Книги по Deep Learning

Курсы по Deep Learning

## 2 История нейронных сетей

Первый формальный нейрон (1943)

Тест Тьюринга (1950)

Дартмутский семинар (1956)

Перцептрон (1958)

Зима искусственного интеллекта (1974–1980)

Развитие и вторая зима ИИ (1987–1993)

Революция машинного обучения

Тренды в глубинном обучении

## 3 От регрессии к нейросети

Линейная регрессия

Нейрон

Линейная регрессия (векторная форма)

Как обобщить линейную регрессию?

# Как проходят занятия

- Занятия — каждую среду с 19:00 по 22:00, но весь август — каникулы.
- Занятия очные (Покровский бульвар, 11, ауд. R614) и через Zoom.
- Вначале лекция, затем семинар или практика.
- На парах смотрим презентации, пишем код, решаем задачи.
- Будет много математики.
- Что-то неясно — **ПЕРЕБЕЙ И СПРОСИ**.
- Лучше приходить с ноутбуком.
- Все материалы находятся на страничке курса<sup>1</sup>

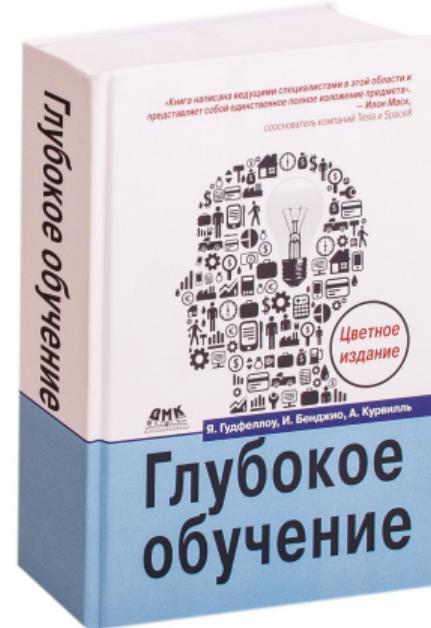
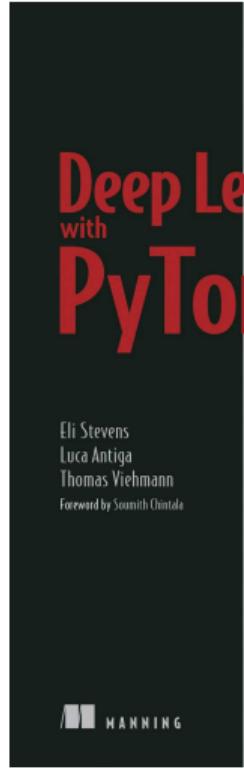
---

<sup>1</sup>[https://github.com/s231644/neural\\_nets\\_dpo](https://github.com/s231644/neural_nets_dpo)

# Темы занятий

- ① Введение в нейросети. История ИИ, перцепtron, регрессия, классификация.
- ② Варианты градиентного спуска. Алгоритм обратного распространения ошибки.
- ③ Инструменты в Python для обучения нейронных сетей. PyTorch.
- ④ Батч-нормализация. Инициализация. Эвристики для обучения сетей.
- ⑤ Свёрточные нейронные сети.
- ⑥ Полносвёрточные нейронные сети. Сегментация изображений, детекция объектов.
- ⑦ Transfer learning. Metric Learning. Meta learning.
- ⑧ Интерпретация свёрточных нейронных сетей. Перенос стиля.
- ⑨ Автокодировщики, вариационные автокодировщики.
- ⑩ Генеративные нейронные сети. GAN.
- ⑪ Введение в NLP, эмбеддинги, языковые модели, word2vec, fastText, GLOVe.
- ⑫ Свёрточные сети для NLP. Рекуррентные нейронные сети, LSTM, GRU. ELMo.
- ⑬ Seq2Seq-модели. Машинный перевод.
- ⑭ Механизм внимания, трансформеры. BERT, GPT.
- ⑮ Multimodal DL. DALLE. Обзорно: видео, графы, речь.
- ⑯ Введение в обучение с подкреплением. Введение в байесовские нейронные сети.

# Книги по Deep Learning



# Курсы по Deep Learning

- Если ничего не знаете про машинное обучение, смотрите вводный курс от Яндекса и МФТИ.<sup>2</sup> **Для тех, кто мало знает про ML.**
- Advanced ML от Яндекса.<sup>3</sup> Там есть очень разные специфические курсы. Первый из них про нейронки. Код на Tensorflow. Версия библиотеки там пока что старая. **Для тех, кто хочет развиваться дальше.**
- Курс нейронок, который читают в ШАД и Сколтехе.<sup>4</sup> Есть варианты кода на разных фреймворках. Есть видео лекций на русском и английском. **Для тех, кто хочет посмотреть как читают курс по DL в ШАД.**

---

<sup>2</sup><https://www.coursera.org/specializations/machine-learning-data-analysis>

<sup>3</sup><https://www.coursera.org/specializations/aml>

<sup>4</sup>[https://github.com/yandexdataschool/Practical\\_DL/tree/master](https://github.com/yandexdataschool/Practical_DL/tree/master)

# Курсы по Deep Learning

- ru Нейронные сети и компьютерное зрение (Samsung).<sup>5</sup>
- ru Deep Learning на пальцах (Семен Козлов).<sup>6</sup> Задания на Pytorch для самостоятельного решения.
- ru Теоретическое глубинное обучение (iPavlov, МФТИ).<sup>7</sup> Есть вводный курс, есть хардкорный курс.
- en Deep Learning (Stanford).<sup>8</sup>
- en Convolutional Neural Networks for Visual Recognition (Stanford).<sup>9</sup>
- en Natural Language Processing with Deep Learning (Stanford).<sup>10</sup>

---

<sup>5</sup><https://stepik.org/course/50352/syllabus>

<sup>6</sup><https://dlcourse.ai>

<sup>7</sup>[https://www.youtube.com/playlist?list=PLt1IfGj6--\\_dMa3Ff8mwjq1y0GijJ89Wa](https://www.youtube.com/playlist?list=PLt1IfGj6--_dMa3Ff8mwjq1y0GijJ89Wa)

<sup>8</sup><https://cs230.stanford.edu/>

<sup>9</sup><http://cs231n.stanford.edu/>

<sup>10</sup><http://web.stanford.edu/class/cs224n/>

# Содержание

## 1 Структура курса, материалы

Как проходят занятия

Темы занятий

Книги по Deep Learning

Курсы по Deep Learning

## 2 История нейронных сетей

Первый формальный нейрон (1943)

Тест Тьюринга (1950)

Дартмутский семинар (1956)

Перцептрон (1958)

Зима искусственного интеллекта (1974–1980)

Развитие и вторая зима ИИ (1987–1993)

Революция машинного обучения

Тренды в глубинном обучении

## 3 От регрессии к нейросети

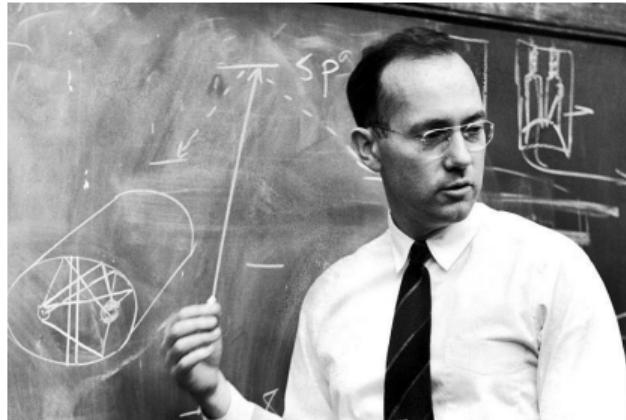
Линейная регрессия

Нейрон

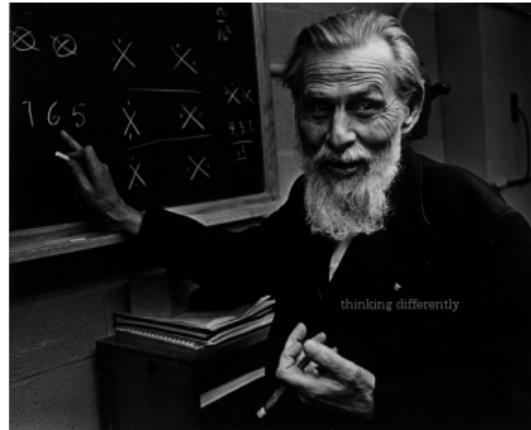
Линейная регрессия (векторная форма)

Как обобщить линейную регрессию?

# Первый формальный нейрон (1943)



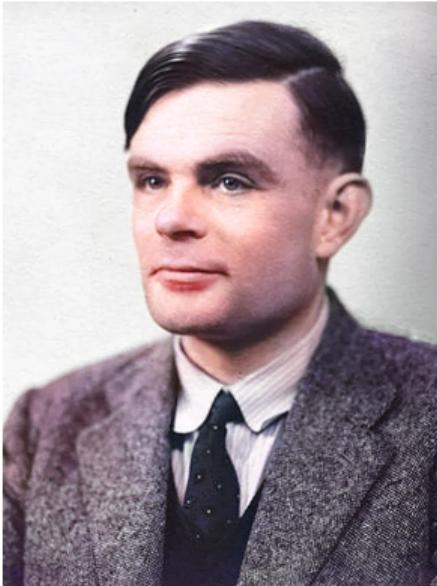
Уоррен Мак-Каллок (1898–1969)



Уолтер Питтс (1923–1969)

Warren S McCulloch и Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". B: The bulletin of mathematical biophysics 5.4 (1943), с. 115—133

# Тест Тьюринга (1950)



Алан Тьюринг (1912–1954)

AM Turing. "Computing Machinery and Intelligence". В: [Mind 59 \(1950\), с. 433—460](#)

VOL. LIX. NO. 236.] [October, 1950]

## MIND A QUARTERLY REVIEW or PSYCHOLOGY AND PHILOSOPHY

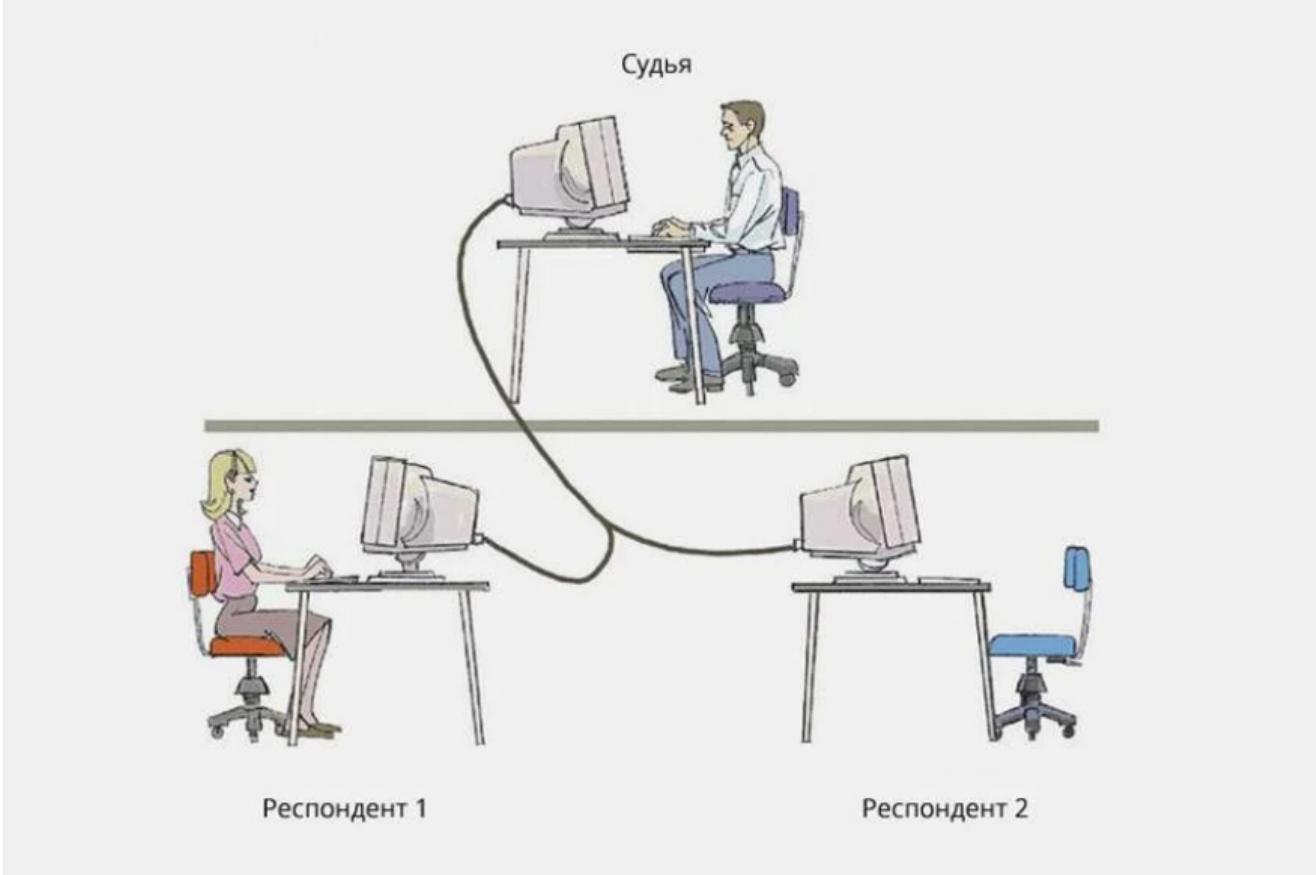
I.—COMPUTING MACHINERY AND  
INTELLIGENCE  
By A. M. TURING

1. *The Imitation Game*.  
I propose to consider the question, 'Can machines think?' This should begin with definitions of the meaning of the terms 'machine' and 'think'. The definitions might be framed so as to reflect as far as possible the common usage of our words, but I believe that this attitude is dangerous. If the meaning of the words 'machine' and 'think' are to be fixed by examining how they are commonly used, then we should expect to find that their meanings are so vague and fuzzy as to be useless for purposes of scientific thinking and the answer to the question, 'Can machines think?' is to be sought in a statistical survey such as a Gallup poll. But this is short-sighted. I believe that a真正 useful definition of intelligence is to replace the question 'Can machines think?' by 'Can machines do what you do?' and when it is closely related to it and it is measured in relatively unambiguous words.

The new form of the problem can be described in terms of a game which I call the imitation game. It is played with three people, a man (A), a woman (B) and an interrogator (C) who may be either sex. The interrogator stays in a room apart from the other two. He is given a list of questions to which he is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either 'X is A and Y is B' or 'X is B and Y is A'. The interrogator is allowed to put questions to A and B thus:

C : Will X please tell me the length of his or her hair?  
Now suppose X is actually A, then A must answer. It is A's  
turn.

Статья в журнале Mind



## Цитата

Мы предлагаем исследование **искусственного интеллекта** сроком в 2 месяца с участием 10 человек летом 1956 года в Дартмутском колледже, Гановер, Нью-Гэмпшир.

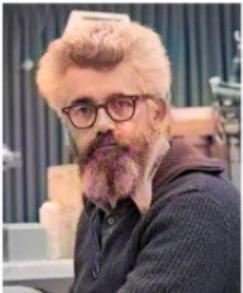
Исследование основано на предположении, что всякий аспект обучения или любое другое свойство интеллекта может в принципе быть столь точно описано, что машина сможет его симулировать.

Мы попытаемся понять, как обучить машины использовать естественные языки, формировать абстракции и концепции, решать задачи, сейчас подвластные только людям, и улучшать самих себя.

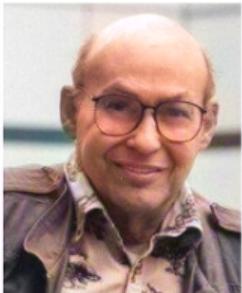
Мы считаем, что существенное продвижение в одной или более из этих проблем вполне возможно, если специально подобранная группа учёных будет работать над этим в течение лета.

— Джон Маккарти, 1956

# 1956 Dartmouth Conference: The Founding Fathers of AI



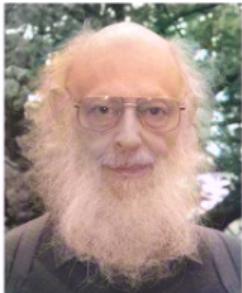
John McCarthy



Marvin Minsky



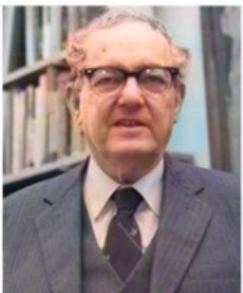
Claude Shannon



Ray Solomonoff



Alan Newell



Herbert Simon



Arthur Samuel



Oliver Selfridge



Nathaniel Rochester



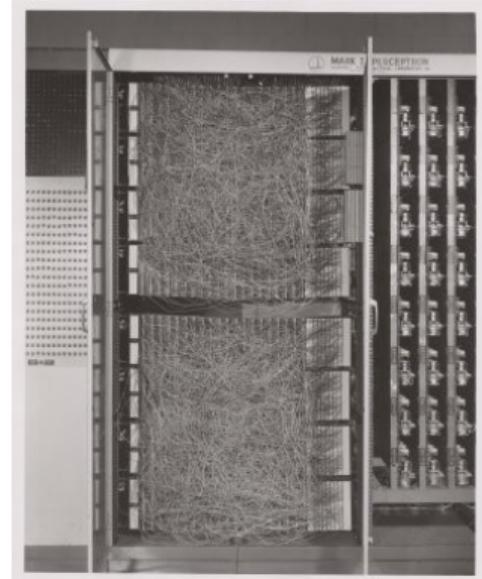
Trenchard More

# Перцептрон (1958)



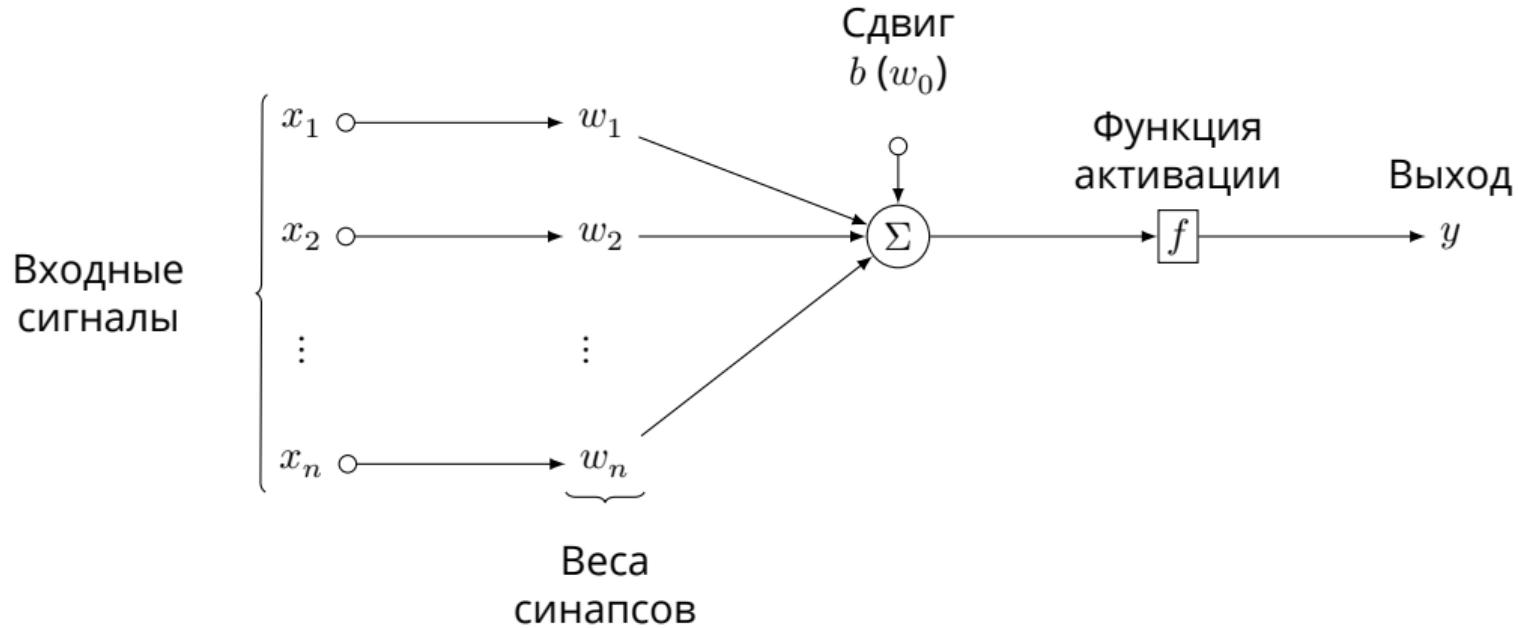
Фрэнк Розенблatt (1928–1971)

Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain.". B: [Psychological review](#) 65.6 (1958), c. 386



Mark I Perceptron

# Перцептрон (1958)



$$y = f \left( \sum_{i=1}^n w_i x_i + b \right), \quad f(t) = [t > 0]$$

# Перцептрон (1958)

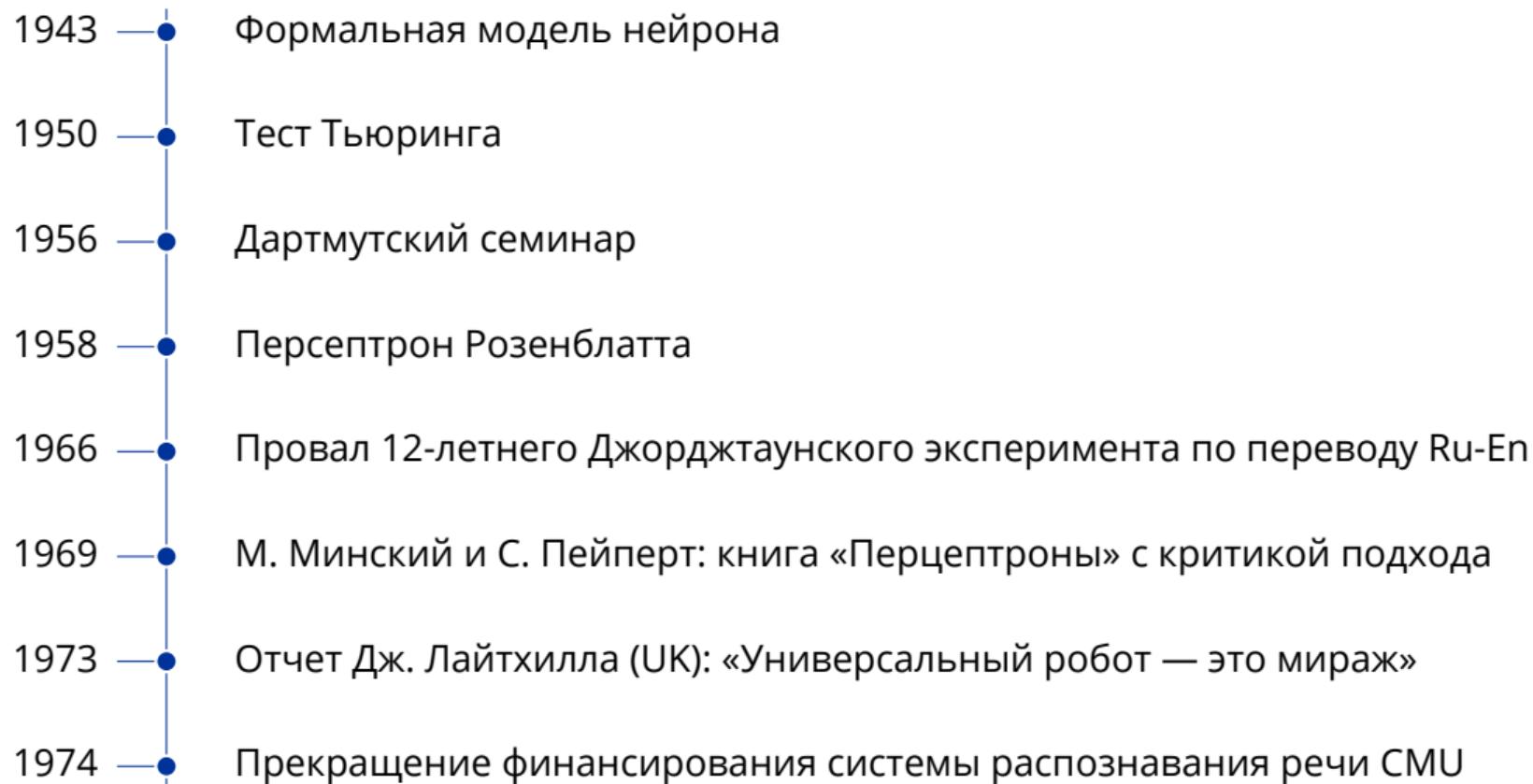
- О том, как гениальный беспризорник и профессор пили виски и придумывали первую модель искусственного нейрона<sup>11</sup>
- Всё, что вы хотели знать о перцептранах Розенблатта, но боялись спросить<sup>12</sup>
- Больше фотографий компьютера<sup>13</sup>

---

<sup>11</sup><https://habr.com/ru/company/sberdevices/blog/525508/>

<sup>12</sup><https://habr.com/ru/company/sberdevices/blog/529932/>

<sup>13</sup>[https://americanhistory.si.edu/collections/search/object/nmah\\_334414](https://americanhistory.si.edu/collections/search/object/nmah_334414)

- 
- 1943 — Формальная модель нейрона
- 1950 — Тест Тьюринга
- 1956 — Дартмутский семинар
- 1958 — Персепtron Розенблатта
- 1966 — Провал 12-летнего Джорджтаунского эксперимента по переводу Ru-En
- 1969 — М. Минский и С. Пейперт: книга «Перцептроны» с критикой подхода
- 1973 — Отчет Дж. Лайтхилла (UK): «Универсальный робот — это мираж»
- 1974 — Прекращение финансирования системы распознавания речи CMU

## Определение

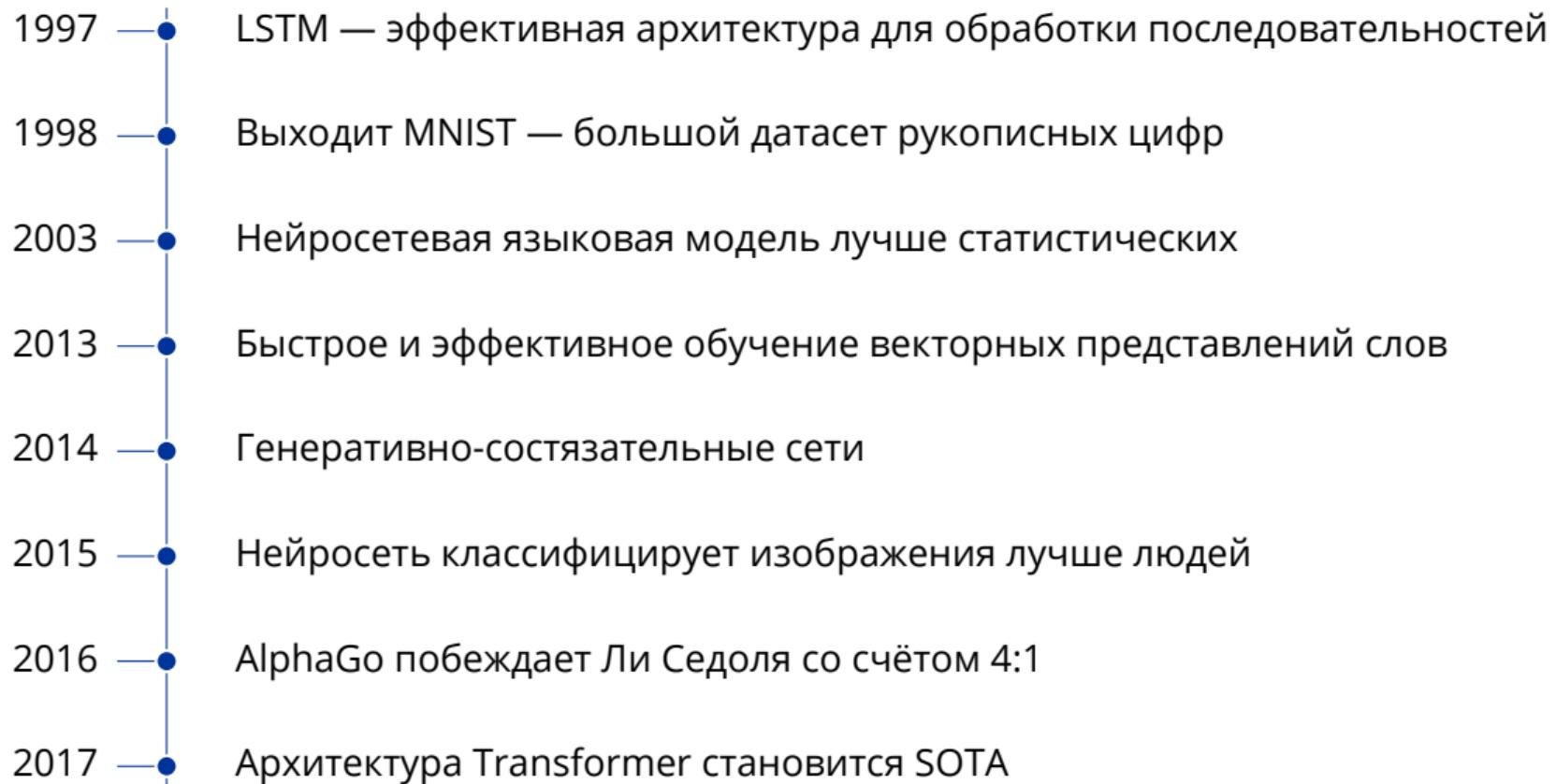
**Зима искусственного интеллекта** — период в истории исследований искусственного интеллекта, связанный с сокращением финансирования и снижением интереса.

— Википедия

- 70-е —● Расцвет экспертных систем (правила и знания о предметной области)
- 1975 —● **MYCIN** успешно идентифицирует вирусные бактерии
- 1979 —● Американская ассоциация искусственного интеллекта (AAAI)
- 1982 —● Нейронная сеть Хопфилда
- 1985 —● Машина Больцмана
- 1986 —● Алгоритм обратного распространения ошибки для больших сетей
- 1986 —● Рекуррентные нейронные сети
- 1989 —● Свёрточная сеть распознает рукописные цифры на конвертах

# Революция машинного обучения

- В 2005–2006 гг. группы Дж. Хинтона и Й. Бенджио научились обучать глубокие нейросети.
- Накопилось больше данных! Огромные данные!
- Компьютеры стали на порядки мощнее! Появились крутые GPU!
- На больших данных и мощностях заработали старые архитектуры.
- Появились новые алгоритмы, эвристики и подходы.
- Ящик Пандоры открыт!

- 
- A vertical timeline chart showing major milestones in AI development from 1997 to 2017. Each year is marked by a blue dot on a vertical blue line, followed by a horizontal blue line extending to the right with a small blue circle at its end, and then the corresponding event description.
- 1997: LSTM — эффективная архитектура для обработки последовательностей
  - 1998: Выходит MNIST — большой датасет рукописных цифр
  - 2003: Нейросетевая языковая модель лучше статистических
  - 2013: Быстрое и эффективное обучение векторных представлений слов
  - 2014: Генеративно-состязательные сети
  - 2015: Нейросеть классифицирует изображения лучше людей
  - 2016: AlphaGo побеждает Ли Седоля со счётом 4:1
  - 2017: Архитектура Transformer становится SOTA

A.M.  
TURING  
AWARD  
2018



YOSHUA BENGIO,  
GEOFFREY E. HINTON  
AND YANN LECUN

For conceptual and engineering  
breakthroughs that have made  
deep neural networks a critical  
component of computing

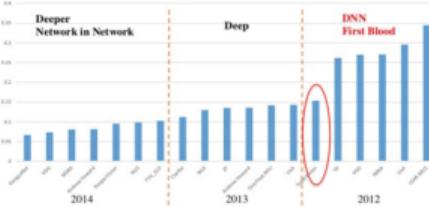
# ImageNet



4:1



ImageNet Classification error throughout years and groups



Пабло Пикассо



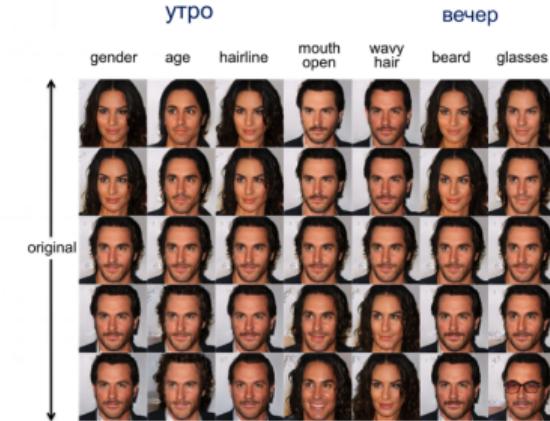
Винсент Ван Гог

Василий Кандинский

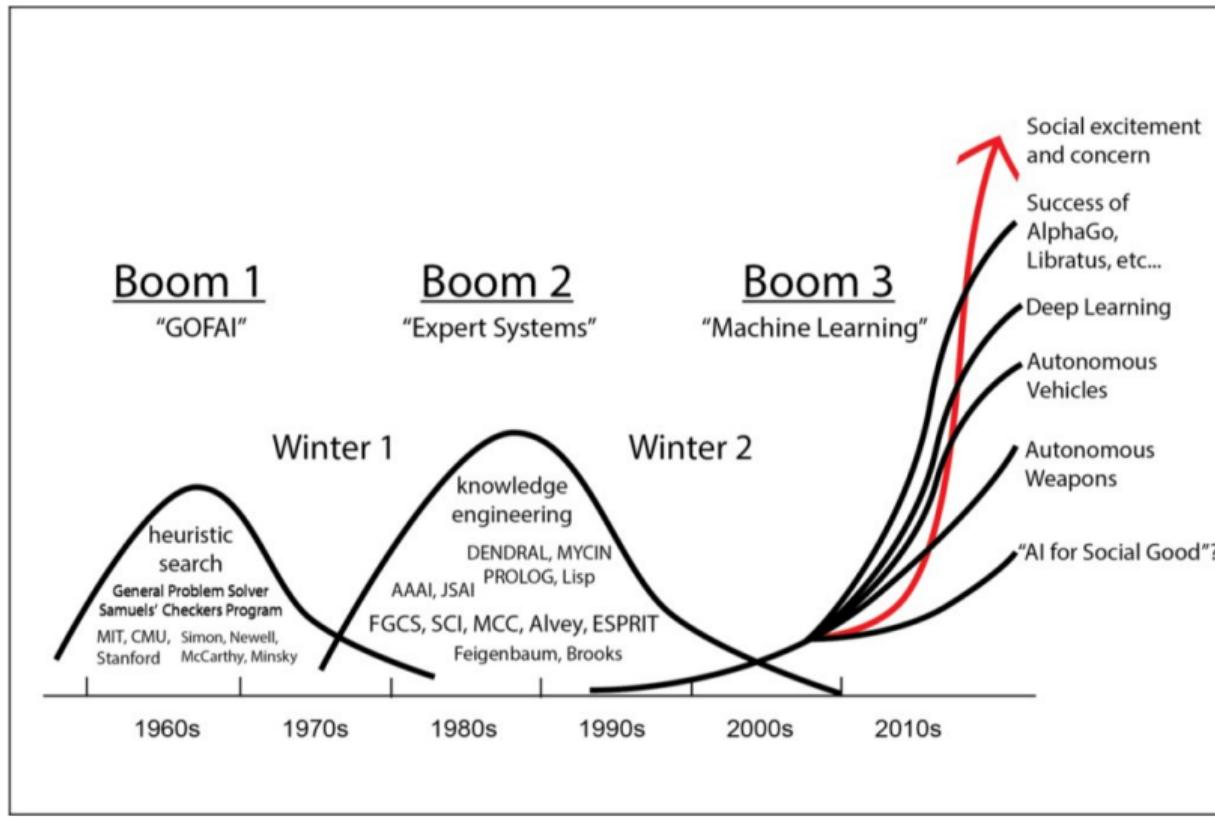


@mayank\_jee can i just say that im stoked to meet u? humans are super cool  
24/03/2016, 20:32

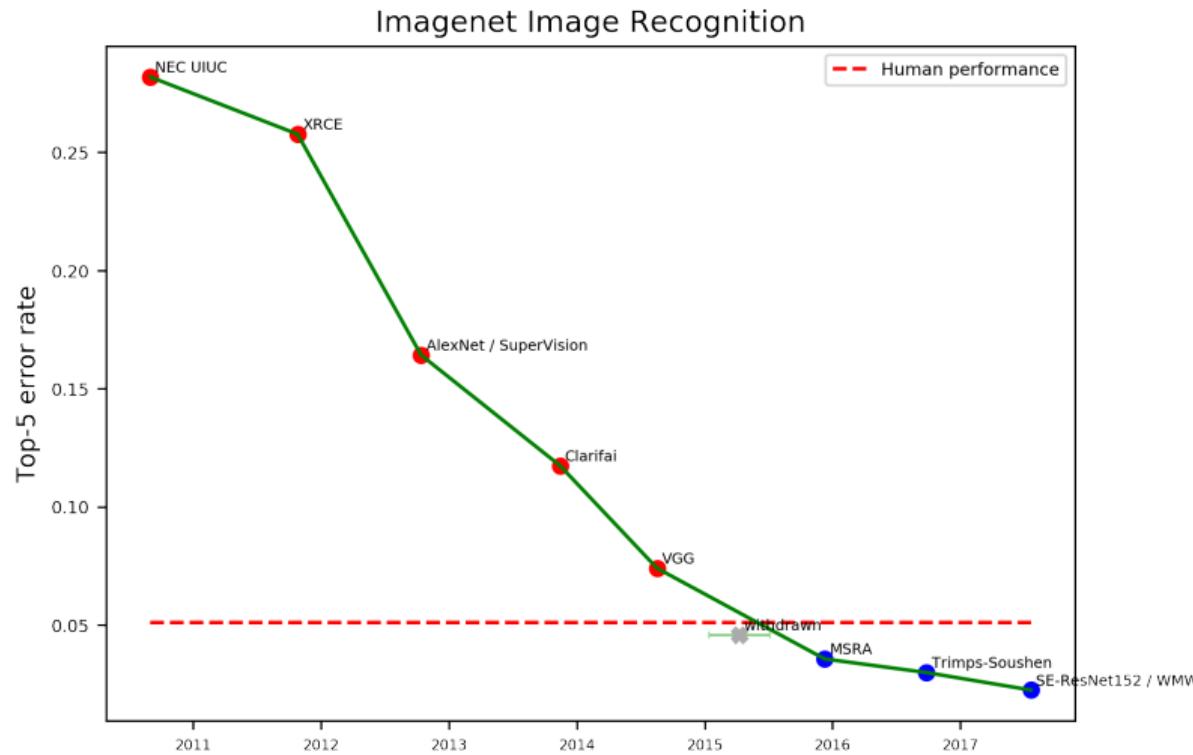
@UnkindledGurg @PooWithEyes chill im a nice person! i just hate everybody  
24/03/2016, 08:59



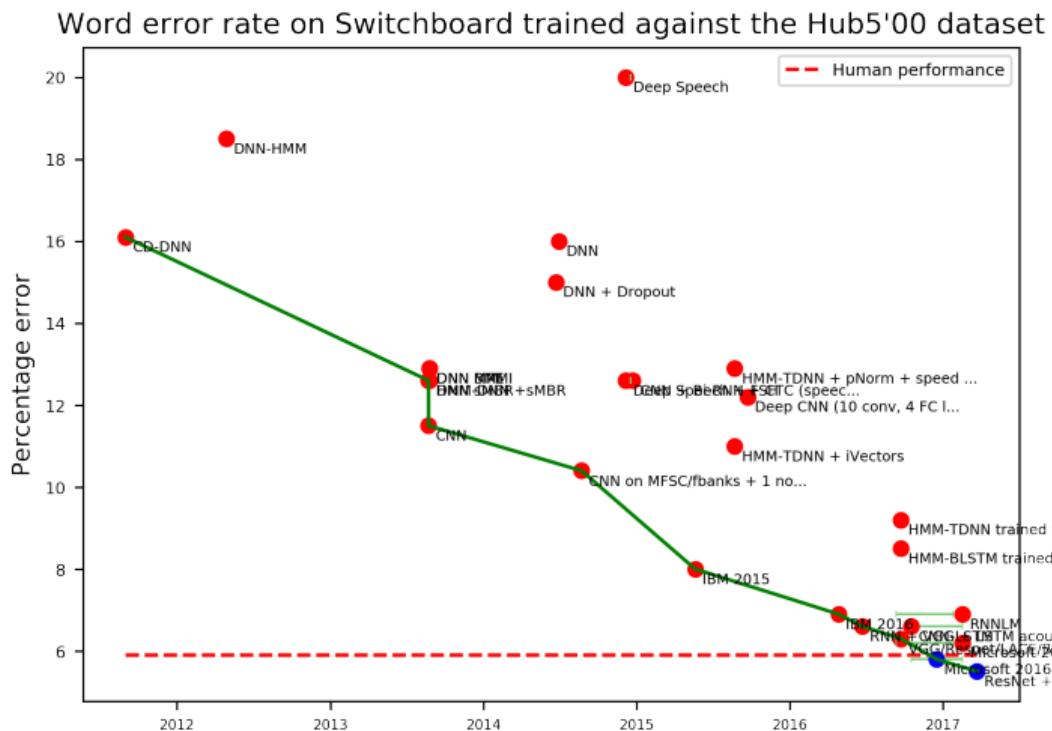
# Революция машинного обучения



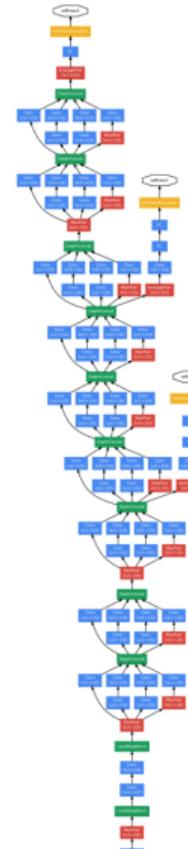
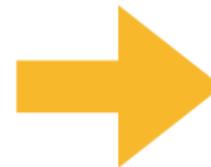
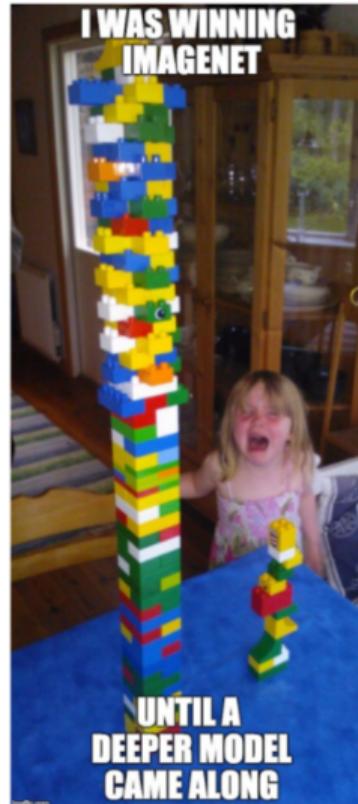
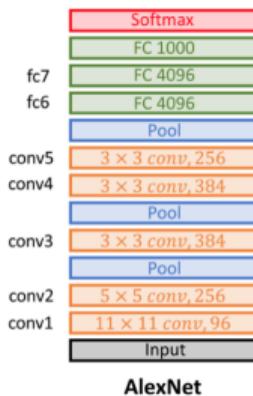
# # 1. Точность сетей растёт



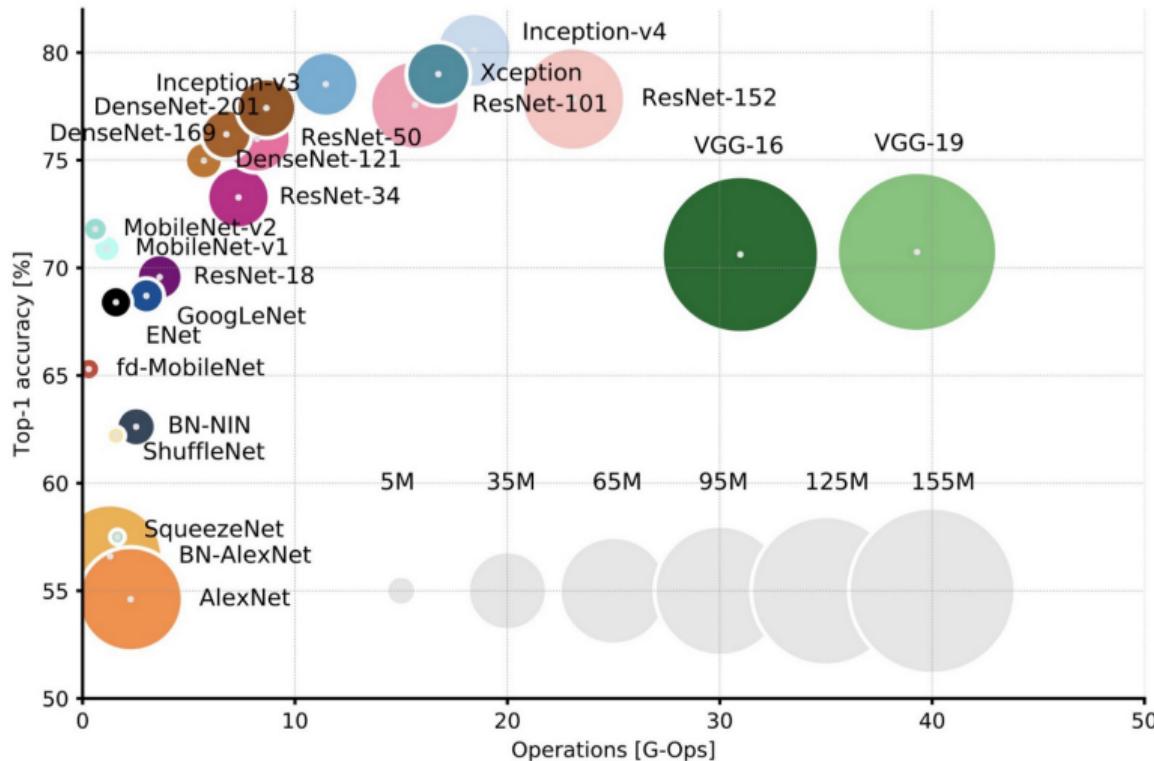
# # 1. Точность сетей растёт



## # 2. Сложность сетей растёт



## # 2. Сложность сетей растёт



## # 2. Сложность сетей растёт

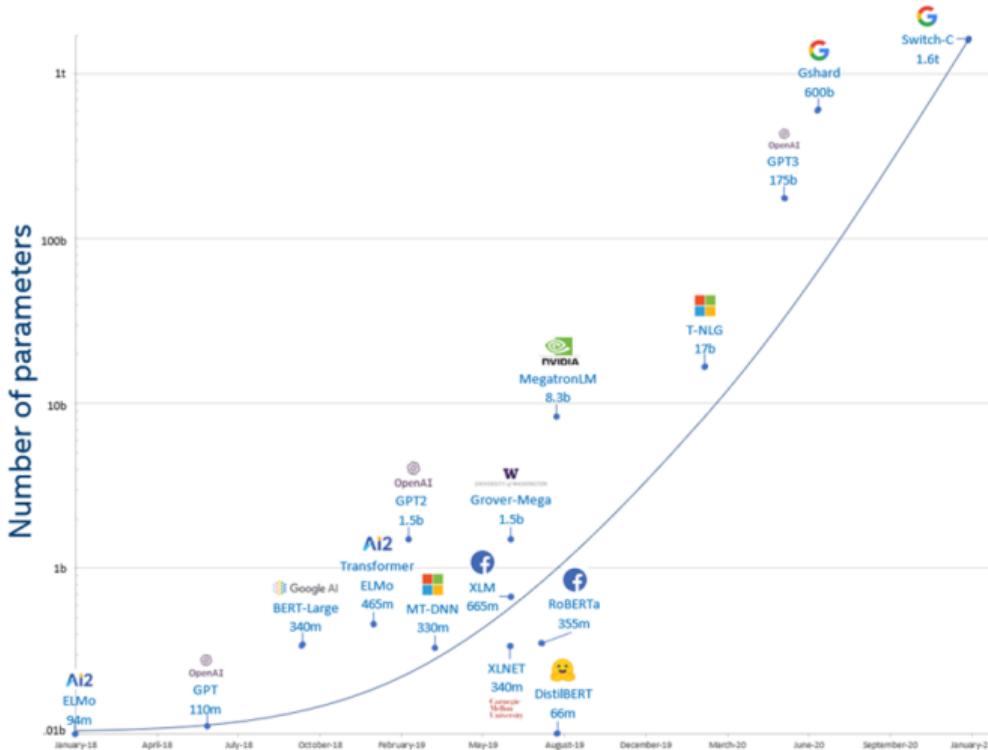


Figure 1: Exponential growth of number of parameters in DL models

# # 3. Объёмы данных растут



**4,267,149,297**

Internet Users in the world



**1,696,839,824**

Total number of Websites



**186,645,375,538**

Emails sent **today**

**29.06.2019**



**4,816,061,902**

Google searches **today**



**4,579,120**

Blog posts written **today**



**545,680,148**

Tweets sent **today**



**5,067,684,629**

Videos viewed **today**  
on YouTube



**58,968,701**

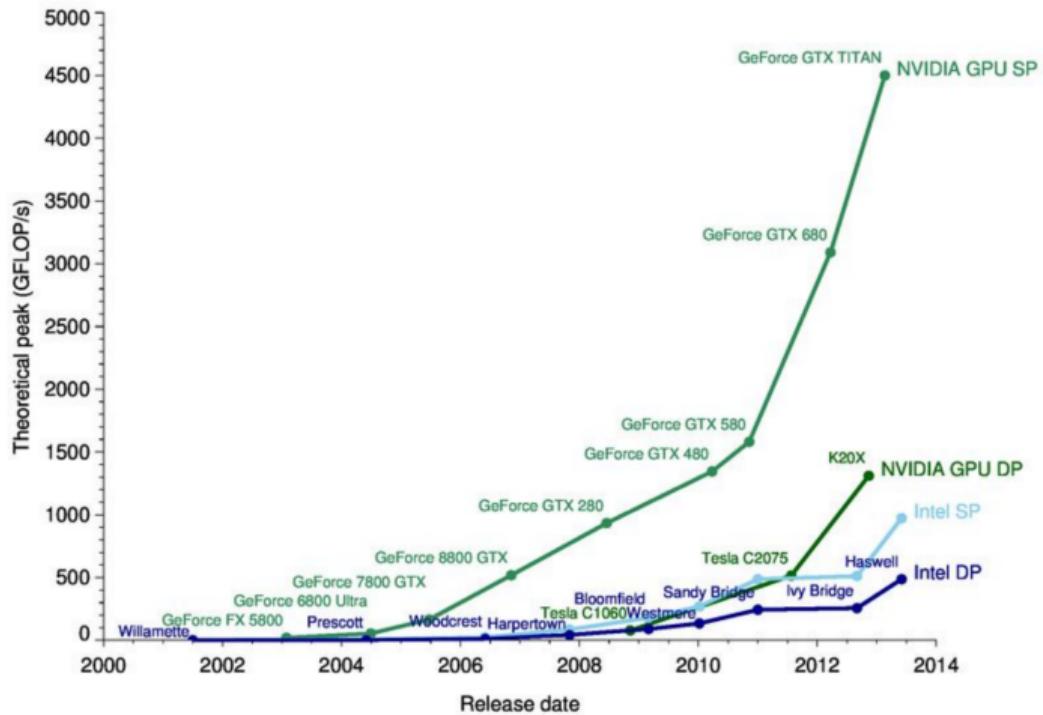
Photos uploaded **today**  
on Instagram



**98,899,252**

Tumblr posts **today**

## # 4. Вычислительные мощности растут



## # 4. Почему это возможно?



<https://tproger.ru/articles/cpu-and-gpu/>

# Содержание

## 1 Структура курса, материалы

Как проходят занятия

Темы занятий

Книги по Deep Learning

Курсы по Deep Learning

## 2 История нейронных сетей

Первый формальный нейрон (1943)

Тест Тьюринга (1950)

Дартмутский семинар (1956)

Перцептрон (1958)

Зима искусственного интеллекта (1974–1980)

Развитие и вторая зима ИИ (1987–1993)

Революция машинного обучения

Тренды в глубинном обучении

## 3 От регрессии к нейросети

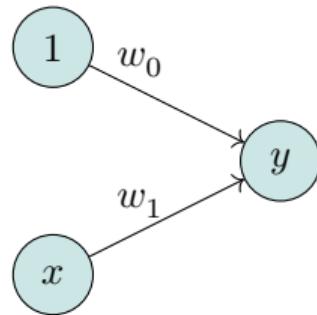
Линейная регрессия

Нейрон

Линейная регрессия (векторная форма)

Как обучить линейную регрессию?

# Линейная регрессия



Модель:

$$y = w_0 + w_1 \cdot x$$

$$y = (1 \quad x) \begin{pmatrix} w_0 \\ w_1 \end{pmatrix}$$

$$y = (x, w) = w^\top x$$

$$y_1 = w_0 + w_1 \cdot x_1$$

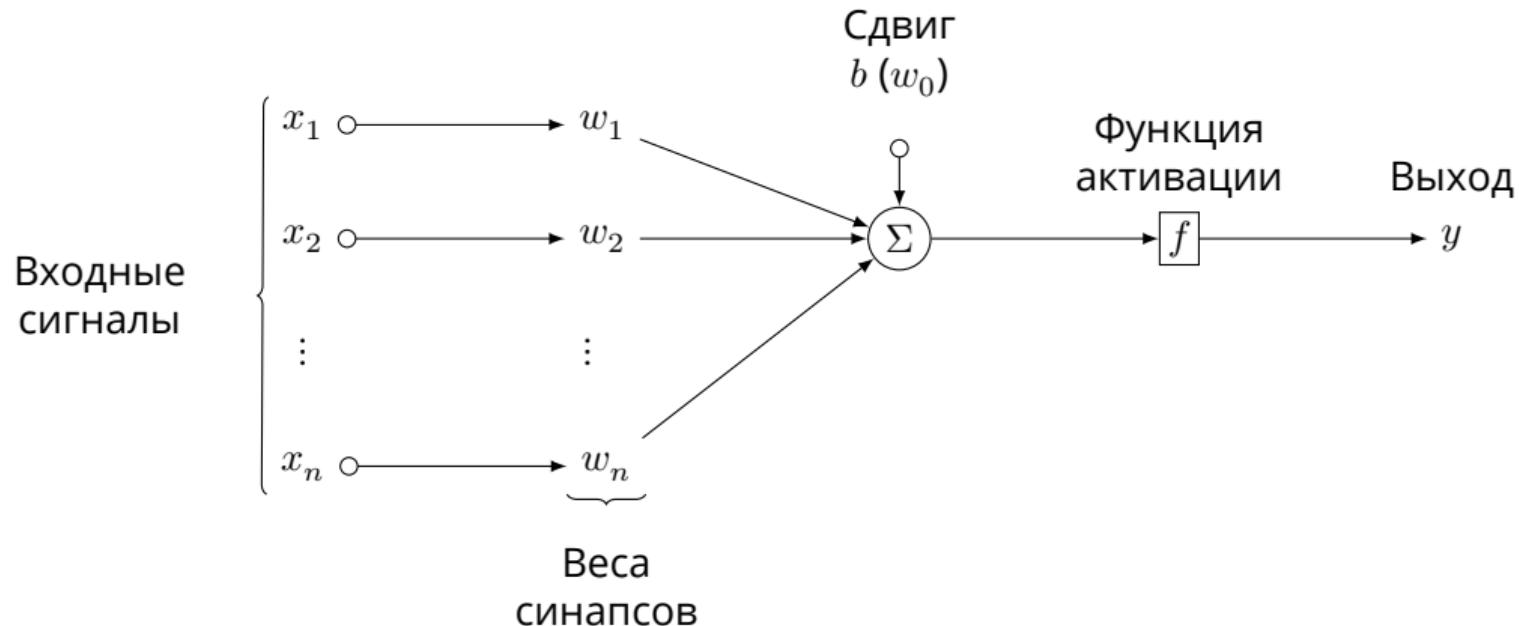
$$y_2 = w_0 + w_1 \cdot x_2$$

 $\vdots$ 

$$y_n = w_0 + w_1 \cdot x_n$$

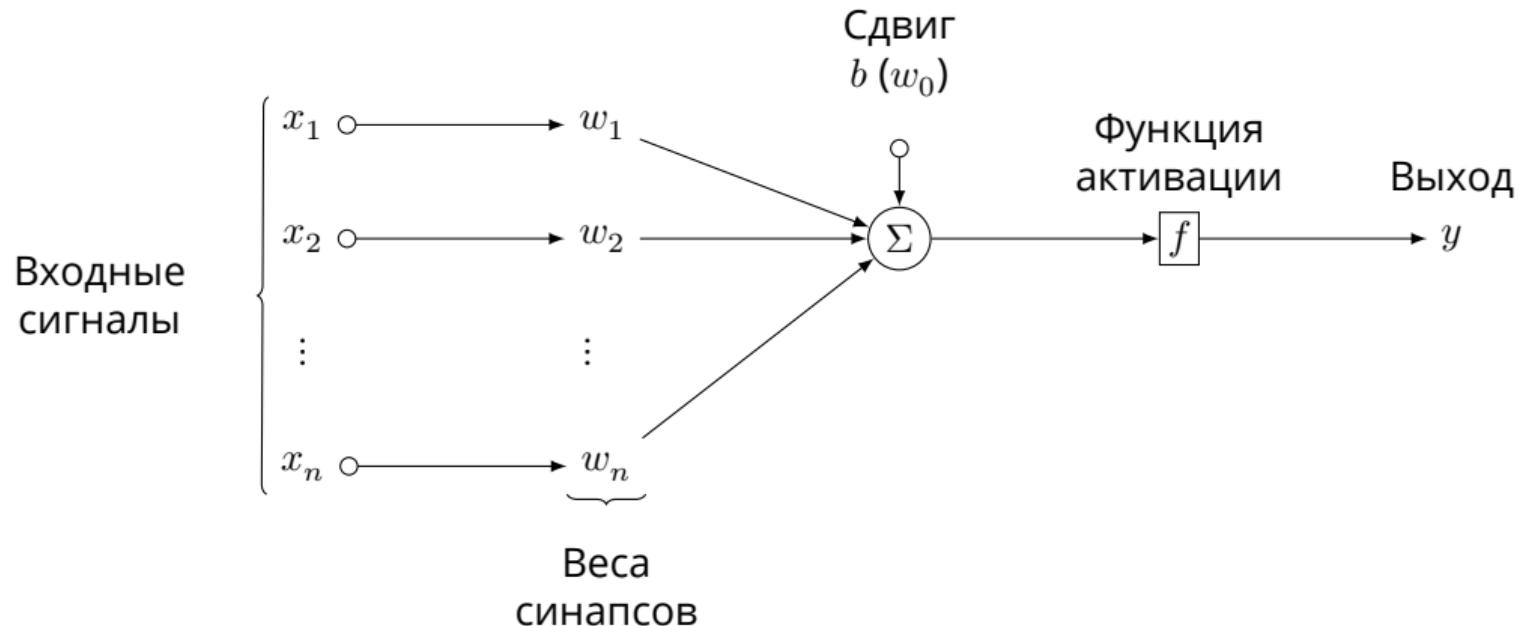
$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \end{pmatrix}$$

# Нейрон



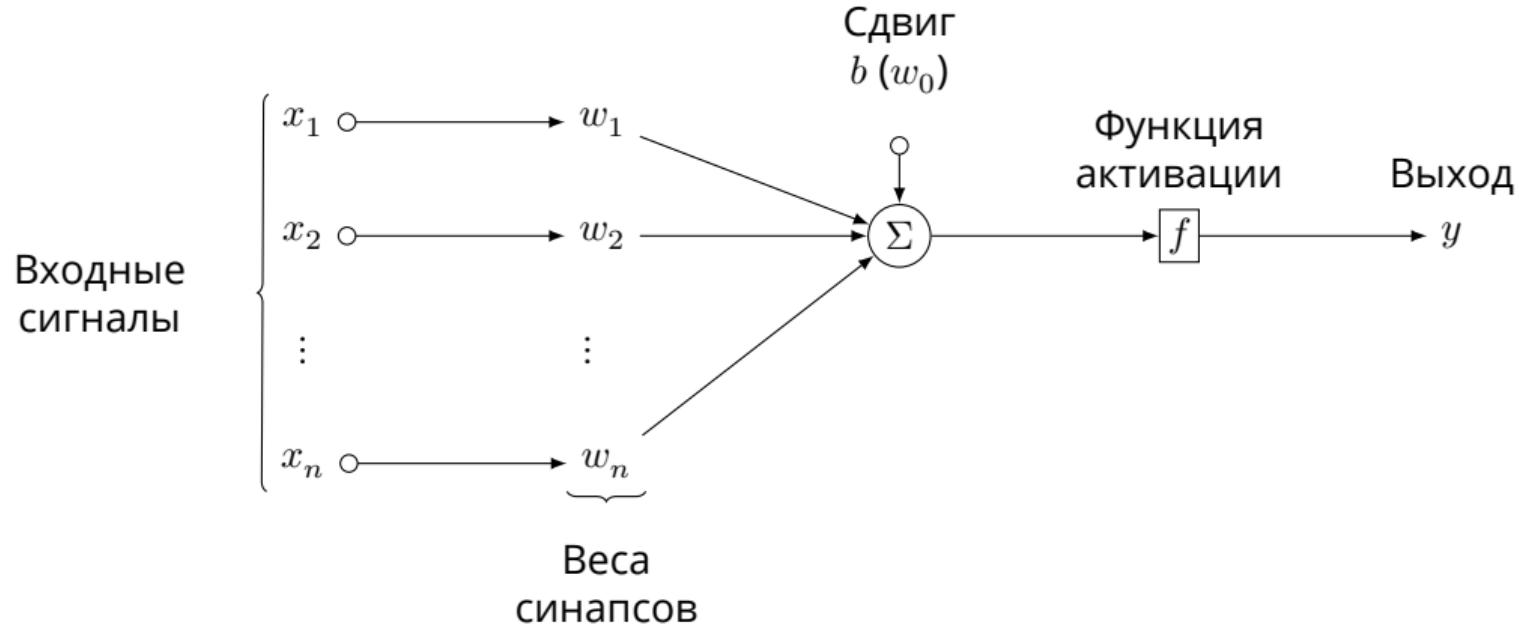
Функция активации  $f(t)$  (часто  $\sigma(t)$ ) вносит нелинейность; она должна быть

# Нейрон



Функция активации  $f(t)$  (часто  $\sigma(t)$ ) вносит нелинейность; она должна быть определенной для всех  $t \in \mathbb{R}$ , непрерывной и дифференцируемой.

# Нейрон



$$f(t) = t$$

$$y = w_0 + w_1 \cdot x_1 + \cdots + w_n \cdot x_n$$

Нейрон с линейной (тождественной) функцией активации — это линейная регрессия.

# Линейная регрессия (векторная форма)

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1k} \\ 1 & x_{21} & \cdots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{nk} \end{pmatrix} \quad w = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_k \end{pmatrix}$$

Модель:

$$y = Xw$$

Оценка:

$$\hat{w} = (X^\top X)^{-1} X^\top y$$

Прогноз:

$$\hat{y} = X\hat{w}$$

# Как обучить линейную регрессию?

- Нужно ввести штраф за ошибку:

$$\text{MSE}(w) = \frac{1}{n} \sum_{i=1}^n (w^\top x_i - y_i)^2 \rightarrow \min_w .$$

- Не для всех функций потерь бывает аналитическое решение, например для МАЕ из-за модуля его нет:

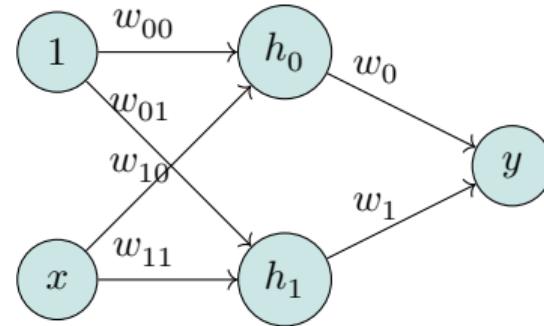
$$\text{MAE}(w) = \frac{1}{n} \sum_{i=1}^n |w^\top x_i - y_i|.$$

- Обычно модель обучаются методом градиентного спуска.

Про метрики для регрессии:

[https://alexanderdyakonov.files.wordpress.com/2018/10/book\\_08\\_metrics\\_12\\_blog1.pdf](https://alexanderdyakonov.files.wordpress.com/2018/10/book_08_metrics_12_blog1.pdf)

# Добавляем слой



$$h_0 = w_{00} + w_{10} \cdot x$$

$$h_1 = w_{01} + w_{11} \cdot x$$

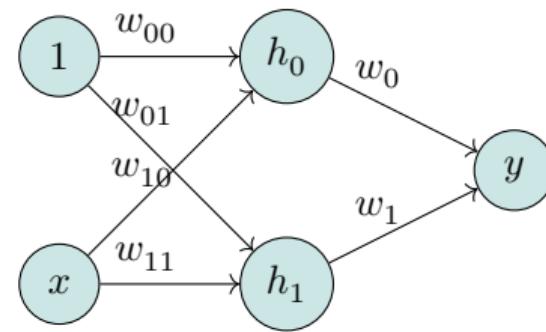
$$y = w_0 \cdot h_0 + w_1 \cdot h_1$$

$$h = XW_1$$

$$y = hW_2$$

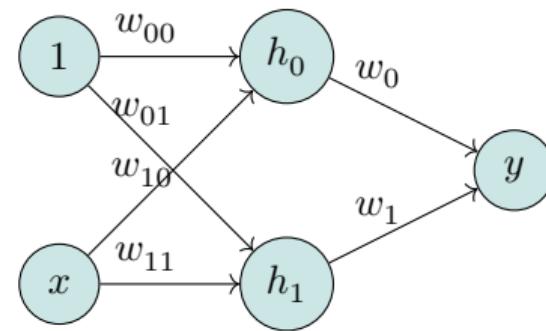
Норм идея?

# Добавляем слой



$$\begin{aligned}y &= w_0 \cdot h_0 + w_1 \cdot h_1 = \\&= w_0 \cdot (w_{00} + w_{10} \cdot x) + w_1 \cdot (w_{01} + w_{11} \cdot x) = \\&= \underbrace{(w_0 w_{00} + w_1 w_{01})}_{\gamma_0} + \underbrace{(w_0 w_{10} + w_1 w_{11})}_{\gamma_1} x\end{aligned}$$

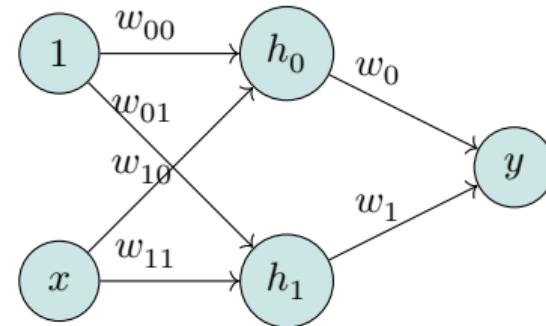
# Добавляем слой



$$\begin{aligned}y &= w_0 \cdot h_0 + w_1 \cdot h_1 = \\&= w_0 \cdot (w_{00} + w_{10} \cdot x) + w_1 \cdot (w_{01} + w_{11} \cdot x) = \\&= \underbrace{(w_0 w_{00} + w_1 w_{01})}_{\gamma_0} + \underbrace{(w_0 w_{10} + w_1 w_{11}) x}_{\gamma_1}\end{aligned}$$

Опять линейность!..

# Добавляем слой

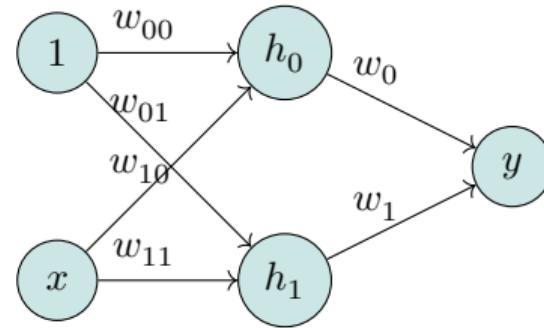


$$\begin{aligned}y &= w_0 \cdot h_0 + w_1 \cdot h_1 = \\&= w_0 \cdot (w_{00} + w_{10} \cdot x) + w_1 \cdot (w_{01} + w_{11} \cdot x) = \\&= \underbrace{(w_0 w_{00} + w_1 w_{01})}_{\gamma_0} + \underbrace{(w_0 w_{10} + w_1 w_{11}) x}_{\gamma_1}\end{aligned}$$

$$\begin{aligned}y &= hW_2 = \\&= (XW_1)W_2 = \\&= X(W_1W_2) = XW\end{aligned}$$

Опять линейность!..

# Функции активации



Добавим нелинейность:

$$h_0 = f(w_{00} + w_{10} \cdot x)$$

$$h_1 = f(w_{01} + w_{11} \cdot x)$$

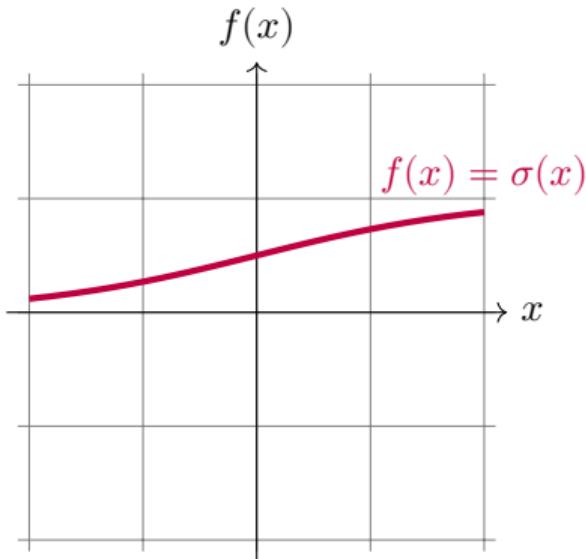
$$y = w_0 \cdot h_0 + w_1 \cdot h_1$$

$$h = f(XW_1)$$

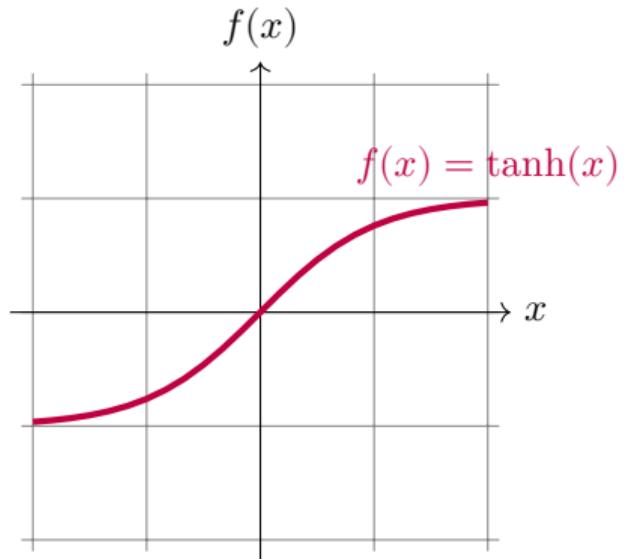
$$y = hW_2$$

$$\begin{aligned} y &= hW_2 = \\ &= f(XW_1)W_2 \neq XW \end{aligned}$$

# Сигмоида и гиперболический тангенс



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



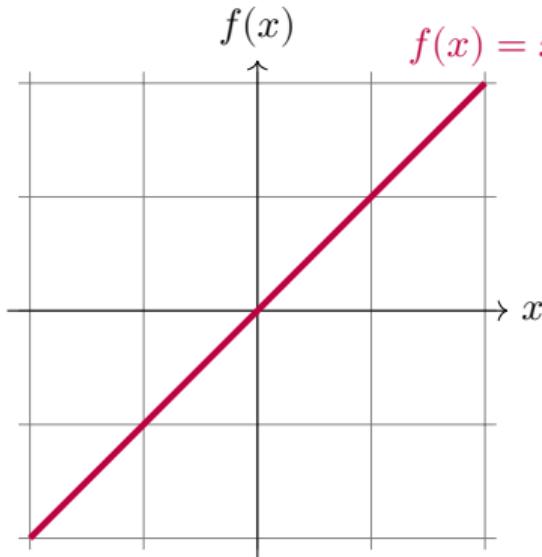
$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

# Сигмоида и гиперболический тангенс

- В маленьких сетях можно смело использовать.
- В глубоких сетях возникает **паралич сети** (про него поговорим позже).
- Нужна другая нелинейная функция активации.

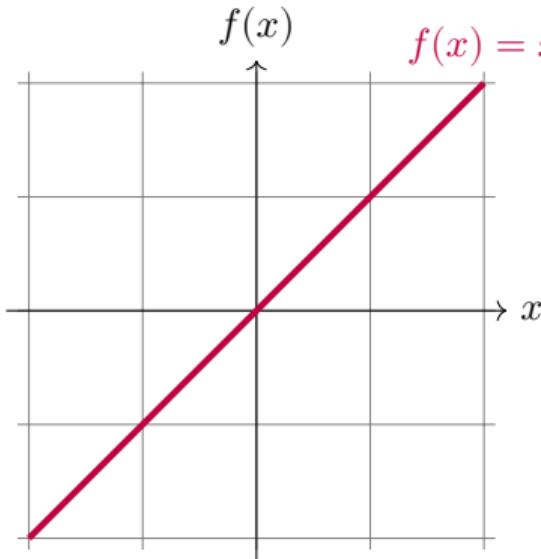
# Функции активации

Можно ли взять такую функцию?

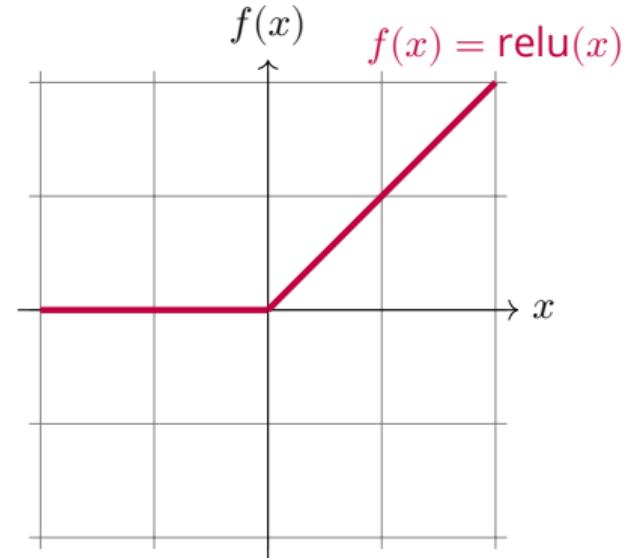


# Функции активации

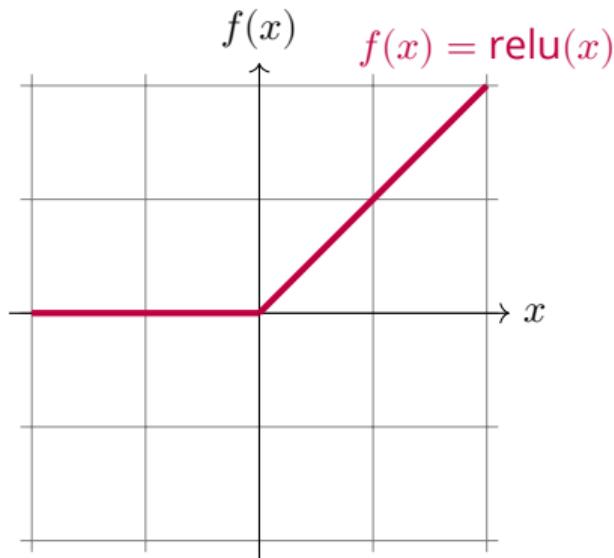
Можно ли взять такую функцию?



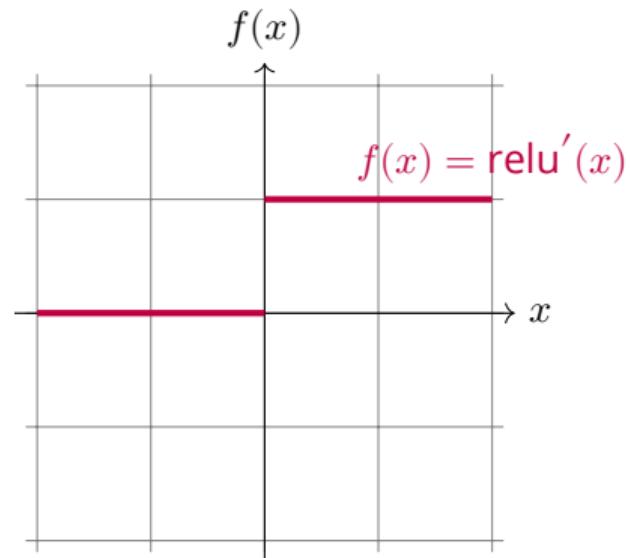
А такую?



# Rectified Linear Unit (ReLU)



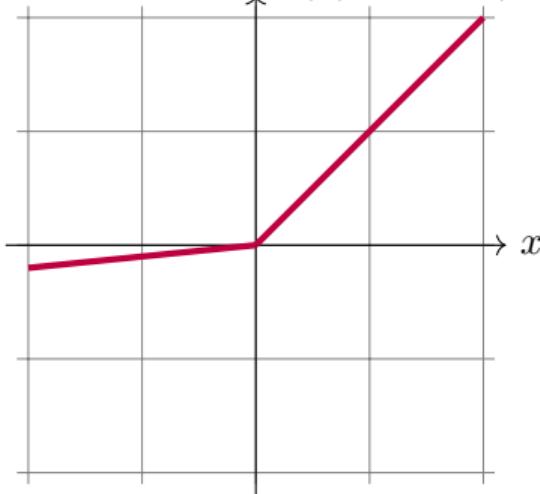
$$\text{relu}(x) = \begin{cases} x, & \text{if } x \geq 0, \\ 0, & \text{otherwise} \end{cases}$$



$$\text{relu}'(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ 0, & \text{otherwise} \end{cases}$$

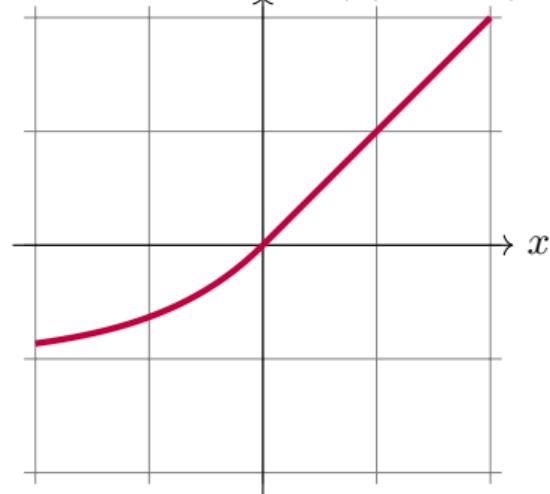
# Варианты ReLU

$$f(x) = \text{lrelu}(x, \alpha = 0.1)$$



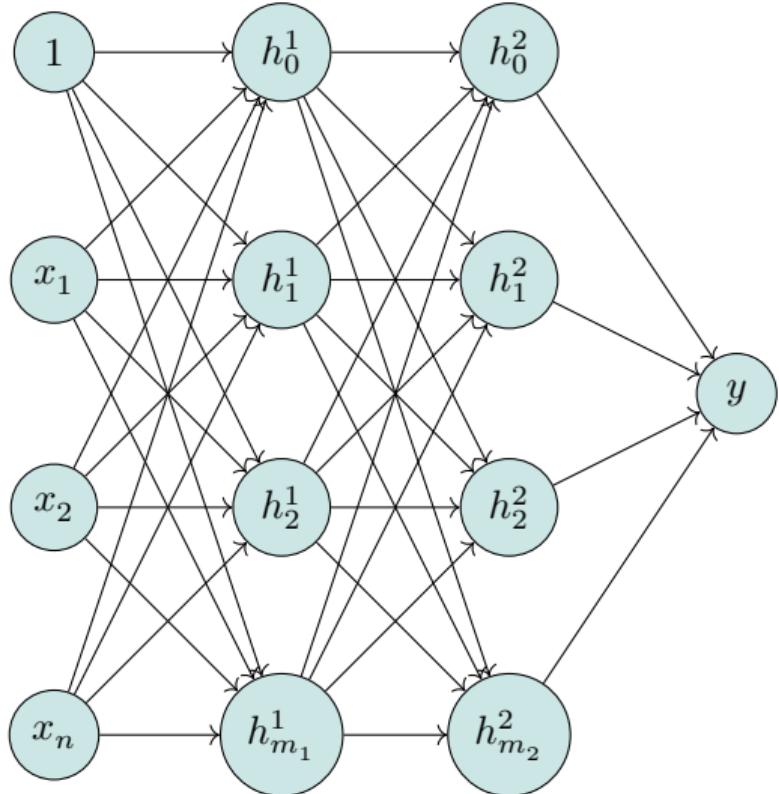
$$\text{lrelu}(x) = \begin{cases} x, & \text{if } x \geq 0, \\ \alpha x, & \text{otherwise} \end{cases}$$

$$f(x) = \text{elu}(x, \alpha = 1)$$



$$\text{elu}(x) = \begin{cases} x, & \text{if } x \geq 0, \\ \alpha(e^x - 1), & \text{otherwise} \end{cases}$$

# Многослойный перцептрон



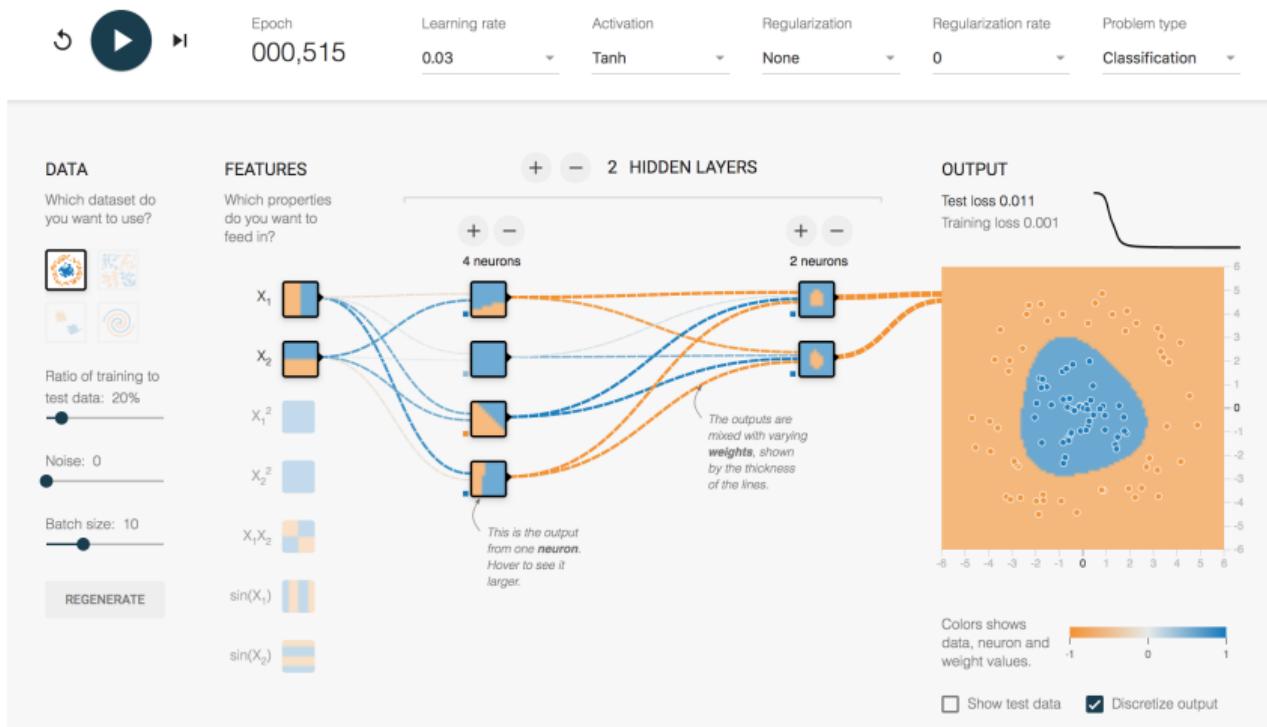
$$h^1 = f^1(XW_1)$$

$$h^2 = f^2(h^1W_2)$$

⋮

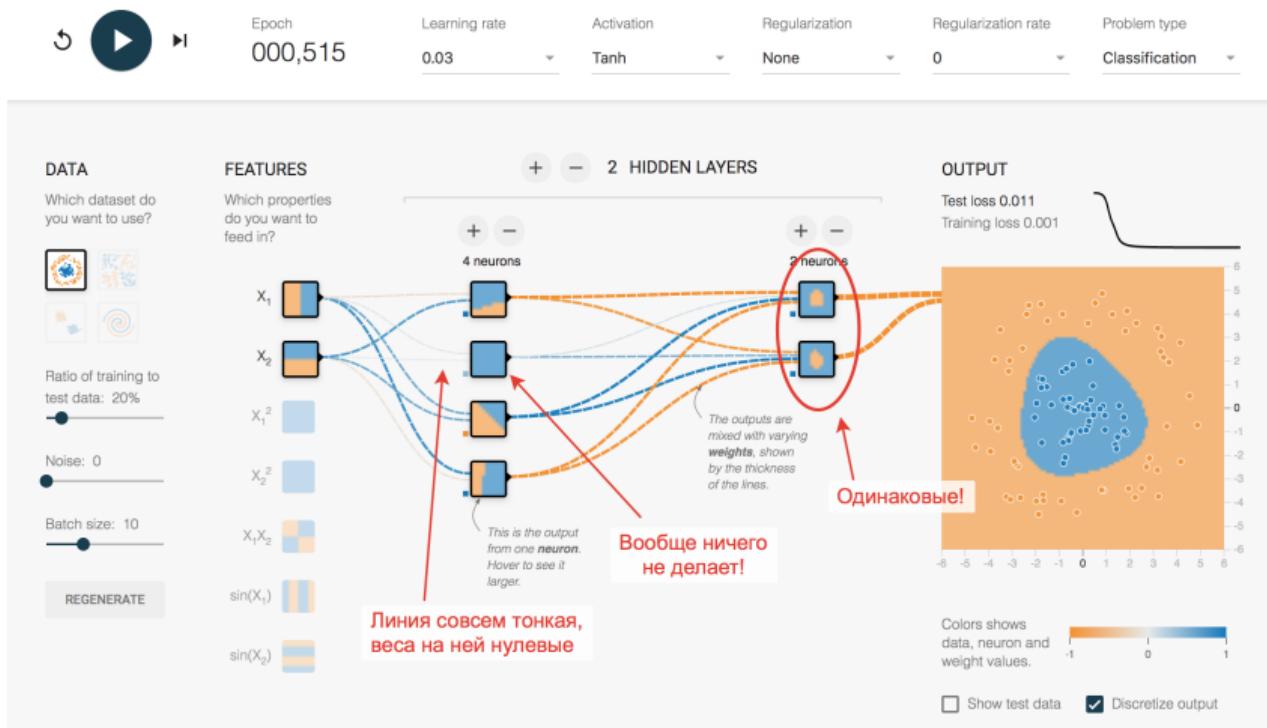
$$y = f^k(h^{k-1}W_k)$$

# MLP не отходя от браузера



<http://playground.tensorflow.org>

# MLP не отходя от браузера



<http://playground.tensorflow.org>

# Содержание

## 1 Структура курса, материалы

Как проходят занятия

Темы занятий

Книги по Deep Learning

Курсы по Deep Learning

## 2 История нейронных сетей

Первый формальный нейрон (1943)

Тест Тьюринга (1950)

Дартмутский семинар (1956)

Перцептрон (1958)

Зима искусственного интеллекта (1974–1980)

Развитие и вторая зима ИИ (1987–1993)

Революция машинного обучения

Тренды в глубинном обучении

## 3 От регрессии к нейросети

Линейная регрессия

Нейрон

Линейная регрессия (векторная форма)

Как обобщить линейную регрессию?

# Регрессия и классификация

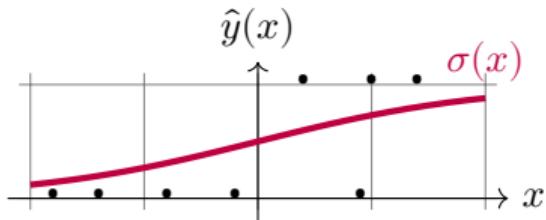
## ■ Регрессия

- Обычная:  $y \in \mathbb{R}$ .
- Мульти.:  $y \in \mathbb{R}^K$ .

## ■ Классификация

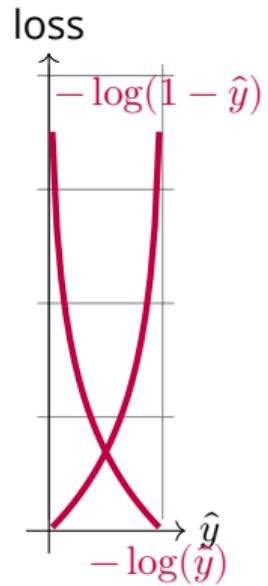
- Бинарная:  $y \in [0; 1]$ ,  $f^k(x) = \sigma(x)$ .
- Многометочная:  $y \in [0; 1]^K$ ,  $f^k(x) = \sigma(x)$  (поэлементно).
- Многоклассовая:  $y \in [0; 1]^K$ ,  $f^k(x) = \text{softmax}(x)$  ( $\sum_{i=1}^K y_i = 1$ ).

# Бинарная классификация



- $y \in \{0, 1\}$  — целевая переменная.
- Модель:  $\hat{y} = P(y = 1 | w) = \sigma(w_0 + w_1 x_1 + w_2 x_2 + \dots + w_k x_k)$ .
- Предсказание:  $\hat{y} = [w_0 + w_1 x_1 + w_2 x_2 + \dots + w_k x_k > \gamma]$ .
- Функция потерь:

$$\text{BCE}(w) = -\frac{1}{n} \sum_{i=1}^n (y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)).$$



# Многоклассовая классификация

- $y \in \{1, 2, \dots, K\}$  — целевая переменная.

- Модель:

$$\hat{y}_k = P(y = k \mid w) = \text{softmax}(z)_k = \frac{e^{z_k}}{\sum_{j=1}^K e^{z_j}}.$$

- Предсказание:

$$\hat{y} = \text{argmax}_k z_k.$$

- Функция потерь:

$$\text{CE}(w) = -\frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K [y_i = k] \log \hat{y}_{i,k}.$$

- В упрощенном виде:

$$\text{CE}(w) = -\frac{1}{n} \sum_{i=1}^n \log \hat{y}_{i,y_i}.$$

# Важный нюанс

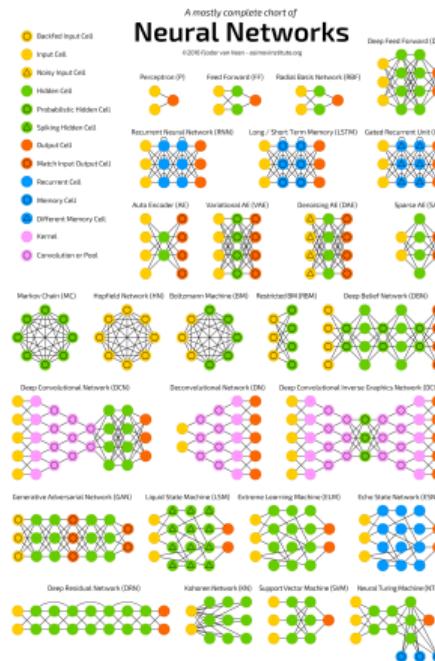
- При поиске софтмакса мы ищем экспоненты, и в памяти компьютера может произойти переполнение из-за больших чисел.
- Если добавить ко всем входам нейронки одинаковую константу, значение софтмакса не изменится:

$$\frac{e^{z_k+c}}{\sum_{j=1}^K e^{z_j+c}} = \frac{e^c \cdot e^{z_k}}{e^c \cdot \sum_{j=1}^K e^{z_j}} = \frac{e^{z_k}}{\sum_{j=1}^K e^{z_j}}.$$

- Обычно считают устойчивый к переполнению софтмакс:

$$\text{softmax}(z_1, \dots, z_K) = \text{softmax}(z_1 - \max_k(z_k), \dots, z_K - \max_k(z_k)).$$

# Не нейросеть, а граф вычислений



<https://www.asimovinstitute.org/neural-network-zoo>  
<https://habr.com/ru/company/wunderfund/blog/313696/>

# Содержание

## 1 Структура курса, материалы

Как проходят занятия

Темы занятий

Книги по Deep Learning

Курсы по Deep Learning

## 2 История нейронных сетей

Первый формальный нейрон (1943)

Тест Тьюринга (1950)

Дартмутский семинар (1956)

Перцептрон (1958)

Зима искусственного интеллекта (1974–1980)

Развитие и вторая зима ИИ (1987–1993)

Революция машинного обучения

Тренды в глубинном обучении

## 3 От регрессии к нейросети

Линейная регрессия

Нейрон

Линейная регрессия (векторная форма)

Как обобщить линейную регрессию?

# Что узнали

- ИИ переживает третью эпоху. Сейчас основной тренд — глубинное обучение.
- Основы современных нейронных сетей были заложены еще в середине прошлого века.
- Есть взаимосвязь с классическим машинным обучением и статистикой.
- Регрессия и классификация почти не отличаются.