

Анализ данных

Хашин С.И.

<http://math.ivanovo.ac.ru/dalgebra/Khashin/index.html>

Ивановский университет

Искусственный интеллект

Иваново-2024

План

Нейроны

Количество

Процессор

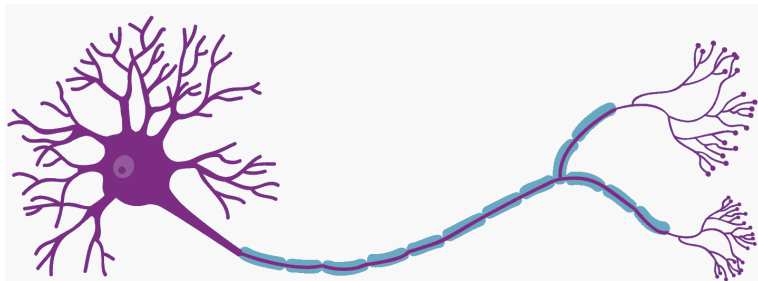
Программа

Данные

Сравнение

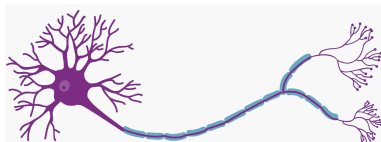
Питон

Нейрон



Слева — синапсы. Справа - аксон.

Что такое «нейрон»?

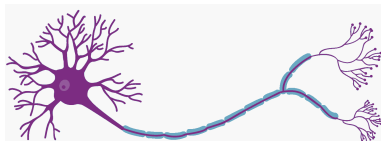


$$F = f(x_1 + x_2 + \dots + x_n).$$

где

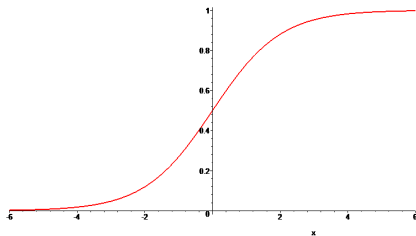
$$f(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geq 0 \end{cases}$$

Что такое «нейрон»?

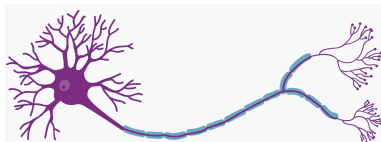


$$F = f(x_1 + x_2 + \dots + x_n).$$

где $f(t)$:



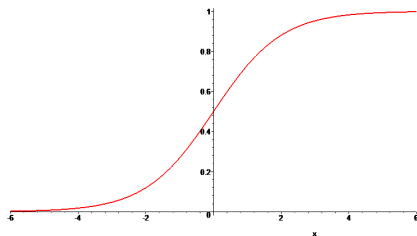
Что такое «нейрон»?



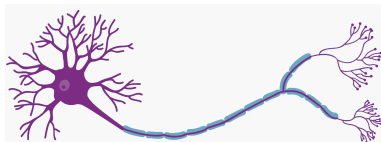
Веса нейрона:

$$F = f(w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n).$$

где $f(t)$:



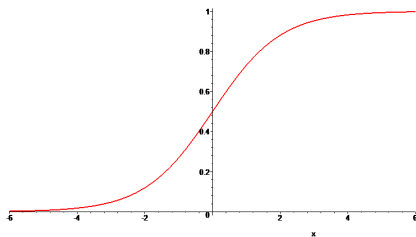
Что такое «нейрон»?



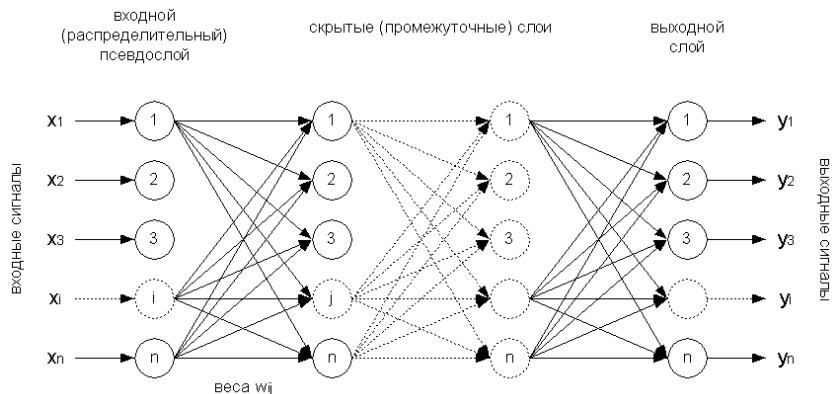
И ещё смещение (bias):

$$F = f(w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n).$$

где $f(t)$:



Нейронная сеть



Количество нейронов

Муха: 25К

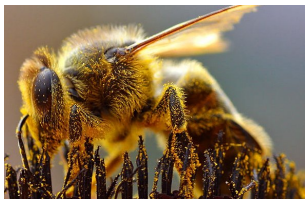


Комар: 200К



Количество нейронов

Пчела: 960К



Лягушка: 16М



Количество нейронов

Мышь: 71М

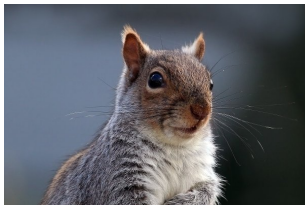


Синица: 226М



Количество нейронов

Белка: 450M



Кошка: 800M



Количество нейронов

Собака: 2.2Г



Макака: 3.8Г



Количество нейронов

Лев: 4.7Г



Медведь: 9.6Г

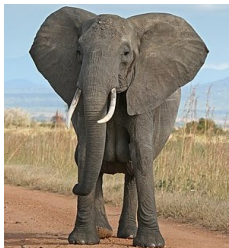


Количество нейронов

Орангутан: 33Г



Слон: 257Г



Количество нейронов

Человек: 86Г



Нейроны не складываются! Собрав 100 кошек, мы не получим интеллект человека!

Количество нейронов

муха	25К	комар	200К
пчела	960К	лягушка	16М
мышь	71М	синица	226М
белка	450М	кошка	800М
собака	2.3Г	макака	3.8Г
лев	4.7Г	медведь	9.6Г
орангутан	33Г	слон	257Г

Человек: 86Г

Сложность процессора

Количество транзисторов.

Процессор	год	разр.	Mhz	нм.	транз.
i8008	1972	8	0.2	10000	35K
i8086	1978	8	8	3000	29K
i80286	1982	16	10	1500	134K
i80486	1989	32	25	1000	1.2M
Pentium 60	1993	32	60	600	3.1M
Pentium4	2000	32	1300	180	42M
iCore2	2006	64	2660	65	291M
iCore i-7	2010	64	3333	32	731M
iBroadwell	2014	64	4000	14	1.9Г
Zen2	2019	64	4700	7	3.9Г
M1 ultra	2022	64	3200	5	114Г
Cerebras WSE3	2024	64	3200	5	4Т

M1 ultra

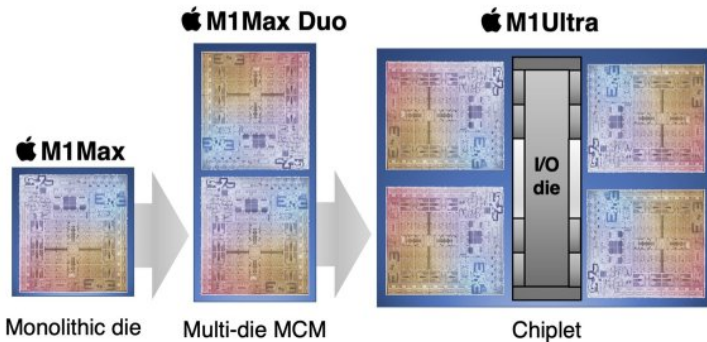
M1 ultra — процессор сравнительно обычных размеров:



Next generations after M1Max



@Frederic_Orange
invited to learn together



Cerebras WSE3

Cerebras WSE3 — это уже «монстр»:



Новости-24

01.07.2022 Представители компании TSMC заявляют, что 3-нм процесс потребляет на 45% меньше энергии, повышает производительность на 23% и использует на 16% меньшую площадь поверхности, чем 5-нм процесс.

20.10.2023 TSMC заявила, что её 3-нм технология сопоставима с перспективным 1.8-нм техпроцессом Intel

13.03.2024 Установлен первый литографический сканер ASML для выпуска 2-нм чипов по технологии Low-NA EUV

Что дальше?

Расстояние между атомами кремния в кристалле ≈ 0.3 нм.
То есть 3нм — это 10 атомов.

Какие перспективы:

1. Многослойные процессоры. Для них обязательно водяное охлаждение.
2. Охлаждение жидким азотом/водородом/гелием.
3. На сегодняшний день на один транзистор приходится порядка триллиона (10^{12}) атомов. Уменьшить это количество до миллиона (10^6).

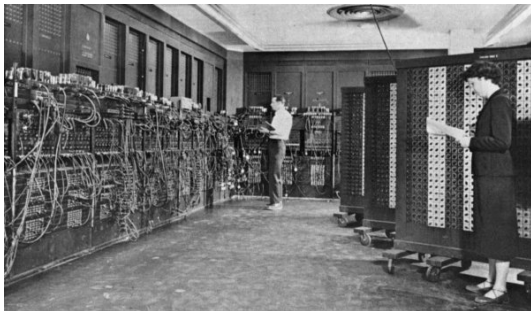
Таким образом, ещё есть принципиальная возможность увеличить количество транзисторов в миллион раз.

Но количество транзисторов — не самый главный показатель!

Архитектура процессора

Первые процессоры (ещё далеко не микро), появившиеся в начале 50-х годов прошлого века были очень простыми, но стали быстро усложняться.

Затем процессоры стали более сложными, более эффективными, увеличивалось количество регистров, разрядность, появилась конвейеризация и т.д..



Архитектура: Мини ЭВМ

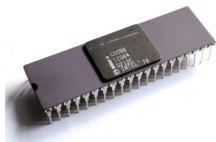
Первая революция связана с появлением PDP-8 (1965 г.), 12-битного компьютера и PDP-11 (1970 г.), 16-битового. Это были миникомпьютеры. Резкий скачок в эффективности был связан именно с переходом от больших, сложных, многоразрядных (64 и более бит) процессоров к самым простым, дешевым и эффективным.



Затем миникомпьютеры стали более сложными, более эффективными, увеличивалось количество регистров, разрядность, появилась конвейеризация и т.д..

Архитектура: Микропроцессоры

Вторая революция - Intel 8080 (1974 г., 8 бит) и Intel 8086 (1978 г., 8-16 бит) Это были микропроцессоры. Резкий скачок в эффективности был связан именно с переходом от больших, сложных, многоразрядных (64 и более бит) минипроцессоров к самым простым, дешевым и эффективным.



Затем микропроцессоры стали более сложными, более эффективными, увеличивалось количество регистров, разрядность, появилась конвейеризация и т.д..

Архитектура

Теперь нам надо ждать третью революцию. Резкий скачок в эффективности будет связан именно с переходом от больших, сложных, многоразрядных (64 и более бит) процессоров к самым простым, дешевым и эффективным.

Сейчас на наших глазах происходит следующая революция. Процессоры Core i9: 8 Р-ядер и 16 Е-ядер да ещё графический ускоритель.

Важная деталь: память интегрируется на кристалл!

Сложность программы

В нашем мозге "hard" и "soft" объединены вместе, нераздельны. В компьютере же — отдельно. И препятствием к появлению ИИ является именно недостаточная сложность программ. Количество **ОЧИЩЕННЫХ** строк в программах:

Программа	год	строк
Ядро Linux 0.1	1991	10K
Ядро Linux 1.0	1993	170K
Windows NT 3.1	1993	4M
Windows 2000	2000	29M
Windows 7	2009	42M
Windows 10	2015	55M
Ядро Linux 4.11.7	2017	12M
Android	2017	18M
GCC	2019	15M

Сложность программы

Вполне реалистичная программа обучения нейронной сети на keras-е выглядит так:

```
model = tf.keras.Sequential( [  
    tf.keras.layers.Dense(N1, activation = tf.nn.tanh,  
        input_shape=(k,)),  
    tf.keras.layers.Dense(N2, activation="tanh"),  
    tf.keras.layers.Dense(1, activation="linear"),] )  
model.compile(optimizer = tf.keras.optimizers.Adam(),  
    loss = tf.keras.losses.MeanSquaredError(),  
    metrics = [tf.keras.metrics.mean_squared_error])  
model.fit(x=X, y=Y, epochs=nEpochs)
```

Значит ли это, что её сложность 9 строк?

Сложность данных

В искусственном интеллекте есть показатель, который отсутствует в естественном — сложность данных.

Например, обученная нейронная сеть включает в себя, помимо самой программы, ещё и результаты обучения в виде нескольких массивов данных. Иногда они не слишком большие (килобайты), а иногда мегабайты и даже гигабайты.

В чем измерять сложность данных — непонятно. Конечно, не в байтах.

У нас в голове всё слито вместе: архитектура, программа, данные. В искусственном интеллекте эти понятия разделены.

Сравнение интеллекта

Пример — умение распознавать (узнавать) лицо человека. Эта возможность появилась у наших компьютеров в середине 2010-х годов.

Для этого компьютеру требуется операционная система, компиляторы, программы работы с графикой, нейронные сети и многое другое. Глядя на приведённую выше таблицу, можно ожидать, что для распознавания лиц требуются программы, общей сложностью около 100 М строк: операционная система, БД, библиотеки ML, компилятор и т.д..

В живой природе узнавать человека умеют делать животные, начиная с кошки, собаки, то есть примерно с 1 млрд. нейронов. У человека нейронов - почти в 100 раз больше.

Вывод

Для появления реального ИИ, должны появиться программные системы, в 100 раз более сложные, чем нынешние, порядка 10 млрд. строк кода программы.

Google утверждает, что у них суммарно около 2 млрд. строк кода.

Но строки не складываются!

Так же как и 100 кошек не дадут в сумме интеллект человека, то и записав на диск 10 000 программ по 1 млн. строк каждая, мы не получим требуемой сложности. Все эти программы должны работать в единой системе!

А.и Б. Стругацкие. Далекая радуга.

Полсотни лет назад в Массачусетсе запустили самое сложное кибернетическое устройство, когда-либо существовавшее. С каким-то там феноменальным быстродействием, необозримой памятью и все такое. . . И проработала эта машина ровно четыре минуты. Ее выключили, зацементировали все входы и выходы, отвели от нее энергию, заминировали и обнесли колючей проволокой. Самой настоящей ржавой колючей проволокой — хотите верить, хотите нет.

— А в чем, собственно, дело? — спросил Банин.

— Она начала вести себя, — сказал Горбовский.

— Не понимаю.

— И я не понимаю, но ее едва успели выключить.

— А кто-нибудь понимает?

— Я говорил с одним из ее создателей. Он взял меня за плечо, посмотрел мне в глаза и произнес только: «Леонид, это было страшно».

Питон — интерпретатор

Гармонический ряд: $S(n) = 1 + 1/2 + \dots + 1/n \approx \ln n$.

C: (1.15 секунды)

```
double S;  
for(int i=1; i<1000000000; i++) S+=1/i;
```

Питон: (5.82 сек.)

```
S = sum(1/k for k in range(1,10**9))
```

То есть Питон \approx в 5 раз медленнее, чем C++!

Пять лет назад Питон был медленнее, чем C в 30..70 раз в зависимости от процессора.

Но, фактически, Питон работает не медленнее, а часто даже быстрее, чем C!

Питон, примеры

Сортировка строк матрицы A по 7-му столбцу:

```
A[A[:,7].argsort()]
```

Из каждого столбца вычесть его среднее значение:

```
A -= A.mean(axis=0)
```

Удаление нулевых столбцов из матрицы A:

```
A =A[:,~(A == 0).all(0)]
```

Питон, примеры

Бинаризация признака со значениями (0,1,2,3):

```
v = np.array([[2,3,0,0,3,3]])
A = np.eye(4)[a]
print(A)
[[[0. 0. 1. 0.]      ! 2
  [0. 0. 0. 1.]      ! 3
  [1. 0. 0. 0.]      ! 0
  [1. 0. 0. 0.]      ! 0
  [0. 0. 0. 1.]      ! 3
  [0. 0. 0. 1.]]]   ! 3
```