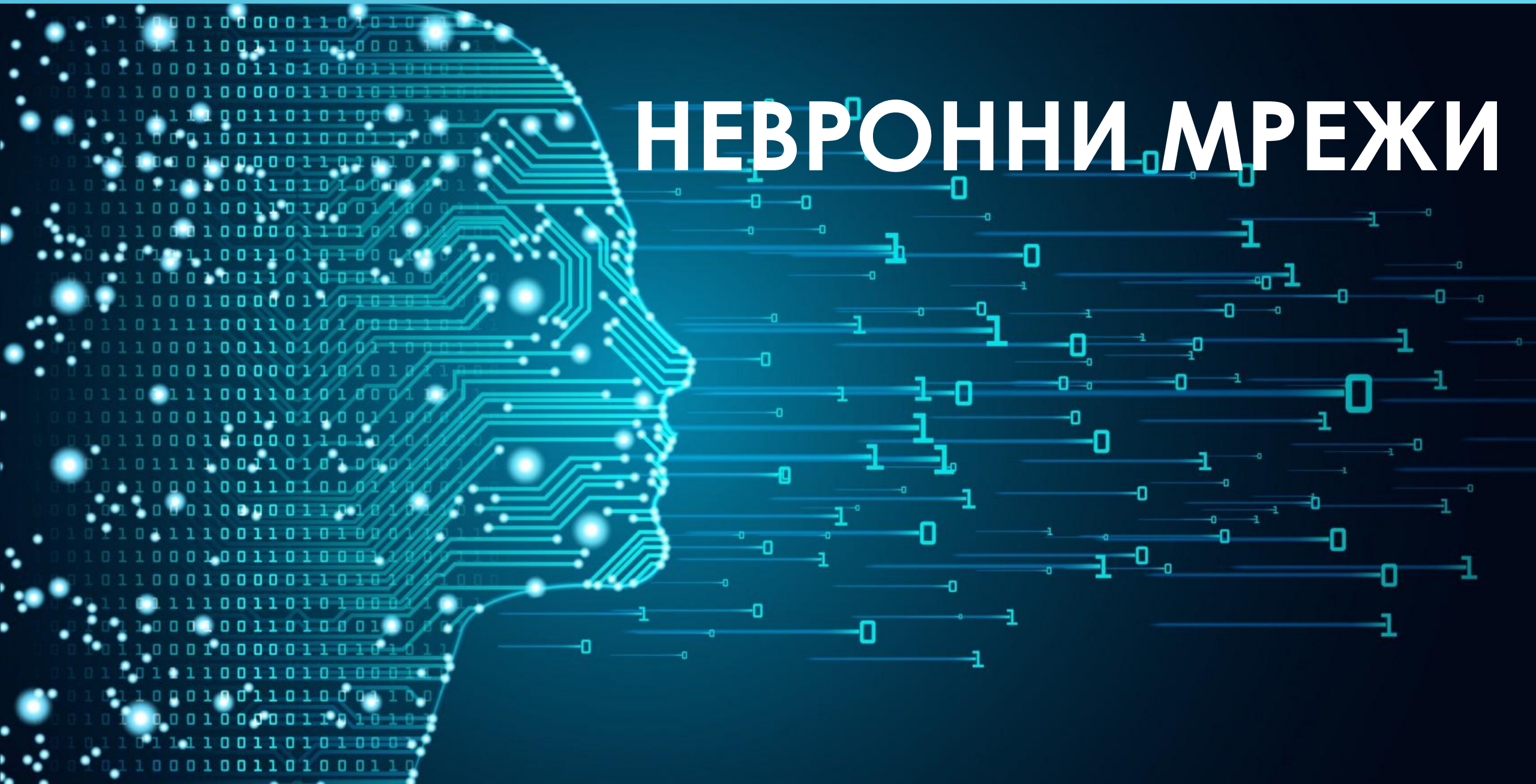


ИЗКУСТВЕН ИНТЕЛЕКТ



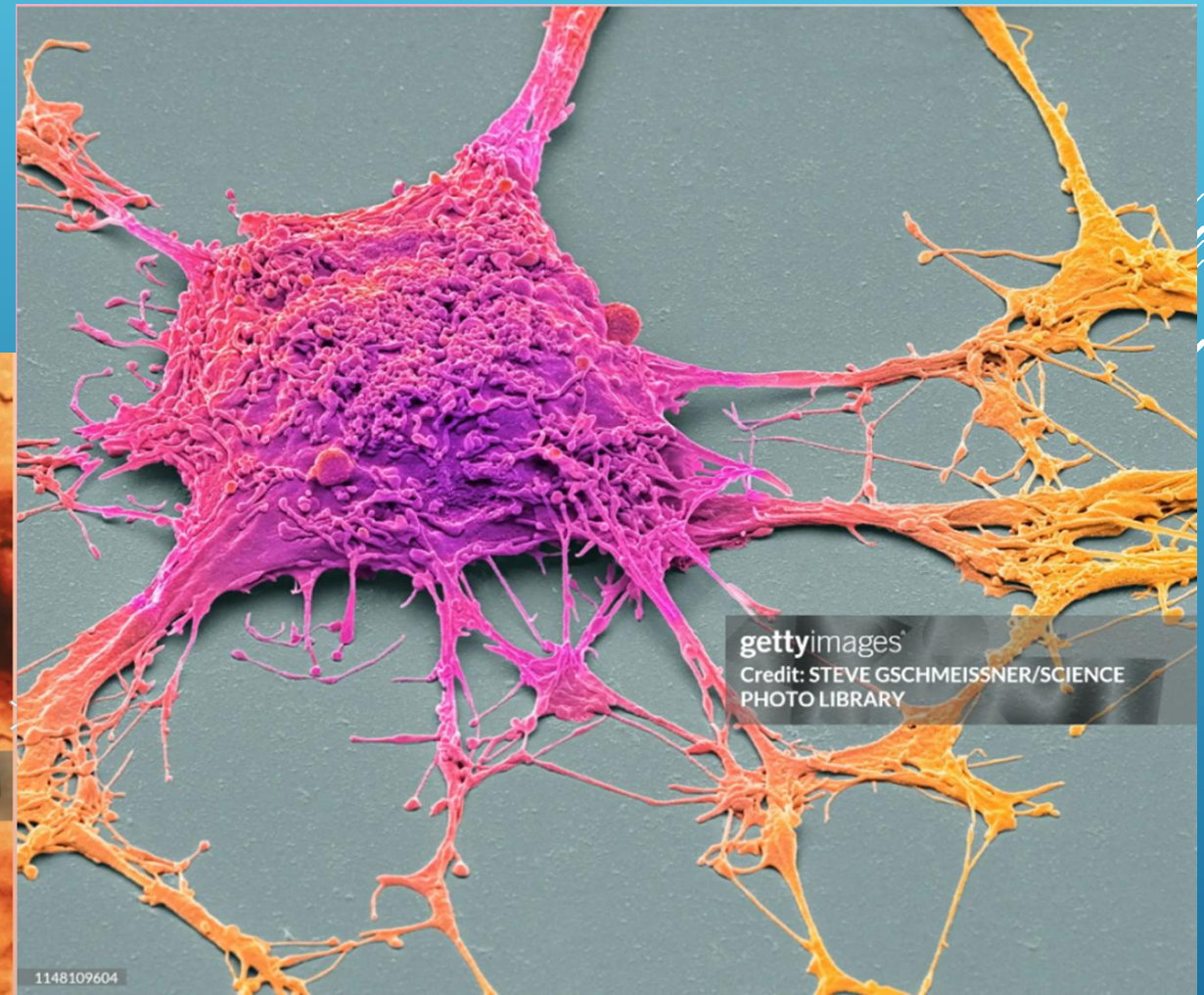
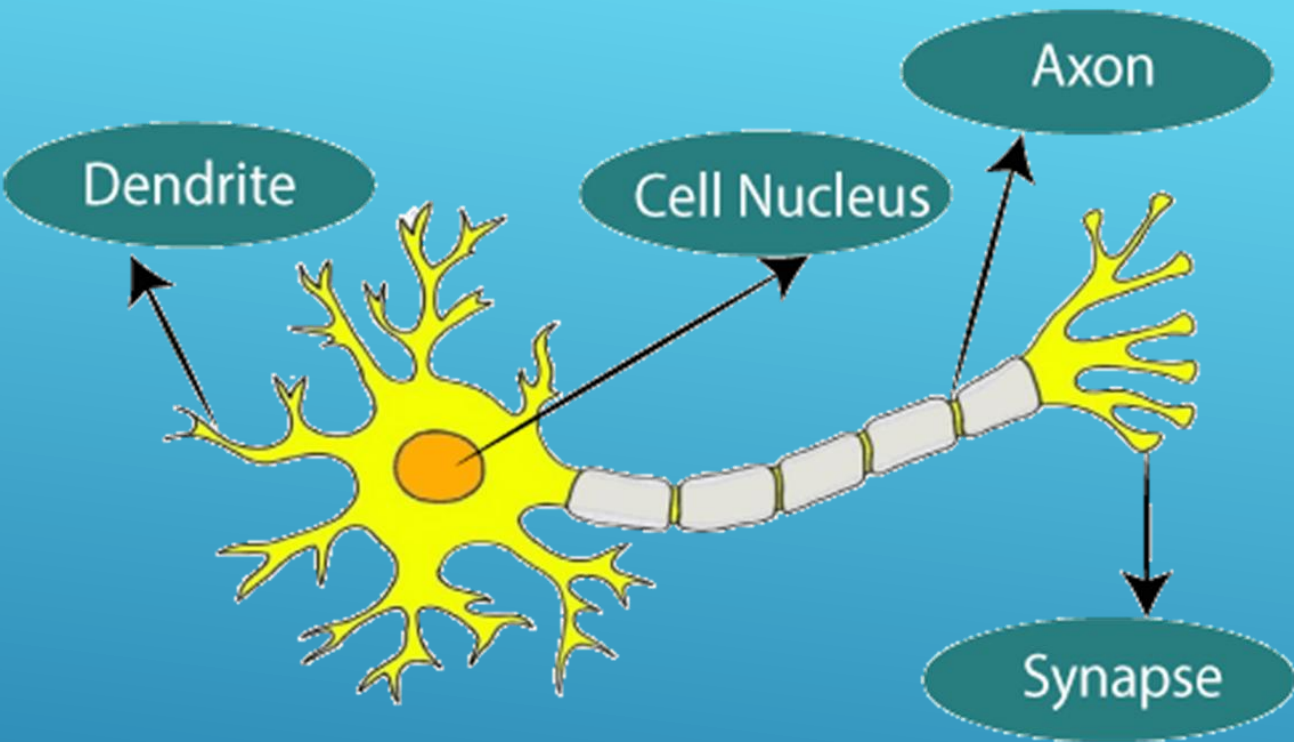
ИЗКУСТВЕН ИНТЕЛЕКТ

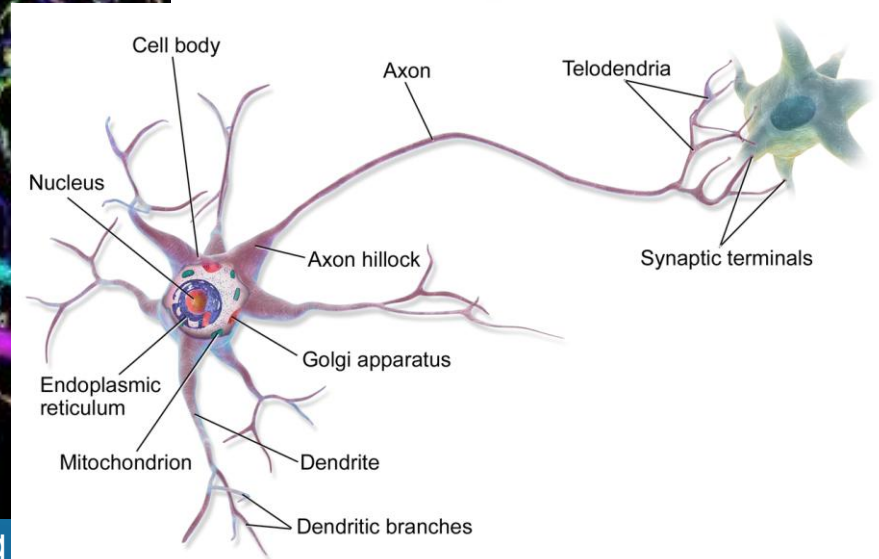
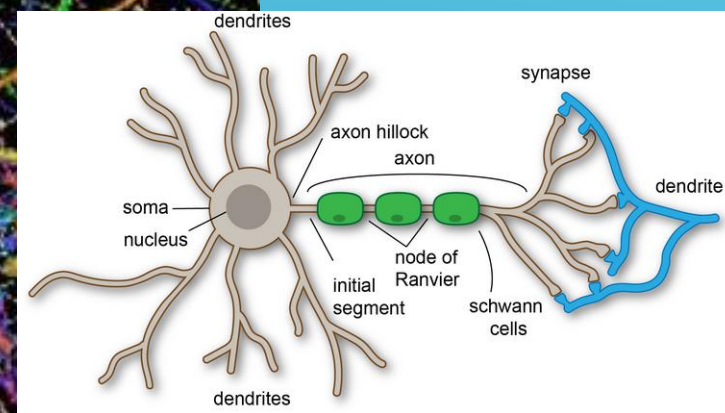
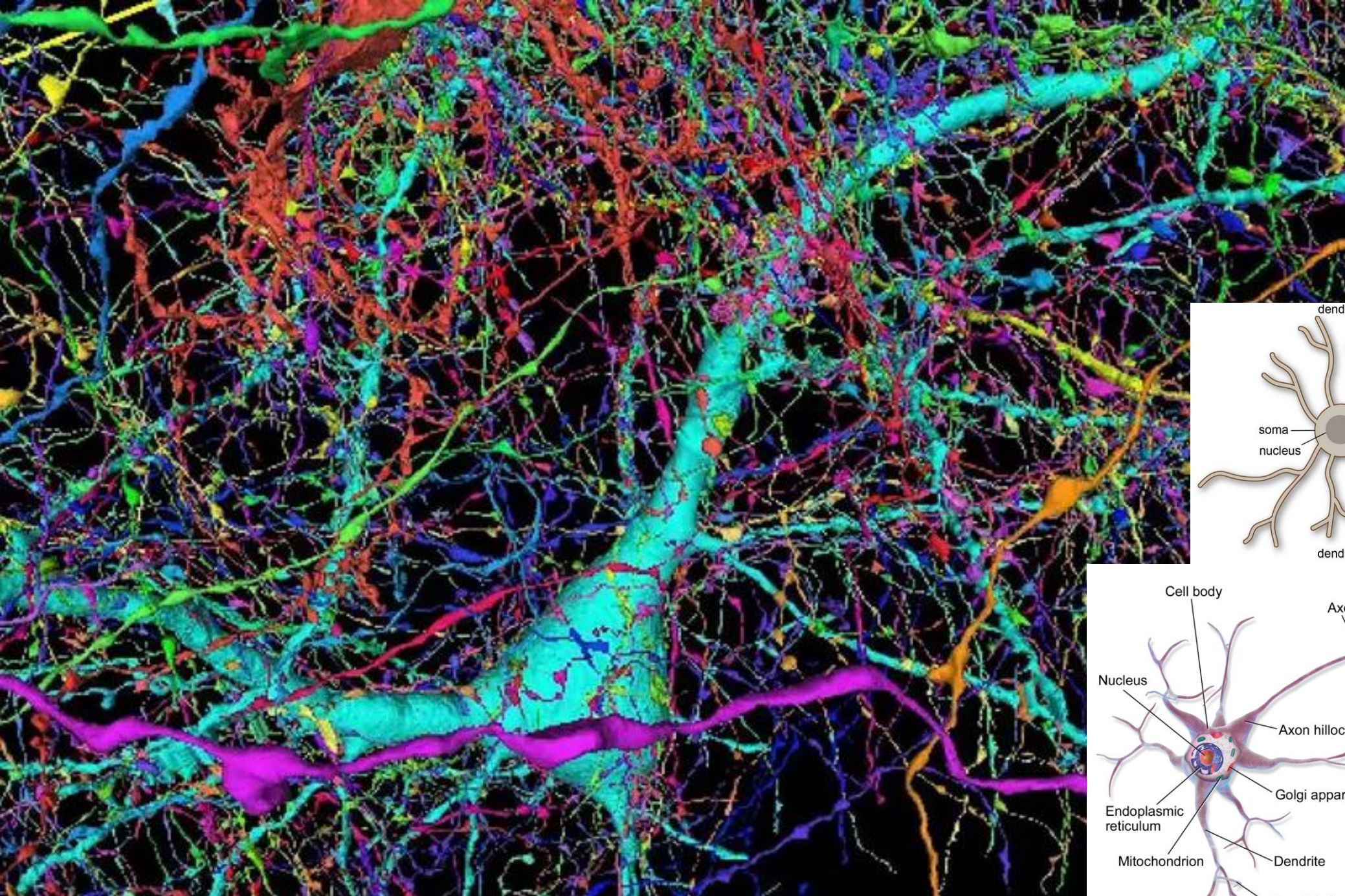
НЕВРОННИ МРЕЖИ



ЧОВЕШКИ НЕВРОН

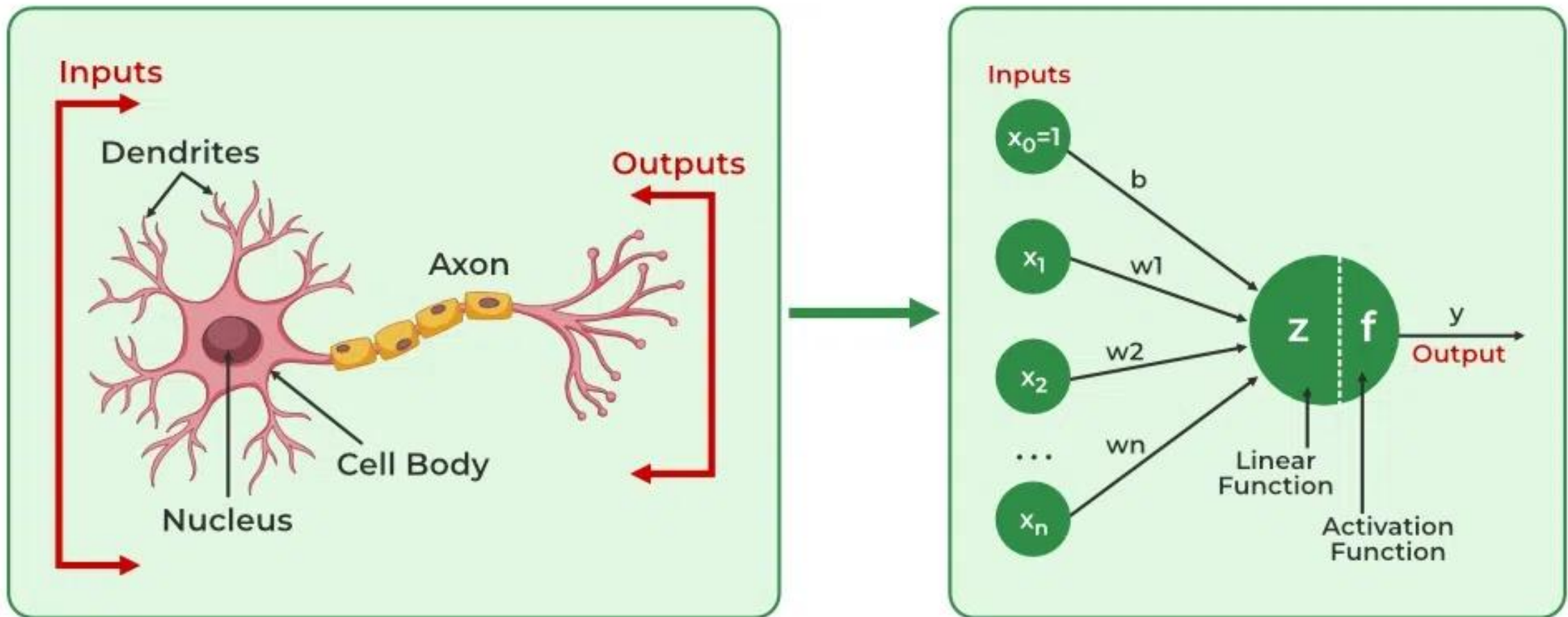
Човешкият мозък се състои от 86 милиарда нервни клетки, наречени неврони. Всеки неврон е свързан с хиляди други клетки чрез аксони.



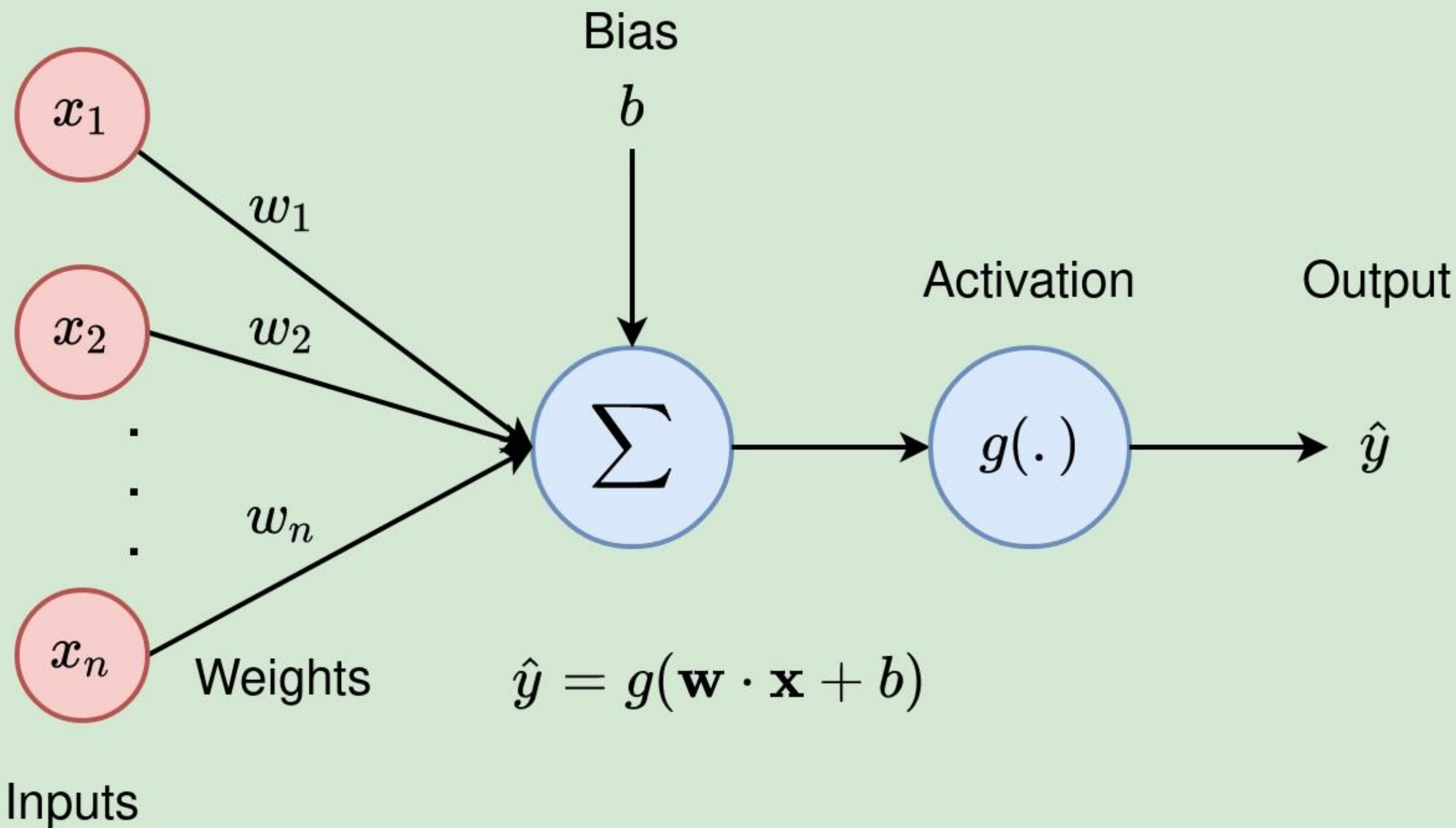


Around 4000 nerve fibres connect to this single neuron Google/Lichtma

ЕСТЕСТВЕН И ИЗКУСТВЕН НЕВРОН



Единичният неврон извежда продукт на функцията от входно ниво, заложеното отклонение/пристрастие (BIAS) и активиране (напр. Сигмоидна функция), който определя, дали той да се активира, или не. В допълнение, връзките между невроните също репрезентират тежести (под формата на тенденции). В процеса на обучение входът първо се обработва чрез предаване на информация от него, през невроните, в изхода (Forward Propagation). След това, стойностите на теглата и отклоненията се регулират, за да се сведе до минимум грешката между очаквания и действителния изход на невронната мрежа. Този процес се нарича обратно разпространение (Backpropagation или обратна връзка за грешка). Така, чрез многократно въвеждане на тренировъчни модели и автоматично минимизиране на грешката, мрежата се модифицира и напасва, докато стойността на грешката падне под предварително определена стойност. По този начин невронните мрежи се научават да организират информацията самостоятелно и осигуряват като изходен резултат генерализация.



В изкуствения неврон **bias (пристрастие)** е допълнителен параметър, който се използва за регулиране на изхода на неврона, независимо от входните стойности. Той играе ключова роля в адаптирането на модела, като позволява на неврона да се активира дори при нулеви входове.

Формула на изкуствения неврон:

Обикновено невронът изчислява своя изход чрез претеглена сума на входовете плюс bias:

$$y = f \left(\sum w_i x_i + b \right)$$

където:

- x_i са входните стойности,
- w_i са теглата на входовете,
- b е bias,
- f е активационната функция.

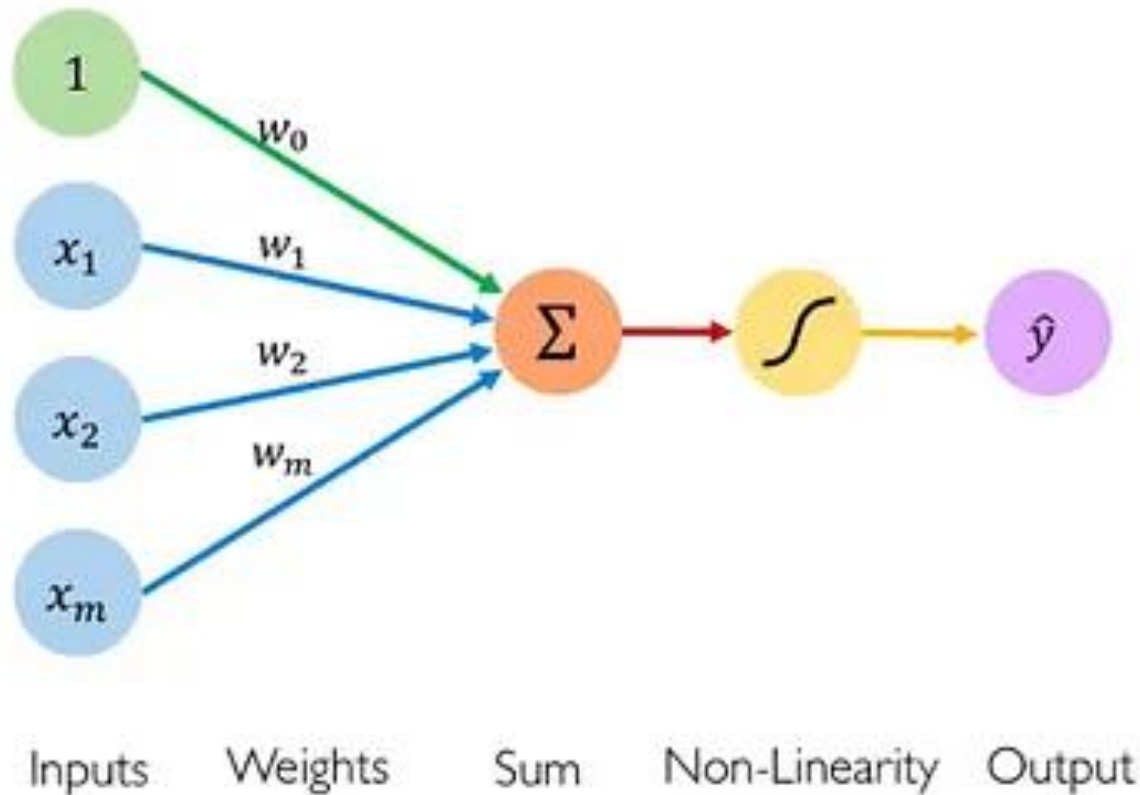
Защо е необходим bias?

1. Осигурява гъвкавост – Без bias, всички неврони биха преминавали през началото на координатната система (0,0), което ограничава способността им да научават различни зависимости.

2. Позволява по-добро обучение – Bias позволява на неврона да регулира границата на активацията си и да моделира по-сложни функции.

3. Аналогия с прага на активация в биологичните неврони – В реалните неврони има минимално количество възбуждане, необходимо за предизвикване на отговор.

The Perceptron: Forward Propagation

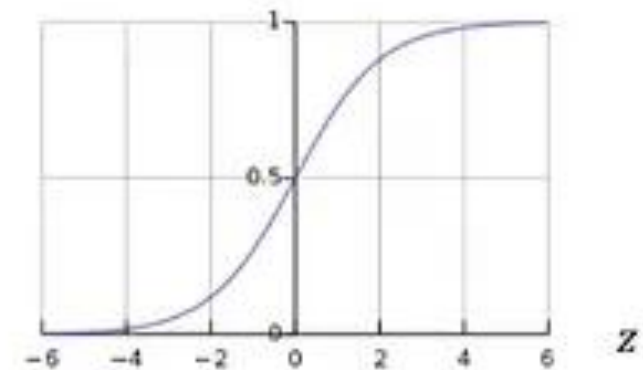


Activation Functions

$$\hat{y} = g(w_0 + X^T W)$$

- Example: sigmoid function

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



▶ Целта на активационната функция е да добави нелинейност към невронната мрежа.

Целта на активационната функция е да добави нелинейност към невронната мрежа.

ReLU

$$f(x) = \max(0, x)$$

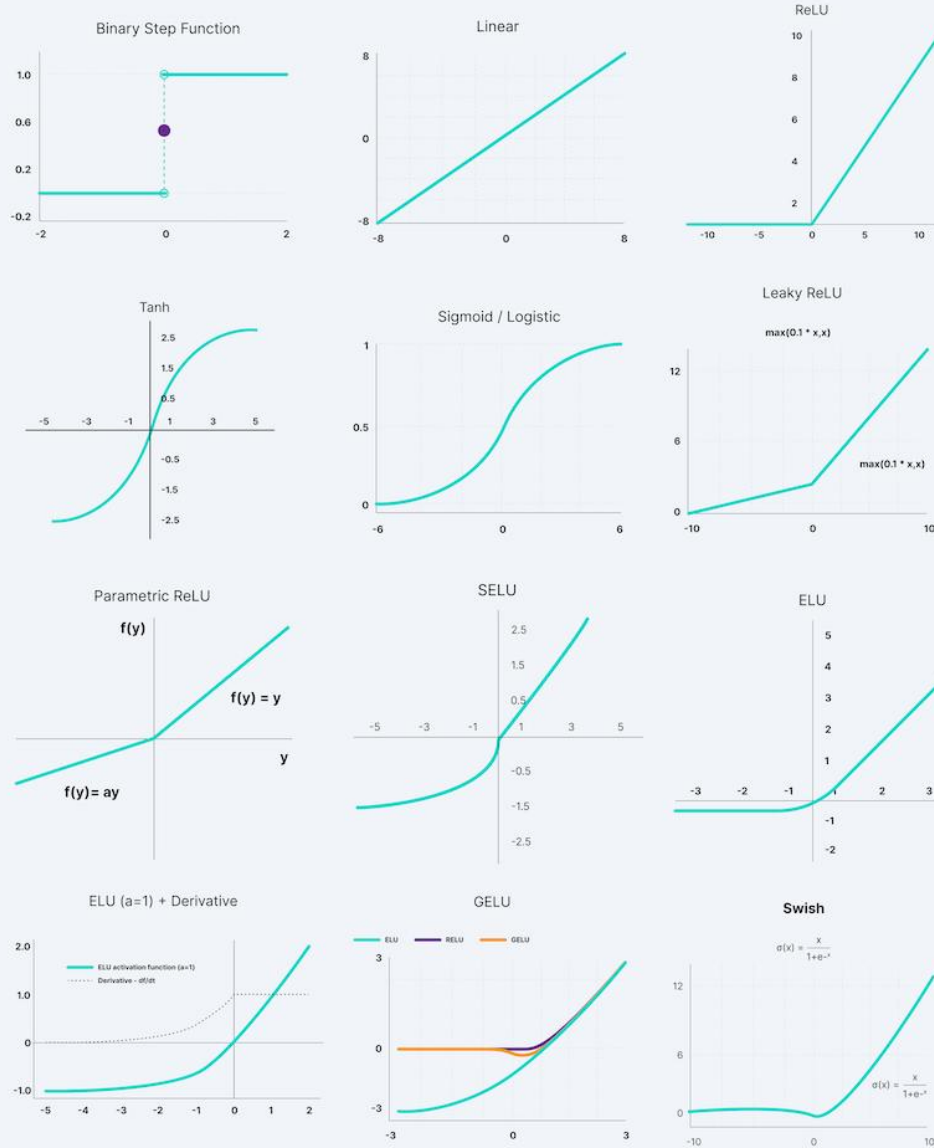
Leaky ReLU

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

Swish

$$f(x) = x * \text{sigmoid}(x)$$

Neural Network Activation Functions



Binary step

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

Sigmoid / Logistic

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

ELU

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

GELU

$$f(x) = xP(X \leq x) = x\Phi(x) = 0.5x \left(1 + \tanh \left[\sqrt{2/\pi} (x + 0.044715x^3) \right] \right)$$



Какво е bias (отклонение/пристрастие) в невронна мрежа?

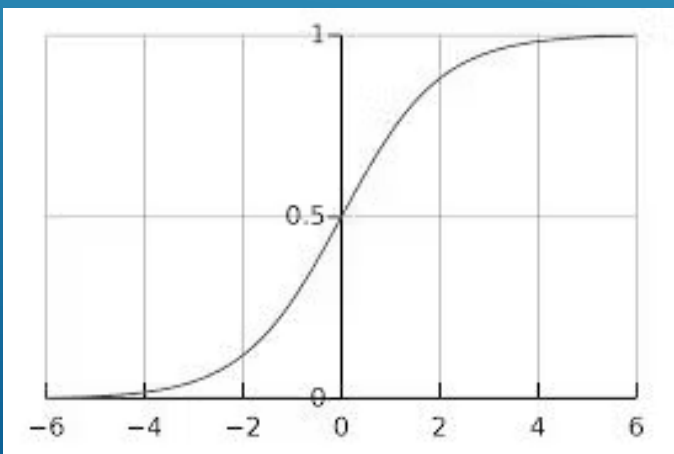
С прости думи, bias (отклонението) на невронната мрежа може да се дефинира като константата, която се добавя към произведението на характеристики и тегла.

Използва се за компенсирание на резултата.

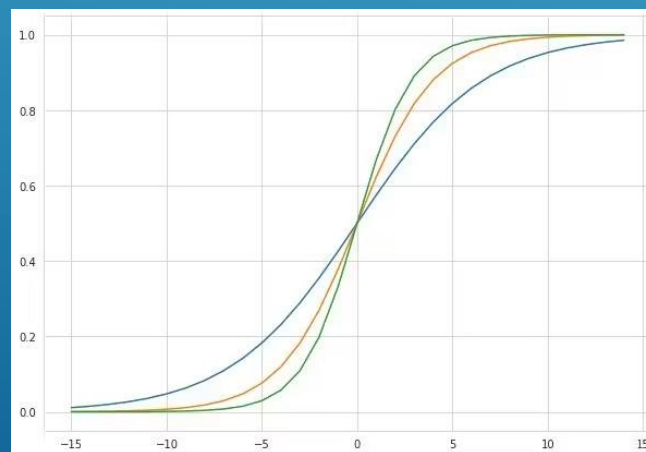
Помага на моделите да изместят функцията за активиране към положителната или отрицателната страна.

$$\text{sigmoid function} = \frac{1}{1 + e^{-(w*x+b)}}$$

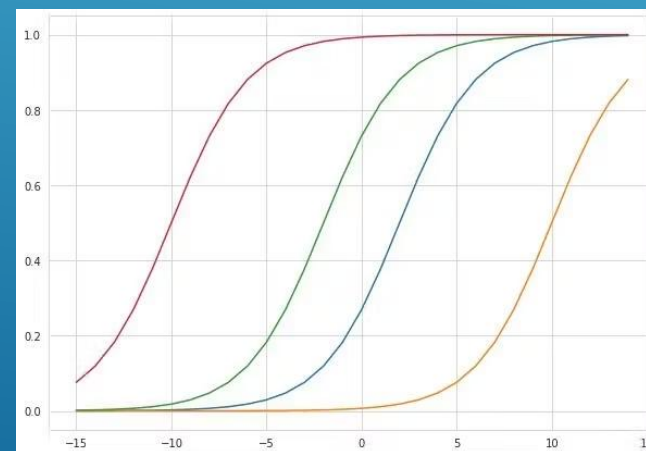
b = 0



b = 0, променяме w

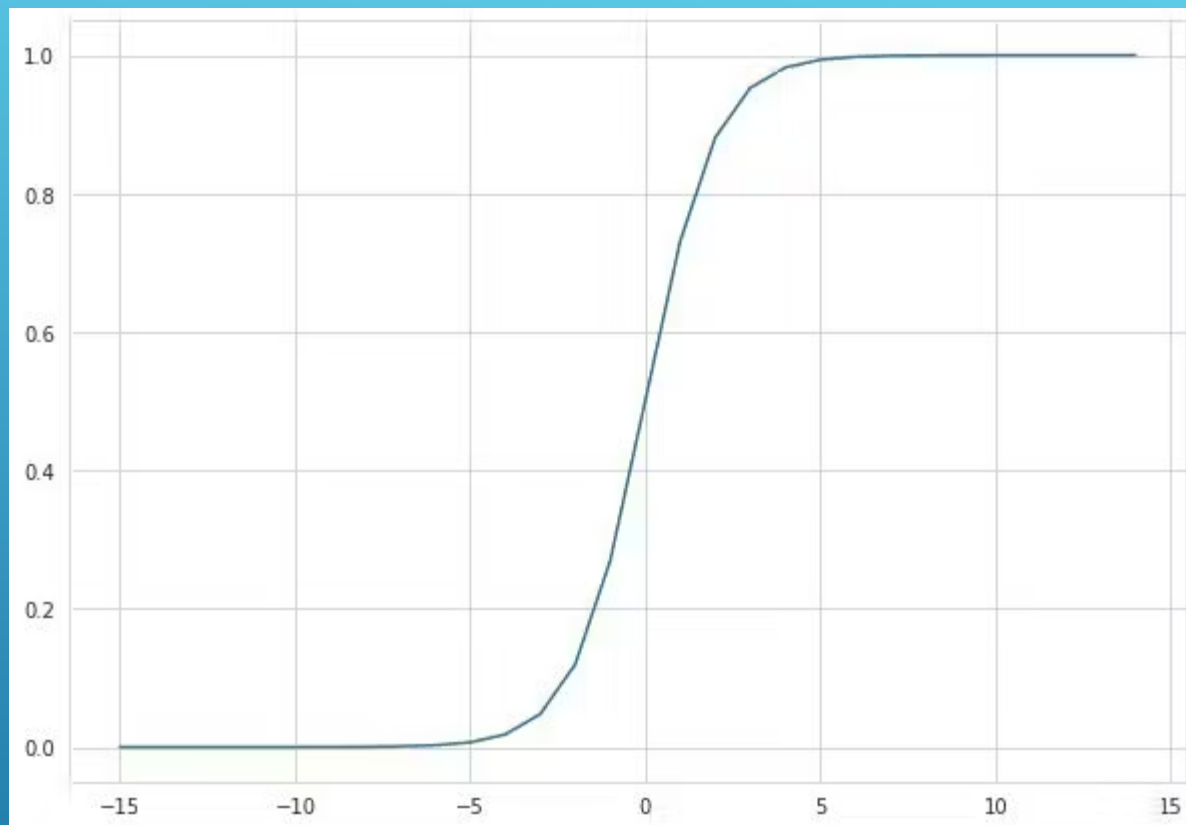


w е фиксирано, променяме b



Как да преместим кривата наляво или надясно

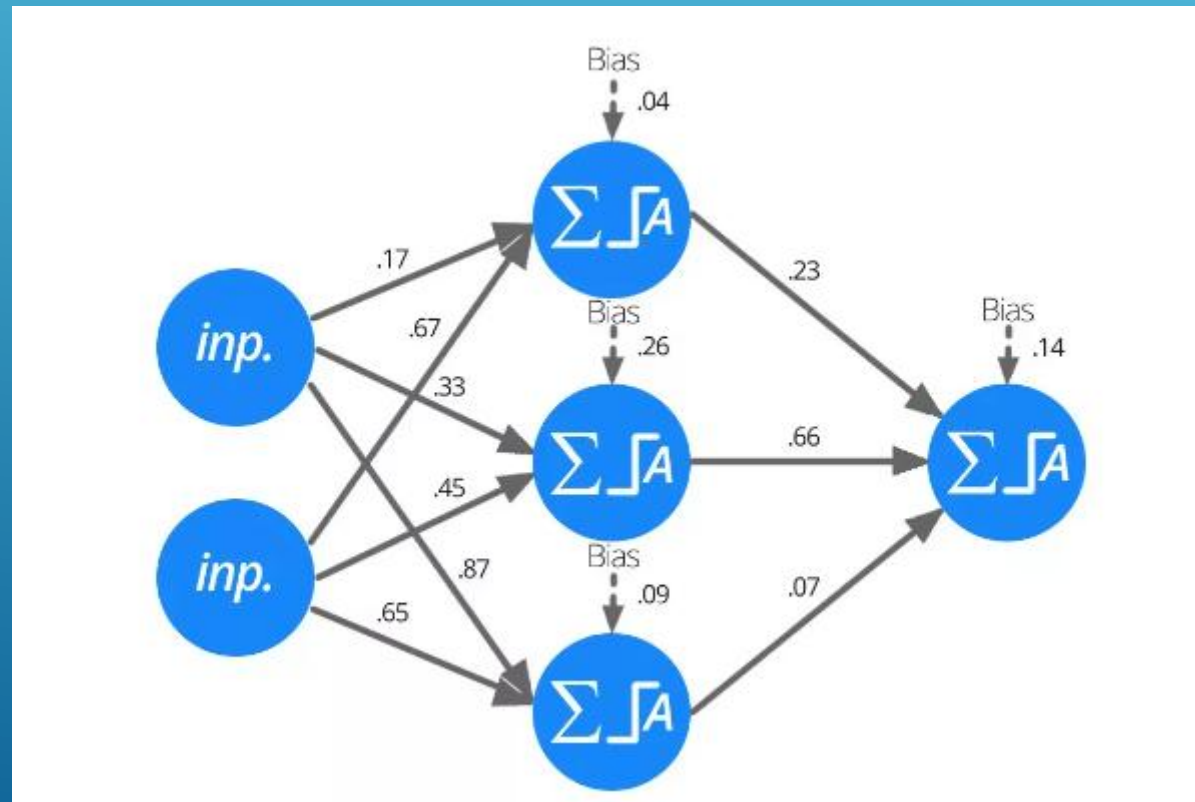
Ако заместим стойността на $w = -1$ и $b = 0$ в уравнението, графиката на функцията за активиране ще изглежда както е показано на графиката.

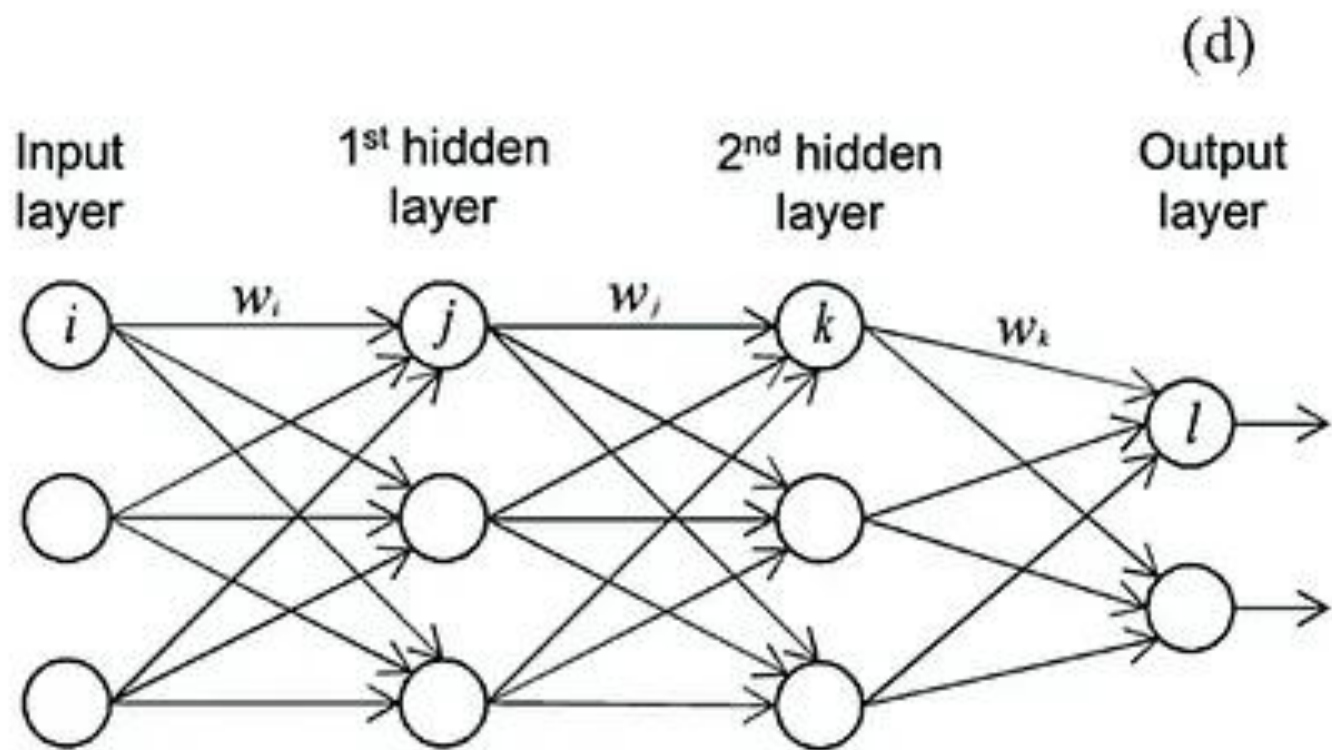
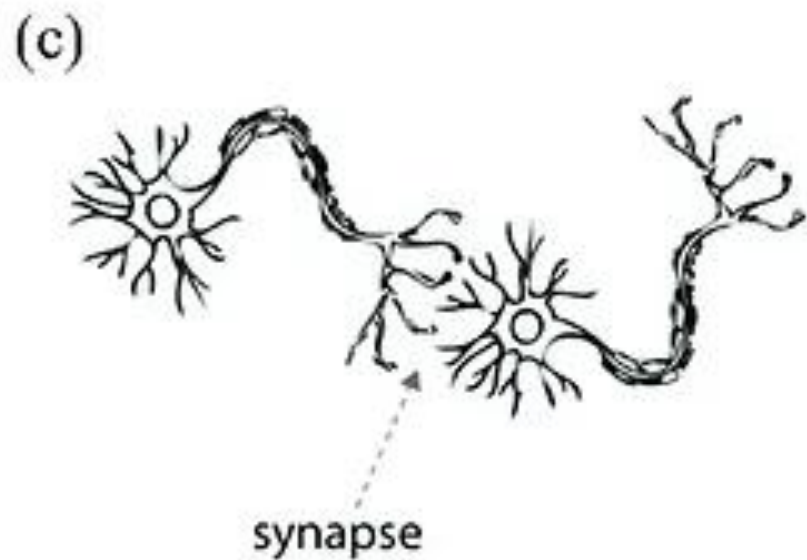
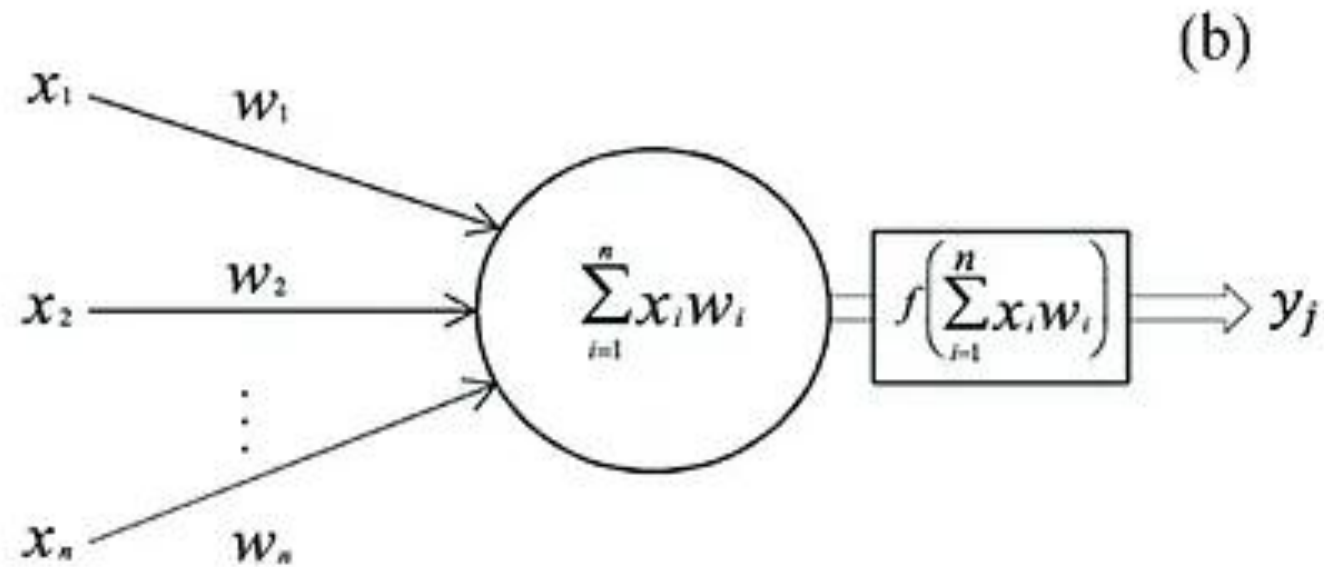
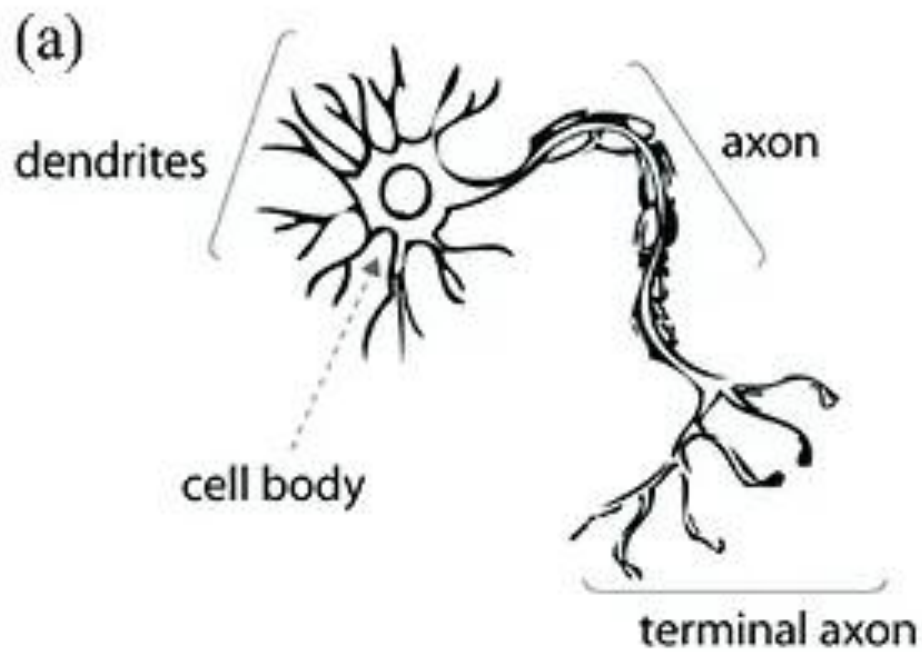


Може да се види, че изходът е равен на 0 за стойностите на „ x “ по-малки от 0 и равен на 1 за стойностите на „ x “ по-големи от 0.

Сега, как трябва да проектираме нашето уравнение, така че изходът на функцията за активиране да е равен на 1 за всички стойности на „ x “ по-малки от 5? Постигаме това, като въвеждаме термина отклонение „ b “ в нашето уравнение със стойност „ $b = -5$ “.

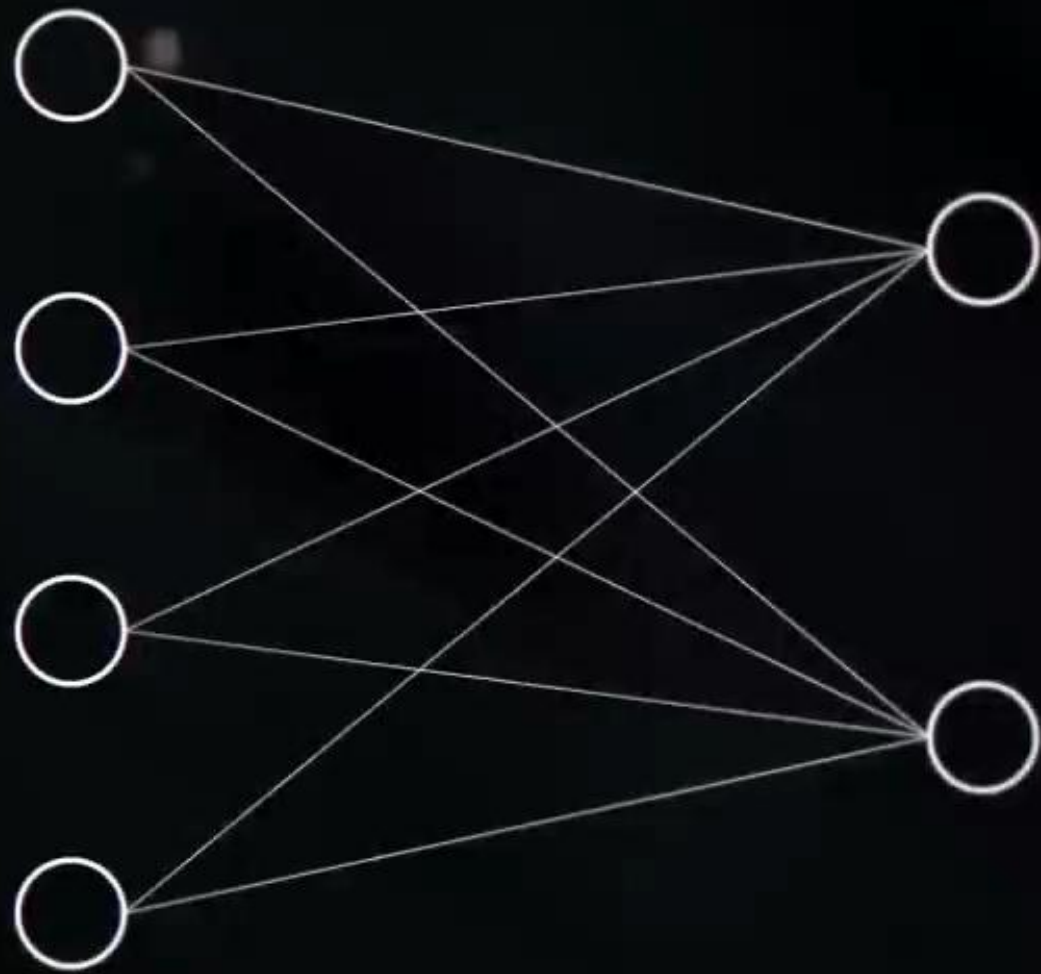
Вдъхновени от природата и базирани на принципа на работа на биологичните невронни структури, изкуствените невронни мрежи имат входно ниво, скрито ниво с произволен брой неврони (най-малко 1) за обработка на информацията и изходно ниво, между което съществуват няколко връзки.







1	0
0	1

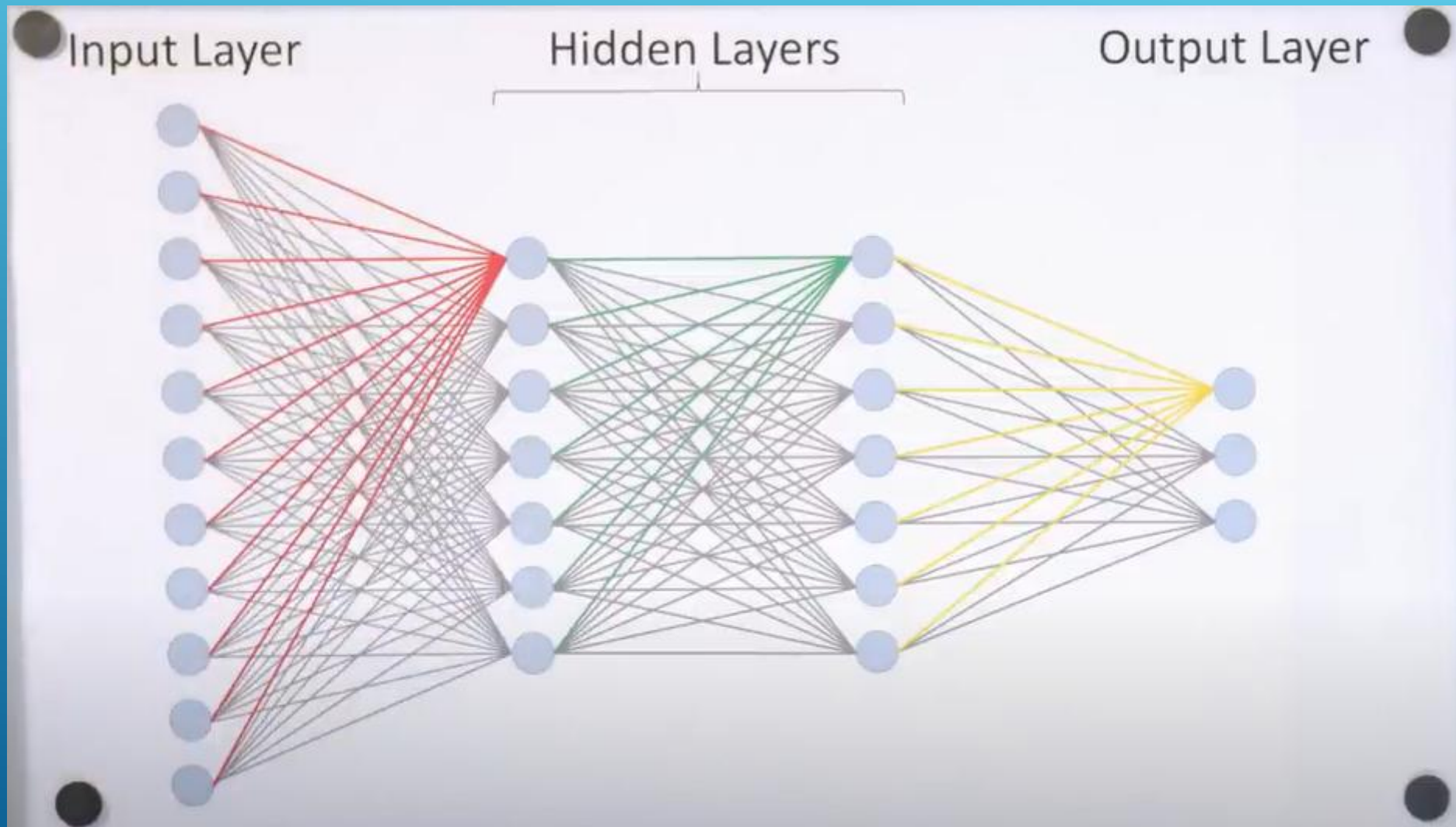


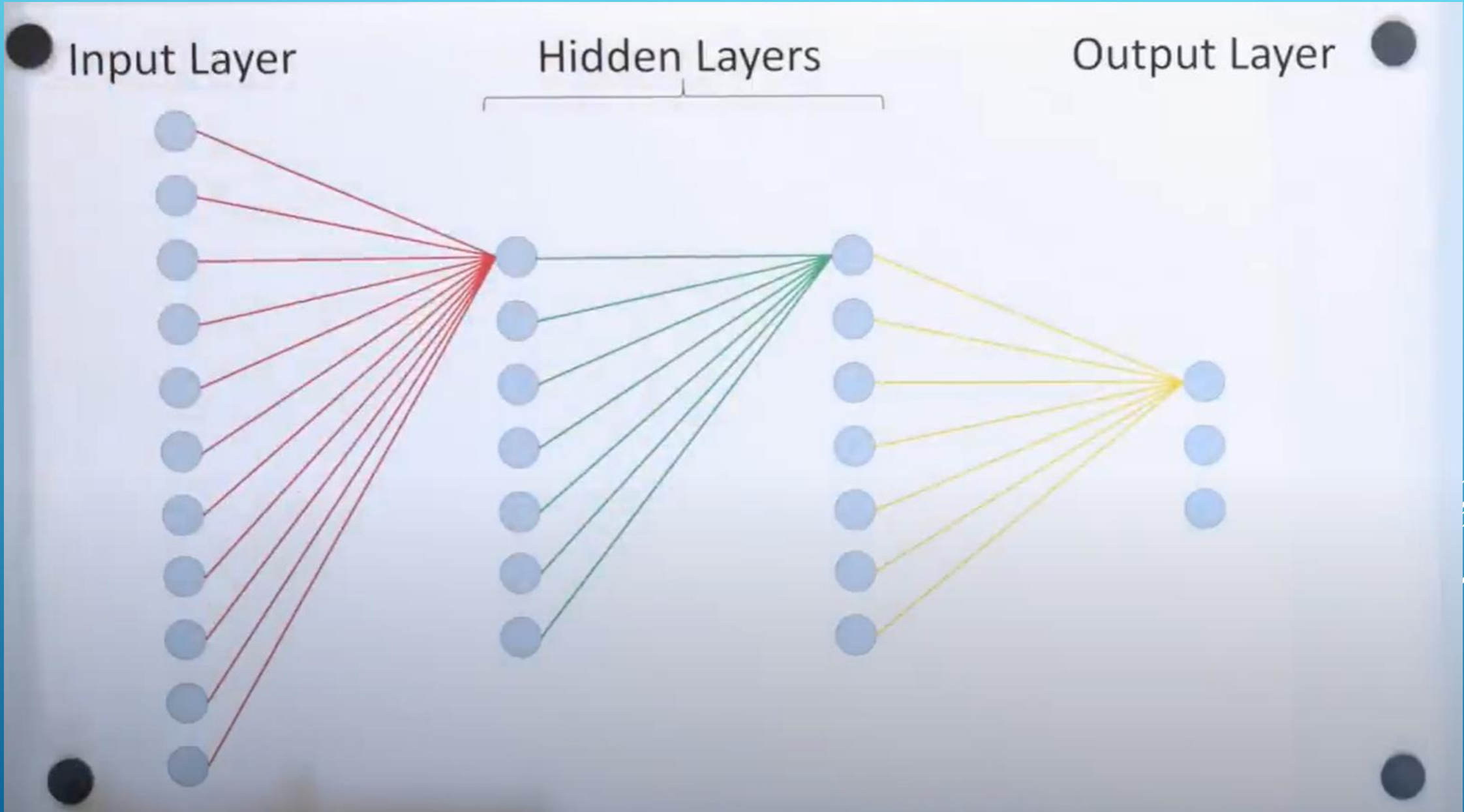
LLM Model Size

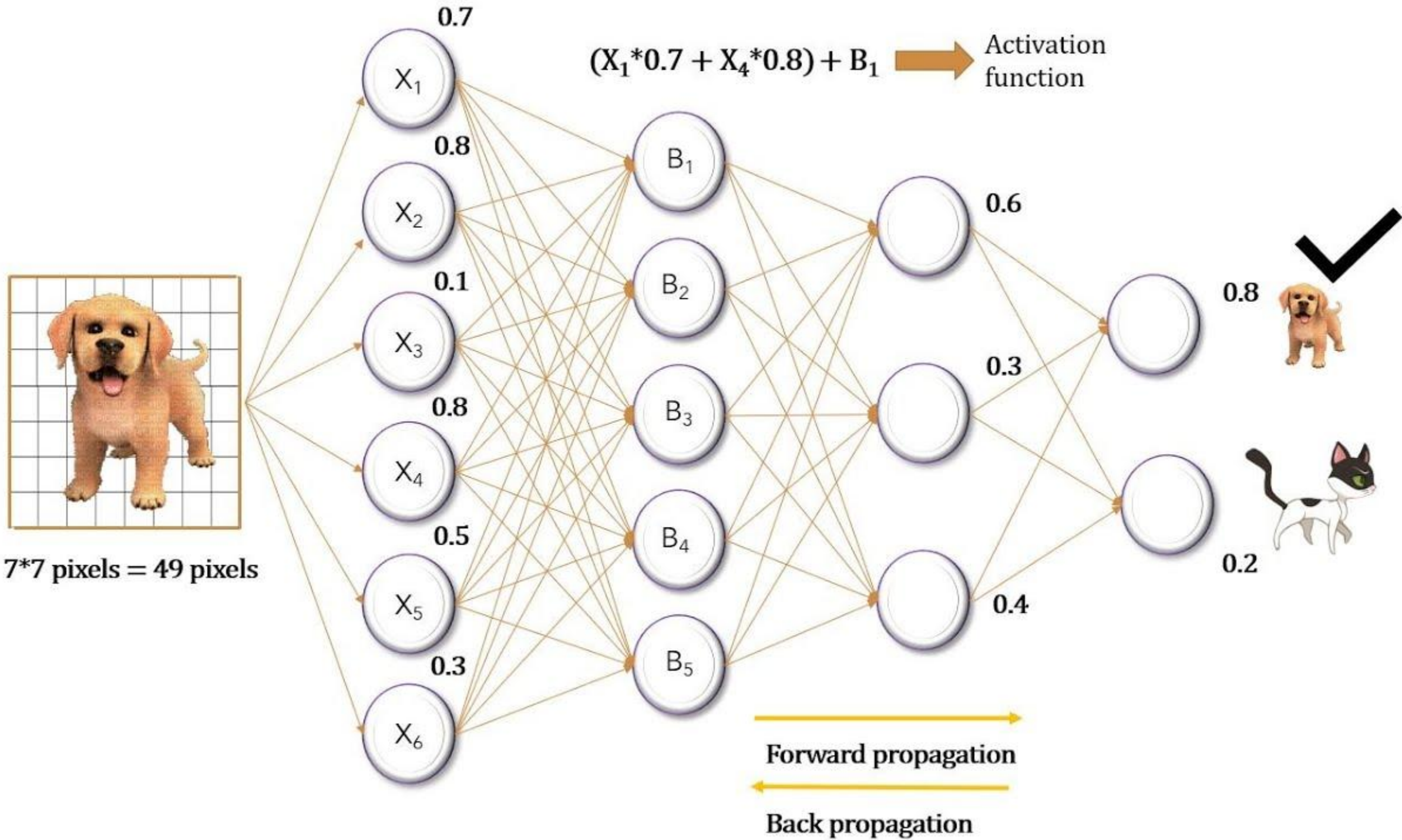
- GPT-4:** GPT-4 stands out with a massive parameter count of **1.5 Trillion**, making it one of the largest LLMs available. This size enables it to capture intricate language patterns effectively.
- BARD:** BARD boasts an impressive **1.6 Trillion** parameters, similar to GPT-4. This substantial size empowers it to generate precise and contextually relevant text.
- LLaMA:** LLaMA is no slouch, offering a substantial **1.2 Trillion** parameters, which allows it to generate coherent text across various topics.
- DeepSeek-V3,** a strong Mixture-of-Experts (MoE) language model with **671B total parameters** with 37B activated for each token.
- Flan-UL2:** Flan-UL2, although smaller in scale with **20 billion parameters**, still excels in producing natural and consistent text based on partial input.
- BLOOM:** BLOOM boasts **176 billion parameters**, which, while smaller compared to GPT-4 and BARD, enables it to discern detailed linguistic patterns and optimize text generation.

ИЗКУСТВЕНА НЕВРОННА МРЕЖА

Artificial Neural Network - ANN







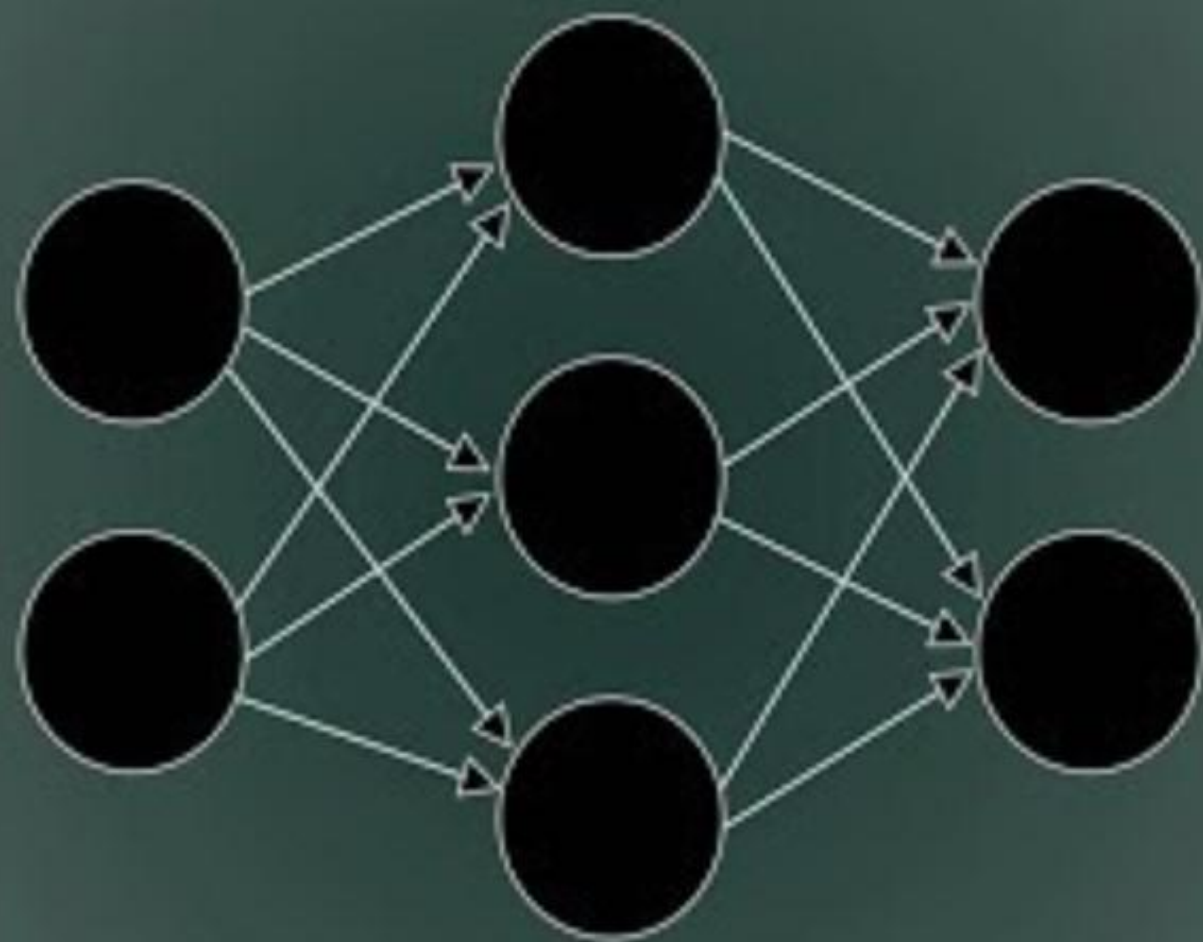


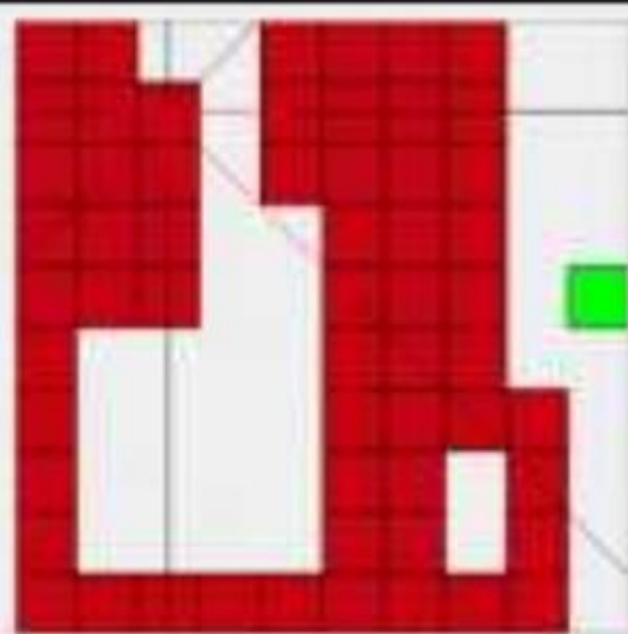
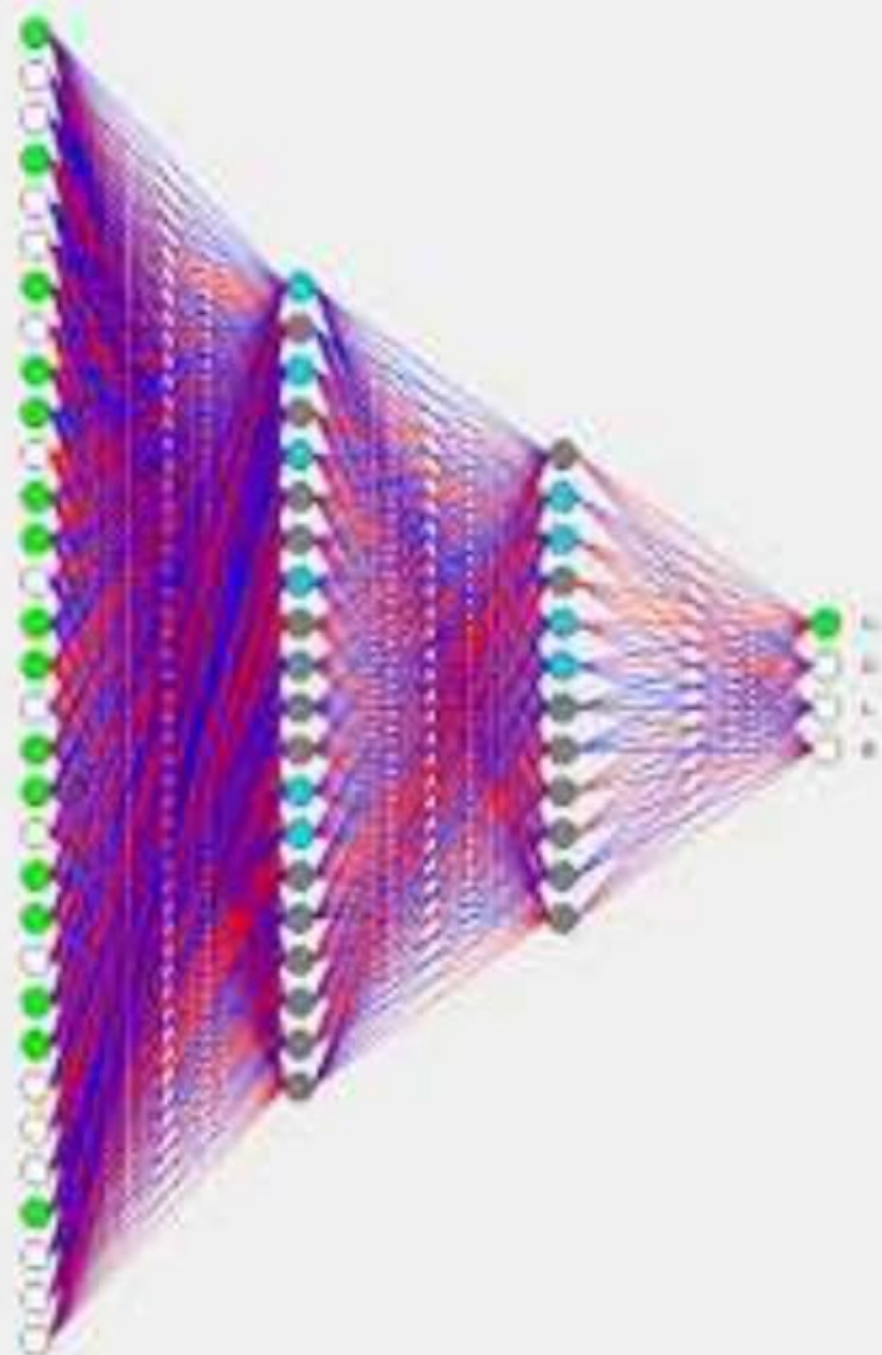
Neural Network

Explained
Simply



NEURAL NETWORKS EXPLAINED (EASY!)





Generation: 500
 Individual: 1/100
 Best Score: 10
 Best Fitness: 1.27E+11

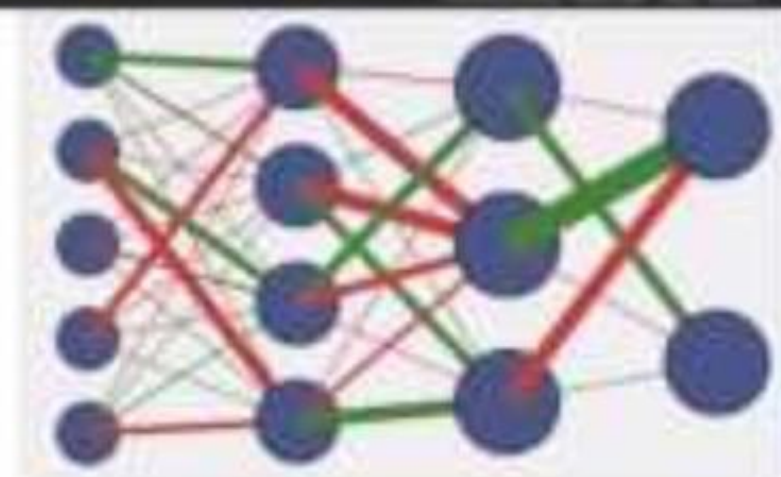
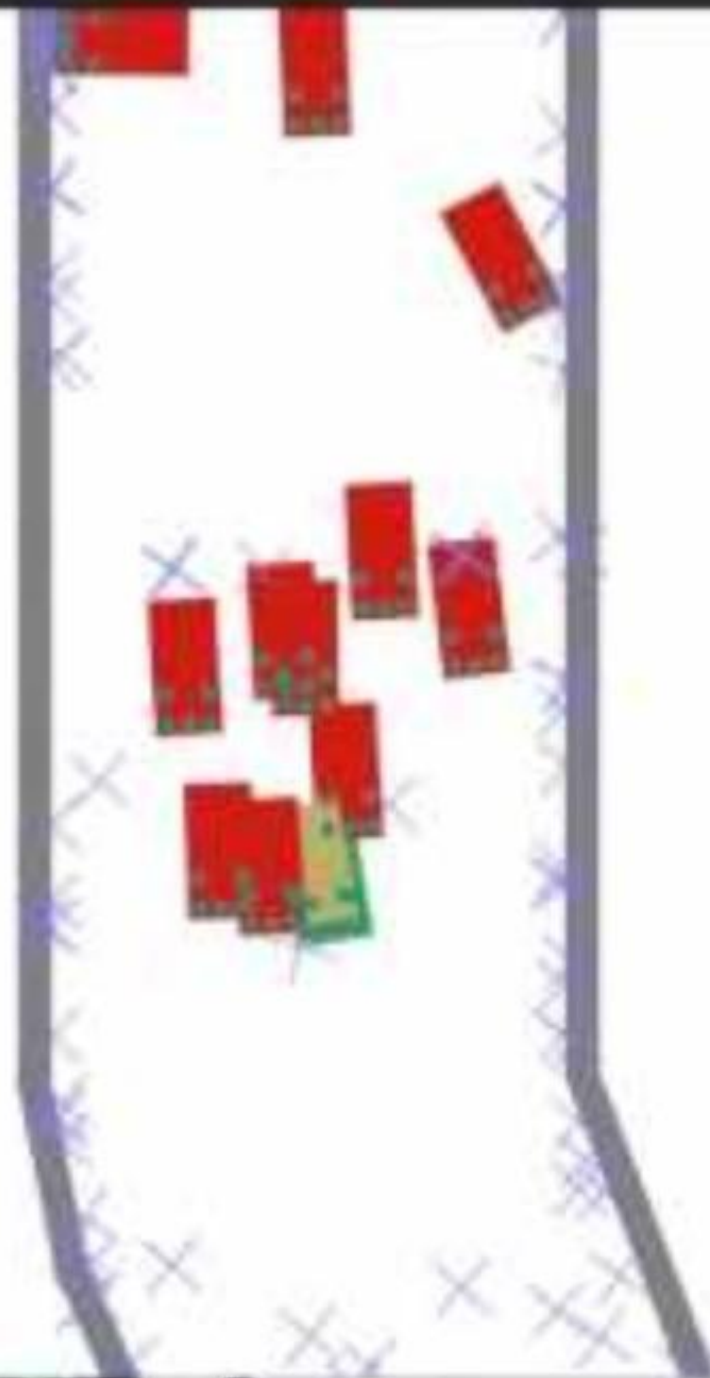
GA Settings

Selection Type: Rn
 Crossover Type: 50% Sbx
 50% Spxx
 Mutation Type: 100% Gaussian
 0% Uniform
 Mutation Rate: 1% + Static
 Lifespan: rbits

NN Settings

Hidden Activation: ReLU
 Output Activation: Sigmoid
 NN Architecture: (20, 12, 12, 4)
 Scale Values: 8 directions
 Apple/Self Values: default

Turn: 0.99975
Engine: 0.99999
Fitness: 0.04980



Generation: 15



~200 machine learning models active

