



# Глубокое обучение



## Лекция 1. Введение в глубокое обучение



# Знакомство с глубоким обучением



## ARTIFICIAL INTELLIGENCE

Techniques allowing computers to copy a human behavior



## MACHINE LEARNING

AI techniques allowing computers to learn to solve a specific task



## DEEP LEARNING

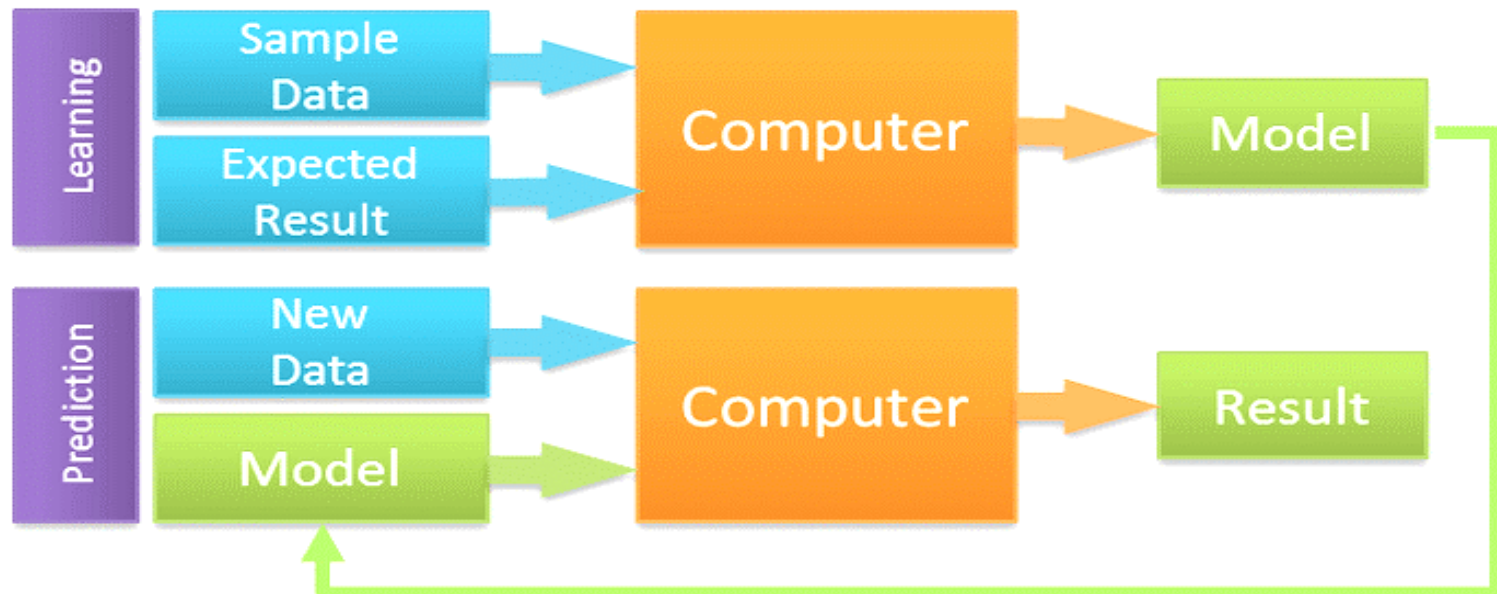
A subset of Machine Learning based on the use of neural networks



## Traditional modeling:

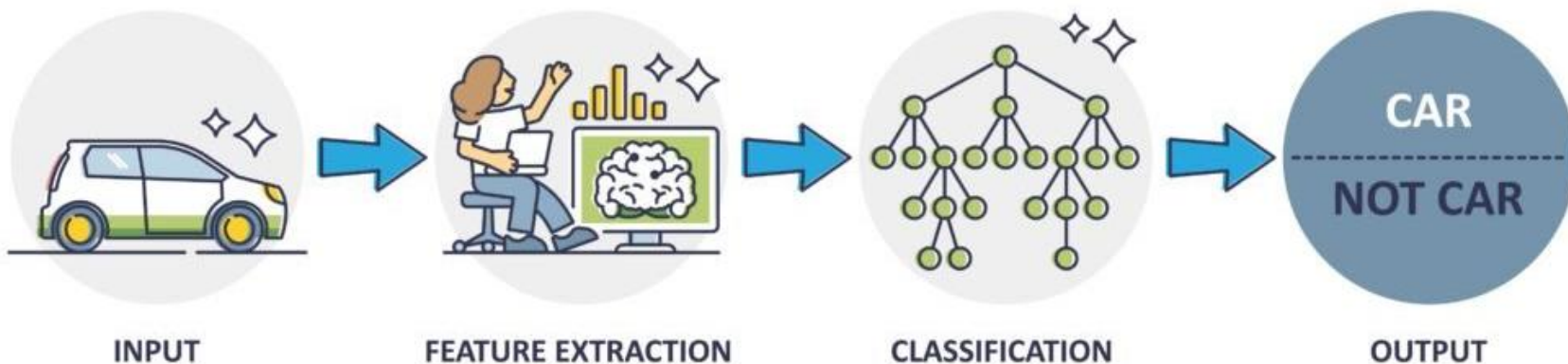


## Machine Learning:





## MACHINE LEARNING



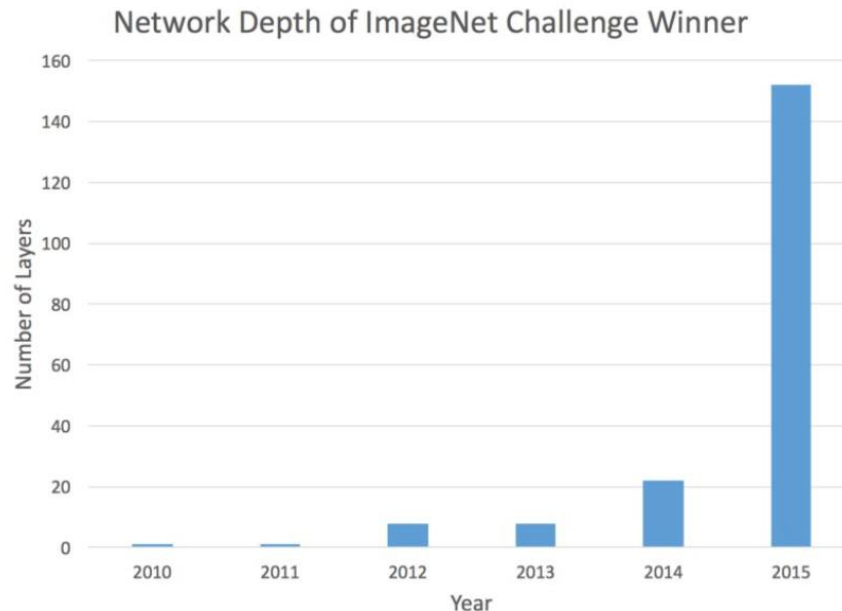
## DEEP LEARNING





Глубокое обучение — раздел машинного обучения, делающий упор на изучение последовательных слоев (или уровней) все более значимых представлений.

- Количество слоев, на которые делится модель данных, называют глубиной модели.
- Современное глубокое обучение часто вовлекает в процесс десятки и даже сотни последовательных слоев представления.
- Все они автоматически определяются под воздействием обучающих данных.
- Другие подходы к машинному обучению ориентированы на изучении одного-двух слоев представления данных, по этой причине их иногда называют поверхностным обучением.

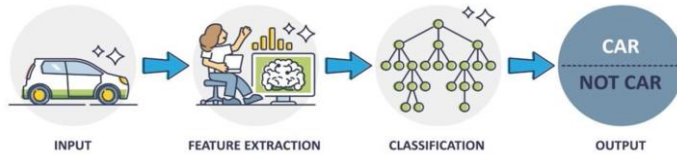






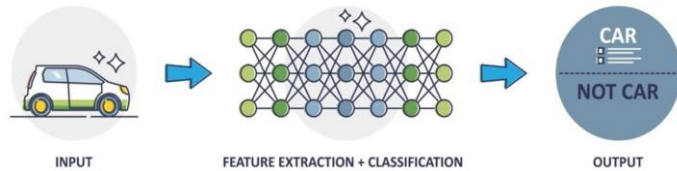
# КЛЮЧЕВОЕ ОТЛИЧИЕ ГЛУБОКОГО ОБУЧЕНИЯ ОТ МАШИННОГО ОБУЧЕНИЯ 7

ML



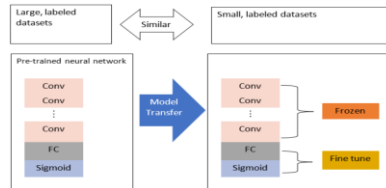
- + Небольшой датасет
- /+ Человеческая экспертиза
- Низкая выразительность модели

DL



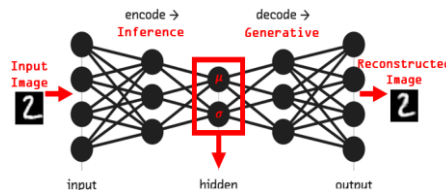
- + Высокая выразительность модели
- + Без эксперта
- Большой датасет

DL: Pretrained models



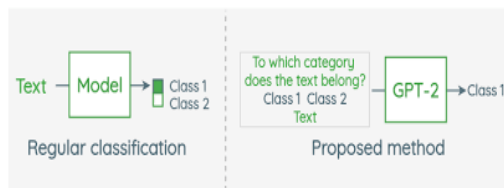
- + Высокая выразительность модели
- + Без эксперта
- + Небольшой датасет при дообучении

DL: Pretrained, unlabeled dataset



- + Очень высокая выразительность модели
- + Очень большой размеченный датасет
- Очень большие затраты на обучение

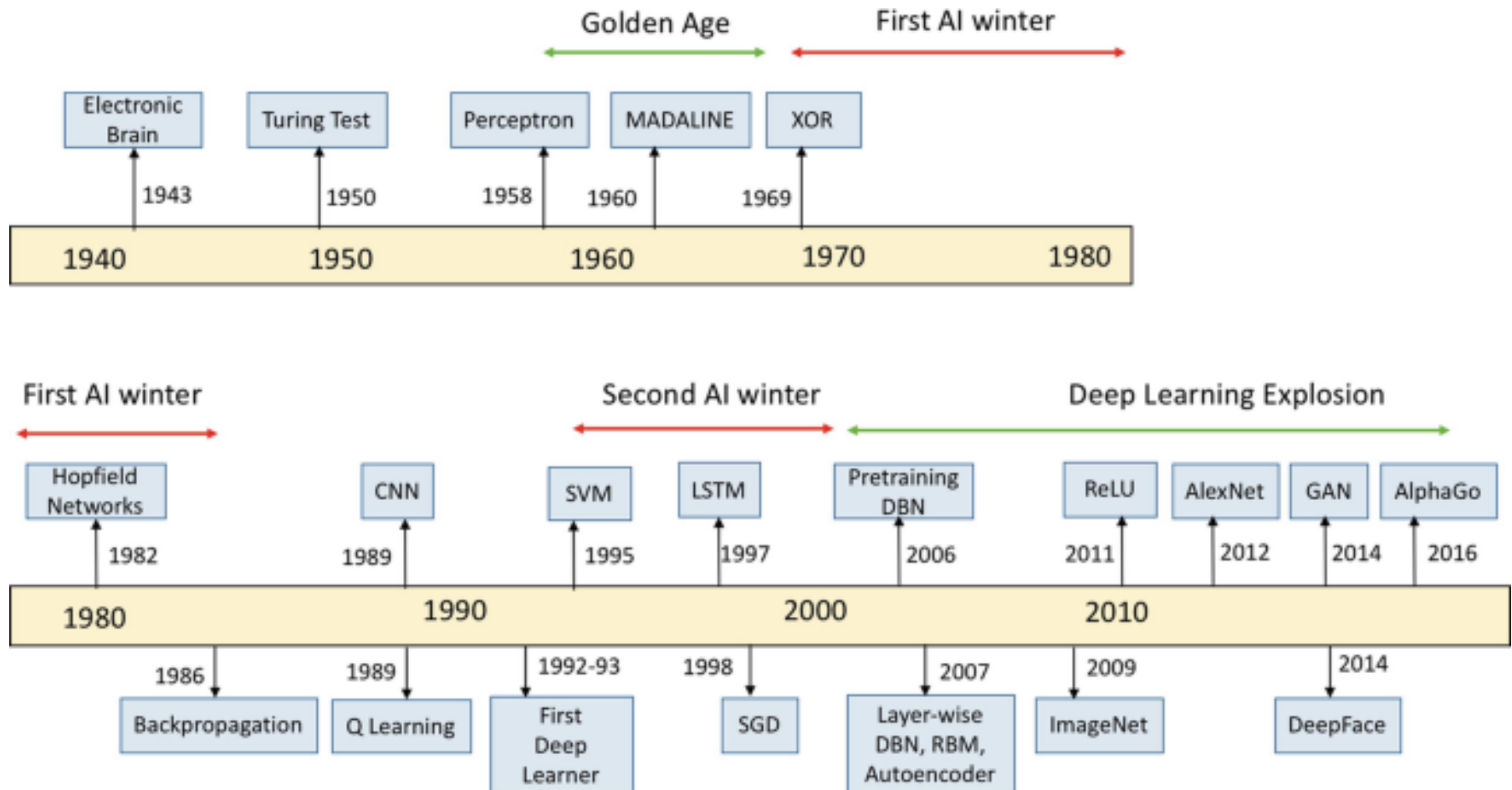
GPT: few-shot learning



- + Тривиальное обучение во время инференса
- + Крайне высокая выразительность модели
- Огромные затраты на первичное обучение
- Дорогой инференс



Развитие ИИС и глубокого обучения включает две «зимы искусственного интеллекта» во время которых интерес к развитию технологий ИИС существенно охладел.

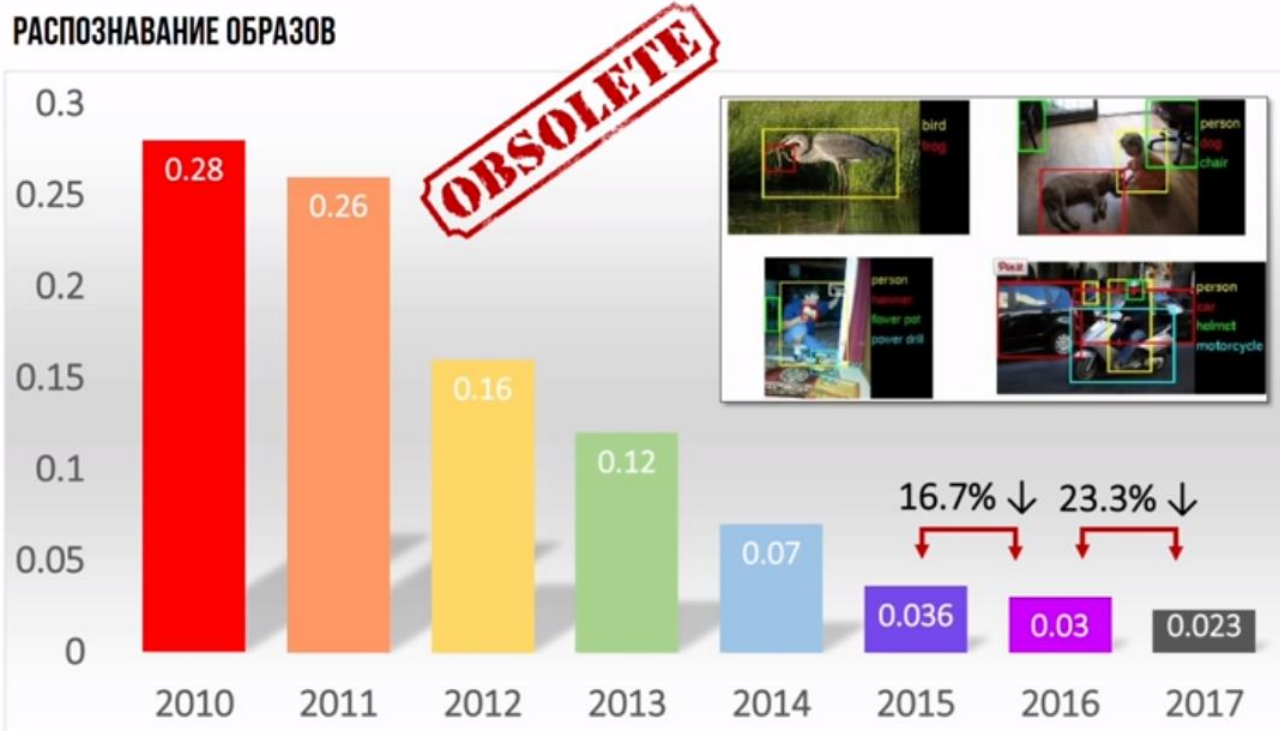




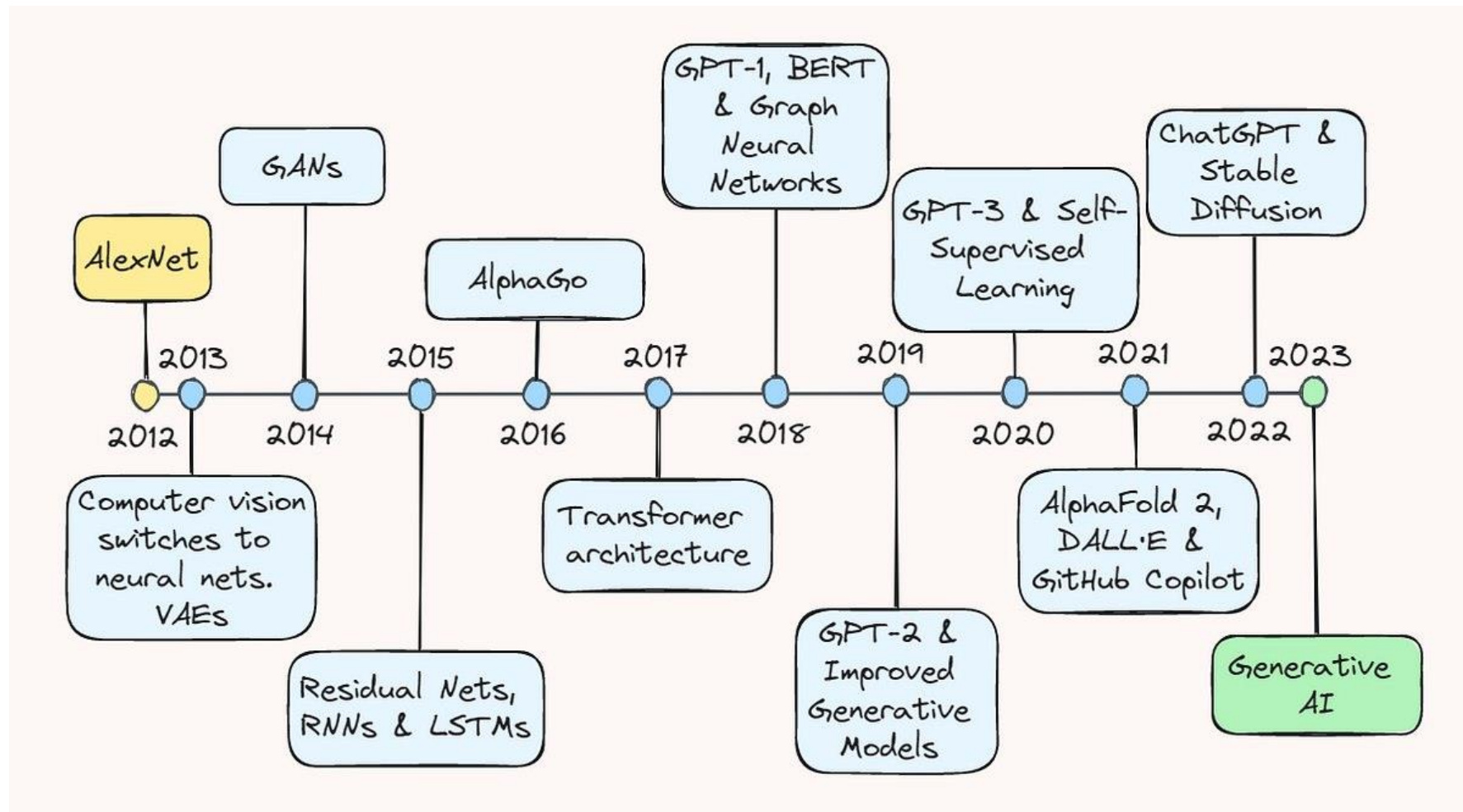


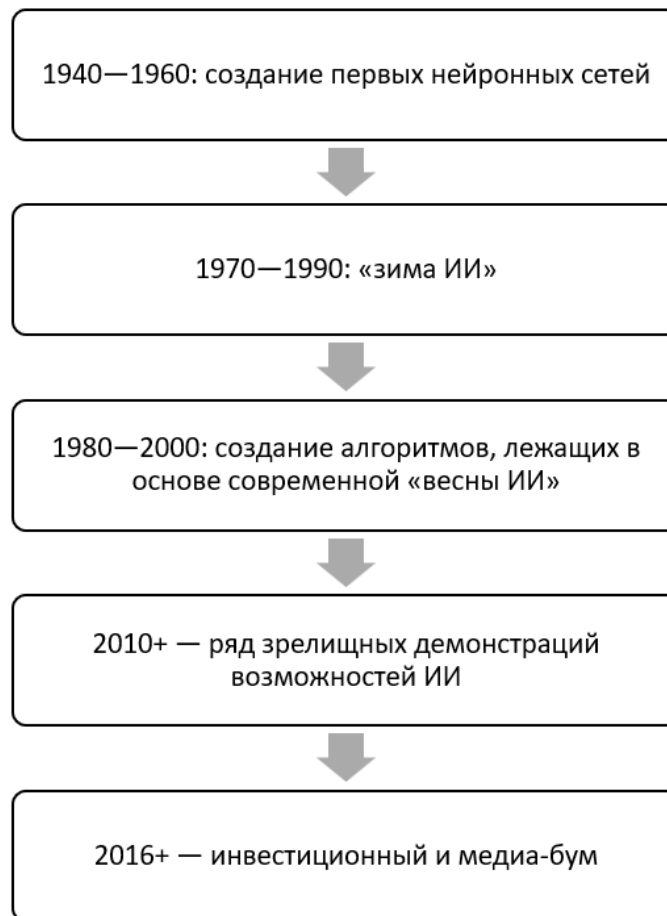
Соревнование **ImageNet** - крупномасштабное распознавание образов. Классификации цветных изображений с высоким разрешением на 1000 разных категорий после обучения по выборке, включающей в себя **1,4 миллиона изображений**.

- В 2011 году модель-победитель, основанная на классических подходах к распознаванию образов, показала точность лишь 74,3% .
- В 2012 году команда Алекса Крижевски (Alex Krizhevsky), советником в которой был Джеффри Хинтон (Geoffrey Hinton), достигла точности в 83,6% — значительный прорыв методов глубокого обучения.



Результаты ежегодных соревнований по распознаванию изображений ImageNet (ILSVRC, классификация)

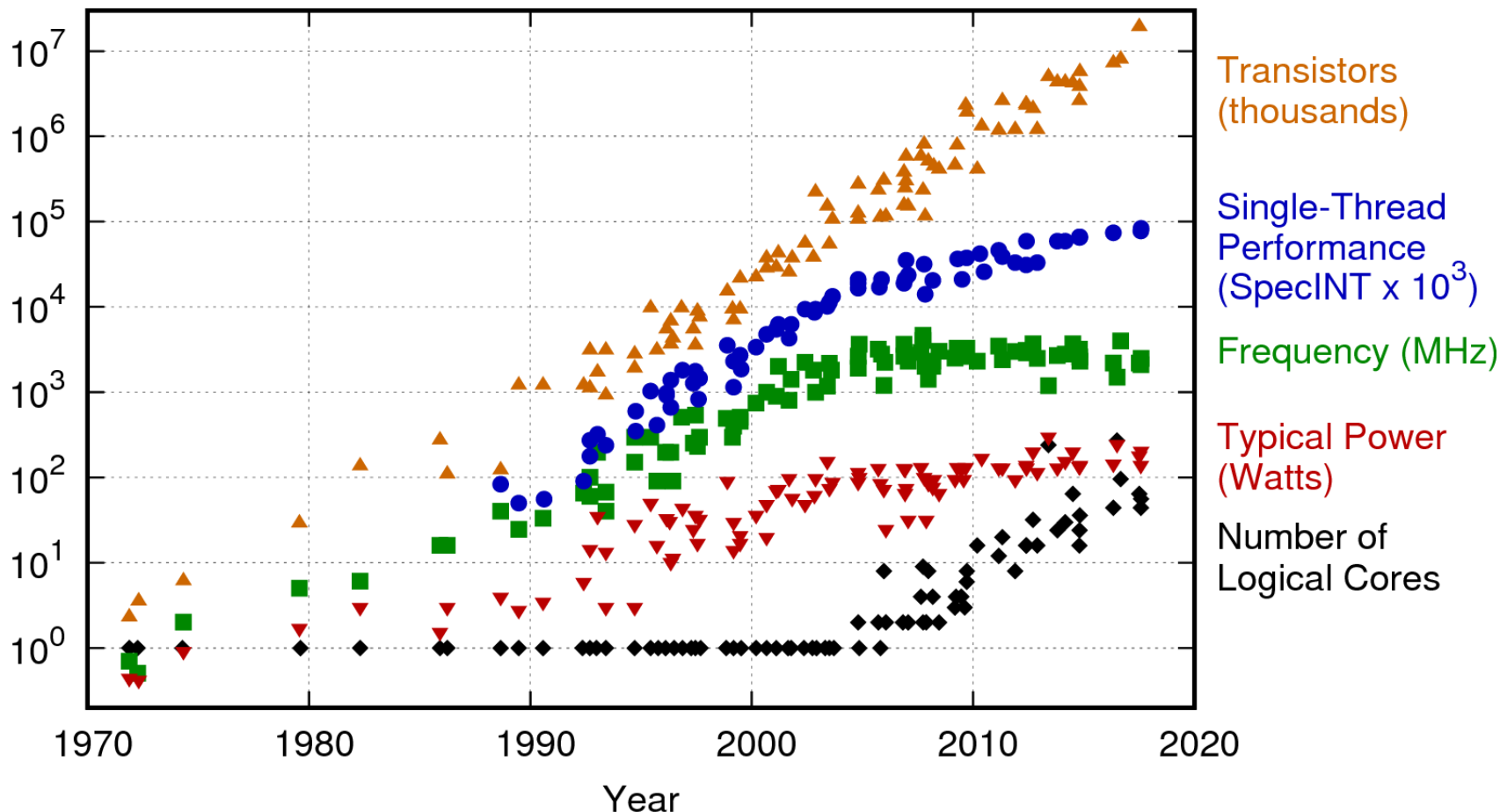






- Между 1990 и 2010 годами быстродействие стандартных процессоров выросло примерно в 5000 раз (закон Мура).

42 Years of Microprocessor Trend Data

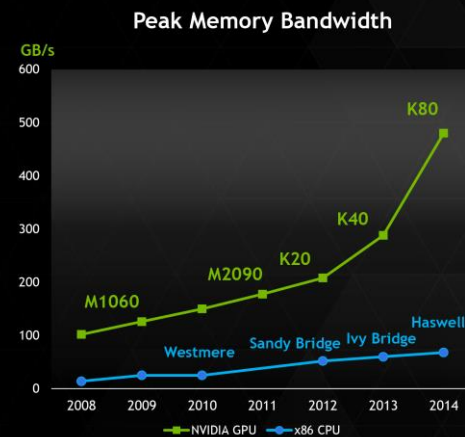
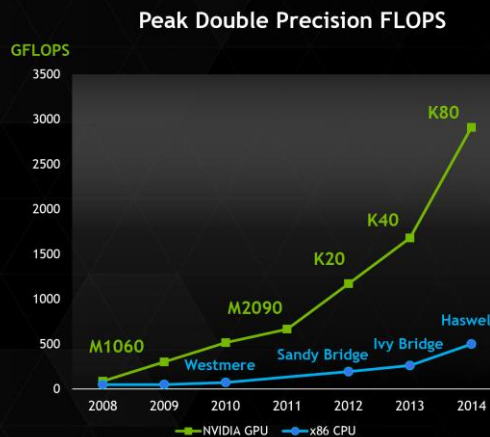


Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten  
New plot and data collected for 2010-2017 by K. Rupp

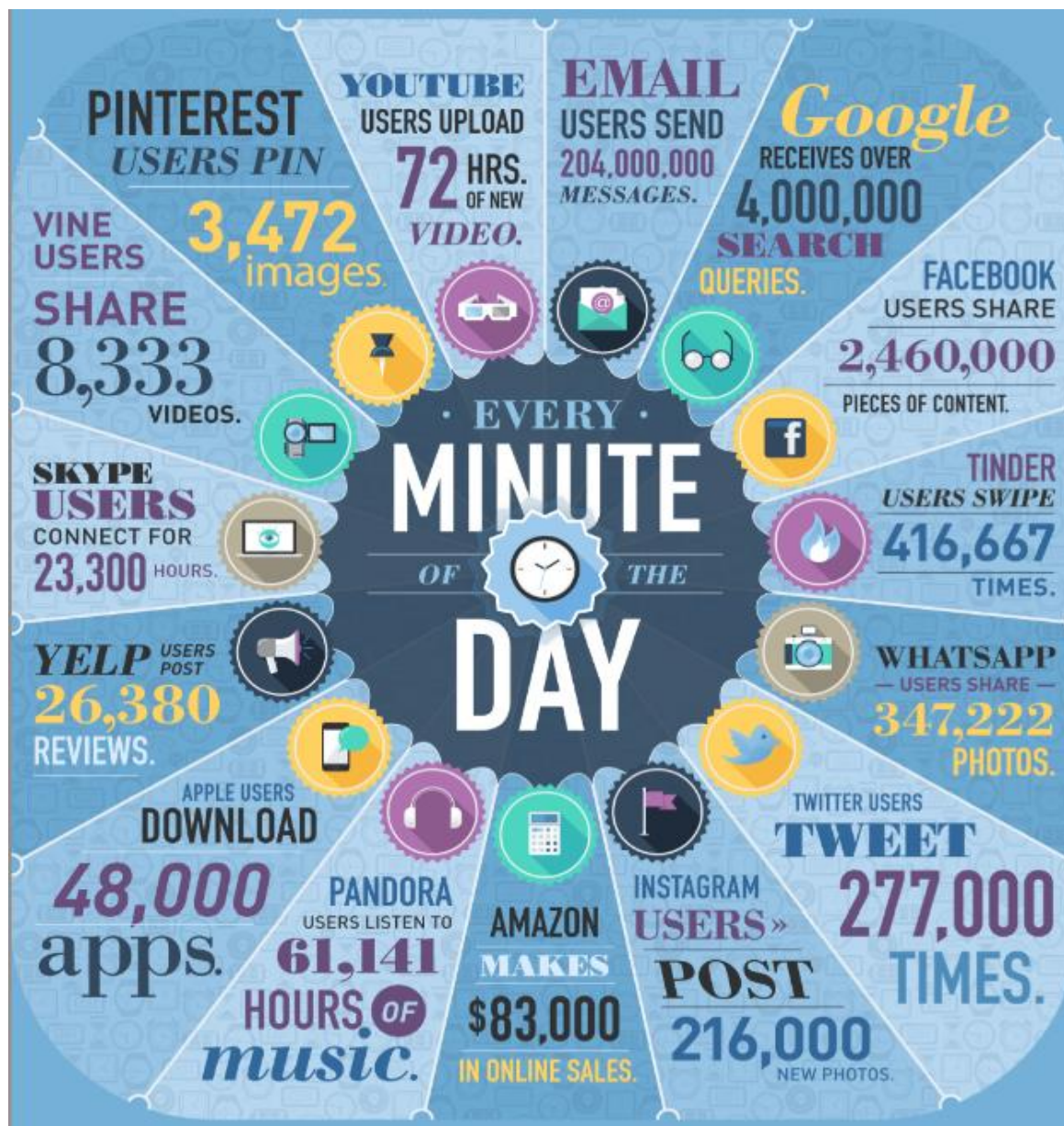
- Глубокие нейронные сети допускают высокую степень распараллеливания т.к. выполняют в основном умножение множества маленьких матриц.
- Компании NVIDIA и AMD, вложили миллионы долларов в разработку быстрых процессоров с массовым параллелизмом Graphical Processing Unit (GPU).
- В 2007 году компания NVIDIA выпустила CUDA (Compute Unified Device Architecture) - интерфейс позволяющий использовать GPU для вычислений общего назначения.
- Ведется разработка специализированных процессоров (TPU) для решения задач DL.



## Performance Lead Continues to Grow









## Персептрон, введение в ИНС



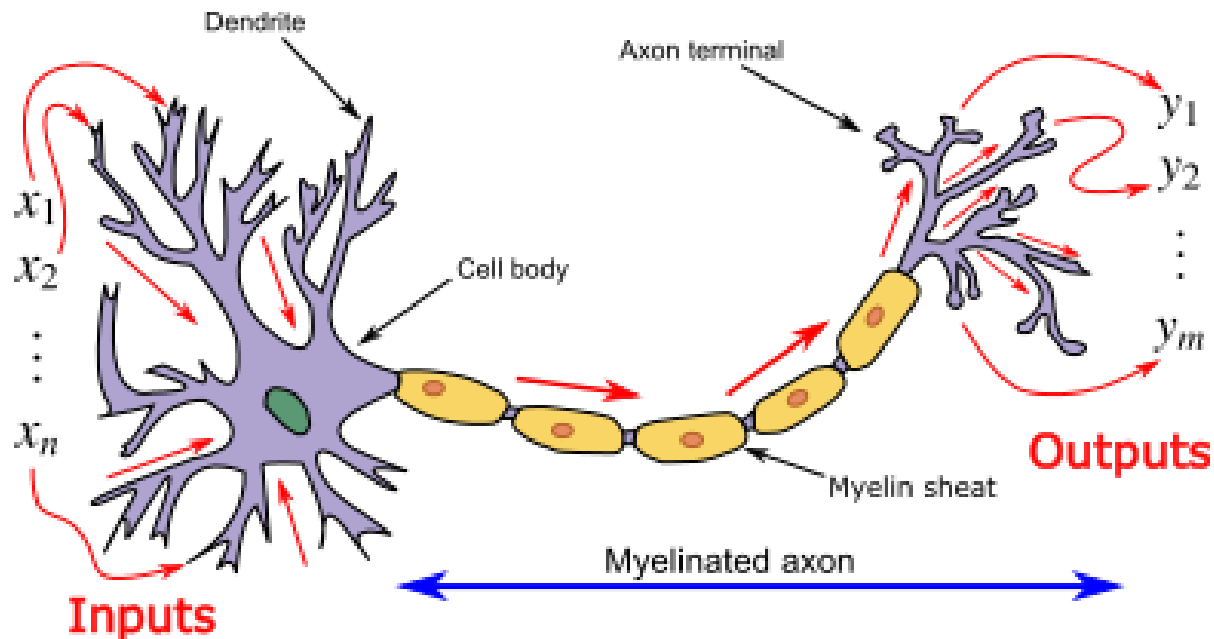




Искусственная нейронная сеть (ИНС) – модель разработанная под влиянием изучения организации и функционирования биологических нейронных сетей.

ИНС представляет собой сеть взаимодействующих между собой простых вычислительных элементов (искусственных нейронов), которые, в общем случае, умеют:

- реагировать на входной сигнал, возвращая реакцию на него
- хранить параметры, обеспечивающие вариативность реакции на входной сигнал
- обучаться - менять свои параметры в ходе выполнения операции обучения



Модели нейронов, используемые в ИНС, в большинстве аспектов очень сильно отличаются от нейронов и их принципов работы в живых организмах

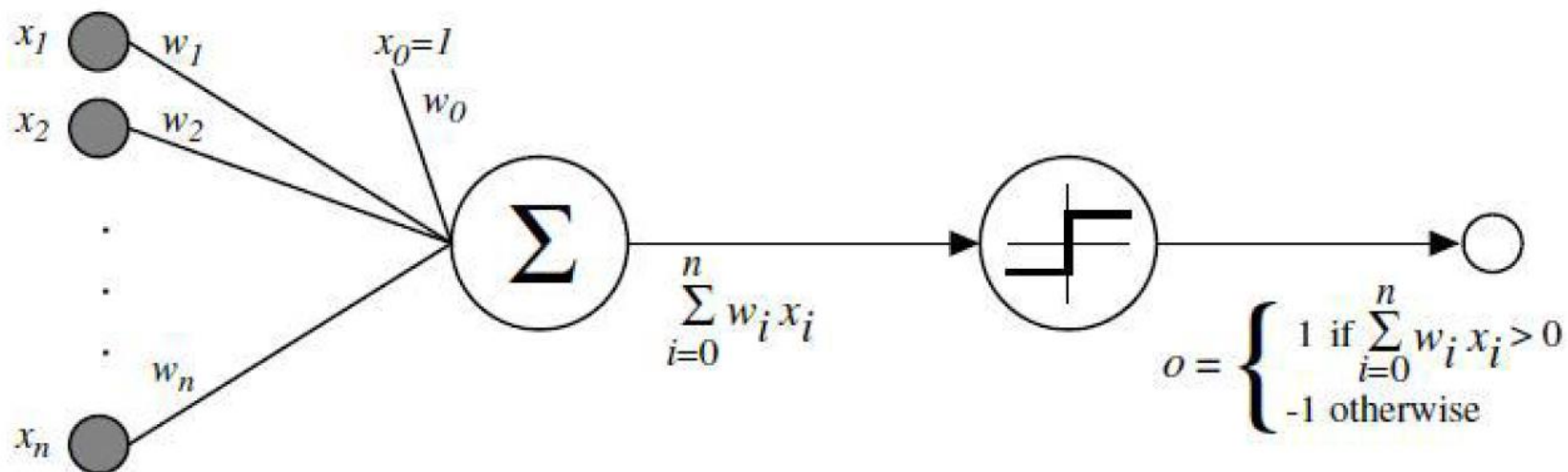


- Модели нейронов, используемые в ИНС, в большинстве аспектов очень сильно отличаются от нейронов и их принципов работы в живых организмах
- Существует совершенно отдельное направление исследований, нацеленное на моделирование работы нейронов в живых организмах (neuroscience)



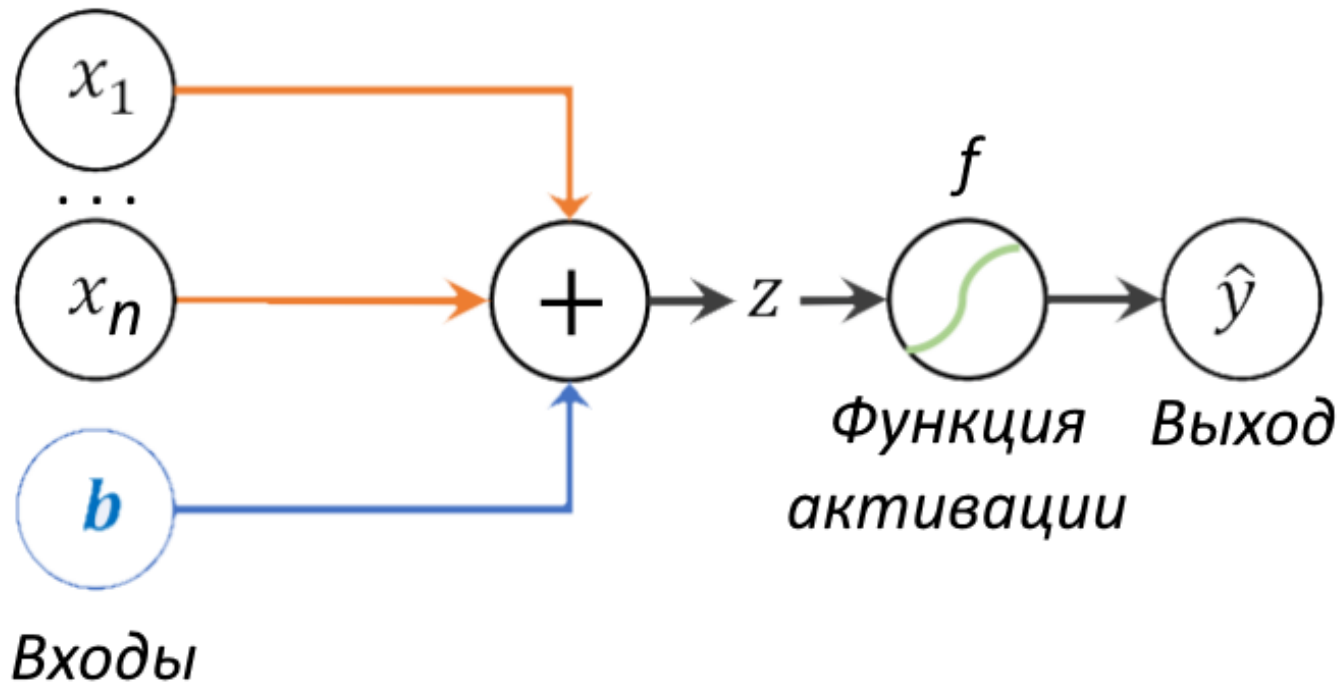


В 1958 г. Фрэнк Розенблатт изобретает однослойный перцептрон и демонстрирует его способность решать задачи классификации.



Основной инновацией Розенблатта была разработка алгоритма обучения перцептрона:

- изначально веса инициализируются случайным образом
- поочередно берется один обучающий пример, включающий набор входов  $x_i$  и верное значение  $y$
- для ошибочных предсказаний  $\hat{y}$ :
  - веса увеличиваются, если  $\hat{y}=0$ , а  $y=1$
  - веса уменьшаются, если  $\hat{y}=1$ , а  $y=0$
- процедура повторяется до исчезновения ошибок.



Общий вид перцептрона:

$$\hat{y} = f(w_0 + w_1 x_1 + \dots + w_n x_n) = f(\mathbf{w}^T \mathbf{x})$$

, где  $\mathbf{x} = (1, x_1, \dots, x_n)$

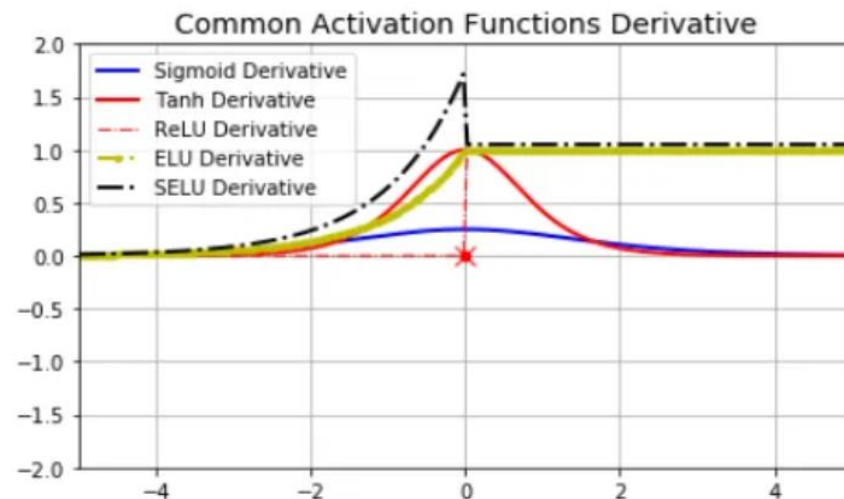
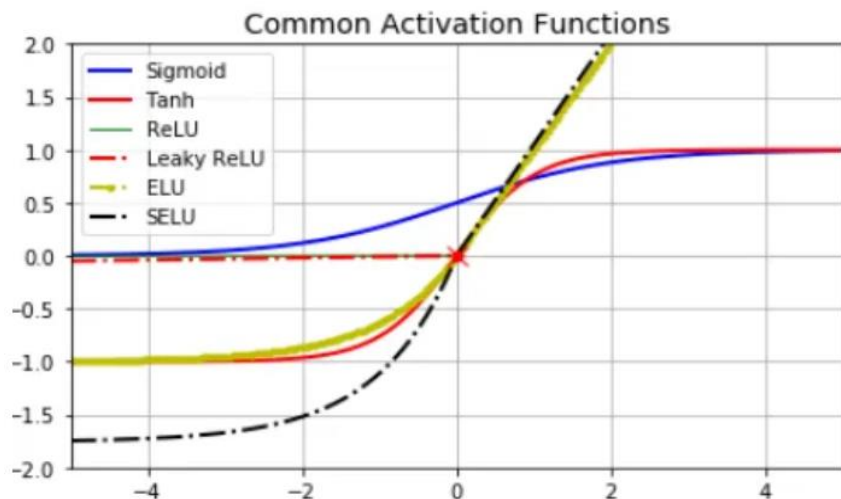
В качестве функции активации  $f$  могут использоваться функции:

- Сигмоида (логистическая функция):  $\sigma(z) = \frac{e^z}{1+e^z}$
- Гиперболический тангенс:  $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$



Требования к функциям активации:

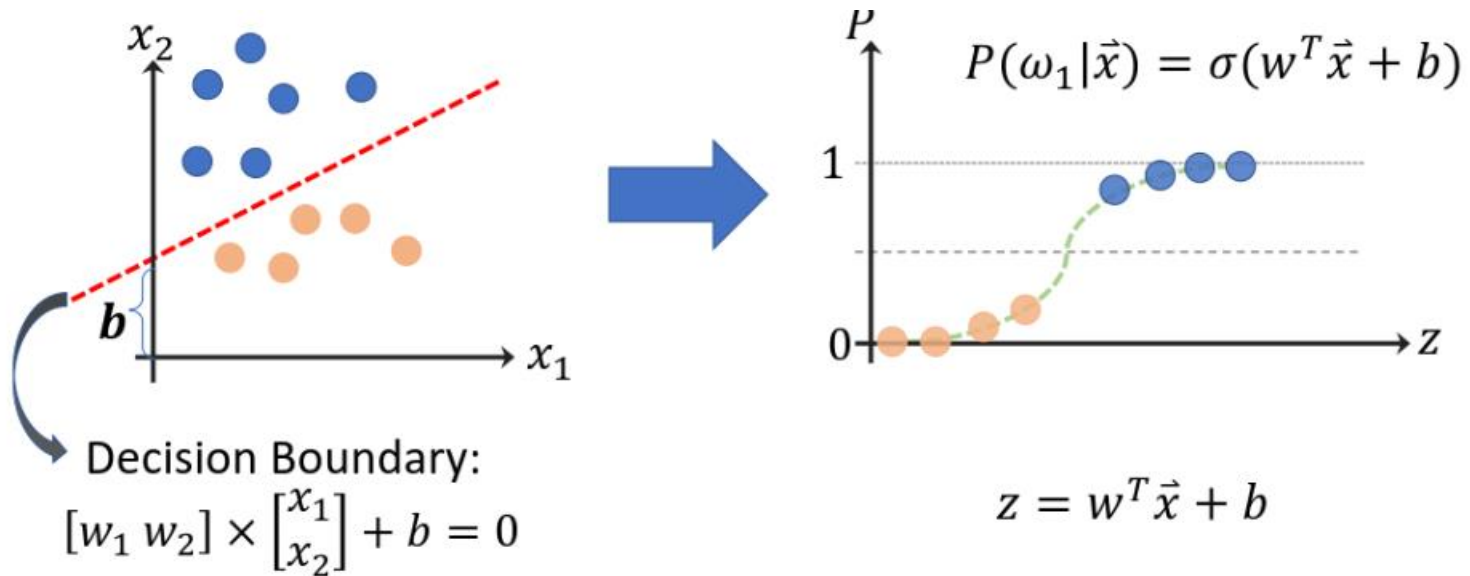
- функция должна быть монотонной (обычно монотонно не убывающая)
- иметь первую производную почти всюду (необходимо для обучения нейронной сети)



- Сигмоида (логистическая функция):  $\sigma(z) = \frac{e^z}{1+e^z}$
- Гиперболический тангенс:  $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$
- Единичная ступенчатая функция (функция Хевисайда):  $step(z) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases}$
- Rectified linear unit (вентиль):  $ReLU(z) = \max(0, z)$
- Leaky Rectified Linear Unit (leaky ReLU):  $ReLU(z) = \begin{cases} z, & \text{if } z > 0 \\ 0.01z, & \text{if } otherwise \end{cases}$
- Exponential Linear Unit (ELU):  $ELU(z) = \begin{cases} z, & \text{if } z > 0 \\ a(e^z - 1), & \text{if } otherwise \end{cases}$
- Scaled Exponential Linear Unit (SELU):  $SELU(z) = \lambda \begin{cases} z, & \text{if } z > 0 \\ ae^z - \alpha, & \text{if } otherwise \end{cases}$



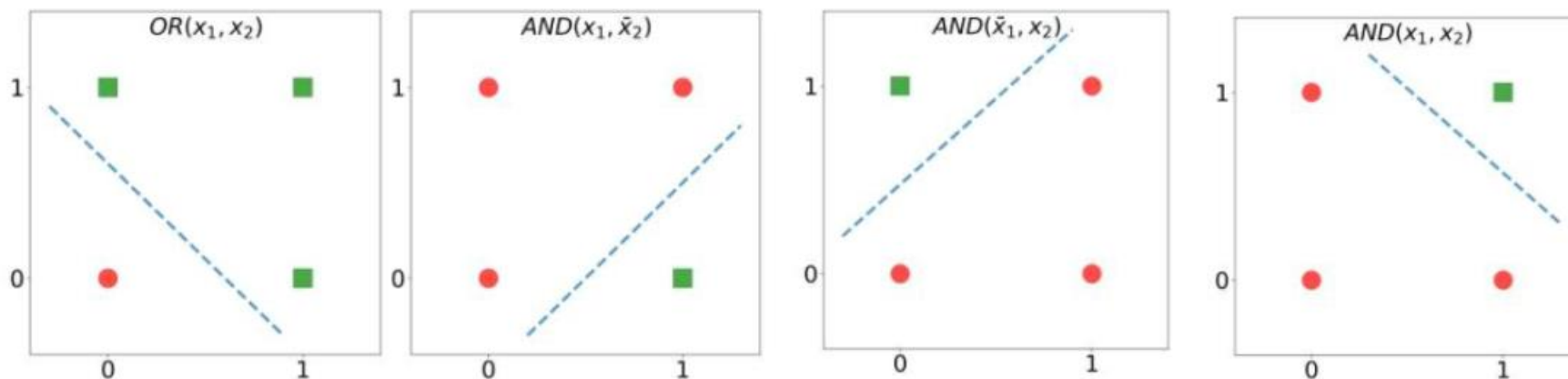
Один перцептрон (с любой функцией активации) может научиться классифицировать только линейно разделимые множества объектов.



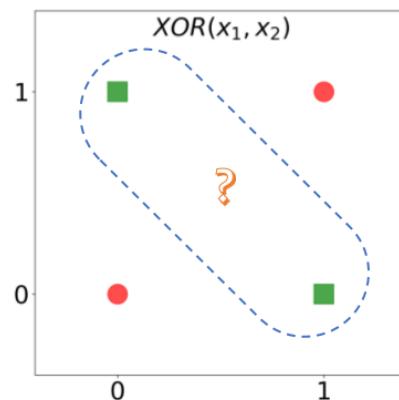
Пример поиска границы разделяющей два класса с помощью перцептрона с  $f = \sigma$



В 1969 году Марвин Мински публикует формальное доказательство ограниченности перцептрона и показывает, что он неспособен решать некоторые задачи в частности, один перцептрон не может реализовать функцию XOR.



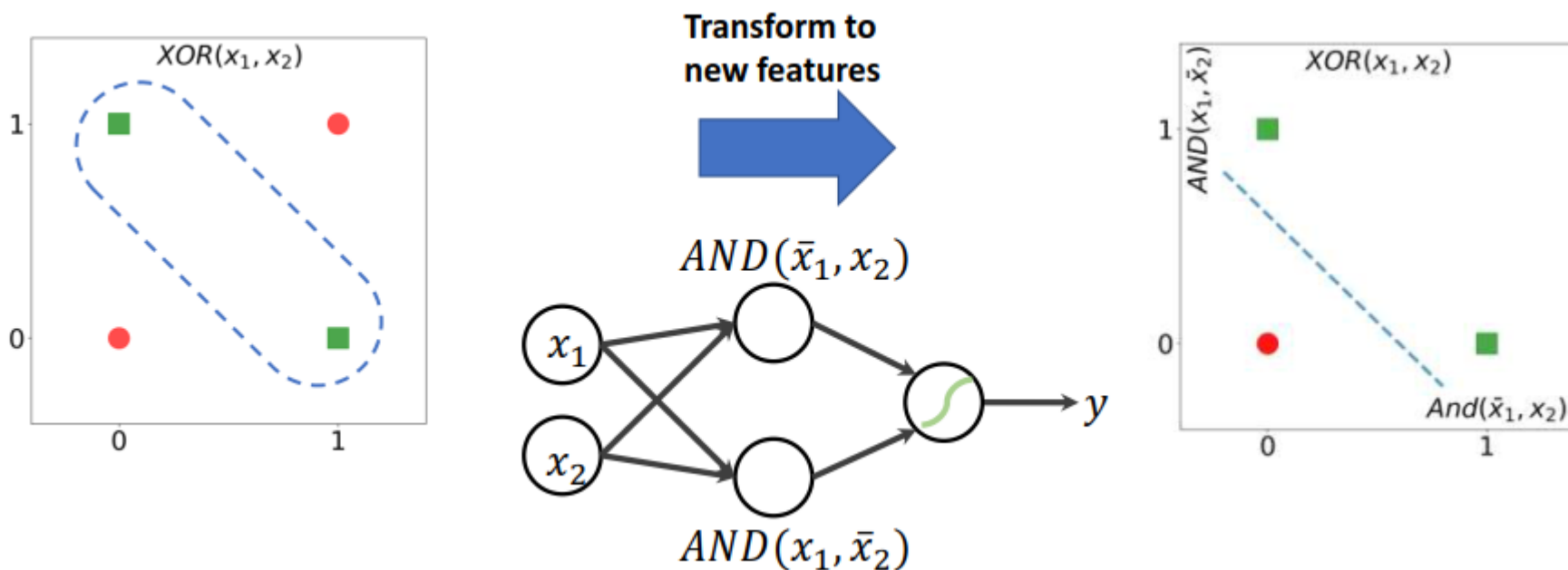
**Пример классификации перцептроном логических функций**







Сам М. Мински показал, что **XOR** может быть реализован многослойной нейронной сетью из перцептронов. **Нелинейная функция активации** является критически важным элементом перцептрона, без нее линейная комбинация перцептронов позволяла бы строить только линейные разделяющие поверхности.



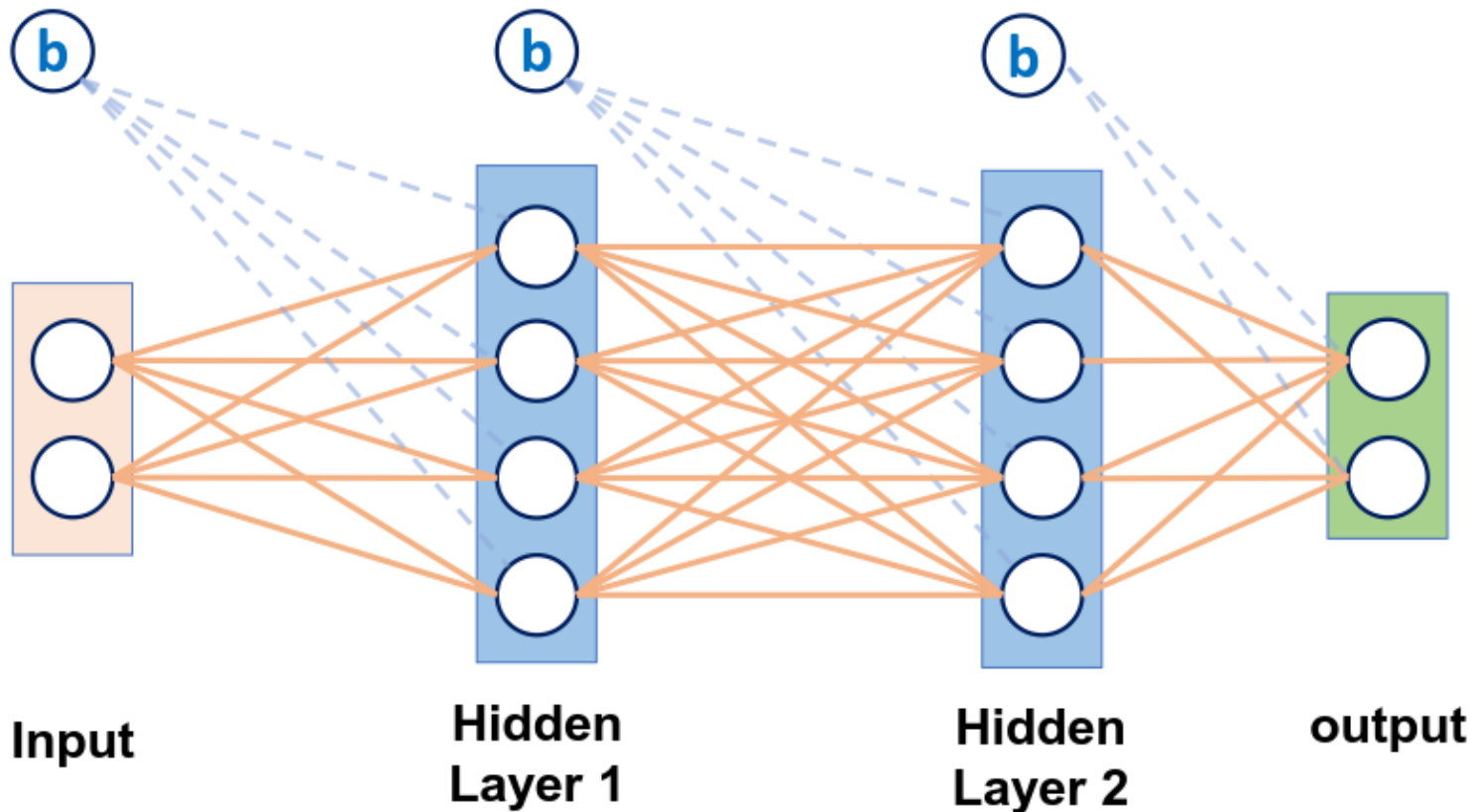
Несмотря на возможность работы многослойных перцептронов о механизм обучения, предложенный Розенблаттом, **не позволяет обучить многослойную сеть**.

После данных результатов интерес к нейронным сетям резко спадает, наступает период, позже названный "Первой зимой искусственного интеллекта".



Для решения сложных задач из перцептронов создается нейронная сеть:

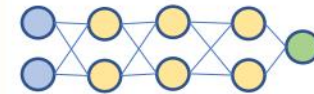
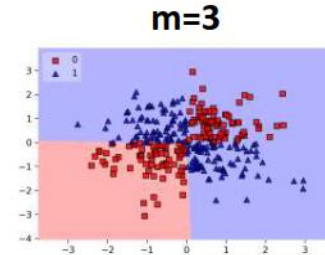
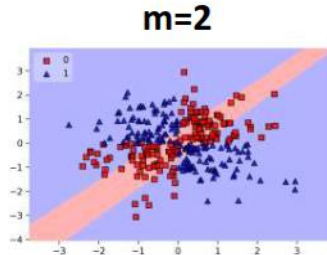
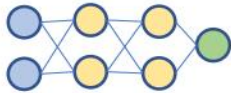
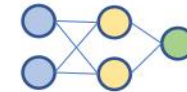
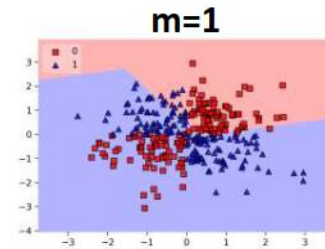
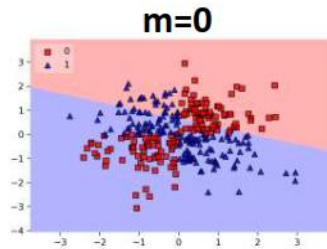
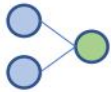
- для получения нескольких результатов нейроны организуются в слой (layer) содержащий столько перцептронов, сколько требуется выходов.
- выходы одного слоя могут использоваться в качестве входов следующего слоя - многослойные нейронные сети.





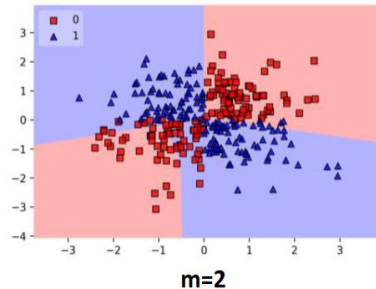
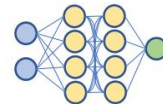
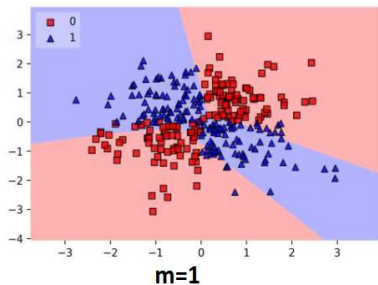
## Decision Boundaries

Number of hidden units:  $n=2$   
Varying the number of hidden layers ( $m$ )



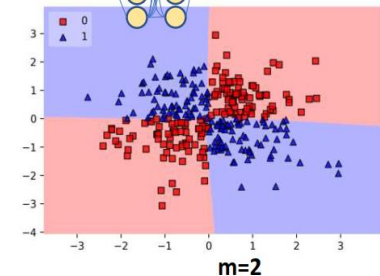
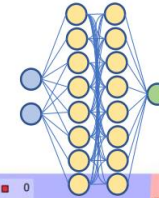
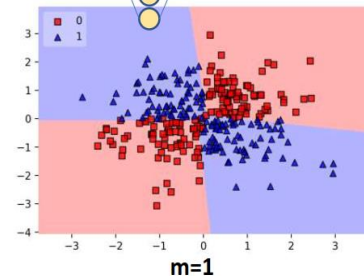
## Decision Boundaries

Number of hidden units:  $n=4$



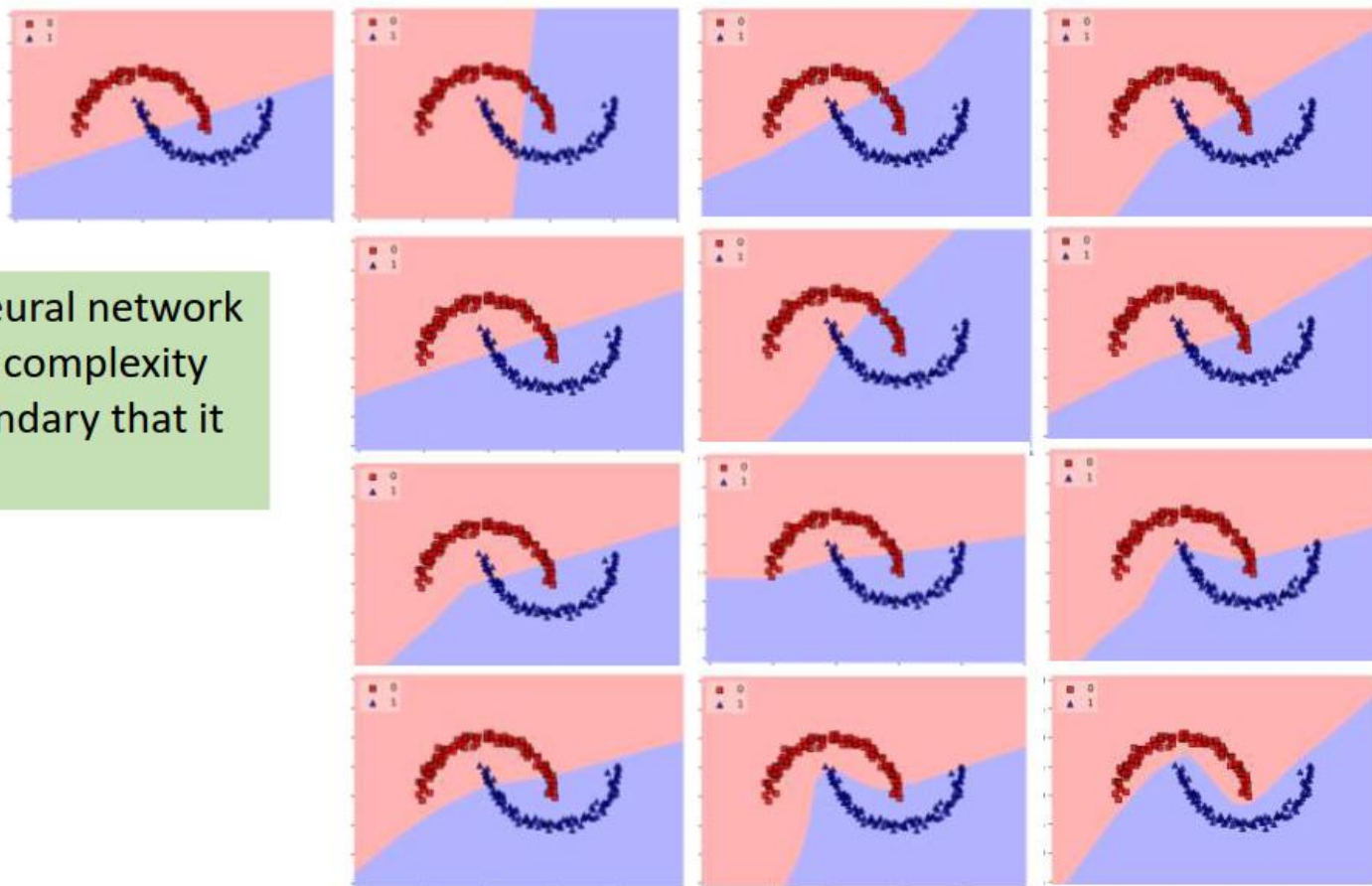
## Decision Boundaries

Number of hidden units:  $n=8$



 $m=0$  $m=1$  $m=2$  $m=3$  $n=2$  $n=4$  $n=8$  $n=16$ 

Capacity of a neural network determines the complexity of decision boundary that it can handle.





Спасибо за внимание!



111