



Нейронные сети

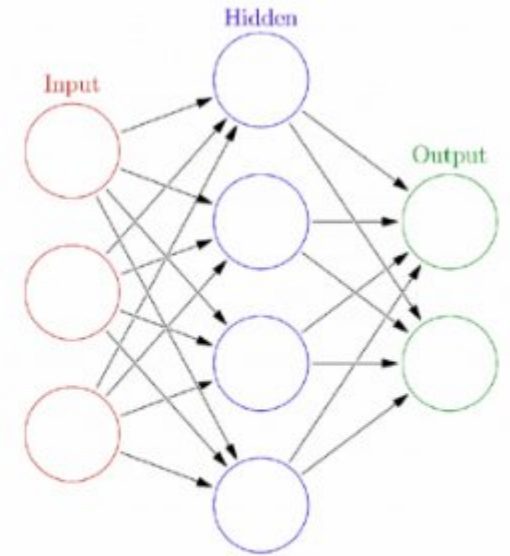
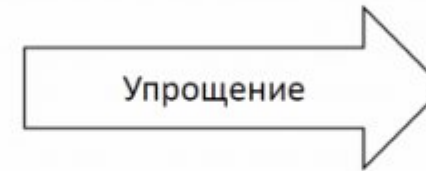
100 м\с

Понятие нейрона

упрощенная модель биологической нейронной сети, представляющая собой совокупность искусственных нейронов, взаимодействующих между собой.



Биологическая нейронная сеть



Искусственная нейронная сеть

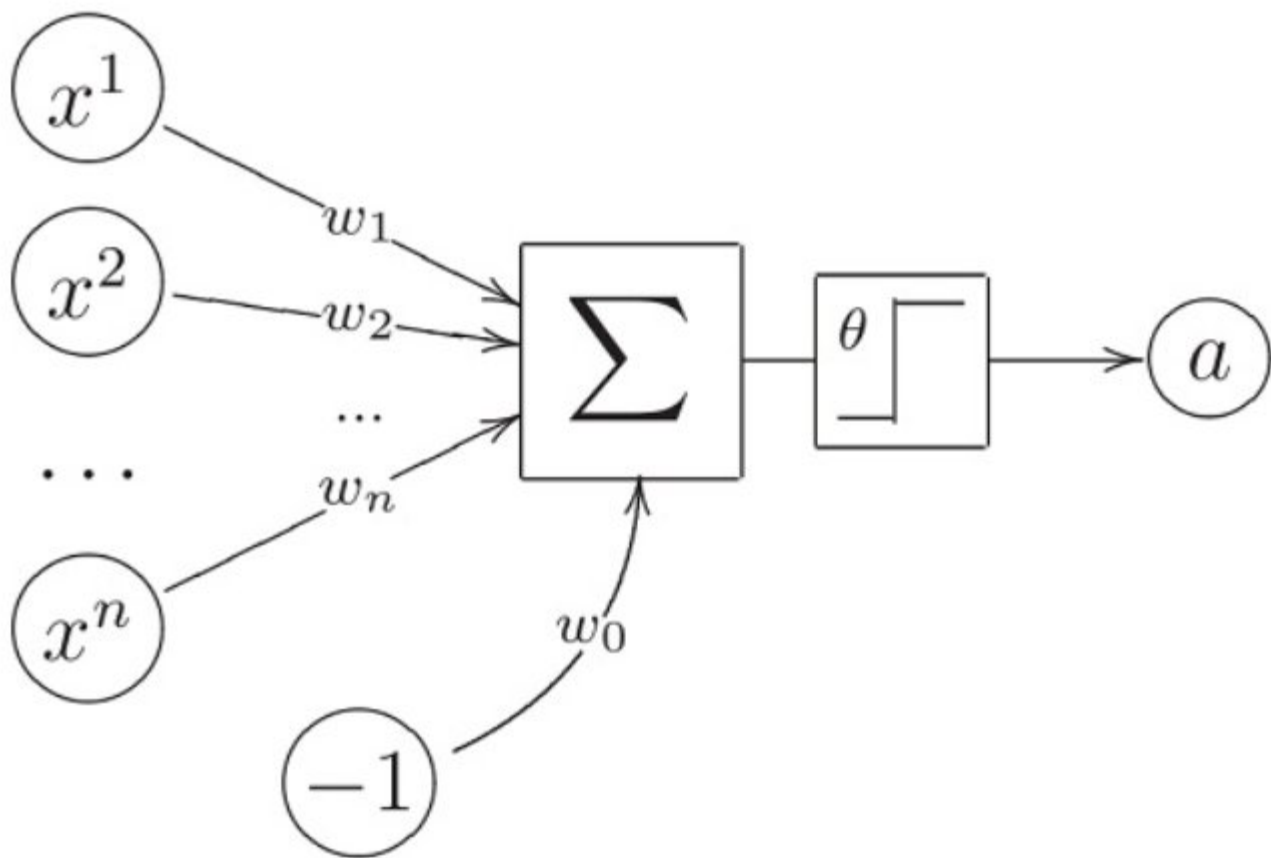
Клетка имеет множество разветвлённых отростков дендритов, и одно длинное тонкое волокно аксон, на конце которого находятся синапсы, примыкающие к дендритам других нервных клеток. Каждая нервная клетка может находиться в двух состояниях: обычном и возбуждённом. В возбуждённом состоянии клетка генерирует электрический импульс величиной около 100 мВ и длительностью 1 мс, который проходит по аксону до синапсов. Синапс при приходе импульса выделяет вещество, способствующее проникновению положительных зарядов внутрь соседней клетки.

Если суммарный заряд, попавший в клетку, превосходит некоторый порог, клетка возбуждается и генерирует импульс, который распространяется по аксону и доходит до синапсов, что способствует возбуждению следующих клеток. После возбуждения клетки наступает период релаксации некоторое время она не способна генерировать новые импульсы. Благодаря этому клетки работают по тактам, наподобие дискретных автоматов, а сеть в целом передаёт направленную волну импульсов.

задача

- X – выборка, n -мерное признаковое описание
- Y – множество допустимых ответов
- Построить алгоритм (модель) аппроксимации
- $A: Y \rightarrow X$

Модель МакКаллокаПитса

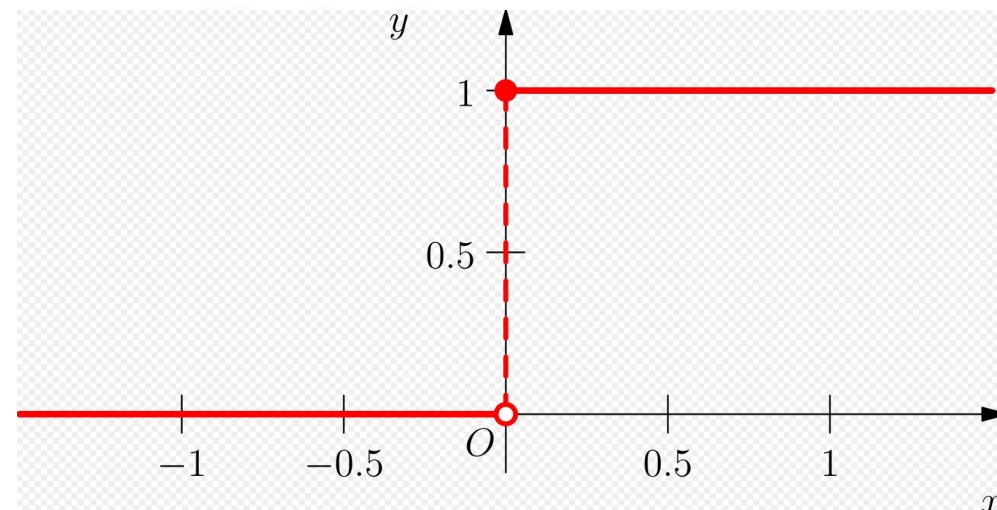


$$a(x) = \varphi\left(\sum_{j=0}^n w_j x^j\right) = \varphi(\langle w, x \rangle).$$

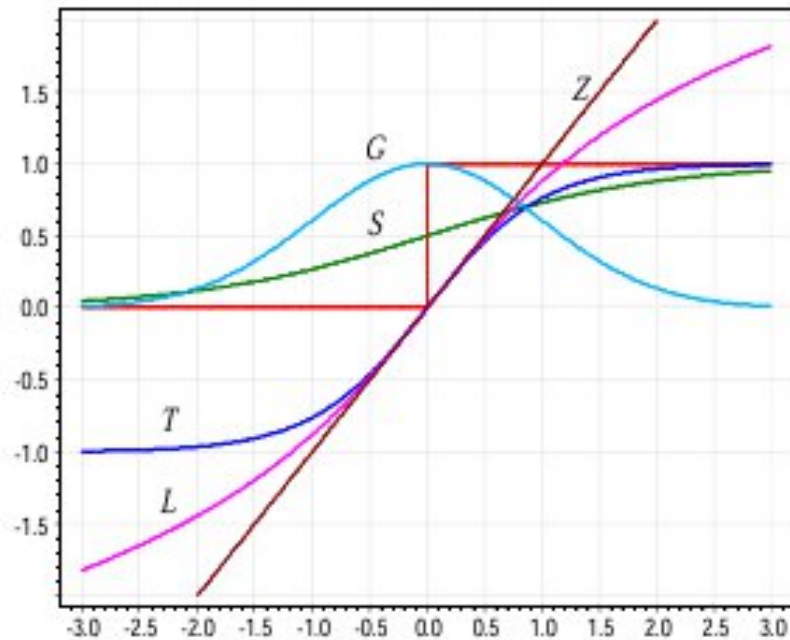
Пусть имеется n входных величин x_1, \dots, x_n бинарных признаков, описывающих объект. Значения этих признаков будем трактовать как величины импульсов, поступающих на вход нейрона через входных синапсов. Будем считать, что, попадая в нейрон, импульсы складываются с весами $\omega_1, \dots, \omega_n$.

Если вес положительный, то соответствующий синапс возбуждающий, если отрицательный, то тормозящий. Если суммарный импульс превышает заданный порог активации ω_0 , то нейрон возбуждается и выдаёт на выходе 1, иначе выдаётся 0.

$$a(x) = \varphi\left(\sum_{j=1}^n \omega_j x^j - \omega_0\right)$$



Функции активации



$$\theta(z) = [z \geq 0]$$

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

$$\text{th}(z) = 2\sigma(2z) - 1$$

$$\ln(z + \sqrt{z^2 + 1})$$

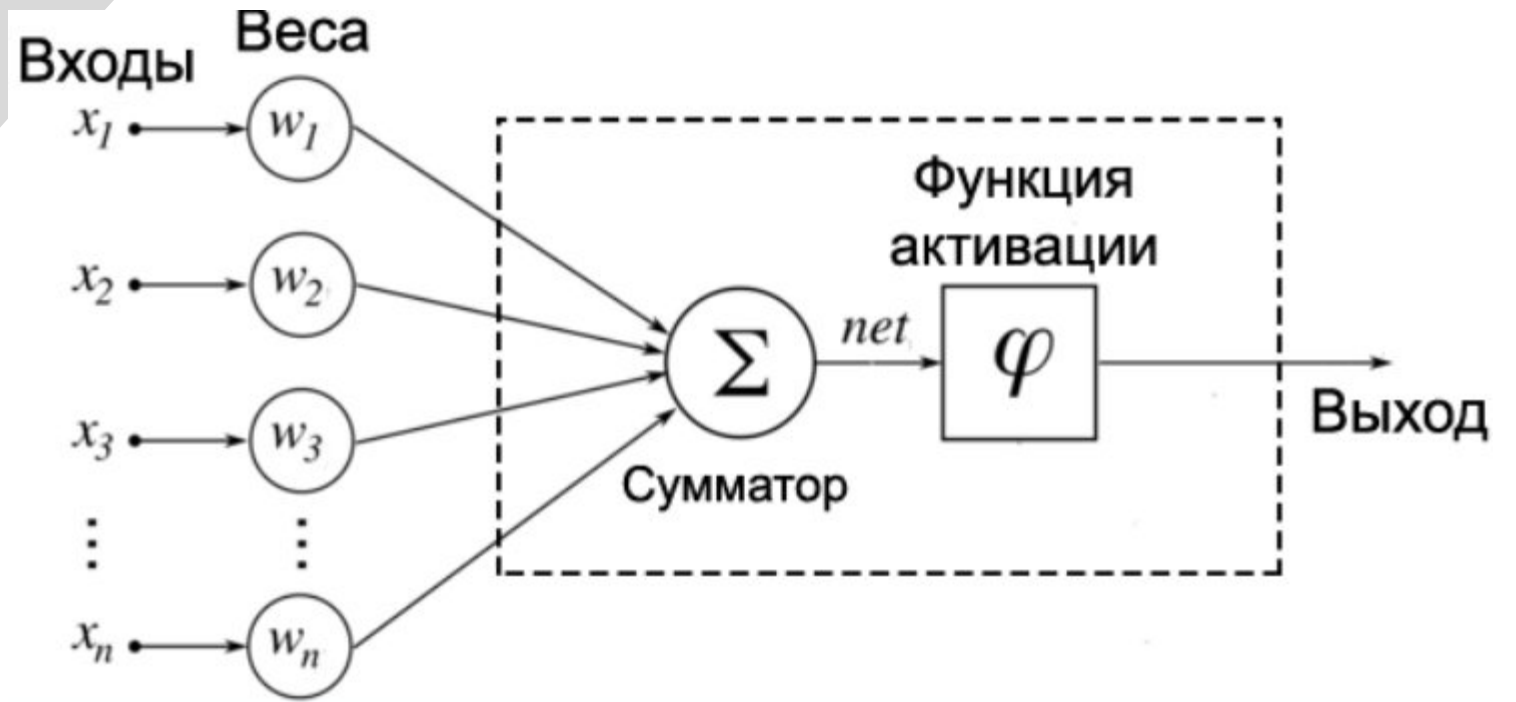
$$\exp(-z^2/2)$$

$$z$$

ступенчатая функция Хэвисайда;
сигмоидная функция (S);
гиперболический тангенс (T);
логарифмическая функция (L);
гауссовская функция (G);
линейная функция (Z);

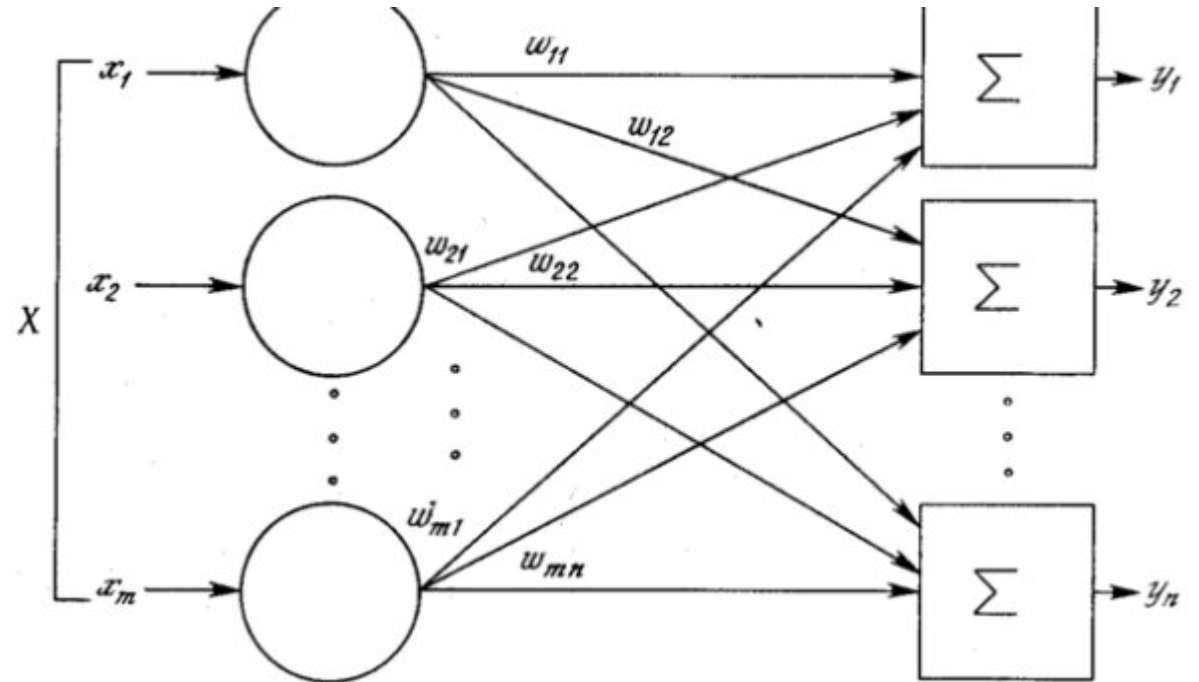
Линейная функция активации соответствует однослойной нейронной сети

Искусственный нейрон

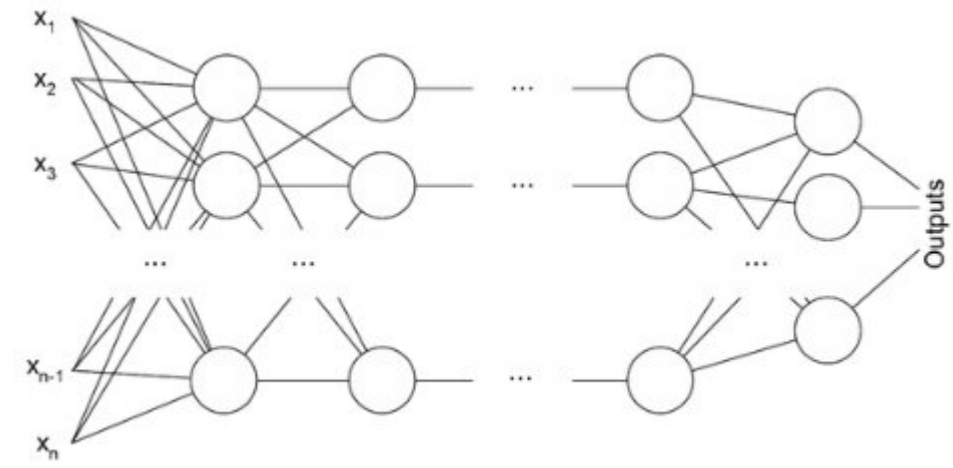
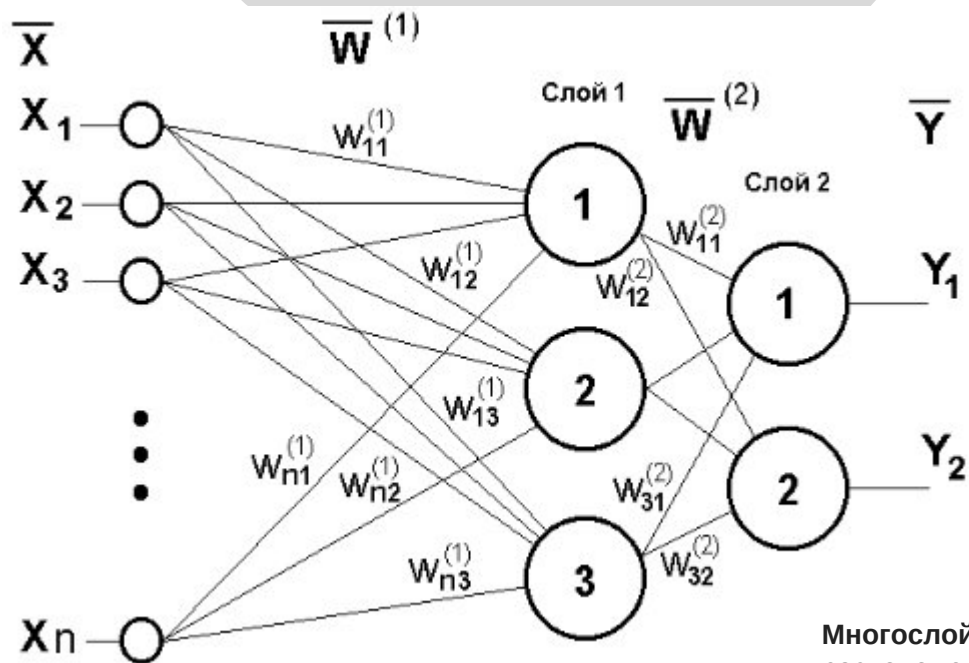


Однослойная нейронная сеть

сеть, в которой сигналы от входного слоя сразу подаются на выходной слой, который и преобразует сигнал и сразу же выдает ответ.



Двухслойная, многослойная нейронная сеть



Многослойная нейронная сеть (англ. *Multilayer neural network*) — нейронная сеть, состоящая из входного, выходного и расположенного(ых) между ними одного (нескольких) скрытых слоев нейронов. Помимо входного и выходного слоев эти нейронные сети содержат промежуточные, *скрытые слои*.

входной слой - один скрытый слой - выходной слой

Полносвязные сети

Теорема

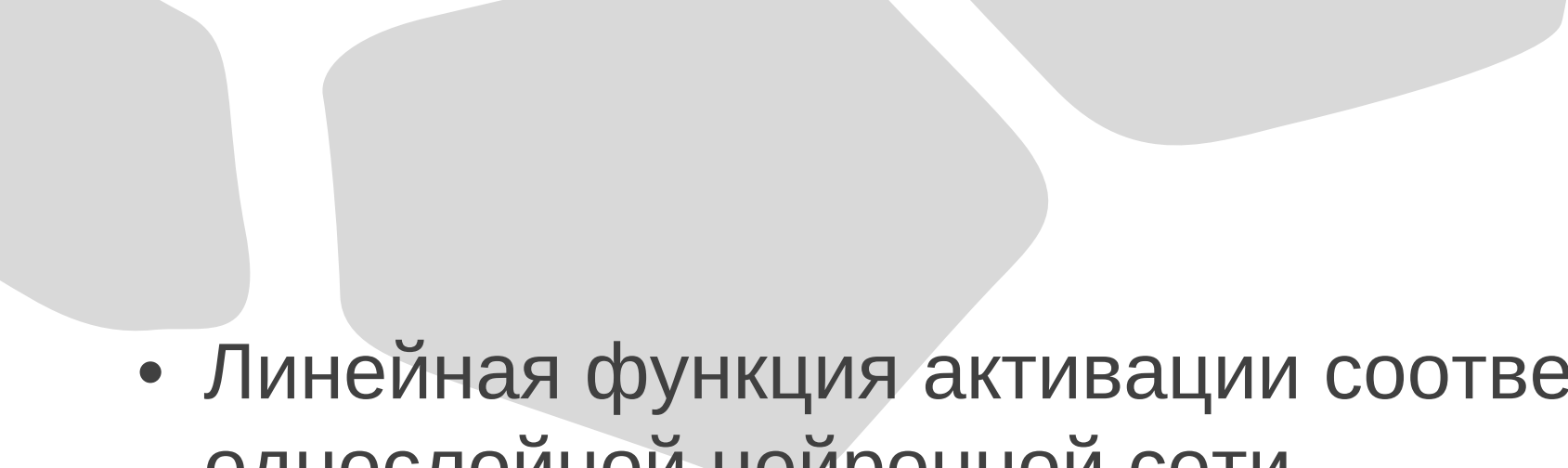
искусственная нейронная сеть прямой связи (англ. *feed-forward*; в которых связи не образуют циклов) с одним скрытым слоем может аппроксимировать любую непрерывную функцию многих переменных с любой точностью. Условиями являются:
достаточное количество нейронов скрытого слоя, удачный подбор $w_1, w_2, \dots, w_n, \alpha$, и θ , где
 w_i — веса между входными нейронами и нейронами скрытого слоя,
 α — веса между связями от нейронов скрытого слоя и выходным нейроном,
 θ — смещения для нейронов входного слоя.

Сети прямого распространения

- Сети прямого распространения (англ. Feedforward neural network) (feedforward сети) — искусственные нейронные сети, в которых сигнал распространяется строго от входного слоя к выходному. В обратном направлении сигнал не распространяется.

Сети с обратным распространением

- Сети с обратными связями (англ. Recurrent neural network) — искусственные нейронные сети, в которых выход нейрона может вновь подаваться на его вход. В более общем случае это означает возможность распространения сигнала от выходов к входам.

- 
- Линейная функция активации соответствует однослойной нейронной сети

Обучение нейронной сети

- Обучение нейронной сети — поиск такого набора весовых коэффициентов, при котором входной сигнал после прохода по сети преобразуется в нужный нам выходной.
- Обучающая/тестовые выборки

- Чтобы обучать эту функцию, сначала надо выбрать функцию ошибки, которую потом можно оптимизировать градиентным спуском. Число неверно классифицированных примеров не подходит на эту кандидатуру, потому что эта функция кусочно-гладкая, с массой разрывов: она будет принимать только целые значения и резко меняться при переходе от одного числа неверно классифицированных примеров к другому.

Обучение нейронной сети

- Случай бинарных признаков
- Перед началом обучения вектор весов некоторым способом инициализируется, например, заполняется нулевыми или случайными значениями
- Если выход сети совпадает с целевым значением, веса не меняются

Если $a(x_i) = 0$ и $y_i = 1$, то вектор весов w увеличивается. Увеличивать имеет смысл только те веса w_j , которые соответствуют ненулевым компонентам x_i^j , так как изменение других компонент не повлияет на результат. Поэтому можно положить $w := w + \eta x_i$, где η — некоторая положительная константа, называемая *темпом обучения* (learning rate).

Если $a(x_i) = 1$ и $y_i = 0$, то вектор весов уменьшается: $w := w - \eta x_i$.

Правило Хэбба

- Класс – число -1, 1
- Результат выхода нейрона:

$$a(x) = \text{sign}(\langle w, x \rangle).$$

Тогда несовпадение знаков $\langle w, x_i \rangle$ и y_i означает, что нейрон ошибается на объекте x_i .
При этом выражение (1.2) преобразуется в следующее правило модификации весов:

$$\text{если } \langle w, x_i \rangle y_i < 0 \text{ то } w := w + \eta x_i y_i, \quad (1.3)$$

Метод стохастического градиента

- L – функция потерь
- Задача: найти оптимальный набор весов w

$$Q(w) = \sum_{i=1}^{\ell} \mathcal{L}(a(x_i), y_i) \rightarrow \min_w,$$

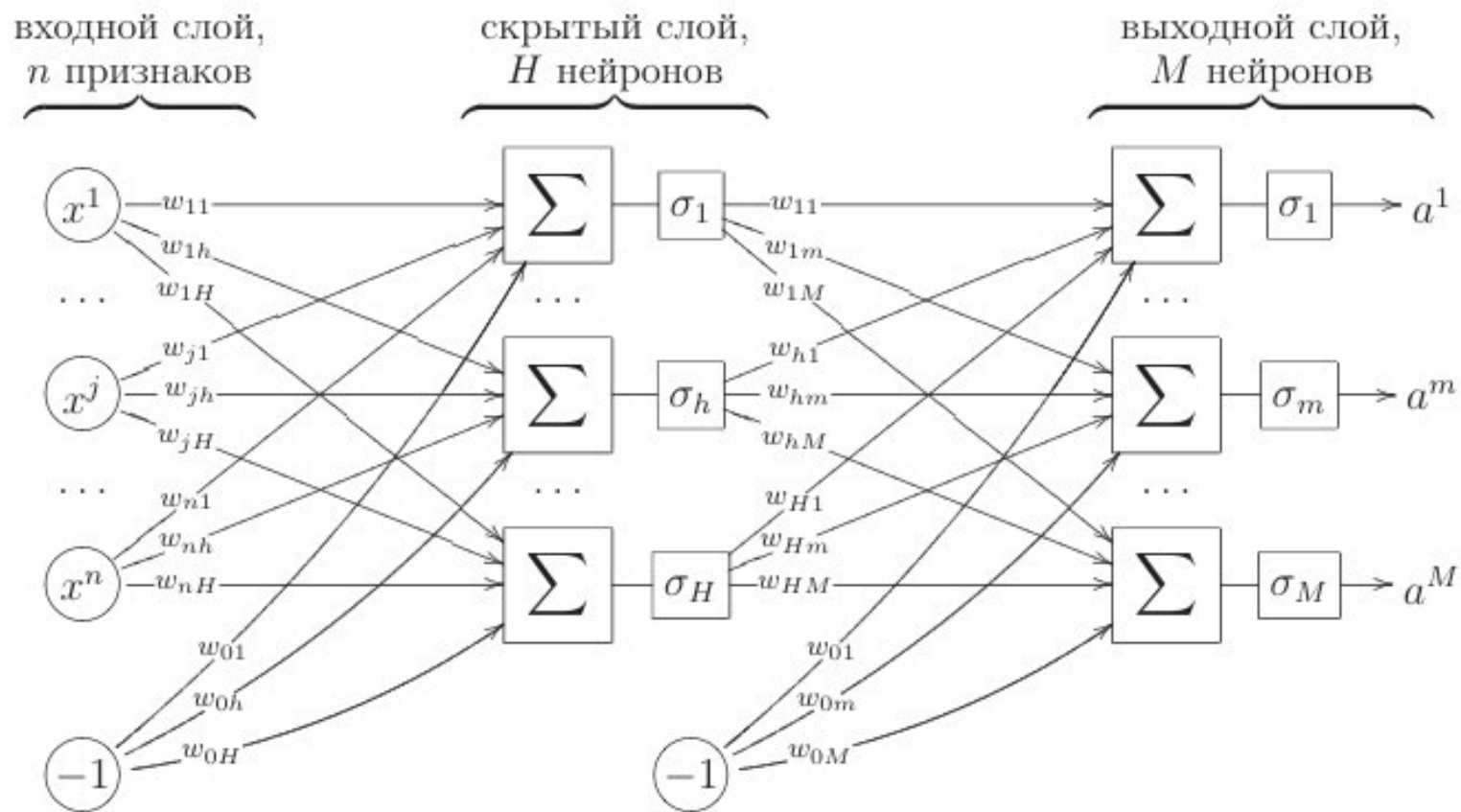
$$w := w - \eta \frac{\partial Q}{\partial w},$$

$$w := w - \eta \sum_{i=1}^{\ell} \mathcal{L}'_a(a(x_i), y_i) \varphi'(\langle w, x_i \rangle) x_i.$$

Если выбирать объекты по одному: стохастический
Правило выбора следующего объекта: случайное

$$w := w - \eta \mathcal{L}'_a(a(x_i), y_i) \varphi'(\langle w, x_i \rangle) x_i.$$

Многослойная нейронная сеть с одним скрытым слоем



Метод обратного распространения ошибки

Алгоритм 1.3. Обучение двухслойной сети методом back-propagation — обратного распространения ошибки

Вход:

$X^\ell = (x_i, y_i)_{i=1}^\ell$ — обучающая выборка, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}^M$;
 H — число нейронов в скрытом слое;
 η — темп обучения;

Выход:

синаптические веса w_{jh} , w_{hm} ;

1: инициализировать веса небольшими случайными значениями:

$$w_{jh} := \text{random}\left(-\frac{1}{2n}, \frac{1}{2n}\right);$$

$$w_{hm} := \text{random}\left(-\frac{1}{2H}, \frac{1}{2H}\right);$$

2: **повторять**

3: выбрать объект x_i случайным образом;

4: прямой ход:

$$u_i^h := \sigma_h\left(\sum_{j=0}^J w_{jh} v^j(x_i)\right), \text{ для всех } h = 1, \dots, H;$$

$$a_i^m := \sigma_m\left(\sum_{h=0}^H w_{hm} u^h(x_i)\right), \text{ для всех } m = 1, \dots, M;$$

$$\varepsilon_i^m := a_i^m - y_i^m, \text{ для всех } m = 1, \dots, M;$$

$$Q_i := \sum_{m=1}^M (\varepsilon_i^m)^2;$$

5: обратный ход:

$$\varepsilon_i^h := \sum_{m=1}^M \varepsilon_i^m \sigma'_m w_{hm}, \text{ для всех } h = 1, \dots, H;$$

6: градиентный шаг:

$$w_{hm} := w_{hm} - \eta \varepsilon_i^m \sigma'_m u^h, \text{ для всех } h = 0, \dots, H, m = 1, \dots, M;$$

$$w_{jh} := w_{jh} - \eta \varepsilon_i^h \sigma'_h x^j, \text{ для всех } j = 0, \dots, n, h = 1, \dots, H;$$


7: $Q := \frac{\ell-1}{\ell} Q + \frac{1}{\ell} Q_i$;

8: **пока** Q не стабилизируется:

- Метод не всегда сходится.
- Процесс градиентного спуска склонен застревать в многочисленных локальных минимумах функционала Q .
- Заранее фиксированное число нейронов скрытого слоя.
- При чрезмерном увеличении числа весов сеть склонна к переобучению.
- Если применяются функции активации типа
- сигмоидной или \tanh , то сеть может попадать в состояние паралича . Чем больше значения синаптических весов на входе нейрона, тем ближе значение производной σ' к нулю, тем меньше изменение синаптических весов. Если нейрон один раз попадает в такое состояние , то у него практически не остаётся шансов из неё выбраться. Парализоваться могут отдельные связи, нейроны, или вся сеть в целом

Приемы

- Нормализация данных
- Выбор функции активации: сигмоида – оценка вероятности принадлежности к классу, Применение нечётных функций, таких как $\text{th}(z) = 2\sigma(2z) - 1$, увеличивает скорость сходимости примерно в полтора раза. Для изменения эффекта паралича:
- Начальное приближение:
 - K – количество нейронов в слое
- Формирование начального приближения $\tilde{z} = \tilde{\gamma}z$ учать нейроны каждого слоя на различных случайных подвыборках $\tilde{x} \in [-\frac{1}{2k}, \frac{1}{2k}]$, либо подавать им на вход различные случайные подмножества признаков.

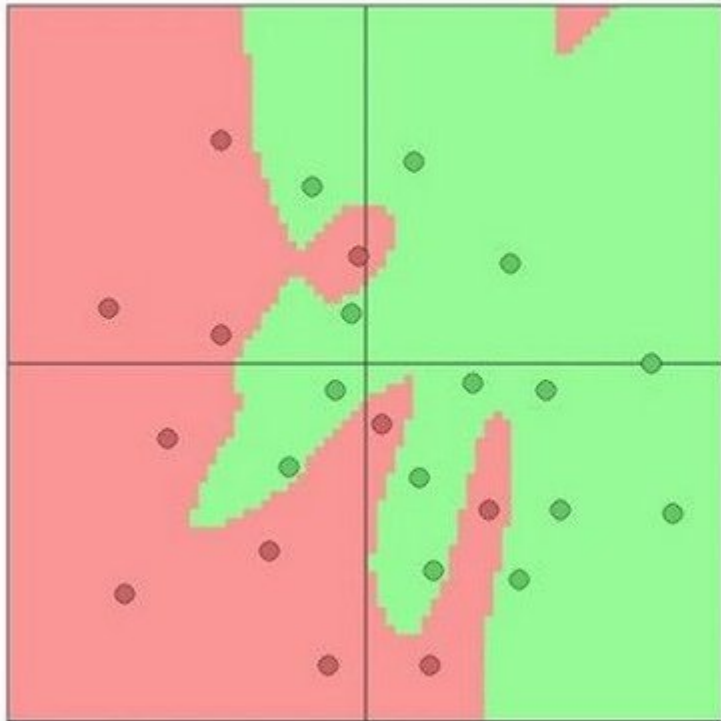
- 
- Перетасовка объектов: попеременно подавать объекты различных классов
 - Чаще предъявлять те объекты, на которых допущена ошибка (когда нет выбросов)
 - Не модифицировать веса, если ошибка окажется меньше порога

Регуляризация:

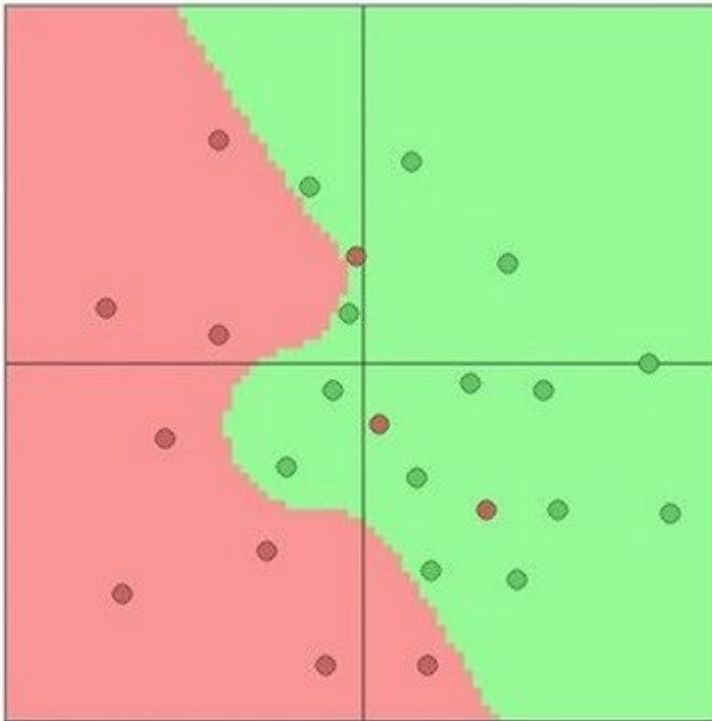
+ когда есть линейно зависимые и
сильно коррелированные признаки

Регуляризация

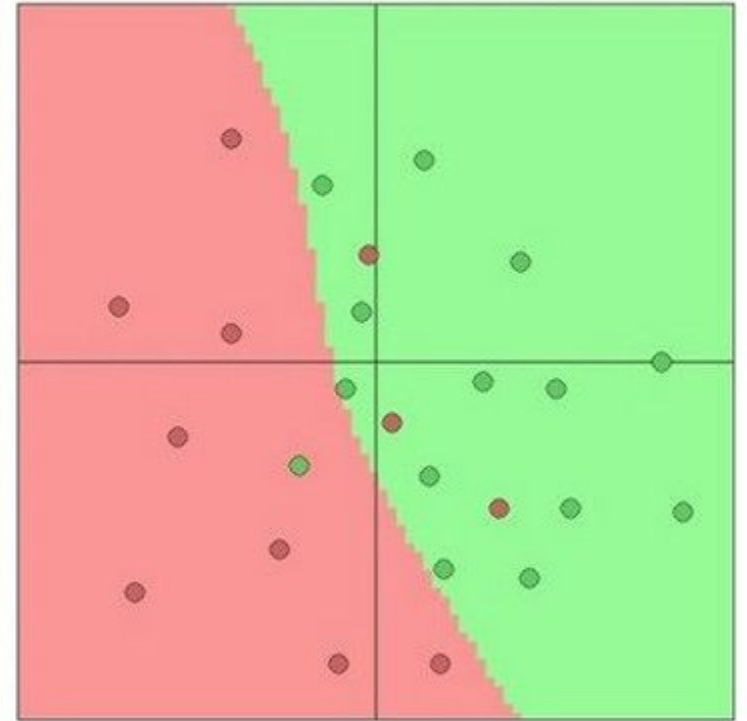
$\lambda = 0.001$



$\lambda = 0.01$




$\lambda = 0.1$

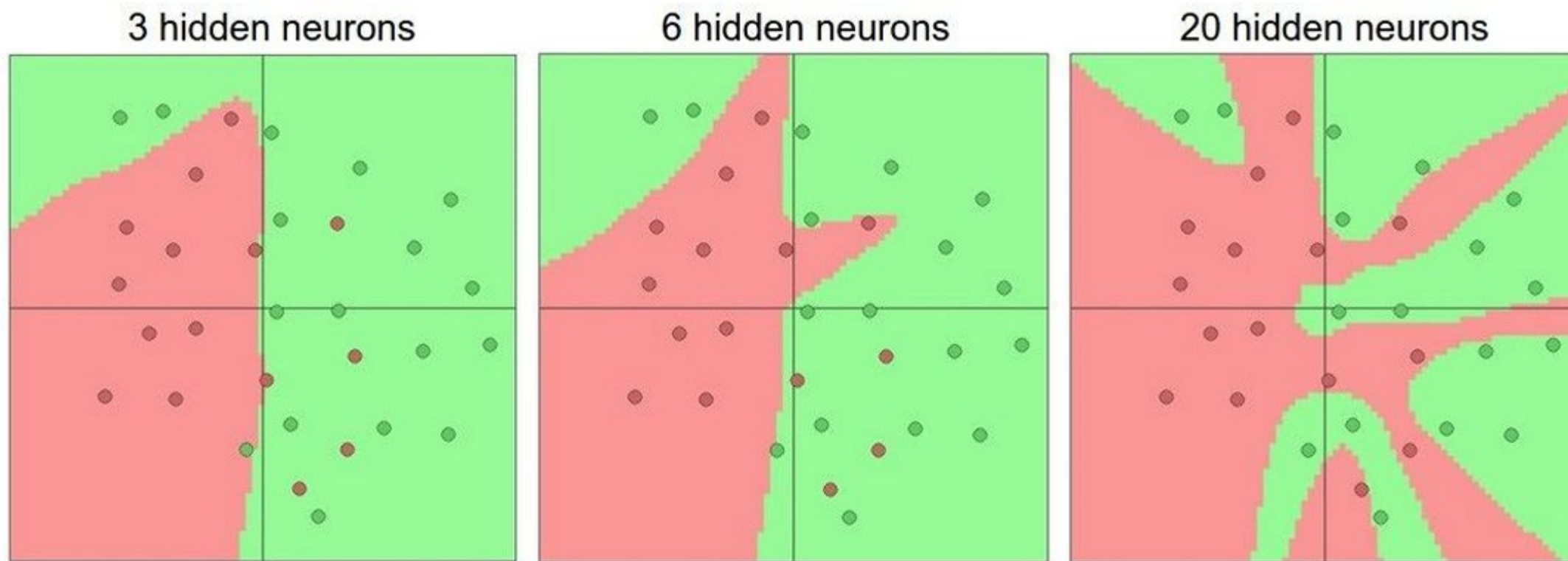


устойчивость к выбросам

$$L(W) = \frac{1}{N} \sum_{i=1}^N L_i(f(x_i, W), y_i) + \lambda R(W)$$

- 
- Постепенное уменьшение скорости обучения
 - Выбор адаптивного шага скорости обучения
 - Выбивание из локального минимума, «потряхивание градиента» – случайная модификация веса
 - Выбор критерия останова: экспоненциальное скользящее среднее, с бОльшим весом объектов, предъявленных последними
 - Ранний останов
 - Обучение пакетами

Выбор количества нейронов в скрытом слое

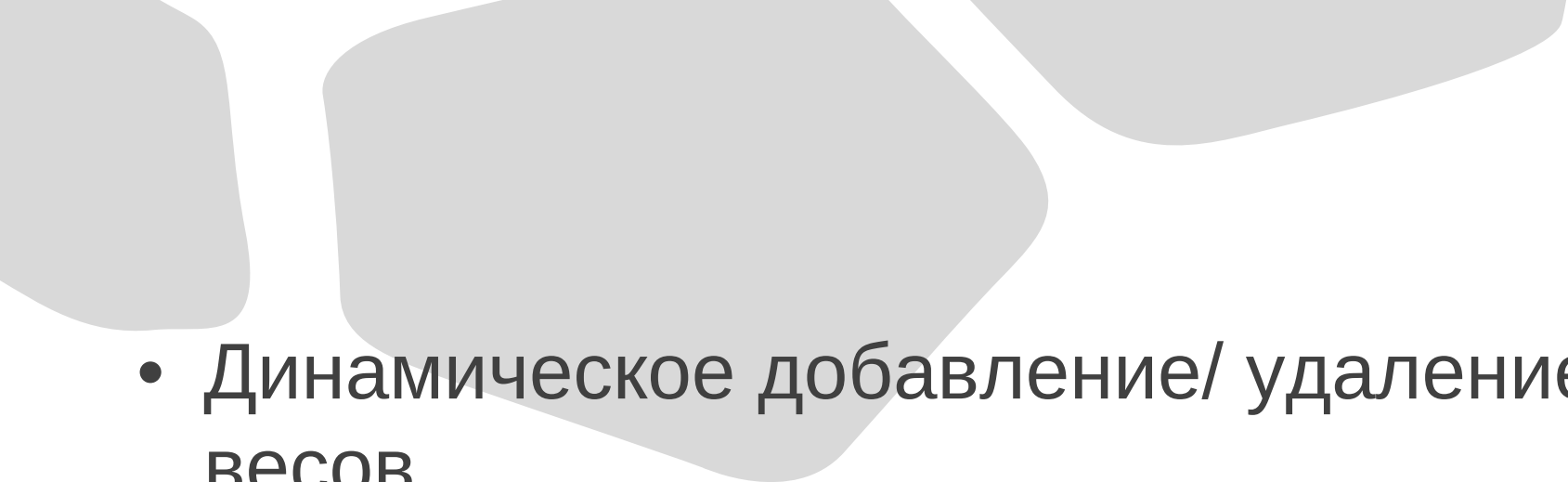


адаптация к данным - переобучение?

Динамически изменять количество весов

Выбор структуры сети

- Линейная разделимость: однослойная
- Иначе: двуслойная
- При сложной зависимости: трехслойная

- 
- Динамическое добавление/ удаление менее значимых весов