

Искусственные нейронные сети

Семинар по вычислительному интеллекту

Докладчик: Цой Ю.Р.

г. Томск, 06 апреля 2010 г.

<http://qai.narod.ru/TomskWorkshop/>



Введение

Искусственные нейронные сети – математические модели, а также их программные или аппаратные реализации, построенные по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма.

Основные решаемые задачи:

- классификация
- аппроксимация
- оптимизация
- кластеризация

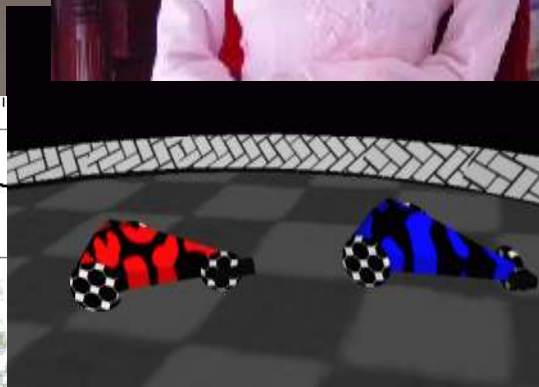
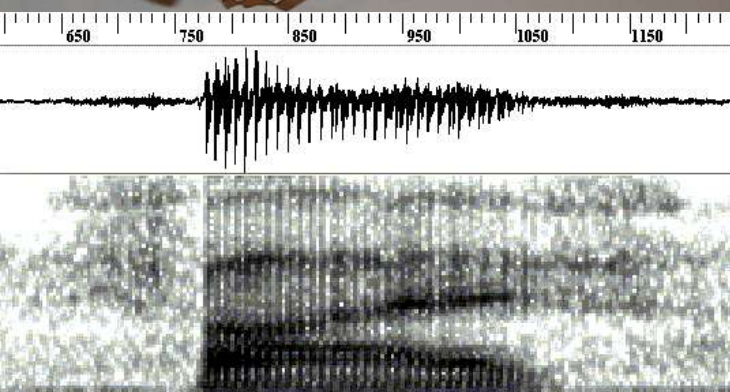
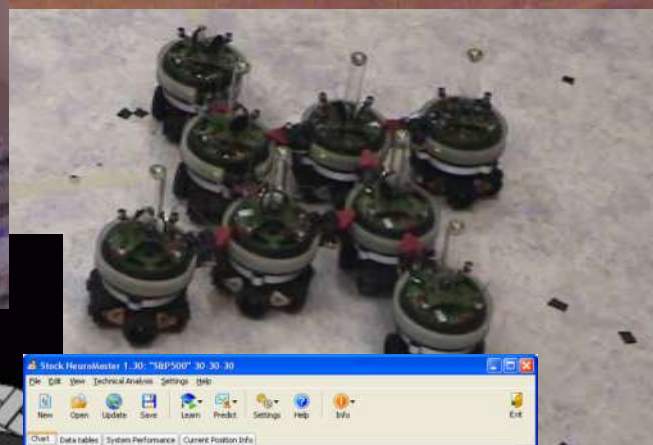
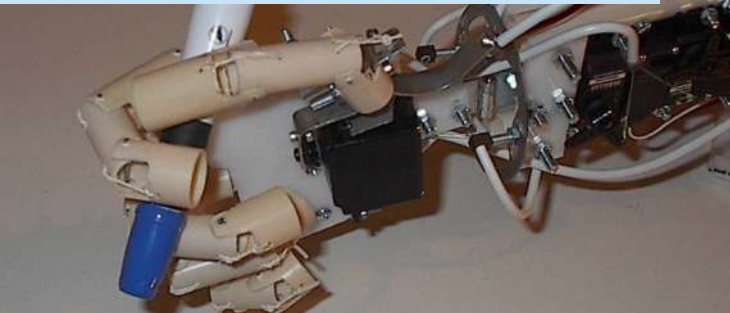
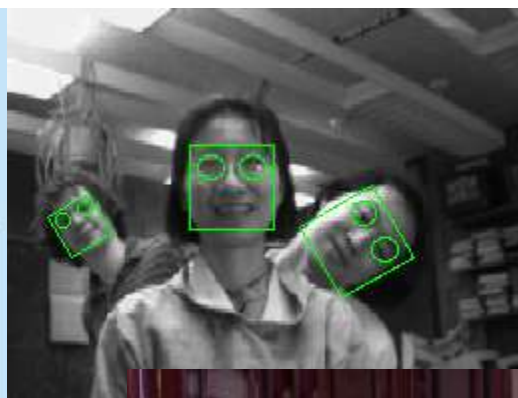
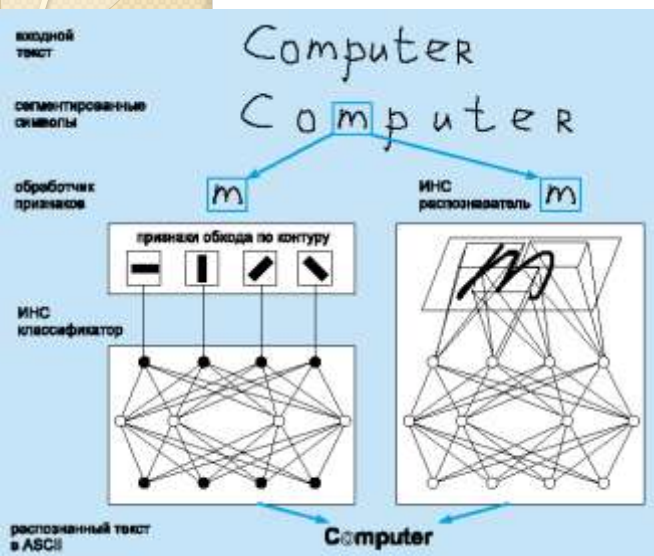


Введение

Цели нейроинформатики (по Д. Хинтону):

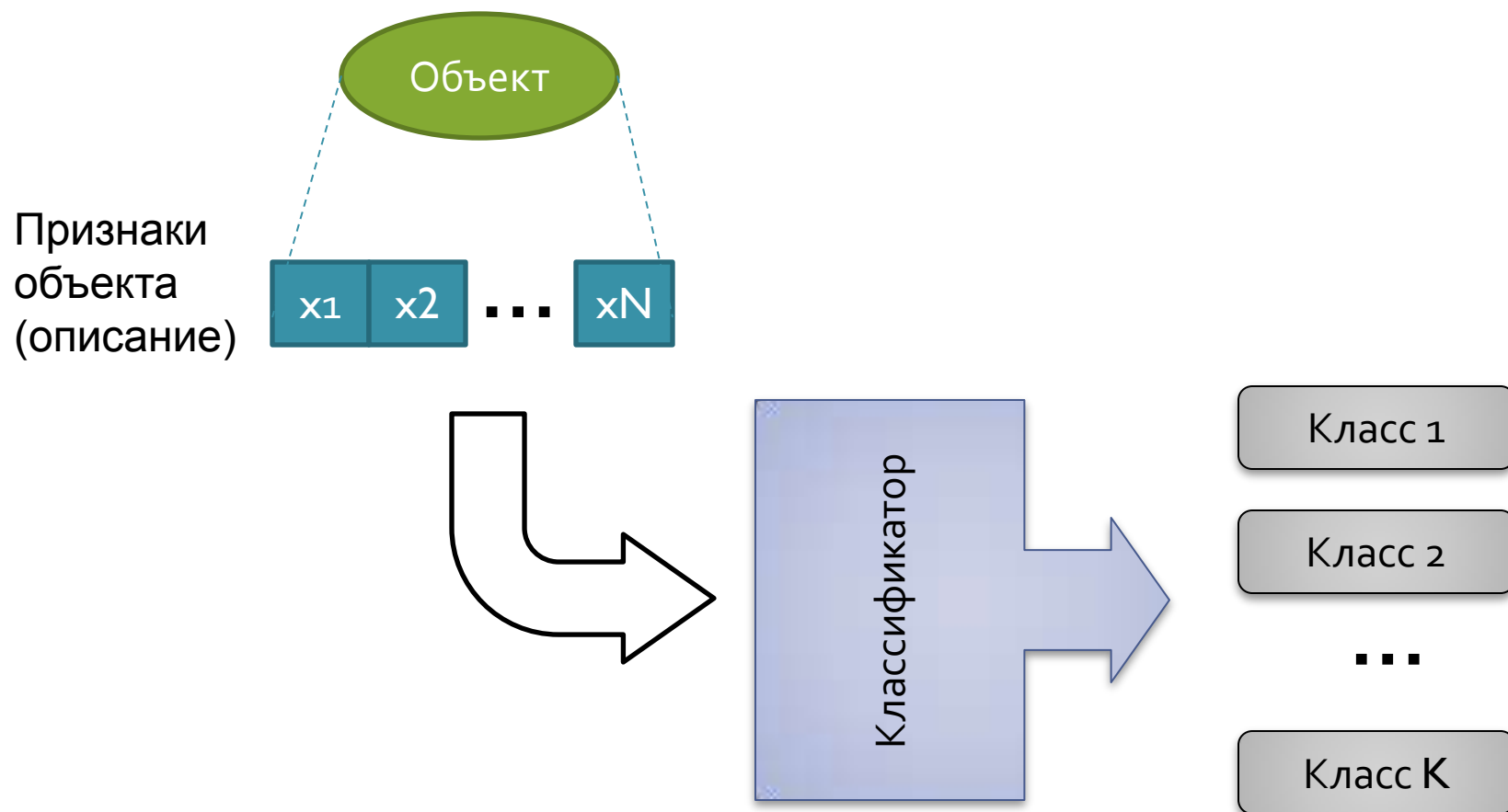
- Понимание принципов работы мозга
(Хинтон: «Он большой, очень сложный и состоит из противного вещества, которое отмирает, если его тыкать» 😊)
- Понимание новых принципов вычислений, сильно отличающихся от традиционных.
- Решение практических задач с применением оригинальных алгоритмов.

Новые алгоритмы могут быть очень полезными, даже если они не имеют ничего общего с работой мозга



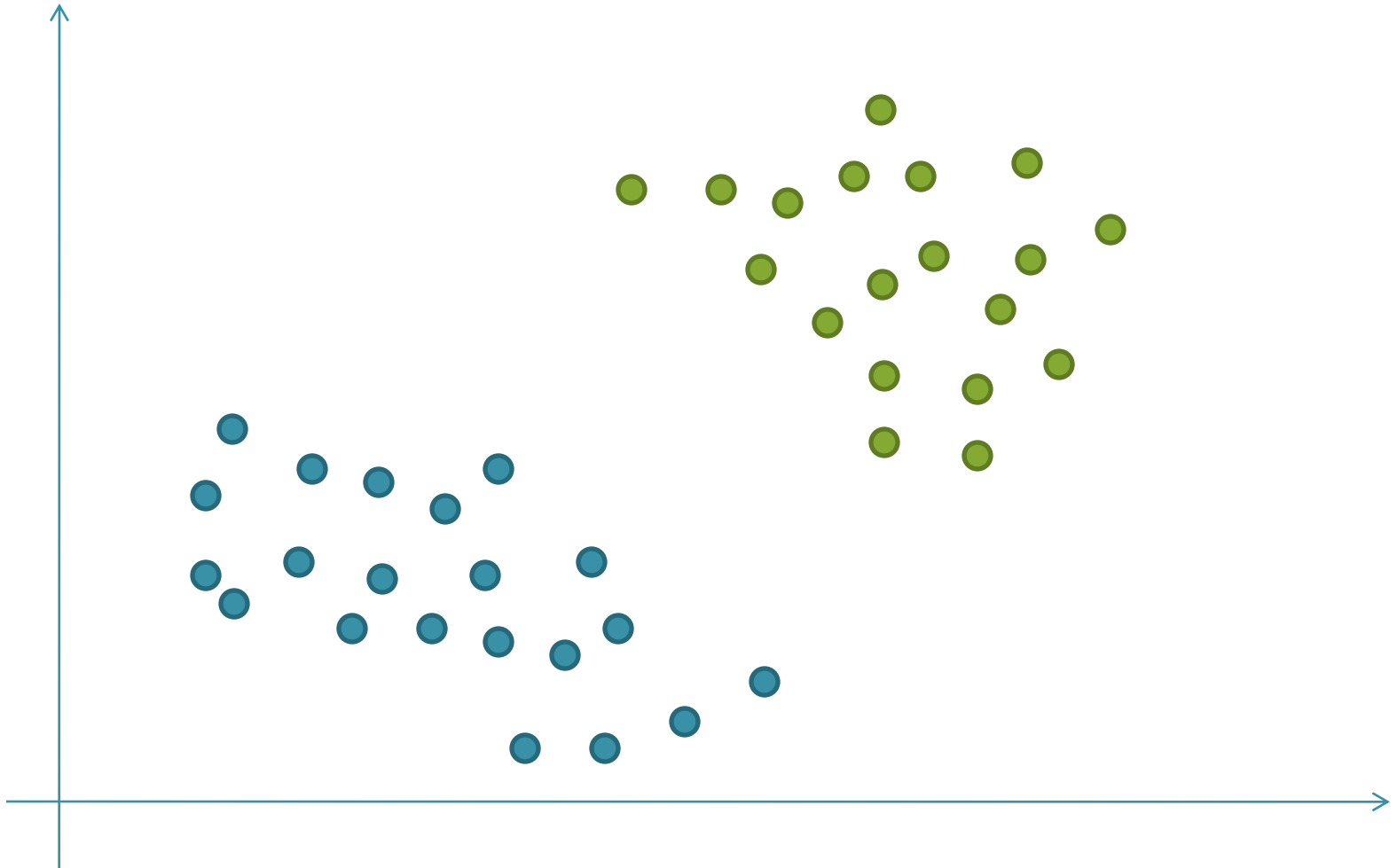
\$sil-f f-\$bck_r \$ret-<al <al> al-\$lab \$fnt_l<v v>\$sil

Задача классификации

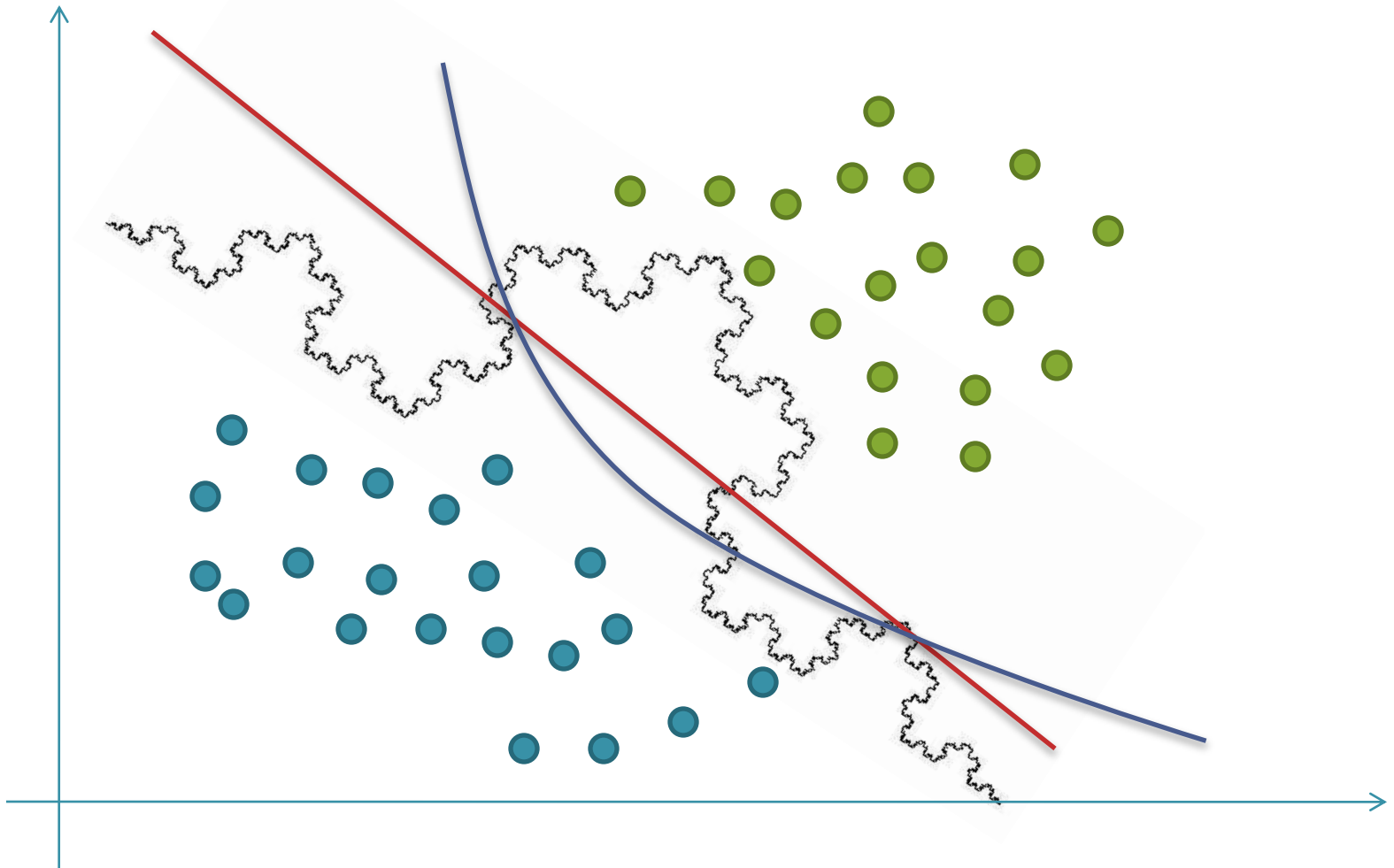


Классификатор осуществляет отображение из пространства признаков в пространство классов. **Как обучить классификатор?**

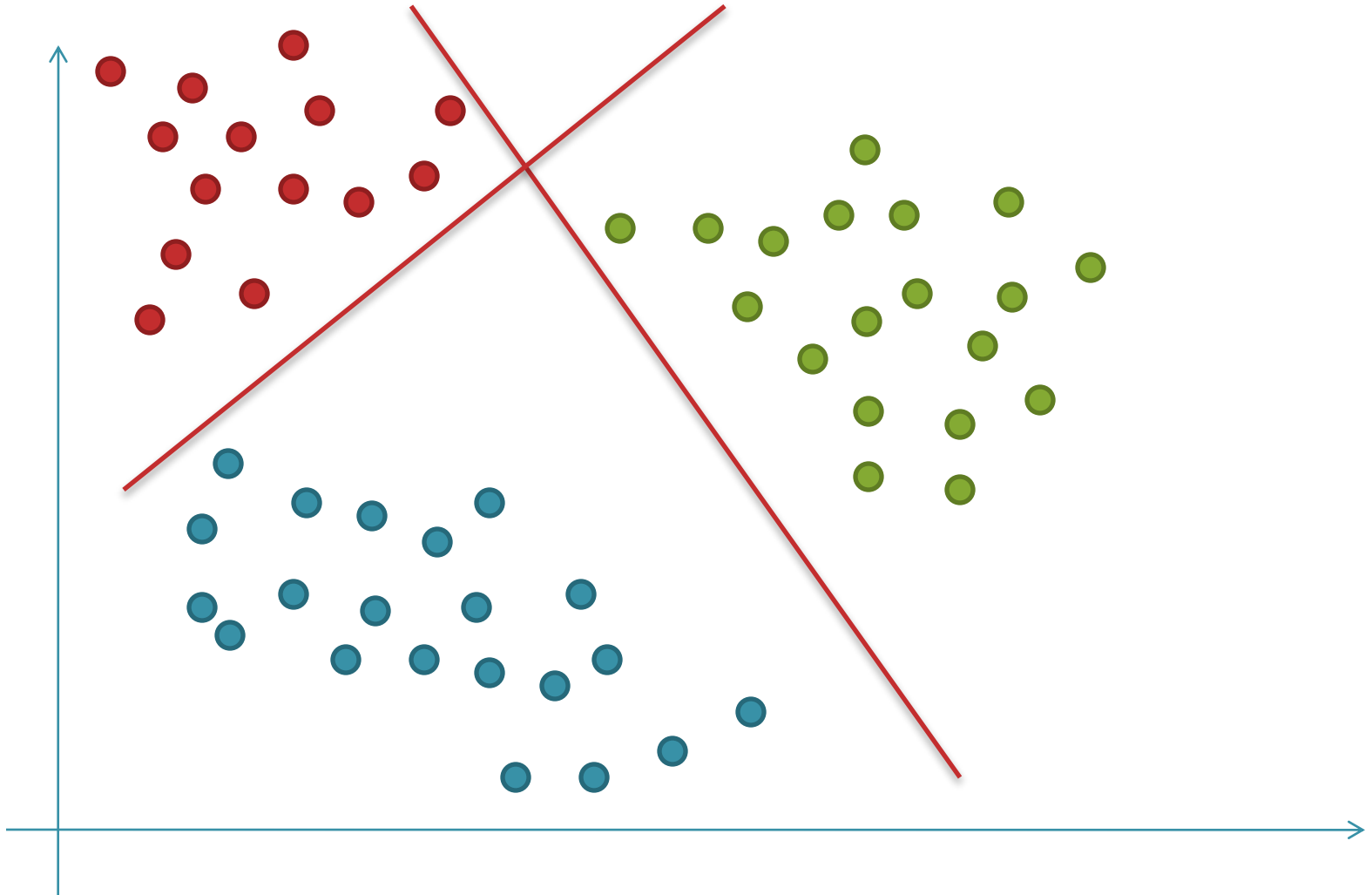
Классификация. Простейший случай



Классификация. Простейший случай. 1 способ.



Классификация. Простейший случай. 1 способ.



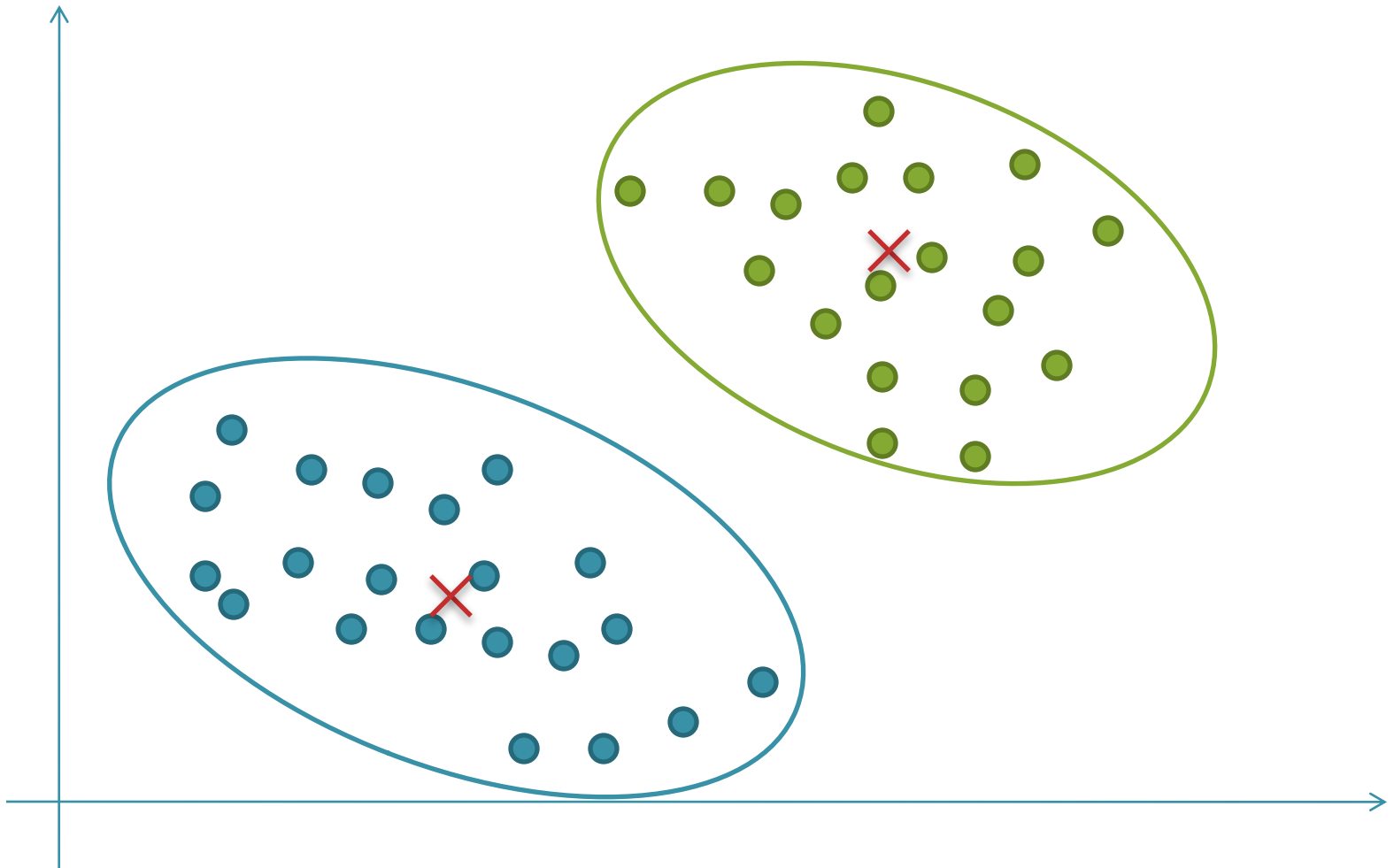
Классификация. Простейший случай. 1 способ.

Всегда ли можно линейно разделить пространство признаков?

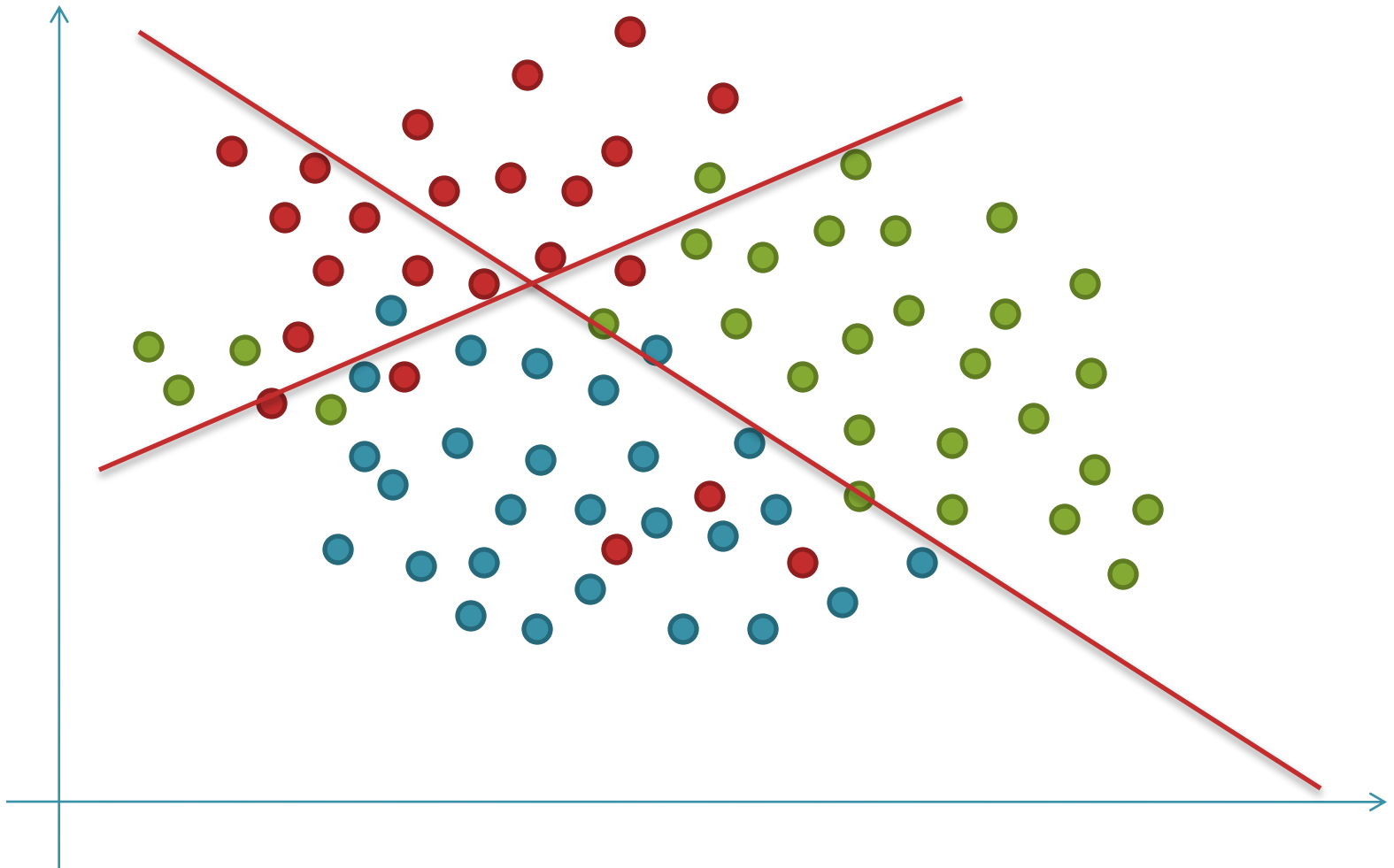
1. Если каждый класс отделяется от всех остальных классов одной линейной решающей функцией.
2. Каждые два класса можно разделить одной линейной решающей функцией.
3. Любой из m классов отделяется от всех остальных классов с помощью m решающих функций.

Возможность разделения объектов из разных классов весьма подробно изучена в рамках **классификации с помощью разделяющих функций**

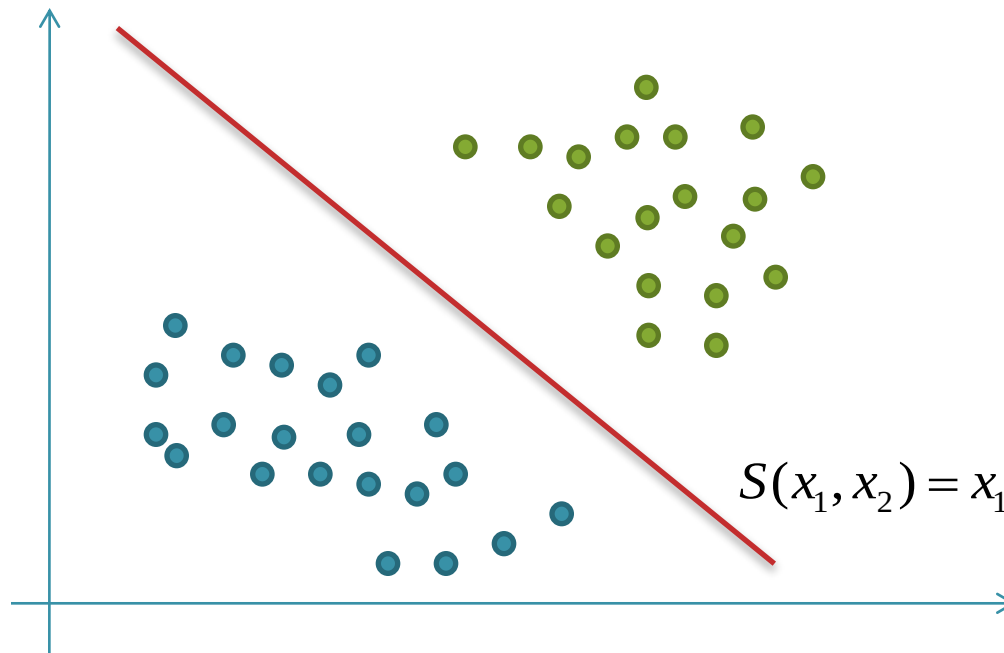
Классификация. Простейший случай. 2 способ.



Классификация. Более «неудобный» и реалистичный случай



Переходим к нейронам. 1 способ.

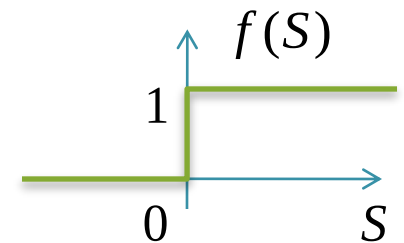


$$S(x_1, x_2) = x_1 * w_1 + x_2 * w_2 + b = \mathbf{x}^T \mathbf{w} + b$$

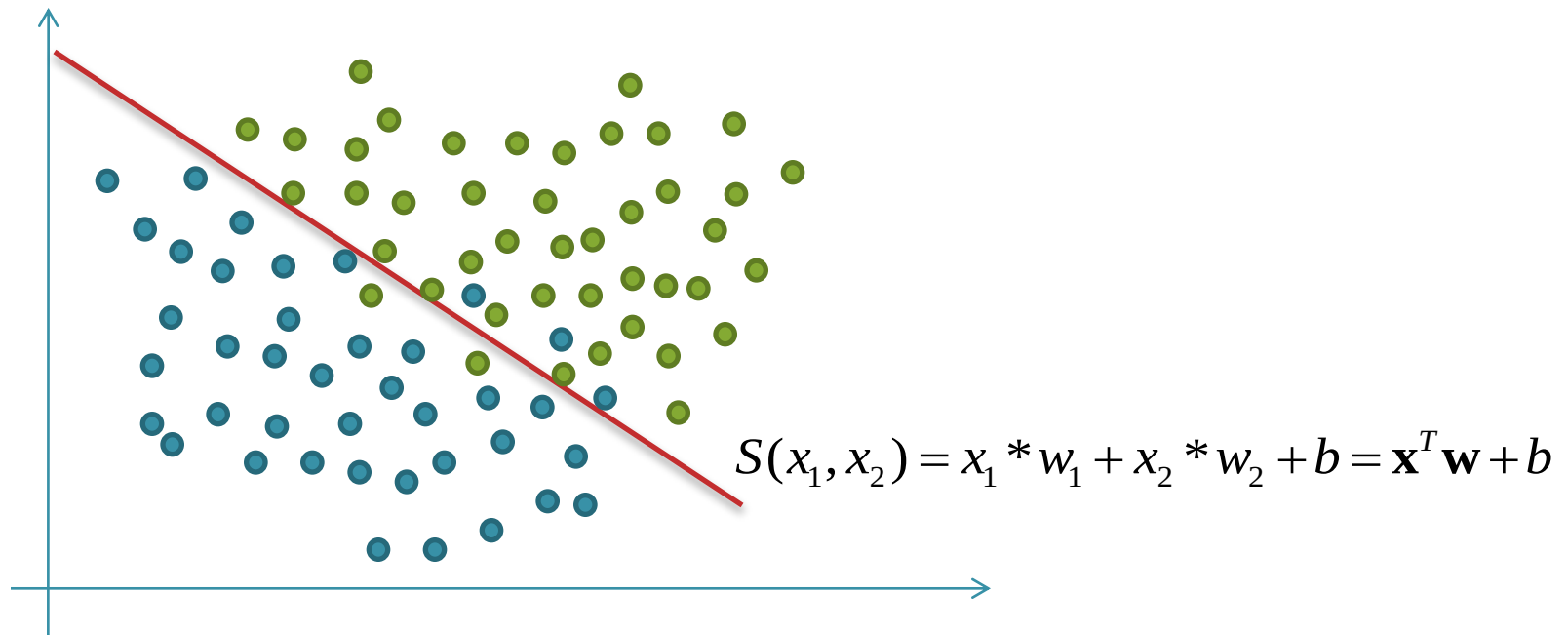
$$f(S) = \begin{cases} 1, S \geq 0 \\ 0, S < 0 \end{cases}$$

$$f(S) = \begin{cases} 1, S \geq 0 \\ -1, S < 0 \end{cases}$$

← Пороговый нейрон



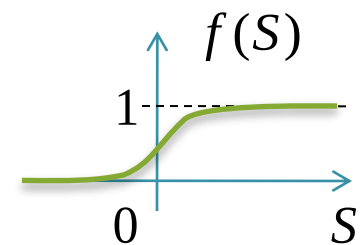
Переходим к нейронам. 1 способ.



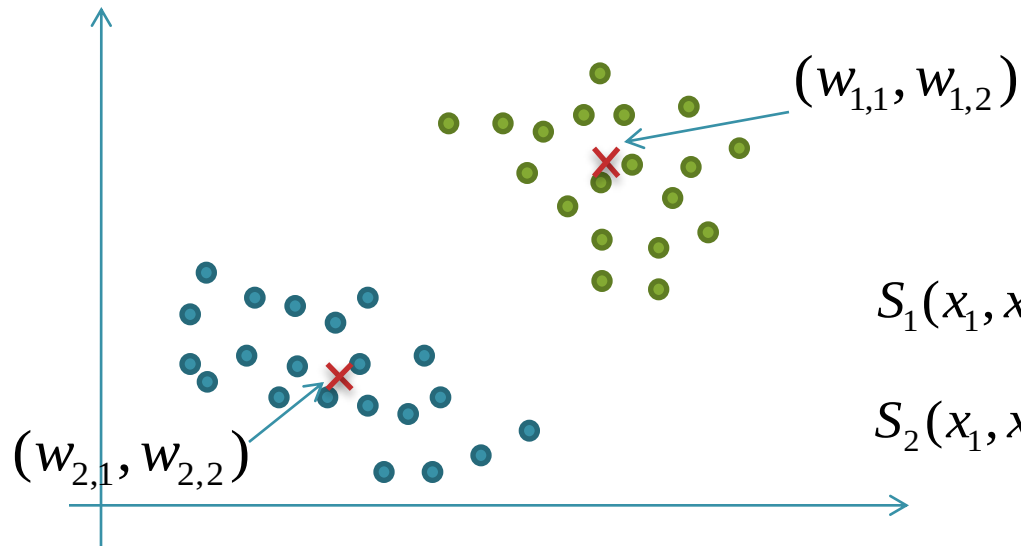
$$f(S) = \frac{1}{1 + \exp(-aS)}$$

$$f(S) = \frac{1 - \exp(-aS)}{1 + \exp(-aS)}$$

Сигмоидный нейрон



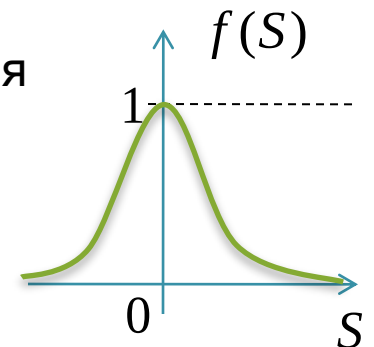
Переходим к нейронам. 2 способ.



$$S_1(x_1, x_2) = x_1 * w_{1,1} + x_2 * w_{1,2} = \mathbf{x}^T \mathbf{w}_1$$

$$S_2(x_1, x_2) = x_1 * w_{2,1} + x_2 * w_{2,2} = \mathbf{x}^T \mathbf{w}_2$$

$$f(S) = \exp\left(-\frac{S^2}{2}\right) \quad \leftarrow \text{Гауссовская функция}$$



От нейронов к сетям



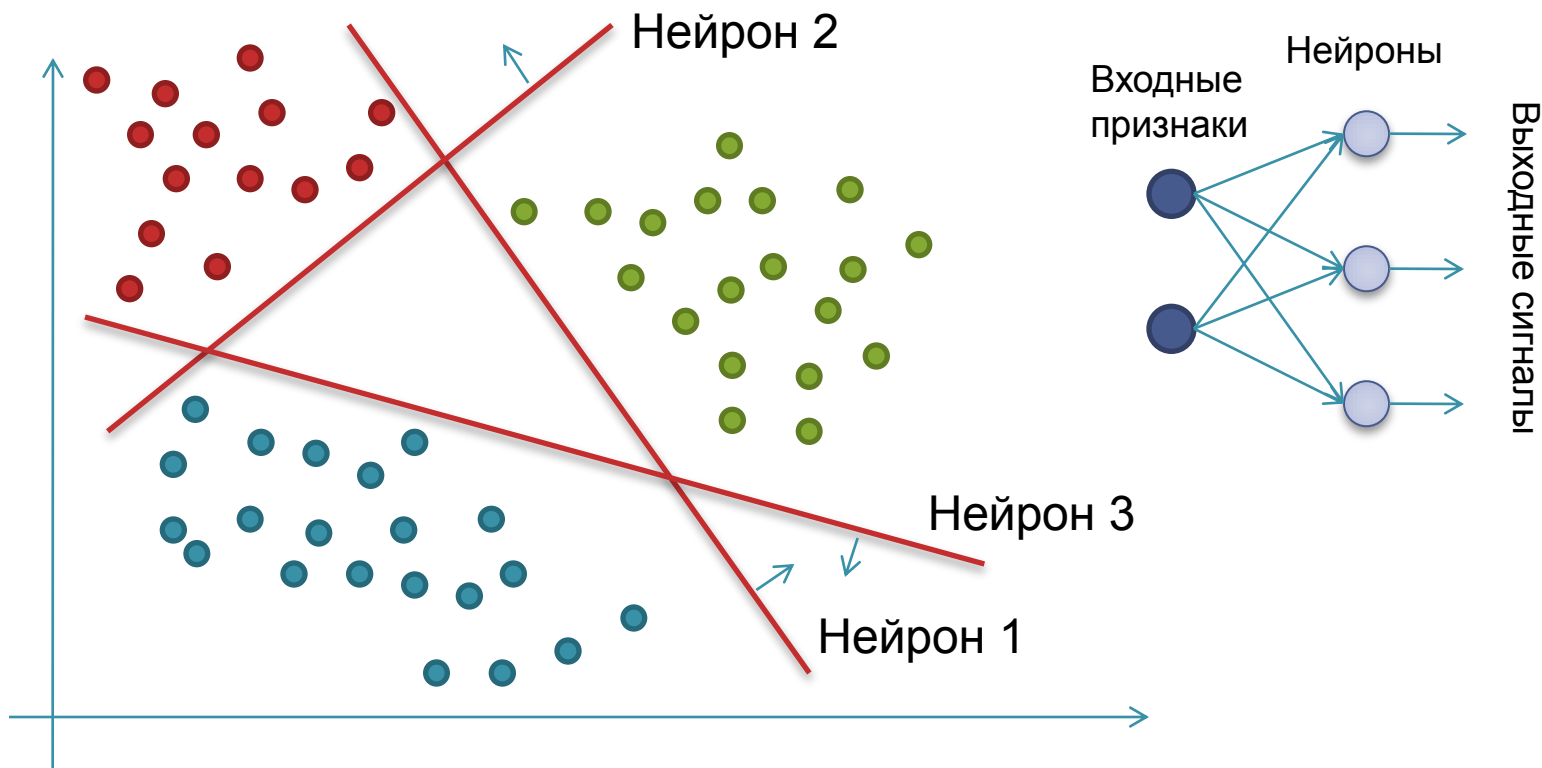
Итак, один нейрон может определить:

1. Где относительно разделяющей функции находится образ в пространстве признаков (выше/ниже, дальше или ближе к границе).
2. Либо как далеко находится образ от «центра» нейрона.

Если нейронов несколько, то:

Пространство входных признаков разбивается на области (обычно пересекающиеся) и каждый нейрон «работает» со своей областью (окрестность, либо полуплоскость).

От нейронов к сетям



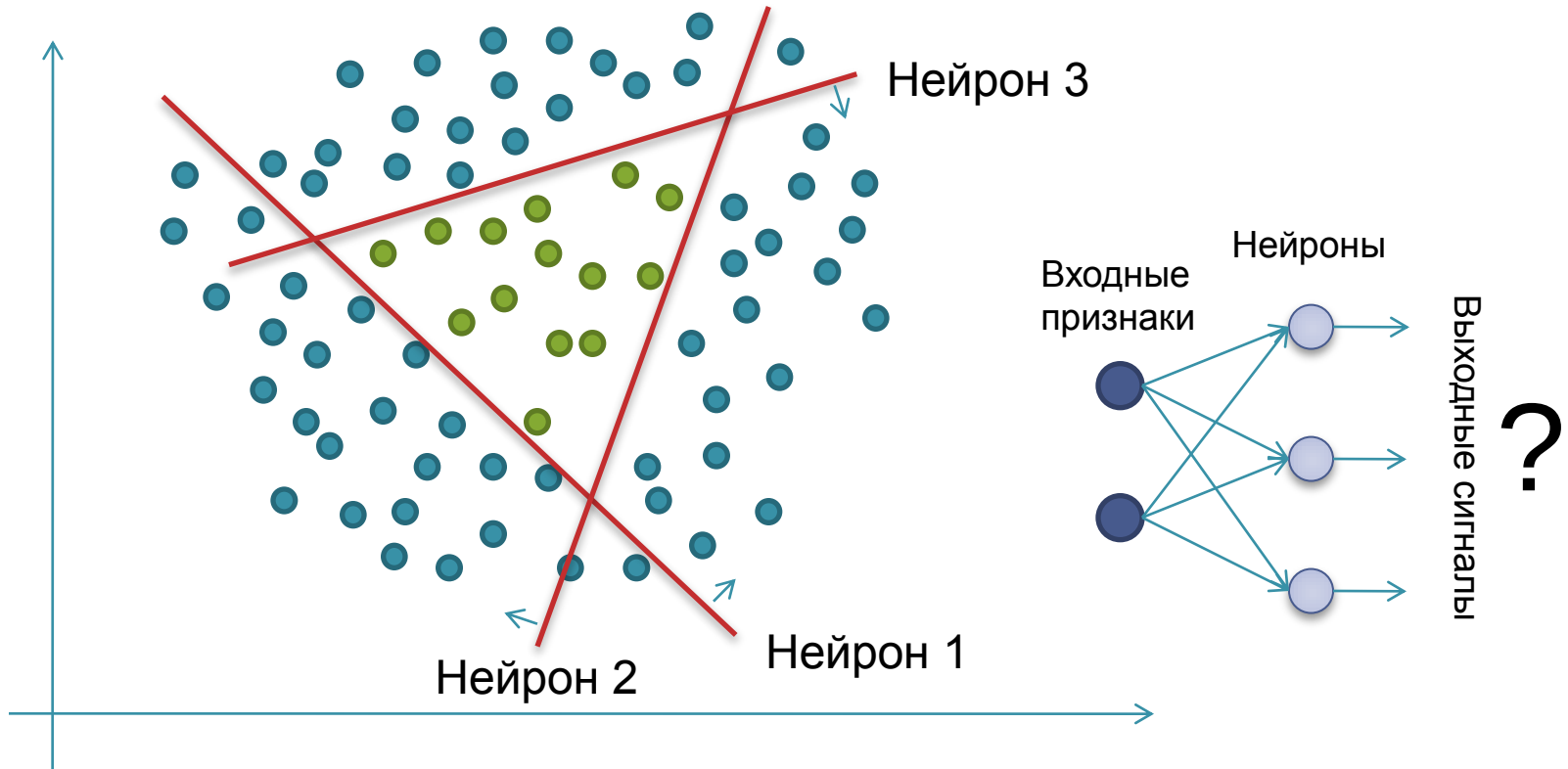
Нейрон 1 «узнает» класс «Зеленый»

Нейрон 2 «узнает» класс «Красный»

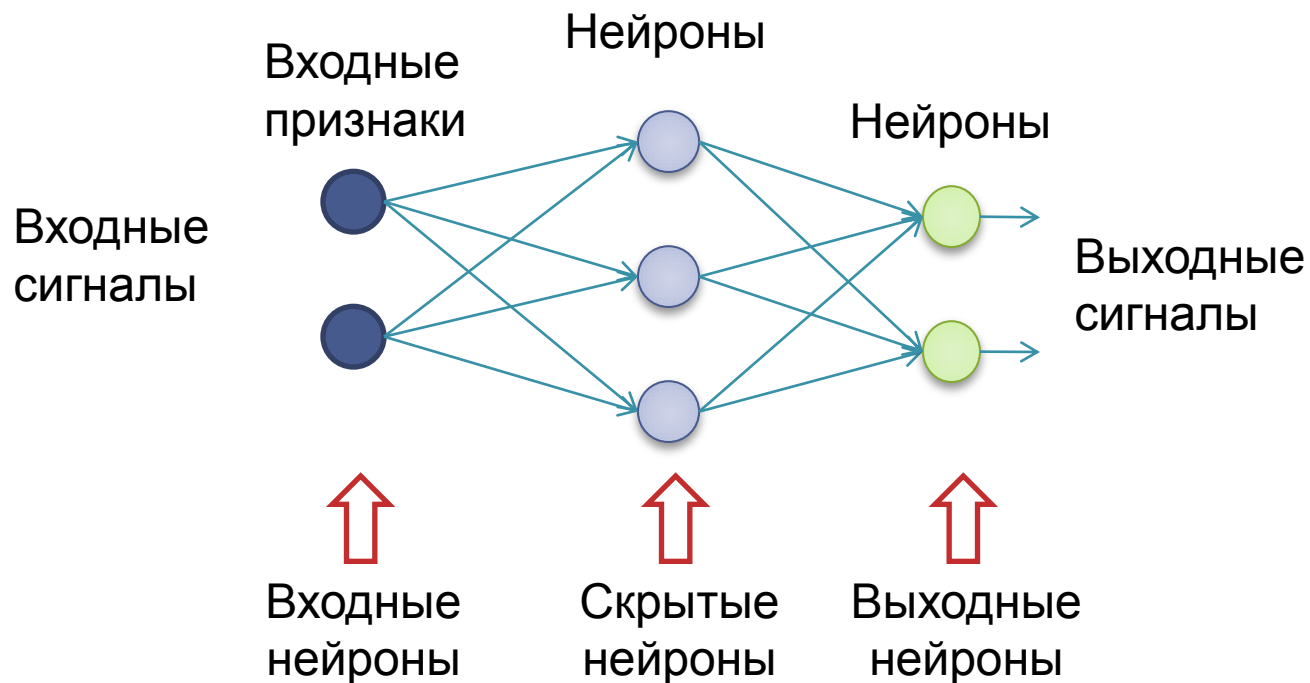
Нейрон 3 «узнает» класс «Синий»

Можно ограничиться даже и 2-мя нейронами.

От нейронов к сетям



От нейронов к сетям



От нейронов к сетям

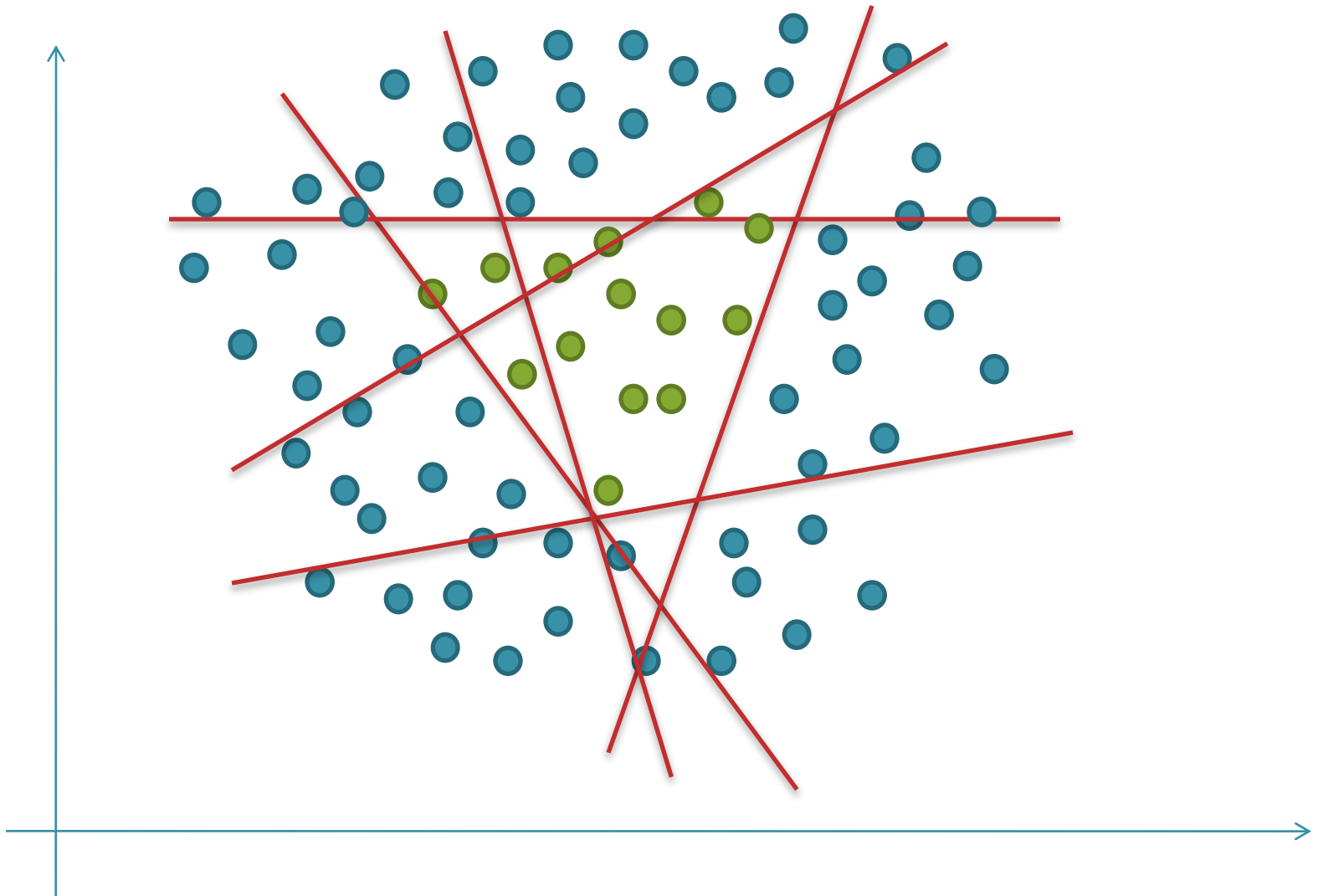
Общая суть:

1. Нейроны первого скрытого слоя производят преобразование входных сигналов (входных признаков) и передают результат нейронам следующего слоя (в примере выходного).
2. Нейроны выходного слоя обрабатывают сигналы нейронов скрытого слоя и выдают результат: вектор выходных сигналов ИНС.

Многослойная ИНС

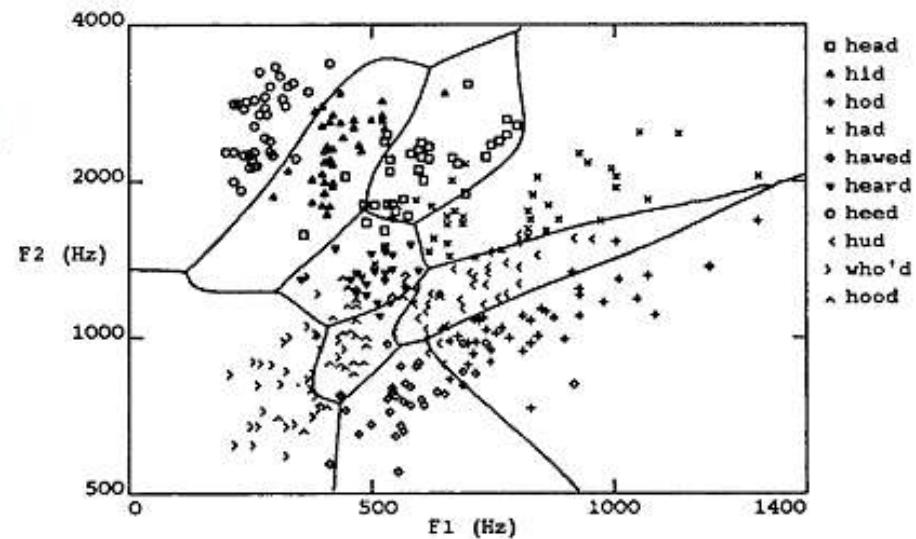
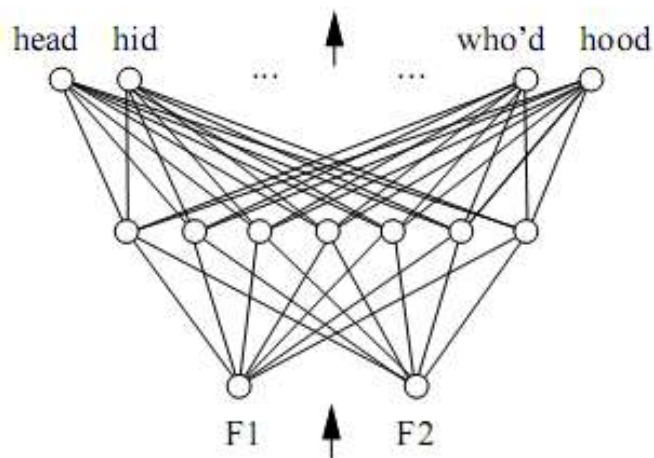
Слоев может быть сколько угодно. Обычно 1-3.

От нейронов к сетям



От нейронов к сетям

Пример: Распознавание речи



Источник: Lecture slides for text book *Machine Learning*, T. Mitchell, McGraw Hill, 1997, p. 88

От нейронов к сетям

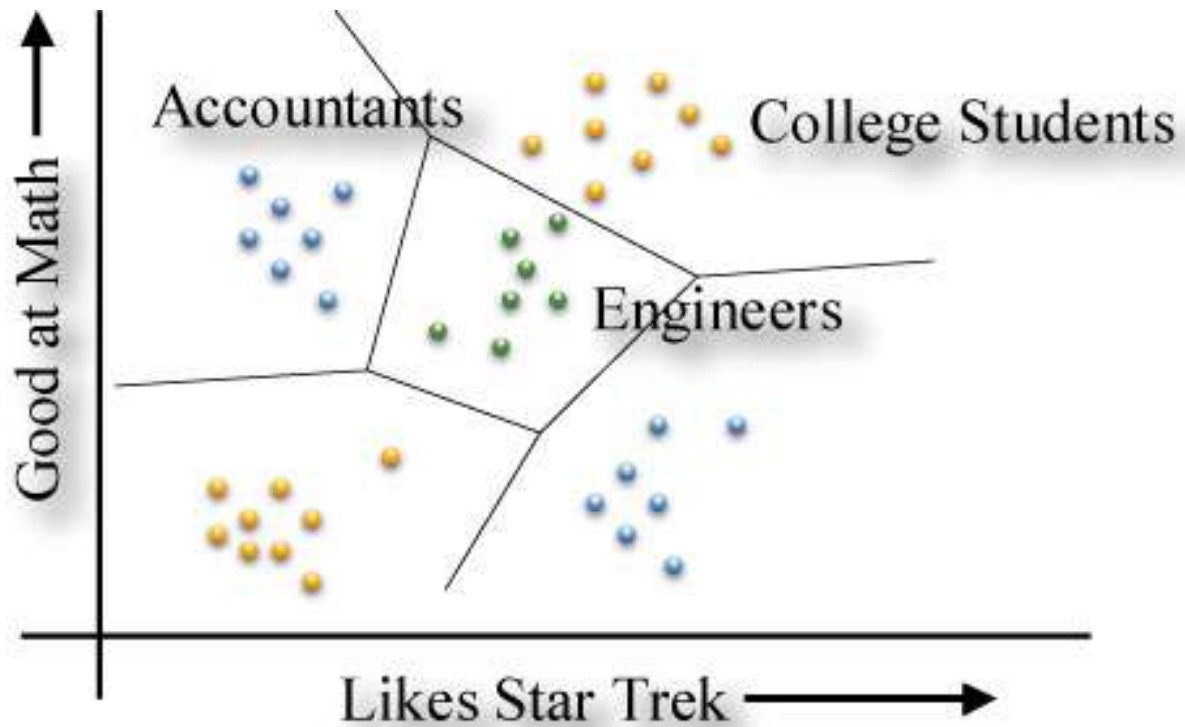


Рисунок: Wikiversity (Neural-Network-Example-5.jpg)

Обучение ИНС



$$F = F(X, W)$$

$$E = |F(X, W) - Y(X)|^2$$

$$E \rightarrow \min$$

Если функции активации нейронов представляют непрерывную дифференцируемую функцию, то НС преобразование также соответствует непрерывной дифференцируемой функции. Минимум E достигается с использованием *любого* известного метода оптимизации.

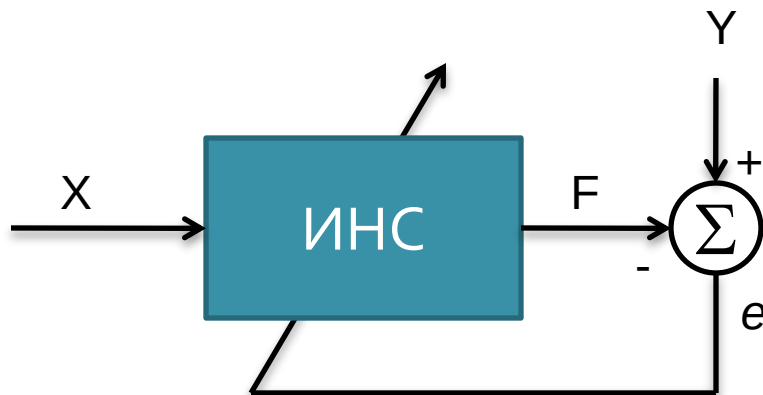


Bernard
Widrow



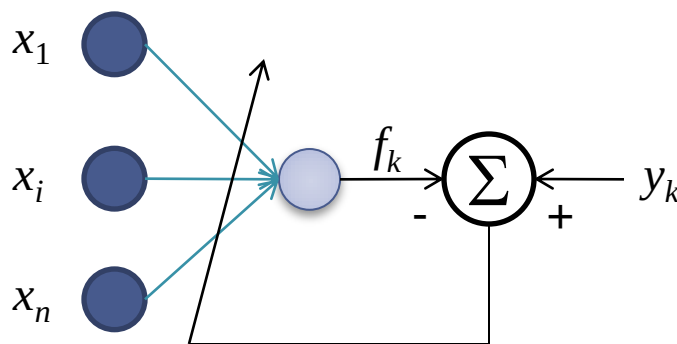
Marcian
"Ted" Hoff

Обучение ИНС. Правило Уидроу-Хоффа (дельта-правило)



$$w_{ik}(t+1) = w_{ik}(t) + \Delta w_{ik}$$

$$\Delta w_{ik} = \eta e_k x_i = \eta (y_k - f_k) x_i$$



Ссылка на статью

Arthur E.
Bryson



Yu-Chi Ho



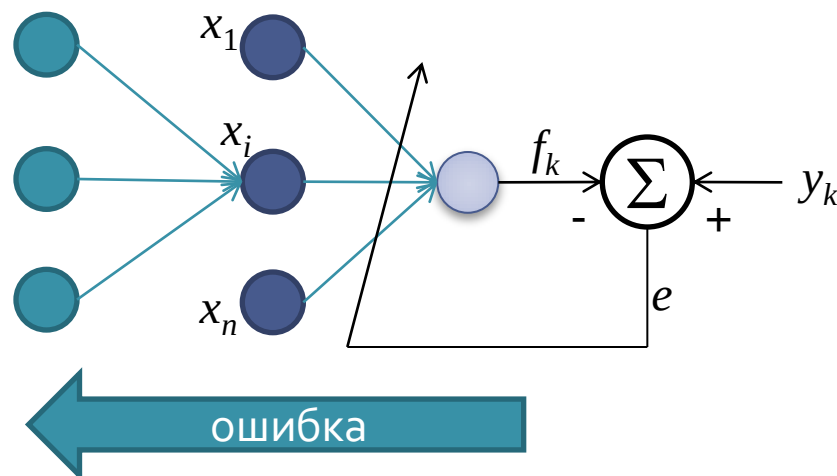
Geoffrey
Hinton



David
Rumelhart

Ronald
J. Williams

Обучение ИНС. Схема обратного распространения ошибки



$$e_k = f'_k(y_k - f_k)$$

$$e_i^{(q)} = -f'_i \sum_j w_{i,j} e_j^{(q+1)}$$

$$w_{ik}(t+1) = w_{ik}(t) + \Delta w_{ik}$$

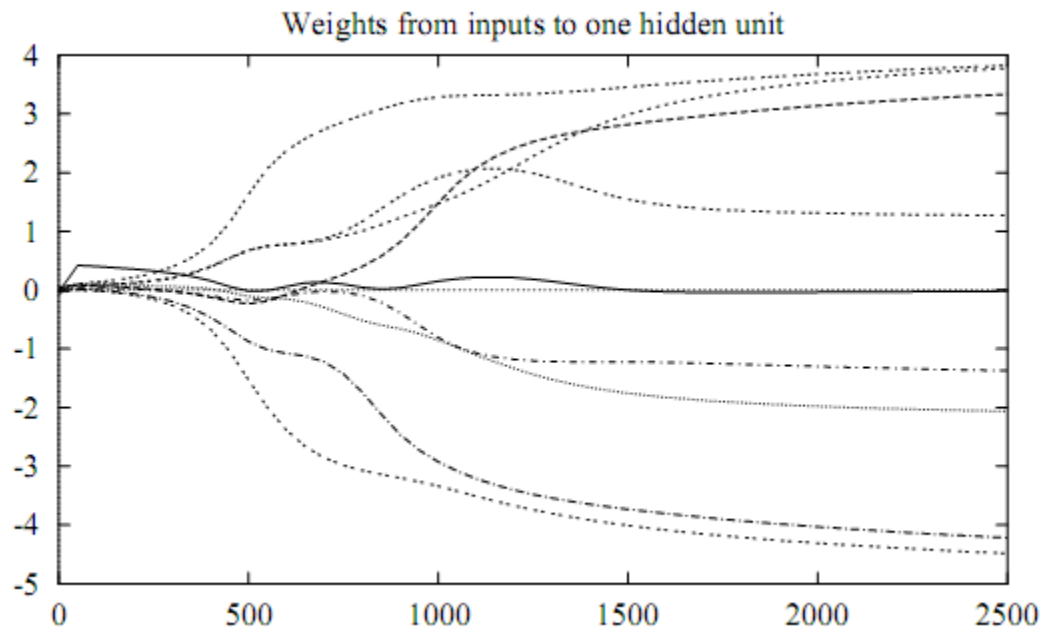
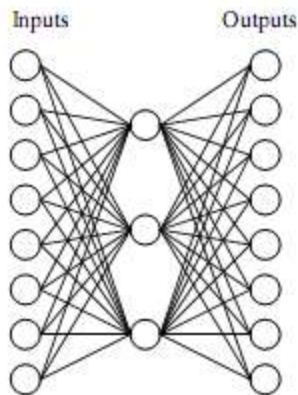
$$\Delta w_{ik} = \eta e_k x_i$$

Bryson A.E., Ho Y.-C. (1969). Applied optimal control: optimization, estimation, and control. Blaisdell Publishing Company or Xerox College Publishing. pp. 481.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning internal representations by error propagation. / In Rumelhart, D. E. and McClelland, J. L., editors, Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume I: Foundations Volume I: Foundations, MIT Press, Cambridge, MA.

Обучение ИНС

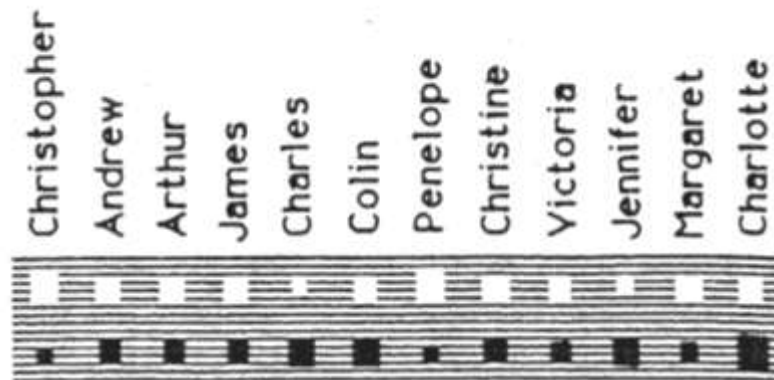
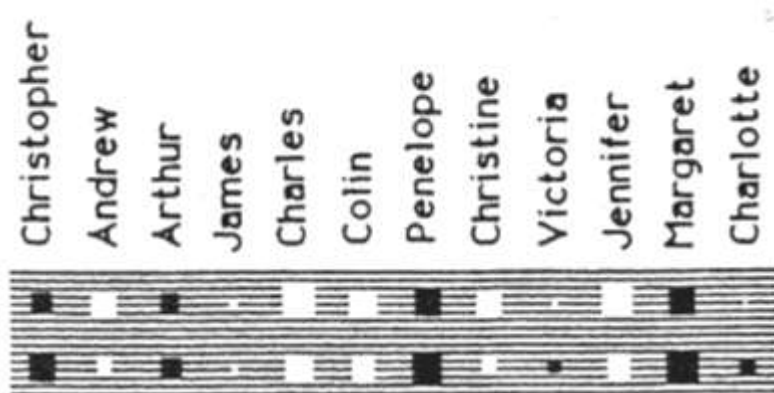
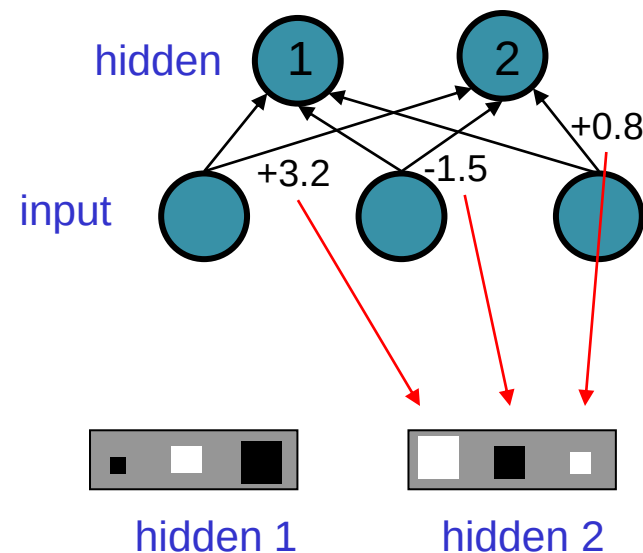
Пример: ИНС для сжатия информации

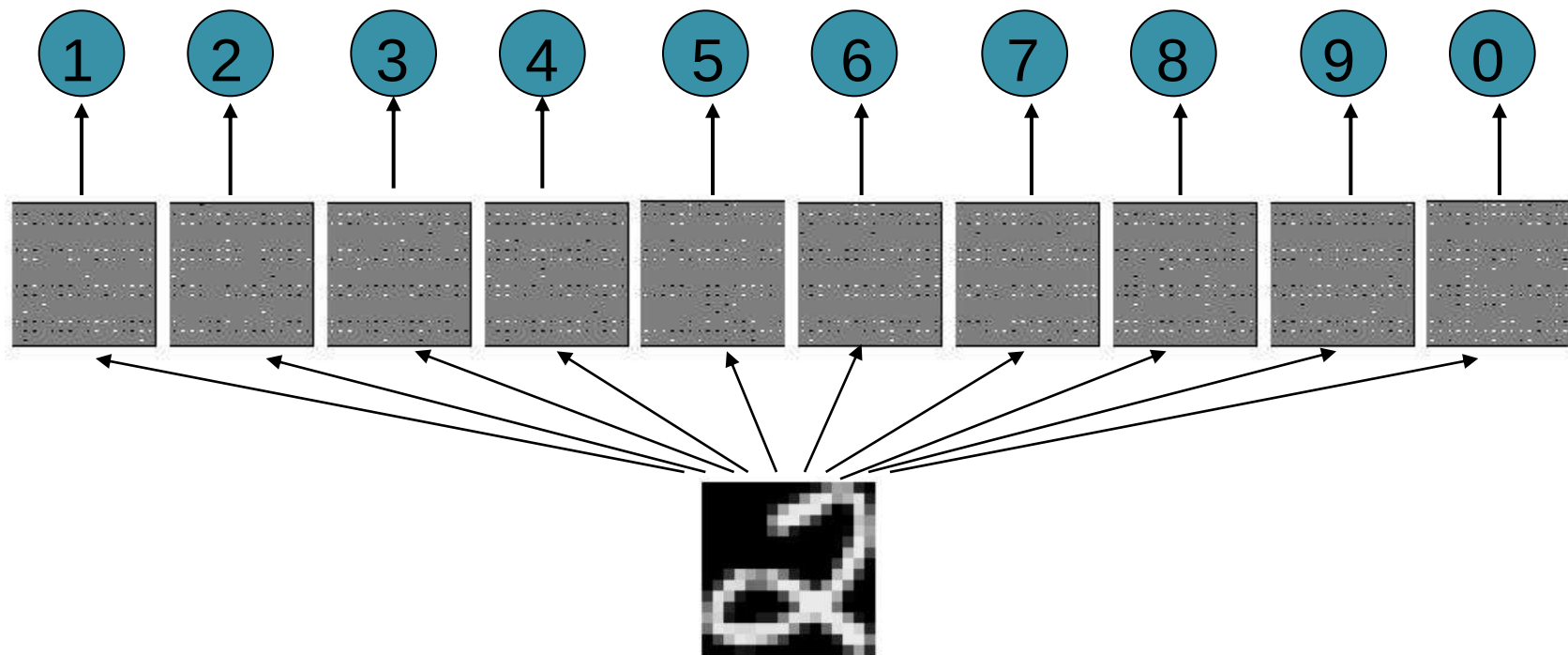


Источник: Lecture slides for text book *Machine Learning*, T. Mitchell, McGraw Hill, 1997, p. 88

Обучение ИНС

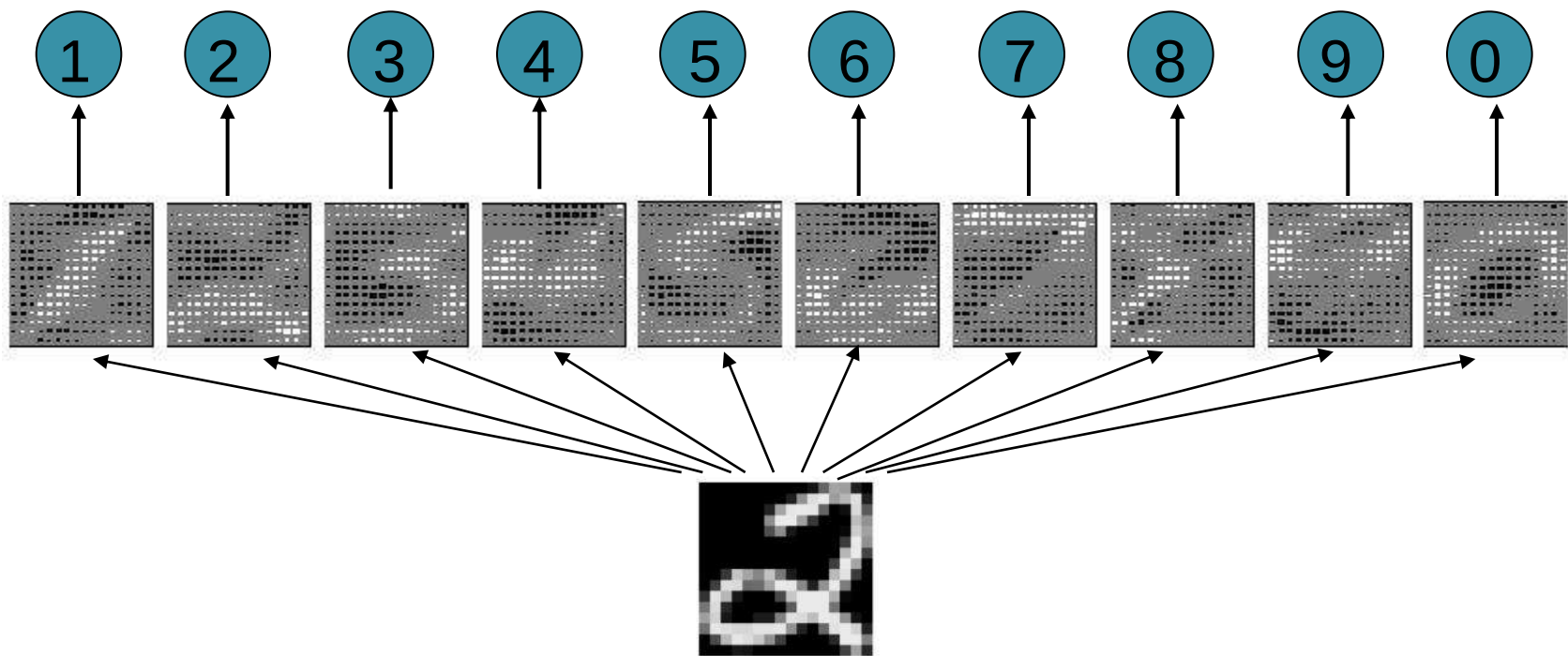
Визуализация результата





Будем на каждом шаге увеличивать веса связей от активных пикселей к «правильному» выходному нейрону, и уменьшать веса от активных пикселей к нейрону-победителю.

Источник: Hinton G. Can computer simulations of the brain allow us to see into the mind?



Источник: Hinton G. Can computer simulations of the brain allow us to see into the mind?

Обучение ИНС

Примеры успешного применения обратного распространения ошибки:

- Распознавание рукописных букв
- Прогнозирование динамики рынка ценных бумаг
- Распознавание мошенничества по кредитным картам
- Распознавание речи
- Прогнозирование следующего слова в предложении по предыдущим

Источник: Hinton G. Lecture 4: Backpropagation.

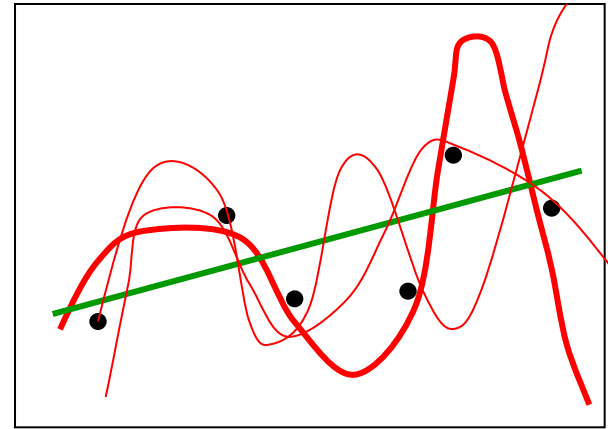
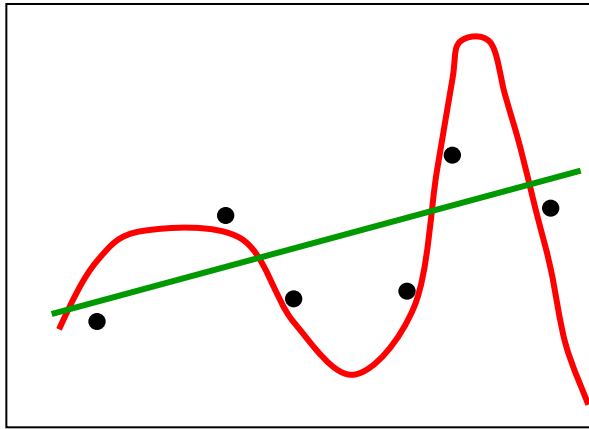
www.cs.toronto.edu/~hinton

Обучение ИНС. Проблемы обратного распространения

- Требуются помеченные обучающие данные.
 - Большинство данных не помечены.
- Нужно настроить примерно 10^{14} весов связей за всего лишь 10^9 секунд.
 - Метки не обеспечивают необходимой информации, если только веса не являются лишними.
- Время обучения плохо масштабируется
 - Обучение занимает много времени, если количество скрытых слоев больше 2 или 3.
- Нейроны должны посылать два различных по типу сигнала
 - Прямой проход: $\text{signal} = \text{activity} = y$
 - Обратный проход: $\text{signal} = dE/dy$

Источник: Hinton G. Can computer simulations of the brain allow us to see into the mind?

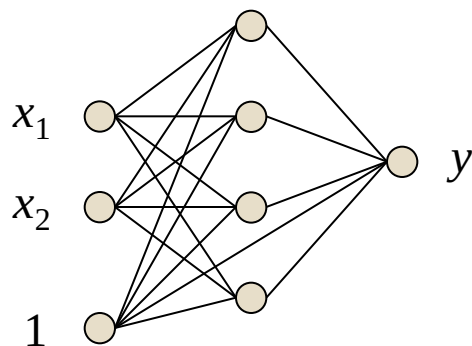
Обучение ИНС. Проблема выбора структуры



Хинтон: Нет никаких оснований полагать, что объем начальных данных должен влиять на сложность модели

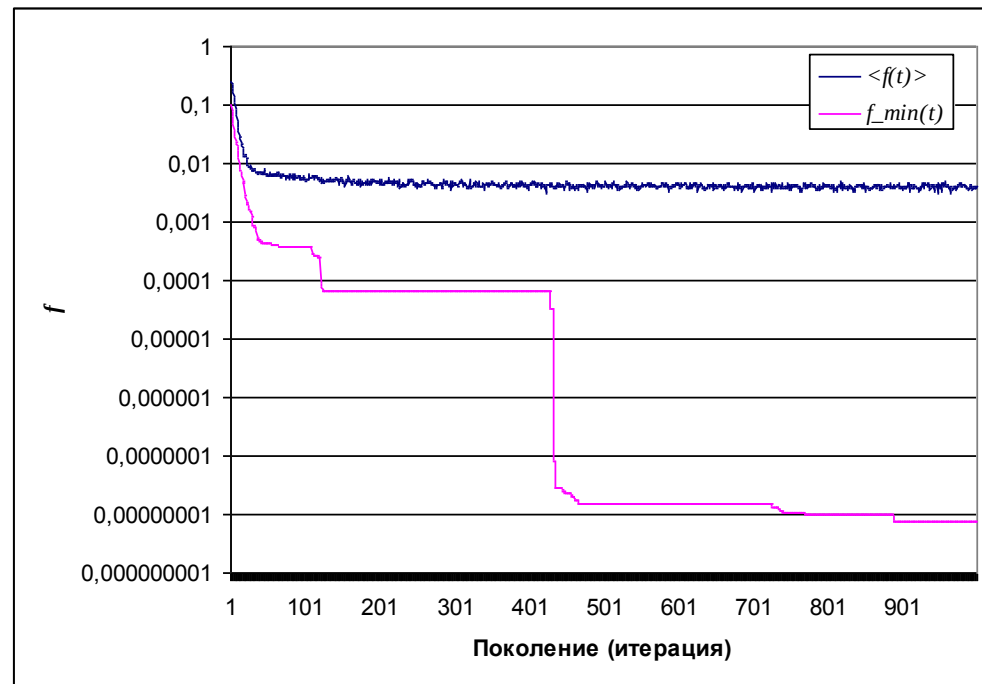
Источник: Hinton G. Lecture 3: Ways to make backpropagation generalize better, and ways to do without a supervision signal

Обучение ИНС. Нейроэволюционный подход



Количество связей: 17
Возможные решения
представлены 17-
мерными векторами:
 $\{w_1, \dots, w_{17}\}$

Пример: Иключающее ИЛИ

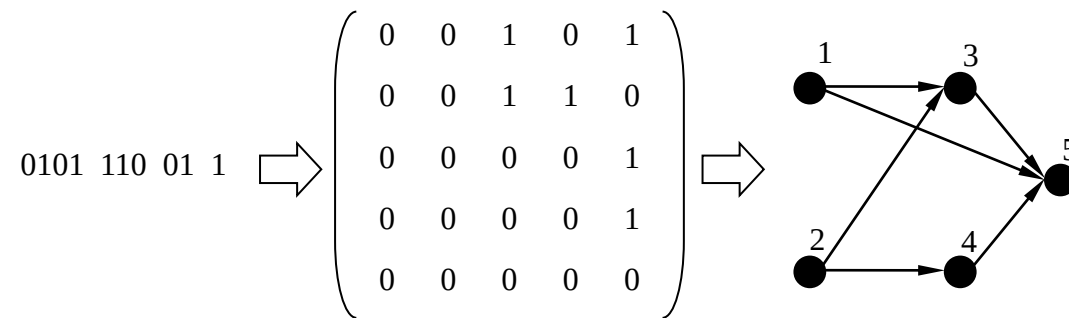


* Усреднено по 100 запускам

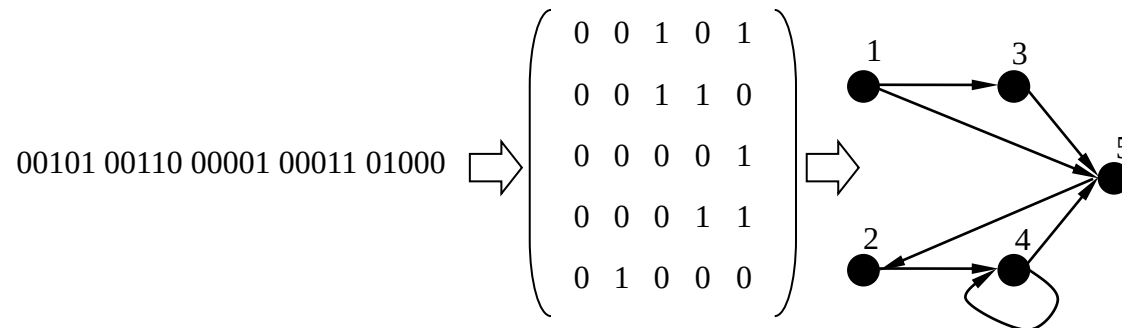
Обучение ИНС.

Нейроэволюционный подход

Пример прямого кодирования: Матрица смежности



а) ИНС без обратных связей



б) ИНС с обратными связями

Обучение ИНС. Нейроэволюционный подход

Пример косвенного
кодирования: Эволюция
грамматических правил

Kitano H. Designing neural
network using genetic
algorithm with graph
generation system // Complex
Systems, 1990, no. 4, pp. 461–
476:

Эволюционный поиск
направлен на генерацию
правил, применяемых на
первом и втором шагах.

ABCDaaaaiaiaacaeae

а)

$$S \rightarrow \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

$$A \rightarrow \begin{pmatrix} a & a \\ a & a \end{pmatrix}$$

$$B \rightarrow \begin{pmatrix} i & i \\ i & a \end{pmatrix}$$

$$C \rightarrow \begin{pmatrix} i & a \\ a & c \end{pmatrix}$$

$$D \rightarrow \begin{pmatrix} a & e \\ a & e \end{pmatrix}$$

$$a \rightarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$c \rightarrow \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

$$e \rightarrow \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

$$i \rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

б)

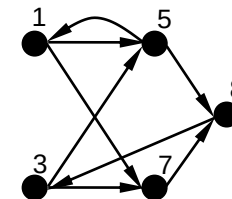
1 шаг
 $S \Rightarrow \begin{pmatrix} A & B \\ C & D \end{pmatrix}$

2 шаг
 $\Rightarrow \begin{pmatrix} a & a & i & i \\ a & a & i & a \\ i & a & a & e \\ a & c & a & e \end{pmatrix}$

3 шаг
 $\Rightarrow$

0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0

в)



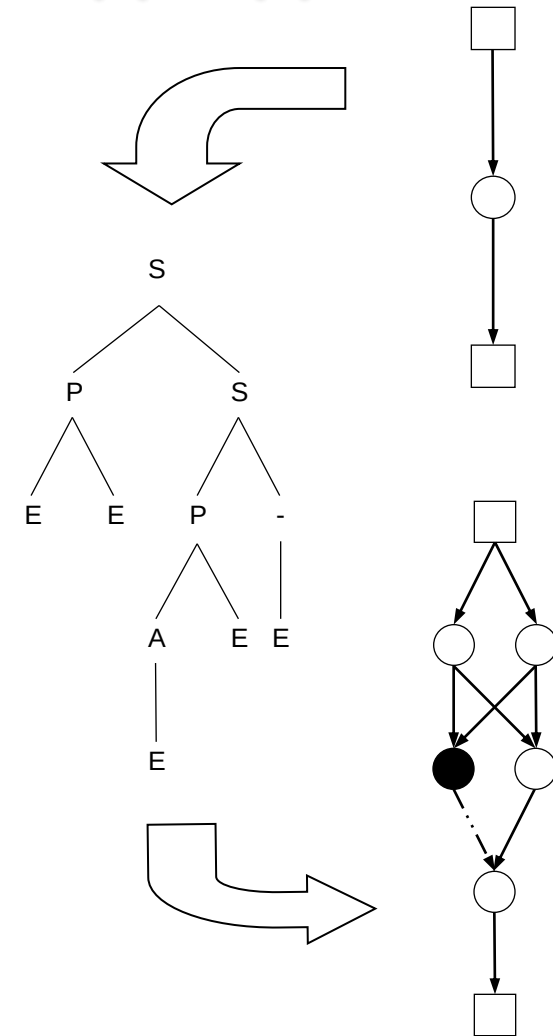
г)

Обучение ИНС. Нейроэволюционный подход

Пример косвенного кодирования: Эволюция последовательности использования правил (*клеточное кодирование, cellular encoding*).

1. Задан набор правил.
2. Эволюционный поиск направлен на генерацию последовательности применения правил, приводящей к желаемому результату.

Gruau F. Genetic synthesis of Boolean neural networks with a cell rewriting developmental process // In Proceedings of the International Workshop on Combination of Genetic Algorithms and Neural Networks (COGANN-92). Los Alamos, CA: IEEE Computer Society Press, 1992. P. 55-74.

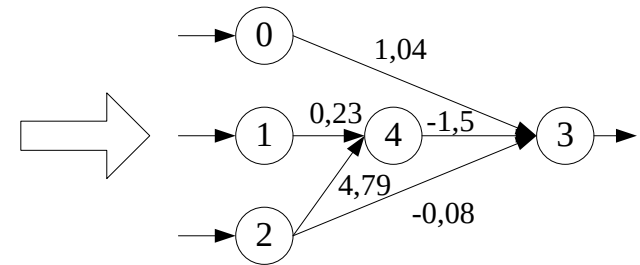


Обучение ИНС. Нейроэволюционный подход

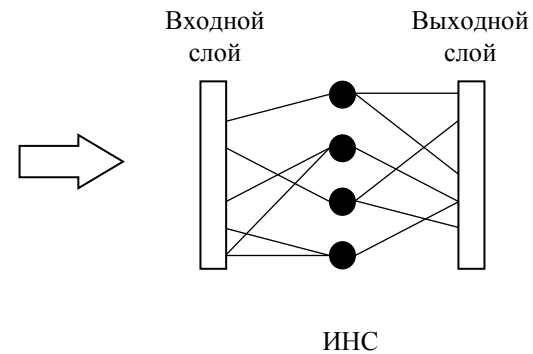
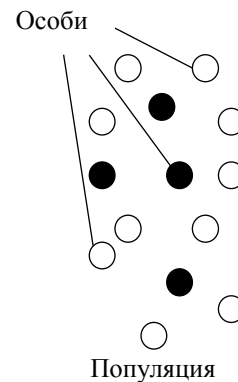
Прямое
кодирование

1	4	4	3	2	3	2	4	0	3
0,23	-1,5	-0,08	4,79	1,04					

Начальный нейрон связи	Конечный нейрон связи
Вес связи	



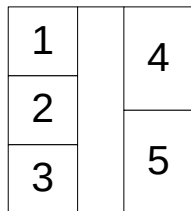
Коэволюционный
подход



Обучение ИНС.

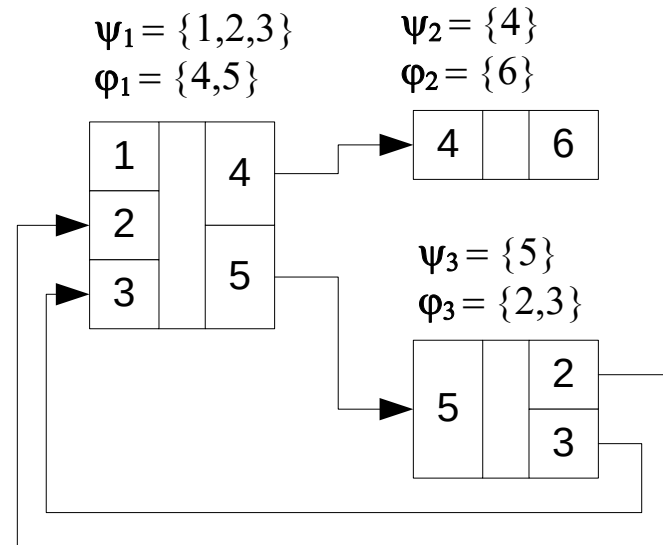
Нейроэволюционный подход

Информация о структуре связей не хранится –
неявное связывание узлов.



$\psi = \{1, 2, 3\}$
 $\phi = \{4, 5\}$

узел



сеть

Возможность получить сеть с большим количеством узлов и связей за относительно небольшое число итераций



Donald
O. Hebb

Обучение ИНС. Правило Хебба.

$$\Delta w_{ij} = \eta y_i y_j$$

Hebb D.O. – The Organization of Behavior.
John Wiley & Sons, New York, 1949

1. Зависимость от времени активации нейронов
2. Локальность
3. Интерактивность
4. Корреляция

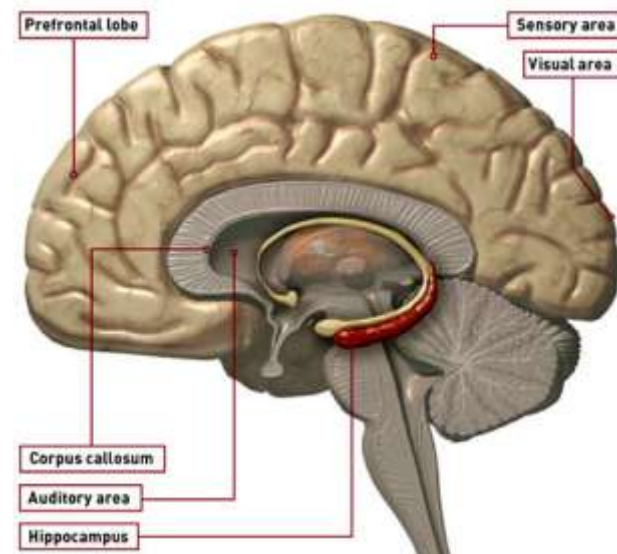


Рисунок:
<http://physiology-physics.blogspot.com/2008/09/binding-of-memory-and-hebbian-learning.html>



Erkki Oja

Обучение ИНС. Правило Ойя.

$$\Delta w_{ij} = \eta y_j (y_i - y_j w_{ij})$$

Oja E. Simplified neuron model as a principal component analyzer // Journal of Mathematical Biology, 1982, vol. 15, no. 3, pp. 267–273

При стабилизации весов связей нейрона, обучаемого по правилу Ойя, эти веса представляют компоненты **главного собственного вектора** матрицы ковариаций вектора входных сигналов.

Массив параллельно связанных нейронов, обучаемых по правилу Ойя, способен «выучить» все собственные векторы.

Выучивание собственных векторов означает, что возможно эффективное **уменьшение размерности** входного вектора, а слой нейронов, обучаемых по правилу Ойя, таким образом выучивает способ **кодировки** входных данных.



Teuvo
Kohonen

Карта Кохонена

- Нейроны находятся в узлах двумерной сетки.
- Векторное квантование.
- Конкуренция нейронов.
- Принципы WTA (Winner Takes All) и WTM (Winner Takes Most).

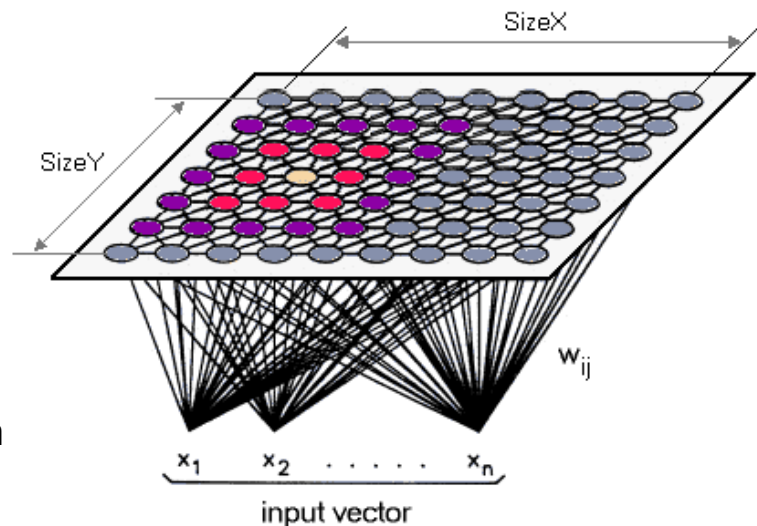


Рисунок:

http://www.lohninger.com/helpcsuite/kohonen_network_-_background_information.htm

Карты Кохонена

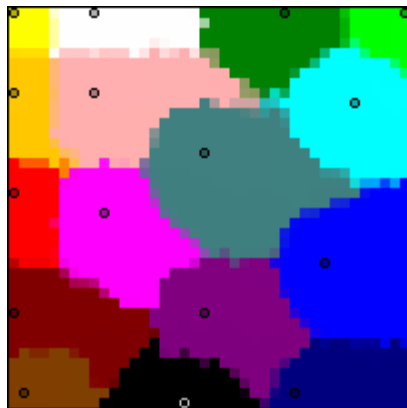
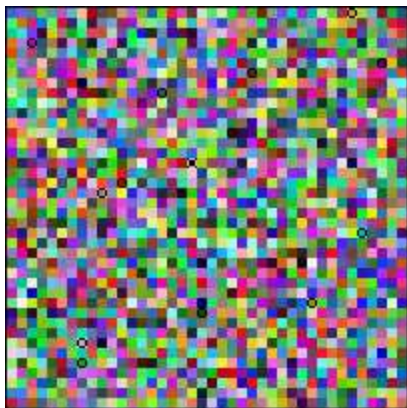


Рисунок (и Java applet):
<http://www.generation5.org/content/2004/kohonenApplications.asp?Print=1>

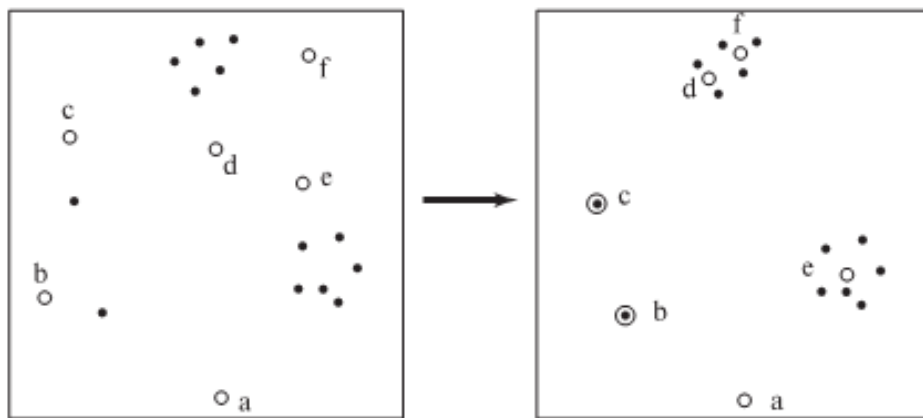


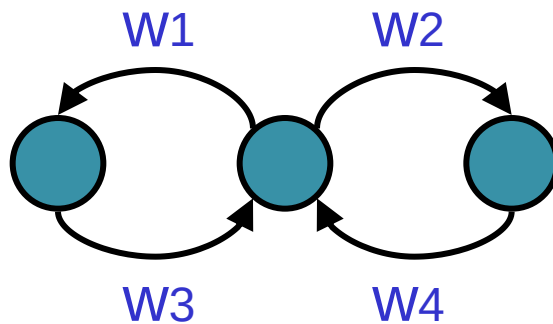
Рисунок: Заенцев И.В. Нейронные сети: основные модели.
Учебное пособие к курсу «Нейронные сети». Воронеж, 1999

ИНС с обратными связями

- Многослойная ИНС с обратными связями (RMLP – Recurrent MultiLayer Perceptron). Алгоритм обратного распространения для рекуррентной ИНС.
- Сеть Хопфилда и ее модификации
- Echo State Network
- Другие модели (здесь не рассматриваем)

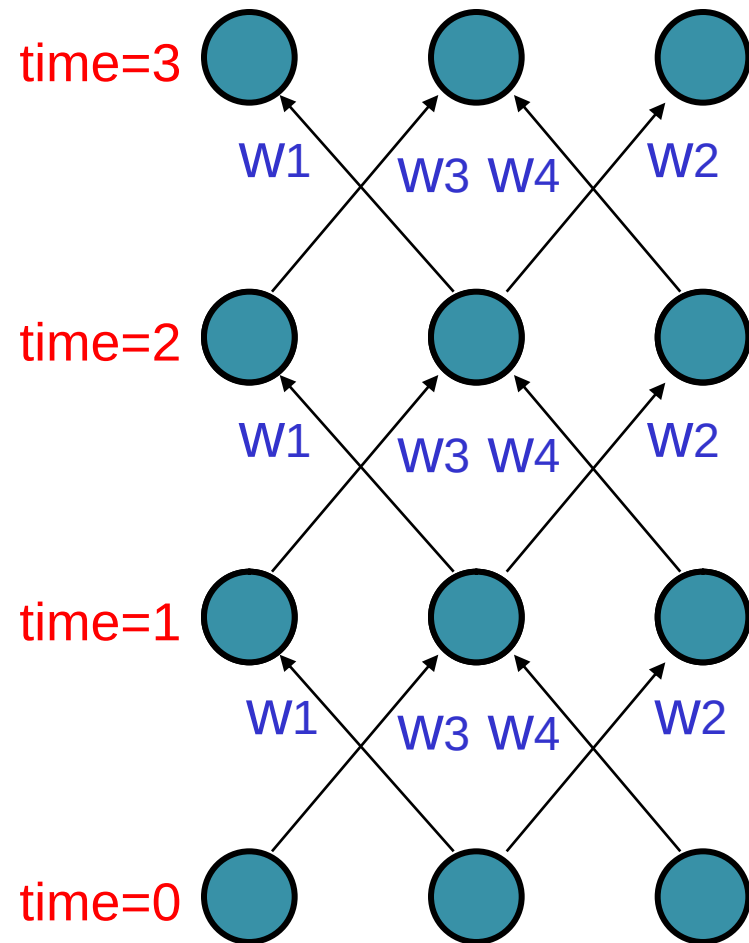


Рекуррентная сеть. Эквивалентность с многослойными сетями



Для каждой связи предполагается единичная задержка.

Таким образом, рекуррентная сеть – это многослойная сеть, которая повторно использует одни и те же веса.



Теорема Коэна-Гроссберга

Системы, описываемые: $\frac{d}{dt}u_j = a_j(u_j) \left[b_j(u_j) - \sum_{i=1}^N w_{ij} \varphi_i(u_i) \right]$

1. Условие симметрии.

$$w_{ij} = w_{ji}$$

2. Условие неотрицательности. Значение функции активации $a_j(u_j)$ должно быть неотрицательным.

3. Условие монотонности. Функция $\varphi_j(u_j)$ отображения входа на выход должна быть монотонной.

При выполнении этих условий функция Ляпунова системы описывающей динамику полносвязной ИНС удовлетворяет условию

$$\frac{dE}{dt} \leq 0$$

Cohen M.A., Grossberg S. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks // IEEE Transactions on Systems, Man and Cybernetics, 1983, vol. SMC-13, pp. 815-821.

Сеть Хопфилда

Используется в качестве автоассоциативной памяти и для задач фильтрации и оптимизации.

Матрица весов связи **симметричная и имеет нули на главной диагонали**.

Обученная сеть в процессе функционирования, **сходится** к состоянию соответствующему **локальному минимуму энергии** сети.

$$E = -\frac{1}{2} \sum_{i,j} w_{ij} s_i s_j + \sum_i \theta_i s_i$$

Емкость сети:

$$M = \frac{N}{2 \ln N} \approx 0,14N$$

Емкость доменной сети Хопфилда:

$$M = \frac{kN}{2 \ln N}$$

B.V.Kryzhanovsky, B.M.Magomedov, A.L.Mikaelian.
A Domaine Model of Neural Network. Doklady
Mathematics, vol.71, N2, pp.310-314 (2005).



John Hopfield

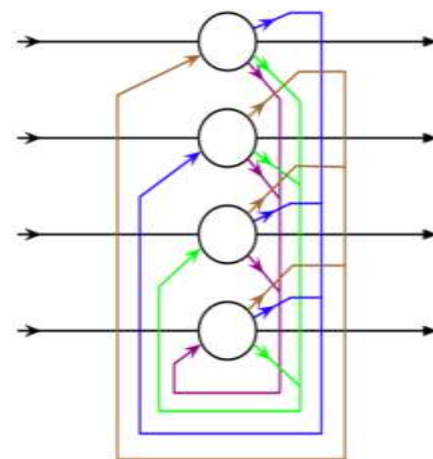


Рисунок из Wikipedia
(Hopfield-net.png)

Сеть Хопфилда

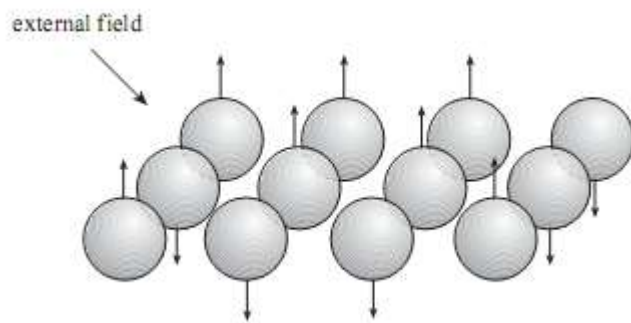
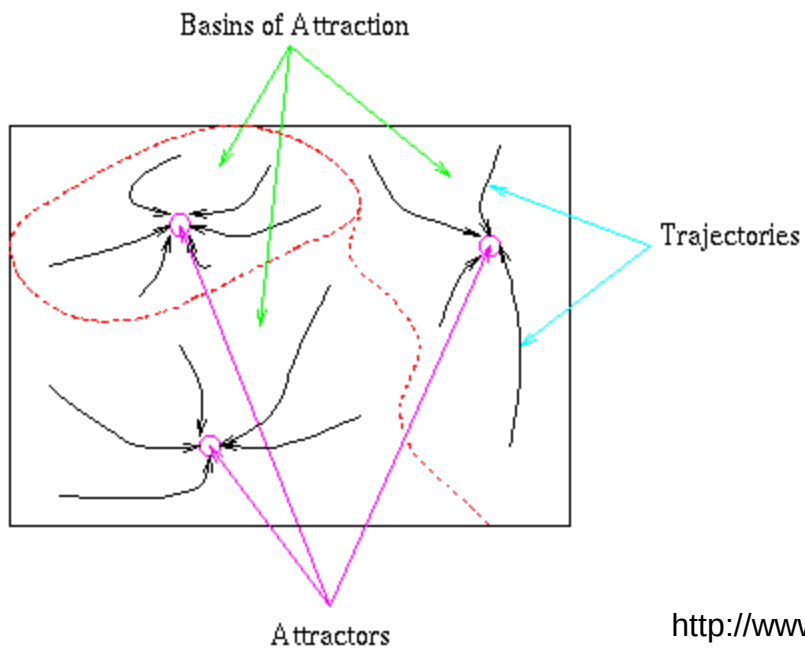


Рисунок из: Rojas R. Neural Networks, Berlin: Springer-Verlag, 1996



Сеть Хопфилда

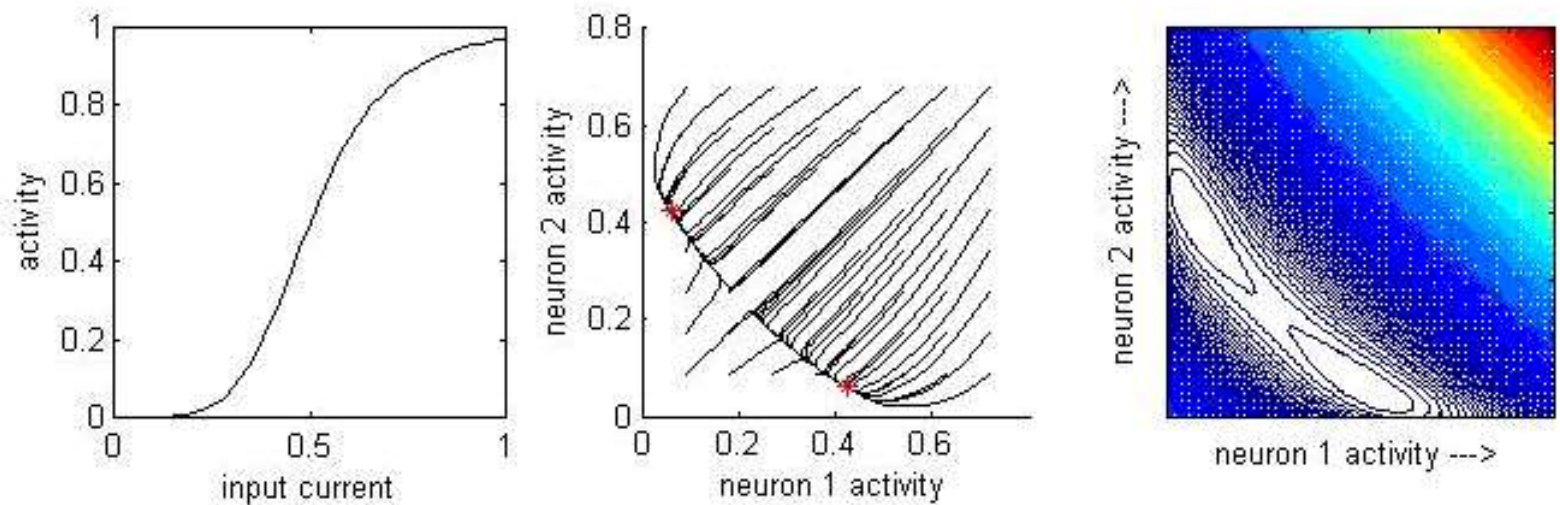
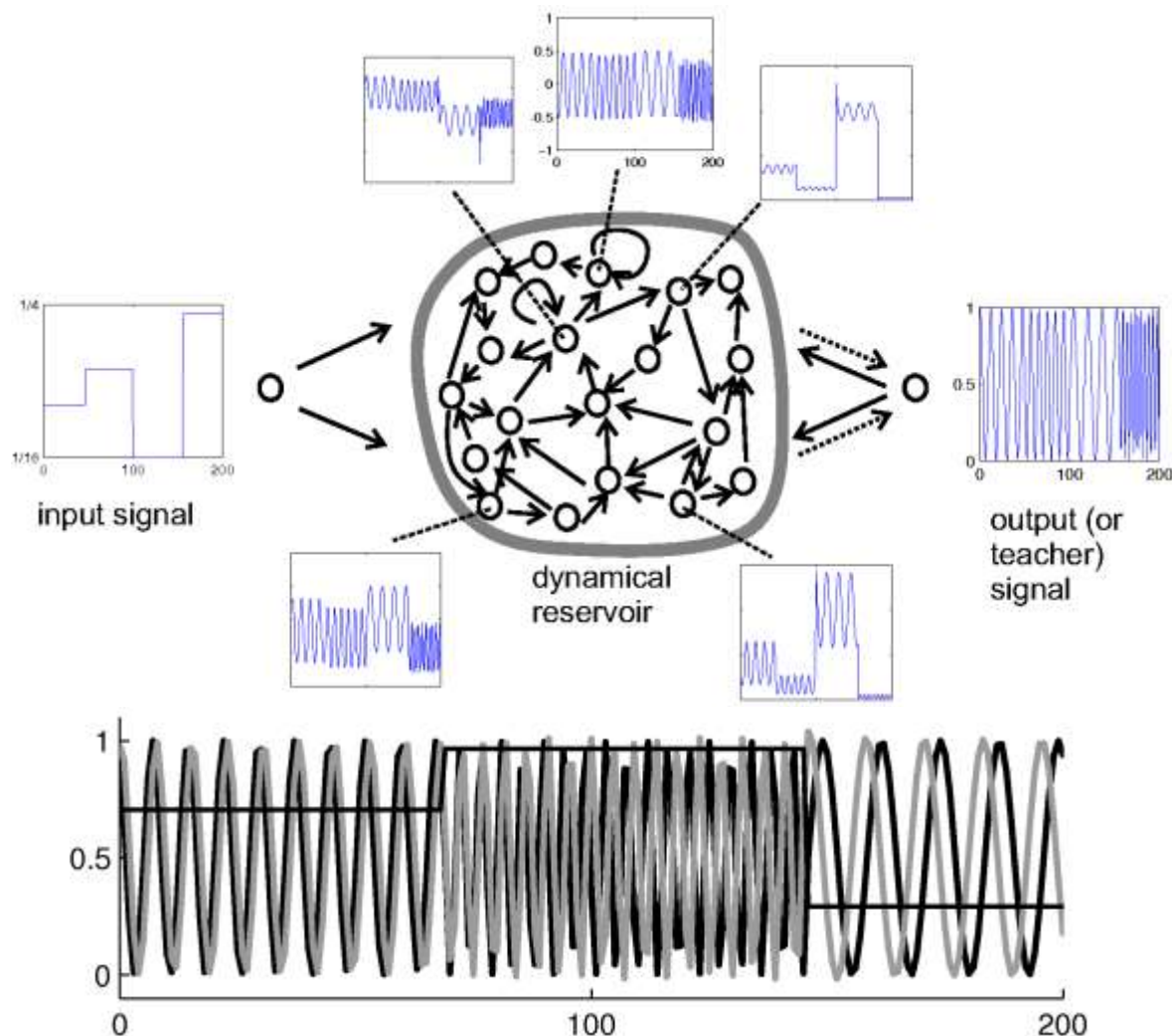


Рисунок из Scholarpedia (Hopfield_Net_fig2.png)



Herbert
Jaeger

Echo state network



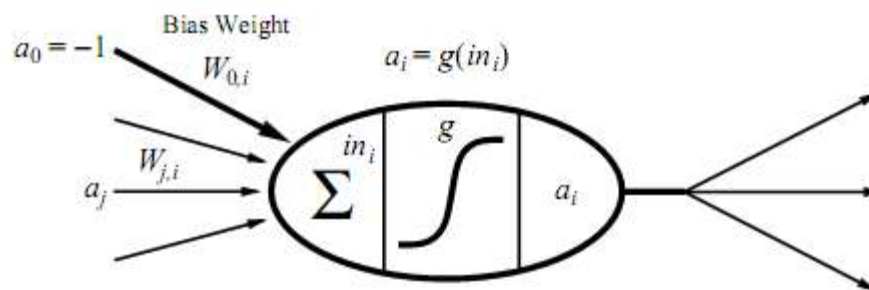
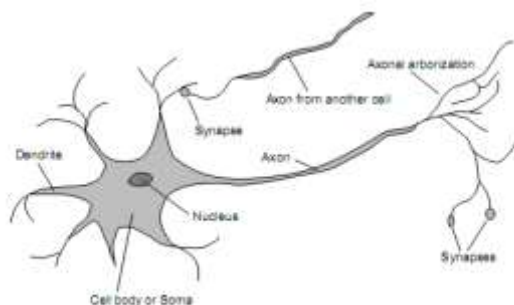
Рисунк: Scholarpedia (FreqGenSchema.png, FreqGenTestOverlay.png)

Echo state network

- ESN может моделировать произвольную динамическую систему (гарантировано для сигмоидных нейронов).
- Для независимых входных переменных с одинаковым законом распределения емкость кратковременной памяти ESN ограничена N .
- Существенно большая скорость обучения за счет отсутствия необходимости настройки всех весов связей.
- Результаты обучения не подвержены влиянию бифуркаций (в отличие от традиционных подходов)

Но начиналось все по-другому...

- Работа Маккалоха и Питтса (1943).
- Эра перцептрона (60-е)
- Эра ассоциативной памяти (70-е)
- Эра обратного распространения (80-е и начало 90-х)

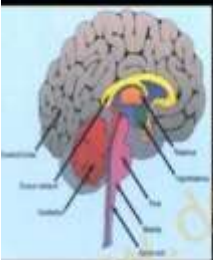


Биология

- Известно около 50 типов нейронов.
- Время переключения нейрона $\sim 1-10$ мс.
- Количество нейронов $\sim 10^{10-11}$
- Количество связей у одного нейрона $\sim 10^4-5$
- Время распознавания сцены $\sim 0,1$ с.
- Около 100 этапов. Основная работа выполняется за счет параллелизма.

What is the most complex information processing system in the universe?.....

AMD
Smarter Graphics

The human brain	Brain	GPU(48xx)
	Compute capability	100 TeraOps
	Memory bandwidth	1 TeraOp
	Power consumption	Terabits/sec
	Compute/Watt	30 W Max!
		150W
		~.007 Teraops/Watt

www.danhabar.com



Warren
Sturgis
McCulloch

Работа Маккалоха и Питтса

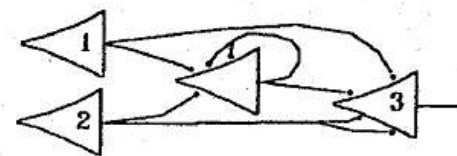
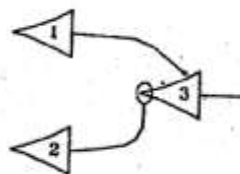
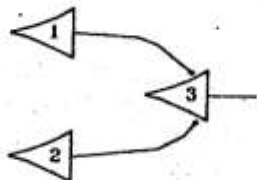
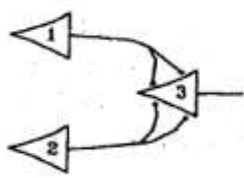
Маккалох Дж., Питтс У. Логические исчисления идей, относящихся к нервной деятельности. // Автоматы. М.: ИЛ, 1956.

McCulloch W., Pitts W. A Logical Calculus of the Ideas Immanent in Nervous Activity // Bulletin of Mathematical Biophysics, 1943, "Vol 5, pp. 115–133.



Walter
Pitts

«Легко показать, что, во-первых, каждая сеть [связанная с окружающей средой] может вычислять лишь такие числа, которые вычисляет машина Тьюринга; во-вторых, что каждое из последних чисел может быть вычислено такой сетью и что сети [изолированные от окружающей среды] с петлями могут вычислять некоторые из вычислимых чисел, но не все из них»





Андрей
Николаевич
Колмогоров

Теорема Колмогорова

Любая непрерывная функция n аргументов на единичном кубе $[0, 1]^n$ представима в виде суперпозиции непрерывных функций одного аргумента и операции сложения:

$$f(x^1, x^2, \dots, x^n) = \sum_{k=1}^{2n+1} h_k \left(\sum_{i=1}^n \varphi_{ik}(x^i) \right)$$

где h_k, φ_{ik} - непрерывные функции, причем φ_{ik} не зависят от выбора f .

Колмогоров А.Н. О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного // Доклады АН СССР, 1958, т. 114, № 5, с. 953–956.



George
Cybenko

Аппроксимация суперпозицией сигмоидных функций

Cybenko G. Approximation by superposition of sigmoidal function // Mathematics of Control, Signals, and Systems, 1989, no. 2, pp. 303-314.

Исследование условий, при которых суммы вида

$$G(x) = \sum_{j=1}^N \alpha_j \sigma(y_j^T x + \theta_j)$$

являются всюду плотными в пространстве непрерывных функций, определенных на n -мерном единичном кубе.

Результат: ИНС с одним скрытым слоем и произвольной сигмоидной функцией активации может с произвольной точностью аппроксимировать непрерывную функцию, при условии отсутствия ограничений на количество скрытых нейронов и значения весов связей.

Отсутствие конструктивности. Проклятие размерности.

Примеры применения ИНС



3-nearest-neighbor = 2.4% error

400-300-10 unit MLP = 1.6% error

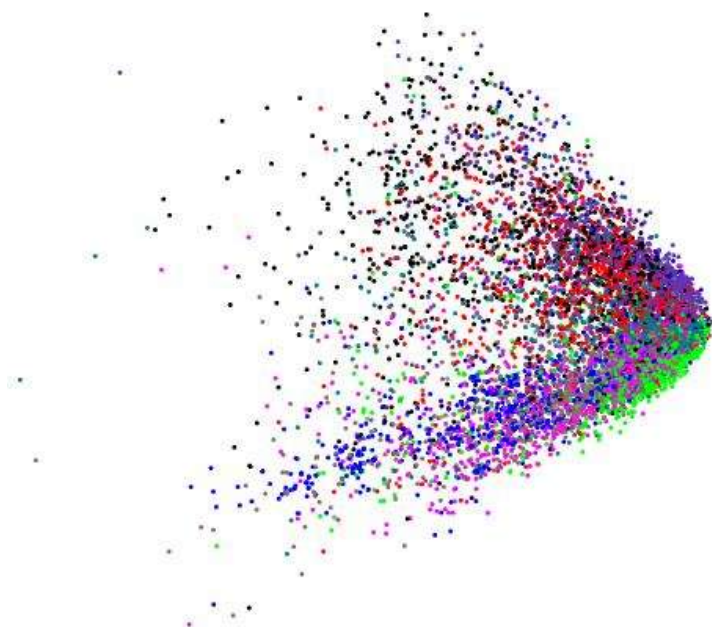
LeNet: 768-192-30-10 unit MLP = 0.9% error

Current best (kernel machines, vision algorithms) $\approx$ 0.6% error

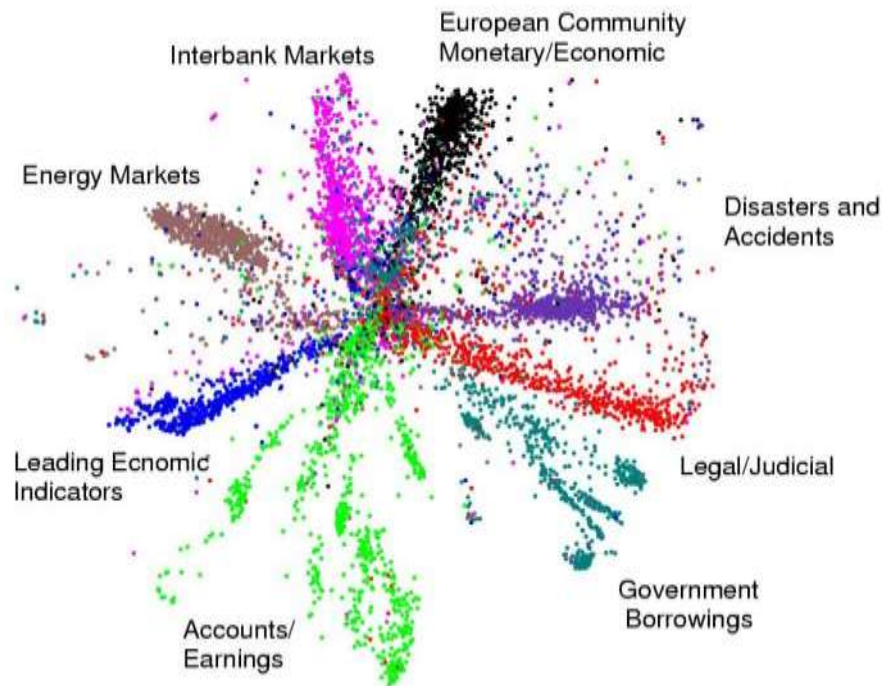
Источник: Russel S., Norvig P. Artificial Intelligence: A Modern Approach

Примеры применения ИНС

LSA 2-D Topic Space
Latent Semantic Analysis



Deep Neural Network
Autoencoder 2-D Topic Space



Каждый документ преобразован в вектор встречаемости слов, который затем был спроецирован на плоскость в виде окрашенной точки. Цвет точки соответствует ручной метке класса.

Источник: Hinton G. Lecture 1: What is Machine Learning?

www.cs.toronto.edu/~hinton

2000 reconstructed counts

output
vector

500 neurons

250 neurons

2

250 neurons

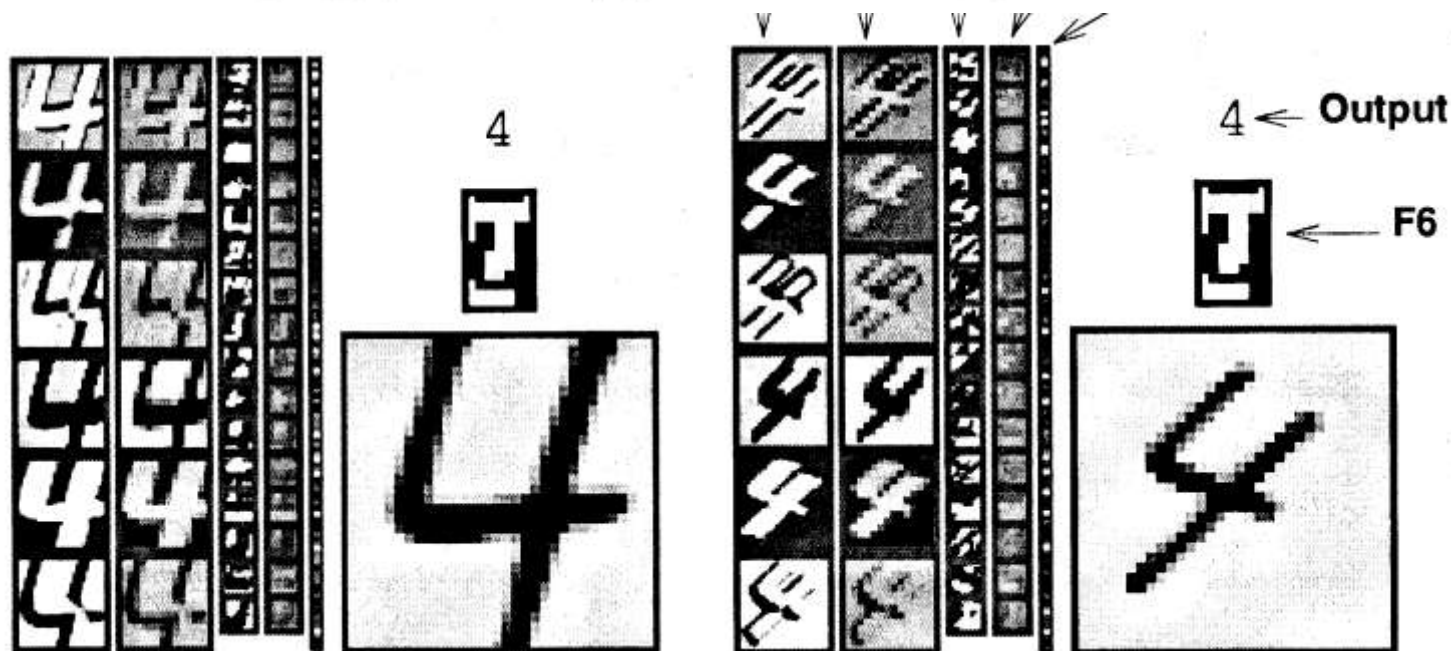
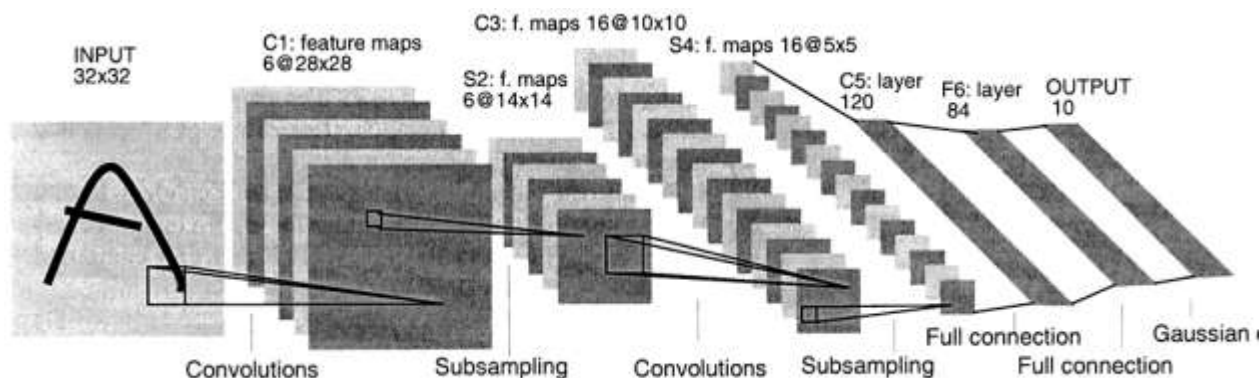
500 neurons

2000 word counts

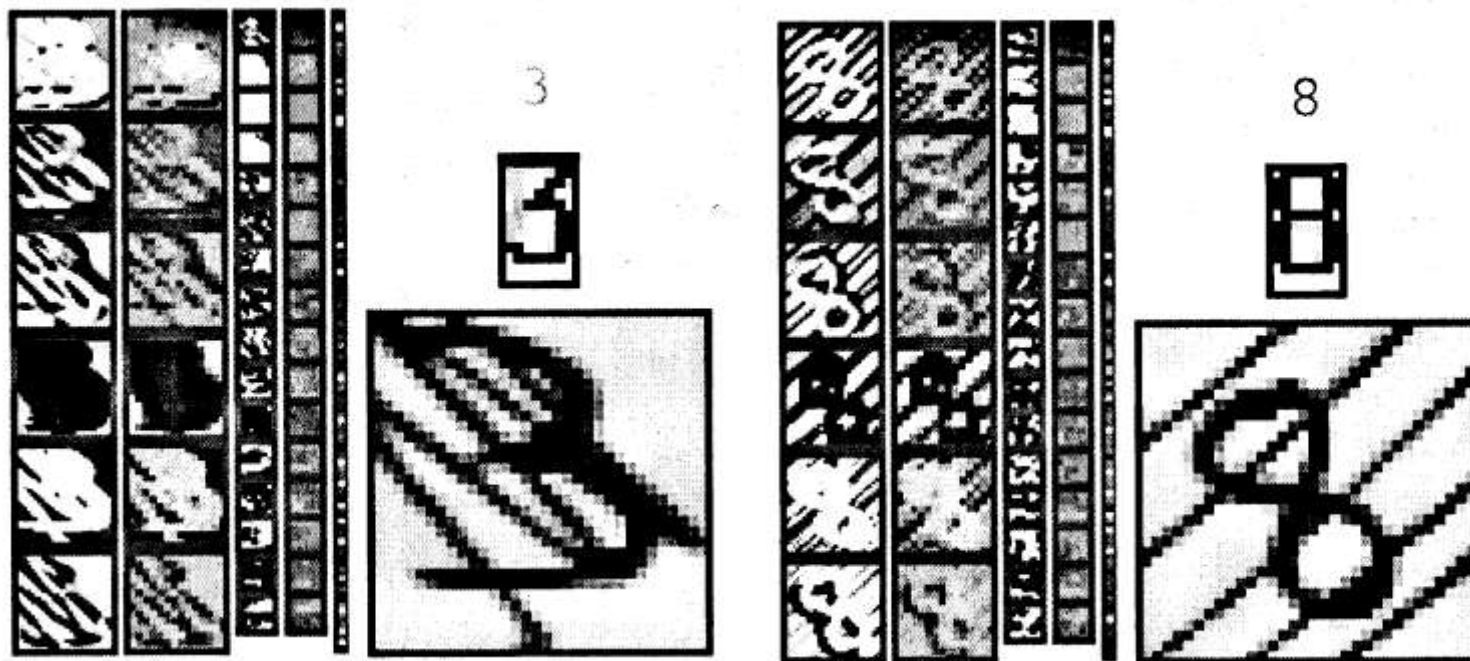
Input vector uses
Poisson units



Сверточная ИНС (LeNet5)



Сверточная ИНС (LeNet5)

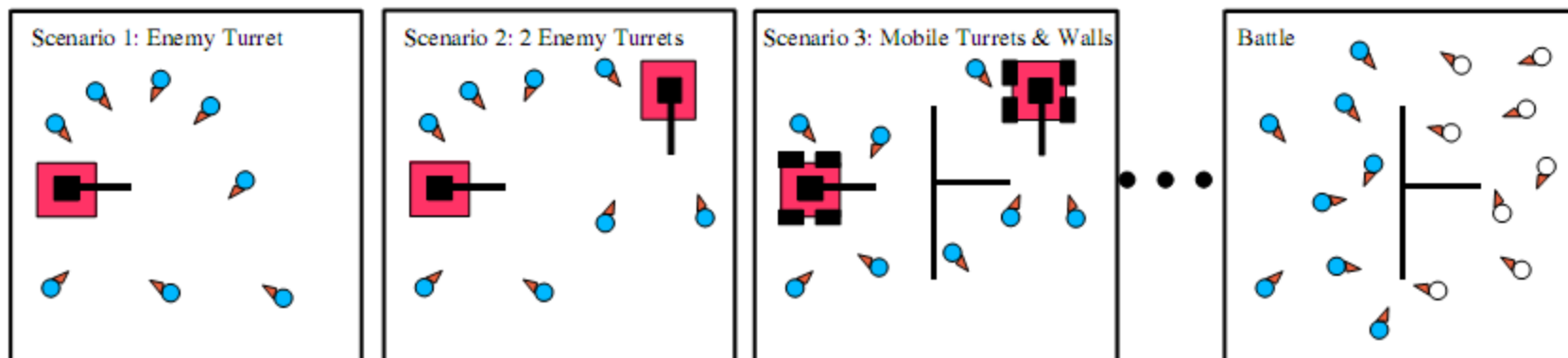


Проект



ИНС управляет действиями
автономного виртуального агента

<http://nerogame.org/>



Использован рисунок из статьи Yong, Stanley, Miikkulainen, Karpov “Incorporating advice into neuroevolution of adaptive agents” (2006)

Проект Aiko

Bio Robot Artificial Intelligence Neural System (aka B.R.A.I.N.S)

- до 8 камер и 32 сенсоров включая гироскопы и акселерометры;
- речь, распознавание образов, чтение текста, слежение за движениями, распознавание лиц и 3-D объектов, общение, взаимодействие с предметами

<http://www.projectaiko.com/>



Ресурсы

- Российская ассоциация нейроинформатики. <http://www.niisi.ru/iont/ni/>
- Европейская ассоциация нейронных сетей. <http://www.e-nns.org/>
- Российская ассоциация искусственного интеллекта. <http://www.raai.org>
- Российская ассоциация нечетких систем и мягких вычислений. <http://ransmv.narod.ru/>
- НИИ нейрокибернетики имени А.Б. Когана. <http://www.krinc.ru/>
- Электронный рецензируемый журнал «Нейроинформатика». <http://www.niisi.ru/iont/ni/library>
- Лекции по нейроинформатике. <http://www.niisi.ru/iont/ni/Library/>, <http://neurolectures.narod.ru>
- Страница В.Л. Дунина-Барковского. http://www.niisi.ru/iont/staff/dbvl/index_rus.php
- Сайт А.Н. Горбаня. <http://www.math.le.ac.uk/people/agl53/homepage/>
- Сайт В. Духа. <http://www.is.umk.pl/~ Duch/>
- Сайт Я. ЛеКуна. <http://yann.lecun.com/>
- Сайт С.А. Терехова. <http://alife.narod.ru/>
- Сайт Д. Хинтона. www.cs.toronto.edu/~hinton
- Сайт Д. Хопфилда. <http://genomics.princeton.edu/hopfield/Index.html>
- Сайт В.Г. Царегородцева. <http://neuropro.ru/>
- UC Irvine Machine Learning Repository, <http://archive.ics.uci.edu/ml>

Искусственные нейронные сети

Эта презентация:

http://qai.narod.ru/TomskWorkshop/lecture_02_2010.pdf

