



# ТЕХНОСФЕРА

## Лекция n2 Softmax слой

### Ограниченнная машина Больцмана

Нестеров Павел

14 декабря 2014 г.

# План лекции

Вспоминаем прошлую лекцию

Softmax слой

Обучение без учителя

Ограниченнная машина Больцмана

Алгоритм contrastive divergence

Заметки про RBM

# Модифицированная модель нейрона МакКаллока-Питтса

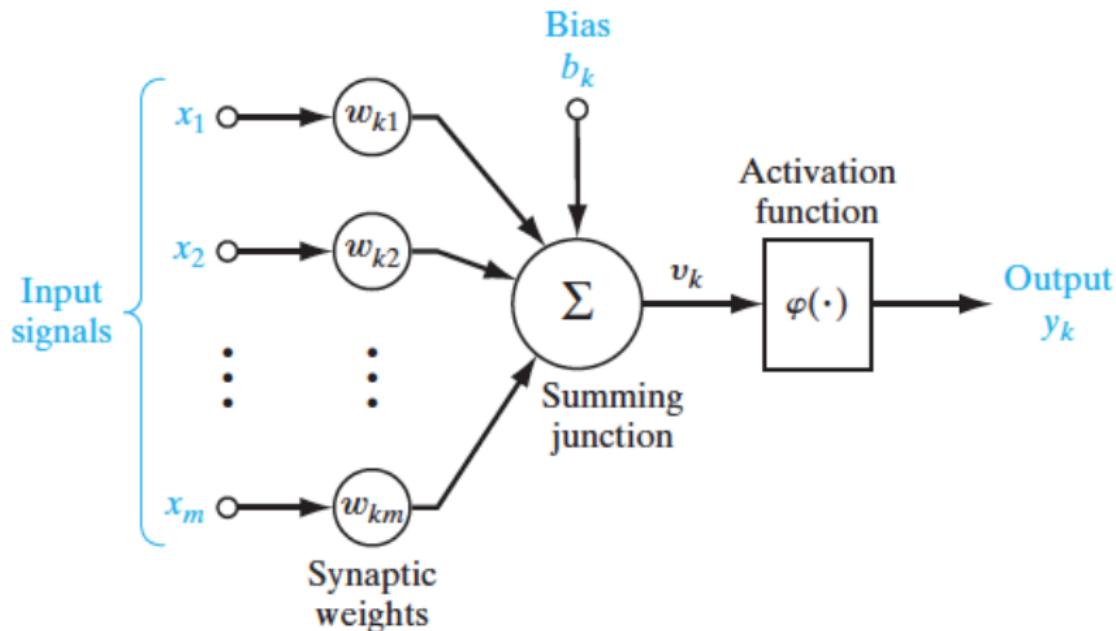


Рис.: Схема искусственного нейрона<sup>1</sup>

<sup>1</sup>Neural Networks and Learning Machines (3rd Edition), Simon O. Haykin

# Многослойная нейронная сеть прямого распространения

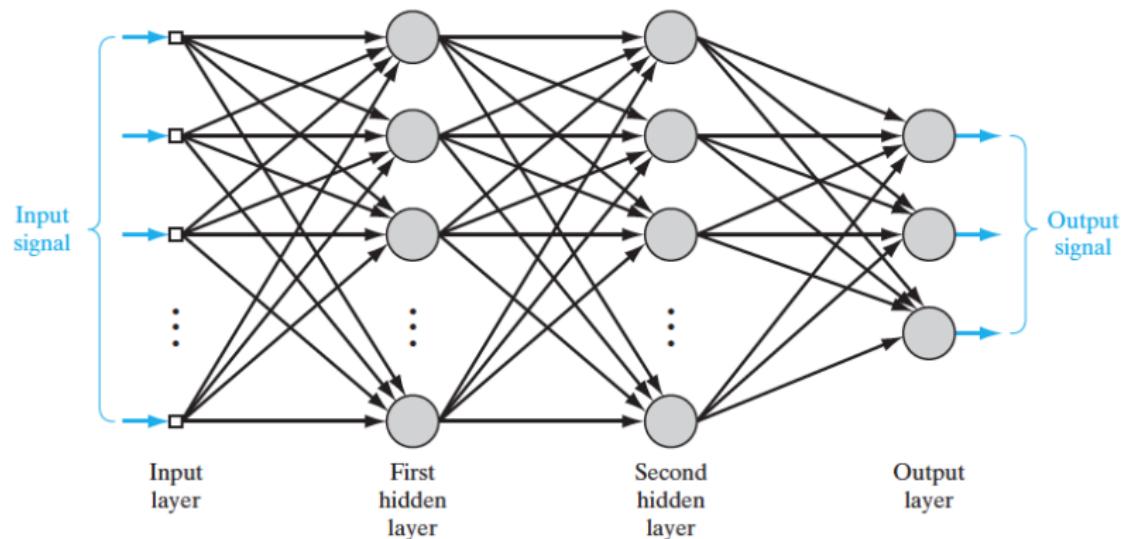


Рис.: Архитектура сети с двумя скрытыми слоями<sup>2</sup>

<sup>2</sup>Neural Networks and Learning Machines (3rd Edition), Simon O. Haykin

# Алгоритм обратного распространения ошибки

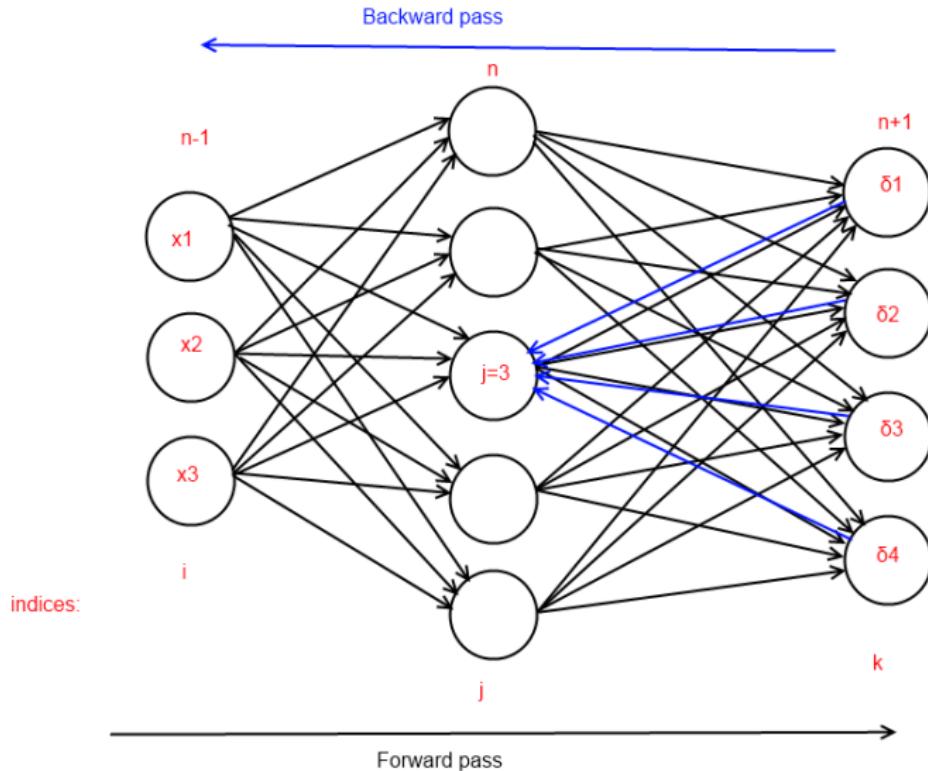


Рис.: Схема прямого (нелинейного) и обратного (линейного) распространения сигнала в сети

# Некоторые функции стоимости

Среднеквадратичная ошибка:

- ▶  $E = \frac{1}{2} \sum_{i \in \text{OUTPUT}} (t_i - y_i)^2$
- ▶  $\frac{\partial E}{\partial y_i} = y_i - t_i$

Логарифм правдоподобия Бернулли:

- ▶  $E = - \sum_{i \in \text{OUTPUT}} (t_i \log y_i + (1 - t_i) \log (1 - y_i))$
- ▶  $\frac{\partial E}{\partial y_i} = \frac{t_i}{y_i} - \frac{1 - t_i}{1 - y_i}$

Для каких задач машинного обучения удобны эти функции?

## Вспоминаем логистическую регрессию, два класса

- $D = \{(x_i, y_i)\}_{i=1\dots m}, \forall x_i \in \mathbb{R}^n, \forall y_i \in \{0, 1\}$

$$\begin{aligned} P(y=1|x) &= \frac{P(x|y) \cdot P(y)}{P(x|y) \cdot P(y) + P(x|\bar{y}) \cdot P(\bar{y})} \\ &= \frac{1}{1 + \exp\left(\frac{P(x|y) \cdot P(y)}{P(x|\bar{y}) \cdot P(\bar{y})}\right)} \\ &= \frac{1}{1 + e^{-a}} = \sigma(a) \end{aligned}$$

где  $\bar{y}$  это  $y = 0$ , а так же  $P(y=0) = 1 - P(y=1)$ , а :  $h(w) = w^T \cdot x$

- $H(p, q) = -\sum_i p_i \cdot \log q_i = -y \cdot \log \hat{y} - (1-y) \cdot \log(1-\hat{y})$
- *какое распределение?*

## Логистическая регрессия, обобщение на N классов

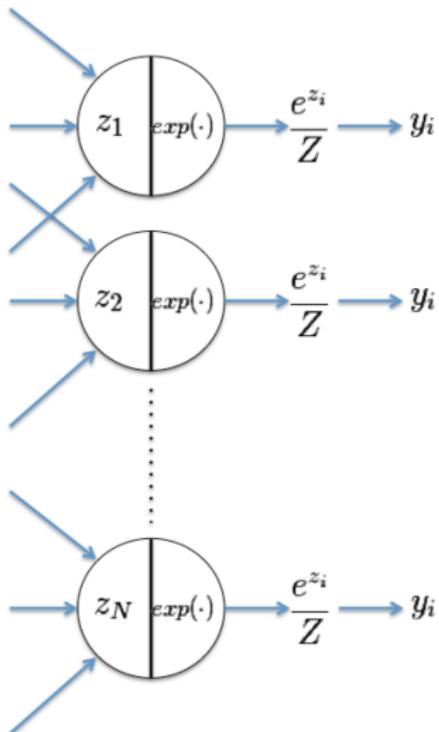
- ▶  $D = \{(x_i, y_i)\}_{i=1\dots m}, \forall x_i \in \mathbb{R}^n, \forall y_i \in \{0, 1, \dots, N\}$
- ▶  $P(y = 1|x) = \frac{1}{1+e^{-a}} = \frac{e^{-a}}{e^{-a}+1} = \frac{1}{Z} \cdot e^{-a} = \frac{1}{Z} \cdot e^{w^T \cdot x}$
- ▶  $Z$  - некоторая нормализующая константа
- ▶  $P(x) = \frac{1}{Z} \cdot e^{-E(x)}$  - распределение Больцмана-Гиббса (почти)

Введем для каждого класса свой вектор весов, получим:

$$P(y = c|x) = \frac{1}{Z} \cdot e^{w_c^T \cdot x} = \frac{e^{w_c^T \cdot x}}{\sum_i e^{w_i^T \cdot x}} \quad (1)$$

- ▶  $\sum_c P(y = c|x) = \sum_c \frac{1}{Z} \cdot e^{w_c^T \cdot x} = \frac{Z}{Z} = 1$
- ▶ как представить в виде нейросети?

# Softmax слой



- ▶  $z_j^{(n)} = \sum_{i=0}^{N_{n-1}} w_{ij}^{(n)} x_i^{(n)}$
- ▶  $y_j = \text{SOFTMAX}(z_j) = \frac{z_j}{Z} = \frac{z_j}{\sum_k z_k}$
- ▶  $E(\vec{y}, \vec{t}) = - \sum_{j=1}^{N_{n-1}} t_j \cdot \log y_j$
- ▶  $\frac{\partial y_j^{(n)}}{\partial z_j^{(n)}} = ???$

## Дифференцирование softmax функции

$$\begin{aligned}\frac{\partial y_j^{(n)}}{\partial z_j^{(n)}} &= \frac{\partial}{\partial z_j} \frac{e^{z_j}}{Z} \\&= \frac{1}{Z^2} \cdot \left( \frac{\partial e^{z_j}}{\partial z_j} \cdot Z - e^{z_j} \cdot \frac{\partial Z}{\partial z_j} \right) \\&= \frac{1}{Z^2} \cdot \left( e^{z_j} Z - e^{z_j} \frac{\partial e^{z_j}}{\partial z_j} \right) \\&= \frac{e^{z_j} Z - (e^{z_j})^2}{Z^2} = \frac{e^{z_j}}{Z} - \left( \frac{e^{z_j}}{Z} \right)^2 \\&= y_j - y_j^2 \\&= y_j \cdot (1 - y_j)\end{aligned}$$

## Вспомним backprop

- ▶  $\frac{\partial E}{\partial w_{ij}^{(n)}} = \frac{\partial E}{\partial z_j^{(n)}} \frac{\partial z_j^{(n)}}{\partial w_{ij}^{(n)}}$
- ▶  $\frac{\partial z_j^{(n)}}{\partial w_{ij}^{(n)}} = \sum_i \frac{\partial w_{ij}^{(n)} x_i^{(n-1)}}{\partial w_{ij}^{(n)}} = x_i^{(n-1)}$

В итоге получим:

$$\frac{\partial E}{\partial w_{ij}^{(n)}} = x_i^{(n-1)} \frac{\partial E}{\partial z_j^{(n)}} \quad (2)$$

Продолжим с этого момента, с учетом того, что функцией стоимости является перекрестная энтропия (опустим индекс слоя для наглядности):

- ▶  $E(\vec{y}(\vec{z}), \vec{t}) = - \sum_{j=1}^{N_{n-1}} t_j \cdot \log y_j(z_j)$
- ▶  $\frac{\partial E}{\partial z_j} = ???$

# Дифференцирование перекрестной энтропии, #1

Раньше было так (для выходного слоя):

$$\frac{\partial E}{\partial z_j} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial z_j}$$

Теперь стало так:

$$\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$$

- ▶ почему так?

## Дифференцирование перекрестной энтропии, #2

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i}$  ???

## Дифференцирование перекрестной энтропии, #2

►  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$

$$\begin{aligned}\frac{\partial E}{\partial y_i} &= -\frac{\partial}{\partial y_i} \left( \sum_k^{N_n} t_k \cdot \log y_k \right) \\ &= -\frac{\partial}{\partial y_i} (t_i \cdot \log y_i) \\ &= -\frac{t_i}{y_i}\end{aligned}$$

## Дифференцирование перекрестной энтропии, #3

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i} = -\frac{t_i}{y_i}$
- ▶  $\frac{\partial y_i}{\partial z_j}$  ???

## Дифференцирование перекрестной энтропии, #4

$$\blacktriangleright \frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$$

$$\blacktriangleright \frac{\partial E}{\partial y_i} = -\frac{t_i}{y_i}$$

$$\frac{\partial y_i}{\partial z_j} = \begin{cases} y_j (1 - y_j), & i = j \\ ???, & i \neq j \end{cases}$$

## Дифференцирование перекрестной энтропии, #5

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i} = -\frac{t_i}{y_i}$
- ▶  $\frac{\partial y_i}{\partial z_j} = \begin{cases} y_j(1-y_j), & i=j \\ ???, & i \neq j \end{cases}$

$$\begin{aligned}\frac{\partial y_i^{(n)}}{\partial z_j^{(n)}} &= \frac{1}{Z^2} \cdot \left( \frac{\partial e^{z_i}}{\partial z_j} \cdot Z - e^{z_i} \cdot \frac{\partial Z}{\partial z_j} \right) \\ &= \frac{1}{Z^2} (0 - e^{z_i} \cdot e^{z_j}) \\ &= -y_i \cdot y_j\end{aligned}$$

# Дифференцирование перекрестной энтропии, #6

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i} = -\frac{t_i}{y_i}$
- ▶  $\frac{\partial y_i}{\partial z_j} = \begin{cases} y_j(1-y_j), & i=j \\ -y_i y_j, & i \neq j \end{cases}$
- ▶  $\frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j} = \begin{cases} -t_j(1-y_j), & i=j \\ y_j t_i, & i \neq j \end{cases}$
- ▶ собираем все вместе

## Дифференцирование перекрестной энтропии, #7

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j} = \begin{cases} -t_j(1-y_j), & i=j \\ y_j t_i, & i \neq j \end{cases}$

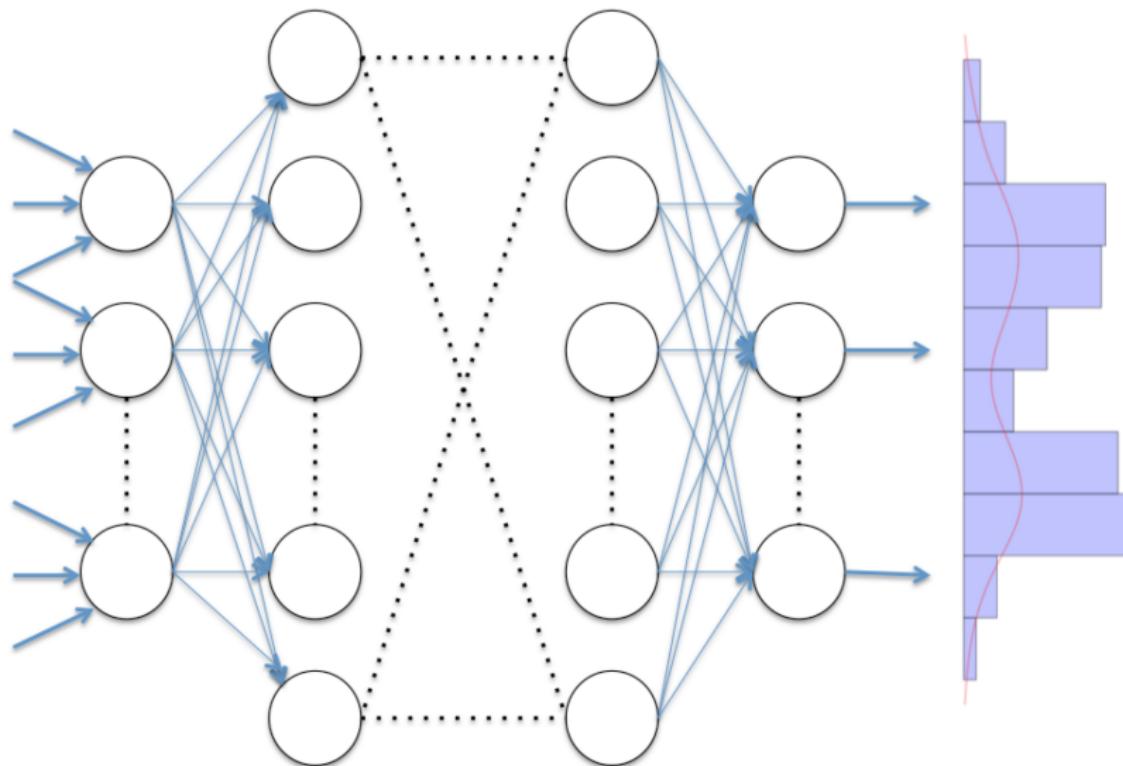
$$\begin{aligned}\frac{\partial E}{\partial z_j} &= -t_j(1-y_j) + \sum_{i=1, i \neq j}^{N_n} y_j t_i \\ &= -t_j + t_j y_j + y_j \cdot \sum_{i=1, i \neq j}^{N_n} t_i \\ &= -t_j + y_j \left( t_j + \cdot \sum_{i=1, i \neq j}^{N_n} t_i \right) \\ &= ???\end{aligned}$$

## Дифференцирование перекрестной энтропии, #8

- ▶  $\frac{\partial E}{\partial z_j} = \sum_{i=1}^{N_n} \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j}$
- ▶  $\frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial z_j} = \begin{cases} -t_j(1-y_j), & i=j \\ y_j t_i, & i \neq j \end{cases}$

$$\begin{aligned}\frac{\partial E}{\partial z_j} &= -t_j(1-y_j) + \sum_{i=1, i \neq j}^{N_n} y_j t_i \\ &= -t_j + t_j y_j + y_j \cdot \sum_{i=1, i \neq j}^{N_n} t_i \\ &= -t_j + y_j \left( t_j + \sum_{i=1, i \neq j}^{N_n} t_i \right) \\ &= y_j - t_j\end{aligned}$$

## Softmax слой, выводы



# Обучение без учителя

*When we're learning to see, nobody's telling us what the right answers are — we just look. Every so often, your mother says "that's a dog", but that's very little information. You'd be lucky if you got a few bits of information — even one bit per second — that way. The brain's visual system has  $10^{14}$  neural connections. And you only live for  $10^9$  seconds. So it's no use learning one bit per second. You need more like  $10^5$  bits per second. And there's only one place you can get that much information: from the input itself.<sup>3</sup>*

---

<sup>3</sup>Geoffrey Hinton, 1996 (quoted in (Gorder 2006))

# Статистическая механика, #1

Представим некоторую физическую систему с множеством степеней свободы, которая может находиться в одном из множества состояний с некоторой вероятностью, а каждому такому состоянию состоянию соответствует некоторая энергия всей системы:

- ▶  $p_i \geq 0$  - вероятность состояния  $i$
- ▶  $\sum_i p_i = 1$
- ▶  $E_i$  - энергия системы в состоянии  $i$

Тогда вероятность состояния  $i$  будет описываться распределением Больцмана-Гиббса, при условии термодинамического равновесия между системой и окружающей средой:

$$p_i = \frac{1}{Z} \exp\left(-\frac{E_i}{k_B \cdot T}\right) \quad (3)$$

где

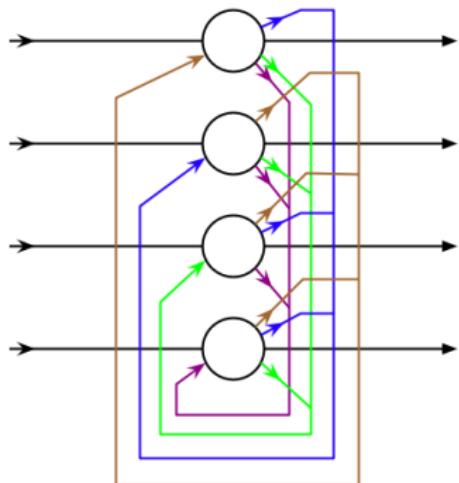
- ▶  $T$  - абсолютная температура (К)
- ▶  $k_B$  - константа Больцмана (Дж/К)
- ▶  $Z = \sum_i \exp\left(-\frac{E_i}{k_B \cdot T}\right)$  - нормализующая константа (partition function, Zustadsumme, статсумма)

# Статистическая механика, #2

Два важных вывода:

1. состояния с низкой энергией имеют больше шансов возникнуть чем состояния с высокой энергией;
2. при понижении температуры, чаще всего будут возникать состояния из небольшого подмножества состояний с низкой энергией.

# Нейросеть Хопфилда



- ▶ обратная связь
- ▶ пороговая функция активации

Такая сеть (рекуррентная нейронная сеть) может находиться как в стабильном состоянии, осциллировать, или даже проявлять признаки детерминированного хаоса.

Хопфилд показал, что при симметричной матрице весов, существует такая функция энергии бинарных состояний системы, что при симуляции система эволюционирует в одно из низко-энергетических состояний.

# Нейросеть Хопфилда, энергия системы, #1

$$E = - \sum_i s_i b_i - \sum_{i < j} s_i s_j w_{ij} \quad (4)$$

- ▶  $s_i$  - состояние нейрона  $i$
- ▶  $b_i$  - смещение нейрона  $i$
- ▶  $w_{ij}$  - вес между нейроном  $i$  и  $j$

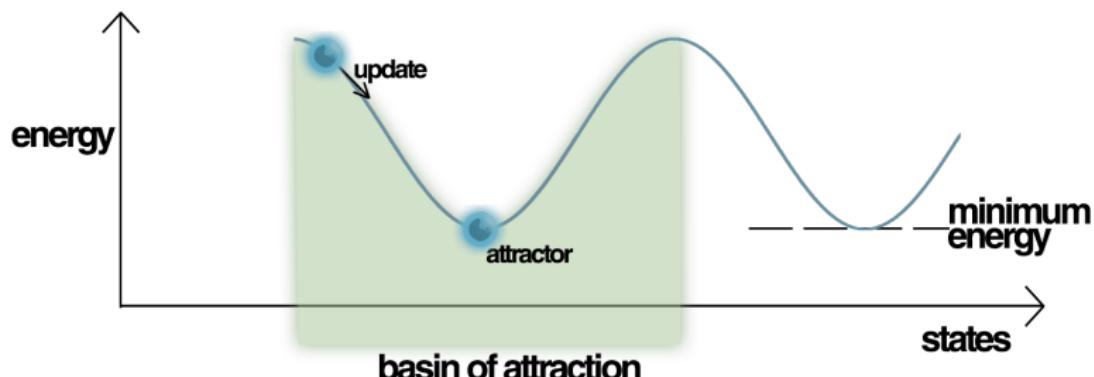


Рис.: Поверхность описываемая энергией сети Хопфилда

## Нейросеть Хопфилда, энергия системы, #2

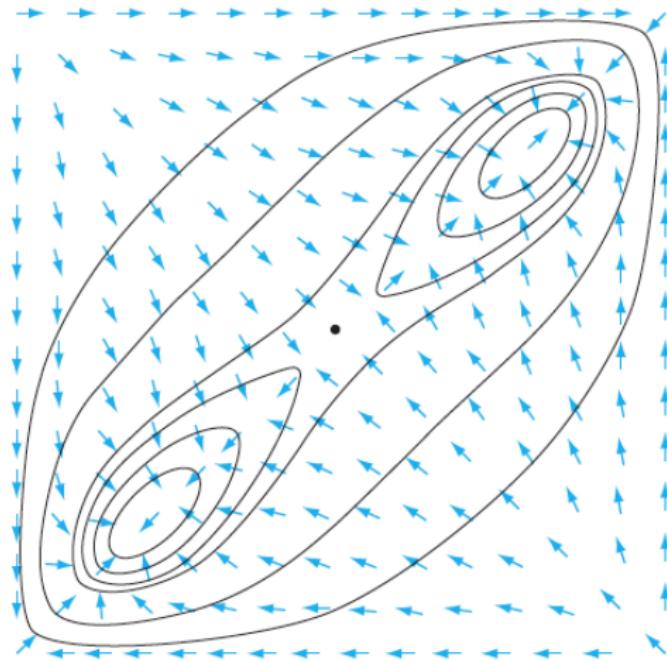
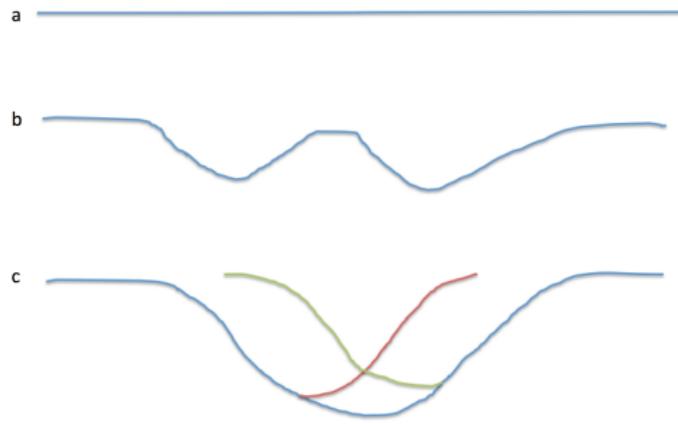


Рис.: Поверхность описываемая энергией сети Хопфилда, два стабильных состояния<sup>4</sup>

<sup>4</sup>Neural Networks and Learning Machines (3rd Edition), Simon O. Haykin

# Нейросеть Хопфилда, как ассоциативная память



- ▶ а - нет образов в памяти
- ▶ б - два образа далеко друг от друга
- ▶ с - два образа накладываются друг на друга

Вместимость  $0.15 \cdot N$  на  $N$  нейронов.

## Нейросеть Хопфилда, алгоритм обучения

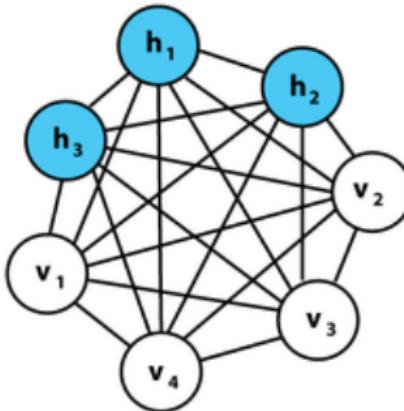
Обучение сети Хопфилда происходит в один прогон по множеству данных по следующему правилу:

$$\Delta w_{ij} = \frac{1}{n} \sum_{i=1}^n s_i s_j, \forall k : s_k \in \{-1, 1\} \quad (5)$$

Это в точности первое правило Хебба:

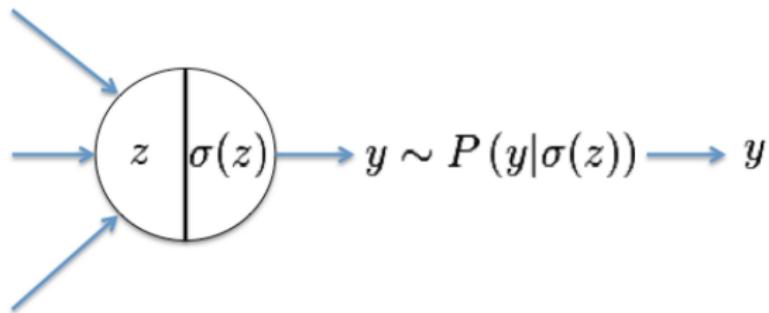
- ▶ если два нейрона по разные стороны от синапсов активируются синхронно, то "вес" синапса слегка возрастает

# Машина Больцмана - стохастический генеративный вариант сети Хопфилда



- ▶ энергия не изменилась:  $E = - \sum_i s_i b_i - \sum_{i < j} s_i s_j w_{ij}$
- ▶ симметричная матрица весов  $w_{ij} = w_{ji}$ , но нет обратных связей:  $w_{ii} = 0$
- ▶ появились скрытые состояния (система ищет такую конфигурацию скрытых состояний которая лучшим образом описывает видимые состояния)
- ▶  $\forall i : s_i \in \{0, 1\}$
- ▶ стохастический нейрон

# Стохастический нейрон



Имитация отжига, идея, #1



## Имитация отжига, идея, #2

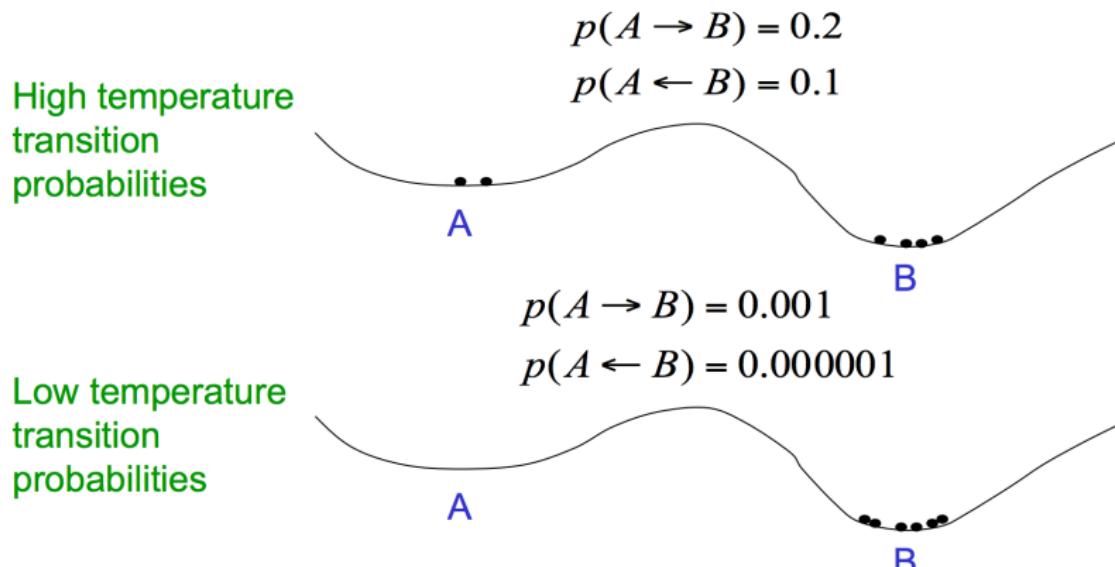


Рис.: Влияние температуры на вероятности переходов<sup>5</sup>

▶ SimulatedAnnealing.gif

<sup>5</sup><https://class.coursera.org/neuralnets-2012-001/lecture>

## Имитация отжига

- ▶  $\Delta E_i = b_i + \sum_j w_{ij} s_j$

- ▶  $p_i = \frac{1}{Z} \exp\left(-\frac{E_i}{k_B \cdot T}\right)$

$$\frac{p_{i=1}}{p_{i=0}} = \exp\left(-\frac{E_{i=0} - E_{i=1}}{k_B T}\right) = \exp\left(\frac{\Delta E_i}{k_B T}\right) \Rightarrow$$

$$\frac{\Delta E_i}{T} = \ln(p_{i=1}) - \ln(p_{i=0}) = \ln(p_{i=1}) - \ln(1 - p_{i=1})$$

$$= \ln\left(\frac{p_{i=1}}{1 - p_{i=1}}\right) \Rightarrow$$

$$-\frac{\Delta E_i}{T} = \ln\left(\frac{1 - p_{i=1}}{p_{i=1}}\right)$$

$$= \ln\left(\frac{1}{p_{i=1}} - 1\right) \Rightarrow$$

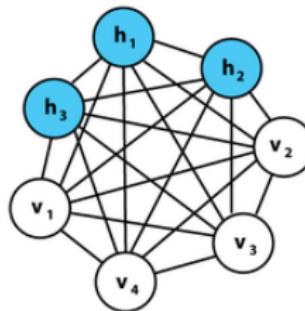
$$\exp\left(-\frac{\Delta E_i}{T}\right) = \frac{1}{p_{i=1}} - 1 \Rightarrow$$

$$p_{i=1} = \frac{1}{1 + \exp\left(-\frac{\Delta E_i}{T}\right)}$$

# Машина Больцмана - выводы

Теоретически такая модель может все (как обычно в нейросетях), но

- ▶ время требуемое для обучения такой модели экспоненциально зависит от размера машины
- ▶ по этой же причине нет возможности вычислить  $Z$
- ▶ так же приходится использовать семплирование Гиббса<sup>6</sup>, в связи с топологией сети (*почему?*)

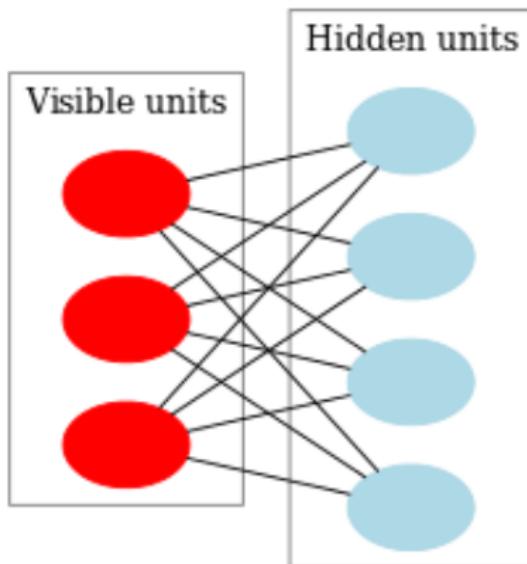


---

<sup>6</sup>Семплирование по Гиббсу не требуется явно выраженное совместное распределение, а нужны лишь условные вероятности для каждой переменной, входящей в распределение. Алгоритм на каждом шаге берет одну случайную величину и выбирает ее значение при условии фиксированных остальных. Можно показать, что последовательность получаемых значений образуют возвратную цепь Маркова, устойчивое распределение которой является как раз искомым совместным распределением.

# Ограниченнная машина Больцмана

- ▶ убираем температуру
- ▶ вводим ограничение на топологию



# Виды RBM

В зависимости от априорного распределения ассоциированного с видимым и скрытым слоями, различают несколько видов RBM:

- ▶ Bernoulli-Bernoulli (binary-binary)
- ▶ Gaussian-Bernoulli
- ▶ Gaussian-Gaussian
- ▶ Poisson-Bernoulli
- ▶ и т.д.

Бинарные (Bernoulli-Bernoulli, binary-binary) RBM играют важную роль в глубоком обучении, по аналогии с выводом алгоритма обучения для бинарной ограниченной машины Больцмана, можно вывести аналогичные правила для остальных типов моделей.

# RBM, обозначения

- ▶  $D = \{\vec{x}_i\}_{i=1\dots N}$  - множество данных;
- ▶  $\vec{v}, \vec{h}$  - значения видимых и скрытых нейронов;
- ▶  $\vec{a}, \vec{b}, W$  - смещения видимых и скрытых нейронов, и матрица весов;
- ▶  $n, m$  - количество видимых и скрытых нейронов;
- ▶  $E(\vec{v}, \vec{h}) = -\sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j = -\vec{v}^T \vec{a} - \vec{h}^T \vec{b} - \vec{v}^T W \vec{h}$
- ▶  $p(\vec{v}, \vec{h}) = \frac{1}{Z} e^{-E(\vec{v}, \vec{h})}$
- ▶  $Z = \sum_r^N \sum_t^M e^{-E(\vec{v}^{(r)}, \vec{h}^{(t)})}$
- ▶  $P(\vec{v}) = \sum_t^M P(\vec{v}, \vec{h}^{(t)}) = \frac{1}{Z} \sum_t^M e^{-E(\vec{v}, \vec{h}^{(t)})}$

Далее значки вектора  $\vec{x}$  будут опускаться для простоты.

## RBM, функция активации

Аналогично обычной машине Больцмана, рассмотрим только для скрытого слоя:

$$\begin{aligned} P(h_k = 1|v) &= \frac{e^{-E_1}}{e^{-E_1} + e^{-E_0}} \\ &= \frac{1}{1 + e^{E_1 - E_0}} \\ &= \frac{1}{1 + e^{-b_k - \sum_i^n v_i w_{ik}}} \\ &= \sigma \left( b_k + \sum_{i=1}^n v_i w_{ik} \right) \end{aligned}$$

*Вопрос:*

- ▶  $P(h|v) = ???$

## RBM, независимость

$$\begin{aligned} P(h|v) &= \prod_{j=1}^m P(h_j|v) \\ P(v|h) &= \prod_{i=1}^n P(v_i|h) \end{aligned}$$

## RBM, целевая функция

$$E(\vec{v}, \vec{h}) = -\sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j \quad (6)$$

$$P(\vec{v}) = \frac{1}{Z} \sum_t^M e^{-E(\vec{v}, \vec{h}^{(t)})} \quad (7)$$

- ▶ максимизировать вероятность данных при заданной генеративной модели
- ▶ что будем делать?

# RBM, дифференцирование $P(v)$ , #1

$$E(\vec{v}, \vec{h}) = - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j$$

$$\frac{\partial E(v, h)}{\partial w_{ij}} = ?$$

$$\frac{\partial E(v, h)}{\partial a_i} = ?$$

$$\frac{\partial E(v, h)}{\partial b_j} = ?$$

$$\frac{\partial e^{-E(v, h)}}{\partial w_{ij}} = ?$$

$$\frac{\partial e^{-E(v, h)}}{\partial a_i} = ?$$

$$\frac{\partial e^{-E(v, h)}}{\partial b_j} = ?$$

## RBM, дифференцирование $P(v)$ , #2

$$E(\vec{v}, \vec{h}) = - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j$$

$$\begin{array}{lcl} \frac{\partial E(v, h)}{\partial w_{ij}} & = & -v_i h_j \\ \frac{\partial E(v, h)}{\partial a_i} & = & -v_i \\ \frac{\partial E(v, h)}{\partial b_j} & = & -h_j \end{array} \quad \begin{array}{lcl} \frac{\partial e^{-E(v, h)}}{\partial w_{ij}} & = & v_i h_j e^{-E(v, h)} \\ \frac{\partial e^{-E(v, h)}}{\partial a_i} & = & v_i e^{-E(v, h)} \\ \frac{\partial e^{-E(v, h)}}{\partial b_j} & = & h_j e^{-E(v, h)} \end{array}$$

## RBM, дифференцирование $P(v)$ , #3

$$Z = \sum_r^N \sum_t^M e^{-E(\vec{v}^{(r)}, \vec{h}^{(t)})}$$

$$\frac{\partial Z}{\partial w_{ij}} = ?$$

$$\frac{\partial Z}{\partial a_i} = ?$$

$$\frac{\partial Z}{\partial b_j} = ?$$

## RBM, дифференцирование $P(v)$ , #4

$$Z = \sum_r^N \sum_t^M e^{-E(\vec{v}^{(r)}, \vec{h}^{(t)})}$$

$$\frac{\partial Z}{\partial w_{ij}} = \sum_r^N \sum_t^M v_i^{(r)} h_j^{(t)} e^{-E(v^{(r)}, h^{(t)})}$$

$$\frac{\partial Z}{\partial a_i} = \sum_r^N \sum_t^M v_i^{(r)} e^{-E(v^{(r)}, h^{(t)})}$$

$$\frac{\partial Z}{\partial b_j} = \sum_r^N \sum_t^M h_j^{(t)} e^{-E(v^{(r)}, h^{(t)})}$$

## RBM, дифференцирование $P(v)$ , #5

$$\frac{\partial P(v^{(k)})}{\partial w_{ij}} = \frac{1}{Z^2} \left( Z \left( \sum_t^M v_i^{(k)} h_j^{(k)} e^{-E(v^{(r)}, h^{(t)})} \right) - \left( \sum_t^M e^{-E(v^{(r)}, h^{(t)})} \right) \left( \sum_r^N \sum_t^M v_i^{(r)} h_j^{(t)} e^{-E(v^{(r)}, h^{(t)})} \right) \right)$$

$$\frac{\partial \ln P(v^{(k)})}{\partial w_{ij}} = \frac{1}{P(v^{(k)})} \frac{\partial P(v^{(k)})}{\partial w_{ij}}$$

## RBM, дифференцирование $P(v)$ , #6

$$\begin{aligned}\frac{\partial \ln P(v^{(k)})}{\partial w_{ij}} &= v_i^{(k)} \sum_t^M h_j^{(t)} P(h^{(t)} | v^{(k)}) - \sum_r^N \sum_t^M v_i^{(r)} h_j^{(t)} P(h^{(t)}, v^{(k)}) \\ &= ???\end{aligned}$$

## RBM, дифференцирование $P(v)$ , #7

$$\begin{aligned}\frac{\partial \ln P(v^{(k)})}{\partial w_{ij}} &= \sum_t^M v_i^{(k)} h_j^{(t)} P(h^{(t)} | v^{(k)}) - \sum_r^N \sum_t^M v_i^{(r)} h_j^{(t)} P(h^{(t)}, v^{(k)}) \\ &= M [v_i^{(k)} h_j]_{\text{DATA}} - M [v_i h_j]_{\text{MODEL}}\end{aligned}$$

$$\frac{\partial \ln P(v^{(k)})}{\partial a_i} = v_i^{(k)} - M [v_i]_{\text{MODEL}}$$

$$\frac{\partial \ln P(v^{(k)})}{\partial b_j} = M [h_j]_{\text{DATA}} - M [h_j]_{\text{MODEL}}$$

## RBM, правила обновления

$$\begin{aligned}\Delta w_{ij} &= \eta \left( M \left[ v_i^{(k)} h_j \right]_{\text{DATA}} - M \left[ v_i h_j \right]_{\text{MODEL}} \right) \\ \Delta a_i &= \eta \left( v_i^{(k)} - M \left[ v_i \right]_{\text{MODEL}} \right) \\ \Delta b_j &= \eta \left( M \left[ h_j \right]_{\text{DATA}} - M \left[ h_j \right]_{\text{MODEL}} \right)\end{aligned}$$

# Алгоритм Contrastive Divergence

- Цель: собрать достаточную статистику для оценки  $M[\dots]_{\text{DATA}}$  и  $M[\dots]_{\text{MODEL}}$

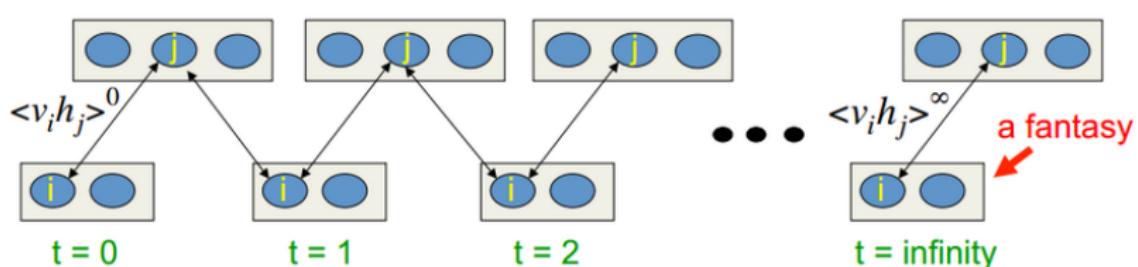


Рис.: Процесс сбора достаточной статистики<sup>7</sup>

- $\Delta w_{ij} = \eta \left( M \left[ v_i^{(k)} h_j \right]^{(0)} - M [v_i h_j]^{(\infty)} \right)$
- $M[\dots]^{(0)}$  - позитивная фаза
- $M[\dots]^{(\infty)}$  - негативная фаза

<sup>7</sup><https://class.coursera.org/neuralnets-2012-001/lecture>

## Практические советы

- ▶ не семплировать видимый слой (семплирование замедляет сходимость, но математически это более корректно);
- ▶ не семплировать значения скрытого слоя при выводе из восстановленного образа;
- ▶ CD- $k$ , уже при  $k = 1$  качество не сильно уступает большим значениям, но выигрыш в скорости значительный;
- ▶ размер минибатча 10-100 экземпляров (*почему?*);
- ▶ кроссвалидаци восстановленных образов;
- ▶ использование момента оказывается крайне положительно на скорости сходимости;
- ▶ <http://www.cs.toronto.edu/~hinton/absps/guideTR.pdf>

## Визуализация восстановленных образов

Z Z

X X

U U

Q Q

O O

M M

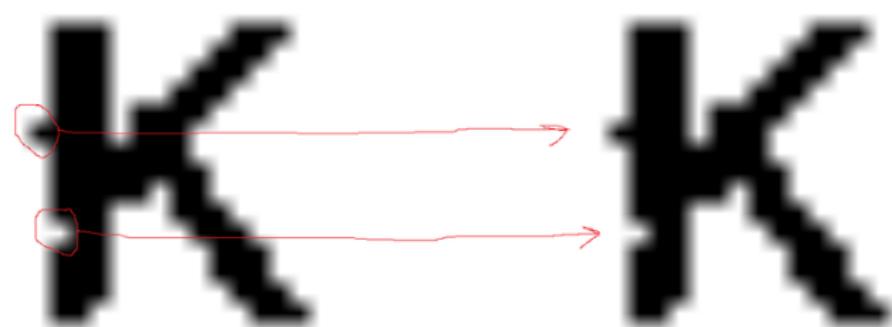
J J

H H

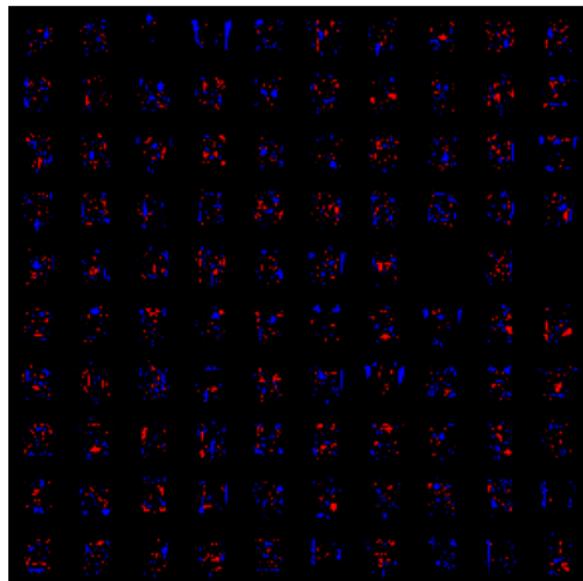
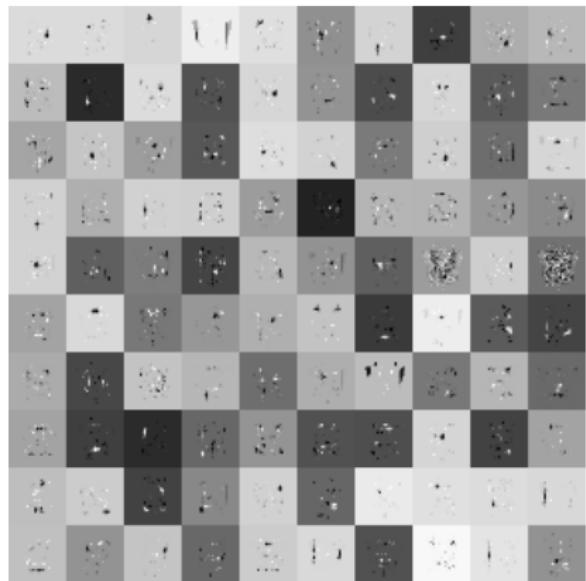
E E

B B

A A



# Визуализация признаков, #1



## Визуализация признаков, #2

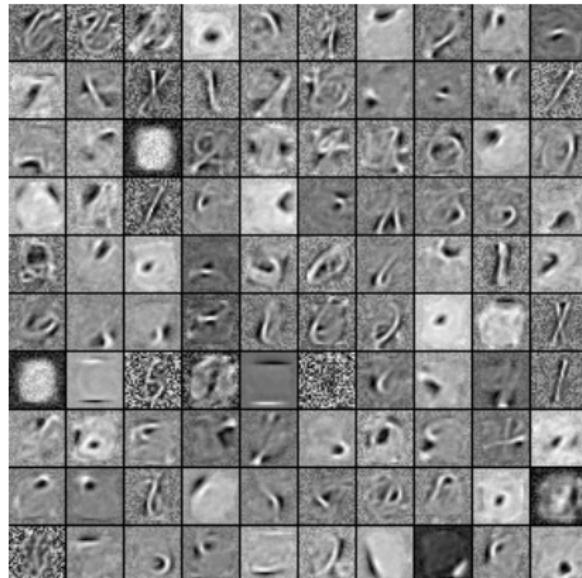
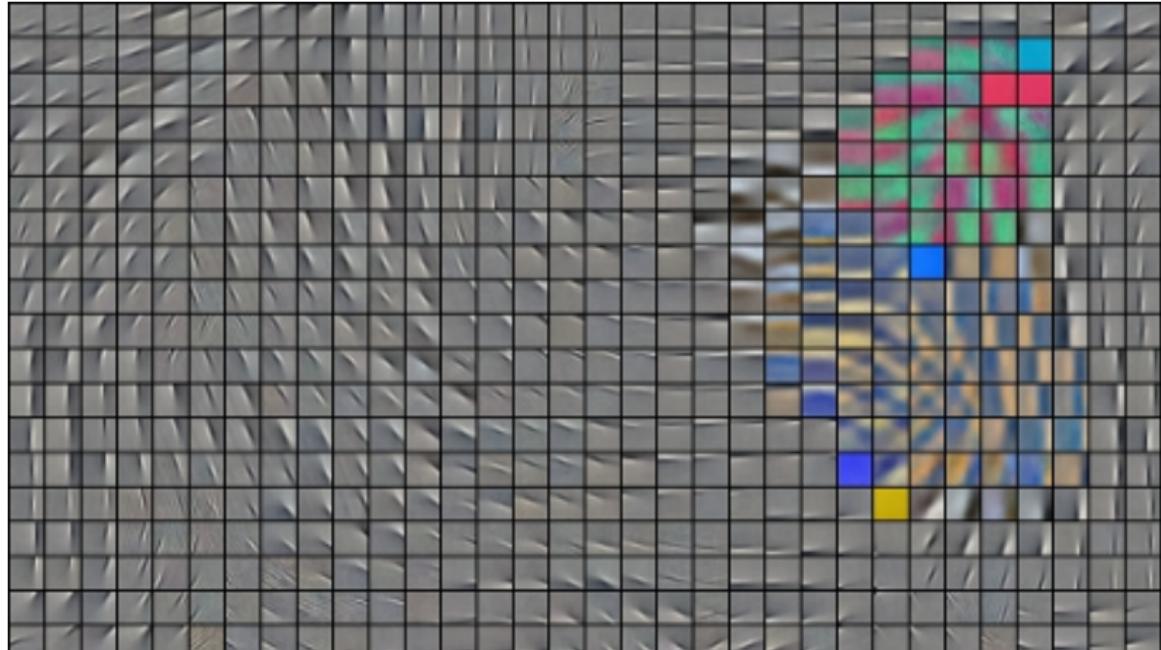


Рис.: Признаки на множестве рукописных цифр MNIST<sup>8</sup>

---

<sup>8</sup><http://deeplearning.net/>

## Визуализация признаков, #3



## Визуализация признаков, #4

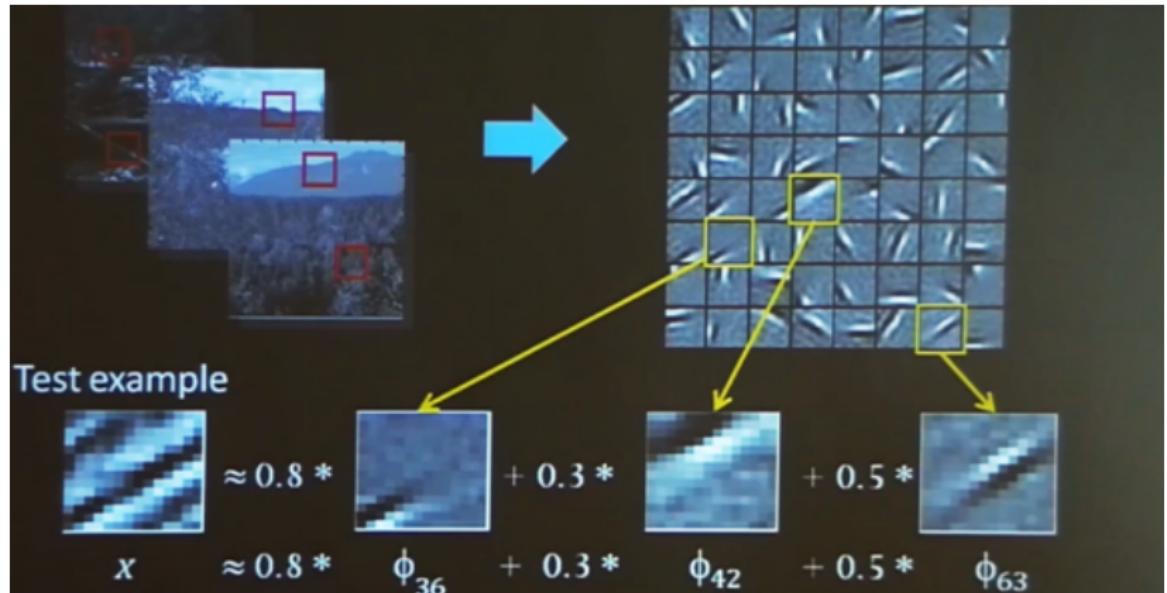
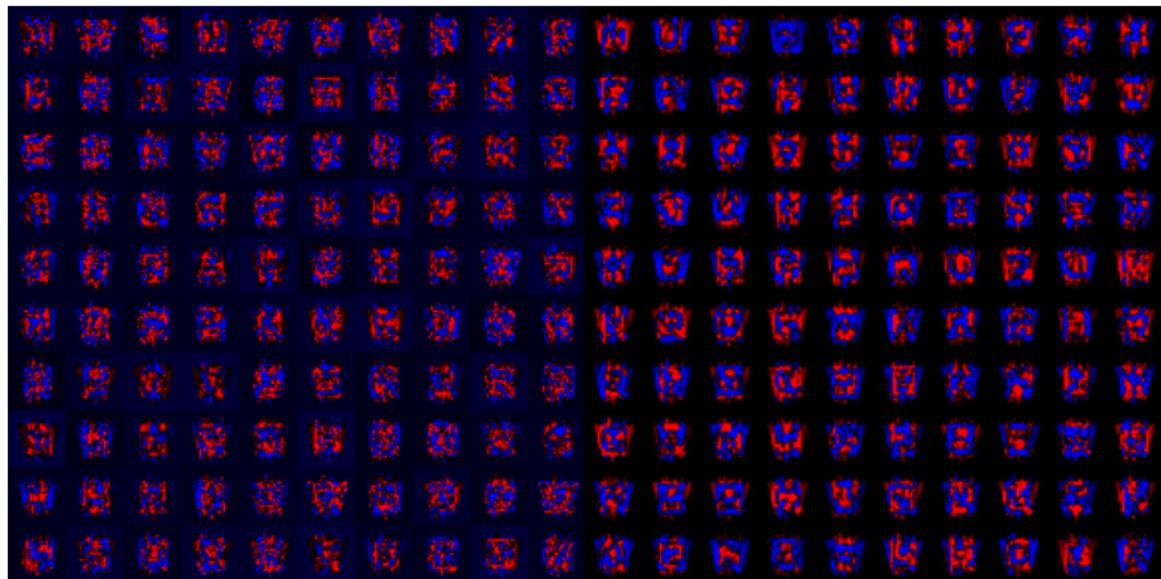


Рис.: RBM как базис<sup>9</sup>

<sup>9</sup><http://cs.stanford.edu/>

# Регуляризация в RBM, #1

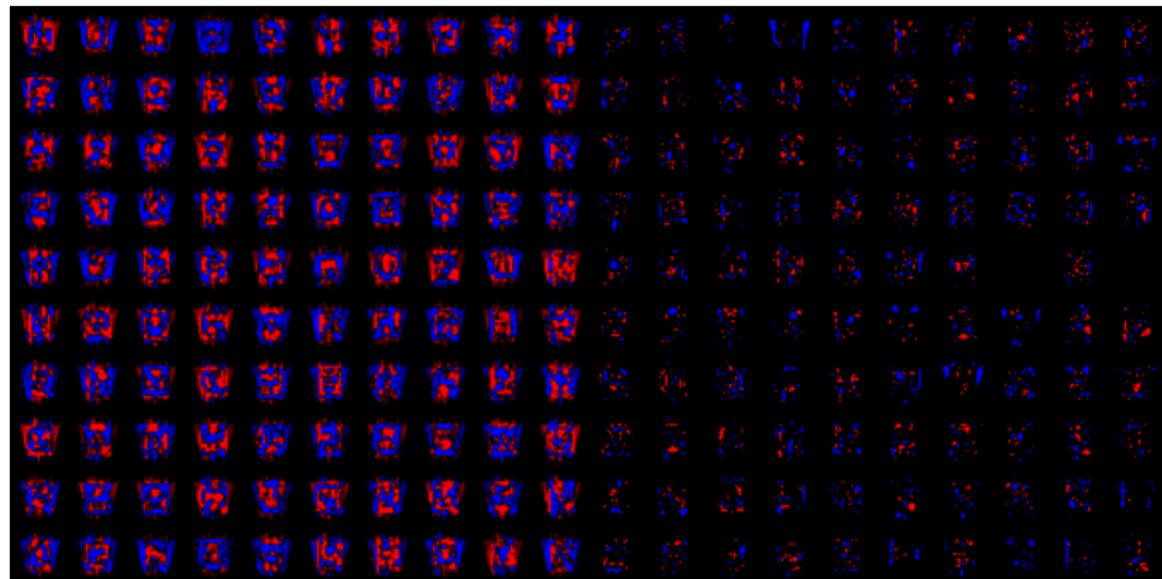


(a) RBM, no reg

(b) RBM, L2 reg

Рис.: Иллюстрация эффекта регуляризации

## Регуляризация в RBM, #2



(a) RBM, L2 reg

(b) RBM, L1 reg

Рис.: Иллюстрация эффекта регуляризации

# Критерий остановки

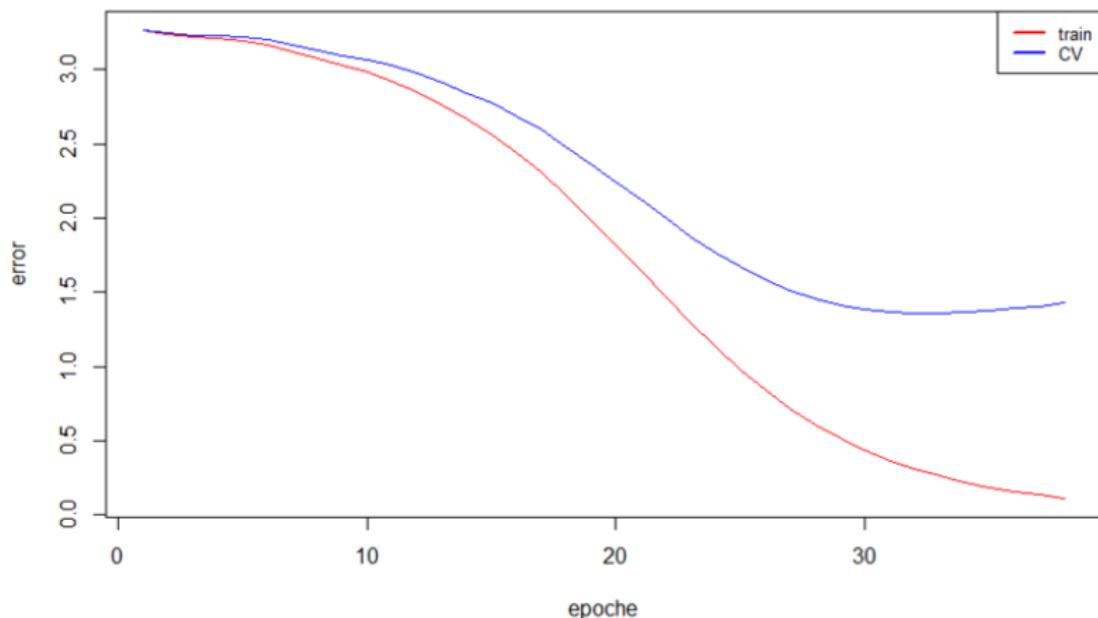


Рис.: Кроссвалидация

Что дальше?

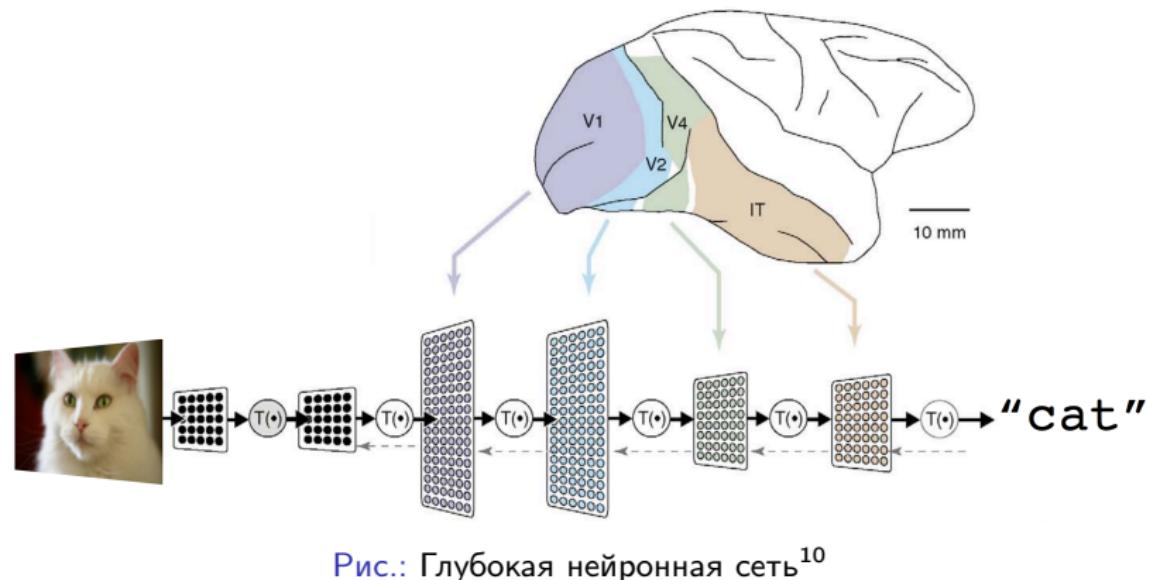


Рис.: Глубокая нейронная сеть<sup>10</sup>

<sup>10</sup>Из презентации Scale Deep Learning, Jeff Dean

## Вопросы

