

# Extreme Learning Machine: алгоритмическая оптимизация, автоматический подбор параметров

Луценко Александр  
344 гр.  
Научный руководитель  
Невоструев Константин

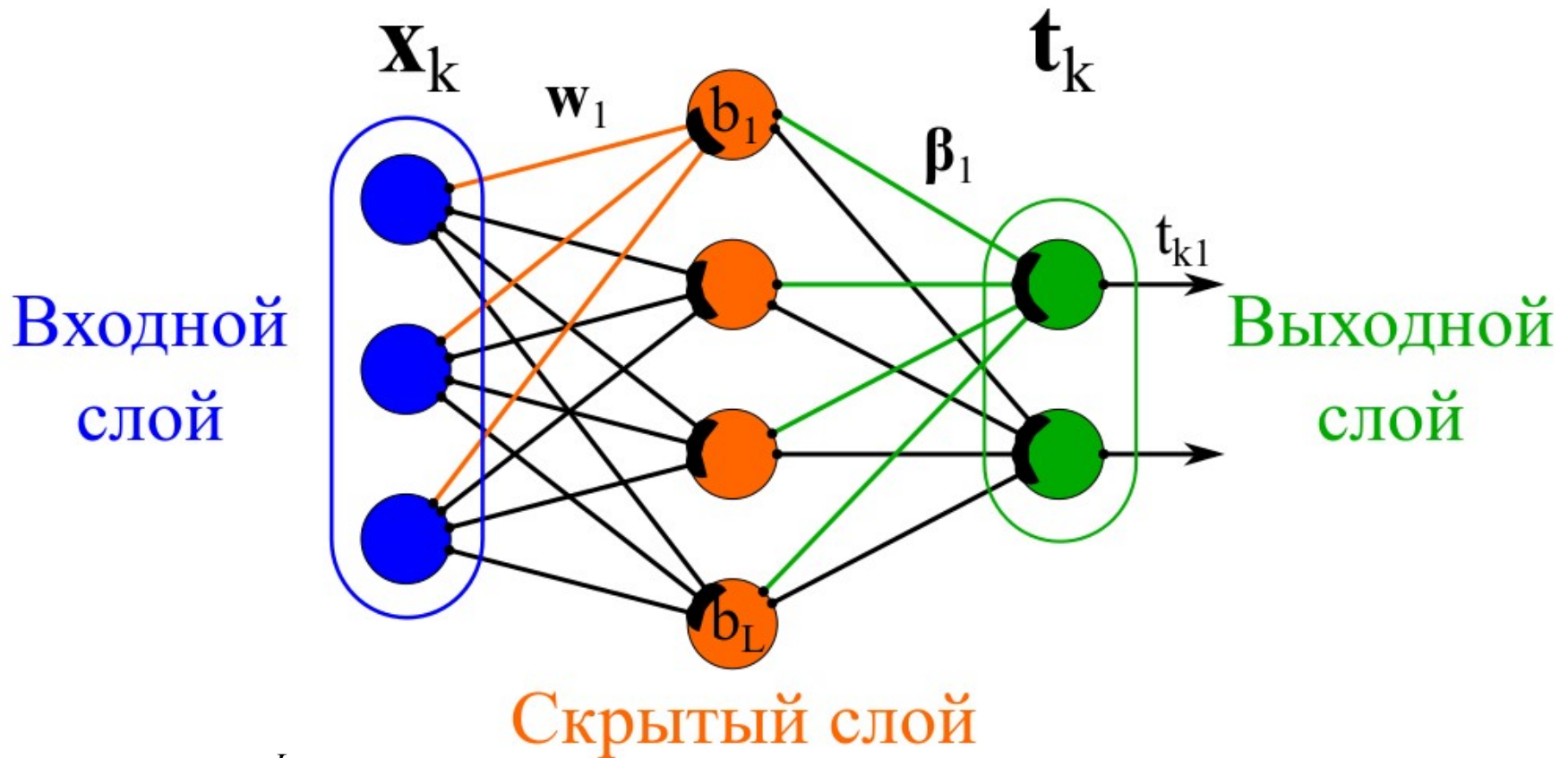
# Глобальная задача

- Конечная цель проекта — создание автоматизированной и максимально простой в использовании системы машинного обучения.

# Текущие задачи

- Реализовать нейронную сеть ELM
- Произвести модификации алгоритма с целью повышения его точности и скорости
- Попытаться найти и реализовать алгоритм автоматического нахождения оптимального числа нейронов.

# ELM



$$t_{k,s} = \sum_{i=1}^L \beta_{s,i} G(\mathbf{w}_i \mathbf{x}_k + b_i), \quad s = 1, \dots, m, \quad k = 1, \dots, N$$

Обучить ELM — значит найти вектора  $\beta_s$

# ELM

$$\mathbf{t}_{k,s} = \sum_{i=1}^L \beta_{s,i} G(\mathbf{w}_i \mathbf{x}_k + b_i), \quad s = 1, \dots, m, \quad k = 1, \dots, N$$

В матричной форме:  $\mathbf{H} \boldsymbol{\beta} = \mathbf{T}$ , где

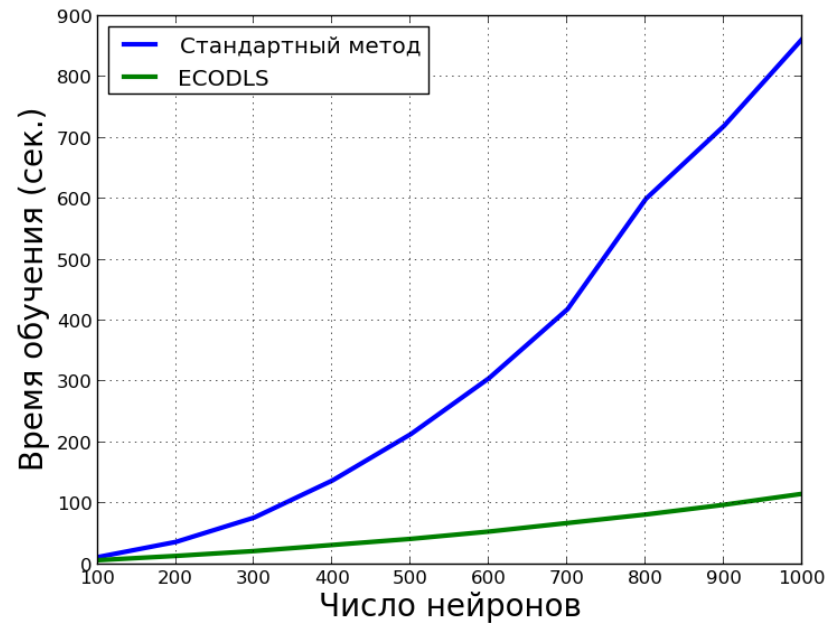
$$\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_m \end{pmatrix}_{L \times m}^T \quad \mathbf{T} = \begin{pmatrix} \mathbf{t}_1 \\ \vdots \\ \mathbf{t}_N \end{pmatrix}_{N \times m}$$

$$\mathbf{H} = \begin{pmatrix} G(\mathbf{w}_1 \mathbf{x}_1 + b_1) & \dots & G(\mathbf{w}_L \mathbf{x}_1 + b_L) \\ \vdots & \ddots & \vdots \\ G(\mathbf{w}_1 \mathbf{x}_N + b_1) & \dots & G(\mathbf{w}_L \mathbf{x}_N + b_L) \end{pmatrix}_{N \times L}$$

Тогда  $\boldsymbol{\beta} = \mathbf{H}^\dagger \mathbf{T}$

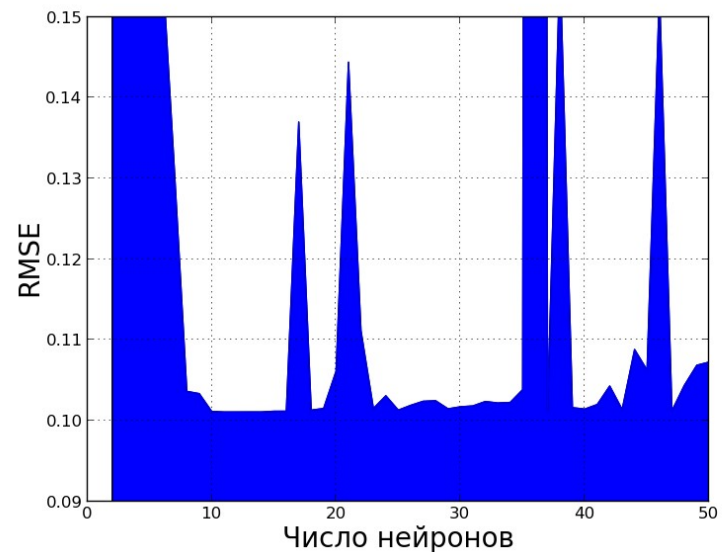
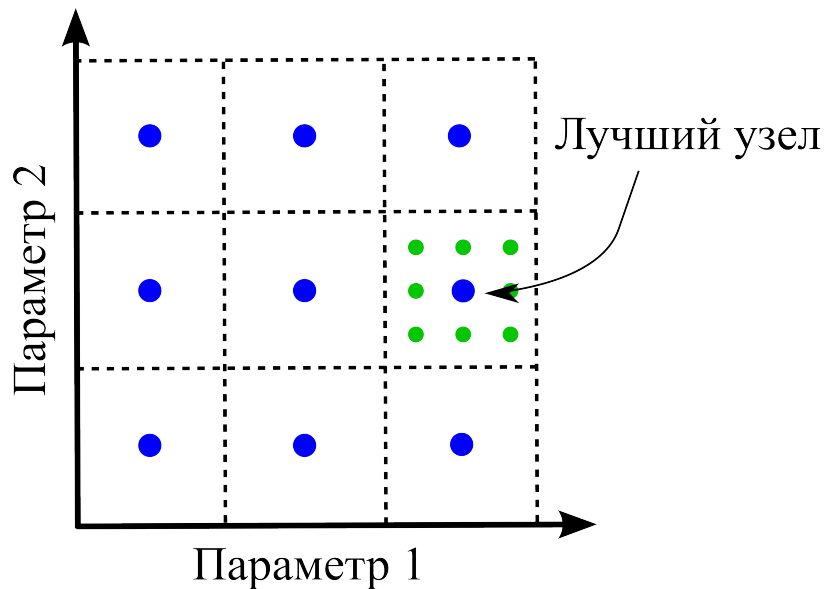
# Улучшение ELM

- Точность
  - RBF-ядро [Huang and Siev 2004]
- Скорость
  - ECODLS [Horata, et al 2013]



# Подбор числа нейронов: Grid Search

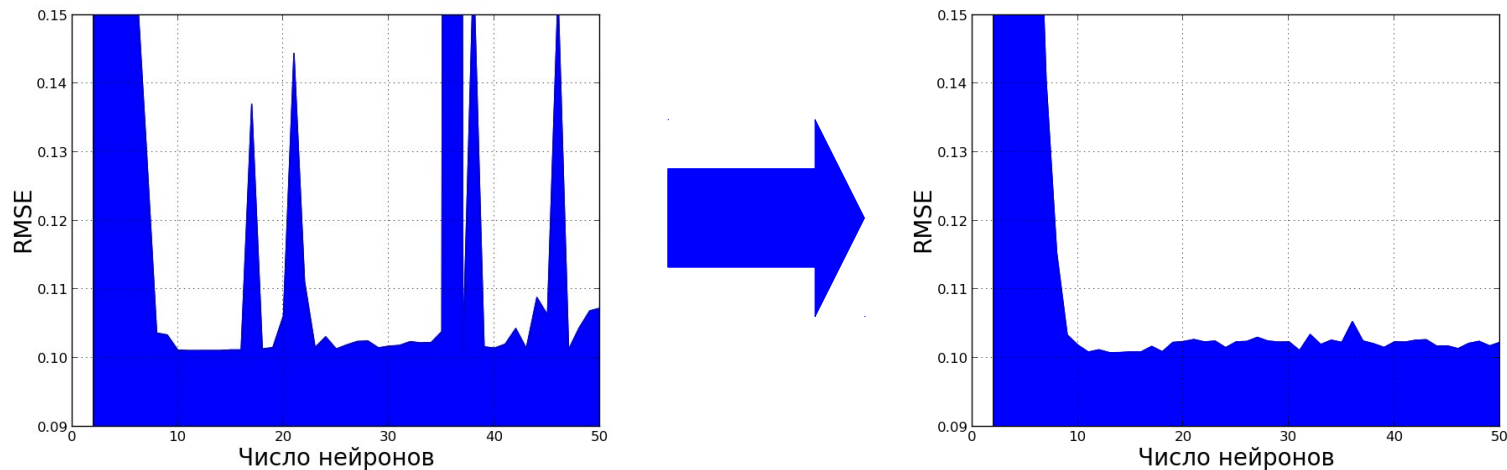
- Самый популярный метод подбора параметров.



- Для ELM неприменим - слишком неровная поверхность поиска

# Подбор числа нейронов: Модифицированный Grid Search

1) Сглаживается поверхность поиска (медиана по  $n$  прогонам)



2) Ищется минимум при помощи Grid Search

В каждом узле решётки ELM не обучается заново, а пересчитывается (из ELM на предыдущем узле)  
[Romero and Alquezar 2006, 2011]



# Подбор числа нейронов: Constructive Selection

- Constructive Selection [Lan, et al 2010]
  - Генерируются  $L_{\max}$  нейронов, а затем сортируются по «полезности»
  - Выбирается такое  $L$ , что «ошибка» нейросети с  $L$  нейронами минимальна
  - На каждой итерации нейросеть не обучается заново, а пересчитывается из предыдущей
- Достоинства
  - Сокращает количество нейронов
  - Достигает лучшей точности, чем Grid Search

# Подбор числа нейронов: Constructive Selection

- Улучшение Constructive Selection
  - Оригинальный алгоритм пересчёта нейронной сети был заменён на предложенный в [Romero and Alquezar 2006, 2011].  
Это позволило использовать CS на больших объёмах данных.

# Результаты тестов: обучение

Набор данных	Размер обучающей выборки	Число нейронов	Время обучения	Время на 1 предсказание	Точность ( $\pm$ Std)	F-score
MNIST	60000	1500	216 сек	1.3 мс	94.8 $\pm$ 0.6%	0.948
		3000	809 сек	1.3 мс	96.1 $\pm$ 0.4%	0.961
Adults	32561	800	39 сек	0.12 мс	85.0 $\pm$ 0.1%	0.648
Poker	25010	2500	232 сек	1.47 мс	72.5 $\pm$ 0.3%	0.693
Titanic	1526	35	0.02 сек	0.03 мс	77.3 + 0.0%	0.521
Diabetes	576	31	0.009 сек	0.04 мс	77.0 $\pm$ 1.4%	0.618

# Результаты тестов: подбор числа нейронов

Набор данных	Алгоритм	Макс. число нейронов	Подобранное число нейронов ( $\pm$ Std)	Время обучения	RMSE ( $\pm$ Std)
Sinc function	ELM	—	16	0.02 сек	0.0044 $\pm$ 0.0002
	Constructive Selection	30	10.3 $\pm$ 0.8	0.7 сек	0.0043 $\pm$ 0.0012
		100	9.9 $\pm$ 0.7	8.9 сек	0.0041 $\pm$ 0.0015
	Модифиц. Grid Search (10 ELMs)	30	23 $\pm$ 12	0.9 сек	0.0043 $\pm$ 0.0053
		100	30 $\pm$ 19	3.0 сек	0.0043 $\pm$ 0.0055

Набор данных	Алгоритм	Макс. число нейронов	Подобранное число нейронов ( $\pm$ Std)	Время обучения	Точность ( $\pm$ Std)	F-score
Diabetes	ELM	—	35	0.01 сек	77.0 $\pm$ 1.4%	0.618
	Constructive Selection	50	23 $\pm$ 8	0.13 сек	77.1 $\pm$ 1.3%	0.629
		100	39 $\pm$ 13	2 сек	77.2 $\pm$ 1.6%	0.635
	Модифиц. Grid Search	50	31 $\pm$ 7	0.3 сек	77.1 $\pm$ 1.4%	0.630
		100	43 $\pm$ 16	0.6 сек	77.1 $\pm$ 1.5%	0.629

# Результаты

- Реализован алгоритм ELM, а также его модификация с RBF
- Произведена алгоритмическая оптимизация, позволившая весьма существенно ускорить процесс обучения
- Подбор количества нейронов
  - Реализован и улучшен метод Constructive Selection (когда предполагаемое количество нейронов невелико)
  - Предложен модифицированный Grid Search (если число нейронов требуется выбрать из широкого диапазона)

**Спасибо!**