

# Выбор предсказательной модели в режиме многозадачного обучения с применением символьных методов

Набиев Мухаммадшариф Фуркатович

Московский физико-технический институт  
Кафедра интеллектуальных систем ФПМИ МФТИ  
Научный руководитель: к.ф.-м.н. Бахтеев Олег Юрьевич

2025

# Цель исследования

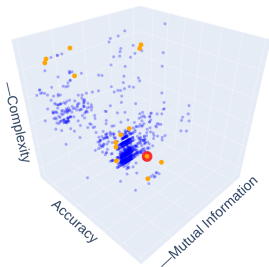
**Проблема:** Выбор оптимальной предсказательной модели сильно зависит от априорного знания человека о природе данных, т.е. от их индуктивного смещения. Определение индуктивного смещения автоматическим образом является открытой проблемой.

*Примером индуктивного смещения могут служить свертки для изображений и временная зависимость для авторегрессии.*

**Цель:** Предложить метод автоматического определения индуктивного смещения.

**Решение:** Построение модели в режиме многозадачного обучения с применением символьной регрессии и дальнейшая ее интерпретация.

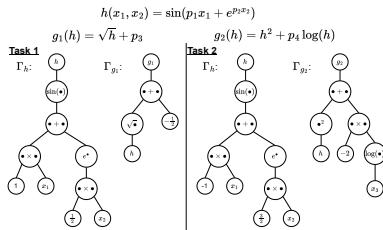
# Архитектура решения



Модели с искомыми критериями.  
Оранжевым выделен Парето-фронт,  
а красным — истинная модель.

Предлагается выбирать модель в режиме многозадачного обучения с учетом информационных критериев.

**Гипотеза:** модель, отражающая индуктивное смещение, располагается на Парето-фронте по следующим критериям: предсказательная способность, длина описания и взаимная информация между исходным объектом выборки и его представлением.



Структура модели в виде дерева разбора.

## Постановка задачи

Пусть  $\mathcal{T} = \{T_i\}_{i=1}^m$  – множество задач. Задаче  $T_i$  соответствует набор данных  $\mathcal{D}_i = (\mathbf{X}^i, \mathbf{y}_i)$ , где  $\mathbf{X}^i \in \mathbb{R}^{N_i \times n}$ , а  $\mathbf{y}^i \in \mathcal{Y}$ . Для регрессии  $\mathcal{Y} = \mathbb{R}^{N_i}$  а для классификации  $\mathcal{Y} = \{1, \dots, K\}^{N_i}$ .

- ▶ Моделью  $\mathbf{f}(\mathbf{x}; \mathbf{w})$  назовем отображение  $\mathbb{R}^n \times \mathbb{R}^l \rightarrow \mathcal{Y}^m$ , структура  $\Gamma_{\mathbf{f}}$  модели которой представима в виде дерева разбора символьного выражения.
- ▶ Декомпозируем модель как  $\mathbf{f} = \mathbf{g} \circ \mathbf{h} = (g_1 \circ \mathbf{h}, \dots, g_m \circ \mathbf{h})$ . Модели  $\mathbf{h}$  и  $\mathbf{g}$  мы назовем *энкодером* и *декодером* соответственно.
- ▶ Назовем *индуктивным смещением* для декомпозируемой модели структуру первой части  $\Gamma_{\mathbf{h}}$  модели  $\mathbf{f}$ .

### Утверждение (Набиев, 2025)

Для любой модели  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathcal{Y}^m$  существует такая модель  $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^d$  и  $\mathbf{g} : \mathbb{R}^d \rightarrow \mathcal{Y}^m$ , что  $\mathbf{f} = \mathbf{g} \circ \mathbf{h}$ .

# Постановка задачи

## Теорема (Набиев, 2025)

Для скрытого пространства  $\mathbb{R}^d$ , такого что  $d \geq m$ , существует разложение  $f = g \circ h$ , где  $g = \text{Id}_Z$ , а  $Z \cong \mathcal{Y}^m$ .

Энкодер  $\mathbf{h}$  задает достаточную статистику относительно  $\mathbf{y}$ , если  $I(\mathbf{h}(\mathbf{X}), \mathbf{y}) = I(\mathbf{X}, \mathbf{y})$ .

**Принцип Information Bottleneck (IB)** — приближение минимальной достаточной статистики:

$$\min_{p(\mathbf{h}(\mathbf{x})|\mathbf{x})} I(\mathbf{X}, \mathbf{h}(\mathbf{X})) - \beta I(\mathbf{h}(\mathbf{X}), \mathbf{y}),$$

где параметр  $\beta$  задаёт баланс между сжатием и значимостью.

Для выполнения условия достаточности минимизируем  $I(\mathbf{X}, \mathbf{h}(\mathbf{X}))$ :

$$I(\mathbf{X}, \mathbf{h}(\mathbf{X})) - \beta I(\mathbf{h}(\mathbf{X}), \mathbf{y}) \approx I(\mathbf{X}, \mathbf{h}(\mathbf{X})) - \beta' \cdot L(f_i(\mathbf{X}), \mathbf{y}),$$

где  $L$  — функция потерь.

## Постановка задачи

Задача оптимизации принимает вид:

$$\Gamma_{\mathbf{h}} = \arg \min_{\mathbf{f}=(\mathbf{g}_1 \circ \mathbf{h}, \dots, \mathbf{g}_m \circ \mathbf{h})} \frac{1}{m} \sum_{i=1}^m L_i(f_i(\mathbf{X}^i; \mathbf{w}_i^*), \mathbf{y}^i) + \lambda_1 I(\mathbf{X}^i, \mathbf{h}(\mathbf{X}^i)) + \lambda_2 C(\mathbf{f}),$$
$$\text{s.t. } \mathbf{w}_i^* = \arg \min_{\mathbf{w}} L_i(f_i(\mathbf{X}^i; \mathbf{w}), \mathbf{y}^i)$$

где  $C : \mathfrak{F} \rightarrow \mathbb{R}$  — длина описания модели, с помощью кодирования Хаффмана.

### Теорема (Набиев, 2025)

*Пусть кол-во задач  $m = 1$ . Тогда существует решение  $f$ , которое можно разложить как  $f = g \circ h$ , где  $g = \text{Id}$ , а  $h = f$ .*

### Следствие

*Пусть имеется  $m > 1$  задач, и пусть каждая задача  $T_i$  аппроксимируется функцией  $f_i(\mathbf{x}) = [\mathbf{f}(\mathbf{x}, \mathbf{w}_i)]_i$ , где:  $\Gamma_{g_i} = \Gamma_{g_j}$  для всех  $i, j \in \{1, \dots, m\}$ . Тогда существует декомпозиция вида  $\mathbf{h}(\mathbf{x}, \mathbf{w}) = \mathbf{f}(\mathbf{x}; \mathbf{w})$ ,  $\mathbf{g}(\mathbf{x}) = \mathbf{x}$ , такая что для каждой задачи  $f_i(\mathbf{x}) = (\mathbf{g} \circ \mathbf{h})(\mathbf{x}; \mathbf{w}_i) = \mathbf{h}(\mathbf{x}; \mathbf{w}_i)$ .*

# Данные для эксперимента

Эксперимент проводился на четырех типов выборок:

1. **Окружность:** 3 задачи: данные — точки окружности. Задачи имеют вид:  $[r > 0]$ ,  $r + 0.5$  и  $\sqrt{r} - 0.5$ . Оптимальная модель — кривая второго порядка.
2. **Авторегрессия:** Задачи вида:  $y_t = \alpha y_{t-1} + \varepsilon$ . Модели оценивались по взаимной информации и длине описания.
3. **4-пикселя:** 8 задач: 6 бинарных классификаций и 2 регрессии. Оптимальная модель — использует локальную пространственную информацию.
4. **MNIST:** Сравнивались модели с масками, имитирующими свёртки, и случайными масками.

**Гипотеза:** структура энкодера будет содержать операции характерные для заданного типа выборки.

# Результаты эксперимента: окружность и 4-пикселя

## Окружность:

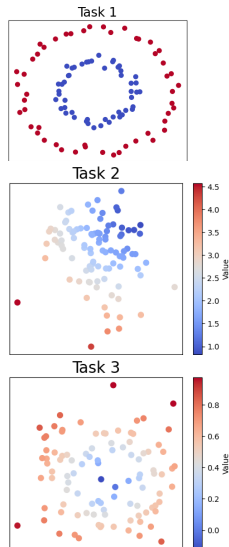
1.  $h_0 = p_1, h_1 = x_0^2 + x_1^2$
2.  $h_0 = (p_2 x_0 + x_1)^2, h_1 = p_1$
3.  $h_0 = x_0^2 + x_1^2, h_1 = p_0$
4.  $h_0 = p_2, h_1 = (x_1 + p_0 x_0)^2$
5.  $h_0 = (x_0 + x_1)^2, h_1 = p_2$

**Вывод:** Модели 1, 3 и 5 демонстрируют высокую точность при минимальной сложности, соответствующая структуре задачи, а 2 и 4 не доминируют.

## 4-пикселя:

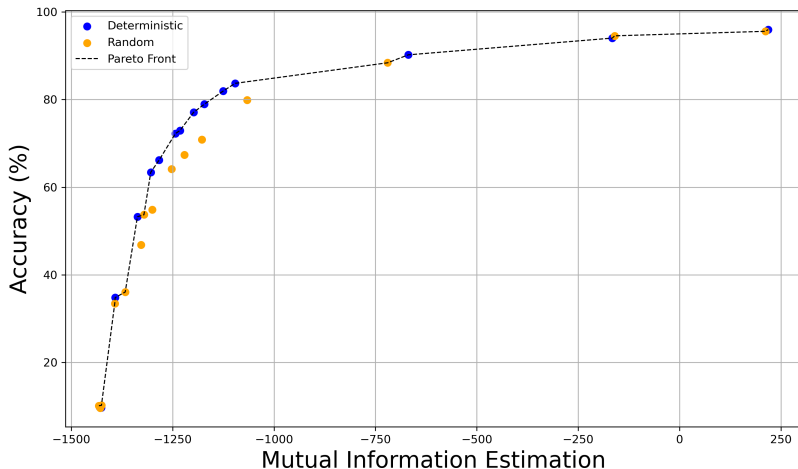
1.  $h_0 = p_3 x_1 - x_0 - x_2 - p_5, h_1 = p_2 x_2 + x_3$
2.  $h_0 = x_1, h_1 = \max(p_4, x_2 - x_0) + p_5 x_2 - x_3$
3.  $h_0 = (x_1 + p_2)(p_3 x_3 + p_1 x_2), h_1 = x_1 + p_1 x_0$
4.  $h_0 = p_3 x_1 - x_3(p_0 + x_2), h_1 = x_0 + p_2 x_2$
5.  $h_0 = p_3(x_2 - p_4)(p_2 + x_0 + x_1), h_1 = x_1(p_1 - x_3)$

**Вывод:** Все модели, кроме 2, локальны (учитывает пиксель и соседей в радиусе 2) и хорошо сбалансированы по точности, длине описания и взаимной информации.





# Результаты эксперимента: MNIST



**Вывод:** Модели учитывающие локальность достигают более высокой точности при заданной взаимной информации, т.е. они лежат на Парето-фронте.

## Выносятся на защиту

1. Предложен критерий выбора модели для определения индуктивного смещения для заданного множества задач.
  2. Приведены теоретические обоснования для предложенного метода.
  3. Проведены эксперименты, которые показали, что модели с оптимальным соотношением качества, информации и сжатия располагаются на Парето-фронте.
- ▶ Результаты были доложены на 67-й конференции МФТИ.
  - ▶ Планируется подача работы в рецензируемый журнал.

- [1] Rich Caruana. Multitask learning: a knowledge-based source of inductive bias. In *Proceedings of the Tenth International Conference on International Conference on Machine Learning*, ICML'93, page 41–48, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
- [2] Cranmer et al. Discovering symbolic models from deep learning with inductive biases. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.
- [3] A.S. Kulunchakov and V.V. Strijov. Generation of simple structured information retrieval functions by genetic algorithm without stagnation. *Expert Systems with Applications*, 85:221–230, 2017.
- [4] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5, 2015.

## Приложение: таблица для окружности

	#	Accuracy	MSE <sub>1</sub>	MSE <sub>2</sub>	$I(h(\mathbf{X}), \mathbf{X})$	C
No reg.	1	1.00	$4.63 \cdot 10^{-2}$	$2.22 \cdot 10^{-2}$	2.98	22
	2	1.00	$6.52 \cdot 10^{-2}$	$3.60 \cdot 10^{-3}$	4.20	34
	3	0.82	$3.56 \cdot 10^{-2}$	$4.68 \cdot 10^{-3}$	4.82	18
	4	1.00	$2.10 \cdot 10^{-2}$	$6.24 \cdot 10^{-3}$	2.98	43
	5	0.57	$2.62 \cdot 10^{-2}$	$3.90 \cdot 10^{-2}$	2.13	38
Reg.	6	0.83	$4.08 \cdot 10^{-14}$	$2.26 \cdot 10^{-15}$	1.85	12
	7	0.80	$6.36 \cdot 10^{-2}$	$2.11 \cdot 10^{-2}$	1.56	16
	8	0.77	$7.14 \cdot 10^{-16}$	$1.14 \cdot 10^{-4}$	1.86	12
	9	0.78	$6.33 \cdot 10^{-2}$	$2.12 \cdot 10^{-2}$	1.56	16
	10	0.80	$6.57 \cdot 10^{-2}$	$2.80 \cdot 10^{-2}$	1.86	10

Найденные модели и метрики для задачи окружности.

## Приложение: таблицы для 4-пикселя и авторегрессии

#	Average accuracy	Average MSE	$l(h(\mathbf{X}), \mathbf{X})$	C
1	0.98	$3.5 \cdot 10^{-3}$	9.57	36
2	0.92	$8.19 \cdot 10^{-4}$	9.14	30
3	0.97	$5.96 \cdot 10^{-3}$	8.58	43
4	0.94	$1.67 \cdot 10^{-13}$	9.96	36
5	0.94	$3.93 \cdot 10^{-2}$	8.30	45

Найденные модели и метрики для 4-пикселя.

#Datasets	Models	$l(h(\mathbf{X}), \mathbf{X})$	C
1, 2, 5	$p_0 x_t$	5.83	5
10	$p_0 \sin^2(x_t) x_t$	3.14	18

Найденные модели и метрики для авторегрессии.

# Приложение: модели для окружности

## Model 1:

$$h_0 = x_0(x_1 + x_1),$$

$$h_1 = (x_0 + p_0) + x_1,$$

$$g_0 = (p_1 + h_1^2 p_0^2) + p_0(p_1 + p_1)(h_0 + h_0),$$

$$g_1 = p_1 h_0 + p_1 + h_1 p_2,$$

$$g_2 = p_2 + (p_1 + h_0)(h_0 p_0).$$

## Model 2:

$$h_0 = (x_0 + p_1)p_0,$$

$$h_1 = \sqrt{x_1^2} + p_2 p_0,$$

$$g_0 = (p_0 + h_0^2) + (p_1^4 + (h_1 + p_0)h_1),$$

$$g_1 = h_0^2 + h_1,$$

$$g_2 = p_2 h_1 + \sqrt{(h_0 + h_0)^2}.$$

## Приложение: модели для окружности

### Model 3:

$$h_0 = p_1(p_2 + p_0 + x_0),$$

$$h_1 = x_1,$$

$$g_0 = h_1 p_0 + (h_1 h_0 + p_2^2),$$

$$g_1 = h_0^2 + p_0^2 + (p_1 p_2 + p_1) + p_2 h_1^2,$$

$$g_2 = (p_2 + h_0^2 + \sqrt{h_1^2}) p_2.$$

### Model 4:

$$h_0 = p_2 x_0 x_1,$$

$$h_1 = x_1 + (2p_2 + p_0 + x_0 + p_1),$$

$$g_0 = (h_0 + p_0 + h_0) p_0 + h_1^2,$$

$$g_1 = (p_1 + h_0)^4 + p_0 p_1 h_1,$$

$$g_2 = 2p_1 h_1^2 + h_0 + p_2^2.$$

## Приложение: модели для окружности

### Model 5:

$$h_0 = x_0 + x_1 x_0 p_1 + x_1 + p_0,$$

$$h_1 = p_2^2,$$

$$g_0 = h_0(h_0 + p_0) + (h_0 + p_0)^2 + 2h_1,$$

$$g_1 = h_1 + 4p_0^2 h_0^2 p_1^2,$$

$$g_2 = p_1.$$

### Model 6:

$$h_0 = p_1,$$

$$h_1 = x_0^2 + x_1^2,$$

$$g_0 = (h_0 h_1 + 2p_0) h_1^2 + h_0,$$

$$g_1 = p_0 + \sqrt{h_1} + p_1 p_2 p_0 + p_0^2 p_1 + p_1,$$

$$g_2 = p_0 + \sqrt{h_0^2 \sqrt{h_1}}.$$



## Приложение: модели для окружности

### Model 7:

$$h_0 = (p_2 x_0 + x_1)^2,$$

$$h_1 = p_1,$$

$$g_0 = h_0 + h_1 + 4h_0^2(h_0 + p_0)^2,$$

$$g_1 = p_2 p_1 + \sqrt{h_0} p_1 + p_0,$$

$$g_2 = p_0 + h_0(h_1 + p_2 h_0).$$

### Model 8:

$$h_0 = x_1^2 + x_0^2,$$

$$h_1 = p_0,$$

$$g_0 = h_1 h_0 + p_1(h_0 + p_1) + h_0^2 p_0^2,$$

$$g_1 = \sqrt{h_0} + h_1,$$

$$g_2 = p_2(h_0 p_0 + p_1 + \sqrt{h_0}).$$

## Приложение: модели для окружности

### Model 9:

$$h_0 = p_2,$$

$$h_1 = (x_1 + x_0 p_0)^2,$$

$$g_0 = (p_2^2 + p_0) h_1^2 p_2^2 + 2h_1 + p_2,$$

$$g_1 = p_2 + \sqrt{h_1} (p_2^2 + p_1 h_1),$$

$$g_2 = h_0 \sqrt{p_1 + h_1 + h_0^2}.$$

### Model 10:

$$h_0 = (x_1 + x_0)^2,$$

$$h_1 = p_2,$$

$$g_0 = (p_1 + h_0) (h_1 + p_0 + h_1 h_0) + 2p_1,$$

$$g_1 = \sqrt{h_0} h_1^2 + h_1 h_0 p_0 + p_1,$$

$$g_2 = h_0 p_0 + p_1 (h_1 + p_2) \sqrt{h_0}.$$

## Приложение: модели для 4-пикселя

### Model 1:

$$h_0 = x_1 p_3 - x_2 - p_5 - x_0,$$

$$h_1 = x_2 p_2 + x_3,$$

$$g_0 = (h_0 + p_1 h_0 - h_1 h_0 - \max(h_1, p_0)) (\max(h_1 h_0, p_4 + h_1) + h_0 + p_4 p_1)$$

$$g_1 = h_0 (p_1 (p_4 - h_1) - \max(p_3, h_0)).$$

### Model 2:

$$h_0 = x_1,$$

$$h_1 = \max(p_4, x_2 - x_0) + x_2 p_5 - x_3,$$

$$g_0 = \max(h_1, \max(p_5, h_1)) - \max(p_3, h_1 p_0) - h_0 p_2 (h_1 - h_0),$$

$$g_1 = \max(h_1 - p_5 - p_0, p_1) + p_1 + 2p_3 h_0 (p_3 + h_1) (p_0 + p_5).$$

## Приложение: модели для 4-пикселя

### Model 3:

$$h_0 = (x_1 + p_2)(p_3x_3 + p_1x_2),$$

$$h_1 = x_1 + p_1x_0,$$

$$g_0 = \max(h_1 + p_0 - p_5h_0, p_5 - p_4),$$

$$g_1 = -p_4 + p_5 - \max(h_1, h_1) - (h_0 + p_2) + p_0.$$

### Model 4:

$$h_0 = x_1p_3 - x_3(p_0 + x_2),$$

$$h_1 = x_0 + x_2p_2,$$

$$g_0 = h_0(h_0 + h_1) + (h_0 + p_4) - p_1p_5 \max(h_1, h_1),$$

$$g_1 = (p_1p_2 + p_4 + h_0 + h_1) \max(p_1p_4, \max(h_1h_0, p_4h_0)).$$

## Приложение: модели для 4-пикселя

### Model 5:

$$h_0 = p_3(x_2 - p_4)(p_2 + x_0 + x_1),$$

$$h_1 = x_1(p_1 - x_3),$$

$$g_0 = (p_5 - h_0)(p_0 + h_1) - p_4,$$

$$g_1 = \max(h_0 + h_1, h_0) + \max(h_1 p_5, p_4) - (h_1^2 + p_3).$$