

# Градиентный бустинг

Харинаев Артём

316 МС ВМК МГУ

29.11.2021

Booking.com Яндекc  
NETFLIX

# Интуитивные соображения

$Y = \mathbb{R}$ ,  $X^\ell$ -обучающая выборка

- ❶  $y$  - целевой признак,  $b_1(x)$  - решающее дерево, настроенное на него
  - ❷  $y - b_1(x)$  - целевой признак,  $b_2(X)$  - решающее дерево, настроенное на него
  - ❸  $y - b_1(x) - b_2(x)$  - целевой признак,  $b_3(X)$  - решающее дерево, настроенное на него
- и т.д.

$a_T(x) = \sum_{t=1}^T b_t(x)$  - композиция алгоритмов

$$\mathcal{L}(y, a(x)) = \frac{1}{2} \|y - a(x)\|^2$$

$$\nabla \mathcal{L}(y, a(x)) = y - a(x)$$

$X$  - пространство объектов,  $Y$  - пространство ответов,  $X^\ell = \{x_i, y_i\}_{i=1}^\ell$  - обучающая выборка,  $\mathcal{L}$  - дифференцируемая функция потерь,  $\mathfrak{B}$  - пространство базовых алгоритмов

$$a_T(x) = \sum_{t=1}^T b_t(x), \quad x \in X, \quad b_t : X \mapsto Y, \quad b_t \in \mathfrak{B}$$

$$a_T(x) = a_{T-1}(x) + b_T(x)$$

$$b_T(x) = \operatorname{argmin}_{b \in \mathfrak{B}} \sum_{i=1}^\ell \mathcal{L}(y_i, a_{T-1}(x_i) + b(x_i))$$

$$b_1(x) = \operatorname{argmin}_{b \in \mathfrak{B}} \sum_{i=1}^\ell \mathcal{L}(y_i, b(x_i))$$

Разложение Тейлора функции потерь до первого члена в окрестности  $(y_i, a_{T-1}(x_i))$  :

$$\begin{aligned}\mathcal{L}(y_i, a_{T-1}(x_i) + b(x_i)) &\approx \mathcal{L}(y_i, a_{T-1}(x_i)) + b(x_i) \frac{\partial \mathcal{L}(y_i, z)}{\partial z} \bigg|_{z=a_{T-1}(x_i)} = \\ &= \mathcal{L}(y_i, a_{T-1}(x_i)) + b(x_i) g_i^{T-1}\end{aligned}$$

Получается следующая оптимизационная задача:

$$b_T \approx \operatorname{argmin}_{b \in \mathcal{B}} \sum_{i=1}^{\ell} b(x_i) g_i^{T-1}$$

решением которой является антиградиент  $-g^{T-1}$ .

На каждой итерации базовые алгоритмы обучаются предсказывать значения антиградиента функции потерь по текущим предсказаниям композиции

# Постановка задачи

- Сравнить качество алгоритма градиентного бустинга с линейными моделями на задачах
  - классификации
  - регрессии
- Исследовать зависимости в данных "Ценообразование недвижимости" ([www.kaggle.com/c/house-prices-advanced-regression-techniques](https://www.kaggle.com/c/house-prices-advanced-regression-techniques))

## Классификация "Болезни сердца"

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

<https://www.kaggle.com/ronitf/heart-disease-uci>

## Регрессия "Ценообразование недвижимости"

	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	...	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
Id													
1	60	RL	65.0	8450	Pave	NaN	...	0	2	2008	WD	Normal	208500
2	20	RL	80.0	9600	Pave	NaN	...	0	5	2007	WD	Normal	181500
3	60	RL	68.0	11250	Pave	NaN	...	0	9	2008	WD	Normal	223500
4	70	RL	60.0	9550	Pave	NaN	...	0	2	2006	WD	Abnorml	140000
5	60	RL	84.0	14260	Pave	NaN	...	0	12	2008	WD	Normal	250000
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1456	60	RL	62.0	7917	Pave	NaN	...	0	8	2007	WD	Normal	175000
1457	20	RL	85.0	13175	Pave	NaN	...	0	2	2010	WD	Normal	210000
1458	70	RL	66.0	9042	Pave	NaN	...	2500	5	2010	WD	Normal	266500
1459	20	RL	68.0	9717	Pave	NaN	...	0	4	2010	WD	Normal	142125
1460	20	RL	75.0	9937	Pave	NaN	...	0	6	2008	WD	Normal	147500

1460 rows x 80 columns

<https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data>



# Предобработка данных

Для обучения линейных моделей необходимо сделать достаточно большую работу по предобработке данных:

- Обработать недостающие значения (заполнить или удалить строки, содержащие их)
- Преобразовать текстовые данные в числовые
- Закодировать категориальные данные (one-hot-encoding, target mean, label encoder)
- Для лучшей сходимости нужно масштабировать данные (к  $\mathcal{N}(0, 1)$ , min-max)
- Выявить и удалить выбросы

Для многих реализаций градиентного бустинга, таких как XGBoost, LightGBM, CatBoost, не обязательно выполнять все эти шаги.



Yandex  
CatBoost

Логистическая регрессия:

Recall = 0.814

	Болен	Здоров
Болен	23	10
Здоров	8	35

Таблица: Матрица ошибок

CatBoost:

Recall = 0.860

	Болен	Здоров
Болен	25	8
Здоров	6	37

Таблица: Матрица ошибок

Ridge-регрессия

$\text{RMSLE} = 0.13748$

CatBoost "из коробки"

$\text{RMSLE} = 0.12948$

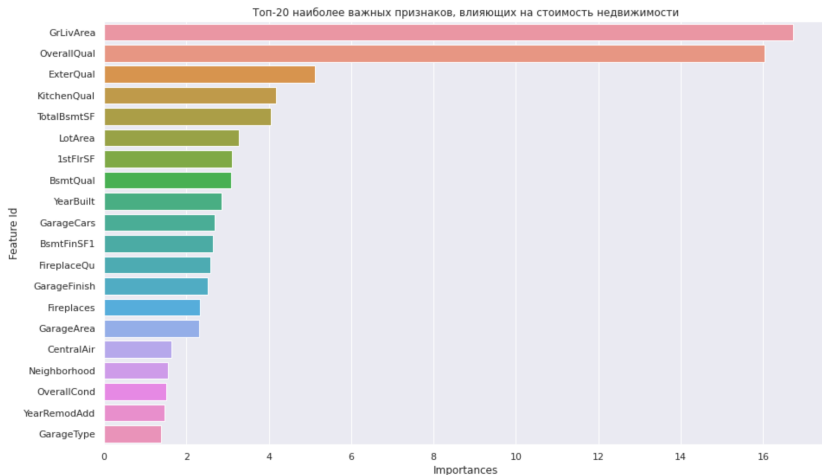
CatBoost с оптимальными параметрами

$\text{RMSLE} = 0.12420$

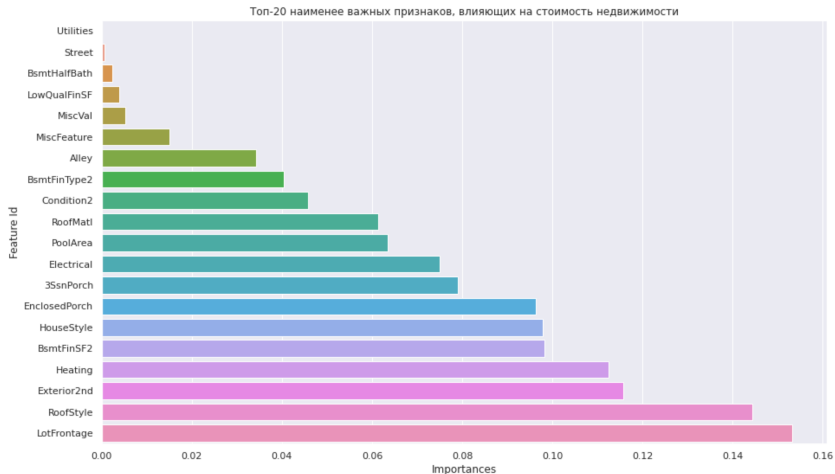
# Важность признаков



# Важность признаков



# Важность признаков



Преимущества градиентного бустинга в сравнении с линейными моделями

- Более высокая точность
- Гибкая настройка гиперпараметров
- Более адекватная оценка важности признаков
- Меньшая зависимость от преобработки данных

Недостатки градиентного бустинга в сравнении с линейными моделями

- Скорость обучения