

# Введение в искусственный интеллект.

## Машинное обучение

### Семинар 9. Примеры реализации бустинга

Бабин Д.Н., Иванов И.Е., Петюшко А.А.

кафедра Математической Теории Интеллектуальных Систем

1 декабря 2020 г.



## 1 AdaBoost — значение алгоритма



- 1 AdaBoost — значение алгоритма
- 2 Практические примеры градиентного бустинга на решающих деревьях:



- 1 AdaBoost — значение алгоритма
- 2 Практические примеры градиентного бустинга на решающих деревьях:
  - 1 XGBoost



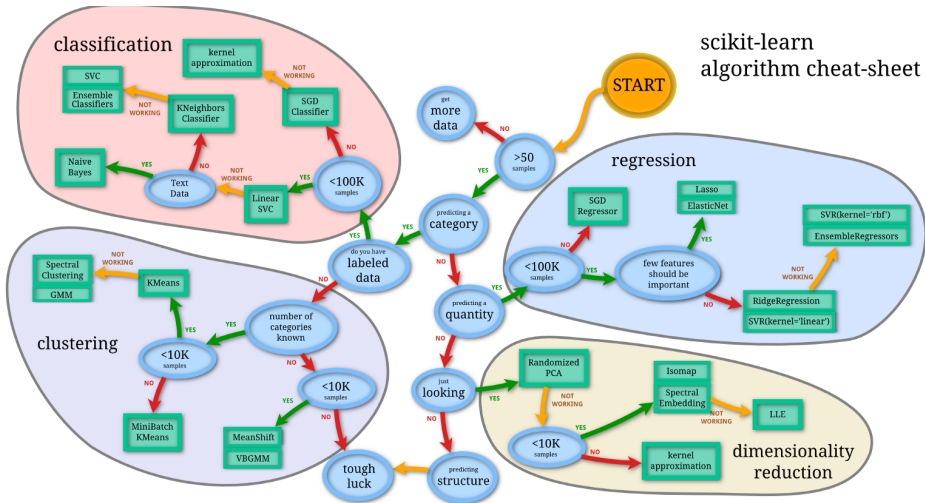
- ① AdaBoost — значение алгоритма
- ② Практические примеры градиентного бустинга на решающих деревьях:
  - ① XGBoost
  - ② LightGBM



- ① AdaBoost — значение алгоритма
- ② Практические примеры градиентного бустинга на решающих деревьях:
  - ① XGBoost
  - ② LightGBM
  - ③ CatBoost

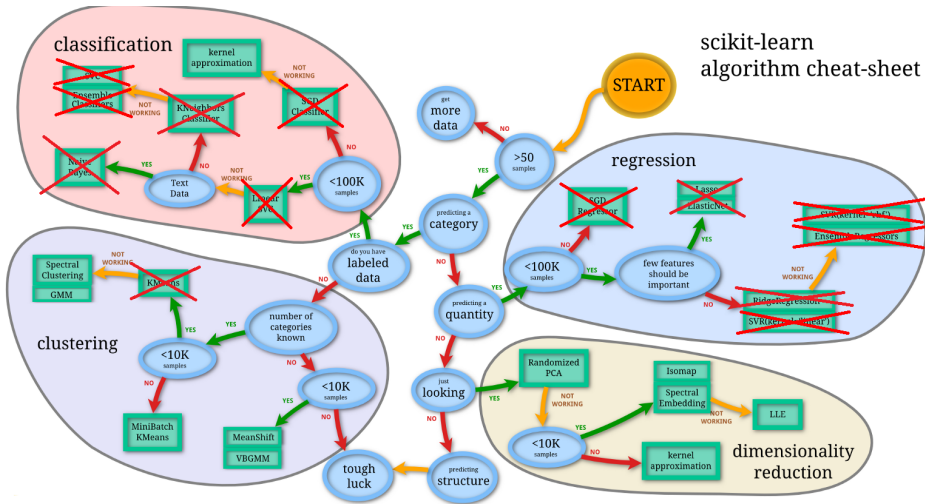


# Дорожная карта Scikit-Learn<sup>1</sup>



<sup>1</sup>[https://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/](https://scikit-learn.org/stable/tutorial/machine_learning_map/)

# Дорожная карта Scikit-Learn<sup>1</sup>



<sup>1</sup>[https://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/](https://scikit-learn.org/stable/tutorial/machine_learning_map/)



# О значении алгоритма AdaBoost

- В 2003 году создатели алгоритма AdaBoost Фройнд и Шапире получили премию Гёделя (вручается за выдающиеся труды по логике и теоретической информатике),

---

<sup>2</sup>Viola and Jones (2001). "Robust Real-time Object Detection"

# О значении алгоритма AdaBoost

- В 2003 году создатели алгоритма AdaBoost Фройнд и Шапире получили премию Гёделя (вручается за выдающиеся труды по логике и теоретической информатике),
- В 2001 году<sup>2</sup> был создан алгоритм, позволяющий обнаруживать объекты на изображениях (прежде всего человеческое лицо) в реальном времени.

---

<sup>2</sup>Viola and Jones (2001). "Robust Real-time Object Detection"

# О значении алгоритма AdaBoost

- В 2003 году создатели алгоритма AdaBoost Фройнд и Шапире получили премию Гёделя (вручается за выдающиеся труды по логике и теоретической информатике),
- В 2001 году<sup>2</sup> был создан алгоритм, позволяющий обнаруживать объекты на изображениях (прежде всего человеческое лицо) в реальном времени.
  - Математической основой послужил модифицированный AdaBoost,

---

<sup>2</sup>Viola and Jones (2001). "Robust Real-time Object Detection"

# О значении алгоритма AdaBoost

- В 2003 году создатели алгоритма AdaBoost Фройнд и Шапире получили премию Гёделя (вручается за выдающиеся труды по логике и теоретической информатике),
- В 2001 году<sup>2</sup> был создан алгоритм, позволяющий обнаруживать объекты на изображениях (прежде всего человеческое лицо) в реальном времени.
  - Математической основой послужил модифицированный AdaBoost,
  - Этот алгоритм детекции был лидирующим для детекции лиц на протяжении более 10 лет (до начала широкого применения сверточных нейросетей).

---

<sup>2</sup>Viola and Jones (2001). "Robust Real-time Object Detection"



# О значении алгоритма AdaBoost

- В 2003 году создатели алгоритма AdaBoost Фройнд и Шапире получили премию Гёделя (вручается за выдающиеся труды по логике и теоретической информатике),
- В 2001 году<sup>2</sup> был создан алгоритм, позволяющий обнаруживать объекты на изображениях (прежде всего человеческое лицо) в реальном времени.
  - Математической основой послужил модифицированный AdaBoost,
  - Этот алгоритм детекции был лидирующим для детекции лиц на протяжении более 10 лет (до начала широкого применения сверточных нейросетей).



<sup>2</sup>Viola and Jones (2001). "Robust Real-time Object Detection"

# Алгоритмы XGBoost, LightGBM и CatBoost

Наибольшую популярность на данный момент имеют реализации градиентного бустинга на решающих деревьях<sup>3</sup>:

---

<sup>3</sup><https://towardsdatascience.com/catboost-vs-light-gbm-vs-xgboost-5f93620723db>

<sup>4</sup>Chen, T., and Guestrin, C. (2016). "Xgboost: A scalable tree boosting system".

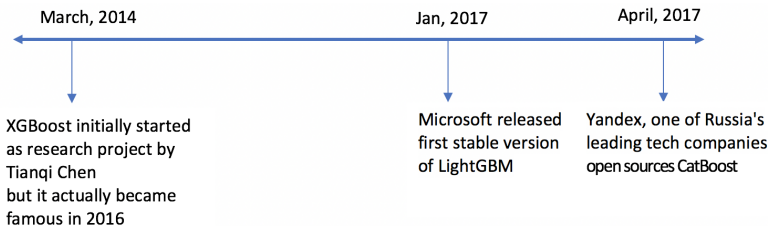
<sup>5</sup>Ke, G. et al. (2017). "Lightgbm: A highly efficient gradient boosting decision tree".

<sup>6</sup>Prokhorenkova, L., et al. (2017). "CatBoost: unbiased boosting with categorical features".



# Алгоритмы XGBoost, LightGBM и CatBoost

Наибольшую популярность на данный момент имеют реализации градиентного бустинга на решающих деревьях<sup>3</sup>:



<sup>3</sup><https://towardsdatascience.com/catboost-vs-light-gbm-vs-xgboost-5f93620723db>

<sup>4</sup>Chen, T., and Guestrin, C. (2016). "Xgboost: A scalable tree boosting system".

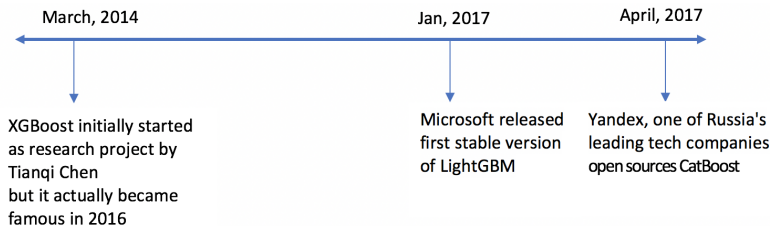
<sup>5</sup>Ke, G. et al. (2017). "Lightgbm: A highly efficient gradient boosting decision tree".

<sup>6</sup>Prokhorenkova, L., et al. (2017). "CatBoost: unbiased boosting with categorical features".



# Алгоритмы XGBoost, LightGBM и CatBoost

Наибольшую популярность на данный момент имеют реализации градиентного бустинга на решающих деревьях<sup>3</sup>:



Одни из наиболее известных на данный момент - это XGBoost<sup>4</sup>, LightGBM<sup>5</sup> и CatBoost<sup>6</sup>.

<sup>3</sup><https://towardsdatascience.com/catboost-vs-light-gbm-vs-xgboost-5f93620723db>

<sup>4</sup>Chen, T., and Guestrin, C. (2016). "Xgboost: A scalable tree boosting system".

<sup>5</sup>Ke, G. et al. (2017). "Lightgbm: A highly efficient gradient boosting decision tree".

<sup>6</sup>Prokhorenkova, L., et al. (2017). "CatBoost: unbiased boosting with categorical features".



# Алгоритмы XGBoost, LightGBM и CatBoost

Эти реализации отличаются:

- Методом ветвления в узлах дерева при его обучении,



# Алгоритмы XGBoost, LightGBM и CatBoost

Эти реализации отличаются:

- Методом ветвления в узлах дерева при его обучении,
- Способом работы с категориальными признаками (например, XGBoost не умеет с ними работать),



# Алгоритмы XGBoost, LightGBM и CatBoost

Эти реализации отличаются:

- Методом ветвления в узлах дерева при его обучении,
- Способом работы с категориальными признаками (например, XGBoost не умеет с ними работать),
- Скоростью обучения / тестирования.



# Алгоритмы XGBoost, LightGBM и CatBoost

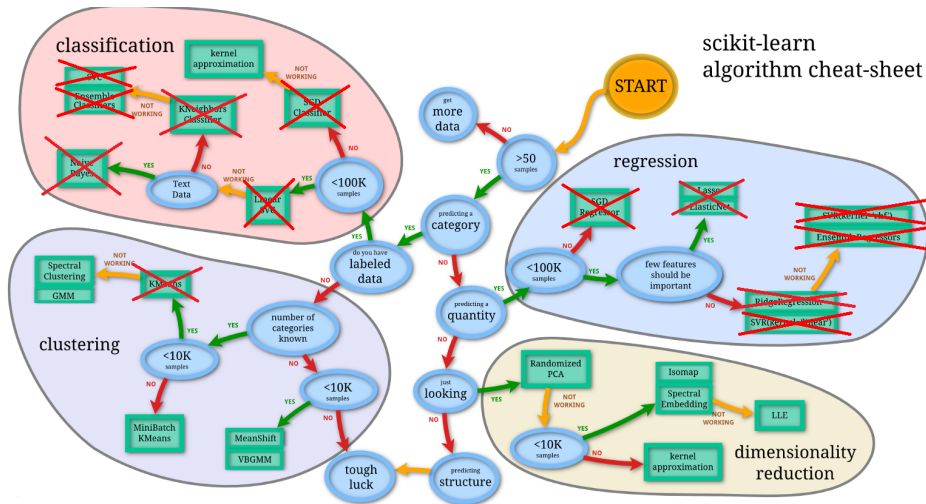
Эти реализации отличаются:

- Методом ветвления в узлах дерева при его обучении,
- Способом работы с категориальными признаками (например, XGBoost не умеет с ними работать),
- Скоростью обучения / тестирования.

	XGBoost	Light BGM		CatBoost	
Parameters Used	max_depth: 50 learning_rate: 0.16 min_child_weight: 1 n_estimators: 200	max_depth: 50 learning_rate: 0.1 num_leaves: 900 n_estimators: 300		depth: 10 learning_rate: 0.15 l2_leaf_reg= 9 iterations: 500 one_hot_max_size = 50	
Training AUC Score	0.999	Without passing indices of categorical features	Passing indices of categorical features	Without passing indices of categorical features	Passing indices of categorical features
		0.992	0.999	0.842	0.887
Test AUC Score	0.789	0.785	0.772	0.752	0.816
Training Time	970 secs	153 secs	326 secs	180 secs	390 secs
Prediction Time	184 secs	40 secs	156 secs	2 secs	14 secs
Parameter Tuning Time (for 81 fits, 200 iteration)	500 minutes	200 minutes		120 minutes	



# Дорожная карта Scikit-Learn<sup>7</sup>



<sup>7</sup>[https://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/](https://scikit-learn.org/stable/tutorial/machine_learning_map/)