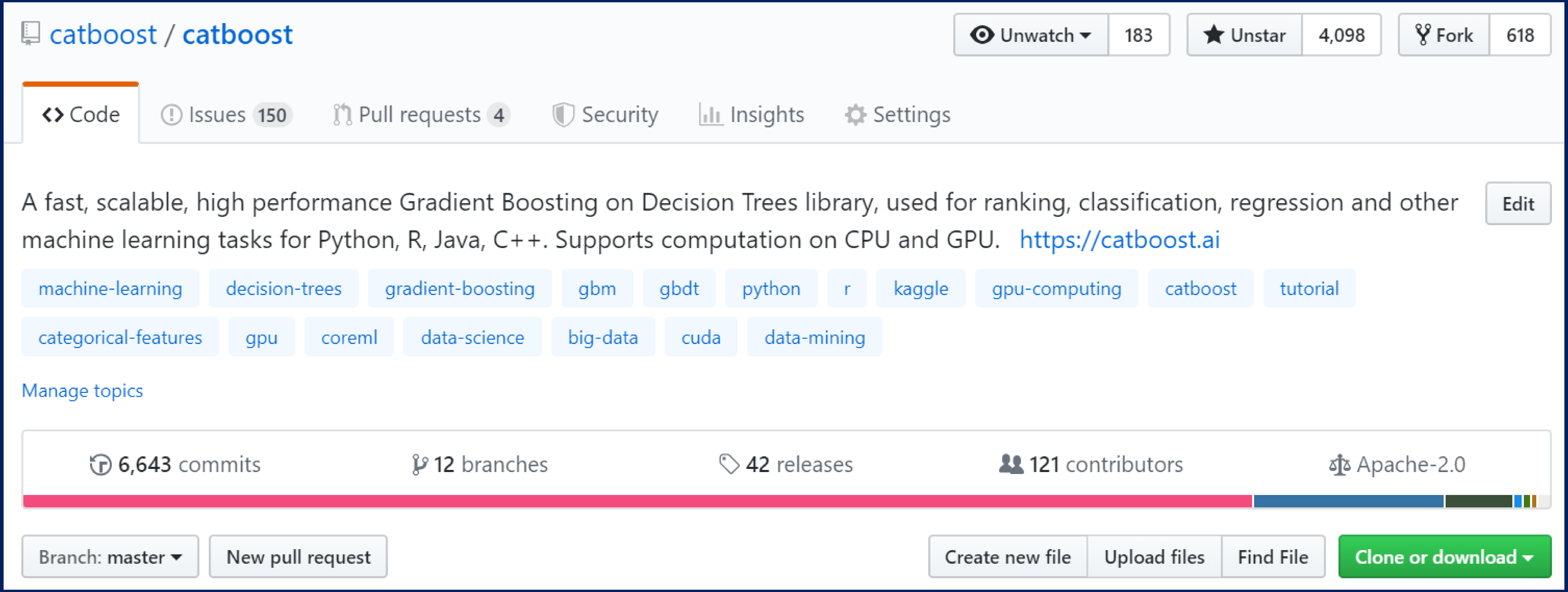
Yandex

CatBoost

The new generation of Gradient Boosting

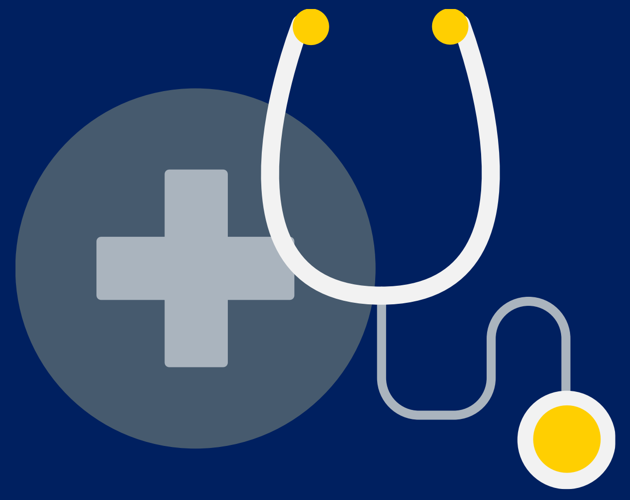
CatBoost


catboost / catboost
Unwatch 183
Unstar 4,098
Fork 618
Code
Issues 150
Pull requests 4
Security
Insights
Settings
A fast, scalable, high performance Gradient Boosting on Decision Trees library, used for ranking, classification, regression and other machine learning tasks for Python, R, Java, C++. Supports computation on CPU and GPU. https://catboost.ai
Edit
machine-learning
decision-trees
gradient-boosting
gbm
gbdt
python
r
kaggle
gpu-computing
catboost
tutorial
categorical-features
gpu
coreml
data-science
big-data
cuda
data-mining
Manage topics
6,643 commits
12 branches
42 releases
121 contributors
Apache-2.0
Branch: master
New pull request
Create new file
Upload files
Find File
Clone or download

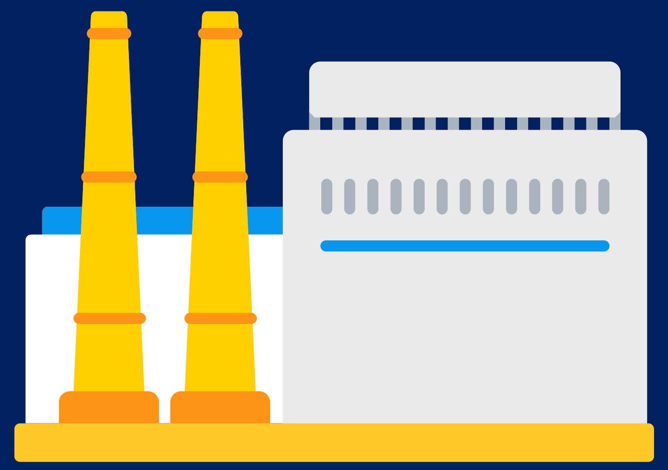
Gradient Boosting

- Best solution for heterogeneous data
- Easy to use
- Works well for small data

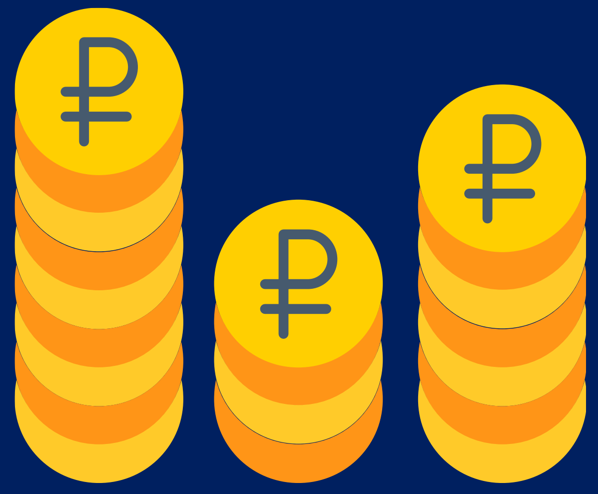
Applications



Medicine



Industry

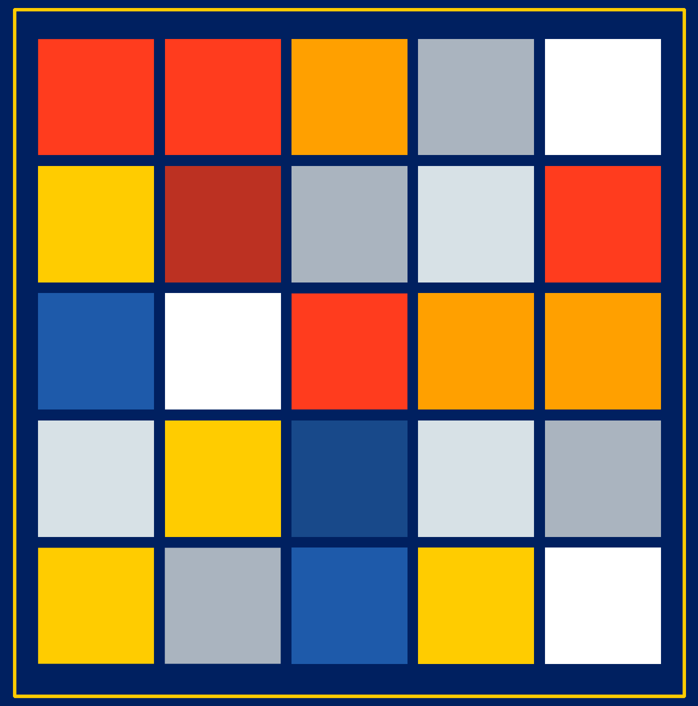


Finance

Music and video recommendations

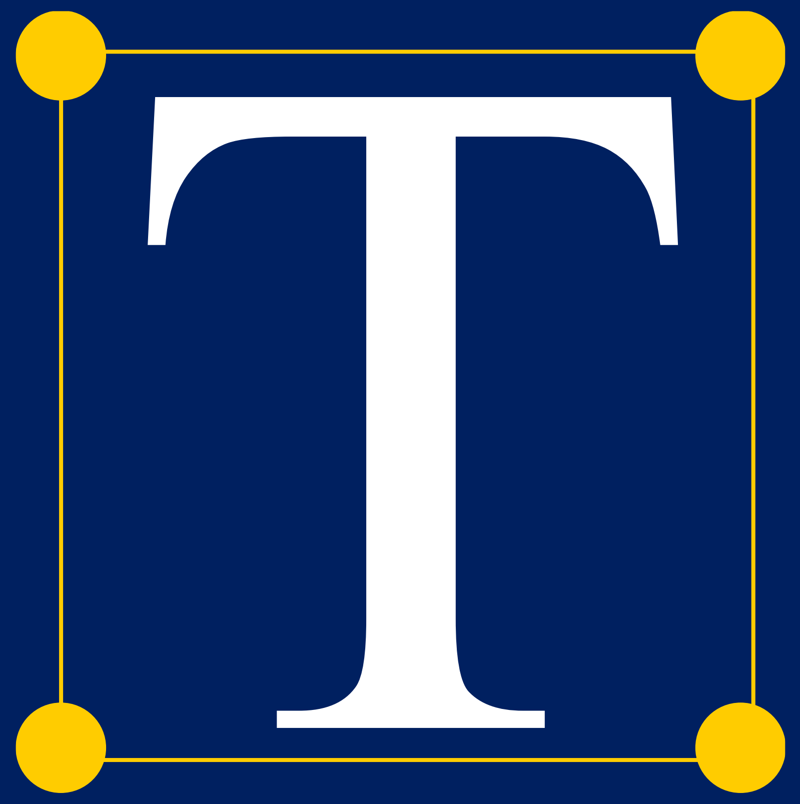
Sales prediction

Neural networks



Images

Sound



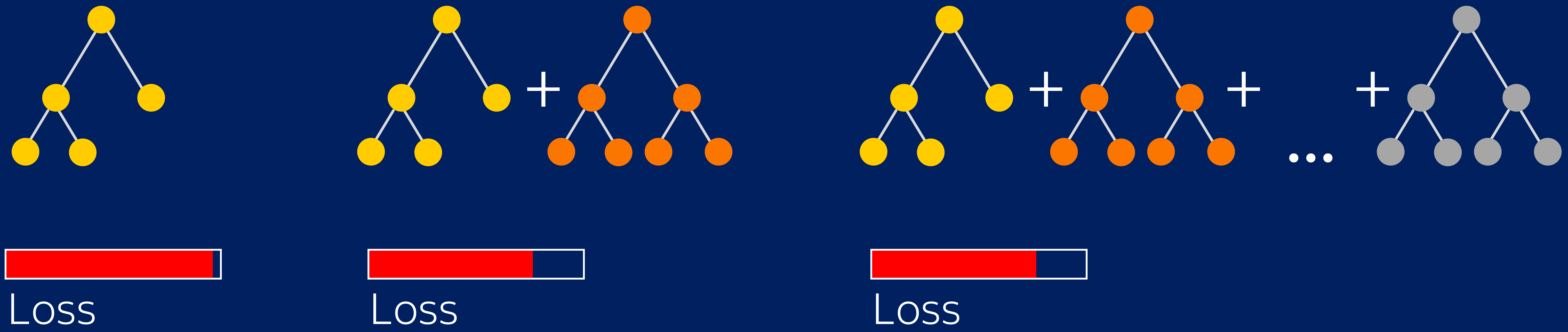
Text

NN + GB

Neural networks

Gradient boosting

Gradient boosting

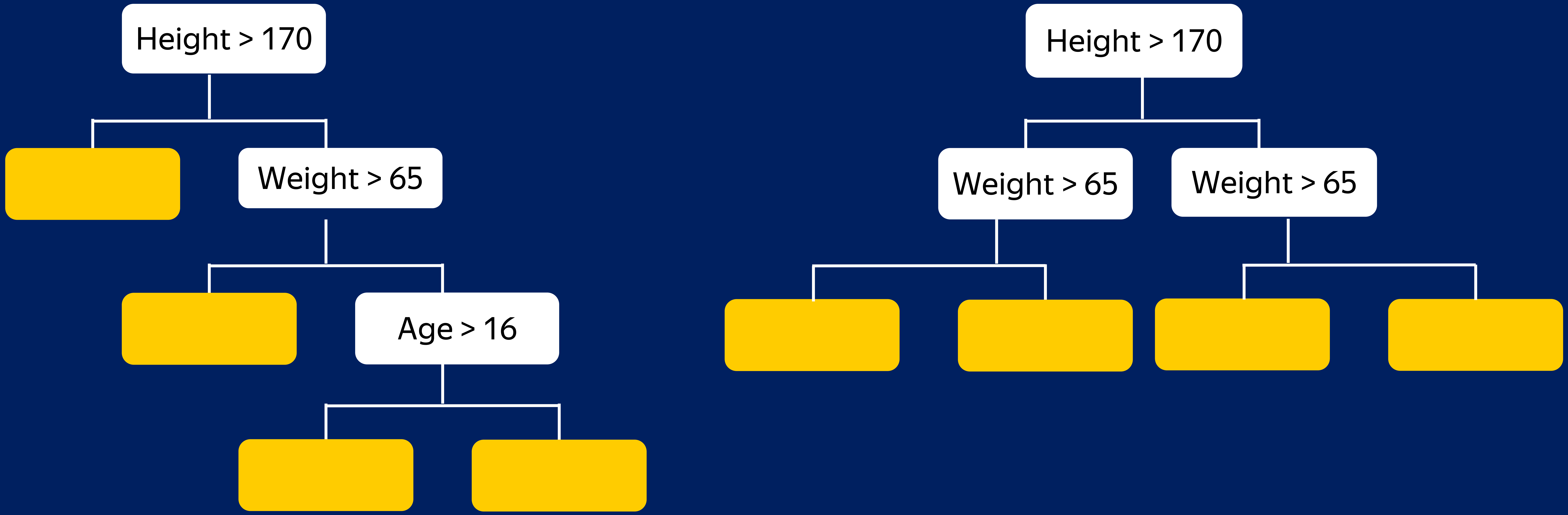


Algorithm comparison

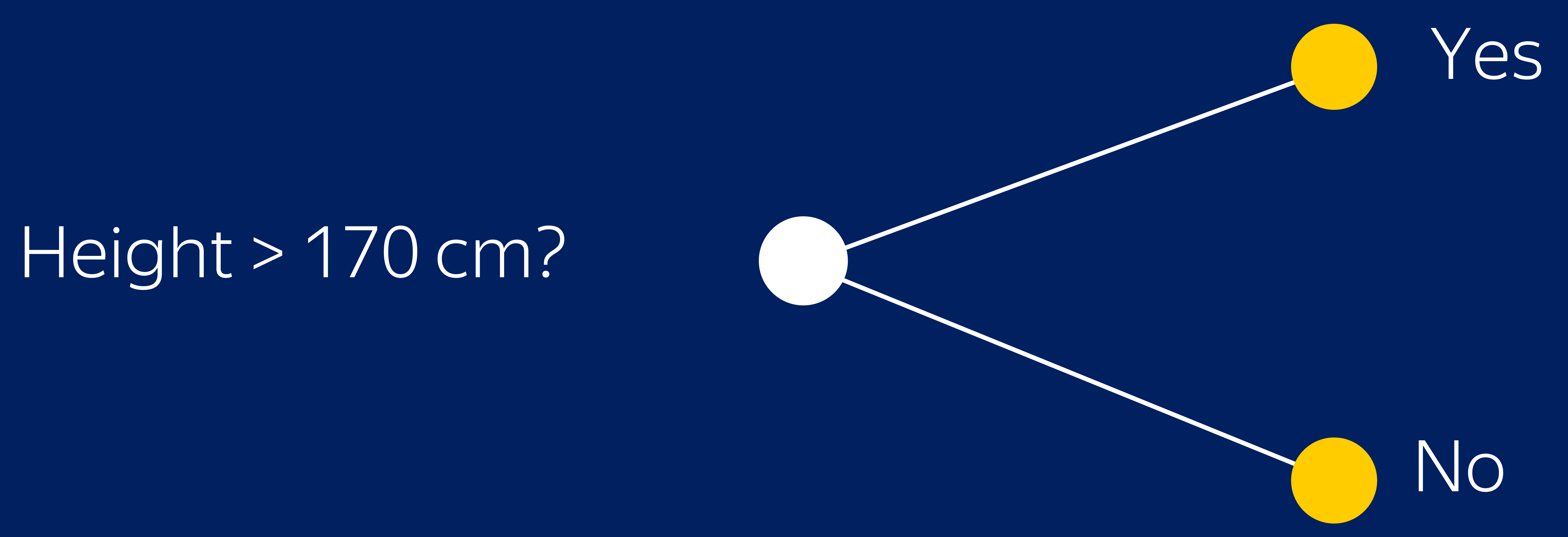
	CatBoost	LightGBM		XGBoost		H2O	
Adult	0.269741	0.276018	+ 2.33 %	0.275423	+ 2.11%	0.275104	+ 1.99%
Amazon	0.137720	0.163600	+ 18.79 %	0.163271	+ 18.55%	0.162641	+ 18.09%
Appet	0.071511	0.071795	+ 0.40 %	0.071760	+ 0.35%	0.072457	+ 1.32%
Click	0.390902	0.396328	+ 1.39 %	0.396242	+ 1.37%	0.397595	+ 1.71%
Internet	0.208748	0.223154	+ 6.90 %	0.225323	+ 7.94%	0.222091	+ 6.39%
Kdd98	0.194668	0.195759	+ 0.56 %	0.195677	+ 0.52%	0.195395	+ 0.37%
Kddchurn	0.231289	0.232049	+ 0.33 %	0.233123	+ 0.79%	0.232752	+ 0.63%
Kick	0.284793	0.295660	+ 3.82 %	0.294647	+ 3.46%	0.294814	+ 3.52%

Logloss

Several tree growing strategies

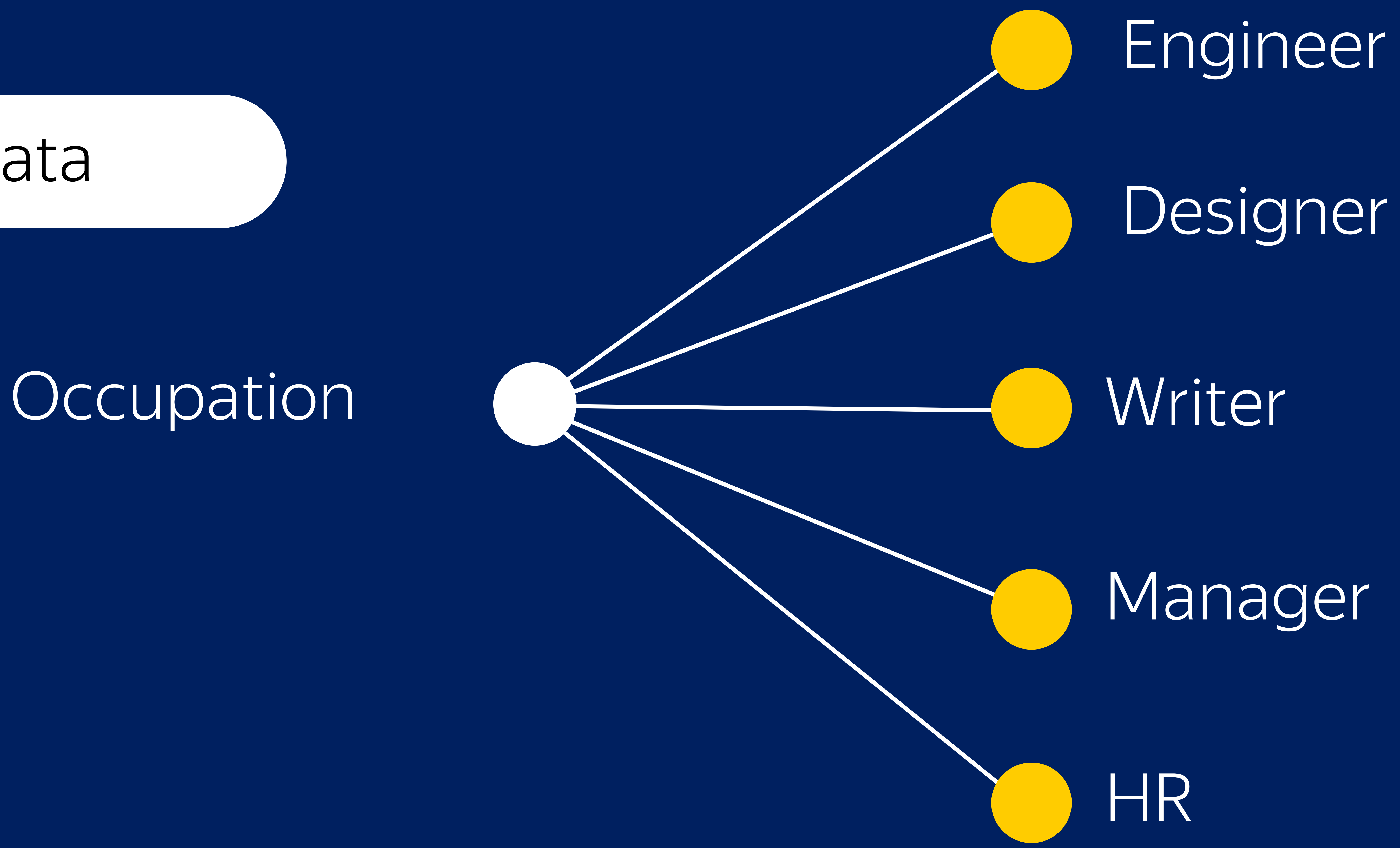


Numerical features



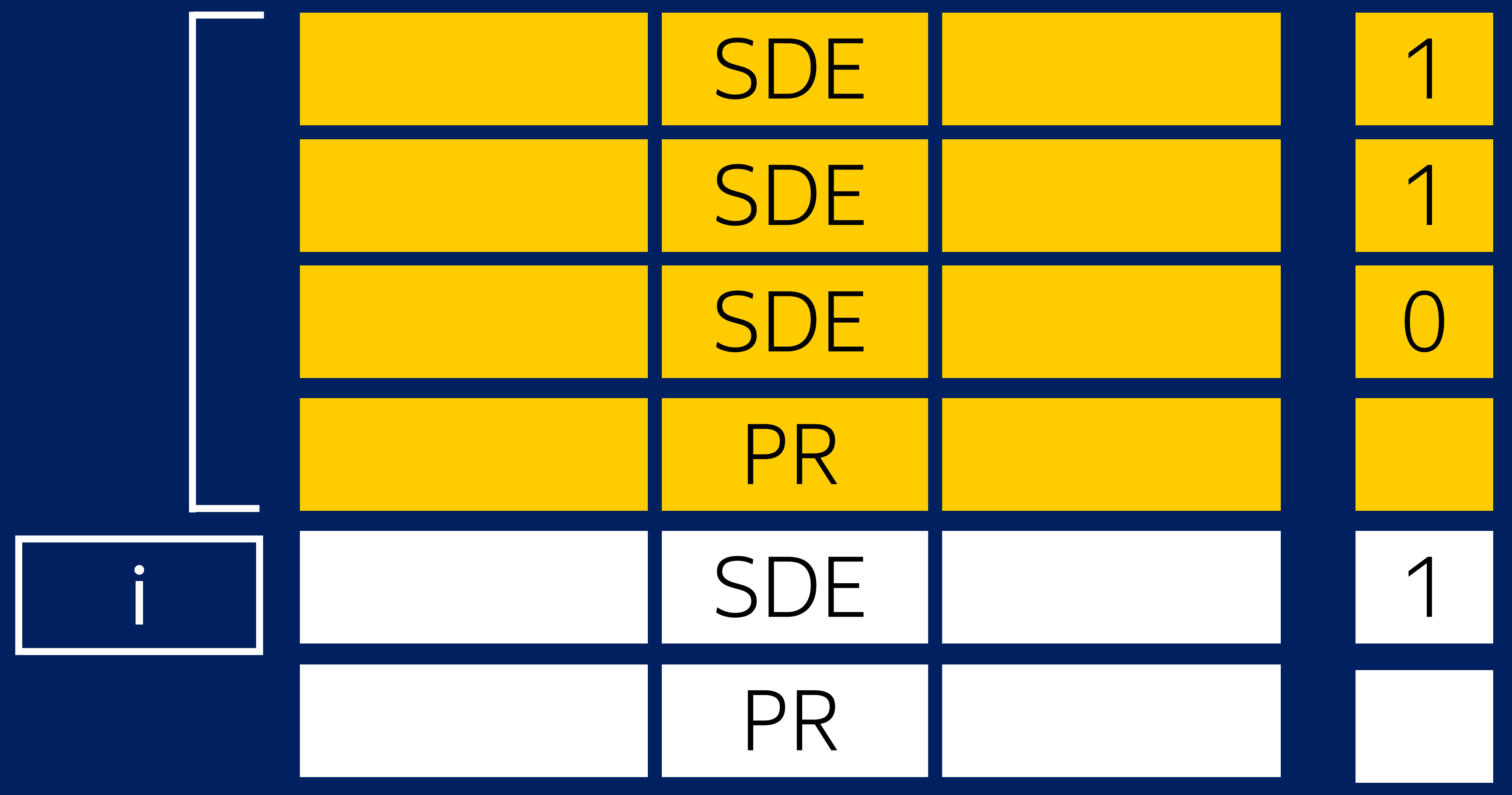
Categorical features

Categorical data



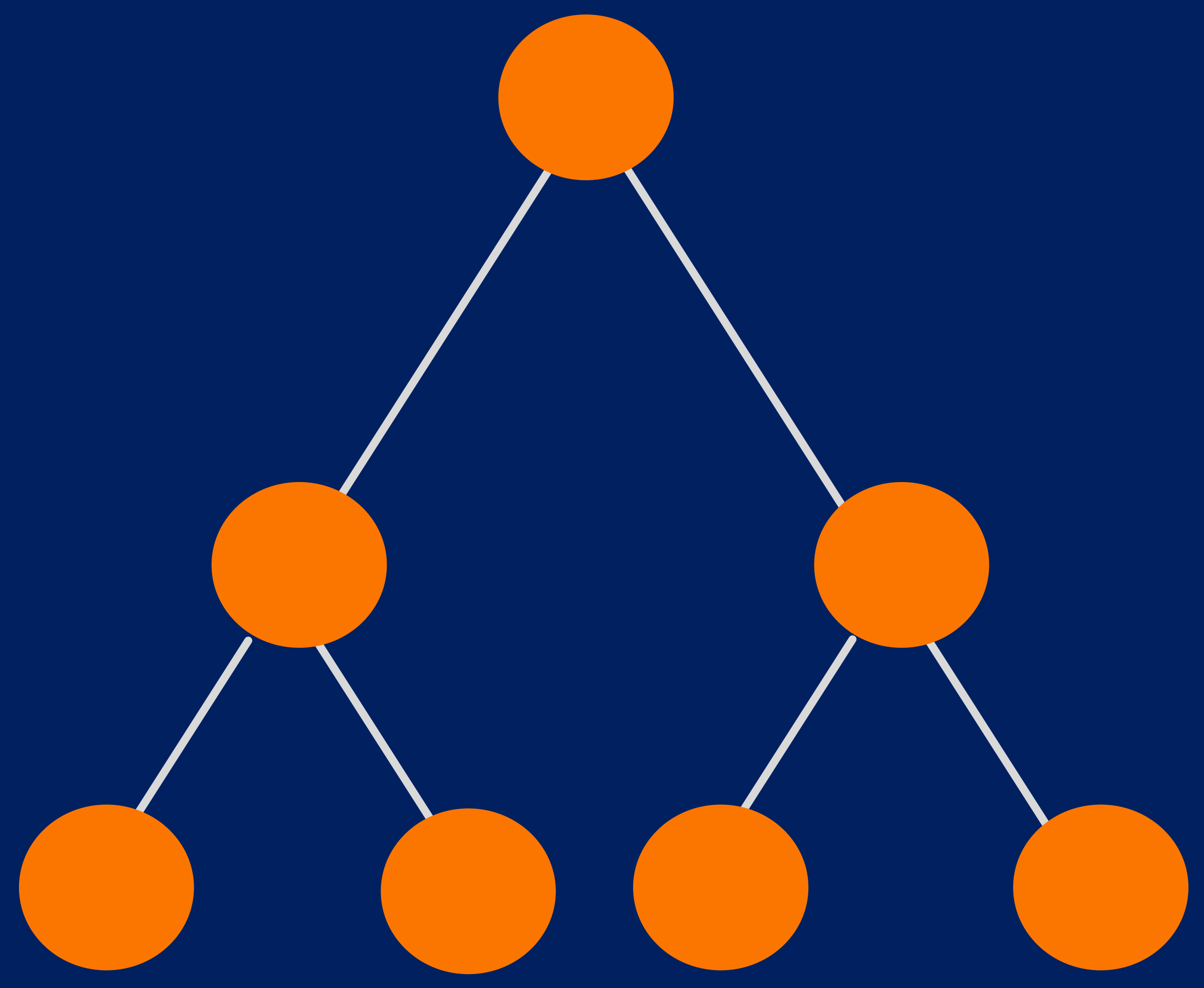
Categorical features support

- One-hot encoding
- Statistics based on category and category plus label value
- Usage of several permutations
- Greedy constructed feature combinations



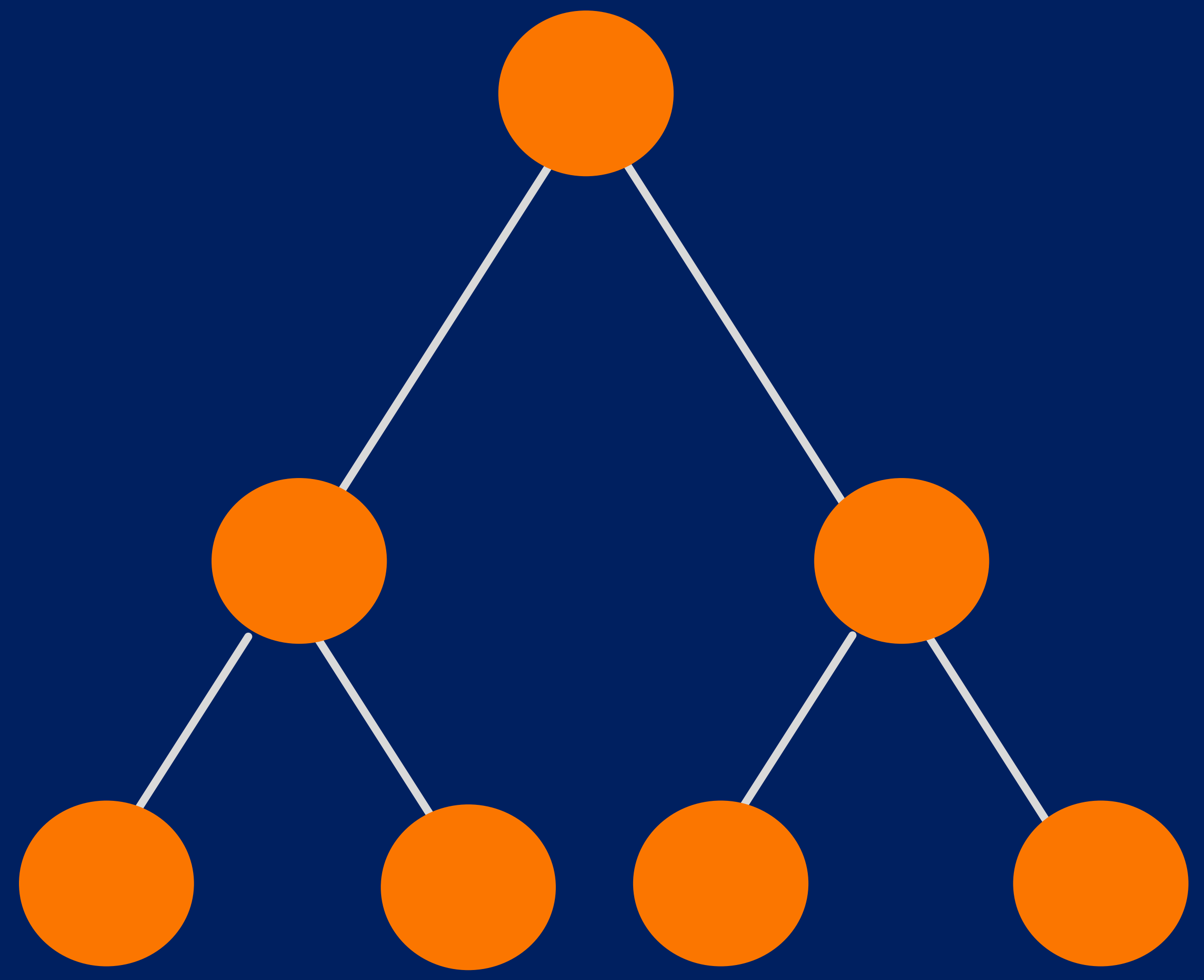
$$i \longrightarrow \frac{1 + 1 + 0 + a * \mathrm{Prior}}{3 + a}$$

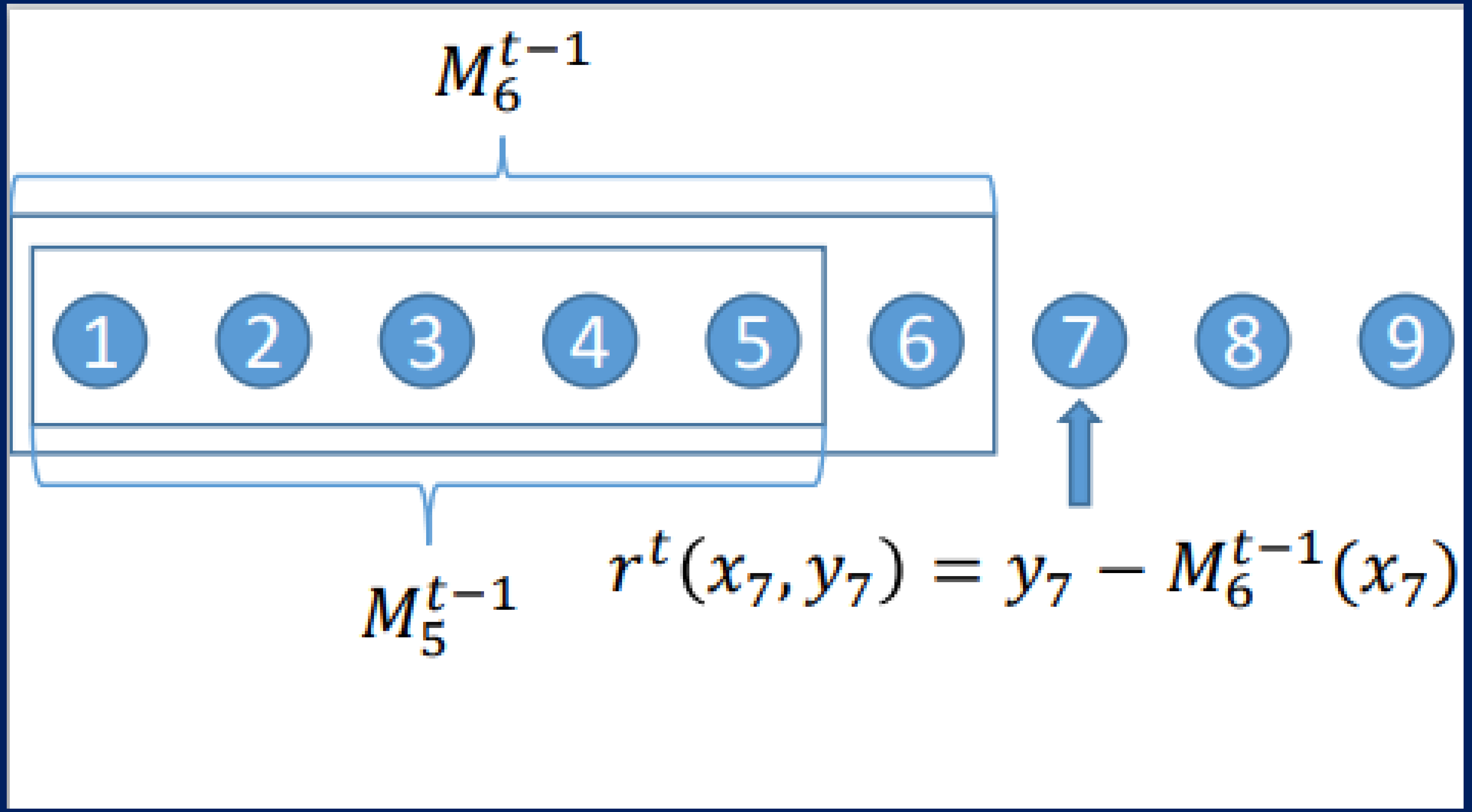
Classical boosting



$$\text{leafValue} = \sum_{i=1}^{n} \frac{\text{g(approx(i), target(i))}}{n}$$

Ordered boosting





Modes

- Classification
- Regression
- Ranking

Ranking

- What are top N hotels in some city?
- Training data: ratings

Predicting rating is not necessary!

- Ranking within a group
- Ranking modes:
    - Ranking (YetiRank, YetiRankPairwise)
    - Pairwise (PairLogit, PairLogitPairwise)
    - Ranking + Classification (QueryCrossEntropy)
    - Ranking + Regression (QueryRMSE)
    - Select top 1 candidate (QuerySoftMax)

Speed

- CPU training
- GPU training
- Prediction speed

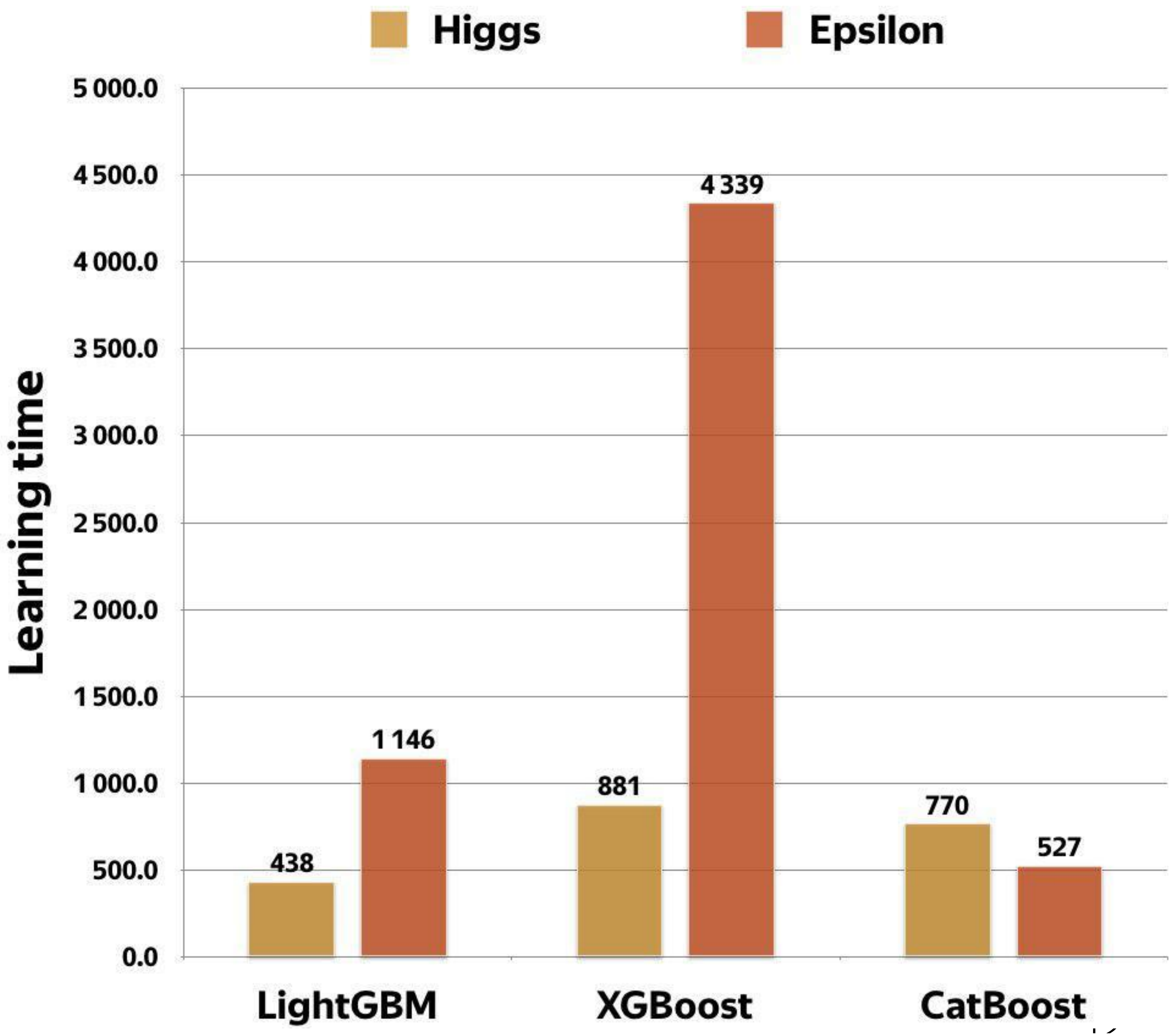
CPU: Comparison with other libraries

- Parameters:

128 bins, 64 leafs, 400 iterations
- Higgs:

28 features, 11M samples
- Epsilon:

2000 features, 400K samples



GPU: Comparison with other libraries

- Parameters:

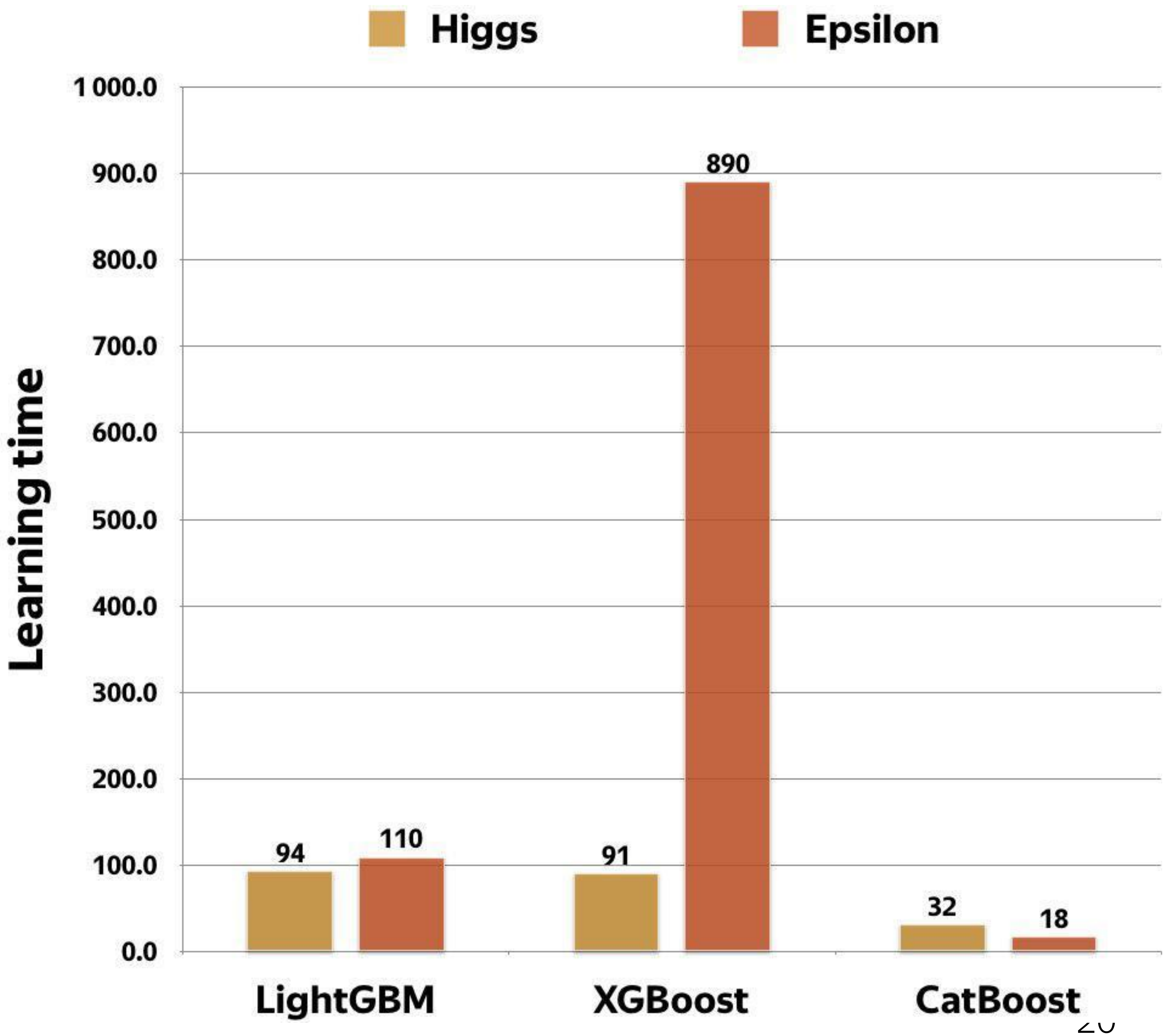
128 bins, 64 leafs, 400 iterations

- Higgs:

28 features, 11M samples

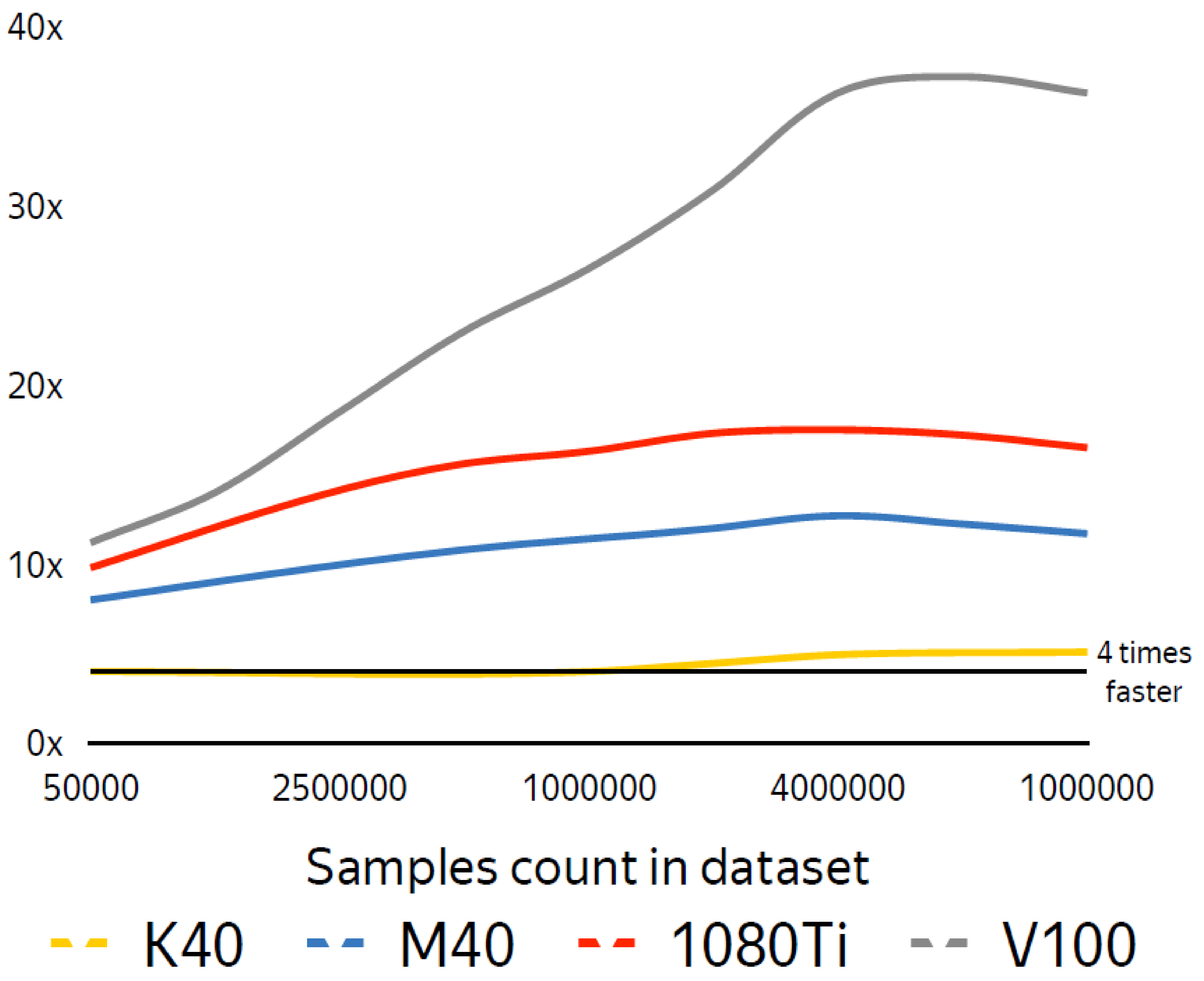
- Epsilon:

2000 features, 400K samples

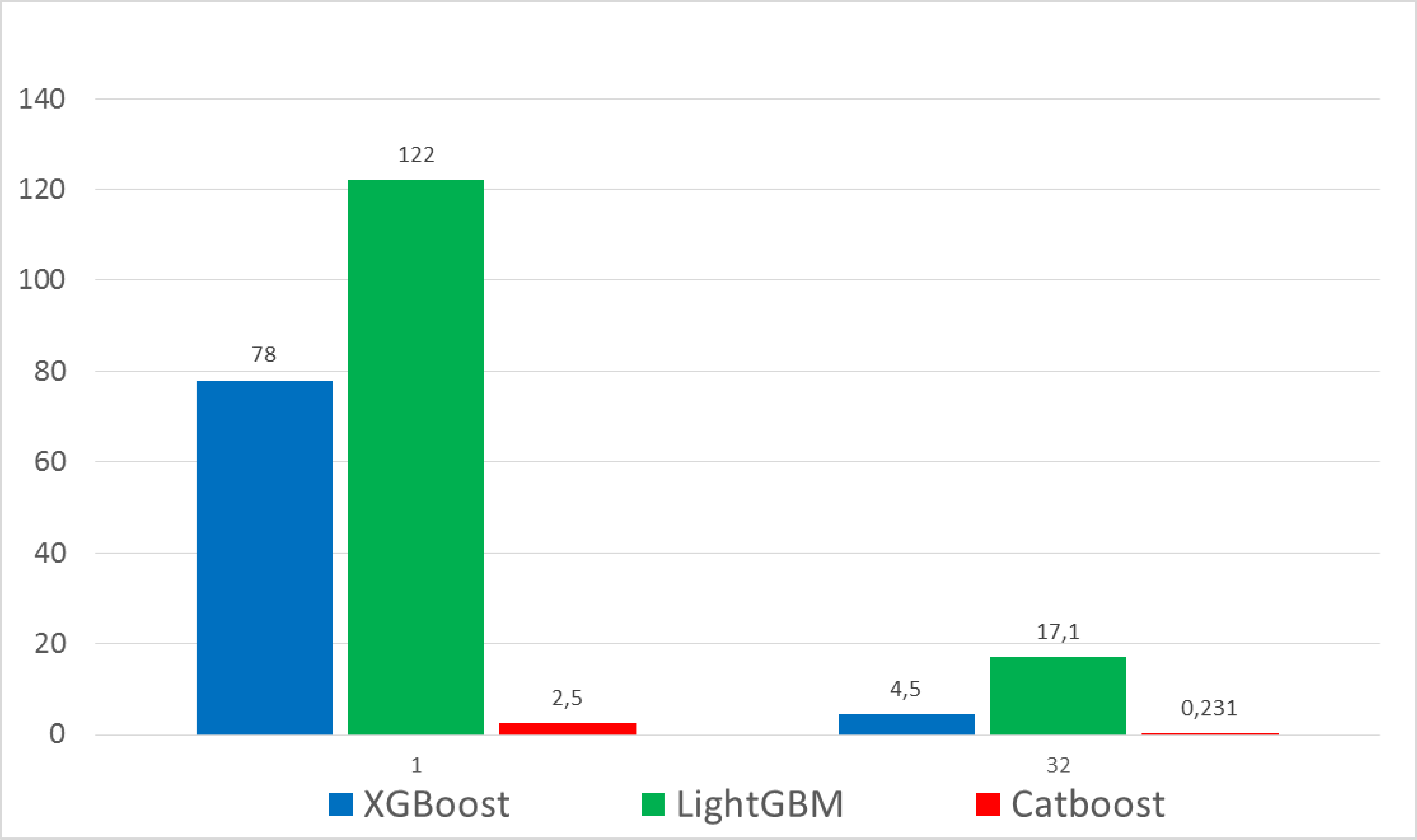


CPU vs GPU

- Dual-Socket Intel Xeon E5-2660v4 as baseline
- Several modern GPU as competitors
- Dataset: 800 features



Prediction time



CatBoost Appliers

Yandex CatBoost

ONNX

CoreML

PMML
Predictive Model
Markup Language

NVIDIA®

Java™

Microsoft
.NET

Ways to explore your data

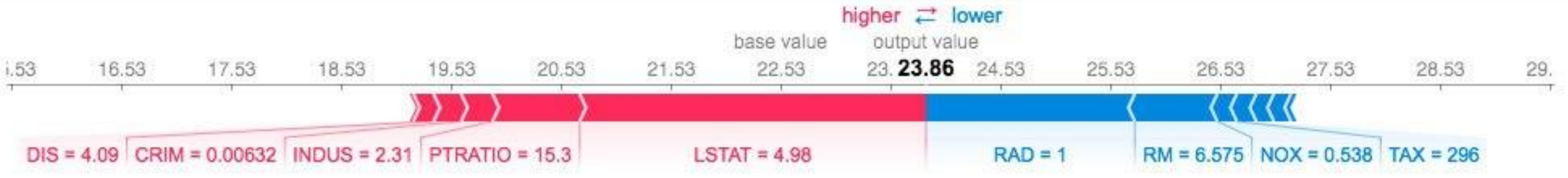
- Feature importance
  - PredictionValueChange
  - LossFunctionChange
- Feature interaction
- Per object feature importance (SHAP)

SHAP values

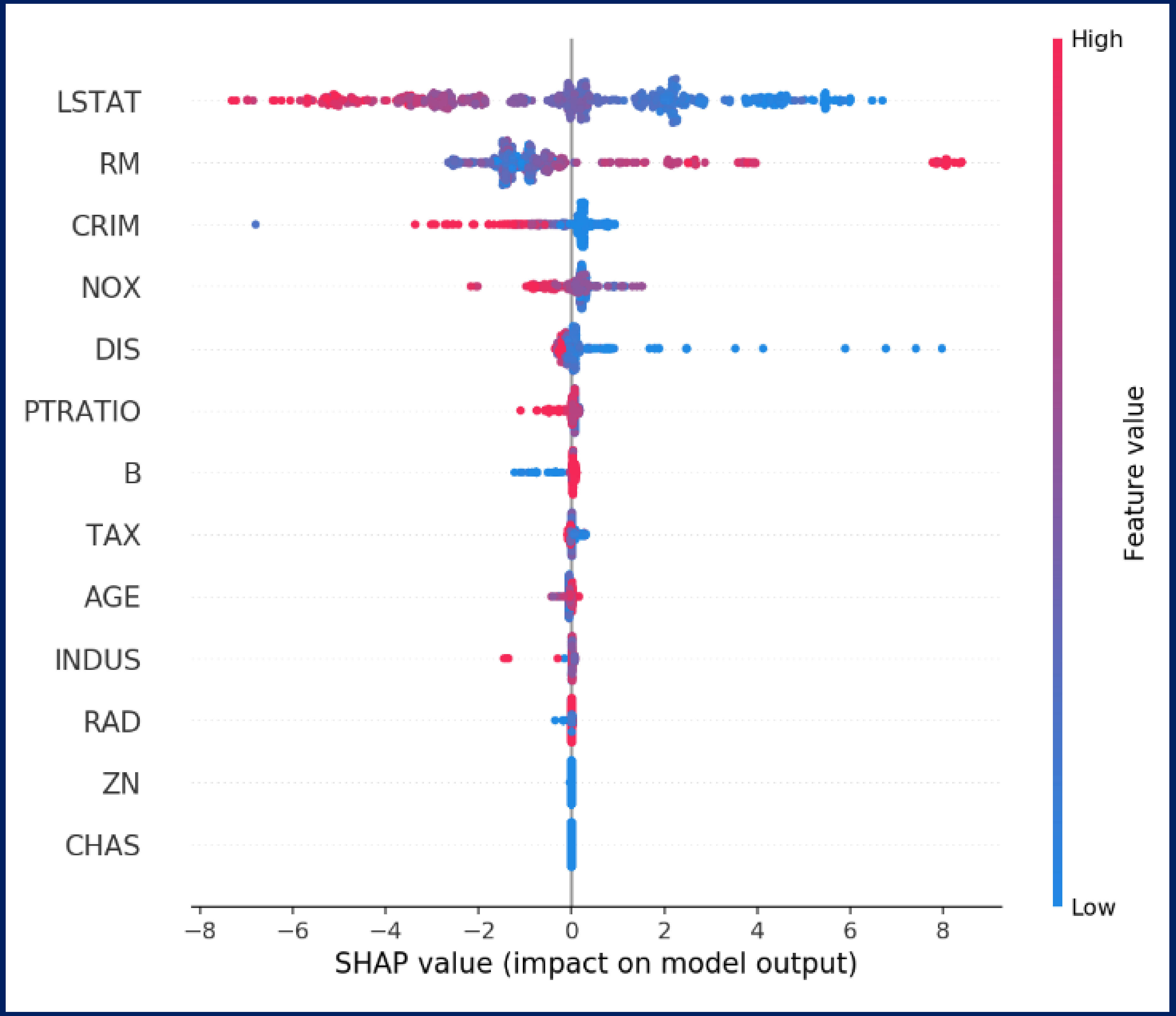
In [46]:

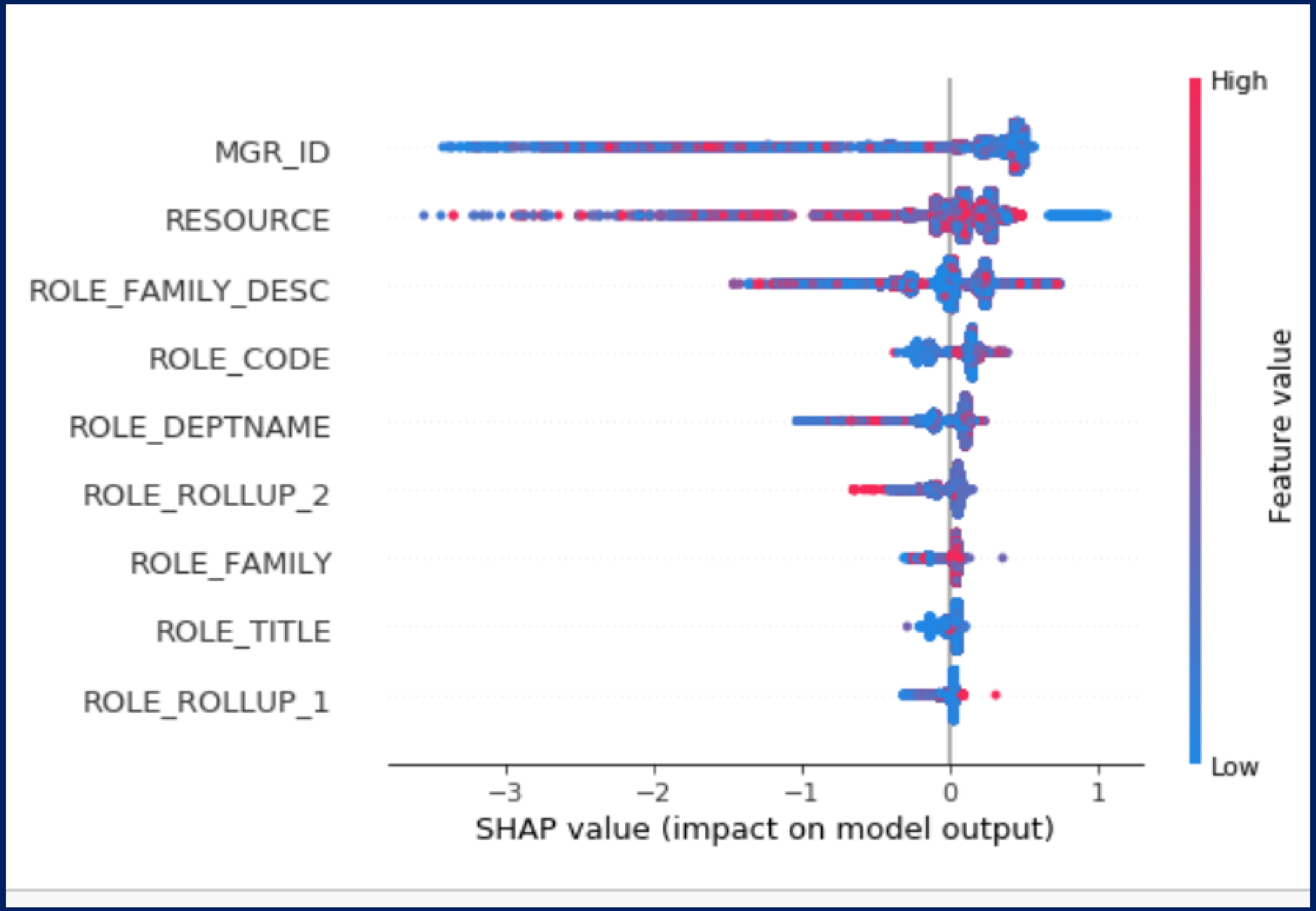
```
shap_values = model.get_feature_importance(data=Pool(X,y), fstr_type='ShapValues')

# visualize the first prediction's explanation
shap.force_plot(shap_values[0,:], X.iloc[0,:])
```

Out[46]:



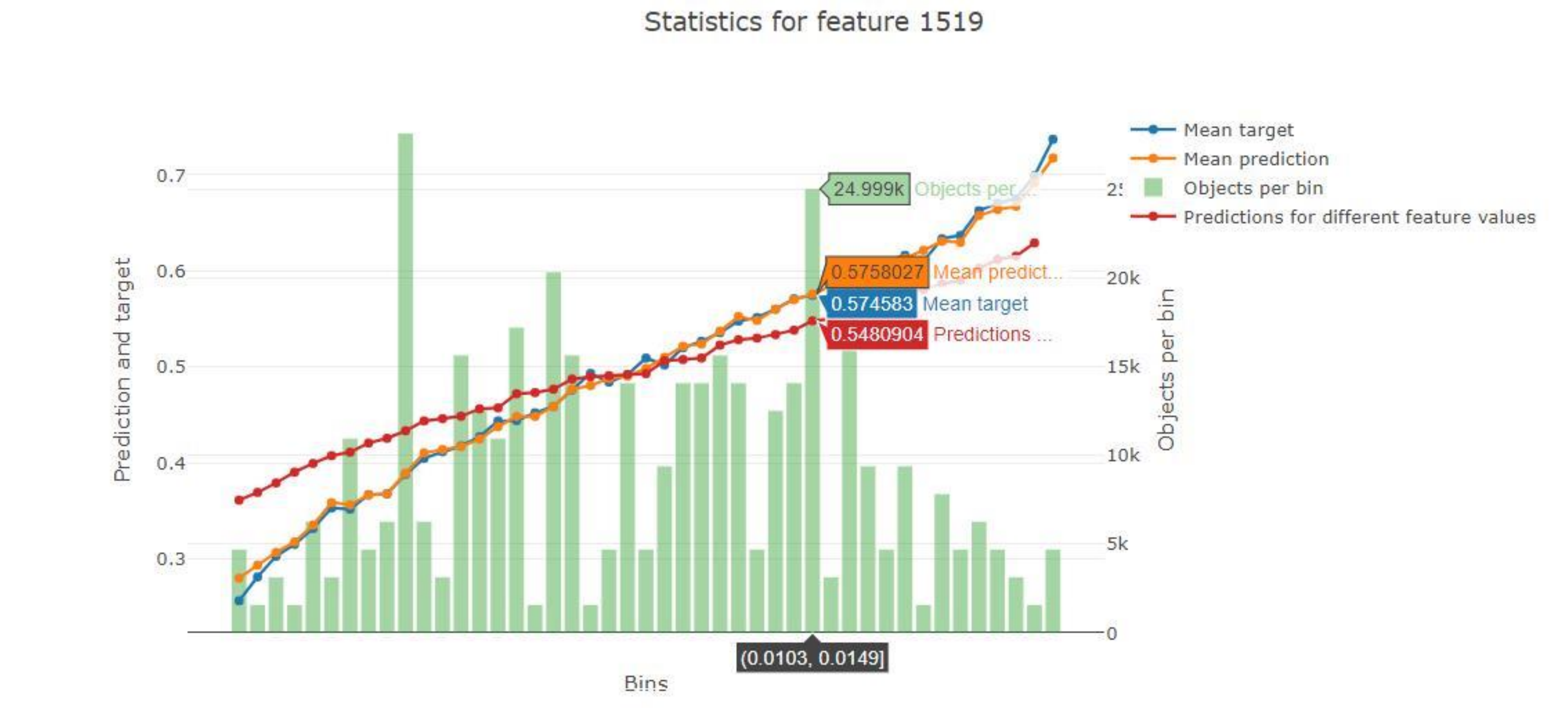
SHAP values


LSTAT
RM
CRIM
NOX
DIS
PTRATIO
B
TAX
AGE
INDUS
RAD
ZN
CHAS
−8
−6
−4
−2
0
2
4
6
8
SHAP value (impact on model output)
High
Low
Feature value


MGR_ID
RESOURCE
ROLE_FAMILY_DESC
ROLE_CODE
ROLE_DEPTNAME
ROLE_ROLLUP_2
ROLE_FAMILY
ROLE_TITLE
ROLE_ROLLUP_1
−3
−2
−1
0
1
SHAP value (impact on model output)
High
Low
Feature value

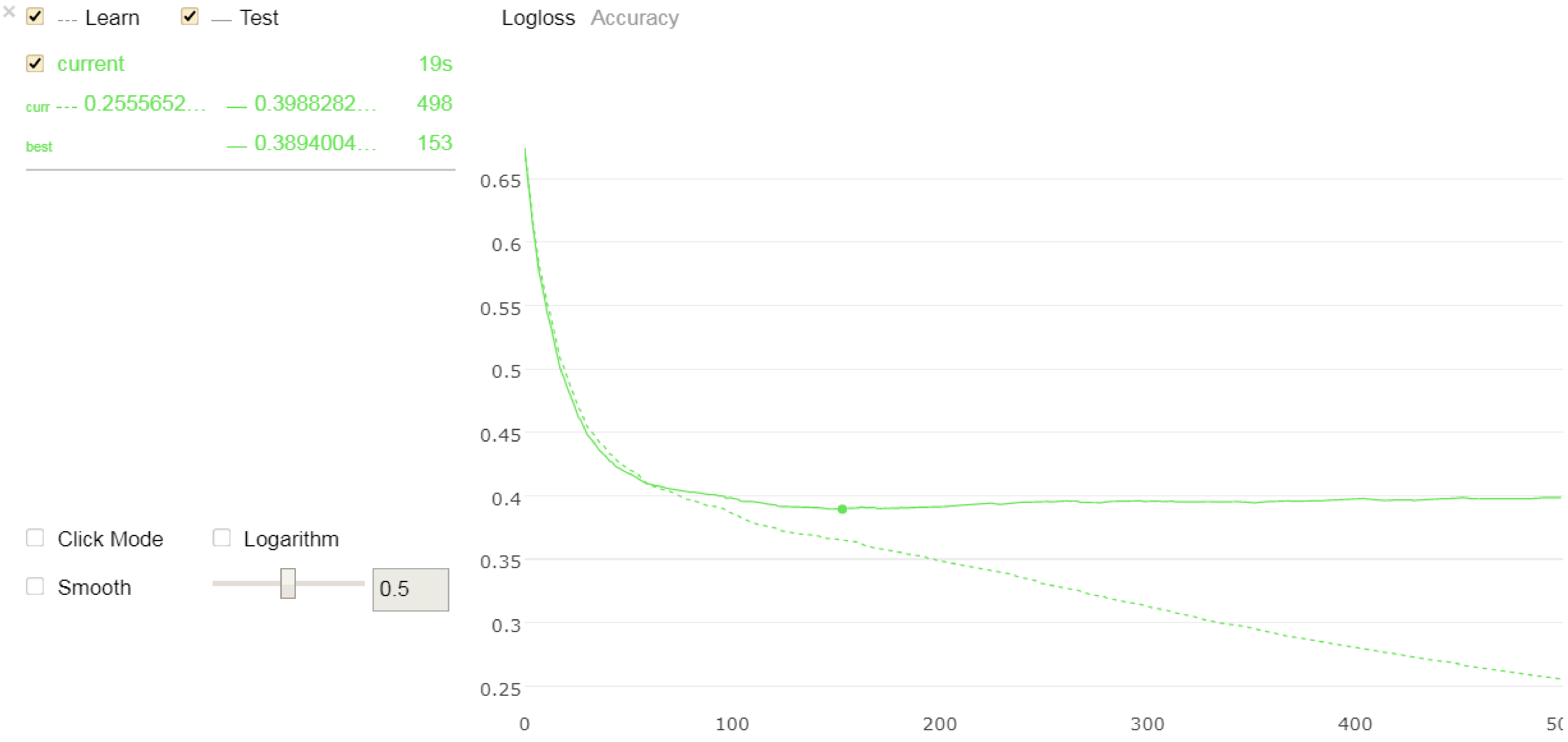
Ways to explore your data

- Feature statistics

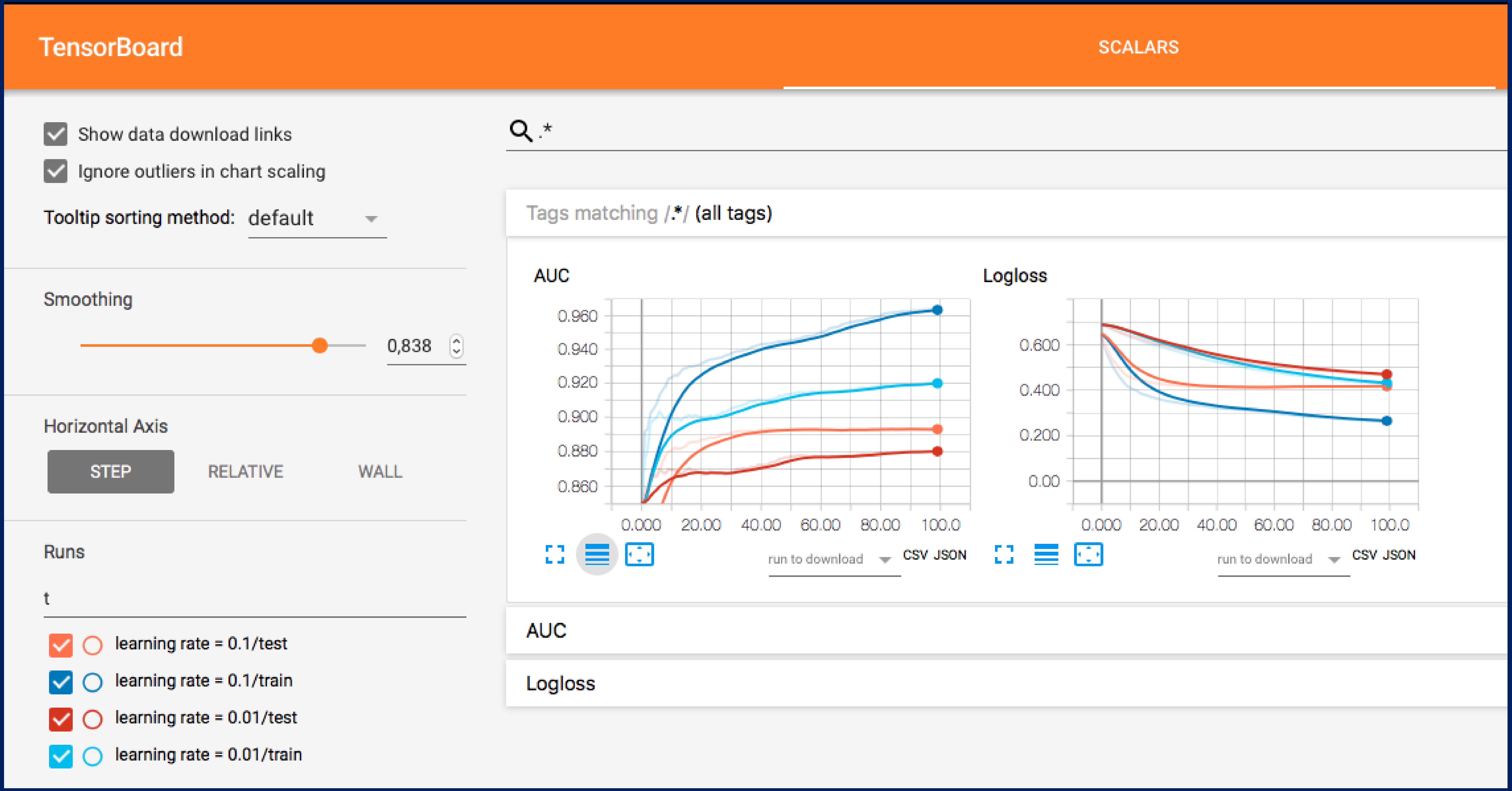


Training visualisation

```
In [11]: model.fit(
             X_train, y_train,
             cat_features=categorical_features_indices,
             eval_set=(X_validation, y_validation),
         #     verbose=True,  # you can uncomment this for text output
             plot=True
         )
```

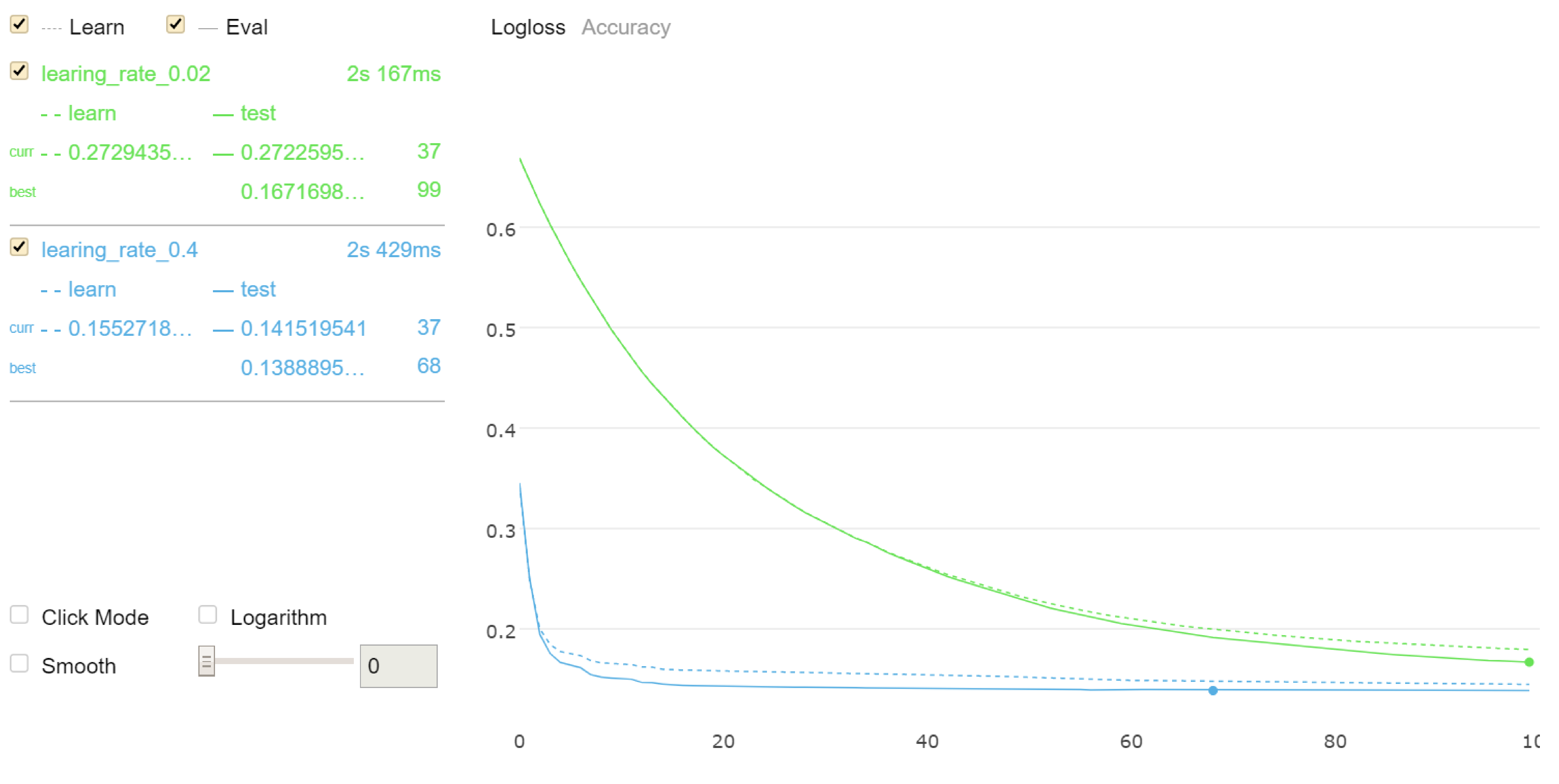


TensorBoard

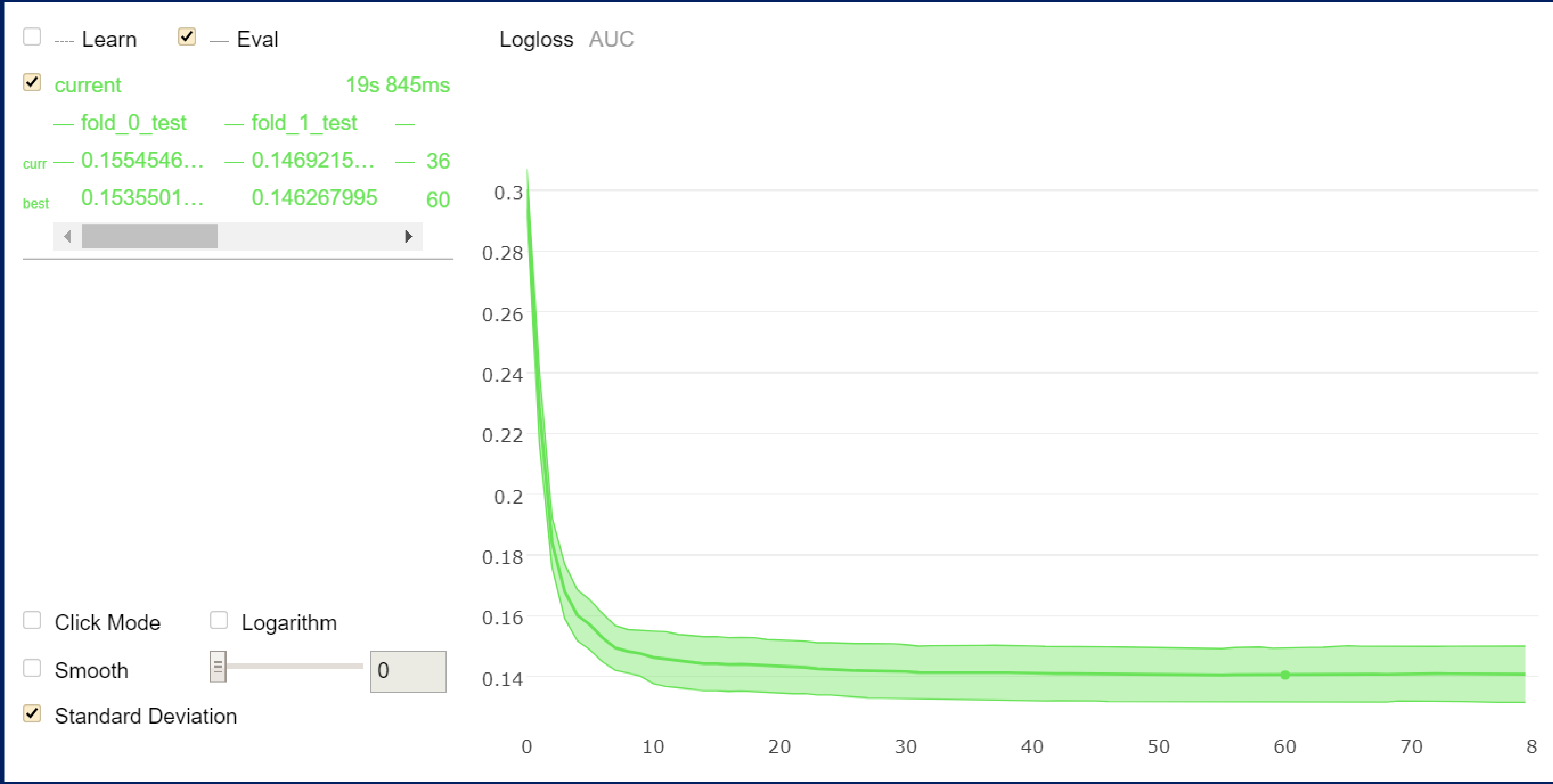


Compare several models

```
from catboost import MetricVisualizer
MetricVisualizer(['learing_rate_0.02', 'learing_rate_0.4']).start()
```



Cross-validation



Ways to explore your data

- Influential documents
- New features evaluation

Algorithm parameters

- learning_rate + iterations
- depth
- l2_regularization
- bagging_temperature / sample_rate
- random_strength
- grow_policy

Reading

- http://learningsys.org/nips17/assets/papers/paper_11.pdf
- https://arxiv.org/abs/1706.09516
- https://github.com/catboost/tutorials

- catboost.ai
- github.com/catboost
- twitter.com/CatBoostML
- t.me/catboost_en, t.me/catboost_ru
- ods.ai => slack (30k people community) => #tool_catboost chanel
- forms.yandex.ru/surveys/10011699

Questions?

Anna Veronika Dorogush

Head of CatBoost team