Yandex

Yandex
CatBoost

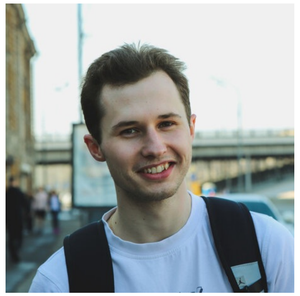
CatBoost: Distributed Training, Uncertainty Estimation and Other News

Kirillov Stanislav,
Head of ML Systems group @ Yandex

Plan

- Introduction
- Main features of CatBoost
- GPU support: single host and distributed training
- CatBoost for Apache Spark release
- Uncertainty estimation
- Further plans

CatBoost core developers



Nikita
Dmitriev

Ekaterina
Ermishkina

Andrew
Khropov

Stanislav
Kirillov

Ivan
Lyzhin

Dmitry
Oganesyan

Eugeny
Petrov

What is CatBoost?

Yandex
CatBoost

- CatBoost is a machine learning algorithm that uses gradient boosting on decision trees (GBDT).
- It is available as an open source library.

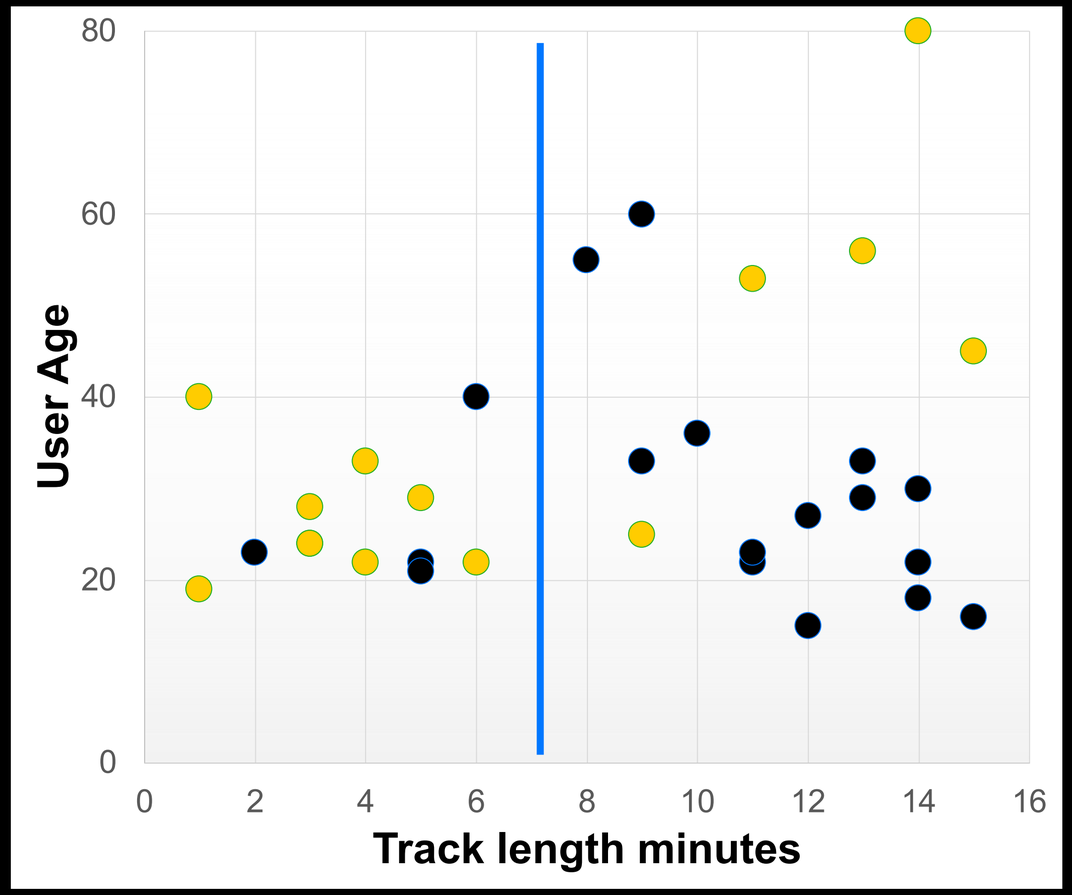
Why CatBoost?

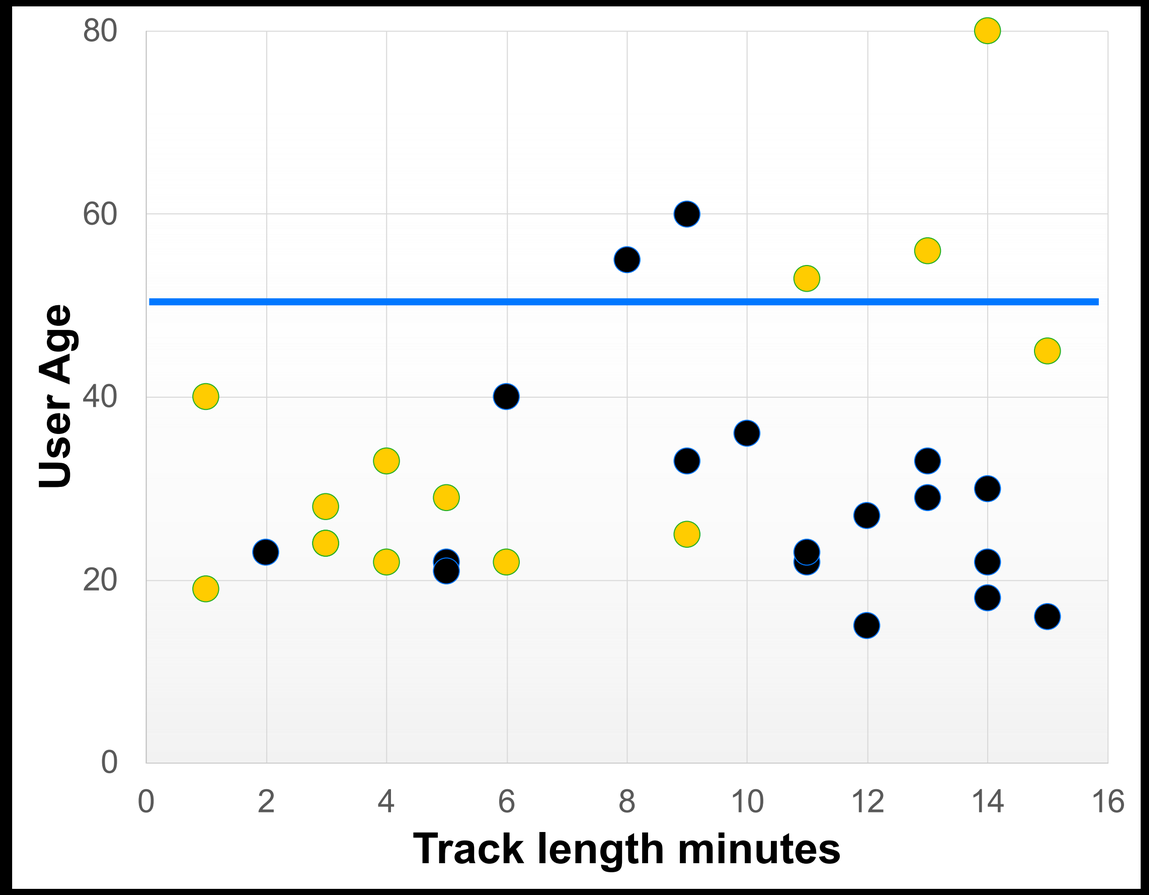
Yandex
CatBoost

- Sophisticated feature types support: categorical, text and embedding features
- Good quality with default parameters
- Multiple tree grow policies: symmetric, loss guide and depthwise
- Fast applier
- Model analysis tools
- Novel uncertainty estimation support

Main features & concepts

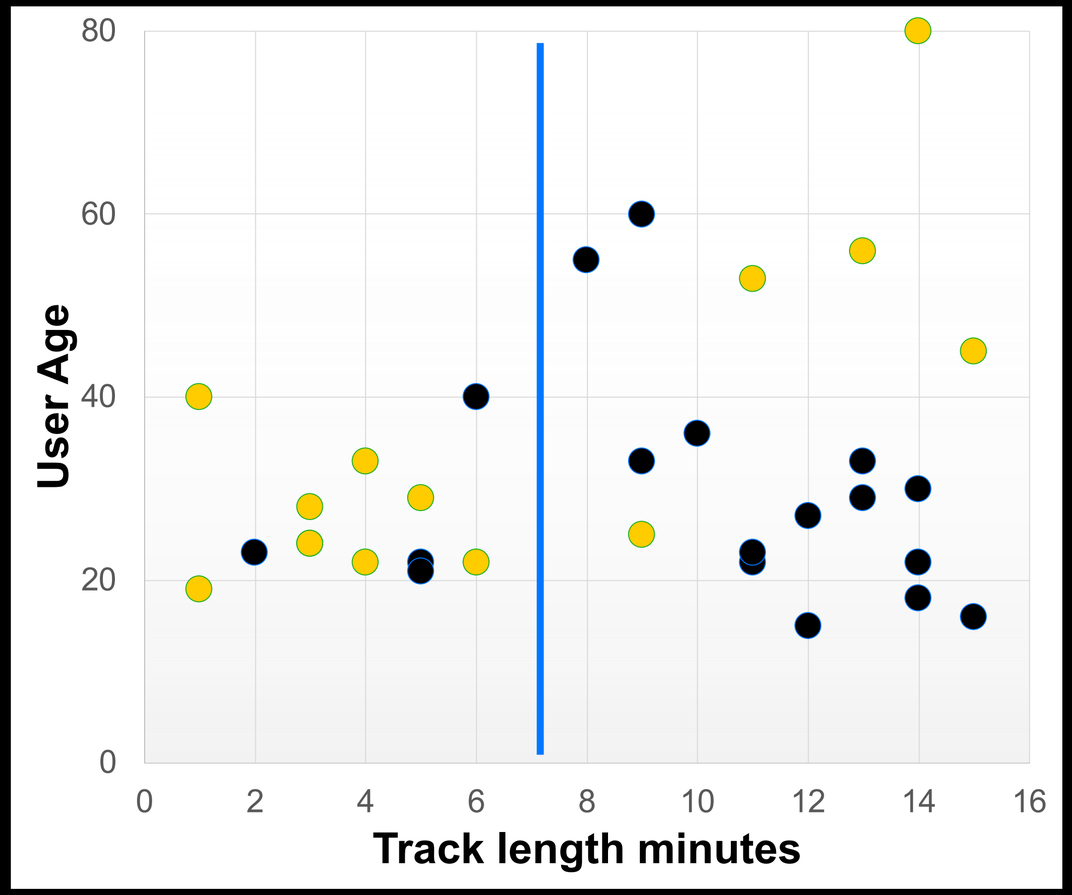
Decision tree: classification

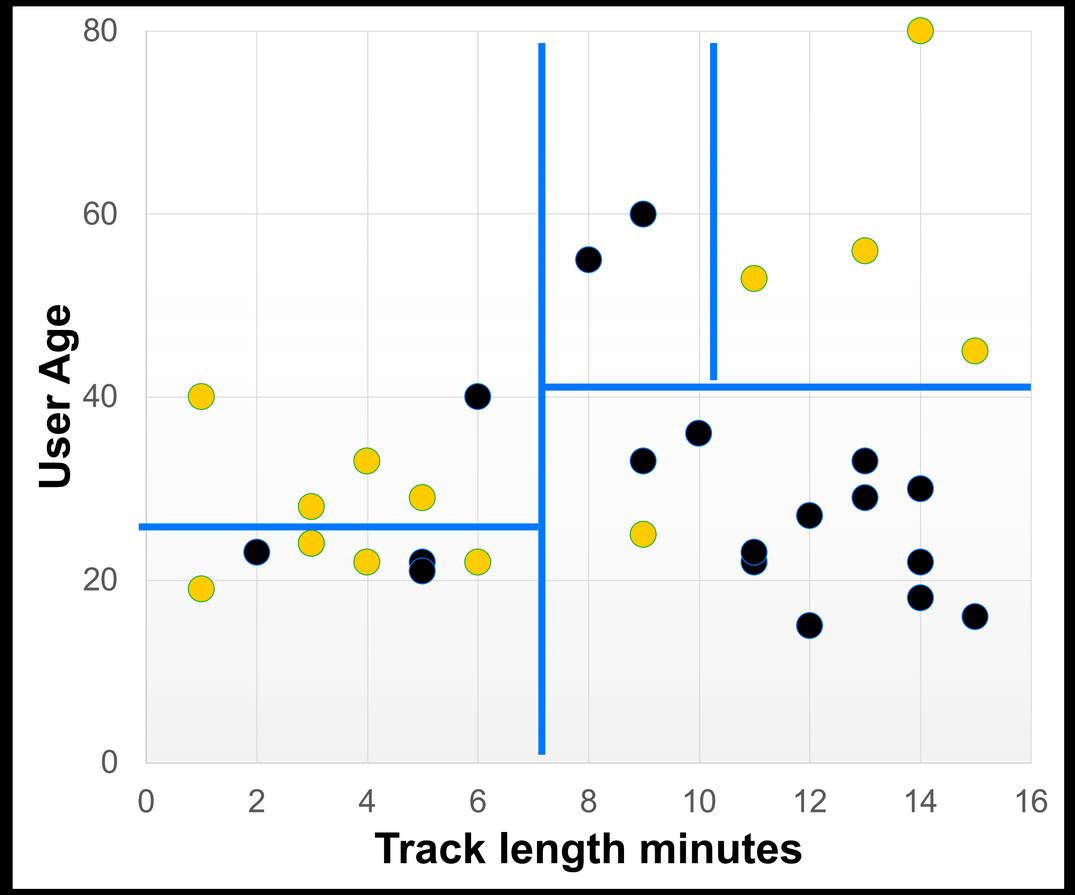




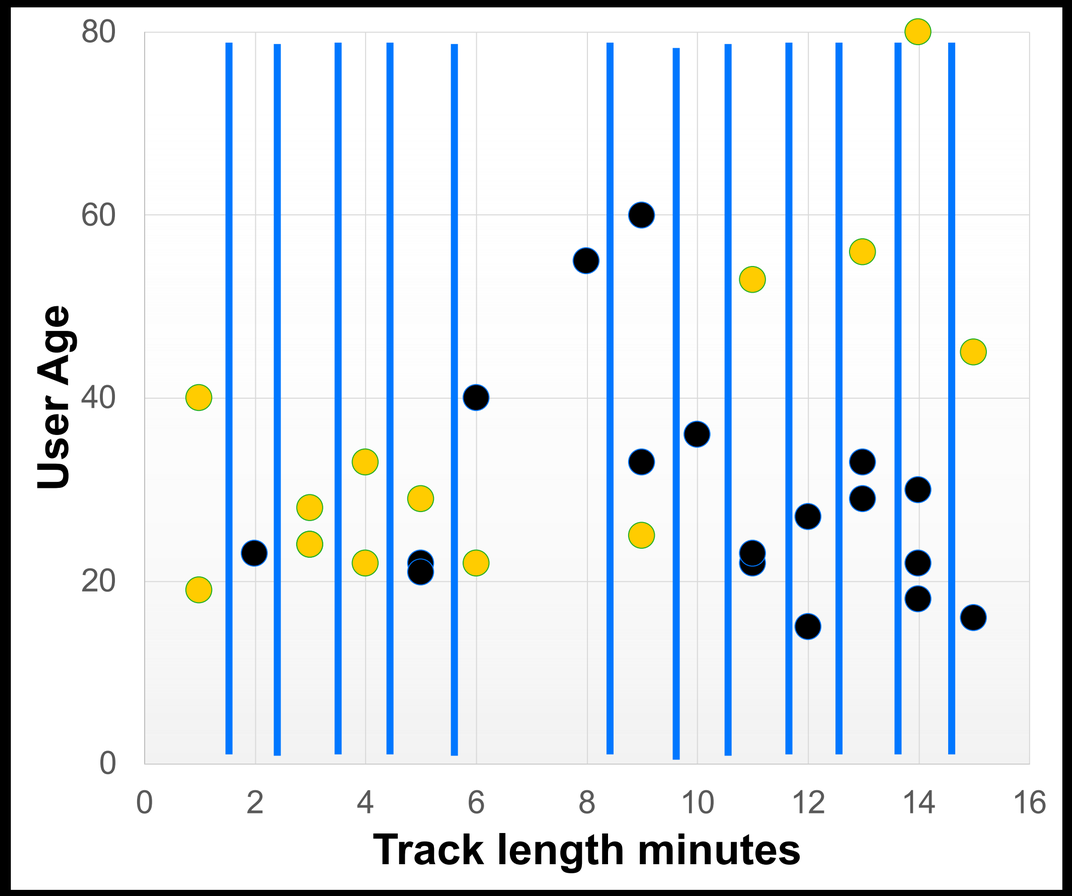
Best split: Track length minutes > 7

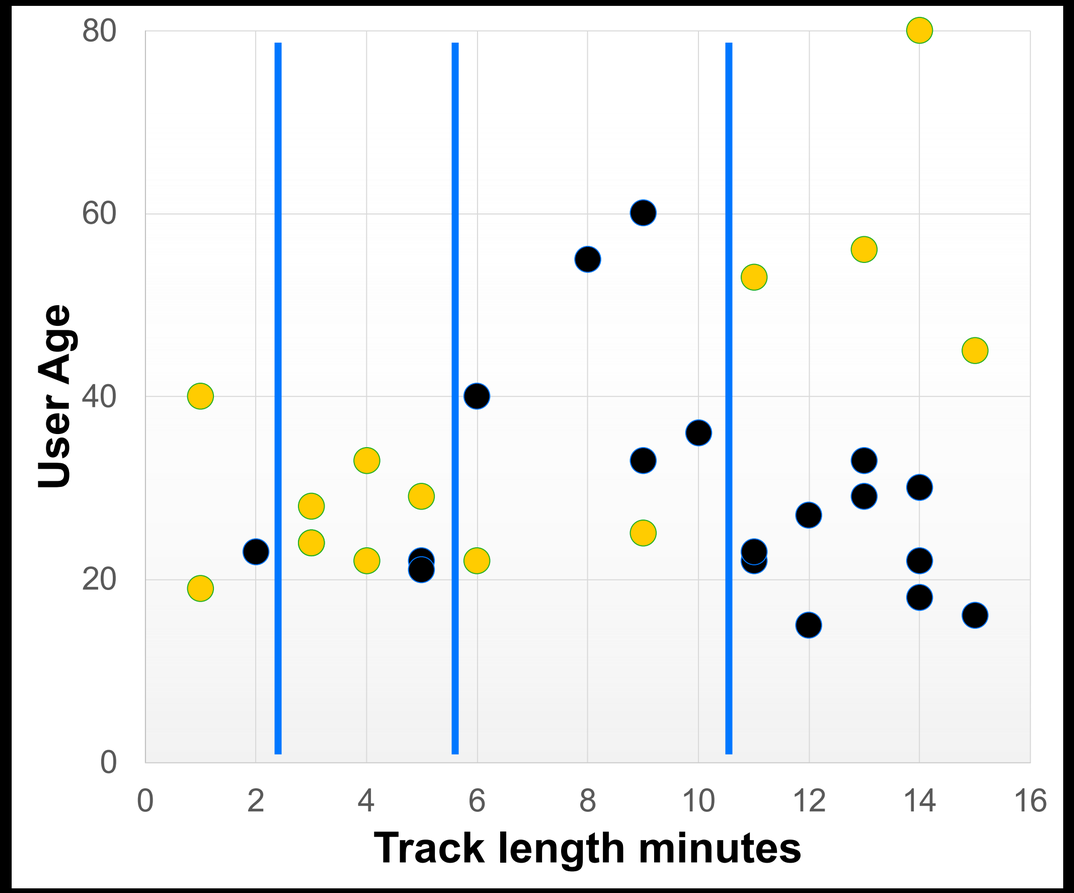
Decision tree: classification



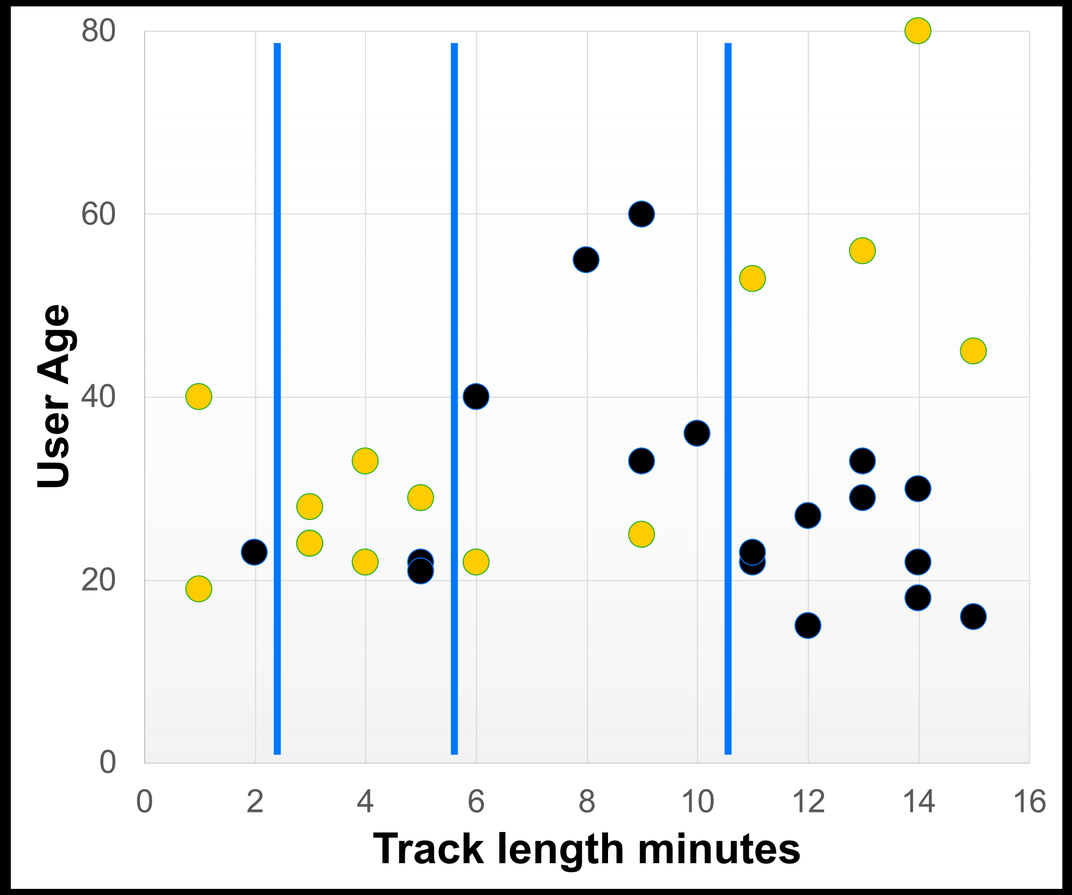


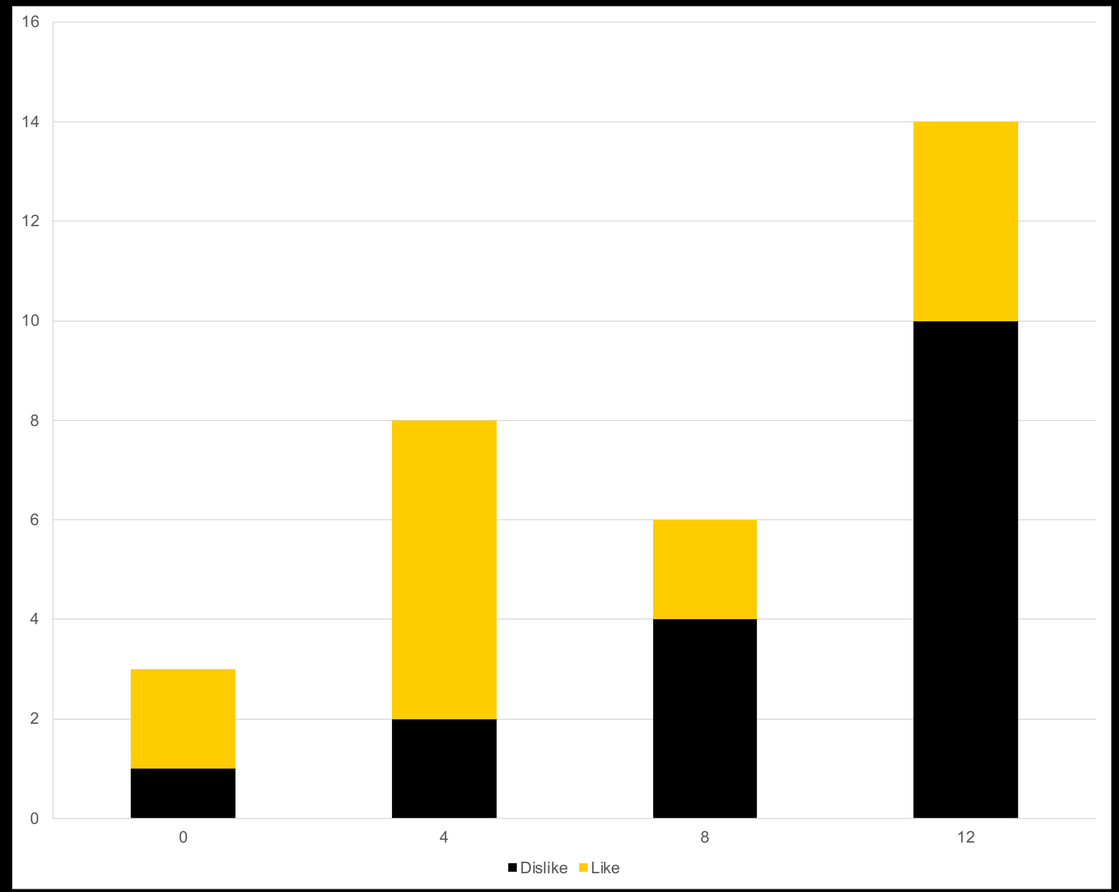
Decision tree: quantization



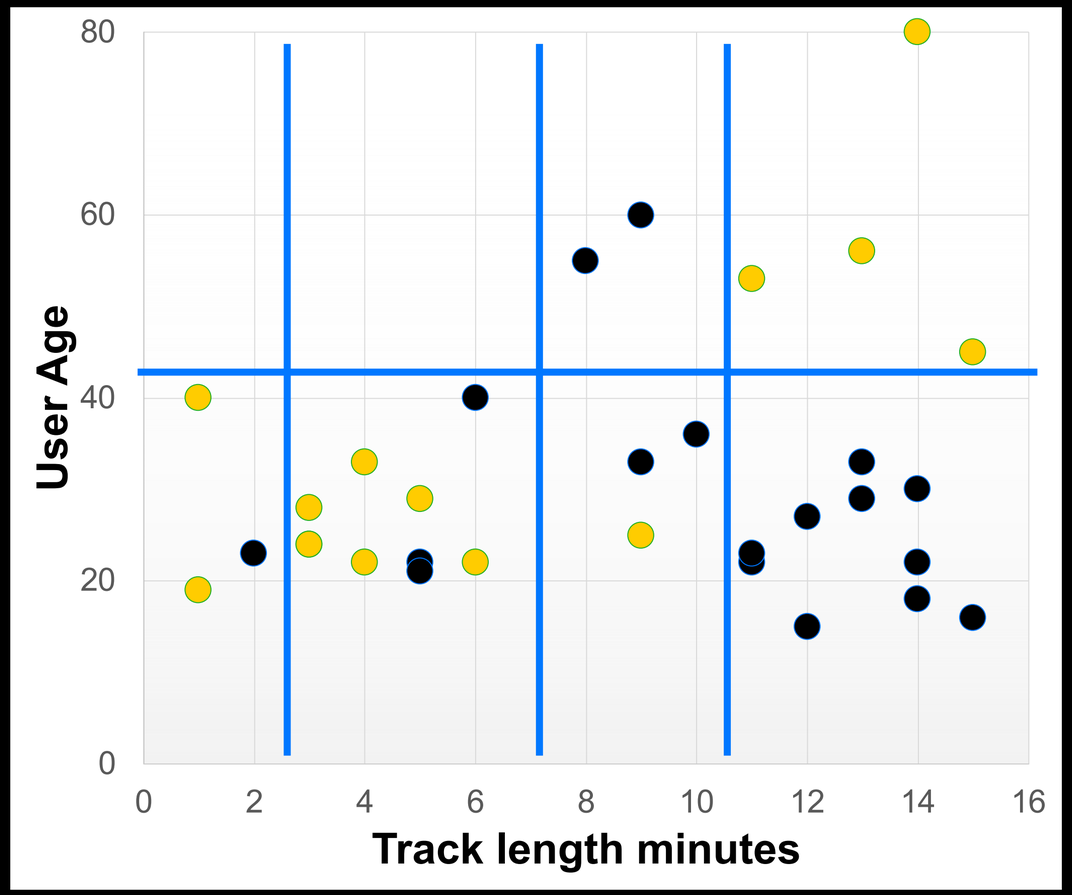


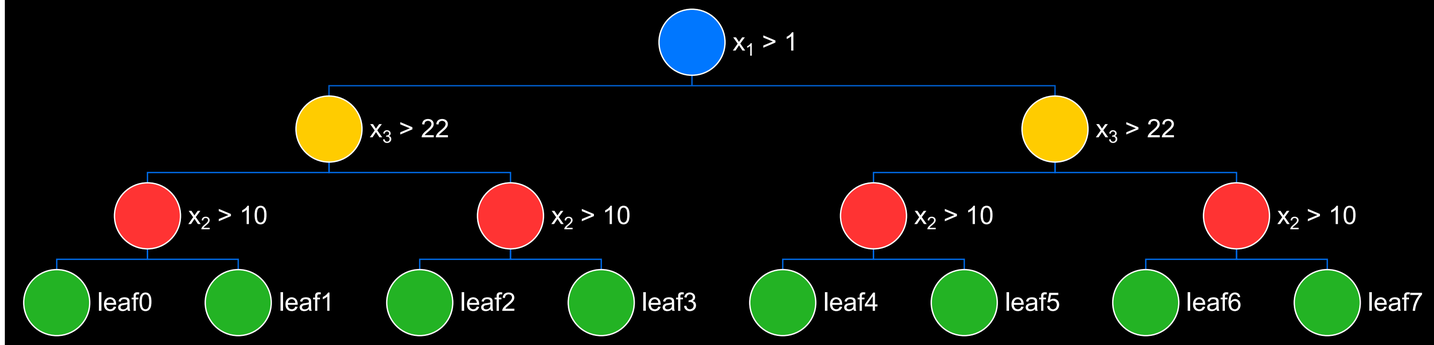
Decision tree: histograms





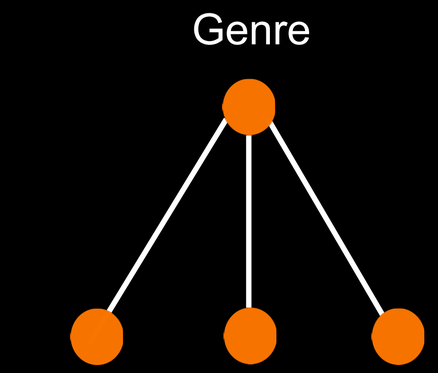
Decision tree: symmetric(oblivious) trees





Catboost supports multiple tree grow policies:
Symmetric (default option)
Lossguide - like in LightGBM
Depthwise - like in XGBoost

Categorical features handling

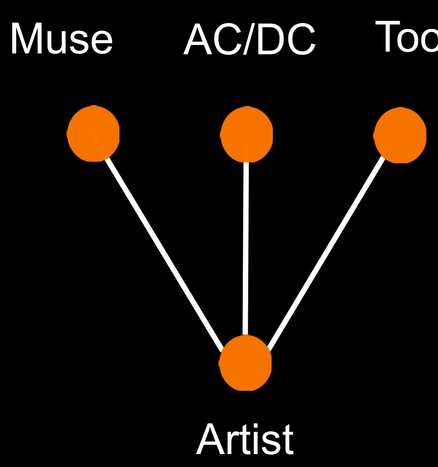


One-hot encoding

Statistics based on category

Category-based

Label based: calculated “online” (aka CatBoost-encoding)

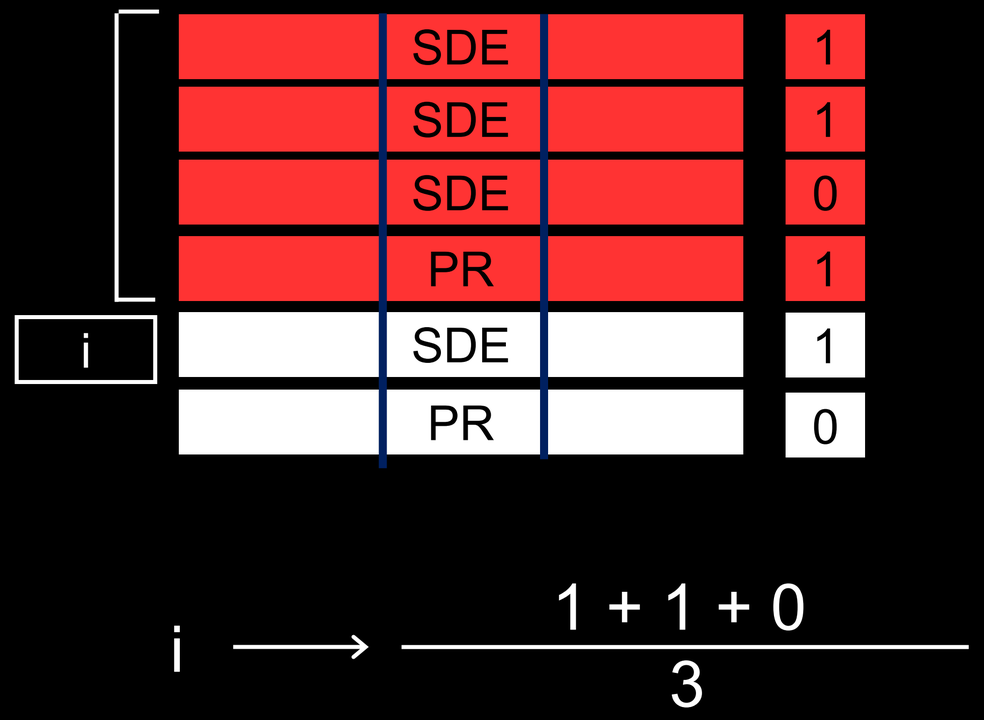


Greedy search for feature combinations

Online features for categories

$$ctr_i = \frac{countInClass + prior}{totalCount + 1}$$

For every sample “online” feature is calculated using objects with the same category before this one



Text features

at first i was afraid i was petrified kept thinkin i could never live without you by my side

Dictionary

[0, 1, 2, 3, 4, 2, 3, 5, 6, 7, 2, 8, 9, 10, 11, 12, 13, 14, 15]

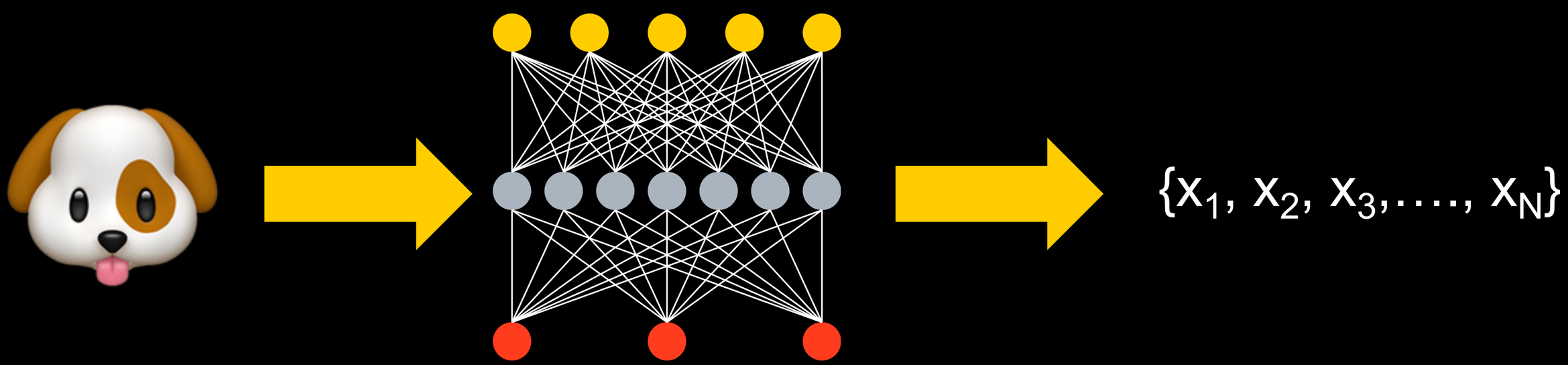
- BM25
- Naïve Bayes
- Bag-of-Words
- …

Profit from text features

Accuracy on Rotten Tomatoes

Numerical + Caterogical	+ BOW	+ Online Text Features
0.4592	0.4616	**0.4714**

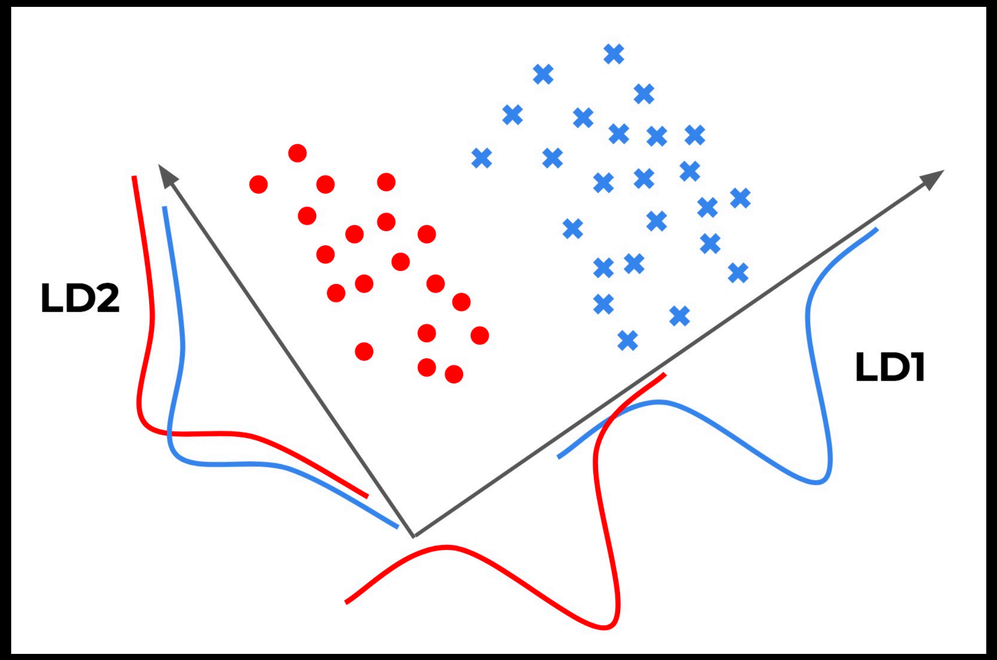
Object embeddings



For example well known text embeddings: Word2Vec, DSSM, BERT, GPT and another language models.

Experimental embeddings support

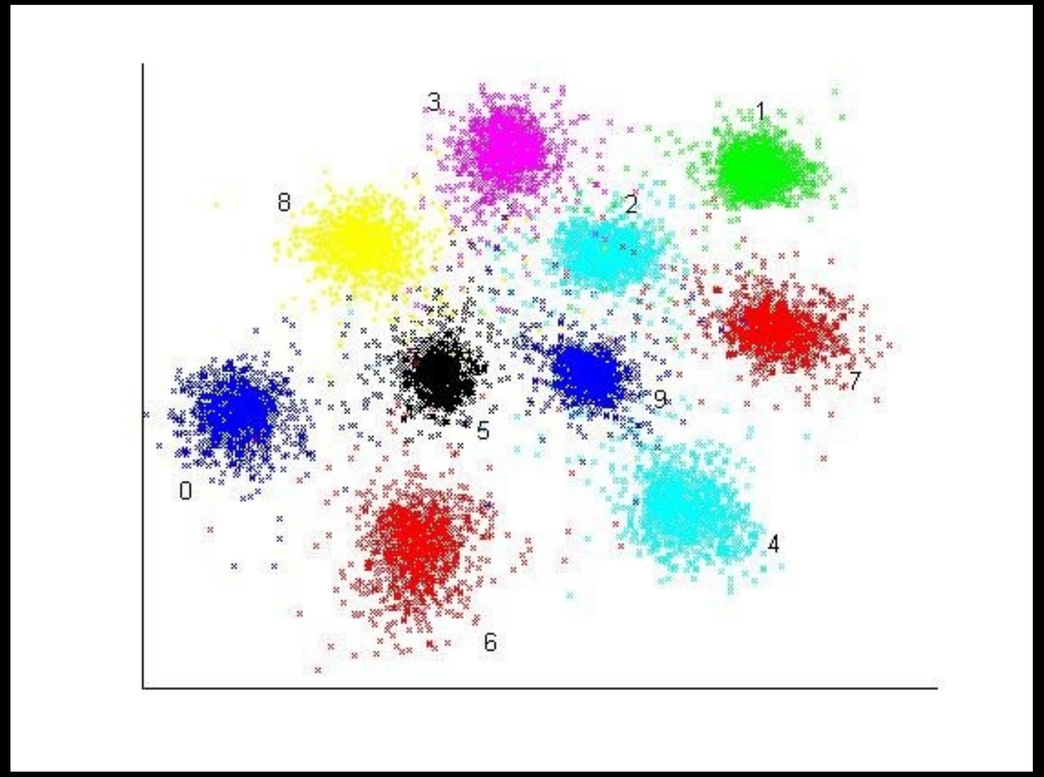
Linear discriminant analysis



$$\log P(y = k|x) = \omega_k^t x + \omega_{k0} + Cst.$$

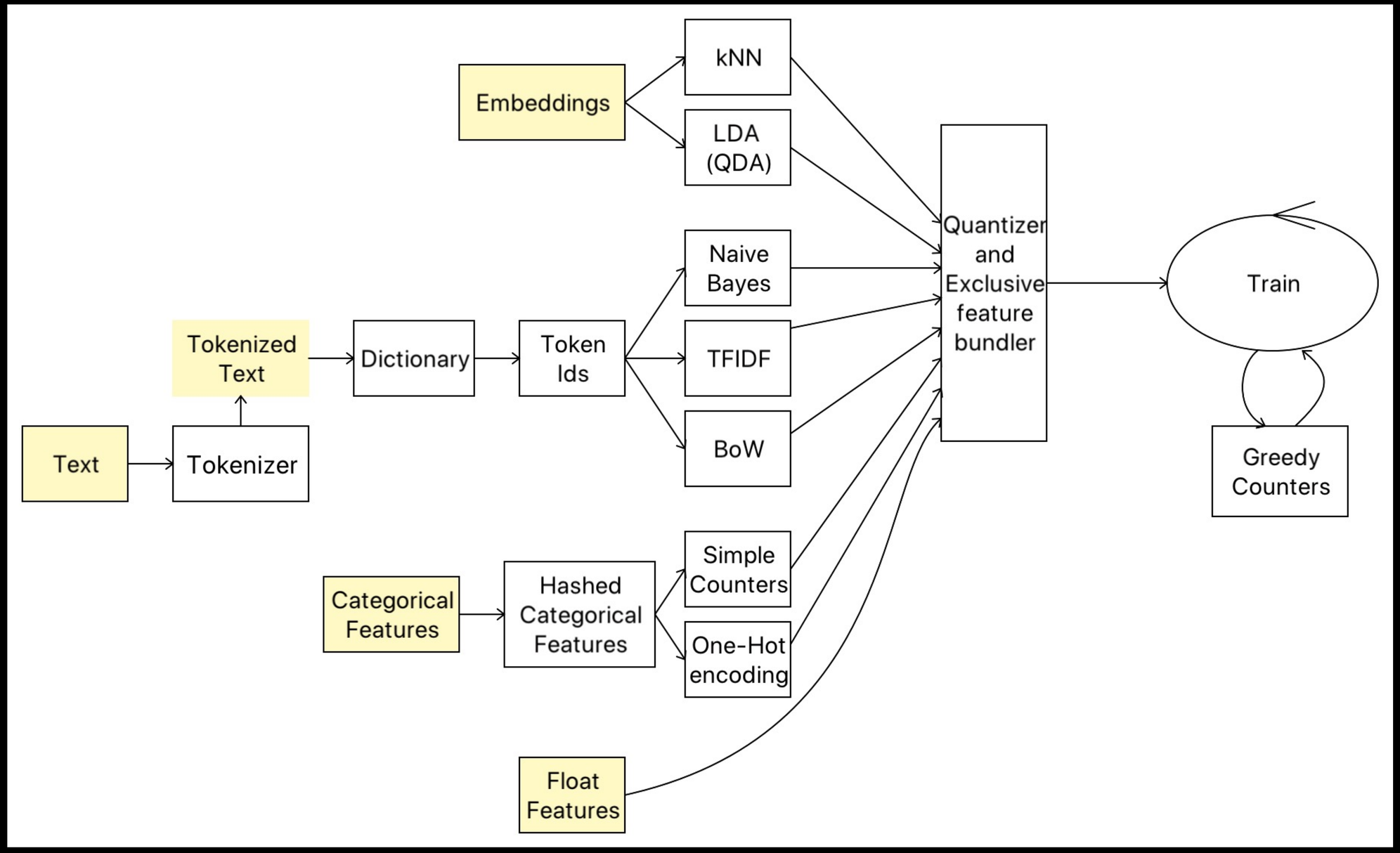
https://raw.githubusercontent.com/eigenfoo/eigenfoo.xyz/master/assets/images/lda-pic.png

kNN



A Deep Non-Linear Feature Mapping for Large-Margin kNN Classification - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Two-dimensional-embedding-of-10-000-MNIST-test-data-using-the-Deep-Neural-Network-kNN_fig9_22076648

CatBoost – bird eye view



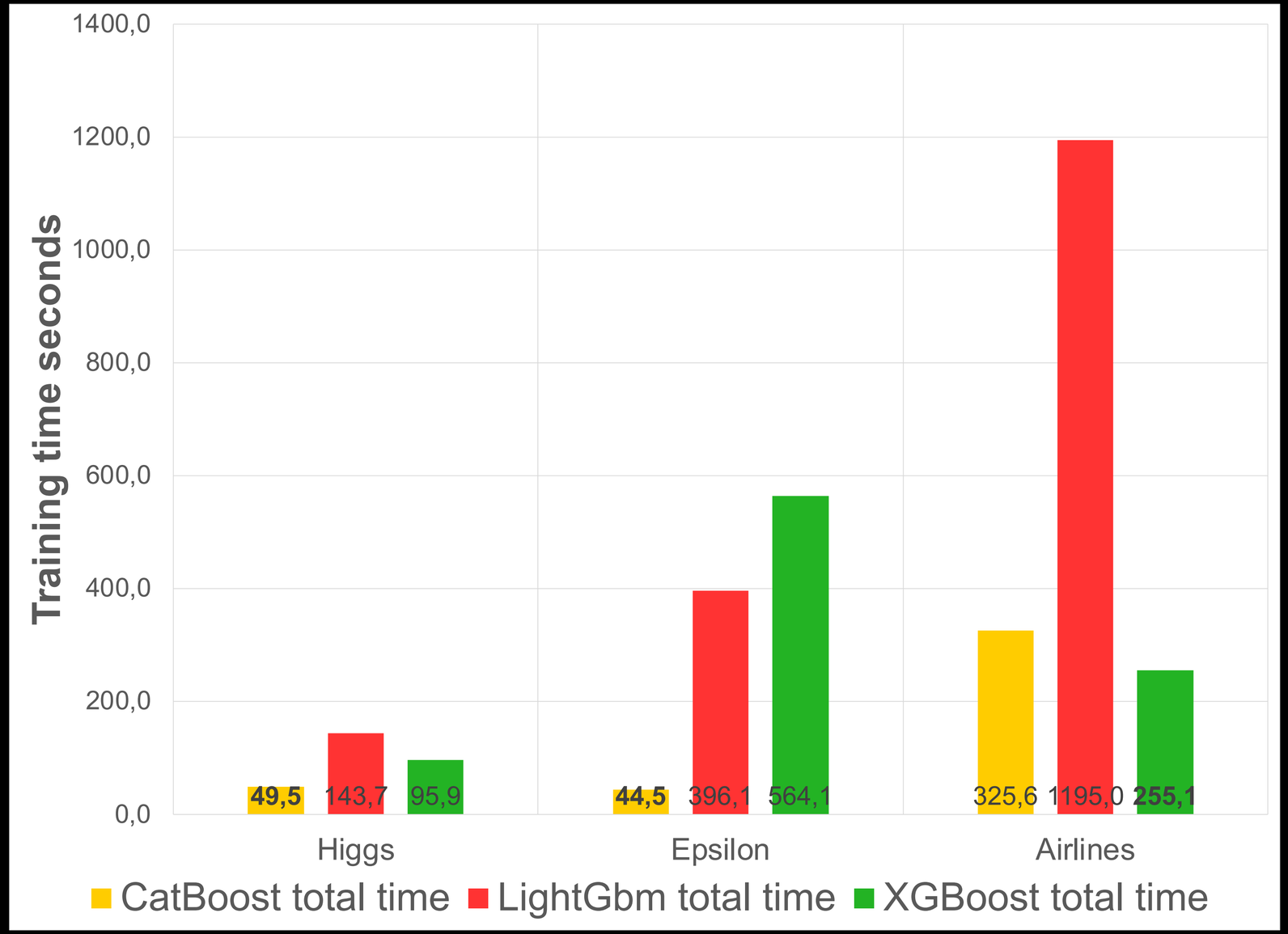
GPU support

V100 Training performance: 1000 trees

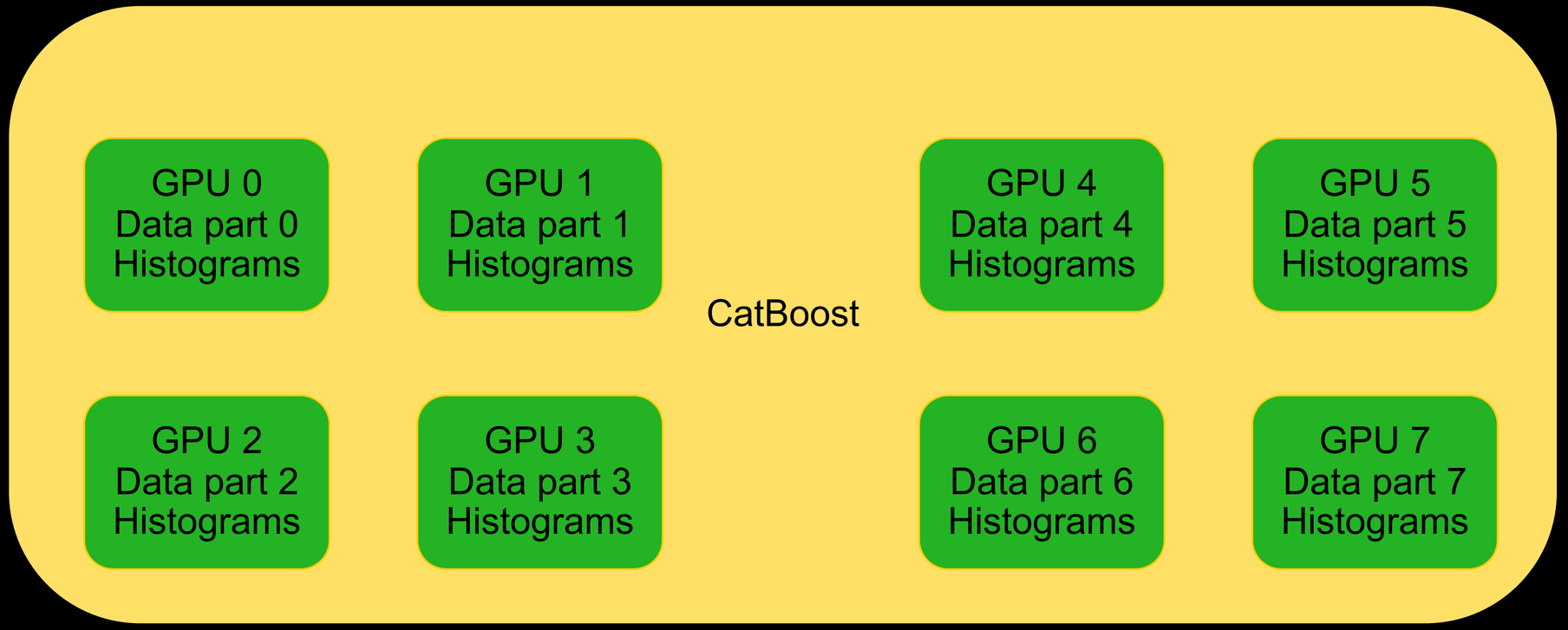
GPU training time on single NVIDIA Tesla V100

- Higgs
10.5mln objects
28 features
- Epsilon
400k objects
2k features
- Airlines
23mln objects
13 features

CatBoost v0.24.1, Lightgbm 2.3.1, XGBoost 1.0.2



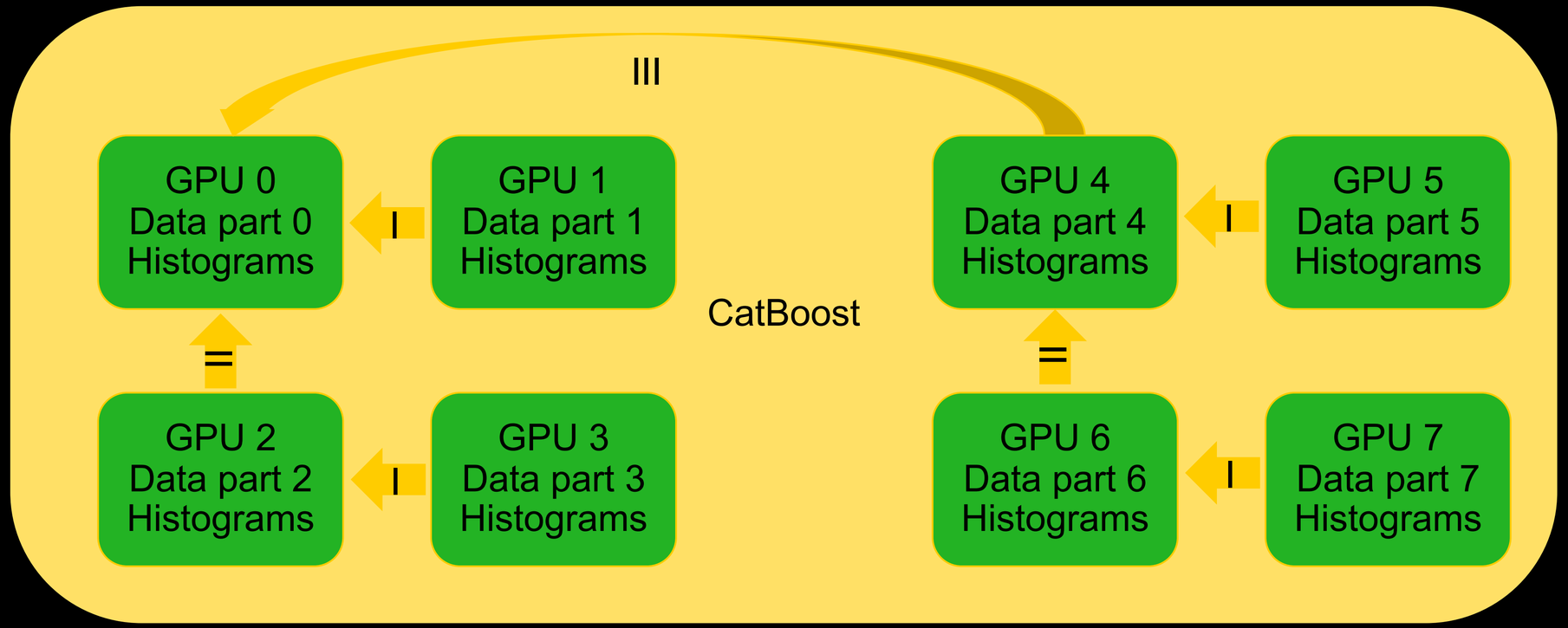
GPU training: single host multi GPU



Two different data mappings

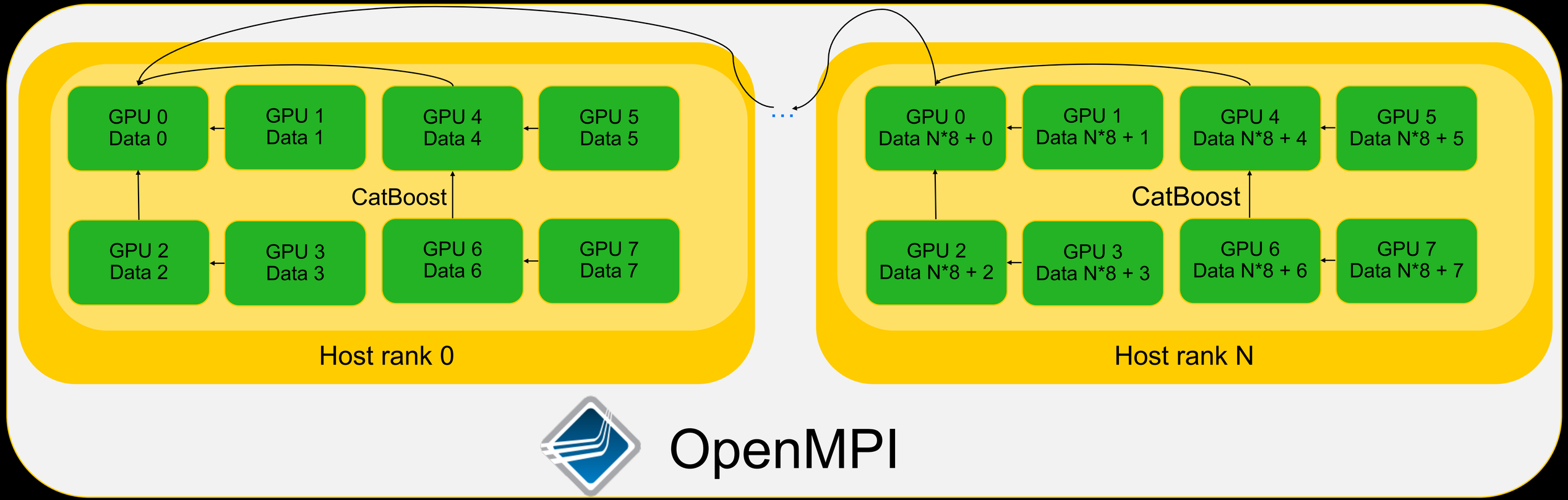
- Document parallel – dataset is splitted by row blocks
- Feature parallel – dataset is splitted by feature column blocks

GPU: histograms reduction scheme



Example of histograms reduction tree.

Multi host multi GPU



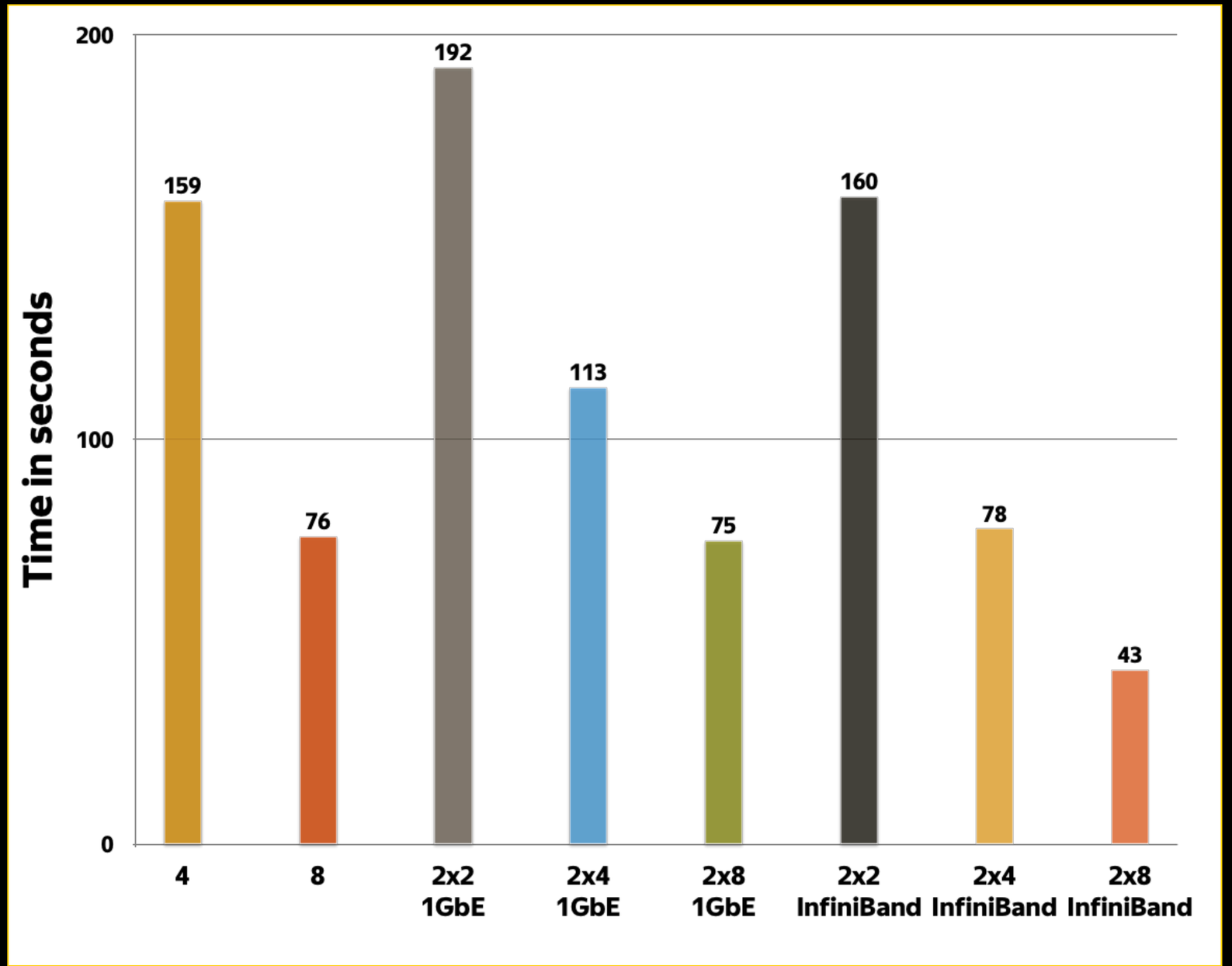
Distributed training: multi-host multi-GPU

CatBoost uses MPI for multi-host communications.
Easy to run:

```
mpirun -np $HOSTS_COUNT -bind-to none -map-by slot
/path/to/catboost fit --learn-set
quantized://path_to_quantized_train --loss-function
Logloss
```

Currenlty, distributed training reads data from `Rank0` host and distributes it across all nodes evenly.

Distributed training: scaling benchmark



CatBoost for Apache Spark

Why CatBoost for Apache Spark?

Yandex
CatBoost

Why CatBoost for Apache Spark?

Yandex
CatBoost

› When data size is too big for a single machine

Why CatBoost for Apache Spark?

Yandex
CatBoost

APACHE
Spark™

- When data size is too big for a single machine
- Processing speed

Why CatBoost for Apache Spark?

Yandex
CatBoost

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark

Why CatBoost for Apache Spark?

Yandex
CatBoost

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API is faster than Python API

Why CatBoost for Apache Spark?

Yandex
CatBoost

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API is faster than Python API
- CatBoost model training and analysis with JVM platform API (PySpark is also supported)

Why CatBoost for Apache Spark?

Yandex
CatBoost

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API is faster than Python API
- CatBoost model training and analysis with JVM platform API (PySpark is also supported)
- API is fully compatible with Spark ML library

API is fully compatible with Spark ML library

```
from pyspark.sql import SparkSession

sparkSession = (SparkSession.builder
    .master("local[*]")
    .config(
        "spark.jars.packages",
        "ai.catboost:catboost-spark_2.4_2.12:0.25-rc4"
    ).getOrCreate()
)

import catboost_spark

trainDf = sparkSession.read().load("/my_datasets/train")

classifier = catboost_spark.CatBoostClassifier(iterations=20)
classifier.write().save("/my_classifier")

model = classifier.fit(trainDf)
model.write().save("/my_model")

applyDf = sparkSession.read().load("/my_datasets/for_application")

dfWithPredictions = model.predict(applyDf)
```

CatBoost for Spark vs Competitors

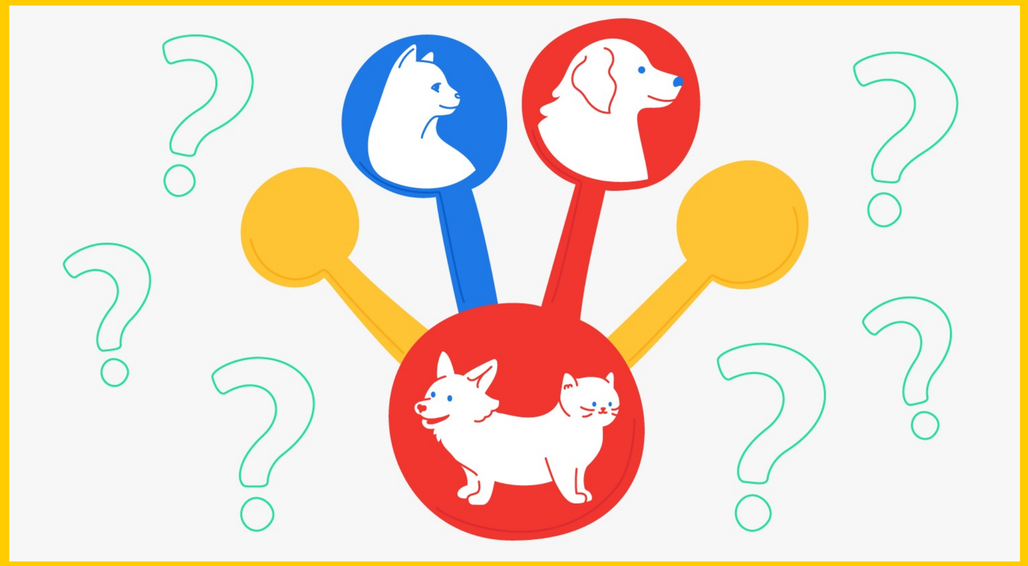
	CatBoost	LightGBM	dmlc XGBoost
GPU support	No (planned)	No	Yes
Categorical features	Yes, including CTR statistics	Yes	Only preprocessed as one hot
Pairs	Yes	No	No
Pre-quantization	Yes	No	No
PySpark support	Yes	Yes	No
SparkR support	No	Beta	No

CatBoost Spark Performance vs Competitors

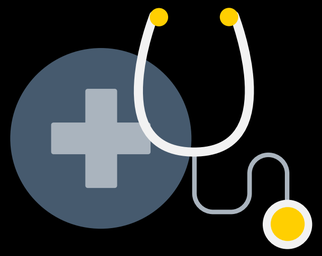
	Criteo derived 170 m samples 65 features		Epsilon 400 k samples 2000 features		Higgs 10.5 m samples 28 features	
	total	per iter	total	per iter	total	per iter
CatBoost	53 m 52 s	2.8 s	1 h 5 m	3.7 s	7 m 40 s	0.31 s
LightGBM	59 m	3.5 s	2 h 36 m	9.4 s	2 h 25 m	8.7 s
dmlc XGBoost	est. 8 h	28.9 s	17 m 25 s	1 s	10 m	0.6 s

- Cluster configuration – 16 nodes x 16 cores
- Time on 1000 iterations, total time includes preprocessing

Uncertainty estimation

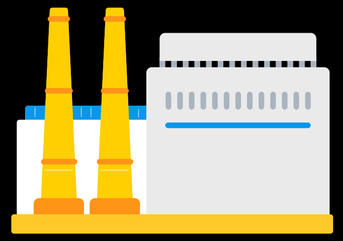


Whom it may concern?

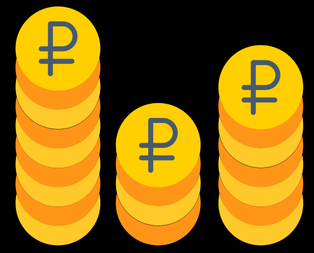


Medcine

Sales prediction



Production processes



Finance & Credit

Data uncertainty

For classification – data unceratainty is entropy of $H(\bar{p}) = -\bar{p}\ln\bar{p} - (1-\bar{p})\ln(1-\bar{p})$ prediction

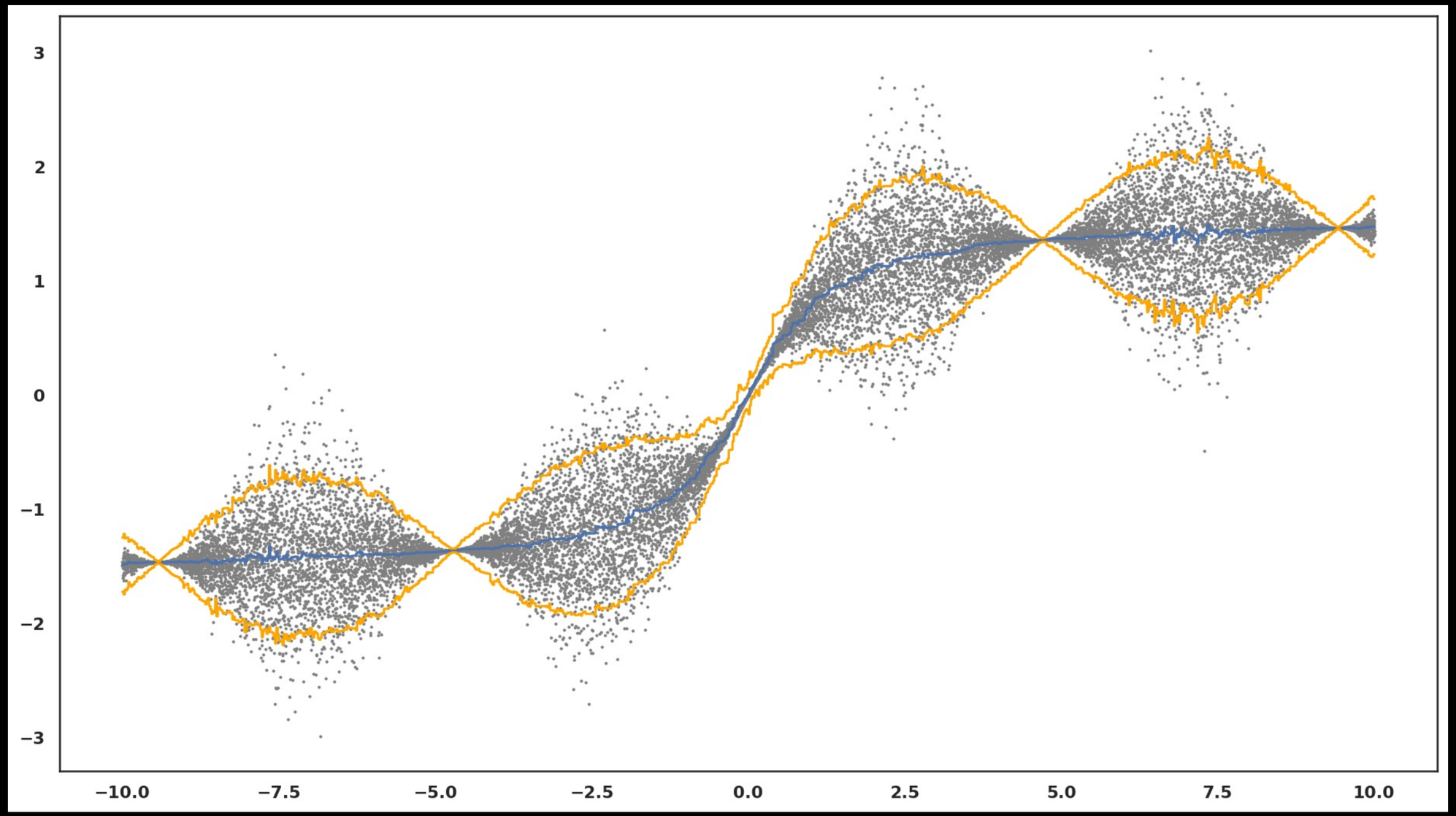
For regression data uncertainty could be estimated with Natural Gradient Boosting (NGBoost)
This loss function is named RMSEWithUncertainty in CatBoost library

$$-\frac{1}{N}\sum_{i=1}^{N}\log p(t_i|a_i) = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{1}{2\pi\sigma^2}\exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)\right) = C + \frac{1}{N}\sum_{i=1}^{N}\left(a_{i,1} + \frac{1}{2}\exp(-2a_{i,1}(t_i - a_{i,0})^2)\right),$$

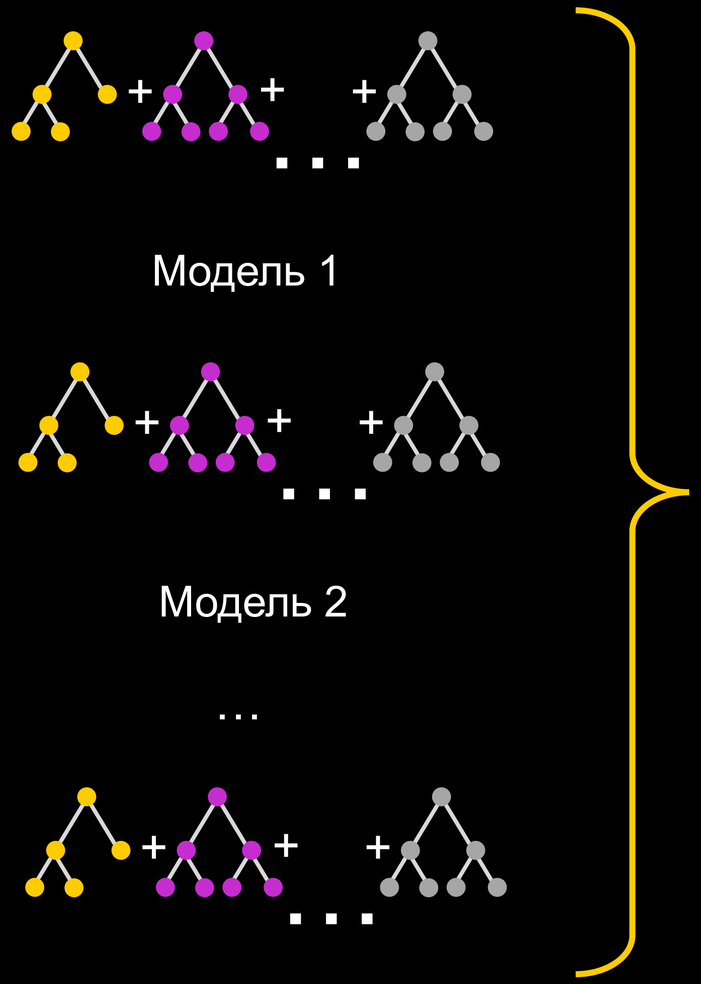
where t is target, a -- 2-dimensional approx a_0 is target predict, a_1 is $\log\sigma$ predict, and p has normal distribution

$p(t|a) = \mathcal{N}(y|a_0, e^{2a_1}) = \mathcal{N}(y|\mu,\sigma^2) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$:

Data uncertainty example



Knowledge uncertainty: ensemble approach



Regression

Prediction variance = knowledge uncertainty

Classification

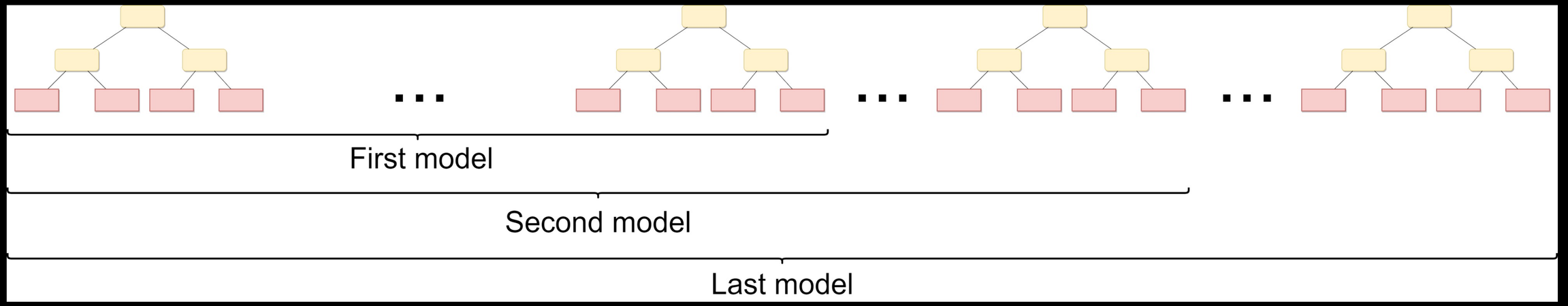
Mean entropy of predictions = full uncertainty

$$H(\bar{p}) = -\bar{p}\ln\bar{p} - (1-\bar{p})\ln(1-\bar{p})$$

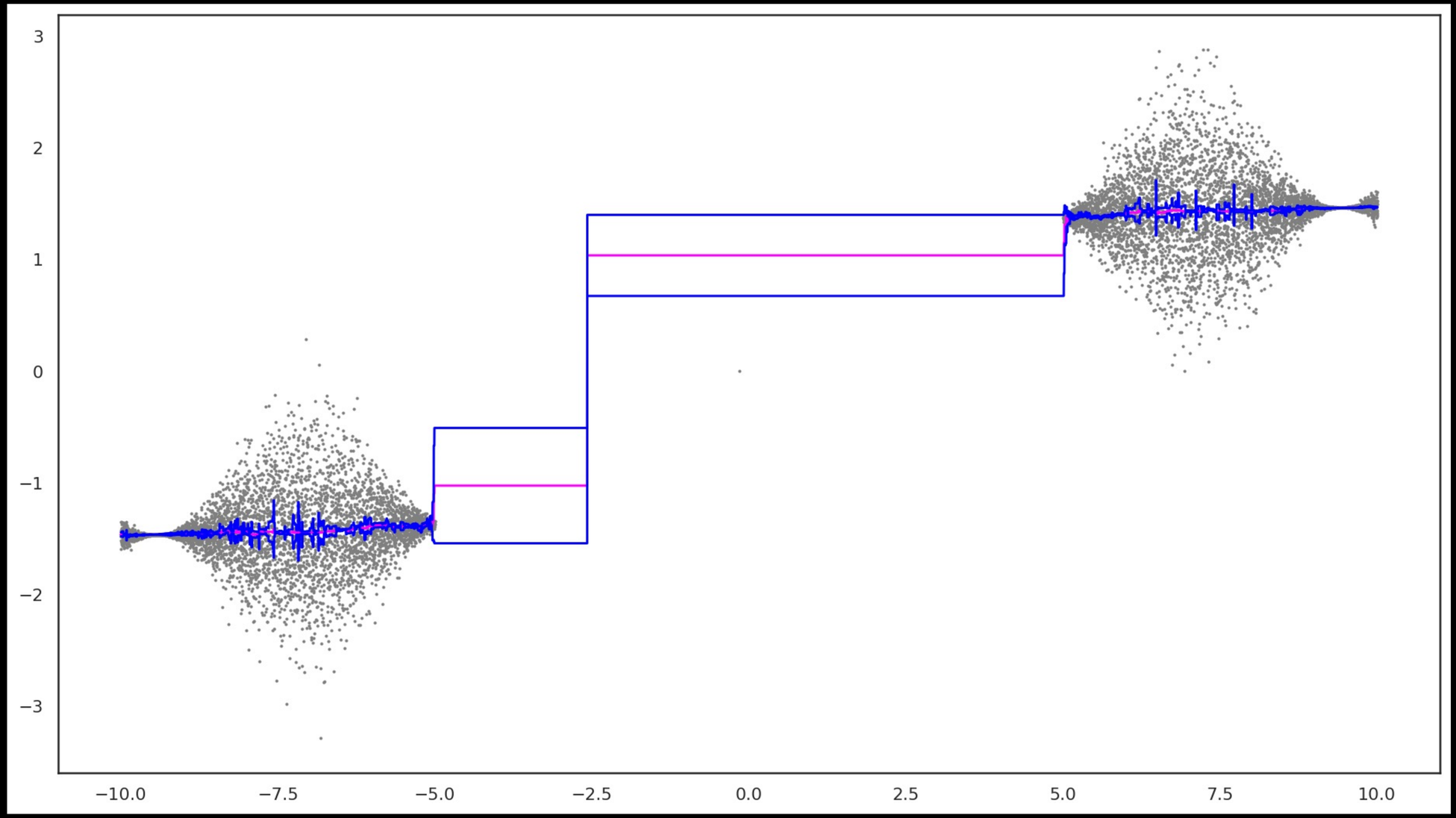
The entropy of mean prediction – data uncertainty

$$H(\bar{p}) = -\bar{p}\ln\bar{p} - (1-\bar{p})\ln(1-\bar{p}) \text{ where } \bar{p} = \frac{\sum_i p_i}{N}$$

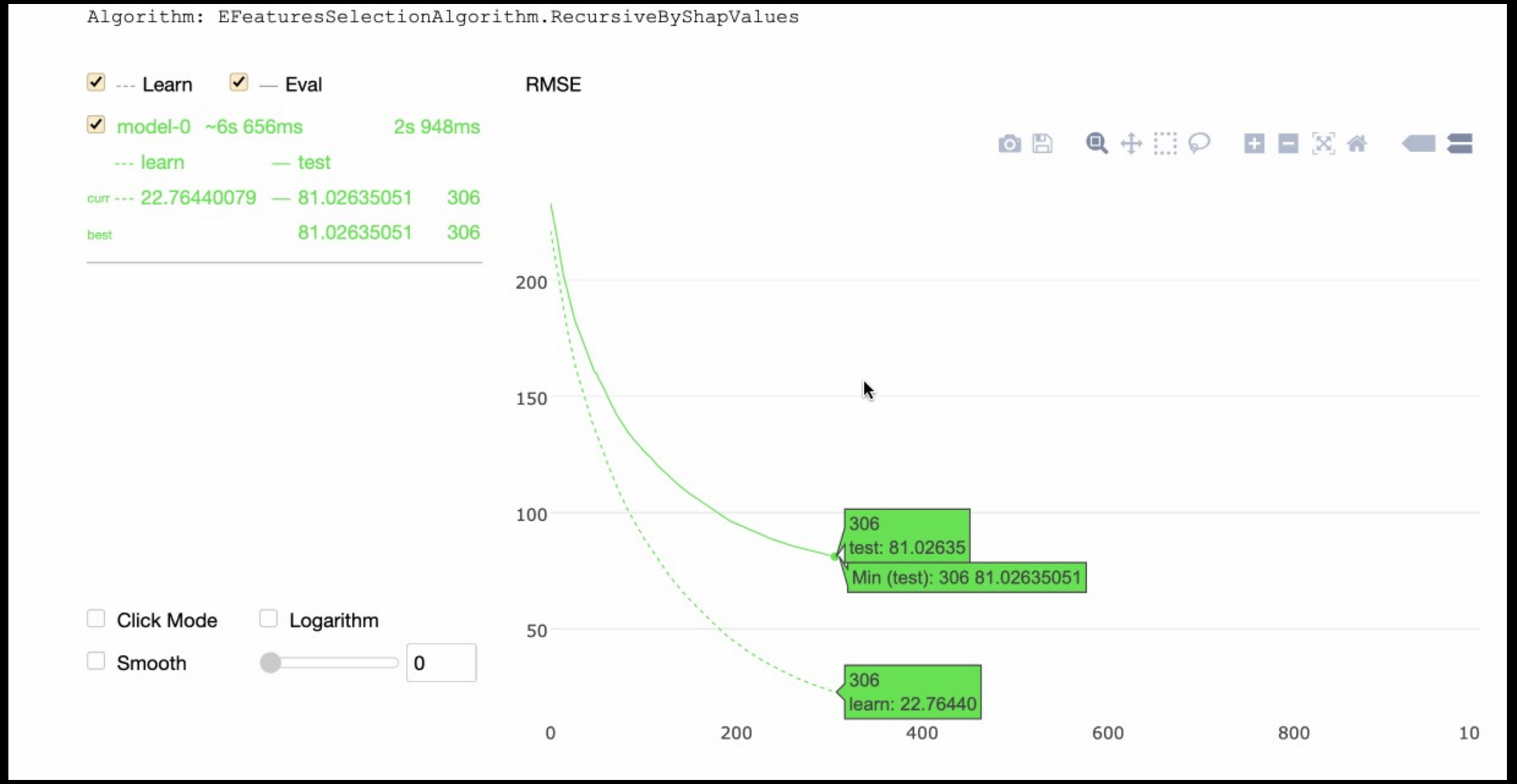
Knowledge uncertainty estimation with virtual ensembles



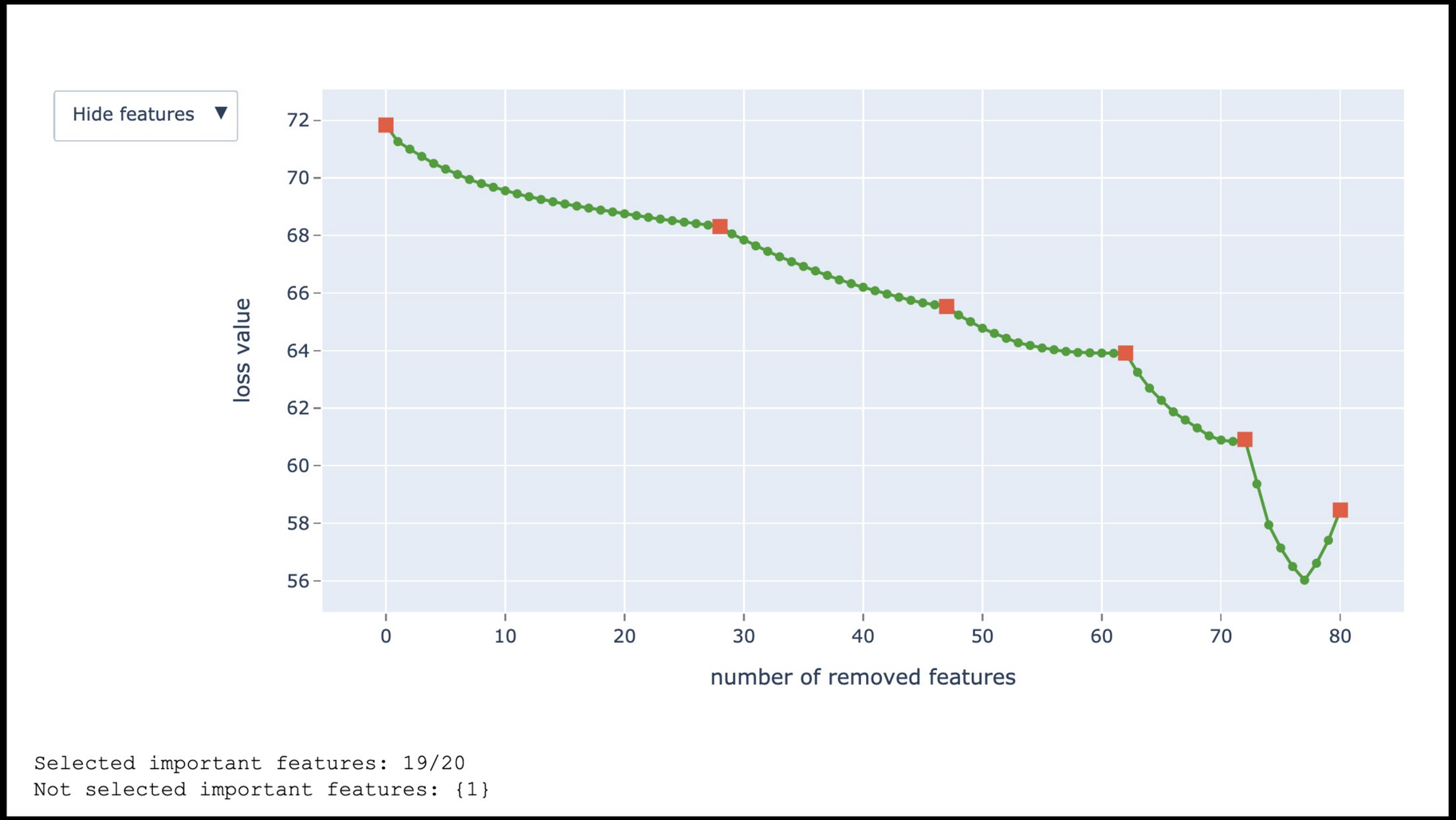
Knowledge uncertainty example



Recursive feature elimination: real time visualization



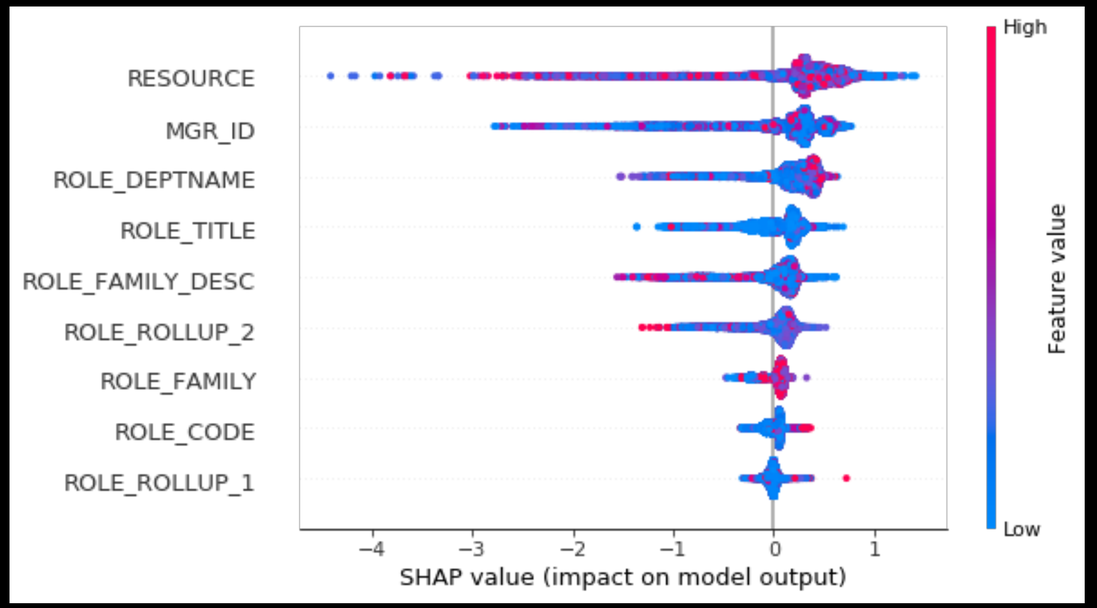
Recursive feature elimination: result visualization

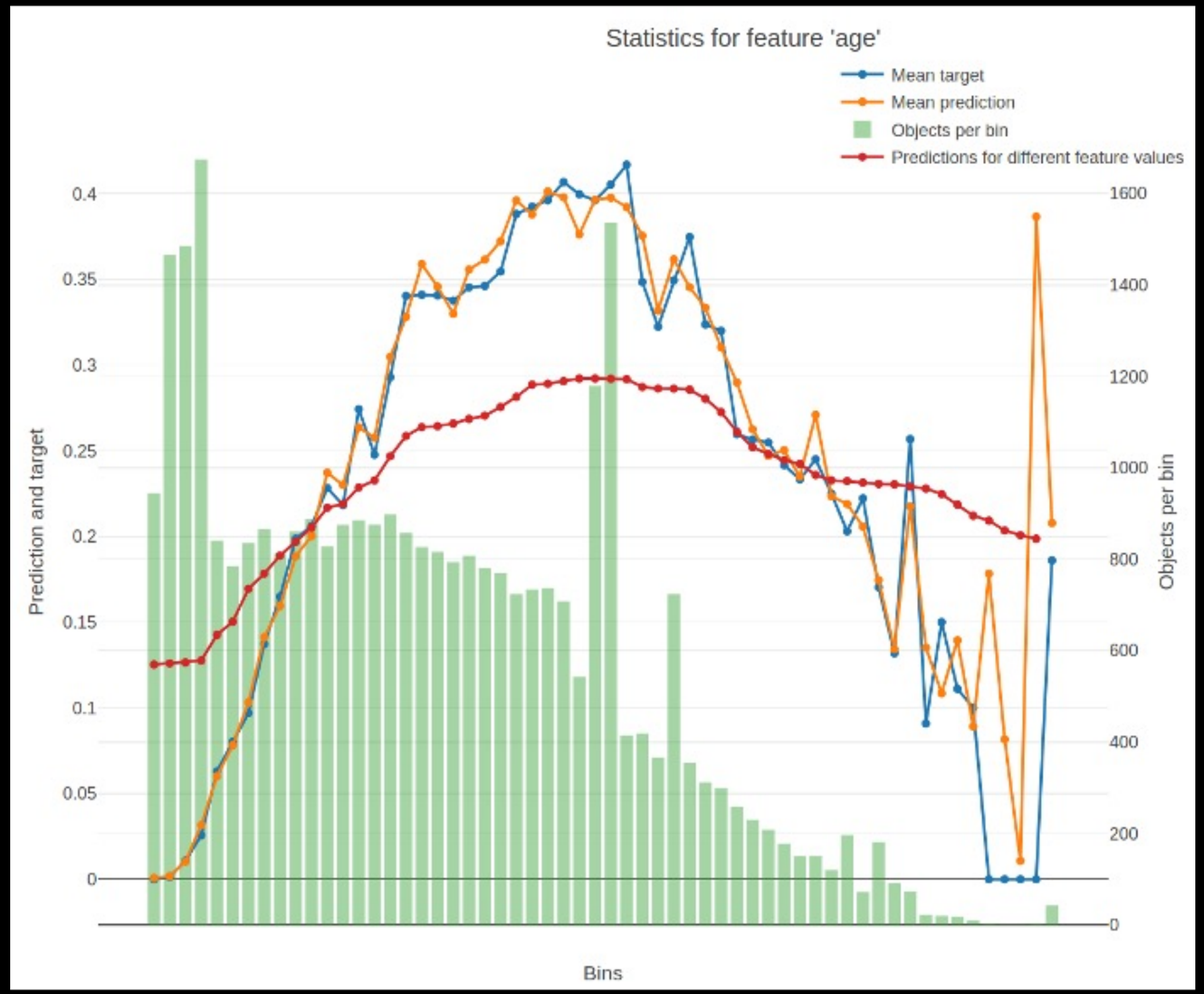



```
Selected important features: 19/20
Not selected important features: {1}
```

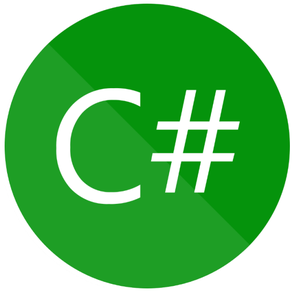
Model Analysis tools

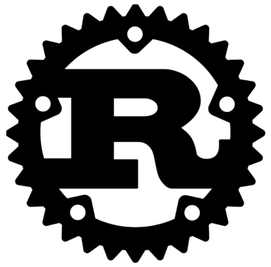
- Per feature model analysis charts
- New types of feature importance
- Tree visualization
- Ranking analysis

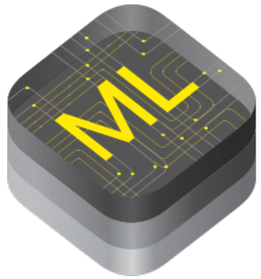




Integration in production







Plans

- CatBoost categorical feature encoder available to use outside of trained CatBoost model
- Solve discrepancy in quality between CPU & GPU version
- Support GPU for Spark training
- Better R library support – solve lot’s of persisting problems,
- Make our HNSW implementation as part of CatBoost library
- Faster and more handy embeddings support
- Model & counters finetuning on new data for trained model

CatBoost: usage examples

Known CatBoost usages

- Recommendations at Netflix
- Hotel ranking in Aviasales
- Protection against bots in CloudFlare
- Particle classification in CERN
- Medical research at University of NSW Sydney
- Destination prediction in Careem taxi service
- ML competitions on Kaggle 😊

Yandex.Search

Task?

- Search document order prediction

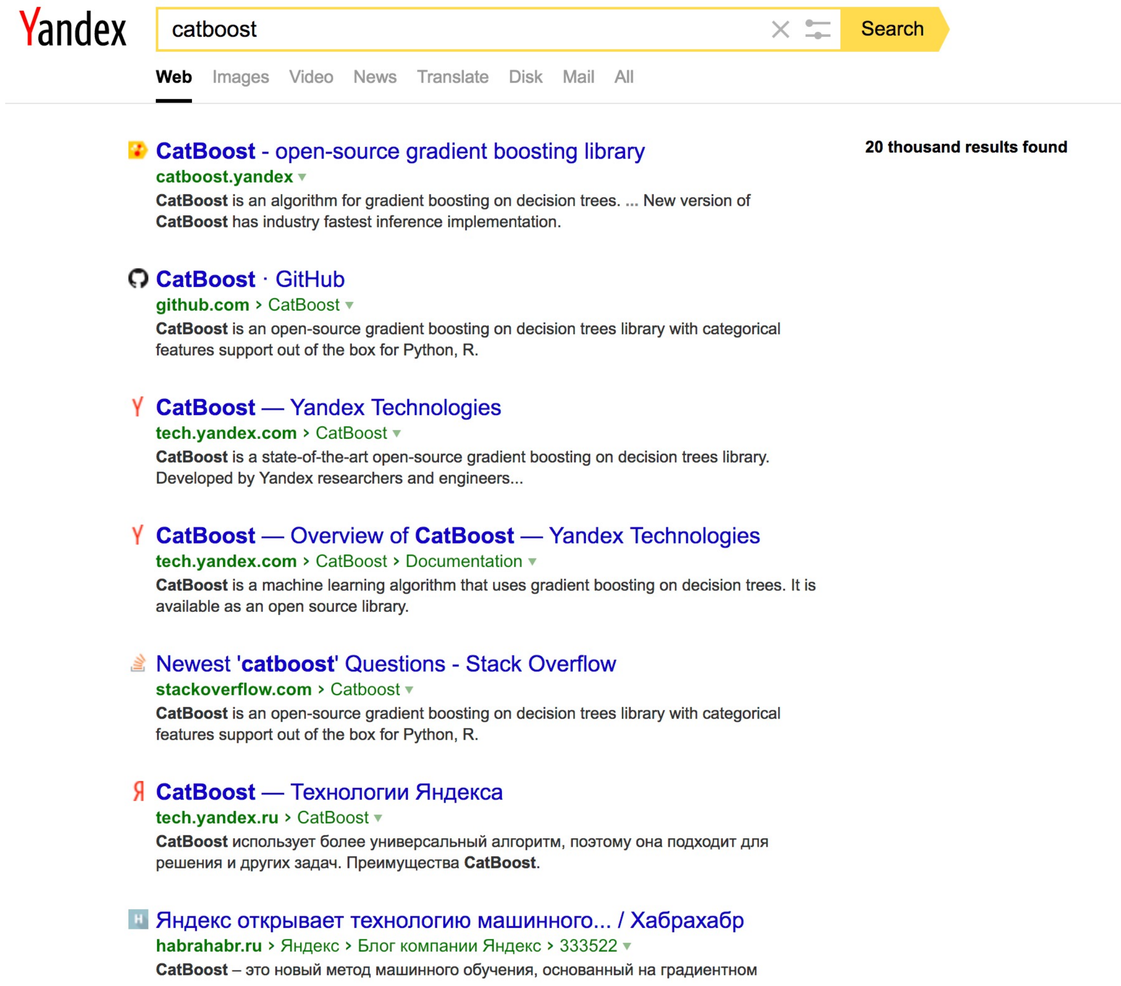
Task type: ranking

Dataset features:

- Classic features (PageRank, BM25 and others)
- Neural Networks output

CatBoost features used:

- YetiRankPairwise target
- Distributed GPU training
- Model blending
- Feature importance analysis
- Ranking analysis



Yandex.Weather

Task?

- Cloudiness type and temperature prediction

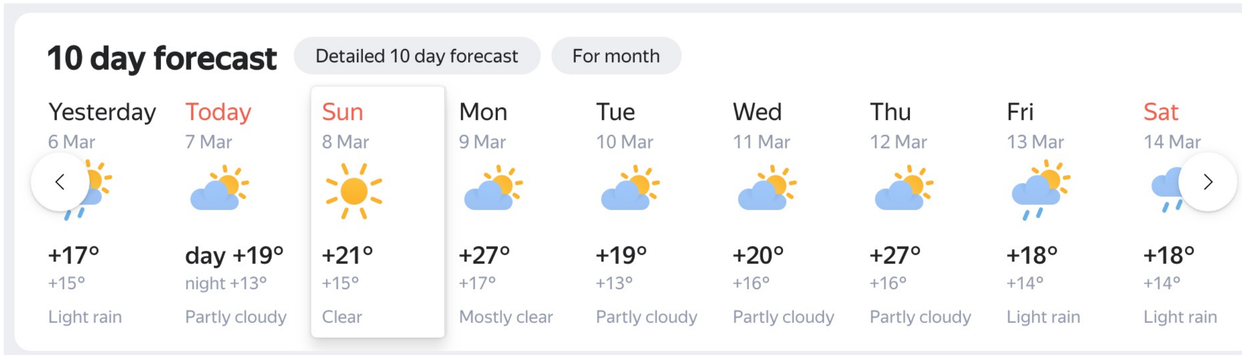
Task type: multiclassification and regression

Dataset features

- Physical weather model output
- Neural network output
- Online-data from weather stations
- Weather historical data

CatBoost features used:

- Multiclassification target and RMSE (for temperature)
- GPU training
- Feature importance analysis
- Training process visualization



Questions?

Kirillov Stanislav

Head of ML systems group @ Yandex

- catboost.ai
- github.com/catboost
- twitter.com/CatBoostML
- t.me/catboost_en, t.me/catboost_ru
- ods.ai => slack => tool_catboost chanel