ML-Тренировки tabular data, часть 1

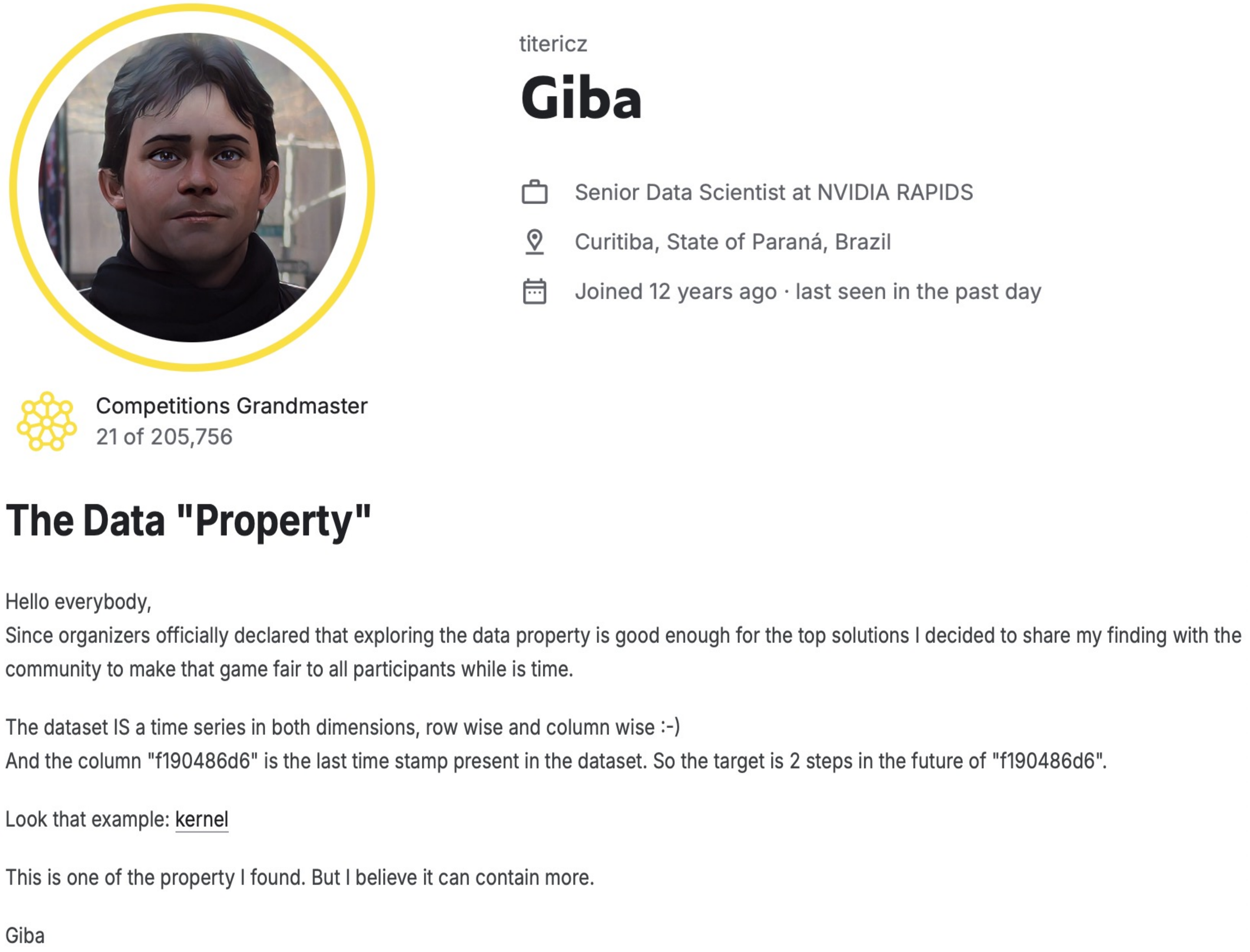
Интро

Программа тренировки:
- сборка стартера
- развитие решения
- decision trees algorithm
- deep learning algorithm
- библиотека pytorch

Сборка стартера

Pipeline stages:
- data collection
- data split
- features engineering
- features transformation
- features treatment
- features selection
- model trainer

data collection



titericz

Giba

Senior Data Scientist at NVIDIA RAPIDS

Curitiba, State of Paraná, Brazil

Joined 12 years ago · last seen in the past day

Competitions Grandmaster
21 of 205,756

The Data "Property"

Hello everybody,
Since organizers officially declared that exploring the data property is good enough for the top solutions I decided to share my finding with the community to make that game fair to all participants while is time.

The dataset IS a time series in both dimensions, row wise and column wise :-)
And the column "f190486d6" is the last time stamp present in the dataset. So the target is 2 steps in the future of "f190486d6".

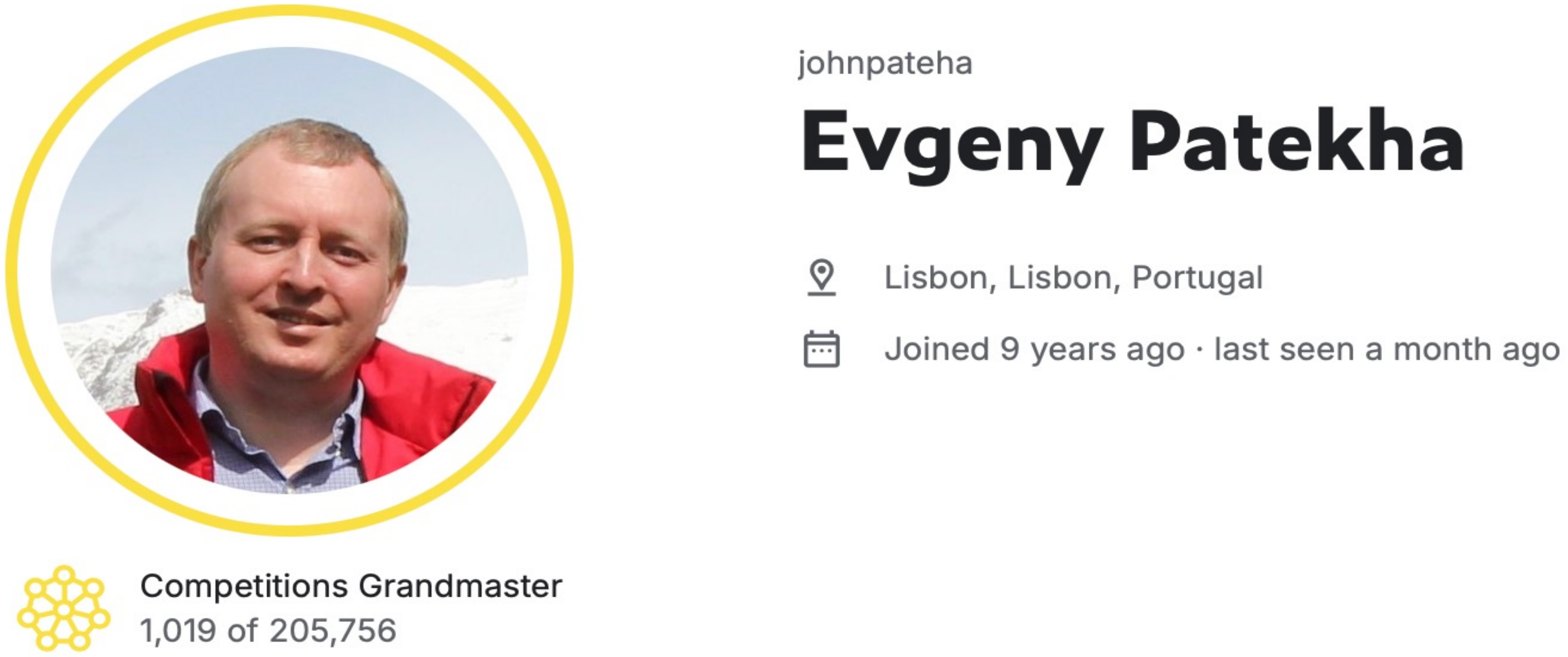
Look that example: kernel

This is one of the property I found. But I believe it can contain more.

Giba

https://www.kaggle.com/competitions/santander-value-prediction-challenge/discussion/61329

data collection



johnpateha

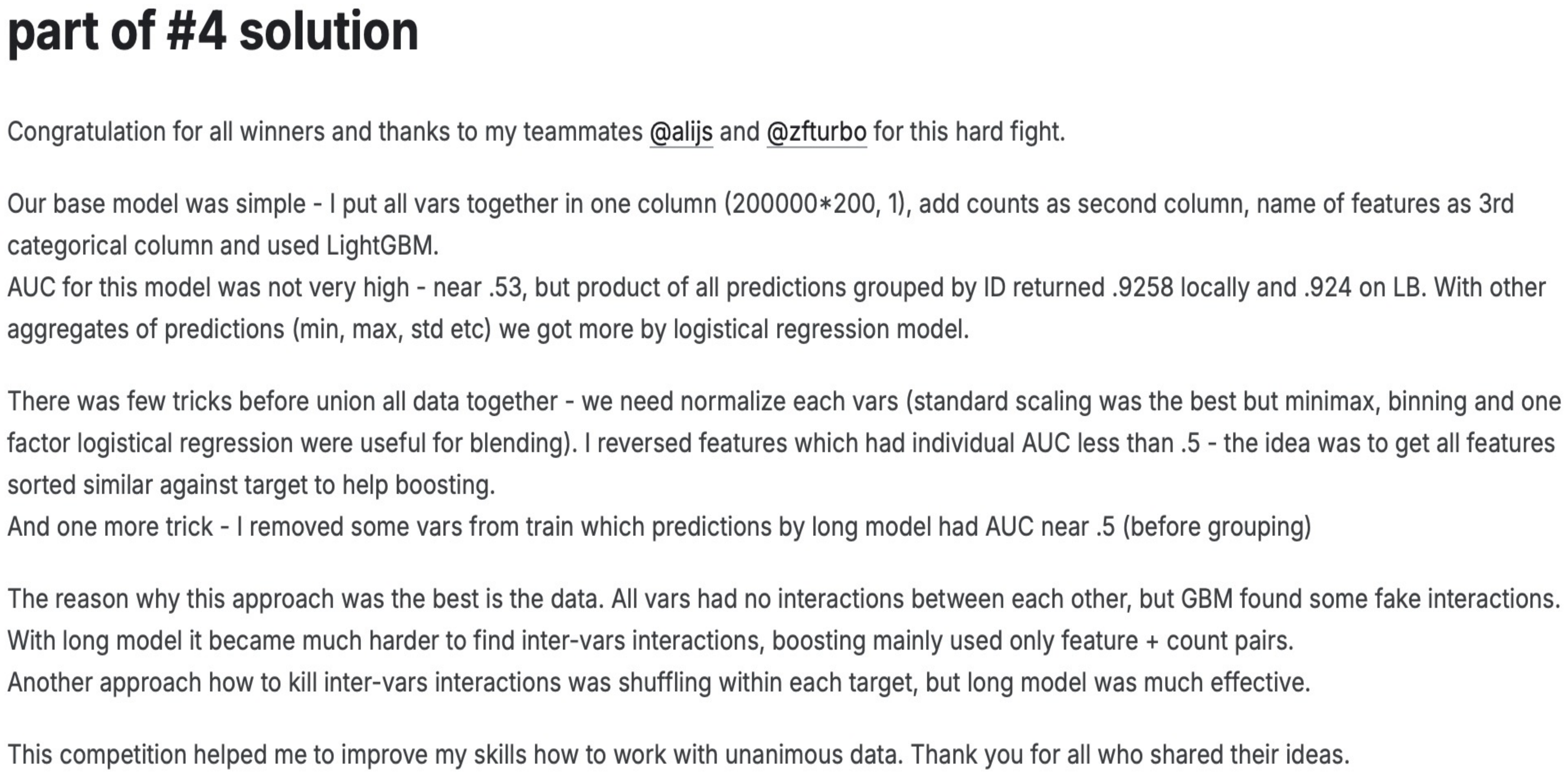
Evgeny Patekha

Lisbon, Lisbon, Portugal

Joined 9 years ago · last seen a month ago

Competitions Grandmaster

1,019 of 205,756



part of #4 solution

Congratulation for all winners and thanks to my teammates @alijs and @zfturbo for this hard fight.

Our base model was simple - I put all vars together in one column (200000*200, 1), add counts as second column, name of features as 3rd categorical column and used LightGBM.
AUC for this model was not very high - near .53, but product of all predictions grouped by ID returned .9258 locally and .924 on LB. With other aggregates of predictions (min, max, std etc) we got more by logistical regression model.

There was few tricks before union all data together - we need normalize each vars (standard scaling was the best but minimax, binning and one factor logistical regression were useful for blending). I reversed features which had individual AUC less than .5 - the idea was to get all features sorted similar against target to help boosting.
And one more trick - I removed some vars from train which predictions by long model had AUC near .5 (before grouping)

The reason why this approach was the best is the data. All vars had no interactions between each other, but GBM found some fake interactions.
With long model it became much harder to find inter-vars interactions, boosting mainly used only feature + count pairs.
Another approach how to kill inter-vars interactions was shuffling within each target, but long model was much effective.

This competition helped me to improve my skills how to work with unanimous data. Thank you for all who shared their ideas.

https://www.kaggle.com/competitions/santander-customer-transaction-prediction/discussion/88970

data split

Базовые подходы:
- train, valid, test
- oof-scheme
- oot-scheme
- groups
- cv

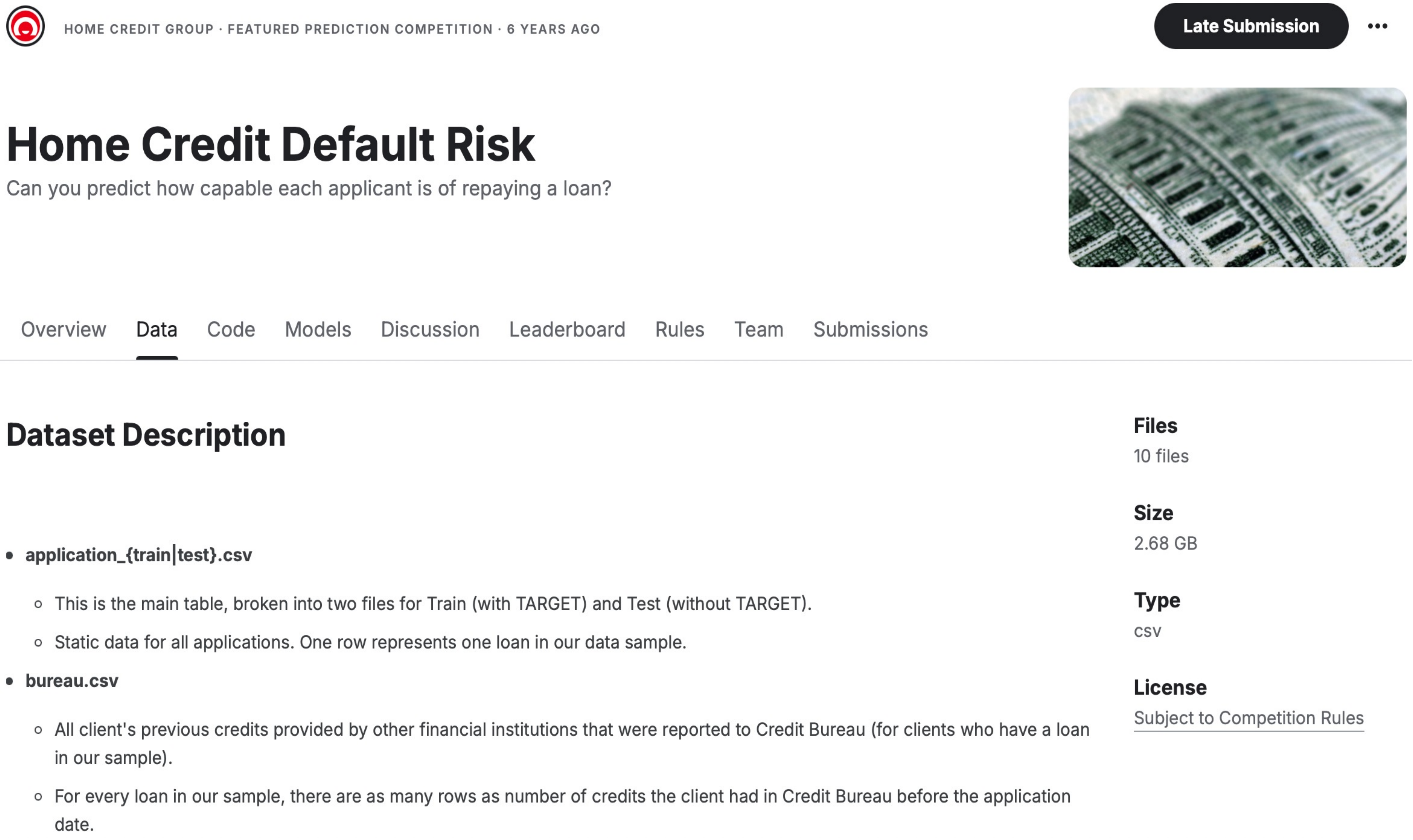
Примеры из соревнований:
- individual model
(https://www.kaggle.com/competitions/sberbank-russian-housing-market/discussion/35684,
https://www.kaggle.com/code/kyakovlev/m5-three-shades-of-dark-darker-magic)
- target outliers
(https://www.kaggle.com/competitions/elo-merchant-category-recommendation/overview)

features engineering

Базовые подходы:
- individual level
- group aggrs
- model

Примеры из соревнований:
- models ensemble
(https://www.kaggle.com/competitions/elo-merchant-category-recommendation/discussion/82055)
- pseudo-labels
(https://www.kaggle.com/code/kneroma/m5-first-public-notebook-under-0-50)
- top1 places
(https://www.kaggle.com/competitions/home-credit-credit-risk-model-stability/discussion/508337,
https://www.kaggle.com/competitions/home-credit-default-risk/discussion/64821)

features engineering



Home Credit Default Risk

Can you predict how capable each applicant is of repaying a loan?

Dataset Description

- **application_{train|test}.csv**
  - This is the main table, broken into two files for Train (with TARGET) and Test (without TARGET).
  - Static data for all applications. One row represents one loan in our data sample.
- **bureau.csv**
  - All client's previous credits provided by other financial institutions that were reported to Credit Bureau (for clients who have a loan in our sample).
  - For every loan in our sample, there are as many rows as number of credits the client had in Credit Bureau before the application date.

Files
10 files

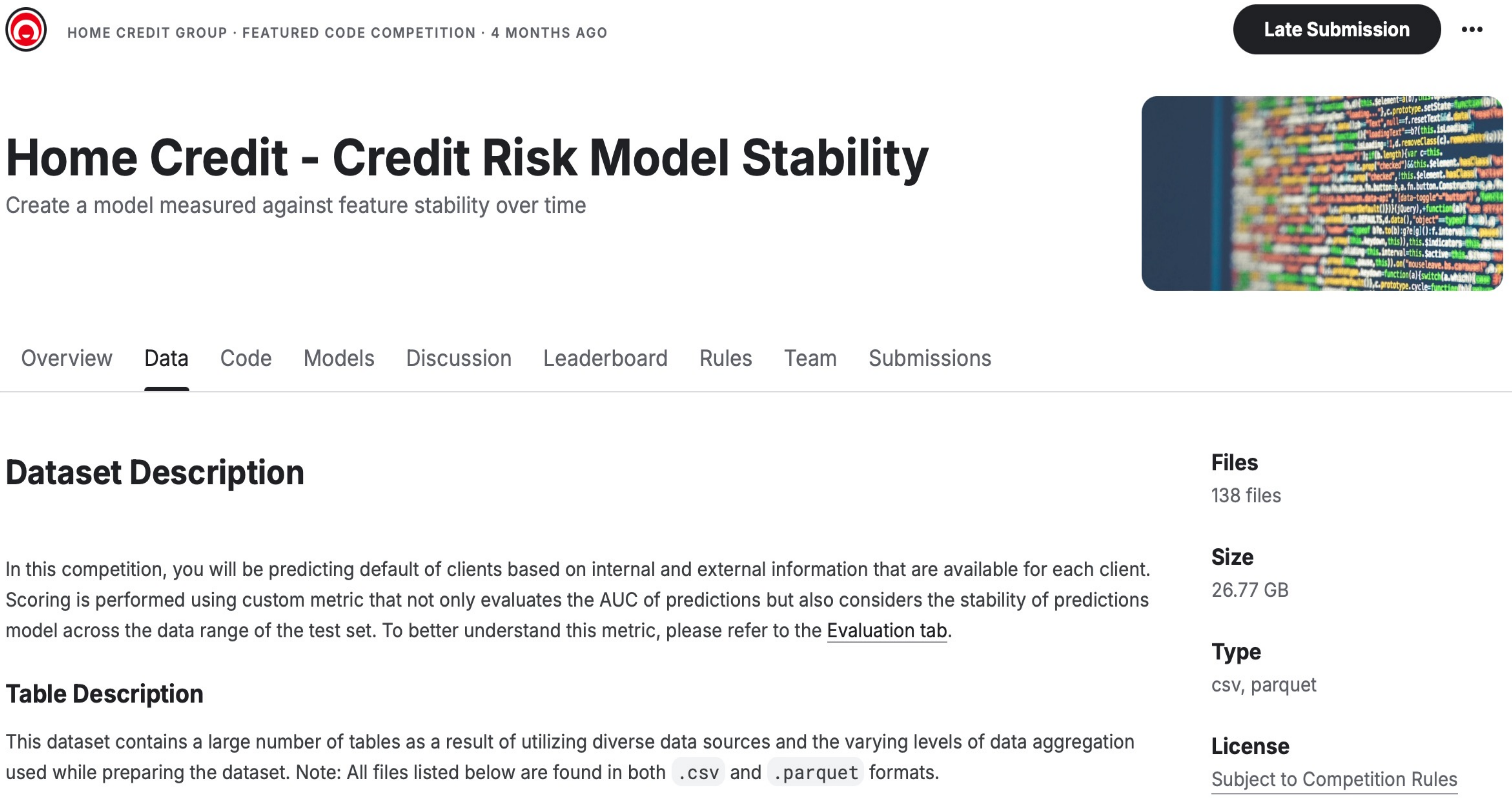
Size
2.68 GB

Type
csv

License
Subject to Competition Rules

(https://www.kaggle.com/competitions/home-credit-default-risk/data)

features engineering



HOME CREDIT GROUP · FEATURED CODE COMPETITION · 4 MONTHS AGO

Home Credit - Credit Risk Model Stability

Create a model measured against feature stability over time

Dataset Description

In this competition, you will be predicting default of clients based on internal and external information that are available for each client. Scoring is performed using custom metric that not only evaluates the AUC of predictions but also considers the stability of predictions model across the data range of the test set. To better understand this metric, please refer to the Evaluation tab.

Table Description

This dataset contains a large number of tables as a result of utilizing diverse data sources and the varying levels of data aggregation used while preparing the dataset. Note: All files listed below are found in both `.csv` and `.parquet` formats.

Files
138 files

Size
26.77 GB

Type
csv, parquet

License
Subject to Competition Rules

(https://www.kaggle.com/competitions/home-credit-credit-risk-model-stability/data)

features transformation

Базовые подходы:
- model
- f(x)

Примеры из соревнований:
- sklearn scalers (https://www.kaggle.com/code/datafan07/optiver-volatility-predictions-using-tabnet)
- denoising autoencoder (https://www.kaggle.com/competitions/porto-seguro-safe-driver-prediction/discussion/44629)

features treatment

Базовые подходы:
- const
- stats
- domain knowledge
- misspell correction

Примеры из соревнований:
- data description
(https://www.kaggle.com/code/serigne/stacked-regressions-top-4-on-leaderboard)

features selection

Базовые подходы:
- стабилизация решения
- удаление неинформативных признаков

Примеры из соревнований:
- adversarial validation (https://www.kaggle.com/code/carlmcbrideellis/what-is-adversarial-validation)
- null importance (https://www.kaggle.com/code/ogrellier/feature-selection-with-null-importances)

model trainer

Основные понятия:
- information gain & gini index
- bagging & boosting
- overfitting

Особенность алгоритма:
- outliers processing
- feat distances

Примеры из соревнований:
- custom metrics (https://www.kaggle.com/competitions/m5-forecasting-accuracy/discussion/133834)
- custom loss function (https://www.kaggle.com/code/jpison/custom-lgbm-obj-weighted-logloss-function)

Сборка стартера

Организация кодовой базы:
- быстрая проверка гипотез
- переиспользование наработок
- возможность для входа нового человека

Примеры из соревнований:
- script / notebook (https://www.kaggle.com/code/ogrellier/lighgbm-with-selected-features, https://www.kaggle.com/code/sggpls/santander-pipeline-kernel-xgb-fe-lb1-38)
- kaggle dataset (https://www.kaggle.com/datasets/yiheng/uw3dmonaitrainingpipeline)
- github project (https://www.kaggle.com/code/lopuhin/imet-2019-submission/script)

Сборка стартера

Доступные ресурсы:
- Disk
- RAM
- CPU
- GPU

Примеры из соревнований:
- data types (https://www.kaggle.com/code/gemartin/load-data-reduce-memory-usage)
- dataset preparation (CPU) (https://www.kaggle.com/code/tommy1028/lightgbm-starter-with-feature-engineering-idea)
- dataset preparation (GPU) (https://www.kaggle.com/code/cdeotte/candidate-rerank-model-lb-0-575/notebook)

Сборка стартера

Библиотеки python:
- pandas
- polars
- numba
- joblib
- cudf
- sklearn
- lightgbm
- catboost
- xgboost
- optuna
- bayes_opt
- hyperopt
- matplotlib
- seaborn