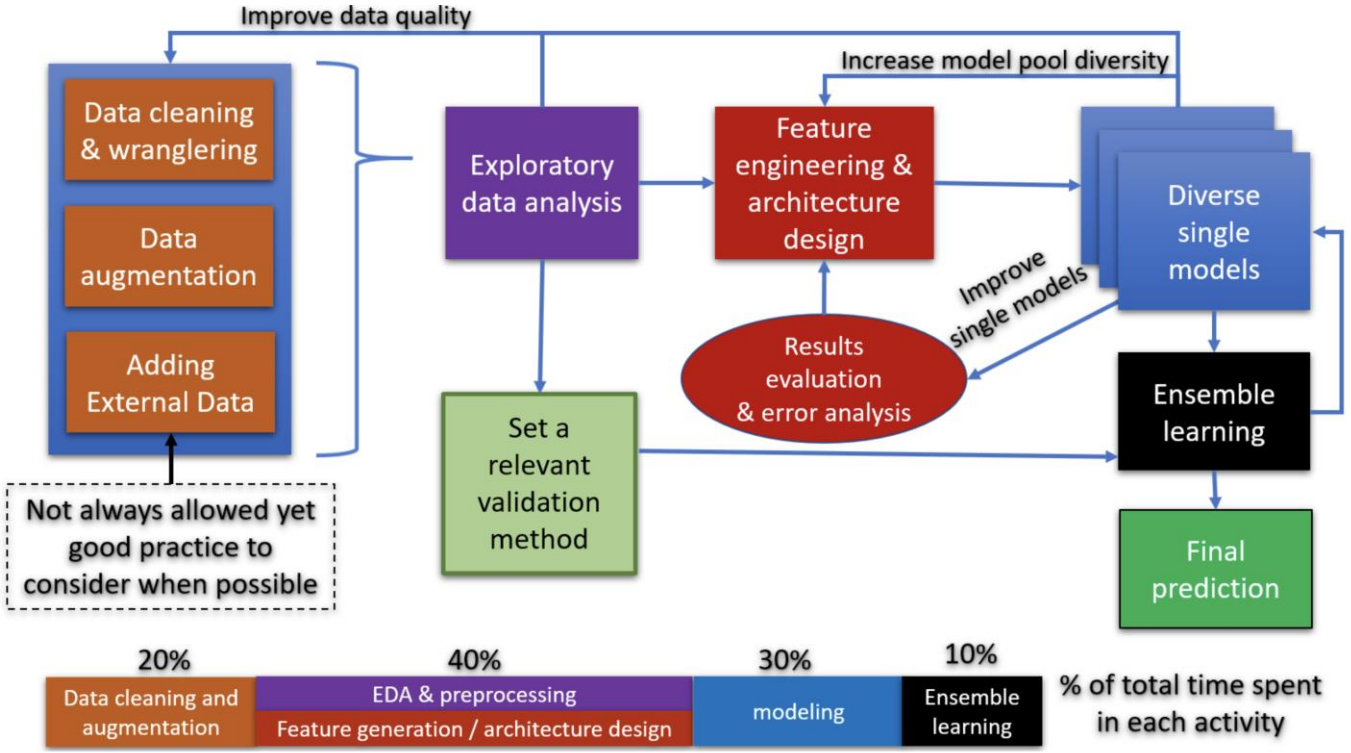
LightAutoML

Антон Вахрушев
Лаборатория AI Сбера

31.05.2023

Типичный pipeline решения ML задач



Довольно сложно… А есть ли способ решить задачу машинного обучения проще?

Классификация AutoML

6 уровней AutoML от Bojan Tunguz*:

1. No automation. You code your own ML algorithms. From scratch. In C++.
2. Use of high-level algorithm APIs. Sklearn, Keras, Pandas, H2O, XGBoost, etc.
3. Automatic hyperparameter tuning and ensembling. Basic model selection.
4. Automatic (technical) feature engineering and feature selection, technical data augmentation, GUI.

Мы здесь (если хватает ресурсов, обычно нет)

5. Automatic domain and problem specific feature engineering, data augmentation, and data integration.
6. Full ML Automation. Ability to come up with super-human strategies for solving hard ML problems without any input or guidance. Fully conversational interaction with the human user.

Из статьи AutoSklearn…

- No single machine learning method performs best on all datasets
- Some machine learning methods (e.g., non-linear SVMs) crucially rely on hyperparameter optimization

Сюда же относятся:

- Neural Architecture Search (NAS)
- Простой Meta Learning
- Стратегии обучения + управление бюджетом

Перспективные направления:

- Продвинутое Meta Learning (больше мета датасетов)
- Domain Specific Language **
- Базы знаний (Графы?)

https://medium.com/@tunguz/six-levels-of-auto-ml-a277aa1f0f38

** some fun examples (Kaggle ARC)

Что такое AutoML?

AutoML в узком смысле – инструмент для автоматического решения задачи машинного обучения

AutoML в широком смысле – технология, помогающая автоматизировать весь процесс моделирования и другие этапы, включая:

- подготовку данных/feature engineering
- отбор признаков
- построение модели, оптимизация параметров
- валидацию/построение отчетов
- внедрение и инференс моделей в проме/мониторинг

Open source решения:

- **LightAutoML (Sber AI Lab)**
- H2O AutoML (H2O.ai)
- AutoGluon (Amazon)
- TPOT
- AutoSklearn

…

Проприетарные решения:

- H2O Driveless AI
- Google Cloud AutoML
- IBM AutoAI
- Microsoft Azure AutoML

…

Мотивация разработки собственного AutoML

SBER AI Lab

Требуется AutoML, который поддерживает разные модальности – числа, категории, даты, тексты, изображения и т.п.

Часто датасет бывает не одной таблицей, а набором связанных таблиц – нужно уметь собрать плоский датасет

Необходимость построения интерпретируемых моделей

Решение ML задачи – не только модель, но и отчет о разработке, интерпретация, интеграция со средами инференса (ML Space в SberCloud), мониторинг и т.п.

Разные типы задач – классификация (бинарная/мультикласс), регрессия, uplift моделирование, multilabel и т.д.

Быстрое решение большого количества ML подзадач единой бизнес постановки – brute-force подход слишком медленный (ориентир на «модель за 10 минут с качеством среднего DS-а на midrange железе»)

Максимальная кастомизация выстраиваемых pipeline-ов, в том числе с написанием своих модулей для SOTA алгоритмов, при едином внешнем интерфейсе

Высокая технологичность и эффективность решения – запуск решения на GPU и Spark, использование данных из БД

LightAutoML (LAMA)

SBER AI Lab

Кросс-платформенный модульный фреймворк включающий в себя набор пресетов пайплайнов для решения **end-to-end** типовых задач, а также возможность разработки кастомных пресетов на разном уровне абстракции

Целевая аудитория – разработчики и DS

На сегодняшний день реализованы

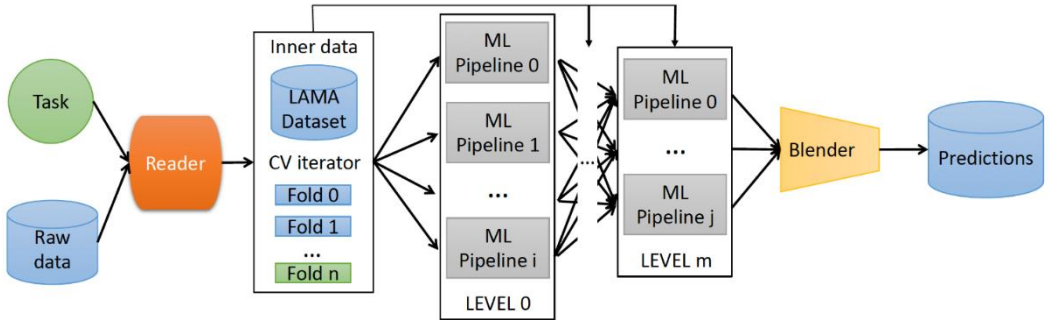
- ✓ BlackBox preset
- ✓ WhiteBox preset
- ✓ NLP preset

GitHub

784 + 484

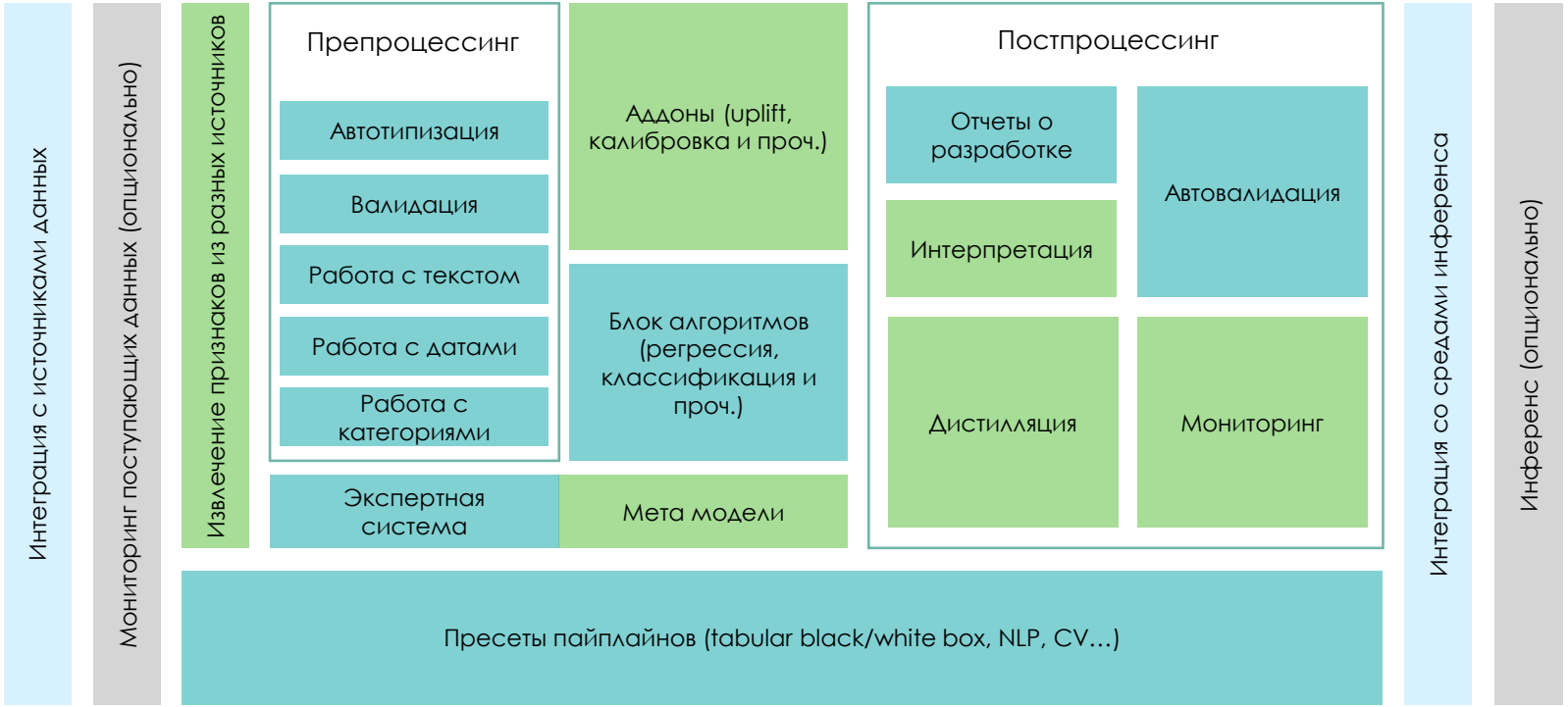
108 тыс.

Скачиваний LightAutoML



https://github.com/sb-ai-lab/LightAutoML

Верхнеуровневое описание LightAutoML (LAMA)



Доступно сейчас

Частично доступно / в разработке

Работы по индустриализации (SberCloud)

Разработки других групп

Пример использования LightAutoML

SBER AI Lab

```
import pandas as pd
from sklearn.metrics import f1_score

from lightautoml.automl.presets.tabular_presets import TabularAutoML
from lightautoml.tasks import Task

df_train = pd.read_csv('../input/titanic/train.csv')
df_test = pd.read_csv('../input/titanic/test.csv')

automl = TabularAutoML(
    task = Task(
        name = 'binary',
        metric = lambda y_true, y_pred: f1_score(y_true, (y_pred > 0.5)*1))
)
oof_pred = automl.fit_predict(
    df_train,
    roles = {'target': 'Survived', 'drop': ['PassengerId']}
)
test_pred = automl.predict(df_test)

pd.DataFrame({
    'PassengerId':df_test.PassengerId,
    'Survived': (test_pred.data[:, 0] > 0.5)*1
}).to_csv('submit.csv', index = False)
```

Курс по LightAutoML от авторов фреймворка

SBER AI Lab

Курс состоит из трех частей с записанными вебинарами, квизами и практическими заданиями:

Часть 1 - general overview, почему сделали свой AutoML, польза для бизнеса, практика применения на конкретных задачах

Часть 2 – нестандартные применения LightAutoML: NLP/CV, AutoUplift, Whitebox модели и генерация отчетов

Часть 3 - что у LAMA под капотом, как писать свои модули и как собирать свой ML pipeline из блоков

https://arxiv.org/pdf/2109.01528.pdf

Сравнение с другими open-source AutoML решениями

SBER AI Lab

Бенчмарки:

- ODS AutoML benchmark – топ-1 по качеству
- OpenML benchmark – вошли в топ-3 по качеству

	Tpot	MLJar	AutoGluon	H2O	LightAutoML
All datasets	4.4256%	13.4466%	13.8290%	14.0555%	14.2106%

Основной вывод – серебряной пули не существует, но к ней можно приблизиться и облегчить работу DS…

https://ods.ai/competitions/automl-benchmark

	Tpot	MLJar	AutoGluon	H2O	LightAutoML
credit-g	4.5263%	0.3384%	5.2524%	5.8536%	7.1453%
segment	3.9669%	4.0503%	4.0662%	4.0415%	4.0884%
kc1	1.9731%	4.6170%	4.3764%	3.4514%	4.3977%
adult	8.7560%	10.1771%	10.2830%	10.0099%	10.7383%
APSFailure	0.7326%	2.9269%	2.8785%	2.9182%	3.1428%
jungle_chess_2p	21.1461%	21.6394%	23.8115%	21.0370%	21.5449%
jannis	7.8074%	10.1464%	10.0499%	9.8560%	10.6147%
Public Result	6.9869%	7.6994%	8.6740%	8.1668%	8.8103%
	Tpot	MLJar	AutoGluon	H2O	LightAutoML
Australian	-0.1075%	0.6740%	1.6828%	1.7655%	2.5489%
blood-transfusion	1.1255%	3.7273%	4.5049%	2.4485%	2.5994%
jasmine	4.9719%	4.7390%	5.7296%	6.2263%	5.4881%
kr-vs-kp	1.5306%	1.8719%	1.8438%	1.8834%	1.8276%
phoneme	15.3687%	17.6236%	18.0752%	17.7147%	17.6690%
christine	8.2872%	44.3333%	48.5931%	47.5563%	49.2230%
guillermo	24.1032%	33.0760%	32.8832%	32.7489%	33.2271%
riccardo	2.6704%	8.6528%	8.6433%	8.6332%	8.6051%
Amazon_employ	31.9851%	56.9123%	61.4628%	61.5476%	67.1970%
nomao	-7.5334%	1.4202%	1.4435%	1.4000%	1.6287%
bank-marketing	-5.1903%	7.2434%	7.5410%	7.2266%	6.8950%
KDDCup09_app	3.0263%	16.9375%	19.3707%	18.2408%	19.6266%
higgs	6.2353%	27.8541%	15.9513%	33.6727%	15.3974%
MiniBooNE	-9.1812%	3.6579%	3.7463%	3.5905%	3.6961%
albert	5.3842%	8.3612%	4.6184%	6.2567%	10.0091%
airlines	1.8004%	19.5060%	19.2035%	19.4160%	16.9867%
connect-4	-1.8769%	47.2913%	45.6030%	48.7328%	47.7235%
vehicle	-16.7940%	-0.9291%	3.5984%	4.7849%	2.7386%
dilbert	0.3463%	9.5782%	9.7509%	9.1540%	9.7808%
fabert	-4.0466%	4.5736%	6.0223%	5.1054%	5.5430%
covertype	-2.4152%	9.5197%	9.6756%	7.7375%	9.5059%
cnae-9	31.3196%	35.6319%	36.0236%	29.0659%	35.6767%
mfeat-factors	0.1297%	0.6726%	0.8200%	0.8429%	0.7891%
robert	21.2027%	37.6371%	40.0294%	44.3273%	43.8943%
volkert	-3.9274%	9.3570%	10.0814%	7.5759%	9.2988%
Fashion-MNIST	-1.6510%	3.2091%	3.3119%	2.5074%	3.3279%
sylvine	-2.1660%	3.0265%	3.0413%	2.9512%	3.1579%
numerai28,6	1.3915%	0.5786%	0.0471%	1.6630%	1.6390%
Private result	3.7853%	14.8835%	15.1178%	15.5277%	15.5607%

Преимущества LightAutoML

SBER AI Lab

Преимущества внедрения LightAutoML:

- Сокращение времени разработки модели около 10 раз. Сокращение time-to-market до 70% (при условии интеграции со средой инференса)
- Способ доставки state-of-art решений индустриальным разработчикам
- Повышение качества разрабатываемых моделей. LightAutoML работает на уровне топ 10% «белковых» data scientists
- Работа с длинным правым хвостом – разработка множества моделей, которые приносят мало прибыли сами по себе
- Автовалидация, бенчмаркинг, быстрая проверка гипотез, нестандартные подходы
- Устранение гандикапа – модели больше не нужно разрабатывать «на века»

Устранение гандикапа

SBER AI Lab

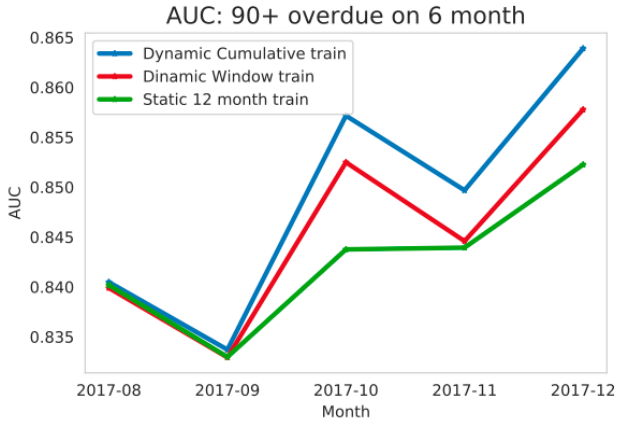
→ Текущая парадигма построения моделей

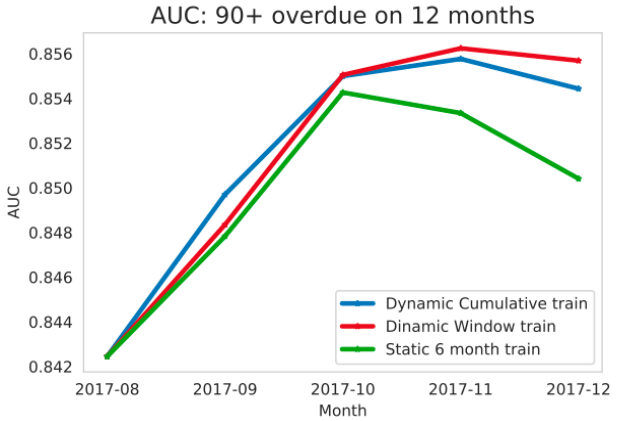
Эксперимент	2017-2018 год	Январь 2019	Февраль 2019	Март 2019	Апрель 2019	Май 2019	Июнь 2019
1	Train	Test					
2	Train		Test				
3	Train			Test			
4	Train				Test		
5	Train					Test	
6	Train						Test

→ Парадигма «устранение гандикапа» (стала реализуема благодаря LightAutoML):

Эксперимент	2017-2018 год	Январь 2019	Февраль 2019	Март 2019	Апрель 2019	Май 2019	Июнь 2019
1	Train	Test					
2	Train		Test				
3	Train			Test			
4	Train				Test		
5	Train					Test	
6	Train						Test

Борьба с деградацией моделей во времени путем регулярного перестроения





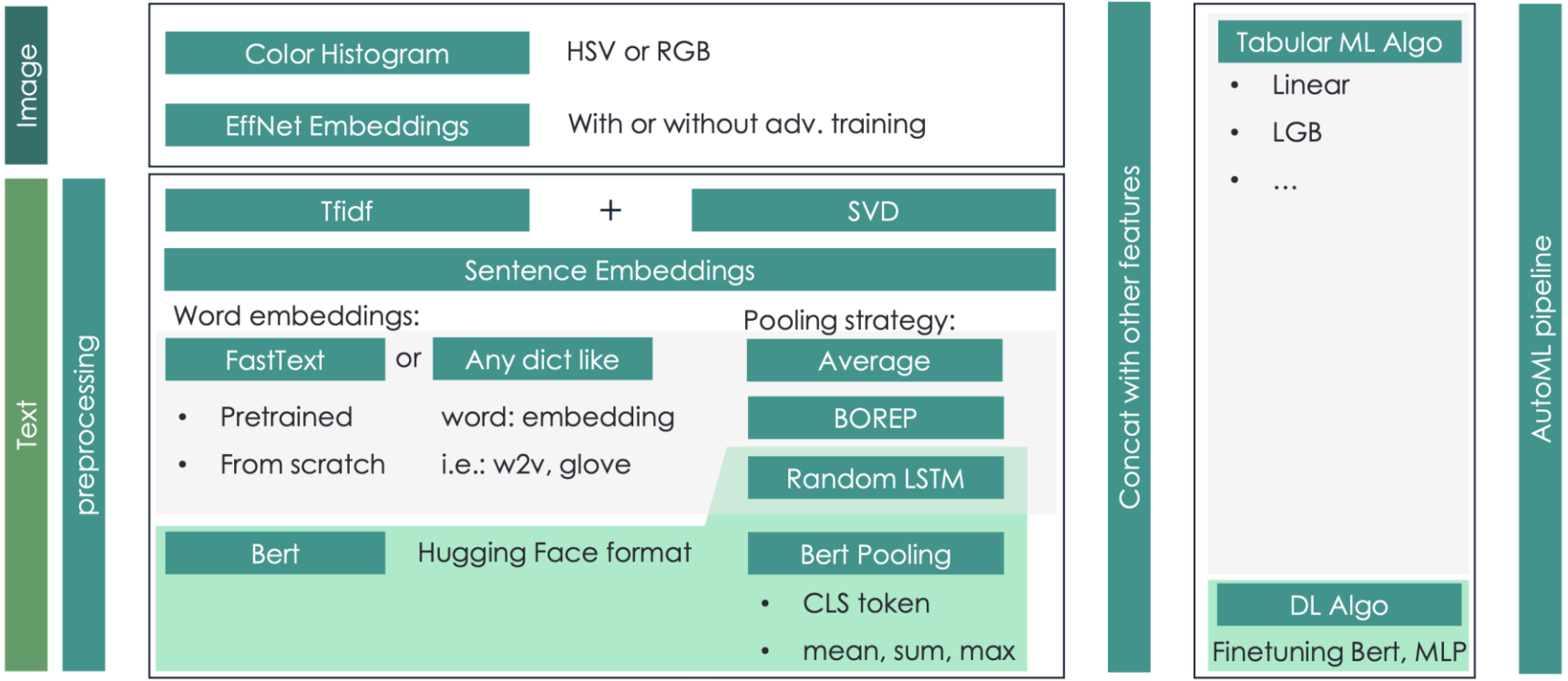
NLP & CV пресеты в LightAutoML

SBER AI Lab

Извлечение числовых признаков

Специализированные модели

NLP & CV пресеты в LightAutoML



Поддерживаемые языки: RU, EN, Multilanguage.

Интерпретация: локальная & глобальная

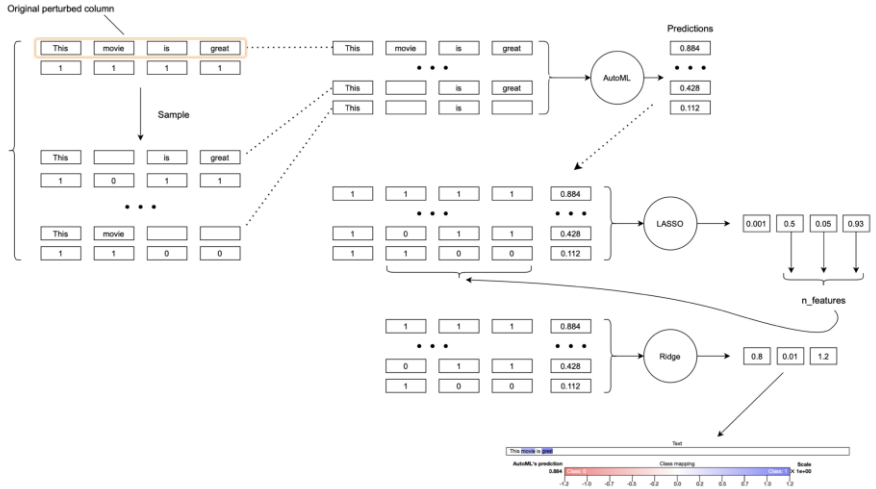
SBER AI Lab

Методы интерпретации NLP моделей в LightAutoML:

- LIME
- L2X

Посредством LIME можно узнать, в сторону какого класса и насколько сильно «тянет» конкретный токен, а L2X позволяет выделить набор токенов, наиболее важных для принятия моделью решения

LIME:



L2X – основные идеи:

- Выделить наиболее информативный набор токенов (НТ) = оптимизировать совместную информацию между НТ и таргетом
- Используем релаксацию через Gumbel-Softmax trick (аналог re-parametrization trick для категориальных переменных, которые можно генерировать из непрерывных)
- **Дополнительное улучшение: добавляем в функционал штраф за разреженность выделенного набора токенов (т.е. поощряем выделение полноценных длинных фраз)**

Pours a orange and straw yellow. The head is nice and bubbly but fades very quickly with a little lacing. Smells, a little yeast in there too. There is some fruit in there too, but you have to take a good whiff to get it. The taste is of wheat, a bit of malt, and a little fruit flavour in there too. Almost feels like drinking Champagne, medium mouthful otherwise. Easy to drink but not something I'd be trying every night.

Рис.: Appearance: 3.5 Aroma: 4.0 Palate: 4.5 Taste: 4.0 **Overall: 4.0**

```
In [33]: expl_valid[66].visualize_in_notebook()
```

Text

<START> lot exploder lost about 3 of this beer down the drain as foam whats left is a cloudy medium brown color with floaties plenty of head obviously which dissipates quickly aroma is tons of malt and dark fruit the flavor is again very fruity with bready malt and caramel notes a bit of roast malt no hint of spices anywhere full bodied with plenty of silky crispness <PAD>

```
In [34]: expl_valid[55].visualize_in_notebook()
```

Text

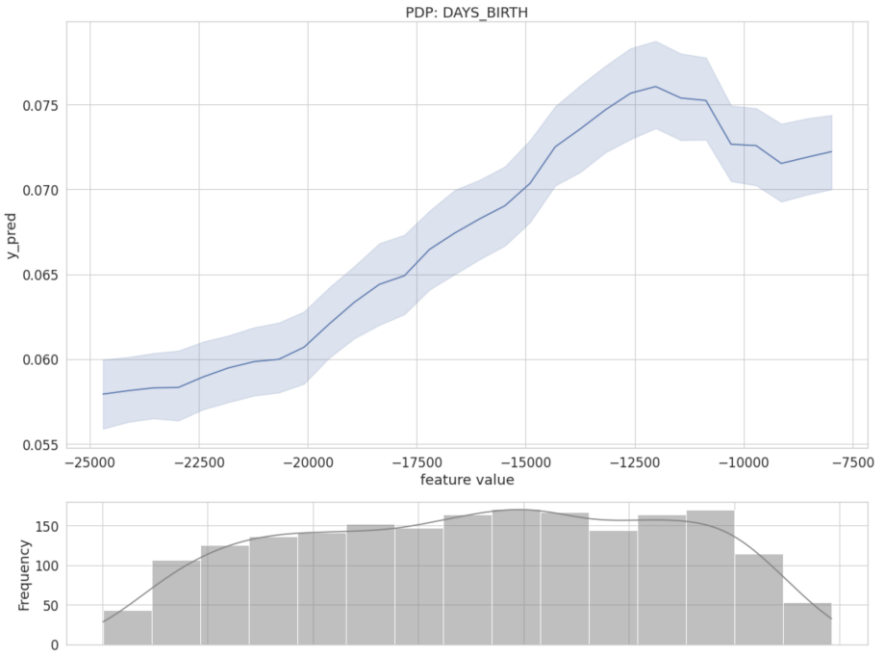
<START> whoa is right with this one this is a big brew in my opinion hence its name pours a thick creamy head and has a dark brown color with hints of amber the taste ha thick hops in here think of biting into a big juicy fruit terrapin comes out strong with this seasonal taste of alcohol is well hidden but will creap up on you in a hurry i have found this most of the year for some reason i guess they distributed alot of it in the atlanta area <PAD>

https://github.com/sb-ai-lab/LightAutoML/blob/master/examples/tutorials/Tutorial_4_NLP_Interpretation.ipynb

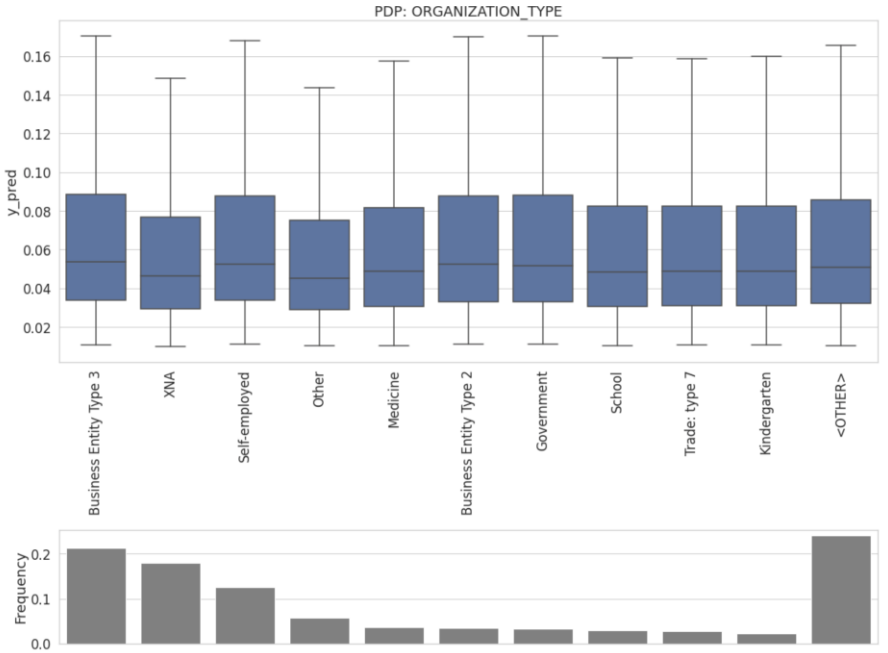
Интерпретация: локальная & глобальная

Глобальная интерпретация - реализована возможность построения графиков ICE и PDP на основе обученной модели

```
In [9]: automl.plot_pdp(test_data, feature_name='DAYS_BIRTH')
```

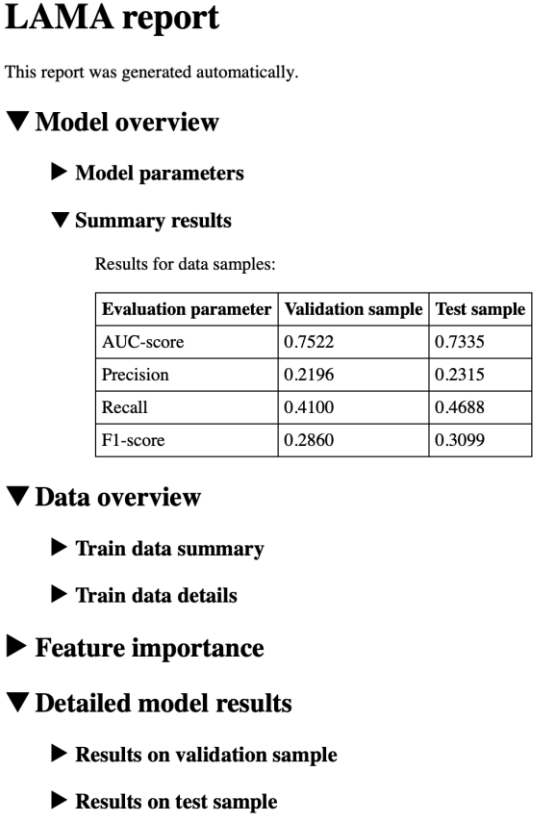



```
In [12]: automl.plot_pdp(test_data, feature_name='ORGANIZATION_TYPE')
```



Отчеты и мониторинг в LightAutoML

- Есть **встроенный декоратор ReportDeco** построения интерактивного отчета по модели – интерфейс идентичен обычному запуску
- Реализован **прототип мониторинга моделей** - на обучении запоминаем распределение признаков, первичных предсказаний и доли NULL, а при дальнейших предсказаниях:
  - Объединяем данные в эпохи достаточного размера (неделя, месяц, и т.д.)
  - Для каждой эпохи считаем PSI (Population Stability Index):
    - для каждого признака
    - для распределения предсказаний
  - Сравниваем долю NULL с тренировочной выборкой



LAMA report

This report was generated automatically.

▼ **Model overview**

▶ **Model parameters**

▼ **Summary results**

Results for data samples:

Evaluation parameter	Validation sample	Test sample
AUC-score	0.7522	0.7335
Precision	0.2196	0.2315
Recall	0.4100	0.4688
F1-score	0.2860	0.3099

▼ **Data overview**

▶ **Train data summary**

▶ **Train data details**

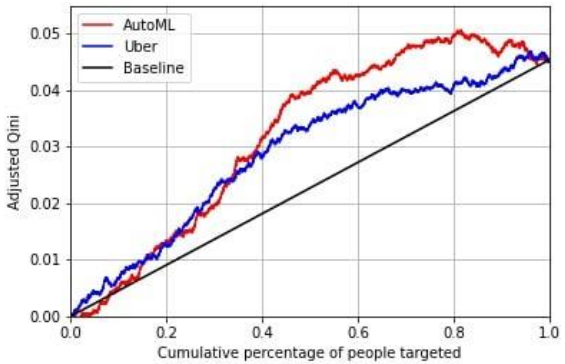
▶ **Feature importance**

▼ **Detailed model results**

▶ **Results on validation sample**

▶ **Results on test sample**

AutoUplift в LightAutoML: основные принципы



Results on Hillstrom dataset

Blackbox base learners

meta learner	LightAutoML	xgb (uber)
TLearner	**0,222**	0,203
Xlearner	**0,223**	0,198

Linear base learners

meta learner	LightAutoML	sklearn (uber)
TLearner	**0,228**	0,223
Xlearner	**0,223**	0,222

TLearner

1) Строим независимые модели на целевой и контрольной группах

$$model^T = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pk} \end{pmatrix}, \begin{matrix} y_1 \\ \cdots \\ y_p \end{matrix}\Big), \quad model^C = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{q1} & \cdots & x_{qk} \end{pmatrix}, \begin{matrix} y_1 \\ \cdots \\ y_q \end{matrix}\Big)$$

X_{train_treat} Y_{train_treat} $X_{train\_control}$ $Y_{train\_control}$

2) Вычисляем величину uplift как разницу 2х моделей – вероятности при коммуникации и без

$$model^T_{predict\ proba}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mk} \end{pmatrix} - model^C_{predict\ proba}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mk} \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix}$$

X_{test} X_{test} $uplift$

XLearner

- LightAutoML предоставляет обертки (meta learners) для удобного построения сильных базовых моделей, используя TabularAutoML пресет и оценке на их основе величины uplift.
- Помимо этого доступна функция автоматического поиска лучшей комбинации базовых AutoML + metalearners.

1) Строим независимые модели на целевой и контрольной группах

$$model^T = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pk} \end{pmatrix}, \begin{matrix} y_1 \\ \cdots \\ y_p \end{matrix}\Big), \quad model^C = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{q1} & \cdots & x_{qk} \end{pmatrix}, \begin{matrix} y_1 \\ \cdots \\ y_q \end{matrix}\Big)$$

X_{train_treat} Y_{train_treat} $X_{train\_control}$ $Y_{train\_control}$

2) Предсказываем прокси uplift на клиента в целевой и контрольной группах

$$\hat{Y}^C = model^C_{predict}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pk} \end{pmatrix}; \quad \hat{Y}^T = model^T_{predict\ proba}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{q1} & \cdots & x_{qk} \end{pmatrix}$$

X_{train_treat} $X_{train\_control}$

$$\bar{D}^T = Y_{train\_treat} - \hat{Y}^C; \quad \bar{D}^C = \hat{Y}^T - Y_{train\_control}$$

3) Строим независимые модели на прокси uplift в целевой и контрольной группах

$$model^T_{new} = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pk} \end{pmatrix}, \begin{matrix} \bar{d}_1^T \\ \cdots \\ \bar{d}_p^T \end{matrix}\Big), \quad model^C_{new} = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{q1} & \cdots & x_{qk} \end{pmatrix}, \begin{matrix} \bar{d}_1^C \\ \cdots \\ \bar{d}_q^C \end{matrix}\Big)$$

X_{train_treat} $\bar{D}^T$ $X_{train\_control}$ $\bar{D}^C$

4) Оценка uplift – взвешенное среднее прогнозов на прокси uplift 2х моделей

$$g \cdot model^C_{new\ predict}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mk} \end{pmatrix} + (1-g) \cdot model^T_{new\ predict}\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mk} \end{pmatrix} = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix}$$

X_{test} X_{test} $uplift$

https://github.com/sb-ai-lab/LightAutoML/blob/master/examples/tutorials/Tutorial_5_uplift.ipynb

AutoUplift в LightAutoML: отчеты

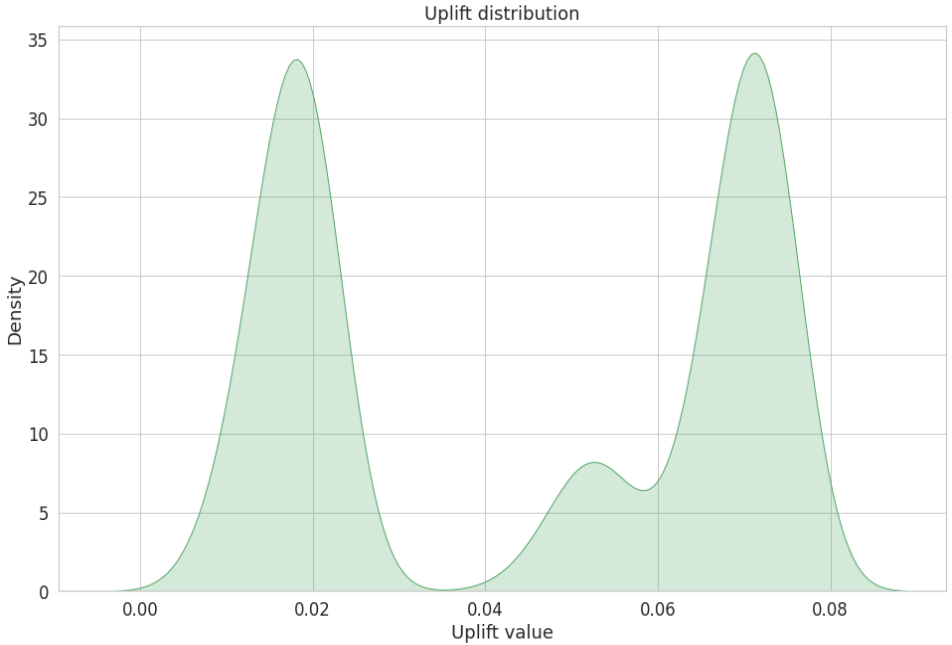
SBER AI Lab

▼ **Test sample summary**

Parameter	Value
Number of records	12808
Share of treatment	0.500937
Mean target	0.128826
Mean target on treatment	0.15134
Mean target on control	0.106227

▼ **Uplift distribution by bins**

Bin number	Amount of objects	Min uplift	Mean uplift	Max uplift	Uplift fact
0	1281	0.006775	0.012237	0.014686	0.011384
1	1281	0.014687	0.016168	0.017456	0.013744
2	1281	0.017457	0.018420	0.019439	0.021174
3	1281	0.019440	0.020591	0.021860	-0.005444
4	1280	0.021860	0.036508	0.052474	0.021594
5	1281	0.052474	0.059125	0.066409	0.059540
6	1281	0.066419	0.067920	0.069274	0.085208
7	1281	0.069275	0.070493	0.071554	0.085346
8	1281	0.071558	0.072540	0.073615	0.073221
9	1280	0.073622	0.075052	0.077110	0.085762



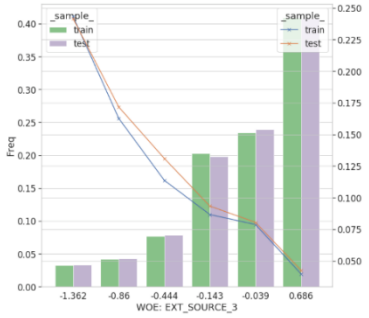
https://github.com/sb-ai-lab/LightAutoML/blob/master/examples/tutorials/Tutorial_5_uplift.ipynb

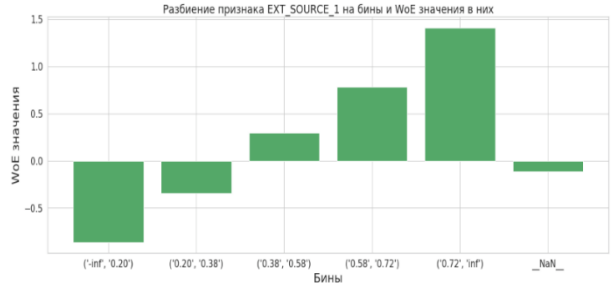
WhiteBox AutoML: интерпретируемые модели

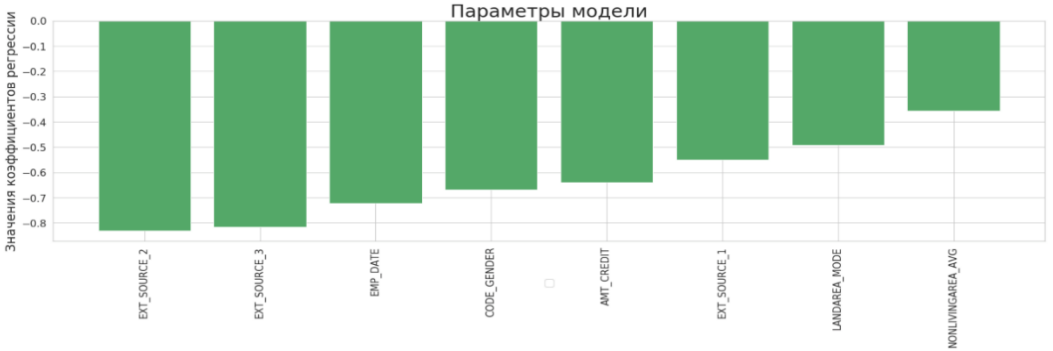
SBER AI Lab

Разработка моделей в формате классической скор карты с минимальными потерями в качестве относительно BlackBox

Variable	Value	WOE	COEF	POINTS
Intercept	None	None	-2.48	-2.48
EXT_SOURCE_2	EXT_SOURCE_2 <= 0.05	-1.58	-0.83	1.31
EXT_SOURCE_2	0.05 < EXT_SOURCE_2 <= 0.16	-1.04	-0.83	0.86
EXT_SOURCE_2	0.16 < EXT_SOURCE_2 <= 0.45	-0.41	-0.83	0.34
EXT_SOURCE_2	0.45 < EXT_SOURCE_2 <= 0.67	0.24	-0.83	-0.2
EXT_SOURCE_2	EXT_SOURCE_2 > 0.67	1.05	-0.83	-0.87
EXT_SOURCE_2	__NaN_0__	0	-0.83	0.0
EXT_SOURCE_3	EXT_SOURCE_3 <= 0.14	-1.36	-0.82	1.11
EXT_SOURCE_3	0.14 < EXT_SOURCE_3 <= 0.22	-0.86	-0.82	0.7
EXT_SOURCE_3	0.22 < EXT_SOURCE_3 <= 0.32	-0.44	-0.82	0.36
EXT_SOURCE_3	0.32 < EXT_SOURCE_3 <= 0.53	-0.04	-0.82	0.03
EXT_SOURCE_3	EXT_SOURCE_3 > 0.53	0.69	-0.82	-0.56
EXT_SOURCE_3	__NaN__	-0.14	-0.82	0.12
EXT_SOURCE_1	EXT_SOURCE_1 <= 0.2	-0.87	-0.55	0.48
EXT_SOURCE_1	0.2 < EXT_SOURCE_1 <= 0.38	-0.34	-0.55	0.19
EXT_SOURCE_1	0.38 < EXT_SOURCE_1 <= 0.58	0.3	-0.55	-0.16
EXT_SOURCE_1	0.58 < EXT_SOURCE_1 <= 0.72	0.78	-0.55	-0.43
EXT_SOURCE_1	EXT_SOURCE_1 > 0.72	1.41	-0.55	-0.78
EXT_SOURCE_1	__NaN__	-0.11	-0.55	0.06
AMT_CREDIT	AMT_CREDIT <= 273330.0	0.01	-0.64	-0.01
AMT_CREDIT	273330.0 < AMT_CREDIT <= 480618.0	-0.34	-0.64	0.22
AMT_CREDIT	480618.0 < AMT_CREDIT <= 798322.5	-0.09	-0.64	0.06
AMT_CREDIT	AMT_CREDIT > 798322.5	0.48	-0.64	-0.31
AMT_CREDIT	__NaN_0__	0	-0.64	0.0







https://github.com/sb-ai-lab/AutoMLWhitebox

WhiteBox AutoML: интерпретируемые модели

SBER AI Lab

WhiteBox модель может быть автоматически сохранена в виде SQL запроса для инференса на стороне БД

```
SELECT
  1 / (1 + EXP(-(
    -2.43
    -0.831*WOE_TAB.EXT_SOURCE_3
    -0.778*WOE_TAB.EXT_SOURCE_2
    -0.599*WOE_TAB.DAYS_EMPLOYED
    -0.533*WOE_TAB.EXT_SOURCE_1
    -0.6*WOE_TAB.AMT_GOODS_PRICE
    -0.574*WOE_TAB.CODE_GENDER
    -0.614*WOE_TAB.NAME_EDUCATION_TYPE
    -0.444*WOE_TAB.YEARS_BEGINEXPLUATATION_MEDI
    -0.66*WOE_TAB.OWN_CAR_AGE
    -0.675*WOE_TAB.DEF_30_CNT_SOCIAL_CIRCLE
    -1.064*WOE_TAB.NAME_CONTRACT_TYPE
  ))) as PROB,
  WOE_TAB.*
FROM
    (SELECT
    CASE
      WHEN (EXT_SOURCE_3 IS NULL OR EXT_SOURCE_3 = 'NaN') THEN -0.164
      WHEN EXT_SOURCE_3 <= 0.1478 THEN -1.226
      WHEN EXT_SOURCE_3 <= 0.26575 THEN -0.762
      WHEN EXT_SOURCE_3 <= 0.33365 THEN -0.425
      WHEN EXT_SOURCE_3 <= 0.50558 THEN -0.045
      ELSE 0.647
    END AS EXT_SOURCE_3,
    CASE
      WHEN (EXT SOURCE 2 IS NULL OR EXT SOURCE 2 = 'NaN') THEN 0
```

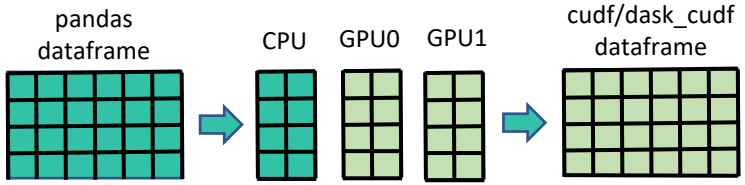
https://github.com/sb-ai-lab/AutoMLWhitebox

Новые пайплайны: GPU & Spark

SBER AI Lab

GPU pipeline – основные особенности:

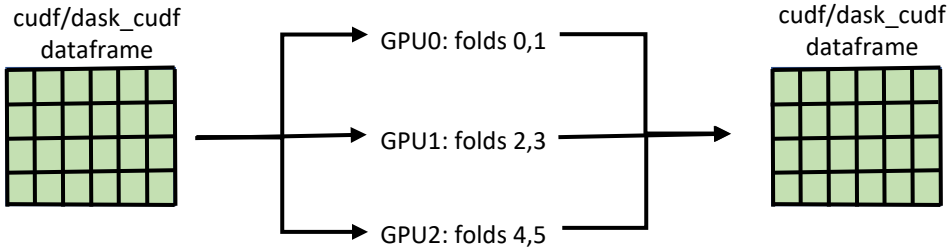
- Гибридный препроцессинг данных
- Параллелизация обучения моделей по фолдам/данным
- Поддержка Single/Multi GPU режимов



hybrid parallel processing of column batches

Результаты на датасете Fashion-MNIST (70k * 785):

- **Стандартный режим** – препроцессинг 283с, линейка 1716с, бустинг 1844с
- **1 GPU – ускорение x6.5** - препроцессинг **x1.3**, линейка **x9**, бустинг **x10**
- **2 GPU – ускорение x9.3** - препроцессинг **x1.7**, линейка **x14**, бустинг **x15**



Multi-GPU data processing during cross-validation.

Официальный репозиторий скоро появится на https://github.com/sb-ai-lab

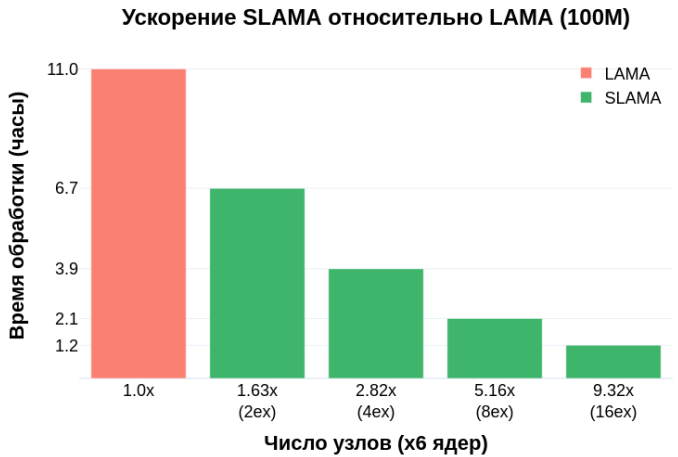
Dev версия: https://github.com/Rishat-skoltech/LightAutoML_GPU

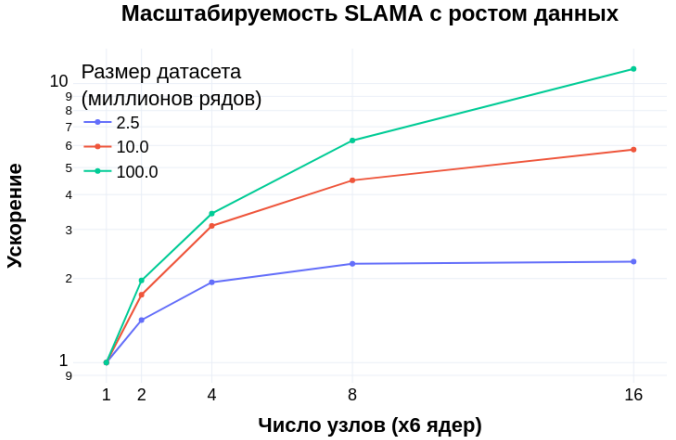
Новые пайплайны: GPU & Spark

SBER AI Lab

Spark pipeline (SLAMA) – основные особенности:

- ❖ Написана под Spark 3.2+ и частично на Scala
- ❖ Автоматическое разбиение эстиматоров признаков на слои с кэшированием / чекпоинтами только между слоями
- ❖ Линейное преобразование датасета без использования join-ов и с минимумом broadcast-ов
- ❖ Обработка категориальных фич (Label , Ordinal, Freq encoding) с помощью кастомизированного StringIndexer
- ❖ Линейный результирующий AutoML трансформер, поддерживающий сохранение и загрузку



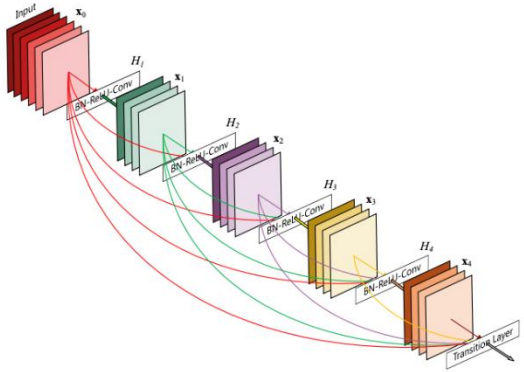


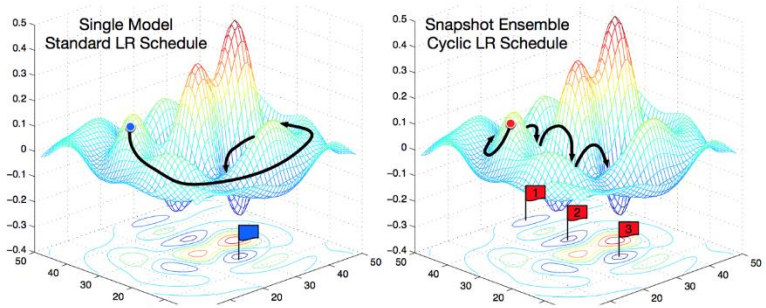
- ❖ SLAMA способна работать с датасетами в **1 миллиард строк и более**
- ❖ **Масштабируемость растет с ростом объема данных** – больше данных = больше ускорения, вплоть до линейной зависимости

https://github.com/sb-ai-lab/SLAMA

Dev версия LightAutoML – табличные нейронки

- Добавились новые NN модели, адаптированные для табличных данных, например, densenet, resnet.
- Можете сильно кастомизировать обучение модели и ее архитектуру (передать лосс и другие параметры)
- При обучение есть возможность использовать SWA для улучшения качества, а также клиппинг градиентов для регуляризации.
- Среди тасков, помимо стандартных задач, поддерживается мульти-регрессия и мультилэйбл.





https://www.kaggle.com/code/alexryzhkov/lightautoml-nn-imputer-tps-june-22

Заключение

SBER AI Lab

LightAutoML (LAMA) это:

- Open source инструмент для программистов, data scientist’ов и аналитиков (https://github.com/sb-ai-lab/LightAutoML)
- Автоматическая разработка типовой ML модели за 10 минут на ноутбуке с качеством лучше среднего DS
- Возможность строить модели в разных постановках, важных для конкретного бизнеса
- Не только AutoML решение, но и конструктор своих ML пайплайнов

СБЕР

Вопросы?

Вахрушев Антон, Лаборатория AI Сбера

AGVakhrushev@sberbank.ru,