

Курсовой проект от Мегафон

Бажанова Екатерина

GeekBrains, факультет искусственного интеллекта

20 января 2022 г.



Постановка задачи

Задача

Построить алгоритм, который для каждой пары пользователь-услуга определит вероятность подключения услуги.

	id	vas_id	buy_time
0	3130519	2.0	1548018000
1	2000860	4.0	1548018000

Метрика

`sklearn.metrics.f1_score(..., average='macro')`

Исходные данные

data train.csv

трейновый датасет с признаками: id; vas id; buy time; target.

features.csv.zip

датасет с признаками: id; <features list>

data test.csv

тестовый датасет с признаками: id; vas id; buy time.

признаки датасетов

id – идентификатор абонента;

vas id – подключаемая услуга;

buy time – время покупки (формат timestamp);

target – целевая переменная (1 – подключение услуги, 0 – не подключение услуги).

Этапы решения

0. Загрузка данных

Загрузка датасетов data train.csv, data test.csv, features.csv

1. Первичный анализ данных

Обзор целевой переменной и датасетов data train, data test и features

2. Построение модели классификации

Создание датасета для обучения модели, подбор моделей, выбор лучшей модели и настройка гиперпараметров

3. Прогнозирование на тестовом датасете

Предсказание target с помощью итоговой модели

4. Формирование рекомендаций подключения услуг

Прогнозирование вероятности подключения услуг

Подготовка данных

Merge датасетов train и features

```
%%time

X_nearest = pd.merge_asof(data_train.sort_values(by=['id']),
                           features.sort_values(by=['id']),
                           on='id',
                           by='buy_time',
                           direction='nearest')

X_nearest.head(2)
```

Wall time: 20min 19s

	id	vas_id	buy_time	target	year	month	day	0	1	2	...
0	2	2.0	2018-12-24	0.0	2018	12	24	-96.799971	229.530888	-110.740786	...
1	4	1.0	2018-08-06	0.0	2018	8	6	-52.309971	-225.139112	-66.250786	...

2 rows × 255 columns

Отбор признаков

Из 255 признаков отобраны 137 (удалены константные, категориальные признаки, а также признаки, не полезные для обучения)

Обоснование выбора модели

Обученные модели

CatBoostClassifier, XGBClassifier, LGBMClassifier и
DecisionTreeClassifier

	model	train_score_f1	test_score_f1	scores.mean() - scores.std()	scores.mean() + scores.std()
0	model_catb	0.713551	0.713236	0.717336	0.718902
1	model_xgb	0.734999	0.703974	0.679931	0.683341
2	model_lgbm	0.715044	0.714455	0.713252	0.714997
3	model_tree	1.000000	0.673154	0.671580	0.672864

Лучшая метрика на валидации

LGBMClassifier

Используемая модель

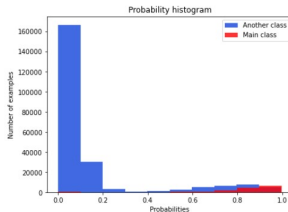
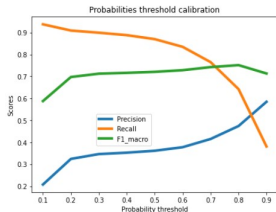
Параметры модели

```
model = lgbm.LGBMClassifier(random_state=21,  
                             class_weight='balanced',  
                             objective='binary',  
                             max_depth=10,  
                             n_estimators=1000,  
                             num_leaves=40,  
                             reg_lambda=0.5)
```

Из 137 признаков взяты топ-126

Порог вероятности предсказания

Оптимальный порог 0.8



Результат модели и прогноз на тесте

Результат модели на data train

f1 macro на train 0.8260610804103954

f1 macro на test 0.7522575067938254

Прогноз на тесте

Тестовый датасет обработан аналогично трейновому

```
result = pd.read_csv('answers_test.csv')  
result
```

форма тестового датасета с important_features_top: (71231, 126)

	vas_id	id	day	226	52	164	month	128	115	247
0	2.0	55	14	-0.767334	-32.171711	0.060492	1	0.354871	0.446143	-253.747724
1	4.0	64	21	10.149332	-64.171711	0.120492	1	-0.575129	-0.393857	-306.747724
2	2.0	151	14	-7.225668	-28.171711	0.250492	1	0.114871	-0.003857	-158.747724

3 rows x 126 columns

	id	vas_id	buy_time	target
0	55	2.0	2019-01-14	0.147746
1	64	4.0	2019-01-21	0.743683
2	151	2.0	2019-01-14	0.005453
3	274	4.0	2019-01-21	0.349173
4	274	2.0	2019-01-14	0.002798
...
71226	4362676	2.0	2019-01-21	0.027134
71227	4362677	2.0	2019-01-14	0.000691
71228	4362697	5.0	2019-01-07	0.001239
71229	4362712	5.0	2019-01-14	0.011264
71230	4362720	2.0	2019-01-07	0.000314

71231 rows x 4 columns

Прогноз сохранен в answers test.csv

Предложения для абонентов

Постановка
задачи

Исходные
данные

Этапы решения

Подготовка
данных

Обоснование
выбора модели

Используемая
модель

Результат
модели и
прогноз на тесте

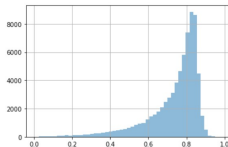
Предложения
для абонентов

Итоговые
материалы

Прогноз вероятности подключения услуг

Для каждого пользователя подсчитана вероятность подключения каждой услуги

id	1	2	3	4	5	6	7	8	9	max_prob	vas_id_max_prob
55	0.104246	0.147746	0.147746	0.107516	0.106805	0.832372	0.030274	0.028534	0.209745	0.832372	6
64	0.162583	0.161608	0.161608	0.743683	0.158476	0.852569	0.018626	0.014808	0.146008	0.852569	6
151	0.004107	0.005453	0.005453	0.623984	0.029704	0.745584	0.018696	0.010220	0.188749	0.745584	6
274	0.001273	0.001102	0.001102	0.340173	0.009649	0.486067	0.001713	0.001206	0.037330	0.486067	6
274	0.002306	0.002798	0.002798	0.574896	0.022755	0.742763	0.005649	0.005505	0.069046	0.742763	6
...
4362676	0.020714	0.027134	0.027134	0.514915	0.017387	0.734564	0.005075	0.003957	0.027335	0.734564	6
4362677	0.000573	0.000691	0.000691	0.318313	0.002409	0.587547	0.002588	0.002435	0.004886	0.587547	6
4362697	0.000288	0.000514	0.000514	0.158031	0.001239	0.407308	0.001190	0.001120	0.011403	0.407308	6
4362712	0.021990	0.024746	0.024746	0.467377	0.011264	0.742375	0.004634	0.004361	0.006433	0.742375	6
4362720	0.000298	0.000514	0.000514	0.158031	0.001239	0.407308	0.001190	0.001120	0.011403	0.407308	6



Формирование рекомендаций

Целесообразно рекомендовать абонентам услуги, вероятность которых ≥ 0.6

Итоговые материалы

Итоговые файлы

- 1) `course project.ipynb` – jupyter-ноутбук с кодом
- 2) `final model.pkl` – модель в формате pickle
- 3) `predict test.py` – файл с кодом (принимает файлы `data test.csv` и `features.csv` из корневой папки и записывает в эту же папку файл `answers test.csv`)
- 4) `answers test.csv` – файл с предсказаниями вероятностей для `data test.csv`
- 5) `answers test class.csv` – файл с предсказаниями классов для `data test.csv`
- 6) `vas id recommended.csv` – файл с предложениями услуг абонентам из `data test.csv`
- 7) `course presentation.pdf` – презентация этапов решения задачи

Ссылка на github

<https://github.com/EkaterinaBazhanova/Megafon>