Курсовой проект от Мегафон

Выполнил Захарченко С.В.

Цель проекта

Согласно заданию, компания обладает массивным нормализованным анонимизированным набором признаков абонентов.

На основании этих данных необходимо построить модель машинного обучения, которая сможет установить закономерности в данных и предсказать вероятность подключения услуги абонентом.

Данные об абонентах

id	vas_id	buy_time	target
540968	8.00	1537131600	0.00
1454121	4.00	1531688400	0.00
2458816	1.00	1534107600	0.00
3535012	5.00	1535922000	0.00
1693214	1.00	1535922000	0.00

Набор признаков

0	1	2	3	...	243	244	245	246	247	248	249	250	251	252
-31.56	327.36	-45.50	274.75	...	-845.37	-613.77	-21.00	-37.63	-28.75	4.17	7.31	-12.18	21.54	0.00
547.27	238.43	533.33	274.80	...	-972.37	-613.77	-26.00	-19.63	-278.75	-24.83	-0.69	-11.18	-0.46	0.00
-92.14	-95.47	-106.08	-139.60	...	-977.37	-613.77	-26.00	-37.63	-304.75	-25.83	-0.69	-12.18	-0.46	0.00
54.88	12.97	54.08	-9.12	...	-977.37	-613.77	-26.00	-18.63	-133.75	-14.83	-0.69	-1.18	-0.46	0.00
45.16	295.24	64.68	344.28	...	-965.37	-612.77	-23.00	-32.63	-127.75	-4.83	-0.69	-12.18	-0.46	0.00

Этап 1. Подготовка данных

В ходе подготовки признаков было установлено, что для некоторых абонентов "id" встречаются сразу несколько записей признаков. Соединение признаков также по времени покупки "buy_time" было невозможно, потому как признаки собирались либо ранее, либо позднее.
Поэтому был разработан алгоритм, который позволил использовать наиболее актуальные признаки абонентов на момент покупки.

Дубликаты, возникшие из-за нескольких наборов признаков для одного абонента

	unique_id	id	vas_id	buy_time_train	target	buy_time	0	1	2	3	...
8	2000856_1_1534712400	2000856	1.00	1534712400	0.00	1531688400	-21.37	108.78	178.62	770.86	...
9	2000856_1_1534712400	2000856	1.00	1534712400	0.00	1531083600	-92.56	48.60	50.24	609.86	...
58	3577738_1_1532293200	3577738	1.00	1532293200	0.00	1540760400	-96.80	45.21	-104.81	315.48	...
59	3577738_1_1532293200	3577738	1.00	1532293200	0.00	1537736400	-96.80	104.53	-8.22	672.22	...
63	203194_1_1532293200	203194	1.00	1532293200	0.00	1542574800	-96.80	-111.57	-110.74	-164.18	...
64	203194_1_1532293200	203194	1.00	1532293200	0.00	1540760400	-96.80	-170.89	-110.74	-223.50	...

Этап 2. Анализ признаков

Анализ данных показал следующие особенности:

- дисбаланс классов целевой переменной;
- наличие 5 константных признаков, которые не представляли ценности;
- наличие 20 бинарных признаков, которых были закодированы;
- наличие 10 категориальных признаков, с количеством категорий менее 10, которые так же были закодированы;
- наличие 219 вещественных признаков, которые были стандартизированы.

Дисбаланс классов целевой переменной

```
0.00    92.76%
1.00     7.24%
```

Этап 3. Построение моделей

Для решения поставленной задачи были рассмотрены наиболее мощные представители моделей машинного обучения - градиентные бустинги в следующих реализациях:

1. CatBoostClassifier
2. LGBMClassifier
3. XGBClassifier

Для удобства оперирования моделью и портативности были использованы пайплайны.

Импорт моделей

```
from catboost import CatBoostClassifier
from lightgbm import LGBMClassifier
from xgboost import XGBClassifier
```

Пайплайн

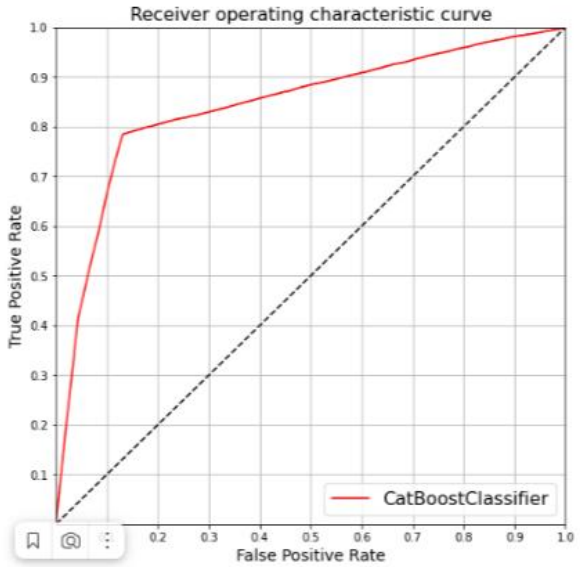
```
# Пайплайн для обработки категориальных признаков.
pipe_cat = Pipeline([
    ('select_cat_features', ColumnSelector(['vas_id'] + list_features_bool + list_features_cat)),
    ('encoding_cat_features', OneHotEncoder(drop='first', handle_unknown='ignore'))
])

# Пайплайн для обработки вещественных признаков.
pipe_num = Pipeline([
    ('select_num_features', ColumnSelector(list_features_num + list_features_others)),
    ('standard_num_features', StandardScaler())
])

# Объединение признаков после обработки.
fu_processing = FeatureUnion([
    ('pipe_cat', pipe_cat),
    ('pipe_num', pipe_num)
], n_jobs=-1)

# Пайплайн с моделью CatBoostClassifier.
pipe_model_cb = Pipeline([
    ('features', fu_processing),
    ('model', CatBoostClassifier(task_type='GPU',
                                 silent=True,
                                 auto_class_weights='Balanced',
                                 thread_count=-1,
                                 random_state=GLOBAL__RANDOM_STATE))
])
```

Этап 4. Метрики качества

Оценка разработанных моделей производилась при помощи метрик качества бинарной классификации. Все модели продемонстрировали сопоставимые результаты.

ROC AUC curve

CatBoostClassifier: AUC_ROC = 0.843

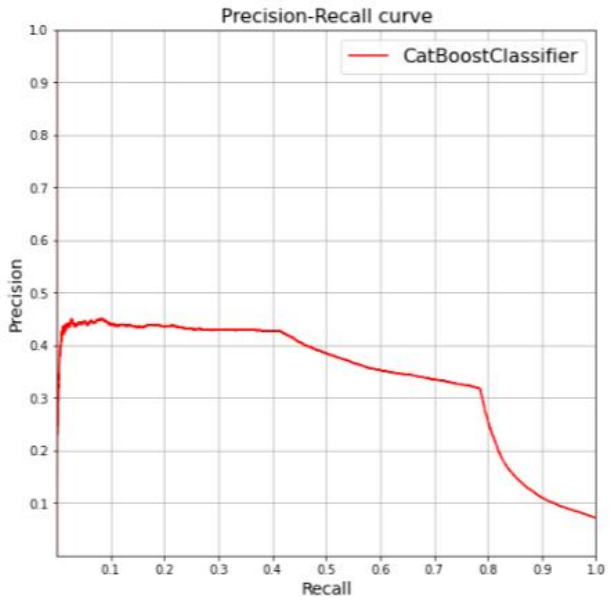


Classification report

	precision	recall	f1-score	support
0	0.98	0.87	0.92	192867
1	0.32	0.79	0.45	15047
accuracy			0.86	207914
macro avg	0.65	0.83	0.69	207914
weighted avg	0.93	0.86	0.89	207914

Precision-Recall curve

CatBoostClassifier: AUC_PR = 0.342



Этап 5. Выводы

Classification report

	precision	recall	f1-score	support
0	0.98	0.87	0.92	192867
1	0.32	0.79	0.45	15047
accuracy			0.86	207914
macro avg	0.65	0.83	0.69	207914
weighted avg	0.93	0.86	0.89	207914

Анализ метрик качества показал, что для класса "1" - подключение услуги, все модели продемонстрировали высокую метрику "Recall" = ~0.8. То есть, удалось установить практически всех потенциальных клиентов.

Однако метрика "Precision" = ~0.33 свидетельствовала о том, что только треть потенциальных клиентов были предсказаны правильно.

Для класса "0" - услуга не подключена, точность прогноза получилась высокой.

Таким образом, при наличии достаточных ресурсов на рекламную компанию можно зафиксировать порог метрики "Recall" в районе ~0.8 и получить около трети подключений услуг от всей целевой аудитории.

	id	vas_id	buy_time	target
0	3130519	2.00	1548018000	0.15
1	2000860	4.00	1548018000	0.82
2	1099444	2.00	1546808400	0.22
3	1343255	5.00	1547413200	0.19
4	1277040	2.00	1546808400	0.21

target - вероятность подключения услуги

Благодарю за внимание!