Прогнозирование оттока клиентов

Отчет о проведенном исследовании

Цели и задачи

Решается задача прогнозирования оттока клиентов, особенно актуальна для сервис провайдеров с распространением услуги близким к 100%. Данные взяты из клиентской базы французской телекоммуникационной компанией Orange, анонимизированы и обфусцированы. Состоит из 50тыс объектов и включает 230 переменных, 190 – числовые, остальные 40 – категориальные.

In [1]:

```
# изначальный вид данных
import pandas as pd
data = pd.read_csv("orange_small_churn_train_data.csv", index_col='ID')
data.head()
```

Out[1]:

	Var1	Var2	Var3	Var4	Var5	Var6	Var7	Var8	Var9	Var10	...	Var222	Var223
ID													
0	NaN	NaN	NaN	NaN	NaN	3052.0	NaN	NaN	NaN	NaN	...	vr93T2a	LM8l689
1	NaN	NaN	NaN	NaN	NaN	1813.0	7.0	NaN	NaN	NaN	...	6hQ9lNX	LM8l689
2	NaN	NaN	NaN	NaN	NaN	1953.0	7.0	NaN	NaN	NaN	...	catzS2D	LM8l689
3	NaN	NaN	NaN	NaN	NaN	1533.0	7.0	NaN	NaN	NaN	...	e4lqvY0	LM8l689
4	NaN	NaN	NaN	NaN	NaN	686.0	7.0	NaN	NaN	NaN	...	MAz3HNj	LM8l689

5 rows × 231 columns

Решенные в рамках проекта задачи:

- проведение описательного анализа данных;
- формирование ключевых метрик оценки качества модели;
- построение baseline-решения;
- эксперименты с различной обработкой входных данных;
- построение модели прогнозирования и ее оптимизация;
- оценка потенциального экономического эффекта от внедрения финальной модели прогнозирования оттока.

Применение

Для того, чтобы получить прогноз, необходимо подготовить данные согласно алгоритму ниже и применить к ним обученный классификатор следующим образом:

In [ ]:

```
result = clf.predict_proba(test)
churn_probabilities = [x[1] for x in result]
```

Получаем массив вероятностей ухода пользователя, откуда можно установить, какие пользователи наиболее склонны к оттоку

Измерение качества и критерий успеха

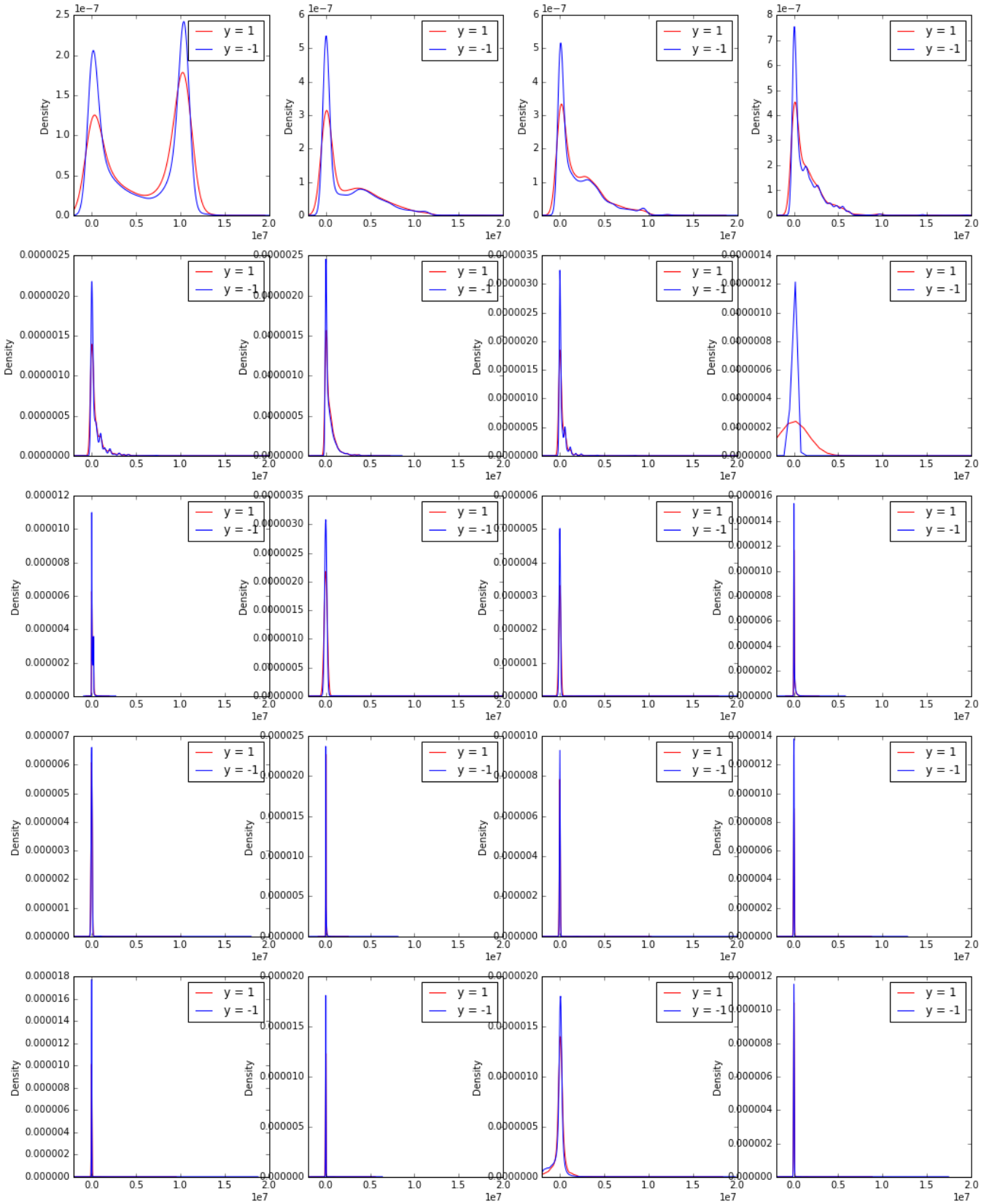
Данные сильно несбалансированы, так как клиентов, которые собираются уходить гораздо меньше, чем тех, кто остается. Поэтому предлагается использовать метрику AUC-PRC (площадь под кривой точность-полнота), так как данная метрика устойчива к дисбалансу классов. Для повышения точности оценки и защиты от переобучения использовалась стандартная техника кросс-валидации. А также замечу, что в дополнение к AUC-PRC в качестве контрольной метрики использовалась AUC-ROC, как наиболее популярная метрика и применимая к широкому кругу задач.

Если не решать задачу прогнозирования оттока, то для произвольного клиента мы можем предположить, что он уйдет с вероятностью 50%. Таким образом, минимальным критерием успеха для нашей модели прогнозирования будет качество выше 50%. Но, естественно, чем выше качество классификации, тем лучше.

Описание развития и подготовки модели.

Анализ данных

В ходе первичного анализа данных было выявлено, что данные достаточно шумные и корреляция с целевой переменно зависит в основном от дисперсии данных.



Наиболее коррелирующие признаки следующие: Var153, Var38, Var133, Var76, Var217, Var214, Var200 Они предположительно вносят наибольший вклад.

Выбор классификатора

При построеннии бейзлайн решения был произведен выбор метрики и подбор классификатора. 1. Ridge классификатор

Ridge classifier:

	AUC-PRC	F1	Acc	Precis	Recall	ROC
3	0.135727	0.015821	0.925367	0.446759	0.008065	0.633292
4	0.138183	0.015766	0.925233	0.440812	0.008065	0.635717
5	0.137340	0.020083	0.925467	0.461859	0.010305	0.637966
7	0.141931	0.020164	0.925633	0.529252	0.010303	0.643937
10	0.142040	0.020990	0.925733	0.539722	0.010758	0.642461

2. Логистическая регрессия

Logistic regression classifier:

	AUC-PRC	F1	Acc	Precis	Recall	ROC
3	0.086963	0.033336	0.916967	0.124202	0.019265	0.540772
4	0.087676	0.033382	0.917000	0.124941	0.019265	0.543011
5	0.087403	0.034789	0.916800	0.127173	0.020159	0.542892
7	0.088865	0.034086	0.916900	0.127013	0.019713	0.544032
10	0.088174	0.036414	0.916933	0.135707	0.021058	0.538644

3. Случайный лес

Random forest classifier:

	AUC-PRC	F1	Acc	Precis	Recall	ROC
3	0.121775	0.005322	0.925633	0.224242	0.001792	0.585742
4	0.118918	0.007083	0.925533	0.229167	0.004480	0.572952
5	0.108963	0.004429	0.925433	0.183333	0.002241	0.584131
7	0.117147	0.008868	0.925267	0.232143	0.002242	0.589793
10	0.116088	0.008838	0.925667	0.143333	0.006270	0.597411

4. Градиентный бустинг

Gradient boosting classifier:

	AUC-PRC	F1	Acc	Precis	Recall	ROC
3	0.211896	0.033925	0.926100	0.559147	0.017473	0.734462
5	0.212360	0.034781	0.926167	0.589283	0.018368	0.738851
8	0.215422	0.038897	0.926167	0.600339	0.020161	0.737432

Градиентный бустинг был выбран в качестве основного классификатора.

Подбор параметров

В ходе работы было проверено множество наборов параметров с помощью GridSearchCV, например

parameters_grid = {

```
    'max_depth' : [5, 10, 15, 20, 25],
    'min_samples_leaf' : [1, 2, 3, 5, 8],
    'n_estimators' : [10, 50, 100, 300],
    'criterion' : ['gini', 'entropy'],
    'max_features' : [0.2, 0.4, 0.6, 0.7, 0.8, 0.9]
```

}

Также производился подбор весов классов

params = {

```
    "class_weight": [{1: 4}, {1: 8}, {1: 10}, {1: 15}, {1: 20}, {1: 50}]
```

}

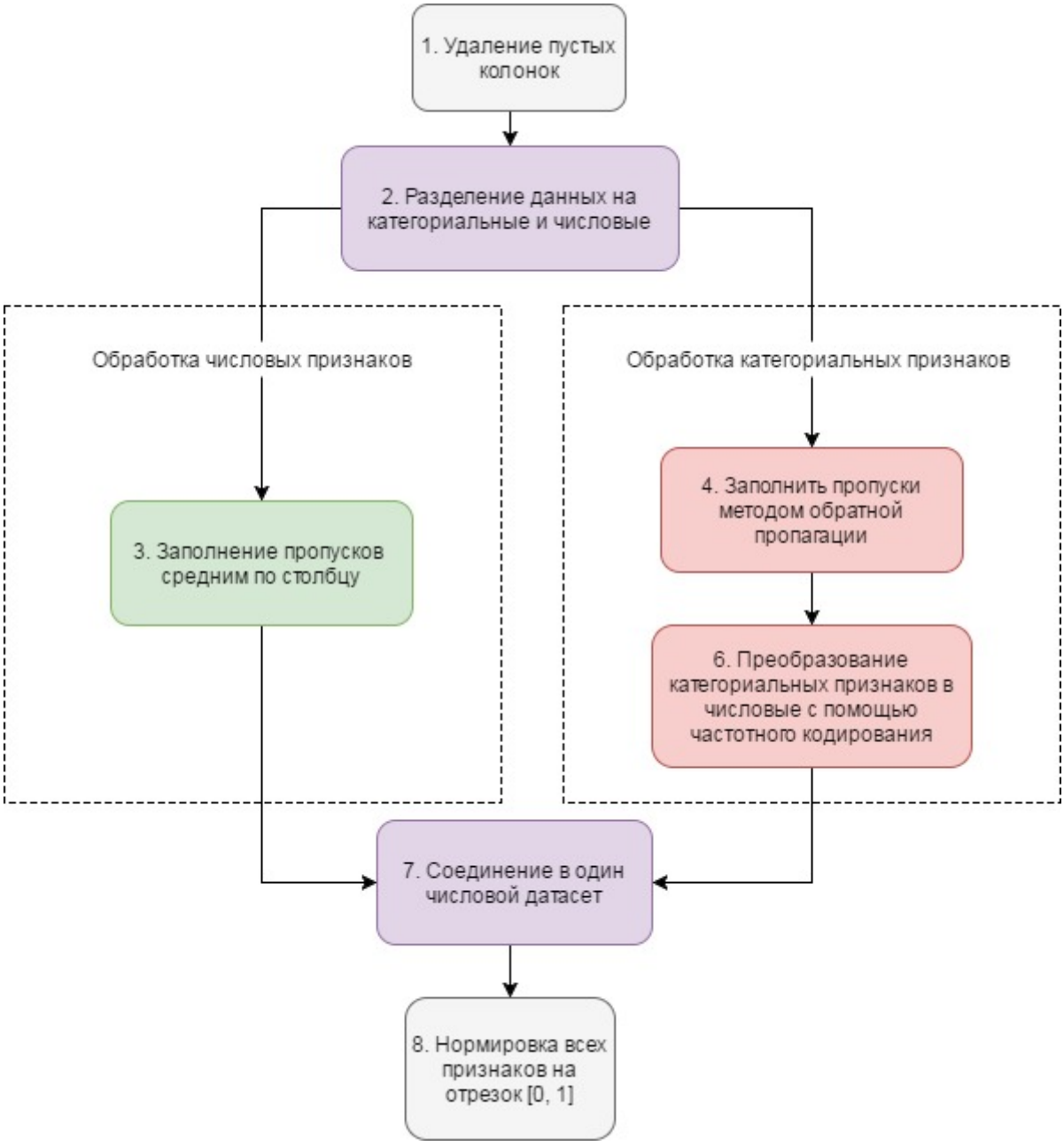
Для градиентного бустинга оптимальный набор параметров:

n_estimators=300, learning_rate=0.2, max_depth=5, loss='exponential', max_features=0.3

Однако, проверка на отложенной выборке показала, что бустинг со стандартными параметрами и n_estimators=200 дает все-таки более хорошее качество, так как, видимо, слабее переобучен.

Построение конечной модели

1. **Процесс подготовки данных можно проиллюстрировать следующим образом:**



После преобработки данные выглядят гораздо лучше!

In [4]:

```
prepared = pd.read_csv("data_prepared.csv", index_col='Id')
prepared.head()
```

Out[4]:

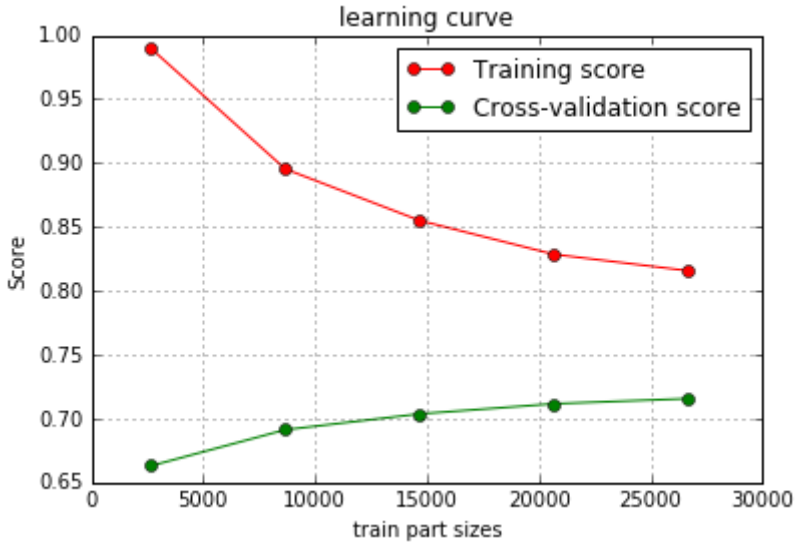
	Var1	Var2	Var3	Var4	Var5	Var6	Var7	Var9	Var10	Var11	...	Var219	Var22
Id													
0	0.0	0.0	0.0	0.0	0.0	0.023163	0.00	0.0	0.0	0.0	...	0.52381	0.729
1	0.0	0.0	0.0	0.0	0.0	0.013760	0.05	0.0	0.0	0.0	...	0.52381	0.731
2	0.0	0.0	0.0	0.0	0.0	0.014822	0.05	0.0	0.0	0.0	...	0.52381	0.081
3	0.0	0.0	0.0	0.0	0.0	0.011635	0.05	0.0	0.0	0.0	...	0.52381	0.986
4	0.0	0.0	0.0	0.0	0.0	0.005206	0.05	0.0	0.0	0.0	...	0.52381	0.418

5 rows × 207 columns

2. **Далее необходимо обучить на подготовленных данных GradientBoostingClassifier и модель готова к работе!**

Оценка результатов

Данная модель имеет уже неплохое качество, так как обучение происходило на достаточно большом массиве данных. На графике кривой обучения можно заметить, что дальнее увеличение датасета не даст значительного прироста качества.



Также отметим, что оценка качества с помощью кросс-валидации и метрики качества AUC-PRC позволяет утверждать, что модель не переобучена и будет хорошо себя проявлять на новых данных.

Экономический эффект

Клиентская база orange насчитывает более 200млн пользователей. По статистике, уходит примерно 7% пользователей в год. Если взять за основу, что средний клиент тратит на услуги 10 евро в месяц, получаем потенциальную потерю прибыли *~140млн* евро в месяц!

Построенная модель имеет качество в 73%, что позволяет определить группу пользователей склонных к оттоку, размером не более 10% от общего числа клиентов, содержащую наши 7% действительно собирающихся уходить пользоватей с точностью 95%.

Таким образом, считая, что пользователи примут наше предложение всего лишь с вероятностью 50% и применяя методики удержания с бюджетом 2 евро на человека только к выделенной группе, получаем дополнительную прибыль в *~30млн* евро!

Итоги

Результатом проделанной работы является обученный на подготовленных данных классификатор, готовый к практическому применению, алгоритм подготовки данных для классификации, а также оценки качества и эффекта от применения.

Улучшение модели

Для дальнейшего развития и улучшения модели можно собрать еще данных, а также дальше поэкспериментировать с новыми моделями и преобразованиями признаков.

Внедрение

Для внедрения классификатора потребуется дополнительно автоматизировать процесс предобработки новых данных согласно описанному алгоритму.

Тестирование

При оценке экономического эффекта была предложена конкретная вероятность принятия предложения. Но перед внедрением в продакшен необходимо этот параметр уточнить с помощью А/Б тестирования