Проект "Предсказание вероятности подключения услуги"

Гурина Ольга, факультет Geek University Искусственного интеллекта

Май, 2021

Обзор данных

В качестве исходных данных представлена информация об отклике абонентов на предложение подключения одной из услуг. Каждому пользователю может быть сделано несколько предложений в разное время, каждое из которых он может или принять, или отклонить.

Отдельным набором данных является нормализованный анонимизированный набор признаков, характеризующий профиль потребления абонента. Эти данные привязаны к определенному времени, поскольку профиль абонента может меняться с течением времени.

Данные train и test разбиты по периодам – на train доступно 4 месяцев, а на test отложен последующий месяц.

Итого, в качестве входных данных представлены:

- **data_train.csv**:
  - id,
  - vas_id,
  - buy_time,
  - target
- **features.csv.zip**:
  - id,
  - ..feature_list..

И тестовый набор:

- **data_test.csv**:
  - id,
  - vas_id,
  - buy_time

Описание датасета

- **id** - идентификатор абонента
- **vas_id** - подключаемая услуга
- **buy_time** - время покупки, представлено в формате timestamp, для работы с этим столбцом понадобится функция datetime.fromtimestamp из модуля datetime.
- **target** - целевая переменная, где 1 означает подключение услуги, 0 - абонент не подключил услугу соответственно.

Информация о модели, ее параметрах, особенностях и основных результатах

Задача

Требуется на основании имеющихся данных об абонентах Мегафон построить алгоритм, который для каждой пары пользователь-услуга определит вероятность подключения услуги.

Модель CatBoost

Для решения задачи применена модель CatBoost со следующими параметрами:

Константные параметры:

- loss_function='Logloss' - показатель, используемый для обучения
- eval_metric='F1' - метрика, используемая для обнаружения переобучения
- auto_class_weights='Balanced' - автоматический подбор весов для балансировки классов
- random_state=42 - случайное зерно, используемое для обучения
- logging_level='Verbose' - вывод оптимизированных метрик, затраченного и оставшегося времени обучения
- task_type='GPU' - используется CPU или GPU. По умолчанию стоит CPU
- cat_features=f_categorical - массив с категориальными признаками
- one_hot_max_size=20 - максимальное количество уникальных значений среди категориальных признаков
- early_stopping_rounds=50 - отслеживание переобучения

Лучшие подбираемые параметры (с использованием сетки гиперпараметров):

- depth=10 - глубина дерева
- learning_rate=0.03 - скорость обучения
- iterations=100 - максимальное количество построенных деревьев
- l2_leaf_reg=20.0 - коэффициент при члене регуляризации L2 функции потерь
- bagging_temperature=2.0 - настройка интенсивности байесовского бутстрапа, по умолчанию=1

Подбор гиперпараметров модели CatBoost выполняется при помощи рандомизированного поиска по сетке с использованием кросс-валидации, проверяется 30 наборов гиперпараметров.

Результаты модели CatBoost

F1 = 0.47 по качеству прогноза для класса 1 – подключение услуги абонентом.

F1 = 0.92 по качеству прогноза для класса 0 – не подключение услуги абонентом.

AUC_ROC = 0.859

AUC_PR = 0.353

Обоснование выбора модели и ее сравнение с альтернативами

Модель CatBoost имеет чуть лучший показатель F1 по сравнению с альтернативной моделью логистической регрессии.

Модель логистической регрессии

Построение модели логистической регрессии выполняется с автоматической балансировкой классов class_weight='balanced'с применением пайплайнов:

- Предобработка: StandardScaler(), OneHotEncoder()
- Селекция: SelectPercetile()
- Модель: LogisticRegression()

Подбор гиперпараметров модели LogisticRegression выполняется при помощи поиска по сетке с использованием кросс-валидации.

Подобранные параметры модели логистической регрессии:

- model__C=5 - обратная сила регуляризации
- selector__percentile=5 - процент лучших признаков

Результаты модели логистической регрессии

F1 = 0.46 по качеству прогноза для класса 1 – подключение услуги абонентом.

F1 = 0.92 по качеству прогноза для класса 0 – не подключение услуги абонентом.

AUC_ROC = 0.845

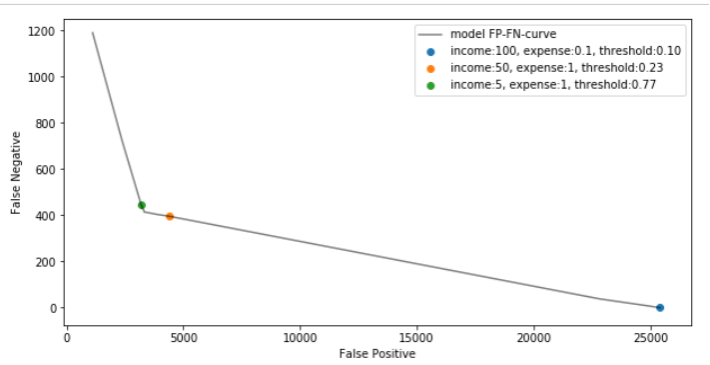
AUC_PR = 0.350

Принцип составления индивидуальных предложений для выбранных абонентов

Предлагаемый принцип - максимизация получаемой прибыли при заданных значениях дохода от подключенной услуги и затраты на рассылку предложения.

Используем формулу минимизации и, задавая конкретные значения Дохода от услуги и Затрат на предложение, найдем порог классификации, при котором достигается максимальная выгода.

График количества NFN и NFP при разных порогах на предсказаниях модели Catboost



- доход 100 р., затраты 10 коп. - наиболее оптимальный порог для предсказаний с максимальной выгодой - 0.10
- доход 50 р., затраты 1 руб. - наиболее оптимальный порог для предсказаний с максимальной выгодой - 0.23
- доход 5 р., затраты 1 руб. - наиболее оптимальный порог для предсказаний с максимальной выгодой - 0.77

Прибыль оператора от положительного отклика клиента на услугу = Доход от клиента - Затраты на рассылку предложения этому клиенту

- ошибка первого рода отражает доход, который оператор потерял, не отправив предложение.
- ошибка второго рода отражает затраты оператора на рассылку, которые оказались напрасными.

Прибыль от рассылки предложений: $REAL = N_{TP} \cdot (Доход - Затраты) - N_{FP} * Затраты$ Упущенная прибыль: $LOSS = N_{FN} * (Доход - Затраты)$ Максимально возможная прибыль: $MAX = (N_{TP} + N_{FP}) * (Доход - Затраты)$	где: N_{TP} - количество положительных откликов на отправленное предложение N_{FN} - количество упущенных клиентов, готовых подключить услугу (ошибка первого рода, FN) N_{FP} - количество напрасно отправленных предложений (ошибка второго рода, FP)

Для получения максимальной выгоды необходимо минимизировать разницу MAX - REAL, то есть упрощая выражения:

$$(Доход - Затраты) / Затраты * N_{FN} + N_{FP} \rightarrow min$$

Так как Доход от подключения услуги, как правило, на несколько порядков превышает Затраты на рассылку предложения (например, услуга со стоимостью подключения 100 р., затрата на смс-рассылку 10 коп.), то N_{FN} гораздо сильнее влияет, чем N_{FP}, на изменение выгоды. Максимально возможная выгода достигается лишь, когда $N_{FN} = N_{FP} = 0$, то есть оператор абсолютно безошибочно разослал все предложения. Это практически невозможно и является идеальным случаем.