Командный проект по исследованию данных в рамках научно исследовательского семинара

Павлов Степан
Ворсин Андрей

Ноябрь 2023

Содержание

1 Введение

В связи с большой конкуренцией банкам нелегко убедить потенциального вкладчика открыть срочный депозит, поскольку на решение клиентов о покупке будет влиять множество факторов, например, время звонка, настроение рынка, наличие работы, возраст и пол клиента и так далее. Чтобы решить эту проблему, мы используем современные методы анализа данных и машинного обучения, выявляя закономерности в клиентском поведении. В частности, мы сравниваем три метода машинного обучения: метод k соседей, случайный лес в sklearn и catboost.

В рамках данного исследовательского проекта нам предстоит изучить данные [1] о результатах маркетинговой кампании банка. Задачей будет являться бинарная классификация на предмет открытия вкладов в банке. Таким образом нам предстоит выяснить, какие из представленных параметров оказывают наибольшее влияние на успешный исход телефонного разговора.

2 Проделанная работа

2.1 Разведочный анализ данных

Для начала с помощью библиотеки **dataprep** мы создали автоматически дэшборд с основными графиками нашего распределения. Дэшборд дал базовое представление о том, с какой страной мы имеем дело. Португалия - развитая страна с большой долей сектора услуг в экономике, высоким уровнем образования (почти нулевая безграмотность, наибольшая группа людей имеет высшее образование). Мы выяснили, что в 88 процентах случаев банк получает отказ. Всего про трёх клиентов известно, что у них есть текущие просроченные задолженности, все такие клиенты не согласились сделать вклад по очевидным причинам. Среди остальных клиентов 80% точно не объявили дефолт, про остальные 20% эта информация неизвестна. Причём, клиенты, про которых неизвестен статус дефолта, в 2.5 раза реже соглашаются на вклад. Вероятно, это происходит из-за относительно большей доли тех, кто на самом деле объявил дефолт среди тех, про кого это не известно. Этот факт усложняет обучение модели.

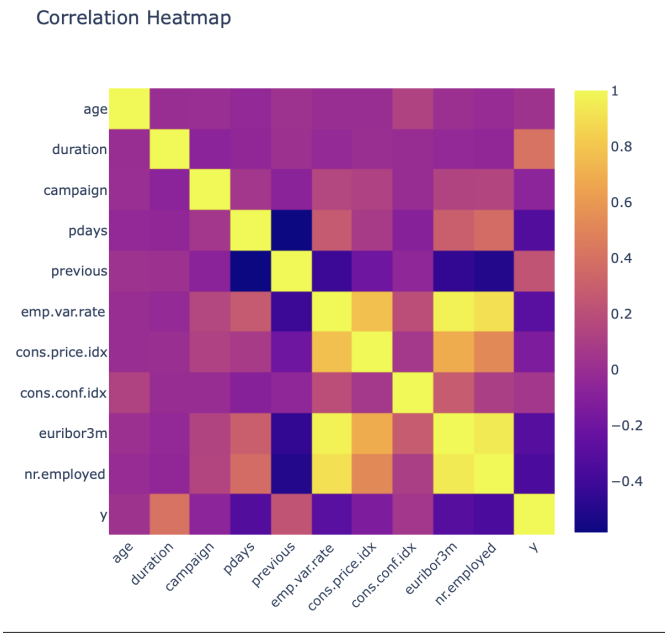

Рис. 1: Correlation heatmap

Карта корреляций даёт больше инсайдов о наших данных. Параметр, имеющий самую большую корреляцию с таргетом - это **duration**. Владельцы датасета рекомендуют убрать этот параметр из процесса обучения, поскольку **duration** известная только по окончании диалога, когда становится известен результат диалога. Следующим по важности параметром был **nr.employed**. Мы выяснили, что он показывает поквартальную занятость в португальской экономике. Этот параметр важен, поскольку он хорошо коррелирует с общим состоянием экономики, о чём говорит известный закон Оукена. Похожим смыслом обладает показатель **emp.var.rate**, отражающий процентный рост занятости в выбранный квартал. Эти два параметра сильно скоррелированы (коэффициент корреляции пирсона более 0.9).

Ещё одним важным параметром является **euribor3m**, описывающий среднесуточную ставку по депозитам с интервалом в 3 месяца. Данный показатель имеет отрицательную взаимосвязь с предсказываемым параметром модели, так как со снижением ключевой ставки, становится меньше клиентов, желающих открыть вклад.

Так же стоит обратить внимание на **pdays** и **previous**. Чем больше дней прошло с последнего контакта, тем меньше вероятность, что клиент согласится на вклад. Обратная ситуация с параметром **previous**: чем больше было контактов до этого, тем больше вероятность, что и сейчас клиент согласится сделать вклад.

Категориальные признаки также дали какое-то количество дополнительной информации. Например, студенты и пенсионеры значительно чаще соглашаются открыть вклад (с вероятностями 0.31 и 0.25 соответственно). Одинокие люди также значительно чаще соглашались на вклад, чем женатые (0.14 против 0.10 при 10000+ наблюдений в каждой группе). Также значительно чаще соглашались те, кому звонили на мобильный телефон (0.14 против 0.05

с 10000+ наблюдениями в каждой группе). Успех при прошлом звонке сильно повышают шансы (до 0.65) что клиент снова согласится на предложение.

2.2 Оценка обученных моделей

Правильная и точная оценка любой модели имеет ключевое значение для её правильного дальнейшего развития. Важным замечанием по исследуемому датасету станет, крайне неравномерное распределение исходов, немногим более 11%. Данный фактор нужно учитывать, поскольку базовая модель оценки точности **accuracy** будет давать завышенные показатели. Было принято решение так же использовать метрику **ROC-AUC** [2] (receiver operating characteristic - area under the curve). Данный способ оценки широко применяется для задач бинарной классификации. Также для каждой модели будет представлена матрица ошибок (Confusion Matrix), которая показывает количество правильно либо же неверно оценённых классов, а также ложно положительные и ложно отрицательные исходы.

Для справки, есть ещё одна интересная метрика **F-1**. Она является гармоническим средним между полнотой (метрика recall), которая показывает как много из фактических успешных значений были правильно классифицировано как успешные, а также точностью (метрика presicion). Она отображает как много из предсказанных успешных прогнозов действительно успешны. Подытожим:

$$\text{Precision} = \frac{\text{TP (True Positive)}}{\text{TP (True Positive)} + \text{FP (False Positive)}}$$

$$\text{Recall} = \frac{\text{TP (True Positive)}}{\text{TP (True Positive)} + \text{FN (False Negative)}}$$

$$\text{F-1} = \frac{\mathbf{2 \cdot Recall \cdot Precision}}{\mathbf{Recall + Precision}}$$

2.3 KNN

Далеко не секрет, что многие люди имеют схожие предпочтения и поведение, особенно в рамках описанной ситуации об открытии вклада. Таким образом сразу на ум приходит идея применить модель KNN (K-Nearest Neighbors).

Данная модель является одной из основополагающих для задач классификации и регрессии. Основная идея KNN заключается в том, чтобы давать прогноз целевой переменной для нового наблюдения, основываясь на значениях ближайших соседей. Основная задача при обучение данной модели - выявить оптимальное количество соседей.

Для начала модель была обучена с параметром 5. Это дало базовое понимание жизнеспособности данной модели, и оно оказалось положительным. Модель дала accuracy = 0.89, ROC-AUC = 0.63. Это говорит о том, что модель куда лучше чем просто случайное угадывание.

Первая модель будет брать во внимание только исходные числовые параметры. Следующим шагом будет поиск оптимального параметра соседей. Для этого можно просто пробежаться циклом от 1 до 30 (тут этого будет достаточно) и построить график точности модели при соответствующих значениях.

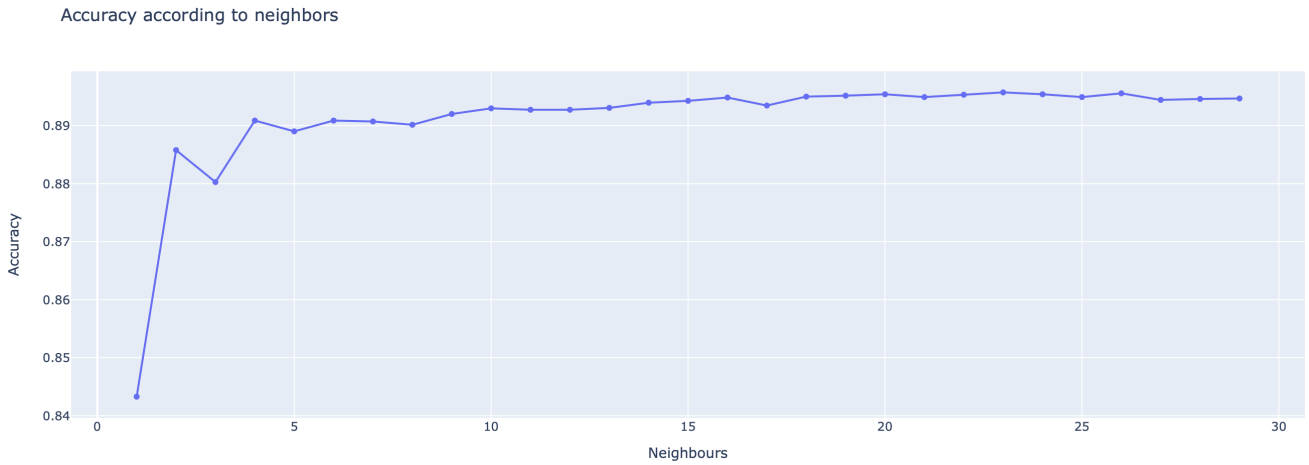

Рис. 2: Accuracy vs N

Судя по графику (Рис.2) оптимальное значение будет в промежутке от 10 до 30. Чтобы упростить поиски воспользуемся методом оптимизации гиперпараметров, например Grid Search Cross-Validation. Итогом её работы станет комбинацию параметров, которая дала наилучшую производительность на кросс-валидации. В результате получились следующие значения: оптимальное число соседей (k = 28), accuracy на тренировочных данных = 0.90, на тестовых = 0.89, ROC-AUC же равен 0.59. Матрица ошибок вышла такой (Рис.3). Теперь попробуем учитывать категориальные признаки. Для этого необходимо перевести их в числовые. Далее всё также проведём поиск гиперпараметров. На выходе получили следующие показатели: число соседей резко сократилось до 6, остальные значения изменились в пределах ±0.01. Вероятнее всего при добавлении новых признаков размерность данных увеличилась, то есть точки, представляющие данные теперь расположены более разряжено. Как следствие, возможно, получилось так, что соседи которые раньше были расположены близко, теперь находятся далеко друг от друга и между ними втиснулись другие, поэтому модели теперь требуется выбирать меньше ближайших точек.

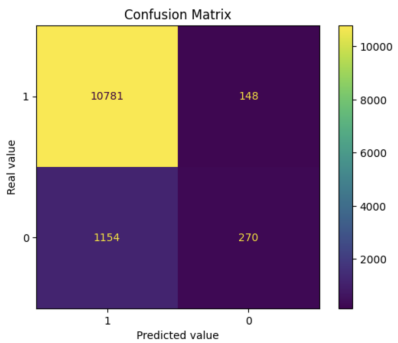

Рис. 3: Confusion matrix KNN

2.4 Random forest

В качестве бейзлайна мы выбрали обычный случайный лес,
поскольку его очень легко реализовать и интерпретировать. Так как классы оказались совершенно неравны (один составляет 88% наблюдений, а другой оставшиеся 12%, мы не хотим использовать метрику **accuracy**. Для нашей задачи больше подойдёт метрика **ROC-AUC** [2]. Затем, мы используем библиотеку optuna, которая может перебрать гиперпараметры быстрее, чем gridsearch. **ROC-AUC** лучшей модели на тестовой выборке был равен 0.607.

Большая эффективность, чем в KNN связана, вероятно, с тем, что RF имеет более гибкий подход к кластеризации, возвращая значение от 0 до 1

2.5 XGBoost

Экстремальный градиентный бустинг по праву считается одним из лучших алгоритмов машинного обучения. Мы использовали его реализацию из библиотеки XGBoost. В работе нам также помогла optuna, с помощью которой мы подобрали гиперпараметры модели. ROC-AUC на тесте сильно вырос по сравнению с бейзлайном и knn и составил 0.77. Причём, время обучения значительно сократилось, что позволило перебрать большее число гиперпараметров.

Комбинация градиентного бустинга и optuna является бенчмарком в современном машинном обучении, который позволят очень быстро обучать модели, имюещие высокую точность. К этим двум моделям мы также добавили CatBoost category encoder, который позволил превратить категориальные признаки в количественные, сохранив внутреннюю логику их влияния на таргет.

```
loan: 0.17129847816290572
contact: 0.12047683893931882
poutcome: 0.10934393236527223
nr.employed: 0.09869490881073797
housing: 0.09096835246458305
default: 0.08614304123416855
month: 0.056852468777536816
day_of_week: 0.04732182631531878
euribor3m: 0.04597033493268773
pdays: 0.04290251733336162
education: 0.02701841743234143
job: 0.0250422944606633
marital: 0.018775538328775994
cons.conf.idx: 0.018164065084529175
cons.price.idx: 0.01620667987476788
age: 0.008686464080702729
emp.var.rate: 0.008024936830797835
previous: 0.004897435806302406
campaign: 0.003211468765227849 7
```

(a) RF feature importance

```
nr.employed: 0.7888926863670349
loan: 0.04807936027646065
pdays: 0.03466339036822319
month: 0.011982926167547703
poutcome: 0.011599799618124962
default: 0.010682146064937115
contact: 0.009467512369155884
previous: 0.008406191132962704
emp.var.rate: 0.007677559275180101
day_of_week: 0.0075682527385652065
euribor3m: 0.007271585986018181
education: 0.006998923607170582
marital: 0.006967867258936167
job: 0.006963368505239487
cons.conf.idx: 0.006895795930176973
housing: 0.006715795025229454
campaign: 0.006595059297978878
age: 0.006566652096807957
cons.price.idx: 0.006005161441862583
```

(b) XGB feature importance

Рис. 4: RF and XGB feature importance

3 Результаты

В целом, несмотря на простоту устройства модели k-ближайших соседей и плохо распределённых данных, получилось добиться вполне неплохих результатов. Тем не менее, существуют более продвинутые модели, дающие более точные результаты. Таким образом, сравнив три модели, можно сказать, что XGBoost показал самую высокую точность (Рис. 5). Однако, данная модель не является ультимативной, и подходит далеко не для всех задач.

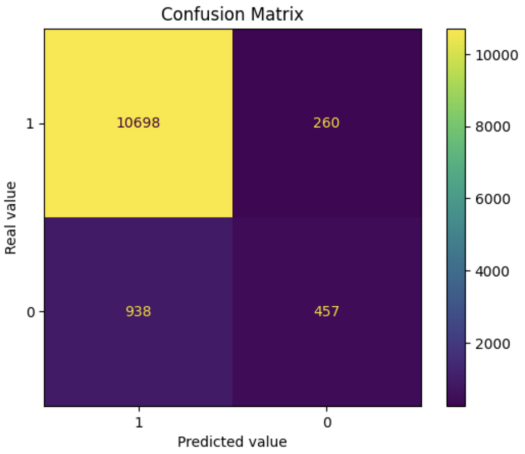

Рис. 5: Confusion matrix XGB

Список литературы

[1] S. Moro, R. Laureano and P. Cortez. (2011). Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121.

[2] Zengchang Qin. (2008). ROC analysis for predictions made by probabilistic classifiers