DS Хакатон МКБ: Обсуждение решений задачи

Сюй Минчуань

Прикладные задачи анализа данных. Соревнование МКБ

8 декабря 2021 г.

Описание задачи и данных

- **Задача** — построить аппликационную модель **оценки вероятности** предъявления по гарантии, чтобы минимизировать потери от кредитного риска. В качестве предикторов для моделирования используются анкетные данные клиента и его финансовые показатели. Решения оцениваются по метрике **ROC/AUC** Score.
- Данные: 124 фичей, вещественные (F-фичи), категориальные, временные, много пропусков. TARGET - бинарный.

Описание данных, EDA

```
plt.plot(train.isnull().sum())
```

```
[<matplotlib.lines.Line2D at 0x1d1d7e894c0>]
```

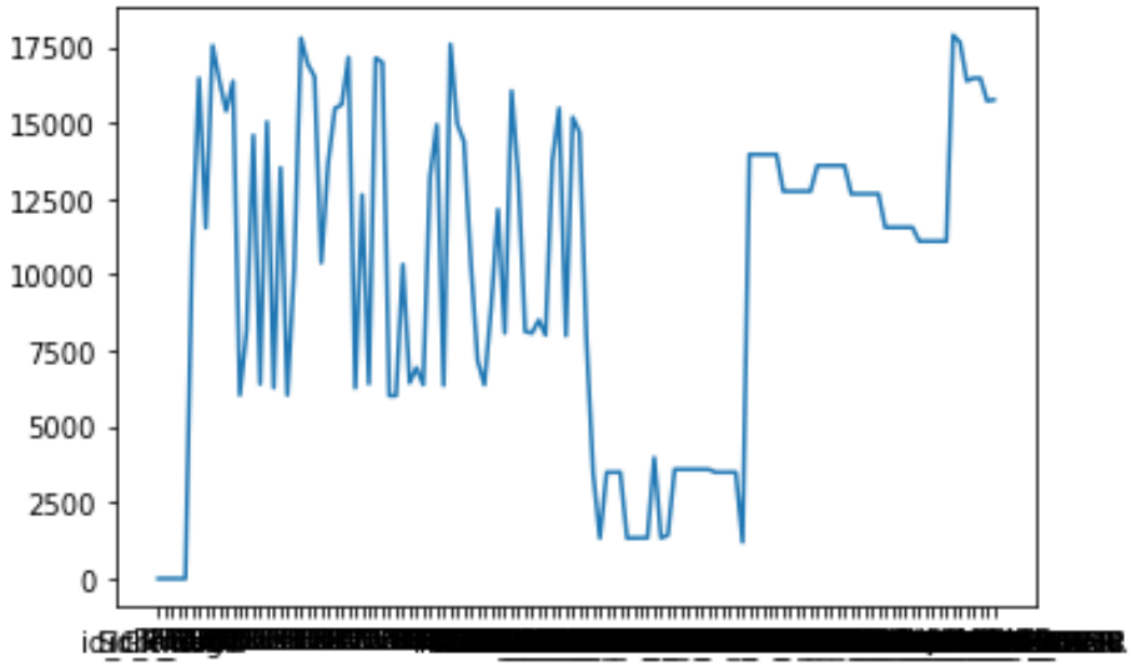

Рис.: Пропуски фичей

Описание данных, EDA

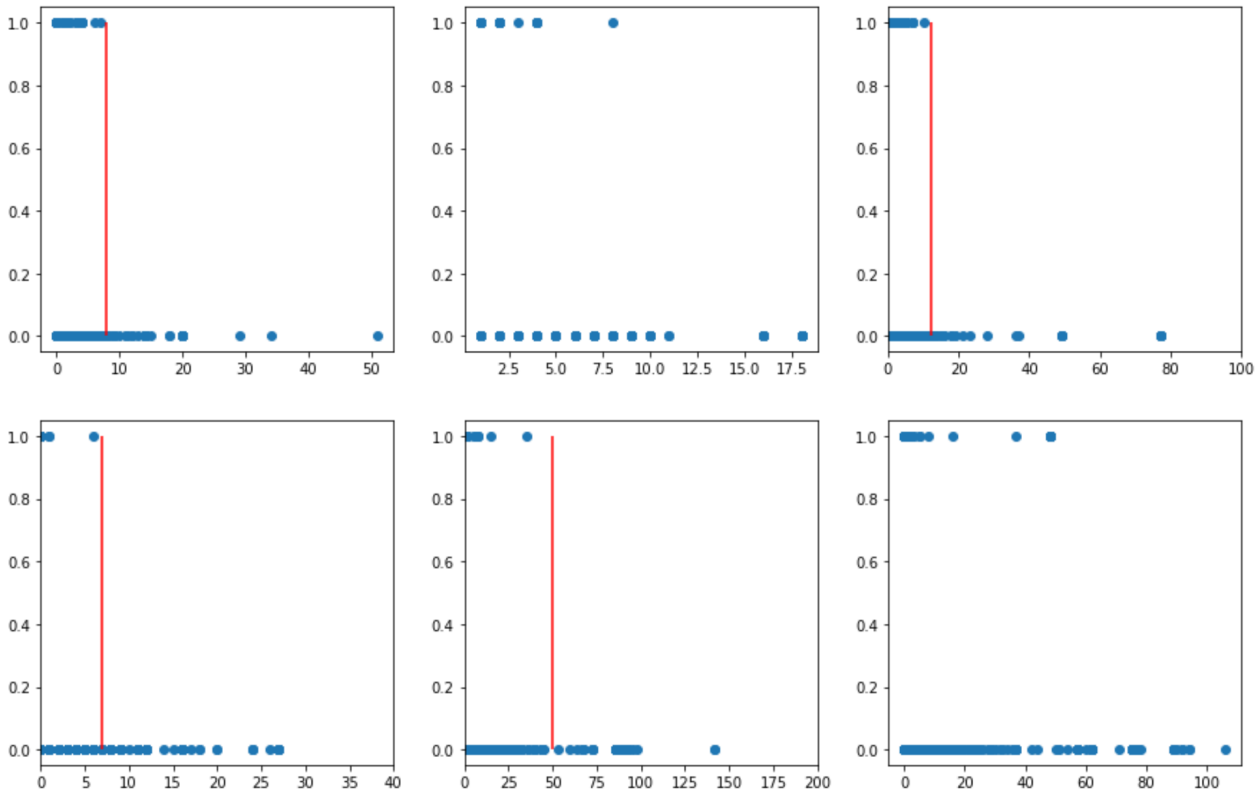

Рис.: Вещественные признаки (кроме F-фичей)

Описание данных, EDA

```
plt.plot(train['F1200'])
```

```
[<matplotlib.lines.Line2D at 0x28cab779280>]
```

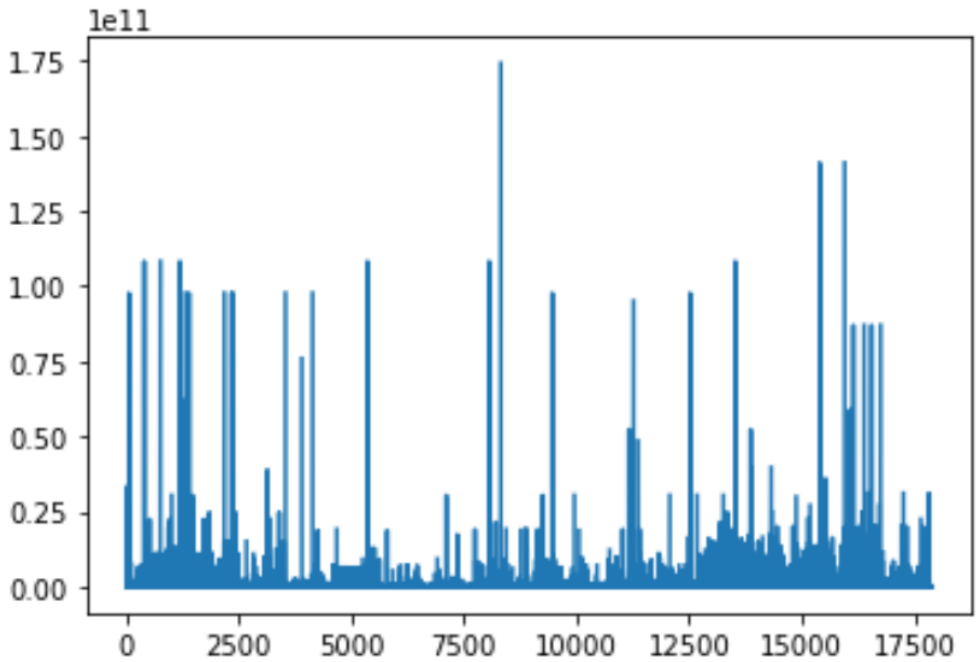

Рис.: F-фичей (в частности F1200)

Первоначальные FE и используемые модели

Сгруппировали признаки на категориальные, временные, бинарные, вещественные. Отдельно выделил F-фичей

- для бинарных: data[name].fillna(data[name].mean(),inplace = True)
- для временных: .dt.day, dt.month, dt.dayofweek, dt.dayofyear, когда был предыдущий кредит.
- для категориальных: get_dummies()
- для вещественных: агрегация mean, std, max, min
- prp.StandardScaler()

А модель я использовал случайный лес + GridSearchCV tuning

- rf = RandomForestClassifier(n_estimators = 70, max_depth = 13, min_samples_split = 50)
- roc_auc_score на валидации: 0.8869
- на тесте: 0.8629

Гипотезы и результаты проверок

Что дали прирост скора:

- vari_diff, vari_diff_mean +
- dayofweek +
- 'OKATO_FED','OKTMO_CODE' - коды регионов +

Что не дали:

- Логарифмирование на F-фичей -
- dayofyear, day, month -
- MinMaxScaler of F -
- Пороги для вещественных фичей -

FE и модель после мозгового штурма

- Кодирование категориальных (CITIZENSHIP_NAME, SEX_NAME) на вещественных (как были в бенчмарке)
- для временных: dt.dayofweek, когда был предыдущий кредит, и mean of diff.
- убрал MinMaxScaler, агрегацию, и многие временные признаки

Модель, результаты

- model = lgb.LGBMClassifier(num_leaves=30, learning_rate=0.05, n_estimators=500)
- roc_auc_score на валидации: 0.9280
- на тесте: 0.8911
- был 23 загрузок решений, всего 4-5 суток для попытки придумывания фичей.

Топ-2 (Aleron - 0.932): генерация признаков

- можно посмотреть сюда `https://github.com/a-milenkin/MKB_hack`
- Сгенерировали 210 новых признаков
- Объединение моделей (блендинг: Catboost + Catbost(optuna)): y = y_pred*0.15 + y_pred_optuna*0.85

Какие фичи у него были самые важные:

- F_contract_count_in_day - Количество подписанных контрактов в один день (в день SIGN_DATE)
- F_contract_count_in_day_std
- F_contract_count - Количество подписанных ранее контрактов
- F_contract_count_mean
- F_WORKERSRANGE_change_mean - Среднее значение WORKERSRANGE за историю
- OKVED_CODE_2 - класс оквед кода

Выводы для себя

- Общий пайплайн для DS соревнования.
- Генерация признаков - очень важно
- Тюнинг с использованием GridSearchCV в sklearn
- Опыт использования тех и иных моделей