对外经济贸易大学

2020 — 2021 学年第一学期期末考试

成绩

论文题目 基于分类与回归算法的商业银行贷款信用评分模型

课程代码及课序号 CMP350-1

课程名称 机器学习与数据挖掘

学　　号 201825009

姓　　名 李世杰

学　　院 信息学院

专　　业 信息管理与信息系统

考试时间 2020-12-20

对外经济贸易大学

本科课程期末论文或其他方式考试评阅表

评分要求：*下列仅为建议指标项，任课教师可根据课程需要进行调整或更改。

请选择总分计算方式：1. 总分为各项指标平均分（ ） 2. 总分为各项指标分总和（ ）

指标项*	选题	观点	材料	文字水平	格式与框架	**总分**
分数						

任课教师签字：

年　　月　　日

摘要

本文以构建商业银行贷款信用评分模型为目标。在样本与特征的观察中，经过 GDBT 筛选特征后发现样本特征中借贷标记、月余额与违约标记间有直接的决定性关系，推测违约标记是由借贷标记、月余额计算得出，其规则由银行人为给定；利用 T-SNE 对高维数据可视化发现，虽然样本总体中违约样本与其他样本分布差异不明显，但借贷标记为真的样本中，违约的样本与未违约样本在分布上有明显的区别。

因此，设立了以违约标记为分类目标的评价模型，和以月余额为回归目标的评价模型。前者通过删除借贷标记、月余额特征防止了过拟合与多重共线性。后者从违约判定规则给定的角度考虑，只要实现了对月余额的预测，就能推断贷款违约的风险。两种模型分别用 knn，带 L2 的逻辑回归，XGBoost 回归实现，都在测试集上得到了良好的结果。

最终，分别将主要影响因素归为，资产类、资金流动类与资产配置类。并得出了相关特征的重要性排序以供参考。

关键词

特征工程，高维数据可视化，评价模型

一、算法综述

本次模型用到的算法主要包括 T-SNE，GDBT，knn，逻辑回归，XGBoost。这里对其中几个算法从原理角度简单综述。

1.1 T-SNE

（1）算法简介

t-SNE(t-distributed stochastic neighbor embedding)是用于降维的一种机器学习算法，适用于高维数据降维到 2 维或者 3 维，便于进行可视化。 t-SNE 是由 SNE 发展而来。

（2） SNE 核心思想

通过仿射(affinitie)变换将数据点映射到概率分布上，主要步骤为：

a)构建一个高维对象之间的概率分布，使得相似的对象有更高的概率被选择，而 不相似的对象有较低的概率被选择

b)在低维空间里在构建这些点的概率分布，使得这两个概率分布之间尽可能相似。

（3）原理

先将欧几里得距离转换为条件概率来表达点与点之间的相似度。给定一组高维数据，t-SNE 首先是计算概率两样本的条件概率 $p_{j|i}$，正比于样本间的相似度。将距离转换为概率的概率密度函数人为给定。如：

$$p_{j|i} = \frac{exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}$$

其中σ_i对不同样本点不一致。通常取值为以数据点x_i为中心的高斯均方差。

低维度下的样本 y_i，可以指定高斯分布的均方差为$1/\sqrt{2}$，则相似度为

$$q_{j|i} = \frac{exp(-\|y_i - y_j\|^2 /)}{\sum_{k \neq i} exp(-\|y_i - y_k\|^2)}$$

如果降维的效果比较好，局部特征保留完整，那么$q_{j|i} = p_{j|i}$

因此目标为优化两个概率分布之间的距离即 KL 散度。

$$C = \sum_i KL(P_i|Q_i) = \sum_i \sum_j p_{j|i} log\left\{\frac{p_{j|i}}{q_{j|i}}\right\}$$

KL 散度具有不对称性，在低维映射中不同的距离对应的惩罚权重是不同的。距离较远的两个点来表达距离较近的两个点会产生更大的 cost，相反，用较近的两个点来表达较远的两个点产生的 cost 相对较小。故 SNE 会倾向于保留数据局部特征。

1.2 GBDT

（1）模型结构

GBDT 的核心目标是在不改变原有模型的结构上提升模型的拟合能力，即在已经建立模型的基础上，在建立一个能最小化前一模型残差的模型。

为了防止结构风险过大，还要加上反映模型复杂度的项。

$$Obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

Boosting 采用前向优化算法，即从前往后，逐渐建立基模型来优化逼近目标函数。目标函数可以表示为 n 阶残差拟合函数的和：

$$\begin{aligned}
&\hat{y}_i^0 = 0 \\
&\hat{y}_i^1 = f_1(x_i) = \hat{y}_i^0 + f_1(x_i) \\
&\hat{y}_i^2 = f_1(x_i) + f_2(x_i) = \hat{y}_i^1 + f_2(x_i) \\
&\cdots \\
&\hat{y}_i^t = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{t-1} + f_t(x_i)
\end{aligned}$$

第 t 步拟合的模型预测值可以表示为

$$\hat{y}_i^t = \hat{y}_i^{t-1} + f_t(x_i)$$

$f_t(x_i)$即为第 t 步需要拟合的模型。此时的目标函数为：

$$\begin{aligned}
Obj^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i^t) + \sum_{i=i}^{t} \Omega(f_i) \\
&= \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{t-1} + f_t(x_i)\right) + \Omega(f_t) + constant
\end{aligned}$$

最小化目标函数就求得了第 t 步要拟合的模型。

（2）目标函数

而目标函数可以类比泰勒公式保留二阶变差的形式：

$$f(x + \Delta x) \approx f(x) + f'(x)\Delta x + \frac{1}{2} f''(x)\Delta x^2$$

将 $\hat{y}_i^t = \hat{y}_i^{t-1} + f_t(x_i)$ 中的前项看作 x，后向看作 delta x。

$$Obj^{(t)} = \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{t-1}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + constant$$

目标函数就可简化为带有损失函数一阶导数、二阶导数的表达形式。其中：

$$g_i = \partial_{\hat{y}^{t-1}} (\hat{y}^{t-1} - y_i)^2 = 2(\hat{y}^{t-1} - y_i), \quad h_i = \partial^2_{\hat{y}^{t-1}} (\hat{y}^{t-1} - y_i)^2 = 2$$

在第 t 步 $l(y_i, \hat{y}_i^{t-1})$为常数。

最终目标函数可以表示为：

$$Obj^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t)$$

最后的问题便是用决策树来表示与优化算法。

决策树分类与回归最后都会归结于一个叶子节点，叶节点取值决定代表预测值。假设

存在存在函数 q 将目标函数 $f_t(x)$ 映射为对应叶节点的值 w。

决策树的复杂度可以表示为：

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

设 $I_j = \{i|q(x_i) = j\}$ 表示第 j 个叶子节点的样本集合，则目标函数的表现形式化为：

$$\begin{aligned} Obj^{(t)} &\approx \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)\right] + \Omega(f_t) \\ &= \sum_{i=1}^{n}\left[g_i w_{q(x_i)} + \frac{1}{2}h_i w_{q(x_i)}^2\right] + \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T} w_j^2 \\ &= \sum_{j=1}^{T}\left[(\sum_{i\in I_j} g_i)w_j + \frac{1}{2}(\sum_{i\in I_j} h_i + \lambda)w_j^2\right] + \gamma T \end{aligned}$$

此时对样本集合的运算都变为对叶节点集合的运算。定义

$G_j = \sum_{i\in I_j} g_i$ ， $H_j = \sum_{i\in I_j} h_i$

目标函数求一阶导数为 0 得

$$w_j^* = -\frac{G_j}{H_j + \lambda}$$

最后目标函数最终简化为

$$Obj = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T$$

1.3 Knn

K 临近的分类算法思想为：对于给定的每一个测试样本，基于距离度量找出训练集中与其最靠近的 K 个训练样本，基于周围样本的信息做预测。在分类中经常使用投票法，即周围哪类样本最多自己就取相同的值。

其特点在于不需要训练，完全依赖已有信息。

实现的核心在于：距离，k 值，高效找到临近点的算法。

距离的计算通过闵可夫斯基距离实现：

$$D(x,y) = \sqrt[p]{(|x_1 - y_1|)^p + (|x_2 - y_2|)^p + ... + (|x_n - y_n|)^p} = \sqrt[p]{\sum_{i=1}^{n}(|x_i - y_i|)^p}$$

K 通过交叉验证选择。

找临近点的方式包括：暴力搜索，kd 树，球树。

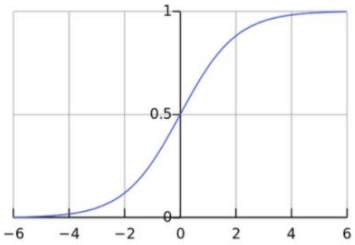
其中 Kd 树划分与决策树相近，选择方差最大的特征切分，切分值为中位数。进而递归到无样本可分，把整个取值空间划分成叶子节点的并集。

预测测试集时，先自上而下找到包含该样本的叶节点。再自下而上回溯，找兄弟节点

中的训练集样本，不足时返回到父节点递归查找。直至达到 k 值或根节点为止。

1.4 Logistic

逻辑回归通过对线性函数进行非线性变换，将拟合的曲线变成如下形状：



拟合曲线

$$h_\theta(x) = sigmoid(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$$

然后根据极大似然估计的原理，在给定样本与拟合曲线的条件下，调整系数 theta 使得残差最小。
及最小化损失函数

$$J(w) = -\frac{1}{n}(\sum_{i=1}^{n}(y_i lnp(x_i) + (1-y_i)ln(1-p(x_i)))$$

通常的最小化方法为梯度下降。即随机选定系数的初始值，对每个系数进行如下操作：
1、系数对损失函数求偏导

$$\frac{\partial J(\Theta)}{\partial \theta_j} = \frac{1}{n}\sum_{i=1}^{n}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)}$$

2、按照一定的学习率调整系数。即系数-学习率×成本函数值对系数的变化率。

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta)$$

经过多次迭代，系数最终收敛。模型的训练也就完成了。

二、问题分析

建模目标为根据商业银行的客户信息，构建客户信用评分模型。其中，被解释变量为 bad_good（是否违约），即要通过对其他特征的观察，找到主要影响违约的因素。但在特征工程中，发现 LOAN_FLAG 与 G_OS_PRCP_SUM 两个属性较为特殊。在现有训练集中凭借两者可以直接推断违约结果，只要发生借贷且账户当月余额为 0 则有违约标记，反之均无。

可能由于这个规律本身就是银行判定当前违约的法则，不能以此为判断依据。模型应能够提前预判客户违约的风险而非在知道当月的信息后确定是否违约。

因此，对建模目标进行了调整。思路有二：

一是对 LOAN_FLAG 为真的样本去掉 G_OS_PRCP_SUM 用其他特征建模，即在

未知本月余额时关注能提前预判违约的特征。

二是利用银行的判定规律，去掉 bad_good，以 G_OS_PRCP_SUM 为被解释变量进行回归。银行可以通过模型预测客户可能的月余额，再结合时间月净支出等特征，自定义风险评估等级。

三、建模过程

3.1 数据预处理

训练集共 8 万条样本，627 个特征（包括被解释变量）。其中 3.8%的客户违约。样本不均衡。去掉客户号，机构号等无关变量，并筛去只有一个取值的特征。观查到有不到总样本量 1%的样本存在缺失值，被解释变量在其上与剩余样本上分布的期望基本一致。判断为完全随机丢失，直接删除对结果无影响。

```
0.0    686
1.0     36
Name: bad_good, dtype: int64
0.0    76247
1.0     3031
Name: bad_good, dtype: int64
```

属性持有货币型基金标志、持有偏债型基金标志、持有偏股型基金标志都有 0、1、N 三种取值。可以看出 0 和 N 的违约率较为接近但 0 值的违约率都高于 N。由于没有详细的特征取值说明，难以推断取值含义，直接进行合并操作，方便数据处理。

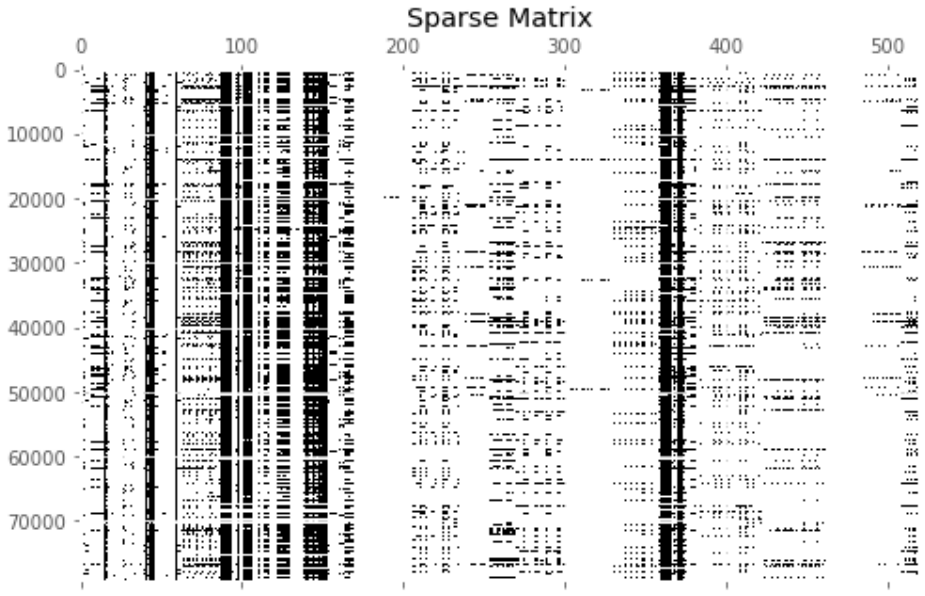
```
 C_FUND_FLAG 取值为0的样本
0.0    6317
1.0     283
Name: bad_good, dtype: int64 违约率: 0.04287878787878788
取值为1的样本
0.0    358
1.0      9
Name: bad_good, dtype: int64 违约率: 0.02452316076294278
取值为N的样本
0.0    70258
1.0     2775
Name: bad_good, dtype: int64 违约率: 0.037996522120137474

 D_FUND_FLAG 取值为0的样本
0.0    6569
1.0     291
Name: bad_good, dtype: int64 违约率: 0.042419825072886296
取值为1的样本
0.0    106
1.0      1
Name: bad_good, dtype: int64 违约率: 0.009345794392523364
取值为N的样本
0.0    70258
1.0     2775
Name: bad_good, dtype: int64 违约率: 0.037996522120137474
```

最后调整变量类型，完成数据预处理工作。

3.2 描述性统计

3.2.1 数据特征

首先观察数据稀疏程度。



可以看出有部分特征绝大部分取值不为 0，而部分特征取值几乎都为 0。严格说不属于稀疏矩阵。考虑到违约样本的不均衡性，不对方差较小的特征筛选。

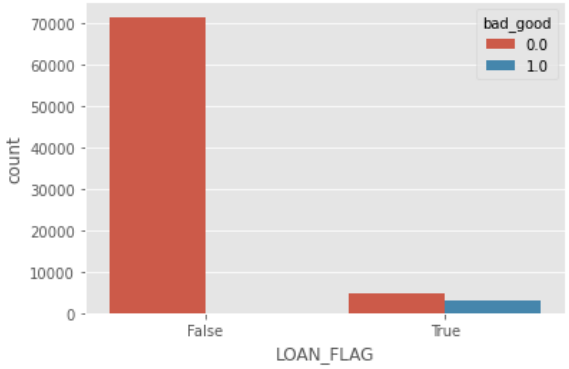
其次观察样本数字特征，发现不同特征取值范围差异很大，且许多特征峰度与偏度都很大。即存在极端值。因此不能使用归一化或标准化等常规方法。

3.2.2 重要特征观察

3.2.2.1 GDBT 第一次特征筛选

GDBT 选出预测违约与否最有影响力的特征为 G_OS_PRCP_SUM 与 LOAN_FLAG。先观察这两个特征的分布。

（1）个贷标识

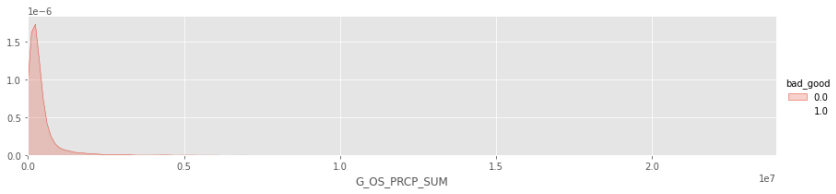


```
In [26]:
1 data2[data2['LOAN_FLAG']==False]['bad_good'].value_counts()
2 # 未借贷的人均未违约

Out[26]: 0.0    71526
Name: bad_good, dtype: int64

In [27]:
1 data2['bad_good'].value_counts()

Out[27]: 0.0    76247
1.0     3031
Name: bad_good, dtype: int64
```

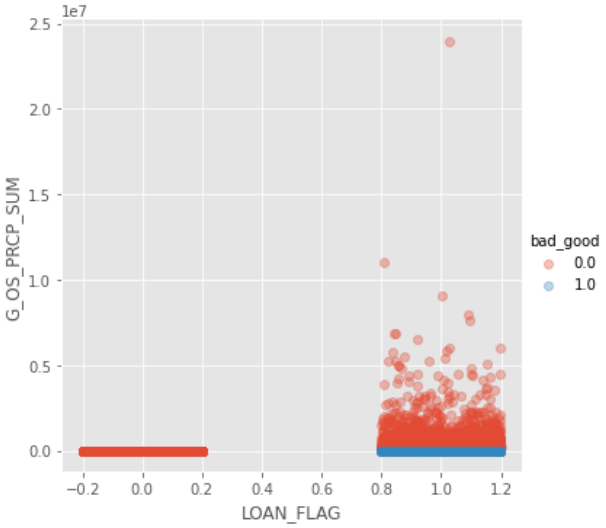
分布图与统计数据显示，样本中大部客户均未借贷。借贷客户中违约与未违约的比例相当，数据较为均衡。借贷客户的违约率高，也说明了商业银行对用户偿债能力的评估需要提升，模型建立有很强的现实意义。

（2）贷款账户月余额

单独观察借贷客户的月余额。可以看到大部分账户余额较小，余额较大的极端值很少。

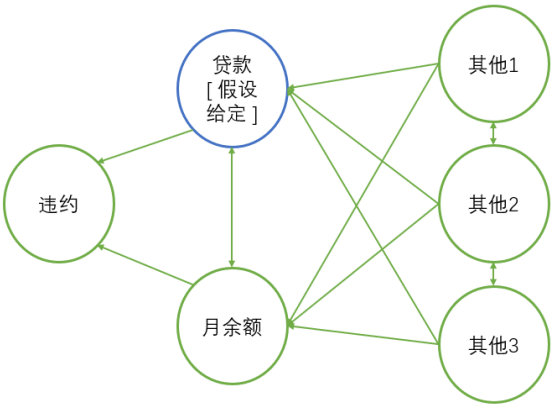


观察月余额与贷款标识的联合分布，发现借贷且账户当月余额为 0 则有违约标记，反之均无。



推测特征间关系为，其他重要特征决定客户是否贷款与当月余额。而贷款与余额直接决定客户是否违约。至此我们有两种建模思路：

1、贷款标识特征是客户决定贷款银行批准后才为真，是预测违约概率外生变量。即需要先假设已经批准贷款的基础上推断是否会违约，及对已贷款用户建立模型。



2、既然违约与其他重要因素通过贷款与月余额联系起来，将贷款标识、月余额与其他数据一起建模必然会出现多重共线性的问题。所以未知贷款标识与月余额时其他因素与违约不独立。可以直接去掉两特征，用其他因素建模。

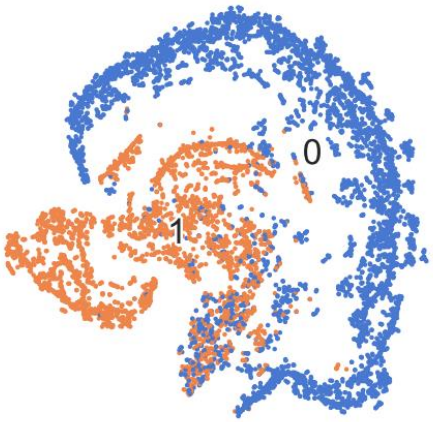
建模方法也分为二：

1、去掉贷款标识与月余额特征直接对违约标记做分类，以违约概率作为评级的依据。

2、直接用其他因素对月余额进行回归，估计在借贷条件下的月余额。银行可根据时间相关的余额波动数据对风险做出评级。

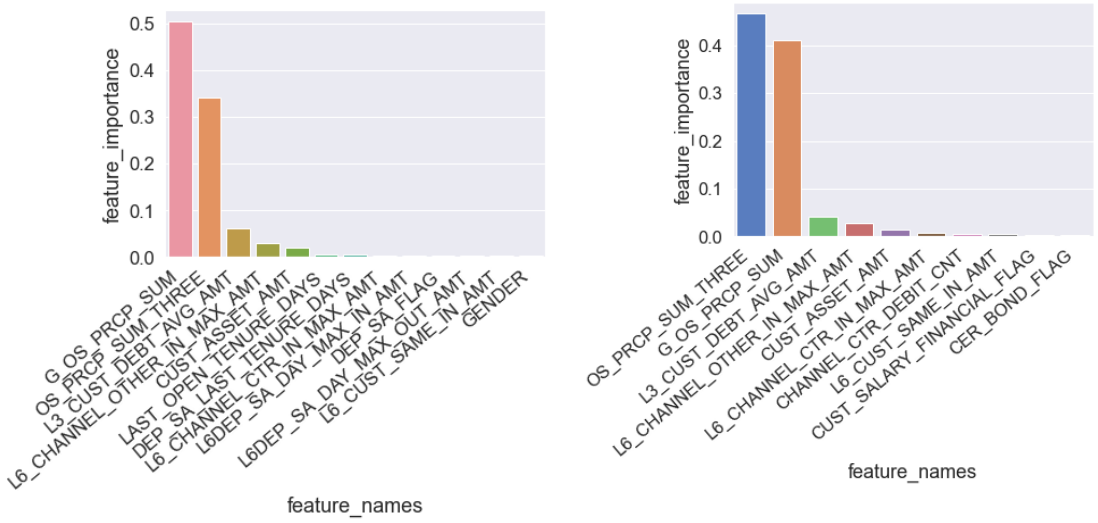
3.2.2.3 T-SNE **数据可视化**

要验证建模方法一的合理性，就要说明贷款条件下，不同违约标识的其他特征联合分布差异很大。故使用 T-SNE 对高维数据可视化。其中黄色点 1 表示违约，蓝色点 0 表示未违约。可以明显看出分布区别较大，方法一可行。



3.2.2.4 **第二次特征筛选**

为发现其他特征中的重要影响因素，我们再次使用梯度提升树筛选特征。本次我们分别从发生借贷的样本（左图）与样本总体（右图）以与月余额对月余额预测的重要性做筛选。



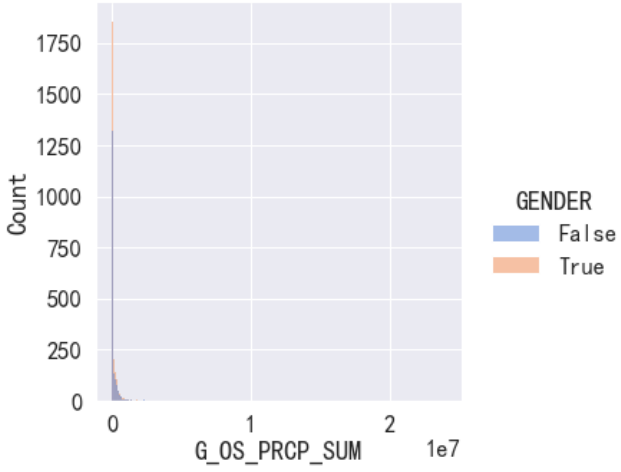
对借贷的客户：'性别', '信用卡最近开户时长', '三个月内贷款账户月均余额', '资产总额', '最近 3 个月客户月平均负债总计', '最近六个月客户跨行同名转入月平均金额', '持有活期产品标志', '活期存款最近开户距今月份', '六个月内单日本币单笔最大转入金额', '六个月内单日本币单笔最大转出金额', '柜面转入六个月内最大交易金额', '其它转入六个月内最大交易金额'。12 个特征重要性较大。

对样本总体：'三个月内贷款账户月均余额', '是否薪资理财', '资产总额', '最近 3 个月客户月平均负债总计', '最近六个月客户跨行同名转入月平均金额', '持有凭证式国债标志', '本期柜面借方交易笔数', '柜面转入六个月内最大交易金额', '其它转入六个月内最大交易金额'。9 个特征重要性较大。

两者的重要特征中，重要性较大的较为接近，略低些的有所不同。可以说明省略未借贷的样本对模型不会产生巨大影响。依据重要特征降维也可以极大提升计算效率。

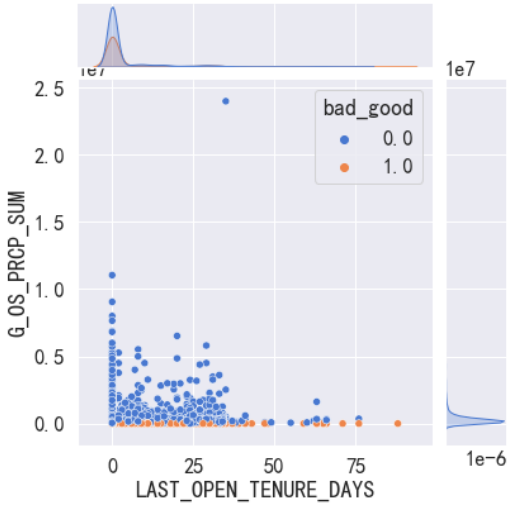
3.2.2.5 **相关特征分布**

针对以上重要特征，分别观察其与月余额的联合分布。



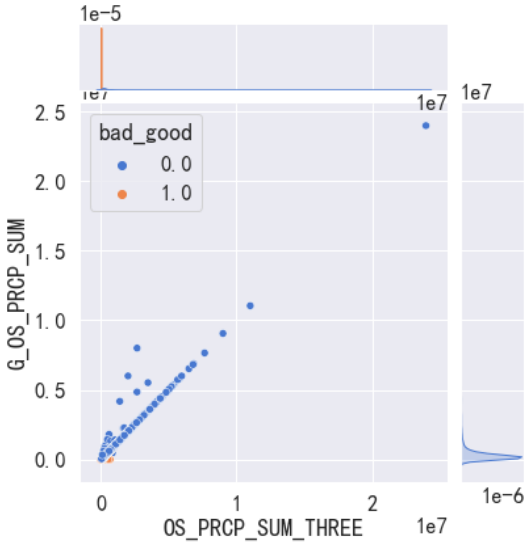
GENDER 性别

男性与女性在分布上看趋势一致，从总量上看，男性样本数在月余额大部分分布区间都高于女性。而在月余额为 0 的账户中，男性普遍高于女性。可能由于性格、财富分配等原因，女性更不容易违约。



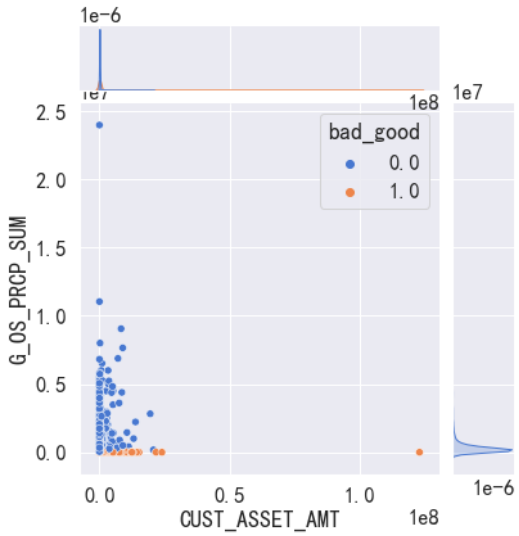
LAST_OPEN_TENURE_DAYS 信用卡最近开户时长

最近一段时间信用卡开户人数激增。相对的，违约人数也大幅度上升。而时间越近月余额平均水平越高。



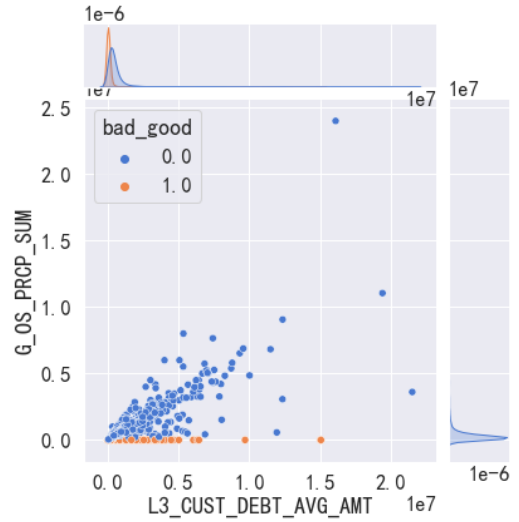
OS_PRCP_SUM_THREE 三个月内贷款账户月均余额

三个月内贷款账户月均余额和月余额呈现显著的正相关关系。这提示银行要重点关注从过去时间的余额水平估计未来短期的余额。



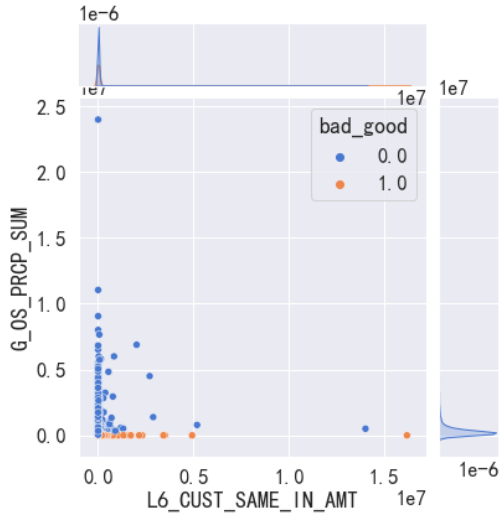
CUST_ASSET_AMT 资产总额

排除少数极端值外，资产总额总体与月余额呈现分级别的正相关性。即相同资产水平下有几个不同级别的平均月余额。在各个级别中，存在不同斜率的正相关。大部分级别下，资产与月余额呈现正相关关系，区别在于总量的大小。违约客户的月余额与资产总额仿佛完全独立。



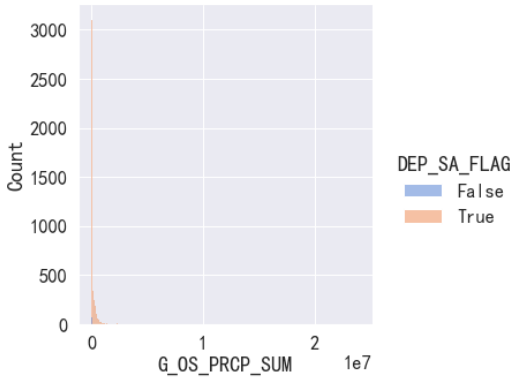
L3_CUST_DEBT_AVG_AMT 最近 3 个月客户月平均负债总计

大部分负债总量的提升都能带来月余额的提升。说明短期负债一定程度上会降低当前违约的可能性。



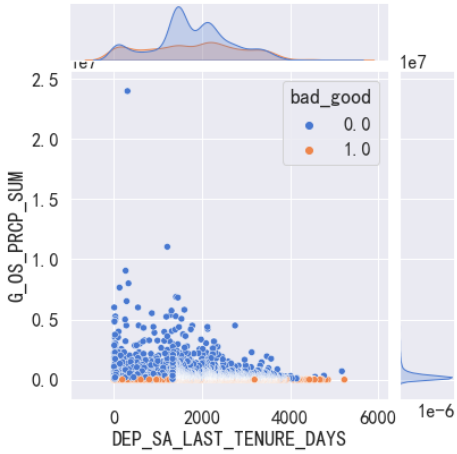
L6_CUST_SAME_IN_AMT 最近六个月客户跨行同名转入月平均金额

两者总体呈现负相关。即当客户近半年转入资金水平的提高，可能意味着正在通过调整资产分布来解决债务问题。这样的客户很可能偿债能力较低。容易出现违约问题。



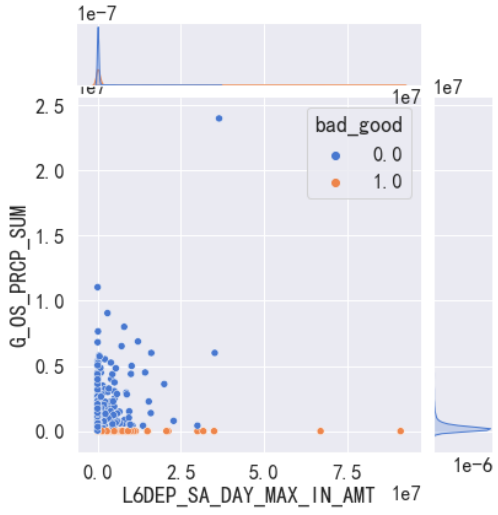
DEP_SA_FLAG 持有活期产品标志

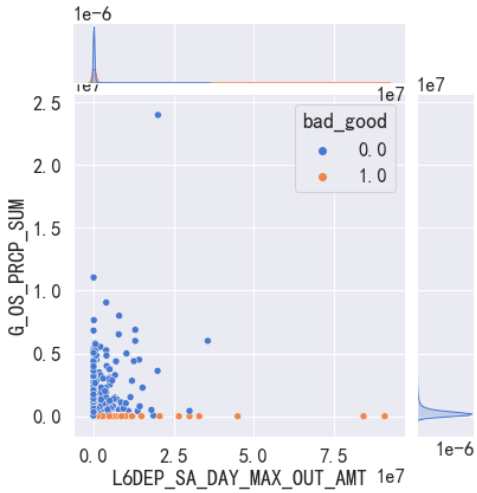
月余额越低的客户越倾向于持有活期产品。银行可以深入分析活期产品持有量做进一步推断。



DEP_SA_LAST_TENURE_DAYS 活期存款最近开户距今月份

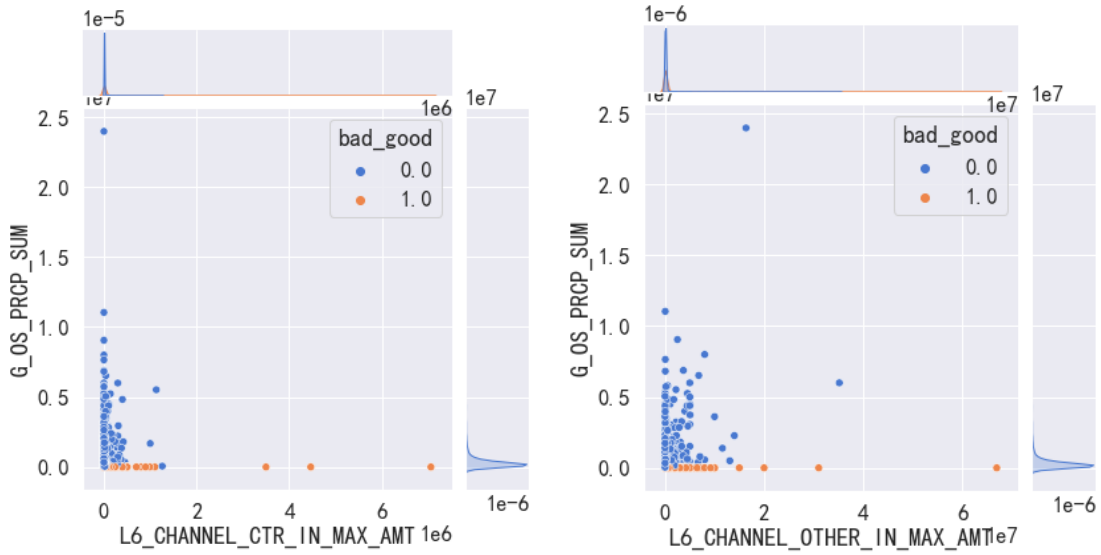
开户时间距离与月余额大体呈现负相关。观察开户时间与违约率的条件分布。可以看出违约与开户时间有周期性关系。表明经济周期的判断对违约与用户余额高度相关，需要密切关注。





L6DEP_SA_DAY_MAX_IN_AMT 六个月内单日本币单笔最大转入金额
L6DEP_SA_DAY_MAX_OUT_AMT 六个月内单日本币单笔最大转出金额

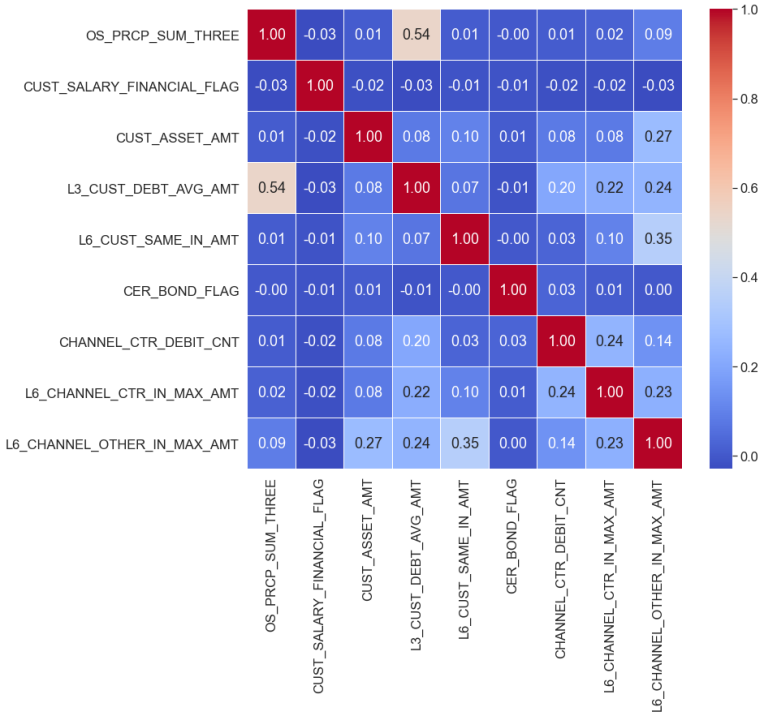
联合观察转入与转出金额，发现两者分布具有相似性。可能存在与月余额结余无关的单纯资金流动，及转入与转出。观查两者的交互相能得出更深层的推断。总体看来小规模的最大资金变动对应更高的月余额水平。



L6_CHANNEL_CTR_IN_MAX_AMT 柜面转入六个月内最大交易金额
L6_CHANNEL_OTHER_IN_MAX_AMT 其它转入六个月内最大交易金额

不同方式转入的最大金额与也具有相似性。最大的共同点在于，小规模的交易往往对应更高的月余额水平。

3.3 重要特征相关性



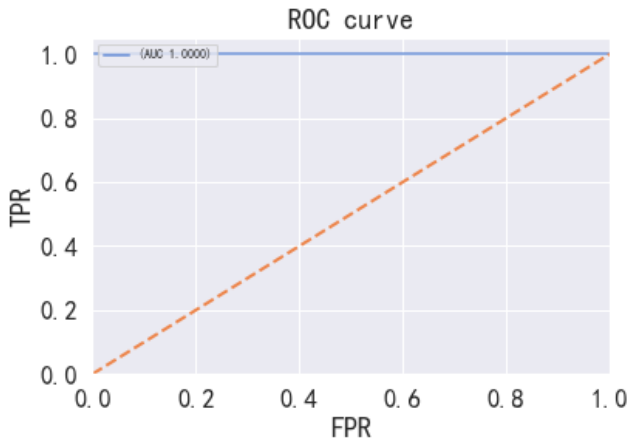
对以上特征相关系数分析，可以看出变量间的关系性较低。不存在多重共线性，可以直接选取以上特征建模。

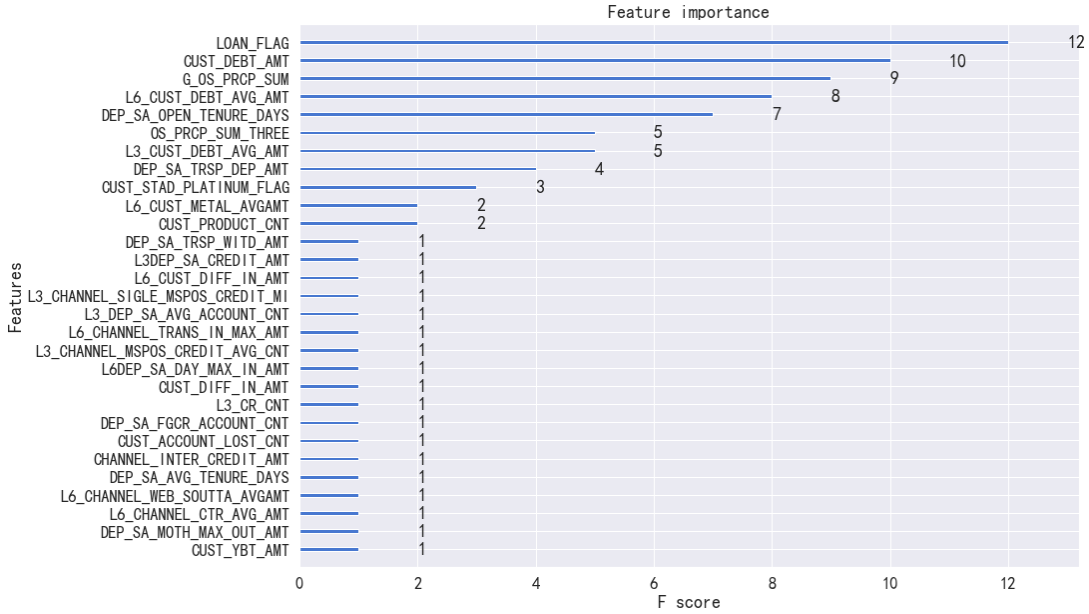
四、评价模型

4.1 XGBclassifier 直接预测

在不排除上面借贷标识与月余额特征下，直接使用树模型进行建模。可以看到 XGBclassifier 很轻松的学到了定义违约的规律，达到了近乎于完美的预测效果。但这也意味着模型的过拟合。在实际应用中必然不会提前获得当月余额的信息，所以这样的模型完全无效。

```
time cost 13.647732019424438
Confusion matrix (testing):
 [[22873     0]
 [    1   910]]

Classification report (testing):
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00     22873
         1.0       1.00      1.00      1.00       911

    accuracy                           1.00     23784
   macro avg       1.00      1.00      1.00     23784
weighted avg       1.00      1.00      1.00     23784

auc 1.0
```





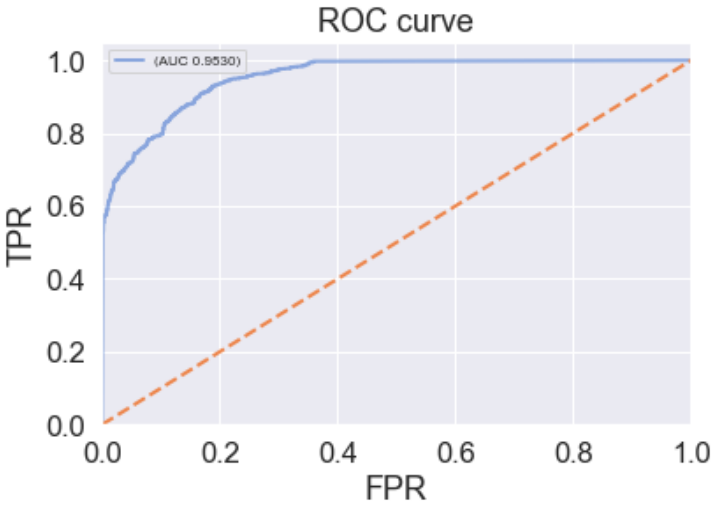
4.2 思路 1 实现——排除借贷标记与月余额分类

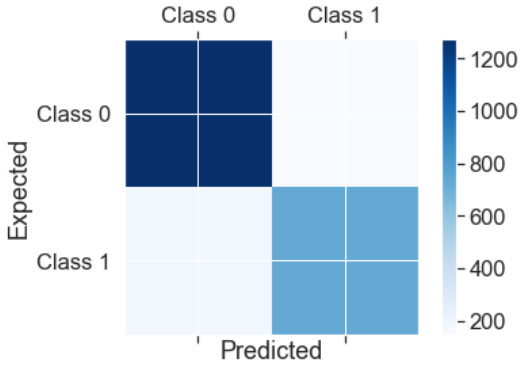
由于在前面 T-SNE 算法展现出了借贷客户是否违约的分布有极大的不同。经过试验尝试，选用发生借贷的客户为样本建立模型。由于降维后分布的差异，推测使用 knn 或是 SVM 类的算法能从几何角度将两种情况分离。为了做好比较这里选择 knn 与逻辑回归两种原理差异较大的算法实现。

4.2.1 knn

调参后，在测试集上达成的最好效果如下：

```
Confusion matrix (testing):
 [[1272  147]
 [ 174  733]]

Classification report (testing):
              precision    recall  f1-score   support

         0.0       0.88      0.90      0.89      1419
         1.0       0.83      0.81      0.82       907

    accuracy                           0.86      2326
   macro avg       0.86      0.85      0.85      2326
weighted avg       0.86      0.86      0.86      2326

auc 0.9529608020928756
```



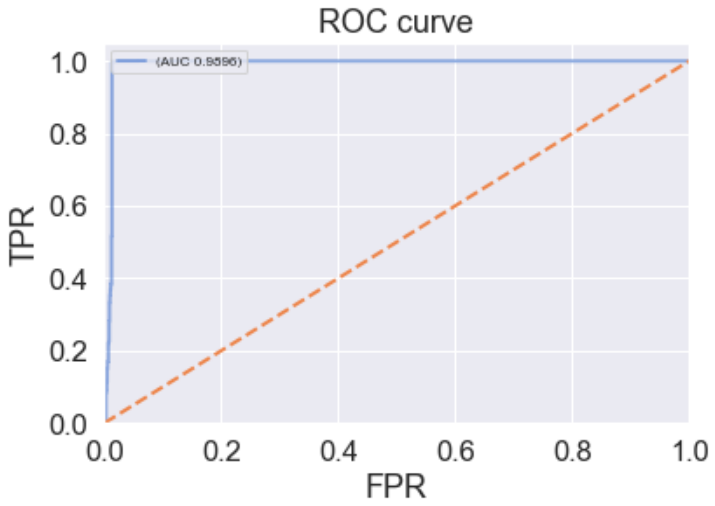


实际操作时，若用户选择贷款，银行可以用模型估计违约可能性。但若对所有客户使用此模型评估，等于假设所有客户贷款的条件下做出预测。由特征筛选的结果，选择贷款与否的条件分布存在不同之处。预测结果可能有偏。

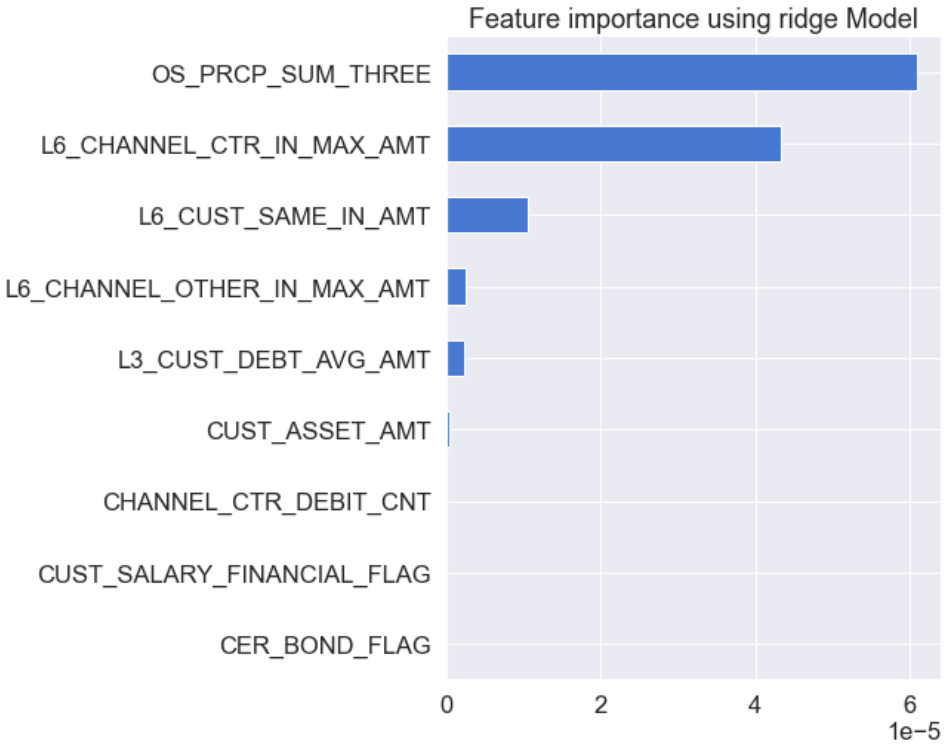
4.2.2 带 L2 正则化的岭回归

调参后，岭回归在测试集上的效果如下：

```
Confusion matrix (testing):
 [[1442   19]
 [   1  864]]

Classification report (testing):
              precision    recall  f1-score   support

         0.0       1.00      0.99      0.99      1461
         1.0       0.98      1.00      0.99       865

    accuracy                           0.99      2326
   macro avg       0.99      0.99      0.99      2326
weighted avg       0.99      0.99      0.99      2326

auc 0.9895819238545142
```

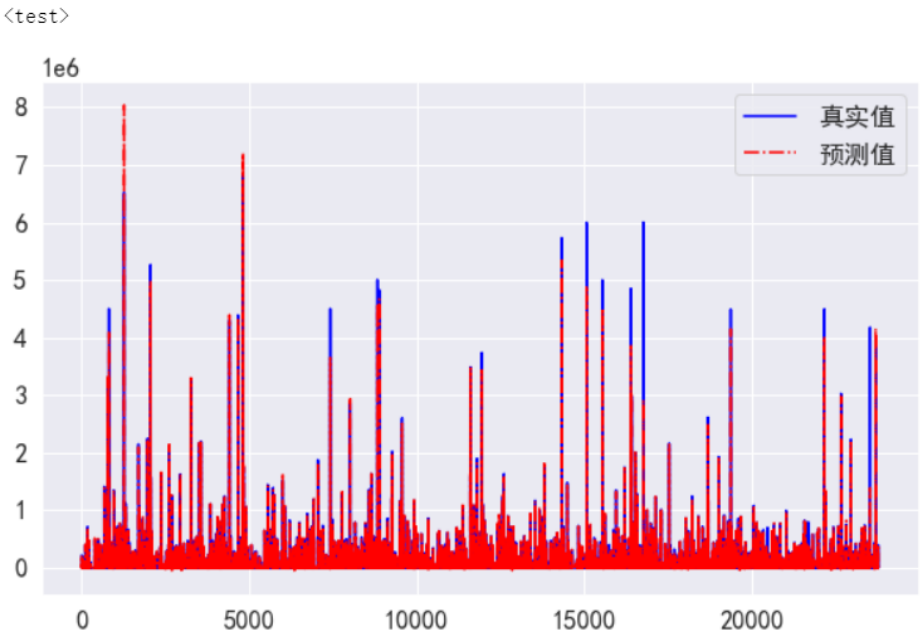


对岭回归的参数取绝对值之后，参数的大小反映对应特征对违约评级影响的重要程度。实际应用中可以以此特征顺序作为决策权重。



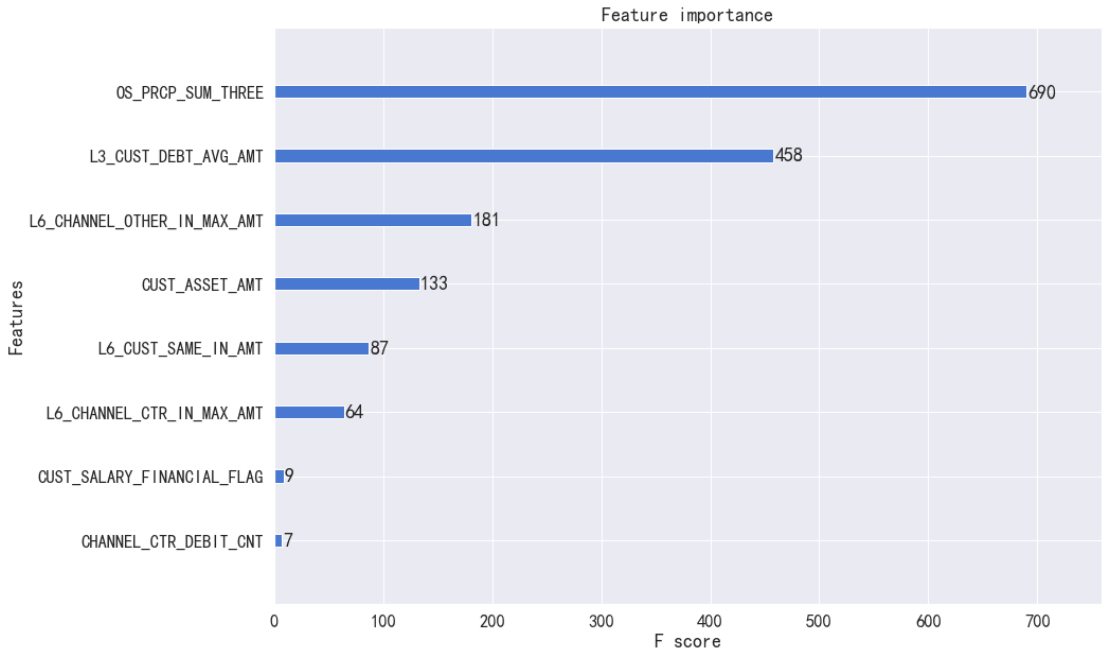
4.3 思路 2 实现

去掉违约标记，用 XGBoost 回归模型直接对月余额进行回归。
模型在测试集上的表现如下：



Boston数据线性回归模型的平均绝对误差为： 2136.366662443119
Boston数据线性回归模型的均方误差为： 1188689394.3368256
Boston数据线性回归模型的中值绝对误差为： 86.55435180664062
Boston数据线性回归模型的可解释方差值为： 0.9658464810836168
Boston数据线性回归模型的R方值为： 0.9658456411693247

XGBoost 模型也输出了预测月余额的特征重要性评估



实际应用中，银行可以使用模型提前预判客户下月余额，并对余额额度建立分级评价指标体系（这需要余额与违约的时间序列数据来进一步研究两者间的关系）。

4.4 模型结果

从最终模型结果来看，影响违约的核心因素可分为 3 类：
资产类：三个月内贷款账户月均余额，资产总额，最近 3 个月客户月平均负债总计。
资金流动类：最近六个月客户跨行同名转入月平均金额，柜面转入六个月内最大交易金额，本期柜面借方交易笔数，其它转入六个月内最大交易金额。
资产配置类：持有凭证式国债标志，是否薪资理财。
其中，资产类特征重要性最高，资金流动类次之，资产配置类相对较低。

五、总结

在平时作业阶段就采用了这个数据集进行实验，结果都是过拟合。当时一直以为此问题的核心在于稀疏矩阵与数据不均衡问题的处理。但通过机器学习建模的全流程，特别是特征工程阶段的操作后。渐渐发现问题并非想象的那么简单。

对数据的预处理与分析才是解决实际问题最重要的一步。特征工程不仅让我了解到样本的实际状况，更是启发我搭建合适的模型。虽然六百多个特征不能一一观察，但通过观察筛选出来的核心特征本身就能解决很大的问题。

当建模过程出现问题时，就应该回归到观察与分析阶段。随着研究的一步步深入，复杂问题最终会化繁为简，解决方式也会浮出水面。

建模目标的转变是最难以忘怀的一步。观察重要特征的分布，使得实验突破了原始题目描述的字面意义，取得了实质性的进展。

六、参考文献

[1]Seaborn, Sklearn, XGBoost, T-SNE 等官方文档

[2]周志华，机器学习，清华大学出版社，2016

[3]李航，统计学习方法，清华大学出版社，2012

[4]庞素琳. Logistic 回归模型在信用风险分析中的应用[J]. 数学的实践与认识, 2006, 036(009):129-137.

[5]曹颖超. 改进的 GDBT 迭代决策树分类算法及其应用[J]. 科技视界, 2017, 000(012):105-105.

[6]Laurens V D M， Hinton G . Visualizing Data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(2605):2579-2605.