北京理工大學珠海學院
BEIJING INSTITUTE OF TECHNOLOGY, ZHUHAI

基于机器学习的信用卡评分模型

队 伍 成 员：王雨聪、郭奇、钟禧儿

队 伍 名 称：你 说 的 都 队

专　　业：大数据分析 & 应用统计学

联 系 方 式：**Tel：18629537461&13543001390**

二〇二〇年十月

摘 要

改革开放以来，我国国民经济开始迅猛增长，金融机构与商业银行的信贷等传统信贷产业得到了飞速的发展。在传统的商业银行以及资信机构的贷款过程中，通常以实地调查以及资产的抵押来进行对借款人的偿还能力进行分析评估，这种方式不仅效率低，速度慢，在此情况下，如何找到一套可以提高升金融风险识别能力，增强风险评估的准确性，减少人力成本以提高风险管理效率、提高网络借贷的成功率的信用评分系统，是我国金融行业的核心工作之一。

在银行与 P2P 借贷中，评分卡是风险评估的重要工具，能够对借贷申请人进行信用评估打分，预测申请人未来的偿还表现并进行排序。本文以 2014 年德国某银行的真实数据为基础，首先进行数据初步清洗，筛选重复值，随机森林方法填补缺失值，观察数据分布并去除异常值。特征工程中，对于不平衡的数据，首先使用 SMOTE 采样方法对样本做平衡化处理，而后绘画每个特征 IV 曲线筛查有效特征，VIF 检验和皮尔逊检验筛选出共线性特征，而后对数据进行卡方分箱处理，分箱后，对数据进行 WOE 编码。

在模型选择上，我们初步使用 KNN、SVM 和 logisitic 模型分别进行拟合，并用学习曲线评价不同模型的拟合能力，最后选择 logistic 模型，而在选定模型后，我们以 ROC 曲线的面积 -AUC 的大小作为模型性能评估标准，对其调整参数，使其最优，而后根据信用分模型和评分卡模型，最终制作出评分卡。

关键词：评分卡　逻辑回归　机器学习　ROC 曲线　WOE 编码

目 录

1 项目背景及流程

评分卡是一个贷前评分体系，用于对未来一段时间内借贷申请人违约、逾期、失联概率的预测，分数越高表示该客户借贷风险越低，反之如果分数较低，表示对该客户的借贷风险较高，银行和资信机构可以拒绝该申请者的贷款业务。

在以往的传统金融学分析中，银行等贷款机构主要采用专家赋值的办法产生供业务人员使用的评分卡，往往由于信息不对等或者决策者的主观意识而造成一部分贷款申请客户难以通过贷款申请。

不过在当代，金融机构在风险管理的每个环节都尽可能地引入统计学计量分析的方法，依托大数据进行后台的分析回顾，不断的优化调整，使得金融机构在风险与收益的博弈过程中更快达到平衡。

我们将以 2014 年德国某银行的真实数据为基础，制作评分卡，评分卡制作出来后，可以对银行中的每一个客户进行打分，客户分的高低，成为衡量客户信用、银行对客户借贷成功率判断的辅助手段,以下为评分卡制作的流程，大图可于附件中查看。

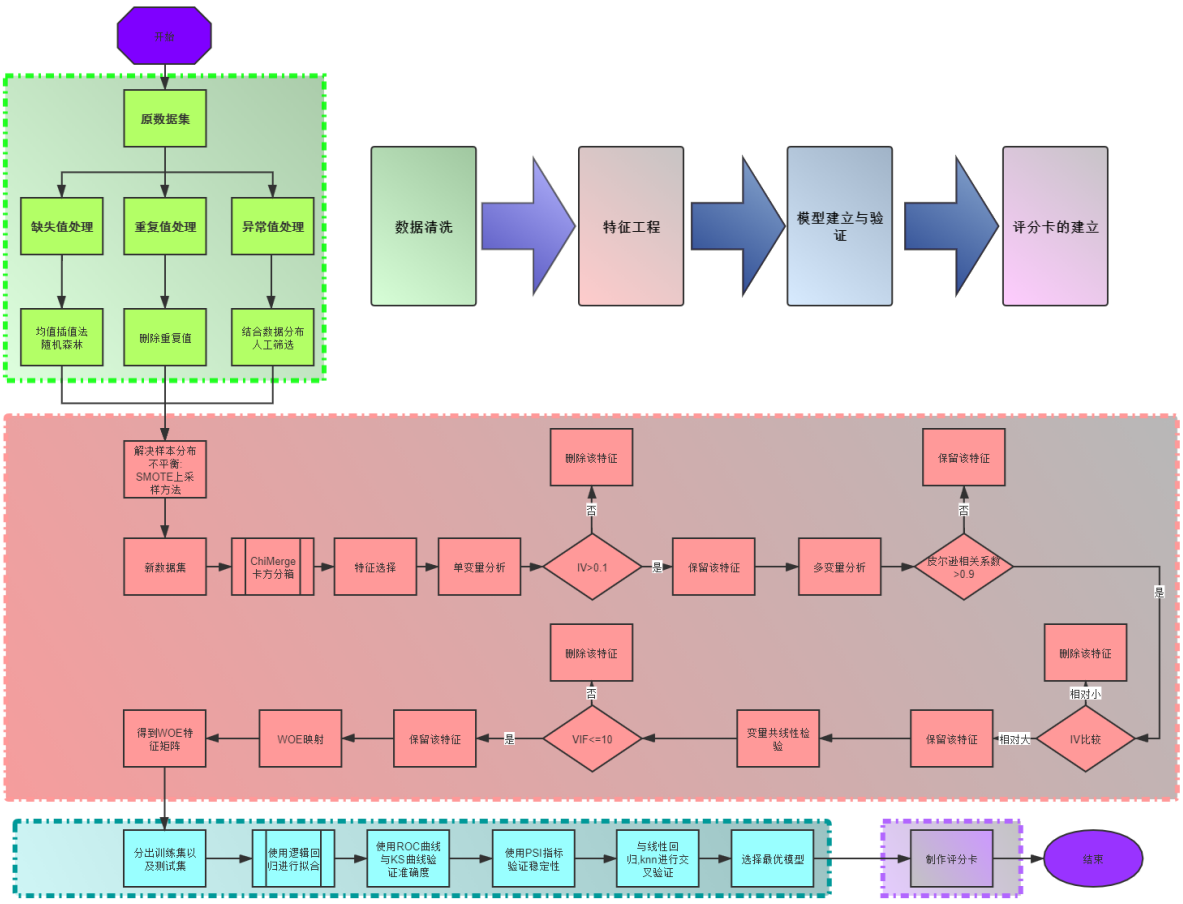


图 1

2 数据清洗

2.1 数据来源及数据特征信息

本文使用的数据集来自于竞赛网站 Kaggle，名字叫做 GivemeSomeCredit, 源自 2014 年一家德国银行的真实数据。经观察，该数据集总共 10 个特征值，1 个标签值，如图所示：

SeriousDlqin2yrs	好客户与坏客户
RevolvingUtilizationOfUnsecuredLines	无担保贷款的循环利用
age	借款人借款时的年龄
NumberOfTime30-59DaysPastDueNotWorse	35-59 天逾期但不糟糕次数
DebtRatio	负债比率
MonthlyIncome	月收入
NumberOfOpenCreditLinesAndLoans	开放式信贷和贷款数量
NumberOfTimes90DaysLate	90 天逾期次数
NumberRealEstateLoansOrLines	不动产贷款或额度数量
NumberOfTime60-89DaysPastDueNotWorse	60-89 天逾期但不糟糕次数
NumberOfDependents	家属数量

表 (1)

由于特征值较多，不便于观察与后续分析。我们依照不同属性把十个特征值进行适当的分类。这些特征经过分类，可以归为以下几个属性：

- 基本属性：包括了借款人当时的年龄。
- 偿债能力：包括了借款人的月收入、负债比率
- 信用往来：两年内 35-59 天逾期次数、两年内 60-89 天逾期次数、两年内 90 天或高于 90 天逾期的次数。
- 财产状况：包括了开放式信贷和贷款数量、不动产贷款或额度数量。
- 其他因素：包括了借款人的家属数量（不包括本人在内）。

2.2 重复值处理

2.2.1 处理前与处理后的表格

特征名	非空值统计	数据类型
SeriousDlqin2yrs	150000 non-null	int64
RevolvingUtilizationOfUnsecuredLines	150000 non-null	float64
age	150000 non-null	int64
NumberOfTime30-59DaysPastDueNotWorse	150000 non-null	int64
DebtRatio	150000 non-null	float64
MonthlyIncome	120269 non-null	float64
NumberOfOpenCreditLinesAndLoans	150000 non-null	int64
NumberOfTimes90DaysLate	150000 non-null	int64
NumberRealEstateLoansOrLines	150000 non-null	int64
NumberOfTime60-89DaysPastDueNotWorse	150000 non-null	int64
NumberOfDependents	146076 non-null	float64

表 (2) 重复值处理前

特征名	非空值统计	数据类型
SeriousDlqin2yrs	149391 non-null	int64
RevolvingUtilizationOfUnsecuredLines	149391 non-null	float64
age	149391 non-null	int64
NumberOfTime30-59DaysPastDueNotWorse	149391 non-null	int64
DebtRatio	149391 non-null	float64
MonthlyIncome	120170 non-null	float64
NumberOfOpenCreditLinesAndLoans	149391 non-null	int64
NumberOfTimes90DaysLate	149391 non-null	int64
NumberRealEstateLoansOrLines	149391 non-null	int64
NumberOfTime60-89DaysPastDueNotWorse	149391 non-null	int64
NumberOfDependents	145563 non-null	float64

表 (3) 重复值处理后

2.3 缺失值处理

对于缺失值，我们一般有两种处理方法，一种是直接删除整行的数据，一种是进行填补，从下图我们可以看出数据的缺失情况：

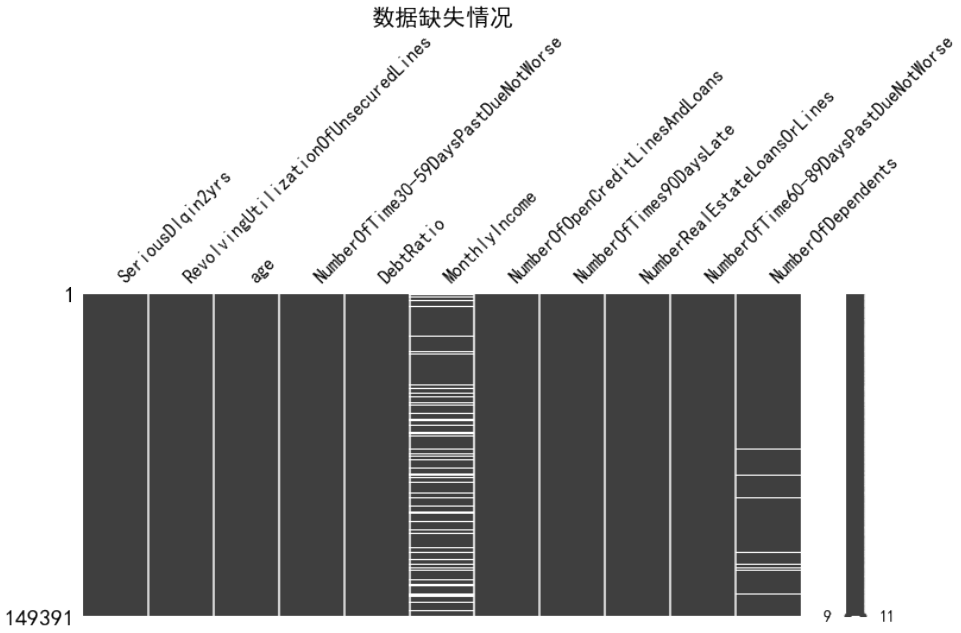


图 2

白色代表缺失，在这里我们需要填补的特征是“月收入”和“家属人数”,家庭人数的缺失不严重，只占到该特征的百分之二左右，对于“家庭人数”，我们可以直接删去，也可以填充，这里使用均值法进行填充，即取缺失数据周围数据的均值进行填充：

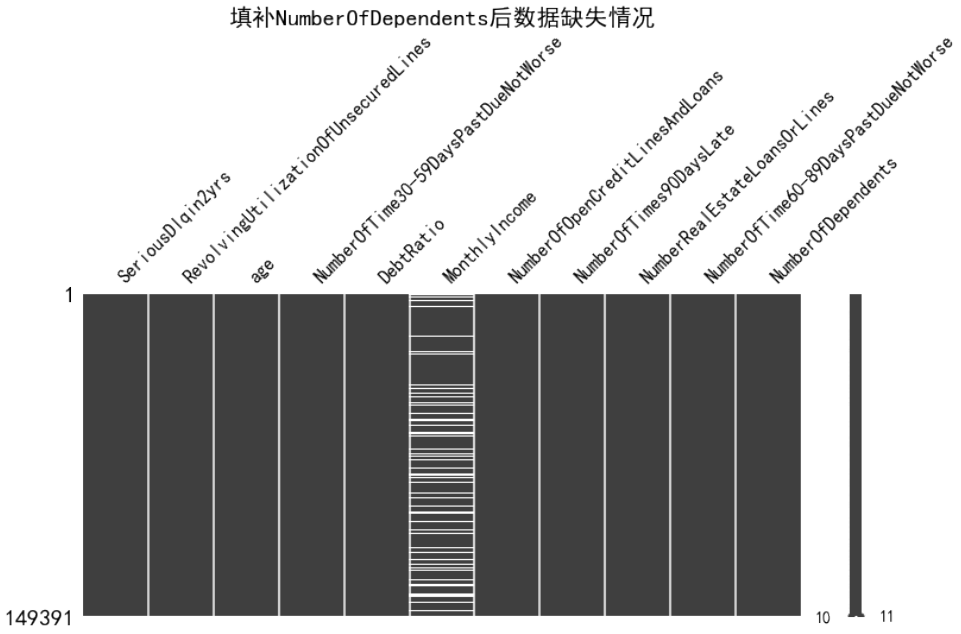


图 3

“月收入”的缺失比较严重，占到了百分之二十左右，同时，从业务的角度考虑，收入应该是对信用评分来说一个很重要的因素，因此这个特征必须要进行填补，但是均值填补法，对于缺失比例较大的“月收入”，显然是不合适的，我们可以这样去考虑这个问题，一个人来借钱，他应该知道，高收入和稳定收入对于他而言应该是一个有益的证据，因此，如果收入较高或者稳定的客户，他会更加倾向于将自己的收入填上，那么收入栏缺失的客户，更有可能是收入不稳定或者收入比较低的，根据这种判断，我们可以用四分位数来填补缺失值，将收入栏空的客户全部当成低收入人群，当然，这种方法并不严谨，也有可能这种缺失确实是数据收集过程中造成的失误，最好的方法是去和业务人员沟通，观察缺失值是如何产生的。在这里，我们采用随机森林法来进行填充。

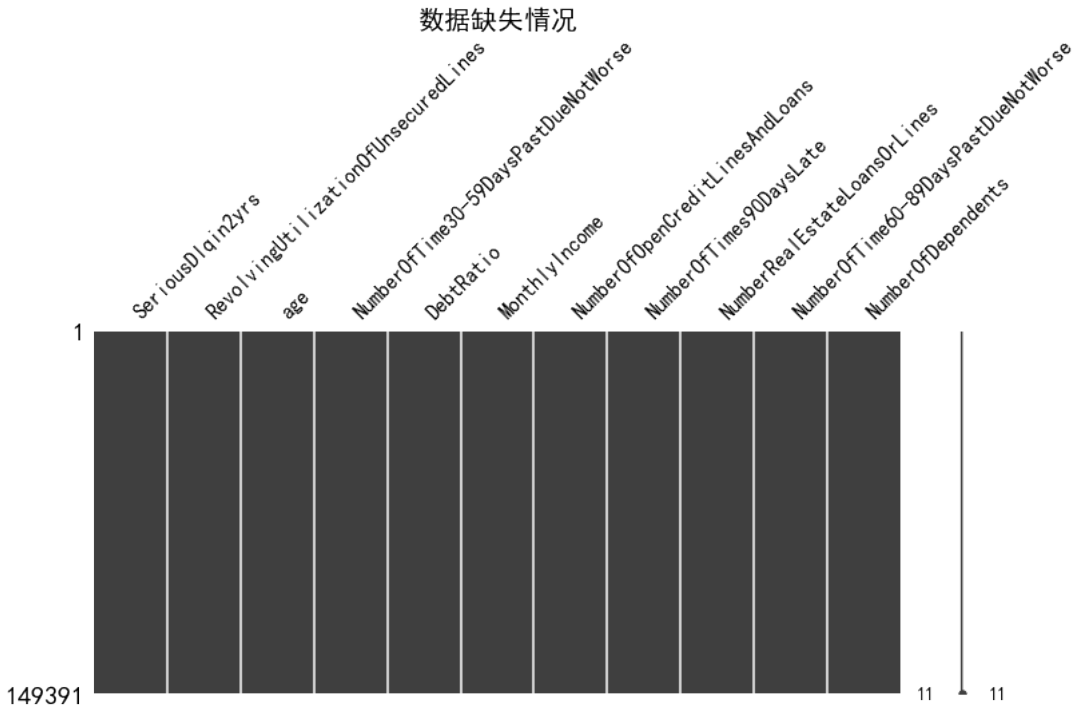


图 4

2.4 异常值处理

在传统的数据分析中，一般要求需要处理的异常值包括极大极小值，离群值，不符合业务逻辑的值，而在银行数据中，我们希望排除的一些异常值并不是一些超高或者超低的值，而是一些不符合业务逻辑，甚至不符合常理的值，在这里我们使用每个特征的核密度估计图来尝试挑出这些异常值，从中可以看出数据的呈现长尾分布，具有高方差的特性，符合我们对银行数据模式的认知

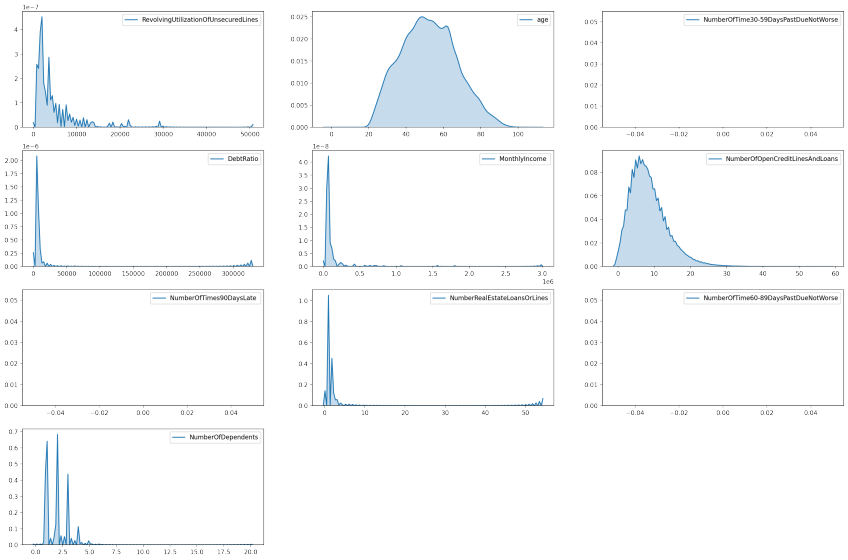


图 5

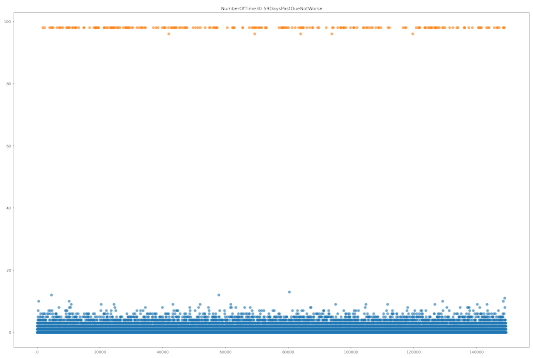
其中，最小的年龄竟然有 0 岁，这不符合常理，银行借贷的最小年龄是 8 岁，经过检查，发现只有一例样本存在这种情况，基本可以确定是异常值，删除即可，另外，有三个指标看起来很奇怪：“NumberOfTime30-59DaysPastDueNotWorse”“NumberOfTime60-89DaysPastDueNotWorse”“NumberOfTimes90DaysLate”

这三个指标分别是：“过去两年内出现 35-59 天逾期但是没有发展的更坏的次数”，“过去两年内出现 60-89 天逾期但是没有发展的更坏的次数”，“过去两年内出现 90 天逾期的次数”。

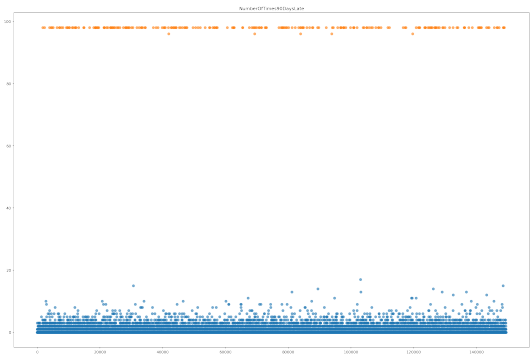
这三个指标，在 99% 的分布的时候依然是 2，最大值却是 98，看起来非常奇怪。一个人在过去两年内逾期 35 59 天 98 次，一年 6 个 60 天，两年内逾期 98 次这是怎么算出来的？我们不知道这个指标是如何计算出来的。经过统计发现，在这三个特征中，大于 90 的值全部属于同一批样本，共 225 个，所以我们先假设上面三个指标中只要每一项均大于 90 就将样本认定为异常的。

	count	mean	std	min	1%	10%	25%	50%	75%	90%	99%	max
SeriousDlqin2yrs	149391	0.066999	0.250021	0	0	0	0	0	0	0	1	1
RevolvingUtilizationOfUnsecuredLines	149391	6.071087	250.26367	0	0	0.003199	0.030132	0.154235	0.556494	0.978007	1.093922	50708
age	149391	52.306237	14.725962	0	24	33	41	52	63	72	87	109
NumberOfTime30-59DaysPastDueNotWorse	149391	0.393886	3.852953	0	0	0	0	0	0	1	4	98
DebtRatio	149391	354.43674	2041.8435	0	0	0.034991	0.177441	0.368234	0.875279	1275	4985.1	329664
MonthlyIncome	149391	5425.4629	13245.409	0	0	0.18	1800	4420	7416	10800	23250	3008750
NumberOfOpenCreditLinesAndLoans	149391	8.480892	5.136515	0	0	3	5	8	11	15	24	58
NumberOfTimes90DaysLate	149391	0.23812	3.826165	0	0	0	0	0	0	0	3	98
NumberRealEstateLoansOrLines	149391	1.022391	1.130196	0	0	0	0	1	2	2	4	54
NumberOfTime60-89DaysPastDueNotWorse	149391	0.212503	3.810523	0	0	0	0	0	0	0	2	98
NumberOfDependents	149391	0.740393	1.108272	0	0	0	0	0	1	2	4	20

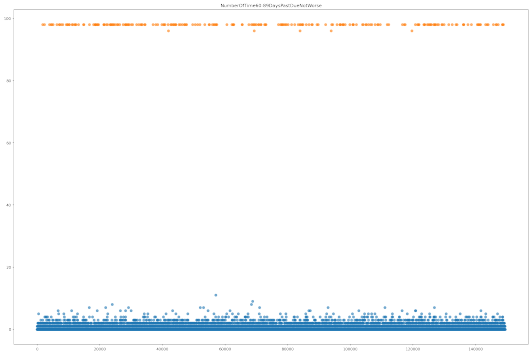
图 6: 特征的分布，大图请于附件查看



(a) NumberOfTime30-59DaysPastDueNotWorse 的分布



(b) NumberOfTimes90DaysLate 的分布



(c) NumberOfTime60-89DaysPastDueNotWorse 的分布

图 7: 三种特征的分布

那这些两年内逾期了 98 次的客户，应该都是坏客户。那么通过小提琴图检查是否是这样：

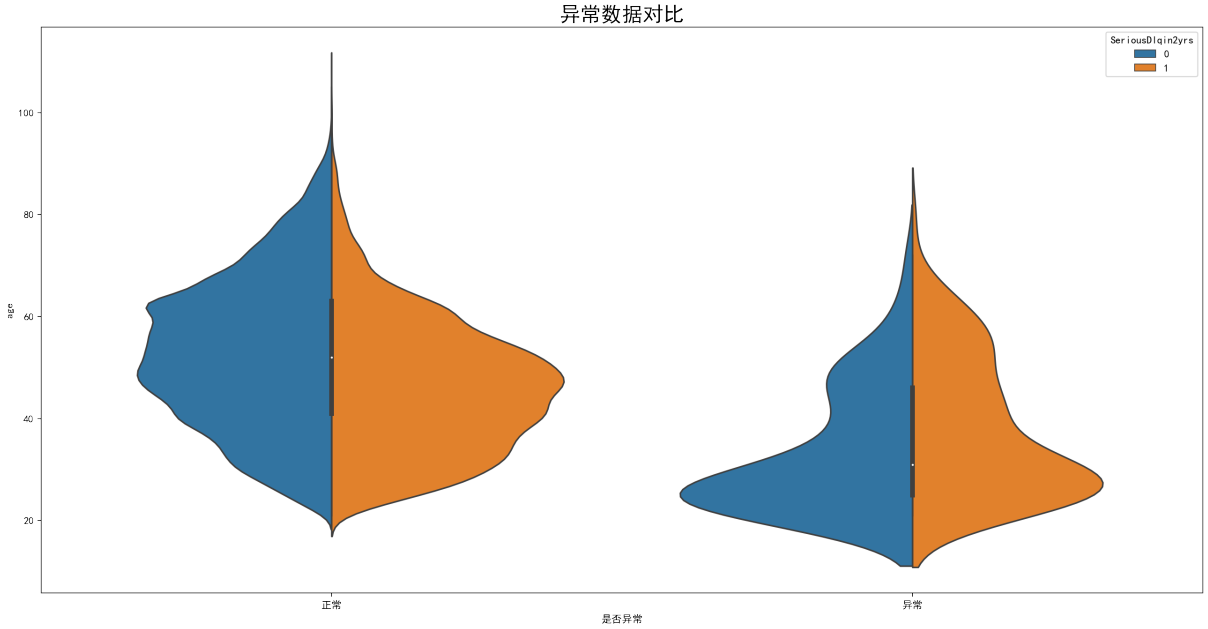


图 8

可以发现，并不是所有人都违约，但是与正常的数据相比，违约的占多数。同时我们还发现一个很有趣的现象，那就是，这部分异常的数据普遍收入很低。

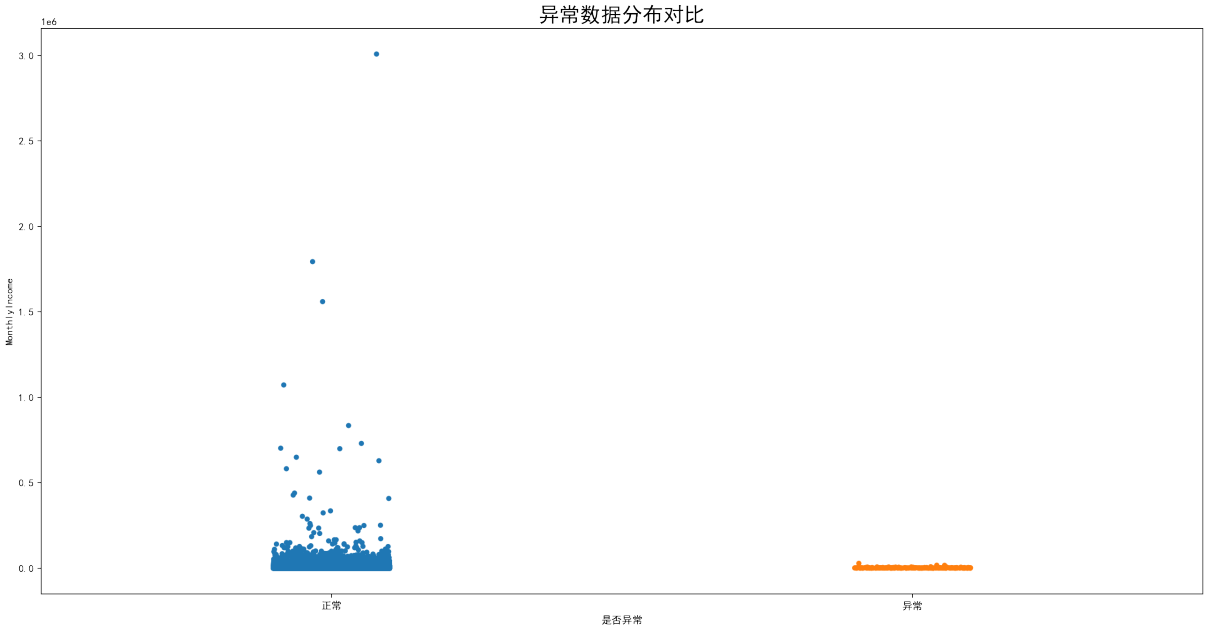


图 9

所以我们最后的结论就是，先去除掉这 225 个样本，之后若是模型的拟合度不好，再选择将其加入。

3 特征工程

3.1 样本平衡化处理

3.1.1 不平衡数据产生的原因

观察样本分布，发现严重不均衡，其原因之一，是人们皆在有意识的避免产生不良信用，故而实际违约者并不多。

从银行的角度出发，贷款给用户，用户遵守协议偿还本金以及利息，银行便可获利，故而银行真正的需求是筛选出“恶意违约“之人，故而原因之二，便是银行并不会死板的将所有逾期的用户记录为坏账，定会尝试与客户沟通、证实，只要能把银行的钱还上，银行不会记此客户为坏账。

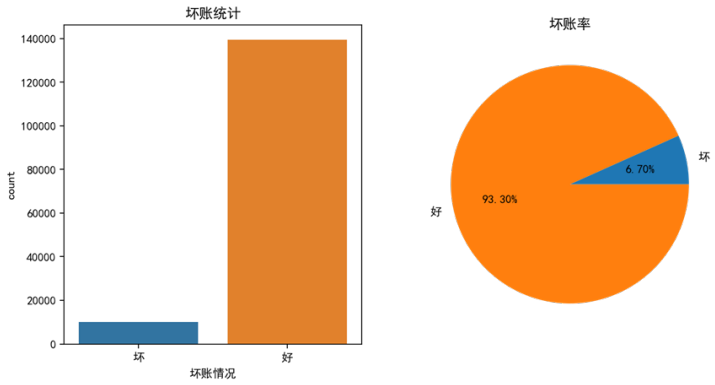


图 10

3.1.2 解决样本不平衡的方法

我们在采样方法上使用 SMOTE 算法来平衡样本，以保证之后逻辑回归的准确度。平衡之后的数据如下所示:

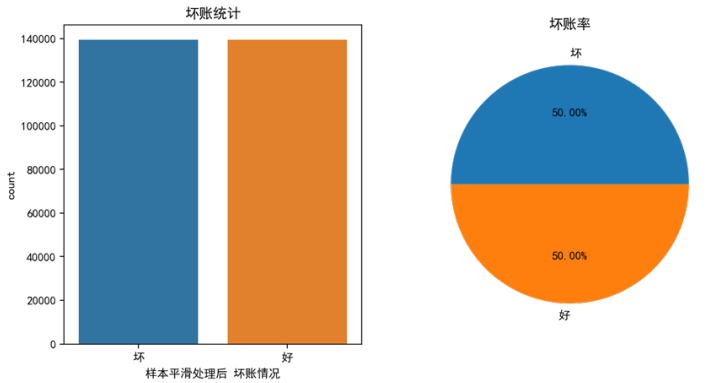


图 11

3.2 分箱

3.2.1 为什么分箱

在前面，我们已经对数据进行了重复值处理，缺失值处理，异常值处理，平衡化处理，但是制作评分卡，目的是给各个特征进行分档，以方便业务人员能够根据新客户的信息为其打分，因此需要分箱，其本质是将连续变量转化为离散变量，让组之间的差异尽可能的小，不同组之间的差异尽可能的大，使不同属性之人划分为不同类别，对应不同的分数。

此外，数据特征的分布大多很极端，存在大量的围绕均值波动的数据，同时有少量远离大部分数据存在的极端高值，但是却不能将其归类为异常值，拿月收入 MonthIncome 来说，存在一部分收入极高，脱离大部分数据分布的值，但是我们却不能判断其为异常，是因为现实中确确实实存在着收入远远高出普通人存在的超高收入人群。

为了避免这些‘异常数据’极端值产生的影响，同时又避免特征中无意义的波动，我们采用分箱的方法，来提高模型的拟合度。 **关于特征分箱，我们后面会有更详细的解释。**

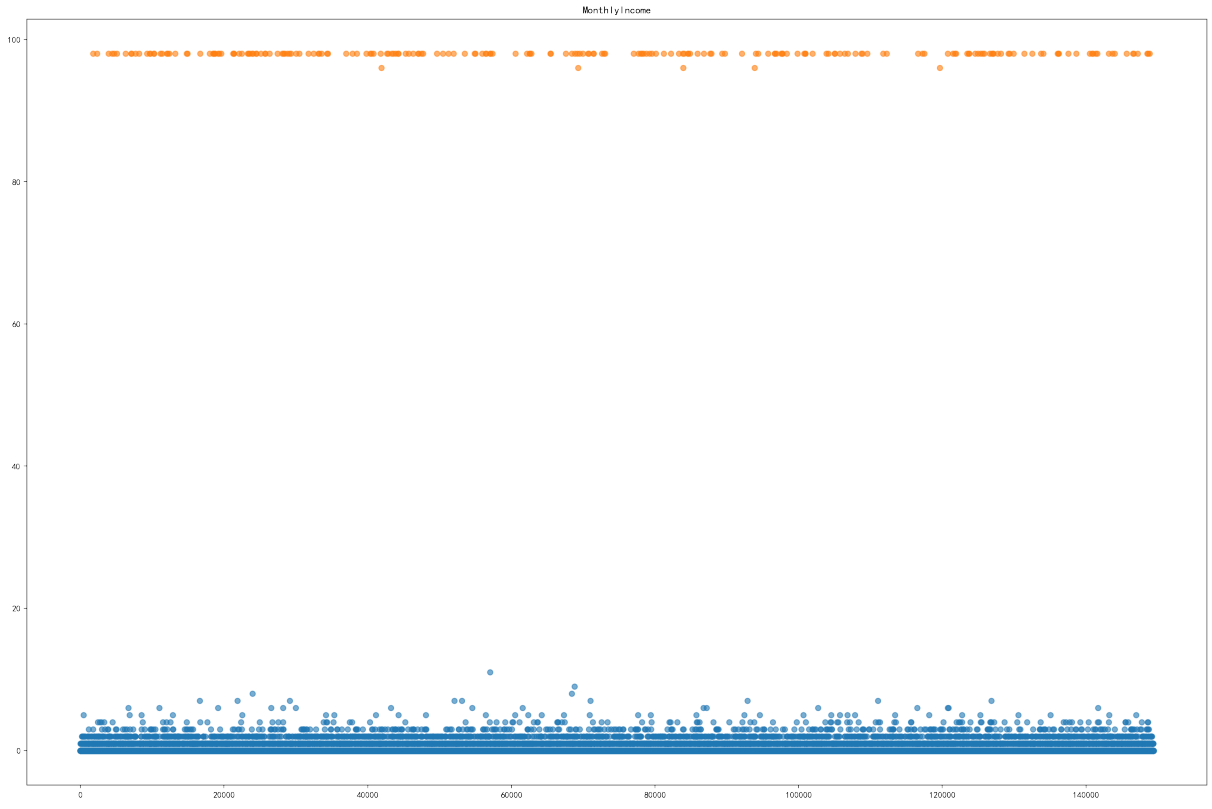


图 12

3.2.2 关于分箱的个数

改成：首先从业务的角度考虑，分箱数不能过多，其次从模型的角度考虑，分箱数过多会导致 IV 值变大，对模型的贡献过大，会压缩其他特征影响，这就意味着，数据的微小浮动导致样本属于不同的分段的可能性变大，导致模型不稳定，在这里，我们使用 Information value(IV) 来进行箱数的确定。

我们会对特征进行分箱，然后计算每个特征在 n 个箱子数目下的 WOE 值，利用 IV 值的曲线，找出每个特征合适的分箱个数。

Information value(IV)：银行业用于衡量特征上的信息量以及特征对预测函数的贡献:

$$IV = \sum_{i=1}^{N}(good\% - bad\%) * WOE_i \tag{1}$$

N 是特征上分箱的个数，i 代表每个箱子，good% 是这个箱内的非坏账率（标签为 0 者之比例，银行认为的优质客户），bad% 是这个箱子中的坏账率（违约者，标签为 1 的比例），而 WOEi 则写作：

$$WOE_i = \ln \frac{good\%}{bad\%} \tag{2}$$

银行业中用来衡量违约概率的指标，中文叫做证据权重 (weight of Evidence)，本质其实就是优质客户比上坏客户的比例的对数。WOE 越大，代表了这个箱子里的优质客户越多。而 IV 是对整个特征来说的，IV 的意义如字面（Information value）意思一般，是特征上的信息量以及此特征对模型的贡献，由下表来控制：

IV	特征对预测函数的贡献
<0.03	特征几乎不带有效信息，对模型没有贡献，这种特征可以删除
0.03 0.09	有效信息很少，对模型的贡献度低
0.1 0.29	有效信息一般，对模型的贡献度中等
0.3 0.49	有效信息较多，对模型的贡献度较高
>= 0.5	有效信息非常多，对模型的贡献超高并且可疑

表 (4)

因此，IV 并非越大越好，我们需要找到 IV 值和箱子个数的平衡点。箱子越少，则 IV 值必然越小，若存在 IV 值足够大，并且箱子个数合适的点，那便是理想情况。

3.2.3 卡方分箱

我们希望不同属性的人有不同的分数，因此我们希望在同一个箱子内的人的属性是尽量相似的，而不同箱子的人的属性是尽量不同的，即”组间差异大，组内差异小“。对于评分卡来说，就是说我们希望一个箱子内的人违约概率是类似的，而不同箱子的人的违约概率差距很大，即 WOE 差距要大，并且每个箱子中坏客户所占的比重（bad%）也要不同。那我们可以使用卡方检验来对比两个箱子之间的相似性，如果两个箱子之间卡方检验的 P 值很大，则说明他们非常相似，那我们就可以将这两个箱子合并为一个箱子。基于这样的思想，我们总结出我们对一个特征进行分箱的步骤：

- 1. 我们首先把连续型变量分成一组数量较多的分类型变量，比如，将几万个样本分成 100 组，或 50 组
- 2. 确保每一组中都要包含两种类别的样本，否则 IV 值会无法计算
- 3. 我们对相邻的组进行卡方检验，卡方检验的 P 值很大的组进行合并，直到数据中的组数小于设定的 N 箱为止
- 4. 我们让一个特征分别分成 [2,3,4.....20] 箱，观察每个分箱个数下的 IV 值如何变，找出最适合的分箱个数
- 5. 分箱完毕后，我们计算每个箱的 WOE 值，观察分箱效果

这些步骤都完成后，我们可以对各个特征都进行分箱，然后观察每个特征的 IV 值，以此来挑选特征。

接下来，在构造完算法和定义完 WOE 和 IV 函数后（具体算法请看附录），我们以特征"age" 和“NumberOfDependents”为例，展示分箱如何完成。

首先，利用卡方分布合并箱体，并画出 IV 曲线，选择转折点处，使得 IV 足够大而箱数足够合适，所以这里对于 age 来说选择箱数为 6。而后为其余特征用相同的方法。

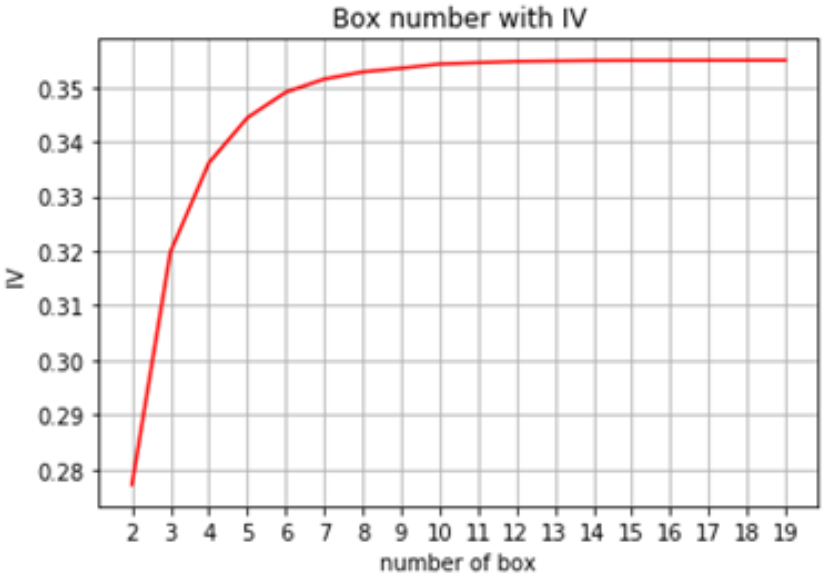


图 13

前文提到并非所有的特征都能使用卡方分箱，一些类别变量，诸如“家人”就无法分出 20 组的箱数，这个时候我们只能采用等频分箱的方法，手动得到分箱区间。如特征“”我们通过观察他的分位点分布手动分出的区间为：[0,2,7]。

```
count    195008.000000
mean          0.224611
std           0.712220
min           0.000000
0%            0.000000
15%           0.000000
25%           0.000000
35%           0.000000
45%           0.000000
50%           0.000000
55%           0.000000
65%           0.000000
75%           0.000000
85%           0.000000
95%           2.000000
100%         17.000000
max          17.000000
Name: NumberOfTimes90DaysLate, dtype: float64
```

(a) 特征分布

(b) 类别变量无法使用卡方分箱

3.2.4 WOE 编码

前者，我们基于对业务又或对模型的需求进行了分箱，但是模型是没法理解我们分箱之后划分的区间，我们要把他们编码，使得机器可以理解。

WOE 编码是一种有监督的编码方式，将预测类别的集中度的属性作为编码的数值，所以将特征的值规范到相近的尺度上便是它的优势，但对于我们的模型而言，需要分箱后每箱都同时有好坏样本，我们当然也在算法考虑并实现了这一点

$$WOE_i = \ln \frac{good\%}{bad_{rate}\%}^{1} \tag{3}$$

计算各箱的 WOE 后，我们以此为依据对数据进行 WOE 映射，并且覆盖原数据，也就是说，我们接下来将用 WOE 覆盖后的数据进行建模，这样一来我们便获取到各个箱的分类结果，即评分卡上各个评分项目的分类结果，同样的操作我们也会在测试集上进行。

图 2 为 WOE 映射后的训练集图 3 为测试集。

[1]good% 为不违约人占总体样本的比率。

[1]bad_{rate}% 为违约人数占总体样本的比例。

	age	RevolvingUtilizationOfUnsecuredLines	DebtRatio	MonthlyIncome	NumberOfOpenCreditLinesAndLoans	NumberOfTime30-59DaysPastDueNotWor
0	-0.227211	2.212034	0.072859	-0.286434	-0.055384	0.3537
1	0.768111	0.667137	0.072859	-0.286434	-0.055384	0.3537
2	-0.457050	-2.049223	-0.313270	-0.286434	-0.055384	-0.8749
3	1.146007	2.212034	-0.313270	-0.286434	0.123403	0.3537
4	-0.227211	-1.076748	-0.313270	0.354919	0.123403	0.3537
...	...	...	...	...	...	
195003	-0.457050	-1.076748	-0.313270	0.067098	0.123403	-1.3792
195004	-0.227211	-1.076748	-0.313270	-0.286434	0.123403	-0.8749
195005	-0.227211	-1.076748	-0.313270	0.354919	0.123403	0.3537
195006	0.768111	-0.464678	-0.313270	0.354919	0.123403	0.3537
195007	-0.227211	-1.076748	0.176437	0.076199	0.123403	0.3537

195008 rows × 8 columns

图 14: 训练集：195008 行

	RevolvingUtilizationOfUnsecuredLines	age	DebtRatio	MonthlyIncome	NumberOfOpenCreditLinesAndLoans	NumberOfTime30-59DaysPastDueNotWors
0	2.212034	0.258529	1.507088	-0.286434	-0.055384	0.35376
1	-1.076748	-0.227211	0.072859	0.354919	0.123403	0.35376
2	2.212034	0.768111	0.072859	0.067098	-0.055384	0.35376
3	2.212034	-0.227211	0.072859	-0.286434	0.123403	0.35376
4	-1.076748	-0.227211	-0.313270	-0.286434	0.123403	0.35376
...	...	...	...	...	...	
83571	-2.049223	-0.227211	0.176437	-0.286434	-0.845052	0.35376
83572	-1.076748	-0.457050	-0.313270	-0.286434	0.123403	-1.37923
83573	-1.076748	-0.457050	0.072859	0.067098	-0.328736	0.35376
83574	-1.076748	-0.457050	0.072859	-0.286434	-0.328736	-0.87496
83575	-1.076748	-0.227211	-0.313270	-0.286434	0.123403	-1.54184

83576 rows × 8 columns

图 15: 测试集：83576 行

3.3 特征选择

3.3.1 单变量分析

依次删除后发现，只有当删除 IV 值最小的特征 (IV=0.04)“”时，MI 和 DR 的 IV 上升，故决定删去“”，并根据新的 IV 重新确定分箱，至此，特征变为 9 列。

	RevolvingUtilizationOfUnsecuredLines	age	NumberOfTime30-59DaysPastDueNotWorse	DebtRatio	MonthlyIncome	NumberOfOpenCreditLinesAndLoans
0	0.015404	53	0	0.121802	4728.000000	5
1	0.168311	63	0	0.141964	1119.000000	5
2	1.063570	39	1	0.417663	3500.000000	5
3	0.088684	73	0	0.522822	5301.000000	11
4	0.622999	53	0	0.423650	13000.000000	9
...	...	...	...	...	...	...
195003	0.916269	32	2	0.548132	6000.000000	10
195004	0.484728	50	1	0.370603	5258.000000	12
195005	0.850447	46	0	0.562610	8000.000000	9
195006	1.000000	64	0	0.364694	10309.000000	7
195007	0.512881	53	0	1968.401488	0.134483	12

195008 rows × 9 columns

图 16

3.3.2 皮尔逊检验 -热力图

可见特征 NumberRealEstateLoansOrLines 和 NumberOfOpenCreditLinesAndLoans 之间的相关系数接近 0.5。

对于广义线性模型而言，若特征间存在近似的共线性，会导致拟合参数 β 估计精度很低，参数估计量经济含义不合理，变量的显著性检验失去意义，可能将重要的解释变量排除在模型之外，模型的预测功能失效。变大的方差容易使区间预测的“区间”变大，使预测失去意义。

因此删除两者中对模型贡献最小的特征 NumberOfOpenCreditLinesAndLoans，模型由 9 个特征变为 8 个。

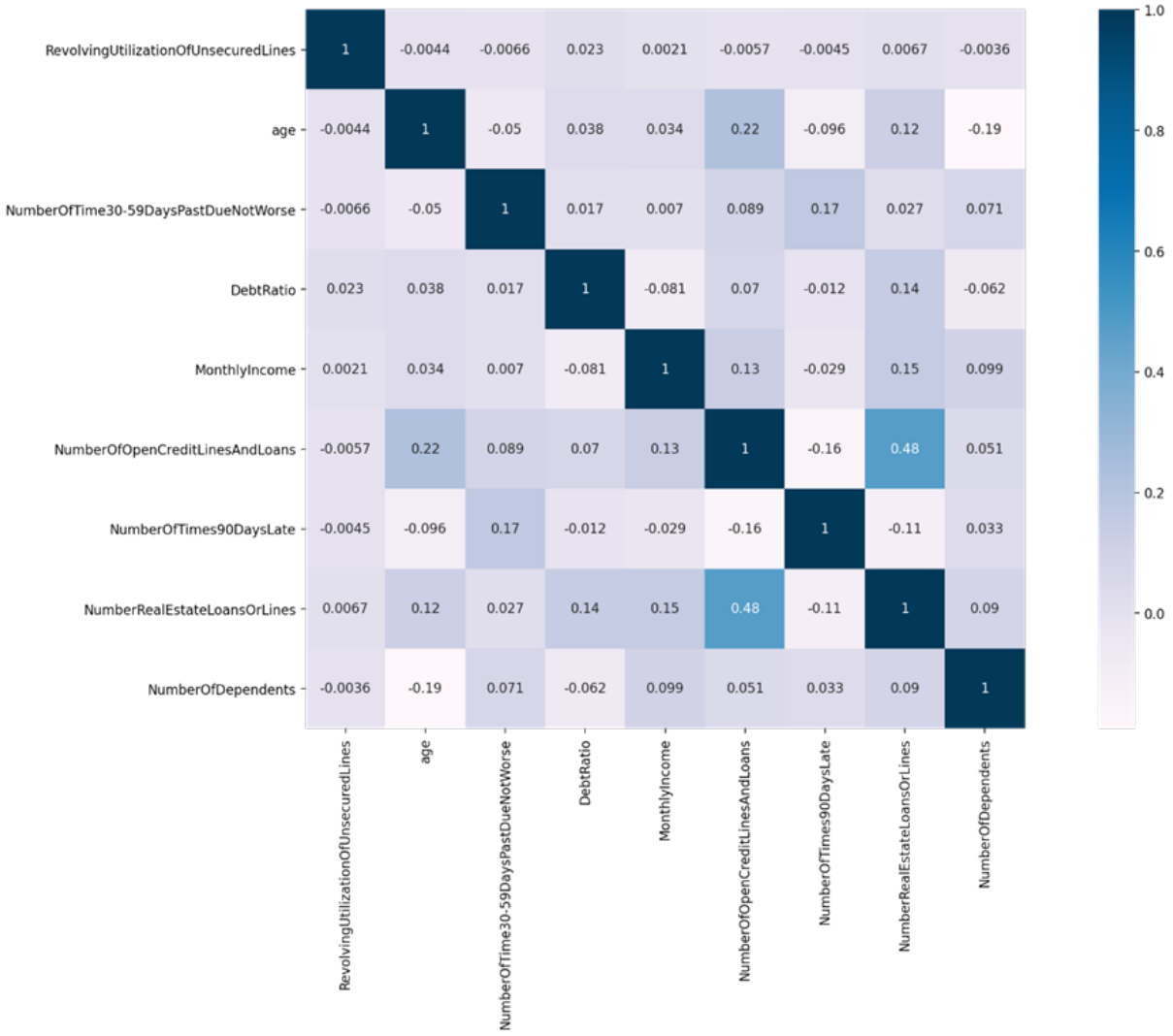


图 17: 热力图

3.3.3 多重共线性 VIF 检验

对数据的 8 个特征的进行 VIF 检验，发现其 VIF 皆小于 2.5，排除这些特征存在多重共线性的可能

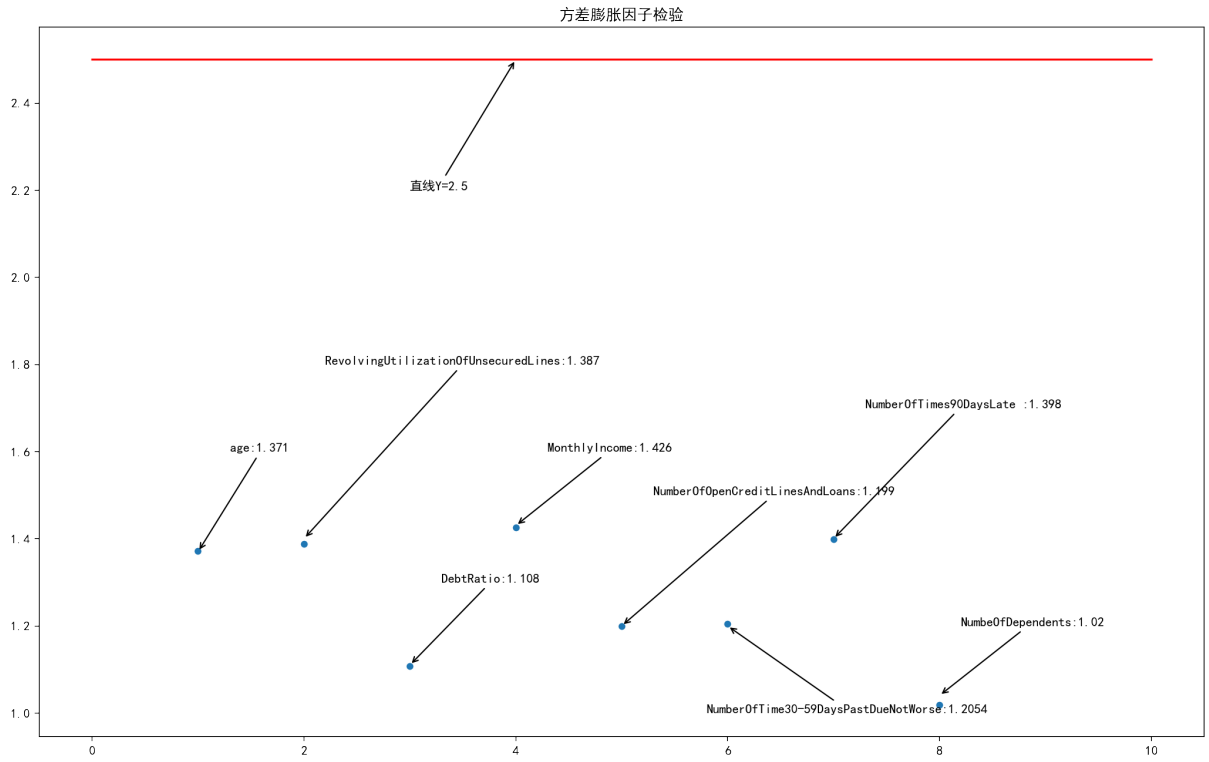


图 18: 各特征 VIF 检验

3.3.4 特征交互性检验

观察交互图，发现产生交互的特征大多为（NumberOfTime60-89DaysPastDueNotWorse 与 NumberOfTime30-59DaysPastDueNotWorse 与 NumberOfTimes90DaysLate）又或（age 与 numberofindepende）认为这一类的特征交互对模型没有价值。

亦增加过特征如：‘月收入的平方除以年龄’意在分辨出年龄高而收入低者，但模型拟合度不尽人意。(交互图尺寸过于庞大，请于附录中查看)

处理完以后，再次映射各箱的 WOE 值，最终成为我们建模数据

4 模型的建立

4.1 信用卡与评分卡建立

4.1.1 信用分模型

目前有 n 个样本，每个样本表示一个贷款客户的信息。则第 i 个样本表示为 $\{x_i.y_i\}$，其中 $y \in \{0,1\}$ 0 代表正常，1 代表逾期写成矩阵的形式：

$$X = \begin{Bmatrix} 1 & x_{1,1} & x_{1,2} & ... & x_{1,r} \\ 1 & x_{2,1} & x_{2,2} & ... & x_{2,r} \\ ... & ... & ... & ... & ... \\ 1 & x_{n,2} & x_{n,3} & ... & x_{n,r} \end{Bmatrix}$$

$$Y = (y_1, y_2, ..., y_n)^T$$

对客户 i 而言，如果其逾期的概率是 p_i，那么其正常的概率就是 $1-p_i$，一个客户要么逾期要么正常，逾期和正常的概率之和必然是 1。因为需要根据 X 评估用户 i 是否会逾期。所以可以考虑使用最常见的广义可加模型，其中线性模型最为简单。如下所示：：

$$z_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + ... + \beta_r x_{i,r} \tag{4}$$

另外，又考虑到概率 p_i 的值域是 $[0,1]$ 区间，希望输出的函数值在这个值域。因此再嵌套一层函数以变换值域：

$$P_i = \frac{e^{z_i}}{1+e^{z_i}} = \frac{1}{1+e^{-z_i}} \tag{5}$$

显然，上式的值域一定是在 [0,1] 区间的，这里使用 logsitic 函数是因为 e^{z_i} 在微分求导上更便利。样本 x_i 对应的借款人，可能是逾期，也可能是正常，可以用一个公式同时表示这两种情况：

$$\begin{aligned} P_i &= P(y_i|x_i\beta_i) = (p_i)^{y_i}(1-p_i)^{(1-y_i)} \\ y_i &\in \{0,1\} \end{aligned} \tag{6}$$

对 n 个样本而言，样本之间是相互独立的，因此对所有样本而言，其全体概率是:$L(\beta) = \prod_{i=0}^{n} P_i$　　那么我们就需要对其中的参数 β_j 进行估计，也就是让 $L(\beta)$ 的值最大，这里采用的是极大似然估计,这样就可以得到损失函数：

$$min{\cdot}ln^{L(\beta)} = -\frac{1}{m}\sum(y_i\frac{1}{p_i} + (1-y_i)\frac{1}{(1-p_i)}) \tag{7}$$

得到损失函数以后，我们再使用梯度下降法求得近似最优解。

4.1.2 评分卡模型建立

评分卡的分值分配，由比率决定。例如，一个评分卡可以设定，评分每降低 20 分，违约的比率升高一倍，600 分账户的违约比率是 620 分账户的两倍。每个得分对应一个特定的违约比率，便于控制预期违约账户。

$$Odds = \frac{p_i}{1-p_i} \tag{8}$$

$Odds$(几率): 不违约概率与违约概率的比值公式 $odds$ 做个变形，也就有：

$$p_i = \frac{Odds}{1+Odds} \tag{9}$$

评分卡的信用分计算：

$$Score = A - B\ln(Odds) \tag{10}$$

其中，A 和 B 是常数。如果违约率 p 很小，那么 Odds 是一个正的小数，比如 0.01、0.02，表示逾期概率很低，此时 $ln(Odds)$ 是大的负数，$Score$ 的分数高。反之可知 $Score$ 的分数低。

我们使用 PDO（指定的违约概率翻倍的分数）方法来计算 A 和 B:

- 1. 设定某个比率为 θ_0 的对应的分值是 p_0，然后，比率为 $2\theta_0$ 的点的分值是 $p_0 + PDO$。
- 2. 将上述两个设定带入公式，有如下两个等式

$$\begin{aligned} p_0 &= A - B\ln(\theta) \\ p_0 + PDO &= A - B\ln(2\theta) \end{aligned} \tag{11}$$

两个等式联合求解，可得：

$$
\begin{aligned}
B &= \frac{PDO}{\ln(2)} \\
A &= p_0 + B\ln(\theta_0)
\end{aligned} \tag{12}
$$

计算示例，违约比 $Odds = \frac{1}{60}$ 的时候，是合理的违约比，此时对应的分值是 $p_0 = 600$ 分，违约比每翻倍，分数变小 20 分，也就是 PDO=20。那么，根据上述公式可以求出 $A = 481.86$，$B = 28.85$，也就是 $Score = 481.89 - 28.85ln(Odds)$。如果 $Odds = \frac{1}{30}$，增加一倍，则 $Score$ 经过计算可得是 580.01，四舍五入后是 580，

信用分分值的分布，由 A、B 和“理论”的 Odds 三者决定。“实际”的 Odds 的分布由 $\beta_j(j = 0, 1, ...r)$ 决定。因此，对 β_j 的计算，跟 A 和 B 的计算无关。

4.2 分类模型的选择

对于分类问题，在机器学习领域中已经有许多成熟的算法可以用来解决。但是对于一个未知的样本集，是没有办法提前确定一套最好的方案的，目前采用最广泛的方案是使用交叉验证集法来进行模型的选择。

在这里，我们横向比较了目前比较流行的三种分类模型的分类能力，这三个模型分别为 Knn（K-NearestNeighbor，最近邻算法）、SVM（Support Vector Machine，支持向量机），LR（logstic Regression，逻辑斯蒂回归），受限于机器的性能，我们只选取了 2% 的样本（即 2000 条数据）进行检测。

4.2.1 KNN 模型

在此数据集中，我们检测了三个 KNN 模型的准确度，从中挑选了准确度最高的一种，加入到交叉验证模型组中，这三组分别为：

-普通 KNN

近邻样本数：10

准确率：0.7049999999999998

-带权重的 KNN

近邻样本数：10, 权重模式：“distance”

准确率：0.721

-指定半径的 KNN

近邻样本数：10, 半径为 500

准确率：0.48999999999999994

4.2.2 SVM 模型

在这里，我们使用了高斯核函数，并且使用 GridSearch 参数最优化确定参数 gamma: **准确度：0.7785**

4.2.3 Logsitic 模型

在这里我们使用 L2 正则化的逻辑斯蒂模型:

准确率：0.78

4.3 三种模型的学习曲线比较

4.3.1 Logsitic 回归

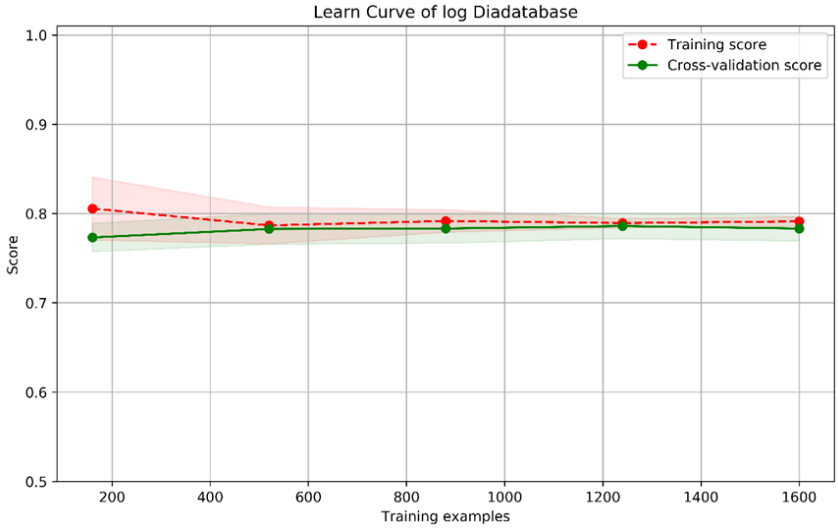


图 19

4.3.2 SVM

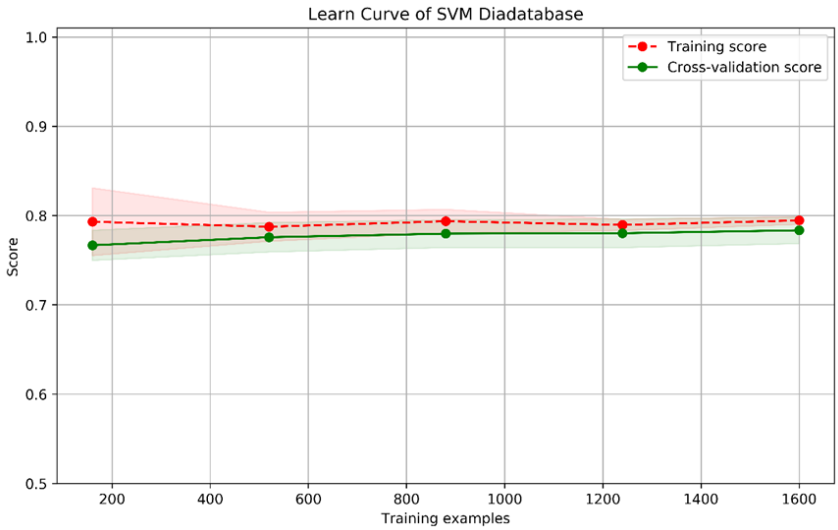


图 20

4.3.3 带权重 KNN

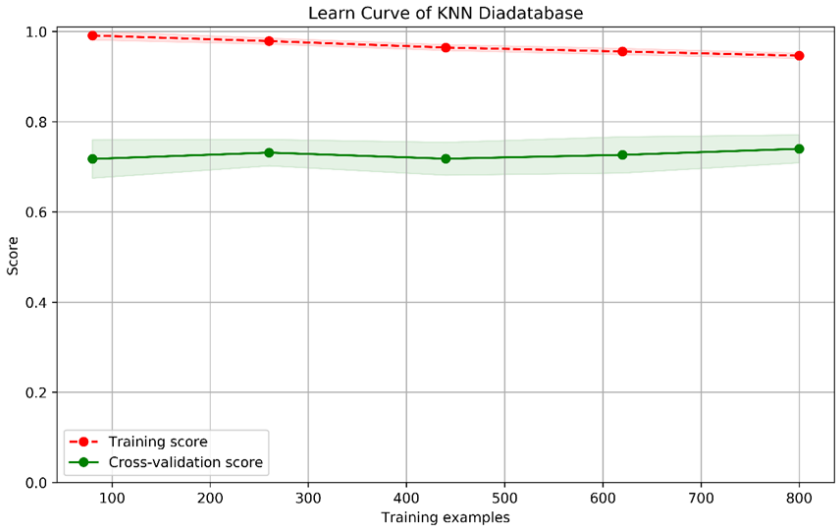


图 21

4.4 Logsitic 模型

4.4.1 参数调整

对于逻辑斯蒂回归模型，主要有两个参数可以优化，一个是正则化参数 C，另一个是迭代轮数，前者属于超参数调整，因为不同的正则化参数，对于损失函数的惩罚力度是不同的，而由于在这里的逻辑斯蒂回归使用的是梯度下降法寻找损失函数最小值的策略，所以容易陷入局部最优，所以，多次迭代选择不同的初始点可以一定程度改善这个问题

对于正则化参数 C，使用线性搜索法，从 0.0 到 0.3，步长为 0.003，逐一搜索：

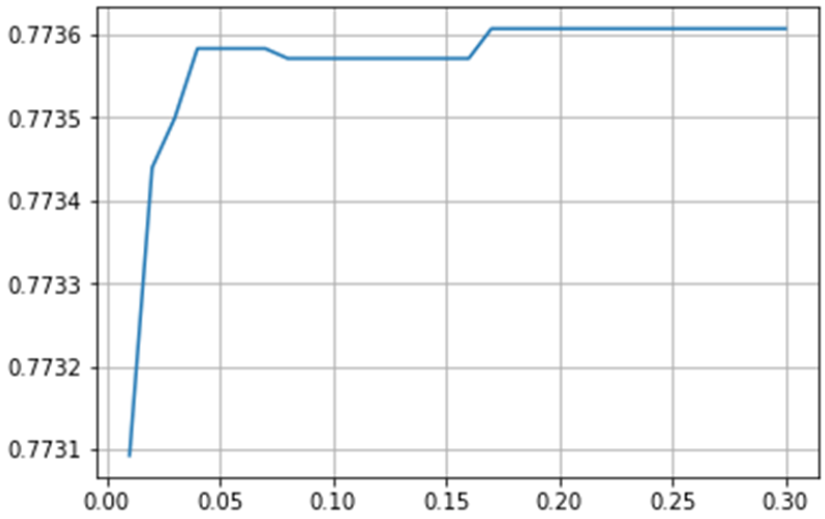


图 22

对于迭代轮数：我们一共选择了 6 轮来观察最优的那个轮数

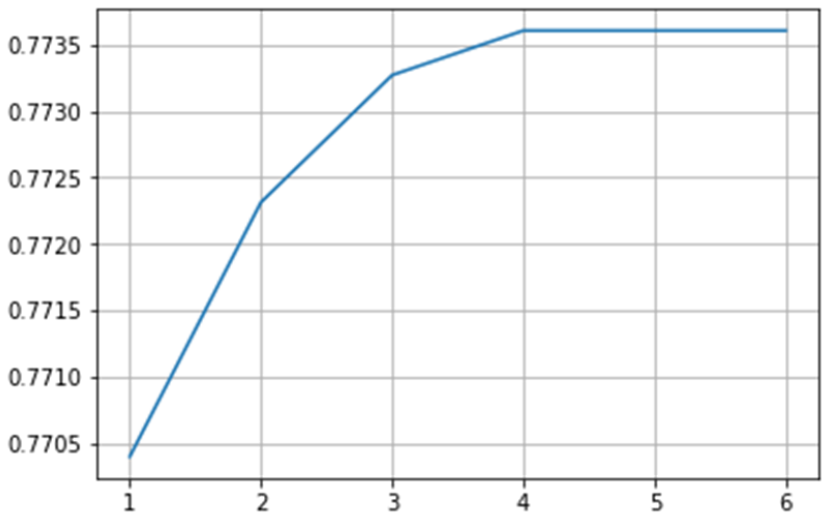


图 23

从上面两张图片中以看出，模型的性能到达瓶颈，无论是调整正则化参数还是增加轮数，模型的准确度均会收敛于 0.7736 左右，不过这为我们下一步的优化指明了方向（L2 正则化（lasso）倾向于删掉变量，而随着 C 的增大，模型的准确度上升说明，更少的特征系数被压缩到 0，或者说越少的特征被删除，可能会导致模型的准确率上升，所以需要增加特征来提高准确度。）

4.4.2 性能评估

对于一个分类器，光用准确率进行评判其实是不准确的，比如说，有 99 个人不违约，但是有一个人违约，这时分类器判断 100 个人全部不违约，那么这个模型的准确率就是 99%. 但是这样的分类器是没有意义的，这个时候我们需要使用 ROC 曲线来评判模型

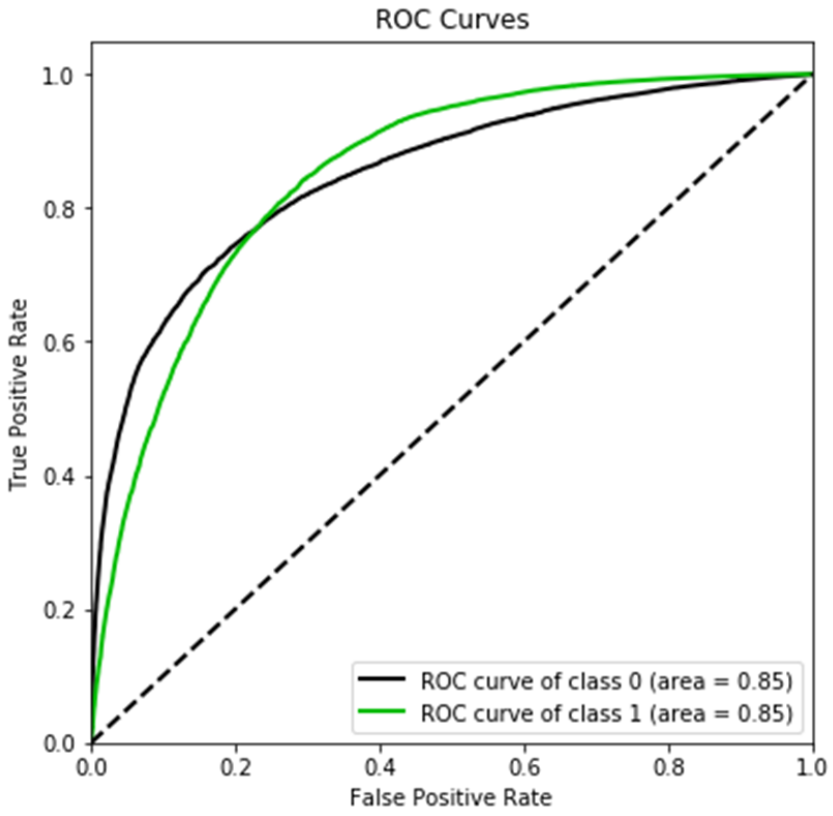


图 24

简而言之，ROC 曲线下面积即 AUC 越大，证明模型的泛化能力越好，越有用，在这里，判断为“不违约”的 AUC 与“违约”的 AUC 是相等的为 0.85，说明模型的分类能力较强，当然，一般地我们希望分类器得到的分类结果是完全正确的，也就是正例样本全部都能够被检测出来，也就是全部都是真正例，或者真反例，这个时候 TPR=1 且 FPR=0，反应在图像上好的分类器的折线应该更加接近左上角。从我们的图像上可以看出，当把把阈值设为 0.5 时，可以看出模型更擅长分类正例（即“违约”）。

5 分析结果

根据信用评分卡模型建立部分，我的得到了信用评分表，如下所示：其中要注意的是，分值越低代表信用越高

个人信用评分表	
base_score	[513.69958055]
age	Score
(-inf, 36.0]	-4.148989508
(36.0, 54.0]	-2.198646424
(54.0, 61.0]	1.97278464
(61.0, 74.0]	7.934720336
(74.0, inf]	13.57802031
RevolvingUtilizationOfUns	Score
(-inf, 0.099]	47.57442306
(0.099, 0.298]	14.45808056
(0.298, 0.465]	-2.757875689
(0.465, 0.982]	-23.20813408
(0.982, 1.0]	-10.41747808
(1.0, inf]	-43.88822935
DebtRatio	Score
(-inf, 0.0175]	33.30704105
(0.0175, 0.402]	0.807701793
(0.402, 1.469]	-8.570715929
(1.469, inf]	3.870049798
MonthlyIncome	Score
(-inf, 0.0963]	18.33255167
(0.0963, 5595.923]	-3.315386274
(5595.923, inf]	3.358845807
NumberOfOpenCreditLine	Score
(-inf, 1.0]	-7.129587455
(1.0, 3.0]	-2.747235448
(3.0, 5.0]	-0.432076422
(5.0, 17.0]	1.029758821
(17.0, inf]	3.882368167

(a) 评分表第一部分

NumberOfTime30-59Day	Score
(-inf, 0.0]	5.467155806
(0.0, 1.0]	-13.563895
(1.0, 2.0]	-21.46016998
(2.0, inf]	-24.01190636
NumberOfTimes90DaysLa	Score
(-inf, 0.0]	3.881818826
(0.0, 1.0]	-28.88851842
(1.0, 2.0]	-37.13290563
(2.0, inf]	-39.73942209
NumberRealEstateLoansO	Score
(-inf, 0.0]	-10.6601121
(0.0, 1.0]	5.345219766
(1.0, 2.0]	16.66339638
(2.0, 4.0]	10.2746443
(4.0, inf]	-7.854064206
NumberOfTime60-89Day	Score
(-inf, 0.0]	1.021526569
(0.0, 1.0]	-11.37043363
(1.0, 2.0]	-14.51039882
(2.0, inf]	-15.05480776
NumberOfDependents	Score
(-inf, 0.0]	14.48663704
(0.0, 1.0]	-13.43869709
(1.0, 2.0]	-12.23968079
(2.0, inf]	-11.06064618

(b) 评分表第二部分

表 (5)

5.1 评分卡的解释

以 age 为例，新客户的 age 信息在 0-36 的区间，则有 -4.148 的分数，在 36 到 54 岁的区间上，则有 -2.198 的分数，在 54 到 61 岁则有 1.972 的分数，以此类推，将每一个特征进行配对，得分，合计分数越高则说明其信用程度越高，越不容易违约。 可以看到 age 特征中，年龄越高，得分越高，越不容易违约，而在数据中（见可视化图 25）也与预想的分布相同.

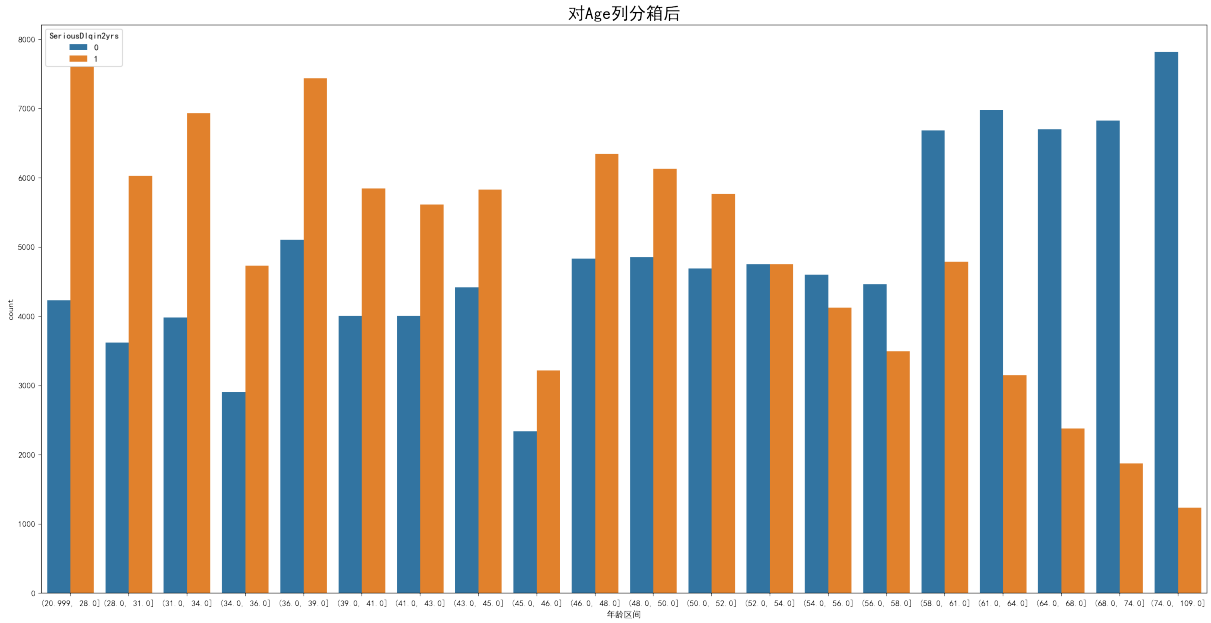


图 25

因此，说明我们的评分卡是有者不错的解释性的。

5.2 抽取样本用评分卡制作分数

我们抽取了三个样本，得到他们的分数如下：

	SeriousDlq	age	RevolvingU	DebtRatio	MonthlyInc	NumberOf	NumberOf	NumberOf	NumberRe	NumberOf	NumberOfDependent	得分
测试样本		40	1.106979	0.347578	2807	5	0	0	0	0	0	
得分		-2.19865	-43.8882	0.807702	-3.31539	-0.43208	5.467156	3.881819	-10.6601	1.021527	14.48664	478.8704
违约样本	1	38	0.025656	0.475841	3000	7	0	0	1	0	2	
得分		-2.19865	47.57442	-8.57072	-3.31539	1.029759	5.467156	3.881819	5.34522	1.021527	-12.2397	551.6955
不违约样本	0	48	0.813752	0.449796	7100	8	3	0	1	1	0	
得分		-2.19865	-23.2081	-8.57072	3.358846	1.029759	-24.0119	3.881819	5.34522	-11.3704	14.48664	472.4424

表 (6)

从中可以明显的看到，不违约样本的得分高于违约样本，其中测试样本为，隐性不违约样本 (隐性即隐藏其是否违约的标签), 求出得分后，发现与不违约样本分数相近，经检查，的确为不违约样本，故我们的评分卡效果良好，可以作为银行业务员判断客户违约概率的辅助手段，

6 优点与创新

创新点：

1. 横向比较了不同分类模型的优点，具有一定的指导意义。

优点：

1. 使用分箱法, 提高了模型的稳定性，避免特征中无意义的波动，提高了模型的鲁棒性：避免了极端值的影响
2. 使用卡方分箱, 一般的信用卡评分建模，使用的分箱方法是等频分箱，等频分箱没有受到目标变量的影响，主观性太强，而使用卡方分箱，则是考虑到，特征对于目标变量的“价值程度”，遵循统计学中“组内差异小，组间差异大”的原则。
3. 模型的泛化能力强，具有推广的价值。
4. 在分箱过程中考虑到对于预测变量的影响而使用 IV 判别法

7 模型改进

从学习曲线表现出来的是模型明显欠拟合，特征决定项目性能的上限，初步考虑是特征工程设计的不全面，没有将现有特征做的细致，删去的两列特征，导致模型获取的信息不足。

首先将删去的特征 NumberOfTime60-89DaysPastDueNotWorse 作为新特征在训练集和测试集上并入，加入新特征后，个别特征在作为分箱依据的 IV 曲线上会有所变化，故而根据曲线图像调整分箱数，　　使用逻辑斯蒂进行模型拟合
在选择不同模型的时候，我们选择的是学习曲线来评价不同模型对数据的合适程度，在选定模型为逻辑斯蒂后，为了评价其泛化能力，我们使用 AUC 作为评判模型有优劣的标准，调整特征后，绘画出对应的 ROC 曲线查看 AUC 的面积，观察到有所增加。

遵循着此种思路，我们将另一个在特征工程删去的特征 NumberRealEstateLoansOrLines 并入测试集和训练集，对数据进行逻辑斯蒂拟合，得到的 AUC 面积有所增加。

基于以上的验证，我们将两组特征重新并入数据集中，此时数据集的特征数是 10。

理论上讲，在特征工程中去除掉具有共线性、模型贡献度 IV 值不高的特征，是可以提高模型的拟合能力的，我们对出现情况的猜测是，8 组特征过少，模型处在欠拟合的状态，而此时即使增加模型贡献度低的特征，依然会对模型整体体现出的拟合度有明显贡献。

基于以上的猜测，我们考虑加入新的特征变量，这种特征在现实理论上能为分辨是否坏账提供一定的贡献，拟定为（月收入的平方/年龄），意图在分辨出年龄高而收入低的‘老无所就者’此种更容易导致坏账率的人群，对训练集和测试集增加特征（月收入的平方/年龄）而后拟合出的模型反应出的 AUC 面积有所增加，但是贡献很低。

我们以特征交互图为依据，将有明显交互效应如 age 与 NumberOfDependents 等多组交互后的变量分别作为新特征并入测试集和训练集，进行模型拟合，观察 ROC 曲线面积 AUC 的大小，发现不增反减，猜测是模型在增加 3 列特征后已经达到饱和，若再添加新特征即过拟合，导致拟合度下降，因此我们仅添加 3 组新特征，共 11 组特征，并且认为逻辑斯蒂模型已经调整至最优。

8 不足与发展方向

1. 目前的不足主要在于模型具有欠拟合的现象。

对于逻辑回归这种广义线性可加模型来说，消除过拟合的方式就是增加特征，或者特征交叉，或者按照特征工程的说法，就是重新构造特征。这些我们在本文中已经做了初步的尝试。

2.bad_{rate} 单调性问题

在分箱中说过，bad_{rate} 要满足单调性，我们的大部分特征都满足，但是有少量的类别型变量不满足单调性，而是“U 型”或者“倒 U 型”，不过这在行业里是可以被允许的。