广发信用卡第二届大学生实习真人秀比赛

团队：**Neo**

成员：梁强

吴飞

2018 年 **11** 月 **27** 日

目录

第一章 引言

1.1 研究背景及意义

随着社会经济、信息技术水平不断提高,越来越多的人开始使用信用卡进行交易,信用卡已成为现代生活中非常重要的支付手段。在我国,虽然信用卡发展时间较短但发展速度迅猛。信用卡透支消费已成为中国新型消费模式,并有望在不久的将来成为主要消费模式。伴随着信用卡使用规模的迅速扩大,信用卡欺诈问题变得越来越严峻,解决此问题已变得刻不容缓。信用卡欺诈问题不仅在经济上令银行蒙受巨额损失,还让银行因此失去大量客户资源,极大地影响了我国商业银行风险控制情况,严重阻碍金融系统的正常发展壮大。

至此如何采用统计方法和机器学习模型对信用卡交易中的欺诈判别成为银行金融机构关注的重点问题，同时针对于信用卡欺诈样本的稀缺性，与非欺诈样本的占比极不均衡方面的问题，将采用何种策略与方法能更好的处理不平衡分类成为金融业界关注的重点方向。

1.2 研究现状

信用卡交易欺诈检测的研究一直受到国内和国外学者的关注，因此从采取的模型算法和处理不均衡的方法两个维度来分析对信用卡交易欺诈的现状和成果。

Rosset 等人提出了基于规则的两阶段信用卡欺诈检测系统，第一阶段运用 C4.5 算法生成欺诈行为相关规则，第二阶段依据欺诈覆盖度进行规则选择，并根据客户信用卡交易行为对欺诈规则进行排序。H Shao 等人运用改进的 C4.5 方法构建扩展的多维评估模型，该模型提升了信用卡欺诈检测的准确度，同时决策树方法的性能也有所提高。Maes 等在构建信用卡欺诈识别模型过程中，综合利用了贝叶斯网络和神经网络两种算法并对它们进行了细致的比较，最后得出结论贝叶斯算法识别欺诈准确率比神经网络算法高，但建模速度比神经网络算法慢。童凤茹等人利用集成方法 AdaBoost 算法建立信用卡欺诈交易识别系统，实验结果表明该系统对于信用卡欺诈风险控制具有重要意义。Phua、Alahakoon Lee 等人基于元学习策略，把贝叶斯算法、BP 神经网络、C4.5 结合起来，提出了信用欺诈检测模型——Minority Report。Ekrem Duman 等人（2011）提出了一种新的组合的元启发式算法，通过使用遗传算法和散射搜索，应用于真实数据集进行欺诈检测。苏亚婷在传统支持向量机(SVM)的基础上，综述了无核二次曲面支持向量机

(QSSVM)的原理和性质，在此基础上提出模糊二范数无核二次曲面支持向量机(F2NQSSVM)来进行信用卡欺诈检测，并和 KNN 分类、神经网络、传统 SVM 作对比取得更好的预测效果。

目前处理不平衡问题的方法可以概括为两类。一种比较普遍的方法是在数据层面通过采用欠采样或过采样的方法，重新分配类别分布，例如文献[10]提出的合成小类过采样技术（Synthetic Minority Over-sampling Technique， SMOTE），文献[11]提出的自适应样本合成方法（Adaptive SyntheticSampling Approach， ADASYN）。 欠采样方法可以提升模型对小类样本的分类性能，但是由于这种方法会造成大类样本数据的信息丢失而使得模型无法充分利用已有的信息。 传统的过采样方法可以生成少数类样本的数据，但是是根据少数类数据生成，只是基于当前少数类蕴含的信息，缺乏数据多样性，一定程度上会造成过拟合。另一种是在算法层面上，算法层面包括集成学习和代价敏感学习。集成学习通过集成多个分类器，来避免单个分类器对不平衡数据分类预测造成的偏差[12]，文献[13]提出在自适应增强模型（Adaptive Boosting， AdaBoost） 每次迭代中引入 SMOTE 的 SMOTEBoost 算法，文献[14]提出在 AdaBoost 每次迭代中引入随机欠采样 （Random Under-Samplingmethod， RUS）的 RUSBoosts 算法。代价敏感学习是在算法迭代过程中，设置少数类被错分时具有较高的代价损失[15]，通常与集成学习算法组合使用。代价敏感方法只是在算法层次进行了修改，没有增加算法的开销，效率较高，有效提高了不平衡数据的分类效果，但是由于主观引入代价敏感损失，损失函数的设计会影响算法的迭代效果，普遍适用性较弱。莫赞等人基于对抗生成网络和集成学习方法，提出了一种新的针对二类不平衡数据集的分类方法——对抗生成网络-自适应增强-决策树（Generative Adversarial Nets- Adaptive Boosting-Decision Tree，GAN-AdaBoost-DT）算法对于信用卡欺诈样本的评判准确性得到 4.5%的提升。Maira Anis 针对信用卡欺诈的不平衡性使用了一种以马氏距离为中心的新的相似性测量方法。这种相似性测量与传统的最近邻度量不同:新方法使用以数据为中心的方法来寻找关键样本，而其他固定重采样技术则使用以数据质心为中心的协方差矩阵进行相似性度量，新的重采样方法是可靠的，在处理具有高召回率的不均衡信用卡数据是有效的。

1.3 行文思路及框架

本文对信用卡交易过程的中的欺诈行为进行预测，使用 AURPC 作为评估模型对欺诈概率的预测的效果。主要经过数据探索、特征工程、异常点检验、模型建立、模型评

估的过程完成文章报告的撰写。整体的框架如下图 1.1 所示。

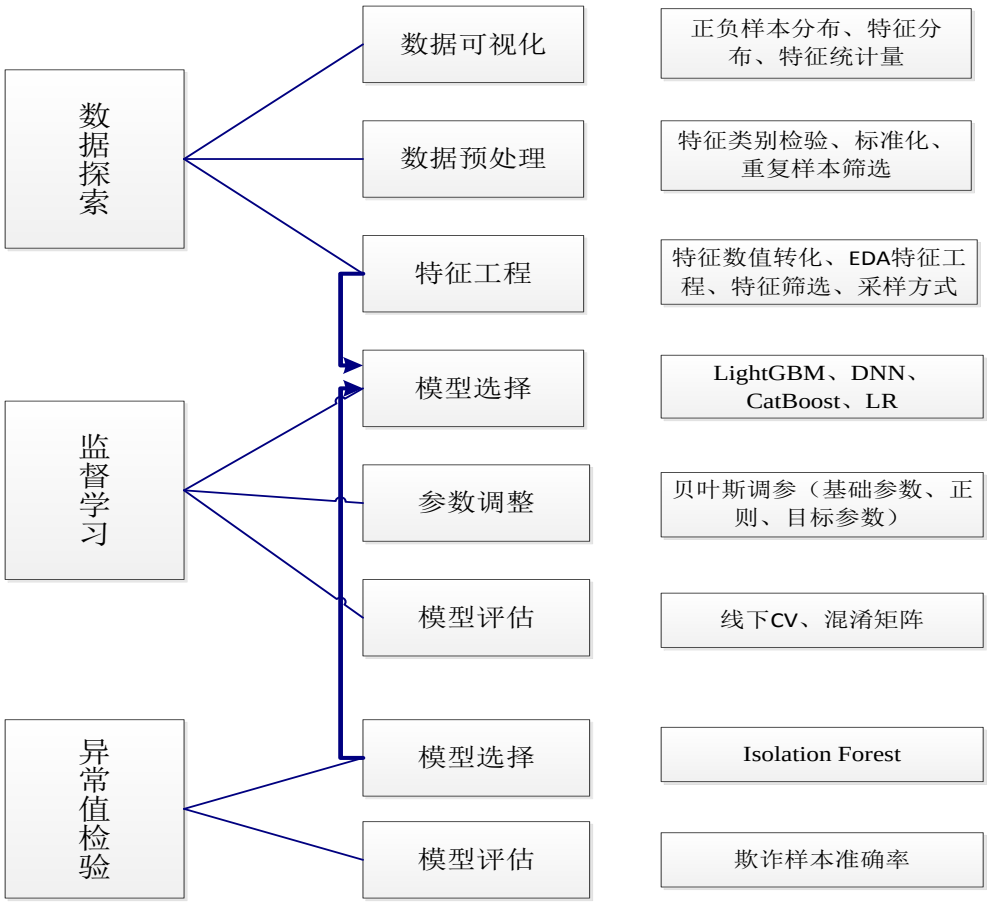


图 1.1 行文思路

第二章 数据探索性分析

2.1 目标变量

1 是欺诈类样本,在训练集中只有 443 条，占比 0.2%，0 是非欺诈类样本，占比 99.8%。数据存在严重的类不平衡问题。

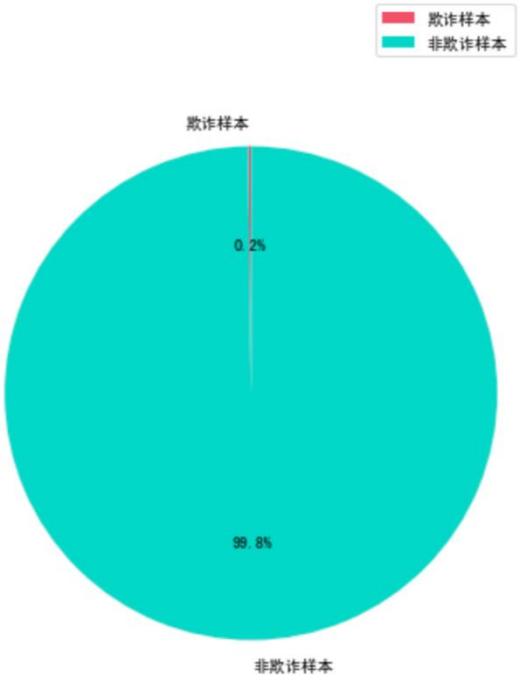


图 2.1 label 饼状图

2.2 特征分布

数据特征除 Time 和 Amount 外的其他变量都进行了 PCA 变换，共有 30 个特征。我们画出每个特征的分布状况，以此来分析在 0,1 不同 label 下特征的分布差异，获取特征工程的灵感。根据图 2.2,我们可以看出 PCA 部分特征在 0,1 条件下的分布具有明显差异，我们可以依据此做特征提取，让这些差异更容易被模型识别。

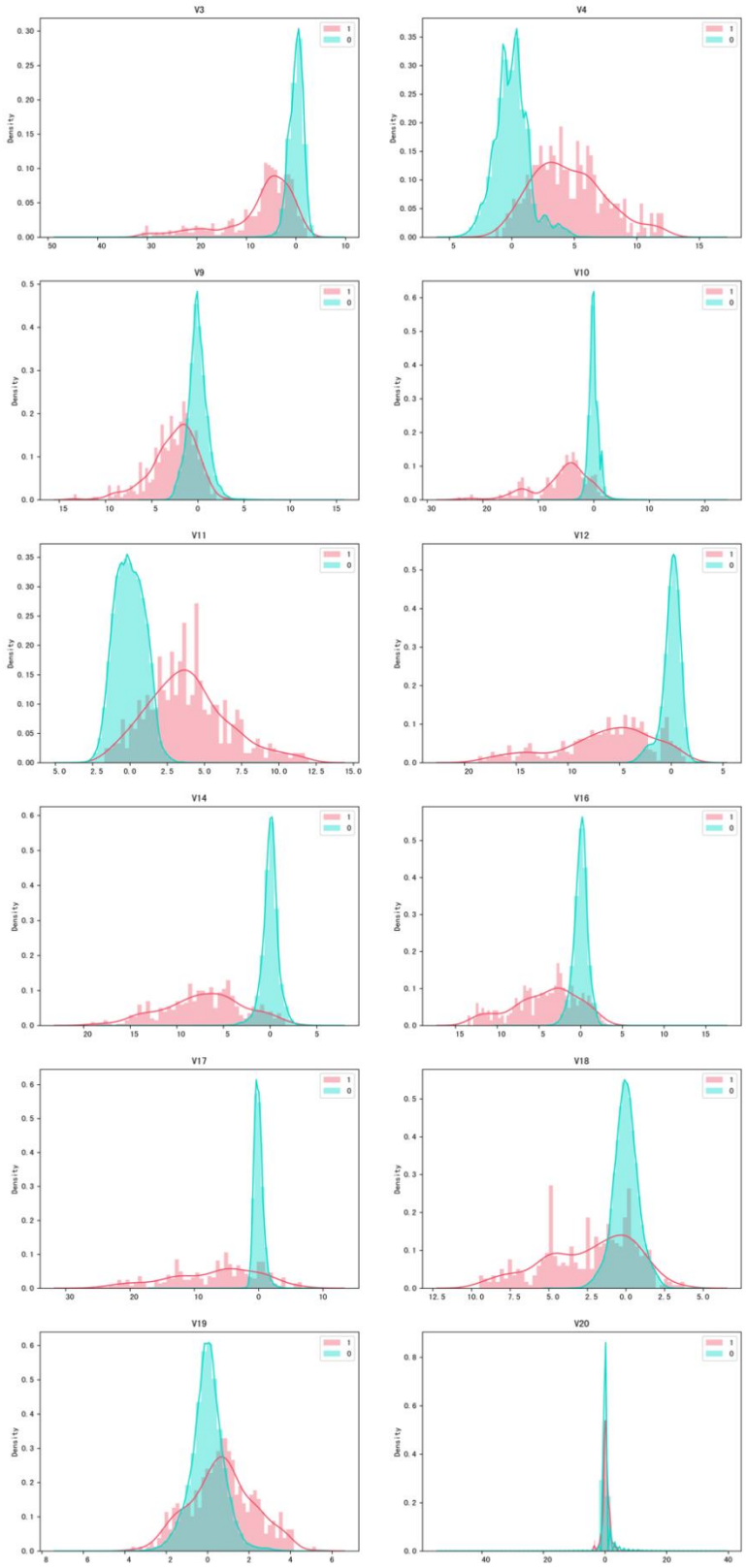


图 2.2 PCA 特征分布图

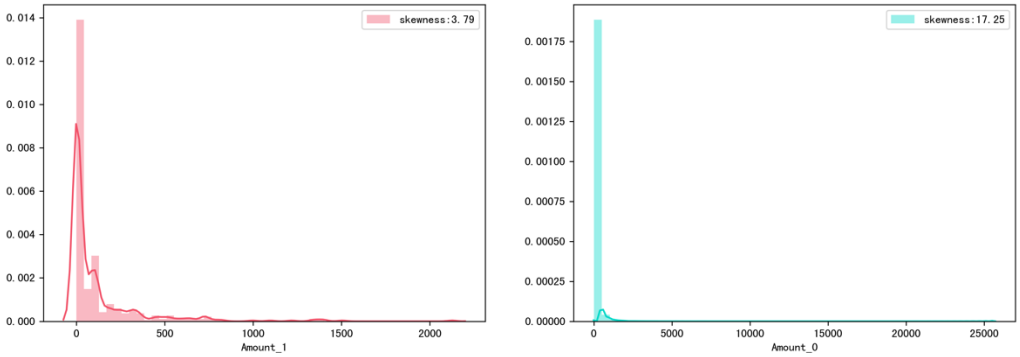


图 2.3 Amount 特征分布图

图 2.3 表明了 Amount 特征分布偏斜，在使用线性模型前，可以进行 log 或者 box-cox 转换。同时可以看出，欺诈类的 Amount 区间范围小于未欺诈类。

2.3 特征与标签相关性分析

由于主要特征来源于 PCA 降维，所以 PCA 特征间不相关，但是我们可以根据特征与标签的相关分析，筛选出与标签相关性较强的特征，尝试特征间的组合或者围绕这些特征构建新的特征。

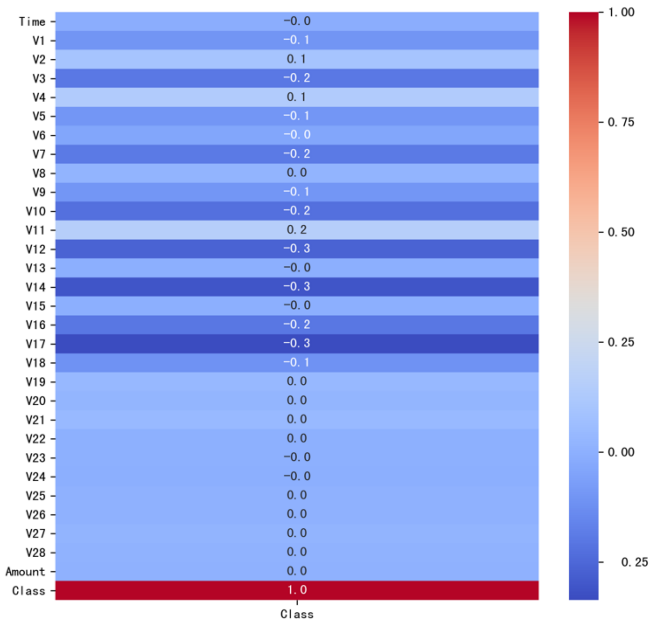


图 2.4 特征与标签的 heatmap 相关图

2.4 离群点分析

我们通常把离群点定义为远离数据主要部分的样本。如果是简单的定性分析，可以通过观察箱形图来查看离群点。有一些模型对离群点具有抗性，例如基于树的模型，会对训练集进行一系列的划分，离群点往往不会对此类模型产生重大影响，但是很多线性模型则会受到离群点影响，在具体建模过程中，我们可以对不同模型进行不同的数据预处理。

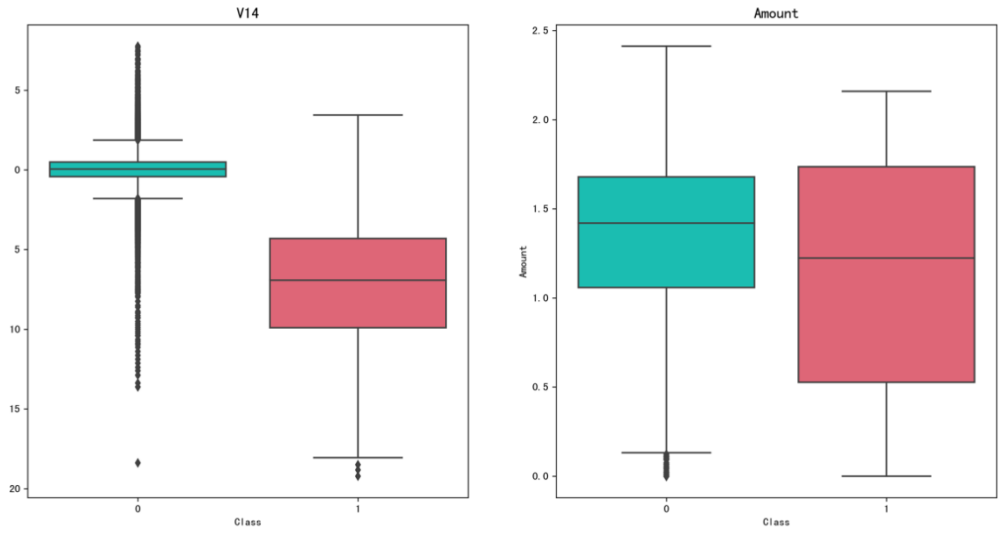


图 2.5 特征箱型图

第三章 数据预处理与特征工程

3.1 数据预处理

我们的数据预处理主要包括四个方面：删除重复样本；离群点处理；对 Amount 进行 Box-Cox 转换；特征标准化。

1.删除重复样本：我们删除 905 条重复样本，同时保留下最近时间的样本，得到的样本包括欺诈类 424 条。

2.离群点处理。为减小盲目删除样本后带来的信息损失，我们采用了一种常用的离群点检测方法—Tukey Method。通过计算特征的 IQR 四分位差，得到 outlier_step =1.5*IQR，如果值大于(上四分位数+outlier_step)或者小于(下四分位数-outlier_step),就判定这个值为离群点。设定阈值 n，如果一个样本中出现离群点个数大于 n，我们就可以删除这个样本。

3.Box-Cox 转换。Box 和 Co 下（1964）提出了一种用一个参数λ进行索引的变换族:

$$x^{*}=\begin{cases}\dfrac{x^{\lambda}-1}{\lambda} & \text{如果}\lambda\neq 0\\ \log(x) & \text{如果}\lambda=0\end{cases}$$

$$h_{g}(x)=\frac{1}{1+e^{-z}}=\frac{1}{1+e^{-\vartheta^{t}x}}$$

$$y=\frac{1}{1+e^{-x}}$$

相比 log 转换，这个变换族还包括平方变换，平方根变换，倒数变换已经在此之间的变换。就使用案例来说,Box-Cox 变换更加直接,更少遇到计算问题,而且对于预测变量同样有效。

4.对 Time,Amount 进行特征标准化处理。

3.2 特征工程

时间特征：特征 Time 是包含每个交易和第一天 00：00：00 之间经过的秒数，所以有个自然的想法是构造每个交易距离起始时间的分钟和小时数。加上新的特征后，配合调参，线下 cross validation 有百分位的提高。

Groupby 特征：对分钟和小时进行分组，统计组内与标签相关性强的特征的统计量，例

如 V14,V12 等。实验发现，构造出的分组特征并未提升模型的预测能力。

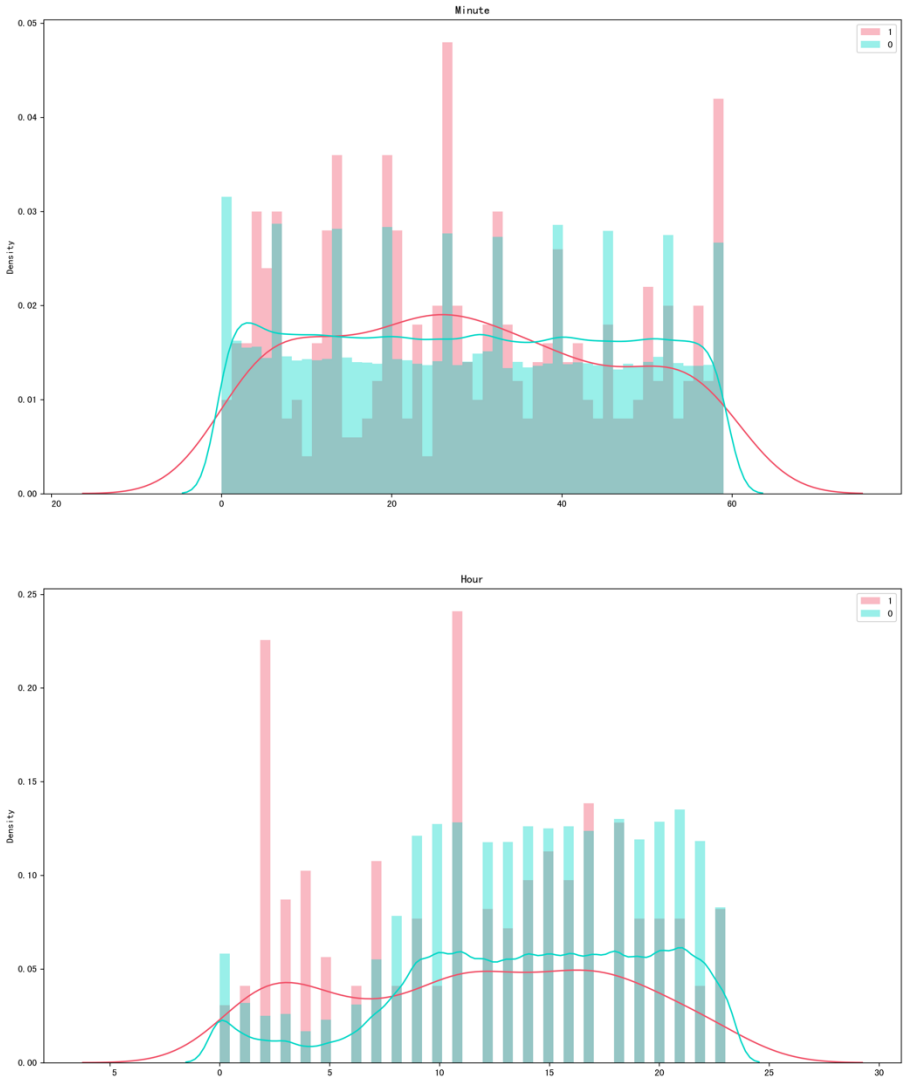


图 3.1 分钟和小时分布图

特征提取：通过 EDA 发现部分特征中 0,1 标签下分布具有明显差异。我们通过提取那些分布差异较大值或者区间来构造新的特征，让模型能更容易的发现这些差异。加入新特征，配合特征选择，线下 CV 有千分位数提高。

构造思路：df[新特征] = df[条件区间或特殊点].astype(int)

交叉特征：尝试将与标签强相关的特征进行组合，构建乘法特征和比例特征。

3.3 特征选择-PIMP

特征选择的方法有方差选择，皮尔逊相关系数，互信息，正则化等。由于树模型的广泛使用，基于树模型的特征重要性排序是一种高效常用的方法。然而模型得到的特征重要性存在一定的偏差，这些往往对特征选择产生干扰。我们采用的特征选择方法是 PIMP 算法(Permutation Importance)，它的主要思想是修正已有的特征重要性。具体算法描述如下:

1.打乱标签的排序，得到新的训练集，重新训练并评估特征重要性。

2.重复第一步 n 次，得到每个特征进行多次评估的特征重要性集合，我们称之为 the null importance.

3.计算标签真实排序时，模型训练得到的特征重要性。

4.利用第二步得到的集合，对每个特征计算修正得分。

5.修正规则如下: Corrected_gain_score=100*

(null_importance_gain<np.percentile(actual_imp,25)).sum()/null_importance.size()

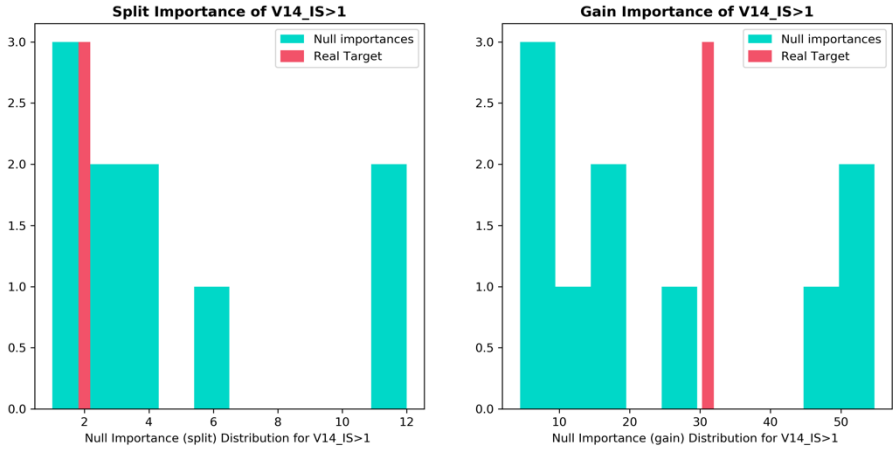


图 3.2 真实标签下特征重要性与其 null_importance 分布对比

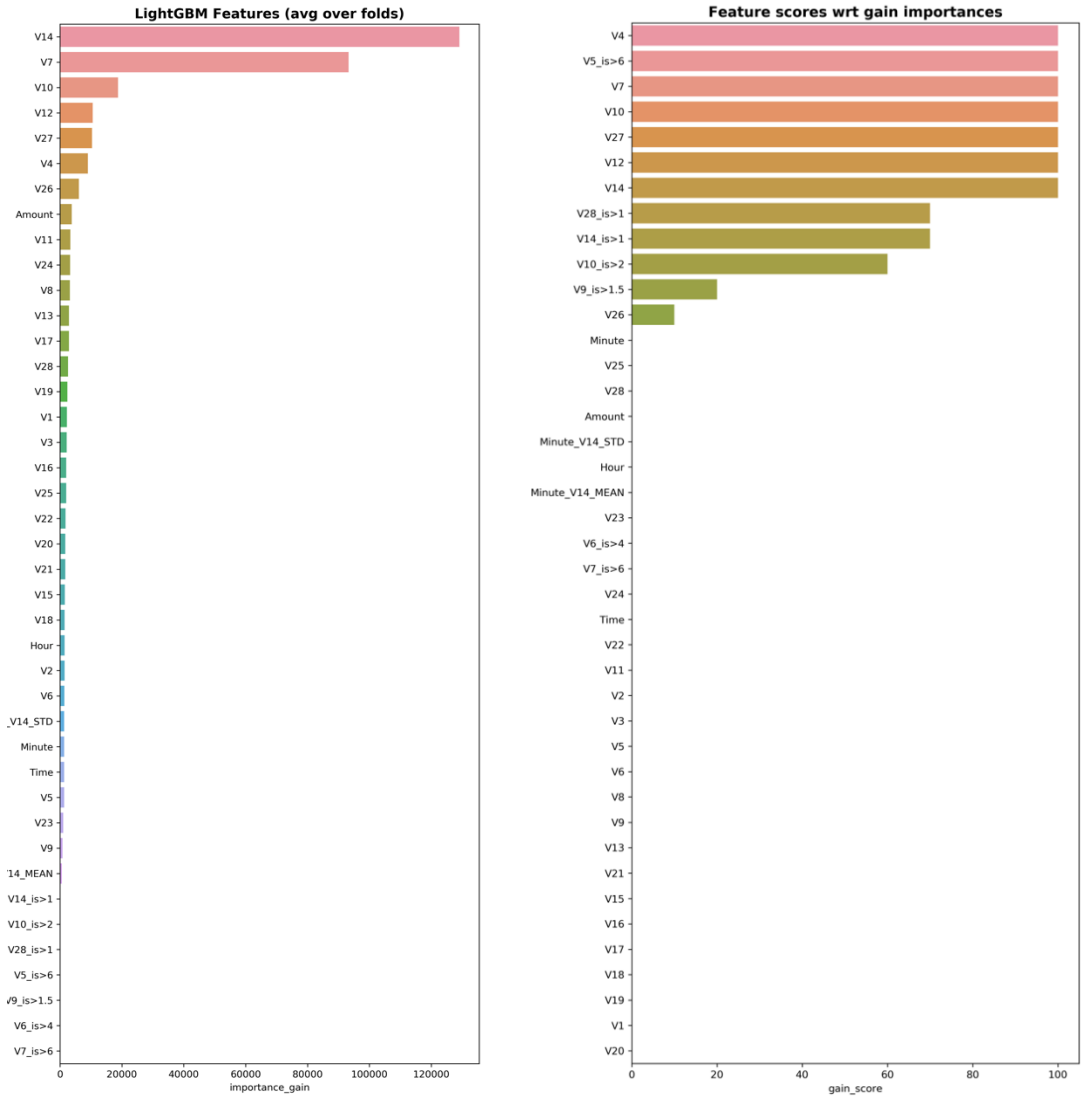


图 3.2 特征重要性修正前后对比

第四章 异常点检测办法

由于给的训练集的样本分布是极不均匀的分布状态，因此在这里把对于识别欺诈样本的问题转化成为异常点的检测问题对待，看能否把欺诈样本当做异常点检测出来。采用的方法是 Isolation Forest(孤立森林)进行处理。

4.1 孤立森林算法简介

它是一个基于 Ensemble 的快速异常检测方法，具有线性时间复杂度和高精准度，适用与连续数据的异常检测，将异常定义为“容易被孤立的离群点 (more likely to be separated)”——可以理解为分布稀疏且离密度高的群体较远的点。用统计学来解释，在数据空间里面，分布稀疏的区域表示数据发生在此区域的概率很低，因而可以认为落在这些区域里的数据是异常的。孤立森林属于无监督学习，对于如何查找哪些点是否容易被孤立（isolated），iForest 使用了一套非常高效的策略。假设我们用一个随机超平面来切割（split）数据空间（data space），切一次可以生成两个子空间（想象拿刀切蛋糕一分为二）。之后我们再继续用一个随机超平面来切割每个子空间，循环下去，直到每子空间里面只有一个数据点为止。

对数据空间的切分是 iForest 的设计核心思想，本文仅介绍最基本的方法。由于切割是随机的，所以需要用 ensemble 的方法来得到一个收敛值（蒙特卡洛方法），即反复从头开始切，然后平均每次切的结果。iForest 由 t 个 iTree（Isolation Tree）孤立树 组成，每个 iTree 是一个二叉树结构，其实现步骤如下：

1. 从训练数据中随机选择Ψ个点样本点作为 subsample，放入树的根节点。
2. 随机指定一个维度（attribute），在当前节点数据中随机产生一个切割点 p——切割点产生于当前节点数据中指定维度的最大值和最小值之间。
3. 以此切割点生成了一个超平面，然后将当前节点数据空间划分为 2 个子空间：把指定维度里小于 p 的数据放在当前节点的左孩子，把大于等于 p 的数据放在当前节点的右孩子。
4. 在孩子节点中递归步骤 2 和 3，不断构造新的孩子节点，直到 孩子节点中只有一个数据（无法再继续切割） 或 孩子节点已到达限定高度 。

获得 t 个 iTree 之后，iForest 训练就结束，然后我们可以用生成的 iForest 来评估测试数据了。对于一个训练数据 x，我们令其遍历每一棵 iTree，然后计算 x 最终落在每个树第几层（x 在树的高度）。然后我们可以得出 x 在每棵树的高度平均值，即 the average path length over t iTrees。*值得注意的是，如果 x 落在一个节点中含多个训练数据，可以使用一个公式

来修正 x 的高度计算，详细公式推导见原论文。

孤立森林的特点：

1. 具有线性时间复杂度。
2. 不适用于特别高维的数据。
3. 对全局稀疏点敏感，不擅长处理局部的相对稀疏点

4.2 实验验证

首先针对训练集中正负样本的抖动性进行检验，分别提取出训练集的中的欺诈样本和非欺诈样本放入到模型中训练，得到欺诈样本的非异常占比 48.53%，非欺诈样本的非异常占比 98.99%。由此可得欺诈和非欺诈样本自身都存在抖动性，欺诈样本的抖动性远大于非欺诈样本，因此为了平衡自身抖动性问题需要增加在训练过程中增添权重分布。

下面采用三种方式对异常点的过程进行检验：

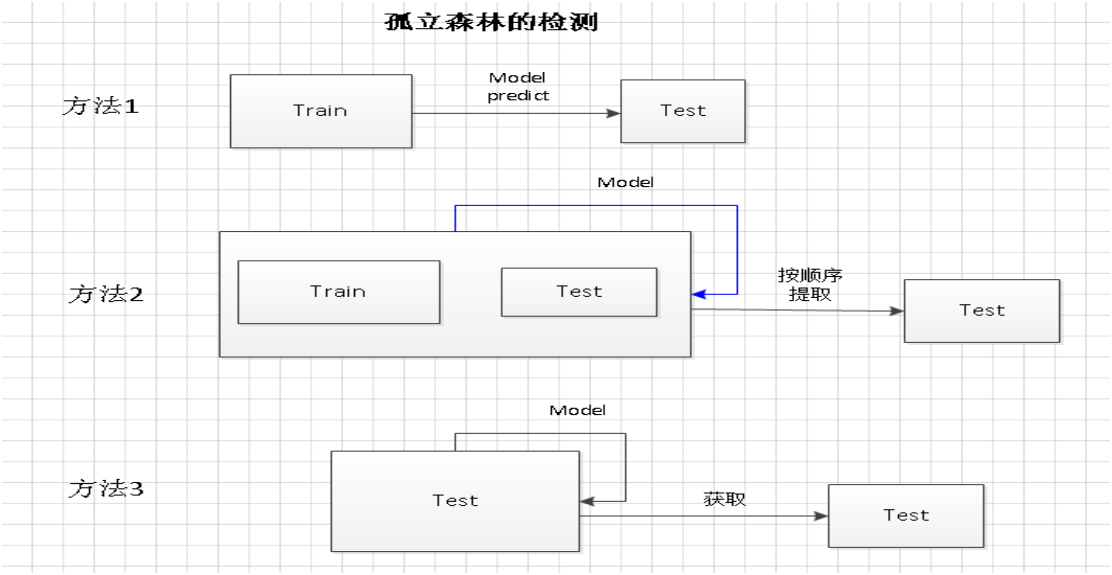


图 4.1 iForest 检验方法

方法一：使用训练集训练 Model,把训练好的模型用于预测测试集 test,获得的结果为：

Label	precision	recall	f1-score	support
0	1.00	1.00	1.00	28431
1	0.26	0.12	0.17	49
avg / total	1.00	1.00	1.00	28480

方法二：把训练集和测试集放在一起训练模型，之后按照标签的顺序提取出 test 集所对应的标签的序号，作为预测 test 的结果：

Label	precision	recall	f1-score	support
0	1.00	1.00	1.00	28431

1	0.14	0.16	0.15	49
avg / total	1.00	1.00	1.00	28480

方法三：把测试集进行使用孤立森林进行单独训练，训练好的模型对 test 集进行预测，输出预测的结果：

Label	precision	recall	f1-score	support
0	1.00	1.00	1.00	28431
1	0.14	0.18	0.16	49
avg / total	1.00	1.00	1.00	28480

由此我们发现，使用训练集训练 Model 后再进行预测在查准率相对较高，而使用后面的两种方法的查全率要高于第一种方法。

三种算法的运行时间对比：

	方法一	方法二	方法三
时间	30.867765s	53.939085s	5.9693415s

4.3 异常点检测的使用建议

尝试把使用孤立森林检测出来的异常点的概率当做特征的进入到监督模型中进行训练，把方法一至方法三获取到的三种异常预测概率作为三个基础特征进行训练的处理。

第五章模型设计与分析

5.1 LightGBM

5.1.1 LightGBM 原理

LightGBM (Light Gradient Boosting Machine)是 2016 年微软亚洲研究院公 布的一个开源、快速、高效的基于决策树算法的提升框架，被用于排序、分类、 回归等多种机器学习的任务，支持高效率的并行训练。LightGBM 的优化算法主要有两点。一是基于梯度的单边采样 GOSS，二是 EFB(可以有效实现无损加速)。类似于 AdaBoost 中赋予样本权重，GOSS 利用样本的梯度信息确定权重。如果某个样本对应的梯度小，说明这个样本的训练误差小，它被训练得很好，所以丢弃这些样本会改变样本分布，对模型的学习精度有影响。GOSS 保留所有梯度值大的样本，从小梯度值中随机抽样，同时为了弥补对数据分布的影响，在计算信息增益时，会在小梯度样本上乘以一个常数值。GOSS 效果是实现了减少训练样本和保持训练精度间的平衡。EFB 的思想则是在减少特征数量的同时实现无损加速。通过把多个稀疏无关的特征组合成单个特征，从 feature bundles 中建立 feature histograms。

5.1.2 模型优化思路

由于数据的类极度不平衡，所以不能采用准确率作为评价指标。这次优化过程采用的是 Precision-Recall Curve (AUPRC)作为评价标准。选择概率阈值不同，查准率和查全率也不同，为了更加综合的考虑模型的性能，我们最终选用的指标是 PR 曲线的面积，在 scikit_learn 中用 average_precision_score 来计算。我们模型优化的过程是先建立 lightGBM 基模型 baseline,在 baseline 上进行手动调参,然后陆续加入新特征,进行微调，最后进行贝叶斯调参。

Step1:建立 baseline,数据预处理完后，用五折交叉验证计算。

Step2:加入时间特征。

Step3:加入 groupby 特征。

Step4:加入提取的特征，进行 PIMP 特征选择。

Step5:贝叶斯调参。

LGB 参数设置

boosting_type	gbdt
Learning_rate	0.01
Colsample_bytree	0.84
subsample	0.9
Max_depth	7
Lambda_l1	0.14
seed	0
nthread	4

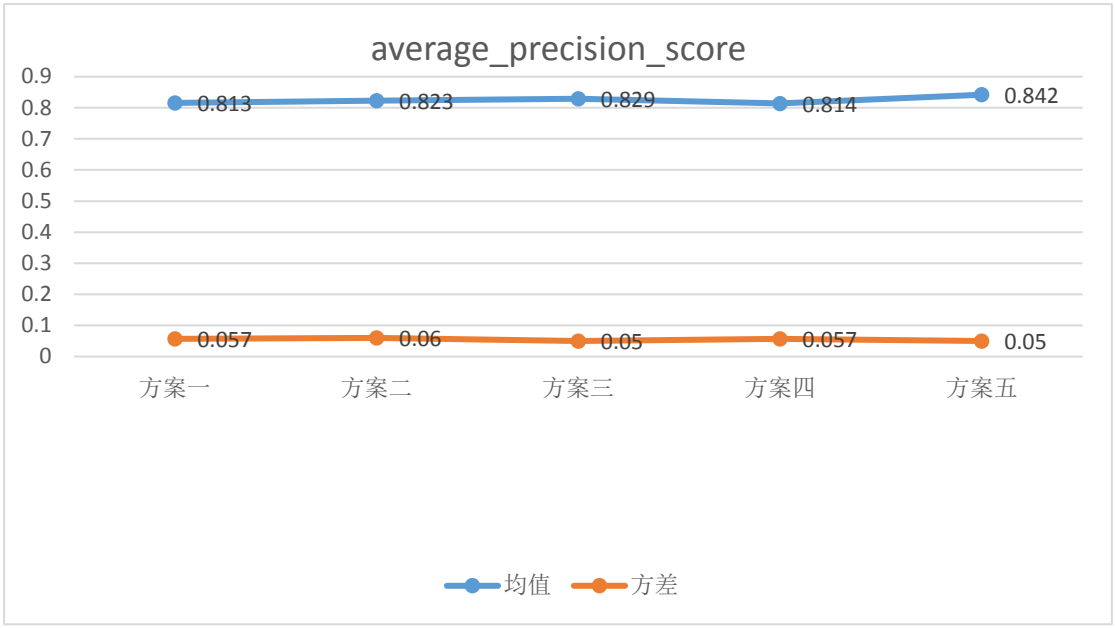
贝叶斯参数优化

boosting_type	gbdt
Learning_rate	0.01
Colsample_bytree	0.8290813068817113
subsample	0.9
Max_depth	7
Lambda_l1	0.16277128856691855
seed	0
nthread	4

5.1.3 实验结果

lightGBM 实验结果对比

baseline	0.813
方案一：时间特征	0.823
方案二：groupby 特征	0.829
方案三：特征提取+特征选择	0.814
方案四：在方案二基础上贝叶斯优化	0.842



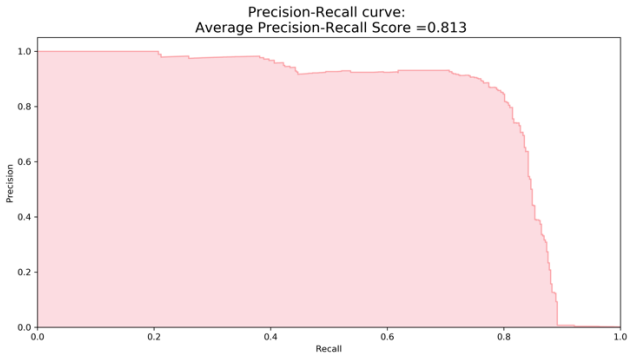


图 5.1 Baseline PR 曲线

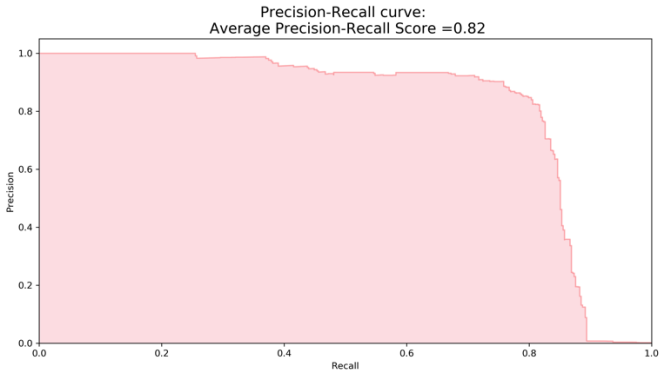


图 5.2 时间特征 PR 曲线

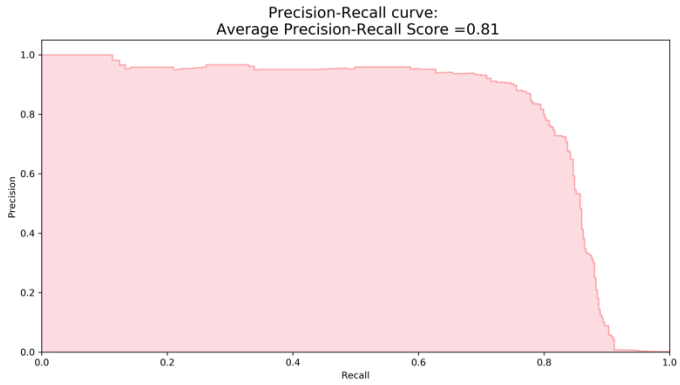


图 5.3 特征提取+特征选择 PR 曲线

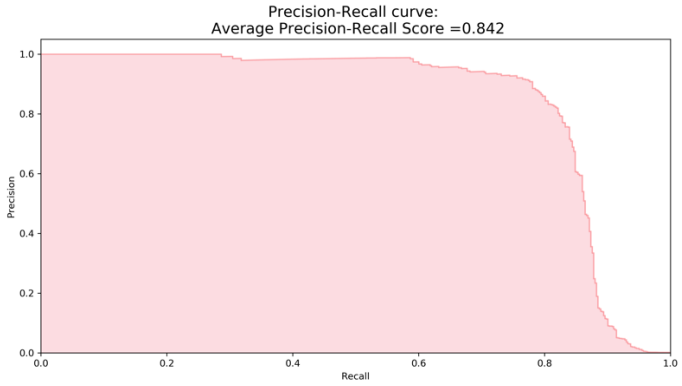


图 5.4 贝叶斯优化 PR 曲线

以上实验说明，lightGBM 在此数据集上的表现良好, 加入时间和 groupby 特征，配合贝叶斯参数优化在交叉验证中效果是最好的，特征提取加上特征选择的方案效果不理想。

5.2 神经网络

5.2.1 神经网络原理

神经网络是一种模拟人类大脑神经元机制的算法。算法主要过程包括前向传播和误差反向传播。前向传播过程是通过定义神经元连接方式和激活函数实现的。输入从输入层经过隐藏层，最后到输出层，依靠隐藏层完成的间接计算可以处理许多复杂问题。隐藏层可以找到数据内在特征，后续层可以在这些基础上进行操作。而神经网络训练参数的方法基础就是梯

度下降算法。BP 算法其实质则是复合函数的链式求导。

5.2.2 神经网络搭建

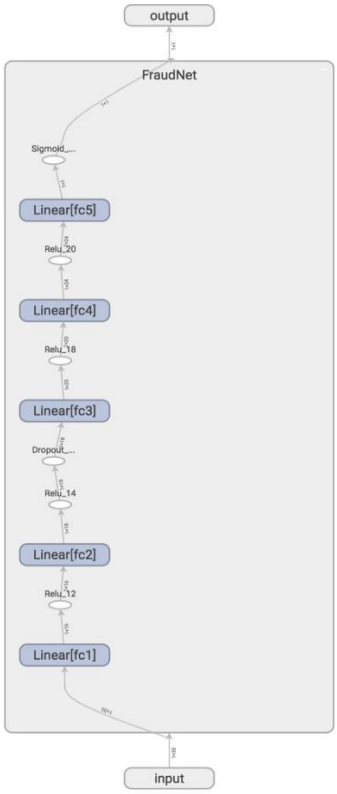


图 5.5 神经网络可视化

我们搭建了一个简单的 5 层神经网络，输入层是 30 个原始特征，经过 16,18,20,24 个神经元的隐藏层，隐藏层的激活都是定义为 relu 函数，输出层激活函数定义为 sigmoid。Relu 相比 sigmoid 减少了很多计算量，同时会使一部分神经元的输出为 0，这样就造成了网络的稀疏性，减少了参数的相互依存关系，缓解了过拟合问题的发生。

5.2.3 实验结果

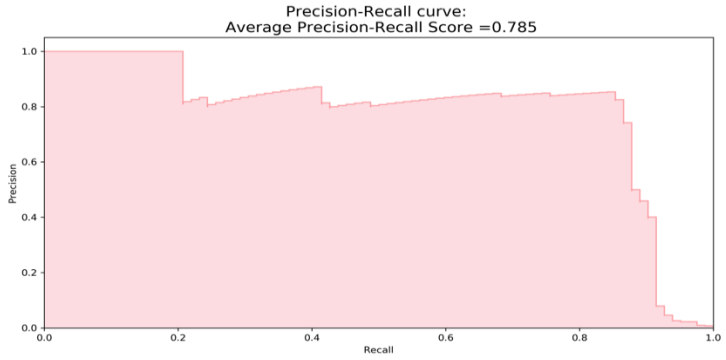


图 5.6 神经网络　PR 曲线

5.3　Catboost

5.3.1　Catboost 原理

Catboost 是从基础的 GBDT 算法改进了两个方面而得来的：其一是对训练样本的变量特征进行转变，其二是从处理梯度预测估计中的偏差角度。整体而言 catboost 本质是，

CatBoost = Category + Boosting

2017 年 7 月 21 日，俄罗斯搜索引擎公司 Yandex 开源 CatBoost，亮点是在模型中可直接使用 Categorical 特征并减少了需要调整的参数个数。Catboost 在分类变量索引方面具有相当的灵活性，可以用在各种统计上的分类特征和数值特征的组合将分类值编码成数字。由于 GBDT 算法存在逐点梯度估计中的偏差问题，Catboost 算法采取有序的 boosting 算法和融合的有序的 TBS 方法解决了存在的问题，进一步提高了运算的效率和准确度。

5.3.2　实验结果

CatBoost 实验结果对比

baseline	0.8132
方案一：时间特征	0.8303
方案二：样本去重	0.8145
方案三：参数贝叶斯优化	0.8361

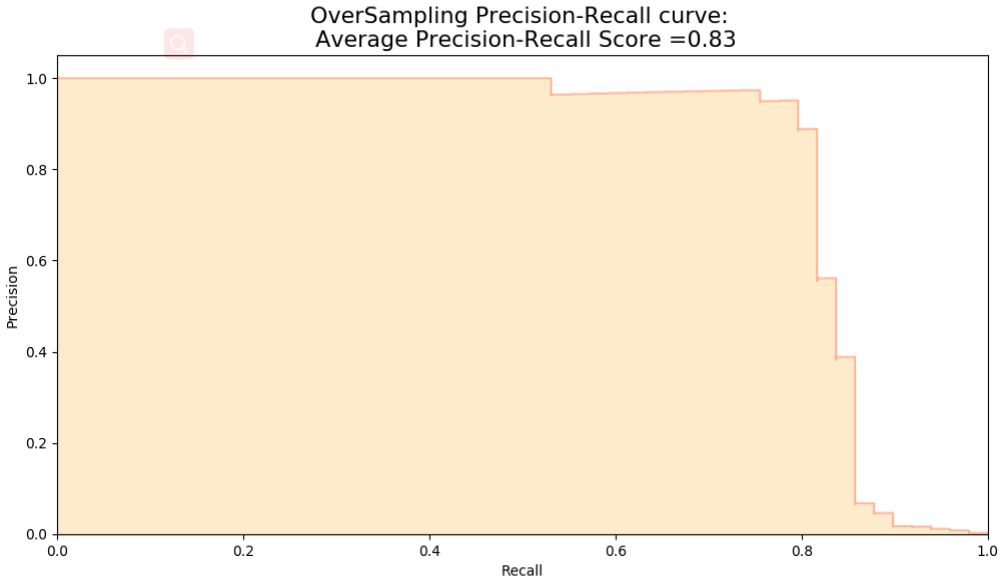


图 5.7 catboost PR 曲线

5.4 Logistic 回归

5.4.1 Logistic 回归原理

对于 Logistic Regression 来说，其思想也是基于线性回归（Logistic Regression 属于广义线性回归模型）。其公式如：

$$h_g(x)=\frac{1}{1+e^{-z}}=\frac{1}{1+e^{-\vartheta^t x}}$$

其中 $y=\frac{1}{1+e^{-x}}$ 被称作 **sigmoid** 函数，我们可以看到，Logistic Regression 算法是将线性函数的结果映射到了 sigmoid 函数中

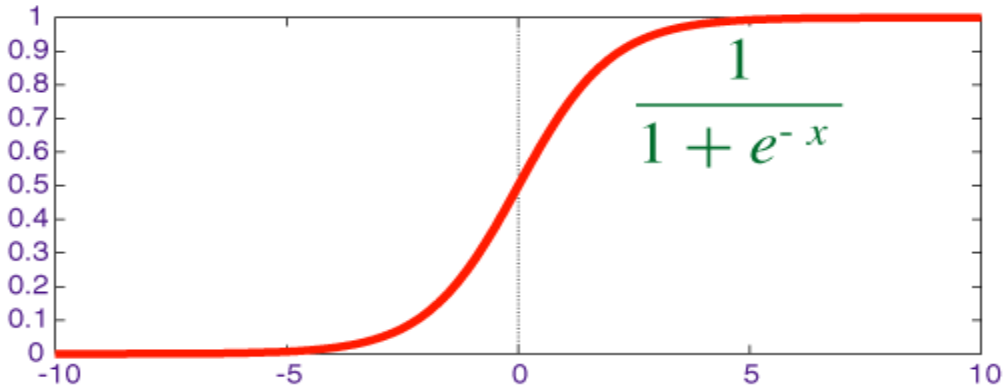


图 5.8　sigmoid 函数图

5.4.2　实验结果

Logistic 回归实验结果对比

baseline	0.649
参数贝叶斯优化	0.686

5.5　模型运行时间对比

算法运行时间对比（单位：秒）

lightGBM	70
CatBoost	260
DNN	83.64
iForest	54.08
LR	2.390

5.6　模型的融合示意图

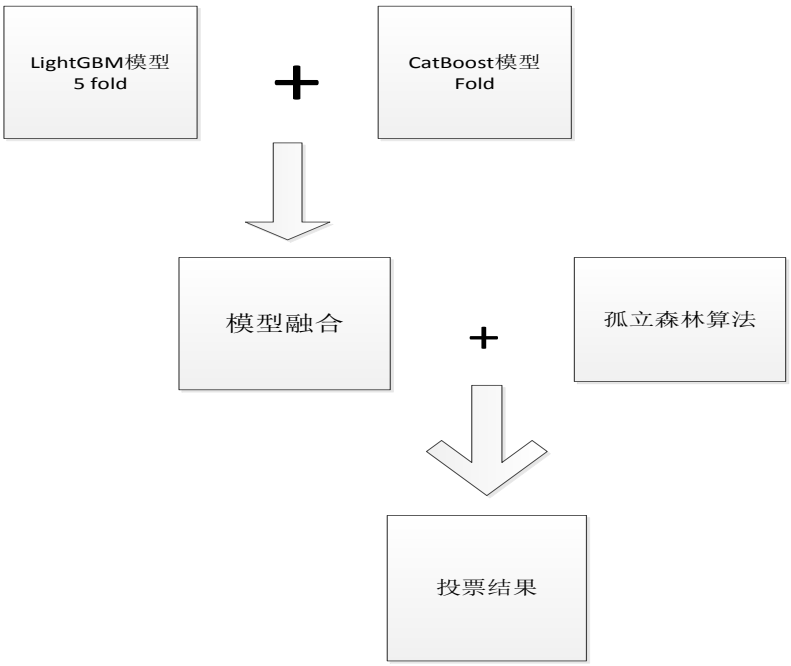


图 5.9 模型融合思路

S

总结

团队尝试了四种模型算法的评判，就模型的效果来讲 LightGBM 算法效果最优，其次是 Catboost 算法，相比于这类提升算来说逻辑回归和神经网络的效果就稍逊一筹。与此同时从数据不平衡性的角度来进行考虑，尝试在异常点检测的角度去分析欺诈问题，由于各类样本自身也存在抖动性所以单独采用 IFroest 算法取得的效果并不理想，因而尝试了将异常点检测办法与模型算法跑出来的结果进行投票优化，带来了新的思路。同时考虑到对银行对欺诈检测的时间要求的属性，进而统计了模型的运行时间。真正运行在工业中的模型在保证准确性的前提下，尽可能选择复杂度低的和运行时长短的模型方法。最后非常感谢广发信用卡中心提供的这次比赛学习的机会，也祝广发银行在金融领域乘风破浪，越办越好！

参考文献

[1]陈晓静.我国商业银行信用卡欺诈风险管理研究[D].长沙：中南大学，2011

[2]童凤茹.基于组合分类器的信用卡欺诈识别研究.计算机与信息技术，2006(7):14-16.

[3] Phua C, Alahakoon D, Lee V. Minority report in fraud detection: classification of skewed data. ACM SIGKDD Explorations Newsletter, 2004, 6(1):50-59.

[4] EkremDuman, M, Hamdi Ozcelik. Detecting credit card fraud by genetic algorithm and scatter search.Expert ystems with Applications. 2011,38:13057-13063.

[5] CHAWLA N V, BOWYER K W, HALL L O, et al. SMOTE:Synthetic Minority Over-Sampling Technique [J].Journal of Artificial Intelligence Research,2002,16, 321-357.

[6] HE H, BAI Y, GARCIA E A, et al. ADASYN: adaptive synthetic sampling approach for imbalanced learning[C]//Proceeding of the 2008 International Joint Conference on Neural Networks. Piscataway,NJ: IEEE,2008:1322-1328

[7] FREUND Y, SCHAPIRE R E. Experiments with a new boosting algorithm[C]//Proceedings of the Thirteenth International Conference on Machine Learning. Murray Hill: ICML, 1996:148–156.

[8] CHAWLA N V , LAZAREVIC A, HALL L O ,et al. SMOTEBoost: improving prediction of the minority class in boosting[C]//Proceedings of the 2003 European Conference on Knowledge Discovery in Databases, Europe: Berlin :Springer-Verlag Heidelberg, 2003, 2838 :107-119

[9] SEIFFERT C, TAGHI M K, HULSE JV,et al. RUSBoost: a hybrid approach to alleviating class imbalance [J].IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans,2010,40(1):185-197.