中山大学

手机 APP 的虚假用户识别

——不平衡数据分析

专　业：　　应用统计学
学　号：　　15338109
姓　名：　　李　　元

2018 年 12 月 11 日

目　　录

0 前言

在机器学习常用算法中，都有一个基本假设，那就是数据分布是均匀的。当我们把这些算法直接应用于实际数据时，大多数情况下都无法取得理想的结果。因为实际数据往往分布得很不均匀，都会存在“长尾现象”，也就是所谓的“二八原理”。本文分析的手机 app 虚假用户的数据真实用户数为 199，虚假用户数为 5615，比例为 3.5%，属于严重的数据不平衡问题。

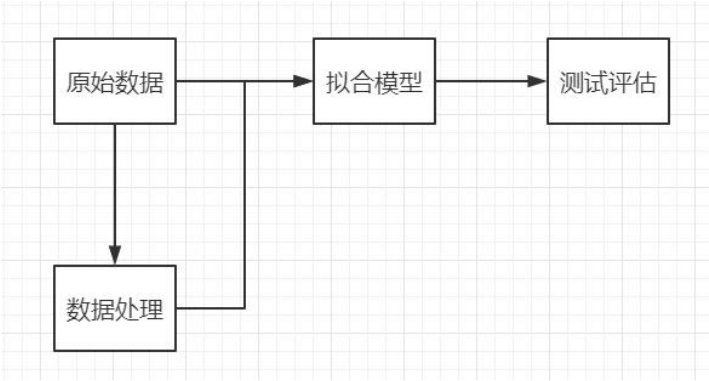


图1

本文首先对数据进行初步探索（第一部分）；然后通过 8 种常用分类算法对原始数据进行探索，对模型进行初步的调参（第二部分）；接着选用四种不平衡抽样的方法对八种模型重新拟合（第三部分）；最后选择出最优的拟合方法以及采样方案（第四部分）。

1 数据探索

原始数据如下表：

ID	X1	X2	X3	...	X20	entrance.type	pages	time	fraud
253	1	0	0	...	0	0	1	12.2	1
254	1	0	0	...	0	1	1	2.555	1
255	1	1	0	...	0	1	2	10.35	1
...	...	...	...	...	...	...	...	...	...
7610	1	1	0	...	0	1	2	52.99	1

图2

变量说明:

变量类型	变量名	详细说明	取值范围
因变量	fraud	响应变量	{0,1}
自变量	X1	X1-X20均为用户行为的属性变量，数值并无实际意义	{0,1,2,...}
	X2		
	...		
	X20		
	entrance.type	进入方式	{0,1}
	pages	浏览页数	0-20
	time	浏览时长	0.371-467.233

图3

对于数据项，我们共有 23 个自变量，对于前 20 个自变量来说：

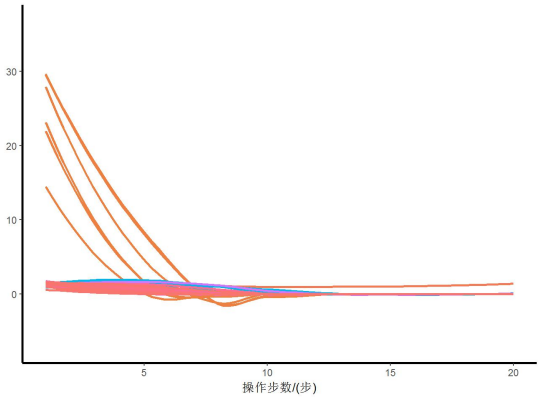


图4

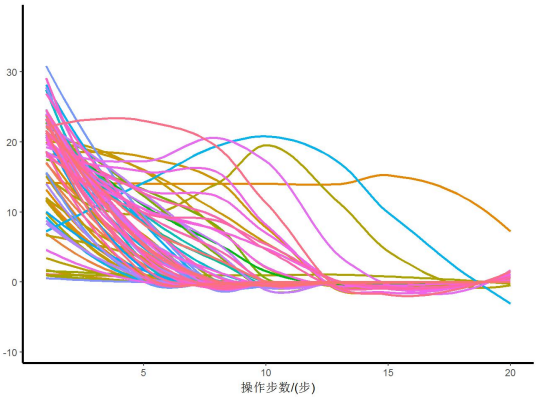


图5

左图为虚假用户的行为，右图为真实用户的行为，可以明显发现自然用户的行为变量的变化明显比虚假用户更为复杂。

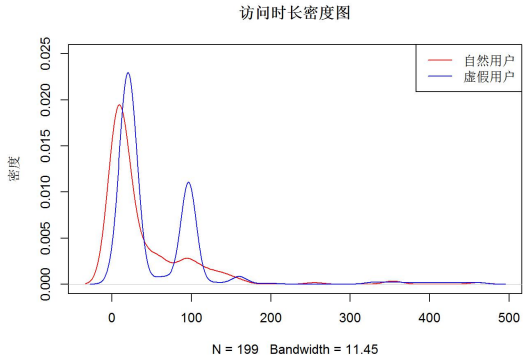


图6

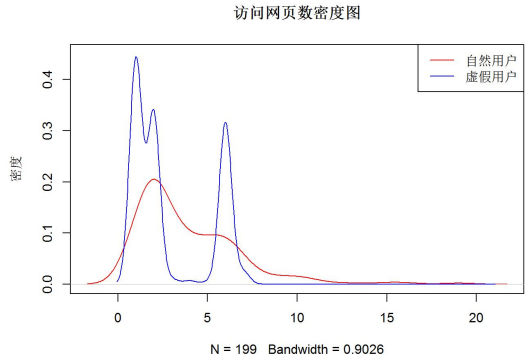


图7

对于访问时长和网页数的密度来看，虚假用户存在明显的双峰，真实用户接近于正态分布。

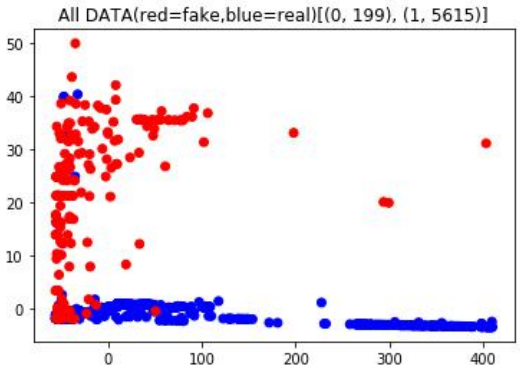


图8

对于整体数据，通过 PCA 降低维后，得到散点图，其中蓝色的点为虚假用户，红色的点为真实用户，可以发现，虚假用户行为非常接近，排列密集；而真是用户行为差异较大，方差较大。

2 数据层面处理

2.1 采样

采样方法是通过对训练集进行处理使其从不平衡的数据集变成平衡的数据集，在大部分情况下会对最终的结果带来提升。

2.1.1 随机过采样(Over-sampling)

随机过采样相当于把小众类复制多份，随机在小众样本中重复抽样，直到两边样本数量相同停止。随机过采样十分接近于在模型中赋予小众样本更大的权重。

优点：速度快，操作简单。缺点：虽然只是简单地将复制后的数据添加到原始数据集中，且某些样本的多个实例都是“并列的”，但这样也可能会导致分类器学习出现过拟合现象，对于同一个样本的多个复本产生多个规则条例，这就使得规则过于具体化；虽然在这种情况下，分类器的训练精度会很高，但在位置样本的分类性能就会非常不理想。

图 9 为随机过采样之后的结果：

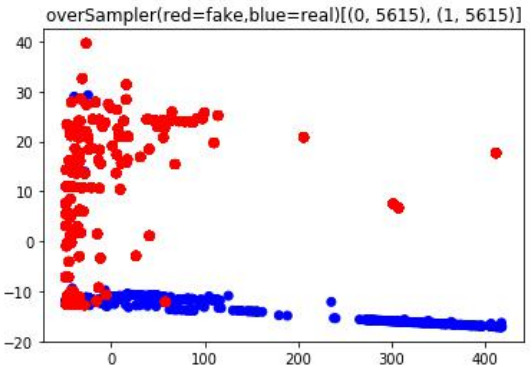


图9

在抽样中，由于抽出的点都完全覆盖在原始点上，所以图像中并不能直观看出新的点的位置。

2.1.2 随机欠采样(Under-sampling)

将大众样本进行随机删除，直到大众样本数量和小众样本数量相同。优点是可以计算量降低。缺点是将多数类样本删除有可能会导致分类器丢失有关多数类的重要信息。 图 10 为随机欠采样之后结果：

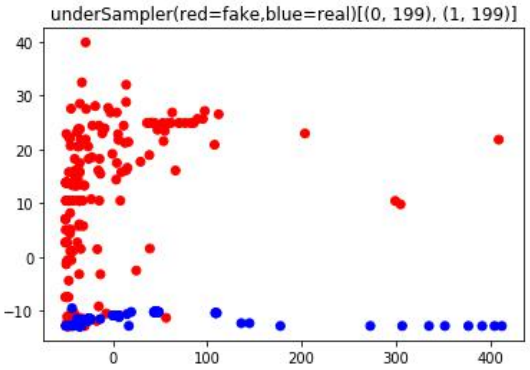


图10

在新的样本中，可以发现蓝色的点大量减少，由于减少为随机的，所以样本保留的可能为异常值，对建模会造成影响。

2.2 生成数据

2.2.1 SMOTE 生成数据随机过采样

Synthetic Minority Oversampling Technique 是常见的生成样本的方法，他的原理是对于少数类样本 a, 随机选择一个最近邻的样本 b, 然后从 a 与 b 的连线上随机选取一个点 c 作为新的少数类样本。通过这种方法可以有效扩充小众样本的样本量，避免了异常值带来的巨大影响，一般优于随机过采样。

优点是有助于简单打破过抽样所产生的关系，使得分类器的学习能力得到显著提高；但是缺点体现在过分泛化问题和方差。

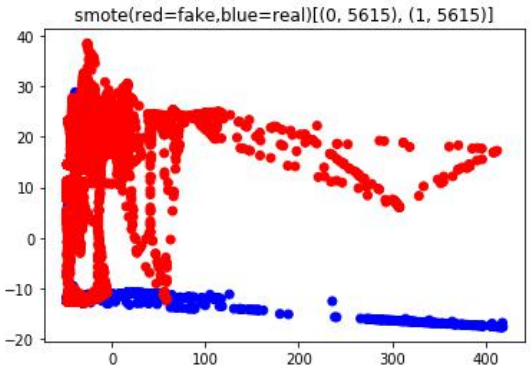


图11

从生成样本中可以发现，从图 11 可以看出，对于红色的小众样本点进行了大量补充。问题在于，对于异常值没有分辨，直接对所有点进行补充，可能会生成更多异常值。

2.2.2 Adaptive Synthetic

ADASYN 是自适应合成抽样方法，解决思路是根据数据分布情况为不同小众样本生成不同数量的新样本。他考虑到了少数样本的邻居数量，然后邻居熟练较多的会有更大的权重，会产生更多的数据点。

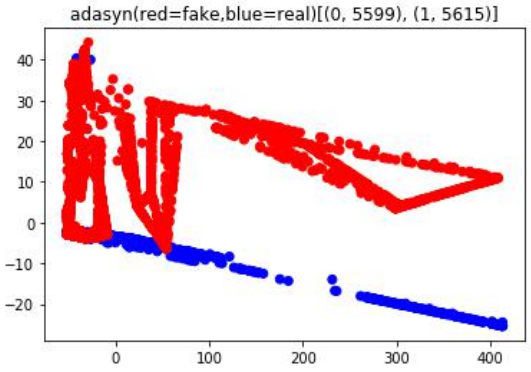


图12

样本生成之后，从图 12 可以看出对于少数有生成新样本，新的样本点还是在小众样本点之间生成，最终样本数量比例接近 1：1。

3 具体模型

接下来的实验中，对原始数据，随机过采样，随机欠采样，SMOTE，Adaptive Synthetic 五组样本分别带入到 9 个模型中。对于测试集和训练集采用 1

比 1 的随机分割，对每一个模型都计算 ACC 准确率，AUC，对于小众样本的撤回率三个指标，来评判样本的拟合情况。

对于过拟合的判断：当对测试集进行计算之后，对训练集进行预测，然后对比训练集的 label 得到新的一组 AUC。在这里我们**定义**一个统计量**过拟合度α：**

$$\alpha = -\ln \frac{AUC_{predict}}{AUC_{train}}$$

当 $AUC_{train} >> AUC_{tpredict}$ 也就是$\alpha > 0$ 的时候，过拟合程度高；当$\alpha > 0$ 或者略大于 0 的时候，不存在过拟合或者过拟合程度低。这样定义的目的是，如果 train 的准确度大于 predict 的准确度，这个程度会被放大。

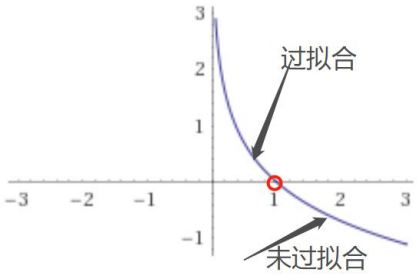


图13

3.1 逻辑回归

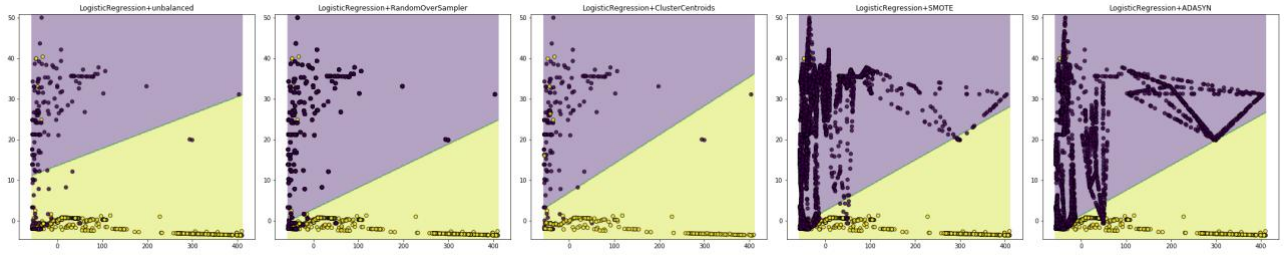


图14

逻辑回归最后模型可以当作线性分类器。从图 14 中第一幅图和其他四幅图的对比，可以明显看出分割线左边有下移趋势，这就说明不平衡的样本之前太多，

导致权重更倾向于大众样本的分对，所以 AUC 在重新抽样之后都有明显提升。

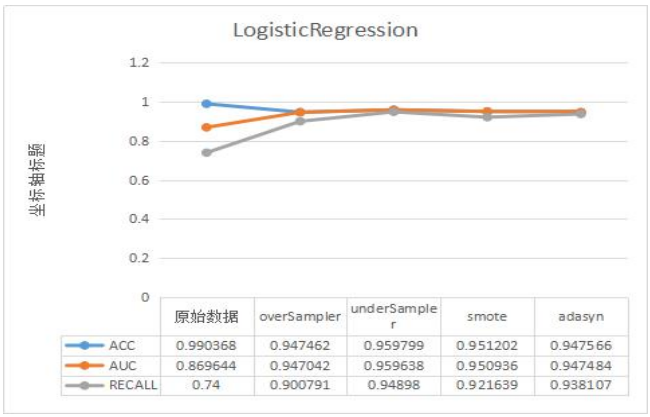


图15

三个评价指标在平衡数据后均有不同程度有所提升。

LogisticRegression	pre_auc	tra_auc	α
origional	0.869644	0.903506	0.038199
overSampler	0.947042	0.938725	-0.008821
underSampler	0.959638	0.984997	0.026082
smotesampler	0.955615	0.957153	0.001608
adasynsampler	0.950719	0.950132	-0.000618

对与α值来说，原始数据较大，存在一定程度的过拟合，但是抽样之后过拟合不存在了。

3.2 K 近邻

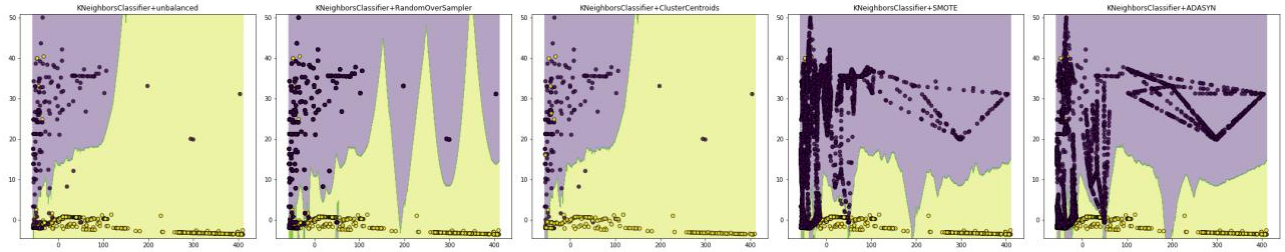


图16

在 K 近邻的分类器中，决策函数的区域随着数据集的变化波动很大。在原始数据中（第一幅），右边的三个异常值没有相邻点，所以直接被忽略；但是在第二幅图过程样中，孤立点重复出现，权重加强，构成了 K 个邻居，所以孤立点周围也变成了小众样本区域。

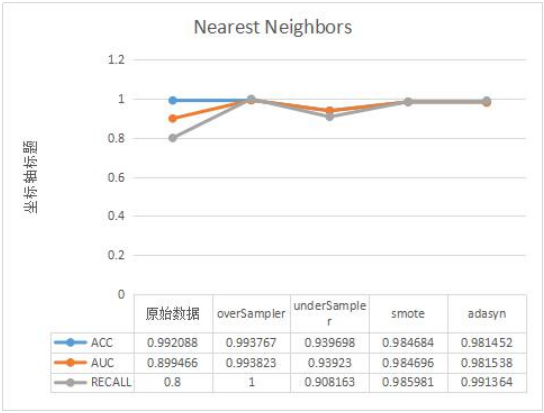


图17

三个评价指标在平衡数据后均有不同程度有所提升，而且准确率很高。KNN算法简单，但是有出色的表现。

Nearest Neighbors	pre_auc	tra_auc	α
origional	0.899466	0.913073	0.015015
overSampler	0.993823	0.995147	0.001332
underSampler	0.93923	0.95004	0.011444
smotesampler	0.986326	0.990526	0.004249
adasynsampler	0.984934	0.989244	0.004366

过拟合判断中可以看出α都十分小，不存在过拟合情况。

3.3 支持向量机

3.3.1 线性支持向量机

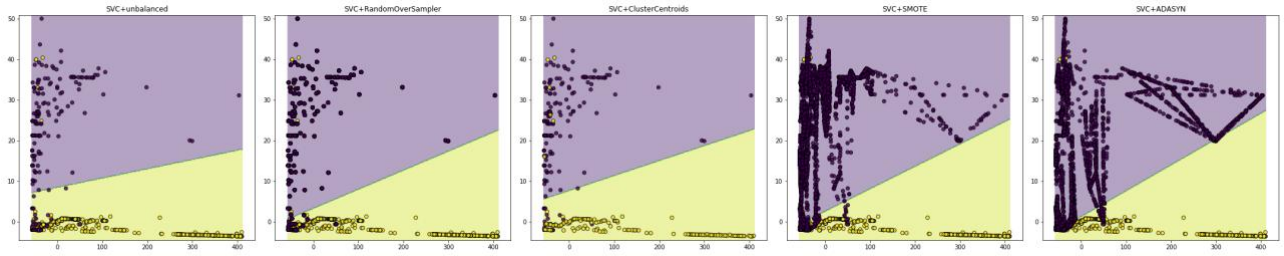
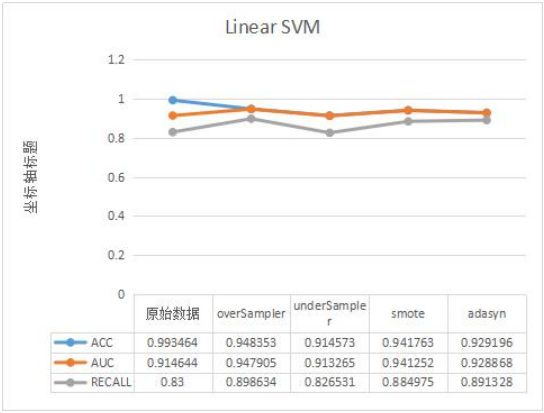


图18

线性支持向量机和逻辑回归分类器的效果十分接近，都属于线性分类，在大样本（2000+）下，决策函数十分接近也不足为奇。



	原始数据	overSampler	underSampler	smote	adasyn
ACC	0.993464	0.948353	0.914573	0.941763	0.929196
AUC	0.914644	0.947905	0.913265	0.941252	0.928868
RECALL	0.83	0.898634	0.826531	0.884975	0.891328

图19

三个评价指标在平衡数据后均有不同程度有所提升，但是即使平衡后撤回率一直略低于其他两个指标，可能后其他深层原因。

Linear SVM	pre_auc	tra_auc	α
origional	0.914644	0.918302	0.003991
overSampler	0.947905	0.938212	-0.010279
underSampler	0.913265	0.940443	0.029324
smotesampler	0.945049	0.941549	-0.00371
adasynsampler	0.931207	0.925529	-0.006116

线性分类器十分简单，α很低，不存在过拟合程度。

3.3.2 RBF 支持向量机

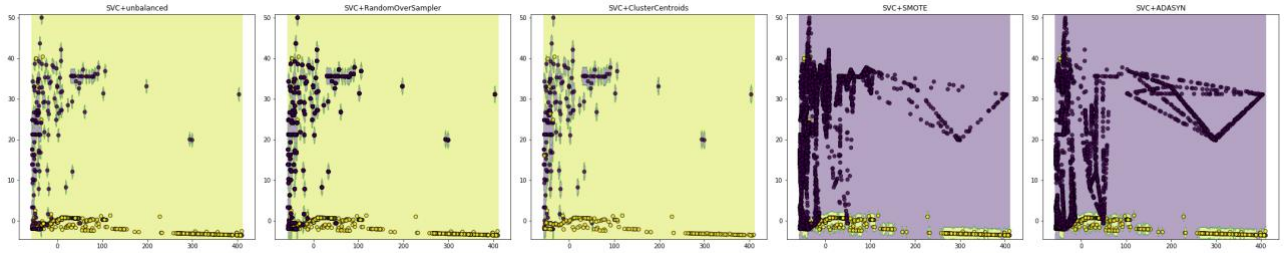


图20

RBF 向量机的核函数是 Radial Based Function，由于不是线性，从图中可以发现一个很有意思的现象：前三幅图决策方式主要将空间分给了大众样本，后两个图主要分给了小众样本。

究其原因主要是因为，在核函数向高维空间投影之后，基于距离的径向基在前三幅图中，小众样本无论数量多少，所占据空间小，决策函数沿着空间边缘分割之后，投影回原空间的大小自然少；相反，在后两种方法生成大量样本之后，新的位置占据了大量空间，所以原来的大众样本空间减少。

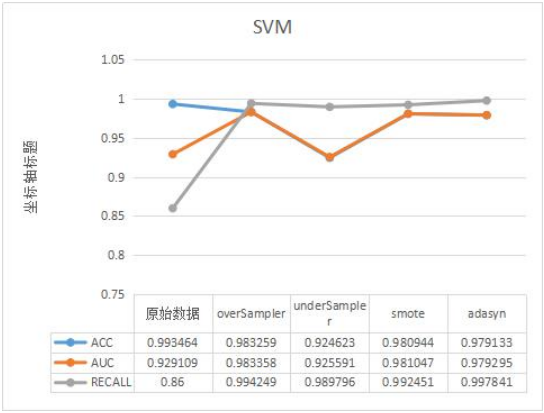


图21

在 5 次模型之中，欠采样对于模型损失极大，原因是模型对于数据拟合程度高，当样本量降低时，模型容易过拟合。

RBF SVM	pre_auc	tra_auc	α
origional	0.929109	0.954545	0.027009
overSampler	0.983358	0.98567	0.002349
underSampler	0.925591	0.979743	0.056858
smotesampler	0.982103	0.985523	0.003476
adasynsampler	0.977512	0.980466	0.003018

从表中我们可以明显看出，第三行，欠采样的α达到了 0.05 十分高，存在过拟合。和我们预期相符，也证明了α定义的合理性。

3.3.3 一分类

对于不平衡数据，我们可以换一个完全不同的角度来看待问题：把它看做一分类（One Class Learning）或异常检测（Novelty Detection）问题。重点不在于捕捉类间的差别，而是为其中一类进行建模，由于时间关系，本文没有进行实验。

3.4 决策树

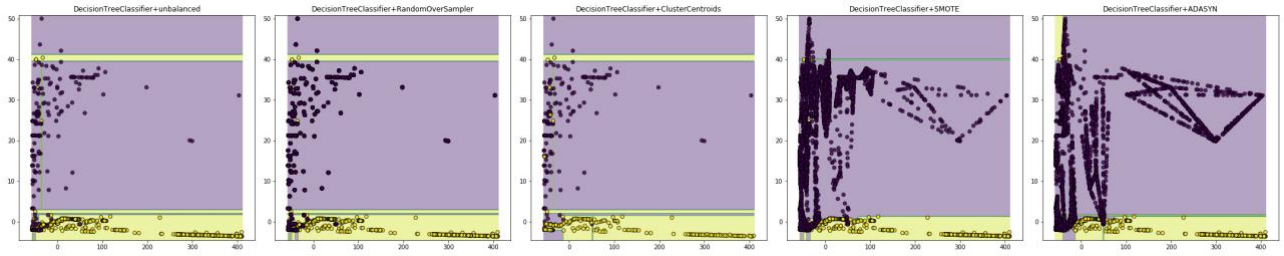


图22

决策树的效果相当于在空间中，把空间用 n 个超平面，把空间分成若干部分，每一部分选择一个属性。（本文 n=5)从图中可以看出拟合效果。

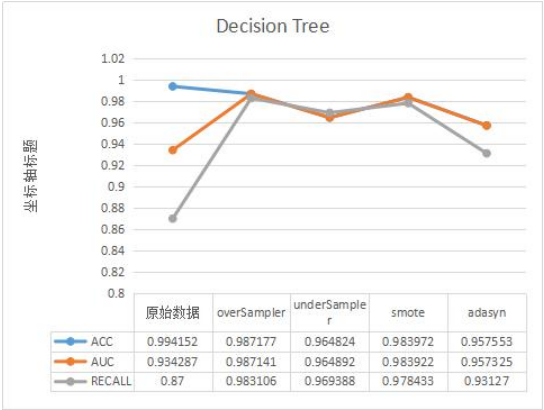


图23

决策树的效果在过采样的时候十分优异，但是在欠采样的时候效果不好，应该是存在过拟合。

Decision Tree	pre_auc	tra_auc	α
origional	0.934466	0.964646	0.031787
overSampler	0.987141	0.987382	0.000244
underSampler	0.964892	0.994898	0.030624
smotesampler	0.983967	0.982898	-0.001087
adasynsampler	0.958579	0.955415	-0.003306

不出所料，欠采样的$\alpha = 0.0306$，存在过拟合。

3.5 Adaboost

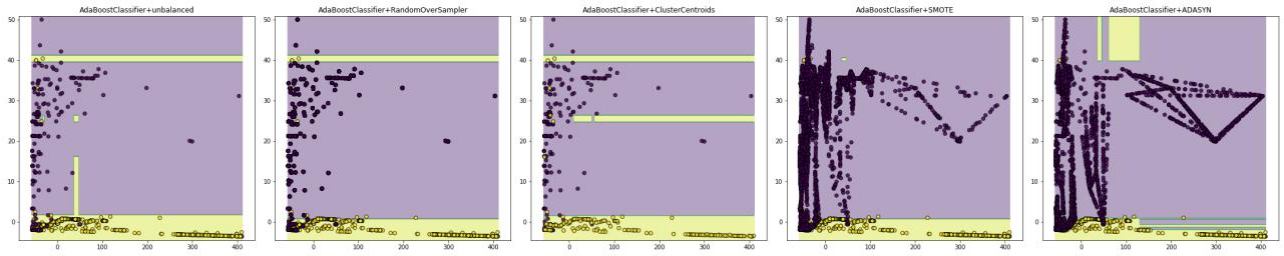


图24

Adaboost 的决策空间相对于决策树来说更细，效果如图。adaboost 整体效果较好，对于决策空间分布较为合理。

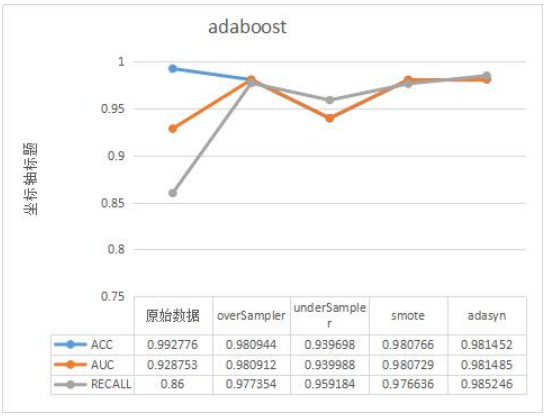


图25

对于α：

AdaBoost	pre_auc	tra_auc	α
origional	0.928753	0.979798	0.053503
overSampler	0.980912	0.98114	0.000233
underSampler	0.939988	1	0.061888
smotesampler	0.982872	0.979362	-0.003578
adasynsampler	0.983448	0.983385	-0.000065

我们可以看出当欠采样时，样本数少，过拟合程度高α达到了 0.061。但是整体来说，adaboost 效果优秀，模型的准确度较为稳定。

3.6 随机森林

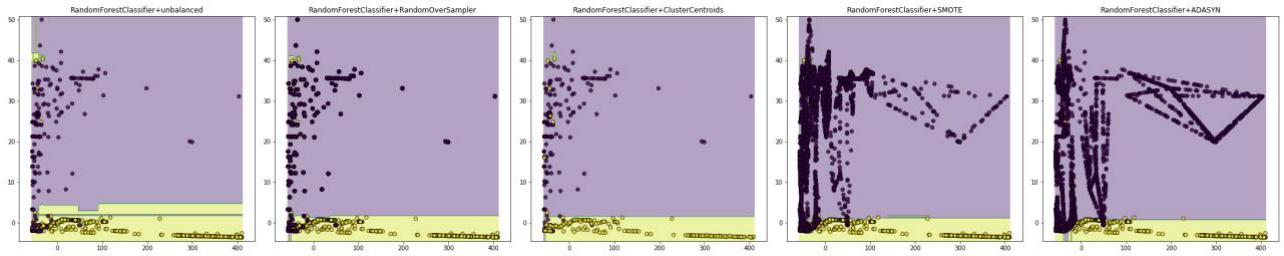
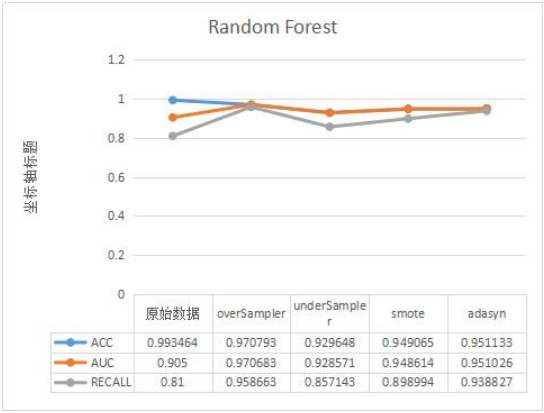


图26

随机森林表现稳定，无论大小样本都可以保存较高的 auc，适合不经过筛选直接使用。



	原始数据	overSampler	underSampler	smote	adasyn
ACC	0.993464	0.970793	0.929648	0.949065	0.951133
AUC	0.905	0.970683	0.928571	0.948614	0.951026
RECALL	0.81	0.958663	0.857143	0.898994	0.938827

图27

对于α：

Random Forest	pre_auc	tra_auc	α
origional	0.915	0.873559	-0.046348
overSampler	0.971036	0.950569	-0.021303
underSampler	0.928571	0.974944	0.048733
smotesampler	0.948614	0.969389	0.021664
adasynsampler	0.940522	0.944817	0.004556

随机森林基本没有过拟合，而且在下采样的时候也保持α较低的水平。原因在于，及时当下采样保留了一些“异常点”，在随机森林随机选择属性建立决策树的时候，只保留了异常点部分信息，大大降低了“异常点”带来的偏差，使得模型不容易过拟合。

3.7 朴素贝叶斯

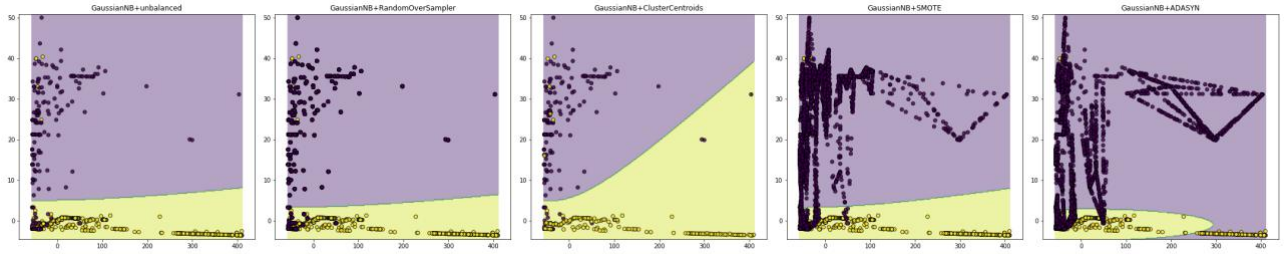


图28

朴素贝叶斯在前四个数据中表现正常，但是在最后一个数据中出现巨大失误，具体原因待研究。

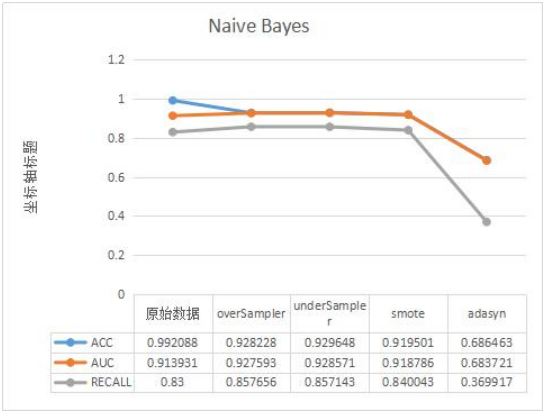


图29

对于α：

Naive Bayes	pre_auc	tra_auc	α
origional	0.913931	0.938326	0.026342
overSampler	0.927593	0.918795	-0.00953
underSampler	0.928571	0.945545	0.018114
smotesampler	0.922381	0.920736	-0.001785
adasynsampler	0.683901	0.675874	-0.011806

没有过拟合情况出现。

3.8 多层感知器分类器

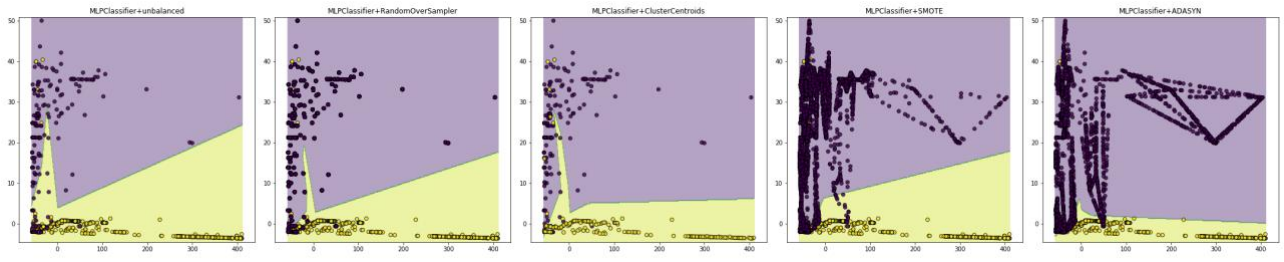


图30

多层感知机在前三组模型表现正常。但是最后两组中，生成的新的样本点（左下角）和另一类混在一起，而模型并不能将其完好的分离，造成 auc 产生了断裂式崩塌，模型使用需要谨慎。

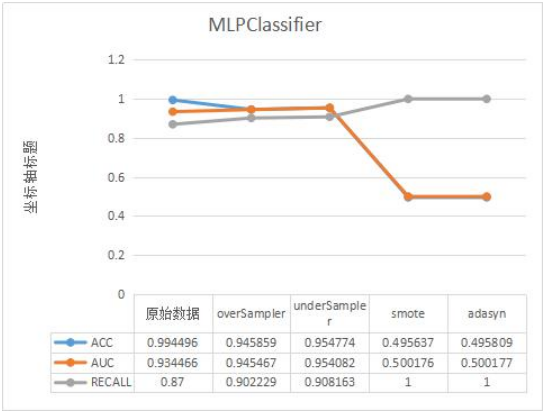


图31

对于α：

MLPClassifier	pre_auc	tra_auc	α
origional	0.934466	0.953655	0.020327
overSampler	0.945467	0.937804	-0.008138
underSampler	0.954082	0.979895	0.026696
smotesampler	0.97495	0.979131	0.004279
adasynsampler	0.5	0.5	0

没有过拟合情况出现。但是对于 adasyn sampler auc 为 0，可能是实验出错造成。

4 整体分析

经过第二部分和第三部分的探索，对 9 个模型和四种抽样方法进行实验，得到 45 组数据。

	ORIGINAL			OVERSAMPLER			UNDERSAMPLER			SMOTE			ADASYN		
Method	ACC	AUC	RECALL	ACC	AUC	RECALL	ACC	AUC	RECALL	ACC	AUC	RECALL	ACC	AUC	RECALL
LogisticRegression	0.990	0.870	0.740	0.947	0.947	0.901	0.960	0.960	0.949	0.951	0.951	0.922	0.948	0.947	0.938
Nearest Neighbors	0.992	0.899	0.800	0.994	0.994	1.000	0.940	0.939	0.908	0.985	0.985	0.986	0.981	0.982	0.991
Linear SVM	0.993	0.915	0.830	0.948	0.948	0.899	0.915	0.913	0.827	0.942	0.941	0.885	0.929	0.929	0.891
RBF SVM	0.993	0.929	0.860	0.983	0.983	0.994	0.925	0.926	0.990	0.981	0.981	0.992	0.979	0.979	0.998
Decision Tree	0.994	0.934	0.870	0.987	0.987	0.983	0.965	0.965	0.969	0.984	0.984	0.978	0.958	0.957	0.931
Random Forest	0.993	0.905	0.810	0.971	0.971	0.959	0.930	0.929	0.857	0.949	0.949	0.899	0.951	0.951	0.939
AdaBoost	0.993	0.929	0.860	0.981	0.981	0.977	0.940	0.940	0.959	0.981	0.981	0.977	0.981	0.981	0.985
Naive Bayes	0.992	0.914	0.830	0.928	0.928	0.858	0.930	0.929	0.857	0.920	0.919	0.840	0.686	0.684	0.370
MLPClassifier	0.994	0.934	0.870	0.946	0.945	0.902	0.955	0.954	0.908	0.496	0.500	1.000	0.496	0.500	1.000

4.1 最优模型

对于所有模型的 auc 进行直接比较，我们可以发现 Adaboost 和 KNN 效果最好，AUC 普遍高于其他模型。

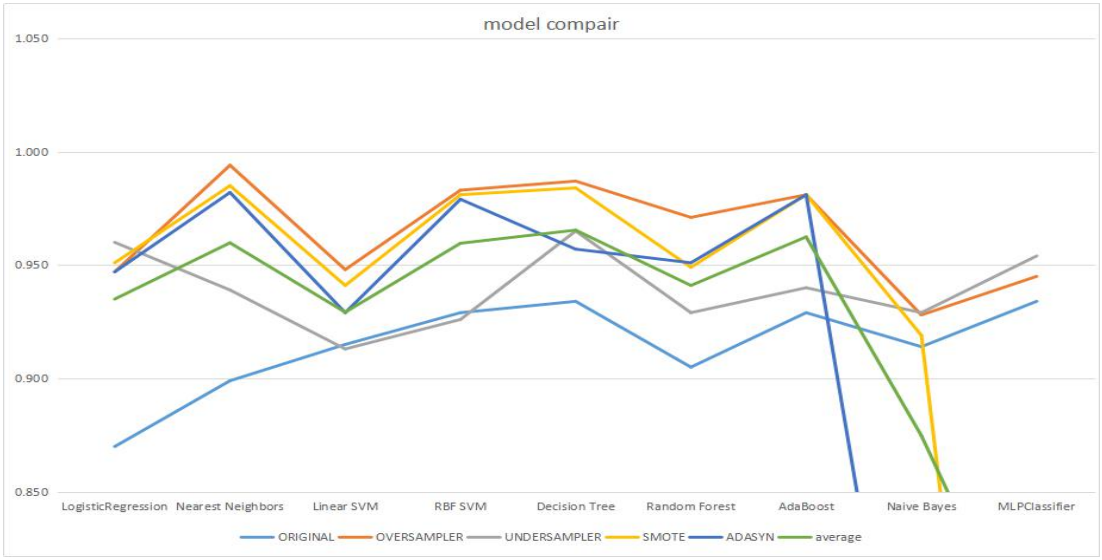


图32

4.2 最优采样方法

排除掉的异常情况，随机过采样和 SMOTE 效果普遍优于随机欠采样；平衡后的结果普遍好于不平衡的结果。

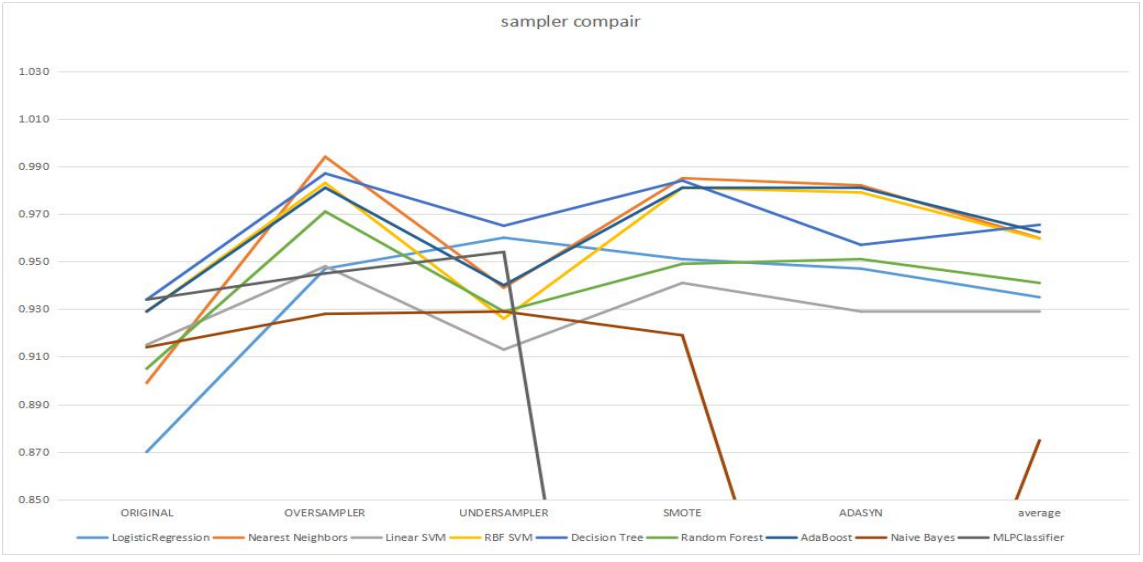


图33

5 结论

从上文中，我们可以看出，不平衡的数据对于模型的影响很大，AUC 损失较多；对于特殊情况，如银行失信者预测，医院疾病诊断的时候，小众类别的损失对模型更为严重，所以在我们建模的过程中一定要对其有所处理。

对于数据：可以对数据进行补充，或者删除。补充包含随机过采样，SMOTE, adaptive synthetic 等算法对于小众样本进行补充来提高其占比，效果较好；但是删除样本就会损失较大量信息，不建议主要使用。

对于模型：我们可以改变小众样本的权重，让小众样本在损失函数中体现更大，在训练模型的时候更注重小众样本的撤回率；也可以采用普适的算法，如随机森林和 adaboost，由于模型本身就含有很多随机成分，过拟合程度不高，对于小众样本也可以很好的预测。

整体来说，采样方法建议对小众样本点数据扩充，模型建议使用随机森林和 adaboost 对于不平衡数据有更好的效果。

附录：

数据和代码：https://github.com/liyuan97/fuzashujufenxi