|  |  |
| --- | --- |
| 队伍编号 | MCB220XXXX |
| 赛道 | （A/B） |

论文题目

摘 要

（摘要正文）

关键词：

目录

键入章级(第 1 级) 1

键入章级(第 2 级) 2

键入章标题(第 3 级) 3

键入章级(第 1 级) 4

键入章级(第 2 级) 5

键入章标题(第 3 级) 6

（这里开始论文正文）

1 语音数据分析

1.1 数据处理思路

**· 数据导入**

导入“附件1语音业务用户满意度数据”，数据共有5433个观测，55个变量。

**· 缺失值处理**

按照附件五的说明，我们对缺失值做以下处理：

1. 填补‘是否关怀用户’，‘是否去过营业厅’字段中的缺失值为“否”。

2. 删除缺失值大于等于5个的观测。

3. 填补‘外省流量占比’字段的缺失值为该字段中位数。

4. 为了简化数据分析，我们对'其他，请注明','用户描述','其他，请注明.1','用户描述.1','终端品牌类型','用户id'，'重定向次数','重定向驻留时长'等缺失率达90%以上的变量采取删除策略。

**· 数据重编码**

1. ‘是否遇到过网络问题’字段中的2，重编码为0，表示没有遇到网络问题。

2. '居民小区', '办公室', '高校', '商业街', '地铁', '农村', '高铁', '手机没有信号', '有信号无法拨通','通话过程中突然中断', '通话中有杂音、听不清、断断续续', '串线', '通话过程中一方听不见' 字段中的-1，重编码为0，其他数值，重编码为1,表示该场景或该问题曾经发生过。

3. '4\\5G用户'字段中，将“2G”重编码为0；“4G”重编码为1；“5G”重编码为2。表示网络制式逐级递增。

4. '语音方式'字段中的'VoLTE'，重新改写为'VOLTE'，此处应该是录入失误，二者并无差别。

5. '是否关怀用户','是否去过营业厅','是否4G网络客户（本地剔除物联网）', '是否5G网络客户', '是否实名登记用户'字段中的“是”，重编码为1；“否”重编码为0。

6. 考虑终端品牌市场占有率，‘终端品牌’字段重编码为“苹果”、“华为”、“小米”和“其他”四类。

7. 查阅北京移动客户分级相关资料，‘客户星级标识’字段重分类为（未评级准星一星二星）/（三星）/（银卡）/(金卡，白金卡，钻石卡)，并且分别赋值1，2，3，4，表示等级由低到高。

8. 最后使用独热编码处理'语音方式','终端品牌'字段。

**· 异常值处理**

定量变量字段有：'套外流量（MB）','套外流量费（元）', '外省语音占比', '外省流量占比','语音通话-时长（分钟）', '省际漫游-时长（分钟）','ARPU（家庭宽带）','当月ARPU', '当月MOU', '前3月ARPU', '前3月MOU','GPRS总流量（KB）', 'GPRS-国内漫游-流量（KB）','当月欠费金额', '前第3个月欠费金额'，考虑处理如下：

1. 画出定量变量箱线图，直方图，发现16个数值变量中大多数为0，呈现右偏分布，存在异常值，但是考虑实际业务场景中通话时长较长或者上网较多的用户相比于一般用户更能够从中通话中发现问题，因此综合考虑，采用截尾方法处理，将大的数据标记为截尾,替换对应数值为列均值加三倍标准差。

次数变量字段有：'脱网次数','mos质差次数', '未接通掉话次数'，考虑处理如下：

1. 对每一个变量使用K均值聚类，目标类数目设定为5，将次数变量离散化处理，保证了极端值包含的信息不缺失。

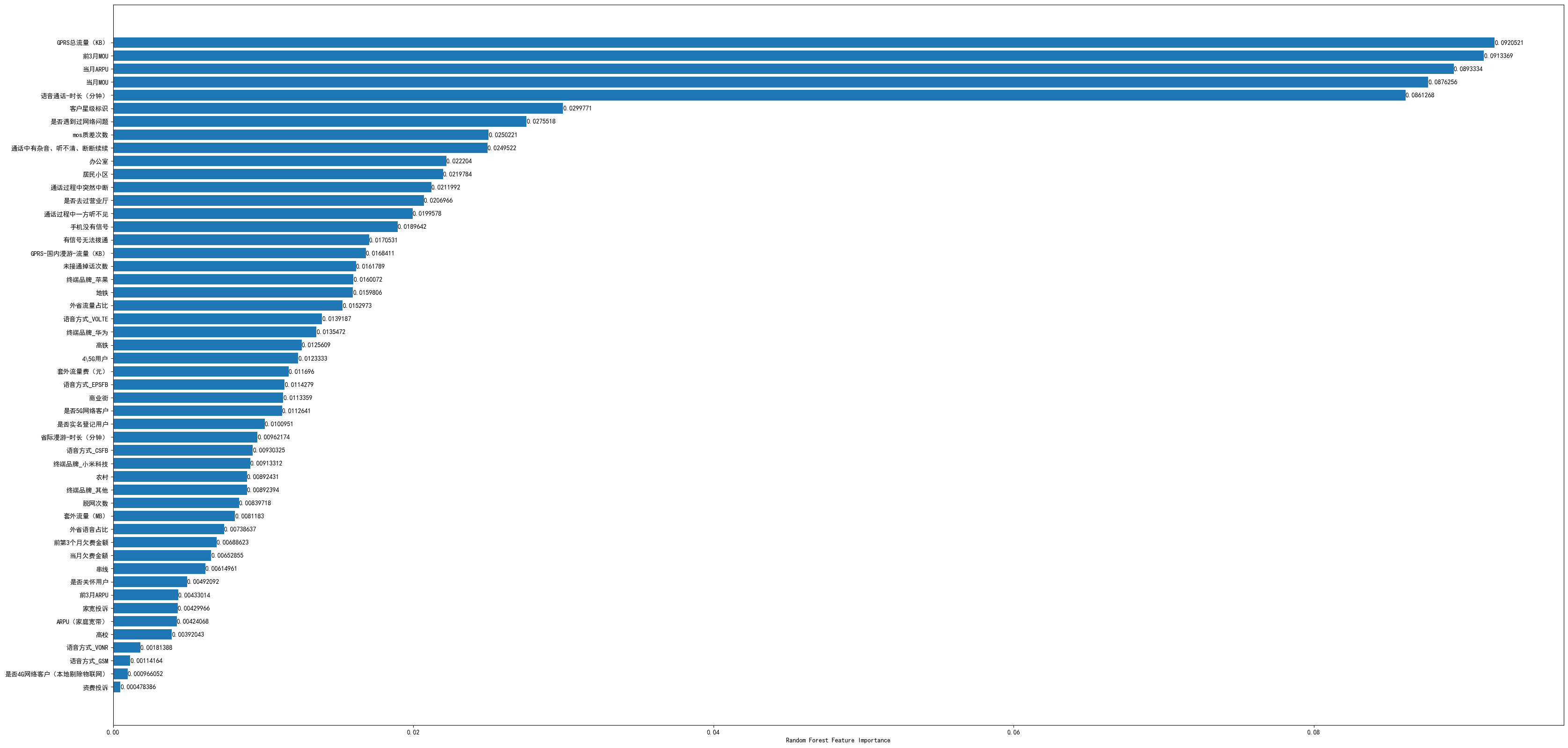
**1.2 数据分析**

**求解问题一：**研究影响语音业务的满意度的主要因素，并给出各因素对客户打分影响程度的量化分析和结果。

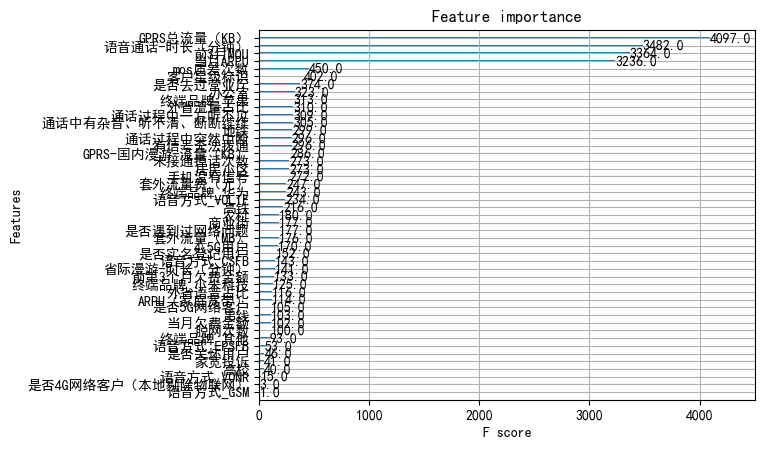
本文选择随机森林、GBDT、XGBoost，BP神经网络、SVM等机器学习算法作为备选，逐一带入训练集进行训练，各机器学习算法的表现如下表所示：

|  |  |
| --- | --- |
| 算法名称 | 分类准确度 |
| 随机森林 | 0.567 |
| GBDT | 0.556 |
| XGBoost | 0.554 |
| BP神经网络 | 0.421 |
| SVM | 0.435 |

本文选择其中表现最好的随机森林算法作为训练器，得出训练集各特征重要性排序如下图所示：

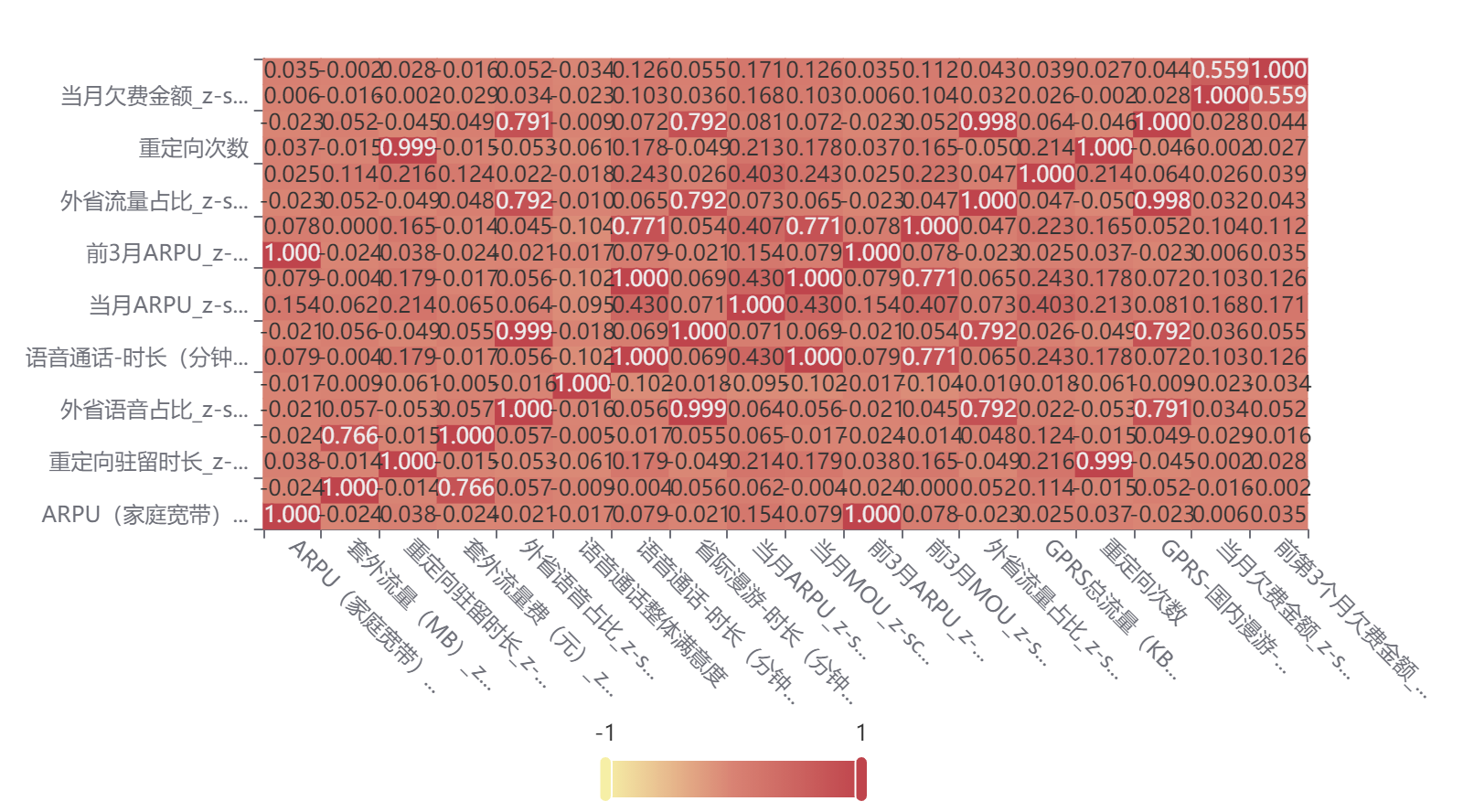


同时使用XGBoost分类器进行训练，得出的特征重要性排序如下：

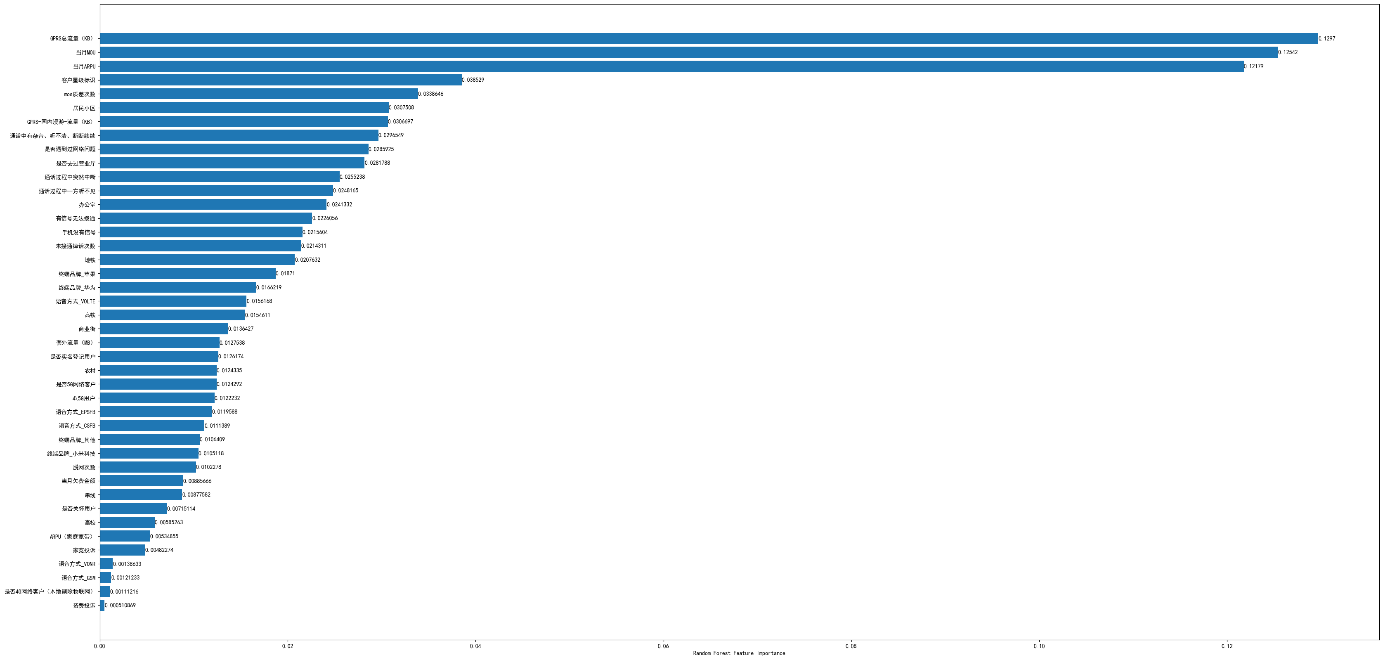


可以看出尽管分类器不同，但是变量重要性排名大致相同，影响客户打分的最重要三个变量为GPRS总流量（KB），前三月MOU和当月ARPU，这些指标分别反映了客户的流量使用情况，通话时长和当月的消费金额，也就是说，客户的使用移动公司服务的时长和月消费金额是影响客户满意度的最主要指标。

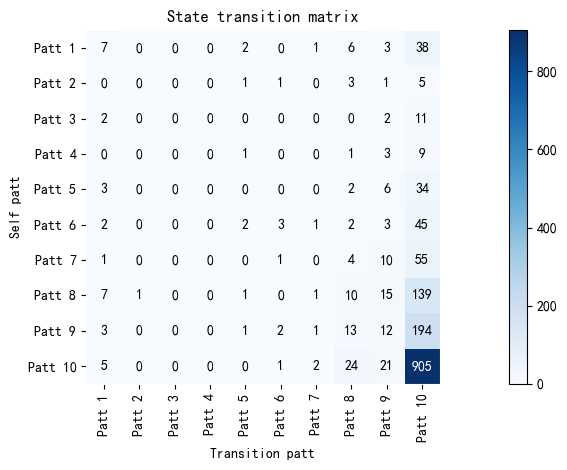
与此同时，原始特征之间存在较强的相关关系，可能会放大用户使用移动公司产品的特征对用户满意度的影响，我们使用Persion和Spearman等级相关系数检验的方法，对原始特征之间的强相关关系进行检验，剔除强相关变量，相关系数热力图如下：



经过检验，定量数据中总共存在五组强线性相关关系，分别为第一组：当月MOU，前三月MOU，语音通话-时长（分钟）；第二组：GPRS-国内漫游-流量（KB），外省流量占比，外省语音占比，省际漫游-时长（分钟）；第三组：当月欠费金额，前三月欠费金额；第四组：套外流量（MB），套外流量费（元）；第五组：当月ARPU，前三月ARPU。本文选择取每组中的第一个变量纳入模型，将每组剩余变量进行剔除，重新进行变量重要性排序，结果如下：



删去线性相关特征之后的特征重要性前三名依旧是GPRS总流量，当月MOU和当月ARPU，在所有场景类别中，最重要的变量是居民小区，说明人们对于小区内的通话质量最为在意，居民小区内的语音业务服务相比其他场景的语音服务更能影响客户满意度。在所遇到的各类问题中，“通话中有杂音，听不清，断断续续”对客户满意度影响最大，其他情况依次是：“通话过程中一方中断”，“通话过程中一方听不见”，“有信号无法拨通”和“手机没有信号”。特征“是否去过营业厅”重要性排名第十，说明营业厅是影响客户打分的一个重要变量，可能的解释是：去过营业厅的客户可能能够得到更细致的服务，这会对他的满意度打分产生影响。最后模型的混淆矩阵为：



**求解问题二：**结合问题 1 的分析，对于客户语音业务建立客户打分基于相关影响因素的数学模型，并据此对附件 3中的客户打分进行预测研究。

基于问题一的分析开展特征工程：以去除线性相关变量的模型为基准模型，分类准确率为：0.576427255985267

**· 特征对齐**

由于附件1和附件3中的特征变量并不相同，也就意味着训练集和测试集存在特征多余或者特征缺失，本文首先进行特征对齐：

1. 针对测试集中存在，但训练集中不存在的变量‘性别', '是否不限量套餐到达用户'进行删除。 '是否投诉'变量保留，同时将训练集中的‘家宽投诉’和‘资费投诉’变量合并为‘是否投诉’变量进行训练。

2. 针对训练集中存在，但测试集中不存在的变量'ARPU（家庭宽带）','前第3个月欠费金额','当月欠费金额', '是否去过营业厅', '是否实名登记用户','语音方式','重定向次数','重定向驻留时长'进行删除。

**· 特征衍生**

基于对其后的特征分别构建问题特征和用户特征。

1. 构建问题特征方法如下：

新增特征'网络覆盖范围问题'='居民小区'+'办公室'+'高校'+'商业街'+'地铁'+'农村'+'高铁'；

新增特征'信号强度问题'='有信号无法拨通'+'手机没有信号'+'脱网次数'；

新增特征'通话清晰度问题'='通话中有杂音、听不清、断断续续'+'通话过程中一方听不见'+'mos质差次数'；

新增特征'通话稳定性问题'='通话过程中突然中断'+'串线'+'未接通掉话次数'；

2. 构建用户特征的方法如下：

（1）找出所有的分类特征和数量特征

（2）将训练集数据分别按照分类特征逐一分类汇总，求出每个数量特征的最大值，最小值，平均数及标准差。

（3）将新生成的特征和原始特征拼接形成新的训练集，新训练集共有235个特征。

**· 特征筛选**

上述衍生方法衍生出的部分特征重要性程度较低，采用下述方法进行剔除：

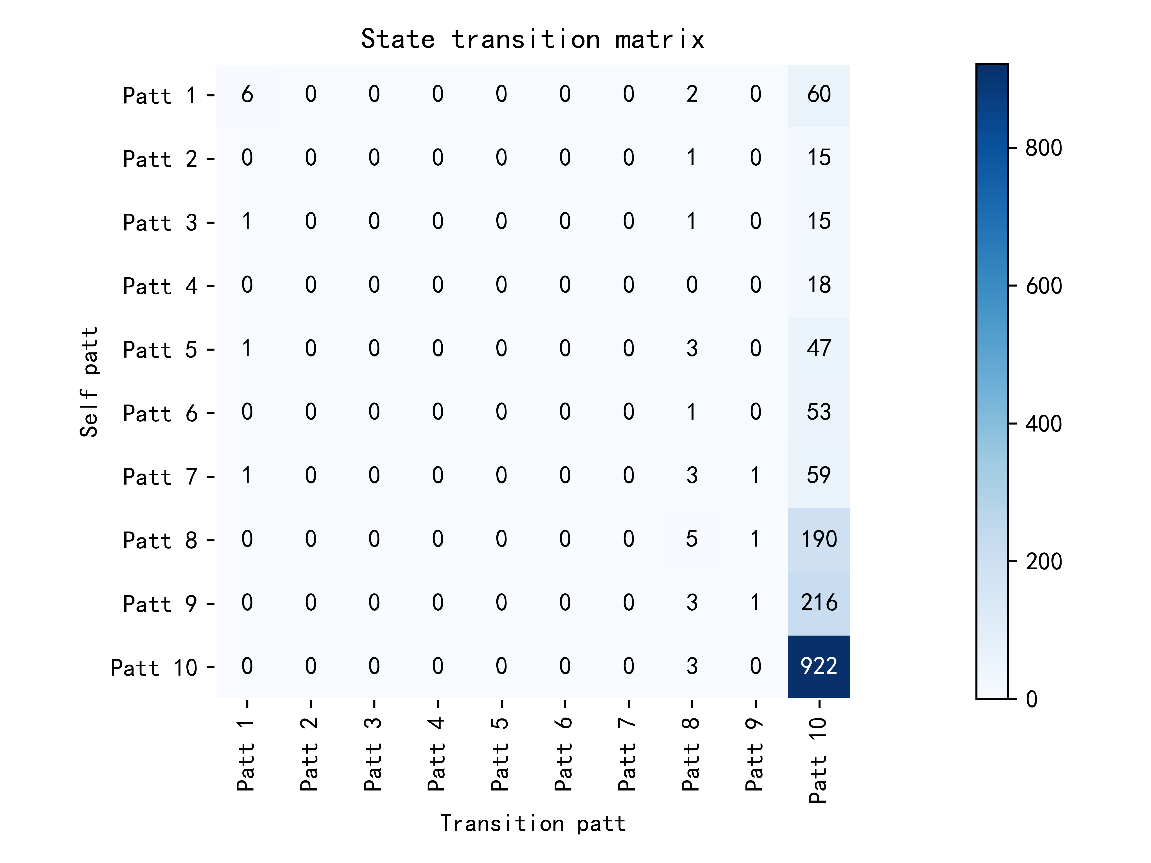
（1）首先剔除所有重要性小于0.003的变量。

（2）其次采用RFE再筛去一半特征。

筛选完成后，训练集包含19个特征为：是否遇到过网络问题, 居民小区, 办公室, 地铁, 手机没有信号, 通话中有杂音、听不清、断断续续,通话过程中一方听不见,mos质差次数, 当月ARPU, 当月MOU,GPRS总流量（KB）,GPRS-国内漫游-流量（KB）, 客户星级标识, 网络覆盖范围问题, 信号强度问题, 通话清晰度问题,通话稳定性问题, 套外流量（MB）\_客户星级标识\_std, 当月MOU\_客户星级标识\_mean。

**· 参数优化**

随机森林分类器重要的参数有：max\_depth, max\_samples, min\_samples\_leaf, min\_samples\_split, n\_estimators,max\_features。使用网格搜索交叉验证方法寻找最优参数结果：criterion='entropy', max\_depth=15, max\_features=2,max\_samples=0.9, min\_samples\_leaf=7, min\_samples\_split=4, n\_estimators=150。得到最优参数下的分类器准确率为：0.5922609389085361。基于最优参数下的随机森林分类器，得到附件3中客户打分的满意度预测数据。最后模型的混淆矩阵为：



2上网数据分析

2.1数据处理思路

· 数据补全

· 根据附件5，指定列（'上网质差次数','脱网次数','重定向次数','2G驻留时长','微信质差次数','王者荣耀质差次数','游戏类APP使用流量','今日头条使用流量','快手使用流量','优酷视频使用流量','腾讯视频使用流量'）空值补全为每列的众数

· 缺失值处理

· 剔除缺失>50%的变量（'场景备注数据','现象备注数据','APP大类备注','APP小类视频备注','APP小类游戏备注','APP小类上网备注','码号资源-激活时间','码号资源-发卡时间'])

· 表示重复信息的变量留一（'操作系统','终端品牌类型','畅享套餐名称'）

· 4个数值型变量（'王者荣耀使用次数','游戏类APP使用天数','游戏类APP使用次数','王者荣耀使用天数'）同步缺失的样本包含414条,使用众数填充

· 4个分类型变量存在<6%的缺失（'当月高频通信分公司','终端品牌','终端类型','近3个月平均消费（剔除通信账户支付'）），按照完全随机缺失处理，直接剔除(共剔除30条样本，剔除量<5%)

· '终端类型' 指标无效，剔除

· 分类型变量重新编码

· '终端品牌' 重分类为5类：苹果，华为，小米，步步高，其他

· '当月高频通信分公司' 重分类为2类：城区分公司 ，其他分公司

· '客户星级标识' 重分类为4类：A，B，C，D （与语音数据保持一致）

· 一致化处理

· 二分类变量 '是','否' 改为 '0','1'

· 分类型数据根据附件5重整为 '0','1'编码 （'居民小区','办公室',etc. ）

· 数值型变量的异常值处理

· 异常值截尾

· 数值型变量标准化

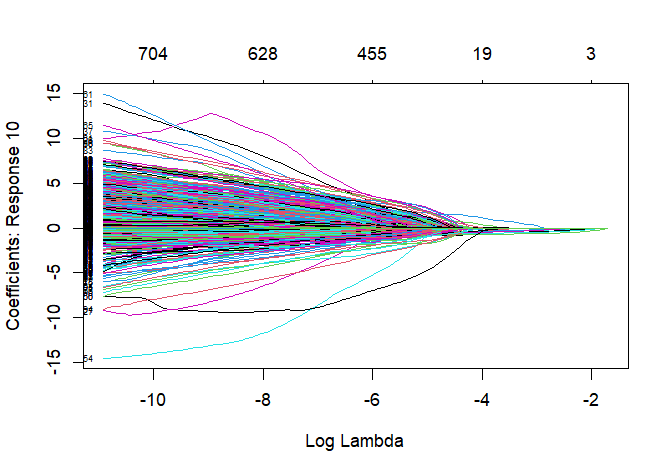
· 分类型变量独热操作

2.2数据处理

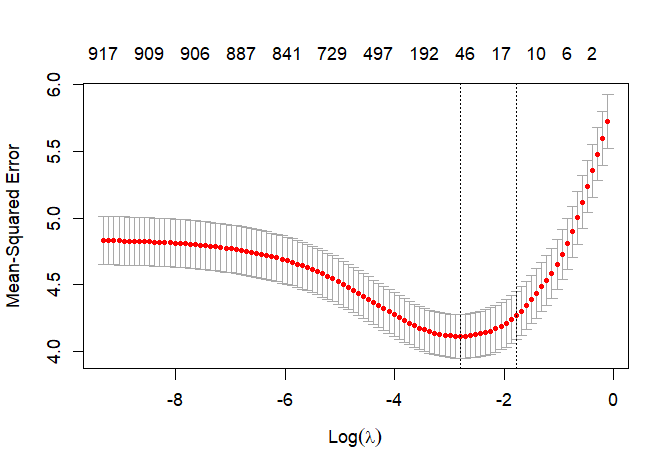
2.3数据分析

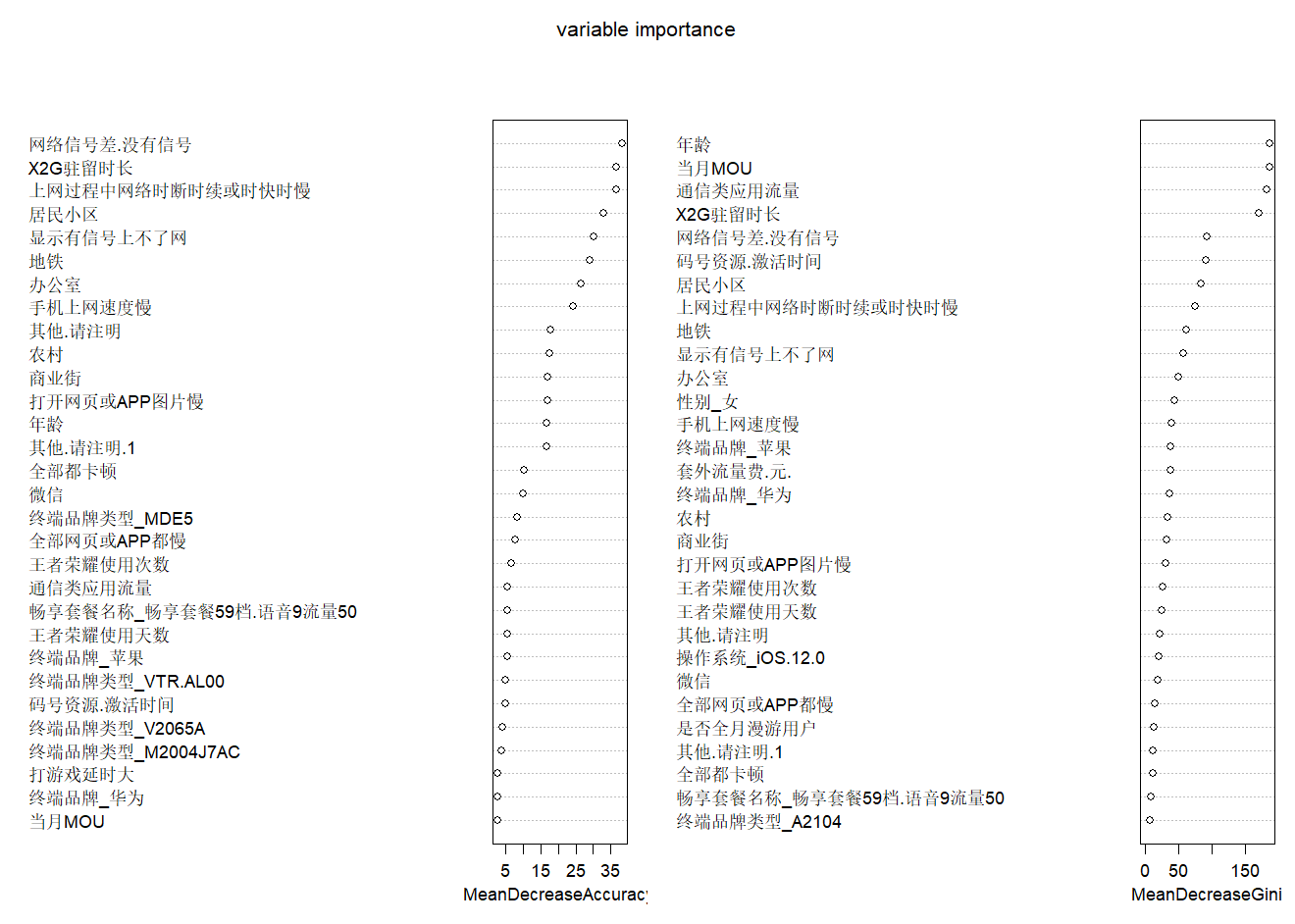
2.3.1Lasso

|  |
| --- |
| data=read.csv("D:/He/VScode/files/其他/Mathorcup/清洗后上网数据.csv")  data=data[,-1]  data\_st=data[,c(-2,-3,-4)]  range01=function(x){(x-min(x))/(max(x)-min(x))}  for (j in 2:length(data\_st[1,])){  if(sum(data\_st[,j]==data\_st[1,j])==length(data\_st[,j])){cat('第',j,'列\n')}  else{data\_st[,j]=range01(data\_st[,j])}  }  library(glmnet)  model=glmnet(data\_st[,-1],data\_st[,1],nlambda=100,family='multinomial',alpha=1)  plot(model,xvar="lambda",label=TRUE)  cvfit=cv.glmnet(as.matrix(data\_st[,-1]),data\_st[,1])  plot(cvfit)  lambda\_min=cvfit$lambda.min #求出最小值  #lambda\_1se=cvfit$lambda.1se #求出最小值一个标准误的λ值  mod1<-coef(cvfit$glmnet.fit,s=lambda\_min,exact = F)  #mod2<-coef(cvfit$glmnet.fit,s=lambda\_1se,exact = F)  #随机森林验证  library(randomForest)  library(pROC)  data\_rf=data\_st  dex=vector()  for (i in 2:length(data\_rf[1,])){  if(mod1[i]!=0){  dex=append(dex,i)}  }  data\_rf1=data\_rf[,dex]  write.csv(data\_rf1,"D:/He/VScode/files/其他/Mathorcup/lasso筛选后的数据.csv")  y\_rf=factor(data[,1])  #80%样例作为训练集,其余作为测试集  sub=sample(1:length(data\_rf1[,1]),0.8\*length(data\_rf1[,1]))  train=data\_rf1[sub,]  test=data\_rf1[-sub,]  model\_rf<-randomForest(train,y\_rf[sub],importance=TRUE,proximity=TRUE)  importance(model\_rf,type=2)  varImpPlot(model\_rf,main="variable importance")#变量的重要性判断  #对测试集进行预测  pre\_rf=predict(model\_rf,newdata=test)  #输出混淆矩阵  table(y\_rf[-sub],pre\_rf,dnn=c("真实值","预测值"))  #准确率  mean(pre\_rf==y\_rf[-sub])  #绘制ROC曲线  pre\_rf\_roc=predict(model\_rf,newdata=test,type='prob')  rf\_roc=multiclass.roc(y\_rf[-sub],pre\_rf\_roc[,1])  plot(rf\_roc$rocs[[1]],col=1,print.auc=TRUE,auc.polygon=TRUE,max.auc.polygon=TRUE,  auc.polygon.col=2)  for (j in 2:10){  #rf\_roc=multiclass.roc(y\_rf[-sub],pre\_rf\_roc[,j])  plot(rf\_roc$rocs[[j]],add=T,col=j)  } |

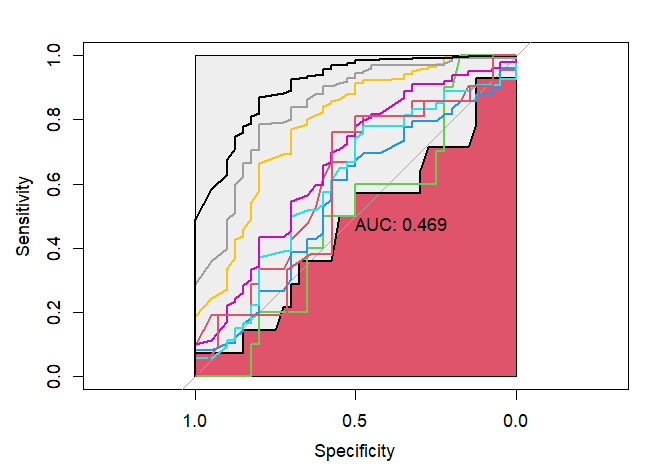


根据上图，可以发现，大部分变量在logλ趋近-4时被压缩到趋近于0，因此可以判断很多变量对于模型的贡献是重复的，进行特征选择是十分必要的，接下来绘制均方误差图来找到最优的λ。

根据均方误差图，第一条虚线所示即为均方误差最小的位置。



随机森林变量重要性排序中，分别通过mean decrease accuracy（随机森林预测准确性的降低程度）和mean decrease gini（每个变量对分类树每个节点上观测值的异质性的影响）来判断变量重要性，越大表示重要性越大，结果如图所示，两个指标的前三名依次为：’网络信号差.没有信号’、’X2G驻留时长’、’上网过程中网络时断时续或时快时慢’和’年龄’、’当月MOU’、’通信类应用流量’。虽然名次不同，但是可以发现通过两种指标被保留的变量基本相同，因此可以相信结果可靠。





由于我们最关心的低分的情况，我们绘制1分的ROC曲线图，在此基础上绘制另外9类的曲线，通过ROC图，可以发现评分为1分的时候AUC为0.469，与随机森林准确率基本吻合。

4.上网打分预测

4.1数据处理思路

· ‘用户id’仅起到序号的作用，对模型的拟合不起作用，因此直接剔除。

· ‘是否遇到网络问题’本质为0-1变量，但是经过观察，该变量的取值取决于 ‘居民小区’到’全部网页或APP都慢’，只要他们其中一个为1，那么’是否遇到网络问题’就为1，因此选择删除该变量，因为后面所有的变量可以表示该变量。

· 所有’注明内容’删除，因为发现这些列的内容极少，且无法通过填充的手段对其内容进行填充。

· 从’居民小区’到’全部网页或APP都慢’，将所有为-1的值用0覆盖，其他值都用1覆盖，至此将这些变量都转化成0-1变量的形式。

· ‘学习强国’出现两次，为重复出现的变量，因此删除掉其中一列，再对比附件2，发现在原来的表格中并未统计’学习强国’的情况，因此选择将’学习强国’和’其他，请注明’合并，使用or语句，只要任一变量的值为1，合并列的值为1，反之为0。

· 对比附件2，发现’上网质差次数’和’脱网次数’位置错误，将两者调换位置。

· 使用excel筛选功能查看’是否投诉’选择是的用户，发现仅有一人，该列数据对模型的拟合起不到重要作用，因此该列直接删除。

· ‘是否关怀用户’、’外省语音占比’、’语音通话-时长（分钟）’、’省际漫游-时长（分钟）’、’终端品牌类型’、’外省流量占比’、’GPRS-国内漫游-流量（KB）’和’是否不限量套餐到达用户’列删除。因为’是否关怀用户’、’外省流量占比’、’GPRS-国内漫游-流量（KB）’和’是否不限量套餐到达用户’无法从前面的变量中获知，同时对比附件2也无法判断，因此删除；’外省语音占比’、’语音通话-时长（分钟）’和’省际漫游-时长（分钟）’与语音通话相关，因此考虑和语音数据结合来看，但是发现在附件5中对用户序号的说明中解释每个表的用户序号没有一一对应的关系，因此无法结合起来看，于是将三个变量删除；’终端品牌类型’在前文已经说明，该变量重要性低，因此选择直接删除该变量。

· 经过对附件4的观察以及筛选操作，发现’4\5G用户’、’是否4G网络客户（本地剔除物联网）’和’是否5G网络客户’三个变量是完全正相关的，只要该用户是5G用户，那么一定是4G网络客户（本地剔除物联网）并且是5G网络客户；若该用户是4G用户，那么一定是4G网络客户（本地剔除物联网）并且不是5G网络客户。因此选择保留’4\5G用户’一个变量即可。

· ‘套外流量（MB）’和’套外流量费（元）’变量位置和附件2中位置不一致，调整到相对位置。

· ‘GPRS-国内漫游-流量（KB）’在附件2中对应变量’当月GPRS资源使用量（GB）’，发现他们单位不一致，因此从附件2中提取该变量时，需要对原数据乘1024的平方。

· 由于附件2中只有’当月MOU’因此选择如何处理’当月ARPU’、’前3月MOU’和’前3月ARPU’三个变量需要进行相关性分析，因此对这四个变量做Pearson系数的相关系数矩阵。

|  |
| --- |
| # ARPU和MOU相关性  data4=pd.read\_excel("C:/Users/90813/Desktop/mathorcup/初赛赛题 2022年MathorCup大数据竞赛/2022年MathorCup大数据竞赛-赛道B初赛/附件4调整.xlsx",header=0,skiprows=1)  cr=data4.iloc[:,67:71].corr()  cr |



发现相同统计量的情况下，当月和前3月的相关系数为0.8左右，具有很强的正相关关系，不同统计量的情况下，相关系数在0.4左右，具有一定的正相关关系，同时通过查询两个统计量的含义，发现MOU和ARPU分别表示平均每户每月通话时间和公司从每用户获得的每月平均收入，通话时长和花费必定是正相关的，因此可以合理证实两者之间的相关关系，又由于’当月ARPU’、’前3月MOU'和’前3月ARPU’无从可知，所以仅保留’当月MOU’。

4.2异常值检测

通常异常值处理通过箱线图和3σ法则来进行剔除，箱线图的剔除标准是超过1.5倍的四分位距，3σ法则则是通过正态分布左右3σ包含约99的数据来实现的，通过对比，3σ法则的区间更长一些，可以保留更多原数据的信息，因此选择该方法进行异常值的处理。

|  |
| --- |
| # 3sigma法  def three\_sigma(dataset,l,n=3):  ss=dataset.iloc[:,l:]  mean = np.mean(ss)  sigma = np.std(ss)  for i in dataset.index:  if sum(abs(ss.loc[i,:] - mean) > n \* sigma)>5:  dataset.drop([i],inplace=True)  return dataset |

4.3数据处理

|  |
| --- |
| # 附件2用于拟合的数据  ct=['手机上网整体满意度','网络覆盖与信号强度','手机上网速度','手机上网稳定性','居民小区','办公室','高校','商业街','地铁','农村','高铁','其他，请注明','网络信号差/没有信号','显示有信号上不了网',  '上网过程中网络时断时续或时快时慢','手机上网速度慢','其他，请注明.1','看视频卡顿','打游戏延时大','打开网页或APP图片慢',  '下载速度慢','手机支付较慢','其他，请注明.2','爱奇艺','优酷','腾讯视频','芒果TV','搜狐视频','抖音','快手','火山','咪咕视频',  '其他，请注明.3','全部都卡顿','和平精英','王者荣耀','穿越火线','梦幻西游','龙之谷','梦幻诛仙','欢乐斗地主','部落冲突','炉石传说',  '阴阳师','其他，请注明.4','全部游戏都卡顿','微信','手机QQ','淘宝','京东','百度','今日头条','新浪微博','拼多多','其他，请注明.5',  '全部网页或APP都慢','上网质差次数','脱网次数','微信质差次数','套外流量（MB）','套外流量费（元）','性别','终端制式','终端品牌',  '当月GPRS资源使用量（GB）','当月MOU','客户星级标识']  data5=data1.loc[:,ct]  data5\_dp=pd.get\_dummies(data5)  data5\_dp=three\_sigma(data5\_dp,4)  data5\_dp['当月GPRS资源使用量（GB）']=data5\_dp['当月GPRS资源使用量（GB）']\*1024\*1024  # 附件4数据清洗  data6=pd.read\_excel("C:/Users/90813/Desktop/mathorcup/初赛赛题 2022年MathorCup大数据竞赛/2022年MathorCup大数据竞赛-赛道B初赛/附件4调整.xlsx",header=0,skiprows=1)  for i in range(52): #将0-1变量转化成正确的表达形式  data6.iloc[:,i][data6.iloc[:,i]==-1]=0  data6.iloc[:,i][data6.iloc[:,i]!=0]=1  # 为保证独热后变量数量相等，将附件2中分类变量独热后的分变量赋给附件4，并判断属于哪类  for i in data5\_dp.iloc[:,59:62].columns:  data6[i]=0  for j in range(len(data6[i])): #判断性别  if data6['性别'][j]==str(i).split('\_')[1]:  data6[i][j]=1  for i in data5\_dp.iloc[:,63:65].columns:  data6[i]=0  for j in range(len(data6[i])): #判断4G还是5G  if data6['4\\5G用户'][j]==str(i).split('\_')[1]:  data6[i][j]=1  for i in data5\_dp.iloc[:,65:97].columns:  data6[i]=0  for j in range(len(data6[i])): #判断手机品牌  if data6['终端品牌'][j]==0: #若为空则用出现次数最多的苹果填充  data6['终端品牌'][j]=str(data6['终端品牌'][j])  data6['终端品牌'][j]='苹果'  elif data6['终端品牌'][j]==str(i).split('\_')[1]:  data6[i][j]=1  else:  pass  for i in data5\_dp.iloc[:,97:].columns:  data6[i]=0  for j in range(len(data6[i])): #判断客户星级  if data6['客户星级标识'][j]==str(i).split('\_')[1]:  data6[i][j]=1  data6.drop(columns=["4\\5G用户", "终端品牌","客户星级标识"],inplace=True)  data6.shape |

4.4数据降维和采样

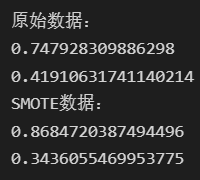
在本环节采用PCA降维来对变量进行筛选，并使用SMOTE、SMOTEENN、….来对不均衡数据进行补充。

|  |
| --- |
| #PCA+分数据集+SMOTE  pca=PCA(n\_components=0.95)  scaler=MinMaxScaler()  x\_scaled=scaler.fit\_transform(data5\_dp.iloc[:,4:])  x\_reduced=pca.fit\_transform(x\_scaled)  x\_train\_svc1,x\_val\_svc1,y\_train\_svc1,y\_val\_svc1=train\_test\_split(x\_reduced,data5\_dp.iloc[:,0],test\_size=0.2,random\_state=32)  sm = SMOTE(random\_state=32)  x\_res1,y\_res1 = sm.fit\_resample(x\_train\_svc1,y\_train\_svc1)  #PCA+分数据集+SMOTEENN  smt = SMOTEENN(random\_state=32)  x\_res2,y\_res2 = smt.fit\_resample(x\_train\_svc1,y\_train\_svc1) |

4.5建立模型

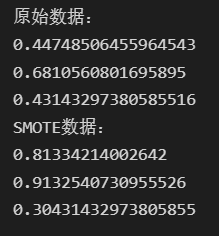
4.5.1 SMOTE+SVC

|  |
| --- |
| # SMOTE SVC  print('原始数据：')  clf1=SVC(C=0.8,gamma=0.8,kernel='rbf',decision\_function\_shape='ovr',random\_state=32)  clf1.fit(x\_train\_svc1, y\_train\_svc1)  print(clf1.score(x\_train\_svc1, y\_train\_svc1))  print(clf1.score(x\_val\_svc1, y\_val\_svc1))  print('SMOTE数据：')  clf1=SVC(C=0.8,gamma=0.8,kernel='rbf',decision\_function\_shape='ovr',random\_state=32)  clf1.fit(x\_res1, y\_res1)  print(clf1.score(x\_res1, y\_res1))  print(clf1.score(x\_val\_svc1, y\_val\_svc1)) |



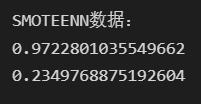
4.5.2 SMOTE+随机森林

|  |
| --- |
| # SMOTE 随机森林  rf=RandomForestClassifier(n\_estimators=2000,min\_samples\_leaf=10,oob\_score=True,random\_state=32)  print('原始数据：')  rf.fit(x\_train\_svc1,y\_train\_svc1)  print(rf.oob\_score\_)  print(rf.score(x\_train\_svc1, y\_train\_svc1))  print(rf.score(x\_val\_svc1, y\_val\_svc1))  print('SMOTE数据：')  rf.fit(x\_res1,y\_res1)  print(rf.oob\_score\_)  print(rf.score(x\_res1, y\_res1))  print(rf.score(x\_val\_svc1, y\_val\_svc1)) |



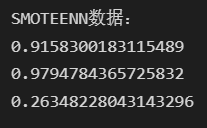
4.5.3 SMOTEENN+SVC

|  |
| --- |
| # SMOTEENN SVC  print('SMOTEENN数据：')  clf1=SVC(C=0.8,gamma=0.6,kernel='rbf',decision\_function\_shape='ovr',random\_state=32)  clf1.fit(x\_res2, y\_res2)  print(clf1.score(x\_res2, y\_res2))  print(clf1.score(x\_val\_svc1, y\_val\_svc1)) |



4.5.4 SMOTEENN+随机森林

|  |
| --- |
| # SMOTEENN 随机森林  print('SMOTEENN数据：')  rf.fit(x\_res2,y\_res2)  print(rf.oob\_score\_)  print(rf.score(x\_res2, y\_res2))  print(rf.score(x\_val\_svc1, y\_val\_svc1)) |



4.5.5 Logit+SMOTEENN

4.5.6 Logit+SMOTETomek

4.5.7 AdaBoost+SMOTEENN

4.5.8 AdaBoost+SMOTETomek

4.5.9 GBDT+SMOTEENN

4.5.10 GBDT+SMOTETomek

4.5.11 XGBoost+SMOTEENN

4.5.12 XGBoost+SMOTETomek

4.5.13 LightGBM+SMOTEENN

4.5.14 LightGBM+SMOTETomek

4.6模型比较