**《统计软件》**

**实 训 报 告**

|  |  |
| --- | --- |
| **实训时间：** | **2022.05.28** |
| **学生姓名：** | **XXXX** |
| **班级学号：** | **XXXXXXXX** |

**1 实训内容**

**目的**：熟悉python中建立预测模型的基本方法。

**实验项目名称**：保险客户购买保险行为分析

**数据文件**：此数据集表示对从保险公司购买保单感兴趣的大量客户的活动。每个Quote\_Id对应一个潜在客户，Quote\_Flag表示该客户是否购买了某个产品。所提供的特性是匿名的，并提供潜在客户和政策的丰富表示。它们包括特定领域、覆盖范围、销售、个人、财产和地理信息。

**第一天：**

(1) 确定各变量类型：并说明理由.

(2) 对于各类变量给出单变量基本分析结果（要求每一个变量都用相应的图表展示）.

(3) 对缺失值进行填充.

**第二天：**

(1)根据第一天内容中，你认为的变量类型，结合各变量取值情况，对各变量的缺失值和异常值进行处理。

**第三天：**

(1) 选用至少6个预测模型进行训练

(2) 查询各模型中可以设置的哪些参数，并简单调节个模型的参数。

(3) 给出各模型的在训练集上的结果。

(4) 选出你认为最好的模型。

**第四天：**

(1) 在数值型变量选择你认为适当的变量加入模型。

对实训报告（3）你认为比较好的预测模型进行调参，并给出你认为模型的最佳参数。

(2) 将多个模型集成为一个模型

(3) 将类别型变量也加入模型，重新选择变量、选择模型、模型调参和模型集成。

**2 实训过程**

基于集成学习的保险行为购买预测模型

摘要

本文针对保险风控领域新客群的购买行为问题, 构建基于可解释性的传统机器学习模型LR与LinghtGBM、CatBoost等高性能的模型的集成学习模型, 建立了适用于新客群的迁移学习模型, 捕捉不同业务中用户基本信息与购买行为之间的关联, 对保险公司根据现有购买行为数据来服务新场景、新客群具有重要研究意义.

首先对数据集进行探索分析，按照数据类型分析用户购买行为的数据特征.对于日期变量提取年月日处理；对于文本变量逐一分析其与label的关系; 对于连续数值特征观察其波动性，以便捕捉业务特征，方便构造特征. 接着进行基本的数据处理，填充缺失值, 去除唯一值等, 使用**原则**去除异常值.

特征工程部分, 对文本特征做简单**的LableEoder**编码, **提取时间特征**的年、月、日；对于文本特征依,类别低的使用编码. 特征优化部分，使用**Fiter方法和Enbedded方法**过滤特征，以相关系数过滤及卡方检验的形式过滤初级特征；以树模型为基础，训练时间原因最终筛选20个特征.

模型训练部分，选用机器学习常见分类模型，**LightGBM、CatBoost、SVM、GBDT、LR、RF**等，按照7：3划分训练集之后，使用**5折交叉验证**的方法训练模型，限于设备问题，使用**网格搜索**加手工调参的方法，最终各个模型的AUC值分别为：0.725、00.732、0.701、0.662、0.765.模型融合部分，采用软投票的机制，以RF、LR、Ensemble、naive Bayes为基础，构建基于不同权重购买概率模型，最终运用于新数据集上，提交结果.

最后，考虑到现有建模方法较为统一，尝试使用automl框架来进行这套流程，选取amazon的自动机器学习框架autogunlon来训练模型，选择限制搜寻时间，以贝叶斯优化模型参数，以AUC作为评价指标，最终选择两层Stacking的WeightedEnsemble\_L2模型作为最终模型，其训练集AUC值为0.916111. 最终运用于新数据集上，提交结果.

**关键词：特征交叉; 模型融合; 交叉验证; LightGBM; 集成学习; AutoGunlon**

# **1. 问题重述**

## 1.1 问题背景

为进一步促进金融普惠的推广落地, 保险公司需要服务许多新的客群.作为对风险控制要求很高的行业, 因为缺乏对新客群的了解, 对新的细分客群的风控处理往往成为金融普惠的重要阻碍.如何利用用户现有购买行为数据来服务新场景、新客群成了一个很有价值的研究方向, 机器学习是其中一个重要手段.

**图1-1 机器学习在金融领域应用**

## 1.2 待解决的问题

要求利用已有的与目标客群稍有差异的另一批购买行为数据, 辅助目标业务购买预测模型的创建, 两者数据集之间存在大量相同的字段和极少的共同用户.此处希望可以利用迁移学习捕捉不同业务中用户基本信息与购买行为之间的关联, 帮助实现对新业务的用户购买预测.

赛题给定的数据有:

TrainingData.csv 个人购买记录数据.

UnknowData.csv 用于测试的数据,提交结果.

依据二分类的评价标准, 使用ROC曲线的面积AUC来衡量得分, , 若有*M*个正样本, *N*个负样本的数据集里, AUC的计算公式为



其中代表第条样本的序号, 即概率得分从小到大排序, 排在第rank个位置.

# 2. 问题分析

本题为标准的表单型二分类数据问题, 评价指标选取为ROC曲线面积AUC的值, 基于此, 需要从TrainingData数据集中抽取数据特征, 构建一个分类预测模型, 计算购买概率, 迁移到新的数据集UnknowData数据集上.

针对TrainingData数据集, 可以考虑与现有的UnknowData进行分布对比, 选择分布接近的数据特征来进行下一步的特征构建, 以提高模型的泛化能力, 进一步提高评价指标AUC的值.

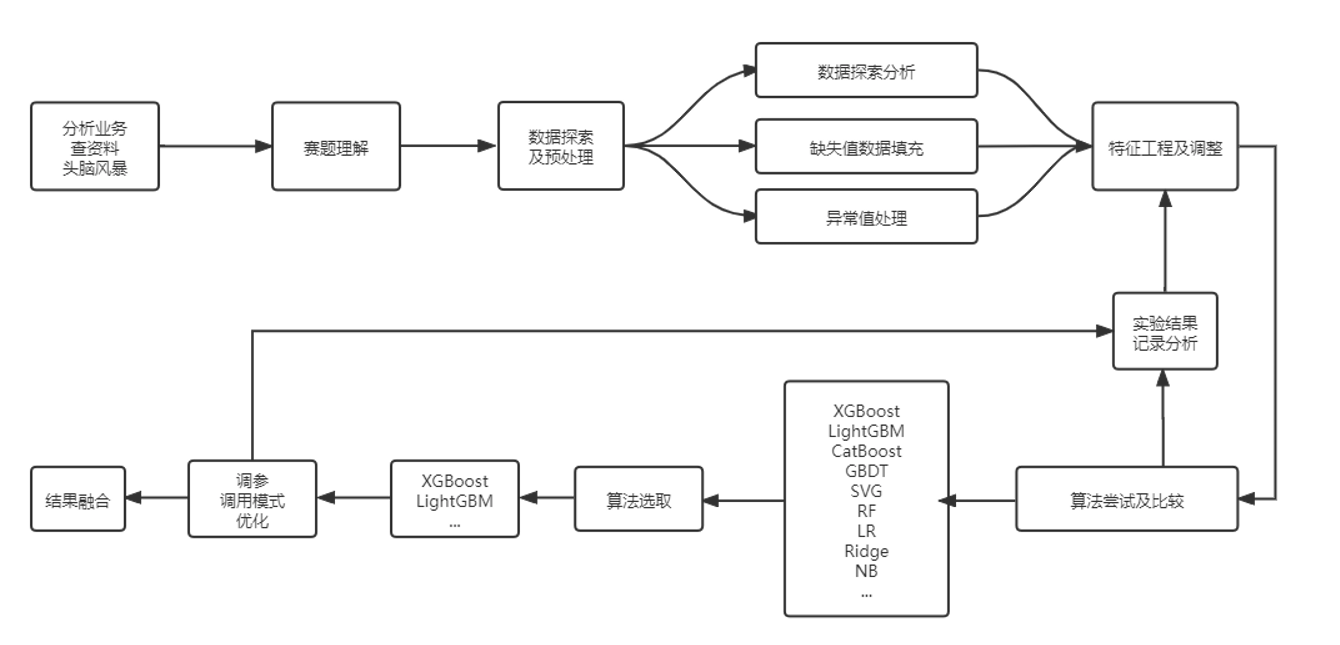
****针对TrainingData数据集, 其有部分TrainingData数据集未有的特征, 且TrainingData数据集也具有是否购买的标签.基于此, 可以先使用现有数据集用树模型进行训练, 得到特征重要性排名, 根据不在两数据集交叉特征里面的特征, 对TrainingData进行数据升维, 进而进行TrainingData的数据集训练, 最终得到模型, 去预测UnknowData数据集.

图2-1思路流程图

# 3. 模型假设

(1) 假设实验在理想状态下进行;

(2) 假设所提供的数据绝大部分真实可靠;

(3) 假设数据满足分布

(4) 假设验证集与训练集分布相差不大

# 4. 符号说明

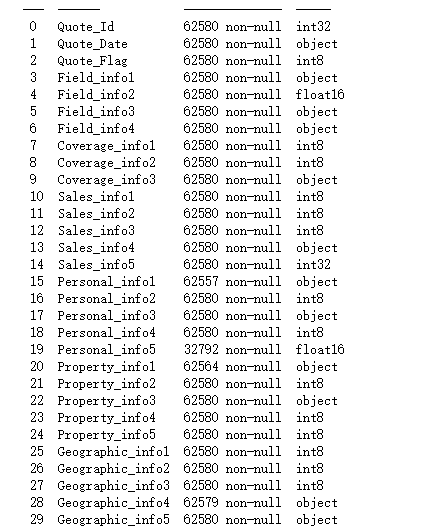
|  |  |
| --- | --- |
| **符号** | **说明** |
|  | 残余误差 |
|  | 组内离差平方和 |
|  | 组间离差平方和 |
|  | Spearman相关系数 |
|  | 信息量特征 |
|  | 表示第个输入的向量 |
|  | 表示第个输出的值 |
|  | 表示实验测量的值即真实值 |
|  | 学习率 |
|  | LR参数 |
| t |  |
|  | 最终数据维度 |
|  | 表示通过模型得到的预测概率 |

# 5. 数据探索

## 5.1 总体特征

首先, 查看银行保险样本数据集, 该数据集维度为10000个样本, 根据数据类型可以

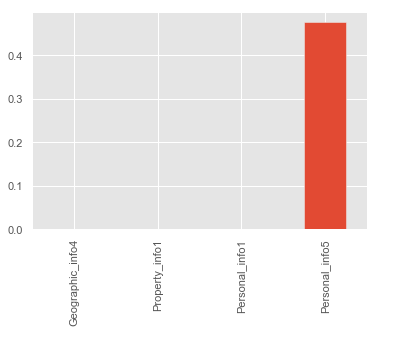
分为日期特征、文本特征、数值特征、ID、以及label.



**图5-1 原始数据特征总体特征查看**

对于文本特征, 针对类别型需要做单独处理, 且文本特征有些特征维度较高, 需要进一步探索；对于日期特征, 数据集给定的是Quote\_Date时间的特征, 可以做时间特征提取.对于数值特征, 可分为离散特征与连续特征, 离散特征可以按照类别特征来探索, 连续特征可以先查看其分布情况, 对于数量级差别较大的数据, 进行分箱处理, 对于分布不满足正态的, 需要进行变换.

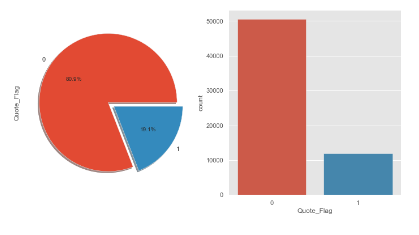
## 5.2 缺失值查看

数据集的特征维度较大, 样本数量多, 首先进行缺失值的探索, 如图5-3.

**图5-2 缺失值数量图**

从图中可以看出, 缺失值还是比例最大的为Personal\_info5匿名特征, 占比为0.4759, 对整体数据集影响比较大, 如果接下来做树模型, 可以依据树模型的特点, 不给予填充；如果是普通普通模型, 需要进形缺失值的填充.考虑样本量大的特点.

## 5.2 重复值唯一值分析及正负样本比例

对于数据集, 使用分组聚合的方式来统计查看单一特征的情况, 从聚合结果可以看出Property\_info2这一特征的值唯一, 即为单一重复值, 对于模型中用处不大, 即考虑舍弃这一特征.接着, 查看了标签的正负样本比例, 见图5-1.

**图5-4 正负样本分布以及比例**

从图中可以看出, 样本标签label的正负样本接近4：1, 对于风控领域该样本比例以及比较均衡, 考虑题目中评价指标为AUC, 即正负样本比例不影响, 即不用采取采样技术扩充负样本数量.

## 5.3 文本特征探索

通过对文本数据分组聚合得到的结果见图5-5.

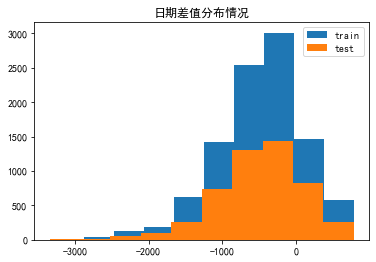
|  |  |
| --- | --- |
|  |  |
|  |  |

**图5-5 文本数据分组聚合结果**

从图中可以看出, 文本指标分散较为明显, 且不同类别和label的关系差异较大, 因此考虑可以使用onehot处理来做.

## 5.4日期数据探索及数值数据分布情况

### 5.4.1 日期数据探索

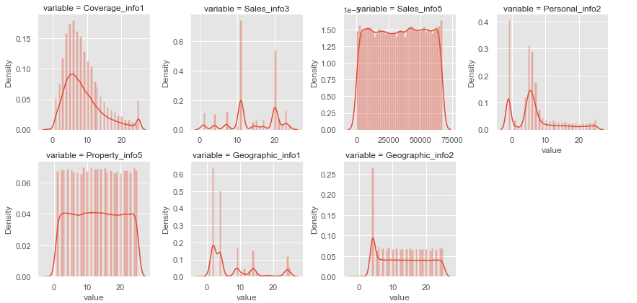
对于issue\_date特征数据, 根据含义为贷款还款时间, 因为数据表中无贷款发放时间, 可以设置一个基准日期, 对时间进行差分处理, 对训练集集和测试集均做处理, 得到两者差值的分布情况, 见图5-6

**图5-6 日期差值分布情况**

如上图, 训练集的范围基本覆盖测试集, 即可以做差分处理.

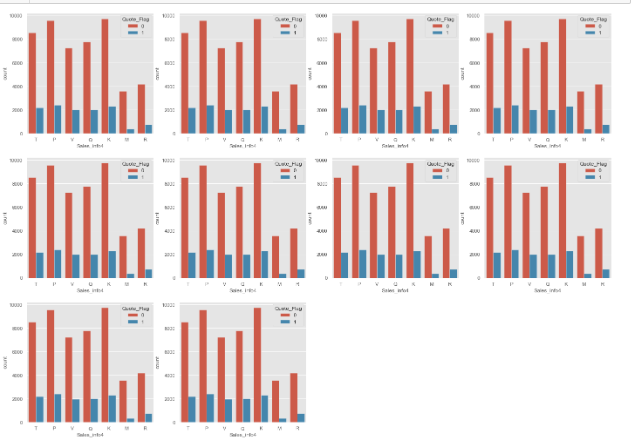
### 5.4.2 数值特征分布情况

对于连续数值特征, 绘制分布图, 来观察数据集的分布是否一致, 以确定其是否满足要求.



**图5-7数值特征分布图**

对于类别型数值特征，绘制条形图查看，如下图：

图5-7 类别数值特征条形图

# 6. 特征工程

## 6.1 非数值数据处理

### 6.1.1 缺失值处理

对于数据缺失值部分, 可以按照建立的模型选择来进行处理,

(1) 对于选择树模型部分, 缺失值无需填充, 树模型在生产下一轮树的时候自动做处理.

(2) 对于非树模型部分, 按数据类型进行填充, 对于字符型数据, 缺失值按照新的类别划分, 对于数值类型的数据, 按照平均值填充.

## 6.1.2 文本类型数据和类别数据编码

针对数据集中字符型数据, 补全后的数据仍然是字符型的, 不利于机器计算、常见的类别特征编码有Label Encoder、one-hot编码、Frequency编码、target编码等.

LabelEncoder即标签编码, 是对不连续的数字或者文本进行编号, 转换成连续的数值型变量, 即进行数字化处理.

One-Hot即独热编码, 是使用 N位状态寄存器来对 N个状态进行编码, 每个状态都有它独立的寄存器位, 并且在任意时候, 其中只有一位有效.它解决了分类器不好处理属性数据的问题, 而且在一定程度上也起到了扩充特征的作用

Target即目标编码, 它使用目标变量的均值编码类别变量.我们为训练集中的每个分组计算目标变量的统计量, 之后合并验证集、测试集以捕捉分组和目标之间的关系.

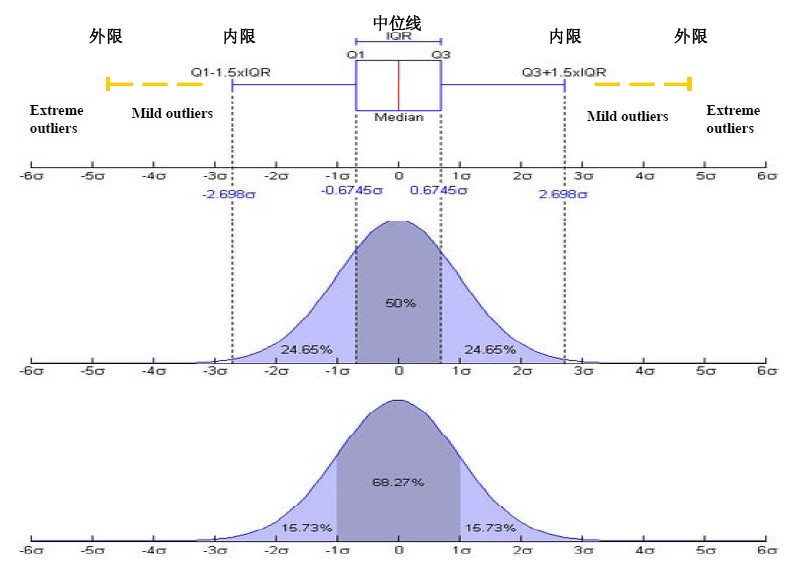
鉴于数据的维度和处理的复杂程度, 考虑使用Frequency编码将数据集中的字符型变量处理成数值型变量, 方便下一步进行计算.

## 6.1.3 日期变量特征提取

对于Quote\_Date特征数据, 根据含义为贷款还款时间, 因为数据表中无贷款发放时间, 可以设置一个基准日期, 对时间进行差分处理, 得到, 以及提取Quote\_Flag的年、月、日.进行多途径处理, 将数据特征维度升上去.

## 6.2 数值数据处理

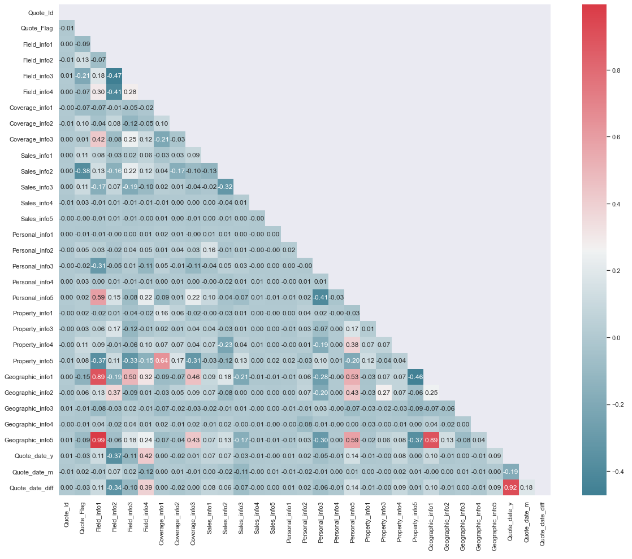
### 6.2.1 异常值处理

对于数据中的异常值, 我们采用统计学上常用的原则进行筛选, 得到不同特征的异常值, 基于此我们考虑一个样本, 如果异常数据指标数量, 即考虑为异常样本, 将该样本从数据集中剔除, 依据此来处理异常数据

**图6-1 选出异常值原理图**

## 6.3 特征筛选

### 6.3.1 Filter方法 (过滤式)

计算各个特征的方差, 然后根据阈值选择方差大于阈值的特征, 或者指定待选择的特征数, 然后选择个最大方差的特征.

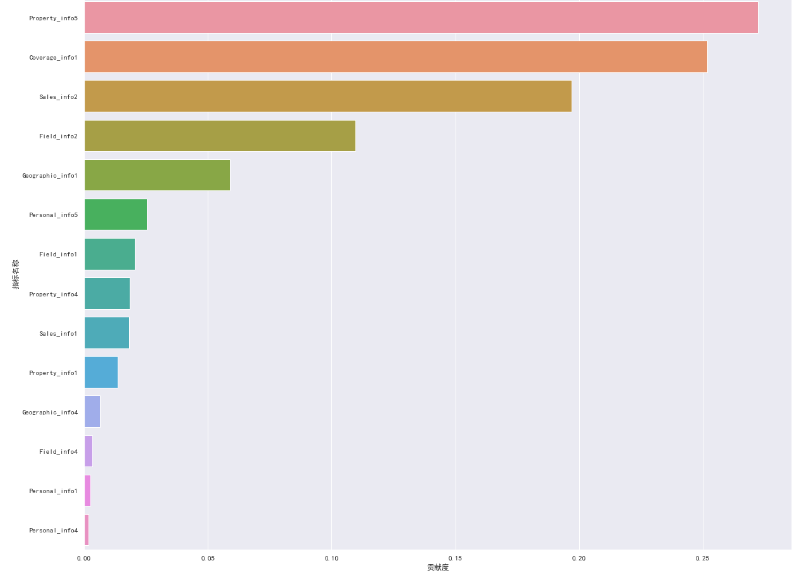
**图6-2 原始特征相关系数热力图**

选取阈值为0.75, 即两个特征之间相关系数大于0.75, 保留一个即可, 基于此过滤特征, 得到新的特征.

### 6.3.2 Enbedded方法 (嵌入式)

嵌入法是一种让算法自己决定使用哪些特征的方法, 即特征选择和算法训练同时进行.

先使树模型的算法和模型进行训练, 得到各个特征的权值系数, 根据权值系数从大到小选择特征, 这些权值系数往往代表了特征对于模型的某种贡献或某种重要性, 这些权值系数往往代表了特征对于模型的某种贡献或某种重要性.



**图6-4 特征重要性排名**

# 7. 模型训练与调参

## 7.1 模型的建立

### 7.1.1 LR (逻辑回归)

针对数据集, 取值概率[5]可记为



为使每个样本属于其真实标记的概率越大越好, 考虑各个样本概率乘积最大, 得到似然函数



对于使然函数极大, 可以化为最小, 整理为



假设对数几率回归模型的函数形式为; 优化的目标函数是



找到一对合理的参数使得目标函数最小, 采用梯度下降的方法



其中为学习率, 设置迭代次数和初始迭代点进行迭代, 找到最优解.

### 7.1.2 SVM (支持向量机)

支持向量机[3]是一种二分类模型, 它是将向量映射到一个更高位的空间里, 在这个空间里建立一个最大间隔的超平面. 在分开数据的超平面的两边建有两个互相平行的超平面, 分隔超平面使两个平行超平面的距离最大化, 平行超平面的距离或者差距越大, 分类器的总误差越小.

首先将划分出来的训练集中的数据定义为集合



同时, 这里将超平面定义为, 则可以得出样本点与超平面的间隔为



那么两类样本点的间隔(支持向量的间隔)为



将问题可以转化为求解支持向量与超平面的最大间隔. 样本点与超平面的最小距离, 即(为样本点个数). 因此, 求解支持向量与超平面的最大间隔就是求解, 则有



设, 进而转化为求. 构造拉格朗日函数.



其中, 为拉格朗日乘子, 为拉格朗日函数, 为惩罚参数, 通过求解上式的最优解来求得, 即



其中是解的分量. 最终求求出最优超平面为:, 最优分类函数



根据的大小, 可以判断预测结果的情况, 若大于0, 则为正预测结果, 否则为负预测结果.

### 7.1.3 GBDT（梯度提升树）

考虑当提升树中的损失函数为一般函数时, 残差计算不是很方便, 因此Freidman 提出利用损失函数的负梯度在当前模型的值作为提升树模型残差的近似值方法, 构建梯度提升树的二分类算法过程.

基于梯度提升树（GBDT）的分类模模型

Step1: 建立初始化评估器;



Step2: 对于1884个样本, 计算下降最快即负梯度的方向;



Step3: 将负梯度方向的函数值作为样本更新后的值, 并把作为下棵树训练的基础数据, 每一个对应的叶子区域, 得到节点个数记为*J*;

Step4: 最后得到加强后的学习器.



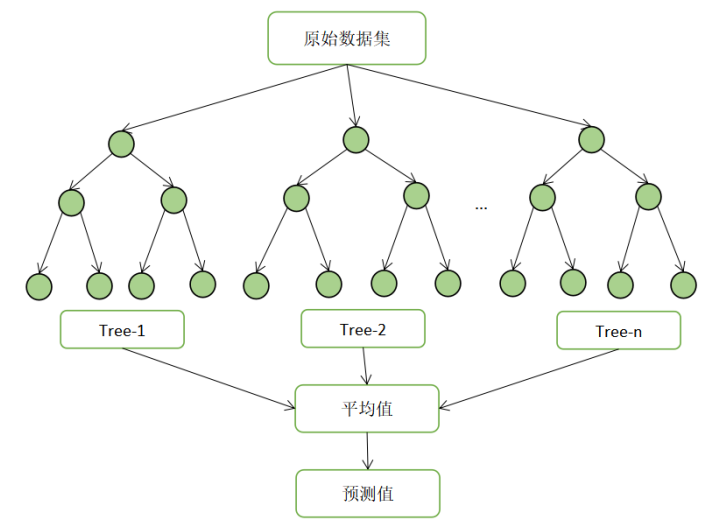
Step5: 每个样本的(最终)预测值为



经过上述过程, 得到了决策树最深3层, 共用12个初始器.

### 7.1.4 RF(随机森林)

本题中的7843个样本的特征变量具有数量多, 高度非线性, 相互强耦联等特点. 因此, 选择利用随机森林算法, 随机森林算法的具体流程见图5-1.



**图7-1 随机森林算法**

利用随机森林算法进行变量筛选的过程主要包括以下几个步骤.

|  |
| --- |
| **随机森林算法** |
| **Step1: 构建*K*棵分类树**  原始训练集为7843个样本中的70%, 应用有放回抽样法又放回地抽取*K*个新的样本集, 与此同时构建*K*棵分类树, 每次未被抽到的样本成为*K*个袋外数据.  **Step2: 生成决策树**  假设特征空间共有*H*个特征, 则在每一轮生成的决策树的过程中,从*H*个特征中随机选择*h*个特征,其中, 组成一个新的特征集, 通过使用新的特征集生成决策树, 在*K*轮中共生成*k*个互不相关的决策树.  **Step3: 组成随机森林**  将生成的多棵树组成随机森林, 互不相关的若干棵决策树的重要性是等价的. |

### 7.1.5 XGBoost

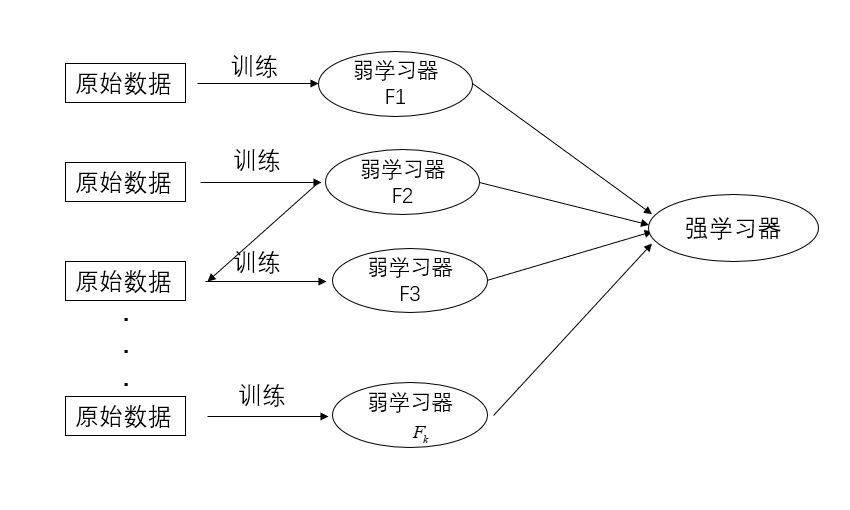
XGBoost(eXtreme Gradient Boosting)极致梯度提升, 是基于GBDT的一种算法.XGBoost的目标函数是由目标函数以及正则化两部分组成.



其中,为叶子节点所包含样本的一节偏导数累加之和；为叶子节点所包含样本的二阶偏导数累加之和.其中求解的算法如下：

|  |
| --- |
| **XGBoost算法** |
| **Step1: 构造每个叶子节点j的目标函数：**    **Step2: 生成决策树**  构建形如一元二次方程, 求最优解, 则在处取得最小值, 最小值为.  **Step3: 代入求得最优的目标值：** |

XGBoost算法的原理如下图所示：

**图7-2 XGBoost原理流程图**

### 7.1.6 LightGBM

LightGBM(Light Gradient Boosting Machine)是一种高速、分布式并且性能较高的梯度提升框架, 它利用弱分类器（决策树）迭代训练以得到最优模型.LightGBM是一个实现GBDT算法的框架, 主要由基于梯度的单边采样（GOSS）和互斥特征捆绑（EFB）算法构成.LightGBM具有训练速度快、效率高、内存占用少、精度高、能够处理大规模数据、支持并行和GPU学习等优点.

### 7.1.7 Adaboost集成学习算法

**Adaboost**

1：输入为样本集

,

2： 输出为{-1, +1}, 弱分类器算法, 弱分类器迭代次数K.输出为最终的强分类器

1. 初始化样本集权重为



（2） 对于

a) 使用具有权重DkDk的样本集训练数据, 得到弱分类器

b)计算的分类误差率



c) 计算弱分类器的系数



（3）构建最终分类器为：



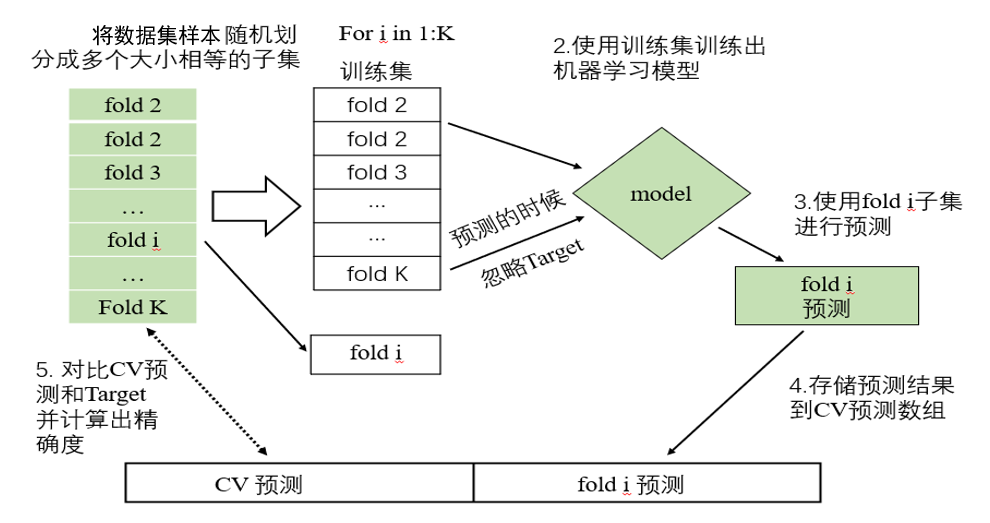
其中R为类别数.从上式可以看出, 如果是二元分类, R=2, 则上式和我们的二元分类算法中的弱分类器的系数一致.

## 7.2 测试集与训练集的划分

将样本个数据集划分为训练集和测试集, 采用训练集进行模型的构建, 测试集验证模型的准确性, 然后利用K折交叉验证方法, 取K=10来将数据集随机划分成十等份, 轮流将其中的7份作为训练集, 3份作为测试集来验证模型.

## 7.3 K折交叉验证

K折交叉验证是指将原始数据分成K组(一般是均分), 将每个子集数据分别做一次验证集, 其余的K-1组子集数据作为训练集来得到K个模型, 用这K个模型最终的验证集的分类准确率的平均数作为此K-CV下分类器的性能指标. 本题取K=10, 做十折交叉验证(见图7-3).

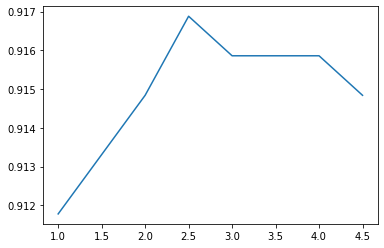


**图7-3 K折交叉验证示意图**

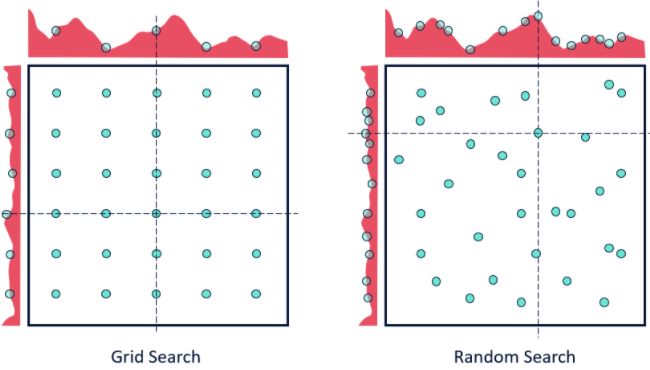
## 7.4 模型超参数调节

### 7.3.1 传统手工搜索

在传统的调参过程中, 我们通过训练算法手动检查随机超参数集, 并选择符合我们目标的最佳参数集.

**图7-4 SVM调参数C结果图**

### 7.3.2 网格搜索

网格搜索是一种基本的超参数调优技术.它类似于手动调优, 为网格中指定的所有给定超参数值的每个排列构建模型, 评估并选择最佳模型.例子:两组超参数k\_value =[2,3,4,5,6,7,8,9,10] & algorithm =[' auto ', ' ball\_tree ', ' kd\_tree ', ' brute '], 在这个例子中, 它总共构建了9\*4 = 36不同的模型.

**图7-5 网格搜索原理示意图**

### 7.3.3 各个模型具体参数

为求解以上模型, 使用网格搜索+手工调参的形式调节参数, 具体参数如下, 详细见附录.

**表7-1 SVM参数**

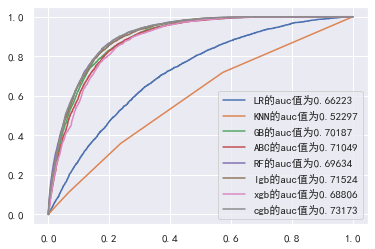
|  |  |
| --- | --- |
| 参数名称 | 参数数值 |
| 'break\_ties': | False, |
| 'cache\_size': | 200, |
| 'class\_weight': | None, |
| 'coef0': | 0.0, |
| 'decision\_function\_shape': | 'ovr', |
| 'degree': | 3, |
| 'gamma': | 'auto', |
| 'kernel': | 'rbf', |
| 'max\_iter': | -1, |
| 'probability': | False, |
| 'random\_state': | None, |
| 'shrinking': | True, |
| 'tol': | 0.001, |

## 7.5 各个模型性能度量

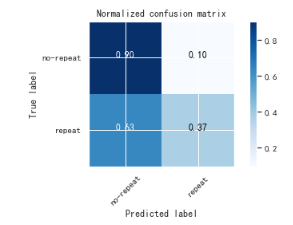
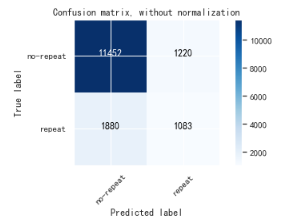
赛题依据二分类的评价标准, 使用ROC曲线的面积AUC来衡量得分, , 若有M个正样本, N个负样本的数据集里, AUC的计算公式为 :



其中 代表第 条样本的序号, 即概率得分从小到大排序, 排在第rank个位置.

图7-6 各个模型的roc曲线图

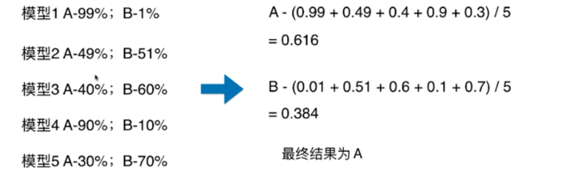
从图中可以看出, Catboost和XGBoost的AUC值较高, 效果较好



**图7-7 部分模型的混淆矩阵图**

## 7.6 基于投票法的模型融合

假设对于一个二分类问题, 有3个基础模型, 现在我们可以在这些基学习器的基础上得到一个投票的分类器, 把票数最多的类作为我们要预测的类别.



**图7-8 模型融合原理图**

# 8. 总 结

## 8.1模型评价

### 8.1.1 模型的优点

(1)建立的集成学习模型不需考虑两组变量中各个变量之间的对应关系, 方便计算多变量之间的相关性.

(2)决策树模型(如随机森林和LightGBM)挖掘出来的分类规则准确性高, 具有很好的解释性, 便于理解.

(3)建立的分类算法预测模型具有自适应能力、容错性强, 能够很好的处理非线性、非局域性的大型复杂系统.

(4)设置 K折交叉验证, 通过分数的平均值的结果, 综合考量了各学习模型的预测准确度和稳定性. 选出来的模型更具有代表性, 泛化效果更好.

(5)模型融合进一步提高模型的性能, 泛化能力强, 具有很好的推广价值.

### 8.1.2 模型的缺点

随机森林算法可能有很多相似的决策树, 掩盖了真实的结果, 而且调参效率不高, 无法控制模型内部的运行, 只能在不同的参数和随机种子之间进行尝试.

## 8.2 模型的改进

(1)使用机器学习模型提取主要特征虽能加快训练速度, 但也会使样本信息提取不完全导致部分信息丢失, 后续可以尝试构造 bagging 特征.

(2)对于特征选择, 先通过搜索的方法找到所有可能的特征, 但搜索的运算量较大, 可以进一步设计优化算法减少运算量.

# **参考文献**

* 1. 王睿. Boosting族算法在信用评分卡模型中的应用研究[D].大连理工大学,2021.
  2. Dharani N P,Bojja Polaiah,Raja Kumari Pamula. Evaluation of Performance of an LR and SVR models to predict COVID-19 Pandemic.[J]. Materials today. Proceedings,2021:
  3. Sakhnovich A, On the GBDT Version of the Bäcklund Darboux Transformation and its Applications to Linear and Nonlinear Equations and Weyl Theory, Mathematical Modelling of Natural Phenomena, 5(4): 340-389, 2012.
  4. 张宗垚. 基于支持向量机的用户回购行为预测[J]. 电脑编程技巧与维护, 2020(06): 3-6+21.
  5. Gorishniy, Y., Rubachev, I., Khrulkov, V., and Babenko, A., “Revisiting Deep Learning Models for Tabular Data”, arXiv e-prints , 2021.

# 附 录

## 附录1：附件文件列表

由于本文模型尝试较多, 这里列出文中结果较好的几个模型的代码部分

支撑材料清单

|  |  |
| --- | --- |
| Jupter文件 | 保险客户购买保险行为分析与建模.ipynb |
| 提交结果文件表格 | submission csv |

## 附录2：程序运行环境

**表1 运行环境**

|  |  |
| --- | --- |
| 系统 | windows11, 64位操作系统 |
| CPU | Intel i5-9300H 2.4GHz |
| GPU | NVIDIA GeForce GTX 1050 3G |

**表2 所用软件**

|  |  |
| --- | --- |
| Anaconda3 | 2.2.0 |
| Jupyter Notebook | 6.1.4 |
| PyCharm Community | 2022.1.1 |

**表3 python相关包版本**

|  |  |
| --- | --- |
| 包名 | 版本 |
| LightGBM | 3.2.1 |
| Scikit-learn | 0.3.2 |
| seaborn | 0.11.0 |
| Autogluon | 0.4.0 |

## 附录3：数据探索源程序 (详细见附件)

1. from sklearn.model\_selection import train\_test\_split
2. from sklearn.model\_selection import GridSearchCV
3. from sklearn.metrics import classification\_report
4. from sklearn.ensemble import RandomForestClassifier
5. *# Split the dataset in two equal parts*
6. X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size=0.7, random\_state=0)
7. *# model*
8. clf = RandomForestClassifier(n\_jobs=-1)
9. *# Set the parameters by cross-validation*
10. tuned\_parameters = {
11. 'n\_estimators': [50, 100, 200]
12. *#                     ,'criterion': ['gini', 'entropy']*
13. *#                     ,'max\_depth': [2, 5]*
14. *#                     ,'max\_features': ['log2', 'sqrt', 'int']*
15. *#                     ,'bootstrap': [True, False]*
16. *#                     ,'warm\_start': [True, False]*
17. }
18. scores = ['precision']
19. for score in scores:
20. print("# Tuning hyper-parameters for %s" % score)
21. print()
22. clf = GridSearchCV(clf, tuned\_parameters, cv=5,
23. scoring='%s\_macro' % score)
24. clf.fit(X\_train, y\_train)
25. print("Best parameters set found on development set:")
26. print()
27. print(clf.best\_params\_)
28. print()
29. print("Grid scores on development set:")

## 附录4：Autogluon源程序

1. *#!/usr/bin/env python*
2. *# coding: utf-8*
3. import pandas as pd
4. df = pd.read\_csv(".\数据集\TrainingData.csv", header=0)
5. df
6. train\_data=df.sample(frac=0.7)*#按0.7比例随机采样*
7. test\_data=df[~df.index.isin(train\_data.index)]
8. train\_data
9. from autogluon.tabular import TabularDataset, TabularPredictor
10. train\_data = TabularDataset(train\_data)
11. test\_data = TabularDataset(test\_data)
12. label="Quote\_Flag"
13. metric='roc\_auc'
14. predictor = TabularPredictor(label,
15. eval\_metric=metric,
16. ).fit(train\_data)
17. predictor.leaderboard(test\_data, silent=True)