**银行客户流失预测**

李嘉伟，赵子宸，康博华，郭豪文，杨澍，谭聪

# 摘要：随着网络技术的高速发展,利率市场化、大数据迅速发展,银行业均表现出明显的"二八定律"现象,20%的优质客户占据了银行的大部分资产。如何防止银行客户流失,尤其是优质客户的流失,已经成为银行越来越关注的问题。因此,建立优质客户流失预测模型就显得尤为重要。使用基于机器学习对银行客户流失预测分析方法，使用Pycaret建立集成模型，对数据集进行分析，进而得出一个能够较好预测银行客户流失的模型。

**关键词：**银行客户流失；数据采样；Pycaret

Bank customer churn forecast

Li Jiawei,Zhao Zichen,Kang Bohua,Guo Haowen,Yang Shu,Tan Cong

**Abstract:** With the rapid development of network technology, the marketization of interest rate and the rapid development of big data, the banking industry shows an obvious phenomenon of the "20-80 law", and 20% of the high-quality customers occupy most of the assets of the bank. How to prevent the loss of bank customers, especially the loss of high quality customers, has become a problem that banks pay more and more attention to. Therefore, it is particularly important to establish a high quality customer churn prediction model. Based on machine learning to forecast bank customer churn analysis method, using Pycaret to build an integrated model, analysis of the data set, and then get a model that can better predict bank customer churn.

**Key words:** bank customer loss; data sampling; Pycaret

# 1 引言

对于客户流失目前还缺乏统一的定义，通常来说，客户终止与本公司的服务或转向其他公司提供的服务，就被认为是客户流失。客户流失按照是否为客户的主观意愿可以分为自愿流失与非自愿流失。

本次实验的目标主要是：识别导致客户流失的因素，并进行可视化。建立一个预测模型，预测分类客户是否流失，并根据模型的性能，选择一个最优模型。

由于流失原因的不同，各种流失的表现相差也比较大，因此，无法找到一种能够预测所有流失的模型和方法，只能针对流失的不同类型分别做相关的分析和研究，建立一个能够相对更加准确预测该类流失的模型。

根据识别出的因素及客户流失的历史数据对现有客户进行流失预测，针对不同的客群提供不同的维护手段，从而降低客户的流失率。

**2 材料准备**

## 2.1 导入所需要的的库与数据

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sns

pd.options.display.max\_rows = None

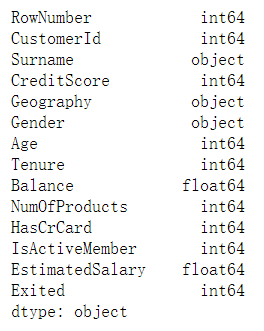
pd.options.display.max\_columns = None

df = pd.read\_csv('Churn\_Modelling.csv', delimiter=',')

df.head()

df.shape

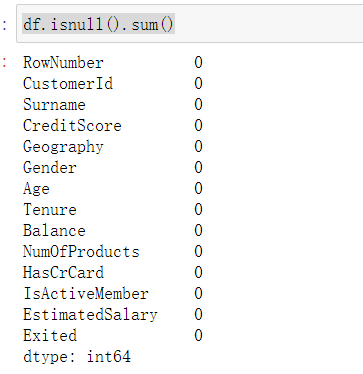
## 2.2 数据描述



特征变量与类型

数据集中共有14个特征变量，其中有8个数字变量，5个分类变量，Exited作为目标变量。

## 2.3 缺失值检测

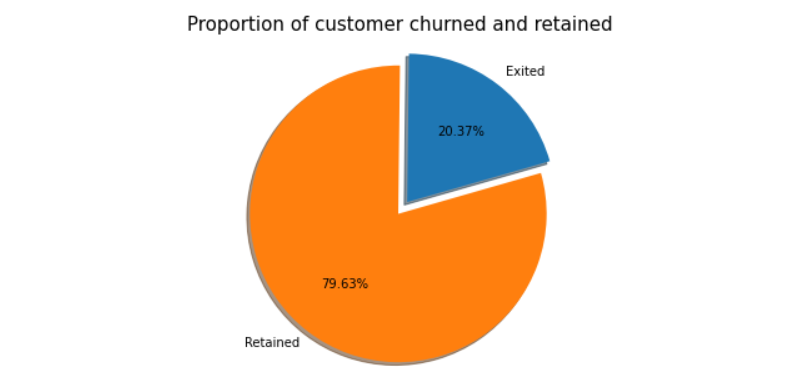


检测缺失值

根据输出结果可知，数据集中不存在缺失值。

# 3 数据预处理

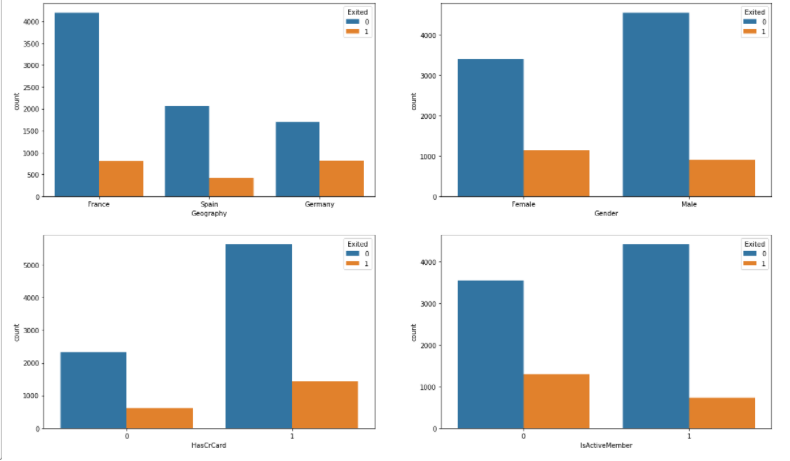
## 3.1 剔除无关量并可视化分析

将前3个无关特征变量——行号（RowNumber）、客户编号（CustomerId）、客户姓氏（Surname）排除在外并进行可视化分析。

流失比例可视化

从上图可得，数据集中大约有20%的客户流失了。由于不知道是什么原因引起了用户流失，所以我们接下来要根据流失用户与现存用户的特点分布来了解用户流失的原因。

## 3.2 数据分类可视化



以地域、性别、是否持有信用卡、活跃程度作为条件可视化数据

由图可以发现数据分布有以下特点：

1、大部分数据来自法国。 但是，流失客户的比例与客户数量成反比，因此银行在客户较少的地区可能存在问题（可能没有分配足够的客户服务资源）。

2、女性顾客流失的比例高于男性顾客。

3、大多数流失的客户是拥有信用卡的客户。

4、不活跃的客户流失率更大。总的来看，不活跃的客户总体比例相当高，银行可能需要采取措施，将该群体转变为活跃客户，从而改善客户流失的状况。

## 3.3 数据条件可视化

现存用户与流失用户特点分布可视化

由图可以发现数据有以下特点：

1. 现存客户和流失客户之间的信用评分分布没有显着差异。
2. 年龄较大的客户比年龄较小的客户流失更多。银行需要采取针对年长客户的措施来进行留存。
3. 对成为银行客户年数而言，处于极端状态的客户与处于平均成为银行客户年数的客户相比，更容易流失。
4. 银行正在流失拥有大量账户余额的客户，这很可能会耗尽其可用的借贷资金。

客户使用产品数量和估计薪资均对于客户流失的概率没有重大影响。

总的来说，可以根据留存用户与流失用户的特点进行特定性针对，例如年龄、用户状态、账户余额等因素。

# 4 建模

## 4.1 搭建环境

from pycaret.classification import \*

exp = setup(data = df\_train , target = 'Exited', session\_id=123)

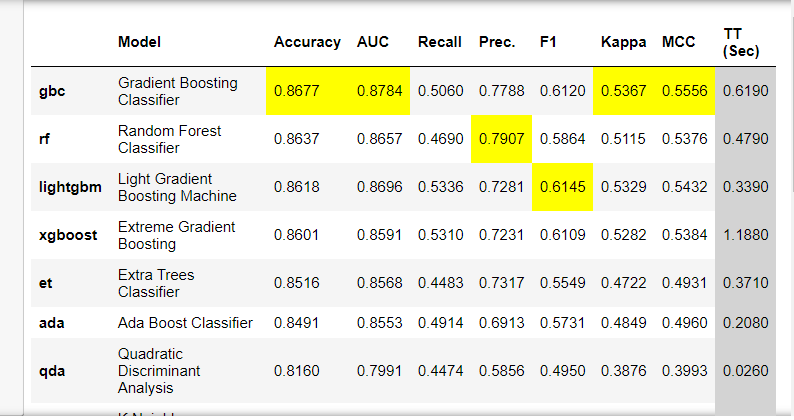
PyCaret中机器学习实验的第一步：通过导入所需的模块并初始化setup()来设置环境。函数setup()初始化pycaret中的环境，并创建转换管道以准备建模和部署数据。在pycaret中执行任何其他函数之前必须调用 setup()。它接受两个必需参数：pandas数据帧和目标列的名称。

当执行setup()时，PyCaret的推理算法将根据某些特征自动推断所有特征的数据类型。但有时并非总是能正确推断数据类型，为此，PyCaret显示一个表，其中包含执行setup()后的特征及其推断的数据类型。在PyCaret中，确保数据类型是正确的是非常重要的，因为PyCaret会自动执行一些预处理任务，这对于任何机器学习实验都是必不可少的。对于每种数据类型，这些任务的执行方式不同，因此正确配置非常重要。

这里使用的模块是pycaret.classification。导入模块后，将通过定义数据框（‘ df\_train’）和目标变量（‘Exited’）来初始化setup()。其中session\_id 是指为所有函数设置的伪随机数种子。如果没有传递session\_id的值，会默认生成一个随机数作为session\_id的值。在此实验中，为了复现实验数据，指定session\_id的值为123。

## 4.2 比较模型

compare\_models()



compare\_models()使用默认的超参数训练模型库中的所有模型，并使用交叉验证评估性能指标。它返回经过训练的模型对象。使用的评估指标是：

分类指标：Accuracy, AUC, Recall, Precision, F1, Kappa

回归指标：MAE, MSE, RMSE, R2, RMSLE, MAPE

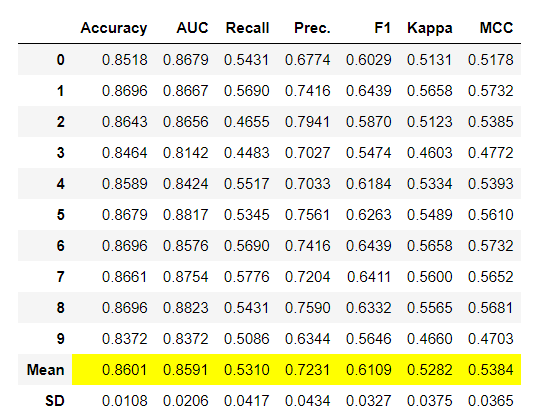
该函数的输出是一个表格，显示了所有模型在折痕处的平均得分。可以使用compare\_model函数中的fold参数定义折叠次数（默认情况下，折页设置为10）。默认情况下，对于分类实验，表按Accuracy排序。如上表显示。

## 4.3 创建模型

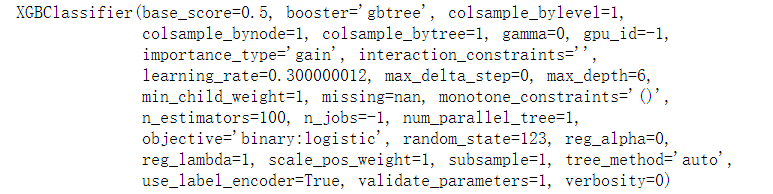
选用三种分类模型（Extreme Gradient Boosting、Gradient Boosting Classifier、Decision Tree）

**1.Extreme Gradient Boosting**

xgboost = create\_model('xgboost')

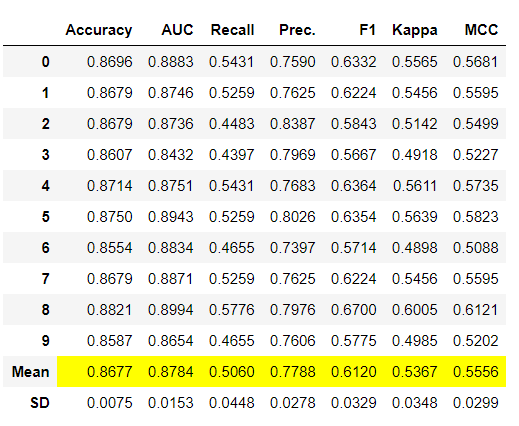


print(xgboost)

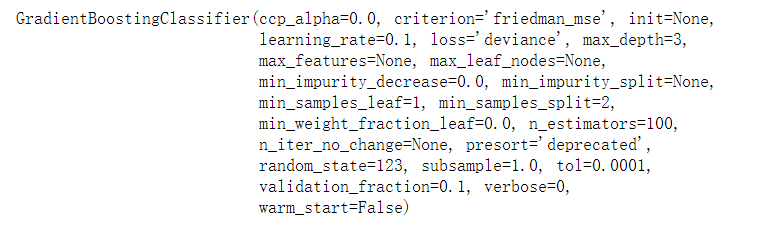


**2.Gradient Boosting Classifier**

gbc = create\_model('gbc')

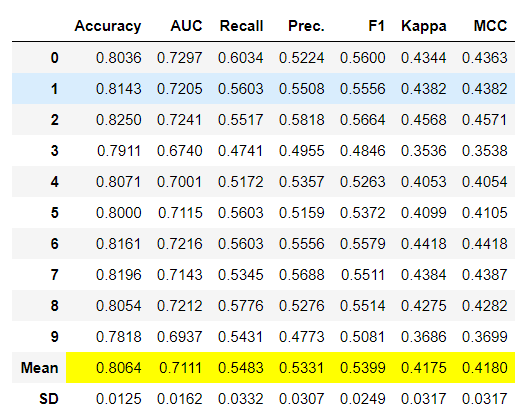


print(gbc)

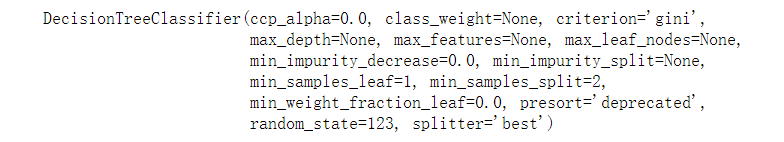


**3.Decision Tree**

dt = create\_model('dt')

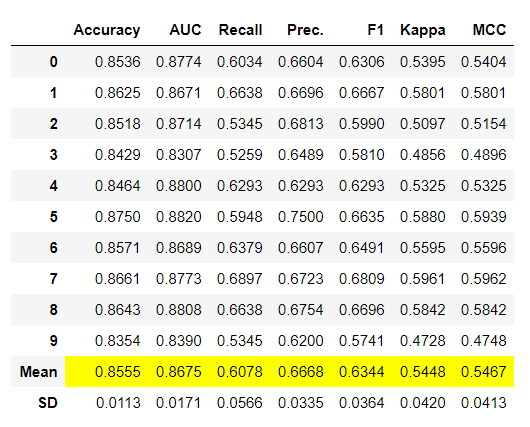


print(dt)

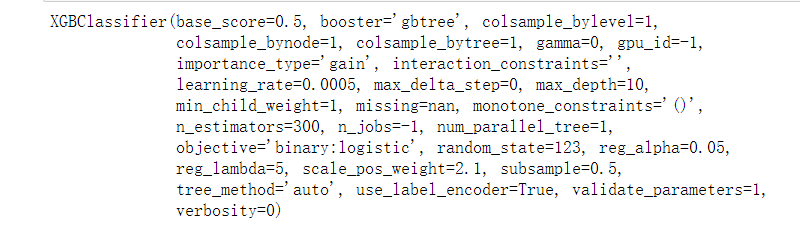


### 4.3.1 调优模型

tuned\_xgboost = tune\_model(xgboost)

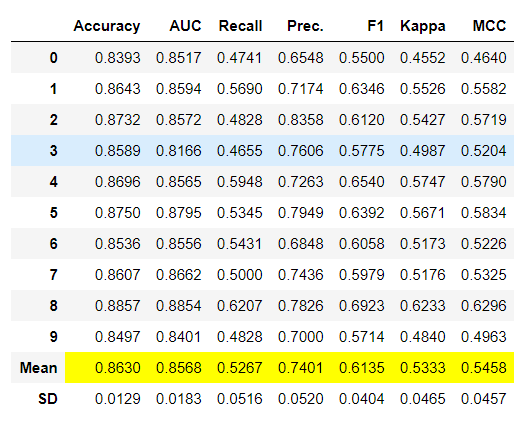


print(tuned\_xgboost)

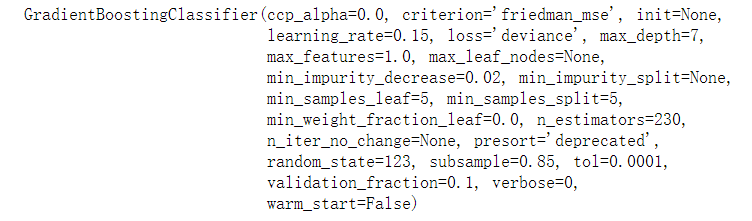


**1.Gradient Boosting Classifier**

tuned\_gbc = tune\_model(gbc)

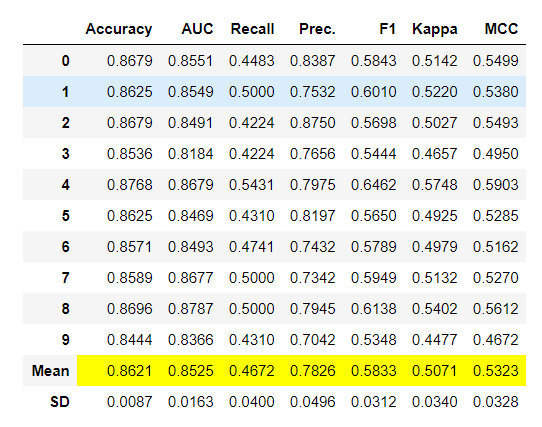


print(tuned\_gbc)

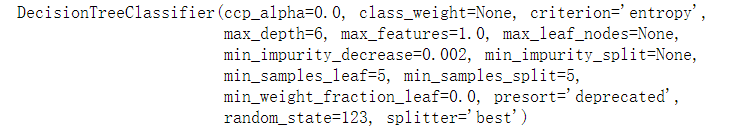


**2.Decision Tree**

tuned\_dt = tune\_model(dt)



print(tuned\_dt)



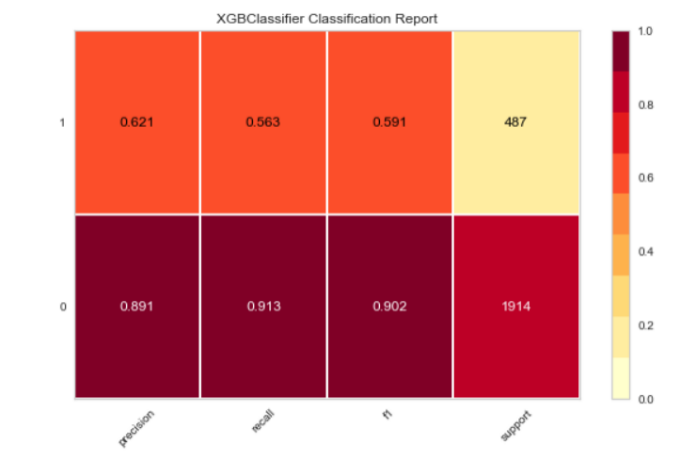
tune\_model（）函数使用带有完全可定制的预定义网格的随机网格搜索来调整作为估计量传递的模型的超参数。优化模型的超参数需要一个目标函数，该目标函数会在有监督的分类实验中自动链接到目标变量。

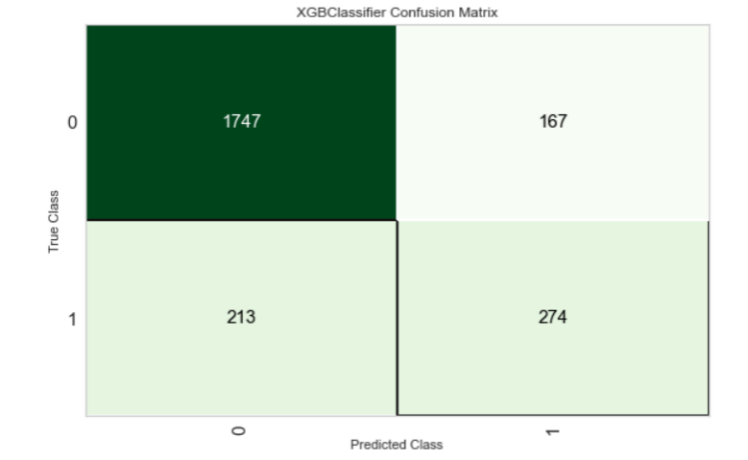
实验主要目的是预测可能会流失的客户，以便银行可以采取措施来防止流失，因此召回率（Recall）指标得分更为重要。根据以上结果的对比，将Extreme Gradient Boosting作为最优模型。

# 5 评价

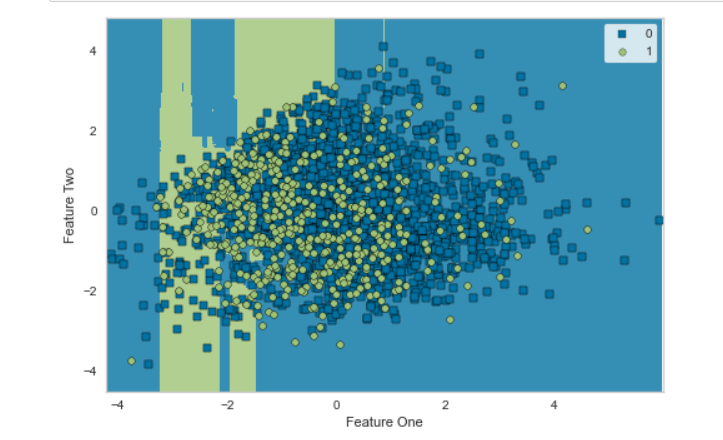
## 5.1 评估模型

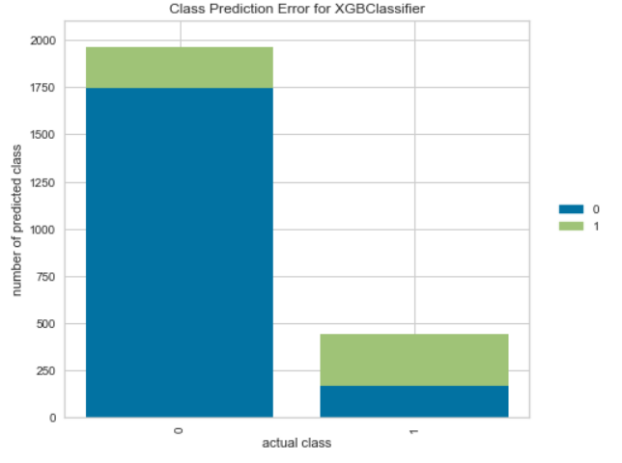
plot\_model(tuned\_xgboost, plot = 'confusion\_matrix')

plot\_model(tuned\_xgboost, plot = 'class\_report')

图1 图2

plot\_model(tuned\_xgboost, plot='boundary')

plot\_model(tuned\_xgboost, plot='error')

图3 图4

evaluate\_model(tuned\_xgboost)

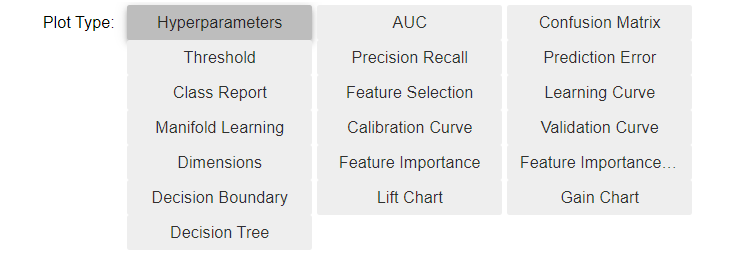


图5

## 图1所示为confusion\_matrix，即混淆矩阵。混淆矩阵是机器学习中总结分类模型预测结果的情形分析表，以矩阵形式将数据集中的记录按照真实的类别、分类模型作出的分类判断，两个标准进行汇总。其中矩阵列代表预测分类（predicted class）,行代表实际分类（True class）,当预测分类与实际分类相同即矩阵对角线为正确的预测结果。可以将其化简为下图：

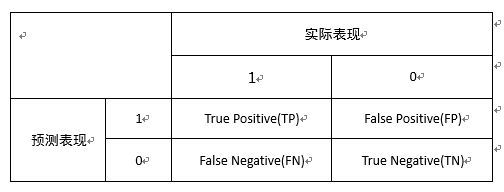


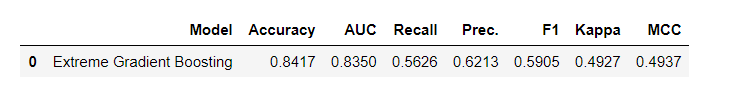
图2所示为class\_report，class\_report函数用于显示主要分类指标的文本报告。图中了显示每个类的精确度（precision），召回率（recall），F1值（精确率和召回率的调和平均数）、support（每个样本出现的次数）等信息。其中精确度=正确预测的个数(TP)/被预测正确的个数(TP+FP)，即可显示模型预测的结果中有多少是预测正确的；召回率=正确预测的个数(TP)/预测个数(TP+FN)，即可确定某个类别测试集中的总量有多少样本预测正确；F1 = 2\*精度\*召回率/(精度+召回率)。

图3为决策边界（boundary），所谓决策边界就是能够把样本正确分类的一条边界，主要有线性决策边界(linear decision boundaries)和非线性决策边界(non-linear decision boundaries)。

图4、图5分别为Class Prediction Error（类预测误差）和evaluate\_model，Class Prediction Error可以预测模型与目标之间的距离，evaluate\_model通过传递模型对象，PyCaret将创建一个交互式窗口，从而以所有可能的方式查看和分析模型。

## 5.2 测试模型

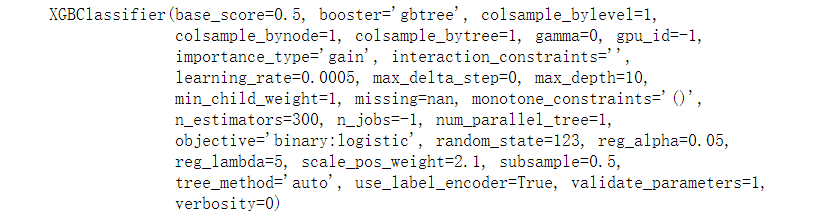
predict\_model(tuned\_xgboost);



## 5.3 部署模型

final\_xgboost = finalize\_model(tuned\_xgboost)

print(final\_xgboost)



## 5.4 模型预测

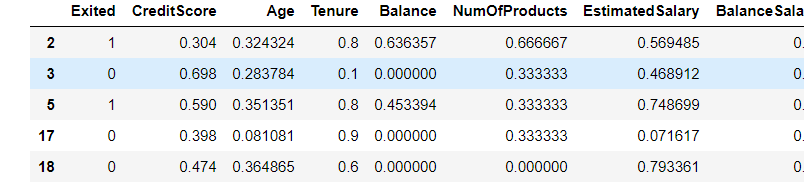
df\_test = DfPrepPipeline(df\_test,df\_train.columns,minVec,maxVec)

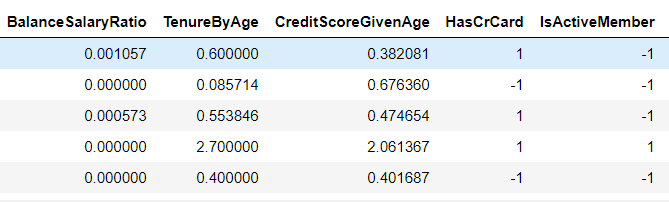
df\_test = df\_test.mask(np.isinf(df\_test))

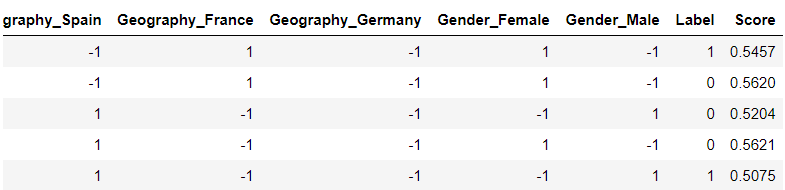
df\_test = df\_test.dropna()

unseen\_predictions = predict\_model(final\_xgboost, data=df\_test)

unseen\_predictions.head()

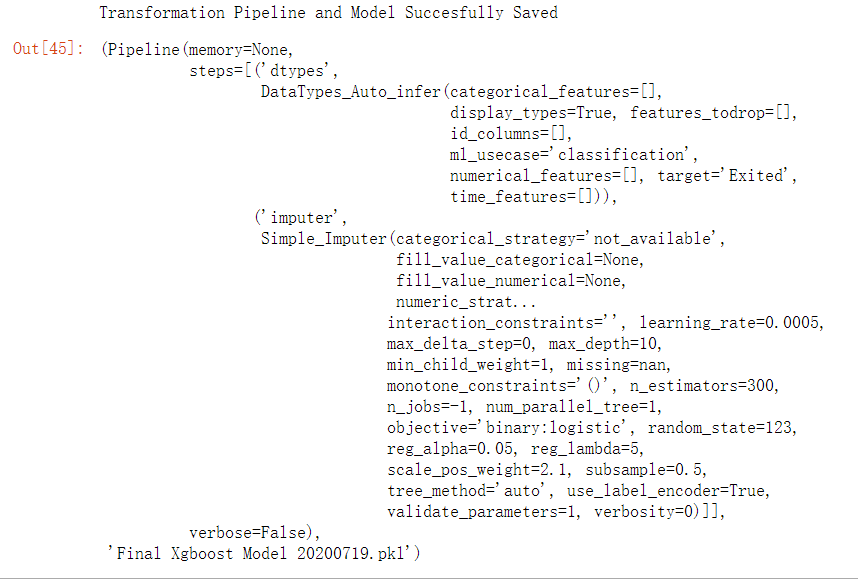






## 5.5 保存实验

save\_model(final\_xgboost,'Final Xgboost Model 20200719')



# 6 结论

在识别影响因素方面，由实验可看出，信用分数、客户所在地、性别、年龄、成为银行客户年数、账户余额、客户使用产品数量、是否拥有本行信用卡、是否为活跃用户、估计薪资均对银行客户流失有一定的影响，银行可以根据识别结果对症下药，采取解决措施。

在预测方面，模型对未见的测试集数据的预测结果精度略高，银行可以运用此模型作为参考，帮助其识别流失客户。但是，尽管此模型具有很高的准确性，但仍存在未识别出流失客户的情况，可能是还有其他的特征变量未考虑进来，或者是数据处理还不够恰当、准确，导致模型存在误差。因此，模型仍有改进的空间。

# 7 建议

1. 目标人群定位。根据特征重要性排序，特征估计薪资和存贷款情况是影响预测客户流失的较为重要因素。结合之前的特征频数分布图得知，该银行客户各收入层分布均匀，说明银行并没有重视目标收入客户定位，没有明确目标人群，更多客户选择存入较少的钱。目标客户定位有助于银行指定针对性策略，提供适合目标客户的商品和服务，提高银行的竞争力

2、用户关怀策略。客户使用该银行产品年数也是影响模型的重要因素。为了挽留老客户，推出相应的关怀和激励策略十分有必要。对于使用产品年数较长的客户，银行可以采用会员积分制、会员优先服务、老客户感恩回馈等活动，令客户感受到银行对客户的重视；对于银行新用户，可通过各种优惠活动吸引客户使用，如与政府和商家合作推出打折活动，同时也要通过会员积分制激励新用户向老用户的转换。

3、高信用评级激励制度。信用得分也是预测客户流失的重要影响因素，其是银行用来评估客户是否具有获得信用卡的资格。为了挽留银行高信用用户，银行可推出一系列信用卡激励策略，一来可以挽留更多高信用用户，二可以刺激其他未办理信用卡的高信用评级用户申请该银行信用卡。