# 技术文档说明

Python 文件目录结构

MLcode :

---| 读取tranaction3ToExcel.py

---| 特征工程.py

---| 模型运行代码.py

---| 随机森林模型调优.py

---| XGBoost森林模型调优.py

---| GradientBoost模型调优.py

## 1 文件介绍

* 1. **读取tranaction3ToExcel.py**

该文件是将原始的压缩包中的transaction3.txt数据转换成 excel文件，方便后续数据的处理。

* 1. **特征工程**

将上面得到的excel文件用特征工程的手段进行处理，得到最终模型训练需要的数据集格式,去掉一些特征列，新增一些特征列。

* 1. **模型运行代码**

使用了随机森林，XGboost模型和 Gradientboost模型进行模型的训练

* 1. **模型调优**

由于随机森林模型，XGboost模型以及 GradientBoost模型的参数需要调优，我们使用了**贝叶斯调优**的方法，找到最好的的参数设置。这一步我已经将结果设置好了（在模型训练中使用了）。我们将其写在三个python文件中。

---|随机森林模型调优.py

---| XGBoost森林模型调优.py

---| GradientBoost模型调优.py

**1.4.1 调优结果**

随机森林(RandomForest)模型调优参数结果 ：

{

'target': 0.04561587665065229,

'params':

{

'max\_depth': 6.890916335318524,

'max\_features':0.2612010992466726,

'min\_samples\_leaf':18.72567375906658,

'min\_samples\_split':23.26154976286628,

'n\_estimators': 40.23885066655579

}

}

XGboost模型调优参数结果 ：

{

'target': 0.9837913699204023,

'params':

{

'max\_depth':8.248147138262466,

'n\_estimators': 263.2082839288204

}

}

Gradientboost调优参数结果 ：

{

'target': 0.8837913699204023,

'params':

{

'max\_depth': 5,

'max\_features':0.4,

'min\_samples\_leaf':10,

'min\_samples\_split':2,

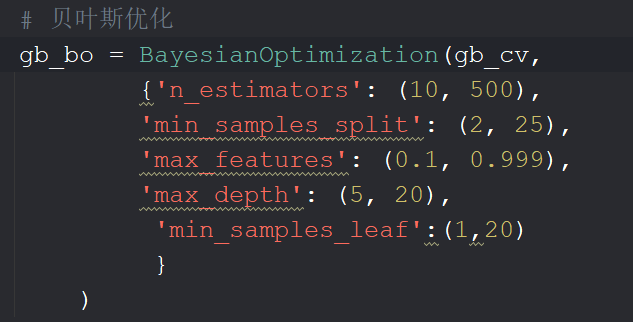
'n\_estimators': 488

}

}

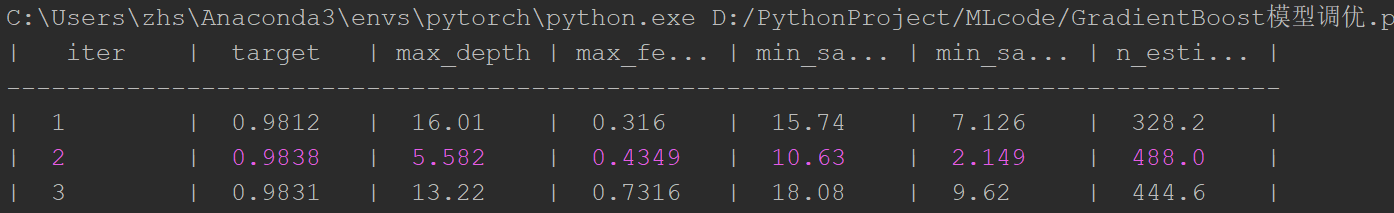
**1.4.2 调优举例**

调优举例 （拿GradientBoost调优举例）：



由于时间仓促，参数的范围设置的可能不一定好，您在代码中可以尝试修改参数调整的范围，以得到更好的调优结果。比如 ‘n\_estimators’ 参数范围可以修改成(10,800)等等,‘max\_depth’可以修改成(5,30)等等。

**调优的过程：**



将调优得到的参数设置在**模型运行.py**中模型中即可。

## 2 未完成的部分

## 第五步特征工程(Feature Engineering)

## 第10步骤未做

* “cardLast4Digits”
  + Since this column indicate different cards used by a single customer, create columns “n cards used by customer”, “number of incorrect enters of CVV by this card”, “numbers of transactions made by this card”

第11 步骤中未做的

* Create column “transactionDateTime\_absolute” to Calculate the absolute eplapsed time for transaction time

但其实最后的效果差别不是很大

特征一共22个，比要求少5 个

n\_transac\_by\_merchant

n\_transac\_by\_this\_c1tomerID

Country\_not\_1

n\_incorrect\_CVV\_by\_card

n\_address\_change\_by\_c1tomer

## 3 实验环境

需要装sklearn相关的包

pip install -U scikit-learn

安装贝叶斯相关的包

pip install bayesian-optimization

贝叶斯调优可参考https://blog.csdn.net/weixin\_43819931/article/details/125172266