北京航空航天大學
BEIHANG UNIVERSITY

风云杯机器学习竞赛

方案说明文档

队伍：改变自己队

成员：李磊 梁旭磊 王国霞

1. 总体思路

1.1. 问题定义

本次题目为二分类问题。所用数据分为三类：基本信息、通话详单、第三方征信，预测标签为 1 的概率。数据具体信息不详，能够猜测含义的文本变量有：职业、家庭住址、注册地址、通话记录（双方通话地址）等。

1.2. 方案设计

我们的方案整体上使用了 Stacking 的框架，如下图：

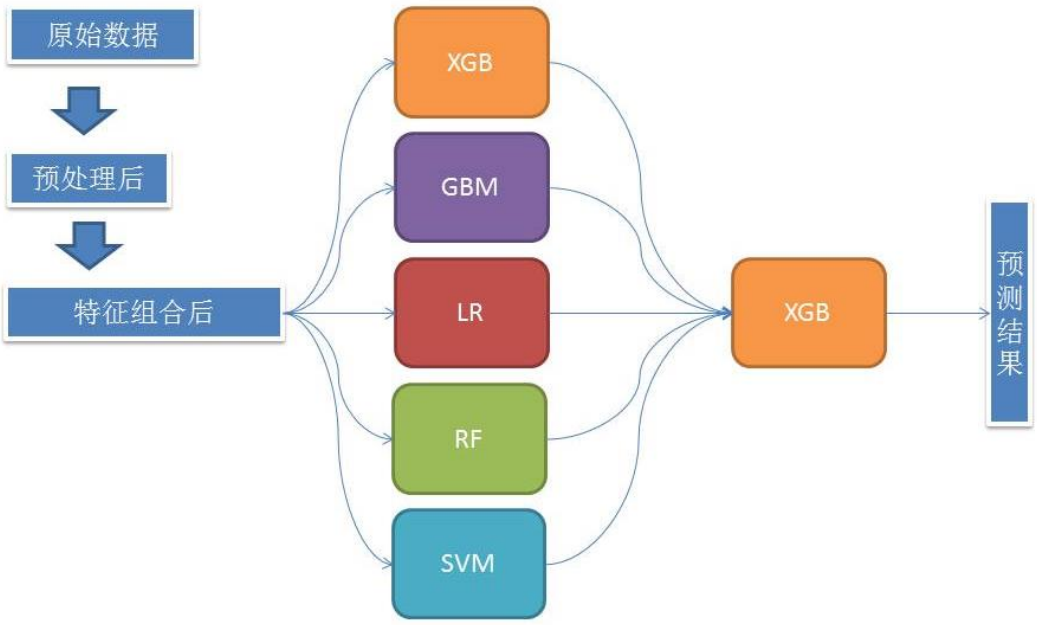

图 1. 整体解决方案框架图

阶段一：处理原始数据，包括预处理步骤，以及特征工程步骤，特征工程步骤主要是特征组合，由预处理后的数据中比较重要的特征组合得到。

阶段二：使用五个不同的机器学习模型进行学习，并调参，生成 stacking 的特征，这里使用五折交叉。

阶段三: 使用上一层的输出进行模型集成，训练一个 XGB 模型并调参，生成最终预测结果。

1.3. 实现技术

Python + Pandas + XGBoost + Sklearn

2. 计算环境

2.1. 服务器

CPU：Intel i7

Memory： 64G

GPU：Nvidia 980

Python3.6

XGBoost(GPU)

Pandas Sklearn Seaborn pyltp

2.2. 笔记本

与服务器相同环境，但硬件配置略低，用于编码。

3. 运行顺序

/src/handle_large_noise.py

/src/preprocessing.py

/src/ models/stacking_data.py

/src/models/stacking_models.py

4. 数据预处理

因为本次问题数据维度比较高，但数据与标签的相关性普遍比较低，经过测试对比，对数据降维会导致模型训练效果更差，降维方法

不太适合用于本题目。

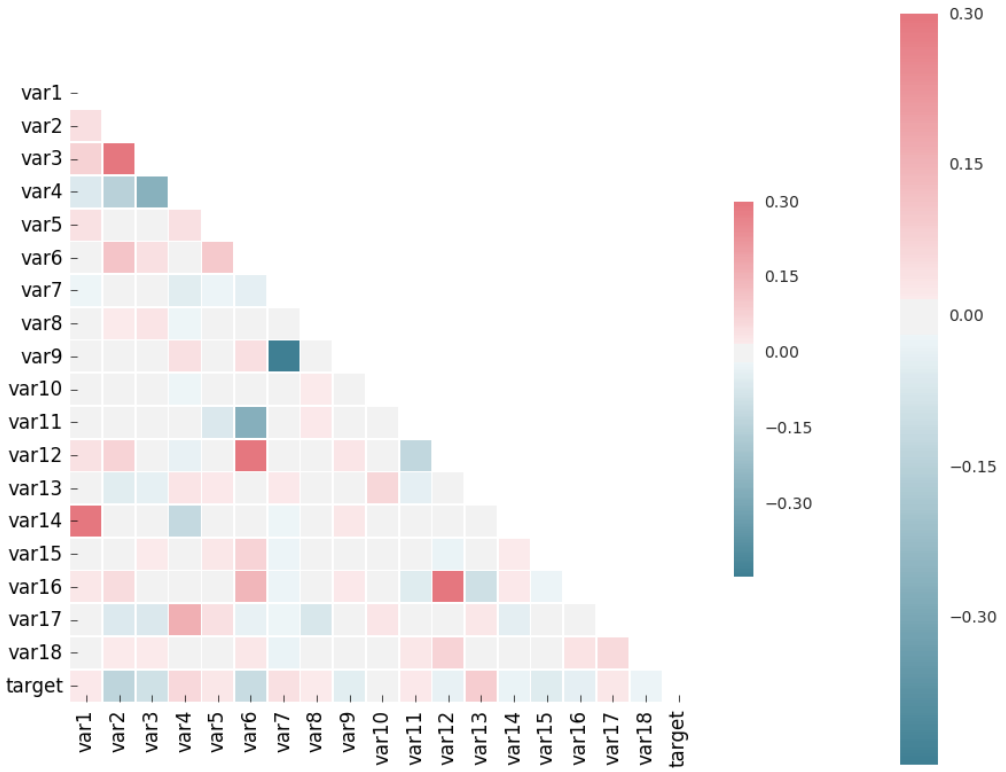

图 2. 基本信息数据与标签的相关性

4.1. 异常值

数值型数据的异常值：

一些异常的数据特别巨大，我们用 max 取代大于 max 的所有数值，max 的计算方法是：从最大值递减，取值为整数*10000，直到本列满足下列条件：

(features[i]>max).sum()<10

且 (features[i]>max-10000).sum() > 10

文本型数据的异常值：

var19 有一些特殊词汇比如‘普通员工’、‘创业人员’、‘家庭主妇’出现频次太少，根据自己理解，使用相近的职业大类填充。

var125 有很多用区号代表地名的用法，也做了处理，但最后特征中并没有用到处理好的这一列。

4.2. 缺失值

数值型：填充 0 或者中位数，视具体分布而定。

文本型：统一填充‘空值’

缺失过多：一些信用数据缺失值过多，直接丢弃。

4.3. 数据变换

基本信息:

基本信息中有 var1/var6/var14 的分布为幂律分布，经过变换可以得到比较好的正态分布。但测试发现，最终特征中同时保留原特征和 log 特征效果更好。

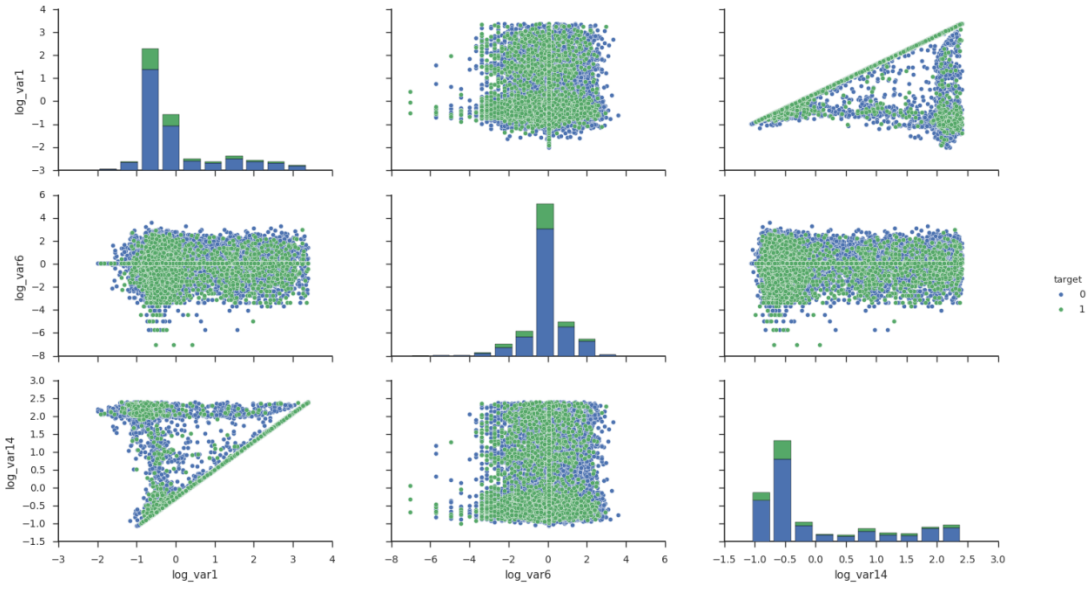

图 3. log_var1/ log_var6/ log_var14 的分布

通话记录:

通话记录中的数据，很多符合幂律分布的，经过 log 变换后可以得到近乎完美的正态分布，对提升预测效果很有帮助。

4.4. 无量纲化

使用标准化方法，将所有数值型数据的分布的方差全部调整为 1。

4.5. 分类数据

将所有包含省名的数据中的省份名字全部放到一起，然后进行哑编码，以及 TFIDF 编码。

5. 特征工程

因为无法知晓变量含义，所以采用交叉构建然后利用树模型筛选的方法。

构建：通过上述处理方法进行预处理后的数据，根据 XGB 的变量重要性排名，取前 70 名，做交叉组合（加、减、乘），产生七千多个维度。

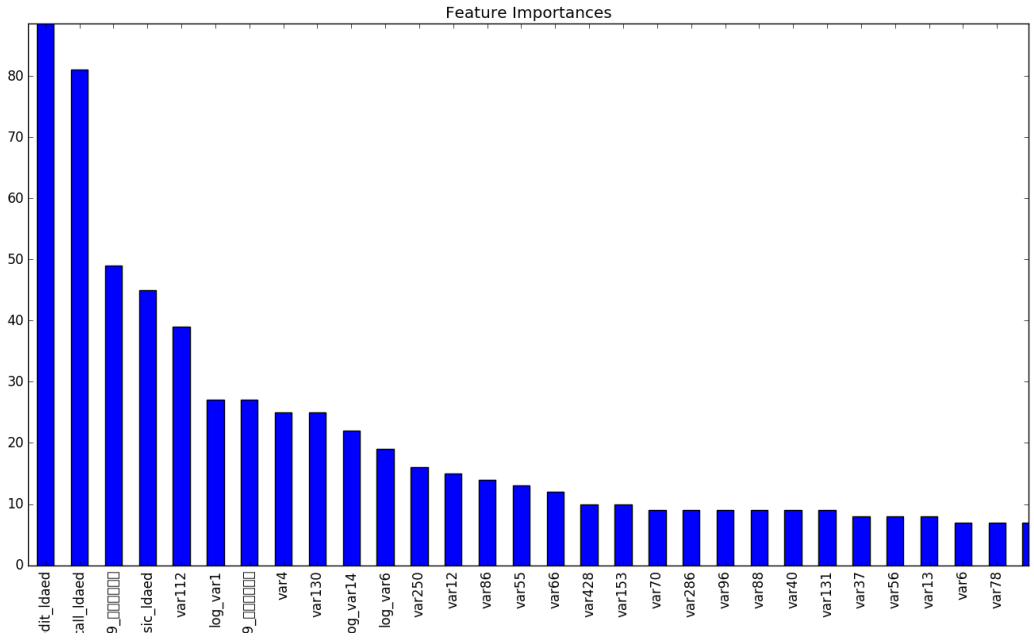

图 4. 预处理后的特征重要性排名

筛选：使用这七千多维度进行训练，选择重要性排名前 180 的组合特征添加到预处理后的训练数据中。

6. 模型选择

树模型 GBDT(XGBoost, sklearn 的 GBM)、Random Forest

线性模型：LR

支持向量机：SVC

使用上述五个模型进行训练并进行相应的调参工作，调参需要使用大量的计算能力。一些模型因为太慢并且效果比较差所以无法使用，如 KNN。

7. 模型融合

最终我们使用的是两层的 Stacking 融合方法。

第一层：上述五个机器学习算法。

第二层：使用一个规模较小的 GBDT(XGB)做为第二层的学习器

Stacking 对学习结果有比较大的提升，每个模型的重要性如下所示：

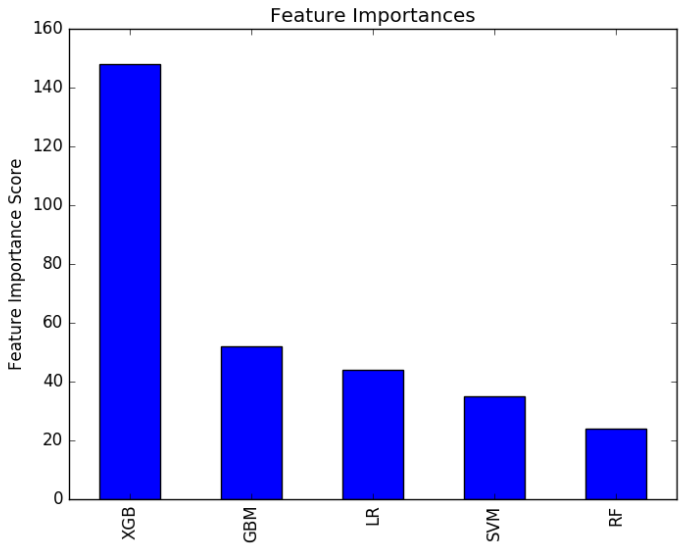

图 5. 不同模型在 Stacking 中的重要程度

8. 算法结果

线下 CV： 0.854076

线上 ： 0.835870

9. 总结和不足

我们三个都是第一次参加机器学习竞赛，都是完全没有经验的小白。从十月二号开始，每天几乎都处在紧张的学习和使用的循环中：前一天看教程知道一个新的库，第二天早上就要看官方英文教程学会，然后下午就要应用到数据上，看对预测结果有没有提升。这期间学习能力真的有很大的提升，拥有快速学习的能力才是这个时代的王道。

排名榜公布后，就更加忙碌。必须要赶在下午四点截止之前把算法参数都调出来，这个时候真的是感叹自己机器性能太弱了，感叹自己没有在晚上写好程序然后直接格点搜索出结果。但后来对榜也看淡了一些，其实隔两天提交一次差不多是正好的节奏，不紧不慢。

还有很多想法没有实现。之前想用 Tensorflow 搭建神经网络做预测，甚至想把每个用户看做一个 20x20 的图像然后用 CNN，但受设备和时间的限制都无法实践，确实还是有些遗憾，但毕竟这次的结果已经是自己拼经全力的结果了，所以整体上还是很满意的。

写给官方组织者：非常感谢这次比赛真的让我们学到了特别多的东西，我们对金融方面也很感兴趣，非常希望能够多了解业内的相关技术以及机器学习实践经验。非常愿意再参加类似的活动，希望将来能获得更多交流的机会~

10. 参考资料

Kaggle 经验

分分钟带你杀入 Kaggle Top 1%

如何在 Kaggle 首战中进入前 10%

Kaggle 首战拿银总结 | 入门指导（长文、干货）

Approaching (Almost) Any Machine Learning Problem | Abhishek Thakur

七月在线-kaggle 案例实战班（原课程链接）【百度网盘链接】 密码：kmg2

特征工程

特征工程到底是什么？

Discover Feature Engineering, How to Engineer Features and How to Get Good at It

调参经验

XGBoost 参数调优完全指南（附 Python 代码）

Complete Guide to Parameter Tuning in Gradient Boosting (GBM) in Python

机器学习算法调优

模型融合

Kaggle 机器学习之模型融合（stacking）心得

【机器学习】模型融合方法概述