**2019数据智能算法大赛**

**复赛算法说明书**

**小组：一剑疾风斩万灵**

目录

[一、数据集划分 3](#_Toc25828508)

[二、数据预处理 3](#_Toc25828509)

[三、特征工程 3](#_Toc25828510)

[四、模型构建思路 4](#_Toc25828511)

[1. 主要模型 4](#_Toc25828512)

[2. 模型搭建 6](#_Toc25828513)

[五、模型后处理 6](#_Toc25828514)

[六、模型亮点 6](#_Toc25828515)

[七、运行环境 6](#_Toc25828516)

# 一、数据集划分

使用了初赛和复赛提供的数据，将俩个数据合并成一个数据集，然后以付款时间为2013年10月31日为分割线分割训练数据与标记。同时在生成提交的验证集时采取了两种方案：第一种是直接对复赛数据进行特征提取，第二种是对初赛和复赛合成之后的数据进行特征提取。

# 二、数据预处理

通过数据清洗，处理掉没用的数据

1. 删除类别唯一的特征
2. 删除某一类别占比超过*95%*的列
3. 删除异常值

# 三、特征工程

针对本次比赛的具体问题我们的特征工程分为三大部分：用户购买特征，用户地域特征，购买商品特征。

用户特征如下：

1. 用户购买次数
2. 用户的所有父订单的付款金额的最大值，最小值，平均值，以及总和
3. 用户的会员状态
4. 用户性别
5. 用户购买商品时间与该商品上架的时间差
6. 商品上架时间
7. 用户购买的订单中商品总数
8. 用户购买的父订单所含商品数
9. 用户购买层级(以上半年所有付款金额的均值为分割)
10. 用户这次购买与上次购买时间的时间差

用户地域特征：

1. 对用户所在省份进行编码
2. 用户所在省份的当年人均*GDP（数据来源于网络，用于反映消费水平）*

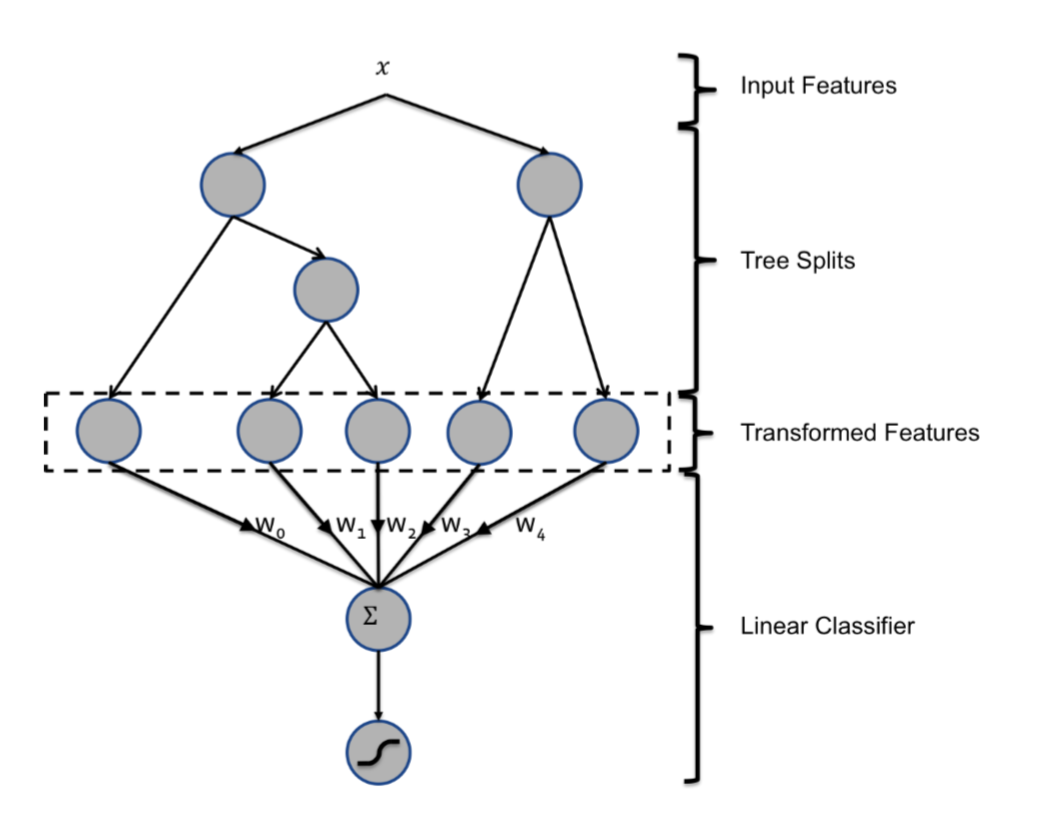
购买商品特征：

1. 商品的计数特征

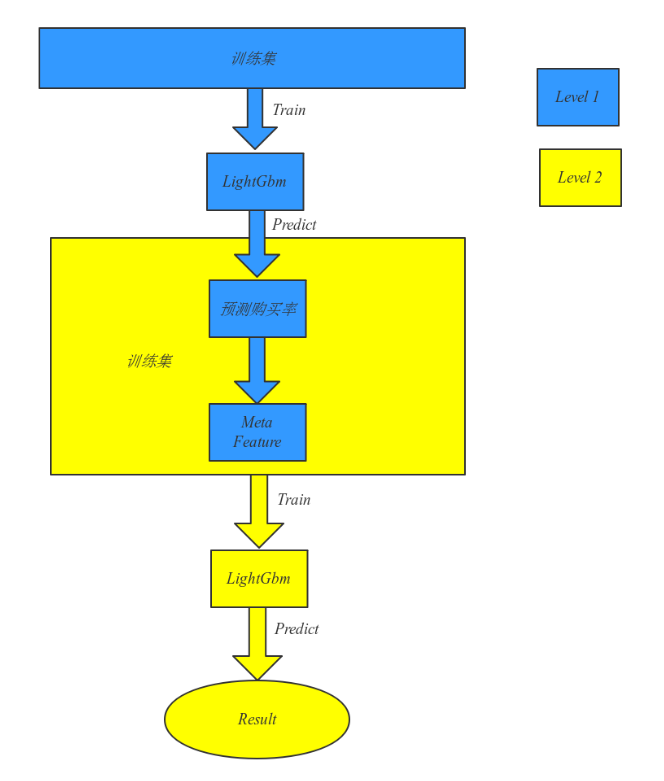
# 四、模型构建思路

## 1. 主要模型

1. Facebook 2014年的文章提出了通过GBDT解决LR的特征组合问题，随后Kaggle竞赛也有实践此思路，*GBDT+LR*模型是业界的经典模型，我们采用这种思路来搭建模型，其模型架构图如下：



1. 第二个模型是参考*LightGbm*的二次训练，采用迁移学习的方法，其思路为首先使用*LightGbm*对数据进行训练，然后将模型的预测值作为新特征加入数据中，在使用另一个*LightGbm*模型对新的数据集进行训练。*Level-1* 将使用前几个月的所有数据对这一时期的购物行为建模，并将它的预测作为第二个模型的输入，这样就能避免前几个月与待预测的双十一购物狂欢之间的剧烈变化。此外，第一个模型的输出特征还为第二个模型提供先验知识，因而能与之前的数据一起更高效地建模购物行为。其模型架构图如下：



## 2. 模型搭建

1） 训练单模型：上述的每个模型都使用同一份数据进行训练，但在预测的时候使用两份测试集进行预测，一份是仅使用复赛数据提取的数据，另一份是使用初赛和复赛结合的数据提取的数据。

2） 模型融合：直接对上述模型产生的四个结果进行均值计算得到最终预测结果。

# 五、模型后处理

在训练*LightGbm*时我们发现其对购买率的最大值仅有0.6，而根据业务的需求，我们需要进一步增加模型的*recall，*故我们对2013年购买次数超过十次的用户将其购买率增加*30%*。在训练*GDBT+LR*模型时尚未发现此现象，故未对其使用后处理。

# 六、模型亮点

1. 简单有效的特征提取；
2. 迁移学习策略*，LightGbm*模型的二次训练，防止前后数据震荡变化对模型的影响；
3. 模型的后处理策略；
4. 模型数量较少，且每个模型复杂度低。

# 七、运行环境

**操作系统： Windows10**

**Python版本：3.5**

**依赖库：scikit-learn,numpy,pandas,tqdm,lightgbm**

# 八、代码运行顺序

首先运行*generate\_feature.ipynb*生成特征文件，其次运行模型文件*gdbt+lr.ipynb，gdbt+lr1.ipynb，lgb\_transfer.ipynb，lgb\_transfer1.ipynb*最后运行模型融合文件*ensemble\_models.ipynb，*其最终结果为在*Res*目录下的*res\_final.csv*