信息与软件工程学院

综合设计 I 中期报告

课程名称：综合课程设计 I

课题名称：基于腾讯广告数据日志的点击率预测方法

指导教师：刘梦娟

所在系别：软件工程（ 互联网+）

执行学期：第三学期

学生信息：

序号	学号	姓名
1（组长）	2016220302033	胡蝶
2	2016220901031	张熙悦
3	2016220901027	饶韵恒
4	2015220301025	姚卓琛

摘 要

本次综合课程设计，我们小组要针对老师给出的“腾讯”移动 app 广告投放日志设计点击率预测算法。其中需要我们掌握的关键操作流程包括：数据清理及训练集和测试集划分、机器学习模型的选择、通过训练集数据学习模型参数、并利用测试集数据对点击率预测模型的性能进行测试。我们选择了 python 语言进行相关代码的学习和利用，截至中期已经掌握了数据分类整理和特征选择相关知识，实现了最基本的学习模型，掌握了相关的学习模型算法。

关键词：点击率预测、机器学习、逻辑回归

目 录

第一章 综合设计的进展情况

1.1 需求分析与建模

1.1.1 点击率预测的需求分析

点击率预测（CTR）是广告系统中最重要的模块，点击率预测的准确度直接跟收入挂钩。传统行为定向广告投放是对用户行为和点击的广告直接建模，面临冷启动，数据稀疏，以及模型更行慢，预测实时性不足的情况。目前来说，CTR 场景的主要难点有：

1. CTR 需要预测单个用户的广告点击率，然而我们不可能给单个用户展示足够多次广告，从而得到点击率的无偏估计。与其说我们预测的是单个用户的 CTR，不如说我们预测的是这一类用户的 CTR。

2. CTR 需要的结果不仅仅是广告的排序，而是需要一个精确的点击率数值，结合点击率才能精确地为本次广告展示出价。所以点击率预测越准，广告系统收入越高。

3. 用户对多个广告的点击率的可比性：如果广告跟广告 2 是由两个不同模型得到的，那这两个点击率能否直接进行数值上的对比？

1.1.2 模型的构建和分析

带着这三个需求，我们可以做个假设。假设用户的兴趣在 2 维平面上服从正态分布（也可以是更高的维度，为了方便画图我们使用 2 维），每个用户就是一个平面上的点。广告根据它的文案，风格，产品内容的不同，也把它抽象成兴趣空间中的一个点（广告画像）。用户兴画像跟广告画像越 match，点击的概率越大，相应的 python 代码如下：

```
# N 个用户
    user = np.array([normal(mu, sigma, N),normal(mu, sigma, N)]).T
# N 次展示机会，随机展示一个广告
```

```
        item=randint(3,size=N)
        for i, ad_i in enumerate(item):
    # 计算用户画像与广告画像的欧式距离
        d = np.linalg.norm(ad_feature[ad_i]-user[i,:])
    # 用户以 exp(-alpha*d) 的概率点击广告 alpha 也是广告的自有属性
    # expo()函数保证点击率在［0，1］
        label[i] = binomial(1,exp(-alpha[ad_i]*d))
```

CTR 问题一般用分类模型输出的“是正样本的概率”来作为广告的点击率，有下面三种建模思路：

1.单一模型二分类：把用户画像跟广告画像怼在一起，例如这种形式 ［用户画像，广告画像， label ］，这样我们就可以把所有数据都输入到一个二分类模型，然后用这个模型针对三个广告输出三个点击率。

2.多模型二分类：对每个广告都根据正负样本生成一个二分类模型，把用户的画像分别输入到三个模型中得到三个点击率。

3.单一模型多分类：构造 4 个类，3 个点击广告的用户群各为一类，没点击任何广告的用户为另一类。

1.2 实施方案与可行性研究

算法框架为：数据处理、特征提取、模型训练、模型融合。

1.2.1 数据处理（包括特征构造和特征划分）

由于数据量较大，需要对训练集进行筛选，合理选用数据作为训练集。对数据进行预处理，过滤掉平均转化回流时间较长的 APP。针对测试集和训练集中的 id 分布做数据分析，和工具有一定关联另外，训练验证集的划分方式可以使用五折交叉验证的方式。可能出现训练集筛选掉了一大半的数据，但是五折交叉验证跑数据的全集仍然十分耗费时间的情况，此时可以考虑选用随机采样的方法来验证特征的有效性，这样能最大限度的提高工作效率。

针对特征构造，首先要明确，特征主要包括三大类：用户信息，广告信息，上下文信息。基础的特征都是离散的 ID 信息，对于一些类别比较少的 id 的可以做

onehot 编码或者使用 wide&deep 这样的模型，根据预测集也可以简单构造一些 leakage 的特征。

在构造的特征的时候，如果直接暴力遍历特征，那样子对机器的要求太高，而且准确度也不一定高，拓展性不强。我们应该选取重要性高的特征，从业务方面分析一些模型没有办法自动组合出的特征。比如一个用户的历史点击序列（01 串）之类的特征，额外的辅助文件生成的特征。其实很多的特征从业务上就能够判断是否关联偶尔，比如上下文的特征。如果使用树类模型的话，不需要做归一化处理，但是至少得在前面生成转率特征之类做平滑处理，处理缺省值的情况。而且在前期做归一化，离散化地处理对后面的其他模型的融合大有帮助。

1.2.2 特征提取

特征提取部分，分为四项：

1）基础特征，包括用户的基本特征、广告的基本特征、上下文特征；

2）统计特征，对基础特征进行交叉后再统计，包括 count 操作和 unique 操作；统计类的特征，如果只是用 python 的字典硬刚统计，会耗费大量时间，这是应该考虑使用 pandas、sklearn 等机器学习的好用的工具，或者抽取一部分样本放进去 xgboost 进行重要性排序。

3）时间相关特征，主要需要统计用户或用户-App 在前一段时间内的点击次数或者安装次数；

4）概率估计特征，需要对很多 ID 类特征，包括交叉 ID 类特征做概率估计。

需要注意的是，包括转化率特征、历史点击量特征，以及用户、App 与广告位的一些组合特征。在利用用户转化率时需要进行恰当的分类，以免出现拟合的现象。

1.2.3 模型训练

模型上，主要设想有 4 种不同的模型，包括一个传统的 GBDT 模型，以及另外三个深度学习模型，分别是 wide&deep 网络、pnn 网络和 nffm 网络。GBDT 模型我们选择的是 lightgbm，训练速度会非常快，在验证特征有效性方面可以大大缩短时间。而 wide&deep 网络、pnn 网络和 nffm 网络，都可以使用 tensorflow 和

tflearn 自己实现。

除此之外还有 FFM、LR、GBDT、ET 模型等，主要使用 xgboost、lightgbm 与 ffm 这三个工具，还有 sklearn 的随机森林与 gbdt 的库。对于 FFM 模型，F 数据预处理十分重要，主要是针对特征进行离散化。不同类型的特征的离散化方案是不一样的，比如对于长尾分布明显的特征（比如某个 id 的点击次数）需要先取 log 进行分箱操作。而对于平滑后的转换率类的特征，其实可以比较简单的进行归一化之后直接分箱。而且 gbdt 输出的叶子节点是一个很强的特征，把 gbdt 的叶子节点加入 FFM，可以让 FFM 有一个很大的提升。

1.2.4 模型融合

模型融合采用 Stacking 方法，特殊的也可考虑 FFM 与 GBDT 的 ensemble。

第二章 存在问题与解决方案

2.1 存在的主要问题

广告点击率预测项目中，我们要基于广告转化数据训练转化率预估模型(pCVR，Predicted Conversion Rate)，在广告排序中引入 pCVR 因子优化广告投放效果，提升 ROI。

给出的数据集中共包含 8 个 csv 文件，其中训练数据从腾讯社交广告系统中某一连续两周的日志中按照推广中的 App 和用户维度随机采样，分为广告特征、用户特征、上下文特征三大类，每类再细分为多个具体特征，显得繁多而杂乱，给后续的特征工程建立和模型训练造成了一定困难。而如何从最大限度地从这些数据中提取特征以供算法和模型使用，也是值得思考且尤为重要的一步。另一方面，当前机器学习领域已存在的模型与算法数量可观，如何在这其中作出恰当的选择，甚至根据自己的实际需求对算法进行相应的改进，也是达到尽可能高的预测准确率的关键。

2.2 解决方案

为了对多个特征集进行合并，可使用 python 的一个数据分析包 pandas 中的 pandas.merge()方法对数据进行合并。具体实现代码如：

```
pandas.merge(left, right, how='inner', on=None, left_on=None,  right_o
n=None, left_index=False, right_index=False, sort=False, suffixes=('_x'
, '_y'), copy=True, indicator=False, validate=None)
```

特征选择方面，由于在广告点击这一应用场景下，特征具有一定可解释性，且考虑到运行成本等因素，基于经验的人工选择是简洁有效的。我们可人工挑选多组特征，再通过交叉验证对该组特征的性能进行评估，并不断调整。

为了解决特征不属于同一量纲、存在缺失值、信息利用率低等问题，我们需要对数据进行预处理。使用 sklearn 中的 preproccessing 库来进行数据预处理，便可解决上述问题。

使用 preproccessing 库的 MinMaxScaler 类对数据进行区间缩放以使数据无量纲化的代码如下：

```
    from sklearn.preprocessing import StandarScaler
#标准化，返回值为标准化之后的数据
    StandarScaler().fit transform(iris.data)
```

使用 preproccessing 库的 Imputer 类对数据进行缺失值计算的代码如下：

```
    from numpy import vstack, array, nan
    from sklearn.preprocessing import Imputer
#缺失值计算，返回值为计算缺失值后的数据
#参数 missing_value 为缺失值的表示形式，默认为 NaN
#参数 strategy 为缺失值填充方式，默认为 mean（均值）
    Imputer().fit_transform(vstack((array([nan, nan, nan, nan]),
iris.data)))
```

使用 preproccessing 库的 OneHotEncoder 类对数据进行哑编码以达到非线性的效果的代码如下：

```
    from sklearn.preprocessing import OneHotEncoder
#哑编码，对 IRIS 数据集的目标值，返回值为哑编码后的数据
OneHotEncoder().fit_transform(iris.target.reshape((-1,1)))
```

在模型与算法的选择方面，我们可进行多次尝试，通过交叉验证[1]对各个算法一一进行测试，进行比较，然后调整参数确保每个算法达到最优解，最后选择最好的一个。

[1]交叉验证(Cross-Validation)： 有时亦称循环估计， 是一种统计学上将数据样本切割成较小子集的实用方法。于是可以先在一个子集上做分析， 而其它子集则用来做后续对此分析的确认及验证。 一开始的子集被称为训练集。而其它的子集则被称为验证集或测试集。

第三章 前期任务完成度与后续实施计划

3.1 课程设计完成过程

首先我们确定 python 作为我们开始综合课程设计的语言，并且通过网上搜索相关资料对于“点击率测试”有了一定的了解。我们集体分析了课程设计的需求和目的，并对老师提供的范例进行了针对性的学习和理解。

点击率预估的主要场景，是在各种长尾的流量上，对一个给定的<用户，广告，上下文>三元组做出精准估计。点击率测试的相关算法也是各大公司关注的重中之重，通过网上搜集相关知识，我们选择了人工选择特征加逻辑回归算法的方式。对于用户特征的分类和划分是我们遇到的第一大难题，通过采用老师建议 onehot 编码进行数据处理和特征转化。这里我们采用了使用 sklearn 中的 preproccessing 库来进行数据预处理，解决的编码转化特征的问题。另外对于所给用户信息中的一些缺省值，一开始我们选择直接删除，觉得不影响总体数据的回归预测。但是通过对少量数据的整理，暴露了这一方法的漏洞。通过老师的提醒我们也通过编程计算的方式解决的缺省值的难题。

目前为止我们完成了对于数据的分类处理部分代码的编写并且通过少量数据的测试，并且通过尝试做出了大量数据中对于训练集和测试集的划分。通过应用 LogisticRegression 的算法模型，我们完成了少量数据的测试和预测。

3.2 中期之前的自我评价和之后的自我规划

对于中期之前的课程设计，我们对自己的工作并不是很满意。因为对于机器学习的了解很少，上手和入门都比较缓慢。目前也只完成了逻辑回归一种算法的代码编写和实现。希望接下来的学习过程中我们能够实现对于更多算法模型的理解和调用，并且通过数据预测结果找到一种最适合的算法模型。

参考文献

[1]杨诚. 基于用户实时反馈的点击率预估算法[J]. 计算机应用,2017,37(10):2866-2870.

[2]刘忠宝. 机器学习方法在个性化推荐系统中的应用[J]. 情报探索,2016,(04):80-82.