商场中精确定位用户所在店铺

一、团队与成绩

团队成员

马璐（charon）：中山大学数据科学与计算机学院 2016 级硕士。

许海城（Seaty）:华南理工大学数学学院 2018 届本科生。

李友（Leonardo）：中山大学数据科学与计算机学院 2016 级硕士。

杨亚涛(sysu_Yang)：中山大学数据科学与计算机学院 2017 级博士。

汪志文(buptwangzw)：北京邮电大学。

比赛成绩

初赛：第 5 名；复赛：第 8 名。

二、赛题背景

本次大赛提供在 2017 年 8 月份大概 100 家商场（复赛为 2017 年 7-8 月大概 500 家商场）的详细数据，包括用户定位行为和商场内店铺等数据，参赛队伍需要对其进行数据挖掘和必要的机器学习训练，并对 2017 年 9 月份的商场内用户所在店铺进行预测，结果以准确率作为评估标准。

三、赛题分析与理解

为什么不处理成多分类

根据统计，每个商场大约有 100 个店铺，如果处理成多分类，那分类的类别数有 10,000 左右，如果使用 XGBoost 训练，那么会需要非常大的内存，每个样本的 loss 时间复杂度也会增加。如果你想按商场划分，为每个商场训练一个多分类器，那么分类类别数就会降到 100 左右。那么事实上，这里多分类还有另一个问题，由于目标是确定用户在哪个店铺——涉及了两个对象，因此在构造特征时

会出现这样的情况：你希望构造用户店铺距离特征，那么你就需要为每个店铺构造一个维度，因此你构造特征的维度也会以类别数成倍增加。

尽管有诸多缺点，但是比赛中实际还是有人开源了一个分商场的多分类的解决思路，特征方面他只有一类商场的所有 WIFI 在这条记录上的强度，仔细调整参数后准确率也能达到 91.5%。

如何转化为二分类

通俗地讲，就是把横着的多分类转换成竖着的二分类。

原始数据：

row_id	特征 1	特征 2	特征 3	用户所在店铺		
1	T11	T12	T13	0	1	0
2	T21	T22	T23	0	0	1
3	T31	T32	T33	?	?	?
4	T41	T42	T43	?	?	?

转换后的训练数据：

row_id	特征	待预测店铺	Label
1	T11	Shop1	0
1	T12	Shop2	1
1	T13	Shop3	0
2	T21	Shop1	0
2	T22	Shop2	0
2	T23	Shop3	1

转换后的预测数据：

row_id	特征	待预测店铺	预测概率	预测 label
2	T21	Shop1	0.6	0
2	T22	Shop2	0.8	0
2	T23	Shop3	0.9	1
3	T31	Shop1	0.1	0
3	T32	Shop2	0.9	1

3	T33	Shop3	0.2	0

转化为二分类的问题

可以看到，转换成二分类后，虽然不会遇到特征维度和类别数过多的问题，但是数据的记录数扩大了。实际上，如果简单的将每条记录扩大为商场的所有店铺，那么数据记录数大约会扩大 100 倍。如此多的记录数，会导致在构造特征和训练模型时效率非常低。因此，我们需要通过为每条记录构造候选店铺集进一步减少转化为二分类之后的记录数。

四、候选店铺构造

数据划分

时间	线下	线上
8.01-8.18	特征构造区间	/
8.18-8.24	训练集	/
8.25-8.31	测试集	训练集
8.1-8.31	/	特征构造区间
9.1-9.14	/	测试集

朴素贝叶斯构造候选集

回顾一下我们二分类的目标——“当用户手机出现这串 WIFI 信息时，用户在这个待预测店铺的可能性”，有没有觉得和垃圾邮件分类的场景非常相似——“当邮件中出现这串文本时，邮件是垃圾邮件的可能性”。因此我们采用了朴素贝叶斯来构造候选集，构造的公式如下。

$$\mathrm{P}\left(\mathrm{in\ shop}_k \middle| \prod\left(wifi_{id}^{n}, wifi_{sig}^{n}, wifi_{link}^{n}\right)\right) = \frac{\mathrm{P}(in\ shop_k)\prod \mathrm{P}\left(\left(\mathrm{wifi}_{\mathrm{id}}^{\mathrm{n}}, \mathrm{wifi}_{\mathrm{sig}}^{\mathrm{n}}, \mathrm{wifi}_{\mathrm{link}}^{\mathrm{n}}\right)|in\ shop_k\right)}{归一化项}$$

这里 WiFi 强度是根据分位数做过离散化处理的，另外我们对先验概率 $\mathrm{P}(\mathrm{in\ shop_k})$还做了一些优化：考虑到了不同店铺有不同的高峰时段——KTV 在晚上、奶茶店在下午，我们引入了时间维度，将先验概率转化为了$\mathrm{P}(\mathrm{in\ shop_k}|\mathrm{at\ time_g})$。

除此之外，在具体实现时需要考虑估计的概率中出现 0 的情况：(1)某个时刻

$P(\text{in shop}_k|\text{at time}_g)$的估计值为 0，(2)某个店铺，$P((\text{wifi}_{id}^n, \text{wifi}_{sig}^n, \text{wifi}_{link}^n)|\text{in shop}_k)$为 0.

对于这种情况，具体处理可以采用拉普拉斯平滑，或者简单的取一个小值。最终用这种方式构造的候选集，候选集平均大小为 5 时召回率为 96%，准确率最高达到 89%。

另外我们也尝试了 pagerank，由于不是很清楚如何调整，效果不是很理想【召回率和朴素贝叶斯相当，但是准确率比较低】，有兴趣的可以自己尝试一下。

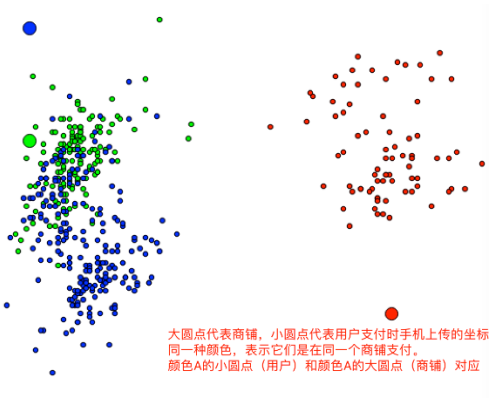
五、特征工程

首先，前面朴素贝叶斯计算到的概率就能直接作为其中的一维特征，另外如果还使用了其他构造候选集的方式都能产出一些指标作为特征。

与传统的二分类不同，这里二分类的目标是“确定用户是不是在待预测店铺”，涉及到了用户和店铺两个对象。因此，在构造特征时需要更加关注于构造用户与店铺的距离这类同时涉及到两个对象的特征，而不是把侧重点放在构造店铺的历史用户数这类只涉及店铺的属性特征。理解了这一点就很好办了。

首先确定用户和店铺有哪些基本信息：WIFI_ID,WIFI_SIG,WIFI_LINK,经纬度。那么 WIFI 方面自然可以构造一些**用户的** WIFI 在**店铺的**出现次数/比例，**用户的** WIFI 的强度与**店铺的** WIFI 强度的比值，**用户的** WIFI 在**店铺的**连接次数/比例，**用户**连接的 WIFI 在**店铺的**出现次数/比例，**用户**连接的 WIFI 的强度与**店铺的** WIFI 强度的比值，**用户**连接的 WIFI 在**店铺的**连接次数/比例。而由于一个用户会涉及多个 WIFI【大多数为 10 个】，可以考虑按 WIFI 强度排序对每个 WIFI 单独构造成多组特征，也可以处理成统计量，合并成一组特征。用户与店铺的距离和方向。

其中方向这一重要特征很容易漏掉，下面结合一幅图解释一下。



如果我们只构造了距离特征，那么红点和蓝点的用户与大红点的店铺距离还算比较相近，这样模型比较难划分出来，如果加入了方向信息，那模型就能很好的把他们划分出来了。

除了这些基本的特征之外，我们还构造了一个经纬度的“KNN 特征”，我们当时是希望构造目标用户的 K 个邻近的历史用户在这个店铺的比例这样一个特征，但是实际运行起来会很慢——不仅要计算距离，还要排序。因此我们最后转换成了计算目标用户的经纬度邻域方格的历史用户在这个店铺的比例——只要计算差值，比较大小，统计比例。

六、算法模型

由于有开源的 XGBoost 多分类，为了提高分数我们还是将多分类的结果作为特征 stacking 到二分类的模型中。而二分类模型我们也是使用了 XGBoost，通过使用不同参数生成多份二分类的概率结果，并对最终的概率进行简单的做平均融合，结果有 2.7‰的提升。