**用户常驻地分析项目**

**《用户常驻地分析项目文档》**

**一、运行环境**

**Python 2.7.10**

**scikit-learn 0.16.1**

**二、数据来源**

JOB 124527 客户手机定位清洗数据

获取一个月内（除去上传日期为法定节假日的数据）的客户手机定位信息，包括clientId （客户手机识别id），latitude（维度），longitude（经度），uploadtime （上传时间） ，作为数据集导入python代码中。

**三、数据清洗和处理**

将数据按照用户分组，针对每个用户的数据进行清洗，主要有三个步骤：

1. 去除单日数据上传次数异常的日期， 即当天上传次数**大于**一个月内平均每日上传次数**三个标准差**的日期。

2. 对同一天中的坐标异常值的点进行处理，去除离当天平均坐标距离**超过两个标准差**的点

3. 去除当日平均坐标点位置与一个月内平均坐标距离**超过两个标准差**的日期的所有数据（用于排除，比如节假日离开常驻地旅游的一些特殊情况）

添加字段，为了区分工作地点和家，增添**是否工作**字段。如果上传时间在:

1. 7am – 9am, is\_work = 0.5
2. 9am – 18pm, is\_work = 1
3. 18pm – 20pm, is\_work = 0.5
4. 其他时间，is\_work = 0

**四、算法实现**

应用DBSCAN + Kmeans的算法对用户常驻地做出预测：

1. 首先，利用DBSCAN算法求出单个用户的n个坐标地簇：
   * 参数设置：
     1. eps = 0.5km / kms\_per\_radian (6371.0088，地球半径)
     2. min\_samples = 2
     3. algorithm='ball\_tree'
     4. metric='haversine' （空间计算距离设置）
2. 选取包含相同标签，即包含坐标点数最多的前两大簇
3. 利用kmeans算法，分别计算每个簇的中心点（坐标），作为前两大常驻地的预测坐标。
   * 参数设置：
     1. n\_clusters=2
     2. random\_state=42
4. 利用是否工作字段计算每个中心点的数值，数值越大，说明该中心点表示为工作地的可能性更高（一般情况小于0.5）；数值越小，说明该中心点为家庭地址所在地的可能性越高（一般情况大于0.5） 。
5. 根据所求得的中心经纬度信息，利用地图API进行逆地理编码，获取该经纬度对应的具体位置、道路名称和城市等信息。

**五、其他说明**

1. 有很多用户由于坐标点记录值过少，利用上述方法可能只能得到一个簇，因此有部分用户没有两个常驻地（办公地址和家庭地址），最后只记录第一簇的中心点。根据**是否工作**对该点上下班时间段进行划分，并标记家庭或者办公地址。
2. 对于样本较少的情况可以根据经纬度和是否上班三个字段绘制三维散点图以供可视化观察，如下（目前已被注释，需要时可开启）：

