****

**2021—2022学年第2学期**

**《机器学习》课程报告**

题目：基于DBSCAN聚类算法的疫情分布分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **学号** | **专业班级** | **姓名** | **分工占比**  **(共100%)** | **得分** |
| 1904020211 | 安全1902 | 刘衍 | 25% |  |
| 1904020204 | 安全1902 | 江柯标 | 25% |  |
| 1904020206 | 安全1902 | 李春龙 | 25% |  |
| 1801040307 | 勘查1803 | 赵灵杰 | 25% |  |

2022年 6 月 1 日

|  |
| --- |
| **题目 基于DBSCAN聚类算法的疫情分布分析** |
| 一、研究意义 |
| 疫情肆虐的背景下，国家的经济，社会的安定，人民的生活受到了极大的打击。尽管我国对于疫情反应迅速，管控有力，但国外感染人数却持续增长，因此我国也仍需对疫情继续进行有效的管控。由于疫情封锁区域政策，虽然使得疫情得到有效的控制，但也给人们增添了生活上的不便以及对不知感染人群位置的恐慌，疫情的持续发酵，牵动着用户的情绪，在这样的情况下，用户会恐惧，感染的人越来越多，是否能够得到控制。  本次模型是基于DBSCAN聚类算法的疫情分布，和地图相结合，以此为依据实现聚类分析，可帮助政府更加精确地实现疫情防控，更加准确地进行封锁政策，加快抗议进程，对结束疫情有着积极作用。 |
| 二、数据描述 |
| 说明：本部分主要描述数据获取途径、数据内容组成、数据预处理过程等。  本模型数据获取途径为：   1. 上海本地宝--记录每日上海地区阳性新增患者的居住地，详尽至街道，小区，并以上海市区级区域划分。 2. 百度坐标拾取系统--通过Selenium模拟浏览器行为，Threading模块   多线程进行爬取，以每个患者所在的街道和小区地址作为搜索值，获取所对应的最佳匹配项的经纬数据。并以Excel格式输出。  数据内容组成。如表1所示。   |  |  |  | | --- | --- | --- | | **地区名** | **经度** | **纬度** | | 永年路126号 | 121.485615 | 31.218081 | | 丽园路43弄 | 121.495629 | 31.21684 | | 永年路128号 | 121.485578 | 31.218068 |     **表1**  数据预处理过程：  将生成的excel表格导入，并拟合所做模型。表格部分如图1所示。  C(OV4JTZT35`N61GOCLJ1$1  图 1 |
| 三、模型描述 |
| 说明：本部分主要描述设计的数据处理流程，以及用到的算法、伪代码、流程图等信息。  本模型数据依据上海本地宝所提供的上海区阳性患者所居住点名称，将其对照爬虫下来这些位置的经度和维度一一对应，再将他们进行可视化，在地图上标记出，由此删除一些异常的点，例如定位经度维度后在河里的点，找出并修正，所得出数据以字典存储，数据格式是{"addr\_name":[经度值,纬度值]}，导入excel表记录下来。  本模型所采用的算法为DBSCAN聚类算法，是一种基于密度的空间聚类算法，是基于python语言实现的。DBSCAN聚类算法的原理简单来说是将空间内具有足够密度数据的区域依据密度进行划分成k个不同的簇。若空间中有任意两点p，q，若p属于簇C，且p密度可达q，则点q也属于簇C。该算法具有聚类速度快、能有效处理噪声点、发现空间中任意形状簇等优点，非常适合于本次问题的解决。  伪代码部分：  算法：DBSCAN  输入：n个经纬度坐标（x,y）  Eps:半径参数  Min\_samples:领域密度阈值  输出：基于密度的集合  方法：   1. 将读入的经纬度向量转化为实际距离矩阵，代替欧氏距离。 2. 标记所有对象为未访问变量对象p 3. Do 4. 随机选择一个未访问变量对象p 5. 标记p为访问 6. If p的eps领域内至少有min\_samples个对象： 7. 创建一个新簇C，并把p添加到C; 8. 令N为p的eps领域中的对象集合 9. For N中每个点p 10. If p未访问： 11. 标记p为已访问; 12. If p的eps领域至少有min\_samples个对象，把这些对象添 加到N; 13. 如果p还不是任何簇的成员，把p添加到C; 14. End for 15. 输出C； 16. Else 标记p为噪声 17. Until访问完所有对象。   流程图。如图1所示。    图 2 |
| 算法实现 |
| 1.首先需要将经纬度坐标转化为距离矩阵  distance\_matrix = squareform(pdist(df, (lambda u, v: haversine(u, v))))  #haversine()函数实现转化的算法，最后得到所有对象两两距离的n\*n矩阵  2.对每一个对象需要判断半径为eps的范围内，对象是否超过min\_samples：  neighbors\_model = NearestNeighbors(radius=self.eps)  #判断半径eps内对象个数 neighbors\_model.fit(X)   1. 判断一个点是否是噪声点，需要累加半径内点的个数做判断   if sample\_weight is None:  n\_neighbors = np.array([len(neighbors) for neighbors in neighborhoods])  else:  n\_neighbors = np.array([np.sum(sample\_weight[neighbors])  for neighbors in neighborhoods])  4．判断：  core\_samples = np.asarray(n\_neighbors >= self.min\_samples,  dtype=np.uint8) #如果中心点在eps范围内的节点数大于min\_samples  dbscan\_inner(core\_samples, neighborhoods, labels) #则将该点标记为中心点，并继续判断临近点   1. 相近的中心点归于同一簇：   self.core\_sample\_indices\_ = np.where(core\_samples)[0] #将两点归于同一簇中  4.根据得到的距离关系为每一个对象分类，并标签，-1表示为噪声点。  labels = np.full(X.shape[0], -1, dtype=np.intp)#为对象标记   1. 可视化   使用高的地图自带的包folium  folium.Map()，绘制以location为中心点的，地图缩放指数为12  folium.CircleMarker()，一次绘制每一个点，并以不同颜色标注，其中黑色为噪声点。 |
| 五、运行结果及意义说明 |
| 图 1静安区疫情聚类分析    图 2浦东区疫情聚类分析    图 3黄浦区疫情聚类分析    图 4长宁区疫情聚类分析    图 5徐汇区疫情聚类分析  图示结果为上海市各区的疫情聚类分析，各区政府可依据此结果进行更为精确且合理的封锁区域，使得疫情得到有效控制，同时又能保证不“一刀切”全境封锁的怠政。 |