

算法课设

姓名：黄思怡 学号：919102610102

题目1 解数独

- [一、问题描述](#)
- [二、算法思路](#)
- [三、算法设计](#)
 - [1、步骤](#)
 - [2、伪码](#)
- [四、算法实现细节](#)
- [五、运行结果](#)
- [六、问题探讨](#)
 - [1、复杂度](#)
 - [2、不可解情况](#)
 - [3、多解情况](#)

题目2 — louvain算法实现

- [一、问题描述](#)
- [二、算法思路](#)
- [三、算法设计](#)
 - [1、步骤](#)
 - [2、伪码](#)
- [四、算法实现细节](#)
 - [1、Graph生成](#)
 - [2、每个社区的信息](#)
 - [3、first_stage的细节](#)
 - [4、second_stage的细节](#)
 - [5、计算Accuracy](#)
- [五、运行结果](#)
 - [1、代码聚类结果（未添加标签）](#)
 - [2、标准数据集：标签 — 集合](#)
 - [3、Accuracy](#)
 - [4、运行时间](#)
- [六、问题探讨](#)

题目1 解数独

一、问题描述

给定一个 9X9 的网格，玩家需要根据盘面上已有的数字，推理出所有剩余空格的数字，使得网格的每一行、每一列、每一个小的 3X3 子网格中的9个位置都被1-9 数字填充，不可重复。

• 输入

```
Input:
grid = { {3, 0, 6, 5, 0, 8, 4, 0, 0},
         {5, 2, 0, 0, 0, 0, 0, 0, 0},
         {0, 8, 7, 0, 0, 0, 0, 3, 1},
         {0, 0, 3, 0, 1, 0, 0, 8, 0},
         {9, 0, 0, 8, 6, 3, 0, 0, 5},
         {0, 5, 0, 0, 9, 0, 6, 0, 0},
         {1, 3, 0, 0, 0, 0, 2, 5, 0},
         {0, 0, 0, 0, 0, 0, 0, 7, 4},
         {0, 0, 5, 2, 0, 6, 3, 0, 0} }
```

• 输出

```
Output:
3 1 6 5 7 8 4 9 2
5 2 9 1 3 4 7 6 8
4 8 7 6 2 9 5 3 1
2 6 3 4 1 5 9 8 7
9 7 4 8 6 3 1 2 5
8 5 1 7 9 2 6 4 3
1 3 8 9 4 7 2 5 6
6 9 2 3 5 1 8 7 4
7 4 5 2 8 6 3 1 9
```

二、算法思路

1. 采用**回溯法**，找到第一个标记为'0'的待填充位置做为起始点 (get_start_pos)。
2. 计算该位置的剩余可填充数字 (leftNums)。由于数独的规则是每行、每列、每个小3X3网格都不能重复1-9数字，那么该位置的剩余可填充数字便是排除了1-9中每行、每列、每个小3X3网格后的剩余数字。
3. 在该待填充位置填入剩余可填充数字(leftNums)中的一个后，找到下一个标记为'0'的待填充位置(get_next_pos)，重复步骤2。
4. 当最后一个带填充位置被成功填充了后，即返回True，可进行“回溯”退出。
5. 如果更改当前位置后，发现下一个待填充位置没有数字能填了，则说明之前填的数字有误，“回溯”回去，并且若当前位置已更改，则需重新设置为'0'。

三、算法设计

1、步骤

1. 找到第一个待填充的位置；
2. 计算该位置剩余可填充数字；
3. 填写剩余可填充数字中的一个；
4. 找到下一个待填充的位置；

5. 如果没有下一个待填充的位置，即全部遍历结束，返回True，进行回溯；
6. 如果下一个位置待填充位置没有数字能填了，则把那个位置重置为'0'，进行回溯。
7. 否则继续递归，进行下一个位置的数字的填充，重复上述操作2-6。

2、伪码

```
input: A - Sudoku 2D Array
output: solved 2D Array

function SolveSudoku(A)
    newA <- A
    x, y <- GetStartPos(A)
    GetLeftNums(x, y)
    GetNextPos(x, y)
    function dfs(x, y)
        leftNums <- GetLeftNums(x, y)
        for n in leftNums do
            newA[x][y] <- n
            x_next, y_next <- GetNextPos(x, y)
            if y_next == -1 then
                return True
            else
                end <- dfs(x_next, y_next)
                if end then
                    return True
                newA[x_next][y_next] <- '0'
            end
        end
    end
    return newA
end
```

四、算法实现细节

语言：python

环境：numpy

关键步骤实现细节：

1. 计算剩余可填充数字, get_leftNums(i, j), 遍历与该位置相关的行空间、列空间、小3X3空间，记录已填充数字，然后用1-9数字去减，做**集合的差集**。
2. 剩下的就是按照逻辑正常写，相当于实现一个深度优先搜索。

五、运行结果

输入：

```
board = [
    ['3', '0', '6', '5', '0', '8', '4', '0', '0'],
    ['5', '2', '0', '0', '0', '0', '0', '0', '0'],
    ['0', '8', '7', '0', '0', '0', '0', '3', '1'],
    ['0', '0', '3', '0', '1', '0', '0', '8', '0'],
    ['9', '0', '0', '8', '6', '3', '0', '0', '5'],
    ['0', '5', '0', '0', '9', '0', '6', '0', '0'],
    ['1', '3', '0', '0', '0', '0', '2', '5', '0'],
    ['0', '0', '0', '0', '0', '0', '0', '7', '4'],
    ['0', '0', '5', '2', '0', '6', '3', '0', '0']
]
```

输出：

```
['3', '1', '6', '5', '7', '8', '4', '9', '2']
['5', '2', '9', '1', '3', '4', '7', '6', '8']
['4', '8', '7', '6', '2', '9', '5', '3', '1']
['2', '6', '3', '4', '1', '5', '9', '8', '7']
['9', '7', '4', '8', '6', '3', '1', '2', '5']
['8', '5', '1', '7', '9', '2', '6', '4', '3']
['1', '3', '8', '9', '4', '7', '2', '5', '6']
['6', '9', '2', '3', '5', '1', '8', '7', '4']
['7', '4', '5', '2', '8', '6', '3', '1', '9']
```

六、问题探讨

1、复杂度

由于该题以回溯法递归为基础，算法运行的时间很大程度上决定于输入网格的质量，如果输入网格有解，则很快就能解出；若无解，则每个空位都会填充1-9任意数字进行尝试。很难找到一个非常精确的渐进紧界，因此，这里给出一个较为宽松的渐进复杂度上界 $O(9^{9 \times 9})$ ，即最多有9X9个空白格，每个格子可以填1-9中的任意整数。

2、不可解情况

当数独所有空白格全部遍历后仍然未返回True，回溯退出，则为不可解情况。可以在代码后面增加一个判断条件，当出现这种情况时，print('unsolved')。

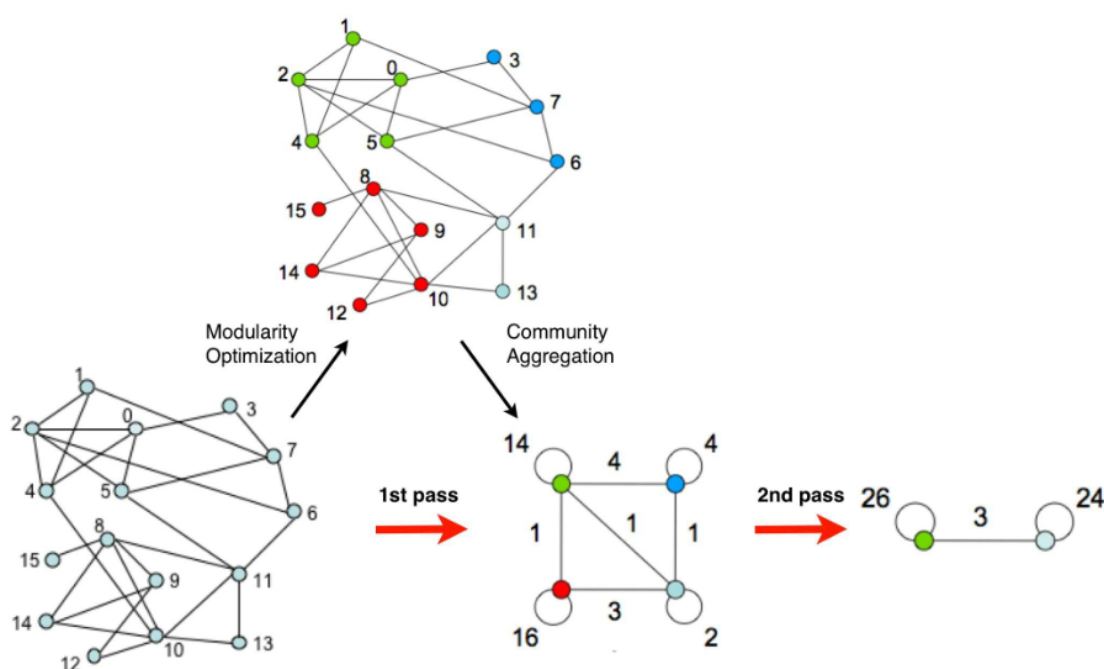
3、多解情况

对于多解情况，则需要每次解出一种情况后，就把这一种保留记录下来，同时按照错误情况处理机制，进行回溯与回填，更换尝试其它方法。如果又找到另一种解法，则继续保留下来，重复上述操作。直到遍历结束。

题目2 — louvain算法实现

一、问题描述

编程实现louvain算法：对SNAP-Email-Eu-core数据集进行聚类，然后输出划分结果、Modularity取值、Accuracy（有标签情况）。



二、算法思路

把数据集输入转化成成一个图，通过louvain算法对该图进行聚类，输出一张新的图。

louvain算法分为两个阶段。

第一阶段，确定节点的新社区。初始，每个节点各自为一个社区。循环遍历每个节点，计算把该节点划入其邻居节点的社区后，所带来的Modularity的增益，选择增益最大且为增益为正的社区，该节点就加入这个社区。循环反复遍历每个节点，直到所有节点的新社区都确定好了，且不再更改后，退出循环。

第二阶段，图压缩。遍历每个节点，把节点加入对应的新社区，同时把节点与社区内部的连接划分给新社区的自权重，把节点与社区外部的连接划分给新社区与外部的连接。

完成第二阶段后，又进入第一阶段，直到所有节点不在更改社区时，退出循环，输出新图。

经过简化计算后，把节点*i*移动到社区*C*造成的modularity的增益可以被简单地用以下公式进行计算：

$$\begin{aligned}\Delta Q &= \left[\frac{\Sigma_{in} + k_{i,in}}{2m} - \left(\frac{\Sigma_{tot} + k_i}{2m} \right)^2 \right] - \left[\frac{\Sigma_{in}}{2m} - \left(\frac{\Sigma_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right] \\ &= \frac{1}{2m} \left[k_{i,in} - \frac{\Sigma_{tot} k_i}{m} \right]\end{aligned}$$

其中：

Σ_{tot} — the sum of the weights of the links incident to nodes in *C*;

k_i — the sum of the weights of the links incident to node *i*;

$k_{i,in}$ — the sum of the weights of the links from *i* to nodes in *C*;

m — the sum of the weights of all the links in the network.

三、算法设计

1、步骤

1. 导入数据集，生成图；
2. 第一阶段: 确定每个节点的新社区
 - 2.1. 循环遍历每个节点
 - 2.2. 遍历邻居节点，计算该节点加入邻居节点的所属社区后的 ΔQ ；
 - 2.3 确定该节点的新社区，使 ΔQ 最大的，且为正的，那个邻居所属的新社区即为该节点的新社区；
 - 2.4. 每个节点可能被遍历到不止一次，直到每个节点所属社区不再改变后，停止并退出循环。
3. 第二阶段：图压缩
 - 3.1. 创建空图；
 - 3.2. 遍历已知图，把每个节点*i*的社区加入空图作为新的社区图节点*c*；
 - 3.3. 节点*i*与社区*c*的连接，转为社区*c*的自权重；

- 3.4. 节点*i*与空图其它节点，即其它社区中心的连接即为新的社区图节点*c*与其它社区中心的连接。
4. 重复步骤2、3，直到步骤2中的社区无改变，保存最后一次图，退出算法。
5. 打印最后的图，计算Accuracy。

2、伪码

```
input: Graph
output: newGraph

function louvain(Graph)
    delta_Q(i, c) //node i, community c

    //phase1
    Loop first_stage
        for i in Graph do
            if  $i.q < \delta_Q(i, c)$  and  $0 < \delta_Q(i, c)$  then
                 $i.c \leftarrow c$ 
        if (have not any i.c changed) then
            exit Loop first_stage

    //phase2
    function second_stage()
        for i in Graph do
             $newGraph[i] \leftarrow i.c$ 
             $newGraph[i].subs \leftarrow i.subs \mid newGraph[i].subs$ 

            for sub in  $newGraph[i].subs$  do
                (update all the new links)
    end function

    Loop excute
        Loop first_stage
        second_stage()
    if didn't find ant changes in first_stage do
        exit Loop excute
    end function
```

四、算法实现细节

语言：python

环境：python标准库，无另外引入库

1、Graph生成

数据集txt每一行的文本中为，节点*i* — 节点*j*，表示节点*i*发送了至少一封邮件给节点*j*，也就是说，该原数据集本身是不含连接权重的。

这就产生了一个选择：在重构图的时候，我们图的连接是恒为1，还是说要包含权重（连接的重复度）。两种方式我都尝试了后，发现其实聚类结果差距不大。所以，还是在重构图的时候，保留权重。

图的结构为：

```
Graph: {node1: {neighbor1: weight, neighbor2: weight, ...}, node2...}
```

2、每个社区的信息

每个社区包含了以下三个内容：

```
class Nodes():
    def __init__(self, subs, inw, next_c):
        self.subs = subs # 节点i的子节点
        self.inw = inw # 节点的自环权重
        self.next_c = next_c # 在firststage中记录最近的neighbor
```

3、first_stage的细节

在第一阶段，特别需要明确的是：找到每个节点最终的所属社区，最终的！所以，不是仅仅遍历一遍节点就能完成的，需要多来几遍，直到每一个节点的所属社区不变就退出。

每次更新，可能都会让该节点当前最大的 ΔQ 改变，所以，使用delta_Qs这一个列表来存储每一个节点的delta_Q。该列表的index是遍历的时候确定的遍历序号loc，该列表的value是这个节点当前最大的 ΔQ 。

```
delta_Qs = [0] * len(self.Graph)
```

使用两个变量来记录第一阶段的状态：

- Changed : while的退出条件。True — while不退出，表示存在节点的最大 ΔQ 改变了；False — while退出，表示没有任何信息更新。
- Res : first_stage() 的返回值，表示是否有节点的社区改变。True — 表示有节点的社区改变，继续进行louvain循环；False — 表示第一阶段没有发现社区结果的改变，退出louvain算法循环。

4、second_stage的细节

在这里有两种思路：

1. 删去当前图中，成为子节点的节点，并更新相关社区连接。这是我尝试的第一种思路，会有一些前后删除顺序造成的错误，而且很难调整。字典，特别是二重字典，不要轻易进行删除、更新，容易出现错误。
2. 记录父节点和其子节点，根据这些记录重新生成图。事实证明，这个实现更加容易，不易出错。

最后，运行成功的为第二种思路。所以，第二阶段的任务，是根据父节点和子节点，重新生成一张图。

需要注意的是，子节点与其它社区子节点之间的连接，需要找到其他社区子节点的父节点后，更新该子节点的父节点与其他社区子节点的父节点之间的连接。要考虑重复判断。

需要双向调整，双向加上连接。

5、计算Accuracy

因为数据是带标签的，所以要把数据中心的标签，为当前社区中子节点数的标签最多的那个标签(res_l)，accuracy就是分类正确的数 / 总的节点数。

但是这里有一个问题，有可能最后确定的出现次数最多的label会有重复，此时如何确定哪一个成为其子集呢？我的选择是，加入。

但这样也是有问题的，所以我不太清楚如何准确计算accuracy。

```
res_l = max(labels, key=labels.get)
if res_l in labeled:
    labeled[res_l] += (dataset[n])
else:
    labeled[res_l] = dataset[n]
```

五、运行结果

```
481
187
89
56
42
41
41
社区个数 1 聚类中心 998 ---- {'869', '817', '849', '382', '644', '712', '782', '533', '593', '189', '147', '457', '434', '52', '348', '349', '352', '656', '943', '835', '129', '398', '909', '425', '291', '694', '595', '936', '280', '557', '177', '3
21', '563', '839', '479', '749', '148', '191', '734', '6', '692', '280', '789', '122', '290', '526', '719', '262', '988', '426', '785', '998', '680', '481', '435', '368', '608', '393', '377', '542', '168', '781', '714', '621', '969', '292', '93',
511', '200', '695', '65', '677', '41', '820', '707', '513', '358', '659', '181', '992', '284', '917', '689', '214', '568', '207', '197', '939', '257', '178', '283', '392', '95', '866', '999', '961', '776', '201', '801', '172', '983', '851', '219',
779', '906', '284', '573', '457', '588', '493', '537', '567', '824', '578', '525', '231', '345', '634', '978', '583', '170', '948', '387', '543', '589', '523', '285', '923', '843', '918', '389', '534', '820', '525', '684', '585', '149', '807',
', '577', '984', '342', '458', '346', '574', '668', '275', '515', '974', '276', '786', '156', '536', '256', '365', '522', '465', '906', '413', '391', '482', '861', '328', '841', '543', '942', '790', '871', '544', '681', '167', '682', '629', '579',
', '277', '483', '128', '448', '182', '194', '751', '310', '873', '905', '824', '562', '136', '528', '510', '1000', '312', '199', '572', '483', '813', '959', '610', '472', '396', '14', '289', '295', '664', '484', '130', '458', '94', '848', '955', '882
', '596', '881', '344', '575', '793', '723', '689'}
社区个数 2 聚类中心 674 ---- {'266', '951', '466', '246', '487', '967', '583', '267', '674', '525', '438', '141', '688', '913', '985', '293', '885', '661', '823', '728', '924', '827', '519', '662', '451', '496', '865', '374', '957', '441', '786', '6
560', '358', '729', '912', '499', '45', '338', '558', '7', '836', '264', '581', '422', '565', '359', '584', '312', '685', '922', '582', '569', '838', '578', '488', '247', '518', '765', '786', '61', '368', '856', '465', '324', '972', '44', '555', '6
38', '11', '486', '998', '161', '573', '728', '833', '211', '680', '586', '893', '407', '12', '589', '984', '699', '563', '569', '729', '729', '488', '699', '11'}
社区个数 3 聚类中心 935 ---- {'763', '388', '58', '712', '743', '902', '877', '195', '241', '412', '619', '1004', '314', '488', '796', '899', '98', '884', '622', '926', '892', '159', '369', '233', '281', '6', '637', '244', '385', '888', '685', '993'
', '888', '1801', '131', '132', '4', '832', '863', '347', '58', '225', '63', '319', '597', '991', '886', '647', '371', '921', '57', '635', '631', '812', '927', '221', '727', '587', '182', '648', '617', '818', '748', '718', '284', '531', '682', '168',
', '980', '810', '283', '986', '838', '931', '981', '838', '55', '778', '710', '750', '683', '625', '248', '618', '688', '887', '538', '21', '253', '659', '718', '481', '788', '54', '528', '589', '738', '5', '298', '789', '930', '994', '734', '1
92', '517', '383', '344', '762', '768', '891', '289', '634'}
社区个数 4 聚类中心 463 ---- {'636', '463', '254', '883', '999', '193', '825', '985', '183', '681', '137', '884', '1081', '981', '845', '774', '726', '832', '416', '665', '174', '68', '854', '184', '384', '998', '382', '886', '211', '516', '3', '8
88', '285', '126', '552', '428', '886', '138', '653', '949', '888', '571', '791', '991', '826', '980', '864', '327', '781', '889', '624', '647', '738', '755', '815', '288', '982', '286', '762', '536', '462', '921', '89', '561', '958', '635'}
社区个数 5 聚类中心 975 ---- {'780', '565', '951', '844', '973', '788', '674', '416', '672', '971', '957', '9', '8', '975', '729', '893', '889', '884', '265', '213', '19', '778', '787', '799'}
社区个数 6 聚类中心 991 ---- {'78', '511', '963', '862', '827', '581', '591', '789', '888', '977', '227', '18', '727', '83', '568', '106', '12', '251', '584', '641', '588', '217', '725', '766', '811', '535', '777', '484', '639', '169', '364', '373'
', '186', '319', '497', '482', '433', '499', '239', '583', '71', '268', '483', '738', '77', '152', '134', '144', '372', '78', '431', '288', '79', '559', '638', '145', '432', '934', '348', '679', '521', '87', '872', '539', '158', '876', '464', '419'
', '637', '212', '553', '445', '554', '867', '242', '363'}
社区个数 7 聚类中心 895 ---- {'888', '270', '727', '123', '655', '897', '489', '844', '261', '26', '878', '816', '333', '436', '883', '278', '38', '547', '722', '993', '478', '361', '444', '914', '28', '228', '36', '251', '336', '455', '725', '683'
', '980', '733', '133', '151', '611', '981', '437', '373', '724', '119', '590', '27', '784', '414', '263', '33', '114', '47', '485', '689', '338', '983', '868', '842', '229', '30', '116', '23', '548', '686', '881', '980', '378', '411', '39', '37'
', '895', '297', '538', '567', '339', '96', '245', '178', '895', '915', '864', '447', '986', '82', '909', '864', '22', '822', '443', '318', '545', '442'}
社区个数 8 聚类中心 789 ---- {'340', '819', '784', '164', '125', '461', '335', '322', '97', '139', '428', '354', '676', '46', '617', '541', '768', '837', '933', '331', '657', '99', '456', '648', '185', '45', '448', '353', '15', '735', '124', '269'
', '323', '446', '422', '935', '529', '688', '273', '188', '764', '222', '687', '789', '838', '592', '770', '584', '216', '447', '64', '484', '768', '181'}
社区个数 9 聚类中心 889 ---- {'112', '889', '568', '722', '474', '16', '108', '1080', '91', '771', '166', '229', '831', '185', '477', '251', '109', '884', '118', '488', '111', '944', '892', '476', '110', '478', '329', '1893', '258'}
社区个数 10 聚类中心 771 ---- {'188', '317', '596', '389', '67', '524', '227', '669', '228', '17', '686', '189', '187', '163', '821', '311', '678', '747', '313', '69', '941', '771', '745', '492', '641', '673', '318', '905', '74', '946', '146', '274'
', '68', '582', '253', '480', '81', '488', '218', '221', '713', '686', '315', '589', '467', '473', '597', '663', '519', '655', '782', '969'}
社区个数 11 聚类中心 816 ---- {'851', '224', '816', '911', '225', '605', '341', '297', '389', '686', '222', '918', '65', '313', '33', '73', '612', '226', '223', '658', '18', '338', '985', '189', '366', '248', '852', '791', '387', '415', '645', '449', '69
4', '846', '569', '623', '778', '215', '488'}
社区个数 12 聚类中心 837 ---- {'287', '866', '736', '779', '588', '678', '756', '187', '558', '941', '92', '679', '834', '787', '549', '837', '667', '418', '186', '146', '582', '186', '589', '180', '887', '627', '120', '614', '885', '671', '638', '615'
', '480', '487', '162', '969'}
社区个数 13 聚类中心 582 ---- {'723', '549', '866', '386', '325', '825', '435', '551', '668', '524', '669', '958', '738', '718', '679', '117', '260', '127', '468', '962', '945', '230', '72', '259', '783', '589', '387', '968', '298', '733', '
923', '799', '614', '618', '885', '742', '508', '715', '888', '538', '938', '299', '952', '728', '42', '613', '168', '969'}
社区个数 14 聚类中心 808 ---- {'184', '62', '788', '994', '645', '512', '724', '888', '175', '48', '366', '40', '532', '693', '162', '853', '59', '121', '432', '155', '283', '792', '433'}
社区个数 15 聚类中心 870 ---- {'741', '998', '869', '394', '947', '150', '568', '376', '388', '932', '385', '179', '183', '398', '395', '389', '296', '184', '632', '943', '948', '343', '386', '628', '491', '985', '397', '538', '258', '396', '717', '878'
', '578', '68', '592', '576', '388', '788', '719', '384', '381', '702'}
社区个数 16 聚类中心 576 ---- {'188', '756', '889', '173', '514', '480', '758', '486', '328', '418', '485', '416', '831', '652', '477', '189', '357', '128', '773', '892', '375', '478', '475', '828', '98', '454', '469', '388', '143'}
社区个数 17 聚类中心 742 ---- {'518', '154', '932', '546', '898', '524', '771', '882', '168', '695', '142', '954', '783', '249', '642', '472', '654', '687', '742', '424', '166'}
社区个数 18 聚类中心 598 ---- {'598', '938', '313', '611', '782', '623', '686'}
社区个数 19 聚类中心 678 ---- {'771', '628', '788', '913', '686', '381', '748', '334', '944', '495', '857', '428', '668', '579', '738', '676', '686'}
社区个数 20 聚类中心 880 ---- {'538', '326', '825', '944', '923'}
社区个数 21 聚类中心 875 ---- {'418', '883', '745', '471', '624', '453', '472', '767', '875', '979', '488', '855', '491', '355'}
社区个数 22 聚类中心 989 ---- {'388'}
社区个数 23 聚类中心 633 ---- {'633'}
社区个数 24 聚类中心 648 ---- {'648'}
社区个数 25 聚类中心 653 ---- {'653'}
社区个数 26 聚类中心 658 ---- {'658'}
社区个数 27 聚类中心 680 ---- {'680'}
社区个数 28 聚类中心 678 ---- {'678'}
社区个数 29 聚类中心 675 ---- {'675'}
社区个数 30 聚类中心 684 ---- {'684'}
```

1、代码聚类结果（未添加标签）

社区个数 1 聚类中心 995 ----- {'829', '515', '567', '368', '692', '275', '820', '868', '60', '483', '425', '659', '939', '536', '440', '203', '65', '936', '544', '199', '906', '345', '712', '522', '841', '680', '128', '965', '491', '206', '956', '352', '343', '846', '605', '122', '719', '959', '734', '378', '695', '232', '180', '177', '93', '585', '513', '801', '575', '450', '434', '14', '401', '207', '793', '392', '779', '942', '284', '292', '741', '970', '874', '604', '414', '781', '824', '681', '557', '682', '526', '276', '694', '219', '426', '688', '402', '53', '985', '542', '621', '316', '790', '129', '391', '960', '0', '562', '205', '277', '840', '320', '312', '202', '689', '157', '992', '843', '130', '344', '603', '289', '256', '916', '457', '903', '761', '201', '149', '849', '938', '182', '974', '321', '697', '280', '572', '533', '350', '403', '396', '346', '342', '295', '543', '917', '995', '955', '909', '629', '782', '156', '168', '484', '873', '839', '204', '176', '588', '257', '908', '290', '677', '948', '399', '611', '398', '749', '813', '870', '196', '198', '918', '871', '523', '776', '511', '393', '940', '191', '705', '656', '262', '148', '947', '534', '714', '644', '797', '664', '910', '922', '537', '376', '904', '751', '449', '181', '41', '465', '528', '869', '563', '835', '231', '943', '479', '349', '383', '817', '380', '600', '178', '382', '387', '197', '907', '458', '861', '593', '706', '568', '377', '413', '789', '95', '961', '94', '935', '200', '379', '668', '1000', '167', '851', '574', '104', '348', '214', '998', '610', '291', '645', '493'}

社区个数 2 聚类中心 674 ----- {'505', '430', '566', '500', '452', '661', '616', '754', '451', '649', '264', '501', '465', '525', '570', '913', '530', '247', '1', '266', '324', '729', '565', '804', '487', '608', '830', '498', '499', '503', '7', '12', '267', '833', '44', '856', '805', '558', '441', '706', '823', '332', '246', '912', '555', '786', '506', '161', '967', '765', '957', '358', '972', '504', '407', '573', '510', '720', '996', '11', '602', '569', '836', '975', '519', '293', '827', '43', '141', '674'}

社区个数 3 聚类中心 463 ----- {'412', '462', '546', '888', '884', '701', '619', '877', '2', '255', '51', '55', '520', '281', '561', '928', '906', '635', '712', '636', '59', '812', '1001', '991', '807', '982', '4', '435', '899', '994', '859', '647', '6', '625', '58', '993', '742', '1004', '344', '517', '850', '718', '252', '147', '132', '740', '54', '814', '814', '408', '56', '863', '57', '192', '195', '685', '347', '305', '63', '463'}

社区个数 4 聚类中心 319 ----- {'762', '183', '138', '924', '536', '137', '304', '571', '48

1', '716', '806', '803', '686', '240', '5', '774', '990', '930', '244', '243', '599', '901', '921', '880', '294', '194', '303', '472', '238', '698', '271', '285', '208', '886', '634', '865', '635', '646', '369', '927', '159', '699', '949', '302', '763', '517', '926', '845', '854', '193', '587', '102', '738', '810', '898', '131', '622', '552', '631', '743', '1001', '815', '88', '737', '210', '665', '314', '809', '630', '241', '857', '864', '420', '528', '832', '755', '234', '637', '126', '298', '931', '286', '683', '233', '516', '991', '301', '211', '706', '647', '209', '319', '601', '993', '826', '586', '750', '3', '89', '902', '507', '610', '174'}
 社区个数 5 聚类中心 503 ---- {'416', '362', '360', '9', '971', '672', '488', '973', '406', '502', '459', '729', '893', '804', '778', '565', '707', '466', '799', '844', '8', '503', '19', '666', '265', '374', '359', '805', '61', '421', '912', '213', '506', '674', '957', '529', '951', '740', '496', '759', '700', '922'}
 社区个数 6 聚类中心 966 ---- {'416', '733', '299', '72', '858', '473', '825', '613', '728', '710', '535', '614', '530', '509', '189', '952', '492', '259', '923', '651', '799', '885', '713', '306', '491', '663', '410', '624', '254', '10', '327', '127', '480', '862', '855', '187', '678', '540', '723', '524', '582', '742', '188', '400', '941', '68', '117', '715', '69', '747', '260', '950', '81', '467', '894', '887', '490', '538', '551', '818', '274', '968', '966'}
 社区个数 7 聚类中心 784 ---- {'953', '299', '278', '417', '784', '983', '745', '803', '229', '897', '85', '978', '114', '170', '556', '767', '472', '997', '414', '39', '318', '190', '855', '76', '545', '411', '527', '736', '47', '75', '925', '590', '13', '778', '330', '26', '822', '844', '588', '36', '988', '883', '116', '547', '442', '453', '355', '875', '860', '333', '113', '795', '423', '569', '725', '220', '987', '785', '900', '336', '864', '24', '409', '444', '721', '339', '28', '263', '722', '919', '979', '37', '753', '32', '337', '119', '136', '895', '23', '171', '471'}
 社区个数 8 聚类中心 709 ---- {'676', '662', '687', '98', '796', '768', '354', '335', '896', '541', '428', '322', '937', '618', '640', '447', '331', '735', '216', '97', '45', '371', '592', '164', '892', '764', '272', '323', '448', '617', '15', '708', '124', '584', '935', '794', '579', '933', '838', '46', '828', '819', '446', '269', '709', '101', '99', '770', '891', '760', '657'}
 社区个数 9 聚类中心 1002 ---- {'396', '489', '595', '781', '944', '309', '279', '308', '869', '905', '394', '696', '282', '315', '771', '330', '397', '943', '163', '311', '395', '522', '821', '74', '389', '384', '17', '745', '118', '866', '948', '385', '105', '719', '576', '702', '390', '16', '435', '920', '296', '150', '386', '524', '523', '381', '250', '52', '596', '230', '1002', '221', '179', '632', '253', '103', '560', '849', '628', '388', '932', '218', '597', '741'}
 社区个数 10 聚类中心 788 ---- {'424', '313', '623', '654', '606', '905', '225', '696', '315', '224', '317', '330', '783', '297', '298', '310', '73', '415', '591', '389', '607', '745', '788', '228', '758', '341', '18', '331', '202', '642', '42', '248', '695', '166', '215', '468', '142', '852', '223', '911', '882', '851', '932', '226', '222', '519', '227', '307', '650', '160', '472'}
 社区个数 11 聚类中心 837 ---- {'146', '549', '80', '945', '704', '509', '287', '92', '418', '969', '837', '86', '885', '550', '663', '184', '866', '589', '508', '620', '627', '958', '678', '679', '582', '20', '800', '667', '107', '787', '615', '21', '638', '834', '779', '887', '538', '325', '962', '671'}
 社区个数 12 聚类中心 997 ---- {'953', '881', '997', '914', '414', '370', '980', '75', '34', '757', '29', '123', '133', '373', '31', '915', '25', '27', '878', '22', '844', '591', '609', '588', '986', '437', '883', '842', '816', '976', '686', '151', '367', '115', '919', '251', '485', '438', '96', '724', '455', '40', '443', '38', '33', '847', '753', '916', '261', '655', '165', '964', '427', '436', '338', '135', '981', '30', '35', '119', '439', '470', '548', '245', '987', '361', '527'}
 社区个数 13 聚类中心 739 ---- {'416', '989', '984', '67', '146', '792', '366', '831', '162', '405', '121', '773', '477', '780', '514', '120', '643', '328', '283', '49', '410', '300', '82', '356', '758', '693', '486', '90', '173', '739', '155', '143', '478', '853', '432', '469', '652', '109', '84', '892', '48', '594', '375', '50', '752', '889', '828', '490', '475', '357', '175', '512', '454', '532'}
 社区个数 14 聚类中心 583 ---- {'553', '598', '583', '777', '811', '482', '419', '581', '62', '460', '730', '848', '559', '769', '554', '539', '862', '679', '445', '468', '87

```

2', '531', '340', '78', '494'}
社区个数 15 聚类中心 676 ---- {'690', '676', '270', '139', '768', '185', '422', '273', '1
00', '937', '611', '722', '353', '64', '456', '125', '404', '429', '935', '140', '57
9', '933', '838', '929', '819', '334', '709', '461', '770', '760'}
社区个数 16 聚类中心 477 ---- {'989', '984', '831', '111', '91', '477', '108', '1003', '4
76', '478', '109', '258', '66', '892', '110', '474', '112', '153', '329'}
社区个数 17 聚类中心 524 ---- {'365', '954', '71', '802', '172', '364', '524', '70', '24
9', '154', '577', '518', '431', '578'}
社区个数 18 聚类中心 909 ---- {'288', '433', '71', '623', '87', '643', '432', '791', '90
9', '626', '217', '612'}
社区个数 19 聚类中心 598 ---- {'79', '766', '598', '554', '867', '583', '268', '77', '49
7', '934', '134', '521', '811', '535', '848'}
社区个数 20 聚类中心 890 ---- {'372', '433', '726', '152', '825', '169', '483', '923', '8
90', '144', '186', '432', '791', '594', '83', '326', '363', '71', '464', '364', '538',
'351', '212', '727'}
社区个数 21 聚类中心 944 ---- {'1002', '330', '106', '560', '613', '944', '717', '756',
'775', '788', '614'}
社区个数 22 聚类中心 927 ---- {'876', '366', '239', '999', '373', '637', '783', '643', '1
45', '927', '564', '158', '930', '109', '319', '236', '800', '963', '237', '879', '66
9', '638', '242', '639', '235', '977'}
社区个数 23 聚类中心 673 ---- {'946', '641', '944', '495', '748', '606', '730', '673', '6
69'}
社区个数 24 聚类中心 580 ---- {'580'}
社区个数 25 聚类中心 633 ---- {'633'}
社区个数 26 聚类中心 648 ---- {'648'}
社区个数 27 聚类中心 653 ---- {'653'}
社区个数 28 聚类中心 658 ---- {'658'}
社区个数 29 聚类中心 660 ---- {'660'}
社区个数 30 聚类中心 670 ---- {'670'}
社区个数 31 聚类中心 675 ---- {'675'}
社区个数 32 聚类中心 684 ---- {'684'}
社区个数 33 聚类中心 691 ---- {'691'}
社区个数 34 聚类中心 703 ---- {'703'}
社区个数 35 聚类中心 711 ---- {'711'}
社区个数 36 聚类中心 731 ---- {'731'}
社区个数 37 聚类中心 732 ---- {'732'}
社区个数 38 聚类中心 744 ---- {'744'}
社区个数 39 聚类中心 746 ---- {'746'}
社区个数 40 聚类中心 772 ---- {'772'}
社区个数 41 聚类中心 798 ---- {'798'}
社区个数 42 聚类中心 808 ---- {'808'}

```

2、标准数据集：标签 — 集合

```

1 -- {'313', '414', '904', '449', '731', '308', '459', '905', '225', '219', '629', '22
4', '1', '315', '317', '696', '479', '120', '311', '297', '73', '74', '310', '415', '1
7', '743', '316', '228', '0', '798', '18', '341', '516', '605', '846', '312', '248',
'307', '177', '215', '764', '517', '85', '603', '775', '468', '916', '1002', '221',
'872', '852', '223', '675', '650', '797', '314', '226', '222', '220', '697', '734',
'218', '227', '645', '309'}
21 -- {'412', '762', '193', '587', '884', '2', '138', '606', '832', '755', '691', '63
4', '622', '520', '281', '865', '137', '928', '286', '635', '571', '636', '648', '59',
'1001', '803', '982', '4', '899', '994', '774', '815', '990', '625', '456', '58', '99
3', '826', '586', '718', '633', '711', '532', '408', '845', '3', '849', '882', '56',

```

```

'57', '670', '863', '192', '880', '901', '195', '921', '685', '194', '305', '63'}
25 -- {'489', '5', '6', '644', '64', '528'}
14 -- {'362', '430', '452', '500', '661', '893', '498', '7', '956', '833', '44', '73
9', '856', '555', '529', '573', '496', '141', '754', '649', '264', '249', '913', '24
7', '729', '804', '487', '830', '19', '267', '374', '359', '699', '332', '421', '912',
'951', '510', '11', '602', '608', '293', '827', '9', '505', '566', '451', '406', '50
1', '502', '570', '266', '324', '565', '707', '466', '8', '666', '265', '805', '558',
'967', '407', '996', '569', '658', '674', '922', '365', '360', '672', '488', '973',
'525', '530', '499', '503', '12', '441', '823', '246', '213', '161', '506', '765', '9
57', '358', '504', '720', '700', '998', '43'}
9 -- {'416', '733', '881', '72', '858', '777', '825', '945', '535', '92', '259', '92
3', '22', '885', '890', '306', '10', '327', '20', '524', '723', '400', '326', '260',
'21', '894', '887', '538', '975', '325', '703', '962'}
26 -- {'301', '651', '624', '420', '13', '677', '118', '409', '810'}
4 -- {'829', '515', '567', '623', '172', '183', '275', '802', '425', '440', '203', '6
5', '199', '544', '936', '522', '965', '206', '959', '232', '93', '450', '585', '801',
'14', '401', '207', '556', '292', '294', '727', '874', '526', '276', '694', '426', '68
8', '53', '542', '790', '129', '562', '840', '576', '202', '689', '992', '843', '256',
'457', '201', '938', '280', '572', '403', '543', '909', '133', '168', '822', '419', '4
84', '176', '844', '290', '399', '486', '611', '749', '813', '198', '620', '523', '94
0', '626', '850', '705', '655', '534', '714', '464', '664', '910', '270', '563', '83
5', '817', '600', '197', '732', '702', '458', '706', '568', '791', '413', '789', '95',
'961', '200', '584', '744', '1000', '167', '902', '574', '836', '291', '493'}
17 -- {'676', '854', '760', '796', '139', '273', '335', '660', '428', '322', '447', '7
35', '216', '45', '592', '164', '272', '429', '15', '617', '140', '579', '794', '933',
'46', '838', '819', '929', '446', '269', '334', '461', '274', '448', '657'}
34 -- {'424', '434', '16', '42', '87', '561', '763', '212', '460', '607', '787', '10
5', '684'}
11 -- {'953', '997', '34', '590', '29', '24', '39', '31', '721', '28', '915', '26', '2
5', '27', '878', '588', '36', '722', '40', '38', '545', '37', '33', '32', '964', '30',
'35', '23', '785'}
5 -- {'970', '196', '593', '861', '751', '204', '128', '917', '918', '205', '908', '77
6', '974', '820', '41', '955', '94', '51'}
10 -- {'47', '370', '75', '757', '925', '417', '704', '444', '263', '983', '976', '54
7', '151', '442', '367', '318', '919', '485', '229', '443', '48', '897', '333', '76',
'753', '847', '113', '114', '795', '261', '981', '423', '119', '725', '895', '245',
'171', '361', '527'}
36 -- {'372', '366', '121', '373', '282', '86', '283', '62', '49', '82', '693', '166',
'594', '601', '83', '107', '142', '752', '950', '932', '494', '160'}
37 -- {'50', '288', '433', '71', '152', '654', '643', '432', '364', '84', '70', '162',
'431', '217', '612'}
7 -- {'396', '376', '295', '368', '181', '868', '869', '60', '394', '397', '943', '38
3', '395', '841', '380', '389', '384', '985', '382', '391', '385', '387', '719', '39
8', '320', '390', '61', '150', '296', '378', '377', '853', '386', '381', '52', '250',
'393', '379', '653', '761', '632', '392', '103', '947', '628', '388', '182', '104',
'772', '321', '741'}
22 -- {'604', '738', '698', '728', '285', '208', '898', '55', '552', '126', '646', '48
1', '683', '986', '988', '866', '211', '949', '469', '455', '1004', '54', '867', '98
7', '809'}
8 -- {'989', '111', '477', '108', '514', '328', '356', '90', '143', '478', '258', '65
2', '66', '375', '110', '475', '112', '357', '329'}
15 -- {'690', '67', '146', '299', '549', '595', '473', '877', '710', '91', '999', '50
9', '287', '189', '778', '418', '969', '536', '780', '952', '298', '873', '799', '71
3', '550', '663', '184', '476', '746', '508', '190', '480', '627', '855', '187', '54
0', '582', '678', '958', '800', '188', '667', '68', '715', '972', '69', '747', '81',
'467', '474', '779', '911', '551', '818', '968'}
3 -- {'483', '531', '77', '539', '497', '581', '445', '557', '78', '811', '79', '848'}
29 -- {'80', '679', '966', '671', '716'}

```

```

20 -- {'806', '808', '740', '924', '750', '302', '96', '630', '89', '304', '369', '303', '88', '252'}
16 -- {'662', '687', '98', '768', '185', '354', '100', '580', '937', '618', '640', '842', '97', '353', '371', '125', '323', '404', '708', '935', '124', '709', '101', '99', '770'}
38 -- {'771', '896', '173', '638', '828', '102', '106', '300', '892', '405', '773', '889', '891'}
27 -- {'736', '769', '410', '355', '109', '153', '577', '454', '578', '559'}
13 -- {'914', '900', '336', '123', '169', '339', '330', '609', '437', '883', '116', '816', '115', '251', '438', '724', '860', '978', '170', '165', '337', '135', '338', '436', '136', '439'}
6 -- {'537', '616', '613', '619', '995', '255', '782', '659', '939', '837', '491', '745', '1003', '960', '948', '686', '127', '435', '642', '453', '647', '117', '834', '942', '490', '548', '610', '472'}
0 -- {'350', '342', '346', '781', '278', '857', '692', '681', '824', '682', '231', '156', '349', '402', '839', '345', '680', '712', '257', '621', '352', '178', '343', '122', '277', '907', '157', '180', '871', '130', '511', '344', '575', '289', '191', '656', '903', '179', '262', '668', '148', '814', '149', '851', '348', '214', '284', '351', '347'}
28 -- {'131', '132', '807', '784', '411', '864', '271', '737'}
2 -- {'553', '766', '598', '482', '554', '583', '340', '862', '134', '521'}
40 -- {'492', '144', '254', '186'}
35 -- {'954', '984', '615', '155', '546', '145', '934', '422', '533', '154', '518', '363', '513'}
23 -- {'944', '495', '279', '465', '641', '541', '783', '906', '591', '748', '730', '946', '589', '920', '979', '695', '786', '742', '596', '673', '669', '230', '147', '793', '519', '512', '597'}
19 -- {'241', '238', '239', '234', '637', '931', '631', '927', '159', '233', '564', '158', '240', '930', '209', '319', '236', '237', '963', '879', '244', '210', '243', '599', '242', '639', '926', '235', '977'}
24 -- {'163', '792', '821', '717', '788', '614'}
32 -- {'991', '886', '859', '888', '971', '665', '175', '812', '174'}
31 -- {'726', '253', '759', '560', '875', '507', '831', '471'}
39 -- {'756', '268', '331'}
12 -- {'470', '427', '980'}
30 -- {'876', '701', '462', '463'}
41 -- {'941', '758'}
18 -- {'767'}
33 -- {'870'}

```

3、Accuracy

```
0.4117117117117117
```

4、运行时间

```
运行时间： 2.125845193862915 s
```

六、问题探讨

打印结果后，发现单个的点相比于标准数据库来说更多，在反复检查并更改代码逻辑后，这种情况仍然存在。查阅资料后发现，这可能与louvain算法的单次局部最优化有关系。

聚类的顺序是不一定的，结果是随机的。有可能在某一轮聚类中，把一个结点聚为了某一社区的子节点，而它在下一轮中更适合作为其它节点的父节点，所以在下一轮又有一部分节点因为该节点加入了社区。但实际上可能，这两轮聚类，更适合把这两次操作分为两个社区。而louvain算法的单次局部最优造成本该是两个社区的节点们，因为某一个“关键节点”的错误聚类，而变成了一个社区。

查阅资料后，可以通过随即邻居Louvain算法进行改进：不考虑所有的邻居社区，而是简单地选择一个随机邻居，并考虑该社区。