数据挖掘与分析第二次上机实验报告

江昱峰 21009200038

2023 年 11 月 19 日

1 背景介绍

聚类分析是数据挖掘和分析领域中常用的一种技术，用于将数据集中的对象按照相似性进行分组。聚类分析可以帮助我们发现数据中的隐藏模式、结构和关系，从而对数据进行更深入的理解和分析。

在现实生活和工业实践中，数据集往往包含大量的样本和特征，而且这些数据的结构和关系不容易直观地观察和理解。聚类分析可以帮助我们从复杂的数据集中提取出有意义的信息，发现数据中的群组或簇，并将相似的样本归类到同一个簇中。

聚类分析在各个领域都有广泛的应用。例如，在市场营销中，可以利用聚类分析来识别具有相似购买行为的消费者群体，从而制定针对性的营销策略。在生物学领域，聚类分析可以用于基因表达数据的分类和研究，帮助科学家发现基因的功能和相互作用关系。在社交网络分析中，聚类分析可以用于识别具有相似兴趣和行为的用户群体，从而推荐个性化的内容和服务。

2 实验目的

实验目的是实践并掌握聚类分析有关的内容，具体包括以下两部分：

- K 均值聚类算法原理与实现；
- 利用真实标签计算聚类指标。

3 任务描述

-

4 数据描述

1. zeisel.pdf：数据来源的文献，包括数据相关介绍，不需要详细了解；
2. Zeisel_df: 小鼠的大规模单细胞 RNA 测序（RNA-seq）数据集，已经数据预处理过，对细胞类型已经筛选过，行是细胞，列是基因；
3. sclabel:Zeisel_df 里细胞对应的真实细胞类型标签。

5 实验原理

5.1 主成分分析（PCA）

假设有 n 个样本，p 个指标，则可构成大小为 n×p 的样本矩阵 x：

$$\mathrm{x}=\begin{bmatrix} \mathrm{x}_{11} & \mathrm{x}_{12} & \cdots & \mathrm{x}_{1\mathrm{p}} \\ \mathrm{x}_{21} & \mathrm{x}_{22} & \cdots & \mathrm{x}_{2\mathrm{p}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{x}_{\mathrm{n}1} & \mathrm{x}_{\mathrm{n}2} & \cdots & \mathrm{x}_{\mathrm{np}} \end{bmatrix}=(\mathrm{x}_1,\mathrm{x}_2,\cdots,\mathrm{x}_{\mathrm{p}}) \tag{1}$$

1. 首先对其进行标准化处理：
   - 按列计算均值: $\bar{x}_{\mathrm{j}}=\frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}}\mathrm{x}_{\mathrm{ij}}$
   - 标准差: $\mathrm{S}_{\mathrm{j}}=\sqrt{\frac{\sum_{\mathrm{i}=1}^{\mathrm{n}}(\mathrm{x}_{\mathrm{ij}}-\overline{\mathrm{x}}_{\mathrm{j}})^2}{\mathrm{n}-1}}$
   - 标准化数据: $\mathrm{X}_{\mathrm{ij}}=\frac{\mathrm{x}_{\mathrm{ij}}-\overline{\mathrm{x}}_{\mathrm{j}}}{\mathrm{S}_{\mathrm{j}}}$
   - 原始样本矩阵经过标准化变化:
     $$\mathrm{X}=\begin{bmatrix} \mathrm{X}_{11} & \mathrm{X}_{12} & \cdots & \mathrm{X}_{1\mathrm{p}} \\ \mathrm{X}_{21} & \mathrm{X}_{22} & \cdots & \mathrm{X}_{2\mathrm{p}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{X}_{\mathrm{n}1} & \mathrm{X}_{\mathrm{n}2} & \cdots & \mathrm{X}_{\mathrm{np}} \end{bmatrix}=(\mathrm{X}_1,\mathrm{X}_2,\cdots,\mathrm{X}_{\mathrm{p}})$$
2. 计算标准化样本查的协方差矩阵：

$$\mathbf{R}=\begin{bmatrix} \mathbf{r}_{11} & \mathrm{r}_{12} & \cdots & \mathbf{r}_{1\mathrm{p}} \\ \mathbf{r}_{21} & \mathrm{r}_{22} & \cdots & \mathbf{r}_{2\mathrm{p}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{r}_{\mathrm{p}1} & \mathrm{r}_{\mathrm{p}2} & \cdots & \mathbf{r}_{\mathrm{pp}} \end{bmatrix} \tag{2}$$

$$r_{ij} = \frac{1}{n-1}\sum_{k=1}^{n}\left(X_{ki} - \overline{X}_i\right)\left(X_{ki} - \overline{X}_j\right) = \frac{1}{n-1}\sum_{k=1}^{n} X_{ki}X_{kj} \qquad (3)$$

3. 计算 R 的特征值和特征值向量：

   - 特征值: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$, （$R$ 是半正定矩阵, 且 $tr(R) = \sum_{k=1}^{p}\lambda_k = p$）
   - 特征向量: $a_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{p1} \end{bmatrix}, a_2 = \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{p2} \end{bmatrix}, \cdots, a_p = \begin{bmatrix} a_{1p} \\ a_{2p} \\ \vdots \\ a_{pp} \end{bmatrix}$

4. 计算主成分共享率以及累计贡献率：
   贡献率 $= \frac{\lambda_i}{\sum_{k=1}^{p}\lambda_k}$, 累加贡献率 $= \frac{\sum_{k=1}^{i}\lambda_k}{\sum_{k=1}^{p}\lambda_k}$, $(i = 1, 2, \cdots, p)$

5. 写出主成分：
   一般取累计贡献率超过 80% 的特征值所对应的第一、第二、... 第 $m(m \leq p)$ 个主成分。第 i 个主成分: $F_i = a_{1i}X_1 + a_{2i}X_2 + \cdots + a_{pi}X_p$

6. 根据系数分析主成分代表的意义：
   对于某个主成分而言，指标前面的系数越大，代表该指标对于该主成分的影响越大。

5.2 Leiden 算法

Leiden 算法步骤伪代码如下图所示：

表 1: Leiden 算法伪代码

序号	具体步骤
1:	**function** LeIDEN(Graph G, Partition $\mathcal{P}$)
2:	**do**
3:	$\mathcal{P} \leftarrow MovenodesFast(G, \mathcal{P})$
4:	done $\leftarrow |\mathcal{P}| = |V(G)|$
5:	**if** not done **then**
6:	$\mathcal{P}_{refined} \leftarrow$ RefinePartition $(G, \mathcal{P})$
7:	$G \leftarrow AggregateGraph\,(G, \mathcal{P}_{refined})$
8:	$\mathcal{P} \leftarrow \{\{v \mid v \subseteq C, v \in V(G)\} \mid C \in \mathcal{P}\}$
9:	**end if**
10:	**while** not done
11:	**return** flat $^*(\mathcal{P})$
12:	**end function**
13:	**function** MoveNodesFast(Graph G, Partition $\mathcal{P}$)
14:	$Q \leftarrow QUeUE(V(G))$
15:	**do**
16:	$v \leftarrow Q$.remove()
17:	$C' \leftarrow \arg\max_{C \in \mathcal{P} \cup \emptyset} \Delta\mathcal{H}_{\mathcal{P}}(v \mapsto C)$
18:	**if** $\Delta\mathcal{H}_P\,(v \mapsto C') > 0$ **then**
19:	$v \mapsto C'$
20:	$N \leftarrow \{u \mid (u, v) \in E(G), u \notin C'\}$
21:	$Q.add(N - Q)$
22:	**end if**
23:	**while** $Q \neq \emptyset$
24:	**return** $\mathcal{P}$
25:	**end function**
26:	**function** RefinePartition(Graph G, Partition $\mathcal{P}$)
27:	$\mathcal{P}_{refined} \leftarrow$ SingletonPartition (G)
28:	**for** $C \in \mathcal{P}$ **do**
29:	$\mathcal{P}_{refined} \leftarrow$ MergenodesSubset $(G, \mathcal{P}_{refined}, C)$
30:	**end for**
31:	**return** $\mathcal{P}_{refined}$
32:	**end function**

function MergeNodesSubset(Graph G, Partition $\mathcal{P}$, Subset S)
  $R = \{v \mid v \in S, E(v, S-v) \geq \gamma \|v\| \cdot (\|S\| - \|v\|)\}$ ∇
  for $v \in R$ **do**
    if v in singleton community **then** D
      $\mathcal{T} \leftarrow \{C \mid C \in \mathcal{P}, C \subseteq S, E(C, S-C) \geq \gamma \|C\| \cdot (\|S\| - \|C\|)\}$
      $Pr\left(C' = C\right) \sim \begin{cases} \exp\left(\frac{1}{\theta}\Delta\mathcal{H}_p(v \mapsto C)\right) & if \Delta\mathcal{H}_P(v \mapsto C) \geq 0 \\ 0 & otherwise \end{cases}$ for $C \in \mathcal{T}$
      $v \mapsto C'$
    end if
  end for
  return $\mathcal{P}$
end function

function AggregateGraph(Graph G, Partition $\mathcal{P}$)
  $V \leftarrow \mathcal{P}$
  $E \leftarrow \{(C, D) \mid (u, v) \in E(G), u \in C \in \mathcal{P}, v \in D \in \mathcal{P}\}$
  return Graph (V, E)
end function

function SingletonPartition(Graph G)
  return $\{\{v\} \mid v \in V(G)\}$
end function

5.3 聚类指标

聚类的混淆矩阵如下表所示：

表 2: 聚类混淆矩阵

	同簇	非同簇
同类	TP	FN
非同类	FP	TN

基于此混淆矩阵，实验中涉及的聚类指标名称、含义与计算公式如下表所示：

表 3: 聚类指标名称、含义与计算公式

聚类指标	意义	计算公式
RI	结果一致性	$RI = \frac{TP+TN}{TP+FP+FN+TN}$
ARI	考虑随机分类影响	$ARI = \frac{2\times(TP\cdot TN-FN\cdot FP)}{(TP+FN)(FN+TN)+(TP+FP)(FP+TN)}$
MI	结果相关性	$\mathrm{MI(U,V)} = \sum_{\mathrm{i=1}}^{\mathrm{R}} \sum_{\mathrm{j=1}}^{\mathrm{C}} \mathrm{p_{i,j}} \log\left(\frac{\mathrm{p_{i,j}}}{\mathrm{p_i}\times\mathrm{p_j}}\right)$
AMI	考虑随机分类影响	$AMI\mathrm{(U,V)} = \frac{\mathrm{MI(U,V)}-E\{\mathrm{MI(U,V)}\}}{\mathrm{F(H(U),H(V))}-E\{\mathrm{MI(U,V)}\}}$
NMI	互信息分数标准化	$NMI\mathrm{(U,V)} = \frac{\mathrm{MI(U,V)}}{\mathrm{F(H(U),H(V))}}$
FMI	细胞分到同簇概率	$\mathrm{FMI} = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP+FP})(\mathrm{TP+FN})}}$
纯度	结果中每簇为同类程度	$Purity = \sum_{\mathrm{i=1}}^{\mathrm{k}} \frac{\mathrm{m_i}}{\mathrm{m}} \mathrm{p_i}$
精确度	正确细胞比例	$\mathrm{Precision} = \frac{TP}{TP+FP}$
准确度	正样本中实际也是比例	$ACC = \frac{\sum_{i=1}^{n} \delta(s_i, map(r_i))}{n}$
召回率	真实分类中正确细胞比例	$Recall = \frac{TP}{TP+FN}$
F1 score	综合准确性和召回率	$F\text{-score} = \frac{2Precision*Recall}{Precision+Recall}$
Jaccard 系数	结果相似性	$Jaccard = \frac{TP}{TP+FN+TN}$

5.4 KM（Kuhn-Munkres）算法

KM 算法步骤如下：

1. 初始化可行顶标的值；
2. 用匈牙利算法寻找完备匹配；
3. 若未找到完备匹配则修改可行顶标的值；
4. 重复步骤 2、3 直到找到相等子图的完备匹配为止。

其中，匈利亚算法步骤如下：

1. 置 M（无向图 G 的一个匹配）为空；
2. 找出一条增广路径 P，通过异或操作获得更大的匹配 M' 代替 M；
3. 重复步骤 2 直到找不出增广路径为止。

6 实验环境

本次实验的实验环境见下表：

表 4: 实验环境

类型	工具配置
操作系统	AWS(亚马逊云) 服务器- Linux（Ubuntu 20.04）
编程语言	Python 3.10（Anaconda 2023 - Jupyter Notebook）
编译器	VScode

7 分析步骤

第一步当然是数据读取。我用 pandas 库的 read_csv 函数对 CSV 文件数据进行读取操作。

由于已经进行过数据预处理，此处便直接进入数据分析步骤。

在进行聚类分析步骤之前，为了简化聚类的计算规模，我先对数据进行了降维操作。常用的降维算法有主成分分析（PCA）、均匀流形近似和投影（UMAP）、线性判别分析（LDA）、因子分析（FC）、独立分量分析（ICA）、等度量映射（ISOMAP）、t-随机邻近嵌入（t-SNE）、局部线性嵌入（LLE）等等。这里我选择最为典型、使用广泛的主成分分析（PCA）算法来进行降维。具体步骤见对应的实验原理部分。

接下来开始正式的聚类分析步骤。首先需要选择使用的算法模型。由于单细胞基因表达数据没有明显的密度、层次等差异特征，也并不要求最终聚类成不同层次，同时为了能够选择更好的初始质心，从而提高算法的收敛速度、降低陷入局部最优解的风险、显著的改善分类结果的最终误差，我决定选择 Leiden 算法。然后，我计算了邻域图，并将图形嵌入二维。接着，我通过 Leiden 算法进行聚类分析并可视化，具体步骤见对应的实验原理部分。

得到聚类结果后，我进一步计算了每类细胞的标记基因并可视化，绘制了标记基因对气泡图、小提琴图等等。

最后，我先用 KM（Kuhn-Munkres）算法在聚类分析结果标签与真实标签之间建立映射，得到与真实标签关联后的聚类标签，具体步骤见对应的实验原理部分。然后，我根据此聚类标签与真实标签计算了一些聚类指标，除了常见的 RI（兰德系数）、ARI（调整兰德系数）、MI（互信息分数）、NMI

（标准化互信息分数）、AMI（调整互信息分数）、FMI（Fowlkes-Mallows 指数）之外，还包括课本中簇有效性的面向分类的度量（熵、纯度 Purity、精确度 Precesion、召回率 Recall、F 度量）和面向相似性的度量（Jaccard 系数）等等。

8 实验结果与可视化

主成分分析（PCA）步骤结果：

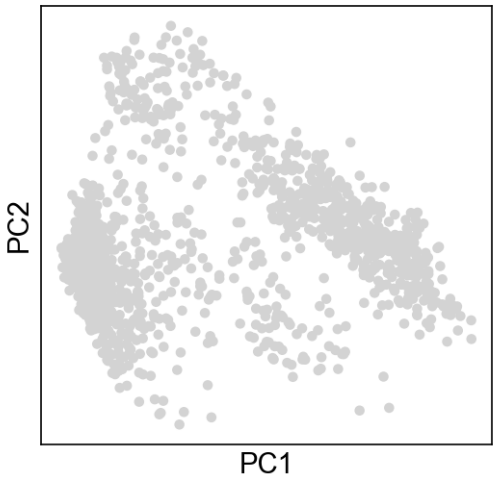

图 1: PCA 散点图

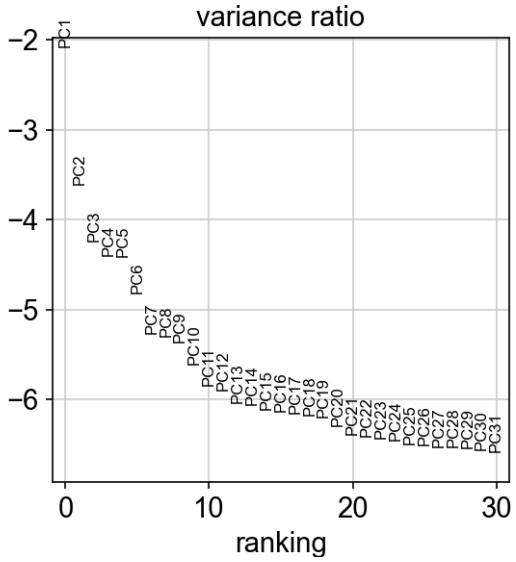

图 2: 单个 PC 对数据总方差的贡献

聚类分析步骤结果：

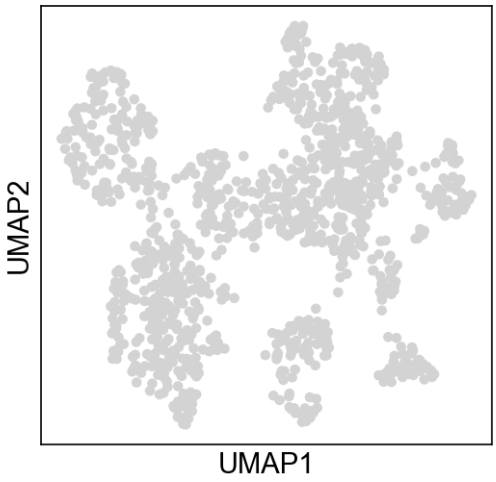

图 3: leiden 聚类邻域图

聚类标签、结果矩阵（未标记细胞类型）的 h5ad 文件见实验结果目录。
每类细胞标记基因计算步骤结果：

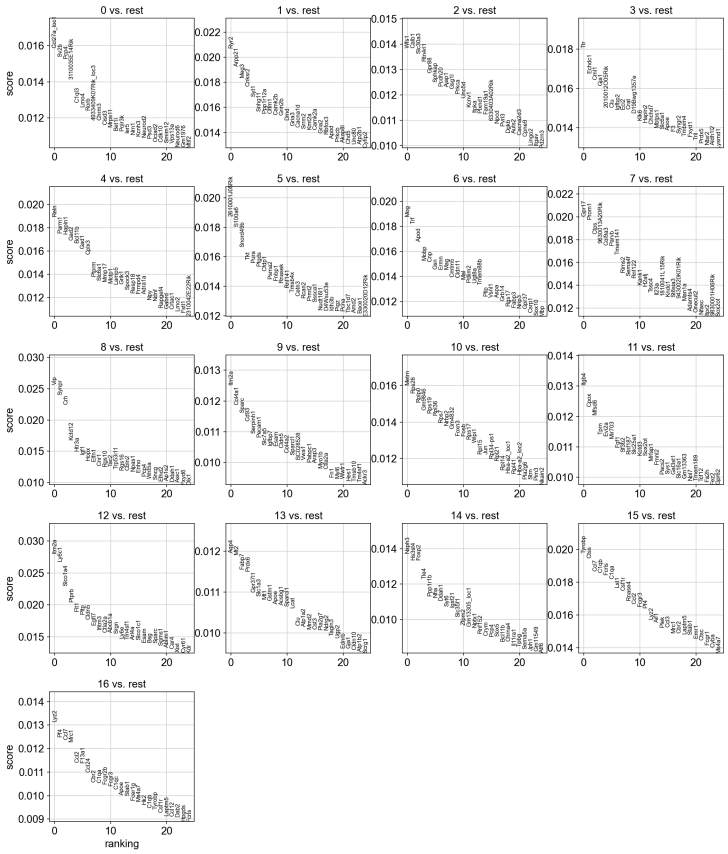

图 4: 每个簇中高度差异基因排名

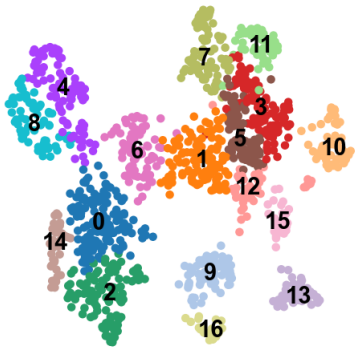

图 5: 聚类结果散点图（标记细胞类型）

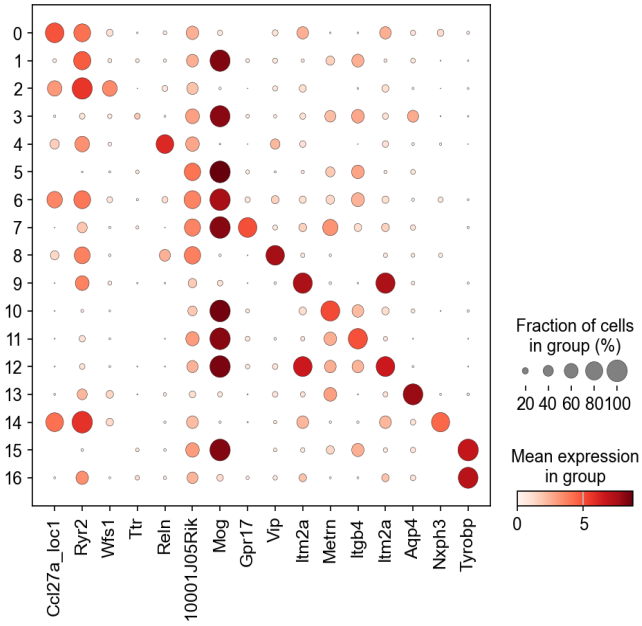

图 6: 标记基因气泡图

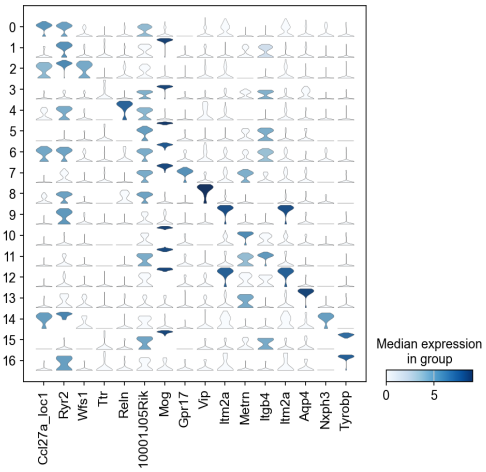

图 7: 标记基因小提琴图

聚类得分矩阵（标记细胞类型）的 h5ad 文件见实验结果目录。
聚类指标计算结果:

表 5: 聚类指标计算结果

聚类指标	计算结果
RI（兰德系数）	0.732
ARI（调整兰德系数）	0.779
MI（互信息分数）	0.779
AMI（调整互信息分数）	0.549
NMI（标准化互信息分数）	0.563
FMI（Fowlkes-Mallows 指数）	0.405
Purity（纯度）	92.217%
Precision（精确度）	91.817%
ACC（准确度）	88.409%
Recall（召回率）	96.435%
F1 score（F 度量）	0.968
Jaccard 系数	0.982

9 结果分析

每个簇中的特征基因各不相同。根据聚类指标的意义，可知聚类效果良好，总体准确性、聚类标签与真实标签的相似度较高。

10 实验讨论

10.1 优缺点分析

本次实验过程有以下三点优点：

- 降维操作所用的主成分分析（PCA）算法是一种经典的线性降维算法，能够有效地捕捉数据中的主要变化方向，减少数据的维度并保留尽可能多的信息，且计算简单、易于理解；
- 聚类分析所用的 Leiden 算法是一种用于图聚类的优化算法，有着高效的聚类性能，支持重叠社区检测，自适应分辨率参数；
- 使用的模型算法具有鲁棒性、可迁移性，聚类结果准确合理。

不过，实验也存在以下两点缺点：

- 降维操作所用的主成分分析（PCA）算法假设数据是线性可分的，对于非线性关系的数据降维效果可能较差；
- 聚类分析所用的 Leiden 算法计算复杂度较高，没有明确的准则来指导参数的选择，对噪声和稀疏图较为敏感。

10.2 改进方法

本次实验有以下两点可以进一步改进的方法：

- 降维操作可以选用对非线性关系数据也较为友好的方法；
- 聚类分析可以选用复杂度较低的方法。

11 心得体会

做完本次实验，除了掌握了实验目的部分中所有内容的收获之外，我还有以下几点心得体会：

- 聚类分析前最好能先对数据进行降维；
- 得到聚类结果之后可以进一步计算每类的特征。

12 参考文献

[1] Xiu Yan,《清风数学建模学习笔记——主成分分析 (PCA) 原理详解及案例分析》, https://blog.csdn.net/weixin_43819566/article/details/113800120

[2] 无尽攀登,《动手学单细胞分析-基础-2.5 聚类之 Leiden》,https://zhuanlan.zhihu.com/p/538605686