数据解读

X（基因表达矩阵），POS（坐标矩阵），y（真实标记）

POS：2*？，2是空间坐标xy，？是spot数目或细胞数目，和X能对应上

这些Rdata是下载自bayesspace的tutorial；zip里面是用spatialPCA 里面的svg算法选了genes，所以可以直接用h5作为输入，具体这些h5是怎么生成的的可以参考R代码。 Histology图片需要大家自己找到数据出处自行下载。

可以直接使用h5文件作为模型的输入

RData里面是原始数据，h5里面的X矩阵存的是做个feature selection的count数据，

大约最后留下来2000个feature/genes，这些feature是比较有信息的feature

想运行stlearn，可以用这个folder，这个里面有数据，有hitology image

大家可以follow这个装python的package读这个里面提供的原始数据也是一样的

当然最推荐的还是大家使用h5文件，毕竟那个是处理过的数据，里面把有用的feature都留下来了，没啥信息的feature全去掉了

第一个方向，不指定具体算法

聚类可以自己设计，如self-training的spatialPCA 用的self-clustering

更常见的，先降维，然后用经典的k-means等

最后聚类的结果用ARI和NMI两个指标计算就可以了，可以根据ref里面的文章画一下ARI和NMI的图，并挑选一两个数据展示一下聚类的结果

ARI和NMI网上都有package算呢

第二个方向，更灵活

高斯过程

12个样本，每个编号看成一个组织切片

h5文件是从rdata里抽出来的，怎么抽的看代码，从原始3000个点，每个点1w+基因（基因需要筛选有用的，筛选方法代码有，筛2000个）。是count数据（12345）不是正态数据，设计loss函数时，如用AE，需要用？？？。用count分布去算似然比较合适

SpaGCN：图转基

有代码，效果不是很好，不能直接用。因为做了很多奇怪的操作

先把所有基因标准化（不推荐），算MSE，需要用Rdata，里面存了标准化后的基因，可以对数化后直接拿来用

矩阵比如2000*3000，一行表示一个基因，一列表示一个cell，测的是每一个cell在空间上的基因表达和坐标
取top50的pc，经过GCN的网络做了聚类，用了self-training
用cell测得的location建了图，用图在pca结果上算convolution
建图：点和周边点算距离，得到kNN的graph，在graph上做convolution，每个点自带一个基因表达向量。

空间转录组技术：把图片分解成点，一个一个穿起来，可以测出每个小圈内遗传物质表达了多少次，测出1w多基因，剩下的测不出来，表达就是0。知道点的空间位置信息，有空间相对位置坐标，存在h5文件pos，matrix里，基因表达存在X里，y代表了不同的点label（细胞类型）
真正用的数据是带15xxxx的数据，希望测的和人标注的ground truth接近。自己聚类算出的lable和ground truth的算matrix，用ARi和NMI算，有包
每个点有两个信息：基因表达信息和空间坐标位置
假设：空间位置相似的点，可能来自同一个细胞类型
关键是要用深度学习做聚类

STAGATE：图注意力（更好）
用AE架构，编码--解码，学出来的latent representation拿去做聚类
建图：在knn上优化，先预聚类，先降维再聚类。（边界地方，knn不行，所以这个方法好）根据聚类结果，优化空间邻居的网络图，不在同一类别的邻居边要去掉，graph有α权重，用graph attention方法
建议算概率，多少概率来自这个类，多少来自另一个类

SpatialPCA：基于PCA
正常PCA从贝叶斯角度分析，可以看成Y=WZ+E（随机扰动）
PCA要求矩阵分解是正交的，W^T@W=I_d

一个spot有一个latent representation向量，拼起来就是Z
原来的Z是对角矩阵，相邻spot没有关系。希望相邻两点的latent representation接近，就在sigma_I非对角线的部分给一些tolerance值（用核函数，距离越远，相关度越小）。越接近，sigma_I会有更大的corelation，即空间距离转化为latent representation的依赖关系
高斯process prior和这个很像，也假设空间接近的点latent representation相近

Histology image
辅助验证，利用信息来增强结果

一个naive方法：（**SpaGCN）点所在位置取一个50*50的小方块，算一下RGB值和其他进行比较，把RGB值当作第三个空间坐标进行建图**

stlearn：用DI学H&E image的feature，feature相近的去处理，用来做基因的归一化，归一化使得spot上

如何能不止用在归一化上，还能用在latent representation上?

搞清楚：

h5文件，查看，使用

R代码看懂?

STAGATE还是SpatialOCA，看论文找代码

ARI，NMI

- ARI调蓝德指数，范围为[-1,1]，值越大越好
- NMI标准化互信息，范围为[0,1]，值越大越好

RI：定义a 为在C中被划分为同一类，在K中被划分为同一簇的实例对数量。定义b为在C中被划分为不同类别，在K中被划分为不同簇的实例对数量

无法保证随机划分的聚类结果的RI值接近0，因此有ARI

（好的评价指标，随机产生的要接近0）

```
from sklearn.metrics import adjusted_rand_score
labels_true = [0, 0, 0, 1, 1, 1]
labels_pred = [0, 0, 1, 1, 2, 2]
print(adjusted_rand_score(labels_true, labels_pred))

from sklearn.metrics.cluster import normalized_mutual_info_score
NMI = lambda x, y: normalized_mutual_info_score(x, y, average_method='arithmetic')
C = [1, 1, 2, 2, 3, 3, 3]
D = [1, 1, 1, 2, 1, 1, 1]
print(NMI(C, D))
```

深度学习怎么聚类?

好多甚至是当前SOTA的方法，还是autoencoder和K-means的结合，基本思路都是通过深度学习+CNN+autoencoder提取特征，即完成降噪降维，而后在embeding space中通过K-means进行聚类

（按照现在深度学习界通用的理解（其实是偏离了原意的），Embedding就是从原始数据提取出来的Feature，也就是那个通过神经网络映射之后的低维向量）

尝试

- 边界识别，增强：预聚类，Louvain algorithm with a small resolution parameter (set as 0.2 by default)
- 深度学习提取特征，改变X矩阵
- 利用真实标记图像，计算分割平行线，改变距离计算方式为：减少平行线上方向的距离权重，可以控制聚类结果也有类似的平行线表现.注意：还要规定所有质心在与平行线垂直的线上

最简单的k-means
把附近k*k个点的平均基因表达，乘上权重系数后，作为当前点的第三维坐标向量，然后用这个三维坐标的欧式距离作为聚类标准
knn原理，得到结果后进行去噪声优化，如果一个点（随机而不是遍历）周围都是其他类，那么改变这个点的类

再挣扎一下

- 质心平行线
- 增加网络层数
- PCA降维看看