会议纪要

2021/11/8

本周汇报内容

1. 完整复现代码，代码运行初步收敛并得到一定的运行结果。
2. **A**.每轮训练结束T\_value Matrix的二次模逐渐收敛至稳定值。

**B.**输出结果中存在部分路径的最优结果是不分割。

**C**.经case study发现影响字典生成的因素包括：

每条路径的密度（即1/T\_value）,代表每条轨迹被不同路径的重合度，重合度越高，该数值越小，该条路径有更大概率被选入Pathlet集合。

λ的数值

若有一条路径b，存在某种分割b1+b2。那么本算法在选择是否分割的时候会去计算diff=λ+，若该值大于零，则我们选择不分割，反之则选择分开。

分析可得，

a)若λ选择无限大，那么其实我们在每次选择的时候都倾向不分开。

b)若λ选取无限小，那么完全是根据密度去决定是否选择分割开路径。

**需要注意的是，该算法实际使用该密度去采样统计路网的主要性权重，但是两者不同点在于路网权重不受路径多少的影响，而密度正比于路径的数量。这就使得本算法输出的结果会随着数据集中路径数量的变化而变化，但是本身应该将这种密度的差异与路网边的重要性差异分开，我们希望输出结果不随数据集数量的变化而变化。**

解决方法，通过调整λ+的计算方法，使其带有概率的含义，不随数据集数量的变化而变化。

**D**.经观察发现每轮训练中，输出的分割集合中元素的数量都是先增后减，说明我们的确发现了部分共有的路径，并且通过分析结果发现的确符合我们在生成数据时期望的结果。下一步可以去观察不同epoch执行结束后输出集合的结果，对比分析。

未来计划：

1. 尝试设置λ使得所有的轨迹都被分开
2. 观察不同epoch执行结束后输出集合的结果，对比分析。
3. 调整λ+的计算方法，使其带有概率的含义，使得输出结果不随数据集数量的变化而变化
4. 寻找输出结果中词典长度为K的最频繁pathlet，最长、短的pathlet等带有统计意义的结果
5. 在输出集合中添加pathlet长度的限制
6. 检验输出集合对于原始路径覆盖率和对于新路径的decomposition
7. 复现最短路径思想的代码