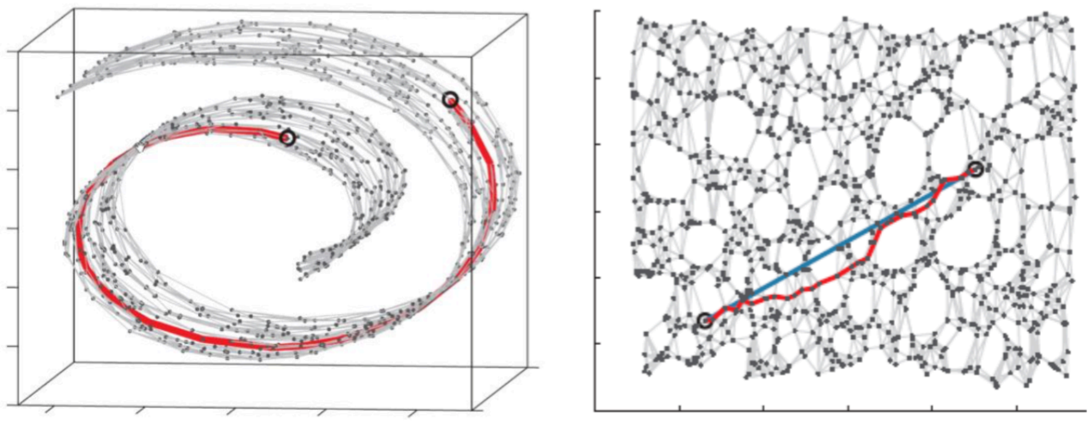
# IsoMap 与 LE

**流形**，定义为很多区平面的叠加，任意一个流形都可以嵌入到足够高维度的欧式空间中；
简单地说，我们认为一张弯曲的纸是一个二维流形被放入了三维的欧式空间中。

在数据处理中，我们假设数据是位于一个低维流形上的，但是其现在在一个高维欧式空间中！
我们的目标就是希望可以找到数据的低维流形表示（假设存在低维流形），从而达到降维的目的，也就是通常说的在高维空间展开一个流形（unfold a manifold）。要求我们在展开的时候能够保持测地距离，也就是我们认为的低维流形空间中的距离。

测地距离：



在上图中，右侧是我们希望展开的低维流形（二维）；左侧是我们在高维（三维）空间的一个数据表示。
我们希望在展开流形的时候，在低维流形上各点之间的距离有所保证，那么就要求我们在高维空间中可以近似地将这个距离计算出来，进而才可做做到保证这个距离相对变化较小。
实践中，不同方法有不同的策略。

**IsoMap：**
图上两点之间的最短路径，可以近似地对应于流行上的测地线距离。当数据点趋向于无穷多的时候，这个估计趋向于真实的测地线距离。
我们假设映射后各点的坐标表示为 yi，那么我们的目标可以写成：

$$\min_{y}\sum_{i,j}\left(d_{\mathcal{M}}(x_i,x_j)-\|y_i-y_j\|\right)^2$$

其中 dm(xi,xj)表示我们使用图上两点间最短路径近似的测地线距离，之后希望高维下的距离和低维流形中的距离尽可能相等。也即取 min。

步骤：

- Step 1. Construct neighborhood graph
  - using the connectivity algorithms such as k nearest neighbor
- Step 2. Find the shortest path between two points
  - gives a good estimate of the geodesic distance between two points.
- Step 3. Low dimensional embedding
  - applying multidimensional scaling on the resulting distance metric

**全局 VS 局部：**

IsoMap 方法需要考虑任意两点间的距离，扩展性较差；
实际中，我们经常考虑使用局部分析方法。

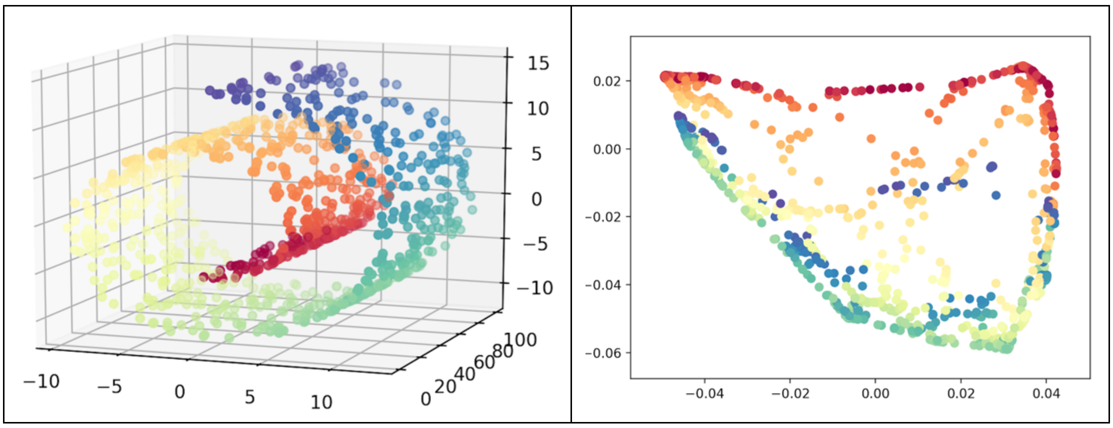
**LE：**

希望保持流形的近邻关系，即希望高维空间中的近似点，在低维流形中一样比较接近。（这样就避免了全局的一些计算）
算法步骤：

- 第一步：构建近邻图
- 第二步：计算每条边的权重(不相连的边权重为0)
  - 热核权重：$w_{ij} = \exp\left(\frac{\|x_i - x_j\|^2}{\sigma^2}\right)$
  - 0-1 权重：$w_{ij} = 1$
- 第三步：求解特征向量方程，$Ly = \lambda y$，将点$x_i$映射到 $(y_1(i), \dots, y_d(i))$

**实践尝试：**

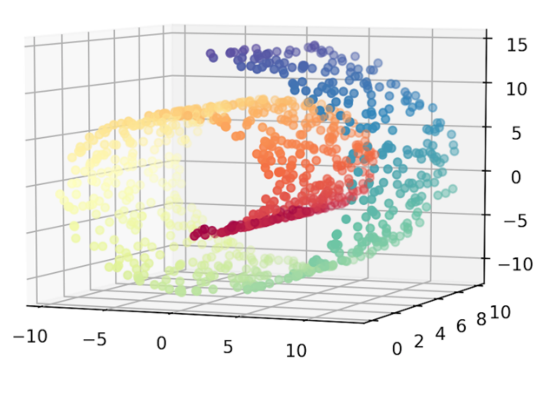
对于 LE 算法，基于生成的 swiss roll data，这里做了一些测试：
尝试 1：

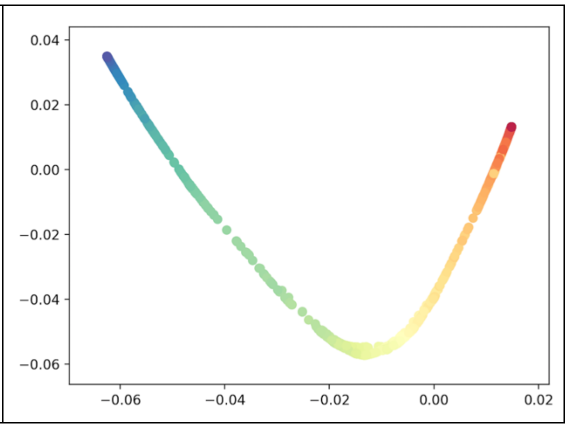



如上图所示，其中有一条轴（深红点展开的轴）的维度为[0,100]，我们将其使用

LE 方法降维，可以发现，此维度上的顶点之间因为原先间距较大的原因，在右侧展开的流形上，距离也较远；

尝试 2：但是当我们将这条轴的幅度减小为：[0,10]




我们发现，红色点间距离变小，在低维流形中汇聚在一起。

Alan
2017/03/13