一、 spectral clustering 算法简介

如果说 K-means 和 GMM 这些聚类的方法是古代流行的算法的话，那么这次要讲的 Spectral Clustering 就可以算是现代流行的算法了，中文通常称为“谱聚类”。由于使用的矩阵的细微差别，谱聚类实际上可以说是一“类”算法。

谱聚类是从图论中演化出来的算法，后来在聚类中得到了广泛的应用。它的主要思想是把所有的数据看做空间中的点，这些点之间可以用边连接起来。距离较远的两个点之间的边权重值较低，而距离较近的两个点之间的边权重值较高，通过对所有数据点组成的图进行切图，让切图后不同的子图间边权重和尽可能的低，而子图内的边权重和尽可能的高，从而达到聚类的目的。

二、 spectral clustering 算法基本概念

- **复杂网络的表示**

在复杂网络的表示中，复杂网络可以建模成一个图 $G=(V,E)$，其中，V 表示网络中的节点的集合，E表示的是连接的集合。在复杂网络中，复杂网络可以是无向图、有向图、加权图或者超图。

- **邻接矩阵**

邻接矩阵 W，它是由任意两点之间的权重值w_{ij}组成的矩阵，获得这个矩阵的基本思想是，距离较远的两个点之间的边权重值较低，而距离较近的两个点之间的边权重值较高，可以通过样本点距离度量的相似矩阵 S 来获得邻接矩阵 W。

一般采用全连接法简历邻接矩阵，当采用高斯核函数时，相似矩阵和邻

接矩阵相同，公式如下：

$$w_{ij} = s_{ij} = exp(-\frac{\|x_i - x_j\|_2^2}{2\sigma^2})$$

- **拉普拉斯矩阵**

  拉普拉斯矩阵 L=D-W，其中 D 是度矩阵，W 是邻接矩阵。

  拉普拉斯矩阵有如下性质：

  - 拉普拉斯矩阵是对称矩阵
  - 所有的特征值都是实数
  - 对于任意的向量f，有

$$f^T\ Lf = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2$$

三、 spectral clustering 算法原理

在复杂网络的网络簇结构存在着同簇节点之间连接密集，不同簇节点之间连接稀疏的特征，是否可以根据这样的特征对网络中的节点进行聚类，使得同类节点之间的连接密集，不同类别节点之间的连接稀疏？

在谱聚类中定义了“截”函数的概念，当一个网络被划分成为两个子网络时，“截”即指子网间的连接密度。谱聚类的目的就是要找到一种合理的分割，使得分割后形成若干子图，连接不同的子图的边的权重尽可能低，即“截”最小，同子图内的边的权重尽可能高。

“截”表示的是子网间的密度，即边比较少。以二分为例，将图聚类成两个类：S类和T类。假设用 $cut(S,T)$来表示图的划分，我们需要的结果为：

$$\min cut(S,T) = \sum_{i\in S; j\in T} e_{ij} = W(S,T)$$

其中$W(S,T)$表示的是类别S和T之间的权重。对于k个不同的类别$A_1, A_2, \dots, A_k$，优化的目标为：

$$\min cut(A_1, A_2, \dots, A_k) = \frac{1}{2}\sum_{i=1}^{k} W(A_i, \overline{A_i})$$

邻接矩阵：

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1n} \\ e_{21} & e_{22} & \cdots & e_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ e_{n1} & e_{n1} & e_{n1} & e_{n1} \end{bmatrix}$$

min *cut*和*ratiocut*中的Laplacian矩阵：

$$L = D - E \Leftrightarrow -\mathrm{L} = E - D$$

*Normalizedcut*中的L′：

$$\mathrm{L}' = D^{-\frac{1}{2}}\mathrm{L}D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}}ED^{-\frac{1}{2}}$$

可见不管是 L、L′都与 E 联系特别大。如果将 E 看成一个高维向量空间，也能在一定程度上反映 item 之间的关系。将 E 直接 kmeans 聚类，得到的结果也能反映 V 的聚类特性，而谱聚类的引入 L 和 L′是使得 G 的分割具有物理意义。

而且，如果 E 的 item(即 n)足够大，将难计算出它的 kmeans，我们完全可以用 PCA 降维(仍为 top 的特征值与向量)。

四、 spectral clustering 算法流程

i. 根据数据构造一个 Graph ，Graph 的每一个节点对应一个数据点，将相似的点连接起来，并且边的权重用于表示数据之间的相似度。把这个 Graph 用邻接矩阵的形式表示出来，记为 W 。

ii. 把每一列元素加起来得到 N 个数，把它们放在对角线上（其他地方都是零），组成一个 N*N 的矩阵，记为 D 。并令 L = D - W 。

iii. 求出 L 的前 k 个特征值以及对应的特征向量。

iv. 把这 k 个特征（列）向量排列在一起组成一个 N*k 的矩阵，将其中每一行看作 k 维空间中的一个向量，并使用 K-means 算法进行聚类。聚类的结果中每一行所属的类别就是原来 Graph 中的节点亦即最初的 N 个数据点分别所属的类别。

五、 spectral clustering 算法优缺点

优点

- 谱聚类只需要数据之间的相似度矩阵，因此对于处理稀疏数据的聚类很有效。这点传统聚类算法比如 K-Means 很难做到
- 由于使用了降维，因此在处理高维数据聚类时的复杂度比传统聚类算法好。

缺点

- 如果最终聚类的维度非常高，则由于降维的幅度不够，谱聚类的运行速度和最后的聚类效果均不好。
- 聚类效果依赖于相似矩阵，不同的相似矩阵得到的最终聚类效果可能很不同。