原 # 从拉普拉斯矩阵说到谱聚类

2014年11月03日 11:33:37 v_JULY_v 阅读数：70667 更多

从拉普拉斯矩阵说到谱聚类

0 引言

11月1日上午，机器学习班 第7次课，邹讲聚类（PPT），其中的谱聚类引起了自己的兴趣，邹从最基本的概念：单位向量、两个向量的正交、方阵 征向量，讲到相似度图、拉普拉斯矩阵，最后讲谱聚类的目标函数和其算法流程。

课后自己又琢磨了番谱聚类跟拉普拉斯矩阵，打算写篇博客记录学习心得， 若有不足或建议，欢迎随时不吝指出，thanks。

1 矩阵基础

在讲谱聚类之前，有必要了解一些矩阵方面的基础知识。

1.0 理解矩阵的12点数学笔记

如果对矩阵的概念已经模糊，推荐国内一人写的《理解矩阵by孟岩》系列，其中，抛出了很多有趣的观点，我之前在阅读的过程中做了些笔记，如

“1、简而言之：矩阵是线性空间里的变换的描述，相似矩阵则是对同一个线性变换的不同描述。那，何谓空间？本质而言， “空间是容纳运动的一个 变换则规定了对应空间的运动”by孟岩。在线性空间选定基后，向量刻画对象的运动，运动则通过矩阵与向量相乘来施加。然，到底什么是基？坐标系

2、有了基，那么在(1)中所言的则应是：矩阵是线性空间里的变换的描述，相似矩阵则是对同一个线性变换在不同基（坐标系）下的不同描述。出来了两个问题，一者何谓 基（坐标系）如何理解？事实上，所谓变换，即空间里从一个点（元素/对象）到另一个（元素对象）的跃迁，矩阵用来描述线性变换。基呢?通过前面已知，矩阵无非不过 性空间中的线性变换的一个东西而已，线性变换为名词，矩阵为描述它的形容词，正如描述同一个人长得好看可以用多个不同形容词"帅""靓"描述，同一个线性变换也可以由 阵来描述，而由哪一个矩阵描述它，则由基（坐标系）确定。

3、前面说了基，坐标系也，形象表述则为角度，看一个问题的角度不同，描述问题得到的结论也不同，但结论不代表问题本身，同理，对于一个线性变换，可以选定一组基 阵描述它，换一组基，得到不同矩阵描述它，矩阵只是描述线性变换非线性变换本身，类比给一个人选取不同角度拍照。

4、前面都是说矩阵描述线性变换，然，矩阵不仅可以用来描述线性变换，更可以用来描述基（坐标系/角度），前者好理解，无非是通过变换的矩阵把线性空间中的一个点 点上去，但你说矩阵用来描述基(把一个坐标系变换到另一个坐标系)，这可又是何意呢？实际上，变换点与变换坐标系，异曲同工！
（@坎儿井围脖：矩阵还可以用来描述微分和积分变换。关键看基代表什么，用坐标基就是坐标变换。如果基是小波基或傅里叶基，就可以用来描述小波变换或傅里叶变

5、矩阵是线性运动（变换）的描述，矩阵与向量相乘则是实施运动（变换）的过程，同一个变换在不同的坐标系下表现为不同的矩阵，但本质/征值相同，运动是相对的， 于坐标系的变换，如点(1,1)变到(2,3)，一者可以让坐标点移动，二者可以让X轴单位度量长度变成原来1/2，让Y轴单位度量长度变成原来1/3，前后两者都可以达到目的。

6、Ma=b，坐标点移动则是向量a经过矩阵M所描述的变换，变成了向量b；变坐标系则是有一个向量，它在坐标系M的度量下结果为a，在坐标系I（I为单位矩阵，主对角为 度量下结果为b，本质上点运动与变换坐标系两者等价。为何？如(5)所述，同一个变换，不同坐标系下表现不同矩阵，但本质相同。

7、Ib，I在(6)中说为单位坐标系，其实就是我们常说的直角坐标系，如Ma=Ib，在M坐标系里是向量a，在I坐标系里是向量b，本质上就是同一个向量，故此谓矩阵乘法计算 别。且慢，什么是向量？放在坐标系中度量，后把度量的结果（向量在各个坐标轴上投影值）按顺序排列在一起，即成向量。

8、b在I坐标系中则是Ib，a在M坐标系中则是Ma，故而矩阵乘法MxN，不过是N在M坐标系中度量得到MN，而M本身在I坐标系中度量出。故Ma=Ib，M坐标系中的a转过来在 量，却成了b。如向量(x,y)在单位长度均为1的直角坐标系中一量，是(1,1)，而在X轴单位长度为2.Y轴单位长度为3一量则是(2,3)。

9、何谓逆矩阵？Ma=Ib，之前已明了坐标点变换a-〉b等价于坐标系变换M-〉I，但具体M如何变为I呢，答曰让M乘以M的逆矩阵。以坐标系

为例，X轴单位度量长度变为原来的1/2，Y轴单位度量长度变为原来的1/3，即与矩阵

1/2 0
0 1/3

相乘，便成直角坐标系I。即对坐标系施加变换，即让其与变换矩阵相乘。"

1.1 一堆基础概念

根据wikipedia的介绍，在矩阵中，n阶**单位矩阵**，是一个$n \times n$的方形矩阵，其主对角线元素为1，其余元素为0。单位矩阵以I_n表示；如果阶 略，或可由前后文确
简记为I（或者E）。如下图所示，便是一些单位矩阵：

$$I_1 = \begin{bmatrix} 1 \end{bmatrix},\ I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \cdots,\ I_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

单位矩阵中的第i列即为单位向量e_i。单位向量同时也是单位矩阵的特征向量，特征值皆为1，因此这是唯一的特征值，且具有重数n。由此可见，单位矩阵的行列式为1，

单位向量又是什么呢？数学上，赋范向量空间中的单位向量就是长度为 1 的向量。欧几里得空间中，两个单位向量的点积就是它们之间角度的余弦（因为它们的长度都是
一个非零向量$\vec{u}$的正规化向量（即单位向量）$\hat{u}$就是平行于$\vec{u}$的单位向量，记作：

$$\hat{u} = \frac{\vec{u}}{\|\vec{u}\|}$$

这里$\|\vec{u}\|$是$\vec{u}$的范数（长度）。
何谓**点积**？点积又称内积，两个向量$\vec{a}$ = [a1, a2,…, an]和$\vec{b}$ = [b1, b2,…, bn]的点积定义为：

$$\vec{a} \cdot \vec{b} = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

这里的Σ指示求和符号。
例如，两个三维向量[1, 3, -5]和[4, -2, -1]的点积是：

$$\begin{bmatrix} 1 & 3 & -5 \end{bmatrix} \cdot \begin{bmatrix} 4 & -2 & -1 \end{bmatrix} = (1)(4) + (3)(-2) + (-5)(-1) = 3$$

$$\vec{a}\cdot\vec{b}=\vec{a}^T\vec{b}$$

这里的$\vec{a}^T$指示矩阵的转置。使用上面的例子，将一个1×3矩阵（就是行向量）乘以一个3×1向量得到结果(通过矩阵乘法的优势得到1×1矩阵也就...):

$$\begin{bmatrix}1 & 3 & -5\end{bmatrix}\begin{bmatrix}4\\-2\\-1\end{bmatrix}=\begin{bmatrix}3\end{bmatrix}$$

除了上面的代数定义外，点积还有另外一种定义：几何定义。在欧几里得空间中，点积可以直观地定义为：

$$\vec{a}\cdot\vec{b}=|\vec{a}|\,|\vec{b}|\cos\theta$$

这里$|\vec{x}|$表示$\vec{x}$的模（长度），θ表示两个向量之间的角度。 根据这个定义式可得：两个互相垂直的向量的点积总是零。若和都是单位向量（长度为1
积就是它们的夹角的余弦。

正交是垂直这一直观概念的推广，若内积空间中两向量的内积（即点积）为0，则称它们是正交的，相当于这两向量垂直，换言之，如果能够定义向
则正交可以直观的理解为垂直。而**正交矩阵**（orthogonal matrix）是一个元素为实数，而且行与列皆为正交的单位向量的方块矩阵（方块矩阵，或简
数及列数皆相同的矩阵。）

若数字λ和非零向量$\vec{v}$满足$A\vec{v}=\lambda\vec{v}$，则$\vec{v}$为A的一个**特征向量**，λ是其对应的**特征值**。 换句话说，在$\vec{v}$这个方向上，A做的事情无非是把$\vec{v}$沿其$\vec{v}$的方向拉长/缩短了-
无规律的多维变换），λ则是表示沿着这个方向上拉伸了多少的比例。 简言之，A对$\vec{v}$做了手脚，使得向量$\vec{v}$变长或变短了，但$\vec{v}$本身的方向不变。
矩阵的**迹**是矩阵$n\times n$的对角线元素之和，也是其n个特征值之和。
更多矩阵相关的概念可以查阅相关wikipedia，或《矩阵分析与应用》。

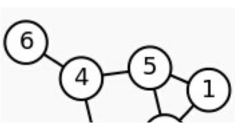
2 拉普拉斯矩阵

2.1 Laplacian matrix的定义

拉普拉斯矩阵（Laplacian matrix)），也称为基尔霍夫矩阵, 是表示图的一种矩阵。给定一个有n个顶点的图$G=(V,E)$，其拉普拉斯矩阵被定义为:

$$L=D-W$$

其中D为图的度矩阵，W为图的邻接矩阵。
举个例子。给定一个简单的图，如下：



把此“图”转换为**邻接矩阵**的形式，记为W：

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

把W的每一列元素加起来得到N个数，然后把它们放在对角线上（其它地方都是零），组成一个$N \times N$的对角矩阵，记为**度矩阵**D，如下图

$$\begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

根据拉普拉斯矩阵的定义 $L = D - W$，可得拉普拉斯矩阵L 为：

$$\begin{pmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & -1 & 0 & -1 & 3 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}$$

2.2 拉普拉斯矩阵的性质

介绍 拉普拉斯矩阵的性质之前，首先定义两个概念，如下：
①对于邻接矩阵，定义图中A子图与B子图之间所有边的权值之和如下：

$$W(A,B) := \sum_{i \in A, j \in B} w_{ij}$$

其中，w_{ij}定义为节点i到节点j的权值，如果两个节点不是相连的，权值为零。
②与某结点邻接的所有边的权值和定义为该顶点的度d，多个d 形成一个度矩阵D （对角阵）

$$d_i = \sum_{j=1}^{n} w_{ij}$$

- $L1=01$，即L的最小特征值是0，相应的特征向量是$\mathbb{1}$。证明：$L*\mathbb{1}=(D-W)*\mathbb{1}=0=0*\mathbb{1}$。（此外 量的定义：若数字$\lambda$ $A\vec{v}=\lambda\vec{v}$，则$\vec{v}$为A的一个**特征向量**，λ是其对应的**特征值**）。
- L 有n个非负实特征值$0=\lambda_1\le\lambda_2\le\ldots\le\lambda_n$
- 且对于任何一个属于实向量$f\in\mathbb{R}^n$，有以下式子成立

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

其中，$L=D-W$，$d_i=\sum_{j=1}^{n}w_{ij}$，$W(A,B):=\sum_{i\in A,j\in B}w_{ij}$。
下面，来证明下上述结论，如下：

$$f'Lf=f'Df-f'Wf=\sum_{i=1}^{n}d_if_i^2-\sum_{i,j=1}^{n}f_if_jw_{ij}$$
$$=\frac{1}{2}\left(\sum_{i=1}^{n}d_if_i^2-2\sum_{i,j=1}^{n}f_if_jw_{ij}+\sum_{j=1}^{n}d_jf_j^2\right)=\frac{1}{2}\sum_{i,j=1}^{n}w_{ij}(f_i-f_j)^2$$

3 谱聚类

所谓聚类（Clustering），就是要把一堆样本合理地分成两份或者K份。从图论的角度来说，聚类的问题就相当于一个图的分割问题。即给定一个图G = (V, E)，顶点集V表 带权的边表示各个样本之间的相似度，谱聚类的目的便是要找到一种合理的分割图的方法，使得分割后形成若干个子图，连接不同子图的边的权重（相似度）尽可能低，同 重（相似度）尽可能高。物以类聚，人以群分，相似的在一块儿，不相似的彼此远离。
至于如何把图的顶点集分割/切割为不相交的子图有多种办法，如

1. cut/Ratio Cut
2. Normalized Cut
3. 不基于图，而是转换成SVD能解的问题

目的是为了要让被割掉各边的权值和最小，因为被砍掉的边的权值和越小，代表被它们连接的子图之间的相似度越小，隔得越远，而相似度低的子图正好可以从中一刀切 本文重点阐述上述的第一种方法，简单提一下第二种，第三种本文不做解释，有兴趣的可以参考文末的参考文献条目13。

3.1 相关定义

为了更好的把谱聚类问题转换为图论问题，定义如下概念（有些概念之前已定义，权当回顾下）：

- 无向图$G=(V,E)$，顶点集V表示各个样本，带权的边表示各个样本之间的相似度
- 与某结点邻接的所有边的权值和定义为该顶点的度d，多个d 形成一个度矩阵D（对角阵）

$$d_i=\sum_{j=1}^{n}w_{ij}$$

- 邻接矩阵W，A子图与B子图之间所有边的权值之和定义如下：

$$W(A,B):=\sum_{i\in A,j\in B}w_{ij}$$

其中，w_{ij}定义为节点i到节点j的权值，如果两个节点不是相连的，权值为零。

- 相似度矩阵的定义。相似度矩阵由权值矩阵得到，实践中一般用**高斯核函数**（也称径向基函数核）计算相似度，距离越大，代表其相似度越小。

$$s(x_i, x_j) = e^{-\|x_i - x_j\|^2/(2\sigma^2)}$$

- 子图A的指示向量如下：

$$\begin{aligned}\mathbb{1}_A &= (f_1, \ldots, f_n)' \in \mathbb{R}^n \\ f_i &= 1 \text{ if } v_i \in A \\ f_i &= 0 \text{ otherwise}\end{aligned}$$

3.2 目标函数

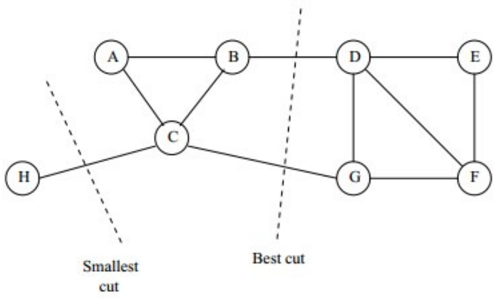
因此，如何切割图则成为问题的关键。换言之，如何切割才能得到最优的结果呢？
举个例子，如果用一张图片中的所有像素来组成一个图，并把（比如，颜色和位置上）相似的节点连接起来，边上的权值表示相似程度，现在要把图片分割为几个区域（要求是分割所得的 Cut 值最小，相当于那些**被切断的边的权值之和最小**，而权重比较大的边没有被切断。因为只有这样，才能让比较相似的点被保留在了同一个子图中，而大的点则被分割了开来。

设 $A_1, \ldots, A_k$ 为图的几个子集（它们没有交集），为了让分割的***Cut*** 值最小，谱聚类便是要最小化下述目标函数：

$$\text{cut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} W(A_i, \overline{A}_i)$$

其中k表示分成k个组，A_i 表示第i个组，$\overline{A}_i$表示A_i 的补集，$W(A_i, \overline{A}_i)$表示第A_i 组与第 $\overline{A}_i$组之间的所有边的权重之和（换言之，如果要分成K个组，那进行分割时去掉的边的权值的总和）。

为了让被切断边的权值之和最小，便是要让上述目标函数最小化。但很多时候，最小化***cut*** 通常会导致不好的分割。以分成2类为例，这个式子通常一个点和其余的n-1个点。如下图所示，很明显，最小化的smallest cut不是最好的cut，反而把{A、B、C、H}分为一边，{D、E、F、G}分为一边很可***cut***：



为了让每个类都有合理的大小，目标函数尽量让A1,A2...Ak 足够大。改进后的目标函数为：

$$\text{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|}$$

其中|A|表示A组中包含的顶点数目。

或：

$$\mathrm{Ncut}(A_1,\ldots,A_k) := \frac{1}{2}\sum_{i=1}^{k}\frac{W(A_i,\overline{A}_i)}{\mathrm{vol}(A_i)} = \sum_{i=1}^{k}\frac{\mathrm{cut}(A_i,\overline{A}_i)}{\mathrm{vol}(A_i)}$$

其中，$\mathrm{vol}(A) = \sum_{i\in A} w_{ij}$。

3.3 最小化*RatioCut* 与最小化$f'Lf$等价

下面，咱们来重点研究下***RatioCut***函数。

目标函数：$\min_{A\subset V} \mathrm{RatioCut}(A,\overline{A})$
定义向量$f = (f_1,\ldots,f_n)' \in \mathbb{R}^n$，且：

$$f_i = \begin{cases} \sqrt{|\overline{A}|/|A|} & \text{if } v_i \in A \\ -\sqrt{|A|/|\overline{A}|} & \text{if } v_i \in \overline{A} \end{cases}$$

根据之前得到的拉普拉斯矩阵矩阵的性质，已知

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

现在把f_i的定义式代入上式，我们将得到一个非常有趣的结论！推导过程如下：

$$
\begin{aligned}
f'Lf &= \frac{1}{2}\sum_{i,j=1}^{N} w_{ij}(f_i - f_j)^2 \\
&= \frac{1}{2}\sum_{i\in A, j\in \bar{A}} w_{ij}\left(\sqrt{\frac{|\bar{A}|}{|A|}}+\sqrt{\frac{|A|}{|\bar{A}|}}\right)^2 + \sum_{i\in \bar{A}, j\in A} w_{ij}\left(-\sqrt{\frac{|\bar{A}|}{|A|}}-\sqrt{\frac{|A|}{|\bar{A}|}}\right)^2 \\
&= \operatorname{cut}(A,\bar{A})\left(\frac{|\bar{A}|}{|A|}+\frac{|A|}{|\bar{A}|}+2\right) \\
&= \operatorname{cut}(A,\bar{A})\left(\frac{|A|+|\bar{A}|}{|A|}+\frac{|A|+|\bar{A}|}{|\bar{A}|}\right) \\
&= |V|\cdot \operatorname{RatioCut}(A,\bar{A})
\end{aligned}
$$

是的，我们竟然从 $f'Lf$ 推出了*RatioCut*，换句话说，拉普拉斯矩阵*L* 和我们要优化的目标函数*RatioCut* 有着密切的联系。更进一步说，因为$|V|$是一最小化*RatioCut*，等价于最小化 $f'Lf$ 。

同时，因单位向量$\mathbb{1}$的各个元素全为1，所以直接展开可得到约束条件：$\|f\|^2 = \sum f_i^2 = n$且$f'\mathbf{1} = \sum f_i = 0$，具体推导过程如下：

$$\sum_{i=1}^{n} f_i = \sum_{i\in A}\sqrt{\frac{|\bar{A}|}{|A|}} - \sum_{i\in \bar{A}}\sqrt{\frac{|A|}{|\bar{A}|}} = |A|\sqrt{\frac{|\bar{A}|}{|A|}} - |\bar{A}|\sqrt{\frac{|A|}{|\bar{A}|}} = 0$$

$$\|f\|^2 = \sum_{i=1}^{n} f_i^2 = |A|\frac{|\bar{A}|}{|A|} + |\bar{A}|\frac{|A|}{|\bar{A}|} = |\bar{A}| + |A| = n$$

最终我们**新的目标函数**可以由之前的 $\min_{A\subset V} \operatorname{RatioCut}(A,\bar{A})$ ，写成：

$$\min_{f\in\mathbb{R}^n} f'Lf \text{ subject to } f\perp \mathbb{1},\ \|f\| = \sqrt{n}$$

其中，$f_i = \begin{cases} \sqrt{|\bar{A}|/|A|} & \text{if } v_i \in A \\ -\sqrt{|A|/|\bar{A}|} & \text{if } v_i \in \bar{A} \end{cases}$，且因 $\|f\| = \sqrt{n}$ ，所以有：f'f = n（注：f是列向量的前提下，f'f是一个值，实数值，ff'是一个N*N的矩阵）。

继续推导前，再次提醒特征向量和特征值的定义：

- 若数字λ和非零向量$\vec{v}$满足$A\vec{v} = \lambda\vec{v}$，则$\vec{v}$为$A$的一个特征向量，$\lambda$是其对应的特征值。

假定$L\ f = \lambda\ f$，此刻，λ是特征值，f 是L 的特征向量。两边同时左乘f'，得到$f'Lf = \lambda f'f$，而f'f=n，其中n为图中顶点的数量之和，因此$f'Lf = $定值，所以要最小化 $f'Lf$ ，相当于就是要最小化λ。因此，接下来，我们只要找到L 的最小特征值λ及其对应的特征向量即可。

但到了这关键的最后一步，咱们却遇到了一个比较棘手的问题，即由之前得到的拉普拉斯矩阵的性质“L最小的特征值为零，并且对应的特征向量正知：其不满足$f\perp\mathbf{1}$的条件，因此，怎么办呢？根据论文“A Tutorial on Spectral Clustering”中所说的Rayleigh-Ritz 理论，我们可以取第2小的特征的特征向量v。

更进一步，由于实际中，特征向量v 里的元素是连续的任意实数，所以可以根据v 是大于0，还是小于0对应到离散情况下的$f = (f_1,\dots,f_n)' \in \mathbb{R}^n$，决定$f$ 是取$\sqrt{|\bar{A}|/|A|}$

所以，问题就转换成了：求拉普拉斯矩阵的前K个特征值，再对前K个特征值对应的特征向量进行 K-means 聚类。而两类的 登录 推 注册 题，即求特征值的，将对应的特征向量排列起来，再进行 K-means聚类。两类分类和多类分类的问题，如出一辙。

就这样，因为离散求解$f=(f_1,\dots,f_n)'\in\mathbb{R}^n$很困难，但*RatioCut*巧妙地把一个NP难度的问题转换成拉普拉斯矩阵特征值（向 问题，将离散的为连续的特征向量，最小的系列特征向量对应着图最优的系列划分方法。剩下的仅是将松弛化的问题再离散化，即将特征向量再 干，便可以得到相能不说妙哉！

3.4 谱聚类算法过程

综上可得谱聚类的算法过程如下：

1. 根据数据构造一个Graph，Graph的每一个节点对应一个数据点，将各个点连接起来（随后将那些已经被连接起来但并不 似的点，通过cut/RatioCut/NCut 的方式剪开），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 W。
2. 把W的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$N\times N$的对角矩阵，记为度矩 并把W - D的结斯矩阵$L=D-W$。
3. 求出L的前k个特征值（前k个指按照特征值的大小从小到大排序得到）$\{\lambda\}_{i=1}^k$，以及对应的特征向量$\{\mathbf{v}\}_{i=1}^k$。
4. 把这k个特征（列）向量排列在一起组成一个$N\times k$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。

或许你已经看出来，谱聚类的基本思想便是利用样本数据之间的相似矩阵（拉普拉斯矩阵）进行特征分解（ 通过Laplacian Eigenmap 的降维方式降将得到的特征向量进行 K-means聚类。

此外，谱聚类和传统的聚类方法（例如 K-means）相比，谱聚类只需要数据之间的相似度矩阵就可以了，而不必像K-means那样要求数据必须是 N中的向量。

4 参考文献与推荐阅读

1. 孟岩之理解矩阵系列：http://blog.csdn.net/myan/article/details/1865397；
2. 理解矩阵的12点数学笔记：http://www.51weixue.com/thread-476-1-1.html；
3. 一堆wikipedia，比如特征向量：https://zh.wikipedia.org/wiki/%E7%89%B9%E5%BE%81%E5%90%91%E9%87%8F；
4. wikipedia上关于拉普拉斯矩阵的介绍：http://en.wikipedia.org/wiki/Laplacian_matrix；
5. 邹博之聚类PPT：http://pan.baidu.com/s/1i3gOYJr；
6. 关于谱聚类的一篇非常不错的英文文献，“**A Tutorial on Spectral Clustering**”：http://engr.case.edu/ray_soumya/mlrg/Luxburg07_tutorial_spectral_clustering.pdf；
7. 知乎上关于矩阵和特征值的两个讨论：http://www.zhihu.com/question/21082351，http://www.zhihu.com/question/21874816；
8. 谱聚类：http://www.cnblogs.com/fengyan/archive/2012/06/21/2553999.html；
9. 谱聚类算法：http://www.cnblogs.com/sparkwen/p/3155850.html；
10. 漫谈 Clustering 系列：http://blog.pluskid.org/?page_id=78；
11. 《Mining of Massive Datasets》第10章：http://infolab.stanford.edu/~ullman/mmds/book.pdf；
12. Tydsh: Spectral Clustering：①http://blog.sina.com.cn/s/blog_53a8a4710100g2rt.html，②http://blog.sina.com.cn/s/blog_53a8a4710100g2rv.html，③http://blog.sina.com.cn/s/blog_53a8a4710100g2ry.html，④http://blog.sina.com.cn/s/blog_53a8a4710100g2rz.html；
13. H. Zha, C. Ding, M. Gu, X. He, and H.D. Simon. Spectral relaxation for K-means clustering. Advances in Neural Information Processing (NIPS 2001). pp. 1057-1064, Vancouver, Canada. Dec. 2001；
14. 机器学习中谱聚类方法的研究：http://lamda.nju.edu.cn/conf/MLA07/files/YuJ.pdf；
15. 谱聚类的算法实现：http://liuzhiqiangruc.iteye.com/blog/2117144。