Graph clustering algorithms utilize spectral techniques such as singular value decomposition, singular decomposition, etc., in order to properly partition (cluster) a graph into \( k \) pieces. Given an undirected graph \( G = (V, E) \), the goal is to partition the set of vertices \( V \) into \( k \) disjoint pieces \( \{V_i\}_{i=0}^k \) such that \( V_i \cap V_j = \emptyset \) and \( \bigcup V_i = V \).

Intuitively, a good clustering should result in clusters in which there are more edges connecting vertices within the same cluster versus outside of it. If \( G \) can be nicely partitioned, then it is worthwhile to look at a class of clustering algorithms, spectral clustering, in order to accomplish this.

Before implementing an algorithm on \( G \), we should make an assumption on how \( G \) was generated. Here, we introduce the Stochastic Block Model (SBM). This is a probabilistic or generative model represented as \( Pr(G|\theta) \) where we can estimate the parameters of \( \theta \) based on \( G \). This is beginning to look like a maximum likelihood problem, however, we will look at a spectral algorithm approach shortly. But first, let’s talk more about this generative model. Let’s consider the simplest partitioning \( k = 2 \) where \( V_1 \) and \( V_2 \) are equally sized partitions containing \( n \) vertices each.

We denote edge \( e_{i,j} \) as the edge connecting vertex \( i \) with vertex \( j \). We also denote \( \text{deg}(i) \) as the degree of vertex \( i \) or simply the number of vertices \( i \) is connected to. We can extend this notation to \( \text{deg}(i)_{V_p} \) in order to indicate the number of vertices \( i \) is connected to that also belong to partition \( V_p \). Now we can define the probabilistic existence of an edge in within the set of edges \( E \) by the following:

\[
Pr[e_{i,j} \in E] = \begin{cases} 
  p & \text{when } i, j \in V_1 \\
  p & \text{when } i, j \in V_2 \\
  q & \text{otherwise}
\end{cases}
\]

Let’s try to understand what kind of relationship \( p \) and \( q \) must satisfy for the recovery of partitions to be possible. It should be obvious that \( p > q \). We would also need the following conditions

1. \( p \geq a\log(n)/n \) (connectivity condition for \( V_1 \) and \( V_2 \))
2. \( \forall i \in V_1, \text{deg}(i)_{V_1} > \text{deg}(i)_{V_2} \)

Notice that \( \text{deg}(i)_{V_1} \) has an expectation of \( np \) and variance of \( np(1-p) \). Similarly, \( \text{deg}(i)_{V_2} \) has expected value \( nq \) and variance \( nq(1-q) \). In order to define a condition for exact recovery given \( p \) and \( q \), let’s say: \( \text{deg}(i)_{V_1} \approx np - \sqrt{np(1-p)} \) and \( \text{deg}(i)_{V_2} \approx nq + \sqrt{nq(1-q)} \). These conditions account for lower values of \( \text{deg}(i)_{V_1} \) by subtracting the standard deviation from the mean, while adding the standard deviation to the mean for \( \text{deg}(i)_{V_2} \). Thus, we’re tightening the margin while maintaining condition (2) from above. We can rearrange condition 2 given these approximation in order to achieve a sufficient condition for exact recovery:

\[
np - \sqrt{np(1-p)} > nq + \sqrt{nq(1-q)}
\]

\[
n(p-q) > \sqrt{n(\sqrt{p(1-p)} + \sqrt{q(1-q)})}
\]
As stated before, there are several ways to go about this problem such as MLE, semi-definite programming, spectral algorithms, etc. We will use spectral clustering methods. This involves a class of algorithms where we define an adjacency matrix \( A \in \mathbb{R}^{N \times N} \) based on our graph \( G \) where \( |V| = N \). Let the singular value decomposition of \( A \) be written as:

\[
A = U \Sigma V^T = \sum_{i=1}^{N} \sigma_i u_i v_i^T
\]

**Algorithm 1 Spectral Algorithm**

1. Construct \( A = U \Sigma V^T = \sum_{i=1}^{N} \sigma_i u_i v_i^T \), the adjacency matrix of \( G \sim SBM(p, q) \)
2. Obtain the singular values \( \sigma_i \) such that \( \sigma_1 \geq \sigma_2 \geq \sigma_3 \ldots \sigma_N \)
3. Obtain singular vectors \( v_1 \ldots v_k \)
4. Construct \( A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T \) using \( v_1 \ldots v_k \)
5. Run clustering algorithm: e.g. \( k\)-means

Taking the expectation value of our adjacency matrix \( E[A] \) gives us an 'ideal matrix' in the sense that instead of traditional ones and zeroes, our matrix entries are \( p \) and \( q \):

\[
E[A] = \begin{pmatrix}
p & \ldots & p & q & \ldots & q \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
p & \ldots & p & q & \ldots & q \\
q & \ldots & q & p & \ldots & p \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
q & \ldots & q & p & \ldots & p
\end{pmatrix}
\]

This can be more elaborately expressed as:

\[
E[A] = \frac{N}{2} (p + q) v_1 v_1^T + \frac{N}{2} (p - q) v_2 v_2^T,
\]

where \( v_1, v_2 \in \mathbb{R}^N \) and \( v_1 = [\frac{1}{\sqrt{N}} \ldots \frac{1}{\sqrt{N}}]^T \) and \( v_2 = [\frac{1}{\sqrt{N}} \ldots \frac{1}{\sqrt{N}}, -\frac{1}{\sqrt{N}} \ldots -\frac{1}{\sqrt{N}}]^T \). Notice that \( v_2 \) can be used to get an exact partition of the graph into two components. Our goal is to compute \( \hat{v}_2 \) which is the second highest singular vector of \( A \) in order to output partitions \( S_1 \) and \( S_2 \) from spectral clustering.

**Theorem 1.** If \( p - q > C \sqrt[3]{\frac{p \log n}{n}} \), then with probability at least \( 1 - \frac{1}{n^2} \) spectral clustering will output \( S_1 \) and \( S_2 \) such that:

\[
|S_1 \Delta V_1| + |S_2 \Delta V_2| \leq \frac{n}{\log n}.
\]
So we might get some values that belong to $V_2$ that are in $S_1$ and vice-versa, but it is fine. We can view $A$ as a slight perturbation of $E[A]$, which we can denote as $M$. For the vectors $v_1$ and $v_2$, which are the two singular singular vectors of $M$, we can find their respective singular values as:

$$\lambda_1 = \frac{N}{2}(p + q) \quad \lambda_2 = \frac{N}{2}(p - q).$$

Spectral clustering will succeed if the singular vectors of $M$ and $A$ are close to each other. The following theorem lets us argue about the closeness of the singular values.

**Theorem 2.** If there is an error matrix $R = A - M$ such that $R_{i,j} = A_{i,j} - M_{i,j}$, $E[R] = 0$, $|R_{i,j}| \leq 1$, and $\text{Var}(R_{i,j}) \leq \sigma^2$ where $\sigma^2$ is the variance of the highest entry in $R$. If $\sigma^2 \geq \frac{c \log n}{n}$ then with probability at least $1 - \frac{1}{n^3}$, $\sigma_1(R) \leq 10\sqrt{\frac{n}{\log n}}$.

Remember that since $M$ has rank 2, thus the other singular values besides the first two are simply zero. Thus we get that with high probability the singular values of $A$ are:

$$\sigma_1 = n(p + q) \pm \sqrt{np(1 - p)}$$

$$\sigma_2 = n(p - q) \pm \sqrt{np(1 - p)}$$

$$\sigma_3 = \pm \sqrt{np(1 - p)}$$

Recall, we wanted to know if $\hat{v}_2$ was good enough. The Davis-Kahan Theorem addresses this problem.

**Theorem 3.** Davis-Kahan Theorem

Let $M, M^o \in \mathbb{R}^{N \times N}$ where the singular vectors and values of $M$ are denoted as $v_1, v_2, \ldots v_N$ and $\sigma_1, \sigma_2, \ldots, \sigma_N$ and the singular vectors and values of $M^o$ are denoted as $w_1, w_2, \ldots w_N$ and $\lambda_1, \lambda_2, \ldots, \lambda_N$. If $M^o$ is the perturbed matrix, then:

$$||v_i - w_i|| \leq \frac{2\sigma_1(M - M^o)}{\min_{j \neq i} |\sigma_i - \sigma_j|}$$

By the Davis-Kahan theorem, we can observe the angular deviation between $v_2$ and $\hat{v}_2$:

$$||\hat{v}_2 - v_2|| \leq \frac{2\sigma_1(R)}{\min(np, n(p - q))}$$

$$\leq \frac{2\sqrt{np(1 - p)}}{n(p - q)}$$

and since, $p - q \geq \sqrt{\frac{p \log n}{n}}$

$$||\hat{v}_2 - v_2|| \leq \frac{1}{\sqrt{\log n}}$$

**Proof of Main Theorem.** If we misclassify $k$ points using the vector $\hat{v}_2$ then by Davis-Kahan we have that $||\hat{v}_2 - v_2|| \geq \sqrt{\frac{k}{n}}$. Hence we get that the number of misclassified points is $k \leq \frac{n}{\log n}$. \qed
1 Additional Reading

- A paper by Mcsherry that solve the problem in full generality. [http://www.cc.gatech.edu/~mihail/D.8802readings/mcsherrystoc01.pdf](http://www.cc.gatech.edu/~mihail/D.8802readings/mcsherrystoc01.pdf)

- An excellent tutorial on spectral clustering. [http://www.kyb.mpg.de/fileadmin/user_upload/files/publications/attachments/Luxburg07_tutorial_4488%5b0%5d.pdf](http://www.kyb.mpg.de/fileadmin/user_upload/files/publications/attachments/Luxburg07_tutorial_4488%5b0%5d.pdf)