Algebraic Connectivity and Fiedler Vector

Consider an undirected graph $\Gamma$ with $n$ vertices. Its adjacency matrix $A$ has the entries $A_{xy} = 1$ if $\{x, y\}$ is an edge in $\Gamma$, and $A_{xy} = 0$ otherwise. The matrix $D$ has the diagonal entries $D_{xx} = \sum_y A_{xy}$ for all vertices $x$ and vanishing off-diagonal elements $D_{xy} = 0$ if $x \neq y$. The graph Laplacian of $\Gamma$ is the matrix $L = D - A$, see [1].

We recall that the vector $\vec{1} = (1, \ldots, 1)$ is an eigenvector of $L$ with eigenvalue 0. Since $L$ is non-negative definite, 0 is the smallest eigenvalue of $L$. It is simple if and only if only $\Gamma$ is connected. The 2nd smallest eigenvalue of $L$ is the so-called algebraic connectivity of $\Gamma$. It is strictly positive whenever $\Gamma$ is connected. It is bounded from above by the vertex connectivity of $\Gamma$ [2]. The associated eigenvector $x_2$ is known as Fiedler vector. Its weak nodal domains, i.e., the subgraph induced by the vertices with non-negative entries in $x_2$ and the subgraph induced by the vertices with non-negative entries in $x_2$ are connected [3].

Hence $\alpha_2$ provides a good measure to determine whether $\Gamma$ is dense (when $\alpha_2$ is close to $n$, the maximally possible value), while the Fiedler vector can be used to partition graphs with small $\alpha_2$ into two subgraphs corresponding to the two nodal domains.

In order to get a numerically stable iteration for $\alpha_2$, we need to transform $L$ so that $\vec{1}$ is not the smallest but the largest eigenvalue also in absolute value. The largest eigenvalue of $L$ is not larger than $2 \max_x D_x x = 2\Delta$.

The auxiliary matrix

$$Q = (2\Delta + 1)I - L = (2\Delta + 1)I - D + A$$

therefore has the desired properties. Its largest eigenvalue is $2\Delta + 1$ with eigenvector $\vec{1}$ and its 2nd largest eigenvalue is $\lambda_2 = 2\Delta + 1 - \alpha_2$. To calculate $\lambda_2$, we start with an arbitrary vector $\vec{x}$. Its coordinates are taken from a random number generator. From the vector, we first construct $\hat{x}$ by subtracting the average from each coordinate.

$$\hat{x} = \vec{x} - \frac{1}{n} \sum_{i=1}^{n} x_i$$

Then we normalize $\hat{x}$ by dividing each coordinate by its length.

$$\hat{x}^* = (1/||\hat{x}||)\hat{x}$$

The resulting vector $\hat{x}^*$ has unit length and is orthogonal to $\vec{1}$. Now we compute $\vec{y} = Q\hat{x}^*$. Component-wise we have:

$$y_i = (Q\hat{x}^*)_i = (2\Delta - d_i)\hat{x}_i^* + (A\hat{x}^*)_i$$

$$y_i = (Q\hat{x}^*)_i = (2\Delta - d_i)\hat{x}_i^* + (A\hat{x}^*)_i$$

1
where the latter term is computed directly via summing over the entries of $\hat{x}^*$ indexed by the chained arrays which represent the adjacency list $l_i = [j_1, \ldots, j_{d_i}]$

$$(A\hat{x})_i = \sum_{j=1}^{n} A_{ij} \hat{x}_j^* = \sum_{k=1}^{d_i} \hat{x}_{j_k}^*$$

Thus, $A\hat{x}^*$ can be calculated in $O(M)$ time and space, where $M$ is the number of edges of $G$. Although $\vec{y}$ is theoretically already orthogonal to $\vec{1}$, we subtract the projection onto $\vec{1}$ in each iteration to stabilize the computation against round-off errors. Finally, we compute its length $\|\vec{y}\|$ which converges to $\lambda_2$. Hence, we record $\|\vec{y}\|$, $\hat{x}^*$ is replaced by $\frac{1}{\|\vec{y}\|}\vec{y}$ and the computation of $\|\vec{y}\|$ is repeated until $\|\vec{y}\|$ and $\hat{x}^*$ do not change more than some prescribed accuracy bound. The vector $\hat{x}^*$ then has converged to the Fiedler vector $x_2$.

Finally, we compute the algebraic connectivity as $\alpha_2 = 2\Delta + 1 - \lambda_2$ and normalized algebraic connectivity is set to $\alpha_2^* = \frac{\alpha_2}{n}$.

References

