Community Detection using Graph Efficiency

Maggie Miller ¹
maggie.miller@utexas.edu

Brendan Shah ²
bds5897@rit.edu

¹University of Texas at Austin

²Rochester Institute of Technology

January 10, 2013
In many applications of graph theory, particularly when working with large graphs, the ability to partition becomes increasingly useful.

Examples
- Decreasing graph size in order to distribute work over processors
- Communities in social networks
- Biological networks
- Information networks

Graph efficiency, proposed by V. Latora and M. Marchiori [Latora and Marchiori, 2001], measures how close a graph is to being complete.

We take the idea of graph efficiency and apply it to graph partitioning.
The efficiency matrix of a graph $E = (\epsilon_{ij})$

$$\epsilon_{ij} = \begin{cases} 
1/d_{ij} & \text{if } d_{ij} \neq 0 \\
0 & \text{if } d_{ij} = 0 \text{ or } v_i \text{ and } v_j \text{ are not connected.}
\end{cases}$$

where $d_{ij}$ is the shortest path length between vertices $v_i$ and $v_j$. 

$$E = \begin{pmatrix}
0 & \frac{1}{3} & \frac{1}{2} & 1 & \frac{1}{4} \\
\frac{1}{3} & 0 & \frac{1}{5} & \frac{1}{2} & 1 \\
\frac{1}{2} & \frac{1}{5} & 0 & \frac{1}{3} & \frac{1}{6} \\
1 & \frac{1}{2} & \frac{1}{3} & 0 & \frac{1}{3} \\
\frac{1}{4} & 1 & \frac{1}{6} & \frac{1}{3} & 0
\end{pmatrix}$$
Global Efficiency

We take from Latora [Latora and Marchiori, 2001] the definition of Global Efficiency

\[ E_{\text{glob}}(G) = \frac{1}{n(n - 1)} \sum_{i,j} \epsilon_{ij} \]
We continue by noting that "communities" presumably have higher efficiency than the graph as a whole.

We derive a new quantity called "partition efficiency".

Let $G$ be a graph with efficiency matrix $E$. Let $P$ be a partition that divides $G$ into two clusters $G_1$ and $G_2$ with $n_1$ and $n_2$ vertices respectively.

Let $E' = (e'_{ij})$ be the updated efficiency matrix, as some entries will change after edges are severed.
The “partition efficiency” of a graph is given by

\[
E_p = \frac{\sum_{v_i, v_j \in G_1} \epsilon'_{ij} + \sum_{v_i, v_j \in G_2} \epsilon'_{ij}}{\left(\sum_{v_i, v_j \in G_1} \max(E)\right) + \left(\sum_{v_i, v_j \in G_2} \max(E)\right)}
\]

\[
= \sum_{i,j} \epsilon'_{ij} \delta(g_i, g_j)
\]

\[
= \frac{\max(E)(n_1(n_1 - 1) + n_2(n_2 - 1))}{\max(E)}
\]

Where \(g_i\) is the cluster that contains vertex \(v_i\).

Note that this quantity falls in the range \([0, 1]\). If both \(G_1\) and \(G_2\) are complete, then \(E_p = 1\).
Example of Partition Efficiency

Figure: $E_{glob}(G) = 0.6741$, $E_{glob}(G_1) = 1$, $E_{glob}(G_2) = 1$, $E_p = 1$
Partition Efficiency

We define a vector \( s \), given by

\[
s_i = \begin{cases} 
1 & \text{if } v_i \in G_1 \\
-1 & \text{if } v_i \in G_2,
\end{cases}
\]

We also see that \( \delta(g_i, g_j) = \frac{1}{2}(s_is_j + 1) \). We can now write

\[
E_p = s^T E' s + \sum_{i,j} E'_{ij}
\]

\[
E_p = \frac{1}{2m} \sum_{i,j} E'_{ij}
\]

Where \( m = \max(E)(n_1(n_1 - 1) + n_2(n_2 - 1)) \). We name the term

\[
e_p = s^T E' s.
\]
Maximizing Partition Efficiency

- We want to maximize \( \frac{1}{2}(e_p + \sum_{i,j} \epsilon'_{ij}). \)
- The second term is approximately \( e_p + C \) for a constant \( C \), so we maximize \( e_p \).
- Assume that
  \[
  \sum_{i,j} \epsilon'_{ij} \delta(g_i, g_j) \approx \sum_{i,j} \epsilon_{ij} \delta(g_i, g_j).
  \]
- We can then say \( e_p \approx s^T Es \).
Not Updating the Efficiency Matrix

While deriving our algorithm, we made the assumption \( \sum_{i,j} \epsilon'_{ij} \approx \sum_{i,j} \epsilon_{ij} \).

Random graph on 1000 vertices:

\[
(E - E') \delta(g_i, g_j)
\]

We find

\[
\frac{\sum_{i,j} \epsilon'_{ij} \delta(g_i, g_j)}{\sum_{i,j} \epsilon_{ij} \delta(g_i, g_j)} = 0.9995.
\]
We can write $s = \sum_{i=1}^{n} a_i u_i$, where each $u_i$ is a normalized eigenvector of $E$ and $a_i = u_i^T s$.

$e_p$ can then be expressed as

$$s^T Es = \left( \sum_{i=1}^{n} a_i u_i^T \right) E \left( \sum_{j=1}^{n} a_j u_j \right)$$

$$= \left( \sum_{i=1}^{n} a_i u_i^T \right) \left( \sum_{j=1}^{n} a_j \lambda_j u_j \right) = \sum_{i=1}^{n} a_i^2 \lambda_i$$
The maximum of $s^T E s$ occurs when $s = 1$ because $E$ is non-negative.

This represents a trivial partition, so we use the eigenvector corresponding to the second largest eigenvalue.

Therefore, in order to maximize $s^T E s$, we choose to define $s = (s_i)$ by the following:

$$s_i = \begin{cases} 
1 & \text{if } u_{2i} \geq 0 \\
-1 & \text{if } u_{2i} < 0.
\end{cases}$$

Algorithm: bisect a network $G$. according to the signs of the components of the second eigenvector of $E$. 
Example Partitioning

Maximum Partition Efficiency - Example

\[ E_p = 0.3917. \]
Information Centrality

- Information Centrality is a graph partitioning algorithm that uses efficiency [Fortunato et al., 2004]. Given a simple undirected graph $G$, let $G_{ij}$ be the graph obtained by removing the edge connecting $v_i$ and $v_j$ in $G$. The information centrality of the edge connecting $v_i$ and $v_j$ is

$$C_{ij}^{\Delta} = \frac{E_{glob}(G) - E_{glob}(G_{ij})}{E_{glob}(G)}$$

[Latora and Marchiori, 2007]
The Information Centrality Algorithm

- Calculate $C_{ij}^A$ for every edge in $G$.
- Remove the edge with the highest centrality.
- Repeat until every vertex in $G$ is isolated. Keep track of modularity.
- Find clusters at maximum modularity.

Figure: [Fortunato et al., 2004]
Information Centrality vs. Partition Efficiency

A graph on thirty vertices with an artificially implanted partition.

Problems with the information centrality method:

- Very slow- $\mathcal{O}(k^3n)$, where $k$ is the number of edges and $n$ the number of vertices in $G$.
- Vertices are often isolated from clusters.
Maximum Modularity, a Graph Partitioning Algorithm

- Given a graph $G$ with adjacency matrix $A$ and with $n$ vertices and $m$ edges, suppose we make a random graph on $n$ vertices in which the vertex degrees correspond with those of $G$.
  - The probabilistic weight of the edge between $v_i$ and $v_j$ is
    \[ p_{ij} = \frac{\text{deg}(v_i)\text{deg}(v_j)}{2m}. \]
- $G$ has modularity matrix $B = A - P$ [Newman, 2006].
- We maximize $Q = \frac{1}{4m} s^T Bs$, where $m$ is the number of edges in $G$. 
### U.S. Senators, June 2012 Legislation Network

#### Maximum Partition Efficiency

<table>
<thead>
<tr>
<th></th>
<th>Value 1</th>
<th>Value 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partition Efficiency</td>
<td>0.2395</td>
<td>0.2378</td>
</tr>
<tr>
<td>Modularity</td>
<td>0.3796</td>
<td>0.3840</td>
</tr>
</tbody>
</table>

---

#### Maximum Modularity

- M. Miller and B. Shah (UTexas and RIT)
Although maximum modularity is faster, maximum partition efficiency is more sensitive to the connectivity of a graph.
To find more than 2 clusters, we recursively partition the graph until meeting a stopping criterion.

- **Criterion 1:** Given a desired minimum global efficiency, we partition a graph until each piece reaches the minimum global efficiency. Global efficiency threshold: 0.50

- **Criterion 2:** Recall $e_p = \sum_{i=1}^{n} \lambda_i (u_i \cdot s)^2$. If $\lambda_2 < 0$, then our method will actually lower the total efficiency. Therefore, a second stopping criterion that requires no additional input is the sign on $\lambda_2$. 

\[ \sum_{i=1}^{n} \lambda_i (u_i \cdot s)^2 \]
Recursive Partitioning: Dolphin Social Network

Maximum Partition Efficiency

Maximum Modularity

Partition Efficiency 0.4245 0.3823
Modularity 0.5349 0.4968

Graph from [Lusseau et al., 2003].
We have developed a new graph partitioning method that maximizes partition efficiency.

Future work:
- Use Johnson’s algorithm to make fast work of sparse matrices
- Use vector partitioning to find multiple clusters
- Consider directed networks

Research was supported by the National Science Foundation and the Department of Defense under the NSF-REU site award #1062128.

With thanks to our advisor, Dr. Anthony Harkin of Rochester Institute of Technology.
Method to find community structures based on information centrality.
*Physical Review E, 70*(056104).

Efficient behavior of small-world networks.

A measure of centrality based on network efficiency.
*New Journal of Physics, 9*(188).

The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations - can geographic isolation explain this unique trait?

Finding community structure in networks using the eigenvectors of matrices.
*Physical Review E, 74*(036104).
Modularity as a Measure

Modularity measures density, but not connectivity.

These two partitions have the same modularity, but the right-hand partition has a higher partition efficiency (as expected).