Permanence: A Vertex-Centric Approach to Evaluating Accuracy of Communities

Sanjukta Bhowmick
University of Nebraska at Omaha

Joint Work With: S. Srinivasan (UNO), T. Chakraborty, N. Ganguly, A. Mukherjee (IIT Kharagpur)
Community Detection

- Finding groups of vertices that are more tightly connected with each other than with other vertices in the network
- Vertices in the same community are generally similar
- Methods of community detection
  - Modularity (more connections than random)
  - Conductance (less connections between in a communities)
  - Random Walk (vertices most often visited are in a community)
Issues with Current Methods

- **Resolution Limit**
  - Communities below a certain size get merged into larger communities

- **Multiplicity of Solutions**
  - Same value can give different community distributions

- **Value of metric does not relate to quality of result**
  - Value depends on size of network rather than the community structure

- **Cannot identify bias in ground truth**
  - Finds communities even when there is no community structure
Criteria for Accuracy

- **Well-Posedness:** Is there sufficient good quality data for analysis?
  - Should not provide solution if data has low quality
  - Quality of results should be mapped to an universal scale for all inputs

- **Sensitivity:** Are changes in data reflected in the analysis results?
  - Small changes should have low impact on results
  - Large changes should have high impact on results

Current metrics neither satisfy well-posedness nor are very sensitive. We propose a new metric Permanence.
Soccer
Movies
Theatre
A
Total Internal connections > maximum external connections to any one of the external communities

Modularity, Conductance
consider total external connections
Theatre

A

Soccer

Theatre
Internal neighbors should be highly connected =>
high clustering coefficient among internal neighbors

Modularity, Conductance do not consider clustering coefficient
Permanence

\[ Perm(v) = \left[ \frac{I(v)}{E_{\text{max}}(v)} \times \frac{1}{D(v)} \right] - \left( 1 - C_{\text{in}}(v) \right) \]

\[ \begin{align*}
I(v) &= \text{internal deg of } v \\
D(v) &= \text{degree of } v \\
E_{\text{max}}(v) &= \text{Max connection to an external neighbor} \\
C_{\text{in}}(v) &= \text{clustering coefficient of internal neighbors}
\end{align*} \]

Permanence of entire network:

\[ Perm(v) = \frac{1}{N} \sum_{v=1:N} Perm(v) \]

\[ Perm(v) = 0.12 \]

Find community by maximizing permanence of the network

\[ I(v) = 4, \quad D(v) = 7, \quad E_{\text{max}}(v) = 2, \quad C_{\text{in}}(v) = 5/6 \]
Important Properties

- **Vertex-based.**
  - Computes the “belongingness” of a vertex in a community

- **Uniform Scale.**
  - Ranges from -1 (vertex placed in completely wrong community) to 1 (vertex in a clique).

- **Value of permanence relatively independent of the size of the network**
Sensitivity Under Perturbation

Permanence is robust to small changes and sensitive to large changes
Community Formation

- **Condition for Merging:**
  - If communities A and B are tightly connected and vertex v is tightly connected to both A and B, then higher permanence is obtained by merging A, B, and v.

- **Condition for Singleton:**
  - If v is loosely connected to its neighboring communities and has nearly equal number of connections to each, then higher permanence is obtained if v remains as singleton.

- **Reduces Multiplicity of Solutions**

- **Condition for Joining a Community:**
  - If v is tightly connected to a community and loosely connected to another community, higher permanence is obtained when v joins the community to which it is more connected.

- **Reduces Resolution Limit Problems**

Detailed proofs in [http://cse.iitkgp.ac.in/resgrp/cnerg/Files/resources/appendix_permance.pdf](http://cse.iitkgp.ac.in/resgrp/cnerg/Files/resources/appendix_permance.pdf)
Empirical Results

- **Algorithm for Maximizing Permanence**
  - Start with initial assignment of vertices to community
  - For each vertex, check if moving it to a different community increases the total permanence
  - If yes move it to the designated community
  - Repeat for all vertices
  - Repeat process until value of permanence stabilizes

- **Comparison with Other Algorithms**
  - Compare obtained community with ground truth communities
  - Using 6 Validation Metrics: NMI, ARI, Purity and Weighted Values of these
  - All values range from 0 (no match) to 1(perfect match)
  - Compare difference of values obtained from two methods. Higher is better
Test Suites with Ground Truth

- Real World Networks
  - Network of Inter-college Football
  - Network of Indian Railways
  - Network of Co-authorship in Technical Articles

- Synthetic Networks
  - LFR networks using different values of mixing parameter ($\mu$)
  - Lower value of $\mu$ indicates tighter community structure
## Experimental Results

<table>
<thead>
<tr>
<th>Method</th>
<th>LFR ($\mu=0.1$)</th>
<th>LFR ($\mu=0.3$)</th>
<th>LFR ($\mu=0.6$)</th>
<th>Football</th>
<th>Railway</th>
<th>Coauthors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Louvain</td>
<td>0.02</td>
<td>0.00</td>
<td>-0.75</td>
<td>0.02</td>
<td>0.14</td>
<td>0.00</td>
</tr>
<tr>
<td>FastGrdy</td>
<td>0.00</td>
<td>0.87</td>
<td>0.02</td>
<td>0.01</td>
<td>0.37</td>
<td>0.14</td>
</tr>
<tr>
<td>CNM</td>
<td>0.14</td>
<td>0.40</td>
<td>-0.13</td>
<td>0.30</td>
<td>0.20</td>
<td>0.05</td>
</tr>
<tr>
<td>WalkTrap</td>
<td>0.00</td>
<td>0.00</td>
<td>-0.50</td>
<td>0.02</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>Infomod</td>
<td>0.06</td>
<td>0.08</td>
<td>-0.20</td>
<td>0.01</td>
<td>0.19</td>
<td>-0.04</td>
</tr>
<tr>
<td>Infomap</td>
<td>0.00</td>
<td>0.00</td>
<td>-0.72</td>
<td>0.00</td>
<td>0.02</td>
<td>-0.02</td>
</tr>
</tbody>
</table>

Differences of our algorithm with the other algorithms averaged over all validation measures (High Value means Max Perm was more accurate)
LFR ($\mu = 0.1$) vs. LFR ($\mu = 0.6$)

For networks with bad community structure ground truth may be biased. Permanence can capture this.
Co-Authorship Results

- By inspecting the meta-data [keywords; subgroups] we find that permanence detects the sub-communities

- Main Communities as per Ground Truth
  - Algorithms and Theory;
  - Databases

- Communities obtained by maximizing permanence have these topics in each group
  - Theory of Computation, Formal Methods, Information and Coding Theory, Computational Geometry, Data Structure
  - Models, Query Optimization, Database Languages, Storage,
## Largest Community Size

<table>
<thead>
<tr>
<th></th>
<th>LFR ($\mu$=.3) Community Size</th>
<th>Football Community Size</th>
<th>LFR ($\mu$=.3) NMI</th>
<th>Football NMI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Truth</td>
<td>49</td>
<td>12</td>
<td>.70</td>
<td>.41</td>
</tr>
<tr>
<td>Louvain</td>
<td>62</td>
<td>24</td>
<td>.70</td>
<td>.41</td>
</tr>
<tr>
<td>Fast Greedy</td>
<td>95</td>
<td>18</td>
<td>.32</td>
<td>.65</td>
</tr>
<tr>
<td>CNM</td>
<td>91</td>
<td>32</td>
<td>.52</td>
<td>.31</td>
</tr>
<tr>
<td>WalkTrap</td>
<td>83</td>
<td>15</td>
<td>.51</td>
<td>.57</td>
</tr>
<tr>
<td>InfoMap</td>
<td>61</td>
<td>16</td>
<td>.79</td>
<td>.86</td>
</tr>
<tr>
<td>InfoMod</td>
<td>59</td>
<td>16</td>
<td>.74</td>
<td>.86</td>
</tr>
<tr>
<td>Permanence</td>
<td>49</td>
<td>13</td>
<td>1</td>
<td>.92</td>
</tr>
</tbody>
</table>

By Reducing Effect of Resolution Limit, Permanence gives the Closest Community to Ground Truth.
Stable Under Vertex Ordering

- As in any combinatorial optimization problem, the initial ordering of the vertex can change the results. Permanence is less affected by vertex ordering than other algorithms.

- **Constant communities**: Communities that do not change across vertex orderings.

- **Sensitivity**: Number of vertices/Number of constant communities (Low is Good).
Core-Periphery Structure of Communities

- Permanence can be used to identify whether the vertex is at the core or the periphery of its community.

- Farness Centrality: Mean shortest path of a vertex to all other vertices in its community.

- Higher permanence $\Rightarrow$ Low Farness Centrality $\Rightarrow$ Closer to the core

![Graphs showing the relationship between permanence and Farness centrality for different communities and datasets.](image-url)
Observations

- Permanence computes “belongingness” of individual vertices and provides an uniform scale for results
- Well-posedness: Reduces resolution limit. Finds closest solution to ground truth
- Detects spurious ground truth
- Sensitive to changes in data
- Can be helpful in understanding structure of communities


T. Chakraborty, S. Srinivasan, N. Ganguly, A. Mukherjee, S. Bhowmick, On the permanence of vertices in network communities. KDD 2014
Overlapping Communities

- Permanence can be generalized for overlapping communities
- A vertex can belong to multiple communities
- Permanence of vertex computed for each community

\[
Perm^C(v) = \left[ \frac{I^C(v)}{E_{\text{max}}(v)} \times \frac{1}{D(v)} \right] - (1 - C^C_{\text{in}}(v)) \times \frac{I^C(v)}{I(v)}
\]

- \(I^C(v)\) = fraction of internal deg of \(v\) in community \(C\)
- \(I(v)\) = Total internal degree of \(v\)
- \(D(v)\) = degree of \(v\)
- \(E_{\text{max}}(v)\) = Max connection to an external neighbor
- \(C^C_{\text{in}}(v)\) = clustering coefficient of internal neighbors in community \(C\)
Overlapping Permanence

\[ D(v) = 6, \quad I(v) = 5 \]

\[ P_{ov}^{C_1}(v) = \frac{1+1+\frac{1}{2}}{2 \times 8} - (1 - \frac{2}{3}) \times \frac{(1+1+\frac{1}{2})}{5} = -0.01 \]

\[ P_{ov}^{C_2}(v) = \frac{1+1+\frac{1}{2}}{2 \times 8} - (1 - \frac{1}{3}) \times \frac{(1+1+\frac{1}{2})}{5} = -0.18 \]

\[ P_{ov}(v) = P_{ov}^{C_1}(v) + P_{ov}^{C_2}(v) = -0.19 \]

Gives “belongingness” of vertex in each community

Value between -1 (bad allocation) to 1 (perfect allocation)

Is sensitive to changes in network

Reduces to non-overlapping permanence if there is one community
Sensitivity of Permanence

Scaled values

Perturbation intensity (p) in log scale
Resolution Limit

- Communities smaller than a certain size fall into the overlap rather than being a separate community.

- Overlapping permanence reduces the effect of resolution limit.

Dotted Lines: Overlapping Permanence
Solid Lines: BIGCLAM
Experimental Results

Permanence gives the best match with ground truth communities

Methods
O: OSLOM
C: COPRA
S: SLPA
M: MOSES
E: EAGLE
B: BIGCLAM
P: MAXOPERM

Accuracy improvement (in %) of MAXOPERM w. r. t. BIGCLAM on large real networks
Stable Under Vertex Ordering

![Graphs showing normalized sensitivity vs. number of different orderings of vertices (rescaled by 100) for LFR (μ=0.3) and Flickr datasets. The graphs compare different methods such as OSLOM, COPRA, MOSES, EAGLE, SLPA, MaxOPerm, and BIGCLAM.]
Core-Periphery Structure

LFR vs. Real-world networks

-\( \langle P^c_{ov} \rangle \)

Farness centrality, d

\( \mu = 0.1 \)
\( \mu = 0.3 \)
\( \mu = 0.6 \)

- LiveJournal
- Amazon
- Youtube
- Orkut
- Flickr
- Coauthorship