Thesis notes

1st June

The Echo Chamber Problem

Goal: given an interaction graph G, find $U \subseteq V$ maximing

$$\xi(U) = \sum_{C \in \hat{\mathcal{C}}} \sum_{T[U] \in S_C(U)} (|T^+[U]| - |T^-[U]|) \qquad (1)$$

where $|T^-[U]|$ and $|T^+[U]|$ denotes the number of negative and positive edges induced in the subgraph, respectively.

The set of users maximing the expression is denoted as $\hat{U}$ and the corresponding score is $\xi(G)$

The rounding algorithm

Algorithm 1: Rounding algorithm

$\hat{G} \leftarrow$ empty graph ;
$\hat{V} \leftarrow$ vertices of $\hat{G}$;
$S = 0$
foreach $e_{ij}^{k} \in \tilde{E}$ **do**
  $\hat{V} \leftarrow \hat{V} \bigcup \{v_i\}$ **if** $v_i \notin \hat{V}$;
  $\hat{V} \leftarrow \hat{V} \bigcup \{v_j\}$ **if** $v_j \notin \hat{V}$;
  $S \leftarrow \xi(\hat{V})$;
  foreach *component C in* $\hat{G}$ **do**
    $S \leftarrow \xi(C)$;
  end
end
return highest S ;

Example

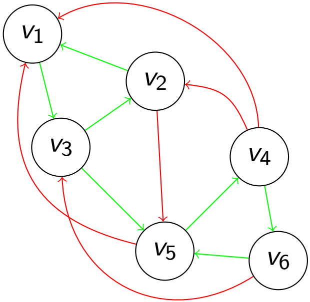


Figure: The original graph

Example

If many edges get the same value in the result of the relaxation, they are selected randomly

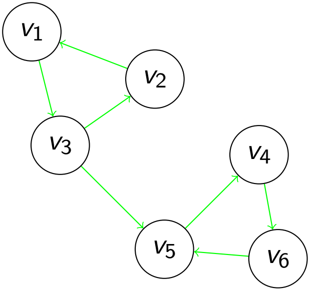


Figure: Only positive edges are reported, getting all value 0.666

Example

When the edge e_{35} is added then the 2 communities cannot be reconstructed.

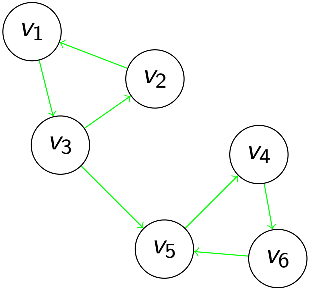
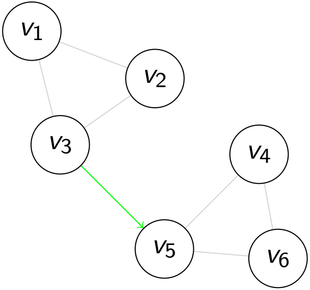


Figure: Only positive edges are reported, getting all value 0.666

Example

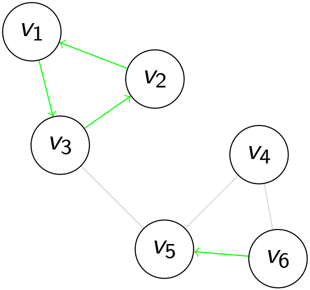
Let us suppose it is added in the first iteration. Then we have a situation as follows



We won't be able to find a whole community during the remaining iterations and the result may even be $U = \{v_3, v_5\}$.

Example

A "luckier" iteration would be the following



Effects on the results

Since the addition of positive cross-community edges degrades the result, we can generally improve the chances of finding better fits on the community

- Reducing the number of positive cross-community edges
- increasing the number of positive edges inside a community
- icreasing the number of threads in order to be less sensitive to single edges

A model for the Echo Chamber Problem

Each node has a group assignment and there are probabilities of positive and negative edges ω_{rs}^{+} and ω_{rs}^{+}, respectively.

1. Generate the *follow* graph G by using a SBM with parameters $\{\phi_{rs}\}$.
2. Each node can be active with probability β_a
3. Any active node activates his inactive neighbours in G with probability β_n
4. active nodes interact according to the categorical $(\omega_{rs}^{+}, \omega_{rs}^{-}, 1 - \omega_{rs}^{+} - \omega_{rs}^{-})$ otherwise (at least one of the 2 nodes is inactive) with categorical $(\theta\omega_{rs}^{+}, \theta\omega_{rs}^{-}, 1 - \theta(\omega_{rs}^{+} + \omega_{rs}^{-}))$, $\theta \leq 1$

A parametrized model (1)

Parameter choice:

$$\phi_{rs} = \begin{cases} 1 & \text{if } r = s \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Users follow other all and only users in the same community.

$\beta_a = 1$, $\beta_n = 1$: all users interact on each post.

$$\omega^{+}_{rs} = \begin{cases} 1 - x & \text{if } r = s \\ \frac{x}{4} & \text{otherwise} \end{cases} \quad \omega^{-}_{rs} = \begin{cases} x & \text{if } r = s \\ \frac{1-x}{4} & \text{otherwise} \end{cases} \tag{3}$$

This means that the probability of having an edge between two nodes in different communities is $1/4$.

Clustering results

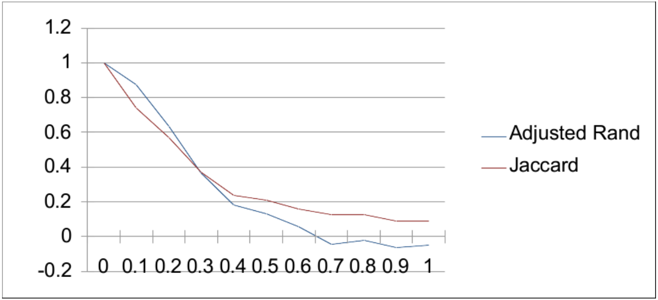


Figure: Approximation algorithm scores

Results obtained with 8 nodes per community and 12 threads.

More observations

The results clearly depends on factors like:

- the number of threads (the higher the number of threads the more the algorithm is robust to noise),
- the number of nodes (the higher the number of nodes the more the algorithm is robust to noise).