MKSE312 Final Report

Susan Greenberg and Ayaka Nonaka

May 6, 2013

1 Introduction

In this report, we analyze the Facebook New Orleans dataset (`http://socialnetworks.mpi-sws.org/data-wosn2009.html`) using different community detection techniques: modularity maximization using KL algorithm, spectral bipartition, and the Louvain-Twitter algorithm.

We extracted 5 disjoint subgraphs of 2,500 nodes from the dataset to make the computation more feasible. Figures 1 - 5 show the adjacency matrices of these subgraphs. The the `.mat` files for these subgraphs are available for download here: `http://db.tt/vd5pFgKP`

We were not able to compute the "flops" because it is obsolete in recent versions of MATLAB. This article explains more about why that is (near bottom of page): `http://www.mathworks.com/company/newsletters/news_notes/clevescorner/winter2000.cleve.html`

2 Modularity maximization using KL algorithm

Currently, our modularity maximization implementation is too slow to get any tangible results. We are not sure why, but it takes a few minutes to get through 1 iteration of just finding the node to move, and that needs to happen 2,500 times per iteration, which means it would take 42 hours just for one iteration of that loop. We tried looking for existing implementations of this algorithm that might be more efficient, but could not find any. We also tried running Newman's fast community finding algorithm, but that also took too long to run for even just one graph, so we did not have much luck with any modularity related routines, except for the Louvain-Twitter algorithm, which ran within minutes.

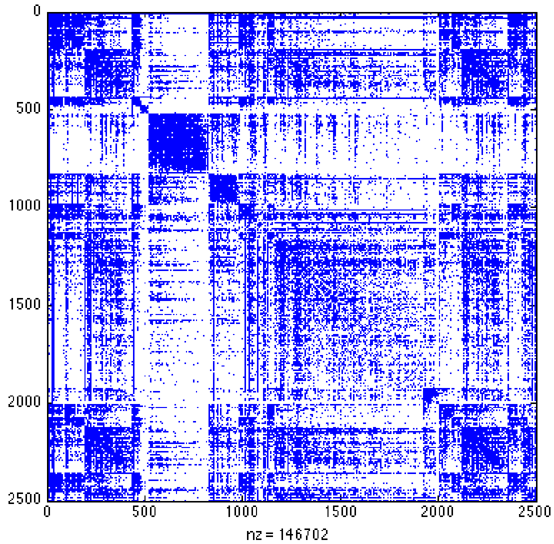

Figure 1: Adjacency matrix of subgraph 1

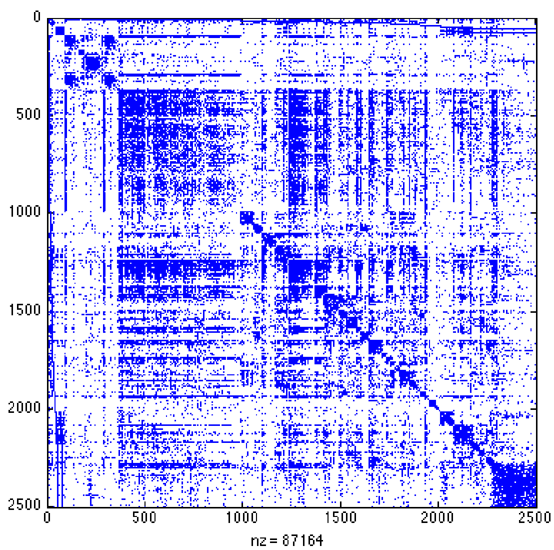

Figure 2: Adjacency matrix of subgraph 2

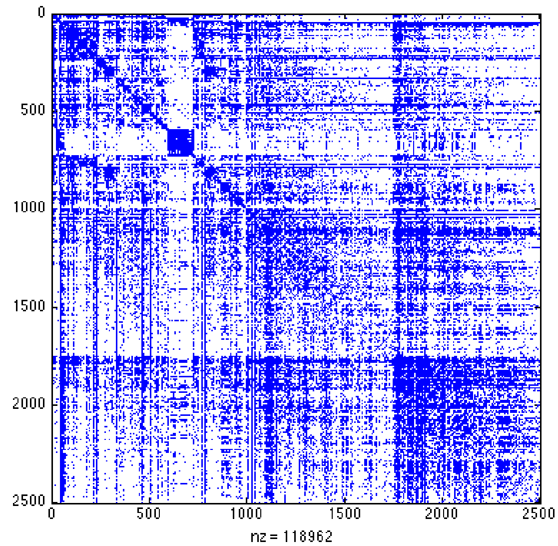

Figure 3: Adjacency matrix of subgraph 3

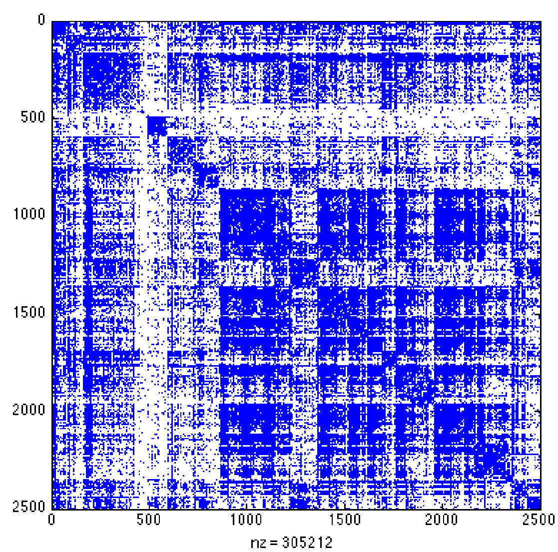

Figure 4: Adjacency matrix of subgraph 4

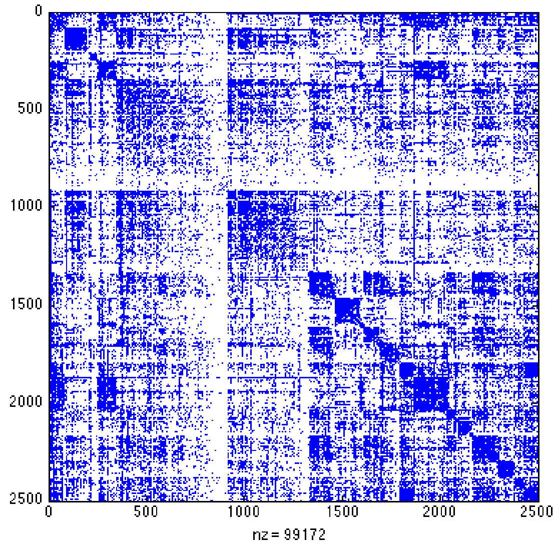

Figure 5: Adjacency matrix of subgraph 5

3 Spectral bipartition

Figures 6 - 10 show the results for spectral bipartition (using the method in the lecture notes) on the 5 subgraphs. The values for λ_2 for each subgraph respectively are: 0.8992, 0.5614, 0.9856, 0.9116, 0.9352. The λ_2 values tell us how easily the network can be split into two groups; the smaller the number, the easier to partition.

Subgraph 2 has the smallest λ_2 (0.5614), suggesting that it is the easiest to split, which seems to make sense since we can identify the two groups in the Figure 7.

On the other hand, subgraph 3 and 5 have the largest λ_2 (0.9856 and 0.9352 respectively), suggesting that they are the most difficult to split, which also seems to make sense because Figure 8 contains some nodes that are highly connected (the long lines that extend throughout the entire plot), and Figure 10 shows a cloud of plots that do not seem to have much obvious grouping.

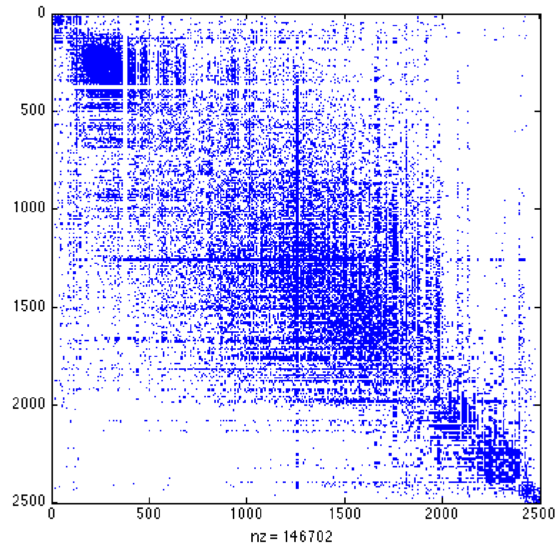

Figure 6: Spectral Bipartition for Subgraph 1

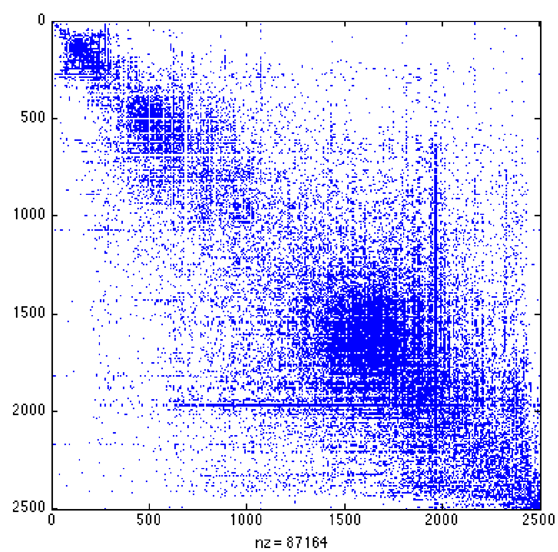

Figure 7: Spectral Bipartition for Subgraph 2

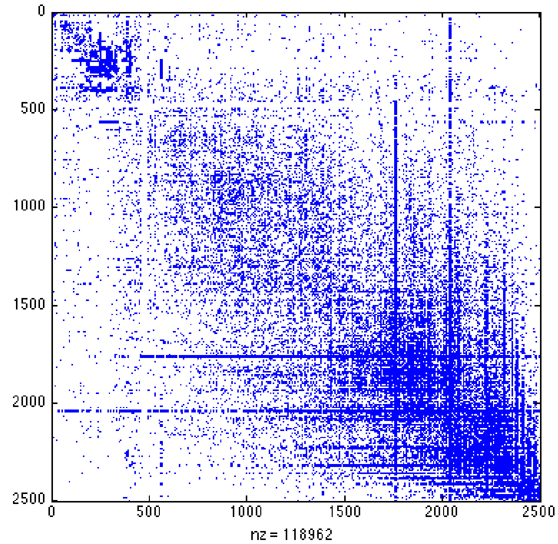

Figure 8: Spectral Bipartition for Subgraph 3

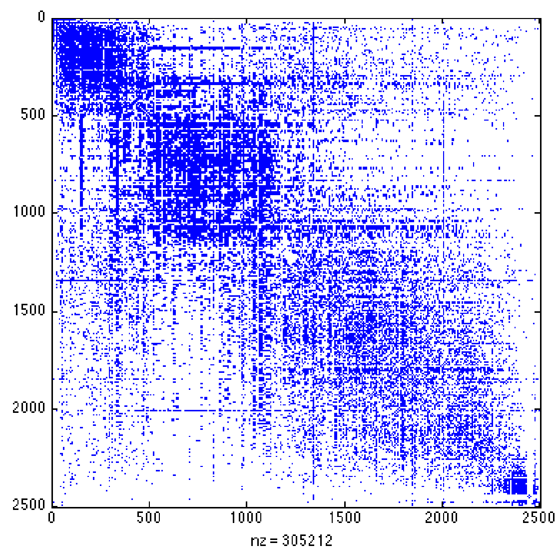

Figure 9: Spectral Bipartition for Subgraph 4

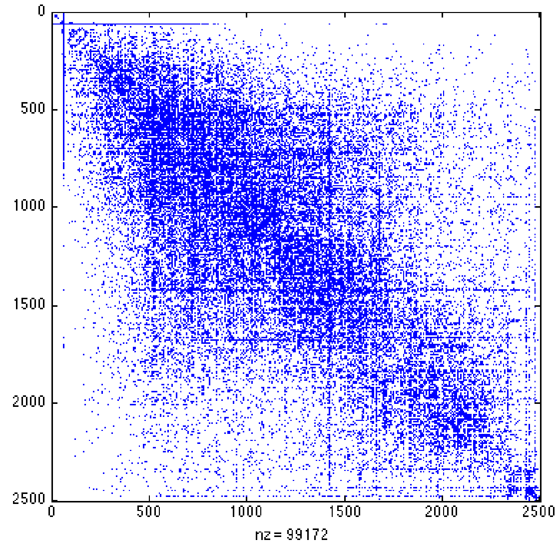

Figure 10: Spectral Bipartition for Subgraph 5

4 Louvain-Twitter algorithm

The Louvain-Twitter algorithm is a greedy optimization algorithm on modularity to detect communities in large networks. We used MATLAB implementation by MIT Strategic Engineering (`http://strategic.mit.edu/downloads.php?page=matlab_networks`). We ran the algorithm on the 5 subgraphs. The results are as follows:

Subgraph	Number of Modules	Modularity
1	1134	0.0133
2	971	0.0287
3	1059	0.0179
4	1231	0.0057
5	1256	0.0204

Subgraph 2 has the highest modularity score, and it also happened to have the smallest λ_2, so out of all the subgraphs, it seems to be the most obviously clustered graph. Subgraph 4 got the lowest modularity score, which might make sense because the network is the most densely connected, based on looking at its adjacency matrix.