CS235 ASSIGNMENT
Yidi Wang

862114701

Question 1
Part 1

```
clear; clc;
edgelist = dlmread('graph.txt');
edgelist = unique(edgelist, 'rows');
A = sparse(edgelist(:,1), edgelist(:,2), 1);
```

Part 2, 3
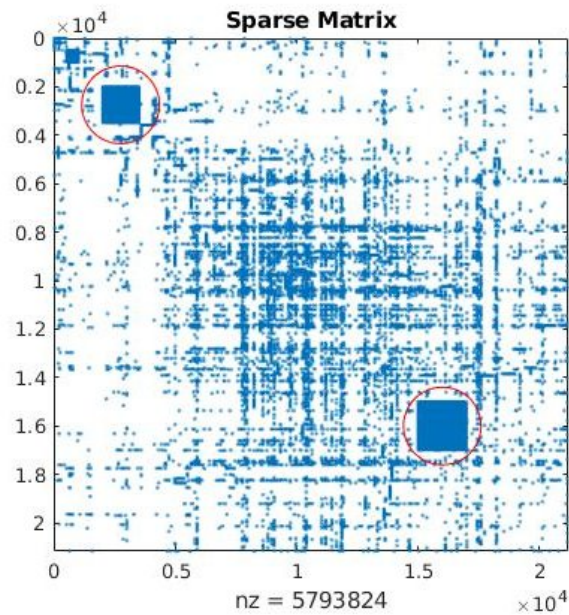There are mainly two dense blocks are annotated by red circles.

Figure 1: spy-plot

Question 2
Part 1
The number of connected nodes are calculated for each node in the edge-list of the graph.

```
max_node = max(edgelist(:));
deg_node = zeros(1, max_node);
for i = 1:max_node
    deg_node(i) = sum(rows(:) == i);
end
```
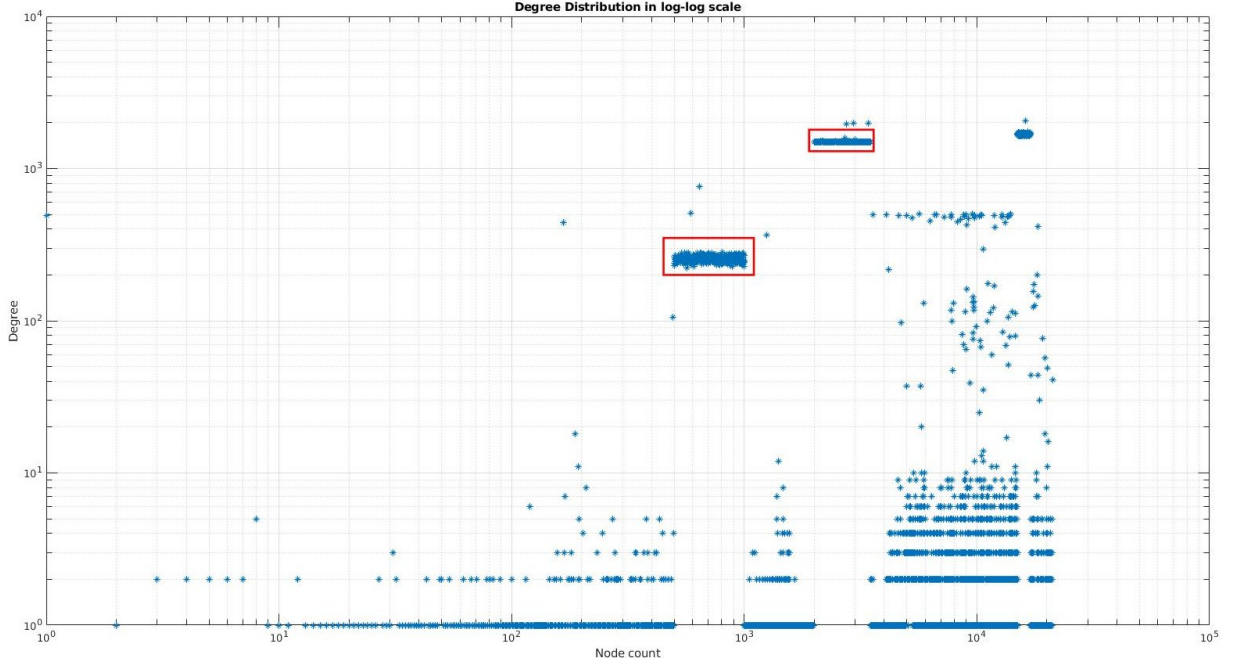
Part 2

Figure 2: Degree distribution in log-log scale

Part 3

Yes. In the clean degree distribution figure, the majority of the nodes have a small degree. While the degree becomes larger, the number of corresponding nodes become less and less. Degree and node counts are negtively correlated. However, in this distribution, node count is increasing while the degree increase. They are positively correlated.

Part 4

(a). There are mainly twoe abnormal blocks. They are annotated in red circles. Above the 10^2 on the y axis in the above figure (when degree is big enough), there are two (near-)horizontal lines, which means that there are many consecutive nodes having similar degrees. This will cause dense blocks in the sparse matrix.c

(b). The lower left thick line in Fig.2 is corresponding to the lowest right dense block in Fig.1. And the upper right thick line in Fig.2 is corresponding to the upper left block in Fig.1.

Question 3

Part 1

Main Problem

The goal of this paper is to extract community-like structures rather than graph-partitioning. The traditional methods, spectral clustering and multi-level graph partitioning can not yield good communities in the graph.

2

Main Idea

They use undirected graphs, which is represented as a adjacant matrix. The matrix is square and symmetric. In such a matrix, the eigenvalues are non-zero and the singular vectors coincide with the non-null eigenvectors. To use the features in their research, they expect to see communities in the form of (near-)cliques or (near-)bi-partite cores among the nodes in the spokes. From the EE-plot, they found that the linear nature of spokes means that the scores of the nodes are linear correcletd; therefore, exploring dominant nodes along one singular vector should be sufficient to extract nodes of a community.

Results

They use SpokEn to extract the communities in four Mobile Call graphs, which shows that SpokEn has successfully stopped at the termination criterion. Then they tested SpokEn with some real world graphs. Some typical results are given in figure 6. A bipartite from patent graph, a clique and a bipartite from provider-customers show that SpokEn works well.

Part 2

(a).

```
[u,s,v] = svds(A);
ds = diag(s);
dsp = norm(ds(n+1:end)) / norm(ds);
```

I tried the above code with different n. When $n = 2$, the error is 0.1157. When $n = 3$, the error is 0.0271. So, if we want a reconstruction correctness larger than 90%, the number of singular value is 3.

$$\begin{bmatrix} 1679.9 \\ 1501.0 \\ 255.22 \\ 37.715 \\ 35.5439 \end{bmatrix}$$
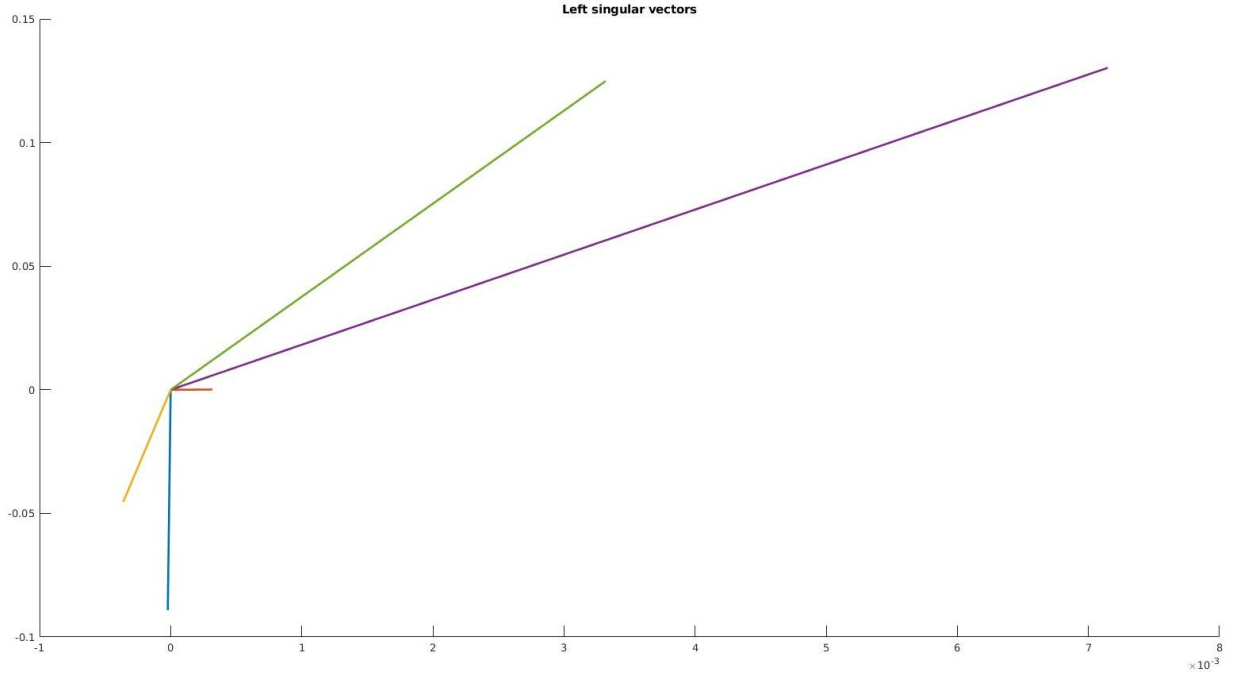
(b).

Figure 3: Left singular vectors

(c). Any abnormal block in the spy-plot is a (near-)clique or a (near-)bipartite in the graph. In the paper, they proved that when they increase the number of communities, the spoke pattern on the EE-plot will become more clear. In the abpve left singular value plot, we have two long spokes, which are corresponding to the two dense blocks in the spy-plot in 1.
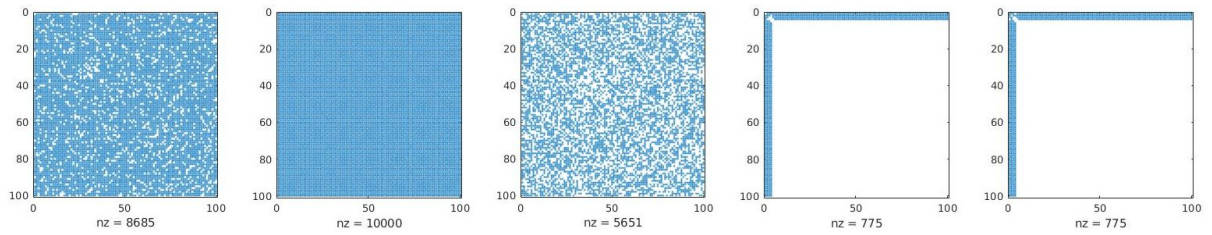
(d).

Figure 4: Left singular vectors