**README File**

**Numerical Studies:**

\*\*\*\*\* Note the following \*\*\*\*\*\*

1. Parallel computation is necessary for the subsampling-based methods. Using a large number of cores can be significantly beneficial in terms of time saving. However, each core needs to access enough memory (RAM) depending on subsample sizes.
2. In all numerical studies we demonstrate our advantage using time elapsed. However, it can vary based on different computing machines.
3. Since the testing method *without subsampling* can take a huge amount of time, we have been conservative in choosing several aspects such as the number of bootstrap resamples, the number of iterations and so on. The user can attempt to run the testing method *without subsampling* as long as they understand the time concerns.
4. Graph sizes and subsampling parameters for the simulation experiments are kept at low values in the codes than in the paper. The user can change them as required.

We keep (the rank of RDPG model) for all the simulations, as well as real data examples.

**Simulation Experiment for Estimation:**

Specify the number of nodes and the vector of subsampling parameter values and . The number of replications to account for randomness, as described in the paper, is denoted by . This experiment will give outputs of mean relative norm difference errors of the and estimation methods, as well as the variations and average times elapsed. This is how Tables 1 and 2 are generated in the paper.

**Simulation Experiment for Testing:**

Specify , and the number of bootstrap resamples used for testing. We calculate the level and power of the tests , and . The variable denotes the that has been used in the paper to indicate the difference between the null and the alternative. Changing the to different values gives us Table 3 presented in the paper. Note that the level corresponds to power at . Runtimes for these methods will also be generated as output.

**Real Data: FriendFeed Data**

We consider undirected, unweighted networks with no self-loops consisting of three layers: and . These networks are compared pairwise using bootstrap testing with subsampling and multiple testing. For convenience, only one pairwise comparison among the three is active, and others are commented out. Only one choice of subsampling parameters is used, and it can be replaced by other parameter choices. The testing without subsampling can be easily included using the appropriate functions (setting in ). This is how Table 4 is generated in the paper.

**Real Data: Sanremo Data**

Based on the Sanremo Network data, choose only nodes that have high weight index and consider undirected, unweighted networks with no self-loops consisting of three layers: and . These networks are compared using bootstrap testing with subsampling and multiple testing. For convenience, only one pairwise comparison among the three is active, and others are commented out. Only one choice of subsampling parameters is used, and it can be replaced by other parameter choices. The testing without subsampling can be easily included using the appropriate functions (setting in ). This is how Table 4 is generated in the paper.

In the following, we list all the functions that we have used for the numerical studies.

**Functions:**

* + Objective: Generates a graph from a probability matrix.
  + Input:
    1. is the probability matrix from which the graph is generated.
  + Output:
    1. An adjacency matrix of the same size as .
  + Objective: Computes the Adjacency Spectral Embedding (ASE) on a fixed dimension.
  + Input:
    1. is an adjacency matrix.
    2. is the fixed dimension for ASE.
  + Output:
    1. The ASE of matrix at dimension .
  + Objective: Finds the Procrustes rotation between and .
  + Input:
    1. Two matrices and of the same dimension.
  + Output:
    1. The minimum Frobenius norm distance between and .
    2. The orthogonal rotation between and .
  + Objective: Computes the probability matrix from a latent matrix.
  + Input:
    1. Latent matrix .
  + Output:
    1. Probability matrix.
  + Objective: Finds the ASE based estimated latent matrix on a fixed dimension with subsampling.
  + Input:
    1. is an adjacency matrix.
    2. is the overlap region size.
    3. is the number of subsamples.
    4. is the fixed dimension for ASE.
    5. is the method for choosing the overlap region. Default is . Another choice is .
  + Output:
    1. The ASE based estimated latent matrix of matrix at dimension .
  + Note: This function makes use of parallel computation. It is recommended to define appropriate parallel clusters with enough cores to get better results.
  + Objective: Finds the estimated latent matrix of based on different methods.
  + Input:
    1. is an adjacency matrix.
    2. is the overlap region size.
    3. is the number of subsamples.
    4. is the fixed dimension for ASE.
    5. is the method for performing estimation. Default is (subsampling). Another choice is .
  + Output:
    1. The estimated latent matrix of matrix at dimension .
  + Note: This is a function combining and , created for convenience in later simulation experiments. The arguments are unnecessary when using , and they can be specified arbitrarily.
  + Objective: Performs bootstrap based testing between two adjacency matrices of two networks.
  + Input:
    1. is an adjacency matrix.
    2. is another adjacency matrix.
    3. is the overlap region size.
    4. is the number of subsamples.
    5. is the fixed dimension for ASE.
    6. is the number of Bootstrap resamples.
    7. is the method for performing estimation. Default is (subsampling). Another choice is .
  + Output: A list containing:
    1. is the p-value for the test.
    2. is the time taken for the method in performing the bootstrap test.
  + Note: To perform the full network bootstrap testing method (ASE), we do not need the values , and they can be specified arbitrarily.
  + Objective: Performs multiple testing between two adjacency matrices of two networks.
  + Input:
    1. is an adjacency matrix.
    2. is another adjacency matrix.
    3. is the overlap region size.
    4. is the number of subsamples.
    5. is the fixed dimension for ASE.
    6. is the number of Bootstrap resamples.
  + Output: A list containing:
    1. is the p-value for the test.
    2. is the time taken for the method in performing the multiple testing.
  + Note: For each of the tests on the subsamples in the multiple testing procedure, ASE based full network estimation is used.