

# Final Report: Graph Two-sample Testing with Node Embeddings

Qiucheng Wu, Yuze Lou, Shucheng Zhong and Jiaxin Wang  
wuqiuche@umich.edu, yuzelou@umich.edu, joezhong@umich.edu, jiaxwang@umich.edu  
University of Michigan  
Ann Arbor, Michigan

## KEYWORDS

Node embedding, two-sample testing, data mining

## 1 ABSTRACT

Graph two-sample testing has been an important tool in applied research fields for decades. However, statistical methods of hypothesis testing to graph structured data are still very challenging as it is hard to figure out what to take to reject the null hypothesis that two graphs come from the same distribution or not. Recent studies in statistics and learning theory use graph explicit properties such as degree distribution and triangle counts to distinguish whether graphs are from the same or different populations, but the practicality of the developed theoretical methods remains an open question.

In our project, we propose a novel method to solve graph two-sample testing problem by combining node embedding methods and statistical two-sample test methods. We have evaluated our method with multiple node embedding methods and conduct experiments on both synthetic datasets and real datasets. We look closely on the performance of each node embedding methods and compare the results with the prior work to demonstrate the effectiveness of our method.

## 2 INTRODUCTION

In many tasks, it is important to understand structural distribution information in graph structured datasets. Sometimes we are interested in whether two populations of graphs come from the same distribution/graph model or not. For example, given two brain graphs, since the data are collected from different patients, the graphs are different from each other. To determine whether the brains are healthy or not, it is helpful to know if the differences are statistically significant. Therefore, we need graph two-sample testing to determine if the graphs are from the same distribution.

The definition of graph two-sample testing is described as follows. Let  $V$  be a set of  $n$  vertices. Let  $G_1, \dots, G_m$  and  $H_1, \dots, H_m$  be two populations of undirected unweighted graphs defined on the common vertex set  $V$ , where each population consists of independent and identically distributed samples. Graph two-sampling test is to verify whether  $(G_i)_{i=1, \dots, m}$  and  $(H_i)_{i=1, \dots, m}$  are generated from the same random model or not [5].

The prior work on graph two-sample testing mainly utilizes the explicit properties of graphs such as adjacency matrix, node distribution and triangle counts [3][4]. Adjacency spectral embedding(ASE) test [11] is one of those methods that is widely used to solve graph two-sample testing. Given the adjacency matrix  $A_G$  of graph  $G$ , adjacency spectral embedding is defined by

$$X_G = U_G \Sigma_G^{1/2}, \quad (1)$$

where  $\Sigma_G \in \mathbb{R}^{r \times r}$  is a diagonal matrix containing  $r$  largest singular values of  $A_G$  and  $U_G \in \mathbb{R}^{n \times r}$  is the matrix of corresponding left singular vectors. Then the test statistic proposed by adjacency spectral embedding test of two graph  $G$  and  $H$  is

$$T_{ASE} = \min\{\|X_G - X_H W\|_F : W \in \mathbb{R}^{r \times r}, W W^T = I\}, \quad (2)$$

where rank  $r$  is assumed to be known. The rotation matrix  $W$  aligns the adjacency spectral embeddings of the two graphs.

Alternatively, instead of focusing on simple graph features, we want to take advantage of the power of node embedding that can represent rich graph information to improve graph two-sample testing. Let  $G = (V, E)$  be a given network. Node embedding is a function  $f : V \rightarrow \mathbb{R}^d$  that maps nodes to feature representations, where  $d$  is a parameter specifying the number of dimensions of our feature presentation [6]. The structural equivalence information represented by node embedding is likely to help improve the accuracy of two-sample test. Limited prior work has been done on graph two-sampling test using node embedding, so it is worth studying and conducting extensive experiments on different embedding methods.

## 3 PROPOSED METHOD

Our approach is to try combinations of different node embedding methods with different test methods to see if they can improve graph two-sample test. We used the node embedding methods to generate vector representations for nodes and applied them to graph two-sample testing methods. Therefore, the method is divided into two parts: embedding and testing.

### 3.1 Embedding

For node embedding methods, we tried methods that focus on node representation and proximity similarity (node2vec [6]) as well as those focusing on structural similarity (struc2vec [10], xNetMF [7] and GraphWave [2]). The embedding methods focus on these different features and assign a vector on each nodes. Therefore, the embedding methods numerically describes features in a graph.

It is worth noting that in our previous report, we apply the embedding methods directly on given edge-list for each graph, which focuses on their local features only and is subject to the noise issue introduced by different graphs, leading to a poor result. With the suggestions from instructors, we concatenate the edge-lists for two graphs and jointly embedding them, reaching a much better result. We will introduce this in details in experiment section.

### 3.2 Testing

We first fixed the test method to be unbiased MMD, and investigated how well each embedding method worked with unbiased MMD. Then we picked embedding methods that work well in the previous

step, and applied it to ASE test to further study which test method performs the best with node embedding.

MMD is a distance measure between two distributions which is defined as the squared distance between their embeddings in the reproducing kernel Hilbert space. Unbiased MMD is an unbiased empirical estimation of the value of MMD. Let  $\mathcal{F}$  be a class of functions  $f : \mathcal{X} \rightarrow \mathbb{R}$ . Let  $p, q$  be two distributions and  $x, y$  are the samples from distribution  $p, q$ . Then MMD is defined as:

$$MMD[\mathcal{F}, p, q] := \sup_{f \in \mathcal{F}} \lim (\mathbf{E}_x[f(y)] - \mathbf{E}_y[f(y)])$$

A unit ball in a reproducing kernel Hilbert space  $\mathcal{H}$  is used as the MMD function class  $\mathcal{F}$ . By the Riesz representer theorem, there exists a feature mapping  $\phi(x) : \mathcal{X} \rightarrow \mathbb{R}$  such that  $f(x) = \langle f, \phi(x) \rangle_{\mathcal{H}}$ .  $\phi(x)$  takes the canonical form  $k(x, \cdot)$ , where  $k(x_1, x_2) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  is positive definite, and the notation  $k(x, \cdot)$  indicates the kernel has one argument fixed at  $x$ , and the second free. Then the unbiased empirical estimate of squared population MMD can be written as:

$$MMD^2[\mathcal{F}, p, q] = \mathbf{E}_{x, x'}[k(x, x')] - 2\mathbf{E}_{x, y}[k(x, y)] + \mathbf{E}_{y, y'}[k(y, y')]$$

Let  $0 \leq k(\cdot, \cdot) \leq K$  and  $m$  is the sample size of  $x$  and  $y$ . A hypothesis test of type-I error level  $\alpha$  for the null hypothesis  $H_0 : p = q$  has the acceptance region

$$MMD_u^2 < (4K/\sqrt{m})\sqrt{\log(\alpha^{-1})},$$

where  $\alpha$  is the upper bound probability that  $H_0$  is rejected while actually being true.

We have introduced the classical ASE[5] test in the introduction part. In short, ASE test uses adjacency matrix decomposition and matrix multiplication to generate node embeddings. Then, the ASE test adopts the idea of using a rotation matrix  $W$  to align two embeddings. This is particularly useful when the embedding does not keep rotation invariant, which is not guaranteed by most of the embedding methods. Thus, we believe this is a good baseline method that can be used in evaluating performances of our methods.

As a conclusion, for graph two-sample test, we applied the representation vectors as input to unbiased MMD, biased fast MMD, and MMD linear respectively. We evaluated the node embedding methods and the extent of improvement in graph two-sample test by applying those methods.

## 4 EXPERIMENTS

### 4.1 Data

**Synthetic datasets:** As described in the proposal report, we used inhomogenous Erdős–Rényi model proposed by Bollobas [1] to generate random graphs and stochastic block model to synthesize graphs that contain community information. To simulate the graphs in real world, stochastic kronecker graph [9] is also used in our experiments.

- *ER*: Generated by Erdős–Rényi model with batch size of 10. Each containing 500 nodes with edge probability of 0.05.
- *SBM*: Generated by stochastic block model with batch size of 10. Containing 3 blocks with 75, 75, and 350 nodes respectively. The internal edge probability is 0.75, and edges between nodes in different blocks have probability of 0.05.

- *Kronecker*: Generated by stochastic kronecker model with batch size of 10. The initiator matrix is  $[0.98, 0.58; 0.58, 0.06]$  and each contains 512 nodes.

**Real datasets:** Previous research of graph two-sampling test mainly focused on synthetic datasets, so we also applied our methods to real datasets.

- *Arxiv GR-QC* [8]: Collaboration network from the e-print arXiv covering scientific collaborations between authors papers submitted to General Relativity and Quantum Cosmology category. Containing 5,242 nodes and 14,496 edges.
- *Arxiv ASTRO-PH* [8]: Collaboration network from the e-print arXiv covering scientific collaborations between authors papers submitted to Astro Physics category. Containing 18,772 nodes and 198,110 edges.

### 4.2 Experiment setup

We have run 3 experiments to test the performance of our proposed methods.

- Experiment 1 tests various combinations of different hyperparameters, node embedding methods and testing methods.
- Experiment 2 compares the performance of our proposed method with the baseline, adjacency spectral embedding testing.
- Experiment 3 tests the robustness of our method using real datasets with random noise involved.

### 4.3 Embedding method and MMD parameter selection

The embedding methods we used are:

- (1) node2vec
- (2) struc2vec
- (3) xNetMF
- (4) GraphWave

And the testing methods we used are

- (1) MMD
- (2) FastMMD-FastFood
- (3) MMD-linear

In terms of the embedding dimension, we have tried several options from dimension 2 to dimension 128, and finally set our embedding dimension to be 2 since it makes our method much faster without losing too much accuracy.

As mentioned in the method section, to ensure the embedding methods are capturing comparable features of nodes from two graphs, in this final report we concatenate the edge-list of two graphs together. To be specific, given two graphs  $G$  and  $H$  with number of nodes  $|N_G|$ ,  $|N_H|$  and number of edges  $|E_G|$ ,  $|E_H|$ , we will put the graphs together to get one graph with number of nodes  $|N_G| + |N_H|$  and number of edges  $|E_G| + |E_H|$ . This graph is a two-component graph; two original graphs are separated with each other, i.e. have no edges connecting them. Therefore, the concatenation preserves order of nodes, and after embedding, we can recover the graphs by the index of nodes.

Then, we apply the embedding methods on preprocessed edge-lists mentioned above, and we use the unbiased MMD test to see the effectiveness of this testing method with different parameters,

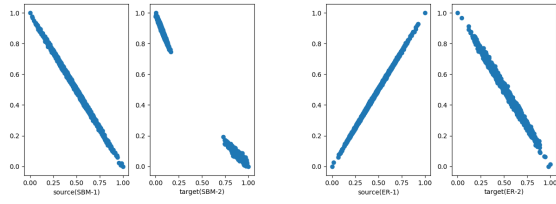


Figure 1: Separate embedding results for node2vec

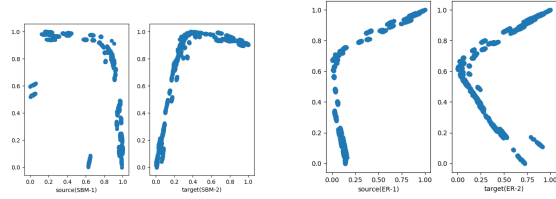


Figure 2: Separate embedding results for struc2vec

namely kernel bandwidth  $\sigma$ . We conducted the experiments on the synthetic datasets we generated. The group of embedding results are

- (1) ER vs ER
- (2) ER vs SBM
- (3) ER vs Kronecker
- (4) SBM vs SBM
- (5) SBM vs Kronecker
- (6) Kronecker vs Kronecker

For the unbiased MMD test, we applied radial basis function(RBF) as our kernel function and took the kernel bandwidth  $\sigma$  ranging from  $10^{-2}$  to  $10^2$ . The upper bound of type-I error probability  $\alpha$  in the test is 0.05. Finally, we set the sample size to 500 and repeated the experiment 10 times to take the average of the results as our MMD scores.

We applied those pairs above to MMD, obtaining MMD scores. By comparing the scores of MMD, we could gain insight on which embedding method and bandwidth  $\sigma$  works best on current datasets.

**4.3.1 Embedding results.** In midterm report, we do the embedding directly on graphs. Figure 1 and Figure 2 give examples of those embeddings. In each of them, all images come from different distributions.

Now we do jointly embeddings on pairs of graphs that we want to compare. Figure 3 and Figure 4 give examples of those embeddings. In each of them, left two images come from same distribution, and right two images come from different distributions.

We will see if separate embedding or joint embedding does better when we feed the embedding results to the MMD test method.

**4.3.2 MMD results and parameter selection.** After we obtain the embedding results, we can feed these embedding vectors to MMD with different kernel bandwidth  $\sigma$  to see which parameter setting can tell the differences between graphs from same distribution and graphs from different distributions. If the method provides a high

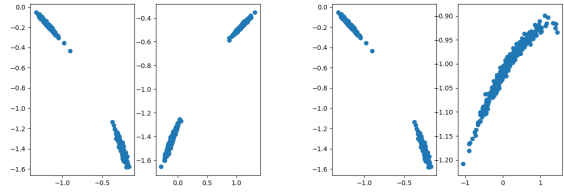


Figure 3: Joint embedding results for node2vec

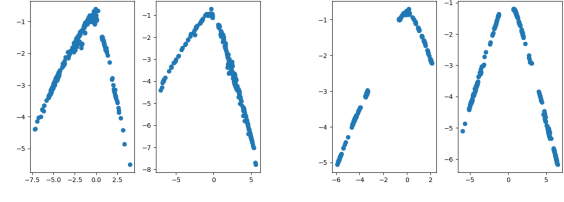


Figure 4: Joint embedding results for struc2vec

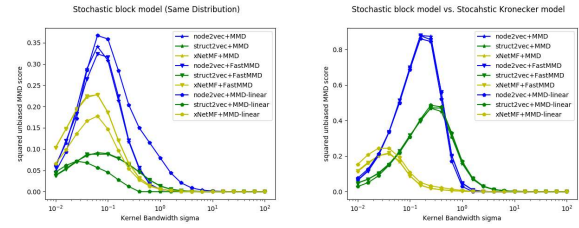


Figure 5: Joint embedding results for node2vec

score on different distributions and a low score on same distribution, then we consider this method to be effective.

We first try the separate embedding methods. The result is shown in Figure 5.

Figure 5-7 describe performances of MMD with different  $\sigma$  on graphs from different distributions.

Figure 8-10 describe performances of MMD with different  $\sigma$  on graphs from same distribution.

Based on results from Figure 5-7, it seems xNetMF, node2vec and struc2vec all provide a high score at different regions of kernel bandwidths, which is our findings in the midterm report. However, after considering results from Figure 8-10, it turns out node2vec usually provides a high score regardless of where the graphs are from. Therefore, node2vec is not a good embedding method on this graph two-sample test task. On the contrary, xNetMF and struc2vec have significant differences between two cases, indicating that they are probably good candidate methods for our task.

Meanwhile, we consider the performance of MMD and its variants. Based on our results, while the fast-MMD seems perform a little bit better, these three test are hard to distinguish. Therefore we can apply any of them in this task.

Finally, we analyze on good bandwidth setting for MMD. Based on the images, xNetMF works when MMD bandwidth  $\sigma$  satisfies  $10^{-2} < \sigma < 1$ , while it works best when  $10^{-2} < \sigma < 0.3$ . The

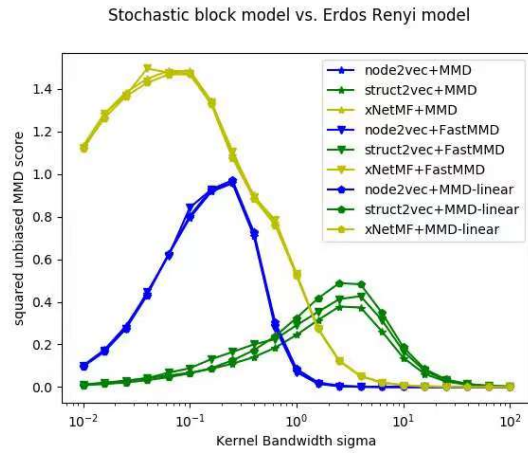


Figure 6: MMD scores comparing SBM with ER

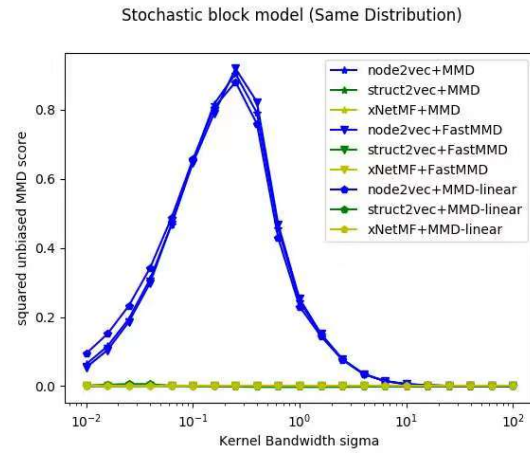


Figure 9: MMD scores comparing graphs both from SBM

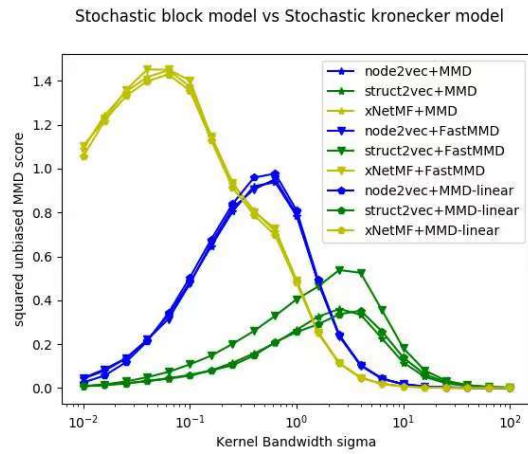


Figure 7: MMD scores comparing SBM with kronecker

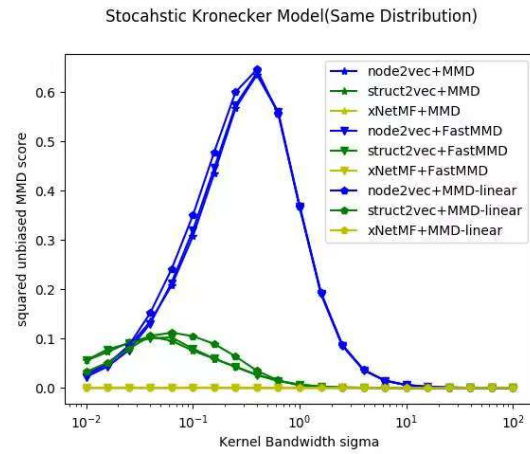


Figure 10: MMD scores comparing graphs both from kronecker

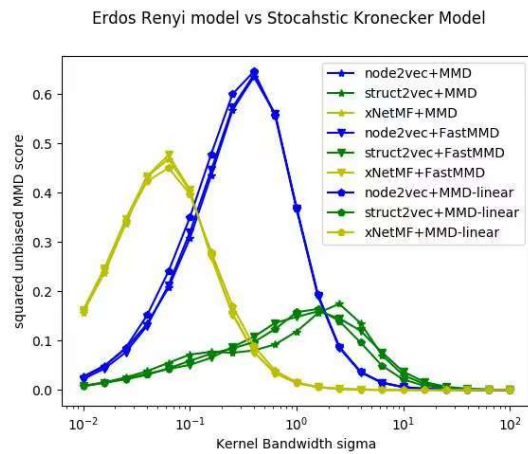


Figure 8: MMD scores comparing ER with kronecker

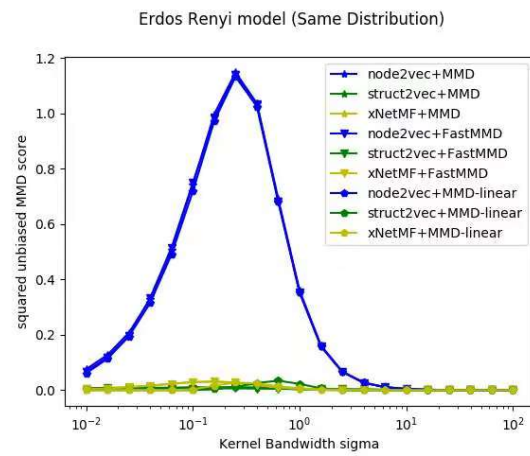


Figure 11: MMD scores comparing graphs both from ER

struc2vec works when  $0.5 < \sigma < 20$ , and it works best when  $1 < \sigma < 10$ . Notice that they work at different bandwidth regions, and more importantly, the struc2vec provides incorrect result at MMD low bandwidth (high score on same distribution and vice versa), indicating that bandwidth is an important factor in this task.

#### 4.4 Baseline test

To measure the performance of our method, we compare the results with adjacency spectral embedding test. However, during the experiments, we found that using the synthetic datasets mentioned in the section 4.1, both two methods can achieve extremely high accuracy. Therefore, we generated another two populations of datasets using Erdős-Rényi model for this experiment. Two populations contain 100 nodes in each dataset and the size of each population is 10. For the first population, the probability  $p$  of having an edge between two nodes is 0.5 and  $p = 0.4$  in the other population.

Since we have already verified that embedding methods node2vec and Graphwave fails to give useful embedding results for the null hypothesis test, we omitted these two methods and only compared adjacency spectral embedding test with embedding method struc2vec and xNetMF. The threshold of passing the null hypothesis test is  $MMD_u^2 < 0.48$  given the error level  $\alpha$  to be 0.05. Then we ran our method and adjacency spectral embedding test within and between two populations to count the occurrences of Type-I and Type-II errors. The results are shown in Table 1.

	ASE	struc2vec+MMD	xNetMF+MMD
Type-I error rate	0.0	0.0	0.28
Type-II error rate	0.62	0.18	0.09

Table 1: Result of baseline test

Add analysis here

#### 4.5 Robustness test against random noise

We tested how effective our testing method is in the presence of noise and investigated how the testing result will change with the increasing amount of noise. In this experiment, We use 2 real datasets: Arxiv GR-QC and Arxiv ASTRO-PH to generate semi-synthetic graphs with noise. We adopted the random edge flipping as our noise. Given a edge list, we set false positive ( $fp$ ) to be the probability of connecting two unconnected nodes and false negative ( $fn$ ) to be the probability of erasing an existing edge. We then generated semi-synthetic graphs of Arxiv GR-QC with different  $fn$  and  $fp$  parameters. We ran our method to compare the real Arxiv GR-QC and the generated Arxiv GR-QC with noise and also compared the real Arxiv GR-QC with the real Arxiv ASTRO-PH as the control groups. We only ran the tests on struc2vec and xNetMF based on the reason mentioned in section 4.4.

##### 4.5.1 Test results. Add test result and analysis here

## 5 RELATED WORK

Related works are divided to *node embeddings* and *graph two-sample test* respectively.

**Node embedding:** Let  $G = (V, E)$  be a given network. Node embedding aims to find an effective function  $f : V \rightarrow \mathbb{R}^d$  mapping nodes to feature representations, where  $d$  is a parameter specifying the number of dimensions of our feature presentation. Previous work purposed tests based on relatively simple features like eigenvalue distribution, while currently, several related work purposed different embedding methods, such as struc2vec which focuses on structural similarity between nodes regardless of their distances, LINE coping with arbitrary type of large-scale networks, and node2vec which learns the similarity brought by proximity of nodes in the graphs. Although each of the method has its own advantages on dealing with certain kinds of graphs, neglecting some kind of similarity may lead to inferior results. The power of different node embedding methods on distinguishing graphs is still worth exploring.

**Graph two-sample test:** Let  $V$  be a set of  $n$  vertices. Let  $G_1, \dots, G_m$  and  $H_1, \dots, H_m$  be two populations of undirected unweighted graphs defined on the common vertex set  $V$ , where each population consists of independent and identically distributed samples. Graph two-sampling test is to verify whether  $(G_i)_{i=1, \dots, m}$  and  $(H_i)_{i=1, \dots, m}$  are generated from the same random model or not. There have been many famous and successful tests in the history such as ASE test, MMD test and etc. However, the size of the observations and the size of graphs can degrade the power of those tests to a certain degree. Moreover, the impact of applying novel node embedding methods on those tests has not been thoroughly explored yet.

## 6 CONCLUSION

## 7 DIVISION OF WORKS

The following division of works only regards to the work already done, not including future work.

Jiaxin is responsible for applying GraphWave to our dataset and generating corresponding embeddings. Qiucheng collects information about xNetMF and use it to generate embeddings on the same dataset. Yuze is responsible for node2Vec. Shucheng is responsible for struc2Vec and applying all the embeddings to unbiased MMD. We also write the report together.

## REFERENCES

- [1] Béla Bollobás, Svante Janson, and Oliver Riordan. 2007. The phase transition in inhomogeneous random graphs. *Random Structures and Algorithms* 31, 1 (2007), 3–122. <https://doi.org/10.1002/rsa.20168>
- [2] Claire Donnat, Marinka Zitnik, David Hallac, and Jure Leskovec. 2018. Learning Structural Node Embeddings via Diffusion Wavelets. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining - KDD '18* (2018). <https://doi.org/10.1145/3219819.3220025>
- [3] Debarghya Ghoshdastidar, Maurilio Gutzeit, Alexandra Carpentier, and Ulrike von Luxburg. 2017. Two-sample Hypothesis Testing for Inhomogeneous Random Graphs. *arXiv:stat.ME/1707.00833*
- [4] Debarghya Ghoshdastidar, Maurilio Gutzeit, Alexandra Carpentier, and Ulrike von Luxburg. 2017. Two-Sample Tests for Large Random Graphs Using Network Statistics. *arXiv:stat.ME/1705.06168*
- [5] Debarghya Ghoshdastidar and Ulrike von Luxburg. 2018. Practical Methods for Graph Two-Sample Testing. In *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). Curran Associates, Inc., 3019–3028. <http://papers.nips.cc/paper/7565-practical-methods-for-graph-two-sample-testing.pdf>
- [6] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

- [7] Mark Heimann, Haoming Shen, and Danai Koutra. 2018. Node Representation Learning for Multiple Networks: The Case of Graph Alignment. *CoRR* abs/1802.06257 (2018). arXiv:1802.06257 <http://arxiv.org/abs/1802.06257>
- [8] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2007. Graph Evolution: Densification and Shrinking Diameters. *ACM Trans. Knowl. Discov. Data* 1, 1, Article 2 (March 2007). <https://doi.org/10.1145/1217299.1217301>
- [9] Mohammad Mahdian and Ying Xu. 2007. Stochastic Kronecker Graphs. In *Algorithms and Models for the Web-Graph*, Anthony Bonato and Fan R. K. Chung (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 179–186.
- [10] Leonardo F.R. Ribeiro, Pedro H.P. Saverese, and Daniel R. Figueiredo. 2017. struc2vec. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '17* (2017). <https://doi.org/10.1145/3097983.3098061>
- [11] Minh Tang, Avanti Athreya, Daniel L. Sussman, Vince Lyzinski, and Carey E. Priebe. 2014. A nonparametric two-sample hypothesis testing problem for random dot product graphs. arXiv:math.ST/1409.2344