Goodness of fit for Dyadic data

Peter Liu and Shane Lubold

December 11, 2020


facebook
December 2010

Introduction

Let G be an $n \times n$ sociomatrix that is drawn from some distribution F.

Introduction

Let G be an $n \times n$ sociomatrix that is drawn from some distribution F.

We are interested in testing the pair of hypothesis

$$H_0 : F \in P_0, \quad H_a : F \in P \setminus P_0.$$

Introduction

Let G be an $n \times n$ sociomatrix that is drawn from some distribution F.

We are interested in testing the pair of hypothesis

$$H_0 : F \in P_0, \quad H_a : F \in P \setminus P_0.$$

i.e. we are interesting in testing if G is drawn from certain graph model P_0.

Methodology

We introduce two results from random matrix theory to construct our test statistics.

Methodology

We introduce two results from random matrix theory to construct our test statistics.

Theorem 1.

Suppose A is a random symmetric matrix with independent entries such that

$$E(A_{ij}) = 0 \quad \textit{and} \quad Var(\sum_{i \neq j} A_{ij}) = 1.$$

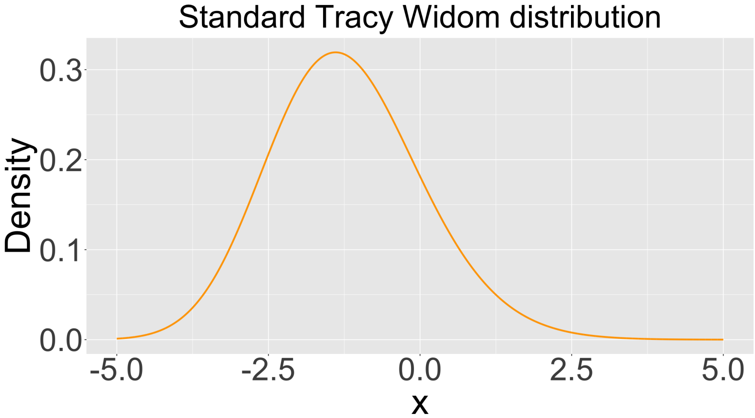
Then

$$n^{2/3}(\lambda_{\max}(A) - 2) \to_d TW_1,$$

$$n^{2/3}(-\lambda_{\min}(A) - 2) \to_d TW_1$$

where TW_1 is the Tracy-Widom distribution with parameter 1.

Tracy Widom distribution



(a) Standard Tracy Widom distribution

Methodology

Theorem 2.

Suppose A is a random asymmetric matrix such that

$$E(A_{ij}) = 0 \quad and \quad Var(A_{ij}) = 1.$$

Then

$$n\sigma_n(A) \rightarrow_d Exp(1).$$

where $\sigma_n(A)$ is the smallest singular value of A.

Methodology

Suppose the sociomatrix G is drawn from $F = P_0(\boldsymbol{\theta})$.

Methodology

Suppose the sociomatrix G is drawn from $F = P_0(\boldsymbol{\theta})$.

- We can always transform G into A using property of P_0 and parameter $\boldsymbol{\theta}$.

Methodology

Suppose the sociomatrix G is drawn from $F = P_0(\boldsymbol{\theta})$.

- We can always transform G into A using property of P_0 and parameter $\boldsymbol{\theta}$.
- For instance, if we can parameterize a symmetric simple graph G with some probability matrix P such that

$$Pr(G_{ij} = 1|P) = P_{ij},$$

Methodology

Suppose the sociomatrix G is drawn from $F = P_0(\boldsymbol{\theta})$.

- We can always transform G into A using property of P_0 and parameter $\boldsymbol{\theta}$.
- For instance, if we can parameterize a symmetric simple graph G with some probability matrix P such that

$$Pr(G_{ij} = 1|P) = P_{ij},$$

Then the normalized adjacency matrix is given as

$$A_{ij} = \frac{P_{ij} - G_{ij}}{\sqrt{(n-1)P_{ij}(1 - P_{ij})}}.$$

Methodology

- The test statistic is given as

$$t_{symm} = n^{2/3}(\lambda_{\max}(A) - 2),$$

$$t_{asymm} = n\sigma(A).$$

Methodology

- The test statistic is given as

$$t_{symm} = n^{2/3}(\lambda_{\max}(A) - 2),$$

$$t_{asymm} = n\sigma(A).$$

- The test statistic should converge to Tracy-Widom/exp(1) distribution for valid models.

Methodology

Now suppose only G is given. How can we access A?

Methodology

Now suppose only G is given. How can we access A?

- First suppose G is drawn from some graph model P_0.

Methodology

Now suppose only G is given. How can we access A?

- First suppose G is drawn from some graph model P_0.
- We can estimate $\hat{\theta}$ using G. In our case, we will plug in $\hat{\theta}_{MLE}$.

Methodology

Now suppose only G is given. How can we access A?

- First suppose G is drawn from some graph model P_0.
- We can estimate $\hat{\boldsymbol{\theta}}$ using G. In our case, we will plug in $\hat{\boldsymbol{\theta}}_{MLE}$.
- Then normalize G into $\hat{A}$ using $\hat{\boldsymbol{\theta}}$ and compute the test statistic.

Methodology

Now suppose only G is given. How can we access A?

- First suppose G is drawn from some graph model P_0.
- We can estimate $\hat{\boldsymbol{\theta}}$ using G. In our case, we will plug in $\hat{\boldsymbol{\theta}}_{MLE}$.
- Then normalize G into $\hat{A}$ using $\hat{\boldsymbol{\theta}}$ and compute the test statistic.
- Reject if p-value $< \alpha = 0.05$.

Methodology

Will this always work?

Methodology

Will this always work?
Several caveats:

Methodology

Will this always work?
Several caveats:

- This is model specific! i.e. for symmetric model, we require

$$n^{2/3}(\lambda_{\max}(A) - \lambda_{\max}(\hat{A})) = o_p(1)/O_p(1).$$

Methodology

Will this always work?
Several caveats:

- This is model specific! i.e. for symmetric model, we require

$$n^{2/3}(\lambda_{\mathsf{max}}(A) - \lambda_{\mathsf{max}}(\hat{A})) = o_p(1)/O_p(1).$$

- Must have access to the MLE.

Methodology

Will this always work?
Several caveats:

- This is model specific! i.e. for symmetric model, we require

$$n^{2/3}(\lambda_{\max}(A) - \lambda_{\max}(\hat{A})) = o_p(1)/O_p(1).$$

- Must have access to the MLE.
- Valid $\hat{P}$, i.e. no $0/1$ values on off-diagonal entries.

Methodology

Will this always work?
Several caveats:

- This is model specific! i.e. for symmetric model, we require

$$n^{2/3}(\lambda_{\max}(A) - \lambda_{\max}(\hat{A})) = o_p(1)/O_p(1).$$

- Must have access to the MLE.
- Valid $\hat{P}$, i.e. no $0/1$ values on off-diagonal entries.
- The convergence to Tracy-Widom is always slow.

Methodology

Will this always work?
Several caveats:

- This is model specific! i.e. for symmetric model, we require

$$n^{2/3}(\lambda_{\max}(A) - \lambda_{\max}(\hat{A})) = o_p(1)/O_p(1).$$

- Must have access to the MLE.
- Valid $\hat{P}$, i.e. no 0/1 values on off-diagonal entries.
- The convergence to Tracy-Widom is always slow.
- But we can fix it via bootstrap!

Valid Models

Below is a list of graph models that can be applied with our method:

Valid Models

Below is a list of graph models that can be applied with our method:

	Undirected	Directed
Binary graph	ER, SBM, Beta, SRM	ER, SRM
Multi-graph	DCSBM	
Continuous graph	SRM	SRM

Valid Models

Below is a list of graph models that can be applied with our method:

	Undirected	Directed
Binary graph	ER, SBM, Beta, SRM	ER, SRM
Multi-graph	DCSBM	
Continuous graph	SRM	SRM

- Model that doesn't work: Latent space model.

Valid Models

Below is a list of graph models that can be applied with our method:

	Undirected	Directed
Binary graph	ER, SBM, Beta, SRM	ER, SRM
Multi-graph	DCSBM	
Continuous graph	SRM	SRM

- Model that doesn't work: Latent space model.
- We are currently working on ERGM!

Algorithm

Suppose we want to fit a graph G with respect to several graph model.

Algorithm

Suppose we want to fit a graph G with respect to several graph model.

- Arrange the models of interest in a descending order of complexity, i.e. ERGM $\supseteq$ Beta $\supseteq$ SBM $\supseteq$ ER.

Algorithm

Suppose we want to fit a graph G with respect to several graph model.

- Arrange the models of interest in a descending order of complexity, i.e. ERGM $\supseteq$ Beta $\supseteq$ SBM $\supseteq$ ER.
- Test starting with the simplest model.

Algorithm

Suppose we want to fit a graph G with respect to several graph model.

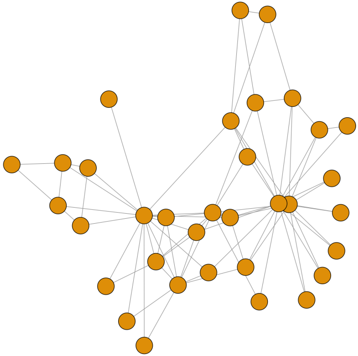
- Arrange the models of interest in a descending order of complexity, i.e. ERGM $\supseteq$ Beta $\supseteq$ SBM $\supseteq$ ER.
- Test starting with the simplest model.
- Our estimate of the model is the first model we fail to reject.

Algorithm

Suppose we want to fit a graph G with respect to several graph model.

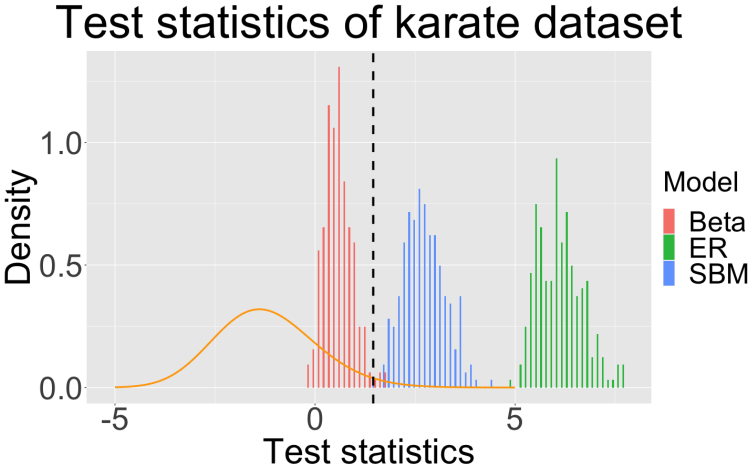
- Arrange the models of interest in a descending order of complexity, i.e. ERGM $\supseteq$ Beta $\supseteq$ SBM $\supseteq$ ER.
- Test starting with the simplest model.
- Our estimate of the model is the first model we fail to reject.
- If all models of interest are rejected, we probably need more complex models to fit our data.

Zachary karate data



(b) Karate club

Zachary karate data



(c) Zachary karate data

Supplementary: Bootstrap correction

1. Given G, compute the MLE $\hat{\boldsymbol{\theta}}$ for a given model P_0.
2. For $b = 1, \ldots, B$:
   2.1 Generate $G_b^\star$ assumed graph model $P_0(\hat{\boldsymbol{\theta}})$.
   2.2 Compute $\lambda_{\max}^\star$ and $\lambda_{\min}^\star$ for $A^\star$:

$$A^\star := \frac{G_b^\star - \hat{P}}{\sqrt{(n-1)\hat{P}(1-\hat{P})}}.$$

3. Reject H_0 when

$$T_{boot} = \mu_{tw} + s_{tw} \max\left(\frac{\lambda_1(\hat{A}) - \hat{\mu}_1}{\hat{s}_1}, -\frac{\lambda_n(\hat{A}) - \hat{\mu}_n}{\hat{s}_n}\right)$$

is bigger than quantile of TW.