Complex Network Analysis on Bike-Sharing Systems

M.Sc. Siyu Chen
Technical Presentation
Supervisor: Prof. Zhenghui Sha

Institute of Information-oriented Control
Technical University of Munich

Motivation

Bike-Sharing Systems (BSS)

Bike-sharing systems (BSS) are emerging transportation systems arising from the sharing economy.

Motivation

Bike-Sharing Systems (BSS)

Bike-sharing systems (BSS) are emerging transportation systems arising from the sharing economy.

Challenges

- How to transform the weighted network to a sparsified binary network with an appropriate weight threshold matrix?
- Given a social network, how to infer the new interactions among its members in the near future?

Motivation

Bike-Sharing Systems (BSS)

Bike-sharing systems (BSS) are emerging transportation systems arising from the sharing economy.

Challenges

- How to transform the weighted network to a sparsified binary network with an appropriate weight threshold matrix?
- Given a social network, how to infer the new interactions among its members in the near future?

- Task 1: minimum absolute spectral similarity (MASS) technique for determining the weight threshold.
- Task 2: classification in machine learning for link prediction.

Task 1 - Methods

- minimum absolute spectral similarity (MASS) [Yan+ 2018]

$$\sigma_{\min} = \min_{|x|=1} \left(1 - \frac{x^T \Delta L x}{\lambda_N} \right) = \frac{\lambda_N - \lambda_N^{\Delta}}{\lambda_N} \tag{1}$$

Task 1 - Methods

- minimum absolute spectral similarity (MASS) [Yan+ 2018]

$$\sigma_{\min} = \min_{|x|=1} \left(1 - \frac{x^T \Delta L x}{\lambda_N} \right) = \frac{\lambda_N - \lambda_N^{\Delta}}{\lambda_N} \tag{1}$$

$$\downarrow$$

Task 1 - Methods

- minimum absolute spectral similarity (MASS) [Yan+ 2018]

$$\sigma_{\min} = \min_{|x|=1} \left(1 - \frac{x^T \Delta L x}{\lambda_N}\right) = \frac{\lambda_N - \lambda_N^{\Delta}}{\lambda_N} \tag{1}$$

$$\downarrow$$

By MASS, we can evaluate what percentage of links are removed, the graph can still remain good graph property σ.

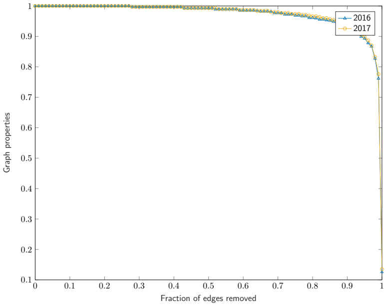
Task 1 - Methods

Algorithm 1 pseudocode for MASS

Generate original weighted matrix W from data file;
Initialization for threshold matrix $X = W$ and sample period p;
$t = 0 : p : 1$
for $i = 1$ to $1/p$ **do**
    Calculate the value of weight x in $t(i) * 100$-th percentile;
    $X(W < x) = 0$;
    Calculate degree matrices D_0 and D of W and X;
    $L = D_0 - W$ and $\tilde{L} = D - X$;
    $\Delta\lambda_N = L - \tilde{L}$;
    Calculate minimum eigenvalues λ_N^Δ and λ_N of ΔL and L;
    Calculate $\sigma(i)$ according to Eq. (1);
end for

Task 1 - Results

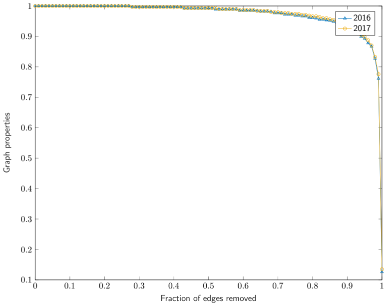
- MASS



Variation of MASS, for networks in 2016 and in 2017

Task 1 - Results

- MASS

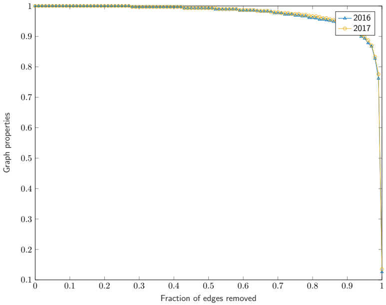


$\Longrightarrow$

Variation of MASS, for networks in 2016 and in 2017

Task 1 - Results

- MASS



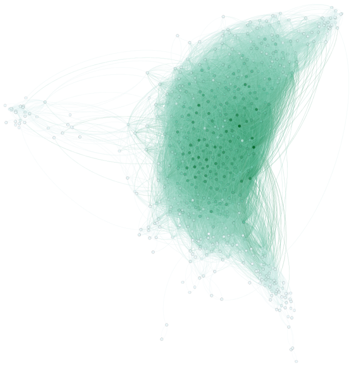
Variation of MASS, for networks in 2016 and in 2017

$\Longrightarrow$

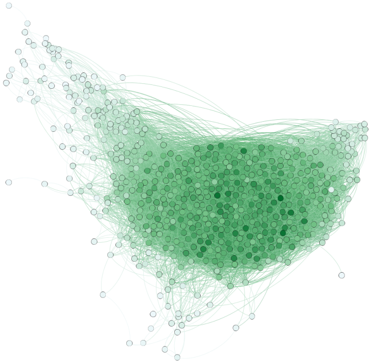
When 30% links are removed, the networks for 2016 and 2017 remain good graph properties!

Task 1 - Results

- Visualization for resulting networks $\tilde{G}_1$ in 2016 and $\tilde{G}_2$ in 2017



Visualization for $\tilde{G}_1$



Visualization for $\tilde{G}_2$

Task 1 - Results

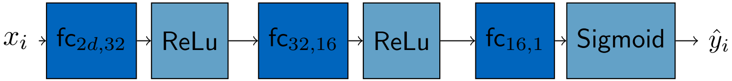
- Network statistics

Statistics for resulting networks $\tilde{G}_1$ and $\tilde{G}_2$

Network statistics	2016	2017
Average Degree	73.4	75.949
Network Diameter	10	9
Graph Density	0.135	0.139
Modularity	0.222	0.268
Average cluster coefficient	0.47	0.469

Task 2 - Methods

- Dataset: We describe the resulting graph $\tilde{G}$ as dataset $\mathcal{D} = \{X, Y\}$, where $X = \{x_1, ..., x_n\} \in \mathbb{R}^{2d \times n}$, $Y = \{y_1, ..., y_n\} \in \mathbb{R}^n$, feature vector $x_i \in \mathbb{R}^{2d}$ and label $y_i \in \{0, 1\}$.
- NN-based classifier:



Structure of the $l = 3$ fully-connected layers

Task 2 - Methods

- Dataset: We describe the resulting graph $\tilde{G}$ as dataset $\mathcal{D} = \{X, Y\}$, where $X = \{x_1, ..., x_n\} \in \mathbb{R}^{2d \times n}$, $Y = \{y_1, ..., y_n\} \in \mathbb{R}^n$, feature vector $x_i \in \mathbb{R}^{2d}$ and label $y_i \in \{0, 1\}$.

- NN-based classifier:

$$x_i \rightarrow \text{fc}_{2d,32} \rightarrow \text{ReLu} \rightarrow \text{fc}_{32,16} \rightarrow \text{ReLu} \rightarrow \text{fc}_{16,1} \rightarrow \text{Sigmoid} \rightarrow \hat{y}_i$$

Structure of the $l = 3$ fully-connected layers

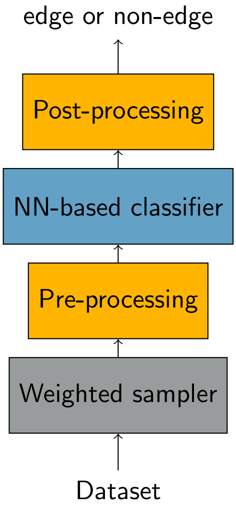
Challenge

How to solve class imbalance for the number of "edge" and "non-edge"?

Task 2 - Results

- Dataset:
  - $n = 338k$ training data points from data file 2016 $n = 342k$ validation data points from data file 2017.
  - *feature vector*: node attributes $a \in \mathbb{R}^3$, "latitude", "longitude" and "dpcapacity" from $\tilde{v}$ to $\tilde{u}$, $x_i \in \mathbb{R}^6$; *labels*: "edge" class $y_i = 1$ and "non-edge" class $y_i = 0$.

- Link prediction framework



Task 2 - Results

- Evaluation

Evaluation based on two samplers

sampler	accuracy	e-precision	e-recall
routine sampler	0.89	0	0
weighted sampler	0.69	0.21	0.84

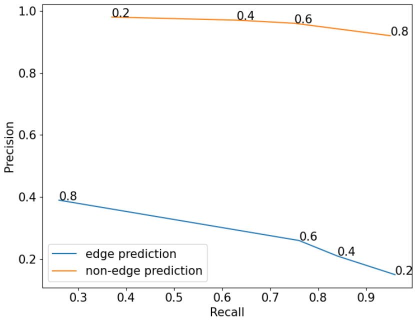
Evaluation based on different class weight ratio

edge: non-edge	accuracy	e-precision	e-recall	n-precision	n-recall
0.8: 0.2	0.80	0.50	0.05	0.90	0.99
0.9: 0.1	0.69	0.39	0.26	0.92	0.95
0.95: 0.05	0.56	0.28	0.65	0.95	0.79
0.99: 0.01	0.30	0.16	0.93	0.98	0.44

Task 2 - Results

- Evaluation

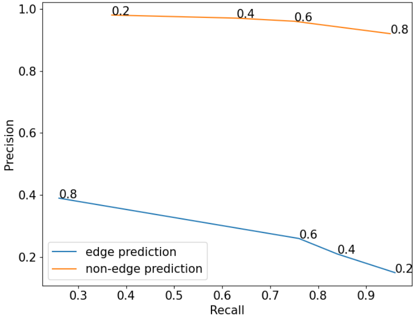
  The precision-recall curve



Task 2 - Results

- Evaluation

  The precision-recall curve

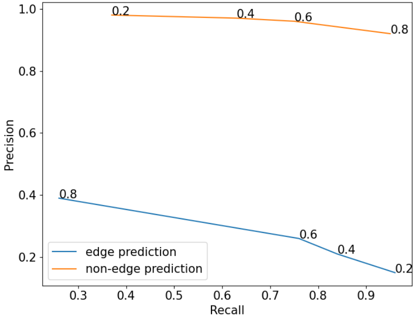


$\Longrightarrow$

Task 2 - Results

- Evaluation

  The precision-recall curve



$\Longrightarrow$

The threshold value 0.6 is chosen for the balance between precision and recall in post-processing.

Conclusion

- The weight threshold matrix is determined by removing 30% of links using the MASS technique.

Conclusion

- The weight threshold matrix is determined by removing 30% of links using the MASS technique.
- Link prediction framework based on NN classifier is designed to establish a model.

Conclusion

- The weight threshold matrix is determined by removing 30% of links using the MASS technique.

- Link prediction framework based on NN classifier is designed to establish a model.

$\Longrightarrow$

Conclusion

- The weight threshold matrix is determined by removing 30% of links using the MASS technique.

- Link prediction framework based on NN classifier is designed to establish a model.

  $\Longrightarrow$

  Weighted sampler is adopted to guarantee a balanced training data for "edge" class and "non-edge" classes and improve the performance of the model.

References

Xiaoran Yan, Lucas Jeub, Alessandro Flammini, Filippo Radicchi and Santo Fortunato. **Weight Thresholding on Complex Networks**. June 2018.