Optimal Intervention on Weighted Networks via Edge Centrality

Dongyue Li, Tina Eliassi-Rad, Hongyang R. Zhang
Email addresses: {li.dongyu, t.eliassirad, ho.zhang}@northeastern.edu

Northeastern University
Khoury College of Computer and Information Sciences

Problem Statement

Network diffusion processes are ubiquitous in our well-connected society, such as the spread of disease or the dissemination of information.

How can we design efficient weight reduction measures to slow down a network diffusion process?

Spectral property. The spreading rate of an epidemic process depends on the largest eigenvalue λ_1 of the graph [CWW+'08; PCV+'12].

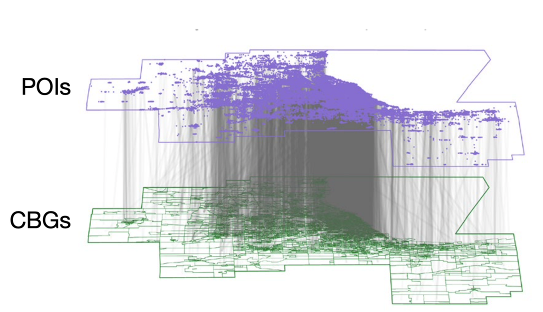
Problem statement: Given a weight reduction budget B, how can we reduce the largest r eigenvalues of $W^\top W$?
- Let W denote the non-negative edge weight matrix of graph G.
- Let λ_k denote the k-th largest singular value.

$$\min_{M} \quad \sum_{k=1}^{r} \lambda_k^2(M)$$
$$\text{s.t.} \quad \sum_{(i,j)\in E} (W_{i,j} - M_{i,j}) \le B$$
$$0 \le M_{i,j} \le W_{i,j}, \text{ for any } (i,j)$$

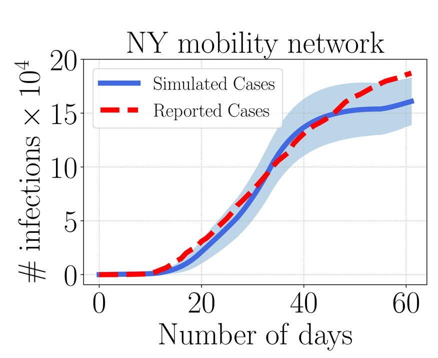
Motivation and Related Work

Mobility Modeling for COVID-19: Mobility networks can accurately fit the reported COVID-19 case counts [CPK+'21; CWL+'21].

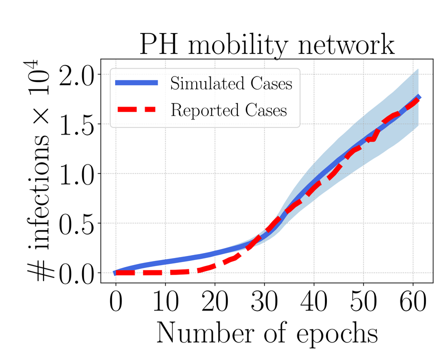
- Mobility networks describe the movement of people from neighborhoods to points of interest, such as restaurants and grocery stores.
- Metapopulation SEIR models are overlaid on neighborhoods.



Mobility network illustration



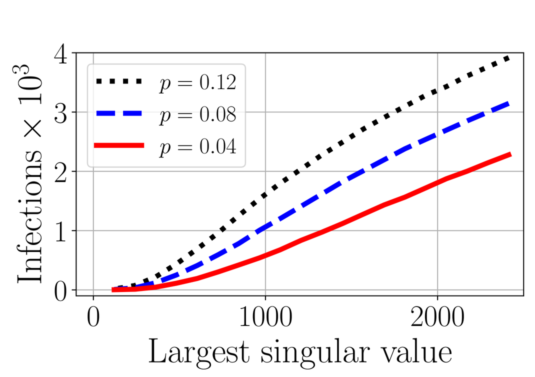
Mobility network on NY



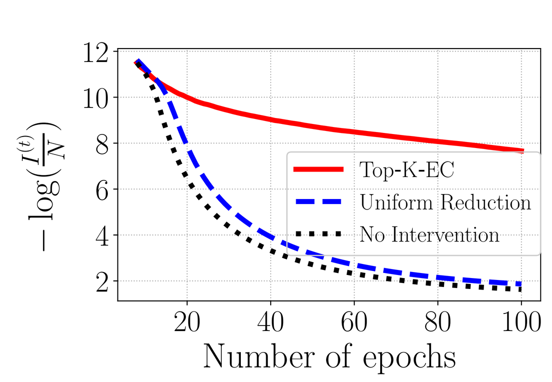
Mobility network on PH

Edge Centrality is a key notion that indicates the influence of an edge in reducing the top singular values.

- The edge centrality score of an edge $e = (x, y)$ is defined as $\vec{u}_1(x) \cdot \vec{v}_1(y)$, where $\vec{u}_1, \vec{v}_1$ are the first left and right singular vector of W.
- Tong et al. (2012) proposed to remove top-k edges with the highest edge centrality scores and found that it is effective in reducing λ_1.



Correlation with λ_1



Effect of Top-k-EdgeCentrality

Iterative Edge Centrality Minimization

Lemma 1: Edge centrality as gradients. The edge centrality scores are equal to the gradient of $\lambda_1^2(W)$ concerning the edge weights upto scaling.

$$\frac{\partial\big((\lambda_1(M))^2\big)}{\partial M_{i,j}} = 2\lambda_1(M)\cdot \vec{u}_1(i)\cdot \vec{v}_1(j).$$

More generally, this applies to any $r = 1, 2, \dots, n$:

$$\nabla f(M) = \frac{\partial\big(\sum_{k=1}^{r}(\lambda_k(M))^2\big)}{\partial M_{i,j}} = 2\sum_{k=1}^{r}\lambda_k(M)\cdot \vec{u}_k(i)\cdot \vec{v}_k(j).$$

Proof: Consider differentiating over both sides of the equation $\vec{u}_k^\top M = \lambda_k \vec{v}_k^\top$.

Algorithm implication. We can leverage the Frank-Wolfe algorithm (1956) for the constrained convex minimization problem.
- Compute the gradient of the convex objective.
- Project the gradient to the constrained set as the descent direction.

Lemma 2: Projection is greedy edge selection. The projection of the gradient into constraints is essentially performing top-k edge deletion.

Proof: Consider finding the best X when minimizing the $\langle X, \nabla f(M)\rangle$ under the same budget constraints. Each variable $X_{i,j}$ is multiplied precisely by the generalized edge centrality of each edge.

Our approach: Iteratively applying a greedy selection of edges with the highest generalized edge centrality scores while recomputing the scores.

- Gradient: Compute edge centrality via SVD.
- Projection: Remove top-k edges with the highest edge centrality.
- Update the weight matrix by moving it along the descent direction.

Extension to Time-Varying Networks

We extend our optimization algorithm to time-varying networks.

- Let $W^{(1)}, \dots, W^{(s)}$ denote a sequence of weighted networks.
- The epidemic threshold of time-varying networks is the largest singular value of the product of matrices $\lambda_1(\prod_{t=1}^{s} W^{(t)})$ [PTV+'10].

How can we reduce the edge weights in each network to minimize the top singular values of the weight product?

$$\min_{\mathcal{M}} \quad f(\mathcal{M}) = \sum_{k=1}^{r}\Big(\lambda_k\Big(\prod_{t=1}^{s} M^{(t)}\Big)\Big)^2$$

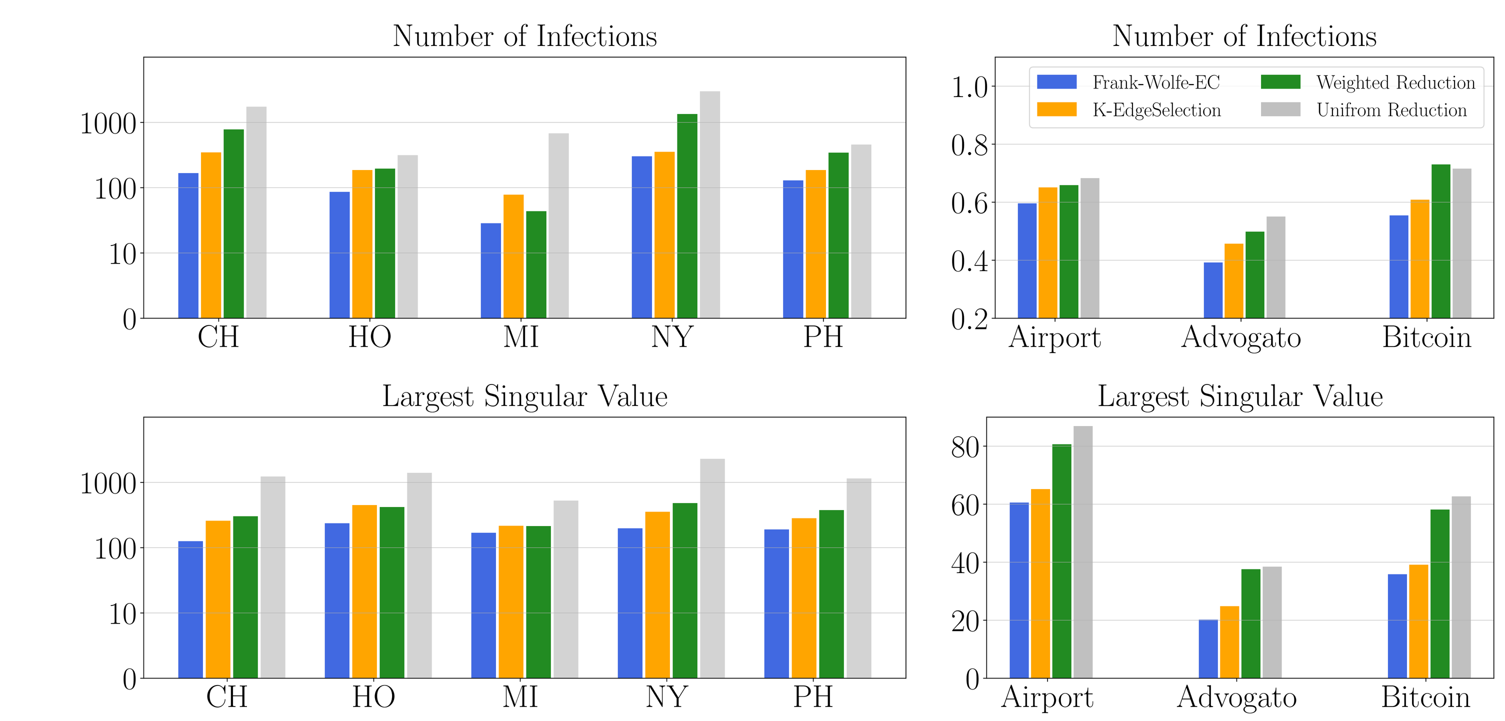
Finding the gradients. From Lemma 1, we can derive the gradient of top singular values to the t-th network. Let $\tilde{X}_r$ be the rank-r SVD of $M^{(t)}$.

$$\frac{\partial f(\mathcal{M})}{\partial M^{(t)}} = 2\Big(\prod_{k=1}^{t-1} M^{(k)}\Big)^\top \tilde{X}_r \Big(\prod_{k=t+1}^{s} M^{(k)}\Big)^\top.$$

Experimental Results

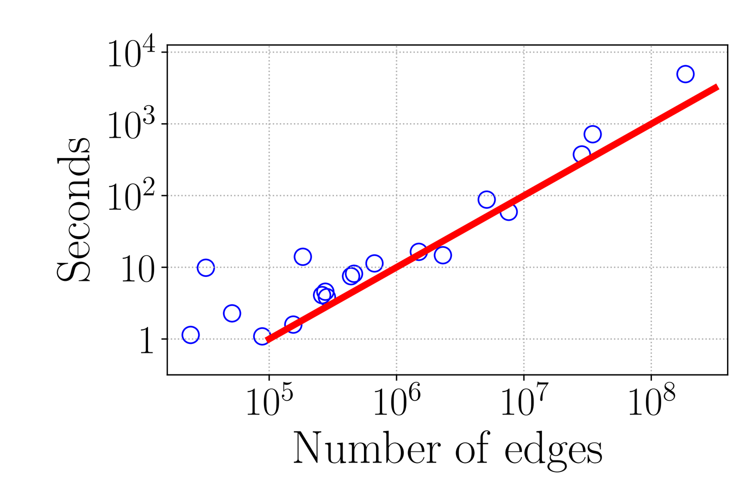
Network intervention in simulated SEIR models. We evaluate our algorithm in eight mobility networks constructed from real-world mobility patterns during the COVID-19 pandemic and three weighted graphs.

- Our algorithm reduces the number of infections by 25.5% more than the baselines.
- Our algorithm decreases the largest singular value by 25.1% more than the baselines.



Comparing our algorithm against baselines: Greedy edge selection based on the rank-1 edge centrality (K-EdgeDeletion); Reducing edge weights in proportional to its edges (Weighted reduction); Reducing edge weights with a uniform ratio (Uniform reduction).

Runtime analysis.
- Our approach converges to the global minimum within 30 iterations.
- Each iteration runs in nearly-linear time in the size of the graph.



Extension to other settings.
- Our algorithm is also effective for **SIS and SIR** epidemic models.
- Our algorithm is also effective for **time-varying networks**, tested on ten weekly mobility networks in Chicago and Houston.

Conclusion

- We revisited edge centrality scores as gradients for the singular values.
- We designed an iterative edge centrality minimization algorithm in which each iteration is greedy edge selection.
- Our algorithm optimally reduces the top singular values of weighted networks in epidemic spreading with nearly-linear time.
- Paper: https://arxiv.org/pdf/2303.09086.pdf
- Code: NEU-StatsML-Research/Intervention-on-Mobility-Networks