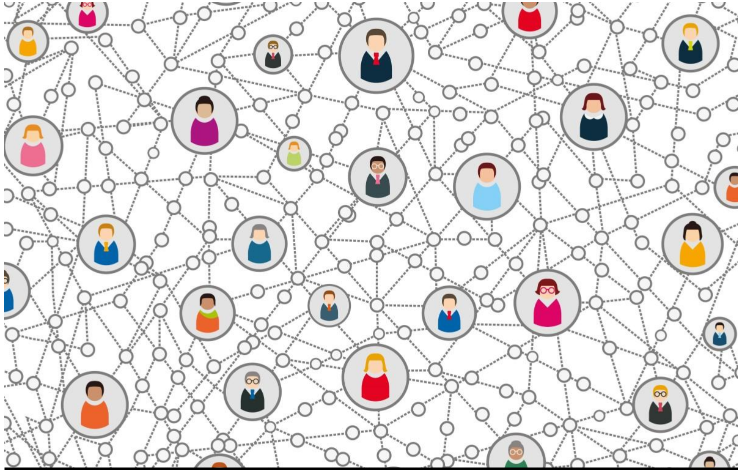
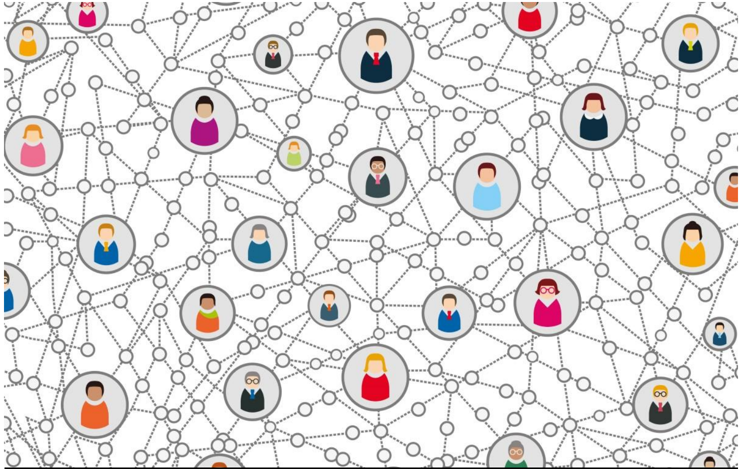
Stopping Disease Spreading with Non-Backtracking Eigenvalues

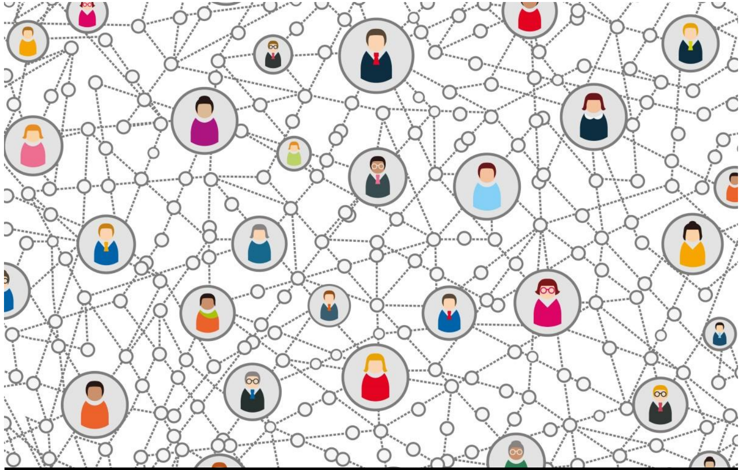
Leo Torres
PhD candidate
Network Science Institute, Northeastern University


https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/



https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

How to stop the **spread** of **disease**?

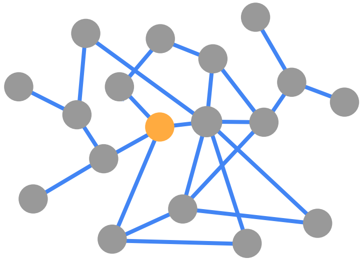


https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

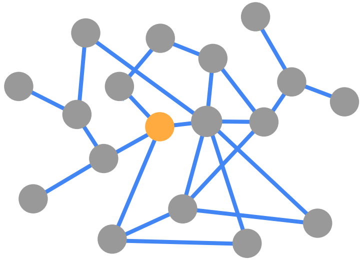
How to stop the **spread** of **disease**?

Consider a graph **G**, where each node is a person and each link is a connection along which disease may spread. How to **modify the graph** in order to **stop the spread?**

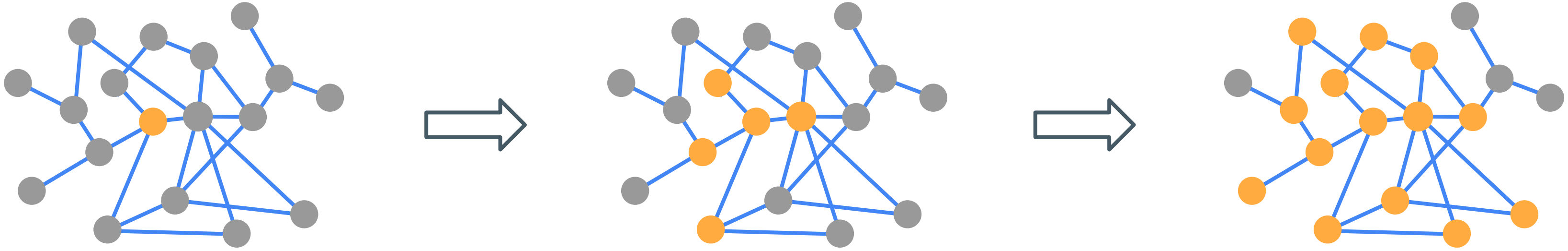
Similar graphs, different spread



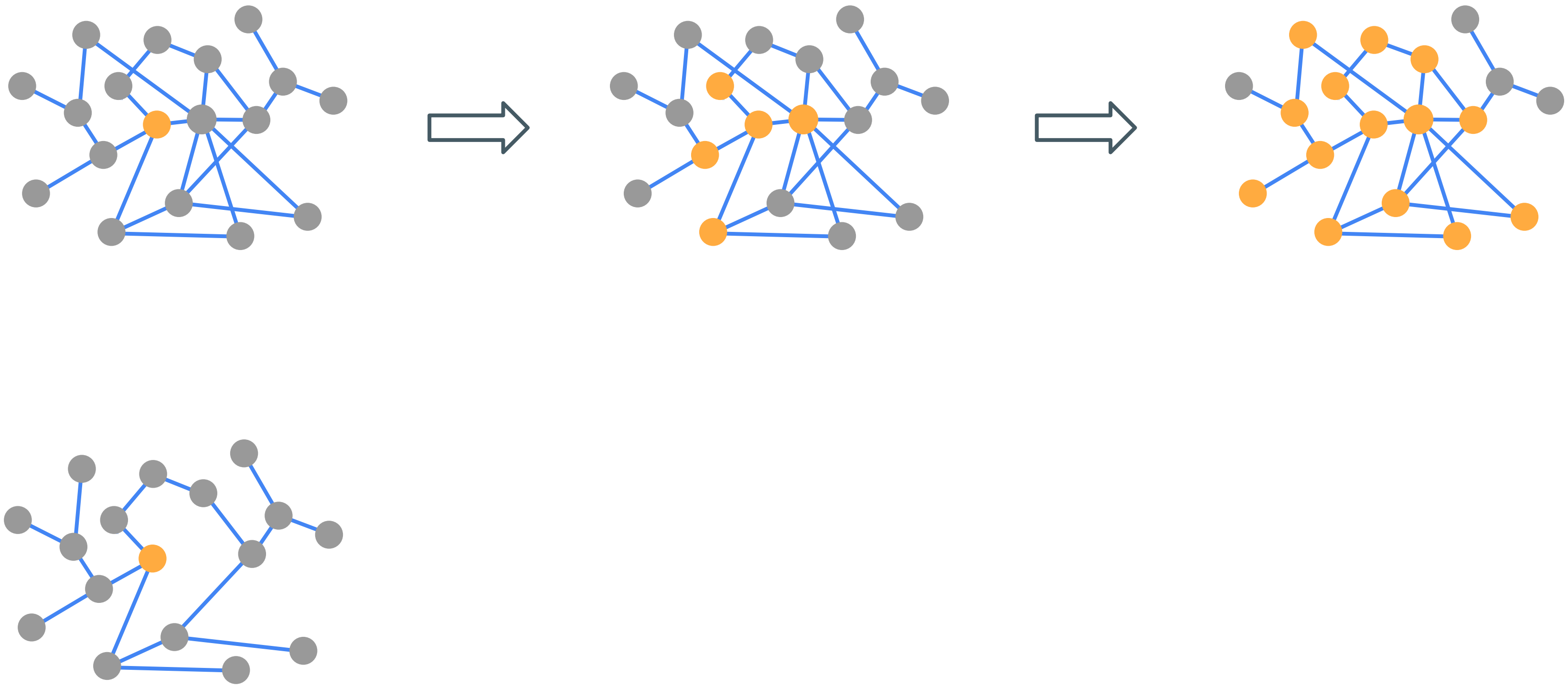
Similar graphs, different spread



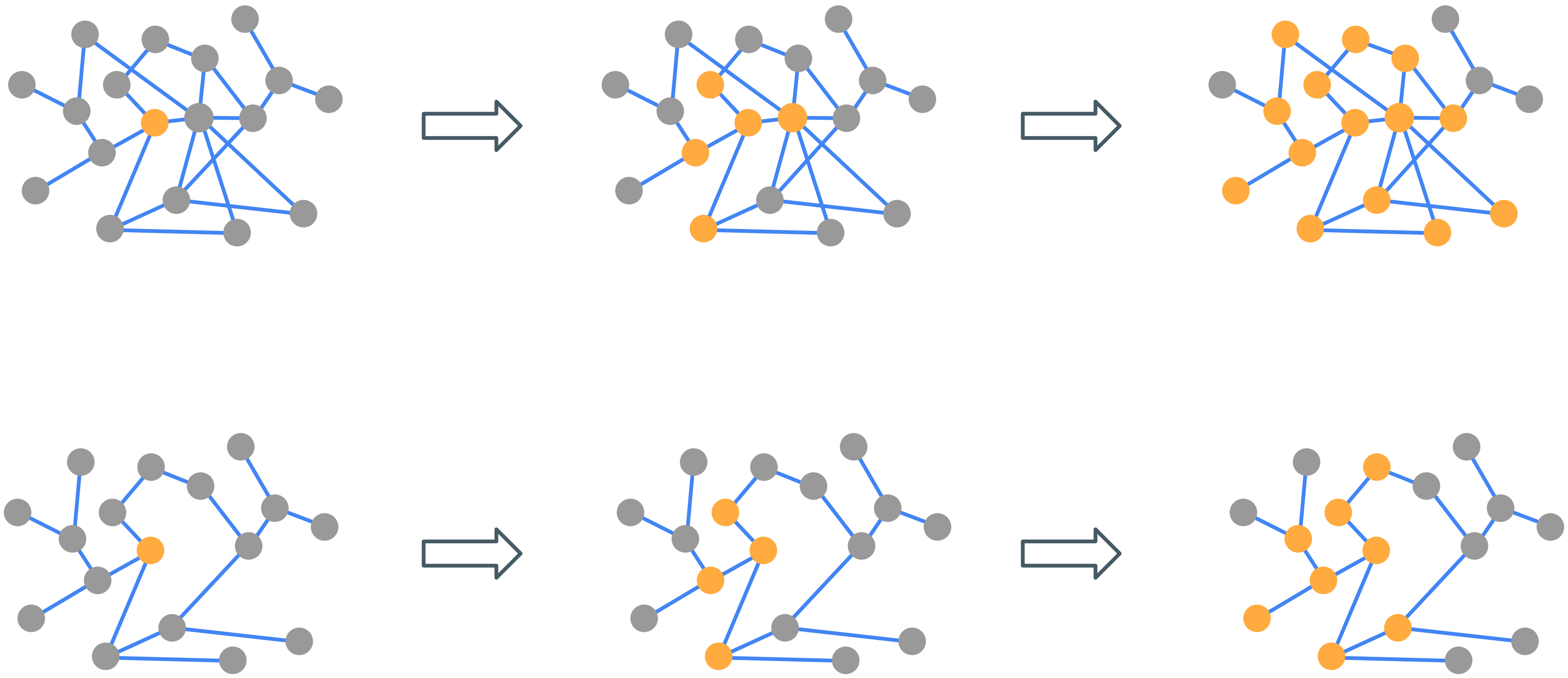
Similar graphs, different spread



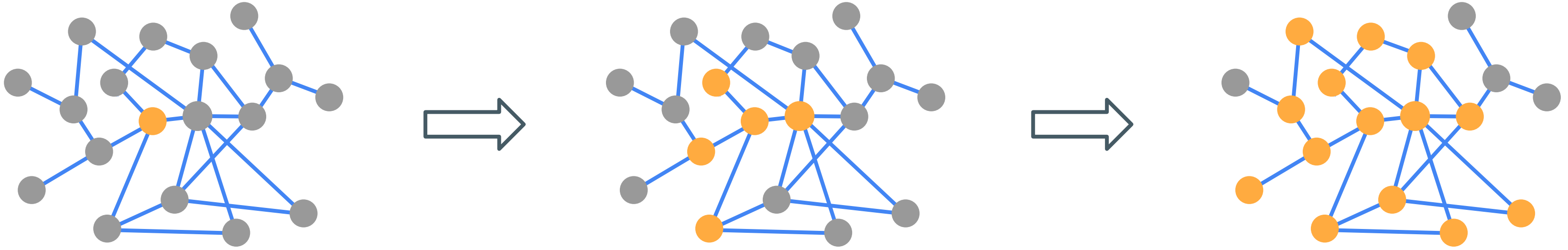
Similar graphs, different spread



Similar graphs, different spread

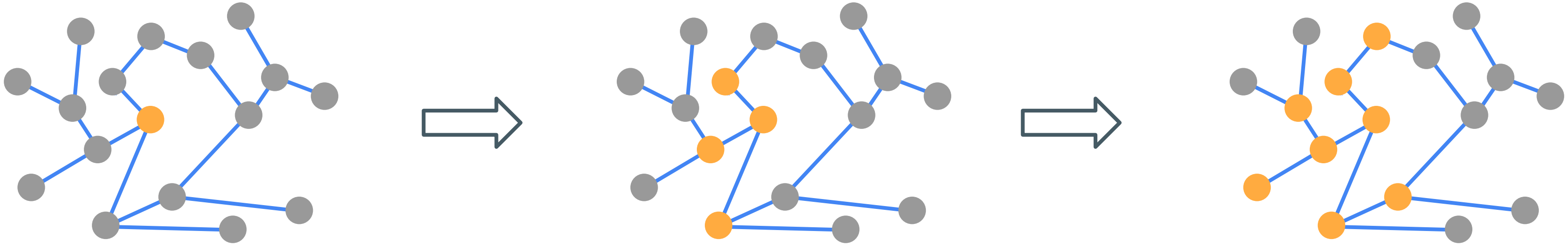


Similar graphs, different spread

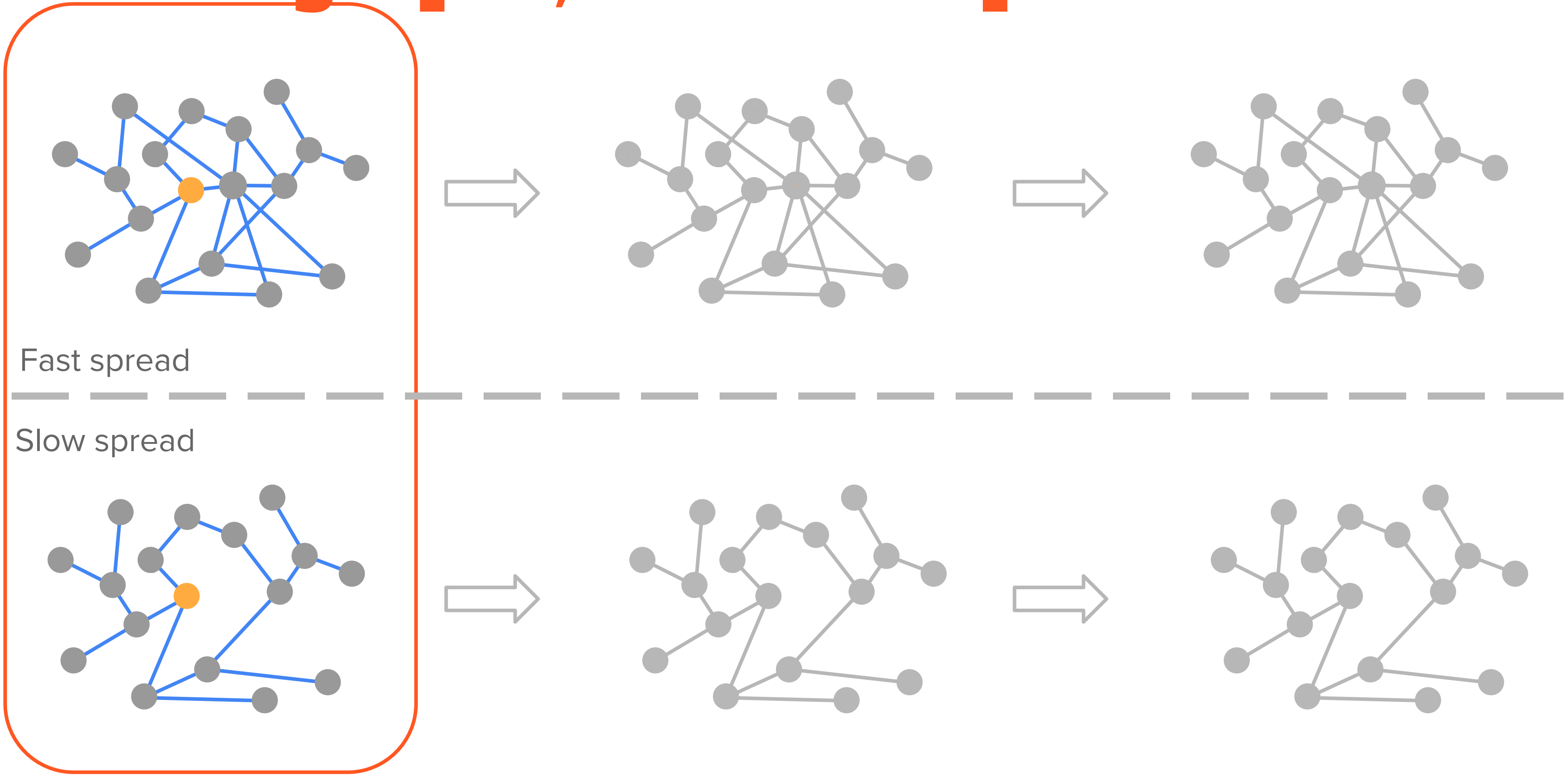


Fast spread

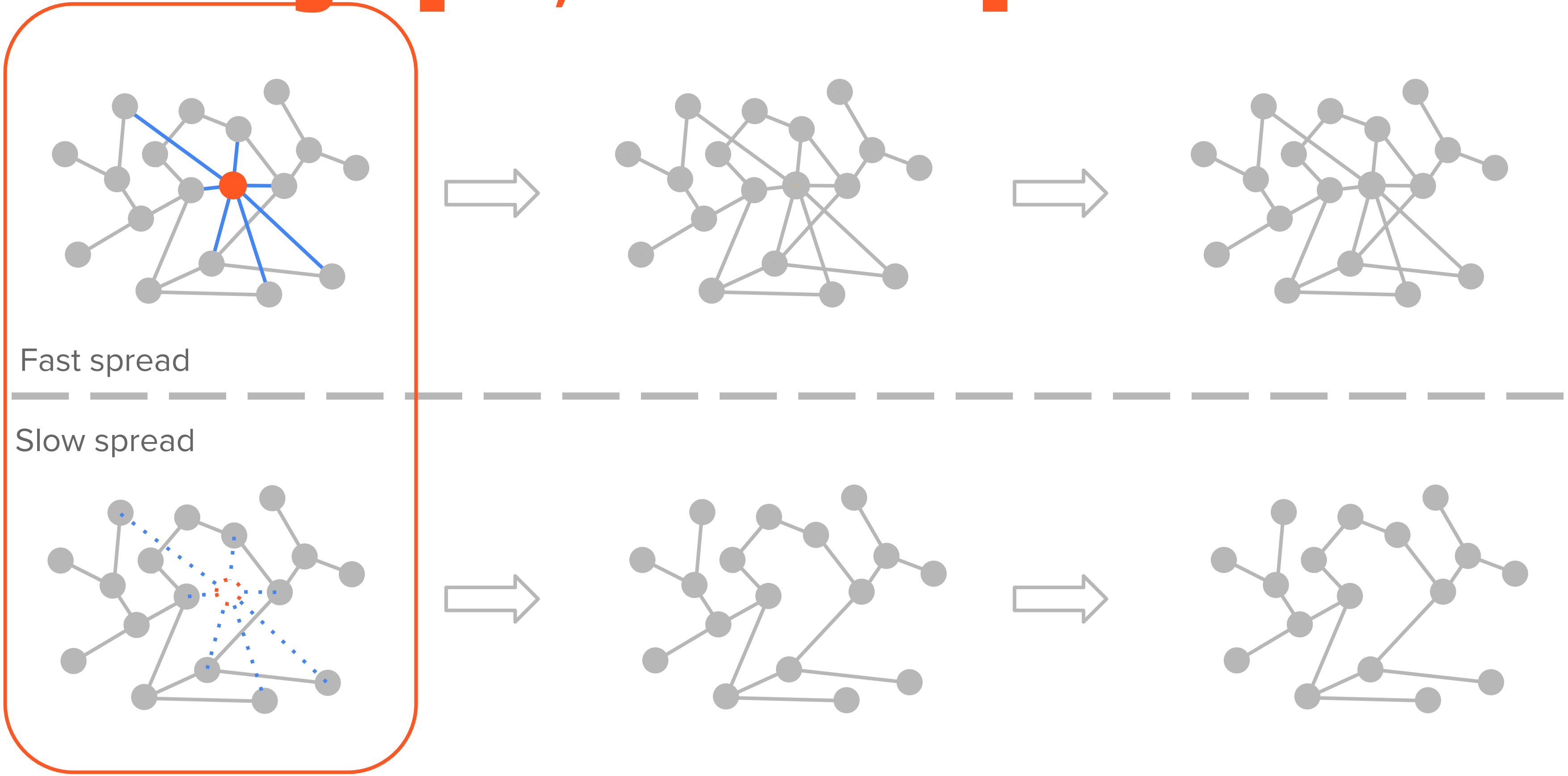
Slow spread



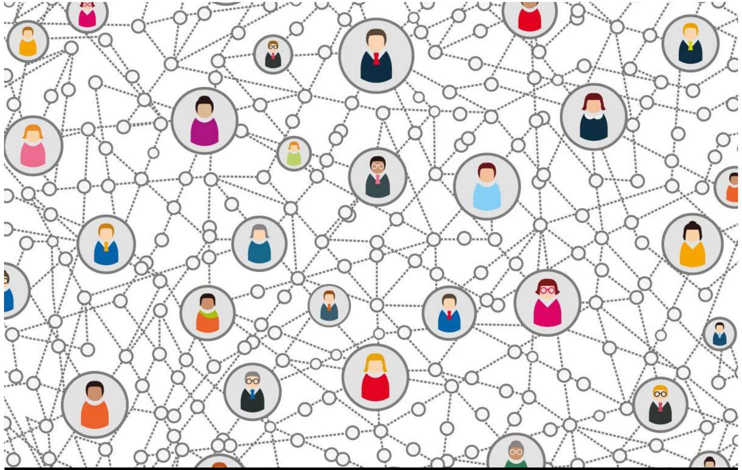
Similar graphs, different spread



Similar graphs, different spread



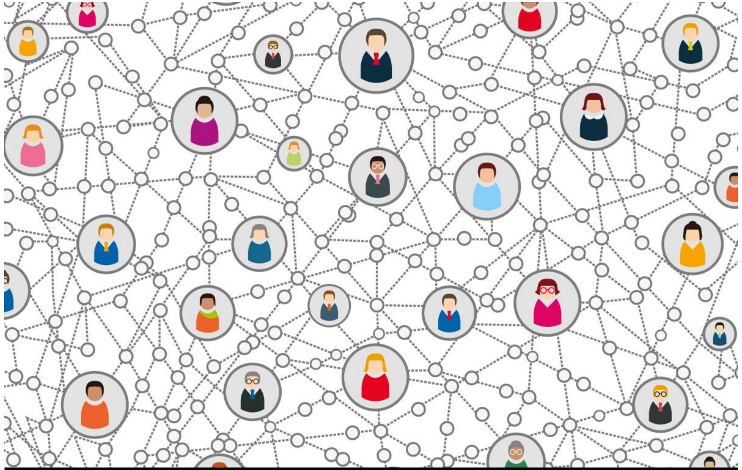
The research question



https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

Consider a graph **G**, where each node is a person and each link is a connection along which disease may spread. How to **modify the graph** in order to **stop the spread?**

The research question

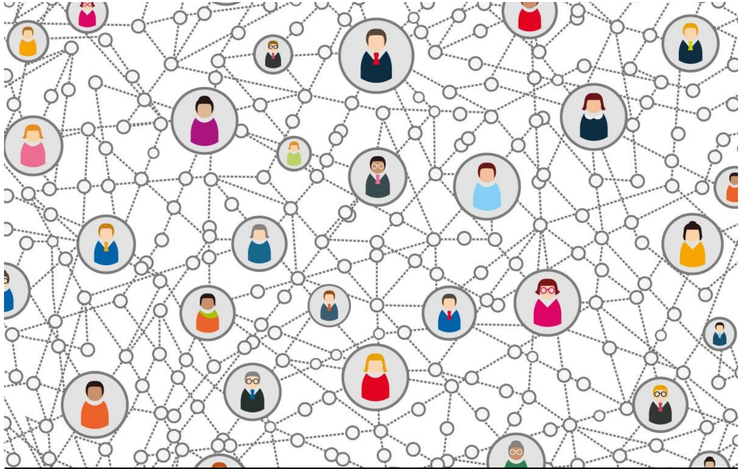

https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

Consider a graph **G**, where each node is a person and each link is a connection along which disease may spread. How to **modify the graph** in order to **stop the spread?**

⇩

Which nodes should we remove in order to **slow the spread?**

The research question



https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

Consider a graph **G**, where each node is a person and each link is a connection along which disease may spread. How to **modify the graph** in order to **stop the spread?**

⇩

Which nodes should we remove in order to **slow the spread?**

“node immunization”

Node Immunization

Given a graph **G**,

1. **Rank the nodes**
2. Pick the first one
3. Remove it (*immunize* it)
4. Repeat

Node Immunization

Given a graph **G**,

1. **Rank the nodes**
2. Pick the first one
3. Remove it (*immunize* it)
4. Repeat

Different ways of **ranking the nodes**:

- Degree (number of edges)

Node Immunization

Given a graph **G**,

1. **Rank the nodes**
2. Pick the first one
3. Remove it (*immunize* it)
4. Repeat

Different ways of **ranking the nodes**:

- Degree (number of edges)
- PageRank
- Eigenvector centrality
- Coreness (k-core index)
- Collective Influence [1]
- NetShield [2]

[1] Morone, F, et al. "Collective influence algorithm to find influencers via optimal percolation in massively large social media." Scientific Reports 6 (2016): 30062.
[2] Chen, C, et al. "Node immunization on large graphs: Theory and algorithms." TKDE 28.1 (2015): 113-126.

Node Immunization

Given a graph **G**,

1. **Rank the nodes**
2. Pick the first one
3. Remove it (*immunize* it)
4. Repeat

Different ways of **ranking the nodes**:

- Degree (number of edges)
- PageRank
- Eigenvector centrality
- Coreness (k-core index)
- Collective Influence [1]
- NetShield [2]
- **This talk**

[1] Morone, F, et al. "Collective influence algorithm to find influencers via optimal percolation in massively large social media." Scientific Reports 6 (2016): 30062.
[2] Chen, C, et al. "Node immunization on large graphs: Theory and algorithms." TKDE 28.1 (2015): 113-126.

Road Map

1. ✓ Disease spreads over networks
2. ✓ The network's structure determines the spread
3. ✓ Change the structure by removing (immunizing) nodes
4. ✓ Many different ways of doing it
5. **Node immunization with non-backtracking eigenvalues**
6. R_0 and the epidemic threshold
7. Non-backtracking matrix
8. Eigenvalue perturbation
9. Experiments

Epidemic Threshold

"This [...] allows to define the concept of ***epidemic threshold***: only if $R_0 > 1$ (i.e. if a single infected individual generates on average more than one secondary infection), an infective agent can cause an outbreak of [substantial] size [...]. If $R_0 < 1$ (i.e. if a single infected individual generates less than one secondary infection), the relative size of the epidemic is negligibly small, [...]."

Pastor-Satorras, Romualdo, et al. "Epidemic processes in complex networks." Reviews of modern physics 87.3 (2015): 925.

$R_0 > 1$ ⇨ Possibility of a substantial outbreak

$R_0 < 1$ ⇨ Most likely no outbreak

These are the **classical definitions**, valid for a setting where **anyone can contact anyone else**.

Epidemic Threshold on Networks

In the case of networks, the ***epidemic threshold*** depends on the network structure.

$$R_0 > \theta \quad \Rightarrow \quad \text{Possibility of a substantial outbreak}$$

$$R_0 < \theta \quad \Rightarrow \quad \text{Most likely no outbreak}$$

Epidemic Threshold on Networks

In the case of networks, the ***epidemic threshold*** depends on the network structure.

$$R_0 > \boxed{\theta} \quad \Longrightarrow \quad \text{Possibility of a substantial outbreak}$$

$$R_0 < \boxed{\theta} \quad \Longrightarrow \quad \text{Most likely no outbreak}$$

$$\theta \approx 1/\tau$$

τ: leading eigenvalue of the **adjacency** matrix

Chakrabarti, Deepayan, et al. "Epidemic thresholds in real networks." ACM Transactions on Information and System Security (TISSEC) 10.4 (2008): 1-26.

Epidemic Threshold on Networks

In the case of networks, the ***epidemic threshold*** depends on the network structure.

$$R_0 > \theta \quad \Longrightarrow \quad \text{Possibility of a substantial outbreak}$$

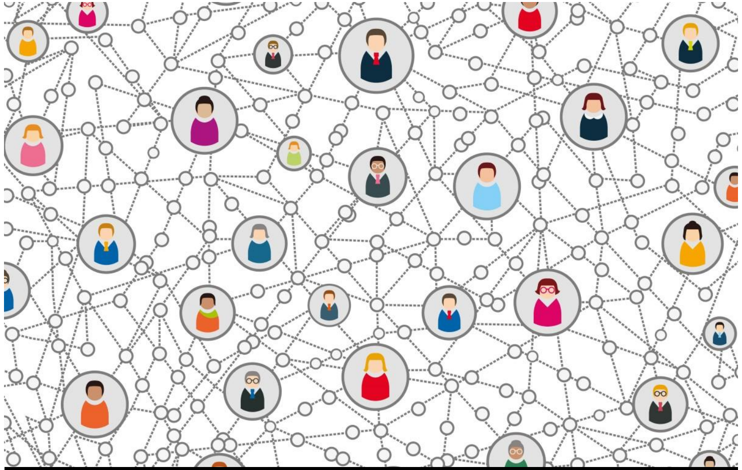
$$R_0 < \theta \quad \Longrightarrow \quad \text{Most likely no outbreak}$$

$$\theta \approx 1/\tau$$ leading eigenvalue of the **adjacency** matrix

$$\theta \approx 1/\lambda$$ leading eigenvalue of the **non-backtracking** matrix

Karrer, Brian, Mark EJ Newman, and Lenka Zdeborová. "Percolation on sparse networks." Physical review letters 113.20 (2014): 208702.

The research question



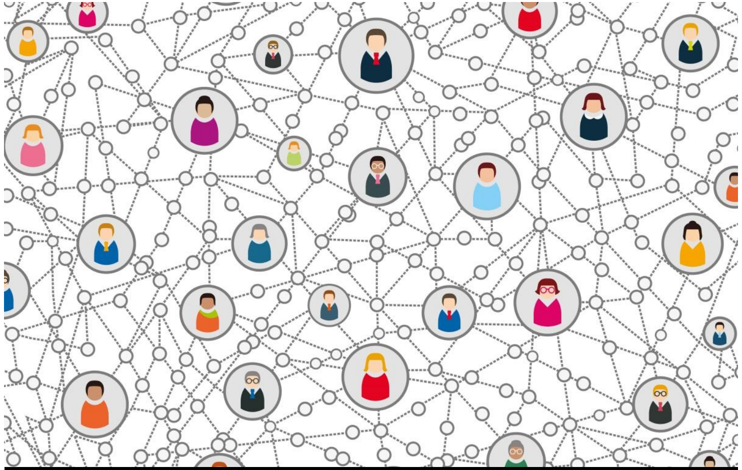
https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

Which nodes should we remove in order to **slow the spread?**

⇓

Which nodes should we remove in order to **increase the epidemic threshold?**

The research question


https://physicsworld.com/a/speed-of-spreading-epidemics-is-predicted-using-analytical-technique/

Which nodes should we remove in order to **slow the spread?**

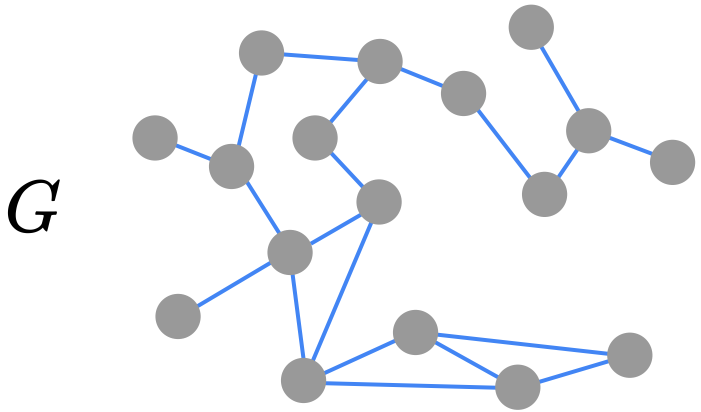
⇩

Which nodes should we remove in order to **increase the epidemic threshold?**

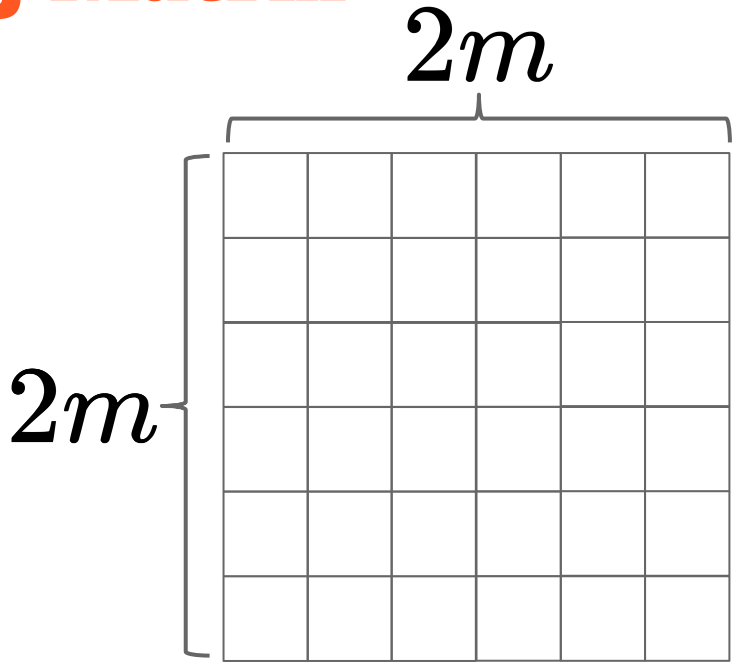
⇩

Which nodes should we remove in order to **decrease the leading eigenvalue?**

The Non-backtracking Matrix



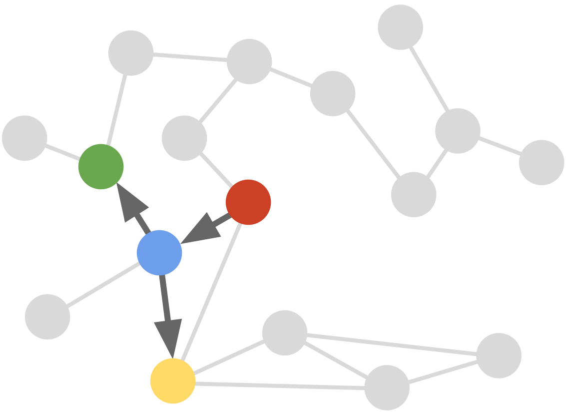
$G = (V, E)$
$|E| = m$



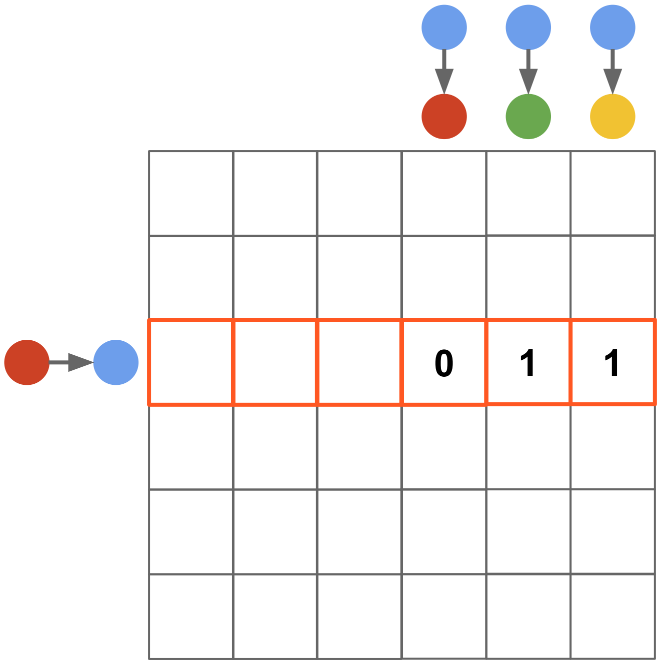
B

The Non-backtracking Matrix

G



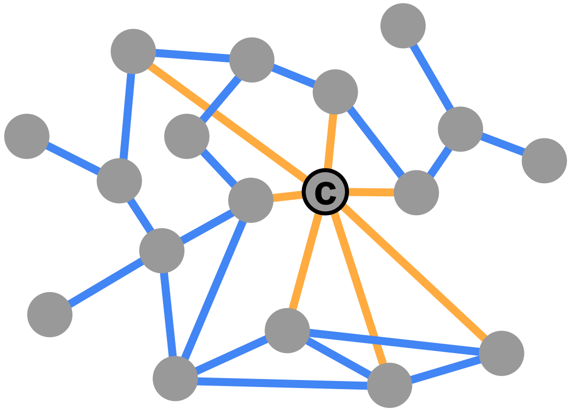
$$G = (V, E)$$
$$|E| = m$$



B

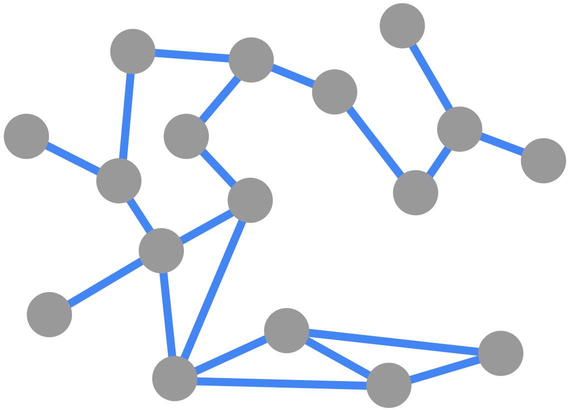
Some notation

G^c



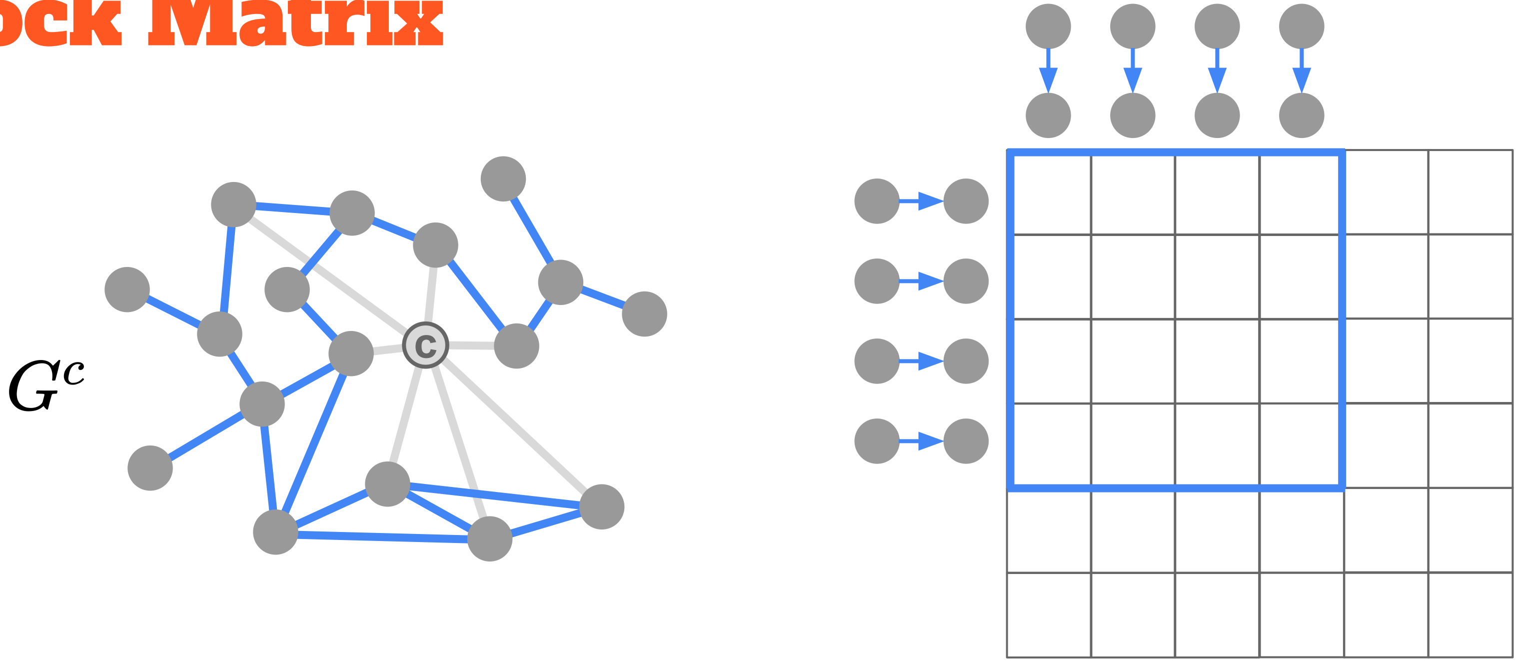
B^c, λ_1^c

G

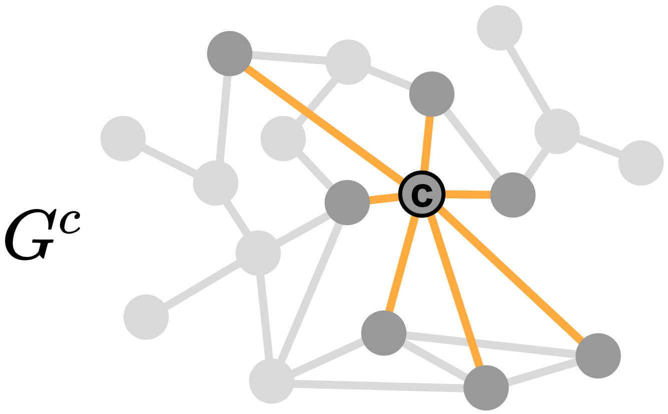


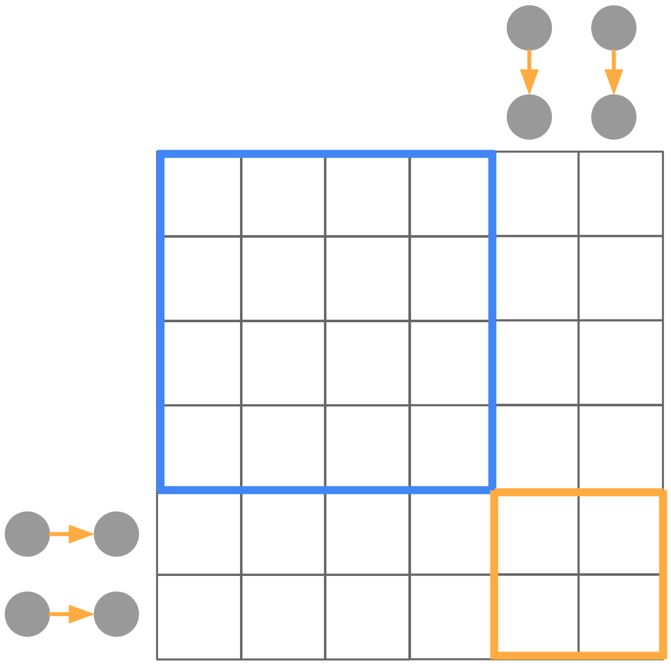
B, λ_1

Block Matrix

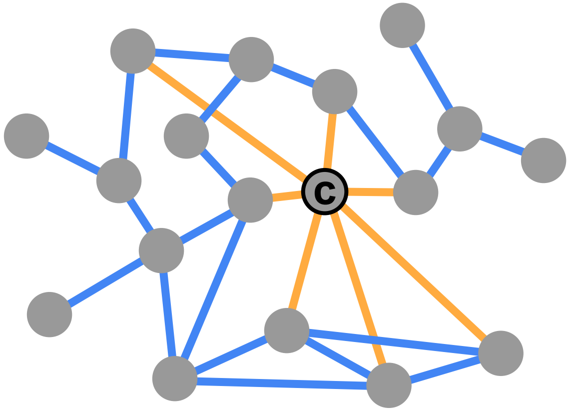


Block Matrix





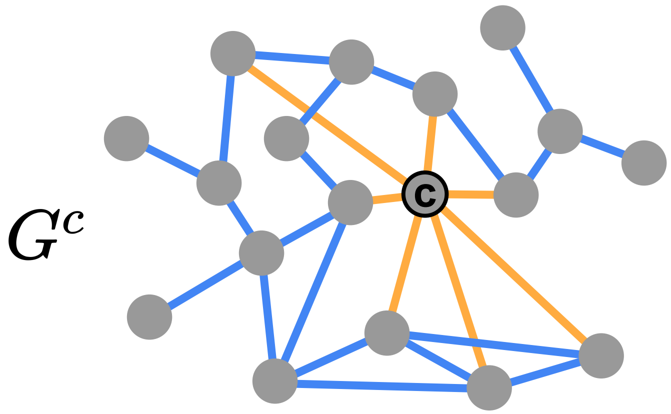
Block Matrix

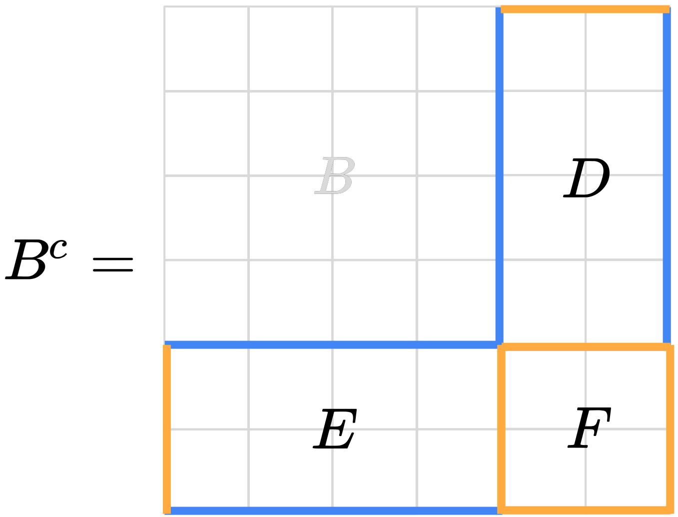


G^c

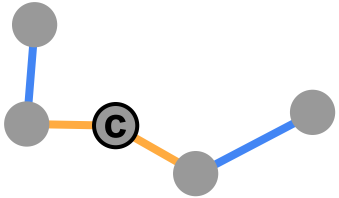
$$B^c = \begin{bmatrix} B & D \\ E & F \end{bmatrix}$$

The X Matrix

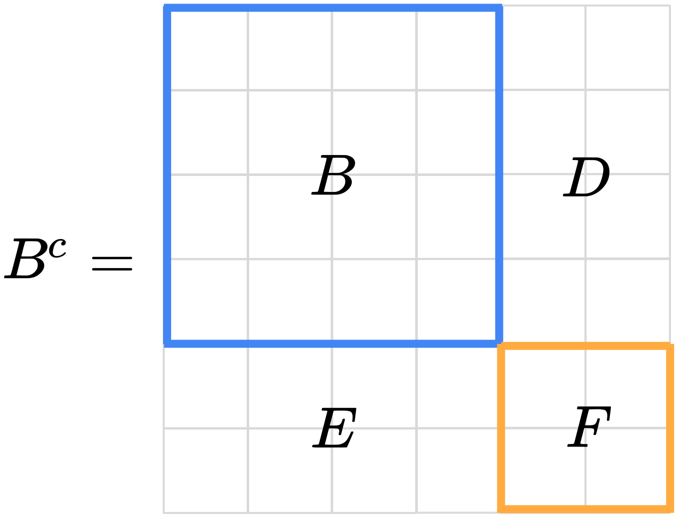




$X = DFE$



Solving for eigenvalues

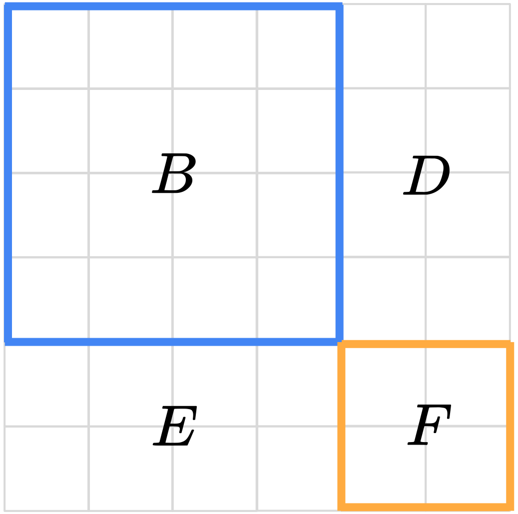


Solve:

$$\det\left(B^c - tI\right) = 0$$

Solving for eigenvalues

$B^c =$



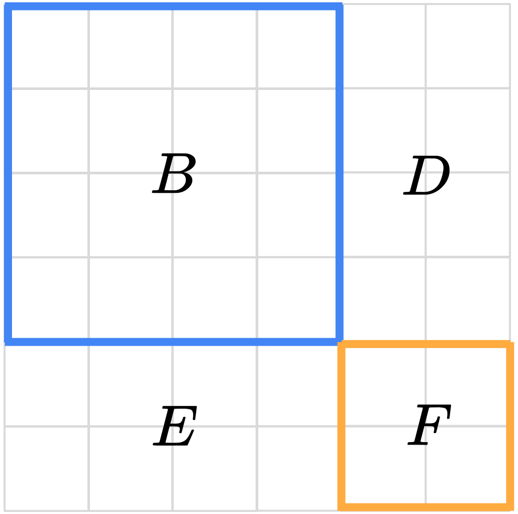
Solve:

$$\det\left(B^c - tI\right) = 0$$

$$\det\left(B^c - tI\right) = t^{2d} \det\left(B - tI\right) \det\left(I + \frac{YX}{t^2}\right) \qquad Y = \left(B - tI\right)^{-1}$$

First Approximation

$B^c =$



Solve:

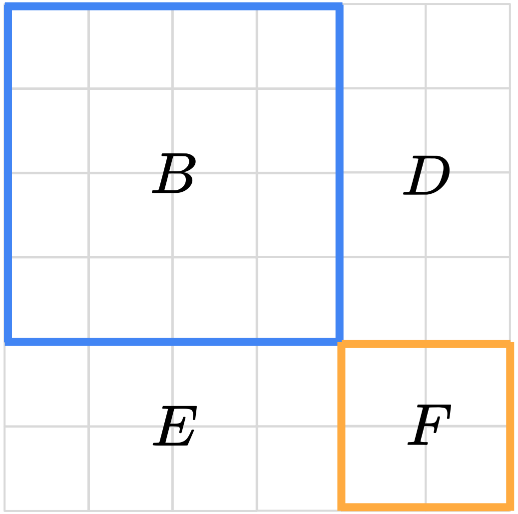
$$\det\left(B^c - tI\right) = 0$$

$$\det\left(B^c - tI\right) = t^{2d}\det\left(B - tI\right)\det\left(I + \frac{YX}{t^2}\right)$$

$$Y = \left(B - tI\right)^{-1}$$

First Approximation

$B^c =$



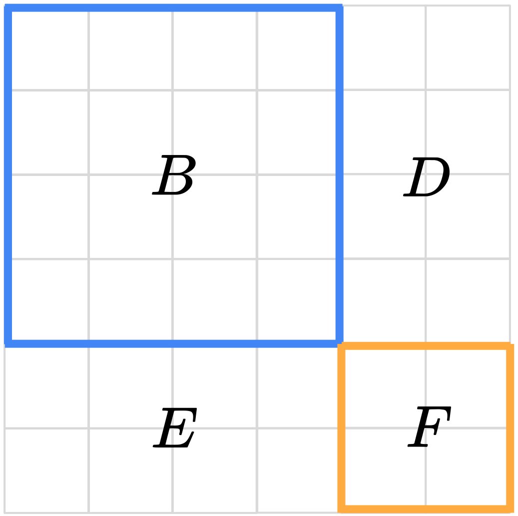
Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

First Approximation

$B^c =$



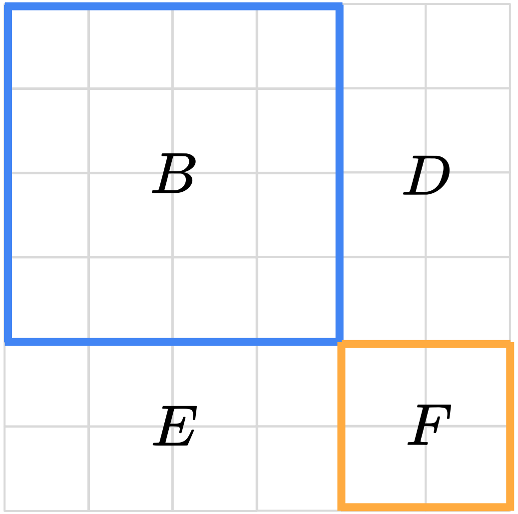
$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2} Tr\,(YX) + \ldots$$

Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

First Approximation

$B^c =$ 

$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2} Tr\left(YX\right) + \ldots$$

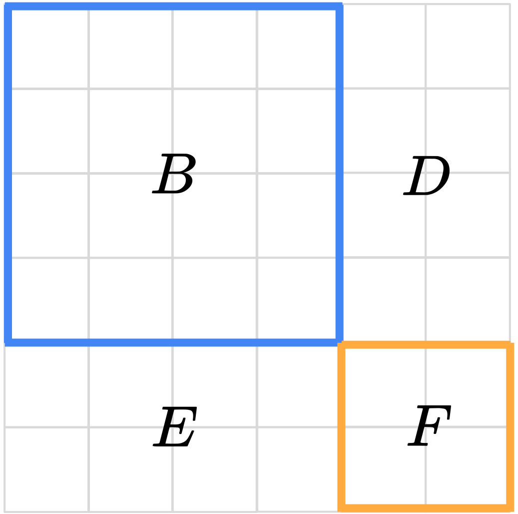
Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

First Approximation

$B^c =$



Solve:

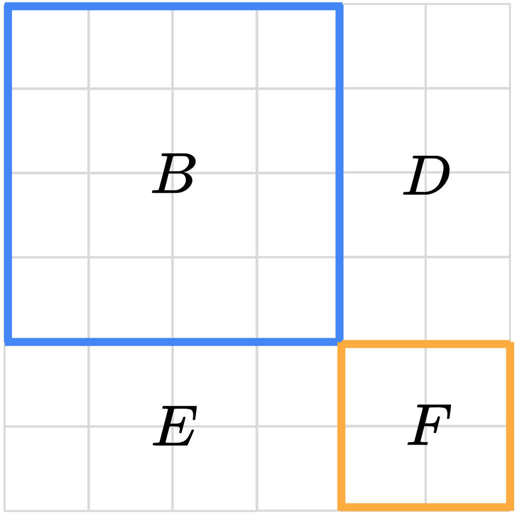
$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2}\sum_i \frac{v_i^T X u_i}{t-\lambda_i} + \dots$$

Second Approximation

$B^c =$



$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2}\sum_i \frac{v_i^T X u_i}{t-\lambda_i} + \dots$$

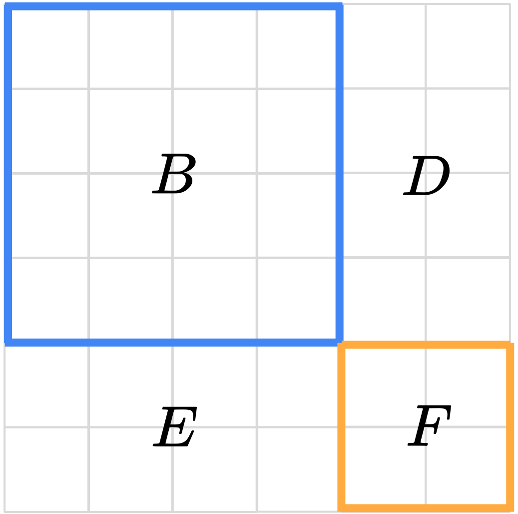
Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

Solving for eigenvalues

$B^c =$



$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2}\frac{v_1^T X u_1}{t - \lambda_1} + \ldots$$

u_1 and v_1 are the **left and right eigenvectors** of the **leading eigenvalue λ_1**

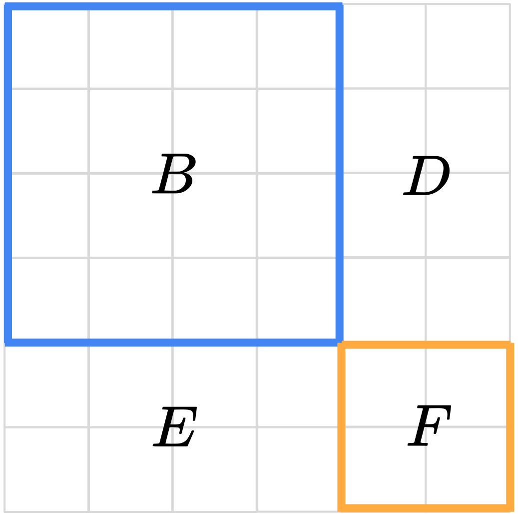
Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

$$X = DFE$$
$$Y = (B - tI)^{-1}$$

Solving for eigenvalues

$B^c =$



Solve:

$$\det\left(I + \frac{YX}{t^2}\right) = 0$$

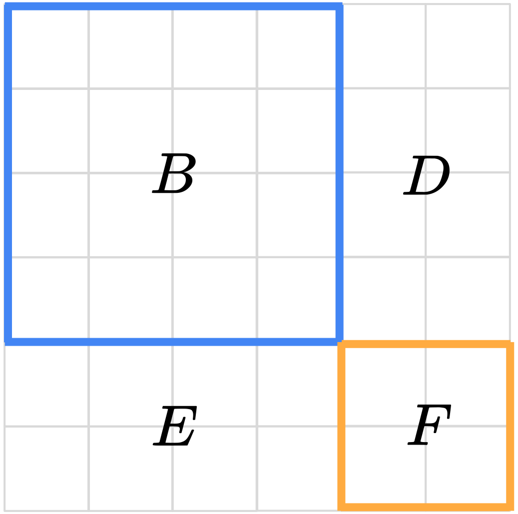
$$X = DFE$$
$$Y = (B - tI)^{-1}$$

$$\det\left(I + \frac{YX}{t^2}\right) = 1 + \frac{1}{t^2}\frac{v_1^T X u_1}{t - \lambda_1} \iff t^2(t - \lambda_1) + v_1^T X u_1 = 0$$

u_1 and v_1 are the **left and right eigenvectors** of the **leading eigenvalue λ_1**

Third Approximation

$B^c =$

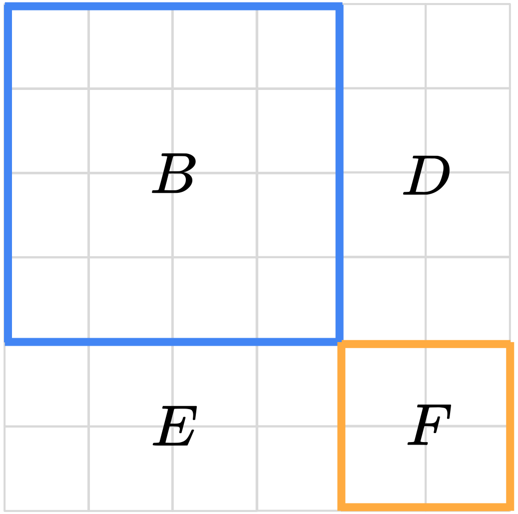


$$t^2(t - \lambda_1) + v_1^T X u_1 = 0 \qquad X = DFE$$

u_1 and v_1 are the **left and right eigenvectors** of the **leading eigenvalue λ_1**

Third Approximation

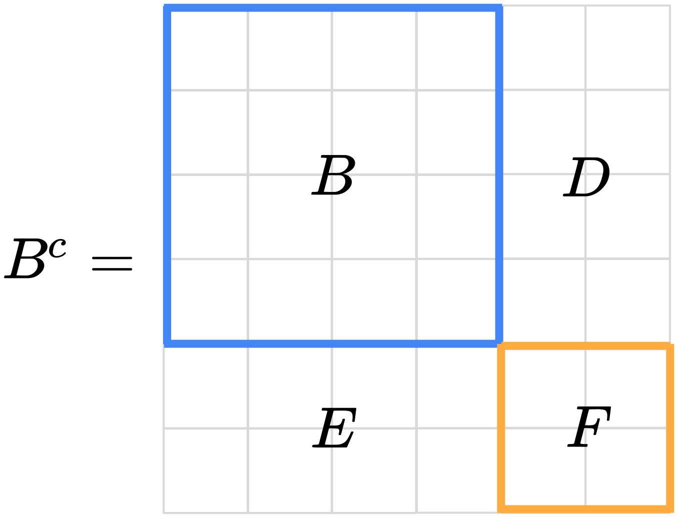
$B^c =$



$$t^2(t-\lambda_1) + v_1^T X u_1 = 0 \qquad X = DFE$$

u_1 and v_1 are the **left and right eigenvectors** of the **leading eigenvalue λ_1**

XNB Centrality



X-Non-backtracking centrality (XNB)

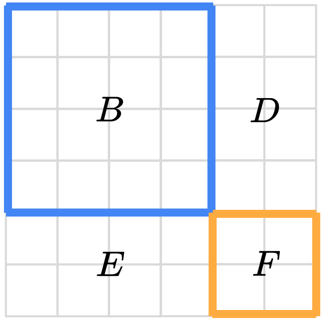
$$t^2(t-\lambda_1) + v_1^T X u_1 = 0 \qquad X = DFE$$

u_1 and v_1 are the **left and right eigenvectors** of the **leading eigenvalue λ_1**

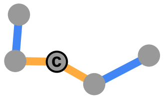
XNB for different target nodes

1. Choose a target node **c**

$B^c =$



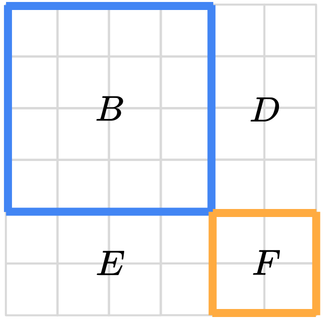
$X_c = DFE$



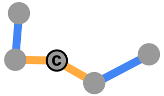
XNB for different target nodes

1. Choose a target node **c**

$B^c =$



$X_c = DFE$



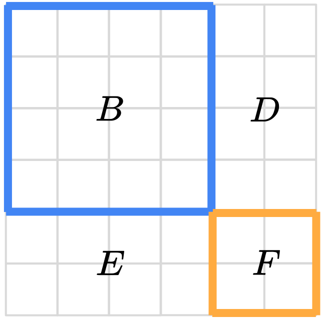
2. Compute u_1, v_1, and XNB

$$XNB(c) = v_1^T X_c u_1$$

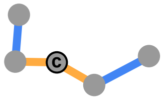
XNB for different target nodes

1. Choose a target node **c**

$B^c =$



$X_c = DFE$



2. Compute u_1, v_1, and XNB

$$XNB(c) = v_1^T X_c u_1$$

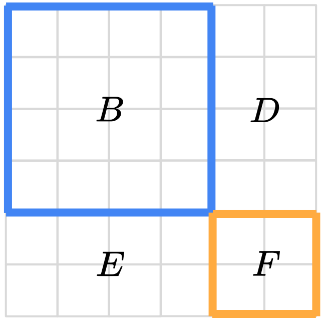
3. Alternative way

$$XNB(c) = \left(\sum_i a_{ci} v_1^i\right)^2 - \sum_i a_{ci} \left(v_1^i\right)^2$$

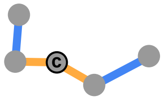
Xdeg for different target nodes

1. Choose a target node **c**

$B^c =$



$X_c = DFE$



2. Compute u_1, v_1, and XNB

$$XNB(c) = v_1^T X_c u_1$$

$$Xdeg(c) = 1^T X_c 1$$

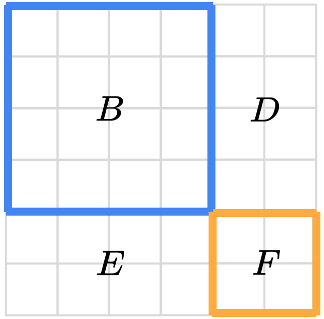
3. Alternative way

$$XNB(c) = \left(\sum_i a_{ci} v_1^i\right)^2 - \sum_i a_{ci} \left(v_1^i\right)^2$$

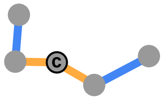
Xdeg for different target nodes

1. Choose a target node **c**

$B^c =$



$X_c = DFE$



2. Compute u_1, v_1, and XNB

$$XNB(c) = v_1^T X_c u_1$$

$$Xdeg(c) = 1^T X_c 1$$

3. Alternative way

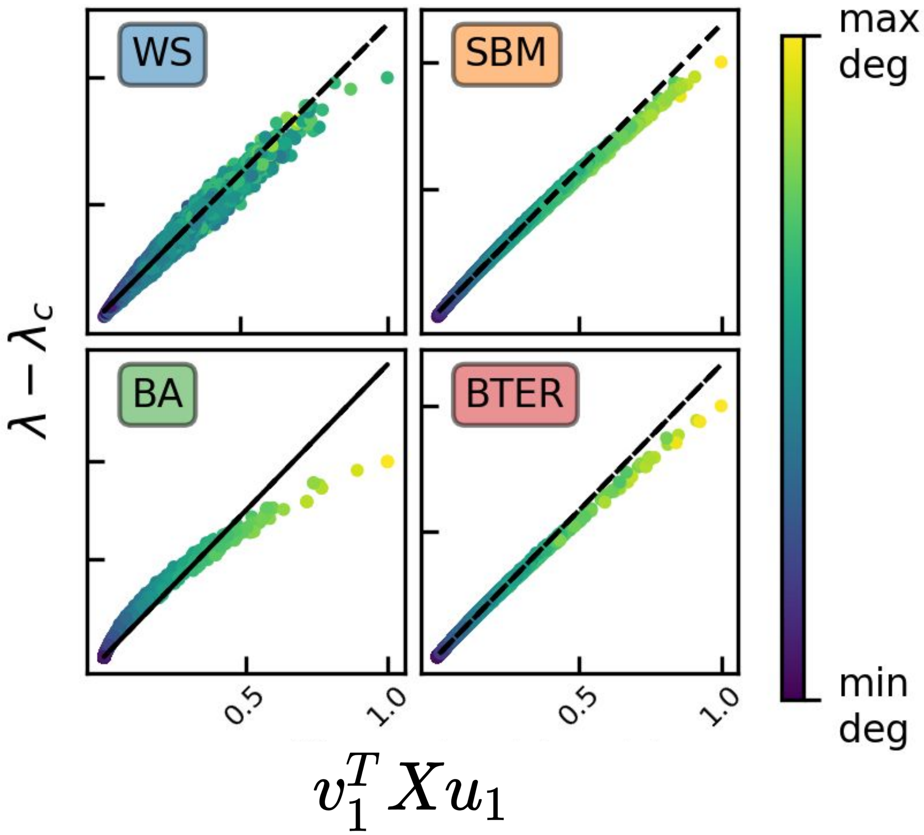
$$XNB(c) = \left(\sum_i a_{ci} v_1^i\right)^2 - \sum_i a_{ci} \left(v_1^i\right)^2$$

v_1^i is the NB-centrality

$$Xdeg(c) = \left(\sum_i a_{ci} d_i'\right)^2 - \sum_i a_{ci} \left(d_i'\right)^2$$

$$d_i' = deg(i) - 1$$

XNB and the true change in eigenvalue



The XNB centrality is **highly correlated** to the change in the leading eigenvalue, and therefore to the change in the epidemic threshold.

Ranking nodes by their XNB values is sure to produce an effective node immunization strategy.

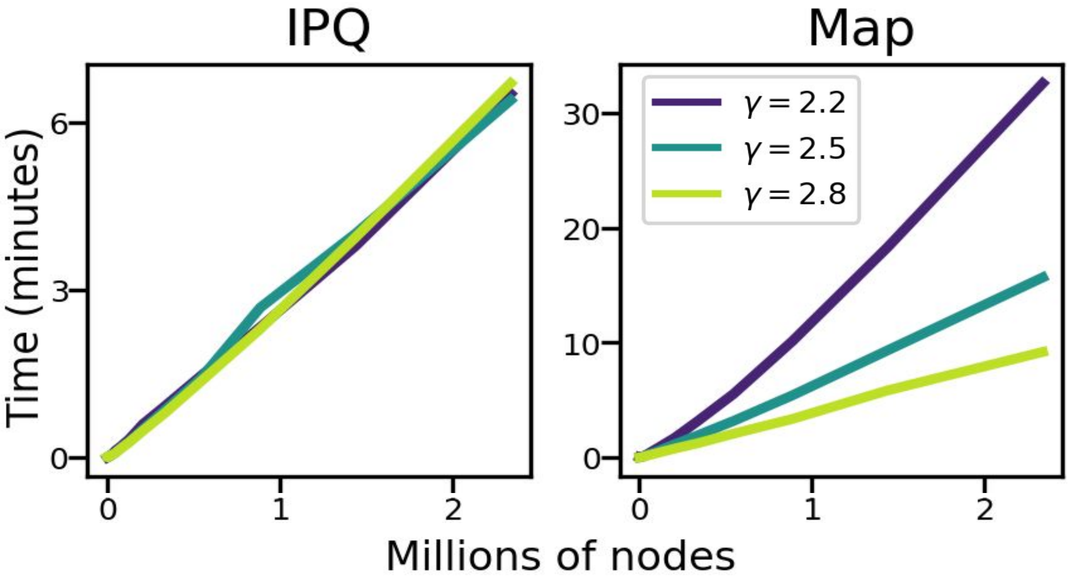
Algorithm: Immunization with XNB

Input: graph G, integer p
Output: removed, an ordered list of nodes

```
removed ← ∅
XNB [i] ← XCent(G,i) for each node i
while length(removed) < p do
    node ← max_i XNB [i]
    foreach i in G.neighbors[node] do
        G.neighbors[i].remove(node)
    foreach i in G.neighbors[node] do
        foreach j in G.neighbors[i] do
            XNB [j] ← XCent(G,i)
    G.neighbors[node] ← ∅
    removed.append(node)
return removed
```

This algorithm can be implemented using one of two data structures: **an indexed priority queue (IPQ), or a hash table (a.k.a. dictionary, Map)**. Each version is more efficient on **different types of networks.**

Algorithm: Scalability



Immunization on graphs with **heterogeneous degree distribution** (e.g. a few large hubs). Smaller γ means larger hubs. Real graphs typically have $2 < \gamma < 3$.

Algorithm: Baselines

		degree	NS	CI	Xdeg	NB	XNB
BA	1%	62.76	61.44	62.88	62.90	62.92	62.91
	2%	68.84	66.94	68.97	68.99	69.01	69.01
	3%	72.42	70.09	72.56	72.57	72.59	72.59
BTER	1%	6.28	6.40	6.41	6.45	6.46	6.46
	2%	10.60	10.72	10.80	10.85	10.86	10.86
	3%	14.31	14.40	14.55	14.61	14.63	14.63
SBM	1%	3.31	3.41	3.40	3.43	3.44	3.44
	2%	6.00	6.16	6.19	6.23	6.25	6.25
	3%	8.52	8.66	8.76	8.80	8.82	8.82
WS	1%	1.41	1.17	1.50	1.52	1.63	1.63
	2%	2.52	2.09	2.97	2.98	3.11	3.11
	3%	3.66	2.94	4.41	4.41	4.57	4.58

Table 1: Average percentage eigen-drop (larger is better) on synthetic graphs after removing 1%, 2%, and 3% of the nodes using different strategies.

	$p = 1$			$p = 10$			$p = 100$		
	degree	CI	Xdeg	degree	CI	Xdeg	degree	CI	Xdeg
AS-1	0.74	0.74	**2.35**	6.70	13.51	**15.43**	71.65	**78.26**	75.92
AS-2	2.02	2.02	**4.00**	17.09	22.36	**28.17**	87.60	**89.61**	87.02
Social-Slashdot	0.95	**1.02**	**1.02**	4.63	6.06	**6.94**	23.65	28.11	**30.30**
Social-Twitter	**2.18**	**2.18**	1.98	13.21	**13.97**	13.68	41.10	42.88	**43.39**
Transport-California	0.00	0.00	**0.65**	**2.65**	0.65	**2.65**	5.09	5.09	**7.80**
Transport-Sydney	**0.00**	**0.00**	**0.00**	0.00	0.00	**6.50**	0.00	7.37	**9.49**
Web-NotreDame	**9.34**	**9.34**	**9.34**	12.10	**13.79**	**13.79**	14.37	14.37	**19.22**

Table 2: Average percentage eigen-drop on real networks (larger is better) when removing $p = 1, 10$, or 100 nodes. Xdeg is effective and has log-linear time in the number of nodes. Details about the sizes of these datasets are in Table 3 of the appendix.

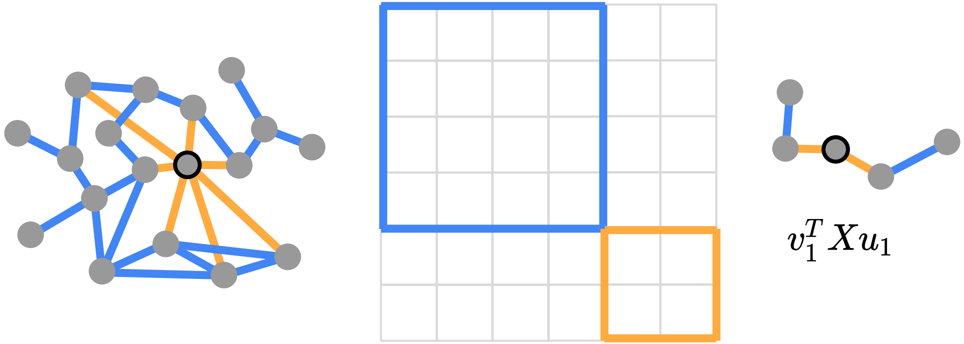
	$p = 1$			$p = 10$			$p = 100$		
	degree	CI	Xdeg	degree	CI	Xdeg	degree	CI	Xdeg
AS-1	0.74	0.74	**2.35**	6.70	13.51	**15.43**	71.65	**78.26**	75.92
AS-2	2.02	2.02	**4.00**	17.09	22.36	**28.17**	87.60	**89.61**	87.02
Social-Slashdot	0.95	**1.02**	**1.02**	4.63	6.06	**6.94**	23.65	28.11	**30.30**
Social-Twitter	**2.18**	**2.18**	1.98	13.21	**13.97**	13.68	41.10	42.88	**43.39**
Transport-California	0.00	0.00	**0.65**	**2.65**	0.65	**2.65**	5.09	5.09	**7.80**
Transport-Sydney	**0.00**	**0.00**	**0.00**	0.00	0.00	**6.50**	0.00	7.37	**9.49**
Web-NotreDame	**9.34**	**9.34**	**9.34**	12.10	**13.79**	**13.79**	14.37	14.37	**19.22**

Table 2: Average percentage eigen-drop on real networks (larger is better) when removing $p = 1, 10$, or 100 nodes. Xdeg is effective and has log-linear time in the number of nodes. Details about the sizes of these datasets are in Table 3 of the appendix.

	$p=1$			$p=10$			$p=100$		
	degree	CI	Xdeg	degree	CI	Xdeg	degree	CI	Xdeg
AS-1	0.74	0.74	**2.35**	6.70	13.51	**15.43**	71.65	**78.26**	75.92
AS-2	2.02	2.02	**4.00**	17.09	22.36	**28.17**	87.60	**89.61**	87.02
Social-Slashdot	0.95	**1.02**	**1.02**	4.63	6.06	**6.94**	23.65	28.11	**30.30**
Social-Twitter	**2.18**	**2.18**	1.98	13.21	**13.97**	13.68	41.10	42.88	**43.39**
Transport-California	0.00	0.00	**0.65**	**2.65**	0.65	**2.65**	5.09	5.09	**7.80**
Transport-Sydney	**0.00**	**0.00**	**0.00**	0.00	0.00	**6.50**	0.00	7.37	**9.49**
Web-NotreDame	**9.34**	**9.34**	**9.34**	12.10	**13.79**	**13.79**	14.37	14.37	**19.22**

Table 2: Average percentage eigen-drop on real networks (larger is better) when removing $p = 1, 10,$ or 100 nodes. Xdeg is effective and has log-linear time in the number of nodes. Details about the sizes of these datasets are in Table 3 of the appendix.

Gracias!



1. Diseases spread on networks!
2. Node immunization: remove node to slow the spread
3. Epidemic threshold ≈ leading non-backtracking eigenvalue
4. **XNB** is a great way to rank nodes for immunization

https://arxiv.org/abs/2002.12309

www.leotrs.com

leo@leotrs.com

@_leotrs