BOSTON UNIVERSITY

big data analytics Lab

Masked Mirror Validation in Graphon Estimation

Huimin Cheng

Boston University

Joint work with Yongkai Chen, Ping Ma, Wenxuan Zhong

Outline

1 Background

- Network, Graphon and Graphon Estimation
- Motivation and Challenges

2 Proposed Method: Masked Mirror Validation (MMV)

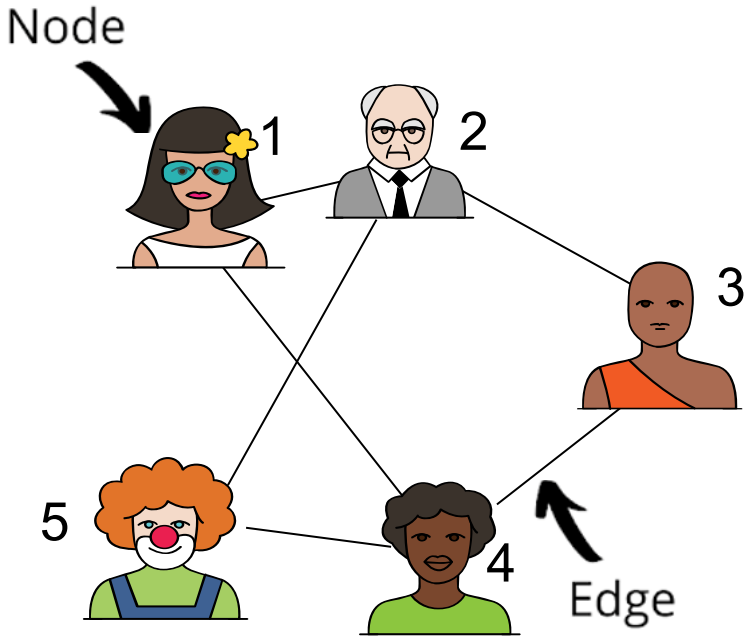
- MMV Procedure
- Theoretical Results
- Simulation Studies

3 Application to Drug Repurposing

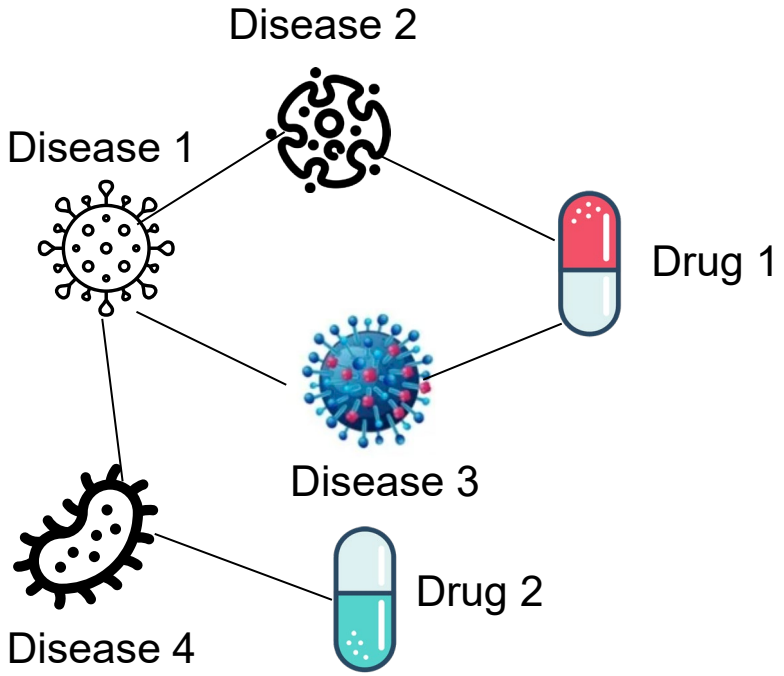
- Drug Repurposing
- Med-Reader AI Tool
- Case Study

Network (Graph)

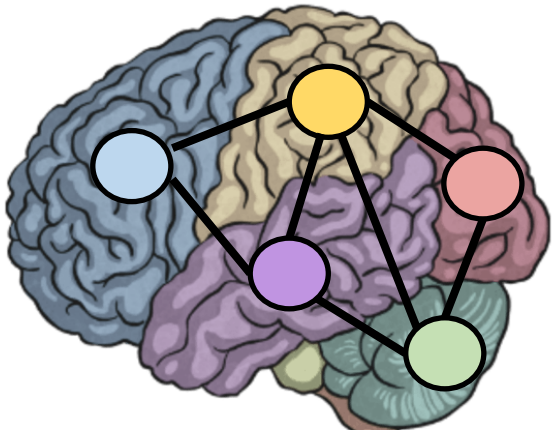
Social network



Drug-disease network



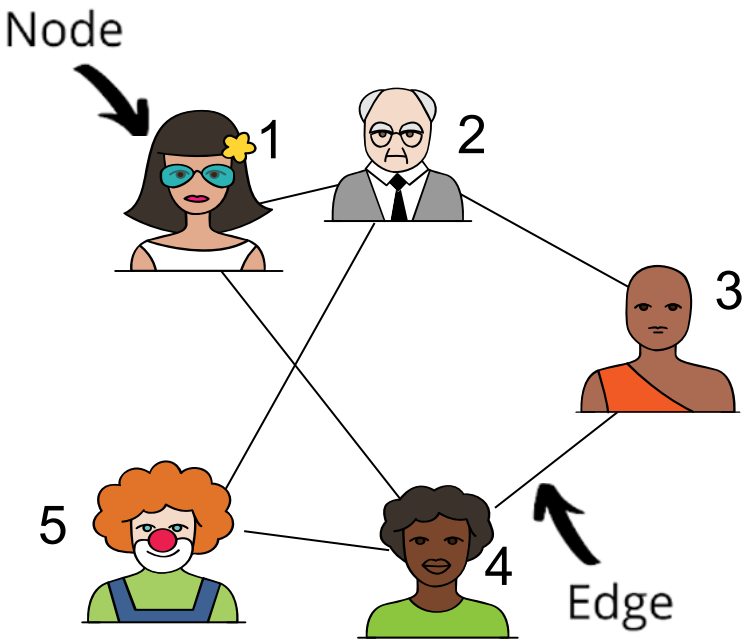
Brain network



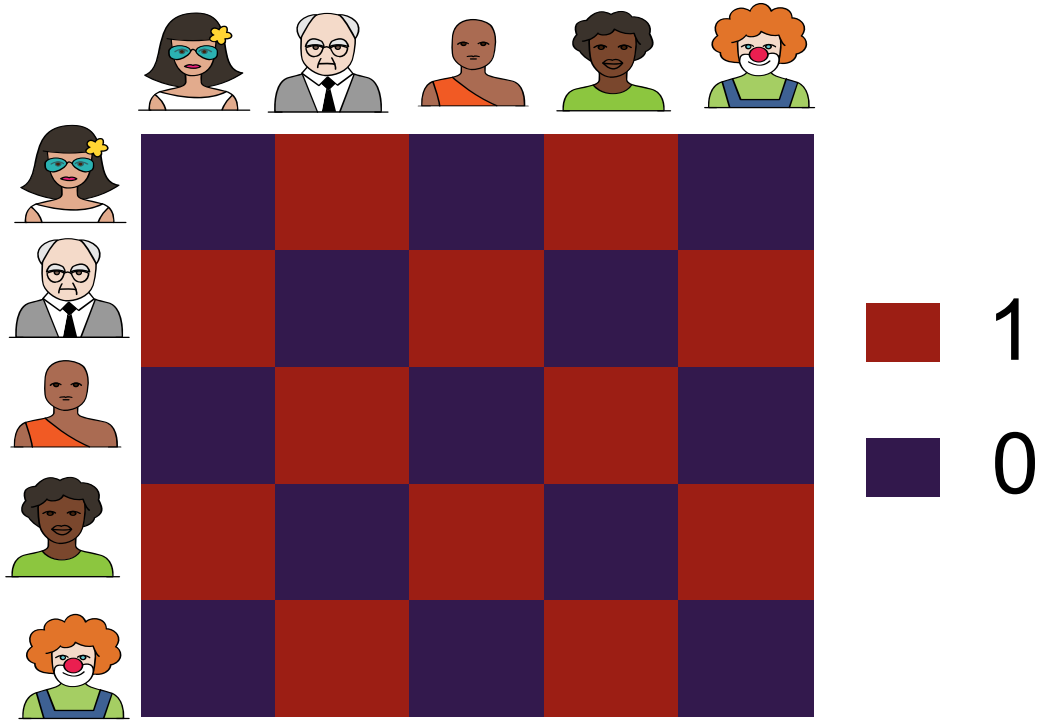
Picture source: https://www.twinkl.com/teaching-wiki/brain-for-kids

Mathematical Representation of a Network

Social network



Adjacency matrix $\mathbf{A} = (a_{ij})$

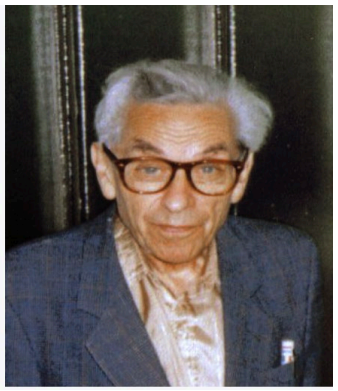


Generating Model for Network

1960

Erdős–Rényi model

Paul Erdős and Alfréd Rényi



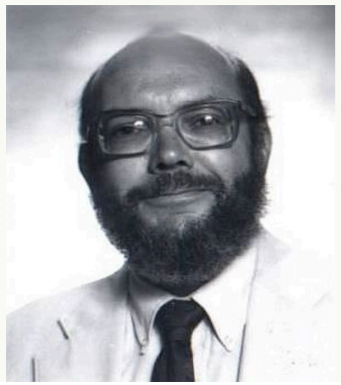
$$a_{ij} \sim \mathrm{Ber}(p)$$

Erdos, and Rényi. Publ. Math. Inst. Hung. Acad. Sci. 1960

1983

Stochastic Block Model (SBM)

Paul W. Holland



$a_{ij} \sim \mathrm{Ber}(p_{in})$, if i, j are in a same group

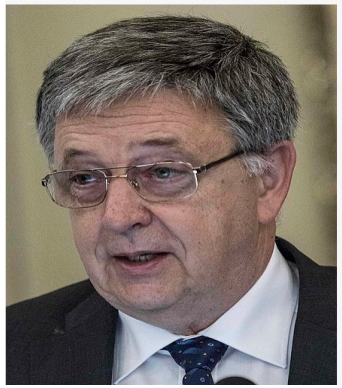
$a_{ij} \sim \mathrm{Ber}(p_{out})$, otherwise

Holland, Laskey and Leinhardt. Social networks 1983.

2006

Graphon model

László Lovász

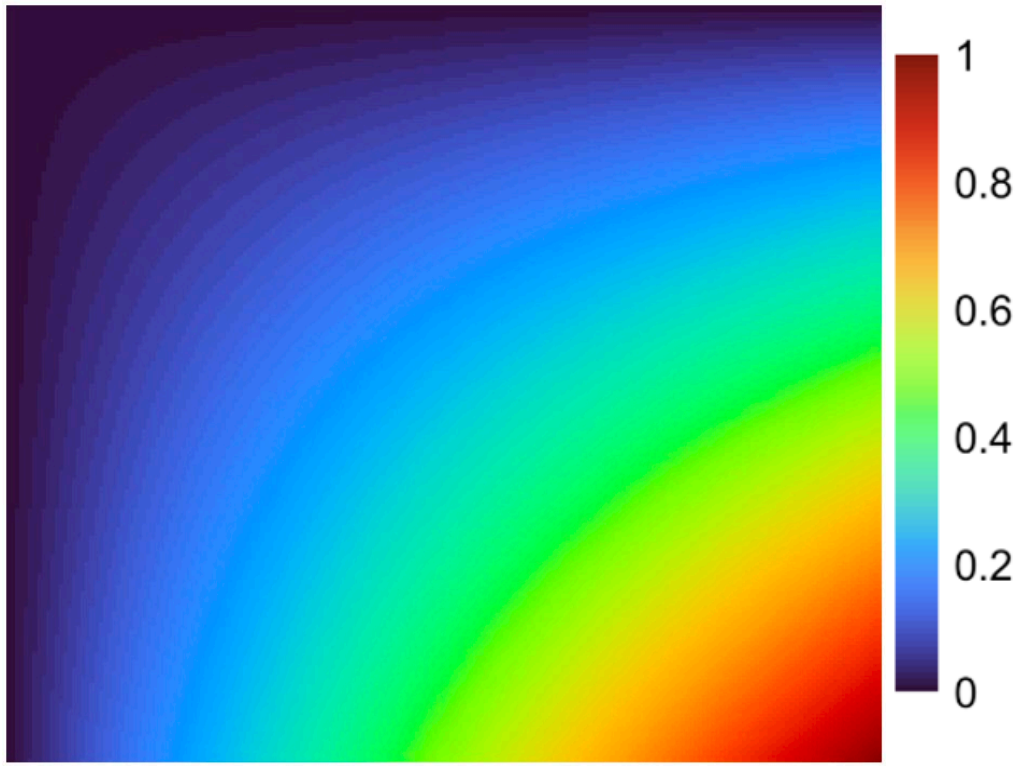


$$a_{ij} \sim \mathrm{Ber}(p_{ij}),$$
$$p_{ij} = f(u_i, u_j)$$
$$u_i \in [0,1],\ u_j \in [0,1]$$

Lovász, and Szegedy. Journal of Combinatorial Theory, Series B 2006

Heuristic of Probability Matrix

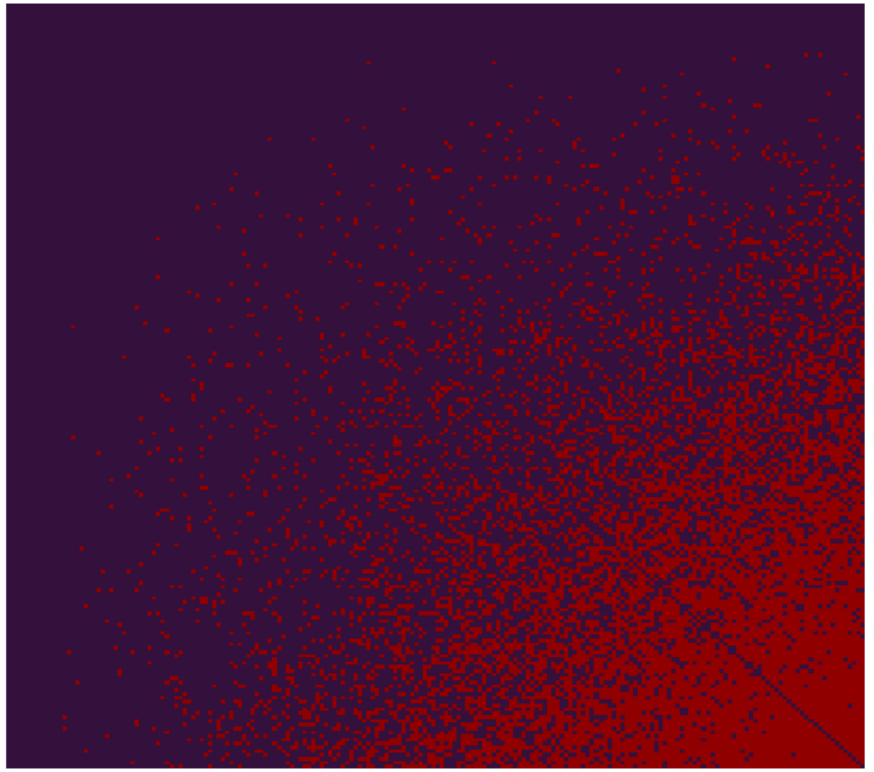
Graphon Probability matrix $\mathbf{P} = (p_{ij})$



Generate

Estimate

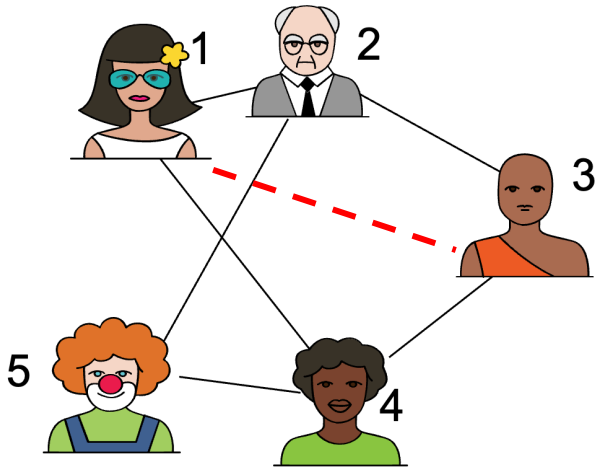
Adjacency matrix

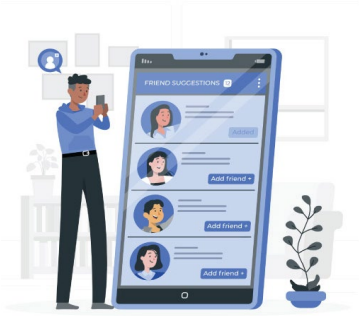


1 0

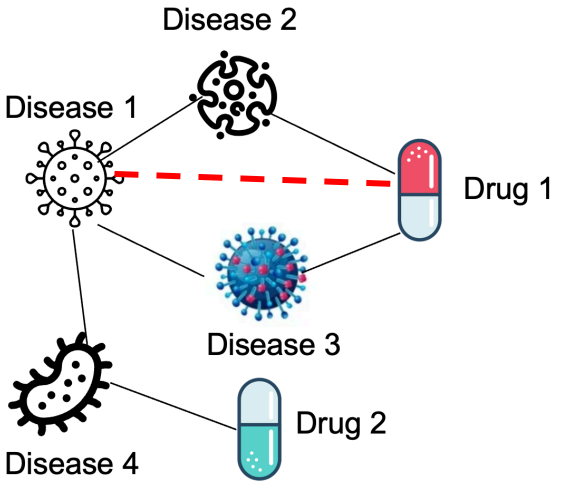
Why Care Graphon Estimation?

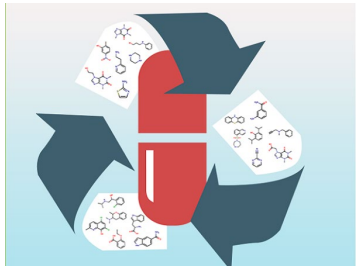
Friend Recommendation





Drug Repurposing

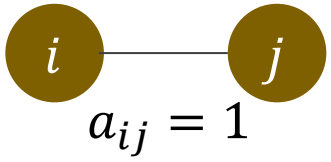




Graphon Estimation Methods

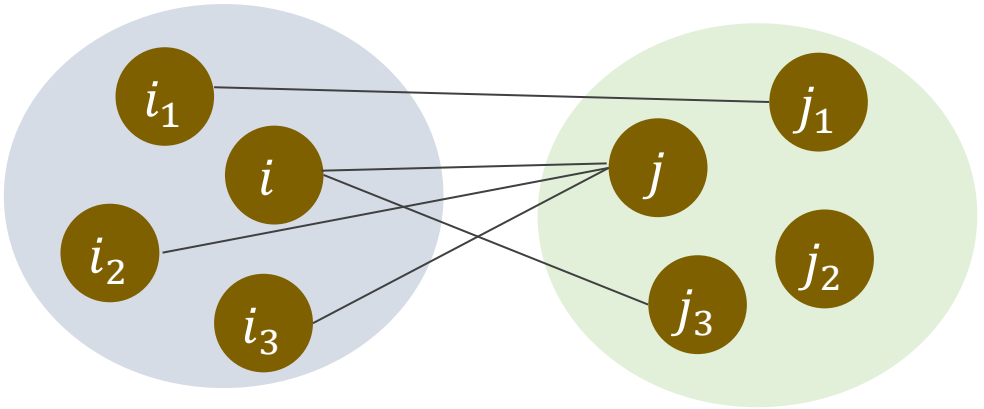
Sort-and-smoothing (SAS) method (Chan and Edoardo. ICML 2014) → Depends on a **hyperparameter:** Neighborhood size $\boldsymbol{h}$

Challenge



Only one observation for p_{ij}!

"Neighbors" of i "Neighbors" of j

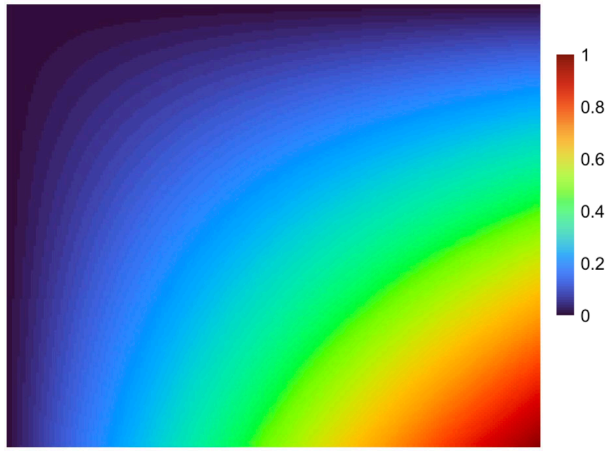


"Replicated observations" for p_{ij}

Neighborhood smoothing (NS) method (Zhang, Levina and Zhu. *Biometrika* 2017) → Depends on a **hyperparameter:** Neighborhood size $m\sqrt{nlogn}$

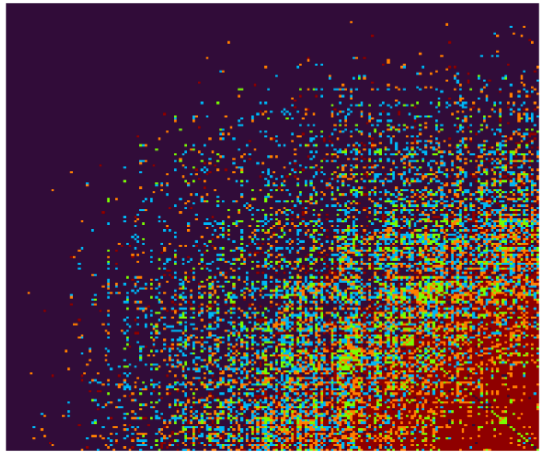
Hyperparameter Influences Graphon Estimation

True $\mathbf{P} = (p_{ij})$

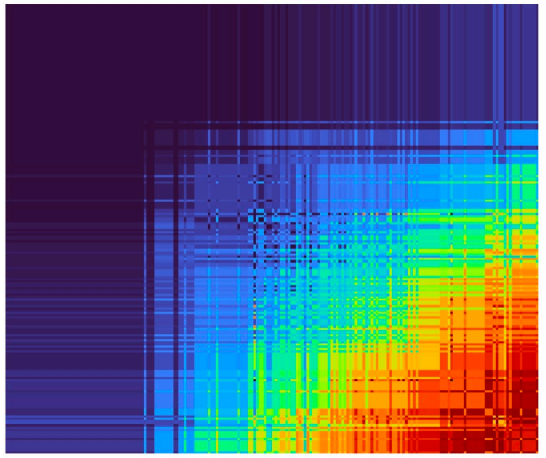


Estimator $\widehat{\boldsymbol{P}} = (\hat{p}_{ij})$

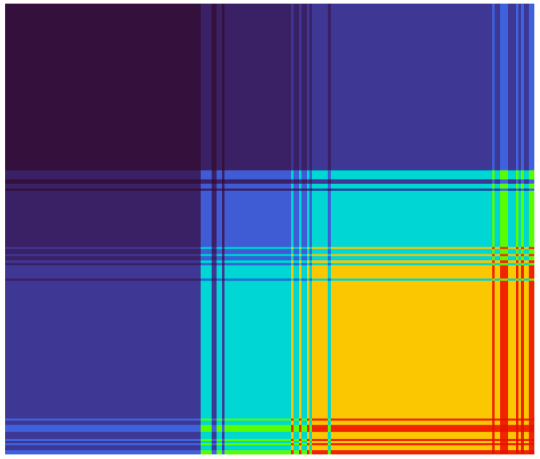
SAS $h = 2$



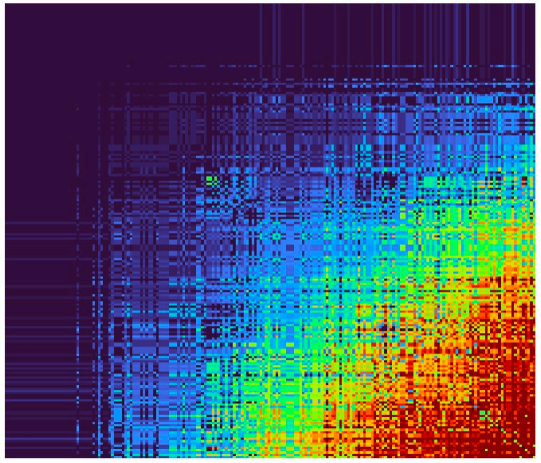
$h = 10$



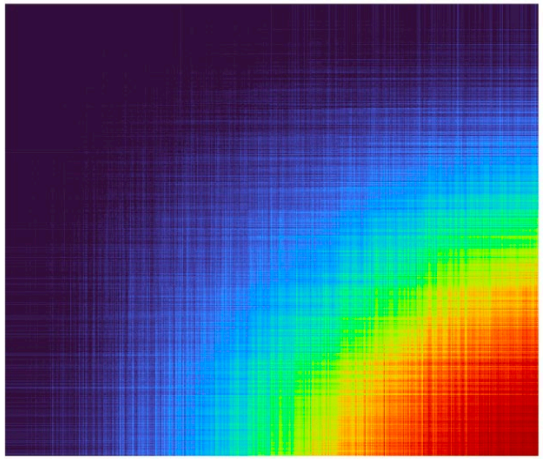
$h = 50$



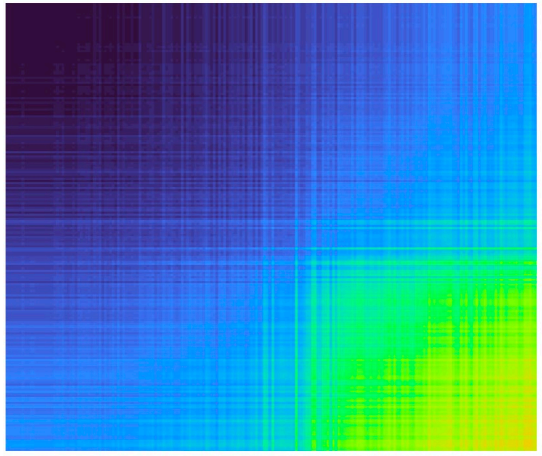
NS $m = 0.5$



$m = 2$



$m = 5$



Hyperparameter Tuning

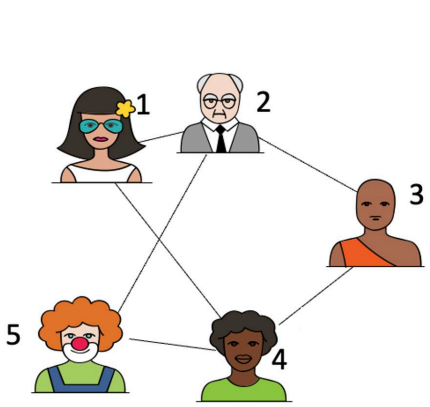
$$\mathrm{MSE}(m) = \frac{1}{n^2}\left\|\mathbf{P} - \widehat{\mathbf{P}}_{\boldsymbol{m}}\right\|_F^2$$

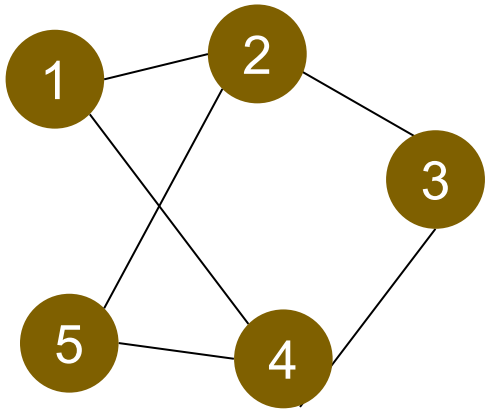
$$m^* = \mathrm{argmin}_{m \in M}\mathrm{MSE}(m)$$

Approaches	Pros and Cons
• Empirical experience	Lack of theoretical guarantee
• Analytic solution	Explicit form between $\widehat{\mathbf{P}}_{\boldsymbol{m}}$ and m is usually elusive
.....	
• Cross-validation	Theoretical guarantee; Effective

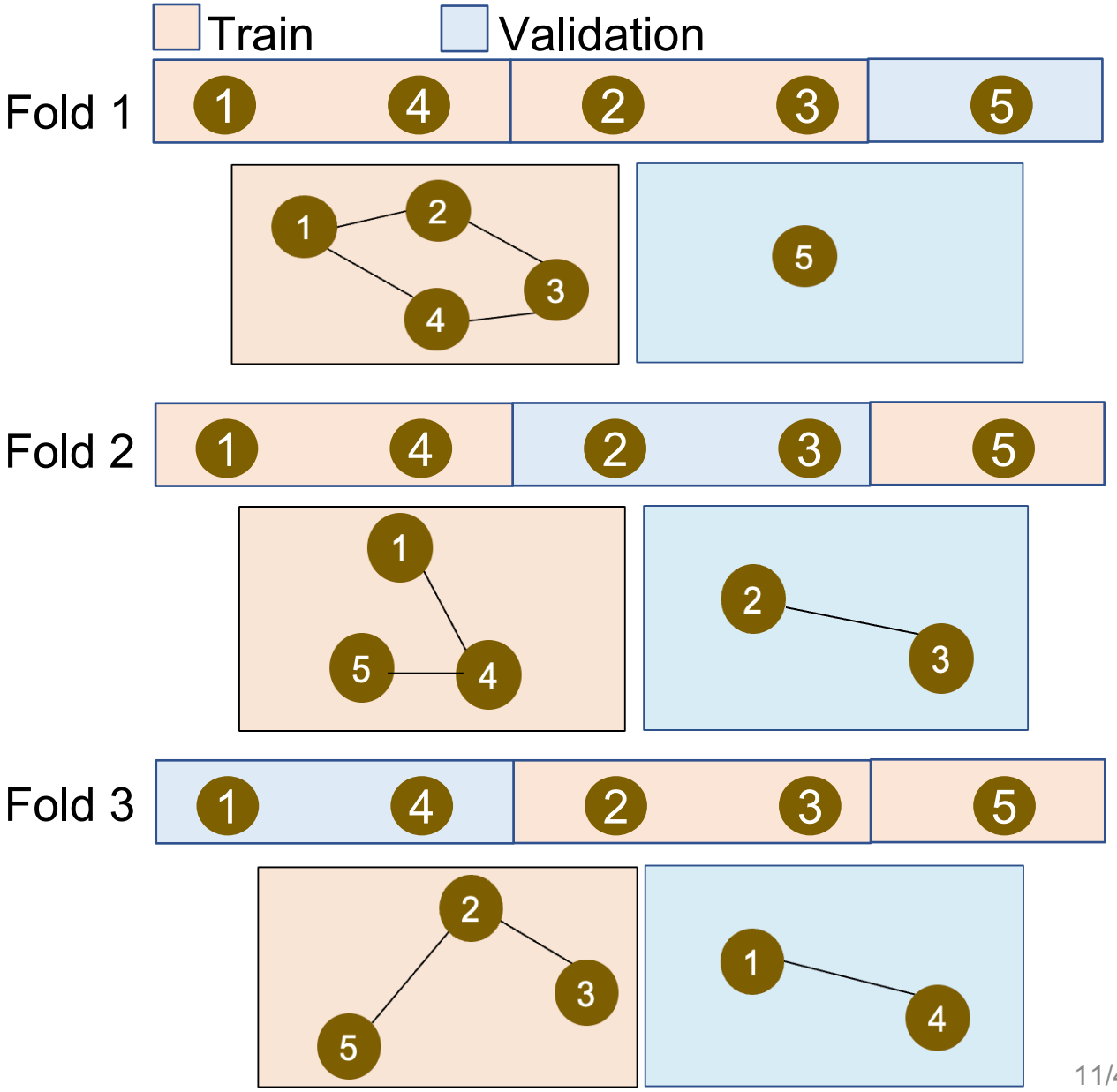
Problems of Node Splitting

Node splitting will **destroy the network structure!**

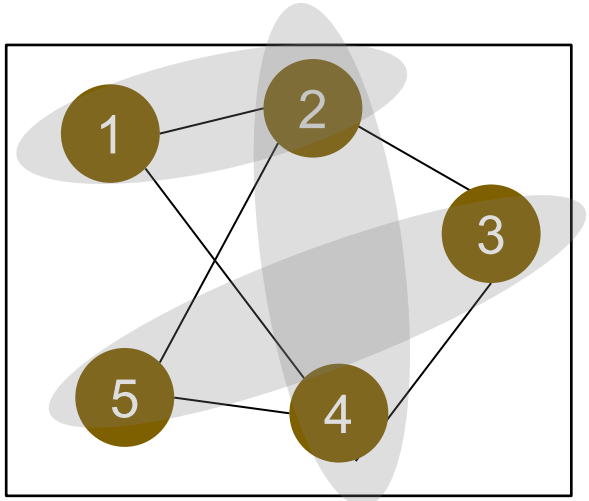




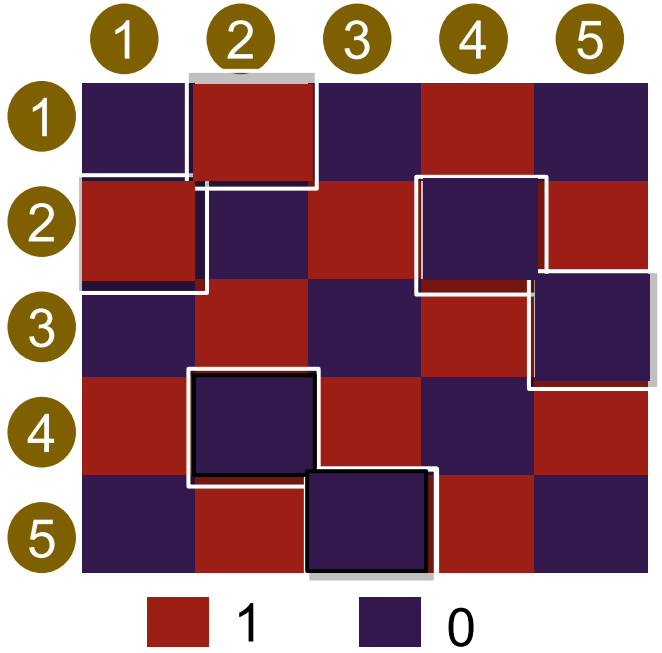
Three folds



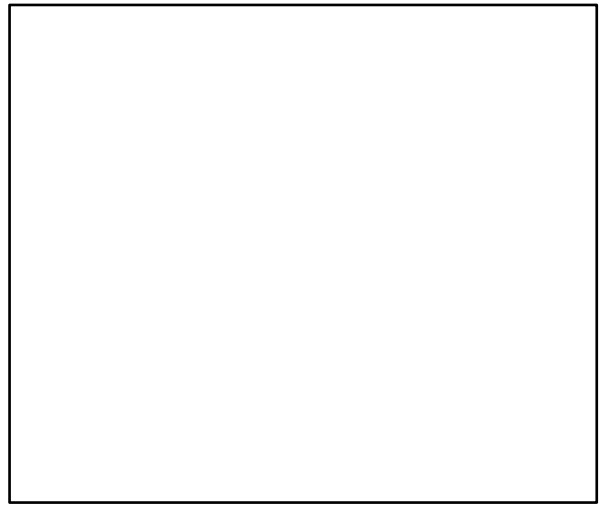
Challenges of Edge Splitting



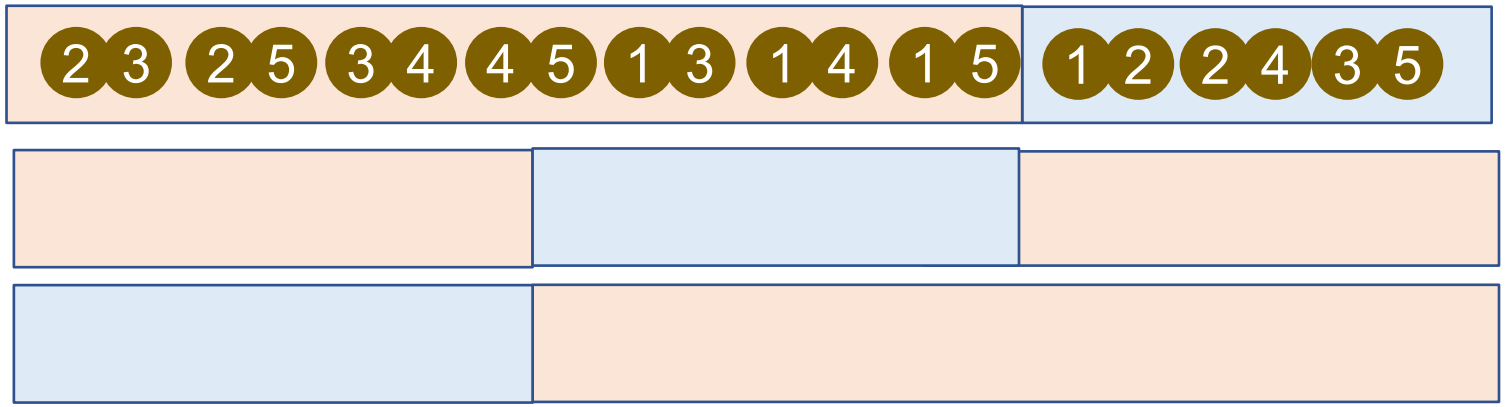
Missing value!!



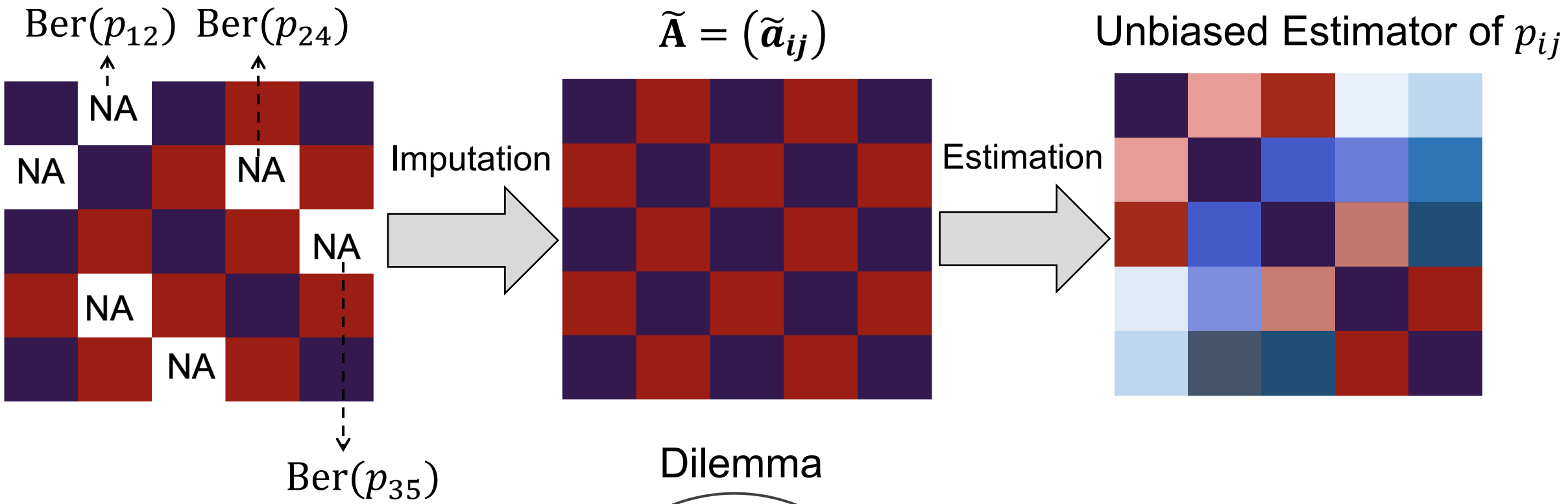
Validation data

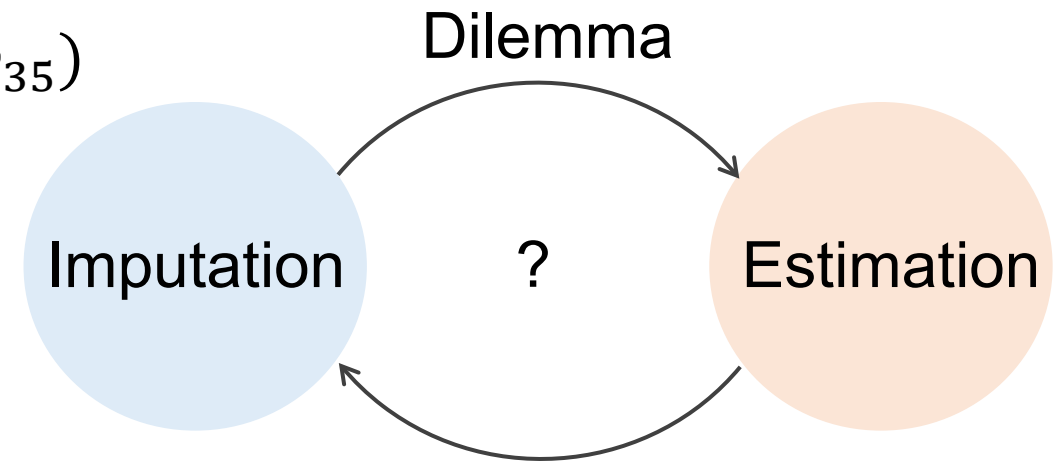


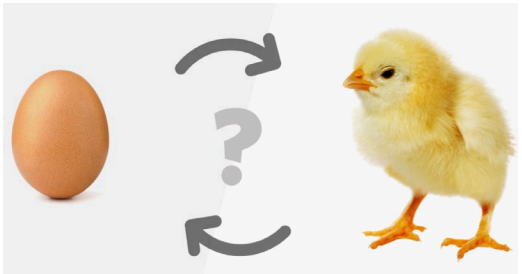
Three-fold split Train Validation



Challenges of Missing Value Imputation



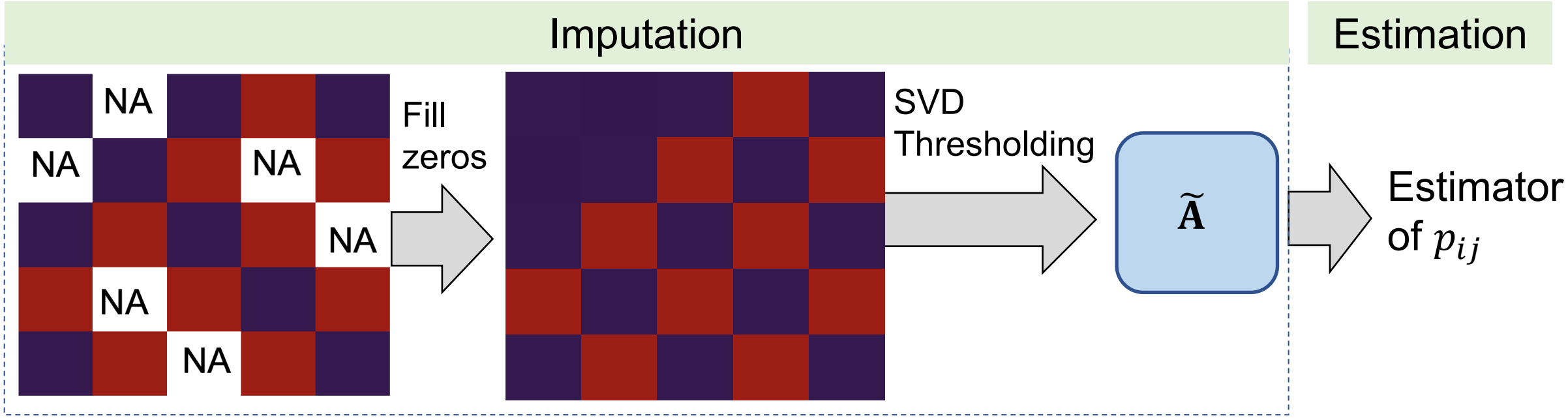




Literature Review of Network Cross-Validation

Edge Cross-Validation (ECV)

(Li, Levina and Zhu. Biometrika 2020)



- Restrictive assumption: Rank(P) $\leq \frac{n}{\delta}$, where δ is average degree
- Introduce additional hyperparameter: Threshold
- Expensive computational cost: O(n^3)

Outline

1 Background

- Network, Graphon and Graphon Estimation
- Motivation and Challenges of Network Cross-Validation

2 Proposed Method: Masked Mirror Validation (MMV)

- MMV Procedure
- Theoretical Results
- Simulation Studies

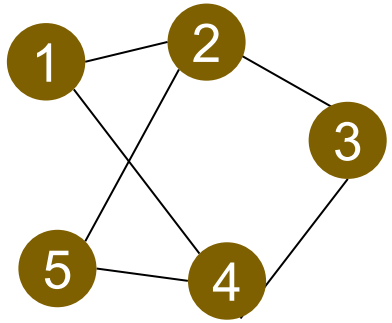
3 Application to Drug Repurposing

- Drug Repurposing
- Med-Reader AI Tool
- Case Study

MMV Procedure

Step 1: Imputation

$$d = \frac{6}{10}$$



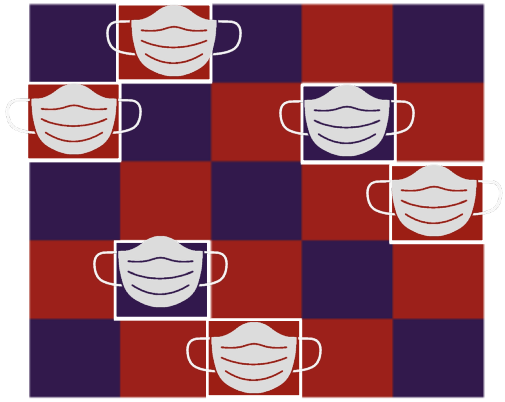
Pooled mean $p_{\cdot\cdot} = \frac{\sum_{i<j} p_{ij}}{\binom{n}{2}}$

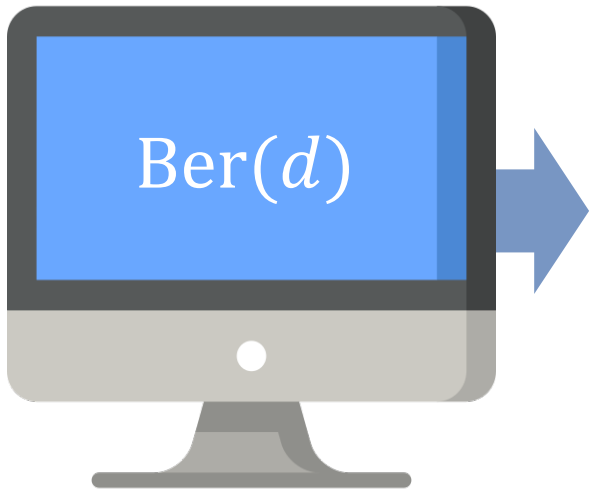
Network density

$$d = \frac{\sum_{i<j} a_{ij}}{\binom{n}{2}}$$

is an unbiased estimator for $p_{\cdot\cdot}$

Training data $\widetilde{\mathbf{A}} = (\tilde{a}_{ij})$



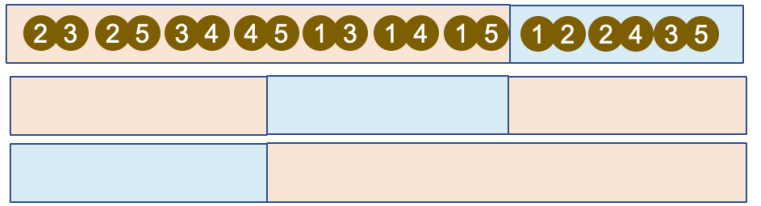


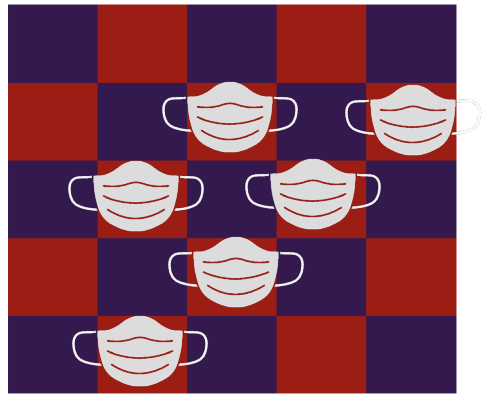
$\tilde{a}_{12} = 1$

$\tilde{a}_{24} = 0$

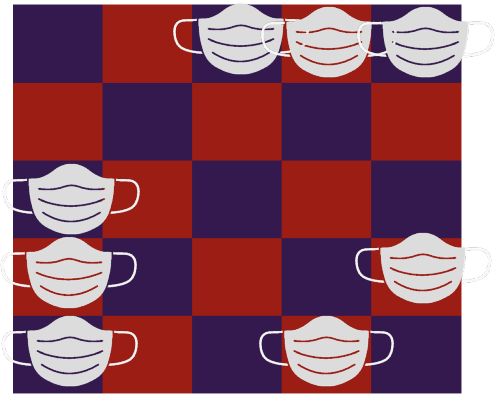
$\tilde{a}_{35} = 1$

Fold 1





Fold 2



Fold 3

Distribution of $\tilde{\mathbf{A}}$

$$\tilde{\mathbf{A}} = (\tilde{a}_{ij}), \tilde{a}_{ij} \sim \text{Ber}(\tilde{p}_{ij})$$

K is the number of folds

If i and j are in validation

$$\tilde{a}_{ij} \sim \text{Ber}(d)$$

$$\mathbb{P}(i \text{ and } j \text{ are in validation}) = \frac{1}{K}$$

$$\mathbb{P}(\tilde{a}_{ij} = 1 | i \text{ and } j \text{ are in validation}) = d$$

If i and j are not in validation

$$\tilde{a}_{ij} = a_{ij} \sim \text{Ber}(p_{ij})$$

$$\mathbb{P}(i \text{ and } j \text{ are not in validation}) = \frac{K-1}{K}$$

$$\mathbb{P}(\tilde{a}_{ij} = 1 | i \text{ and } j \text{ are not in validation}) = p_{ij}$$

By law of total probability

$$\tilde{p}_{ij} := \mathbb{P}(\tilde{a}_{ij} = 1) = \frac{1}{K} d + \frac{K-1}{K} p_{ij}$$

Derivation of Bias Correction Formula

$$\tilde{p}_{ij} := \mathbb{P}(\tilde{a}_{ij} = 1) = \frac{1}{K} d + \frac{K-1}{K} p_{ij}$$

Reverse map

$$p_{ij} = \frac{K\tilde{p}_{ij} - d}{K-1}$$

$$\hat{p}_{ij} = \frac{K\hat{\tilde{p}}_{ij} - d}{K-1}$$

Used for bias correction

Unbiased estimator of p_{ij}

Unbiased estimator of $\tilde{p}_{ij}$

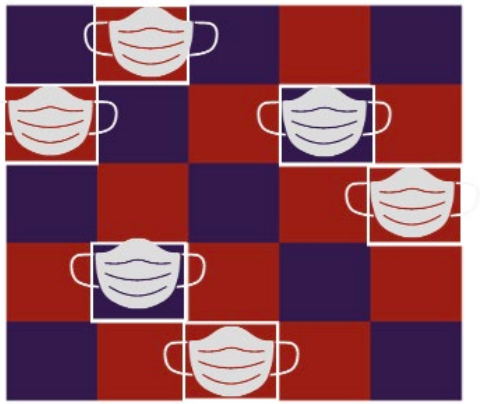
By applying graphon estimation method to $\tilde{\mathbf{A}}$, we can get an unbiased estimator of $\tilde{p}_{ij}$.

Step 2: Estimation and Bias Correction

$$\widehat{\boldsymbol{P}}_{\boldsymbol{m}} = \frac{K\widehat{\boldsymbol{P}}_{\boldsymbol{m}}^{\tilde{\mathbf{A}}} - d}{K-1}$$

Training data

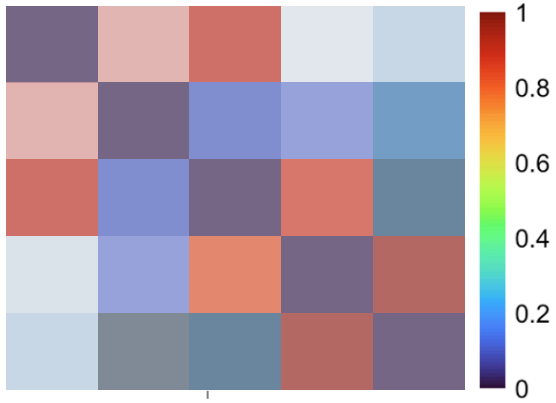
$\tilde{\mathbf{A}} = (\tilde{a}_{ij})$



Applying a graphon estimation method using a given **hyperparameter m**

Estimator

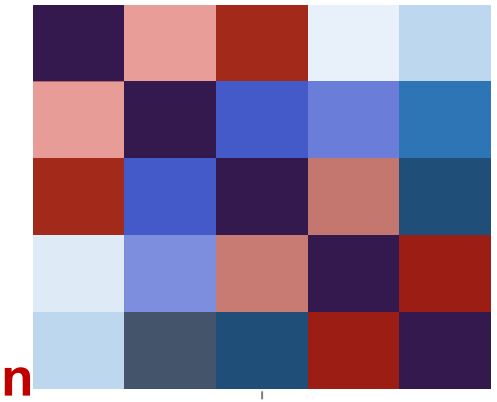
$\widehat{\boldsymbol{P}}_{\boldsymbol{m}}^{\tilde{\mathbf{A}}}$



Bias correction

Unbiased estimator

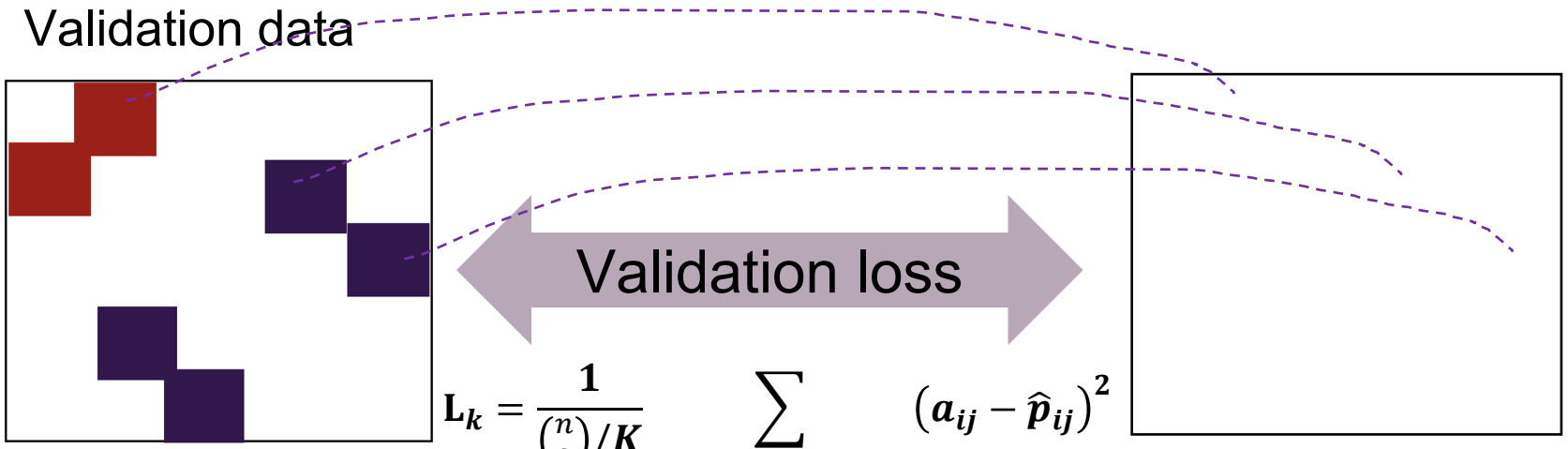
$\widehat{\boldsymbol{P}}_{\boldsymbol{m}} = (\hat{p}_{ij})$



ij−th entry is an unbiased estimator of $\tilde{p}_{ij}$

ij−th entry is an unbiased estimator of p_{ij}

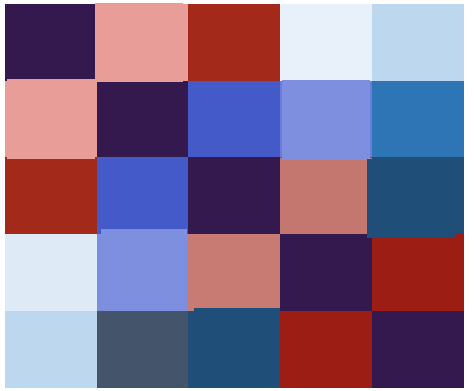
Step 3: Validation Loss



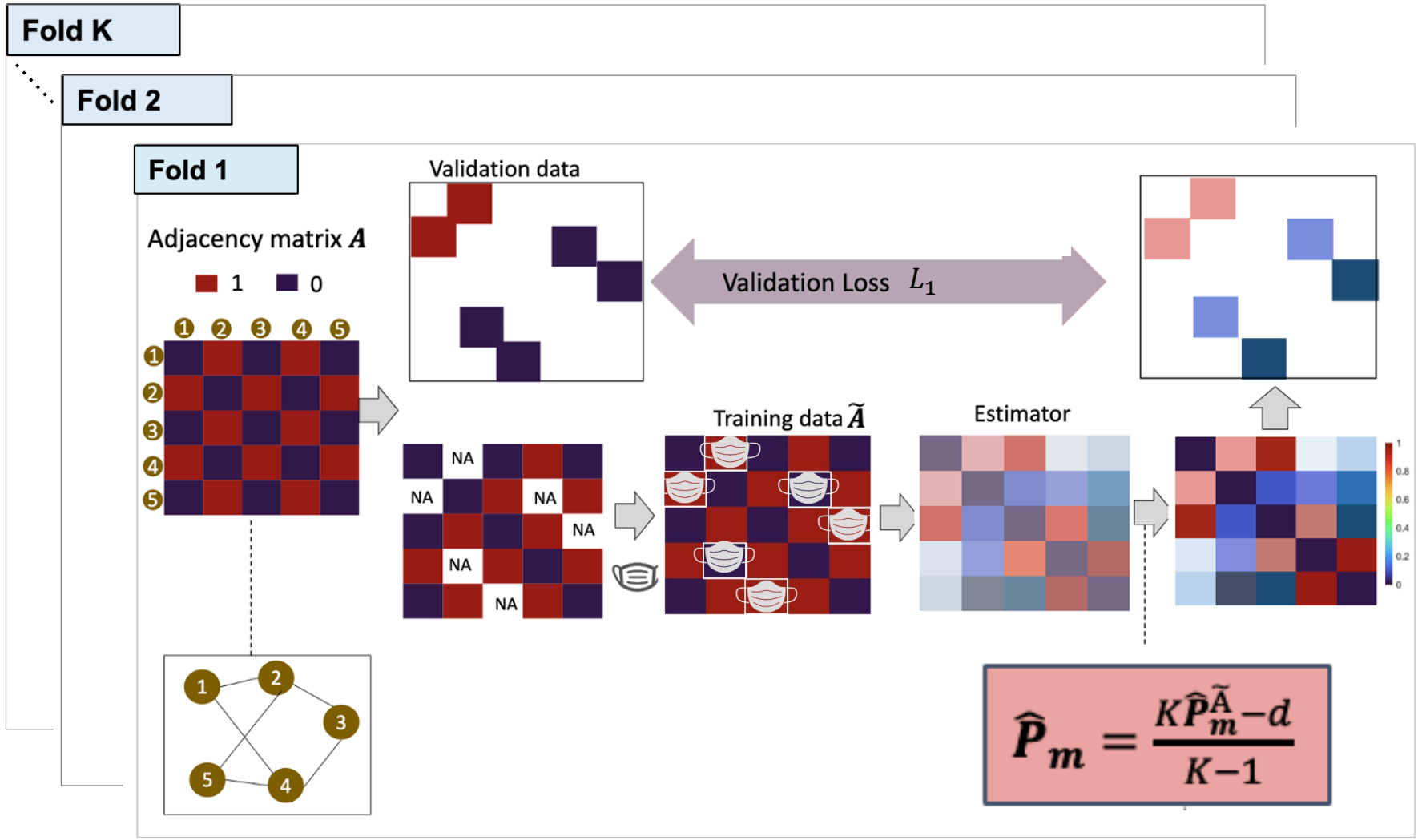
$$\mathbf{L}_k = \frac{1}{\binom{n}{2}/K} \sum_{ij \text{ in kth validation}} (a_{ij} - \hat{p}_{ij})^2$$

$$k = 1, \ldots, K$$

Unbiased estimator $\widehat{\boldsymbol{P}}_{\boldsymbol{m}} = (\hat{p}_{ij})$



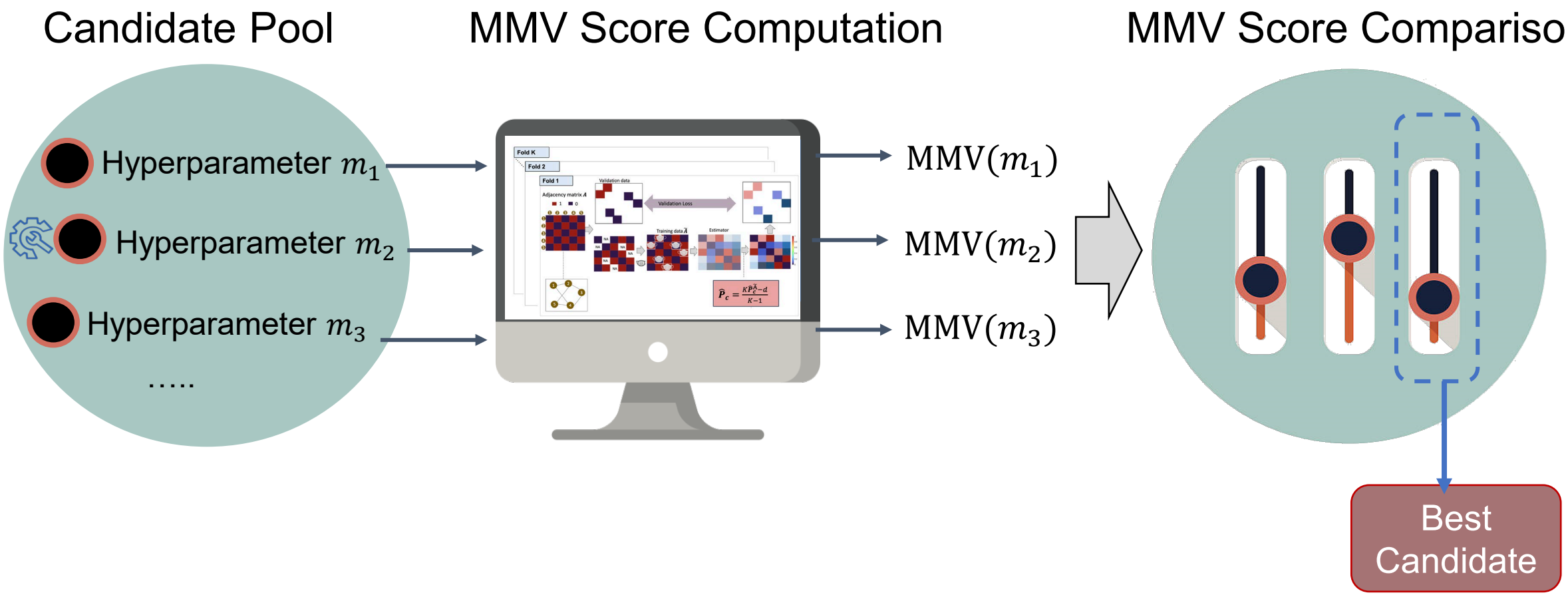
MMV Score of a Hyperparameter



MMV score

$$\mathrm{MMV(m)} = \frac{1}{K}\sum_{k=1}^{K} L_k$$

Hyperparameter Tuning Using MMV score



Theoretical Results

Assumption 1: $K = O(n)$.

Assumption 2: $$\frac{||\widehat{\boldsymbol{P}}_{\boldsymbol{m}}^{\widetilde{\boldsymbol{A}}} - \widehat{\boldsymbol{P}}_{\boldsymbol{m}}^{\boldsymbol{A}}||_F}{||\widetilde{\boldsymbol{A}} - \boldsymbol{A}||_F} = O_p(1).$$

Theorem (Selection consistency of MMV)

Given a set of hyperparameters, under Assumptions 1 and 2, as $n \to \infty$, the probability of MMV selecting the optimal hyperparameter converges to one.

Minimize the MSE

Simulation Setting

Generate μ_i

$$\mu_i \sim U(0,1),\ i = 1, \ldots, n$$

Generate p_{ij}

Graphon Setting 1

$$p_{ij} = \frac{1}{1 + \exp(-10(\mu_i^2 + \mu_j^2))}$$

Graphon Setting 2

$$p_{ij} = 0.5 + \frac{\mu_i \mu_j}{3}$$

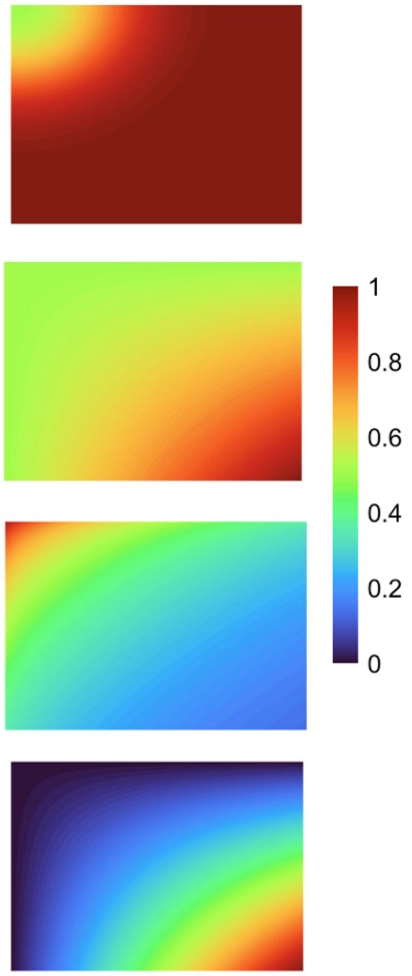
Graphon Setting 3

$$p_{ij} = \mu_i \mu_j$$

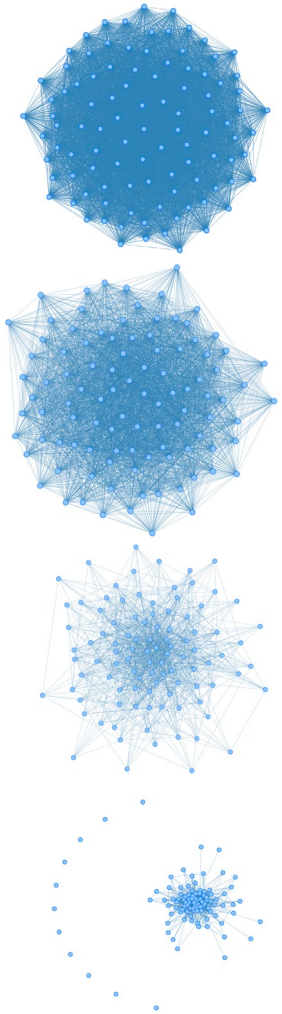
Graphon Setting 4

$$p_{ij} = \exp(-(\mu_i^{0.7} + \mu_j^{0.7}))$$

Probability matrix
$\mathbf{P} = (p_{ij}) \in \mathbb{R}_{n \times n}$



Generate network using $a_{ij} \sim \mathrm{Ber}(p_{ij})$



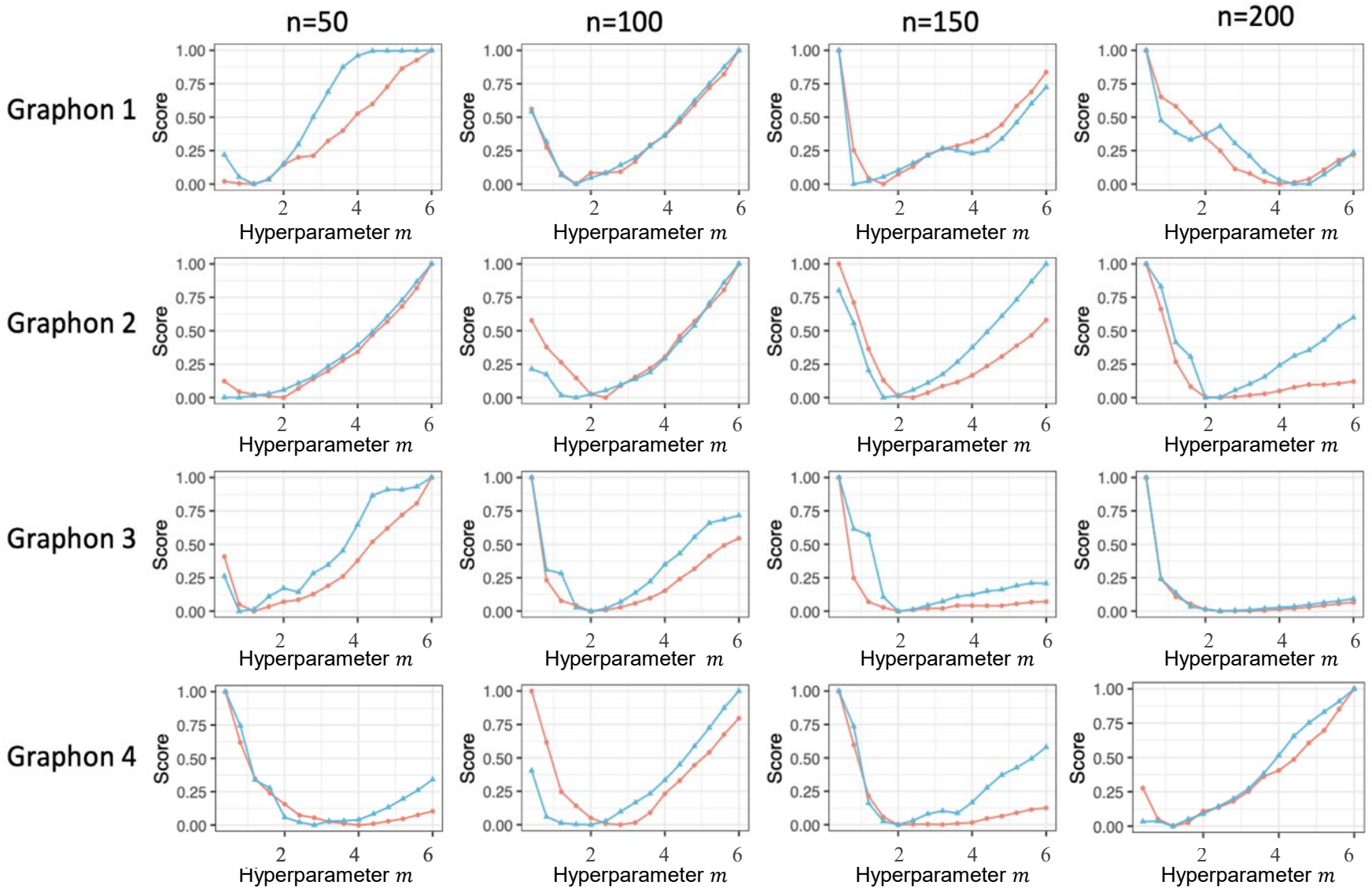
$n = 50, 100, 150, 200$

$K = 0.1n$

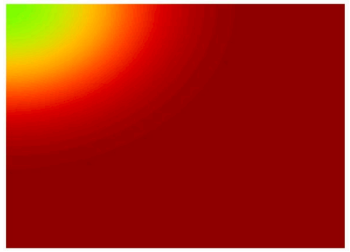
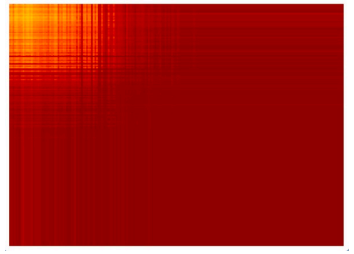
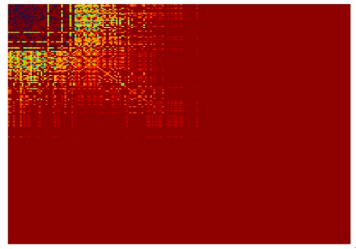
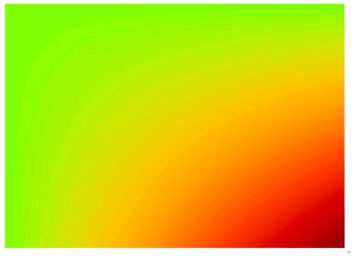
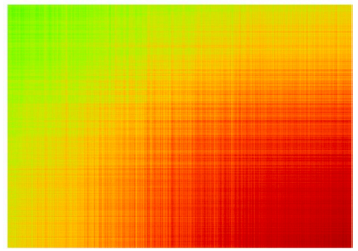
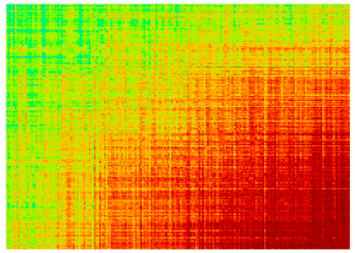
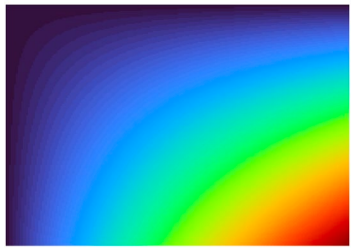
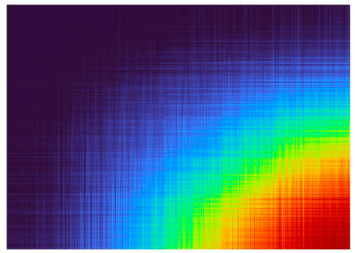
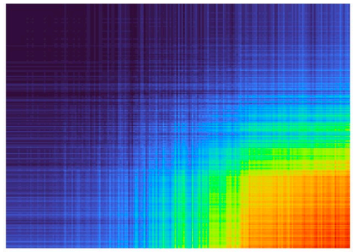
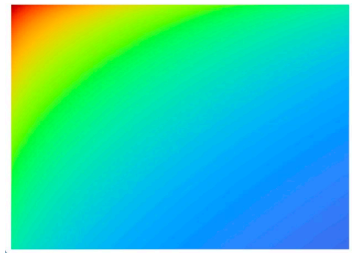
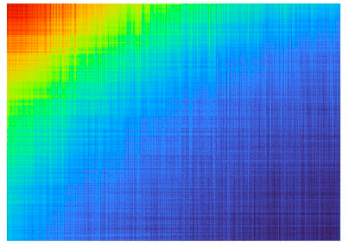
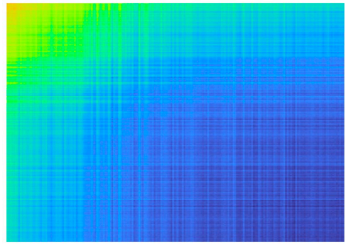
Simulation Results: Validation of Our Theorem

Tuning hyper-parameter m in NS method

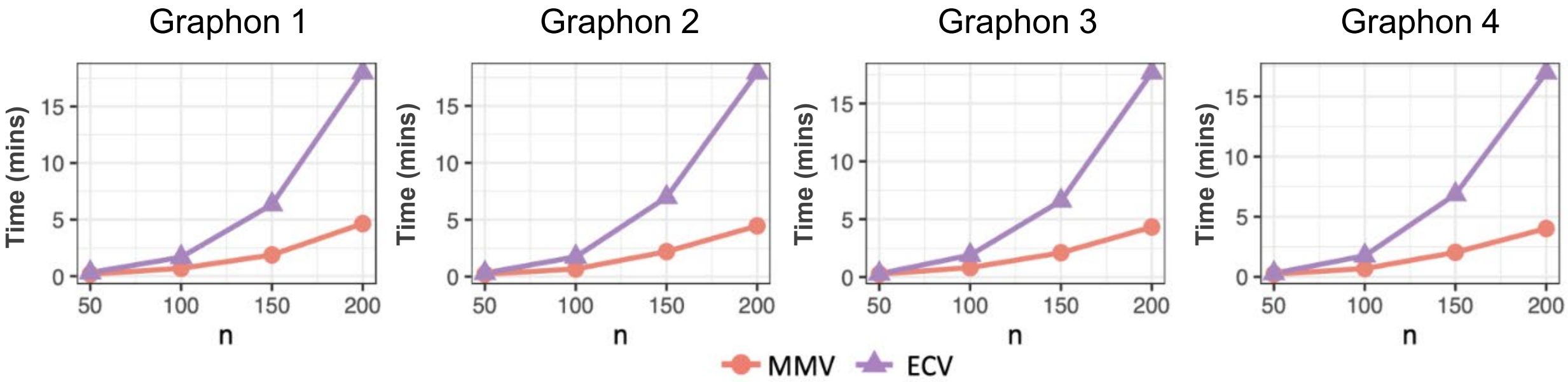
(Zhang, Levina and Zhu. Biometrika 2017)



Simulation Results: Graphon Estimation

	True P	$\widehat{\mathbf{P}}$ (m selected by MMV)	Average MSE	$\widehat{\mathbf{P}}$ (m selected by ECV)	Average MSE
Graphon 1			**$0.51\ (\mathbf{sd} = 0.13)$**		$0.98\ (\mathrm{sd} = 0.26)$
Graphon 2			**$0.79\ (\mathbf{sd} = 0.11)$**		$1.06\ (\mathrm{sd} = 0.09)$
Graphon 3			**$0.94\ (\mathbf{sd} = 0.03)$**		$1.21\ (\mathrm{sd} = 0.07)$
Graphon 4			**$1.39\ (\mathbf{sd} = 0.11)$**		$2.22\ (\mathrm{sd} = 0.22)$

Simulation Results: Computation Time



Outline

1 Background

- Network, Graphon and Graphon Estimation
- Motivation and Challenges of Network Cross-Validation

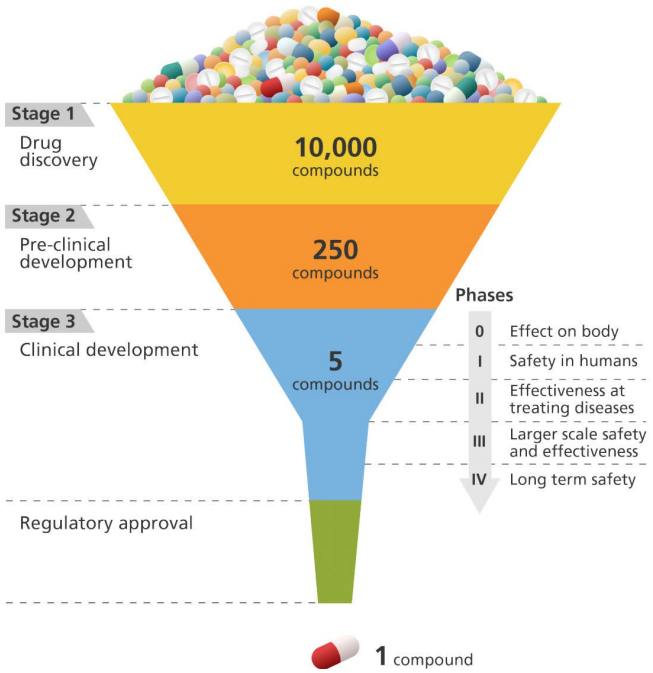
2 Proposed Method: Masked Mirror Validation (MMV)

- MMV Procedure
- Theoretical Results
- Simulation Studies

3 Application to Drug Repurposing

- Drug Repurposing
- Med-Reader AI Tool
- Case Study

Drug Repurposing



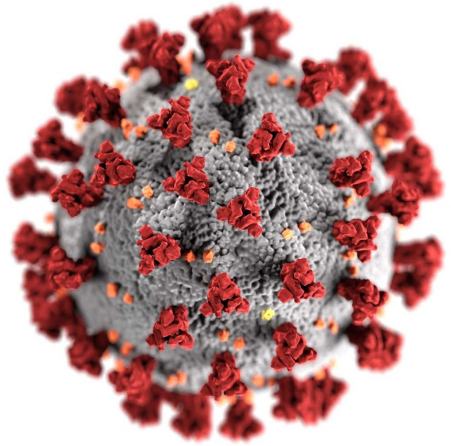
12-15 years

$2-3 billion

Drug repurposing example

Influenza

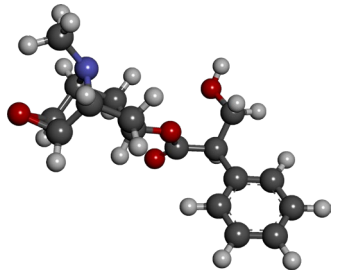
COVID-19



Picture source:
https://discover.hubpages.com/education/How-are-drugs-developed-and-approved-The-drug-development-process
https://www.hopkinsmedicine.org/health/conditions-and-diseases/influenza/influenza-flu-in-children

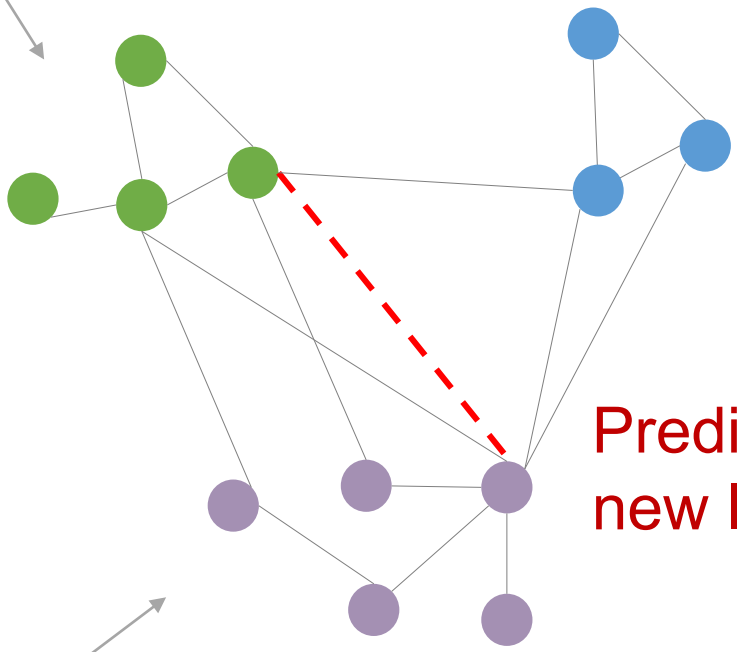
Drug Repurposing Using Knowledge Graph

Drug



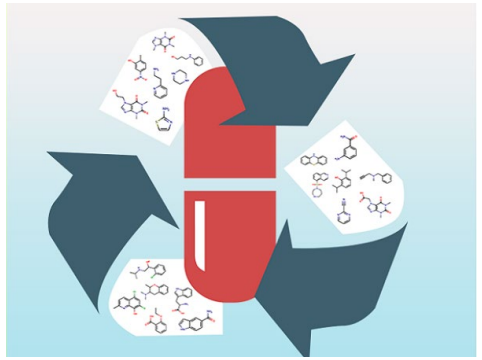
Gene

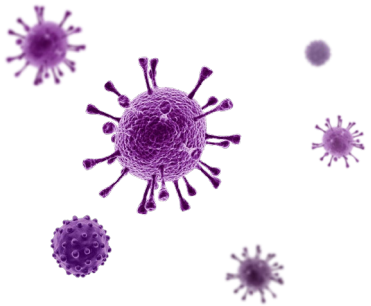
Knowledge graph



Predicted
new link

Drug repurposing



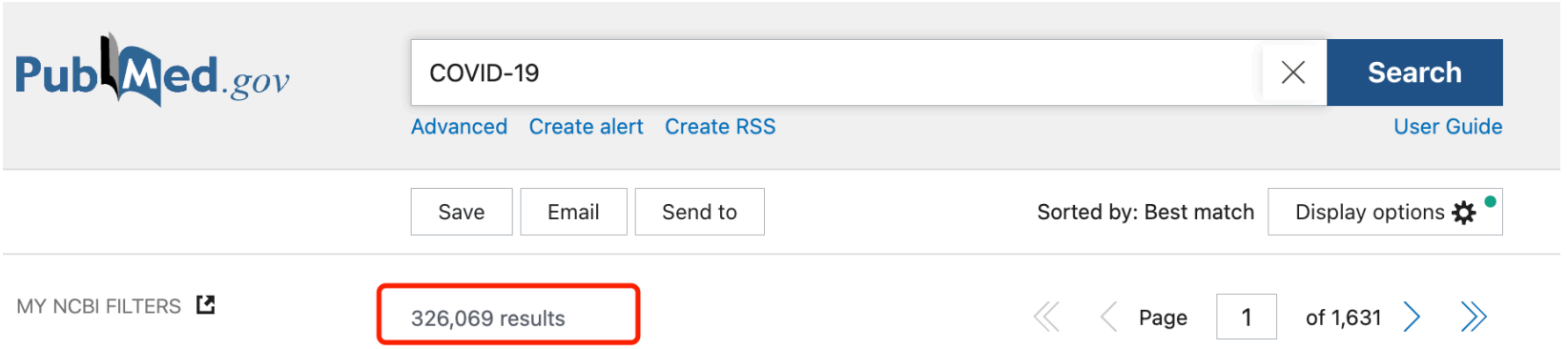


Disease

Step 1: Knowledge graph construction

Step 2: New drug-disease link prediction

Challenge of Step 1

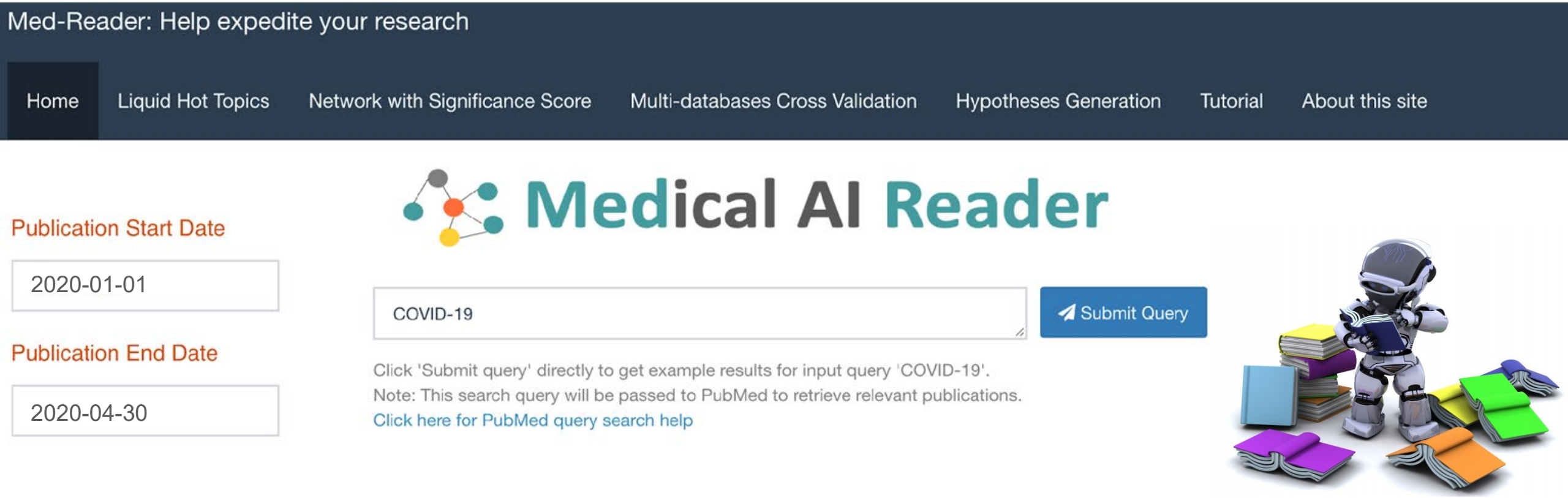


If the reading speed is 30 mins one paper

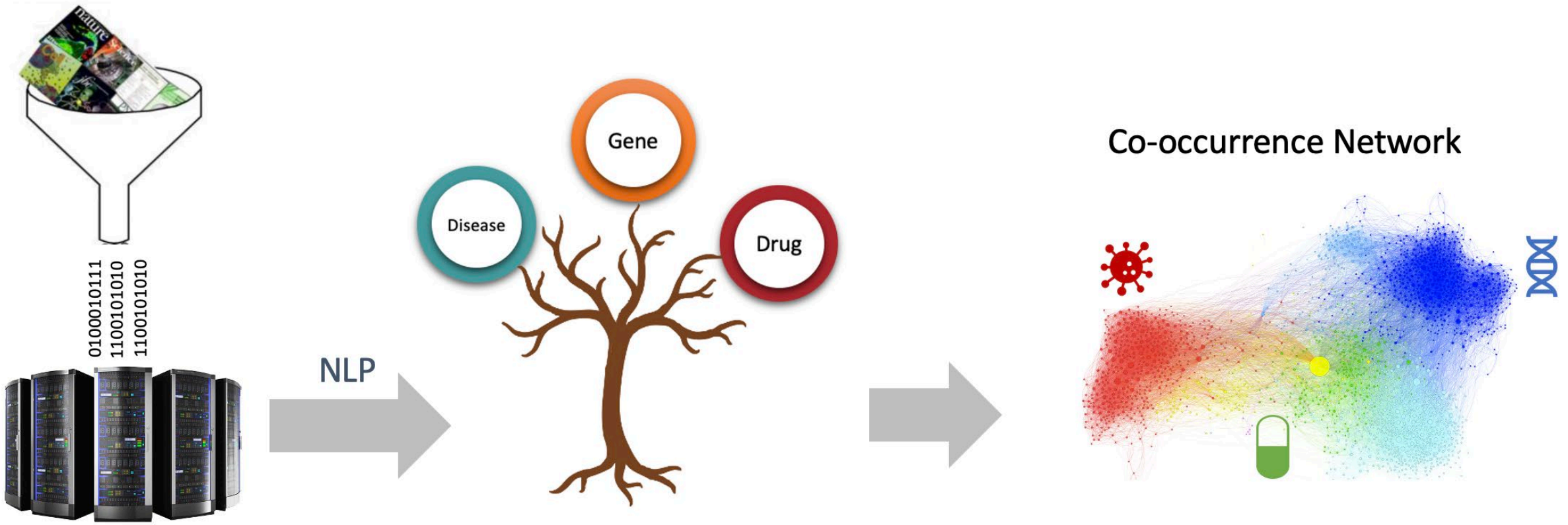
It takes 19 years to read all papers

Picture source: https://discover.hubpages.com/education/How-are-drugs-developed-and-approved-The-drug-development-process

Solution: Our Developed Med-Reader

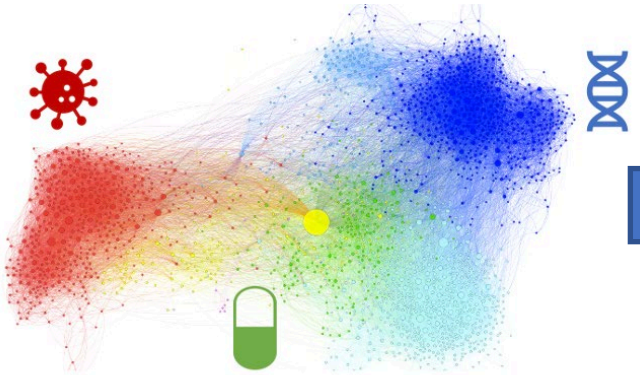


Med-Reader for Knowledge Graph Construction

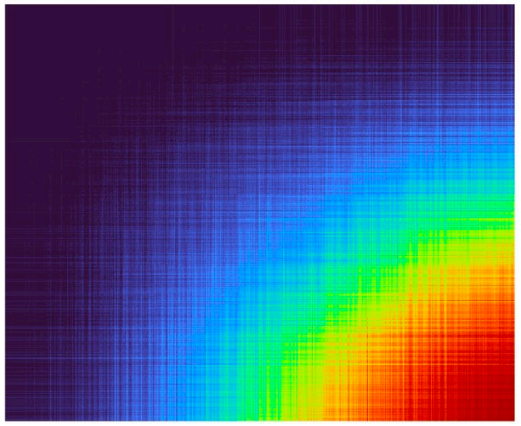


Graphon Estimation in Step 2

Knowledge Graph



Graphon Estimate



Prediction Results

Rank	Drug	Disease	$\hat{p}_{ij}$
1	Drug 1	Disease 1	0.97
2	Drug 2	Disease 2	0.95
…			

Challenge: How to tune hyperparameters?

Solution: Masked Mirror Validation!

Case Study of Drug Repurposing

Step 1: COVID-19 Knowledge Graph Construction

Publication Start Date

2020-01-01

Publication End Date

2020-04-30

Medical AI Reader

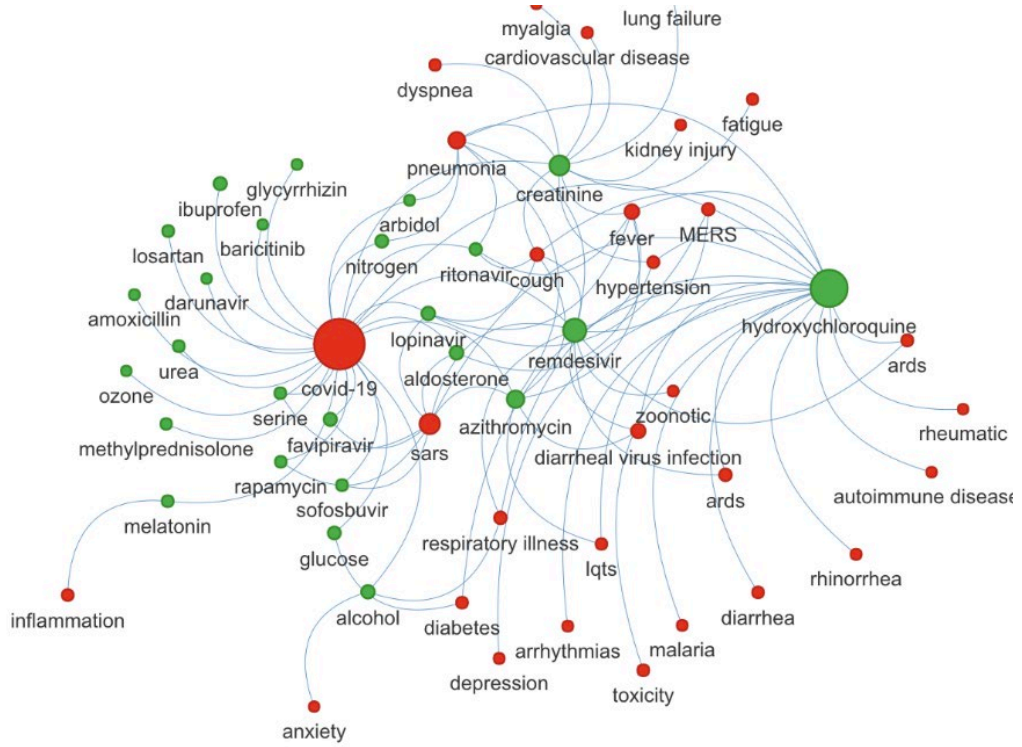
COVID-19

Submit Query

Input: “COVID-19”
(“01/01/2020” – “04/30/2020”)

Co-occurrence Network

1036 nodes 5781 edges



Case Study of Drug Repurposing

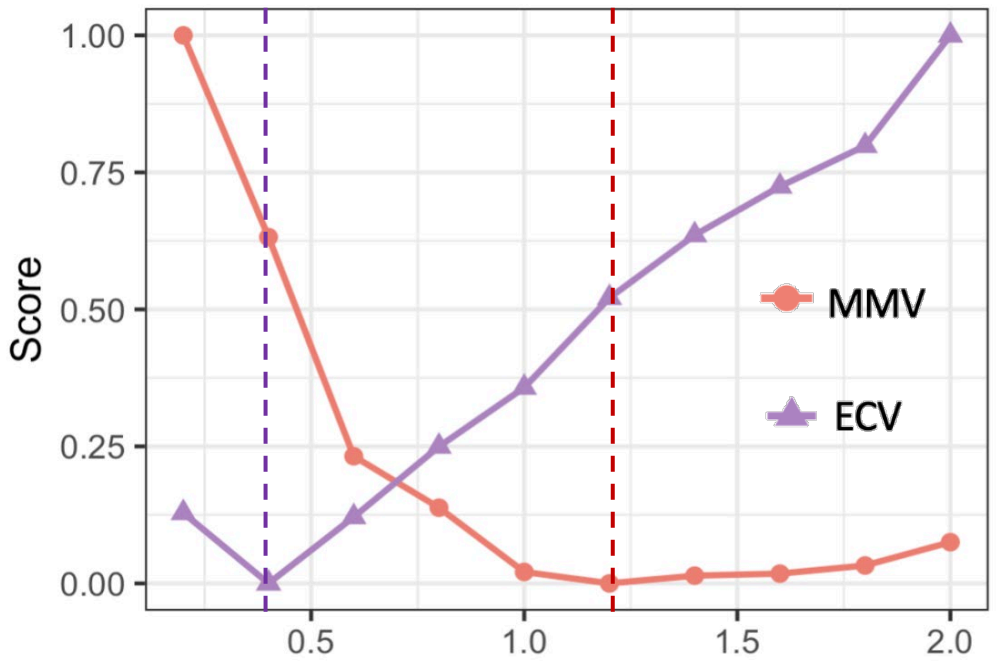
Step 2: Graphon Estimation

Hyperparameter Tuning

Tuning hyperparameter m in NS method

(Zhang, Levina and Zhu. Biometrika 2017)

MMV selects $m = 1.2$



ECV selects $m = 0.2$

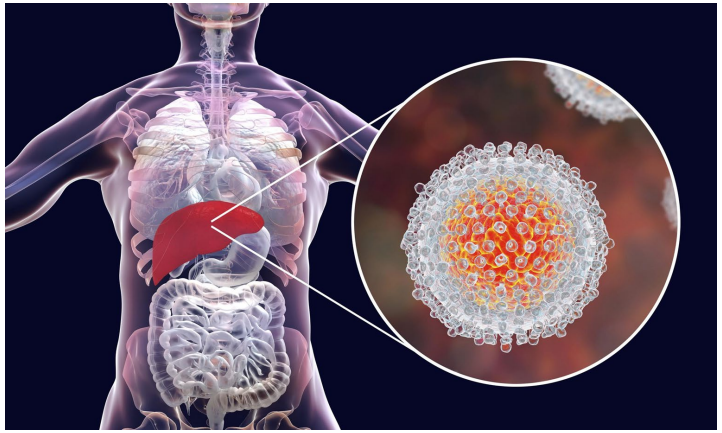
Prediction

Prediction Results using MMV

Rank	Drug	Disease	$\widehat{p}_{ij}$
1	Ledipasvir	COVID-19	0.91
2	Budesonide	COVID-19	0.87
…			

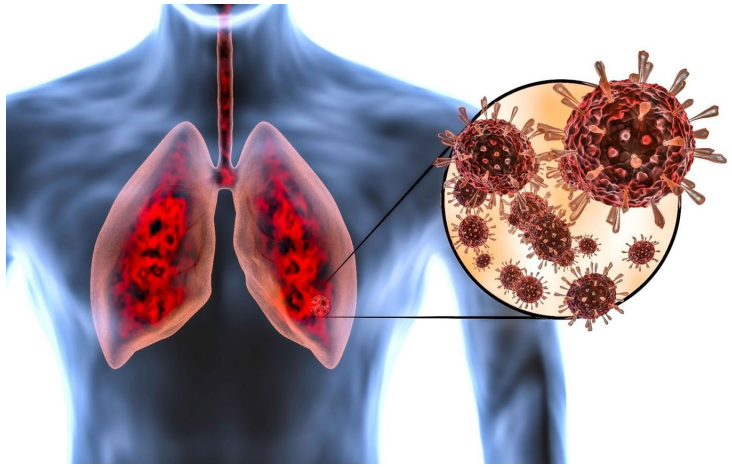
Top Prediction: COVID-19 and Ledipasvir

Hepatitis C



Ledipasvir

COVID-19



Scientific way of validating the prediction

1/1/2020 – 4/30/2020

Used for **making** prediction

5/1/2020 – 5/1/2022

Used for **validating** prediction

cells MDPI

Cells. 2021 May; 10(5): 1052. PMCID: PMC8146643
Published online 2021 Apr 29. doi: 10.3390/cells10051052 PMID: 33946869

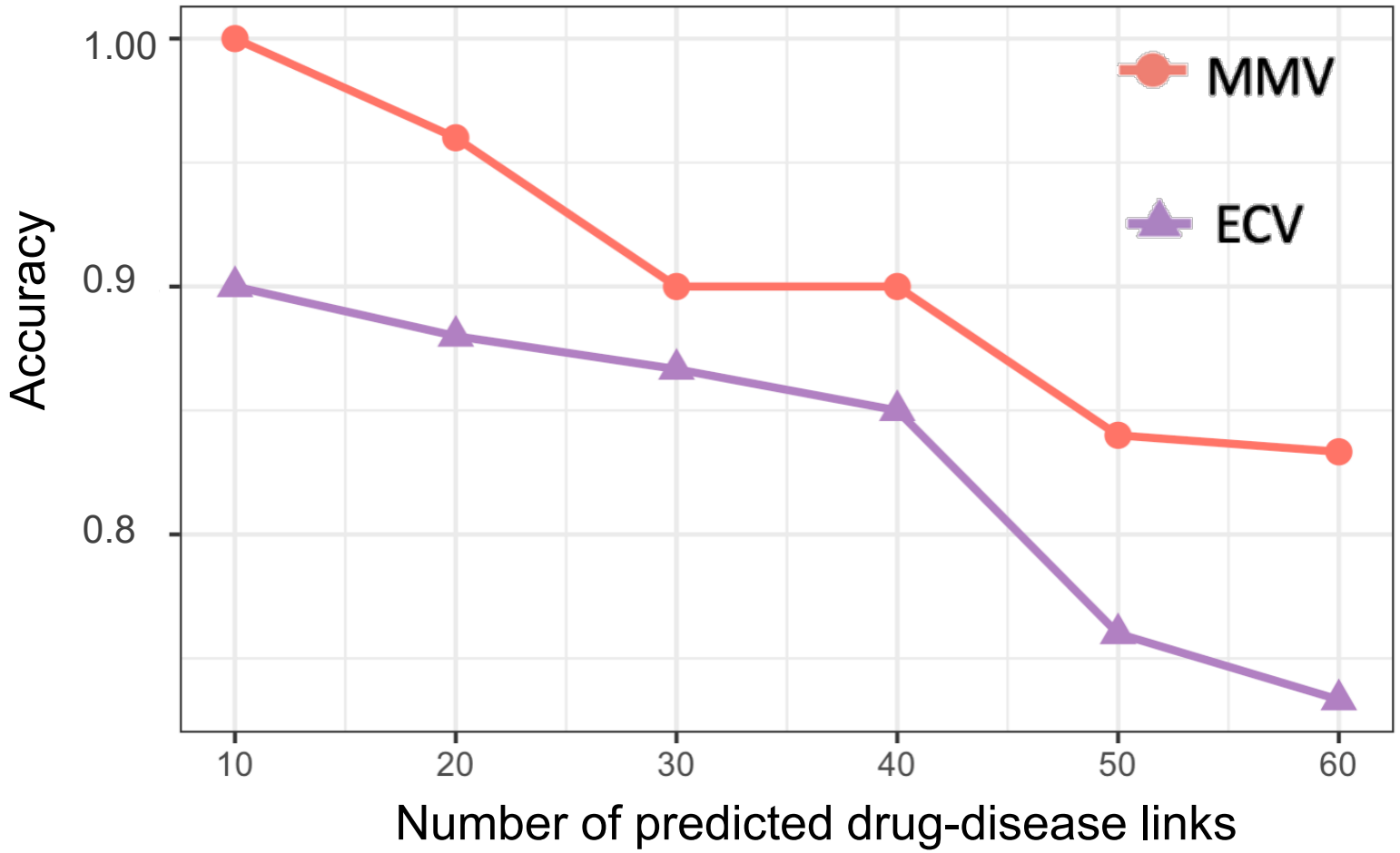
Remdesivir and Ledipasvir among the FDA-Approved Antiviral Drugs Have Potential to Inhibit SARS-CoV-2 Replication

Rameez Hassan Pirzada,[1] Muhammad Haseeb,[1] Maria Batool,[1,2] MoonSuk Kim,[1] and Sangdun Choi[1,2,*]

Comparison Between MMV and ECV

Prediction Accuracy

$$\text{Accuracy} = \frac{\#\ \text{validated predictions}}{\#\ \text{predictions}}$$



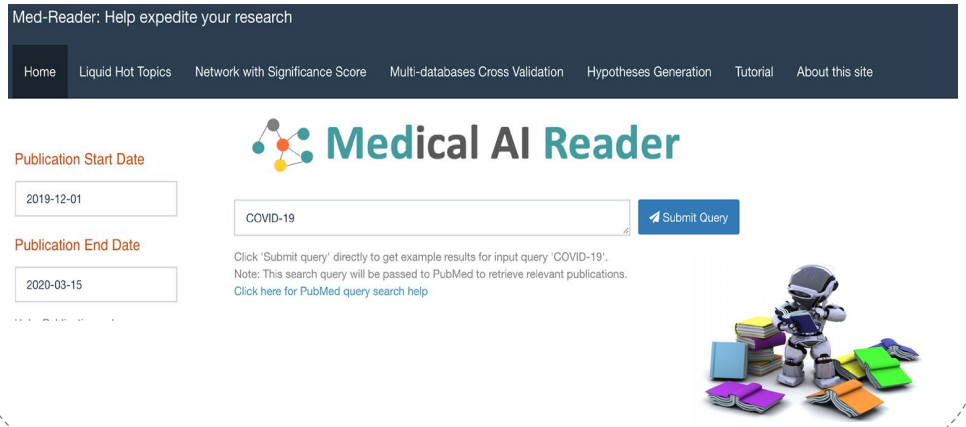
Time

Computational Time (mins)

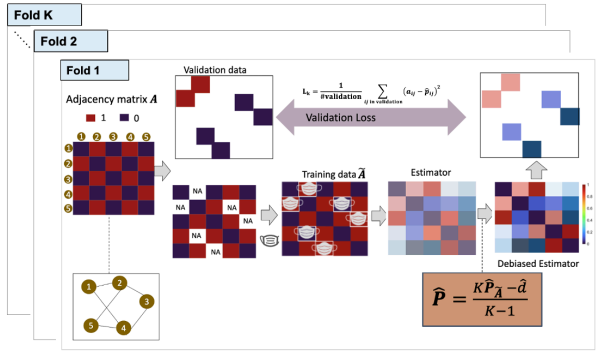
- **MMV: 28.90 ± 1.30**
- ECV: 250.65 ± 2.11

Take Home Message

Med-Reader

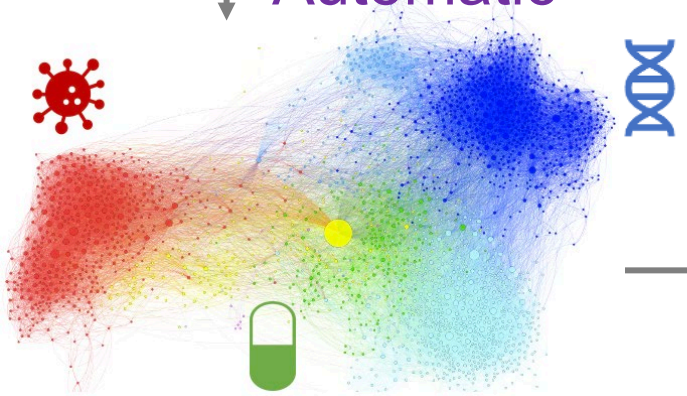


Masked mirror validation (MMV)



- ✓ Theoretical guarantee
- ✓ Fast
- ✓ Effective

Automatic



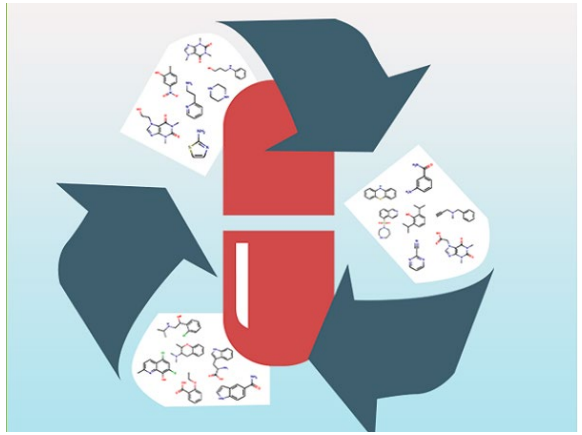
Knowledge graph

Better estimation

Graphon Estimation

Promotes

Drug repurposing



BOSTON UNIVERSITY

big data analytics Lab

Thank you!