# Learning Probabilistic and Causal Models with(out) Imperfect Advice

PhD Defense

13 January 2025

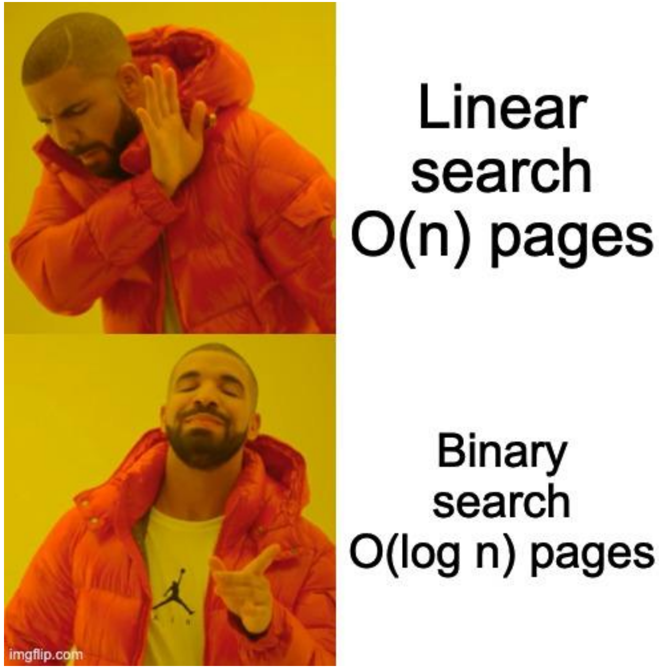
Davin Choo

National University of Singapore

# How do we find words in a dictionary?

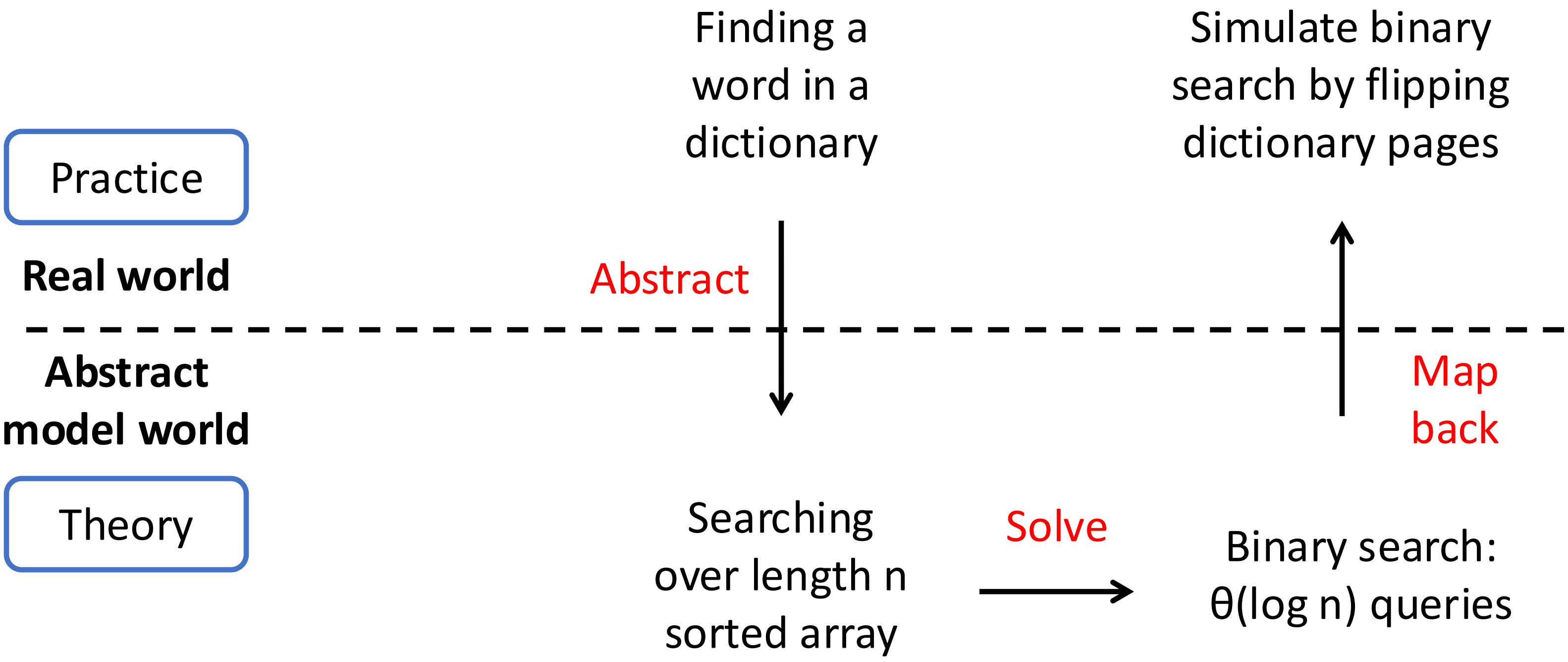
ChatGPT on the prompt “Generate me a cartoon kid finding a word in a dictionary”

# How do we find words in a dictionary?




https://imgflip.com/memegenerator/Drake-Hotline-Bling

# A general problem-solving framework



# A general problem-solving framework

- Complex setting
- Many nuances
- Possibly unseen problem

Finding a word in a dictionary

Simulate binary search by flipping dictionary pages

**Real world**

Abstract ↓ ↑ Map back

**Abstract model world**

- Simplified setting
- Generic problem framing
- Many plug-and-play solution concepts

Searching over length n sorted array

Solve →

Binary search: $\theta(\log n)$ queries

# A general problem-solving framework

- Complex setting
- Many nuances
- Possibly unseen problem

**Real world**

**Abstract**
**model world**

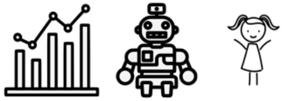
- Simplified setting
- Generic problem framing
- Many plug-and-play solution concepts

## Two useful scientific models

1) **Probabilistic models** for predictive tasks

2) **Causal models** for understanding interventional effects on systems

# Side-information about problem instances



It's on page x

**Real world**

**Abstract model world**

Finding a word in a dictionary

Abstract ↓

Searching over length n sorted array

Solve →

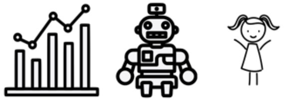
Binary search: θ(log n) queries

Map back ↑

Simulate binary search by flipping dictionary pages

https://thenounproject.com/icon/statistics-7090732/, https://thenounproject.com/icon/girl-1257314/ , https://thenounproject.com/icon/robot-7098785/, Ideogram on the prompt "A cartoon of a kid sitting at a desk in a library, with a friend standing beside him and a robot also standing beside him. The kid is looking through a large dictionary. The kid points to a word in the dictionary and the robot points to a page number in the dictionary. The background contains bookshelves filled with books." https://ideogram.ai/g/QIpUowELS3yRtzrVFmggxA/0

# Side-information about problem instances



It’s on page x

**Real world**

Finding a word in a dictionary

Abstract

Simulate binary search by flipping dictionary pages

Map back

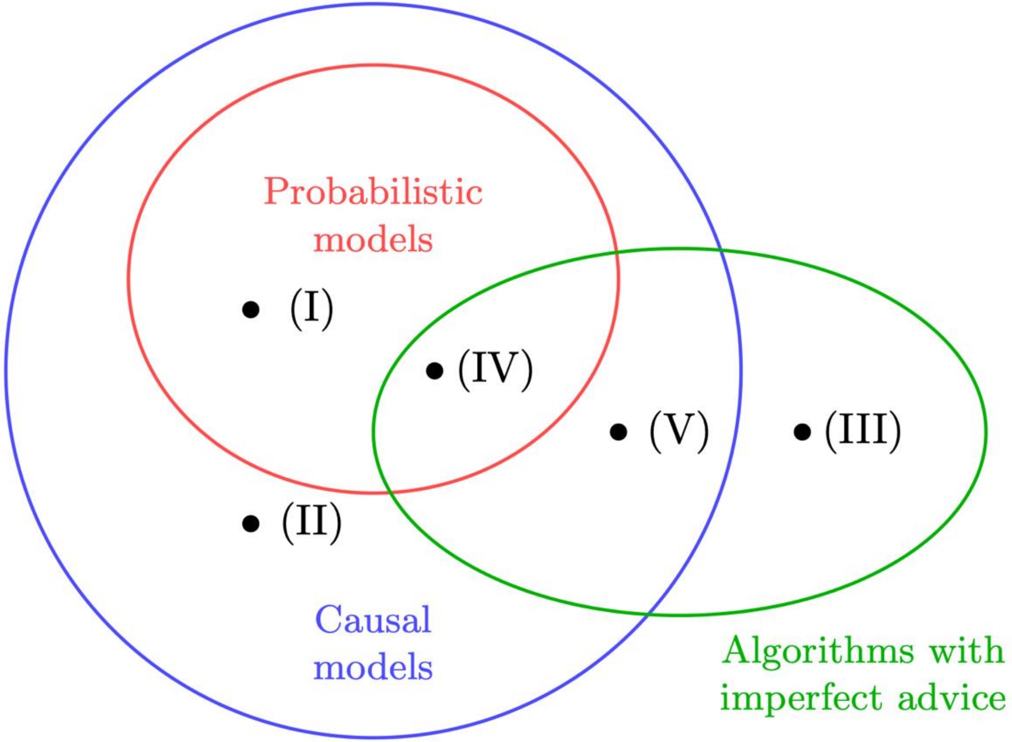
**Abstract model world**

Searching over length n sorted array

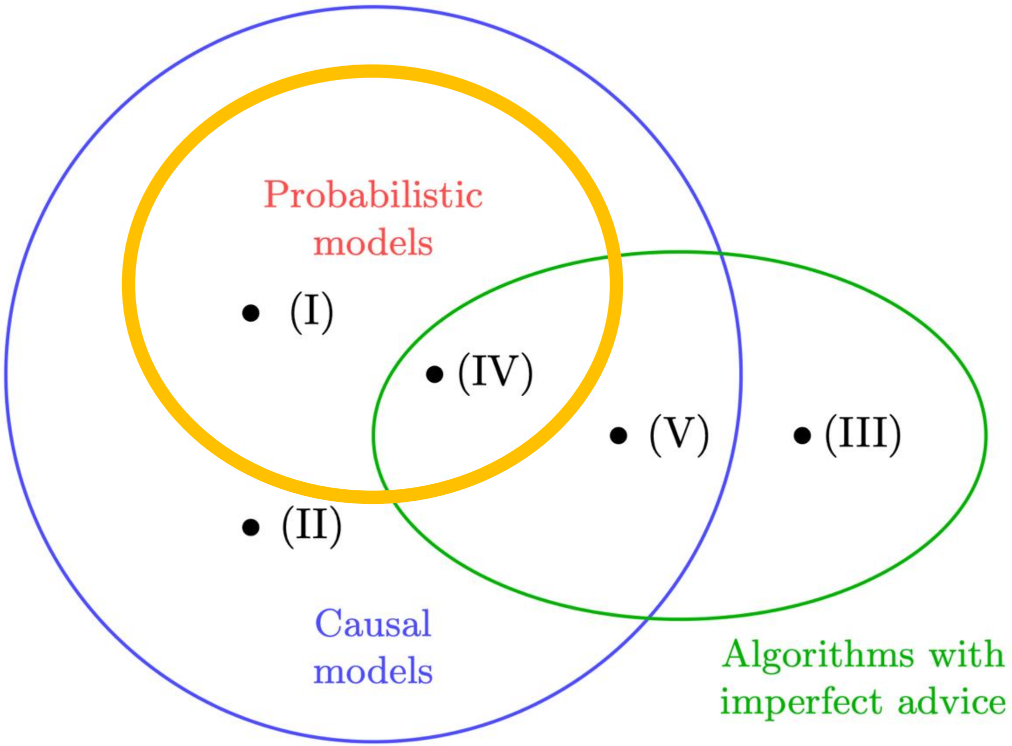
Solve

Algorithm with imperfect advice: $\theta(\log |x - x^*|)$ queries

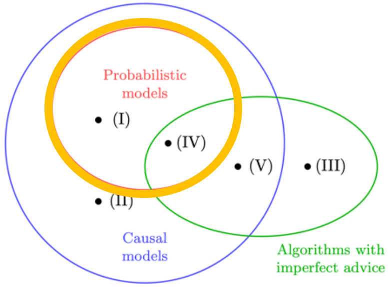
# Main themes explored in my PhD thesis



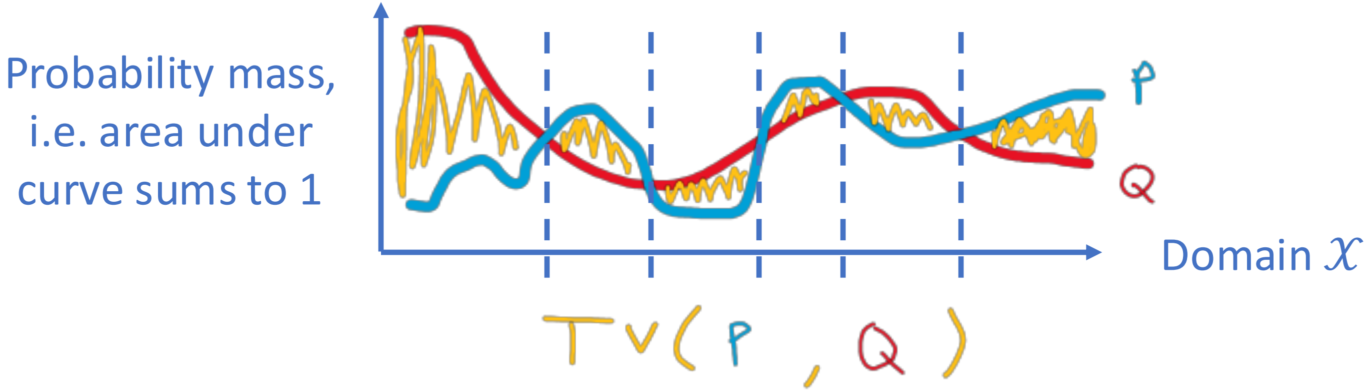
# Main themes explored in my PhD thesis



# (I): Probabilistic models

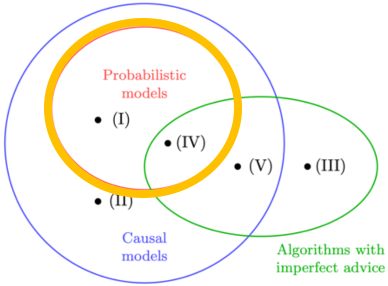



- Classic results in statistics show asymptotic convergence of estimators in the limit of large data
- Probably Approximately Correct (PAC) learning model [Val84]
  - Given sample access to some underlying distribution $\mathcal{P}$, produce $\hat{\mathcal{P}}$ such that $\mathrm{TV}(\mathcal{P}, \hat{\mathcal{P}}) \leq \varepsilon$ with probability $\geq 1 - \delta$




[Val84] Leslie G Valiant. *A theory of the learnable*. Communications of the ACM, 1984.

# (I): Probabilistic models

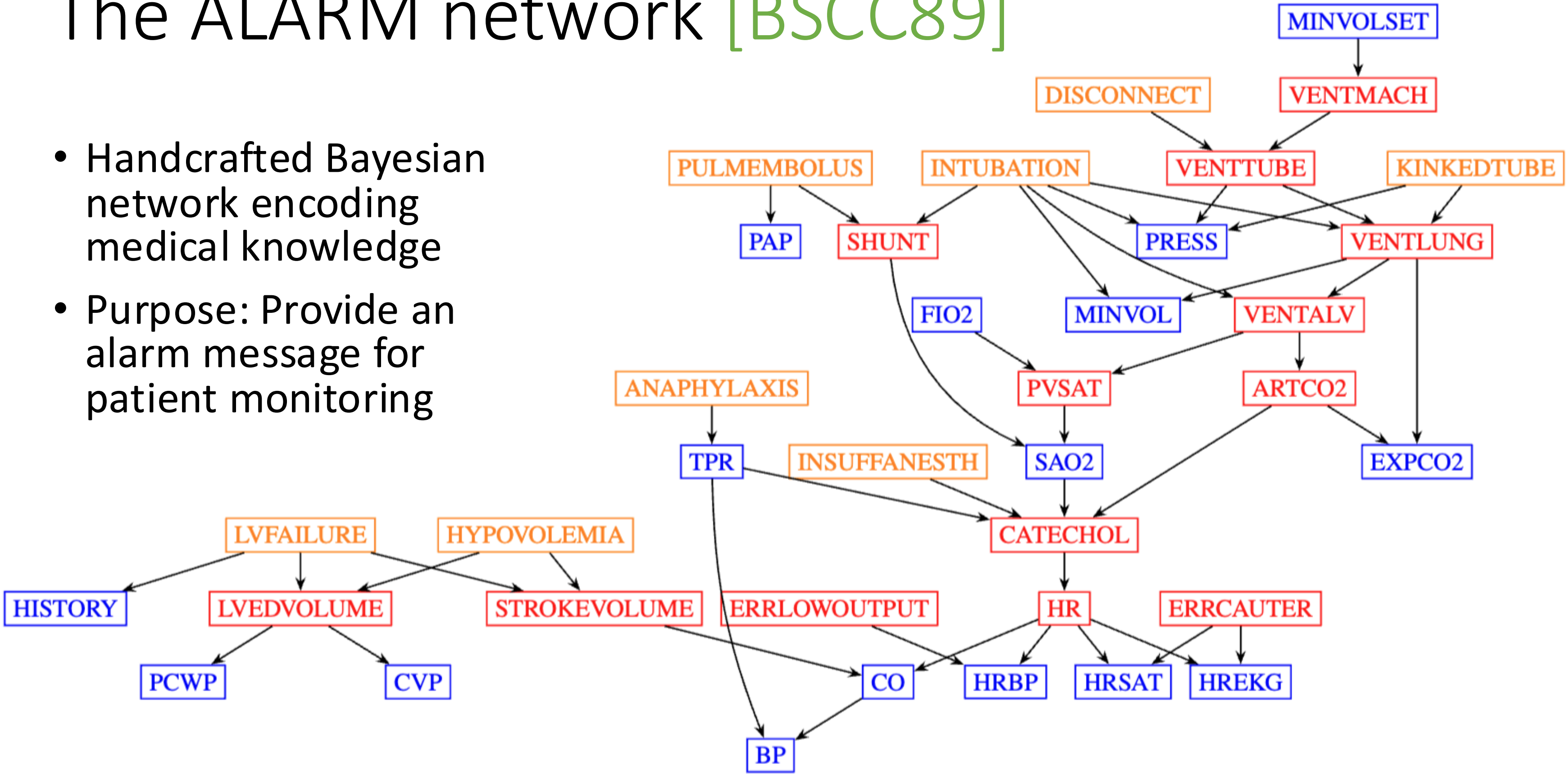



- Classic results in statistics show asymptotic convergence of estimators in the limit of large data
- Probably Approximately Correct (PAC) learning model [Val84]
    - Given sample access to some underlying distribution $\mathcal{P}$, produce $\hat{\mathcal{P}}$ such that $\mathrm{TV}(\mathcal{P}, \hat{\mathcal{P}}) \leq \varepsilon$ with probability $\geq 1 - \delta$
- Bayesian networks [Pea88]
    - Probabilistic graphical model commonly used to model beliefs
    - 2 parts: graph + conditional distributions for each vertex
    - $\approx 2^{n^2}$ candidate directed acyclic graphs (DAGs), one of which is $\mathcal{G}^*$

[Pea88] Judea Pearl. *Probabilistic reasoning in intelligent systems*. Morgan Kaufmann, 1988.

# The ALARM network [BSCC89]

- Handcrafted Bayesian network encoding medical knowledge
- Purpose: Provide an alarm message for patient monitoring





[BSCC89] Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. *The alarm monitoring system: A case study with two probabilistic inference techniques for belief network*. Second European Conference on Artificial Intelligence in Medicine (AIME), 1989

# The ALARM network [BSCC89]

**A sample consultation**

ALARM is a data-driven system. Simulating an anesthesia monitor, ALARM accepts a set of physiologic measurements. An example would be as follows: blood pressure 120/80 mmHg, heart rate 80/min, inspired oxygen concentration 50%, tidal volume 500 ml, respiratory rate 10/min, breathing pressure 50 mbar, and measured minute ventilation 1.2 l/min. These measurements are categorized into 'low', 'normal', 'high', etc. and text messages are generated when measurements are outside of their normal range. These messages will then appear in the *Warning* and *Caution* fields of the monitor depending on their importance *(Fig. 3)*. In the given example, the high breathing pressure of 50 mbar imposes a direct danger to the patient and a warning is issued. The low minute ventilation is less immediate and is displayed as a caution only.

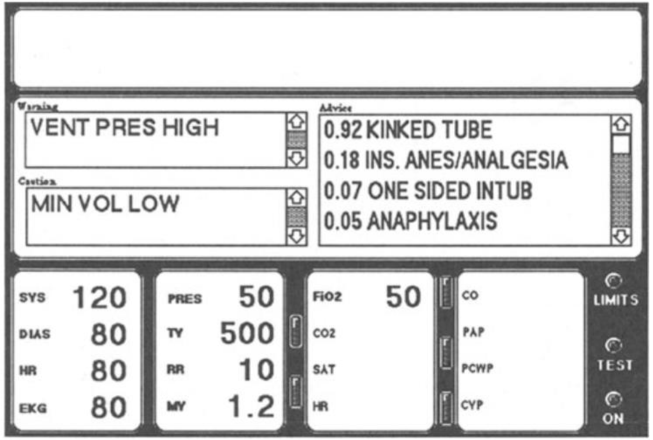
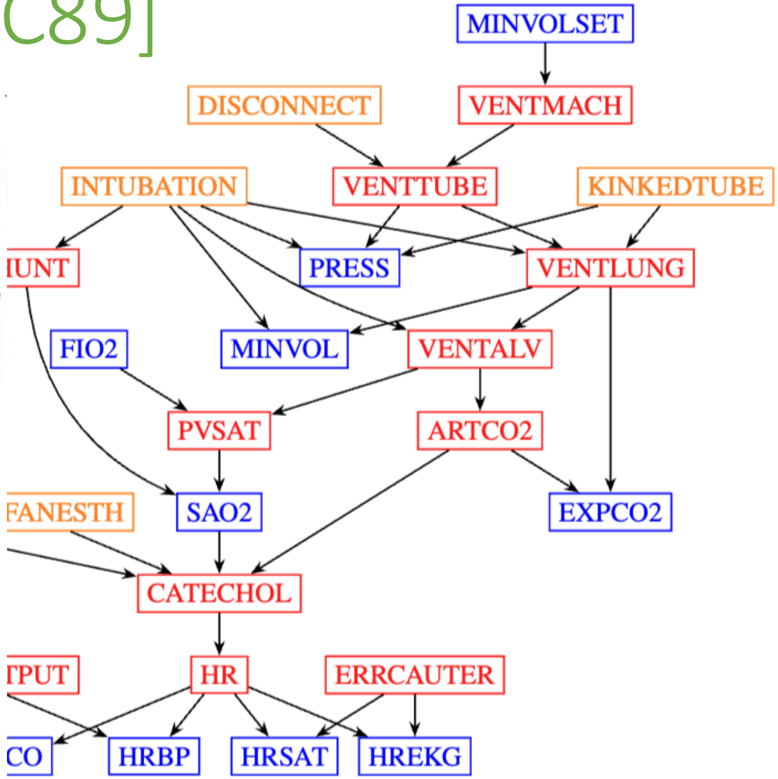



*Fig. 3*

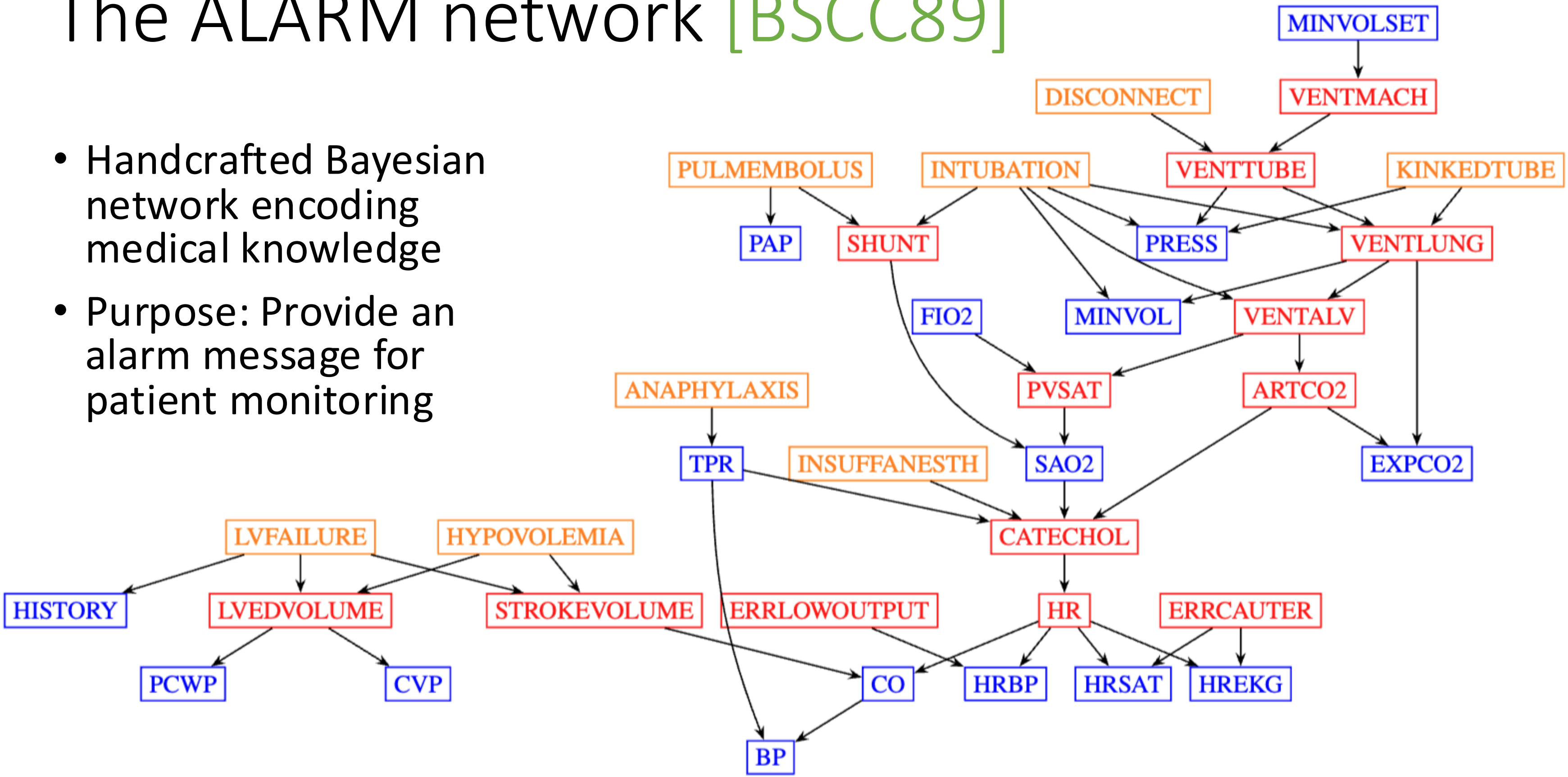
*ALARM simulates an anesthesia monitor. It takes patient measurements, displays warning and caution messages, and lists a differential diagnosis.*




[BSCC89] Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. *The alarm monitoring system: A case study with two probabilistic inference techniques for belief network*. Second European Conference on Artificial Intelligence in Medicine (AIME), 1989

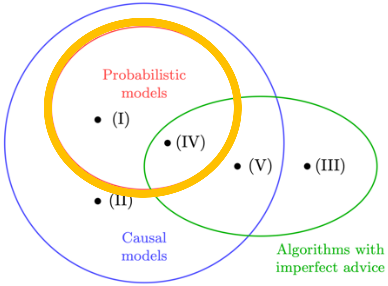
# The ALARM network [BSCC89]

- Handcrafted Bayesian network encoding medical knowledge
- Purpose: Provide an alarm message for patient monitoring



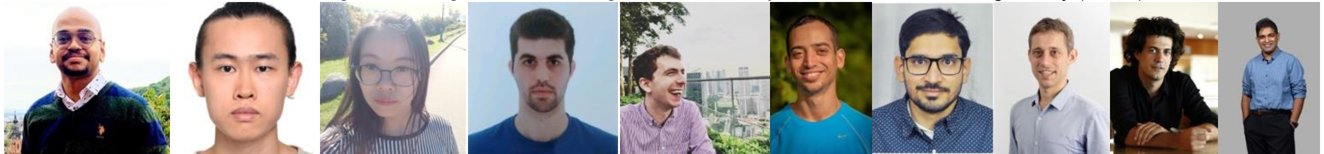

[BSCC89] Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. *The alarm monitoring system: A case study with two probabilistic inference techniques for belief network*. Second European Conference on Artificial Intelligence in Medicine (AIME), 1989

# (I): Probabilistic models

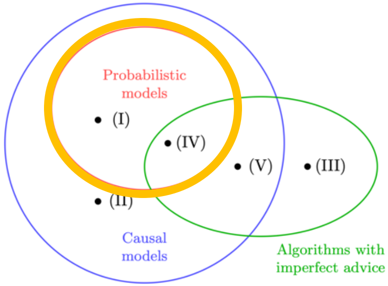



- Suppose data distribution $\mathcal{P}$ is described by Bayesian network
  - NP-hard to find "score maximizing" DAG from data [Chi96] and to decide whether $\mathcal{P}$ can be described by a DAG with p parameters [CHM04]
  - Even under the promise that $\mathcal{P}$ can be described by a DAG with p parameters, it is NP-hard to find such a parameter-bounded DAG [BCGM25]
  - We also have some PAC-style finite sample results in learning the structure and parameters of Bayesian network for $\mathcal{P}$ [BCG+22, DDKC23, CYBC24]
  - Insight: If network's in-degree is bounded, we can use less samples

[Chi96] David Maxwell Chickering. *Learning Bayesian networks is NP-complete*. Lecture Notes in Statistics, vol 112, 1996
[CHM04] Max Chickering, David Heckerman, and Chris Meek. *Large-sample learning of Bayesian networks is NP-hard*. Journal of Machine Learning Research (JMLR), 2004
[BCGM25] Arnab Bhattacharyya, Davin Choo, Sutanu Gayen, Dimitrios Myrisiotis. *Learnability of Parameter-Bounded Bayes Nets*. AAAI Conference on Artificial Intelligence (AAAI), 2025
[BCG+22] Arnab Bhattacharyya, Davin Choo, Rishikesh Gajjala, Sutanu Gayen, Yuhao Wang. *Learning Sparse Fixed-Structure Gaussian Bayesian Networks*. International Conference on Artificial Intelligence and Statistics (AISTATS), 2022
[DDKC23] Yuval Dagan, Constantinos Daskalakis, Anthimos-Vardis Kandiros, Davin Choo. *Learning and Testing Latent-Tree Ising Models Efficiently*. Conference on Learning Theory (COLT), 2023
[CYBC24] Davin Choo, Joy Qiping Yang, Arnab Bhattacharyya, Clément L. Canonne. *Learning bounded degree polytrees with samples*. International Conference on Algorithmic Learning Theory (ALT), 2024

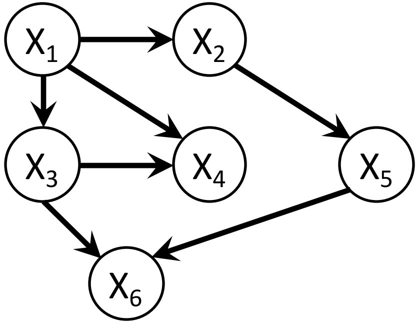


# A glimpse of [BCG+22]




Insight: If network's in-degree is bounded, we can use less samples

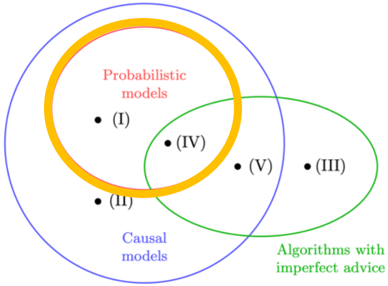
- Suppose we get i.i.d. samples from a linear DAG with Gaussian noise




$$\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix}$$

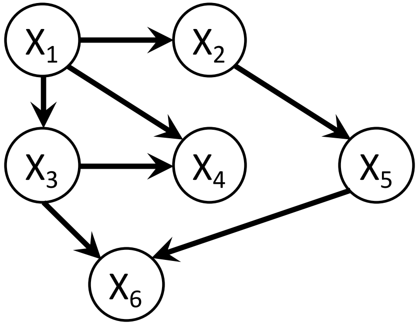
$$X_i = \begin{cases} \eta_i + \sum_{X_j \in \mathrm{Pa}(X_i)} a_{i,j} X_j & \text{if } \mathrm{Pa}(X_i) \neq \emptyset \\ \eta_i & \text{if } \mathrm{Pa}(X_i) = \emptyset \end{cases}$$

# A glimpse of [BCG+22]




Insight: If network's in-degree is bounded, we can use less samples

- Suppose we get i.i.d. samples from a linear DAG with Gaussian noise

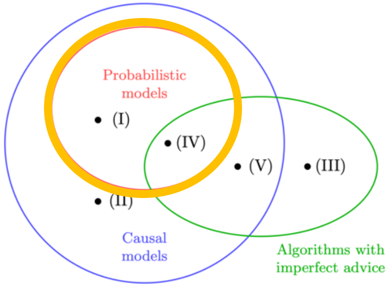



$$\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix}$$

| | $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ |
|---|---|---|---|---|---|---|
| Sample 1 | 1.24 | 1.08 | 0.229 | -0.846 | 0.307 | 1.201 |
| Sample 2 | -0.614 | 0.552 | 0.758 | 1.77 | 1.646 | 0.375 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

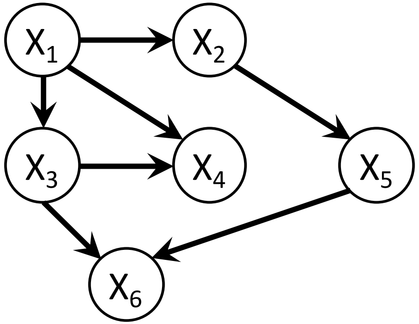
How many samples would we need to learn the coefficients and noise?

# A glimpse of [BCG+22]




Insight: If network's in-degree is bounded, we can use less samples

- Suppose we get i.i.d. samples from a linear DAG with Gaussian noise




$$\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix}$$

$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{\eta}$

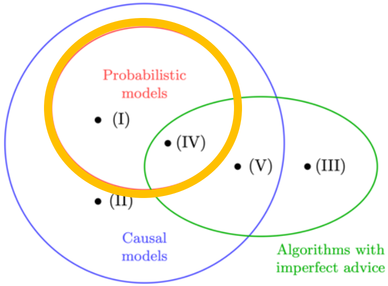
$\Rightarrow \boldsymbol{X} = (\boldsymbol{I}_n - \boldsymbol{A})^{-1} \boldsymbol{\eta}$

$\Rightarrow \boldsymbol{X}$ is a multivariate Gaussian, in general

$\Rightarrow$ Need $\widetilde{\Omega}\left(\frac{n^2}{\varepsilon^2}\right)$ i.i.d. samples to learn $\boldsymbol{X}$ "$\varepsilon$-well"

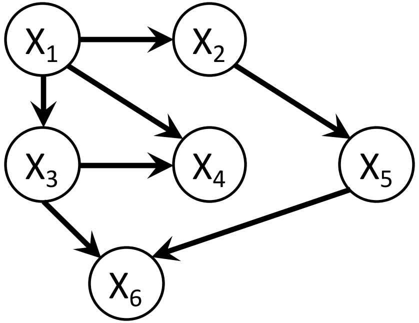
A rough intuition: All $n^2$ covariance matrix entries "matter", in general

# A glimpse of [BCG+22]




Insight: If network's in-degree is bounded, we can use less samples

- Turns out $\tilde{O}\left(\frac{nd}{\varepsilon^2}\right)$ samples suffice with just least squares at each node




$$\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \end{pmatrix}$$

$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{\eta}$

$\Rightarrow \boldsymbol{X} = (\boldsymbol{I}_n - \boldsymbol{A})^{-1}\boldsymbol{\eta}$

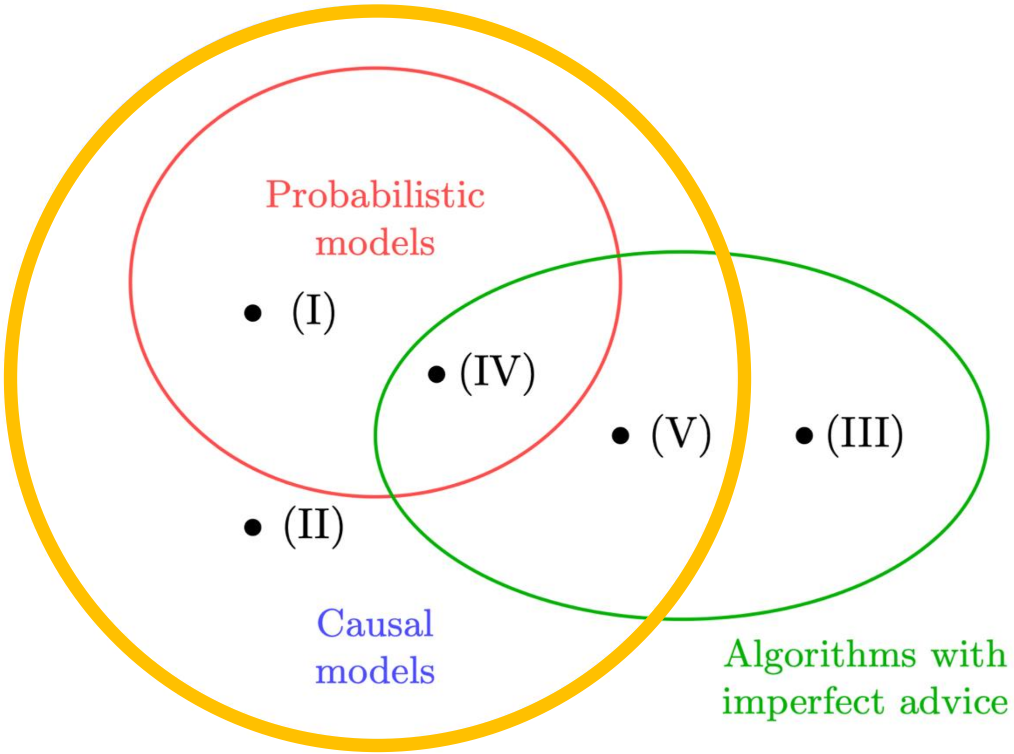
$\Rightarrow \boldsymbol{X}$ is a multivariate Gaussian, in general

- Here, max in-degree $d = 2$

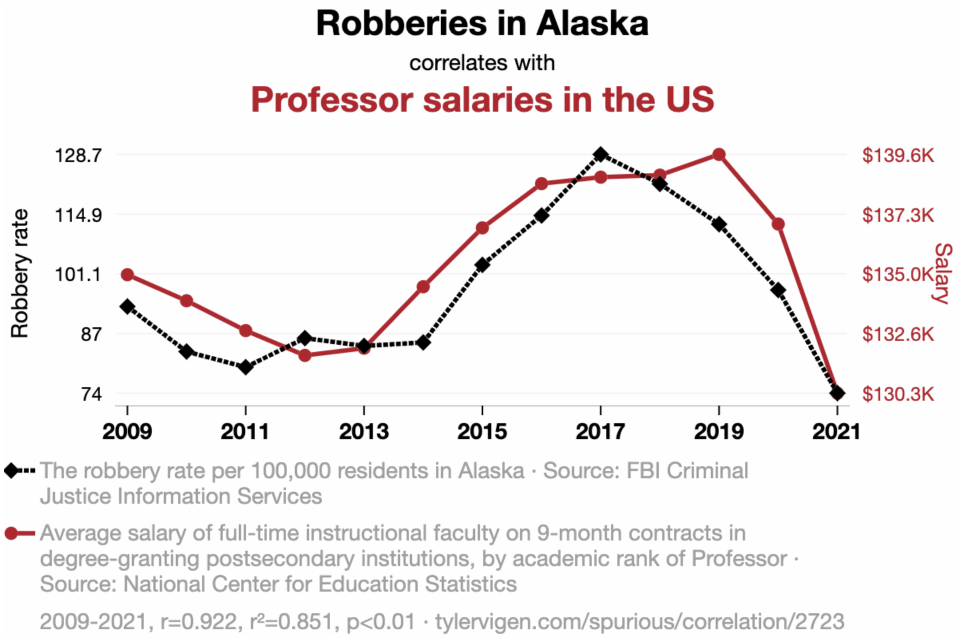
$\Rightarrow$ Need $\widetilde{\Omega}\left(\frac{n^2}{\varepsilon^2}\right)$ i.i.d. samples to learn $\boldsymbol{X}$ "$\varepsilon$-well"

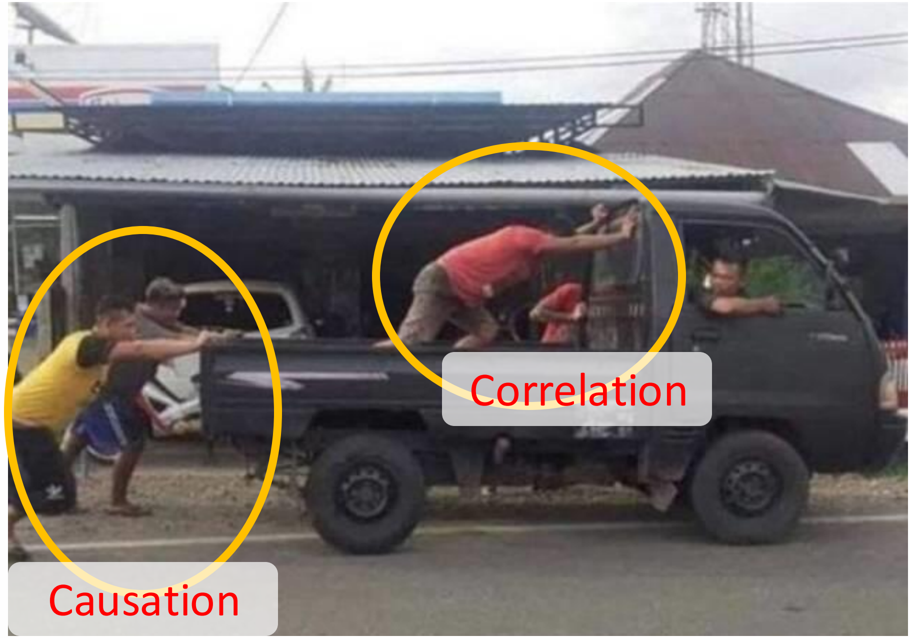
A rough intuition: All $n^2$ covariance matrix entries "matter", in general

# Main themes explored in my PhD thesis



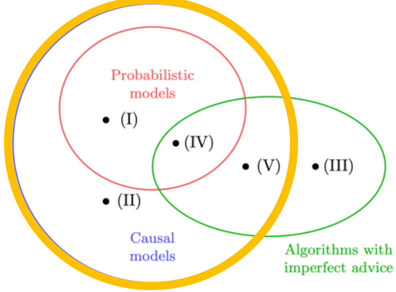
# Correlation does not imply causation



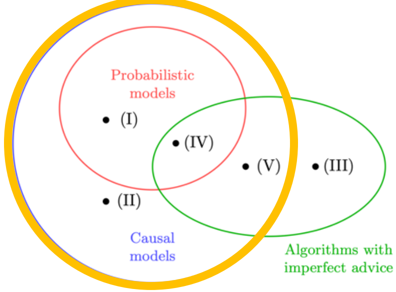

https://tylervigen.com/spurious/correlation/2723_robberies-in-alaska_correlates-with_professor-salaries-in-the-us
https://imgflip.com/memetemplate/327010423/Truck-teamwork

# (II): Causal models



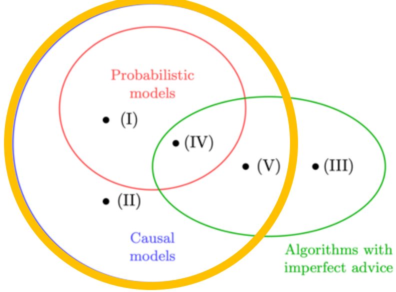

- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
  - Causal effect estimation: Estimate $\mathcal{P}(Y = y \mid do(X = x))$

# (II): Causal models

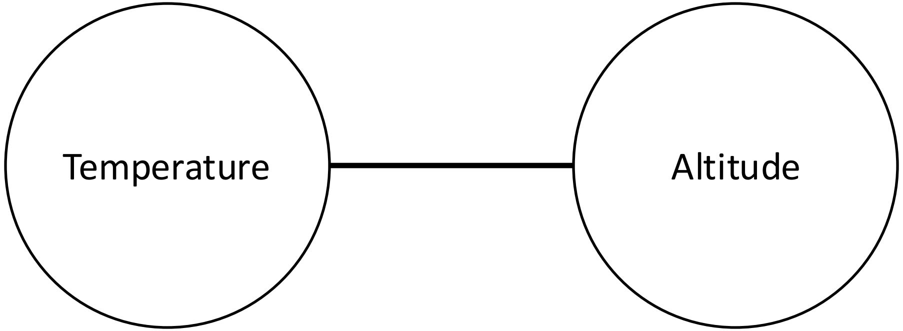



- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
    - Even with infinite observational data, can only determine causal graph up to some equivalence class where all conditional independence relations agree
    - Make distributional/structural assumptions or perform interventions/experiments!

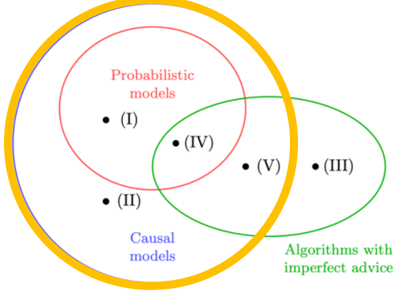
# (II): Causal models




- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
    - Even with infinite observational data, can only determine causal graph up to some equivalence class where all conditional independence relations agree
    - Make distributional/structural assumptions or perform interventions/experiments!

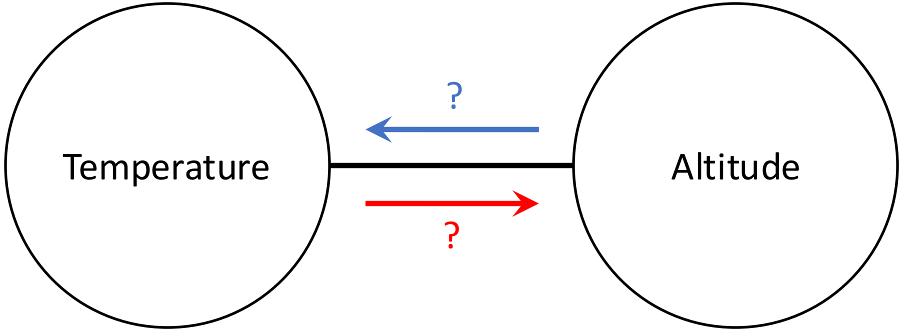



# (II): Causal models



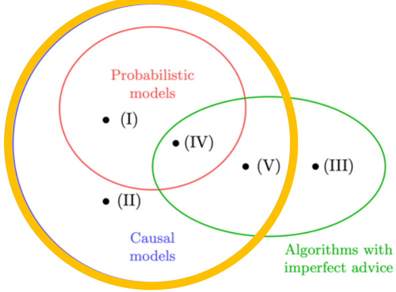

- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
    - Even with infinite observational data, can only determine causal graph up to some equivalence class where all conditional independence relations agree
    - Make distributional/structural assumptions or perform interventions/experiments!

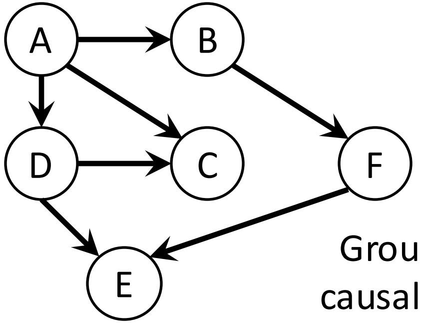
If we change altitude, temperature changes



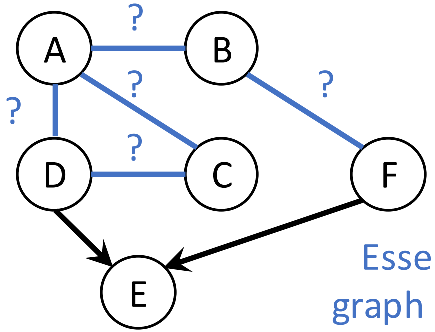

If we change temperature, altitude changes

# Causal discovery via interventions



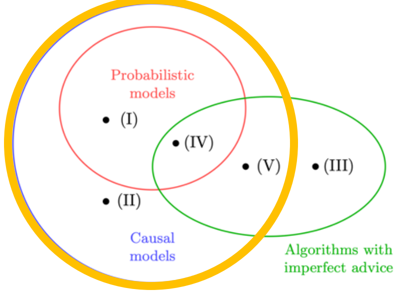

Ground truth causal graph $\mathcal{G}^*$

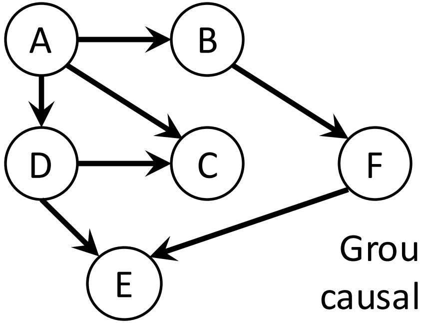



Essential graph $\mathcal{E}(\mathcal{G}^*)$

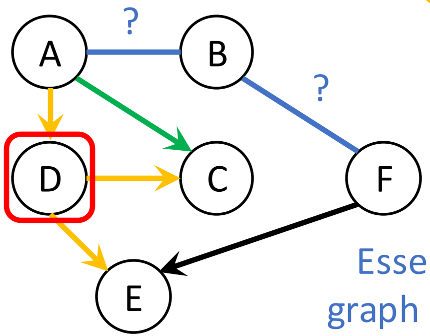
- Want: Recover $\mathcal{G}^*$ starting from partially oriented $\mathcal{E}(\mathcal{G}^*)$ from observational data

# Causal discovery via interventions



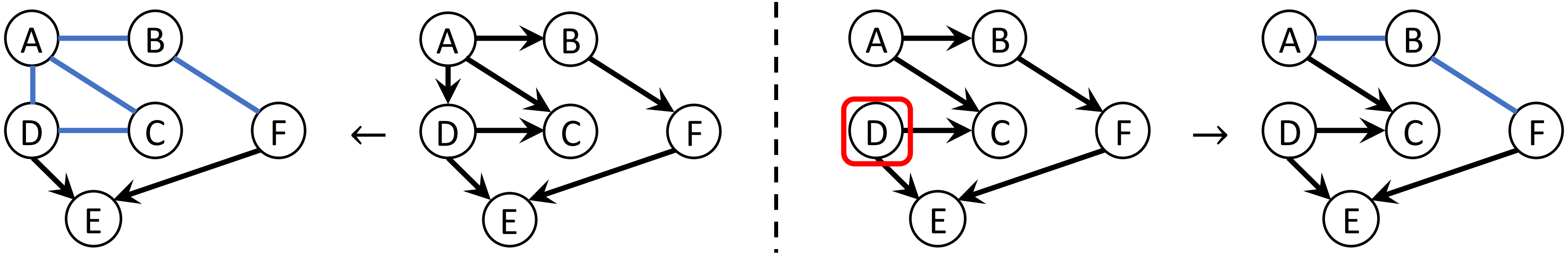




Ground truth causal graph $\mathcal{G}^*$

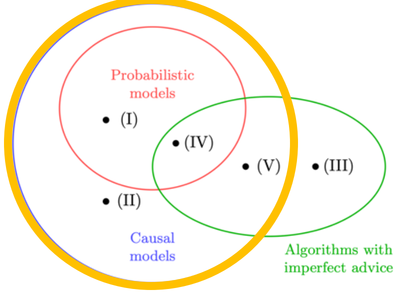


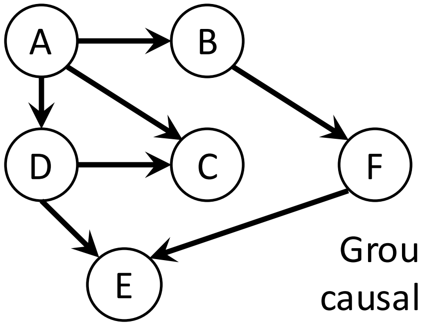

Essential graph $\mathcal{E}(\mathcal{G}^*)$

- Want: Recover $\mathcal{G}^*$ starting from partially oriented $\mathcal{E}(\mathcal{G}^*)$ from observational data
- Interventions reveal arc orientations (incident arcs + Meek rules)
- **Goal: Recover $\mathcal{G}^*$ using as few interventions as possible**

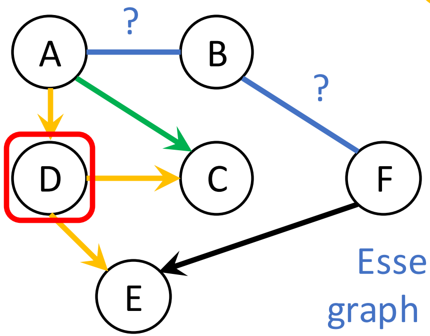


# Causal discovery via interventions




Ground truth causal graph $\mathcal{G}^*$

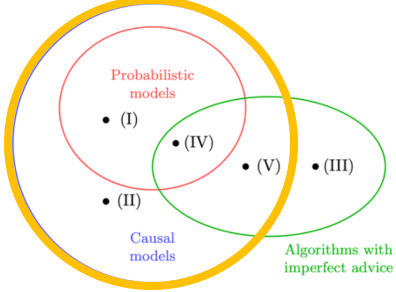



Essential graph $\mathcal{E}(\mathcal{G}^*)$

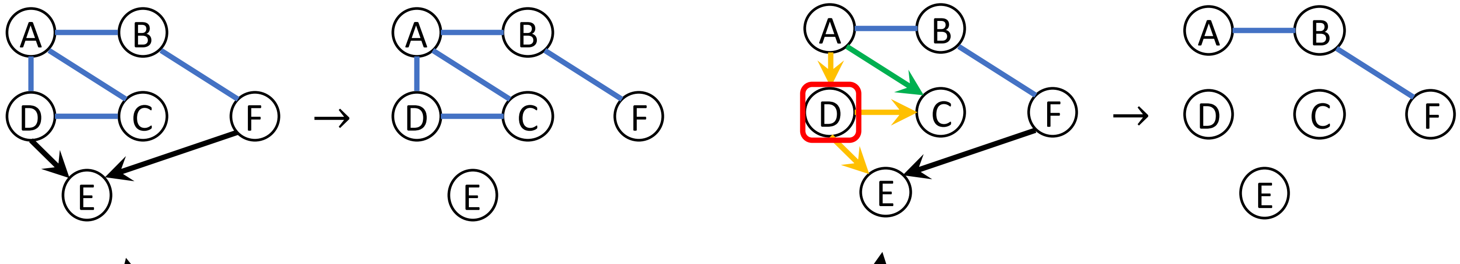
- Want: Recover $\mathcal{G}^*$ starting from partially oriented $\mathcal{E}(\mathcal{G}^*)$ from observational data
- Interventions reveal arc orientations (incident arcs + Meek rules)
- **Goal: Recover $\mathcal{G}^*$ using as few interventions as possible**
- We have some results regarding how to design algorithms to perform optimal adaptive interventions under various scenarios [CSB22, CS23a, CGB23, CS23b, CS23c, CSU24]
- Insight: Reduce to graph / set cover problem with specialized causal operations

[CSB22] Davin Choo, Kirankumar Shiragur, Arnab Bhattacharyya. *Verification and search algorithms for causal DAGs*. Conference on Neural Information Processing Systems (NeurIPS), 2022.
[CS23a] Davin Choo, Kirankumar Shiragur. *Subset verification and search algorithms for causal DAGs*. International Conference on Artificial Intelligence and Statistics (AISTATS), 2023.
[CGB23] Davin Choo, Themistoklis Gouleakis, Arnab Bhattacharyya. *Active causal structure learning with advice*. International Conference on Machine Learning (ICML), 2023.
[CS23b] Davin Choo, Kirankumar Shiragur. *New metrics and search algorithms for weighted causal DAGs*. International Conference on Machine Learning (ICML), 2023.
[CS23c] Davin Choo, Kirankumar Shiragur. *Adaptivity Complexity for Causal Graph Discovery*. Conference on Uncertainty in Artificial Intelligence (UAI), 2023.
[CSU24] Davin Choo, Kirankumar Shiragur, Caroline Uhler. *Causal discovery under off-target interventions*. International Conference on Artificial Intelligence and Statistics (AISTATS), 2024.

# A glimpse of [CSB22]

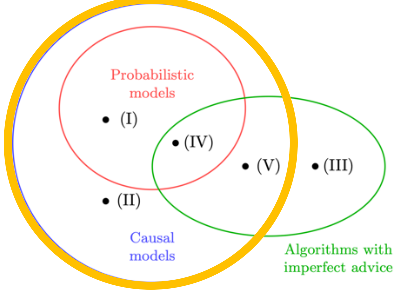



- Insight: Frame as graph problem with causal operations
- Known facts and observations (say n vertices)
  - Remove directed edges in essential graph → chordal graph $G$
  - If $G$ has no (undirected) edges, then whole graph is oriented
  - Intervention on vertex v → Orient all edges incident to v (possibly more)

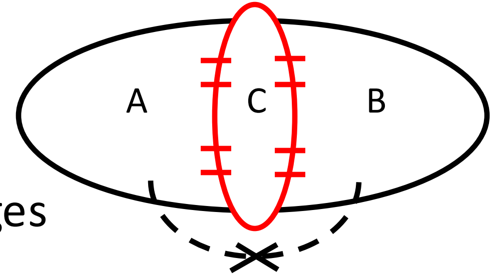



Essential graphs from earlier slides

# A glimpse of [CSB22]

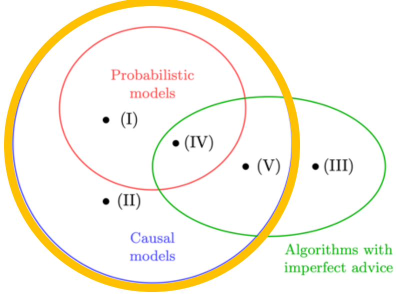



- Insight: Frame as graph problem with causal operations
- Known facts and observations (say n vertices)
  - Remove directed edges in essential graph → chordal graph $G$
  - If $G$ has no (undirected) edges, then whole graph is oriented
  - Intervention on vertex v → Orient all edges incident to v (possibly more)
- Chordal graph separators [GRE84]
  - $|A|, |B| \leq \frac{|G|}{2}$ and $C$ is a clique, i.e., $|C| \leq \omega(G)$
  - Intervene on vertices in $C$ one by one
  - Repeat $O(\log n)$ times → $G$ will have no more edges
- We also show that this is optimal in worst case



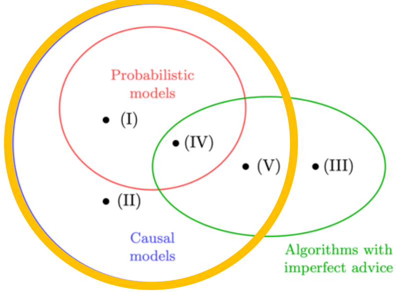

[GRE83] John R. Gilbert, Donald J. Rose, Anders Edenbrandt. *A Separator Theorem for Chordal Graphs*. SIAM Journal on Algebraic Discrete Methods, 1984.

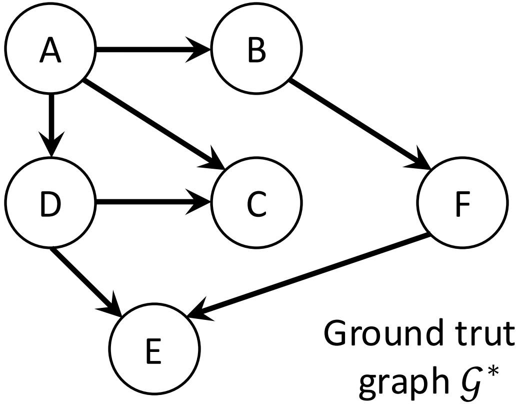
# (II): Causal models



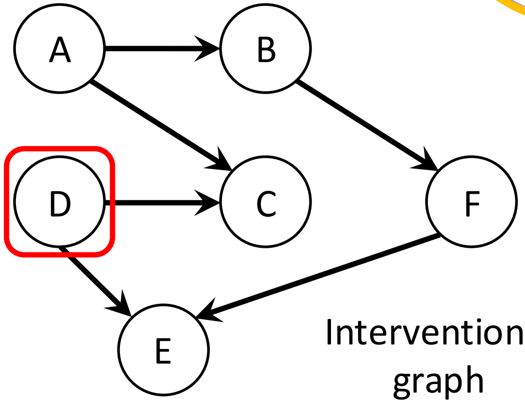

- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
    - Even with infinite observational data, can only determine causal graph up to some equivalence class where all conditional independence relations agree
    - Make distributional/structural assumptions or perform interventions/experiments!
  - Causal effect estimation: Estimate $\mathcal{P}(Y = y \mid do(X = x))$
    - Typically, a 2-stage process: learn $\mathcal{G}^*$, then apply closed-form formulas

# Causal identification (the 2nd step)




Ground truth graph $\mathcal{G}^*$



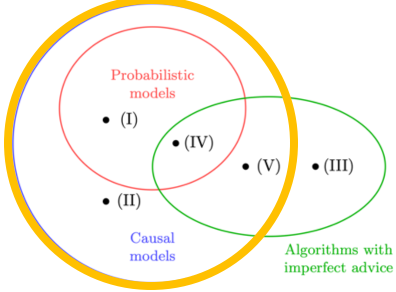

Interventional graph

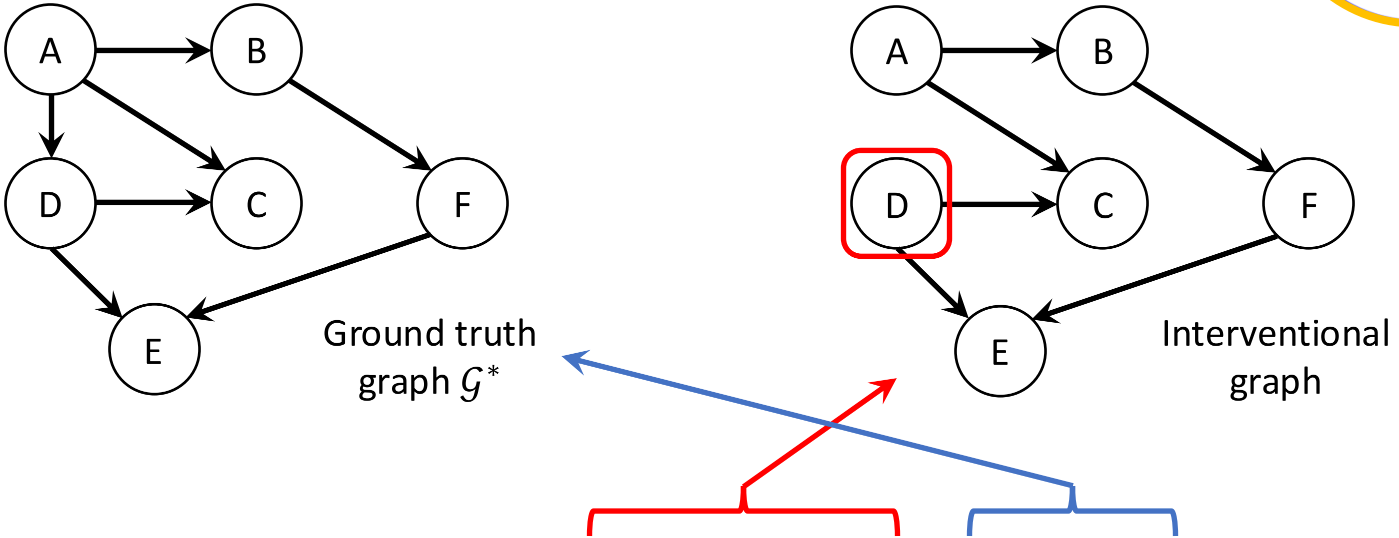
$$\mathcal{P}(E = e \mid do(D = d^*))$$

Interventional query

What is probability of $E = e$ when we fix $D = d^*$?

# Causal identification (the 2nd step)



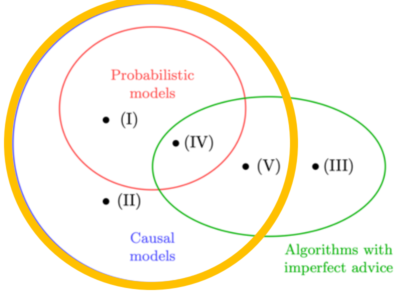

$\mathcal{P}(E = e \mid do(D = d^*)) = \mathcal{P}(e \mid do(d^*)) \neq \mathcal{P}(e \mid d^*)$ in general

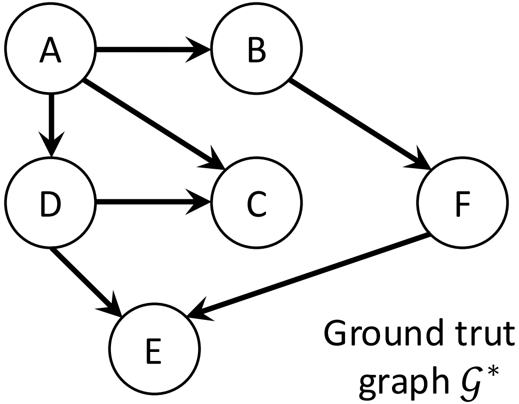
Interventional query

What is probability of $E = e$ when we fix $D = d^*$?

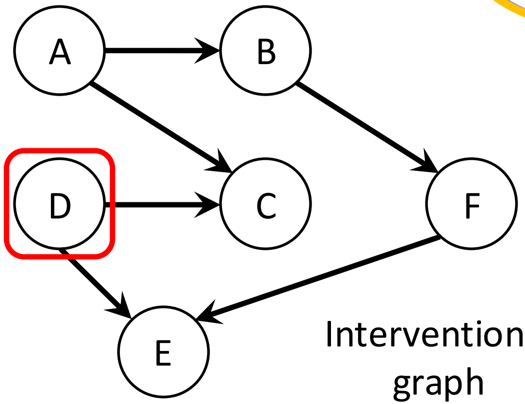
Need to draw samples from interventional graph, i.e., perform experiment and measure

# Causal identification (the 2nd step)




Ground truth graph $\mathcal{G}^*$




Interventional graph

Because of structure of $\mathcal{G}^*$

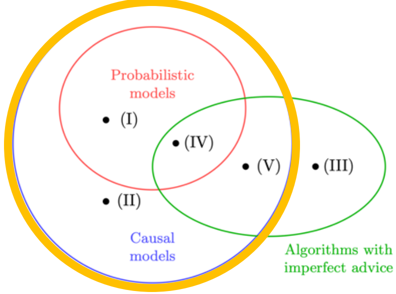
$$\mathcal{P}(E = e \mid do(D = d^*)) = \mathcal{P}(e \mid do(d^*)) = \int \mathcal{P}(e \mid d^*, a) \cdot \mathcal{P}(a)\, da$$

<u>Interventional query</u>

What is probability of $E = e$ when we fix $D = d^*$?

Just observational terms!

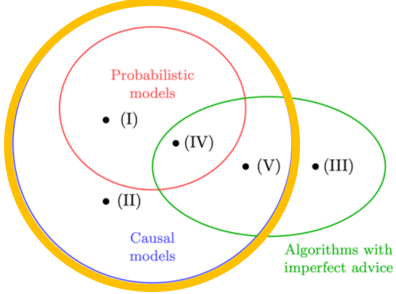
# (II): Causal models




- Two fundamental problems in causal inference
  - Causal graph discovery: Recover true causal graph $\mathcal{G}^*$
    - Even with infinite observational data, can only determine causal graph up to some equivalence class where all conditional independence relations agree
    - Make distributional/structural assumptions or perform interventions/experiments!
  - Causal effect estimation: Estimate $\mathcal{P}(Y = y \mid do(X = x))$
    - Typically, a 2-stage process: learn $\mathcal{G}^*$, then apply closed-form formulas
    - [CSBS24] This is suboptimal as it may require strong assumptions and a lot of samples
    - Insight: "weak edges" shouldn't affect much for PAC-style results

[CSBS24] Davin Choo, Chandler Squires, Arnab Bhattacharyya, and David Sontag. *Probably approximately correct high-dimensional causal effect estimation given a valid adjustment set*. Under submission, 2024.

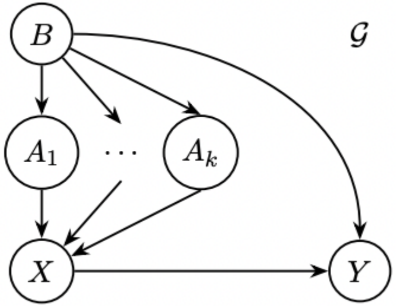
# A glimpse of [CSBS24]



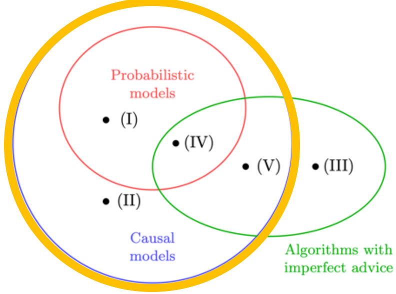

Insight: "weak edges" shouldn't affect much for PAC-style results

- Suppose we draw observational samples from this causal graph of binary variables and wish to estimate interventional effect $\mathcal{P}(Y = y \mid do(X = x^*))$
- Let's estimate $\mathcal{P}(y \mid do(x^*))$ via $\sum_s \mathcal{P}(y|x^*, z)\mathcal{P}(z)$ for some subset $Z \subseteq V$

$V$ excludes $X$ and $Y$



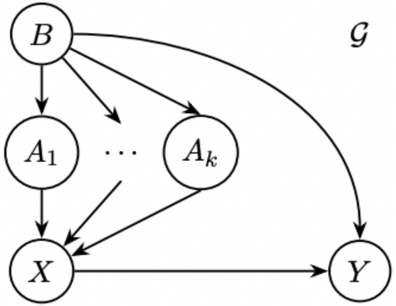
# A glimpse of [CSBS24]



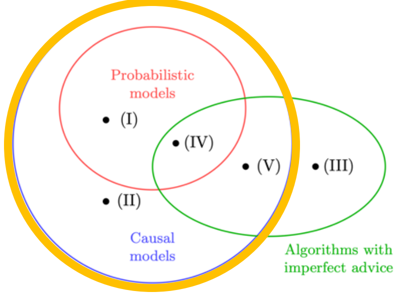

Insight: "weak edges" shouldn't affect much for PAC-style results

- Suppose we draw observational samples from this causal graph of binary variables and wish to estimate interventional effect $\mathcal{P}(Y = y \mid do(X = x^*))$
- Let's estimate $\mathcal{P}(y \mid do(x^*))$ via $\sum_s \mathcal{P}(y|x^*, z)\mathcal{P}(z)$ for some subset $Z \subseteq V$
  - Valid when $Z$ is $\{B, A_1, \ldots, A_k\}$ or $\{A_1, \ldots, A_k\}$ or $\{B\}$, for <u>any</u> underlying $\mathcal{P}$
  - $\{B\}$ is the best: smaller set = less samples for an accurate estimate
  - But… we don't know the graph!

$V$ excludes $X$ and $Y$



# A glimpse of [CSBS24]

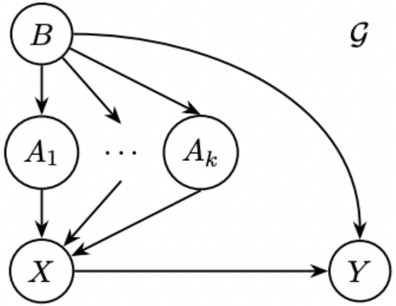



Insight: "weak edges" shouldn't affect much for PAC-style results

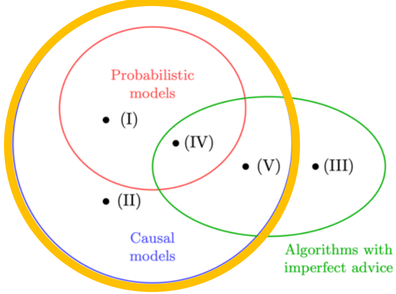
- Suppose we draw observational samples from this causal graph of binary variables and wish to estimate interventional effect $\mathcal{P}(Y = y \mid do(X = x^*))$
- Let's estimate $\mathcal{P}(y \mid do(x^*))$ via $\sum_s \mathcal{P}(y|x^*, z)\mathcal{P}(z)$ for some subset $Z \subseteq V$
  - Valid when $Z$ is $\{B, A_1, \ldots, A_k\}$ or $\{A_1, \ldots, A_k\}$ or $\{B\}$, for <u>any</u> underlying $\mathcal{P}$
  - $\{B\}$ is the best: smaller set = less samples for an accurate estimate
  - But... we don't know the graph!

$V$ excludes $X$ and $Y$

- What CI tests + do-calculus that will validate the estimate?
  - "Markov blanket": $X \perp\!\!\!\perp S \backslash V \mid S \rightarrow$ Get $S = \{A_1, \ldots, A_k\}$
  - "Screening set": $Y \perp\!\!\!\perp S \backslash S' \mid X \cup S'$ and $X \perp\!\!\!\perp S' \backslash S \mid S \rightarrow$ Get $S' = \{B\}$



# A glimpse of [CSBS24]

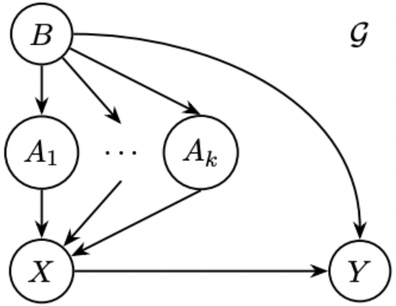



Insight: "weak edges" shouldn't affect much for PAC-style results

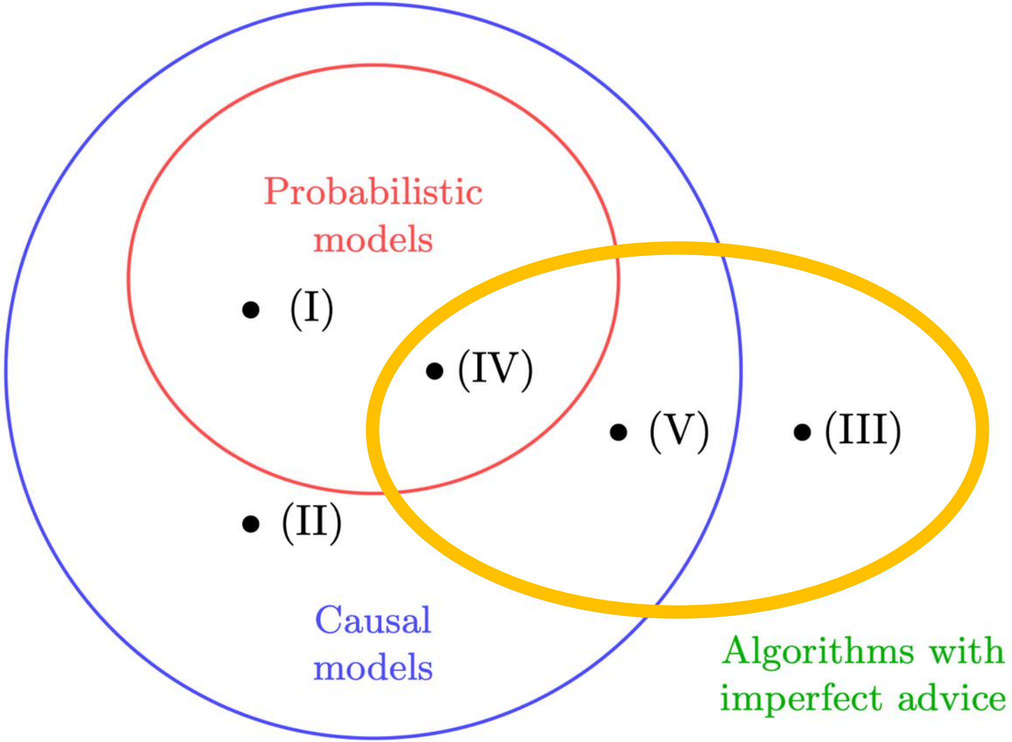
- Suppose we draw observational samples from this causal graph of binary variables and wish to estimate interventional effect $\mathcal{P}(Y = y \mid do(X = x^*))$
- Let's estimate $\mathcal{P}(y \mid do(x^*))$ via $\sum_s \mathcal{P}(y|x^*, z)\mathcal{P}(z)$ for some subset $Z \subseteq V$
  - Valid when $Z$ is $\{B, A_1, \ldots, A_k\}$ or $\{A_1, \ldots, A_k\}$ or $\{B\}$, for any underlying $\mathcal{P}$
  - $\{B\}$ is the best: smaller set = less samples for an accurate estimate
  - But… we don't know the graph!

$V$ excludes $X$ and $Y$

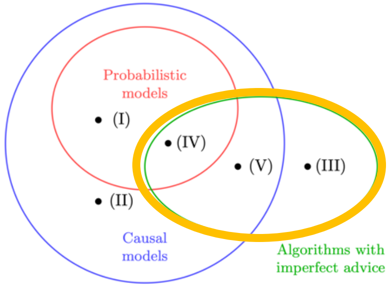
- What CI tests + do-calculus that will validate the estimate?
  - "Markov blanket": $X \perp\!\!\!\perp S \setminus V \mid S \rightarrow$ Get $S = \{A_1, \ldots, A_k\}$
  - "Screening set": $Y \perp\!\!\!\perp S \setminus S' \mid X \cup S'$ and $X \perp\!\!\!\perp S' \setminus S \mid S \rightarrow$ Get $S' = \{B\}$
  - Approximate conditional independence test → PAC estimate



# Main themes explored in my PhD thesis



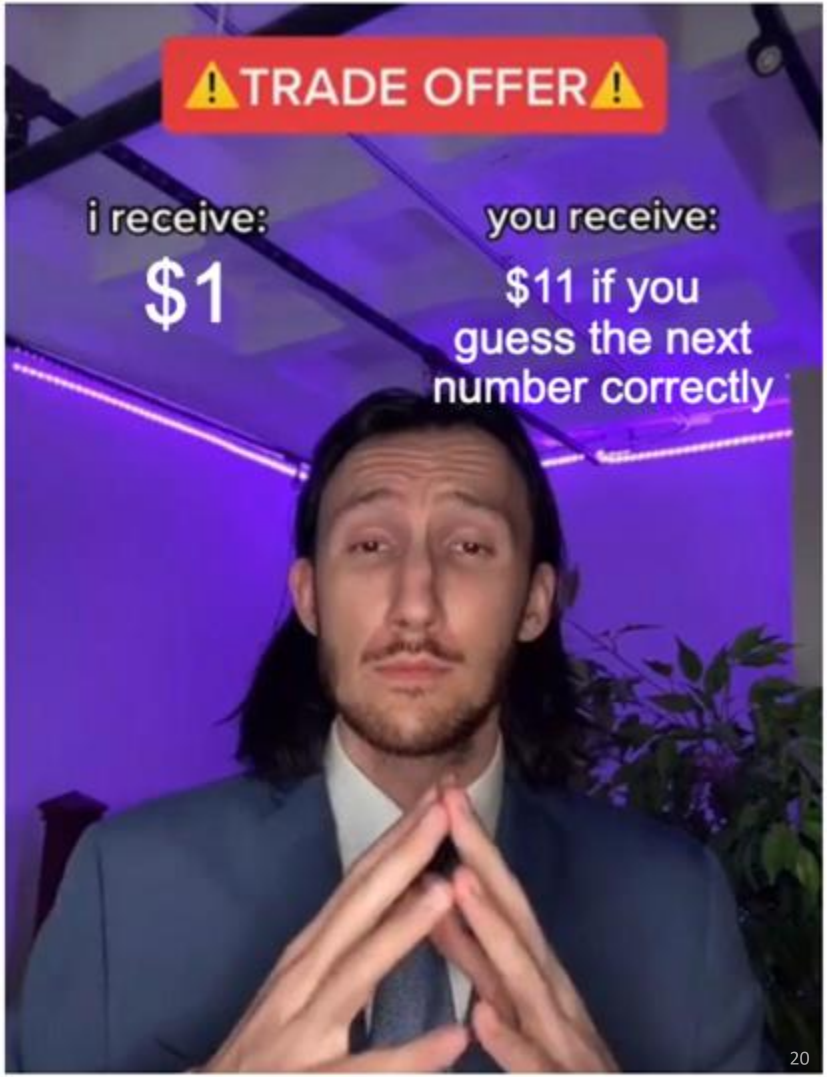
# (III/IV/V): Algorithms with advice




- Two key performance measures
  - Consistency: If advice is “perfect”, how good are things?
  - Robustness: If advice is “garbage”, how bad are things?
- Challenge: We don’t know how good the given advice is a priori!

# Detour: Let’s make a deal

- There are 10 numbers in the universe U = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
- There is an underlying process $\mathcal{P}$ that generates i.i.d. samples from U
  - i.e., We can observe a sequence such as 1, 6, 3, 6, 2, 8, 0, 3, 9, 5, 4, …
- What property of $\mathcal{P}$ will make this deal <u>profitable in expectation</u>?




https://imgflip.com/memegenerator/309868304/Trade-Offer

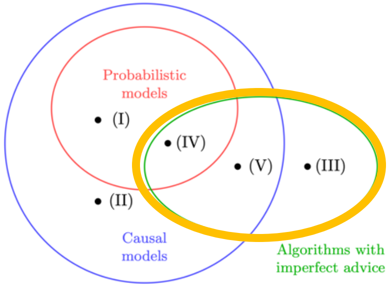
# Detour: Property testing land

- How to test if $\mathcal{P}$ is the uniform distribution over U?
  - Say, we only care about constant success probability (can be amplified)
- Learning a $\varepsilon$-close $\hat{\mathcal{P}}$ then check: $\Theta\left(\frac{|U|}{\varepsilon^2}\right)$ i.i.d. samples from $\mathcal{P}$
- Uniformity testing requires $\Theta\left(\frac{\sqrt{|U|}}{\varepsilon^2}\right)$ i.i.d. samples from $\mathcal{P}$
  - If $\mathcal{P}$ is uniform, output YES w.p. $\geq \frac{2}{3}$
  - If $\mathcal{P}$ is $\varepsilon$-far from uniform, output NO w.p. $\geq \frac{2}{3}$
  - Many existing proofs for this bound. E.g., look at collisions in samples
- See also [Can22] for an excellent property testing survey

→ Allowed to output arbitrarily if not uniform, yet not "far from uniform"

[Can22] Clément L. Canonne. *Topics and Techniques in Distribution Testing: A Biased but Representative Sample*. Foundations and Trends® in Communications and Information Theory, 2022.

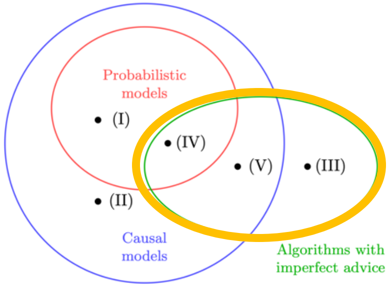
# (III/IV/V): Algorithms with advice




- Two key performance measures
  - Consistency: If advice is "perfect", how good are things?
  - Robustness: If advice is "garbage", how bad are things?
- Challenge: We don't know how good the given advice is a priori!
- Insight: "Testing can be cheaper than learning" → TestAndAct
  - [CGLB24] TestAndMatch: Improve competitive ratio of online bipartite matching (III)
  - [BCGG24] TestAndOptimize: Improve sample complexity of learning multivariate Gaussians (IV)
  - [CGB23] TestAndSubsetSearch: Reduce num of interventions required for causal graph discovery (V)

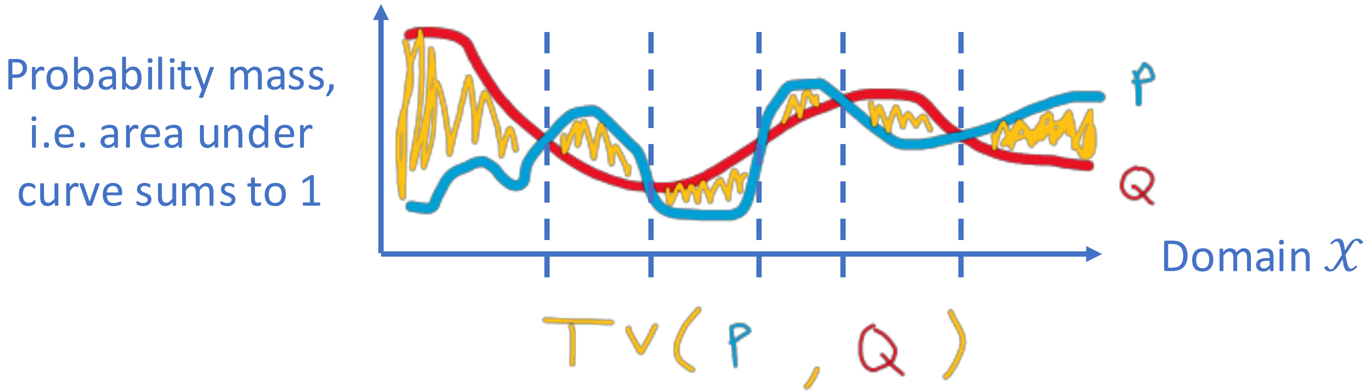

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024.
[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. Under submission, 2024.
[CGB23] Davin Choo, Themistoklis Gouleakis, Arnab Bhattacharyya. *Active causal structure learning with advice*. International Conference on Machine Learning (ICML), 2023.


# A glimpse of [BCGG24]

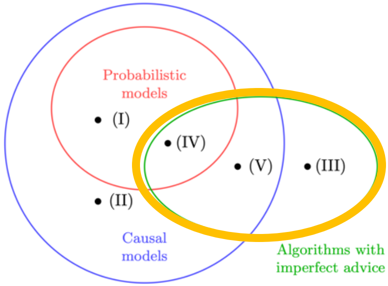



Insight: "Testing can be cheaper than learning"

- Gaussian estimation with i.i.d. samples
  - Given sample access to some underlying distribution $\mathcal{P}$, produce $\hat{\mathcal{P}}$ such that $\mathrm{TV}(\mathcal{P}, \hat{\mathcal{P}}) \leq \varepsilon$ with probability $\geq 1 - \delta$



# A glimpse of [BCGG24]



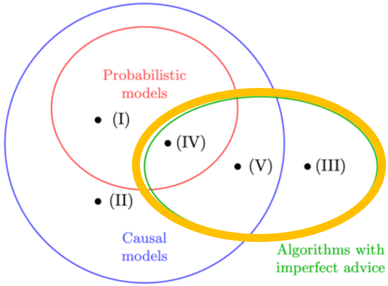

## Insight: "Testing can be cheaper than learning"

- Gaussian estimation with i.i.d. samples
  - Given sample access to some underlying distribution $\mathcal{P}$, produce $\hat{\mathcal{P}}$ such that $\mathrm{TV}(\mathcal{P}, \hat{\mathcal{P}}) \leq \varepsilon$ with probability $\geq 1 - \delta$
- Useful to invoke Pinsker's inequality: $\mathrm{TV}(\mathcal{P}, \hat{\mathcal{P}}) \leq \sqrt{\frac{1}{2} \cdot \mathrm{KL}(\mathcal{P}, \hat{\mathcal{P}})}$
- For multivariate Gaussians over $\mathbb{R}^d$,

$$\mathrm{KL}\left(N(\boldsymbol{\mu}_{\mathcal{P}}, \boldsymbol{\Sigma}_{\mathcal{P}}), N(\boldsymbol{\mu}_{\mathcal{Q}}, \boldsymbol{\Sigma}_{\mathcal{Q}})\right) = \frac{1}{2} \cdot \left[\mathrm{Tr}\left(\boldsymbol{\Sigma}_{\mathcal{Q}}^{-1} \boldsymbol{\Sigma}_{\mathcal{P}}\right) - d + \ln\left(\frac{\det \boldsymbol{\Sigma}_{\mathcal{Q}}}{\det \boldsymbol{\Sigma}_{\mathcal{P}}}\right)\right]$$

- So, we just need to upper bound KL by $\varepsilon^2$

# A glimpse of [BCGG24]




Insight: "Testing can be cheaper than learning"

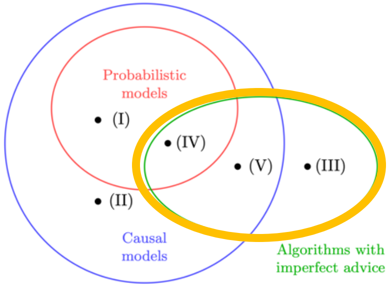
- Let's consider the simple identity covariance setting

$$\mathrm{KL}\big(N(\boldsymbol{\mu}, \mathbf{I}_d), N(\widehat{\boldsymbol{\mu}}, \mathbf{I}_d)\big) = \frac{1}{2} \cdot \|\boldsymbol{\mu} - \widehat{\boldsymbol{\mu}}\|_2^2$$

- Empirical estimator is optimal: need $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ samples to get $\mathrm{KL} \leq \varepsilon^2$

Linear in dimension d

# A glimpse of [BCGG24]




Insight: "Testing can be cheaper than learning"

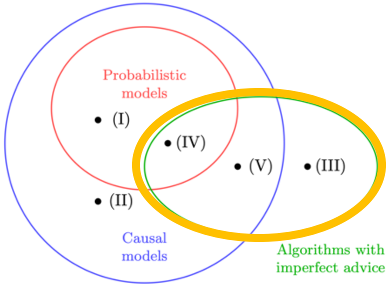
- Let's consider the simple identity covariance setting

$$\mathrm{KL}\left(N(\boldsymbol{\mu}, \mathbf{I}_d), N(\widehat{\boldsymbol{\mu}}, \mathbf{I}_d)\right) = \frac{1}{2} \cdot \|\boldsymbol{\mu} - \widehat{\boldsymbol{\mu}}\|_2^2$$

Linear in dimension d

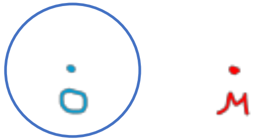
- Empirical estimator is optimal: need $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ samples to get KL $\leq \varepsilon^2$
- Can we do better if someone proposes $\widetilde{\boldsymbol{\mu}}$ as advice?
  - If $\widetilde{\boldsymbol{\mu}} = \boldsymbol{\mu}$, then 0 samples needed, but we cannot blindly trust it
  - W.L.O.G., can treat $\widetilde{\boldsymbol{\mu}} = \mathbf{0}_d$ by pre-processing the samples accordingly
    - Given samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n \sim \mathcal{P}$, consider $(\boldsymbol{y}_1 - \widetilde{\boldsymbol{\mu}}), \ldots, (\boldsymbol{y}_n - \widetilde{\boldsymbol{\mu}})$ instead
    - Once we obtain estimate $\widehat{\boldsymbol{\mu}}$, output $\widehat{\boldsymbol{\mu}} + \widetilde{\boldsymbol{\mu}}$ instead

# A glimpse of [BCGG24]

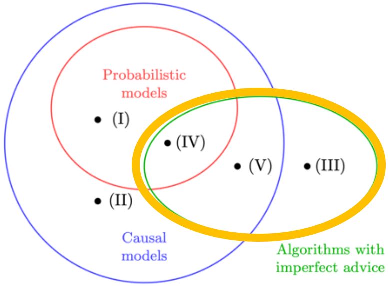



Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$

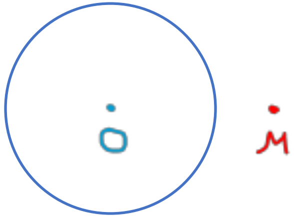


# A glimpse of [B<u>**C**</u>GG24]

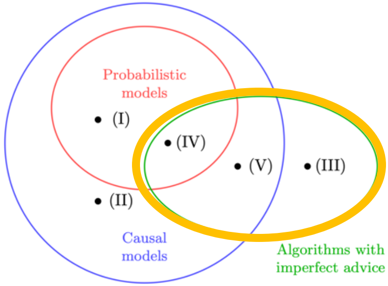



Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$

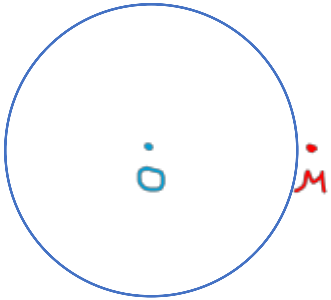


# A glimpse of [BCGG24]

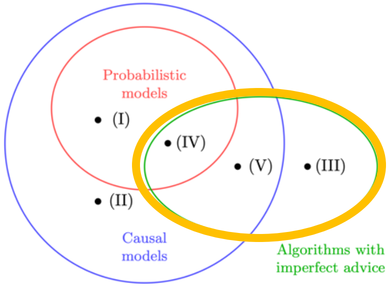



Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$

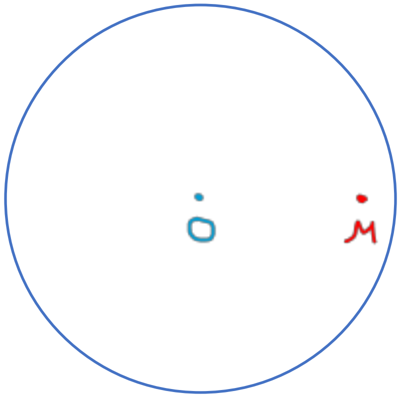


# A glimpse of [BCGG24]

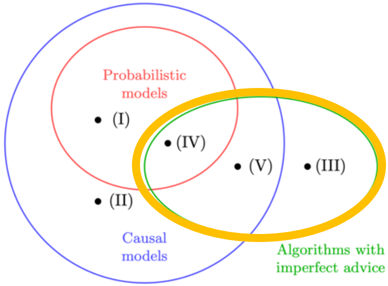



Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$



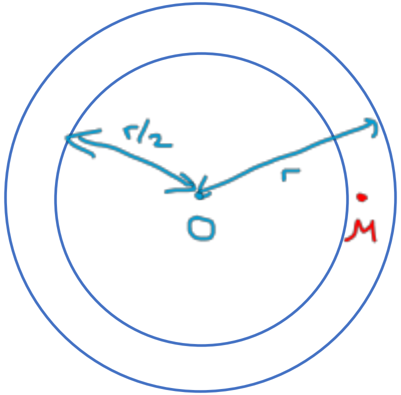
# A glimpse of [BCGG24]



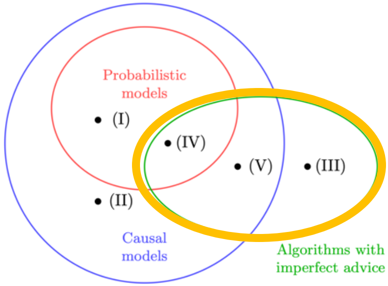

Each test uses $\tilde{O}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$, as compared to $\tilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ for empirical estimator

Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$



# A glimpse of [BCGG24]

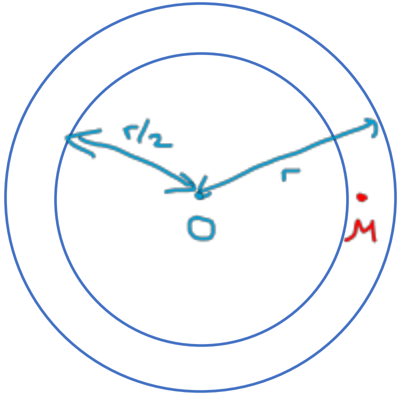



Each test uses $\tilde{\mathcal{O}}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$, as compared to $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ for empirical estimator

Insight: "Testing can be cheaper than learning"

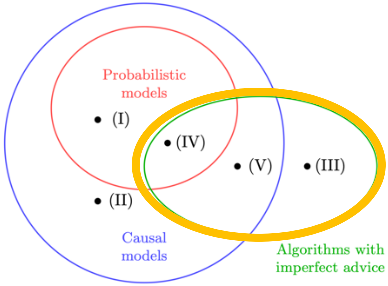
- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$
  - For technical reasons, we need to estimate $\|\boldsymbol{\mu}\|_1$ with some $\lambda$ instead
  - Then, using i.i.d. samples $\boldsymbol{y}_1, \dots, \boldsymbol{y}_n$ from $\mathcal{P}$, solve LASSO in poly time:

$$\widehat{\boldsymbol{\mu}} = \operatorname{argmin}_{\|\boldsymbol{\beta}\|_1 \leq r} \frac{1}{n} \sum_{i=1}^{n} \|\boldsymbol{y}_i - \boldsymbol{\beta}\|_2^2$$




  - When $\|\boldsymbol{\mu}\|_1$ is sufficiently small, our method provably uses $\tilde{o}\left(\frac{d}{\varepsilon^2}\right)$
  - Recall: Empirical estimator is optimal: need $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ samples to get $\mathrm{KL} \leq \varepsilon^2$

# A glimpse of [BCGG24]

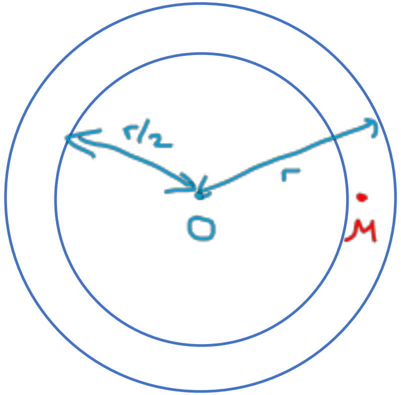



Each test uses $\tilde{\mathcal{O}}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$, as compared to $\tilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ for empirical estimator

Insight: "Testing can be cheaper than learning"

- High-level idea
  - Use sublinear tolerant testing + exponential search to find $r > 0$ s.t. $\frac{r}{2} \leq \|\boldsymbol{\mu}\|_2 \leq r$
  - Then, search within this radius to find a "good enough" $\widehat{\boldsymbol{\mu}}$
  - For technical reasons, we need to estimate $\|\boldsymbol{\mu}\|_1$ with some $\lambda$ instead
  - Then, using i.i.d. samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ from $\mathcal{P}$, solve LASSO in poly time:

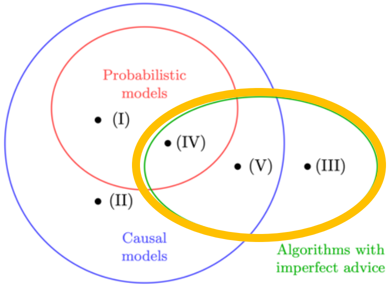
$$\widehat{\boldsymbol{\mu}} = \operatorname{argmin}_{\|\boldsymbol{\beta}\|_1 \leq r} \frac{1}{n} \sum_{i=1}^{n} \|\boldsymbol{y}_i - \boldsymbol{\beta}\|_2^2$$




  - When $\|\boldsymbol{\mu}\|_1$ is sufficiently small, our method provably uses $\tilde{o}\left(\frac{d}{\varepsilon^2}\right)$
  - Recall: Empirical estimator is optimal: need $\tilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ samples to get KL $\leq \varepsilon^2$

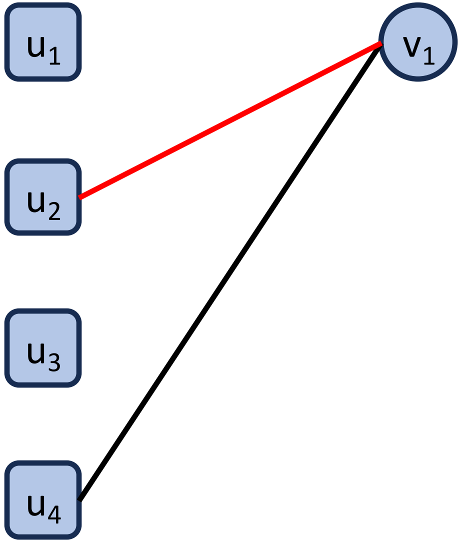
- Remarks
  - Small $\|\boldsymbol{\mu}\|_1$ here actually means small $\|\boldsymbol{\mu} - \tilde{\boldsymbol{\mu}}\|_1$ due to the pre-processing WLOG
  - We also need additional modifications tricks such as partitioning $\boldsymbol{\mu}$ into different coordinates to estimate $\|\boldsymbol{\mu}\|_1$, etc.
  - Similar idea work when the multivariate Gaussian has non-identity covariance matrix, but we use SDP instead of LASSO

# A glimpse of [CGLB24]

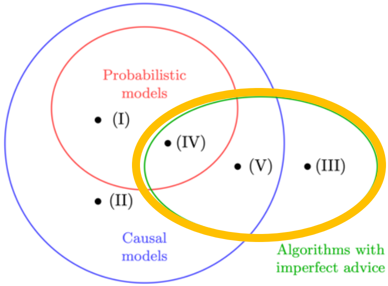



Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \ldots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \ldots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision

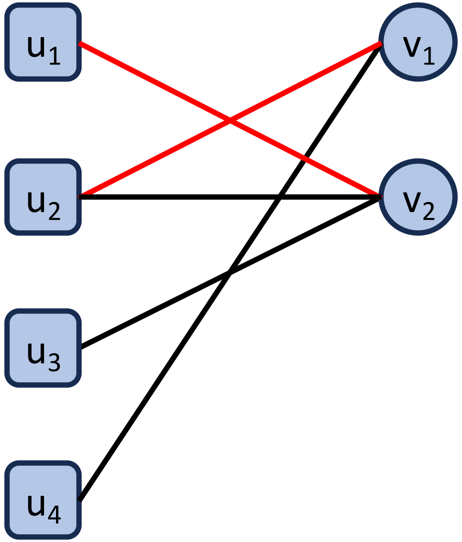


# A glimpse of [CGLB24]

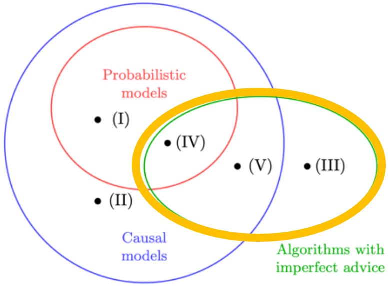



Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \ldots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \ldots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision

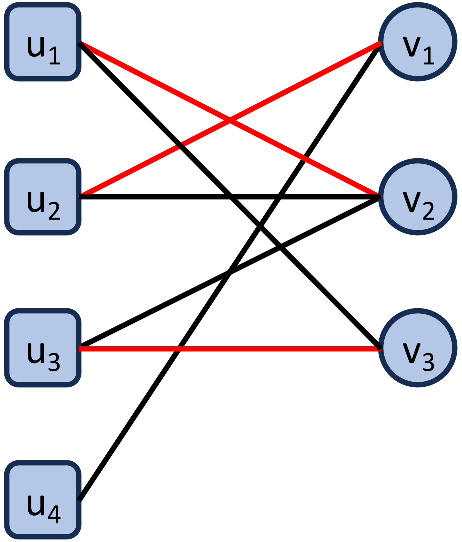


# A glimpse of [CGLB24]

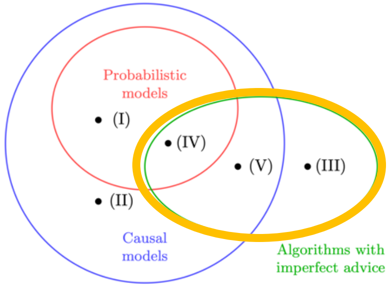



## Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \ldots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \ldots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision

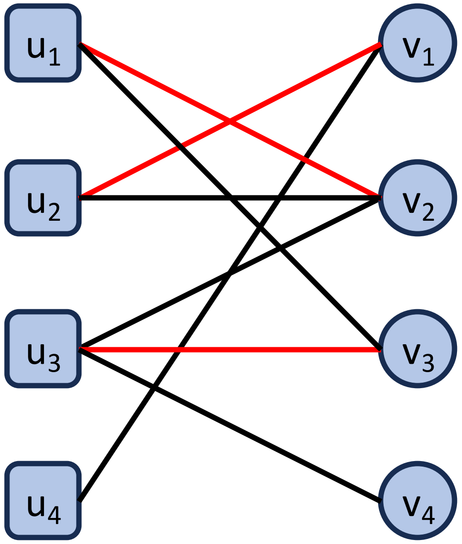


# A glimpse of [CGLB24]

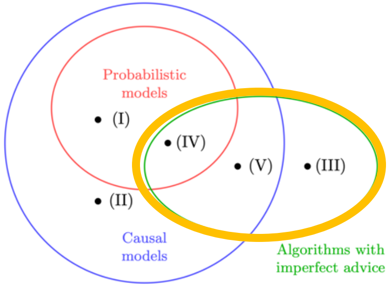



Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \ldots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \ldots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision

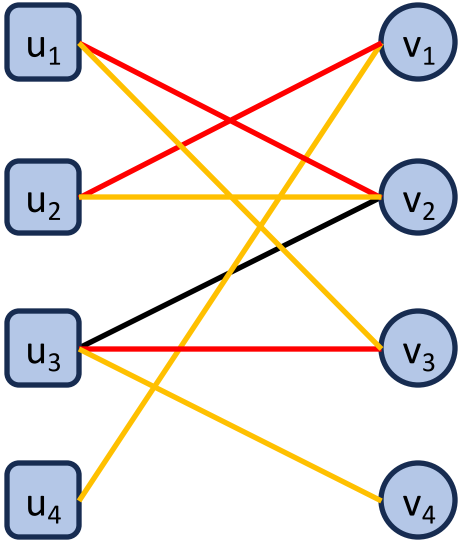


# A glimpse of [CGLB24]

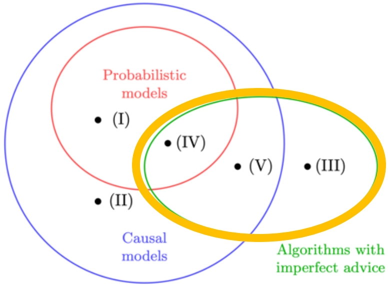



## Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \dots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \dots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision
  - Final offline graph $G^* = (U \cup V, E)$
    - $E = N(v_1) \cup \dots \cup N(v_n)$
    - Maximum matching $M^* \subseteq E$ of size $|M^*| = n^* \leq n$

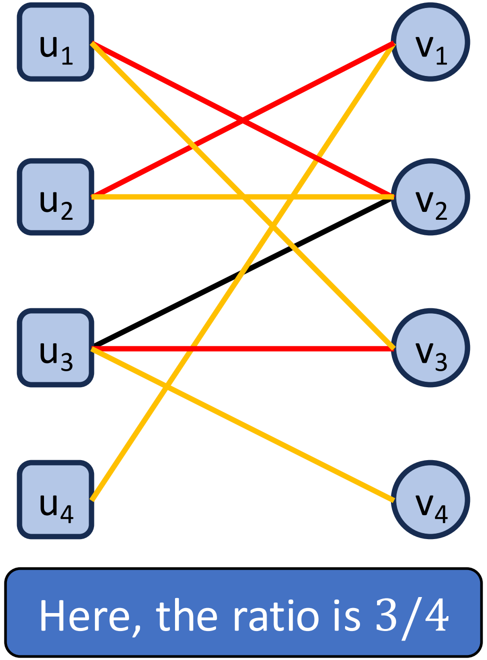


# A glimpse of [CGLB24]

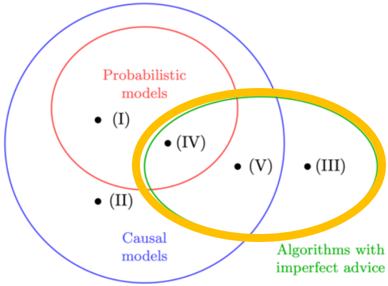



## Insight: "Testing can be cheaper than learning"

- Online bipartite matching
  - Offline set $U = \{u_1, \ldots, u_n\}$ fixed and known
  - Online set $V = \{v_1, \ldots, v_n\}$ arrive one by one
  - When an online vertex $v_i$ arrives
    - $N(v_i)$ are revealed and we make irrevocable decision
  - Final offline graph $G^* = (U \cup V, E)$
    - $E = N(v_1) \cup \cdots \cup N(v_n)$
    - Maximum matching $M^* \subseteq E$ of size $|M^*| = n^* \leq n$
  - Goal: Produce $M$ maximizing competitive ratio $\frac{|M|}{|M^*|}$

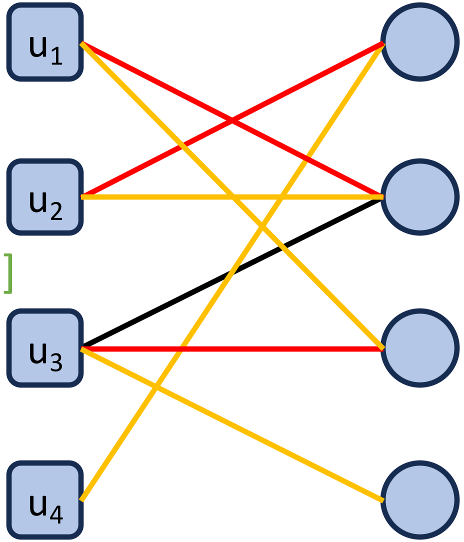


# A glimpse of [CGLB24]




## Insight: "Testing can be cheaper than learning"

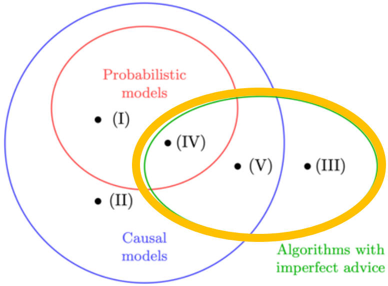
- Online bipartite matching with random arrival
  - Still worst-case final graph $G^*$
  - Online vertex sequence is random permutation of V
  - Ranking achieves comp. ratio of 0.696 [MY11]
  - No algorithm cannot beat comp. ratio of 0.823 [MGS12]
- What we show
  - Advice = Prediction $\tilde{G}$ of $G^*$
  - When advice perfect ($\tilde{G} = G^*$), get comp. ratio 1
  - When advice bad, we get $\approx \beta$ $(0.696 \leq \beta \leq 0.823)$





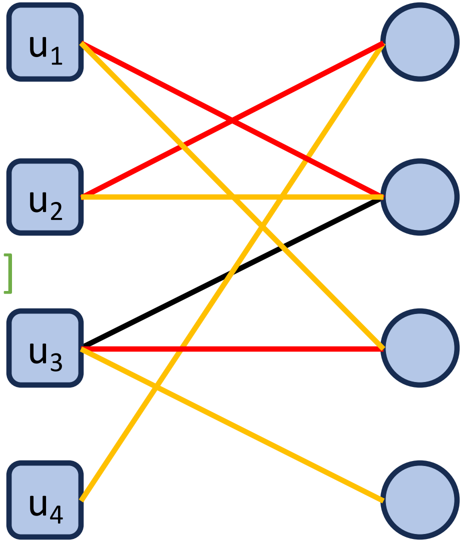
[MY11] Mohammad Mahdian and Qiqi Yan. *Online Bipartite Matching with Random Arrivals: An Approach Based on Strongly Factor-Revealing LPs*. Symposium on Theory of Computing (STOC), 2011
[MGS12] Vahideh H Manshadi, Shayan Oveis Gharan, and Amin Saberi. *Online stochastic matching: Online actions based on offline statistics*. Mathematics of Operations Research, 2012

# A glimpse of [CGLB24]




## Insight: "Testing can be cheaper than learning"

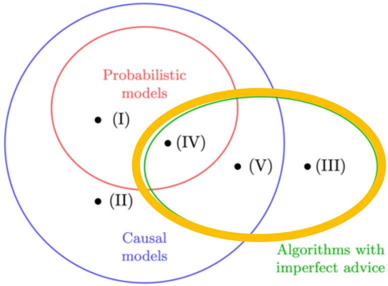
- Online bipartite matching with random arrival
  - Still worst-case final graph $G^*$
  - Online vertex sequence is random permutation of V
  - Ranking achieves comp. ratio of 0.696 [MY11]
  - No algorithm cannot beat comp. ratio of 0.823 [MGS12]
- What we show
  - Advice = Prediction $\tilde{G}$ of $G^*$
  - When advice perfect ($\tilde{G} = G^*$), get comp. ratio 1
  - When advice bad, we get $\approx \beta$ $(0.696 \leq \beta \leq 0.823)$




Say, Baseline achieves this

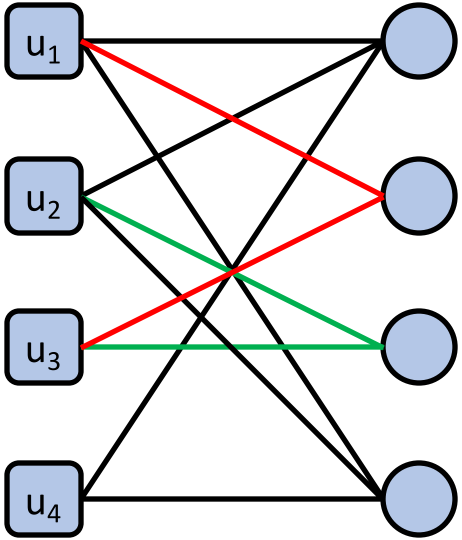

[MY11] Mohammad Mahdian and Qiqi Yan. *Online Bipartite Matching with Random Arrivals: An Approach Based on Strongly Factor-Revealing LPs*. Symposium on Theory of Computing (STOC), 2011
[MGS12] Vahideh H Manshadi, Shayan Oveis Gharan, and Amin Saberi. *Online stochastic matching: Online actions based on offline statistics*. Mathematics of Operations Research, 2012

# A glimpse of [CGLB24]




## Insight: "Testing can be cheaper than learning"

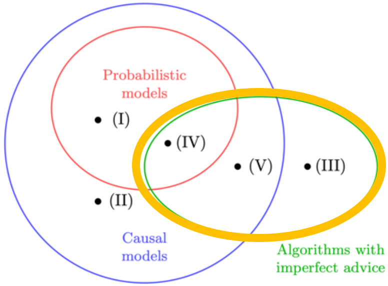
- Realized type counts as advice




$T^*$ = first three rows:

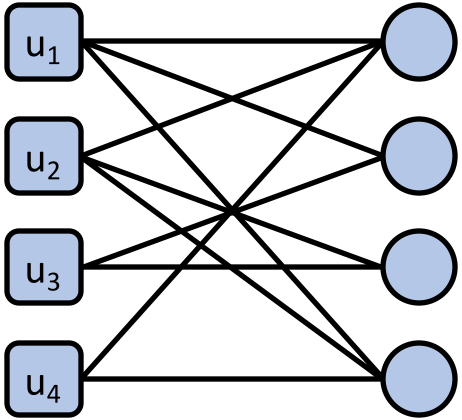
| Type | $c^*$ |
|---|---|
| $\{u_1, u_2, u_4\}$ | 2 |
| $\{u_1, u_3\}$ | 1 |
| $\{u_2, u_3\}$ | 1 |
| $2^U \setminus T^*$ | 0 |

# A glimpse of [CGLB24]



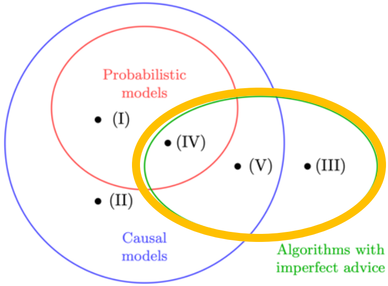

Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched



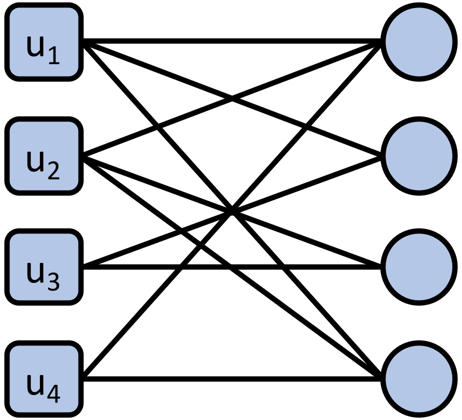

| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

# A glimpse of [CGLB24]

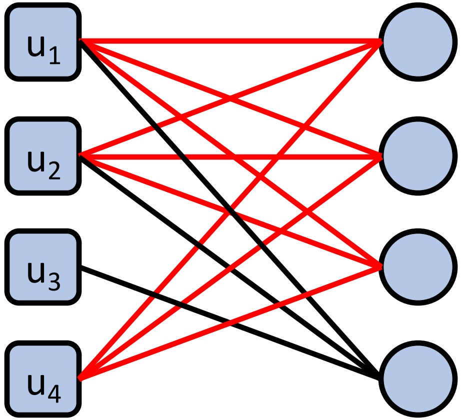


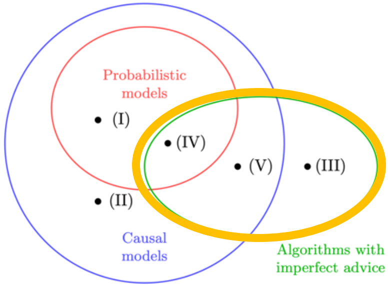

## Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

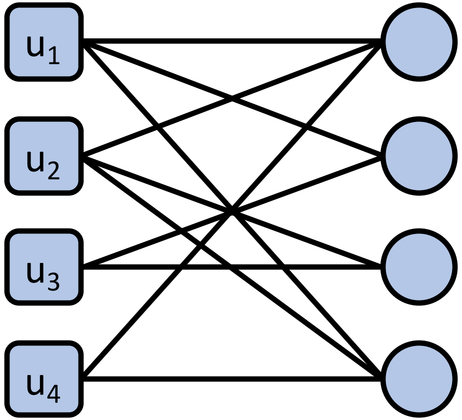





# A glimpse of [CGLB24]

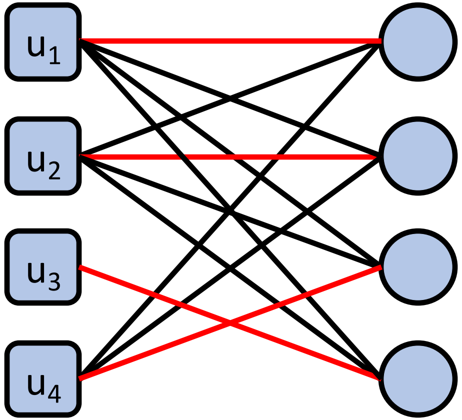
Insight: "Testing can be cheaper than learning"

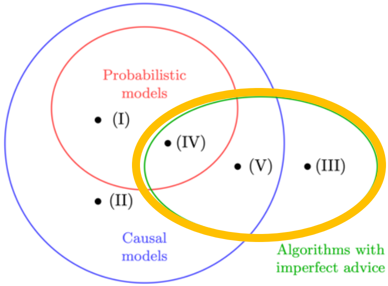
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

$\widehat{M}$

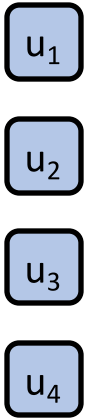




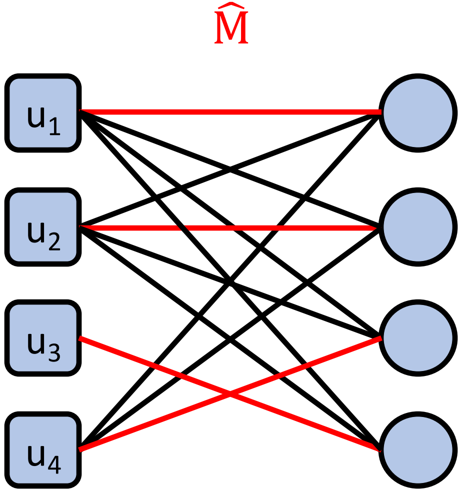

# A glimpse of [CGLB24]

Insight: "Testing can be cheaper than learning"

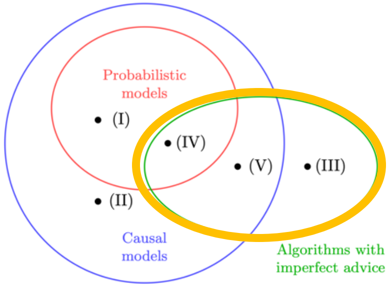
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

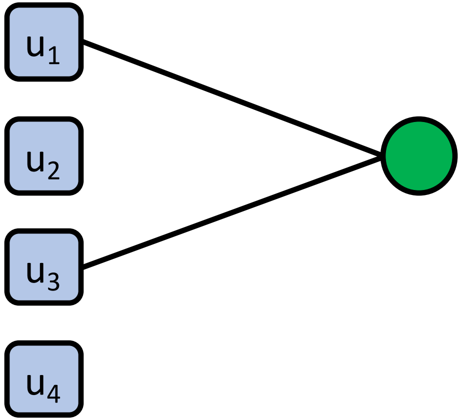


# A glimpse of [CGLB24]



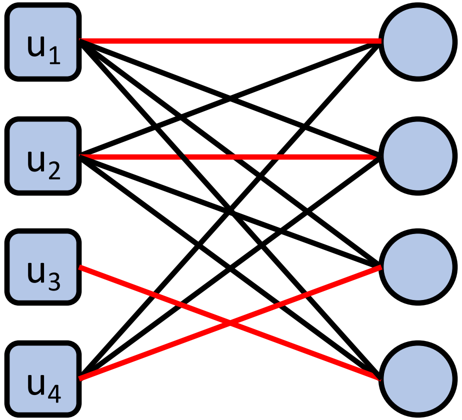

## Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched

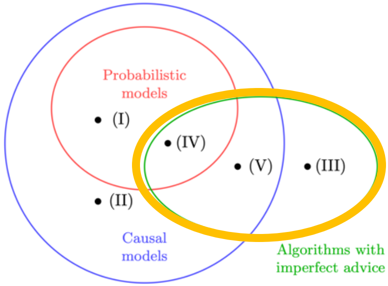



| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

$\widehat{M}$

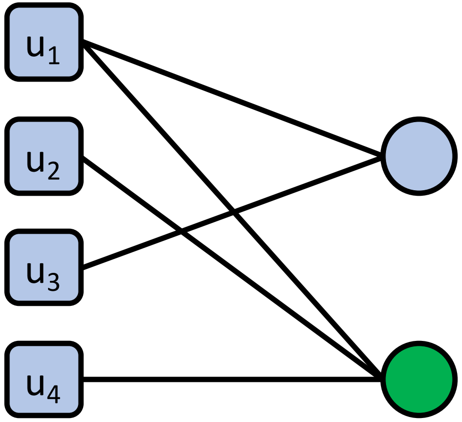


# A glimpse of [CGLB24]

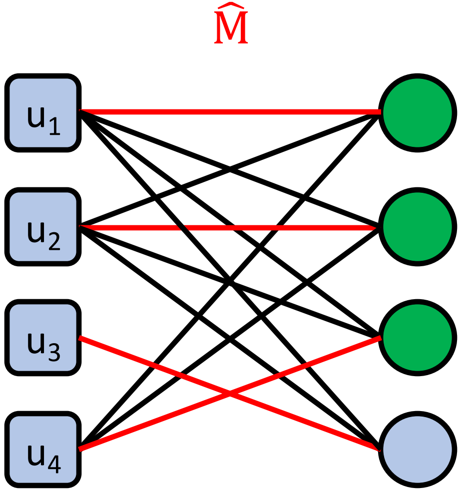



## Insight: "Testing can be cheaper than learning"

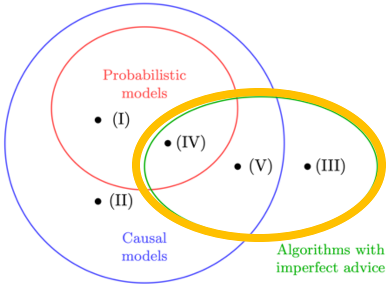
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

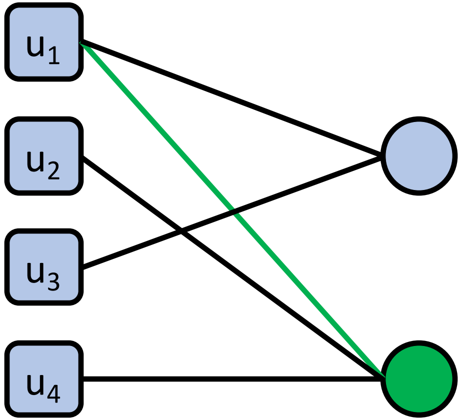


# A glimpse of [CGLB24]

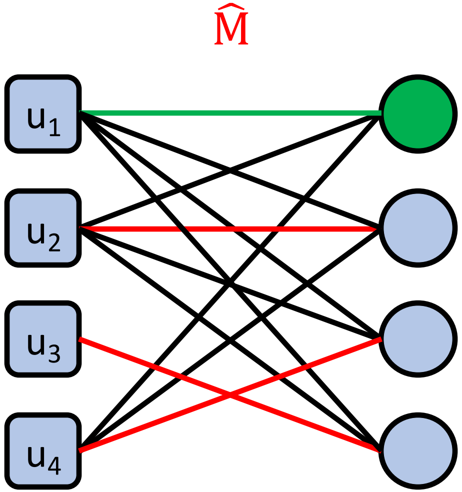



Insight: "Testing can be cheaper than learning"

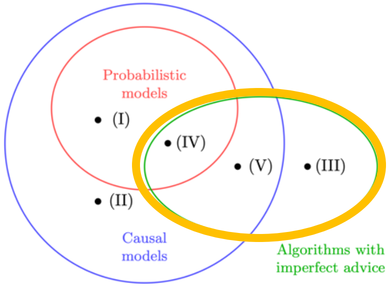
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | ~~3~~ 2 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

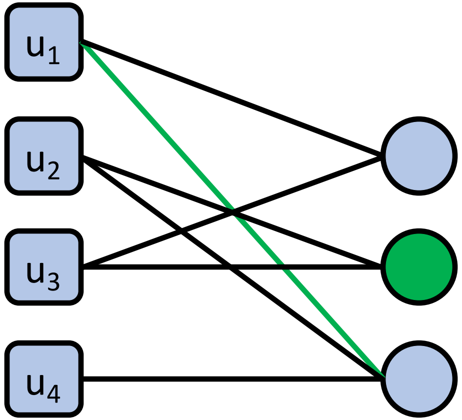


# A glimpse of [CGLB24]

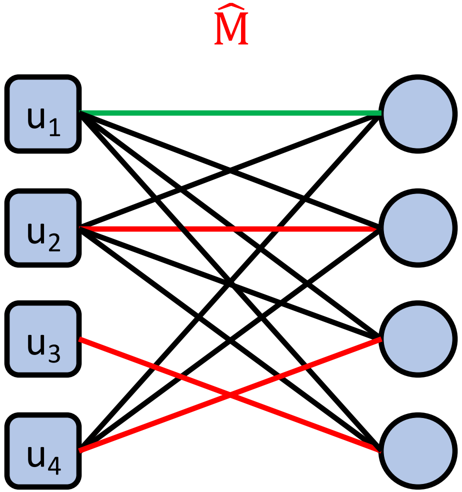



Insight: "Testing can be cheaper than learning"

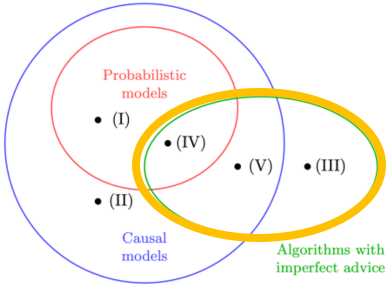
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | ~~3~~ 2 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

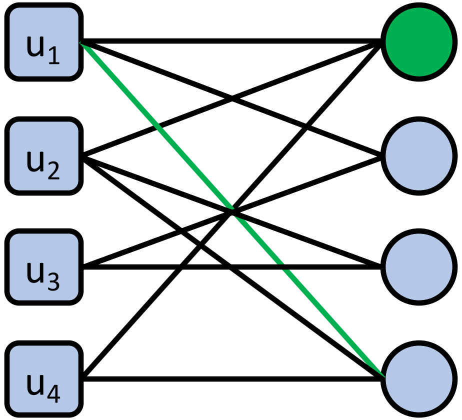


# A glimpse of [CGLB24]

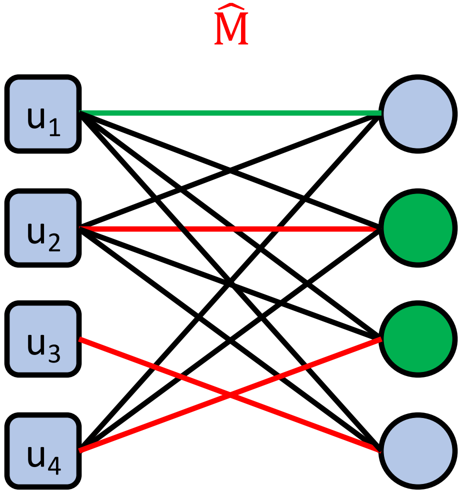



Insight: "Testing can be cheaper than learning"

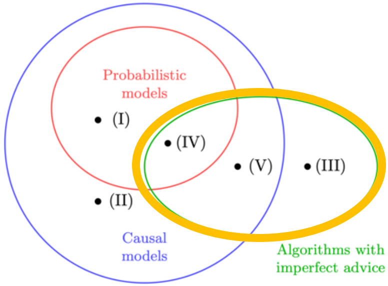
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | ~~3~~ 2 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

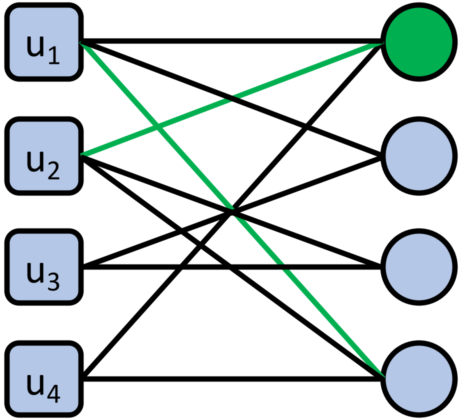


# A glimpse of [CGLB24]

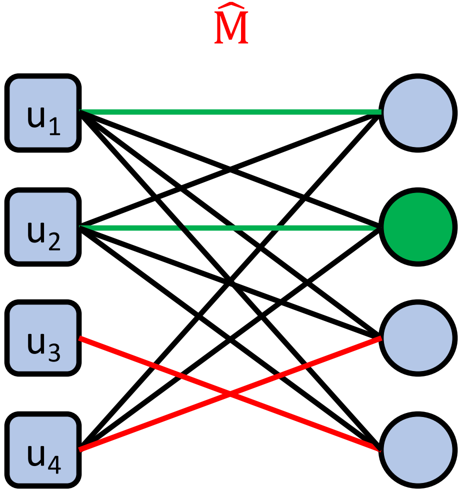



## Insight: "Testing can be cheaper than learning"

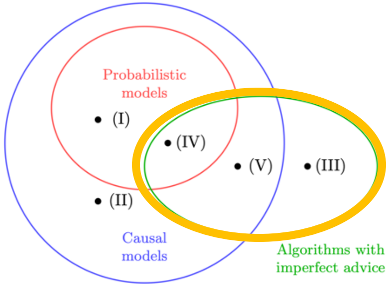
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | ~~3~~ ~~2~~ 1 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

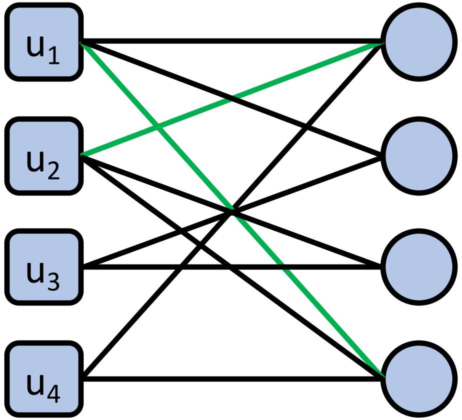


# A glimpse of [CGLB24]




Insight: "Testing can be cheaper than learning"

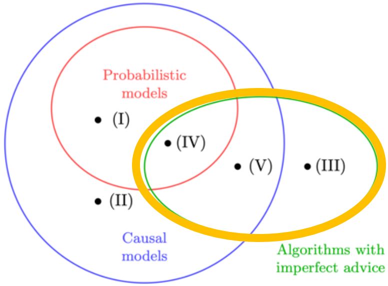
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

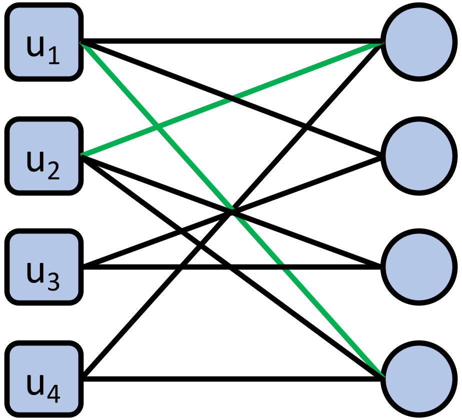
Produced matching size = 2

# A glimpse of [CGLB24]




Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched

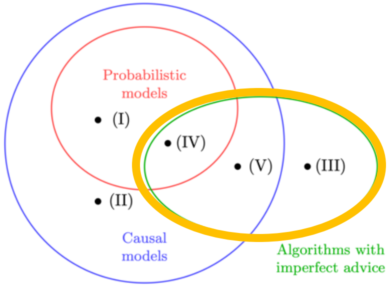



| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

Produced matching size

= 2

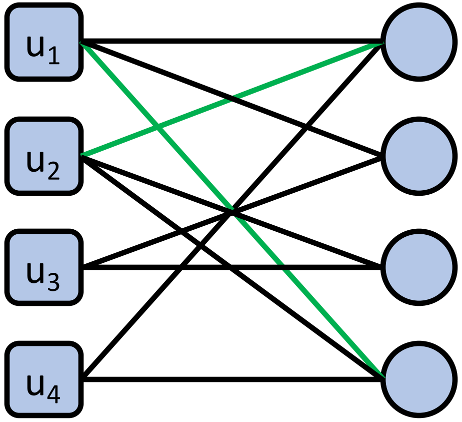
$$L_1(c^*, \hat{c})$$
$$= |3 - 2| + |0 - 1|$$
$$+ |0 - 1| + |1 - 0| + 0 \ldots$$
$$= 4$$




# A glimpse of [CGLB24]

Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched




| Type | $c^*$ | $\hat{c}$ |
|---|---|---|
| $\{u_1, u_2, u_4\}$ | 2 | 3 |
| $\{u_1, u_3\}$ | 1 | 0 |
| $\{u_2, u_3\}$ | 1 | 0 |
| $\{u_1, u_2, u_3\}$ | 0 | 1 |

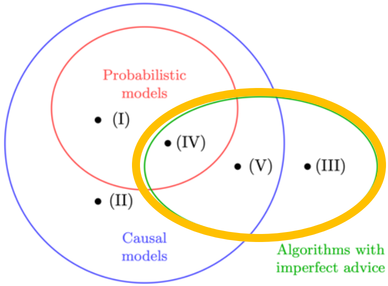
Produced matching size

$= 2 = \left|\widehat{M}\right| - \frac{L_1(c^*, \hat{c})}{2}$

Error is "double counted" in $L_1$

$$L_1(c^*, \hat{c}) = |3-2| + |0-1| + |0-1| + |1-0| + 0 \ldots = 4$$

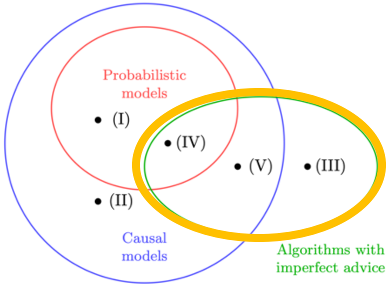
# A glimpse of [CGLB24]




Insight: "Testing can be cheaper than learning"

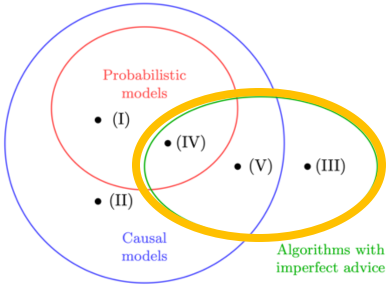
- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched
- Analysis: $0 \leq L_1(c^*, \hat{c}) \leq 2n$ measures how close $\hat{c}$ is to $c^*$
  - By blindly following advice, Mimic gets a matching of size $|\widehat{M}| - \frac{L_1(c^*, \hat{c})}{2}$
  - Mimic beats an advice-free Baseline whenever $|\widehat{M}| - \frac{L_1(c^*, \hat{c})}{2} > \beta \cdot n$

# A glimpse of [CGLB24]




Insight: "Testing can be cheaper than learning"

- Mimic algorithm: Fix arbitrary maximum matching $\widehat{M}$ defined by $\hat{c}$ and try to follow it as much as possible. If unable, leave unmatched
- Analysis: $0 \leq L_1(c^*, \hat{c}) \leq 2n$ measures how close $\hat{c}$ is to $c^*$
  - By blindly following advice, Mimic gets a matching of size $|\widehat{M}| - \frac{L_1(c^*, \hat{c})}{2}$
  - Mimic beats an advice-free Baseline whenever $|\widehat{M}| - \frac{L_1(c^*, \hat{c})}{2} > \beta \cdot n$
- Idea: Use Mimic when $L_1(c^*, \hat{c})$ low; otherwise use Baseline
- Problem: We don't know $c^*$, so cannot evaluate $L_1(c^*, \hat{c})$



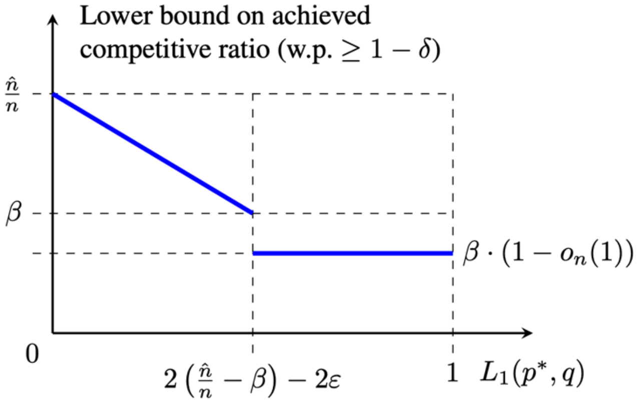

# A glimpse of [CGLB24]

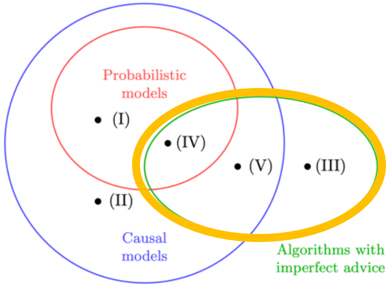
Insight: "Testing can be cheaper than learning"

- Use sublinear property testing to estimate $L_1(c^*, \hat{c})$ ← Random arrival ordering $\equiv$ i.i.d. samples
  - Define $p = \frac{c^*}{n}$ and $q = \frac{\hat{c}}{n}$ as distributions over the $2^U$ types
  - [VV11, JHW18]: Can estimate $L_1(p, q)$ "well" using $o(n)$ i.i.d. samples
    - Some adjustments needed to our problem setting, but it can be done

[VV11] Gregory Valiant and Paul Valiant. *The power of linear estimators*. Foundations of Computer Science (FOCS), 2011.
[JHW18] Jiantao Jiao, Yanjun Han, and Tsachy Weissman. *Minimax estimation of the $L_1$ distance*. IEEE Transactions on Information Theory, 2018.

# A glimpse of [CGLB24]




per than learning”

ting to estimate $L_1(c^*, \hat{c})$ ← Random arrival ordering $\equiv$ i.i.d. samples

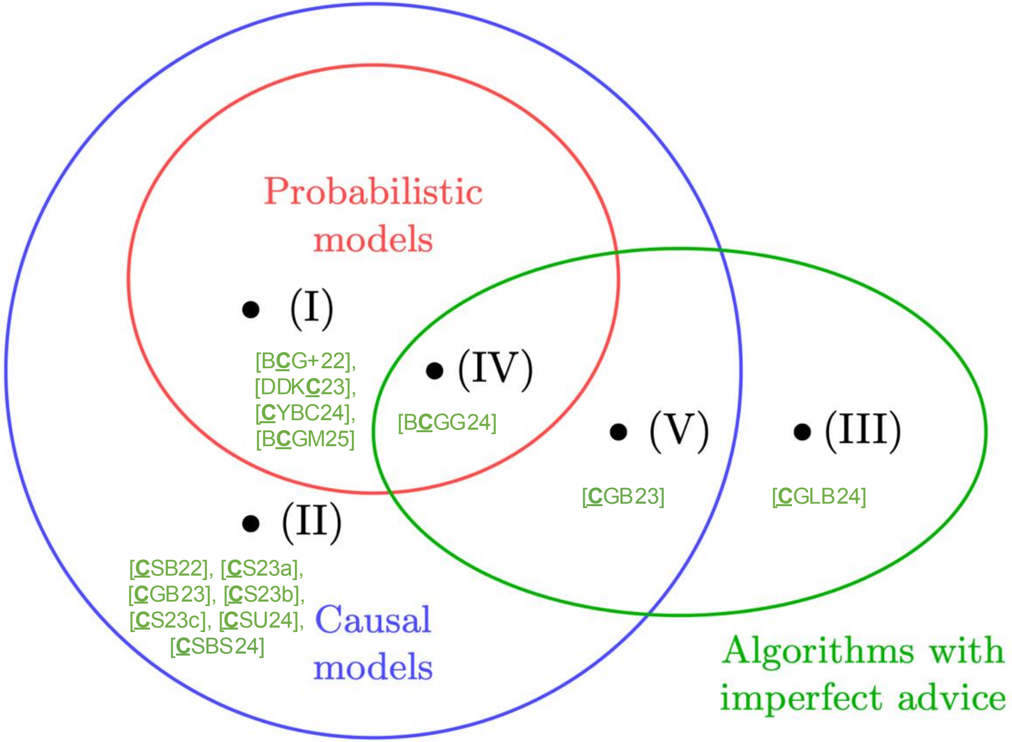
distributions over the $2^U$ types

te $L_1(p, q)$ “well” using $o(n)$ i.i.d. samples

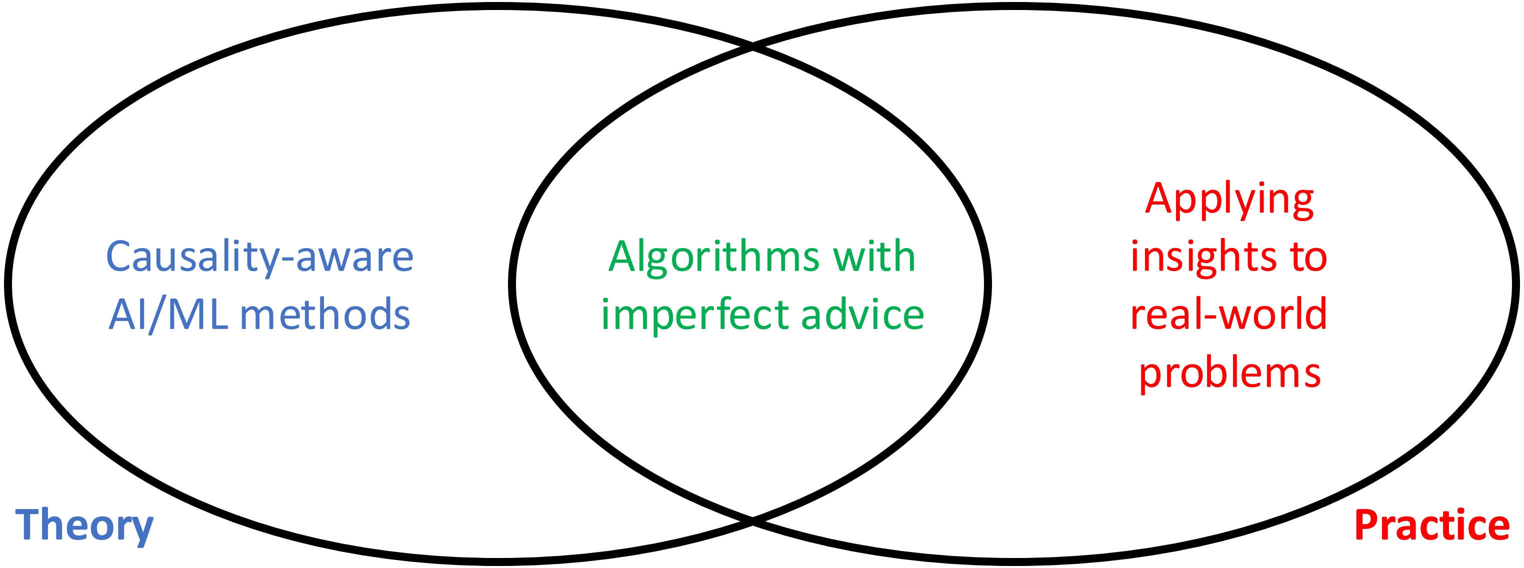
ɔ our problem setting, but it can be done

- TestAndMatch: Use Mimic or Baseline depending on $\hat{L}_1(c^*, \hat{c})$
  - Achieve comp. ratio at least $1 - \frac{L_1(c^*, \hat{c})}{2n} \geq \beta$, when $\hat{L}_1(c^*, \hat{c})$ “small”
  - Achieve comp. ratio at least $\beta \cdot (1 - o(1))$, when $\hat{L}_1(c^*, \hat{c})$ “large”
  - i.e., TestAndMatch is 1-consistent and $\beta \cdot (1 - o(1))$-robust

# Main themes explored in my PhD thesis



# Research vision:
# Principled algorithms with real-world impact



# Research vision:
# Principled algorithms with real-world impact

Causality-aware AI/ML methods

**Theory**

- Beyond simple statistical or association relationships
- Especially important for systems that act on the environment and have impact on real-world decisions


# Research vision:
# Principled algorithms with real-world impact

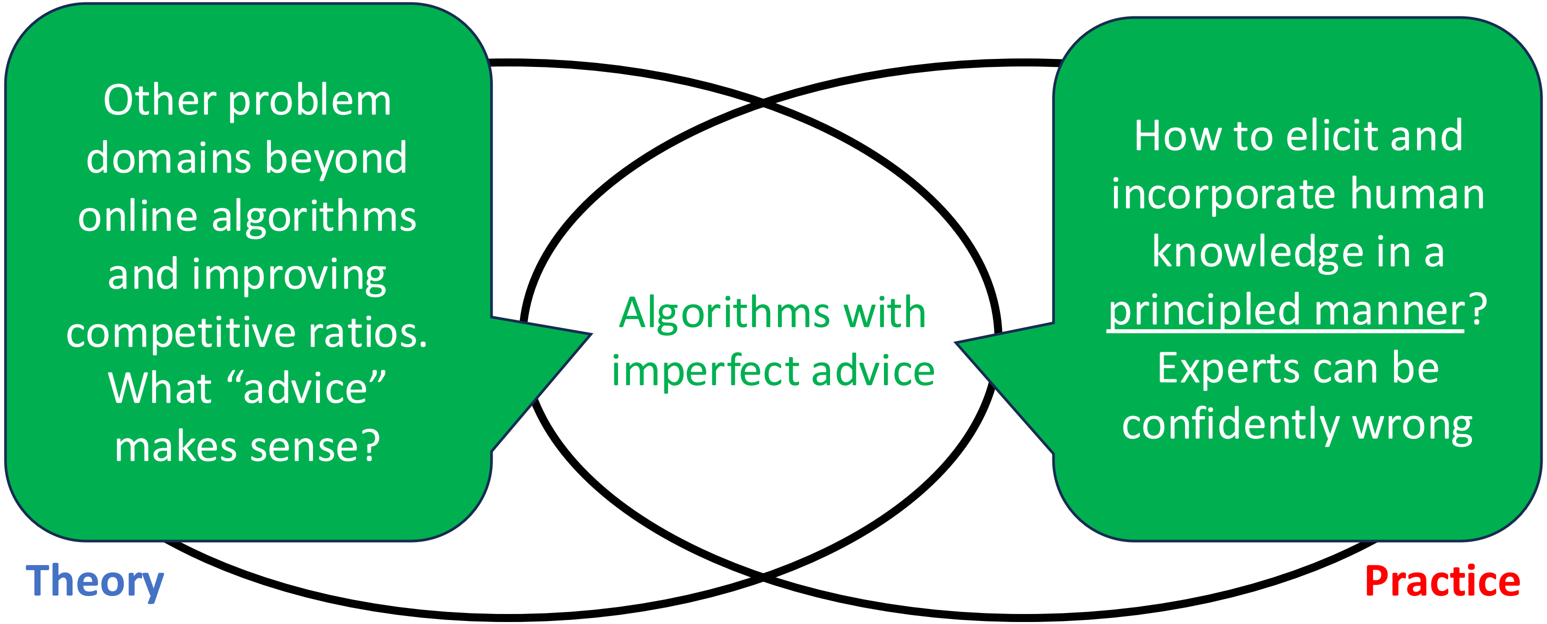
- Translate knowledge to benefit society
- Model and solve real-world problems in a principled manner

Applying insights to real-world problems

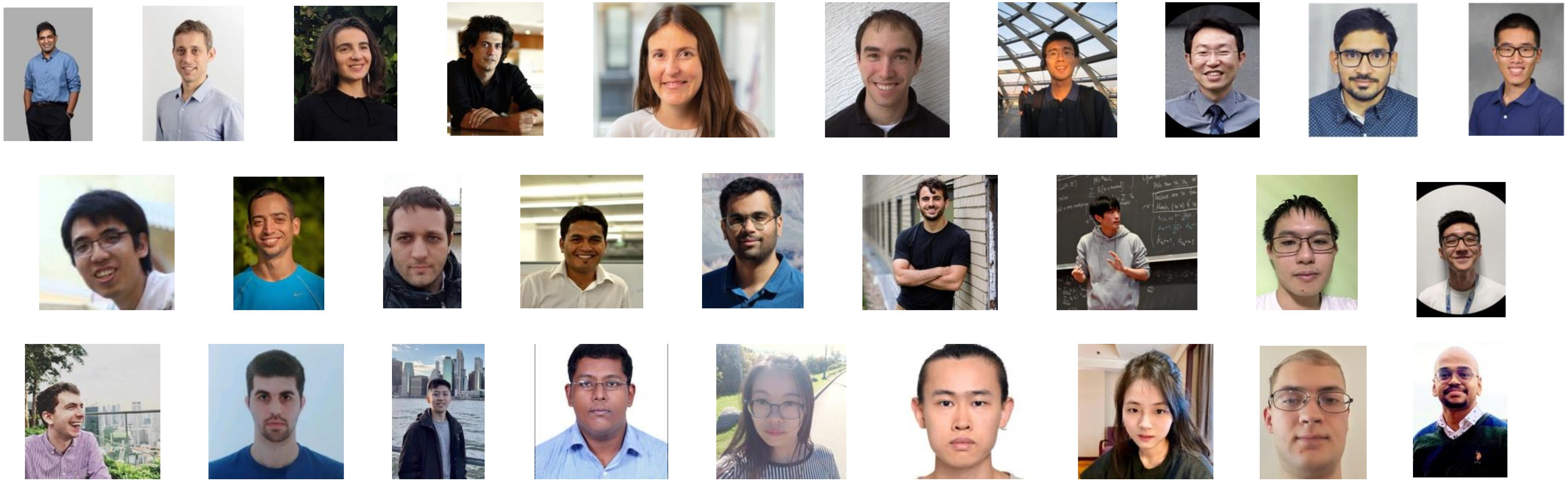
**Practice**

ChatGPT on the prompt "Generate me a picture of using AI/ML to benefit society"

# Research vision:
# Principled algorithms with real-world impact



# Thank you to all my amazing collaborators during my PhD journey!



**Thank you for your kind attention!**

(I have been lucky to work on many interesting projects with these folks since Aug 2021. I have learnt a lot from them! Some of the works are not included in this talk or are upcoming submissions)