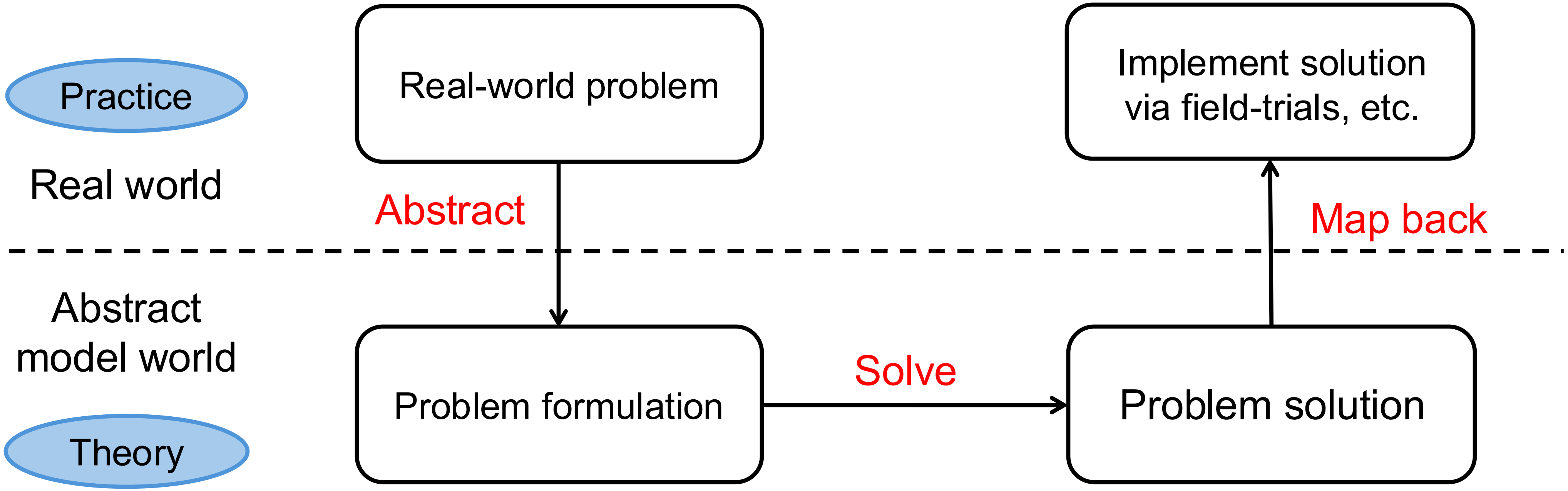
Test-and-Act: A recipe for learning-augmented algorithms inspired by sublinear thinking

UCSD CS Theory Lunch
Dec 5, 2025

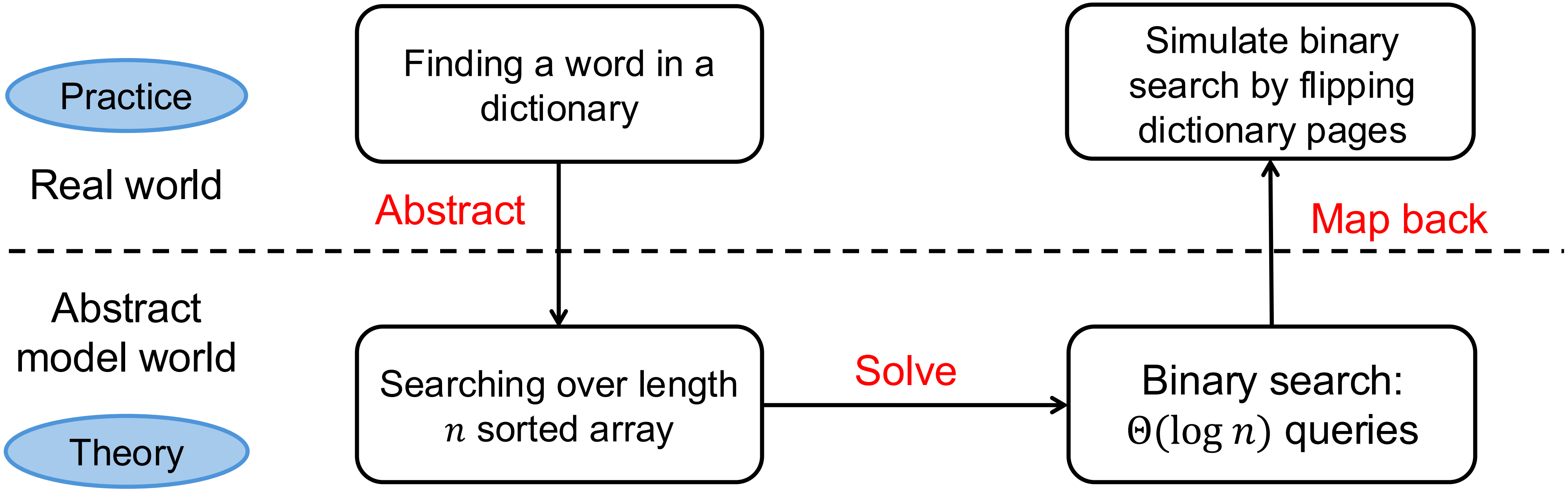
Davin Choo

Postdoctoral Fellow @ Harvard SEAS

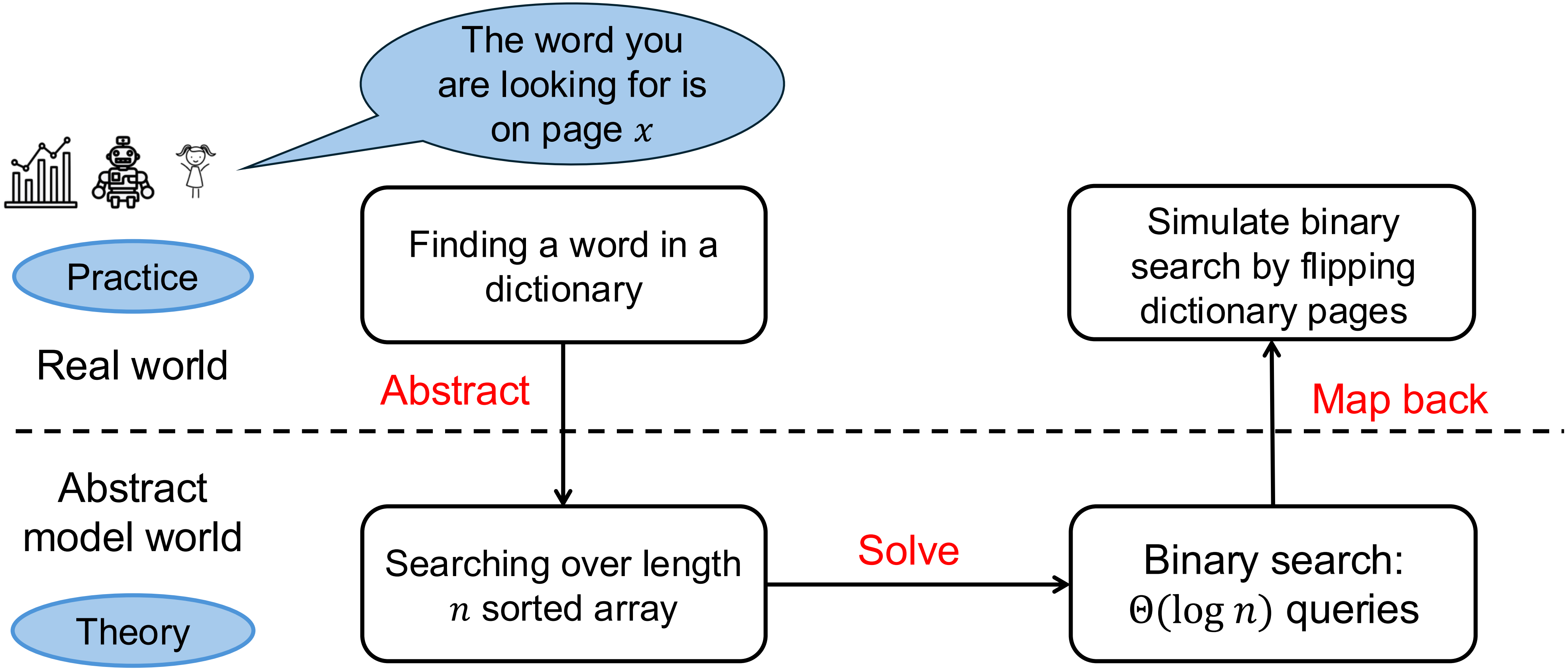
A general problem-solving paradigm



A general problem-solving paradigm


Practice
Real world
Abstract
model world
Theory
Finding a word in a dictionary
Abstract
Searching over length n sorted array
Solve
Binary search: $\Theta(\log n)$ queries
Map back
Simulate binary search by flipping dictionary pages

Learning-augmented algorithms are a way to harness imperfect instance-specific information



The word you are looking for is on page x

Practice

Real world

Finding a word in a dictionary

Abstract

Simulate binary search by flipping dictionary pages

Map back

Abstract model world

Theory

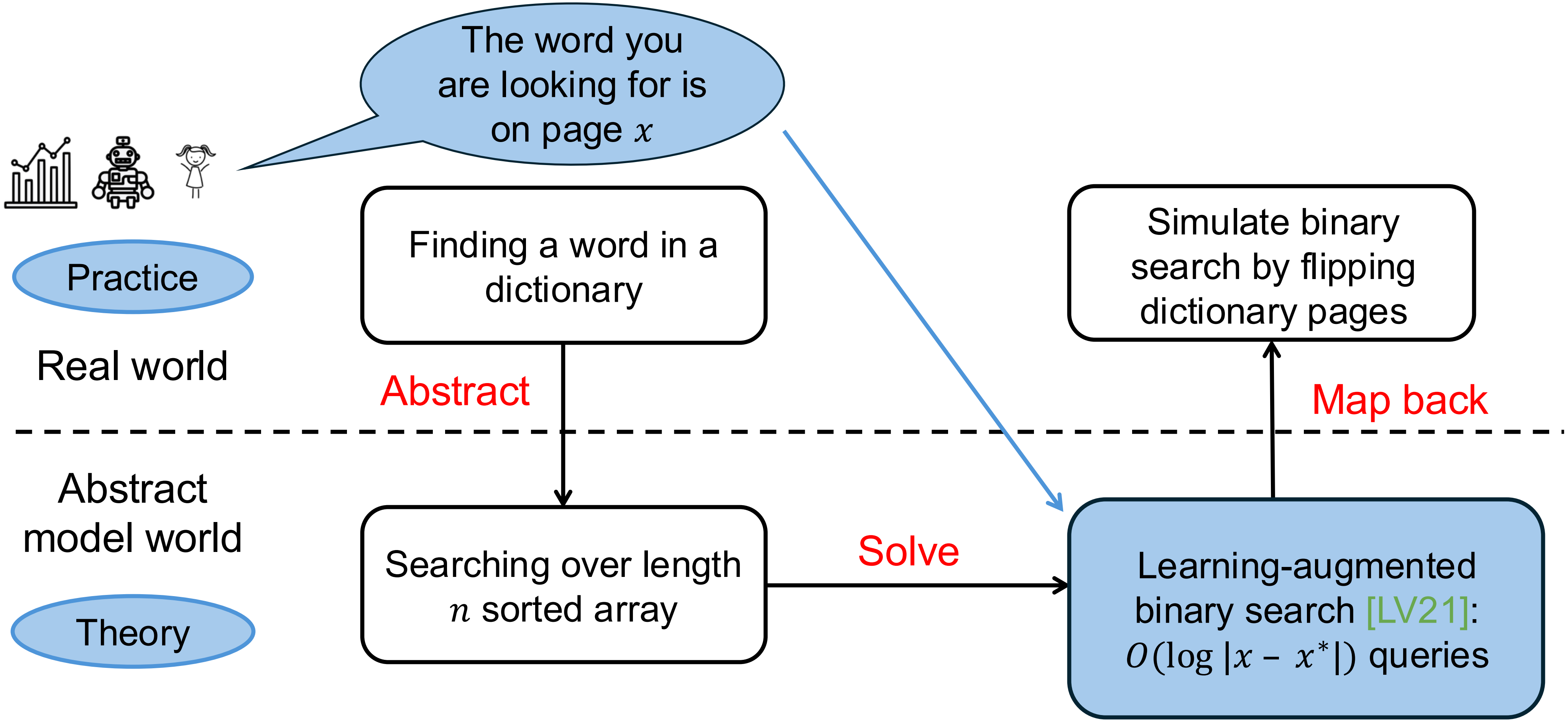
Searching over length n sorted array

Solve

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

https://thenounproject.com/icon/statistics-7090732/
https://thenounproject.com/icon/robot-7098785/
https://thenounproject.com/icon/girl-1257314/

Learning-augmented algorithms are a way to harness imperfect instance-specific information



The word you are looking for is on page x

Practice

Real world

Finding a word in a dictionary

Abstract

Simulate binary search by flipping dictionary pages

Map back

Abstract model world

Theory

Searching over length n sorted array

Solve

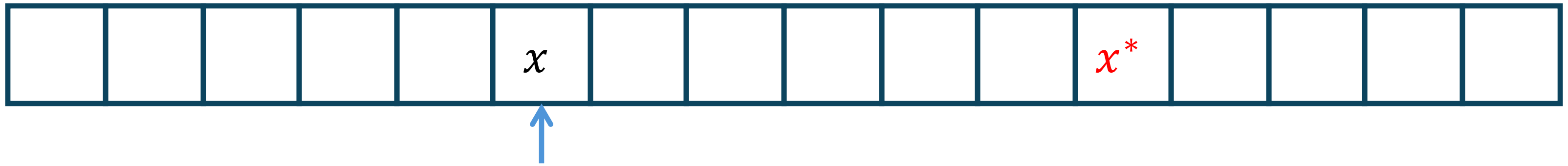
Learning-augmented binary search [LV21]: $O(\log|x - x^*|)$ queries

[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Learning-augmented binary search [LV21]

Algorithm and analysis

- Exponential search from x until we exceed x^*
- Binary search in between
- Each step takes $O(\log|x - x^*|)$ queries

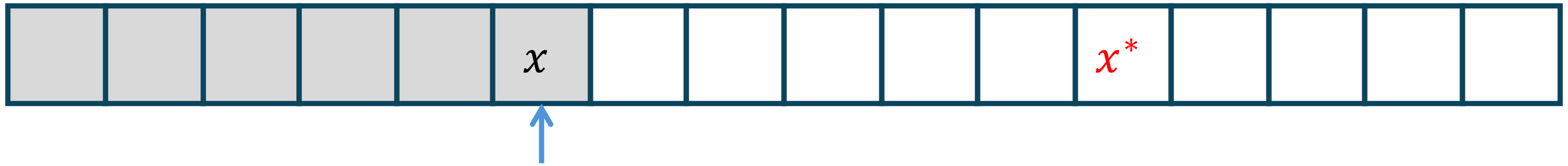


[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Learning-augmented binary search [LV21]

Algorithm and analysis

- Exponential search from x until we exceed x^*
- Binary search in between
- Each step takes $O(\log|x - x^*|)$ queries

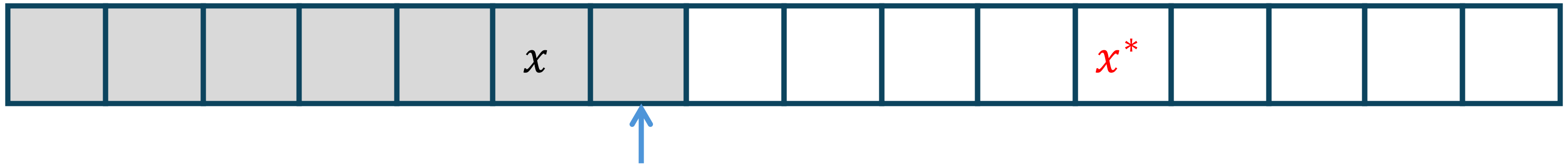


[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Learning-augmented binary search [LV21]

Algorithm and analysis

- Exponential search from x until we exceed x^*
- Binary search in between
- Each step takes $O(\log|x - x^*|)$ queries

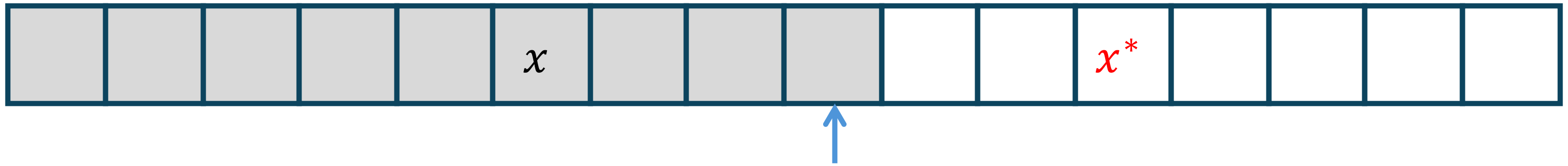


[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Learning-augmented binary search [LV21]

Algorithm and analysis

- Exponential search from x until we exceed x^*
- Binary search in between
- Each step takes $O(\log|x - x^*|)$ queries

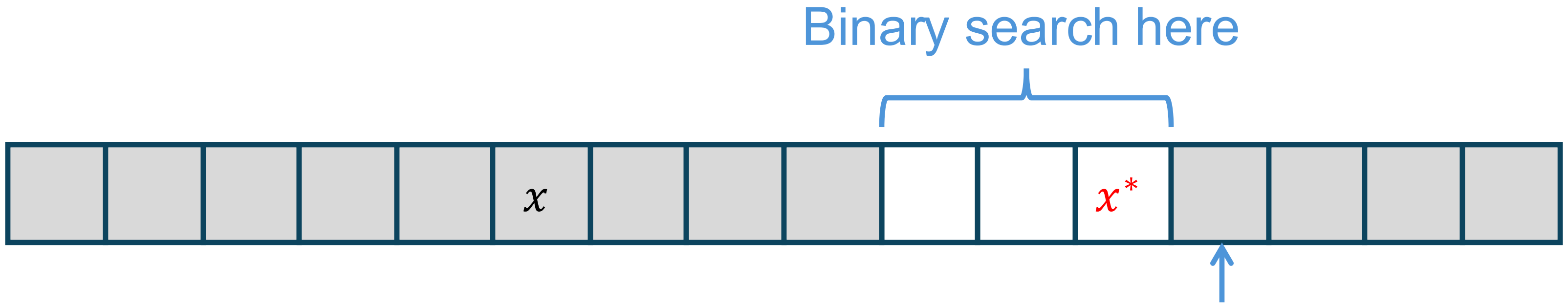


[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Learning-augmented binary search [LV21]

Algorithm and analysis

- Exponential search from x until we exceed x^*
- Binary search in between
- Each step takes $O(\log|x - x^*|)$ queries



[LV21] Thodoris Lykouris, Sergei Vassilvitskii. *Competitive Caching with Machine Learned Advice*. Journal of the ACM (JACM), 2021

Evaluating learning-augmented algorithms

Desiderata

- Consistency: What is the performance when advice is “perfect”?
- Robustness: What is the performance when advice is “garbage”?
  - Ideally, similar performance as best advice-free baseline
- Smoothness: Performance interpolates between extremes of “perfect” and “garbage” advice
- All the other usual “nice-to-haves” in algorithm design of polynomial runtime, etc.

Evaluating learning-augmented algorithms

Desiderata

- Consistency: What is the performance when advice is "perfect"?
- Robustness: What is the performance when advice is "garbage"?
  - Ideally, similar performance as best advice-free baseline
- Smoothness: Performance interpolates between extremes of "perfect" and "garbage" advice
- All the other usual "nice-to-haves" in algorithm design of polynomial runtime, etc.

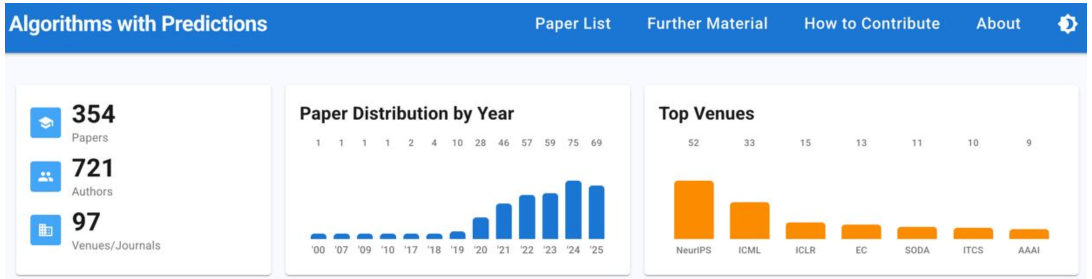
Searching in length n sorted array

- Performance metric of interest: Number of queries
- Best advice-free baseline: $\Theta(\log n)$ queries via binary search
- Learning-augmented binary search uses $O(\log |x - x^*|)$ queries
  - Consistency: When $x = x^*$, only 1 query required
  - Robustness: When x is "garbage", we have $|x - x^*| \leq n$ always, so at most $O(\log n)$ queries

Landscape of learning-augmented algorithms

Relatively new field with lots of potential and unsolved problems!

- https://algorithms-with-predictions.github.io/
- Snapshot on Dec 4, 2025
- See also Chris's [HSSY24]



354 Papers

721 Authors

97 Venues/Journals

What is "learning-augmented"? (From my possibly limited viewpoint)

- Started from online algorithms where we make irrevocable decisions without knowing the future
- Predictions about the future can help circumvent standard hardness results in these settings
- "Learning-augmented" because predictions often from machine learning models
- Personally, I prefer the broader term "algorithms with imperfect advice" as useful instance-specific information could also come human expertise or domain knowledge

[HSSY24] Monika Henzinger, Barna Saha, Martin P. Seybold, Christopher Ye. *On the Complexity of Algorithms with Predictions for Dynamic Graph Problems*. Innovations in Theoretical Computer Science Conference (ITCS), 2024

Test-and-Act framework

Insight: “Testing can be easier than learning”

- Developed as part of my PhD
- Idea: Design suitable testing subroutine to estimate advice quality, then react accordingly
- [**C**GB23] TestAndSubsetSearch: Reduce number of interventions for causal graph discovery
- [**C**GLB24] TestAndMatch: Improve competitive ratio of online bipartite matching
- [B**C**GG24] TestAndOptimize: Improve sample complexity of learning multivariate Gaussians
- [B**C**GG25] TestAndOptimize: Improve sample complexity of learning product distributions

Arnab
Bhattacharyya

Themistoklis
Gouleakis

Philips
George John

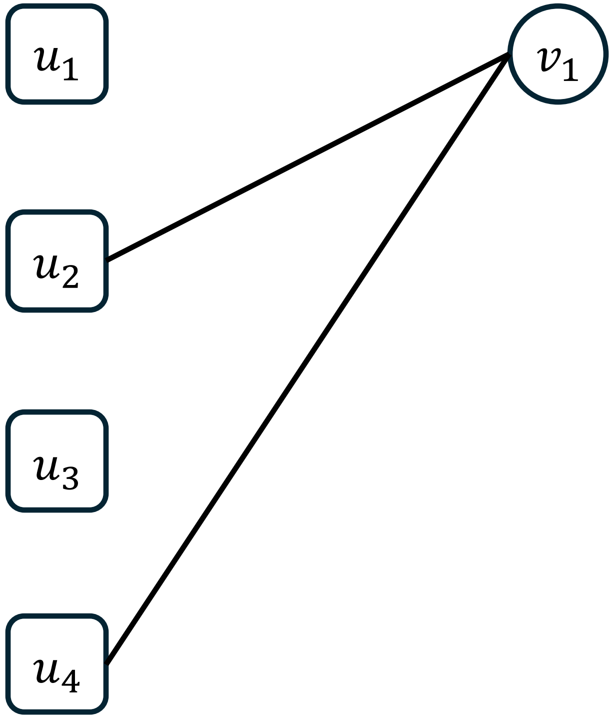
Chun Kai
Ling

[**C**GB23] Davin Choo, Themistoklis Gouleakis, Arnab Bhattacharyya. *Active causal structure learning with advice*. International Conference on Machine Learning (ICML), 2023
[**C**GLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[B**C**GG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[B**C**GG25] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Product Distribution Learning with Imperfect Advice*. Conference on Neural Information Processing Systems (NeurIPS) Spotlight, 2025

Online bipartite matching

Problem setting

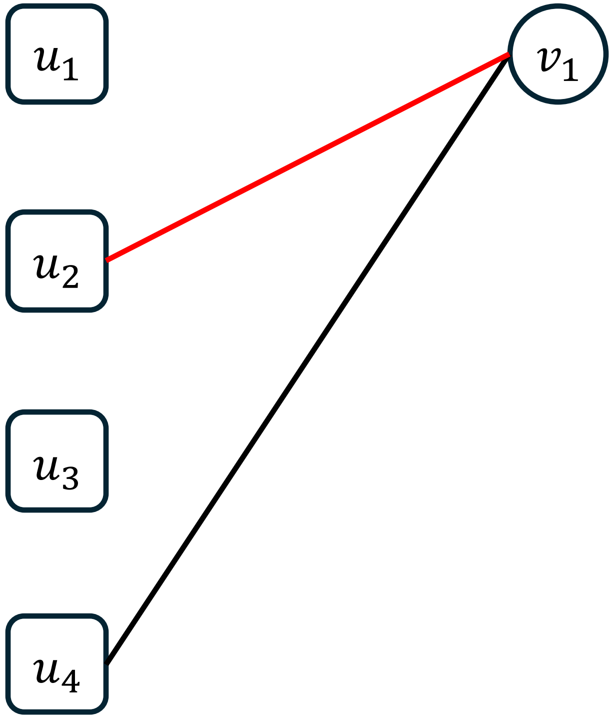
- Offline set $\boldsymbol{U} = \{u_1, \ldots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \ldots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make irrevocable decision



Online bipartite matching

Problem setting

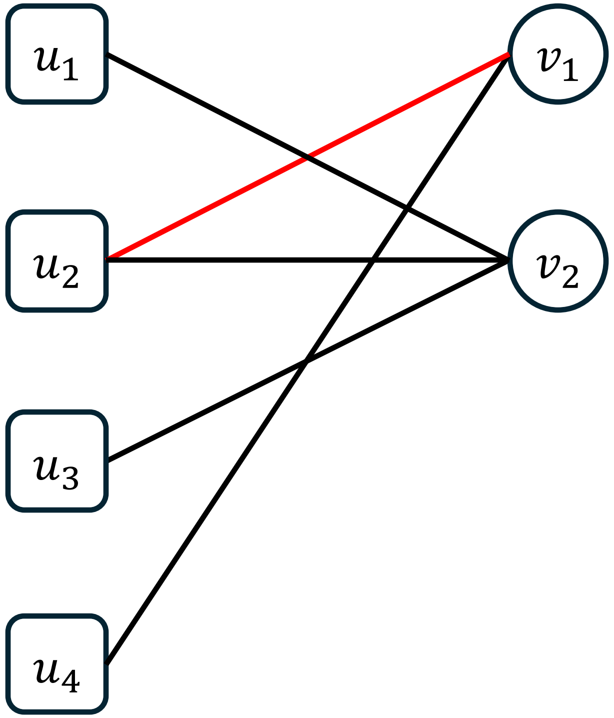
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make irrevocable decision



Online bipartite matching

Problem setting

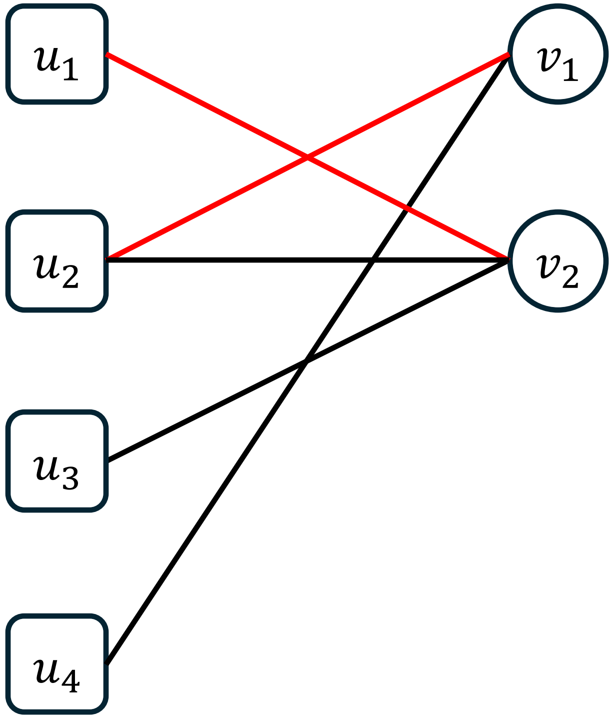
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
    - $N(v_i)$ are revealed and we make <u>irrevocable</u> decision



Online bipartite matching

Problem setting

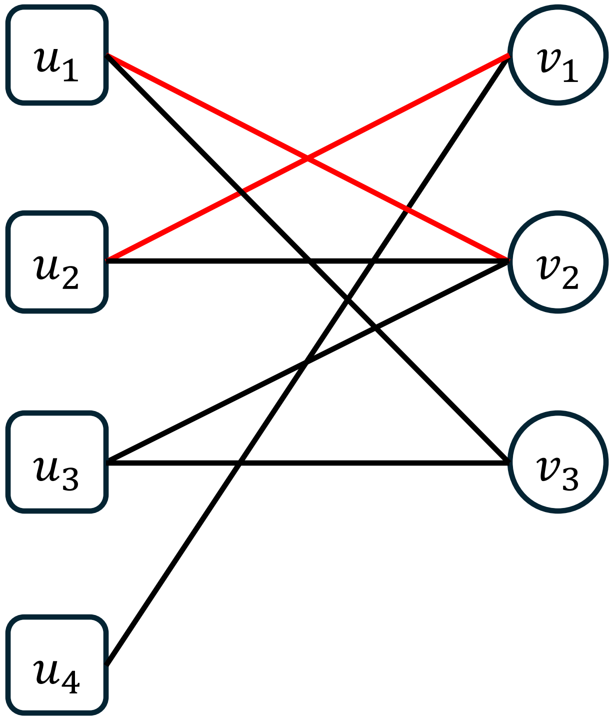
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make irrevocable decision



Online bipartite matching

Problem setting

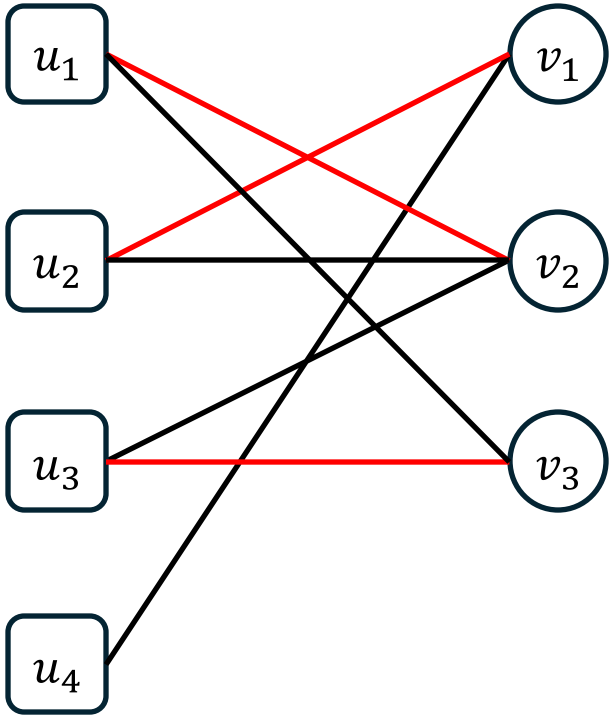
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make irrevocable decision



Online bipartite matching

Problem setting

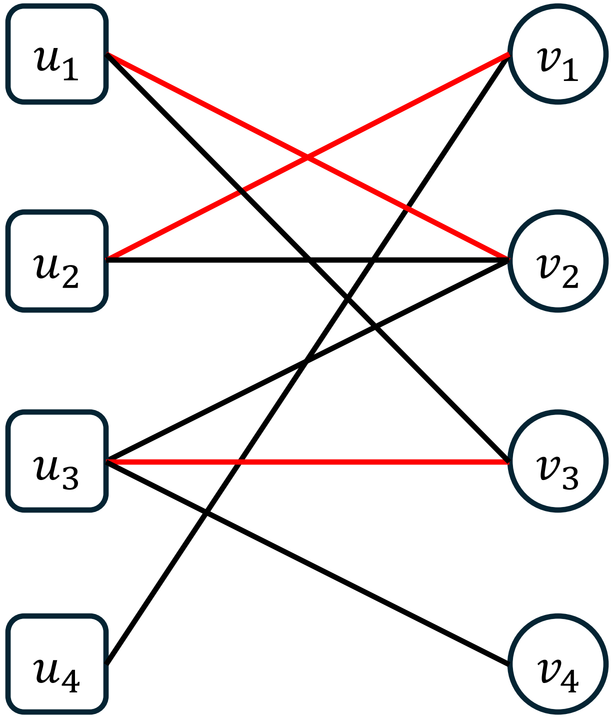
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
    - $N(v_i)$ are revealed and we make irrevocable decision



Online bipartite matching

Problem setting

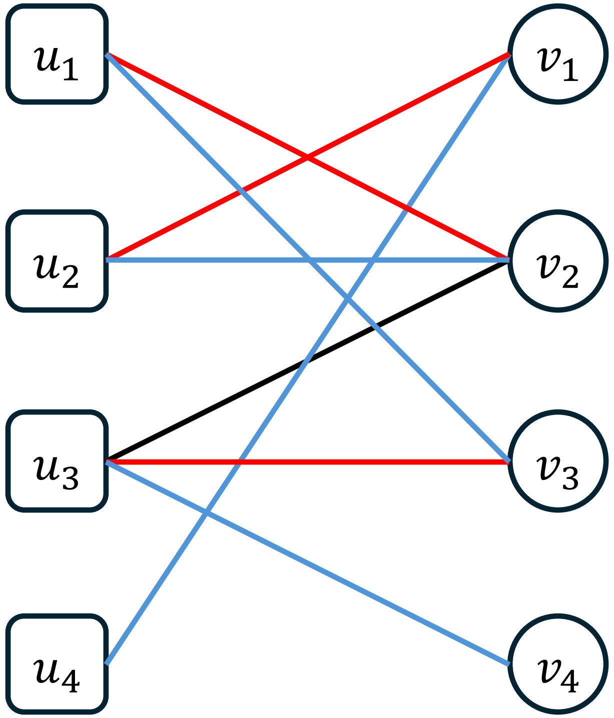
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
    - $N(v_i)$ are revealed and we make <u>irrevocable</u> decision



Online bipartite matching

Problem setting

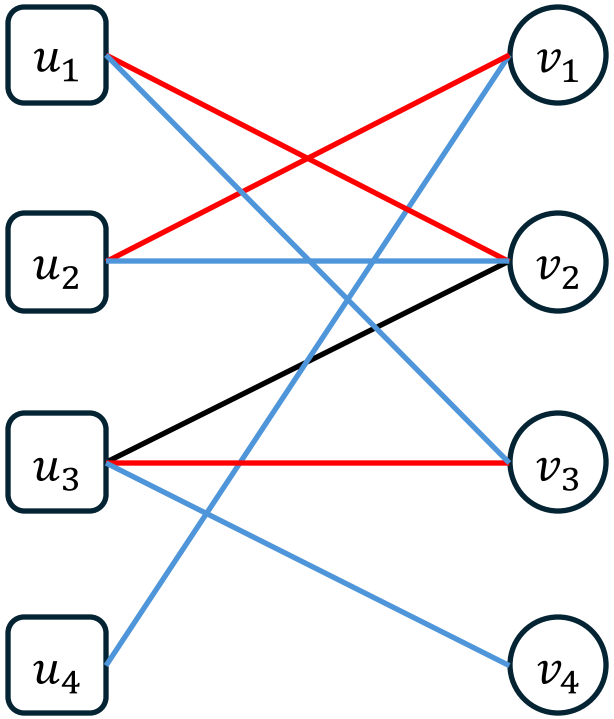
- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make <u>irrevocable</u> decision
- Final offline graph $\mathcal{G}^* = (\boldsymbol{U} \cup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



Online bipartite matching

Problem setting

- Offline set $\boldsymbol{U} = \{u_1, \dots, u_n\}$ fixed and known
- Online set $\boldsymbol{V} = \{v_1, \dots, v_n\}$ arrive one by one
- When an online vertex v_i arrives
  - $N(v_i)$ are revealed and we make irrevocable decision
- Final offline graph $\mathcal{G}^* = (\boldsymbol{U} \cup \boldsymbol{V}, \boldsymbol{E})$
  - $\boldsymbol{E} = N(v_1) \cup \dots \cup N(v_n)$
  - Maximum matching $\boldsymbol{M}^* \subseteq \boldsymbol{E}$ of size $|\boldsymbol{M}^*| \leq n$
- Goal
  - Produce matching $\boldsymbol{M}$ maximizing competitive ratio $\frac{|\boldsymbol{M}|}{|\boldsymbol{M}^*|}$
  - Here, the ratio is 3/4
  - For this talk, suppose $|\boldsymbol{M}^*| = n$



Online bipartite matching

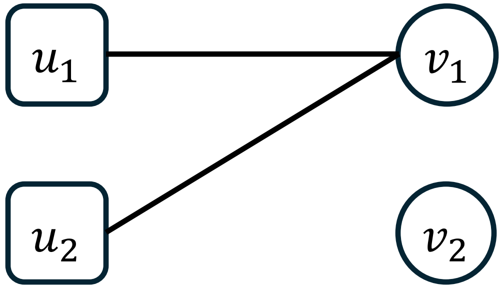
What is known?

- Any reasonable greedy algorithm has competitive ratio $\geq 1/2$
  - Size of maximal matching is at least half of size of maximum matching

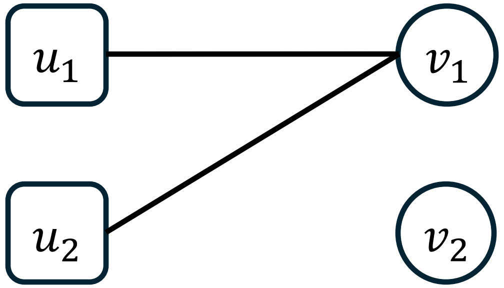
Online bipartite matching

What is known?

- Any reasonable greedy algorithm has competitive ratio $\geq 1/2$
  - Size of maximal matching is at least half of size of maximum matching
- Why is online bipartite matching hard?
  - Maximum bipartite matching is poly time computable…
  - But we don't know the future in the online setting!



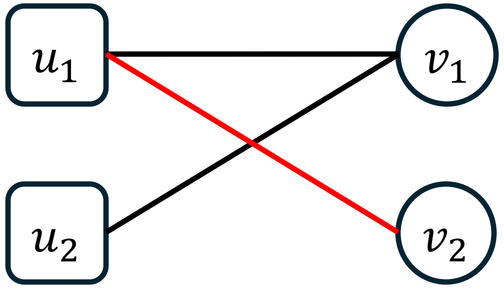
versus



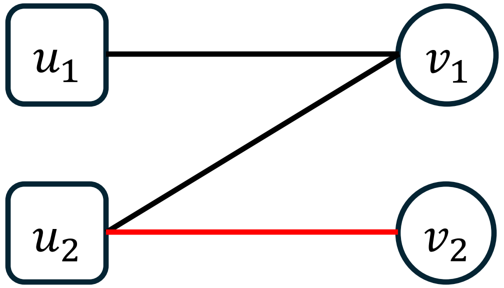
Online bipartite matching

What is known?

- Any reasonable greedy algorithm has competitive ratio $\geq 1/2$
    - Size of maxi<u>mal</u> matching is at least half of size of maxi<u>mum</u> matching
- Why is online bipartite matching hard?
    - Maximum bipartite matching is poly time computable…
    - But we don’t know the future in the online setting!



versus



Online bipartite matching

What is known?

- Expected competitive ratio: $\min\limits_{\mathcal{G}*} \min\limits_{\substack{\boldsymbol{V'}\text{s arrival}\\ \text{sequence}}} \frac{\text{(Expected) number of matches}}{n}$
- The Ranking algorithm [KVV90]
  - Pick a random permutation π over the offline vertices $\boldsymbol{U}$
  - When vertex v_i arrive with $N(v_i)$, match v_i to the smallest indexed (w.r.t π) unmatched neighbor

	(Expected) Competitive ratio
Deterministic algorithm	$\frac{1}{2}$
Deterministic hardness	$\frac{1}{2}$
Randomized algorithm	$1-\frac{1}{\mathrm{e}}$ [KVV90]
Randomized hardness	$1-\frac{1}{\mathrm{e}}+\mathrm{o}(1)$ [KVV90]

← Greedy

← Ranking

[KVV90] Richard M. Karp, Umesh V. Vazirani, Vijay V. Vazirani. *An optimal algorithm for on-line bipartite matching*. Symposium on Theory of Computing (STOC), 1990

Learning-augmented online bipartite matching

Can we design an algorithm that is 1-consistent and $\left(1-\frac{1}{e}\right)$-robust?

- Suppose we have instance-specific prediction / advice / side-information $\hat{\mathcal{G}}$ of $\mathcal{G}^*$
- Consistency: If $\hat{\mathcal{G}} = \mathcal{G}^*$, i.e., $\hat{\mathcal{G}}$ is “perfect”, then competitive ratio is 1
- Robustness: If $\hat{\mathcal{G}}$ is “garbage”, then competitive ratio is $1-\frac{1}{e}$

	(Expected) Competitive ratio
Deterministic algorithm	$\frac{1}{2}$
Deterministic hardness	$\frac{1}{2}$
Randomized algorithm	$1-\frac{1}{\mathrm{e}}$ [KVV90]
Randomized hardness	$1-\frac{1}{\mathrm{e}}+\mathrm{o}(1)$ [KVV90]

← Greedy

← Ranking

[KVV90] Richard M. Karp, Umesh V. Vazirani, Vijay V. Vazirani. *An optimal algorithm for on-line bipartite matching*. Symposium on Theory of Computing (STOC), 1990

Learning-augmented online bipartite matching

Prior attempts are not simultaneously 1-consistency and $\left(1-\frac{1}{e}\right)$-robust

- [AGKK20] Prediction on edge weights adjacent to V under an optimal offline matching
  - Random vertex arrivals and weighted edges. Require hyper-parameter to quantify confidence in advice, so their consistency/robustness tradeoffs are not directly comparable
- [ACI22] Prediction of vertex degrees $\hat{d}(u_1), \ldots, \hat{d}(u_n)$ of the offline vertices in $\boldsymbol{U}$
  - Adversarial arrival model. Optimal under the Chung-Lu-Vu random graph model [CLV03], but unable to attain 1-consistency in general
- [JM22] Advice is a proposed matching for the first batch of arrived vertices
  - Two-staged arrival model [FNS21], where best possible robustness is ¾
  - For any $\mathrm{R} \in [0, ¾]$, they can achieve consistency of $1 - \left(1 - \sqrt{1-\mathrm{R}}\right)^2$
- [LYR23] Augment any “expert algorithm” with a pre-trained RL model
  - For any $\rho \in [0,1]$, their method is ρ-competitive to the given “expert algorithm”

[AGKK20] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. *Secretary and online matching problems with machine learned advice*. Neural Information Processing Systems (NeurIPS), 2020
[ACI22] Anders Aamand, Justin Chen, and Piotr Indyk. *(Optimal) Online Bipartite Matching with Degree Information*. Neural Information Processing Systems (NeurIPS), 2022
[CLV03] Fan Chung, Linyuan Lu, and Van Vu. *Spectra of random graphs with given expected degrees*. Proceedings of the National Academy of Sciences (PNAS), 2003
[JM22] Billy Jin and Will Ma. *Online bipartite matching with advice: Tight robustness-consistency tradeoffs for the two-stage model*. Neural Information Processing Systems (NeurIPS), 2022
[FNS21] Yiding Feng, Rad Niazadeh, and Amin Saberi. *Two-stage stochastic matching with application to ride hailing*. Symposium on Discrete Algorithms (SODA), 2021
[LYR23] Pengfei Li, Jianyi Yang, and Shaolei Ren. Learning for edge-weighted online bipartite matching with robustness guarantees. International Conference on Machine Learning (ICML), 2023

Learning-augmented online bipartite matching

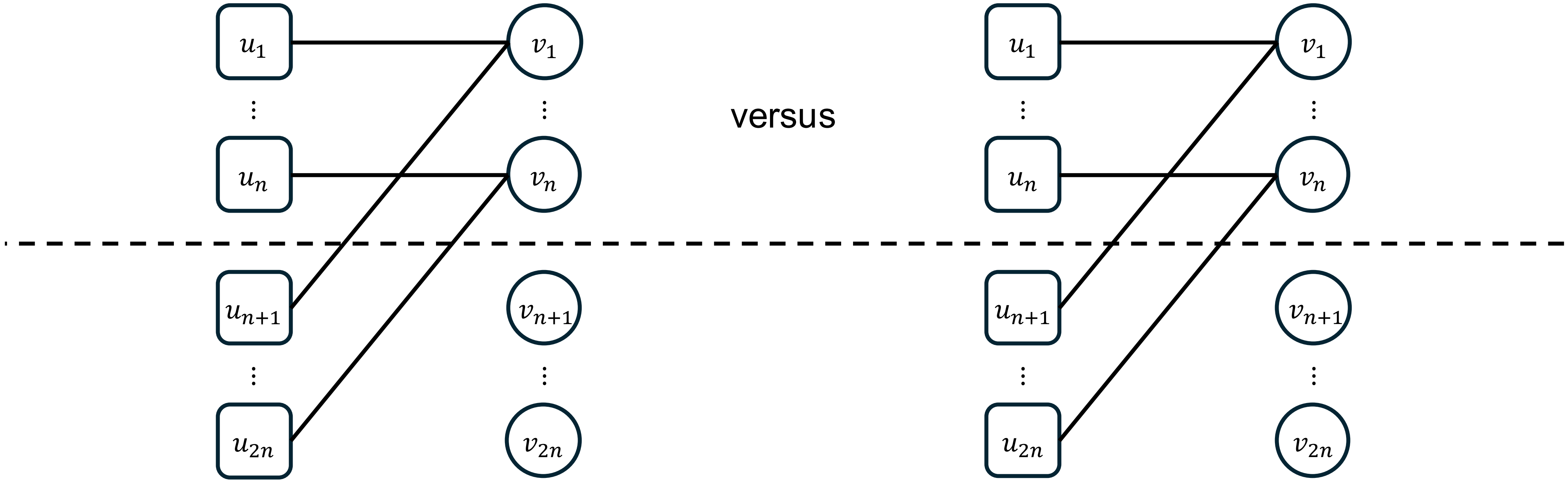
[CGLB24] Impossibility result: Under adversarial vertex arrivals, <u>no</u> algorithm can be both 1-consistent and $> \frac{1}{2}$ -robust, regardless of what advice is used.

- Extends to $(1-a)$-consistent and $\left(\frac{1}{2}+a\right)$-robust, for any $a \in [0, ½]$.
- Proof sketch (for $a = 0$ case):
    - Restrict $\mathcal{G}^*$ to be one of two possible graphs (next slide)
    - <u>Any</u> advice is equivalent to getting 1 bit of information
    - In first half of arrivals, no algorithm can distinguish between the two graphs
    - <u>Any</u> 1-consistent algorithm must behave as if the advice is perfect initially

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

Learning-augmented online bipartite matching

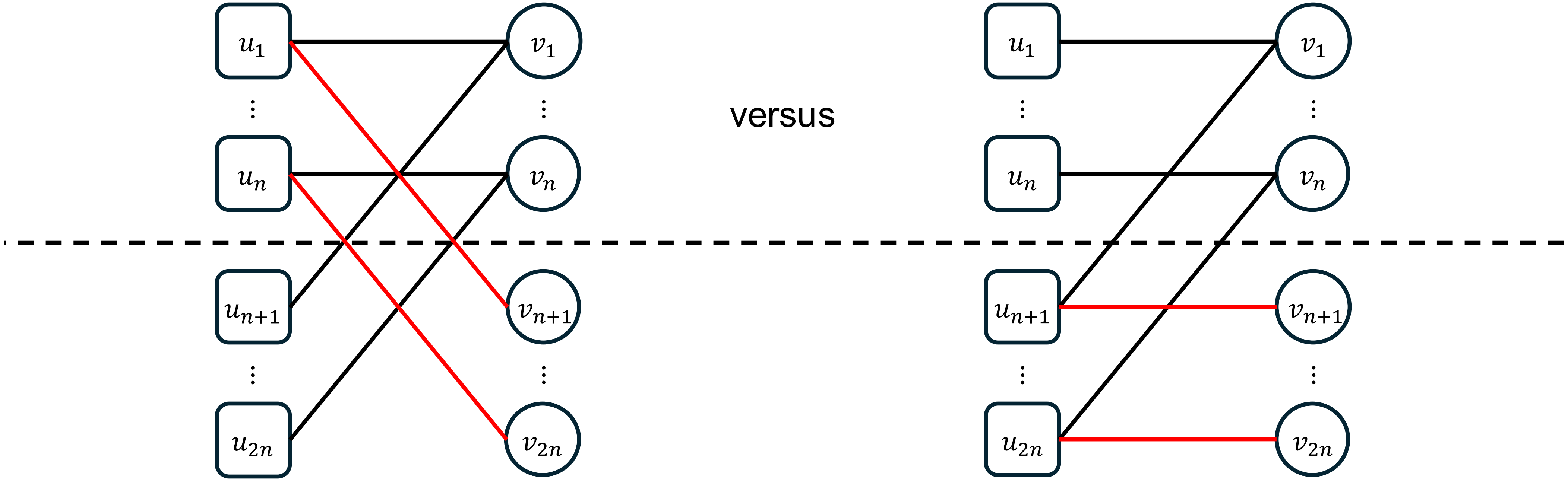
[CGLB24] Impossibility result: Under adversarial vertex arrivals, no algorithm can be both 1-consistent and $> \frac{1}{2}$ -robust, regardless of what advice is used.



[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

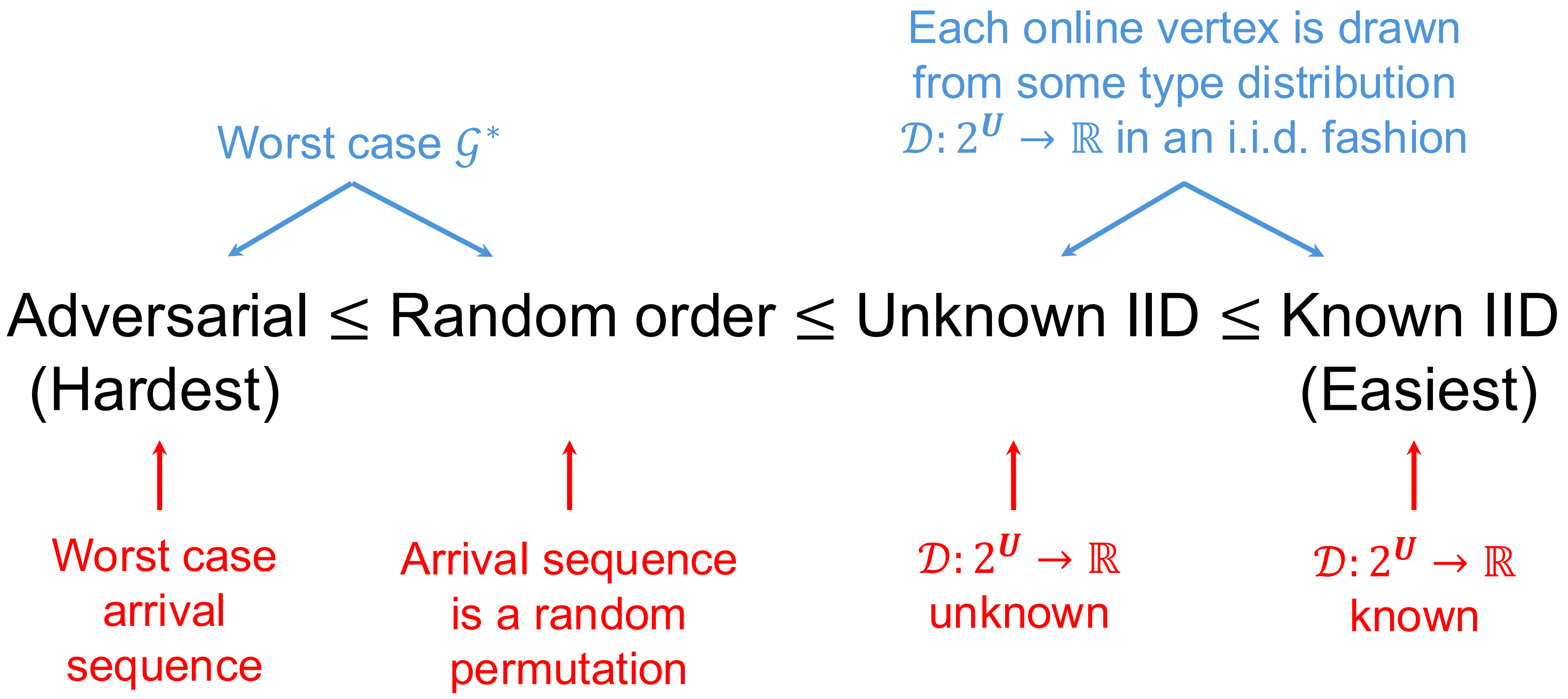
Learning-augmented online bipartite matching

[CGLB24] Impossibility result: Under adversarial vertex arrivals, no algorithm can be both 1-consistent and $> \frac{1}{2}$ -robust, regardless of what advice is used.



[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

Hierarchy of arrival models



Each online vertex is drawn from some type distribution $\mathcal{D}: 2^{\boldsymbol{U}} \rightarrow \mathbb{R}$ in an i.i.d. fashion

Worst case $\mathcal{G}^*$

Adversarial ≤ Random order ≤ Unknown IID ≤ Known IID

(Hardest)

(Easiest)

Worst case arrival sequence

Arrival sequence is a random permutation

$\mathcal{D}: 2^{\boldsymbol{U}} \rightarrow \mathbb{R}$ unknown

$\mathcal{D}: 2^{\boldsymbol{U}} \rightarrow \mathbb{R}$ known

[M13] Aranyak Mehta. *Online matching and ad allocation*. Foundations and Trends in Theoretical Computer Science, 2013

Online bipartite matching: random order arrival

What is known?

	(Expected) Competitive ratio		
	Adversarial arrival	Random order arrival	
Deterministic algorithm	$\frac{1}{2}$	$1-\frac{1}{e}$ [GM08]	← Greedy
Deterministic hardness	$\frac{1}{2}$	$\frac{3}{4}$	
Randomized algorithm	$1-\frac{1}{e}\approx 0.63$ [KVV90]	0.696 [MY11]	← Ranking
Randomized hardness	$1-\frac{1}{e}+o(1)$ [KVV90]	0.823 [MGS12]	

- [KMT11] showed that Ranking cannot beat 0.727 in general
- So, new ideas are needed if you believe the “right bound” is 0.823

[KMT11] 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
[GM08] Gagan Goel and Aranyak Mehta. *Online budgeted matching in random input models with applications to Adwords*. Symposium on Discrete Algorithms (SODA), 2008
[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

Learning-augmented online bipartite matching under random order arrival

Can we design an algorithm that is 1-consistent and $\overset{\beta}{\cancel{\left(1-\frac{1}{e}\right)}}$-robust?

- Suppose we have instance-specific prediction / advice / side-information $\hat{\mathcal{G}}$ of $\mathcal{G}^*$
- Consistency: If $\hat{\mathcal{G}} = \mathcal{G}^*$, i.e., $\hat{\mathcal{G}}$ is "perfect", then competitive ratio is 1
- Robustness: If $\hat{\mathcal{G}}$ is "garbage", then competitive ratio is β, where $0.696 \leq \beta \leq 0.823$

	(Expected) Competitive ratio	
	Adversarial arrival	**Random order arrival**
Deterministic algorithm	$\frac{1}{2}$	$1-\frac{1}{e}$ [GM08]
Deterministic hardness	$\frac{1}{2}$	$\frac{3}{4}$
Randomized algorithm	$1-\frac{1}{e} \approx 0.63$ [KVV90]	0.696 [MY11]
Randomized hardness	$1-\frac{1}{e} + o(1)$ [KVV90]	0.823 [MGS12]

← Greedy

← Ranking

Learning-augmented online bipartite matching under random order arrival

Can we design an algorithm that is 1-consistent and $\overset{\beta}{\cancel{\left(\mathbf{1}-\frac{\mathbf{1}}{\boldsymbol{e}}\right)}}$-robust?

- Suppose we have instance-specific prediction / advice / side-information $\hat{\mathcal{G}}$ of $\mathcal{G}^*$
- Consistency: If $\hat{\mathcal{G}} = \mathcal{G}^*$, i.e., $\hat{\mathcal{G}}$ is “perfect”, then competitive ratio is 1
- Robustness: If $\hat{\mathcal{G}}$ is “garbage”, then competitive ratio is β, where $0.696 \leq \beta \leq 0.823$

[CGLB24] Under random order arrivals, there is an algorithm achieving competitive ratio interpolating between 1 and $\boldsymbol{\beta} \cdot \left(\mathbf{1} - \boldsymbol{o}(\mathbf{1})\right)$, depending on advice quality

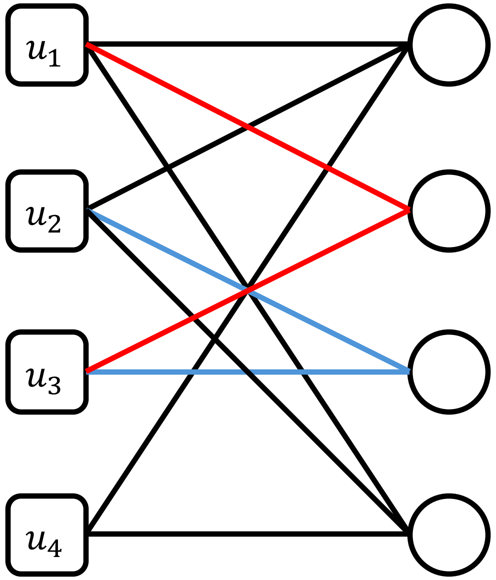
- So, we are simultaneously 1-consistent and $\beta \cdot \left(1 - o(1)\right)$-robust
- Meta-algorithm that uses any Baseline that achieves β, e.g., use Ranking for Baseline

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [CGLB24]

Realized type counts as advice

- Type of v_i = set of offline vertices in $N(v_i)$ that v_i is adjacent to [BKP20]
- True graph $\mathcal{G}^*$ can be represented by integer vector $c^* \in \mathbb{N}^{2^n}$, indexed by all possible types $2^{\boldsymbol{U}}$
- Similarly, advice graph $\hat{\mathcal{G}}$ can be represented using $\hat{c} \in \mathbb{N}^{2^n}$



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

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[BKP20] Allan Borodin, Christodoulos Karavasilis, and Denis Pankratov. *An experimental study of algorithms for online bipartite matching*. Journal of Experimental Algorithmics (JEA), 2020

A glimpse of [CGLB24]

Realized type counts as advice

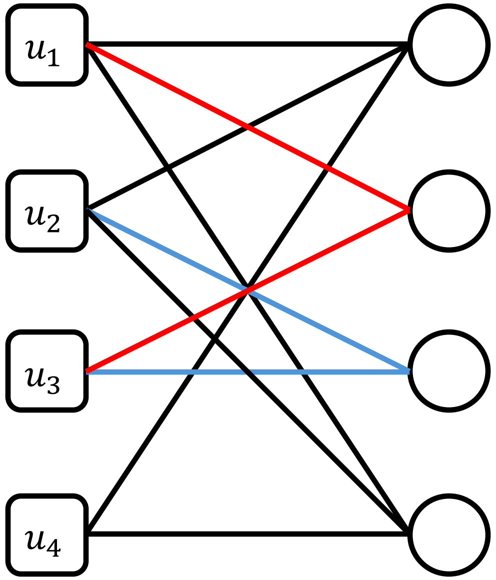
- Type of v_i = set of offline vertices in $N(v_i)$ that v_i is adjacent to [BKP20]
- True graph $\mathcal{G}^*$ can be represented by integer vector $c^* \in \mathbb{N}^{2^n}$, indexed by all possible types $2^{\boldsymbol{U}}$
- Similarly, advice graph $\hat{\mathcal{G}}$ can be represented using $\hat{c} \in \mathbb{N}^{2^n}$
- Representation vectors c^* and $\hat{c}$ are sparse with at most n non-zero entries
  - Let $T^* \subseteq 2^{\boldsymbol{U}}$ be the subset of non-zero counts in c^*
  - Let $\hat{T} \subseteq 2^{\boldsymbol{U}}$ be the subset of non-zero counts in $\hat{c}$
  - Then, $|T^*|, \left|\hat{T}\right| \leq n \ll 2^{|\boldsymbol{U}|} = 2^n$
  - Implication: Can represent using $O(n)$ labels and numbers

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[BKP20] Allan Borodin, Christodoulos Karavasilis, and Denis Pankratov. *An experimental study of algorithms for online bipartite matching*. Journal of Experimental Algorithmics (JEA), 2020

A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, leave unmatched.



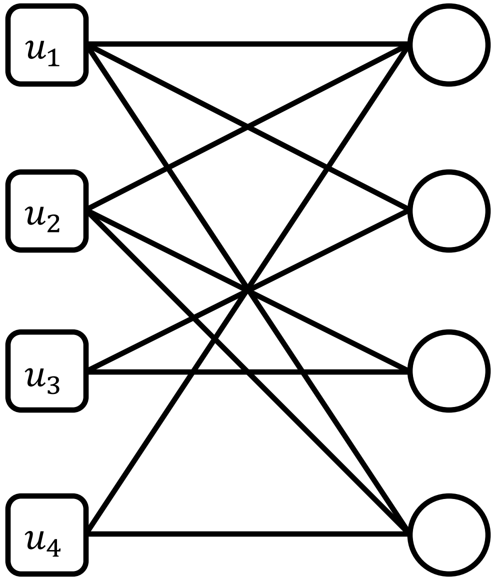
Type	c^*
$\{u_1, u_2, u_4\}$	2
$\{u_1, u_3\}$	1
$\{u_2, u_3\}$	1

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

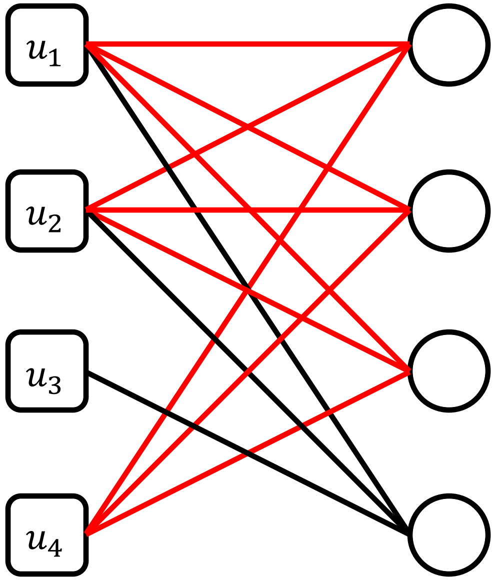
A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

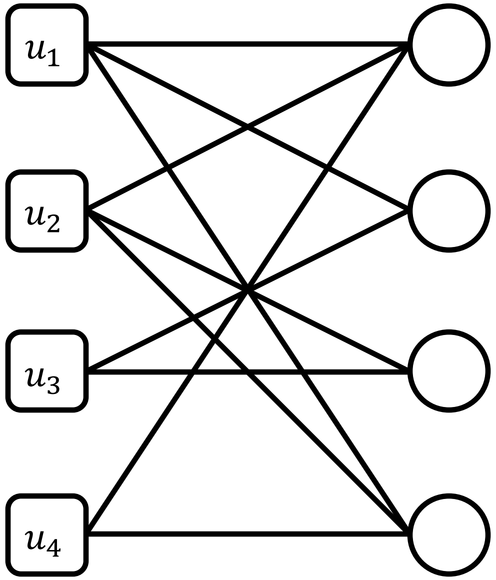


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

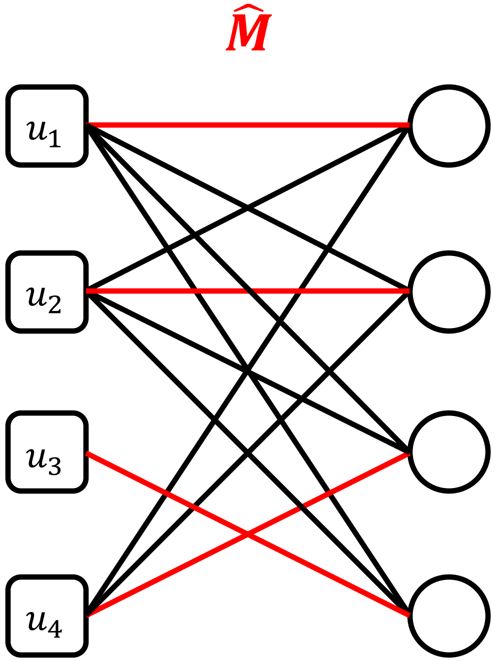
A glimpse of [CGLB24]

The Mimic algorithm, i.e., "Blindly trust advice as much as possible"

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

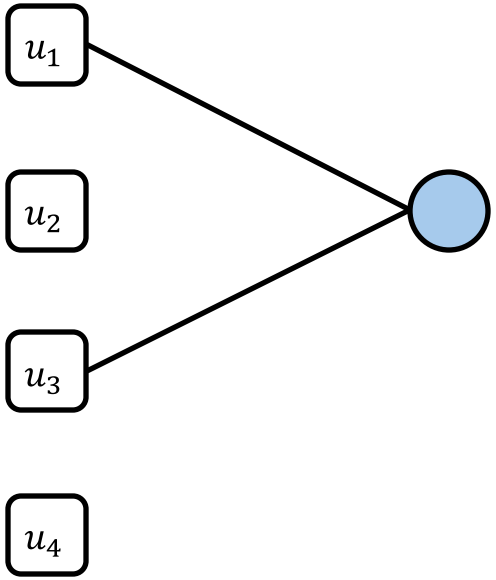


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

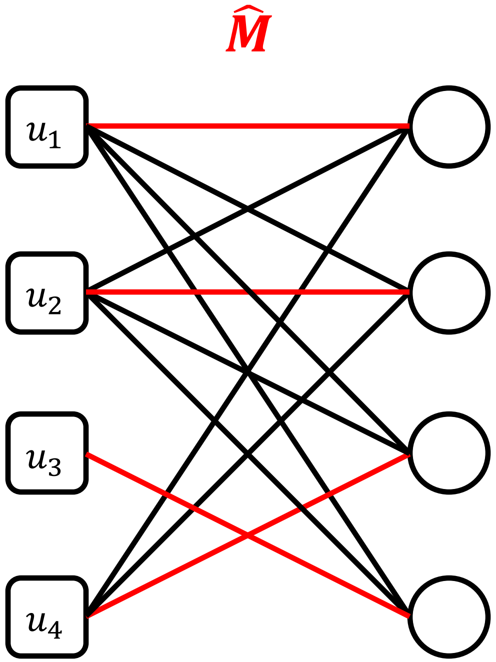
A glimpse of [CGLB24]

The Mimic algorithm, i.e., "Blindly trust advice as much as possible"

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

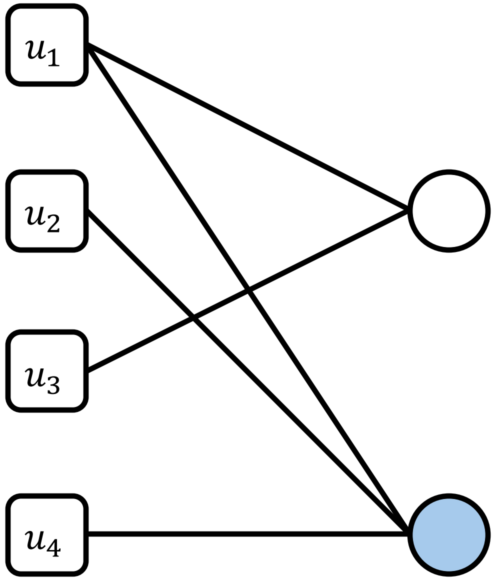


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

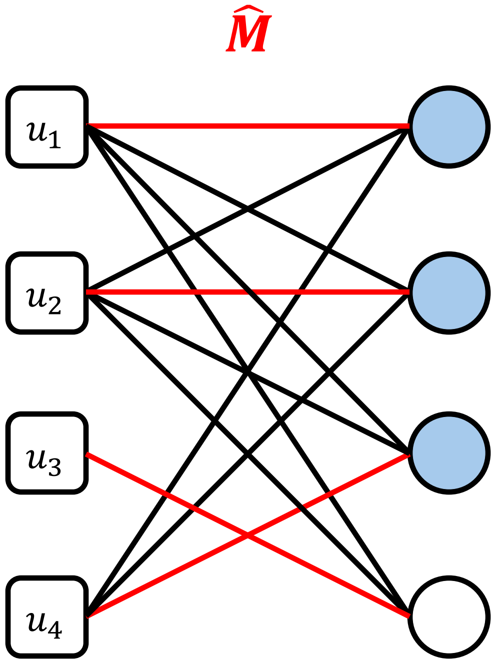
A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

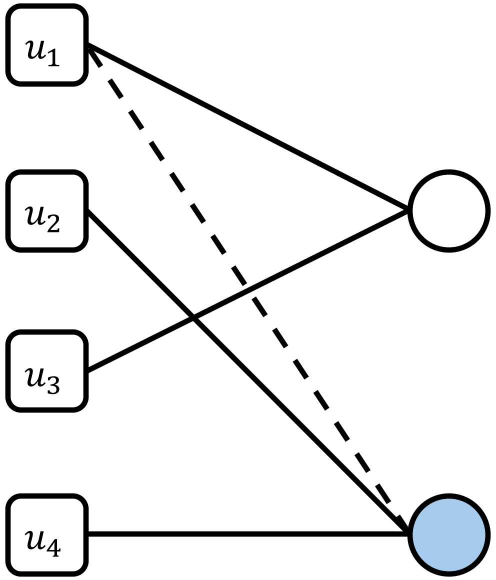


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

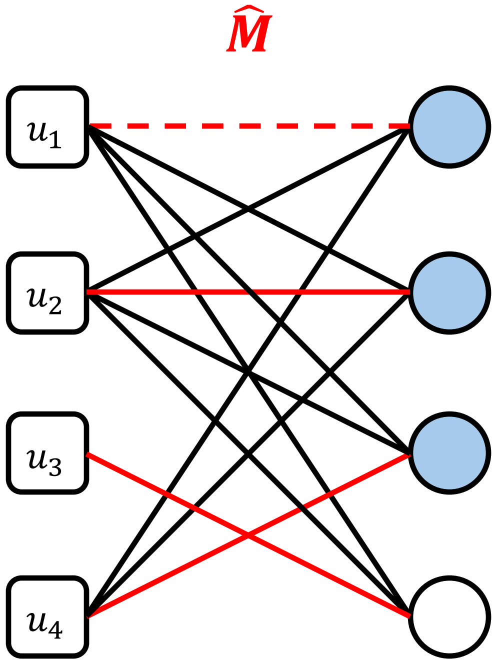
A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

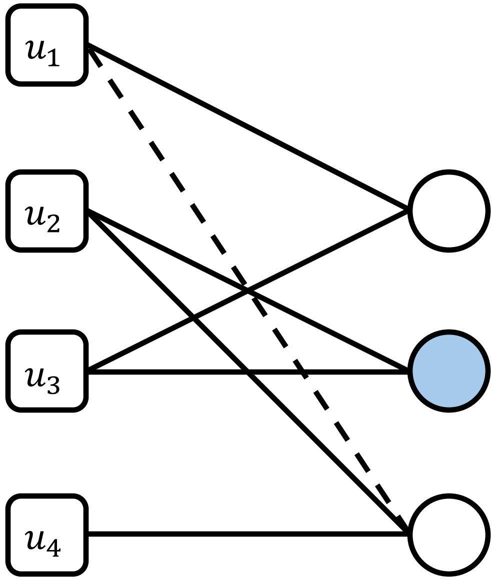


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

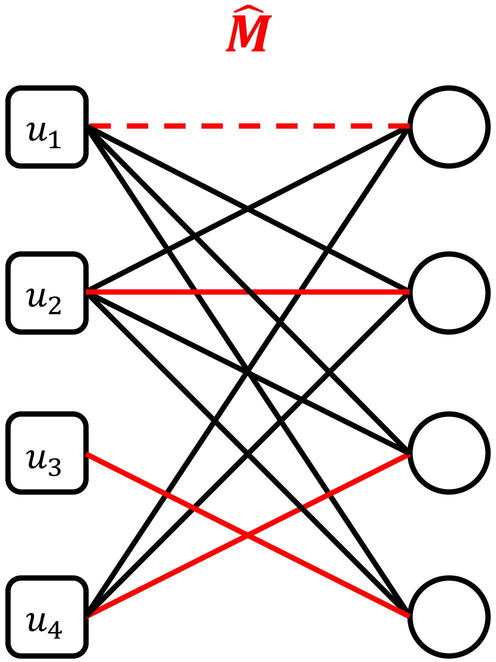
A glimpse of [CGLB24]

The Mimic algorithm, i.e., "Blindly trust advice as much as possible"

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

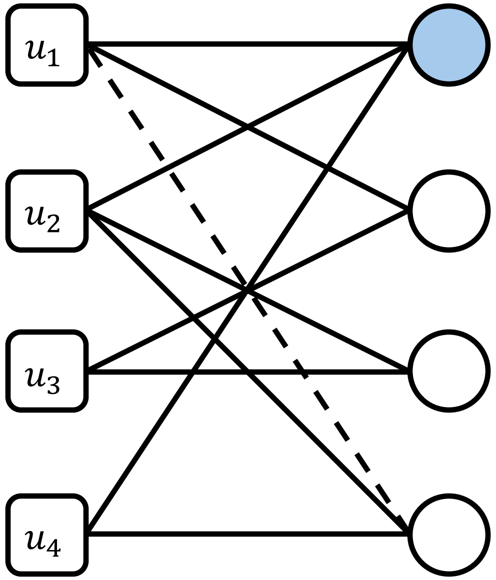


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

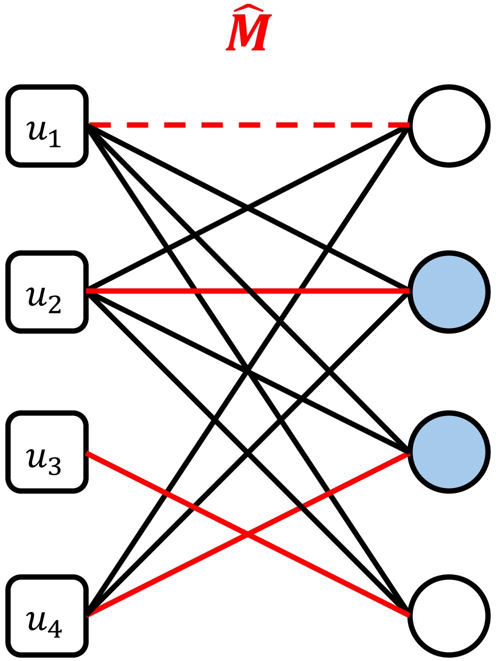
A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

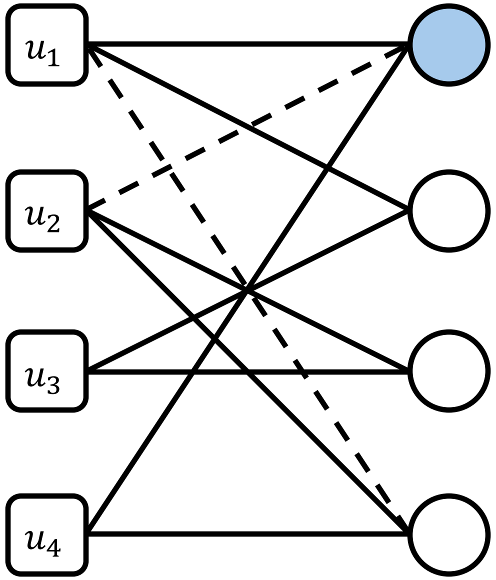


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

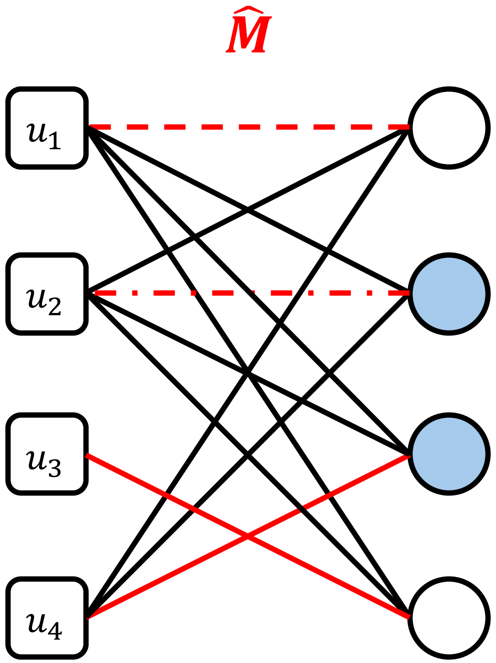
A glimpse of [CGLB24]

The Mimic algorithm, i.e., "Blindly trust advice as much as possible"

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

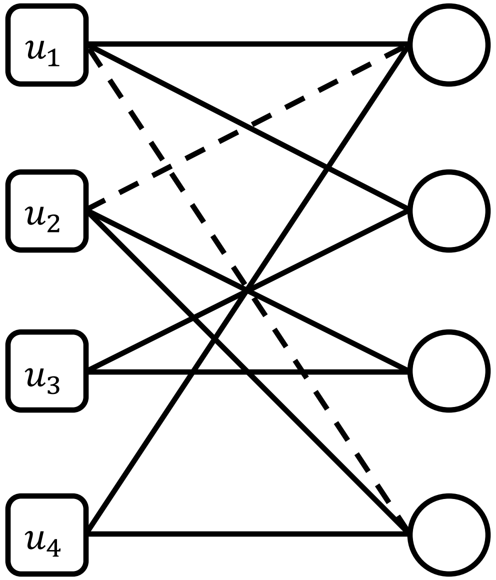


[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, 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

Output matching size

$$2 = |\widehat{\boldsymbol{M}}| - \frac{\ell_1(c^*, \hat{c})}{2}$$

Error is “double counted” in ℓ_1

$$\ell_1(c^*, \hat{c}) = |3 - 2| + |0 - 1| + |0 - 1| + |1 - 0| = 4$$

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, leave unmatched.

Analysis

- $0 \leq \ell_1(c^*, \hat{c}) \leq 2n$ measures how close $\hat{c}$ is to c^*
- By blindly following advice, Mimic gets a matching of size $\left|\widehat{\boldsymbol{M}}\right| - \frac{\ell_1(c^*,\hat{c})}{2}$
- Mimic beats an advice-free Baseline whenever $\left|\widehat{\boldsymbol{M}}\right| - \frac{\ell_1(c^*,\hat{c})}{2} > \beta \cdot n$

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [CGLB24]

The Mimic algorithm, i.e., “Blindly trust advice as much as possible”

- Fix arbitrary maximum matching $\widehat{\boldsymbol{M}}$ defined by $\hat{c}$
- Try to follow it as much as possible. If unable to mimic $\widehat{\boldsymbol{M}}$, leave unmatched.

Analysis

- $0 \leq \ell_1(c^*, \hat{c}) \leq 2n$ measures how close $\hat{c}$ is to c^*
- By blindly following advice, Mimic gets a matching of size $\left|\widehat{\boldsymbol{M}}\right| - \frac{\ell_1(c^*,\hat{c})}{2}$
- Mimic beats an advice-free Baseline whenever $\left|\widehat{\boldsymbol{M}}\right| - \frac{\ell_1(c^*,\hat{c})}{2} > \beta \cdot n$
- For simplicity, let’s suppose that $\left|\widehat{\boldsymbol{M}}\right| = n$ for this talk
  - Mimic beats an advice-free Baseline whenever $\frac{\ell_1(c^*,\hat{c})}{n} < 2 \cdot (1 - \beta)$
  - Problem: We don’t know $\ell_1(c^*, \hat{c})$ upfront!

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [CGLB24]

Sublinear property testing to the rescue

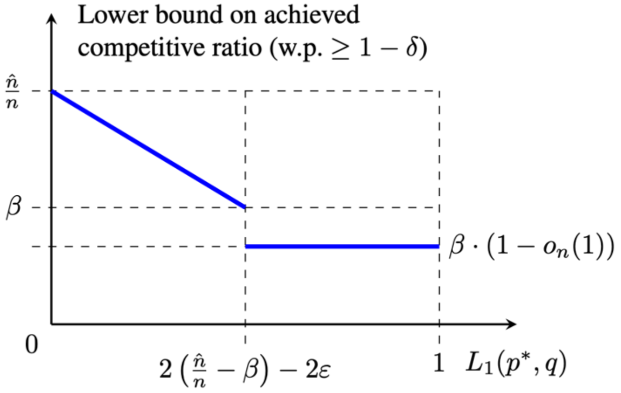
- Define $p = \frac{c^*}{n}$ and $q = \frac{\hat{c}}{n}$ as distributions over the $2^{\boldsymbol{U}}$ types
- [VV11, JHW18]: Can estimate $\ell_1(p, q)$ “well” using $o(n)$ i.i.d. samples
  - To be precise, if p and q have domain size $r \leq n$, then $\Theta\left(\frac{r}{\varepsilon^2 \log r}\right)$ i.i.d. samples are sufficient and necessary to estimate $\hat{\ell}_1$ such that $\left|\hat{\ell}_1 - \ell_1(p, q)\right| \leq \varepsilon$
  - c^* and $\hat{c}$ can be defined over $\left|\hat{T}\right| + 1$ elements, with a not-in-$\hat{T}$ bucket
- Some adjustments needed (included for completeness)
  - Random vertex arrivals are “sampling without replacement”, but we can simulate i.i.d. samples by tracking what has arrived and “reusing” arrivals with probability proportional to number of arrivals
  - ℓ_1 estimator is in expectation, but can be made “with high probability”

[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[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]

[CGLB24] Under random order arrivals, there exists a meta-algorithm (TestAndMatch) that uses any Baseline (achieving $\boldsymbol{\beta}$) and achieves competitive ratio interpolating between 1 and $\boldsymbol{\beta} \cdot \left(\mathbf{1} - \boldsymbol{o}(\mathbf{1})\right)$, depending on advice quality

- The TestAndMatch algorithm
  - Fix any arbitrary maximum matching $\widehat{\boldsymbol{M}}$ on the graph defined by advice $\hat{c}$
  - If $\left|\widehat{\boldsymbol{M}}\right| \leq \beta \cdot n$, run the best advice-free Baseline on all arrivals
  - Otherwise, run Mimic while testing quality of $\hat{c}$ by estimating $\ell_1(c^*, \hat{c})$
  - If test declares $\ell_1(c^*, \hat{c})$ is "large", use Baseline for remaining arrivals
  - Otherwise, continue using Mimic for remaining arrivals
- Analysis
  - If $\hat{\ell}_1 \lesssim 2(1-\beta)$, then TestAndMatch attains ratio of at least $1 - \frac{\ell_1(c^*, \hat{c})}{2n}$
  - Otherwise, TestAndMatch attains ratio of at least $\beta \cdot \left(1 - o(1)\right)$



[CGLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024

Test-and-Act framework

Insight: “Testing can be easier than learning”

- Idea: Design suitable testing subroutine to estimate advice quality, then react accordingly
- [CGB23] TestAndSubsetSearch: Reduce number of interventions for causal graph discovery
- [CGLB24] TestAndMatch: Improve competitive ratio of online bipartite matching
  - No 1-consistent and $> \frac{1}{2}$-robust algorithm under adversarial arrival
  - Meta-algorithm TestAndMatch interpolates between 1 and $\beta \cdot \left(1 - o(1)\right)$ under random order arrival
- [BCGG24] TestAndOptimize: Improve sample complexity of learning multivariate Gaussians
- [BCGG25] TestAndOptimize: Improve sample complexity of learning product distributions

[CGB23] Davin Choo, Themistoklis Gouleakis, Arnab Bhattacharyya. *Active causal structure learning with advice*. International Conference on Machine Learning (ICML), 2023

[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*. International Conference on Machine Learning (ICML), 2024

[BCGG25] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Product Distribution Learning with Imperfect Advice*. Conference on Neural Information Processing Systems (NeurIPS) Spotlight, 2025

Learning multivariate Gaussians

Producing a candidate distribution $\hat{\mathcal{P}}$ when drawing i.i.d. samples from $\mathcal{P} = N(\mu, I)$

- PAC-learning [Val84]: We want $TV(\mathcal{P}, \hat{\mathcal{P}})$ to be small, with “good chance”, using few samples
- For Gaussians with identity covariance, we just need good estimation $\hat{\mu}$ of the mean $\mu \in \mathbb{R}^d$

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

Learning and testing multivariate Gaussians

Producing a candidate distribution $\hat{\mathcal{P}}$ when drawing i.i.d. samples from $\mathcal{P} = N(\mu, I)$

- PAC-learning [Val84]: We want $TV(\mathcal{P}, \hat{\mathcal{P}})$ to be small, with "good chance", using few samples
- For Gaussians with identity covariance, we just need good estimation $\hat{\mu}$ of the mean $\mu \in \mathbb{R}^d$

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

- Idea: Empirical mean works

Quadratic gap!

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

- If $\|\mu\|_2 \leq \varepsilon$, output Accept
- If $\|\mu\|_2 \geq 2\varepsilon$, output Reject
- Otherwise, decide arbitrarily
- Idea: Define statistic Z and threshold τ. Output "Accept" if $Z > \tau$, else output "Reject"

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

A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality

[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

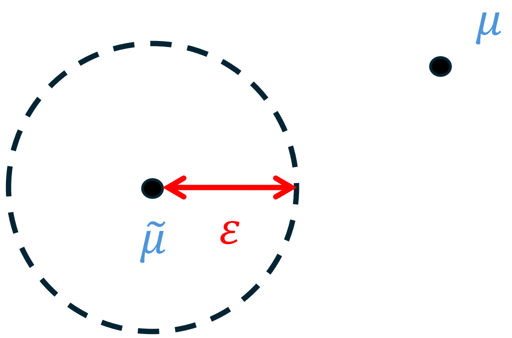
Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality
- Idea: Use tolerant testing to first estimate advice quality of $\tilde{\mu}$

"Guess and double"



[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

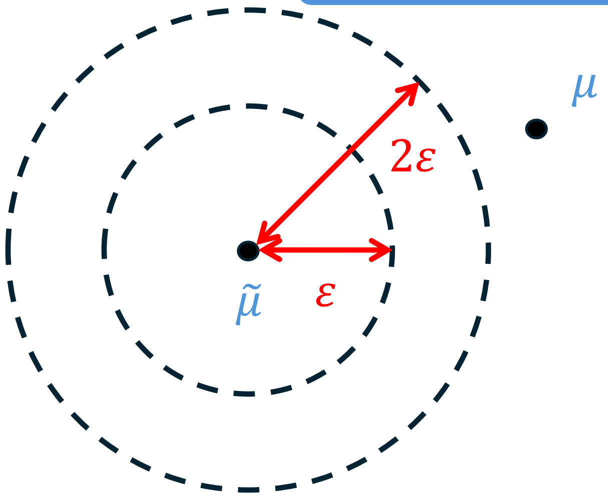
Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality
- Idea: Use tolerant testing to first estimate advice quality of $\tilde{\mu}$

"Guess and double"



[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

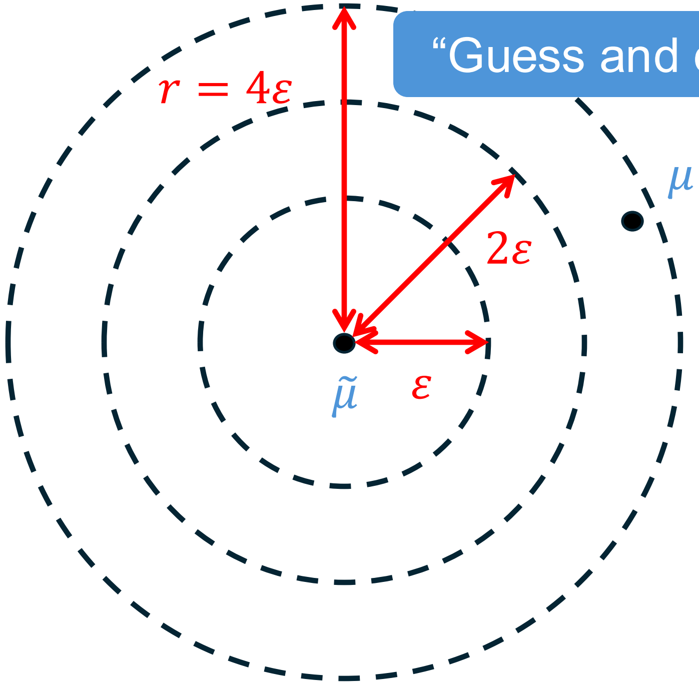
A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce “ε-good” $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to “ε-tolerant-test” μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality
- Idea: Use tolerant testing to first estimate advice quality of $\tilde{\mu}$



“Guess and double”

[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

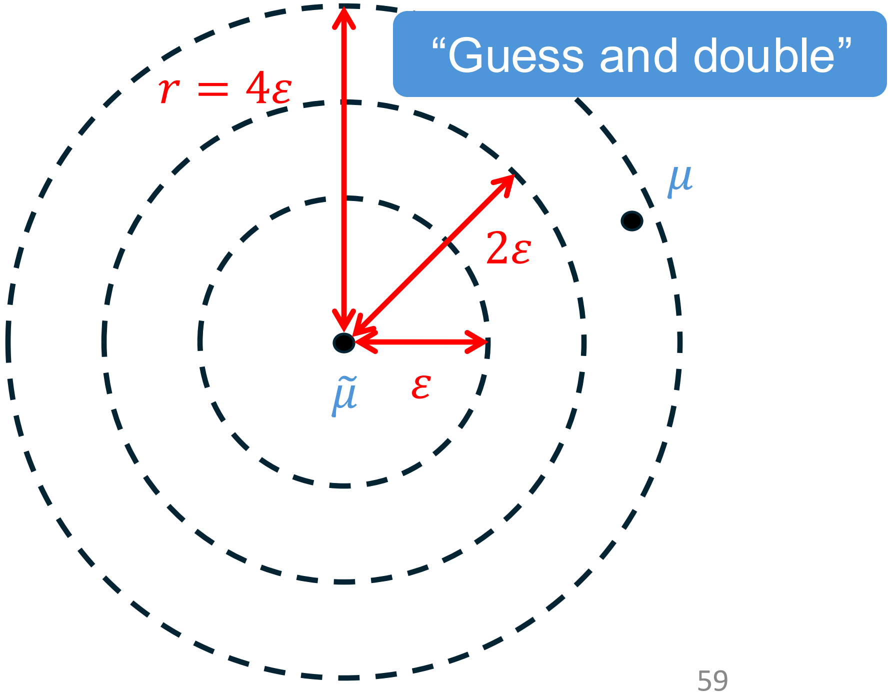
- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality
- Idea: Use tolerant testing to first estimate advice quality of $\tilde{\mu}$, then run LASSO-style optimization to produce $\hat{\mu}$

$$\hat{\mu} = \operatorname{argmin}_{\|\beta\|_1 \leq r} \sum_{i=1}^{n} \left\|x^{(i)} - \beta\right\|_2^2$$

samples

- Polynomial time with overall sample complexity

$$\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\left(\min\left\{1, \frac{\|\mu - \tilde{\mu}\|_1^2}{\varepsilon^2 d}\right\}\right)\right)$$



[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- If $\ell_2(\mu, \tilde{\mu})$ is small enough, just output $\tilde{\mu}$ with zero samples
- Unfortunately, we do not know the advice quality
- Idea: Use tolerant testing to first estimate advice quality of $\tilde{\mu}$, then run LASSO-style optimization to produce $\hat{\mu}$

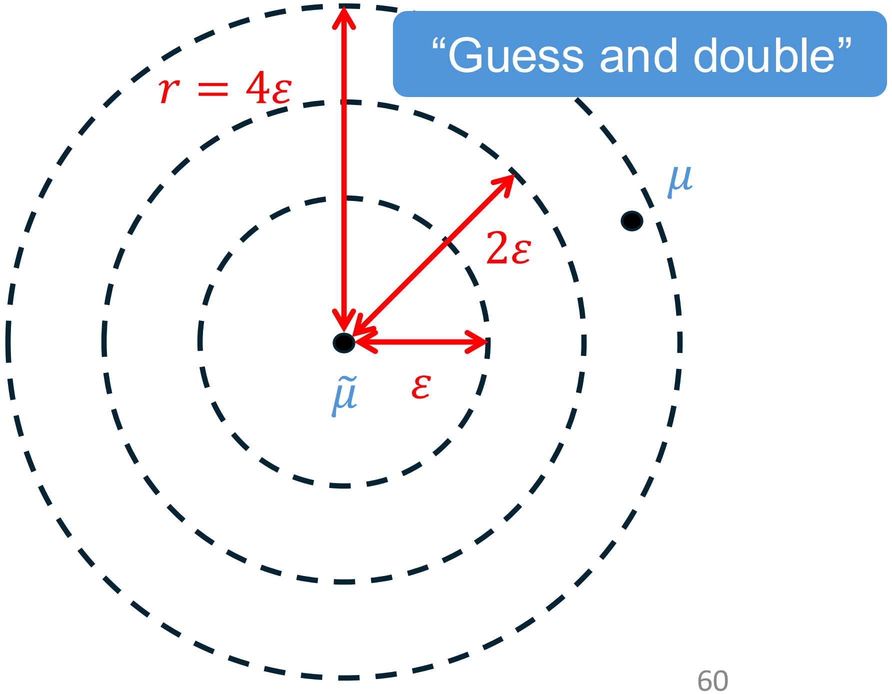
$$\hat{\mu} = \operatorname{argmin}_{\|\beta\|_1 \le r} \sum_{i=1}^{n} \left\|x^{(i)} - \beta\right\|_2^2$$

These ℓ_1 are not typos.

samples

- Polynomial time with overall sample complexity

$$\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\left(\min\left\{1, \frac{\|\mu - \tilde{\mu}\|_1^2}{\varepsilon^2 d}\right\}\right)\right)$$



[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

$$\boxed{\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\left(\min\left\{1, \frac{\|\mu - \tilde{\mu}\|_1^2}{\varepsilon^2 d}\right\}\right)\right)}$$

- Why ℓ_1-norm?
  - Short answer: It just pops out of analysis
  - Small ℓ_1-norm ⇒ Difference is approximately sparse
  - Sample complexity is linear in sparsity, not in dimension d
  - If advice close in ℓ_2-norm, linear $\Omega(d)$ sample complexity lower bounds apply unfortunately
- How to estimate ℓ_1 efficiently using ℓ_2 tolerant tester?
  - Naively using $\|x\|_2 \leq \|x\|_1 \leq \sqrt{d} \cdot \|x\|_2$ does not escape $\Omega(d)$ sample complexity
  - Idea: Exploit independence in the coordinates in the mean vector
    - Estimate ℓ_1 of length k chunks, *then* optimize the parameters in analysis
  - Similar idea works for covariance matrix
    - Disjoint principal submatrices define independent Gaussians of smaller dimensions

[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

A glimpse of [BCGG24]

Learning Gaussians: $\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to produce "ε-good" $\hat{\mu}$

Tolerant testing Gaussians: $\widetilde{\Theta}\left(\frac{\sqrt{d}}{\varepsilon^2}\right)$ i.i.d. samples from $N(\mu, I)$ to "ε-tolerant-test" μ

[BCGG24]: Efficient algorithm for improving sample complexity given advice $\tilde{\mu} \in \mathbb{R}^d$

- Polynomial time with overall sample complexity

$$\widetilde{\Theta}\left(\frac{d}{\varepsilon^2}\left(\min\left\{1, \frac{\|\mu - \tilde{\mu}\|_1^2}{\varepsilon^2 d}\right\}\right)\right)$$

- When $\|\mu - \tilde{\mu}\|_1 \ll \varepsilon\sqrt{d}$, sample complexity is *sublinear in* d
- $\Omega(d)$ samples unavoidable when $\|\mu - \tilde{\mu}\|_1 \in \Omega(\varepsilon\sqrt{d})$
- Bound can be slightly parameterized, and similar idea works for non-identity covariance (with SDP instead of LASSO; matching lower bound exists)

[BCGG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024

Test-and-Act framework

Insight: “Testing can be easier than learning”

- Idea: Design suitable testing subroutine to estimate advice quality, then react accordingly
- [**C**GB23] TestAndSubsetSearch: Reduce number of interventions for causal graph discovery
- [**C**GLB24] TestAndMatch: Improve competitive ratio of online bipartite matching
  - No 1-consistent and $> \frac{1}{2}$-robust algorithm under adversarial arrival
  - Meta-algorithm TestAndMatch interpolates between 1 and $\beta \cdot \left(1 - o(1)\right)$ under random order arrival
- [B**C**GG24] TestAndOptimize: Improve sample complexity of learning multivariate Gaussians
  - Use sublinear tolerant testing to estimate ℓ_1-closeness of $\hat{\mu}$ and $\hat{\Sigma}$ to μ and Σ
  - Set up LASSO-style / SDP optimization constrained on ℓ_1
- [B**C**GG25] TestAndOptimize: Improve sample complexity of learning product distributions
  - Similar high-level idea as [B**C**GG24], but require additional unavoidable “balanced-ness” assumption

[**C**GB23] Davin Choo, Themistoklis Gouleakis, Arnab Bhattacharyya. *Active causal structure learning with advice*. International Conference on Machine Learning (ICML), 2023
[**C**GLB24] Davin Choo, Themistoklis Gouleakis, Chun Kai Ling, and Arnab Bhattacharyya. *Online bipartite matching with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[B**C**GG24] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Learning multivariate Gaussians with imperfect advice*. International Conference on Machine Learning (ICML), 2024
[B**C**GG25] Arnab Bhattacharyya, Davin Choo, Philips George John, and Themistoklis Gouleakis. *Product Distribution Learning with Imperfect Advice*. Conference on Neural Information Processing Systems (NeurIPS) Spotlight, 2025

Test-and-Act: A recipe for learning-augmented algorithms inspired by sublinear thinking

Learning-augmented algorithms are a way to harness imperfect instance-specific information

- Metrics of interest: consistency, robustness, smoothness
- Useful instance-specific prediction / advice / side-information are often available in practice!
- Does your favorite problem have possibly useful advice? Let's talk ☺

Test-and-Act framework

- Idea: Design suitable testing subroutine to estimate advice quality, then react accordingly
- Framework is broadly applicable whenever the problem setting enables "efficient testing"
- Exciting application of property testing and sublinear algorithms

Thank you for your kind attention!