MAX PLANCK INSTITUTE
FOR SECURITY AND PRIVACY

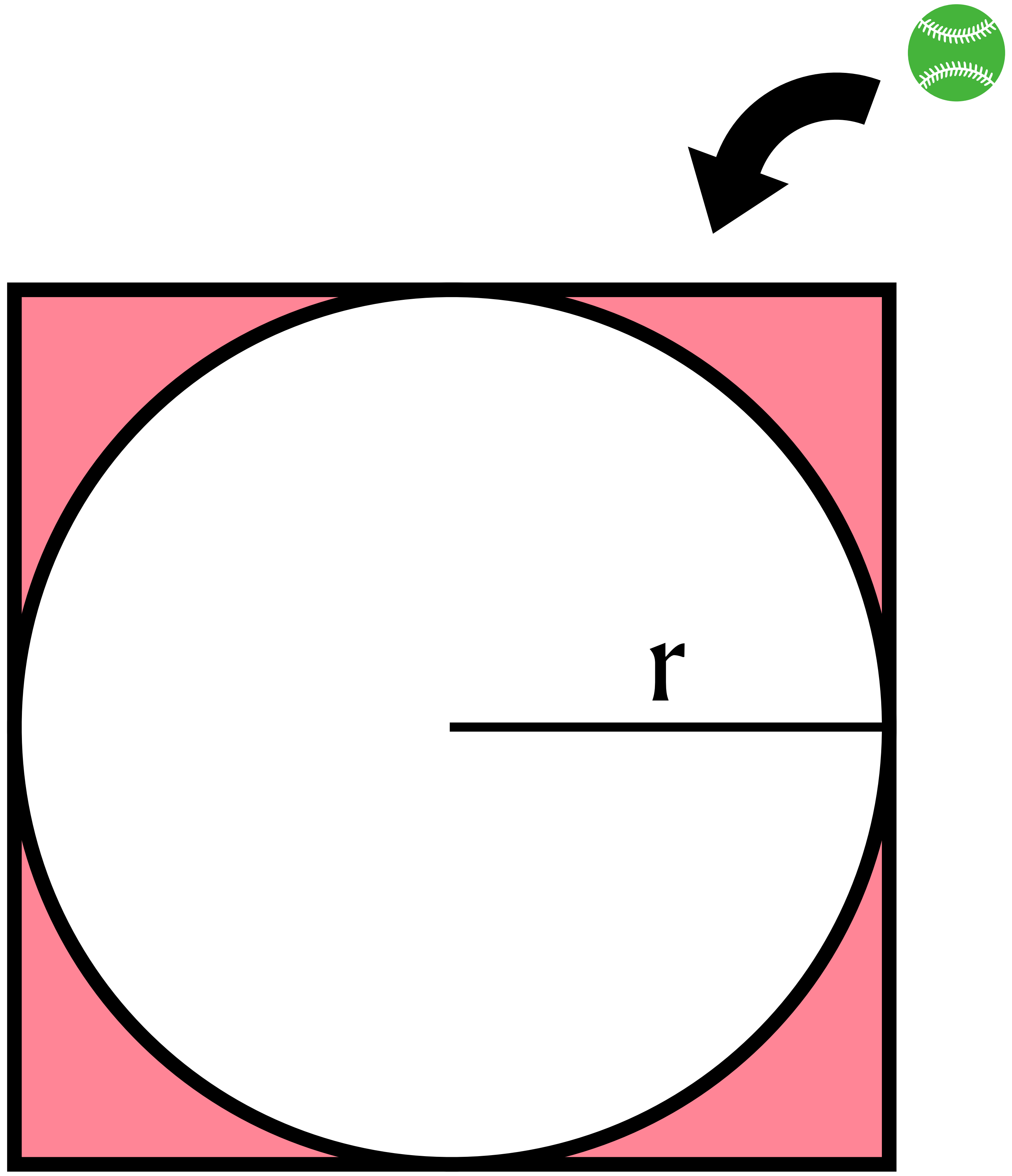
Accounting for Missing Events in Statistical Information Leakage Analysis

Seongmin Lee[1], Shreyas Minocha[2], and Marcel Böhme[1]

1. Max Planck Institute for Security and Privacy (MPI-SP)
2. Georgia Institute of Technology

ICSE 2025

Q. What is the probability of a thrown 🥎 ball to the 🟥 square dropped not into the ? area?



$$P(\neg \text{in white area}) = ?$$

Q. What is the probability of a thrown ball to the square dropped not into the ? area?

Two Ways to Solve the Problem

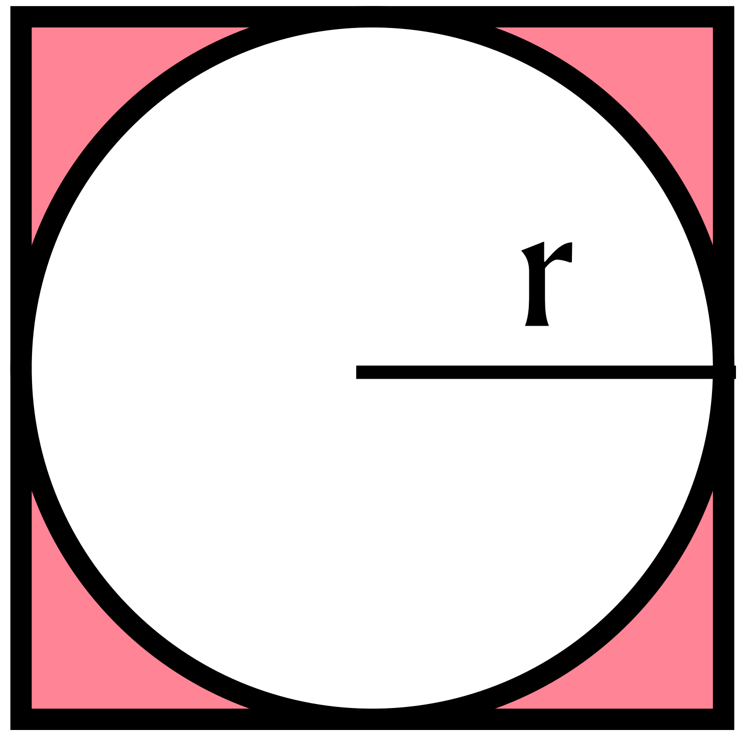
$$P(\neg \text{in white area}) = ?$$

Q. *What is the probability of a thrown ⚾ ball to the 🟥 square dropped not into the ⓠ area?*

1️⃣ Analytic methodology

If the problem can easily be ***mathematically*** *modeled,*

(e.g, area = circle)



$$\Pr(\neg \text{in circle}) = \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146...$$

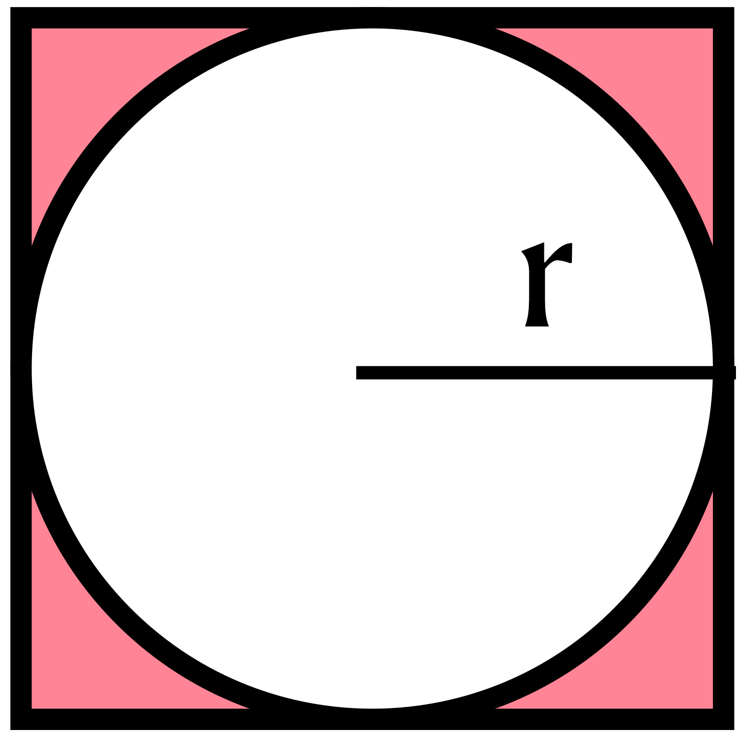
✅ ***Precise result / Formal guarantees***

Q. What is the probability of a thrown ⚾ ball to the 🟥 square dropped not into the ? area?

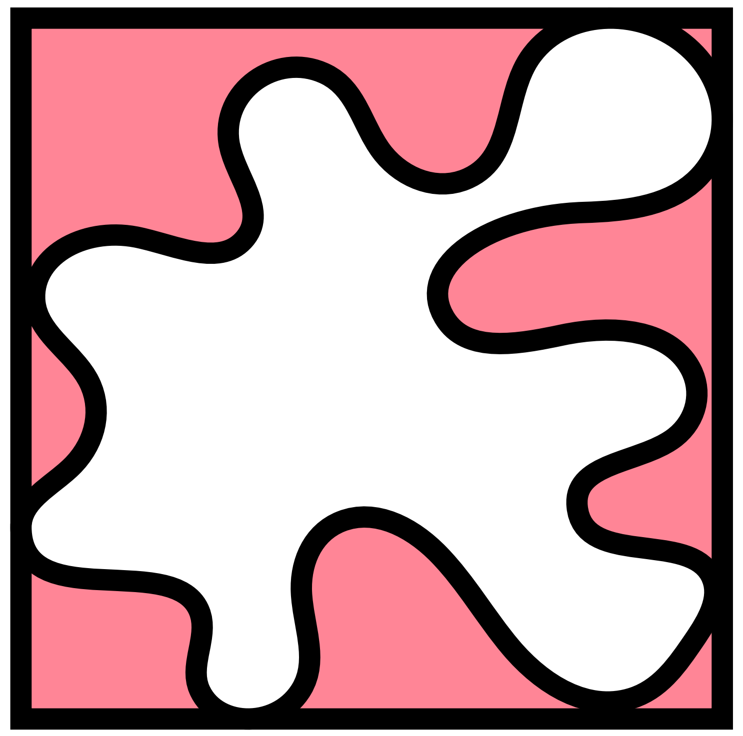
1️⃣ Analytic methodology

*If the problem can easily be **mathematically** modeled,*

(e.g, area = circle)



$$
\begin{aligned}
&\Pr(\neg\text{in circle})\\
&= \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})}\\
&= \frac{(2r)^2 - \pi r^2}{(2r)^2}\\
&= \frac{4-\pi}{4} \approx 0.2146\ldots
\end{aligned}
$$



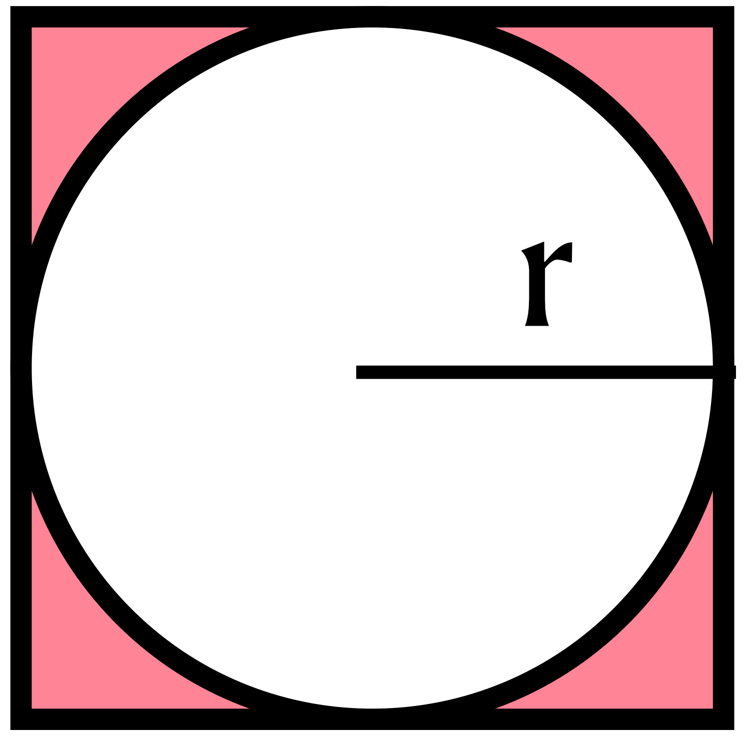
✅ ***Precise result / Formal guarantees***

Q. *What is the probability of a thrown ⚾ ball to the 🟥 square dropped not into the ? area?*

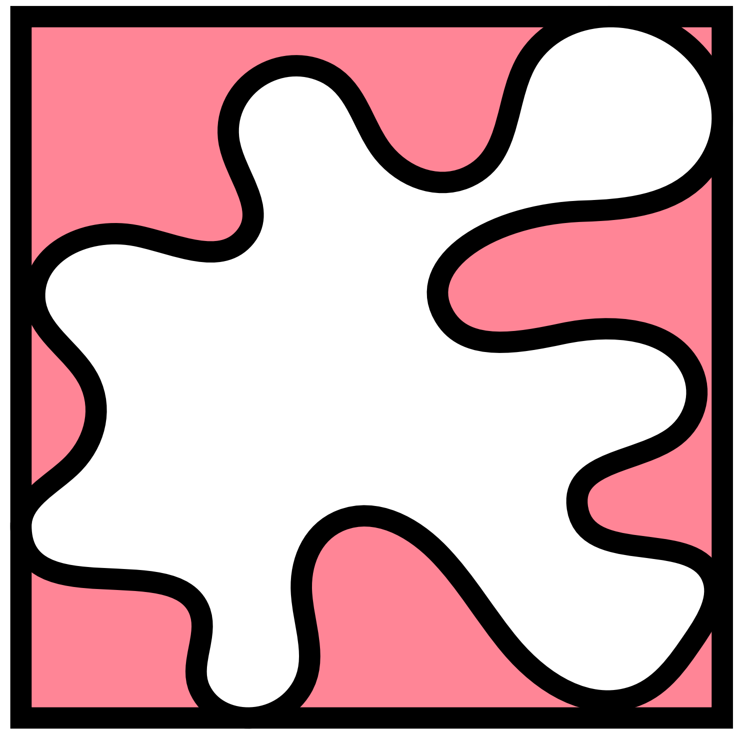
1️⃣ Analytic methodology

If the problem can easily be ***mathematically*** *modeled,*

(e.g, area = circle)



$$\Pr(\neg\text{in circle}) = \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146...$$

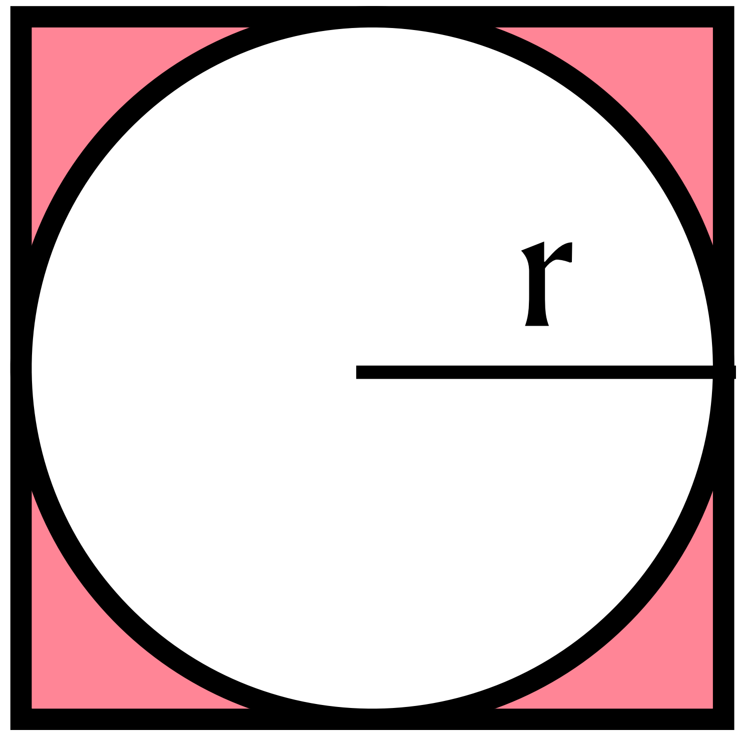


✅ ***Precise result / Formal guarantees***

Q. What is the probability of a thrown ⚾ ball to the 🟥 square dropped not into the ? area?

1️⃣ Analytic methodology

If the problem can easily be ***mathematically*** *modeled,*
(e.g, area = circle)

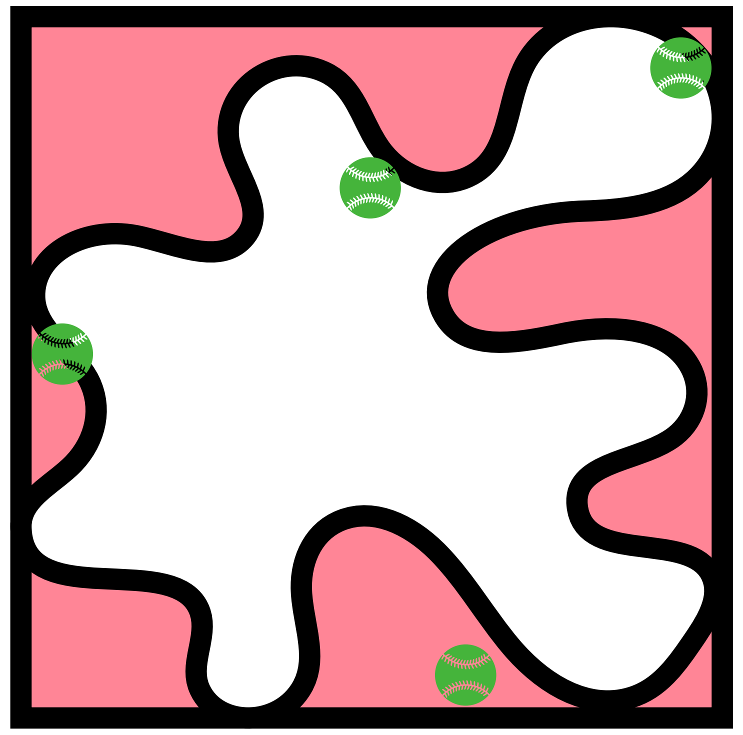


$$
\begin{aligned}
&\Pr(\neg\text{in circle})\\
&= \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})}\\
&= \frac{(2r)^2 - \pi r^2}{(2r)^2}\\
&= \frac{4-\pi}{4} \approx 0.2146\ldots
\end{aligned}
$$

✅ ***Precise result / Formal guarantees***

2️⃣ Empirical methodology

For example, the ***Monte Carlo method****, where we* ***simulate*** *the ball throwing*

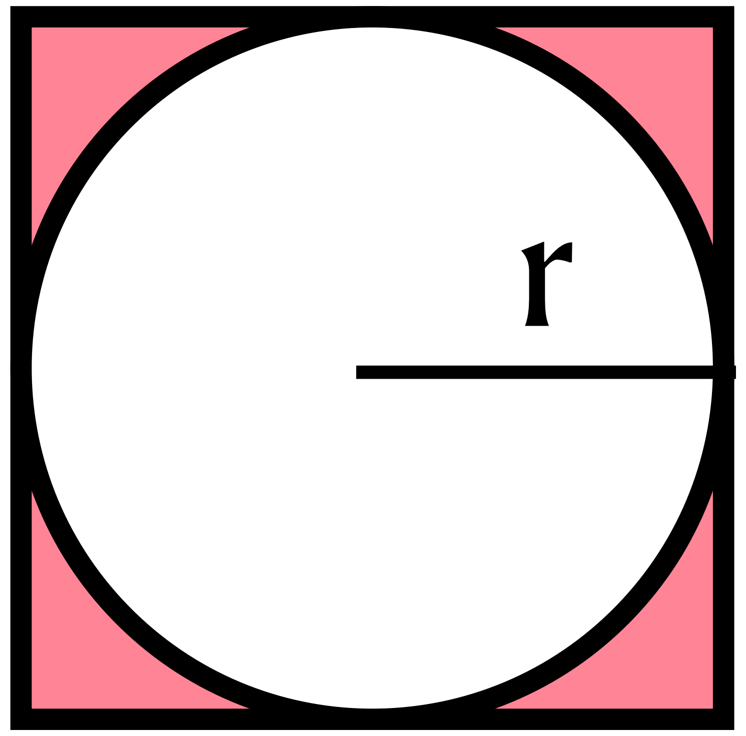


$$
\begin{aligned}
&\hat{\Pr}(\neg\text{in area})\\
&= \frac{\#\text{ of balls outside the area}}{\#\text{ of balls thrown}}\\
&= \frac{1}{4} = 0.25
\end{aligned}
$$

Q. What is the probability of a thrown 🥎 ball to the 🟥 square dropped not into the ? area?

1️⃣ Analytic methodology

*If the problem can easily be **mathematically** modeled,*
(e.g, area = circle)

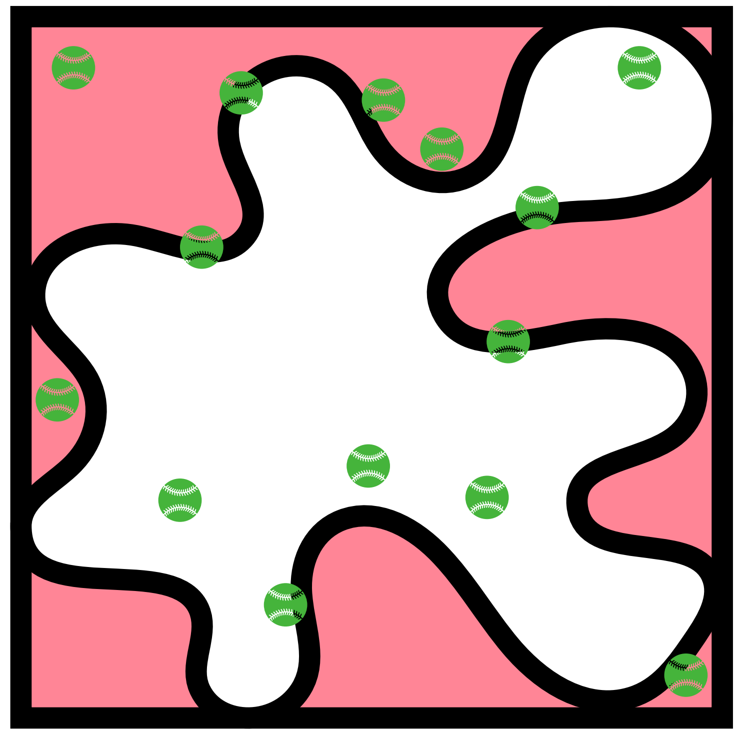


$$\Pr(\neg\text{in circle}) = \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146...$$

✅ ***Precise result / Formal guarantees***

2️⃣ Empirical methodology

*For example, the **Monte Carlo method**, where we **simulate** the ball throwing*



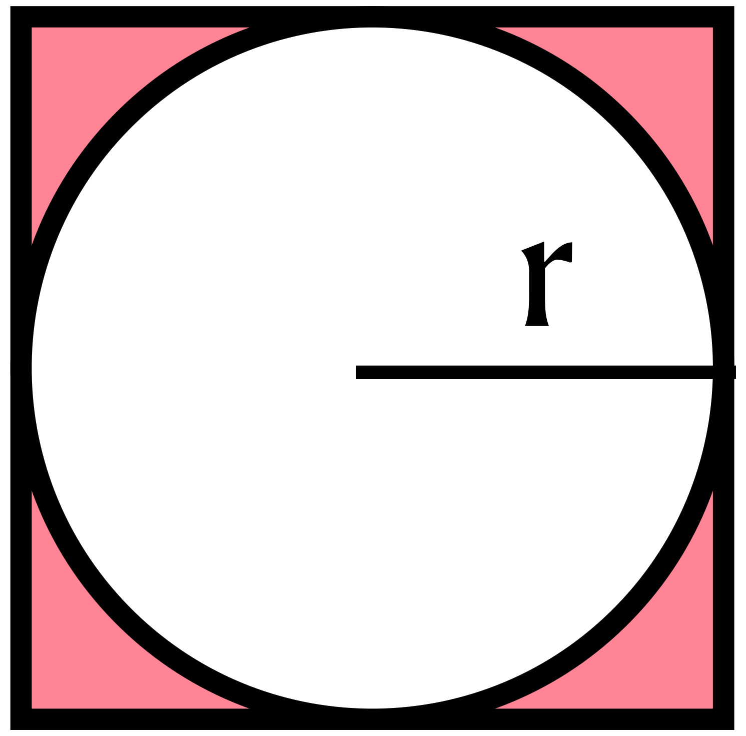
$$\hat{\Pr}(\neg\text{in area}) = \frac{\#\text{ of balls outside the area}}{\#\text{ of balls thrown}} = \frac{5}{14} \approx 0.3571$$

Q. What is the probability of a thrown 🎾 ball to the 🟥 square dropped not into the ❓ area?

1️⃣ Analytic methodology

*If the problem can easily be **mathematically** modeled,*

(e.g, area = circle)

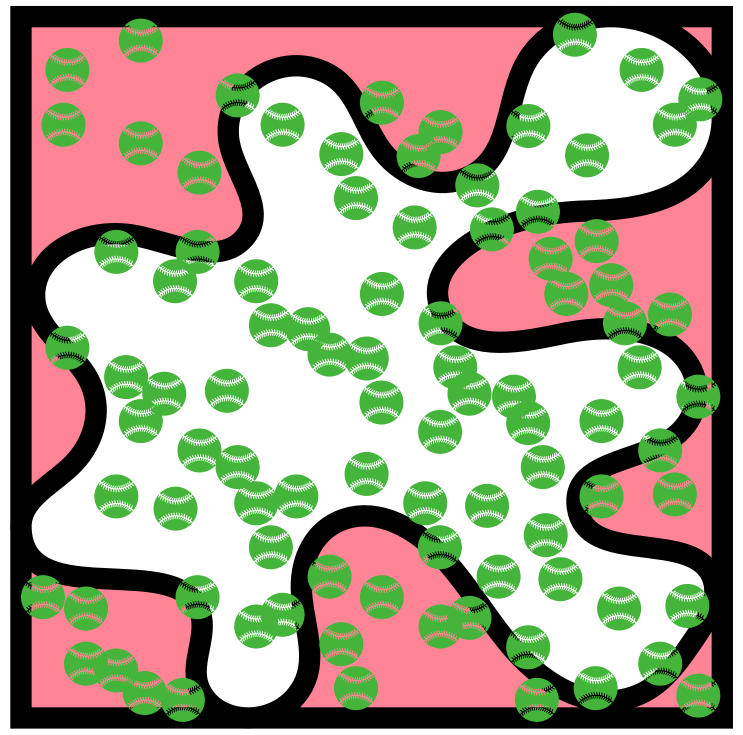


$$\Pr(\neg\text{in circle}) = \frac{\mathit{Area}(\text{Square}) - \mathit{Area}(\text{Circle})}{\mathit{Area}(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146\ldots$$

✅ ***Precise result / Formal guarantees***

2️⃣ Empirical methodology

*For example, the **Monte Carlo method**, where we **simulate** the ball throwing*



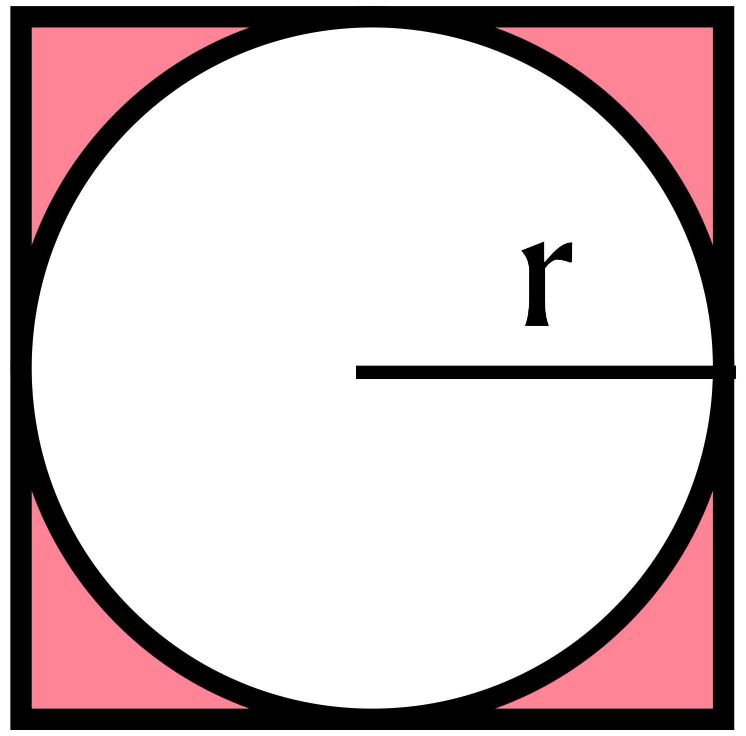
$$\hat{\Pr}(\neg\text{in area}) = \frac{\#\text{ of balls outside the area}}{\#\text{ of balls thrown}} = \frac{3577}{10000} = 0.3577$$

Q. *What is the probability of a thrown 🎾 ball to the 🟥 square dropped not into the ? area?*

1️⃣ Analytic methodology

*If the problem can easily be **mathematically** modeled,*

(e.g, area = circle)

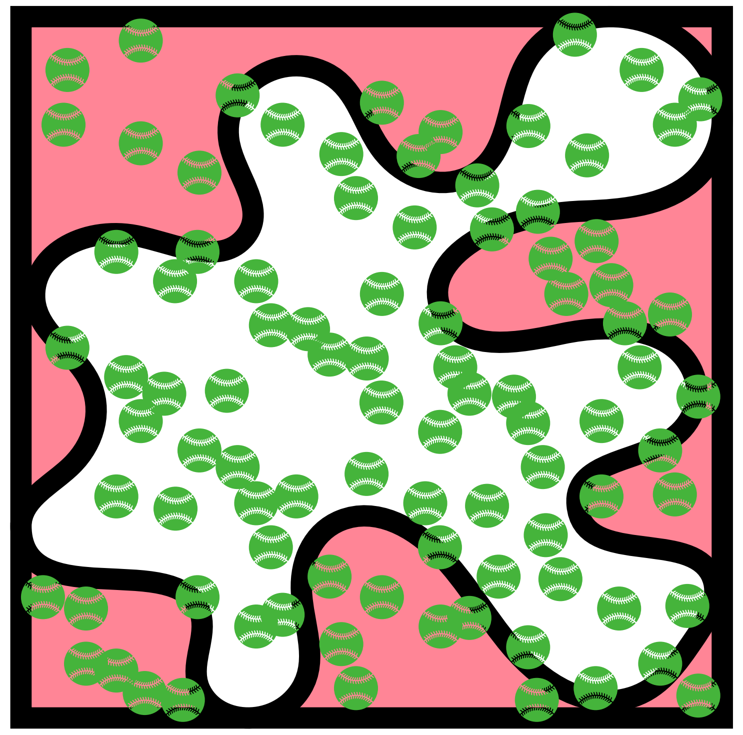


$$\Pr(\neg\text{in circle}) = \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146\ldots$$

✅ ***Precise result / Formal guarantees***

2️⃣ Empirical methodology

*For example, the **Monte Carlo method**, where we **simulate** the ball throwing*



$$\hat{\Pr}(\neg\text{in area}) = \frac{\#\text{ of balls outside the area}}{\#\text{ of balls thrown}} = \frac{3577}{10000} = 0.3577$$

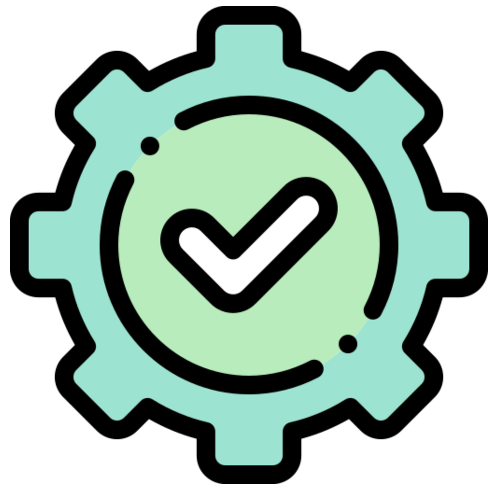
✅ ***Scalable,** i.e., can deal with complex problems*

Information Leakage Analysis

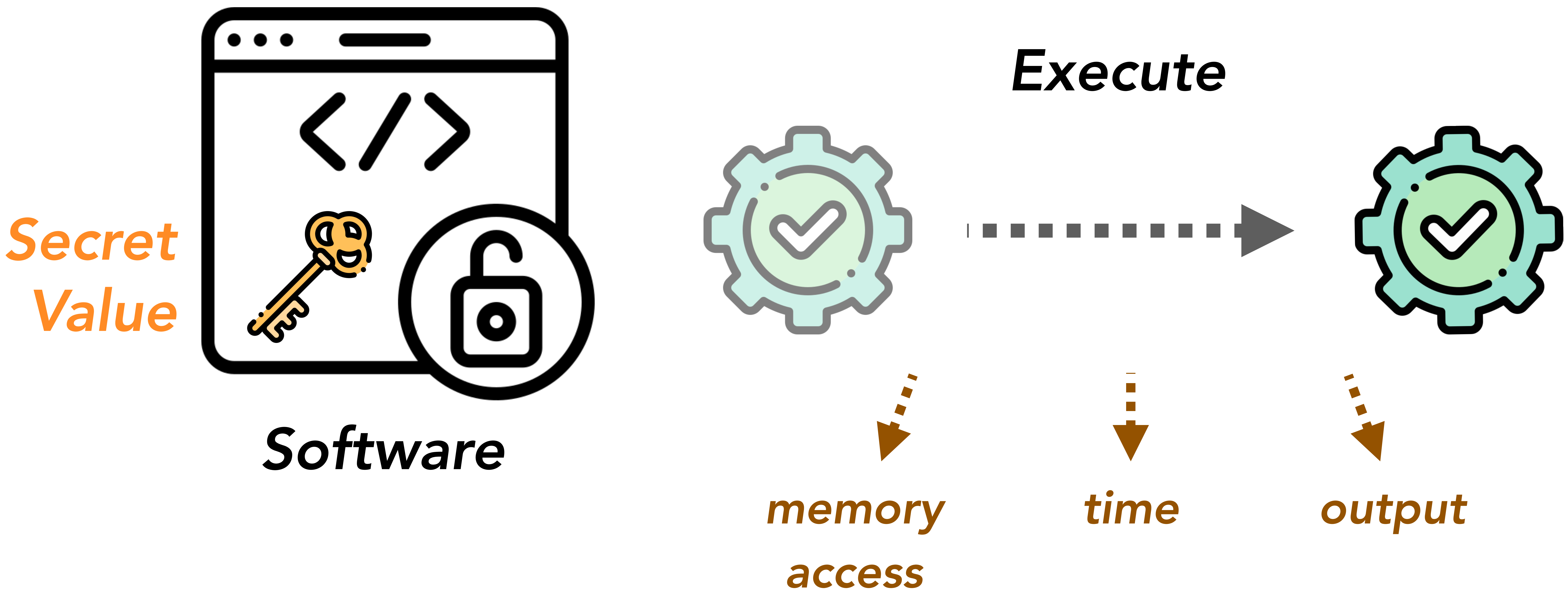
Information Leakage

Secret Value

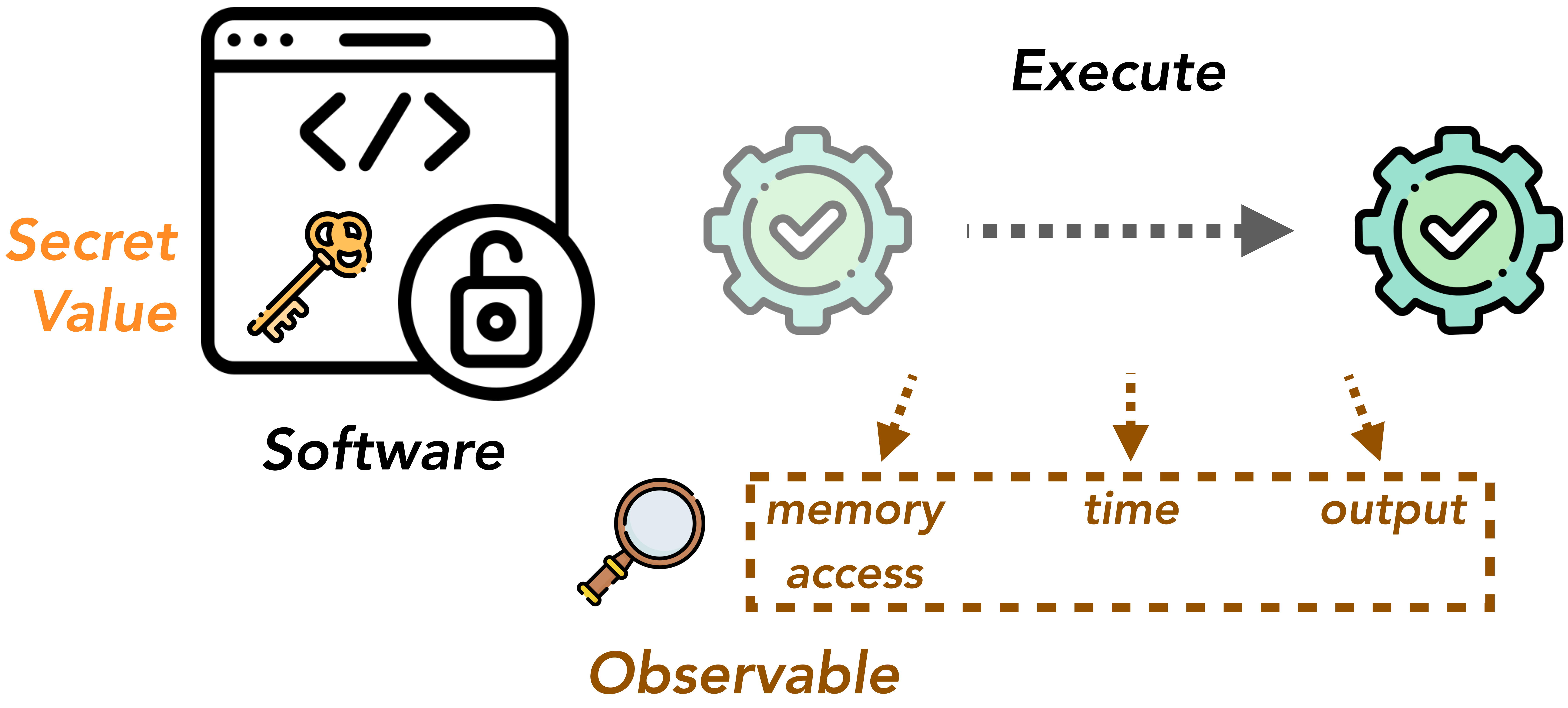
Software



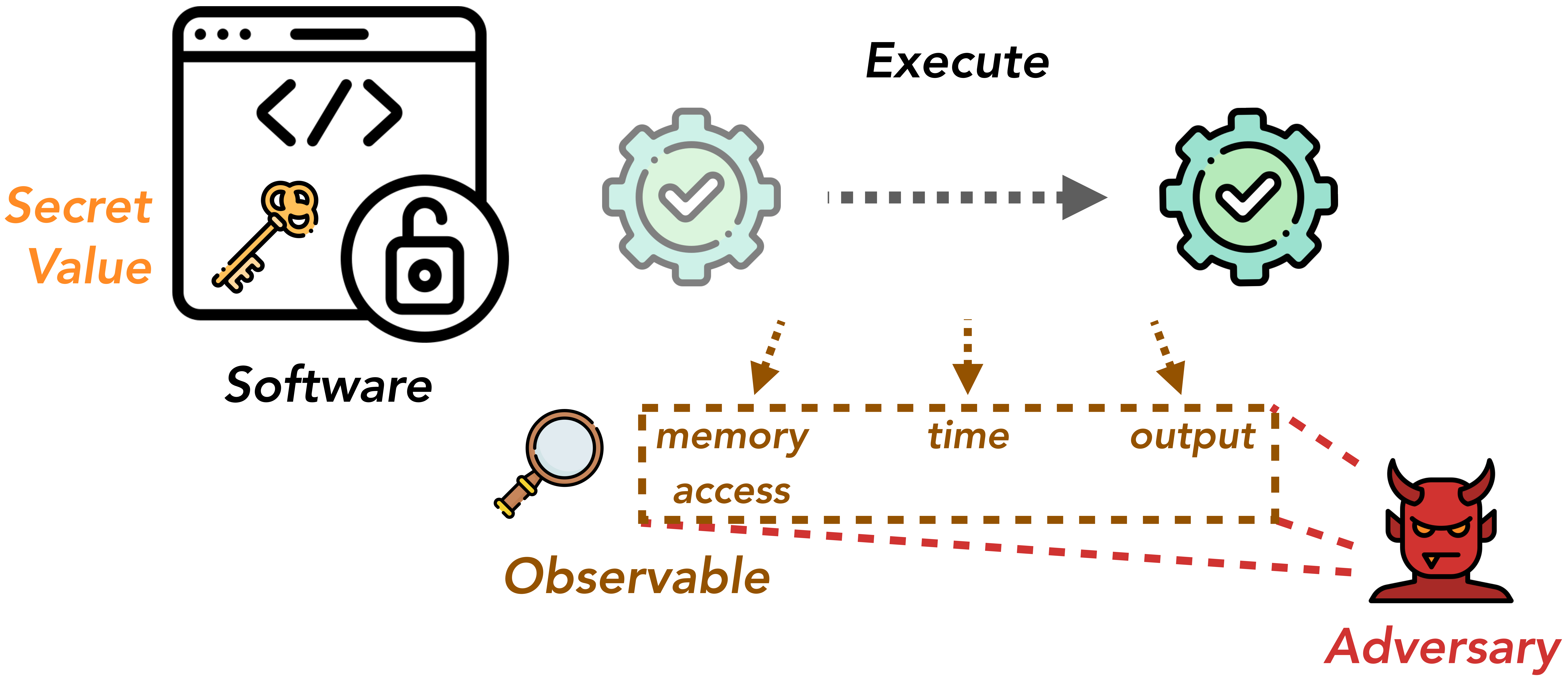
Information Leakage



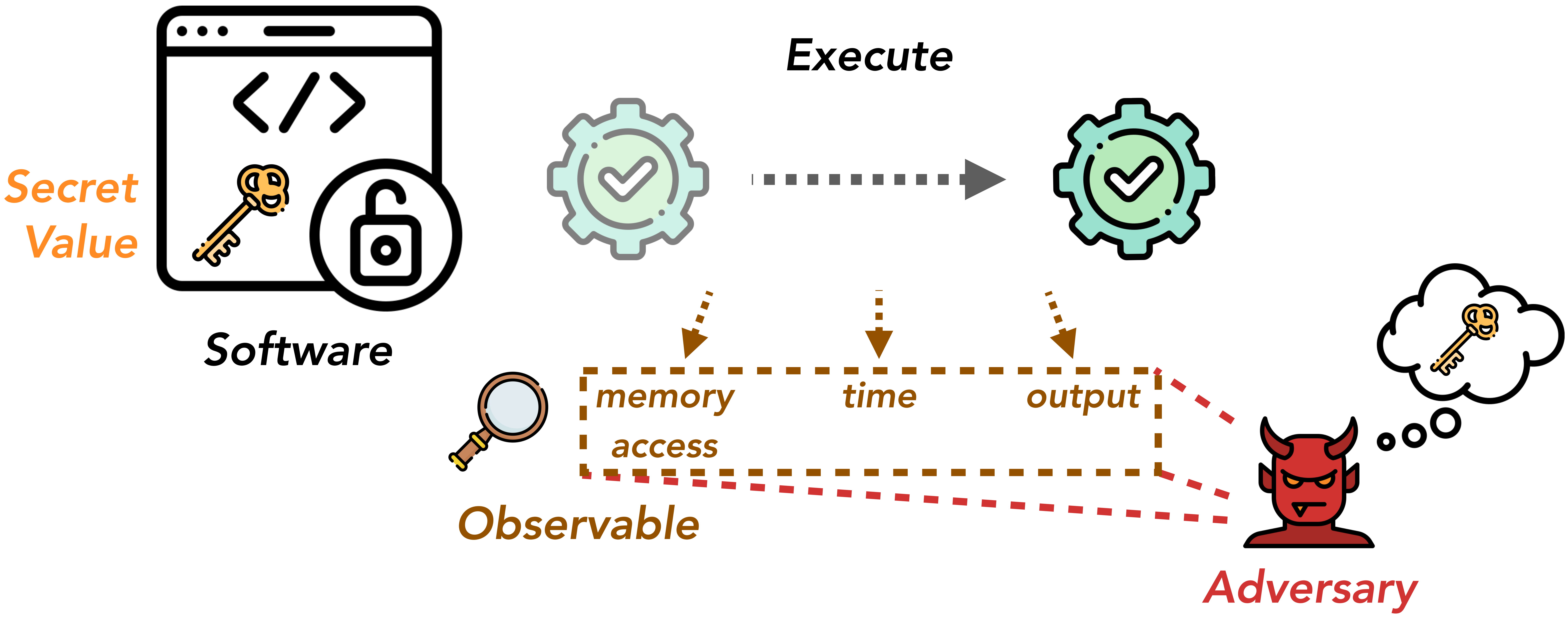
Information Leakage



Information Leakage



Information Leakage



Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O):

Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O):

 $(S$  $)$

Initial uncertainty of the ***secret value***

[Notations]

 : *Secret*

 : *Uncertainty*

 : *Observable*

Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O):

$$?(S) - ?(S \mid O)$$

Initial uncertainty of the ***secret value***

Remaining uncertainty of the ***secret value*** *after checking the* ***observable value***

[Notations]

 : Secret

 : Uncertainty

 : Observable

Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O): $?(S) - ?(S \mid O)$
- The ***Uncertainty*** ? can be measured with **Shannon Entropy** H.
  - *If the distribution D's entropy H(D) is X, it means ~2^X times of guessing are expected to match a sample from D.*

[Notations]

: *Secret*

: *Uncertainty*

: *Observable*

Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O): ?(S) – ?(S | O)
- The ***Uncertainty*** ? can be measured with **Shannon Entropy H.**
  - *If the distribution D's entropy H(D) is X, it means ~2^X times of guessing are expected to match a sample from D.*

$$H(S) = -\sum_{s \in S} \Pr_S(s) \cdot \log_2 \Pr_S(s)$$

*Initial uncertainty of the **secret value***

marginal prob. dist. of secret S

[Notations]

: Secret

: Uncertainty

: Observable

Measure of Information Leakage

- The amount of information about the secret (S) was leaked from the observable (O): $?(S) - ?(S \mid O)$
- The *Uncertainty* ? can be measured with **Shannon Entropy H.**
  - *If the distribution D's entropy H(D) is X, it means ~2^X times of guessing are expected to match a sample from D.*

$$H(S \mid O)$$

*Remaining uncertainty of the **secret value** after checking the **observable value***

$$= - \sum_{(s,o) \in S \times O} \Pr(s, o) \cdot \log_2 \frac{\Pr(s, o)}{\Pr_O(o)}$$

marginal prob. dist. of observable O

[Notations]

: Secret

: Uncertainty

: Observable

Measure of Information Leakage

Observable (O)

Secret (S)

			...	
	0.085	3E-04	...	0.025
	0.002	0.078	...	1E-04
...	...	...	...	...
	0.012	0.042	...	0.040

Joint Probability Distribution

Measure of Information Leakage

$$H(S)$$

*Initial uncertainty of the **secret value***

Observable (O)

Secret (S)

			...	
	0.085	3E-04	...	0.025
	0.002	0.078	...	1E-04
...	...	...	...	...
	0.012	0.042	...	0.040

Joint Probability Distribution

Measure of Information Leakage

$H(S\ )$

*Initial uncertainty of the **secret value***

Observable (O)

Secret (S)

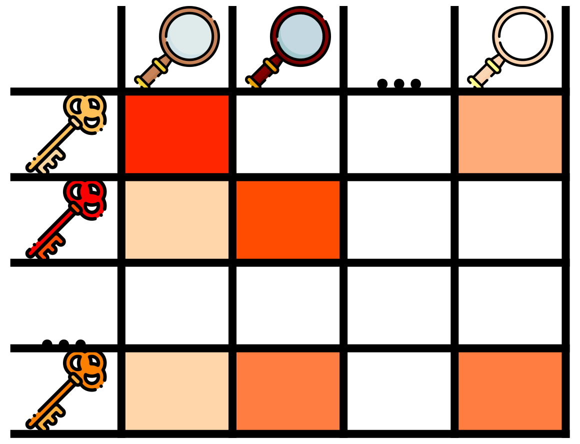
			...	
	0.085	3E-04	...	0.025
	0.002	0.078	...	1E-04
...	...	...	...	...
	0.012	0.042	...	0.040

Joint Probability Distribution

$H(\ \mid O\ )$

*Remaining uncertainty of the **secret value** after checking the **observable value***

Measure of Information Leakage

Mutual Information (MI) I measures the information leakage from  .

$$I(S\,;O\,) \quad = \quad H(S\,) - H(\;\mid O\,)$$

Mutual Information (MI)

*between the **secret** and the **observable***

*Initial uncertainty of the **secret value***

*Remaining uncertainty of the **secret value** after checking the **observable value***

Software

Obtaining Information Leakage Bounds via Approximate Model Counting*

SEEMANTA SAHA†, UC Santa Barbara, USA
SURENDRA GHENTIYALA†, UC Santa Barbara, USA
SHIHUA LU, UC Santa Barbara, USA
LUCAS BANG, Harvey Mudd College, USA
TEVFIK BULTAN, UC Santa Barbara, USA

Information leaks are a significant problem in modern software systems. In recent years, information theoretic concepts, such as Shannon entropy, have been applied to quantifying information leaks in programs. One recent approach is to use symbolic execution together with model counting constraints solvers in order to quantify information leakage. There are at least two reasons for unsoundness in quantifying information leakage using this approach: 1) Symbolic execution may not be able to explore all execution paths, 2) Model counting constraints solvers may not be able to provide an exact count. We present a sound symbolic quantitative information flow analysis that bounds the information leakage both for the cases where the program behavior is not fully explored and the model counting constraint solver is unable to provide a precise model count but provides an upper and a lower bound. We implemented our approach as an extension to KLEE for computing sound bounds for information leakage in C programs.

CCS Concepts: • **Software and its engineering → Formal software verification**; **General programming languages**.

Additional Key Words and Phrases: Quantitative Program Analysis, Symbolic Quantitative Information Flow Analysis, Model Counting, Information Leakage, Optimization

ACM Reference Format:
Seemanta Saha, Surendra Ghentiyala, Shihua Lu, Lucas Bang, and Tevfik Bultan. 2023. Obtaining Information Leakage Bounds via Approximate Model Counting. *Proc. ACM Program. Lang.* 7, PLDI, Article 167 (June 2023), 22 pages. https://doi.org/10.1145/3591281

1 INTRODUCTION

One of the most critical security issues in software systems today is protecting users' private information, which makes analyzing information leakage in software systems a timely and important research problem. A classic approach to address this problem is enforcing *noninterference* which ensures that publicly observable properties of program execution (such as public outputs or side-channels) are independent of secret input values. But, enforcing noninterference is often not possible as software systems need to reveal some amount of information that depends on secret inputs. Consider a password checker where, as public output, the system needs to provide

*This material is based on research supported by NSF under Grants CCF-2008660, CCF-1901098, CCF-1817242.
†These authors have equal contribution to this paper.

Authors' addresses: Seemanta Saha, UC Santa Barbara, USA, seemantasaha@ucsb.edu; Surendra Ghentiyala, UC Santa Barbara, USA, sg974@cornell.edu; Shihua Lu, UC Santa Barbara, USA, shihualu@ucsb.edu; Lucas Bang, Harvey Mudd College, USA, lbang@g.hmc.edu; Tevfik Bultan, UC Santa Barbara, USA, bultan@ucsb.edu.

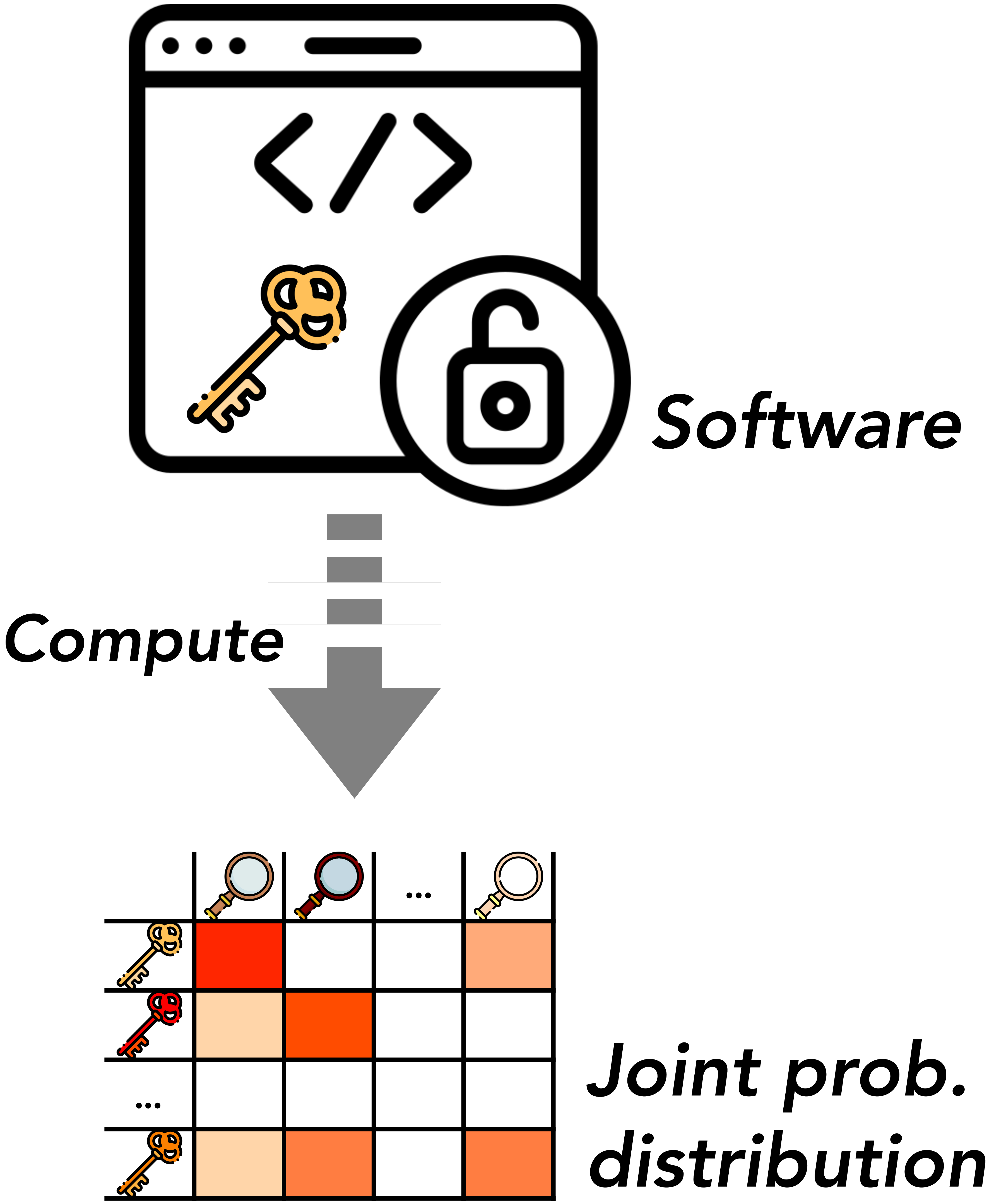
167

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.
© 2023 Copyright held by the owner/author(s).
2475-1421/2023/6-ART167
https://doi.org/10.1145/3591281

Proc. ACM Program. Lang., Vol. 7, No. PLDI, Article 167. Publication date: June 2023.

Analytic approach

Uses model counting



Obtaining Information Leakage Bounds via Approximate Model Counting*

SEEMANTA SAHA†, UC Santa Barbara, USA
SURENDRA GHENTIYALA†, UC Santa Barbara, USA
SHIHUA LU, UC Santa Barbara, USA
LUCAS BANG, Harvey Mudd College, USA
TEVFIK BULTAN, UC Santa Barbara, USA

Information leaks are a significant problem in modern software systems. In recent years, information theoretic concepts, such as Shannon entropy, have been applied to quantifying information leaks in programs. One recent approach is to use symbolic execution together with model counting constraints solvers in order to quantify information leakage. There are at least two reasons for unsoundness in quantifying information leakage using this approach: 1) Symbolic execution may not be able to explore all execution paths, 2) Model counting constraints solvers may not be able to provide an exact count. We present a sound symbolic quantitative information flow analysis that bounds the information leakage both for the cases where the program behavior is not fully explored and the model counting constraint solver is unable to provide a precise model count but provides an upper and a lower bound. We implemented our approach as an extension to KLEE for computing sound bounds for information leakage in C programs.

CCS Concepts: • **Software and its engineering** → **Formal software verification**; **General programming languages**.

Additional Key Words and Phrases: Quantitative Program Analysis, Symbolic Quantitative Information Flow Analysis, Model Counting, Information Leakage, Optimization

ACM Reference Format:
Seemanta Saha, Surendra Ghentiyala, Shihua Lu, Lucas Bang, and Tevfik Bultan. 2023. Obtaining Information Leakage Bounds via Approximate Model Counting. *Proc. ACM Program. Lang.* 7, PLDI, Article 167 (June 2023), 22 pages. https://doi.org/10.1145/3591281

1 INTRODUCTION

One of the most critical security issues in software systems today is protecting users' private information, which makes analyzing information leakage in software systems a timely and important research problem. A classic approach to address this problem is enforcing *noninterference* which ensures that publicly observable properties of program execution (such as public outputs or side-channels) are independent of secret input values. But, enforcing noninterference is often not possible as software systems need to reveal some amount of information that depends on secret inputs. Consider a password checker where, as public output, the system needs to provide

*This material is based on research supported by NSF under Grants CCF-2008660, CCF-1901098, CCF-1817242.
†These authors have equal contribution to this paper.

Authors' addresses: Seemanta Saha, UC Santa Barbara, USA, seemantasaha@ucsb.edu; Surendra Ghentiyala, UC Santa Barbara, USA, sg974@cornell.edu; Shihua Lu, UC Santa Barbara, USA, shihualu@ucsb.edu; Lucas Bang, Harvey Mudd College, USA, lbang@g.hmc.edu; Tevfik Bultan, UC Santa Barbara, USA, bultan@ucsb.edu.

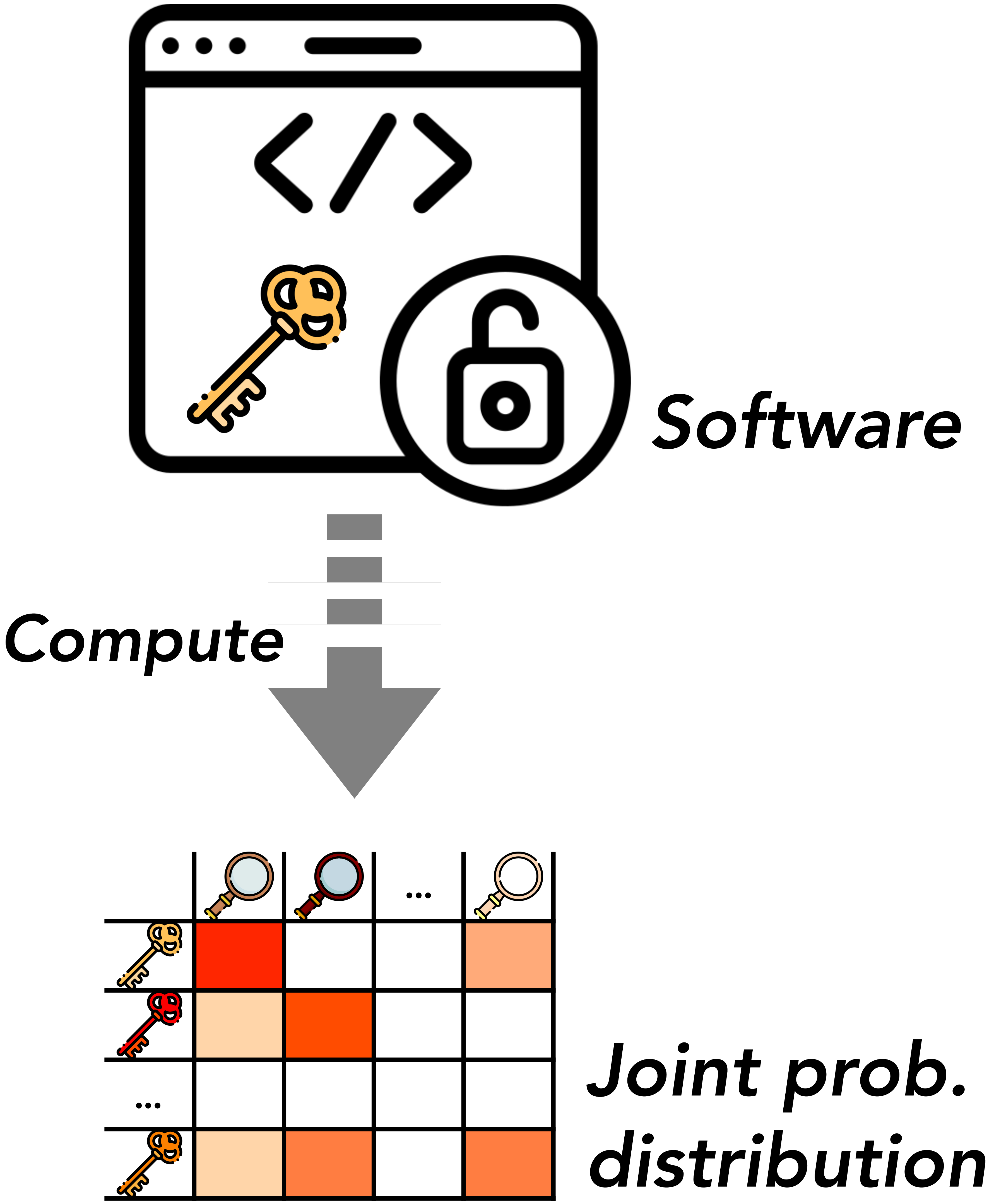
167

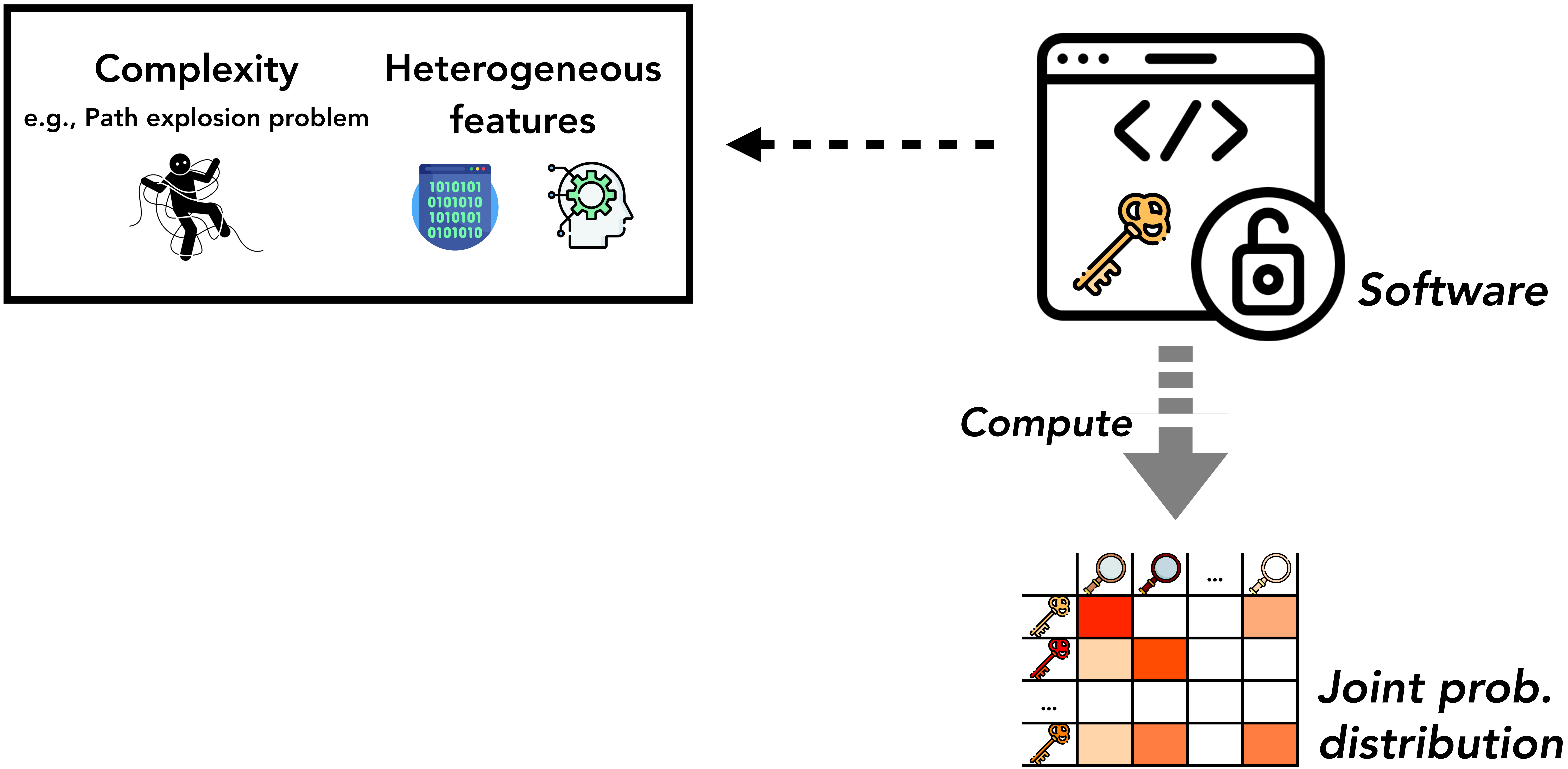
2475-1421/2023/6-ART167
https://doi.org/10.1145/3591281

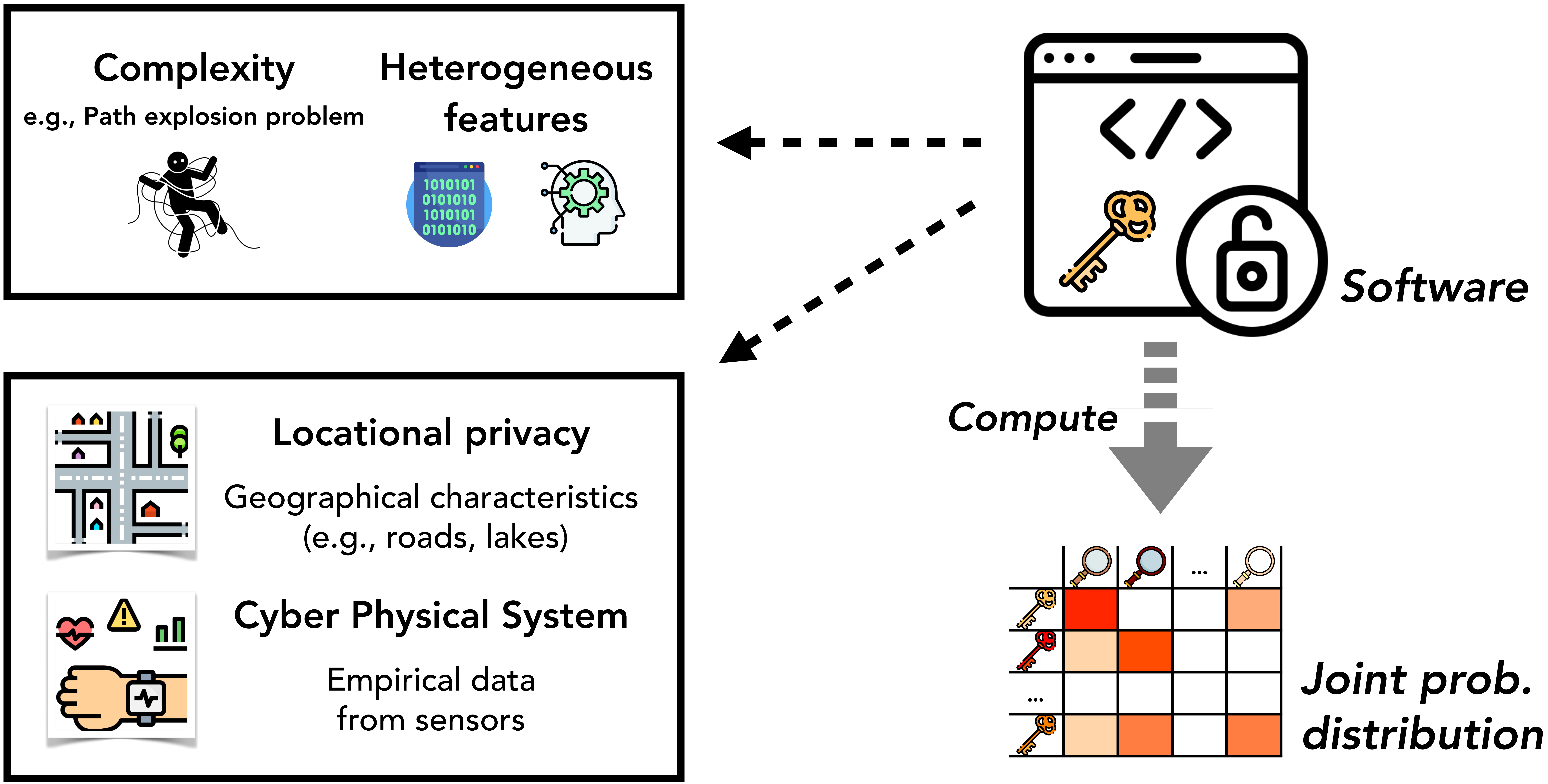
Analytic approach

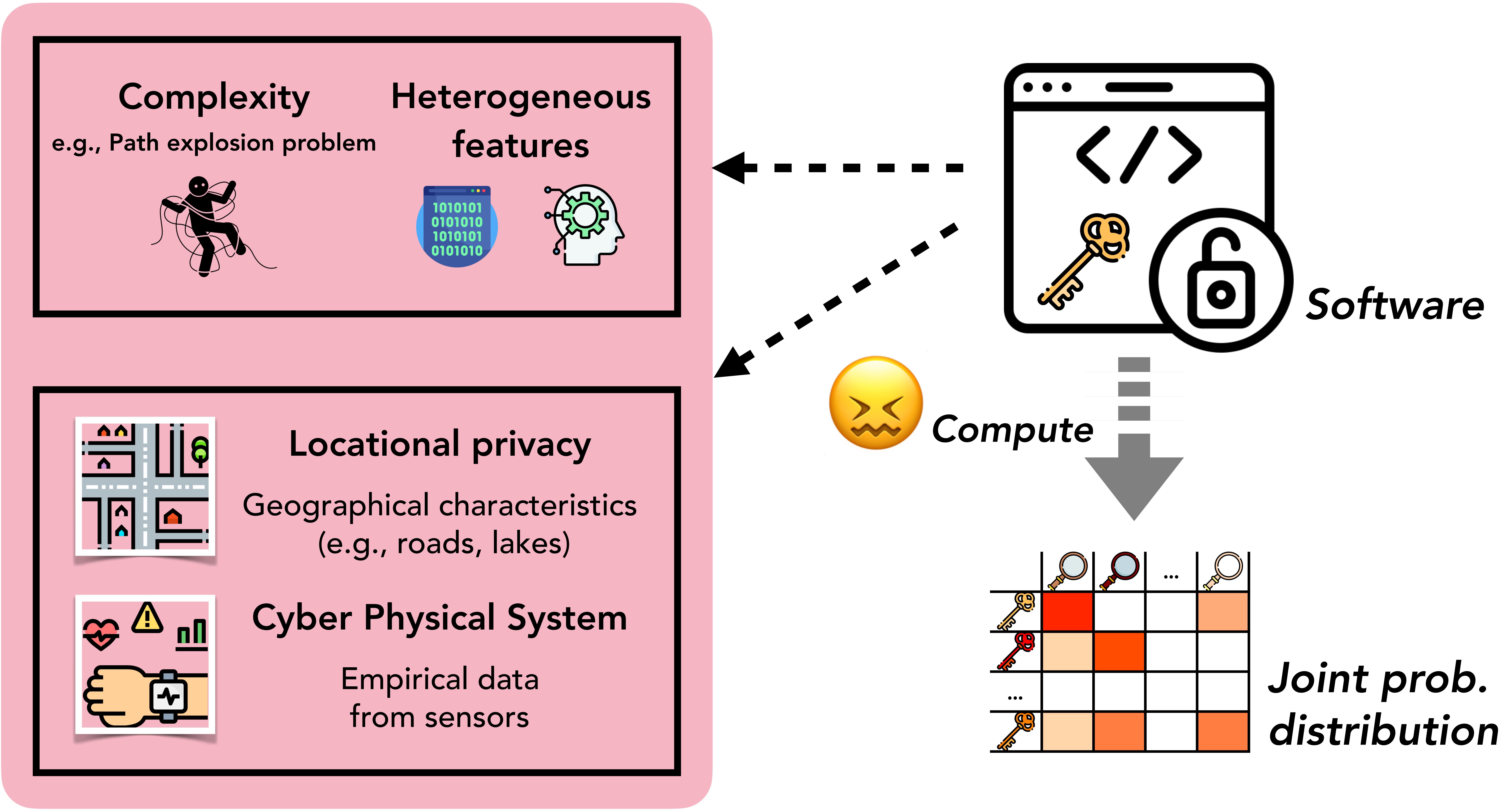
Uses model counting

*An **analytic approach** provides*
*a **precise result** or*
*a **formal guarantee!***



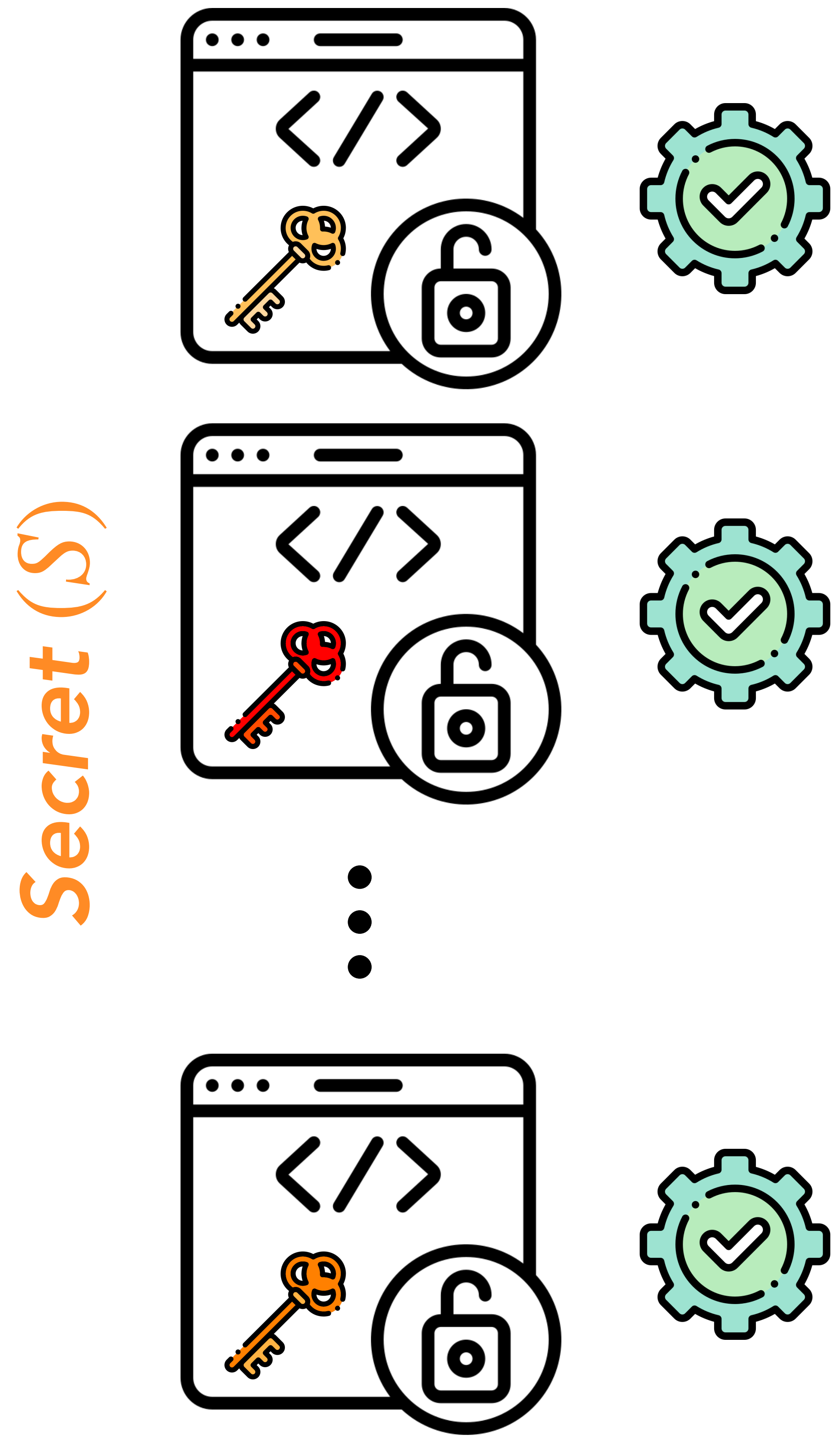

Complexity
e.g., Path explosion problem
Heterogeneous features
1010101
0101010
1010101
0101010
Software
Compute
Joint prob. distribution


Complexity
e.g., Path explosion problem
Heterogeneous
features
1010101
0101010
1010101
0101010
Locational privacy
Geographical characteristics
(e.g., roads, lakes)
Cyber Physical System
Empirical data
from sensors
Software
Compute
...
Joint prob.
distribution

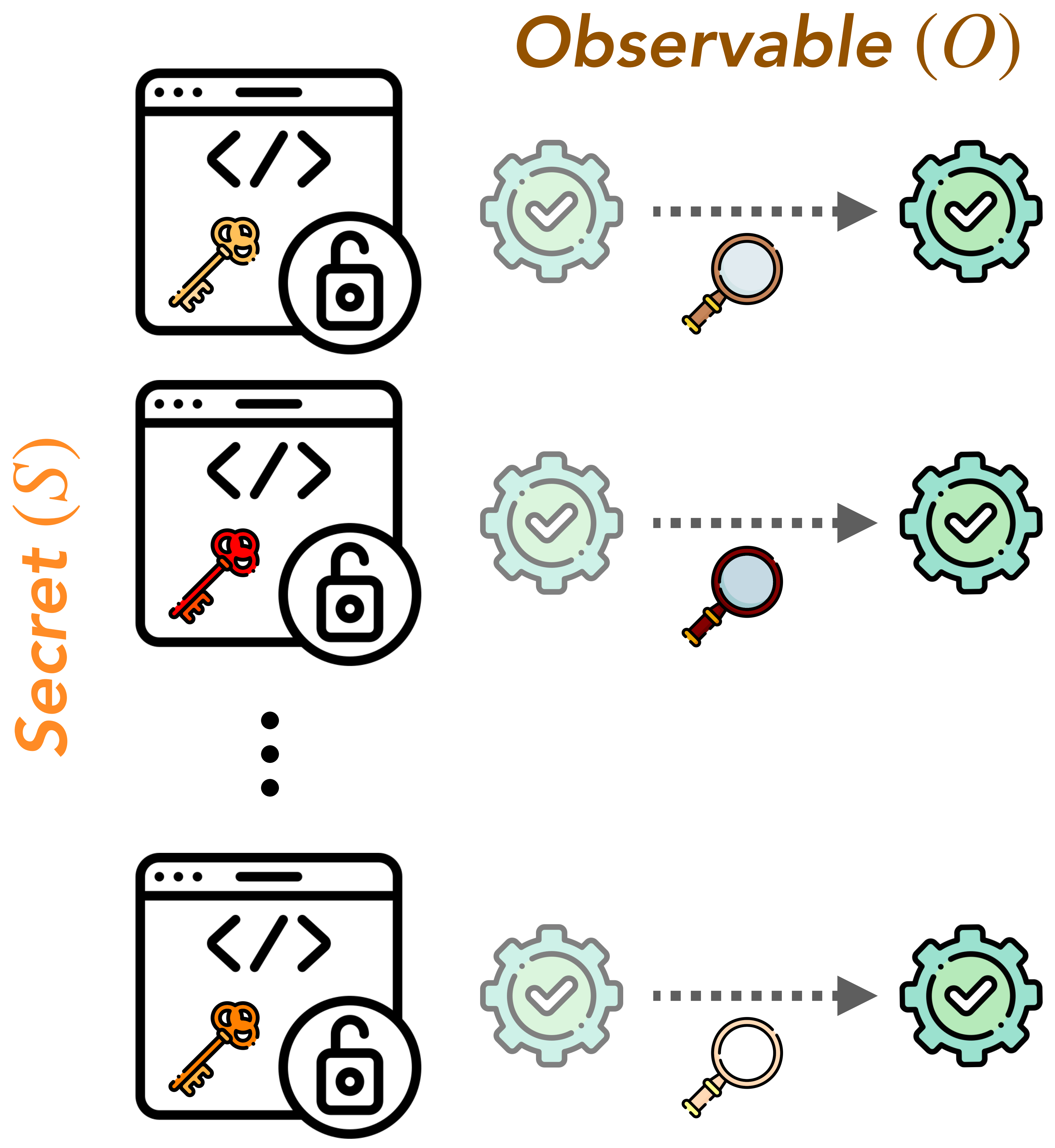


Complexity
e.g., Path explosion problem
Heterogeneous features
1010101
0101010
1010101
0101010
Locational privacy
Geographical characteristics (e.g., roads, lakes)
Cyber Physical System
Empirical data from sensors
Software
Compute
...
Joint prob. distribution

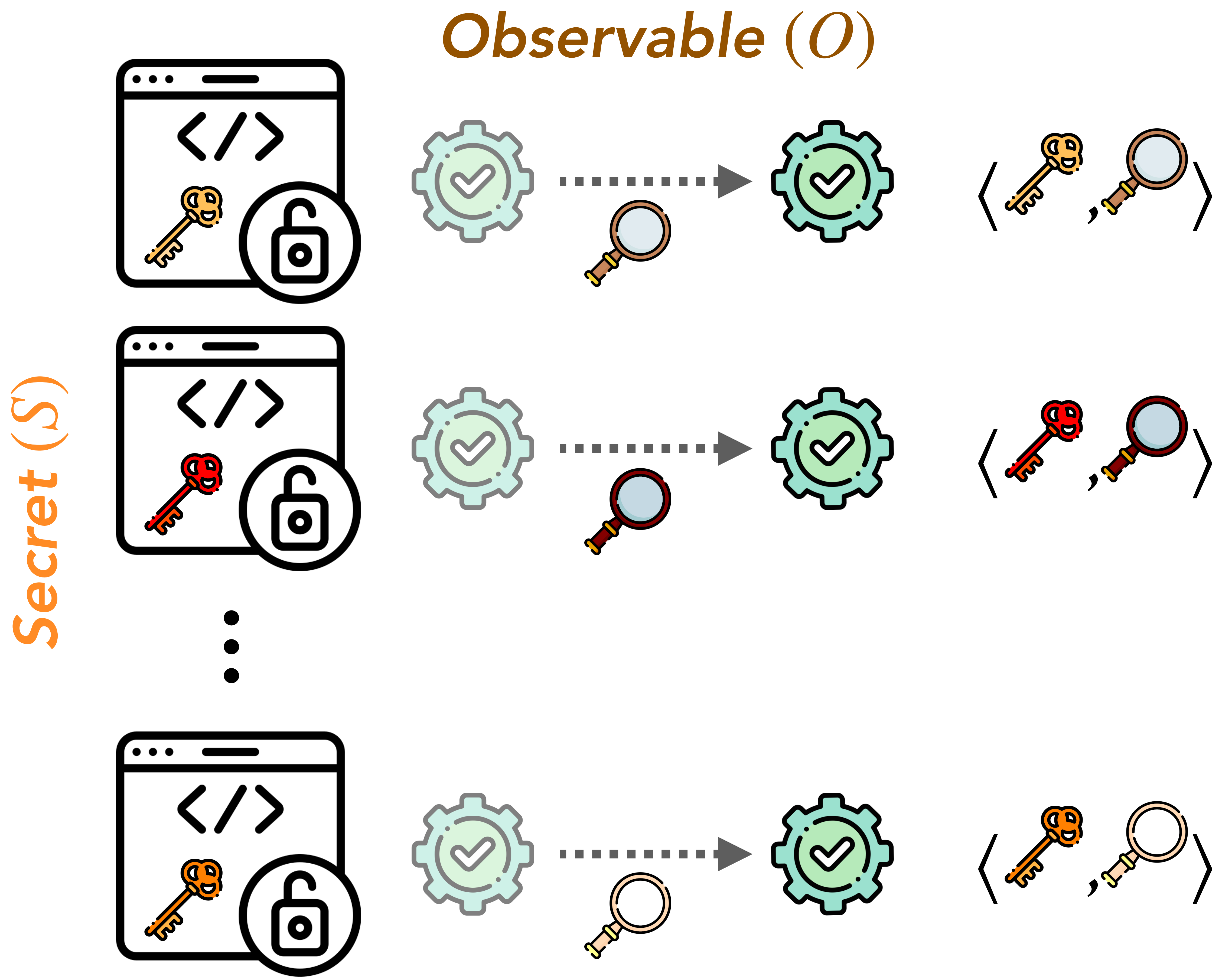
(Existing) Empirical Information Leakage Analysis



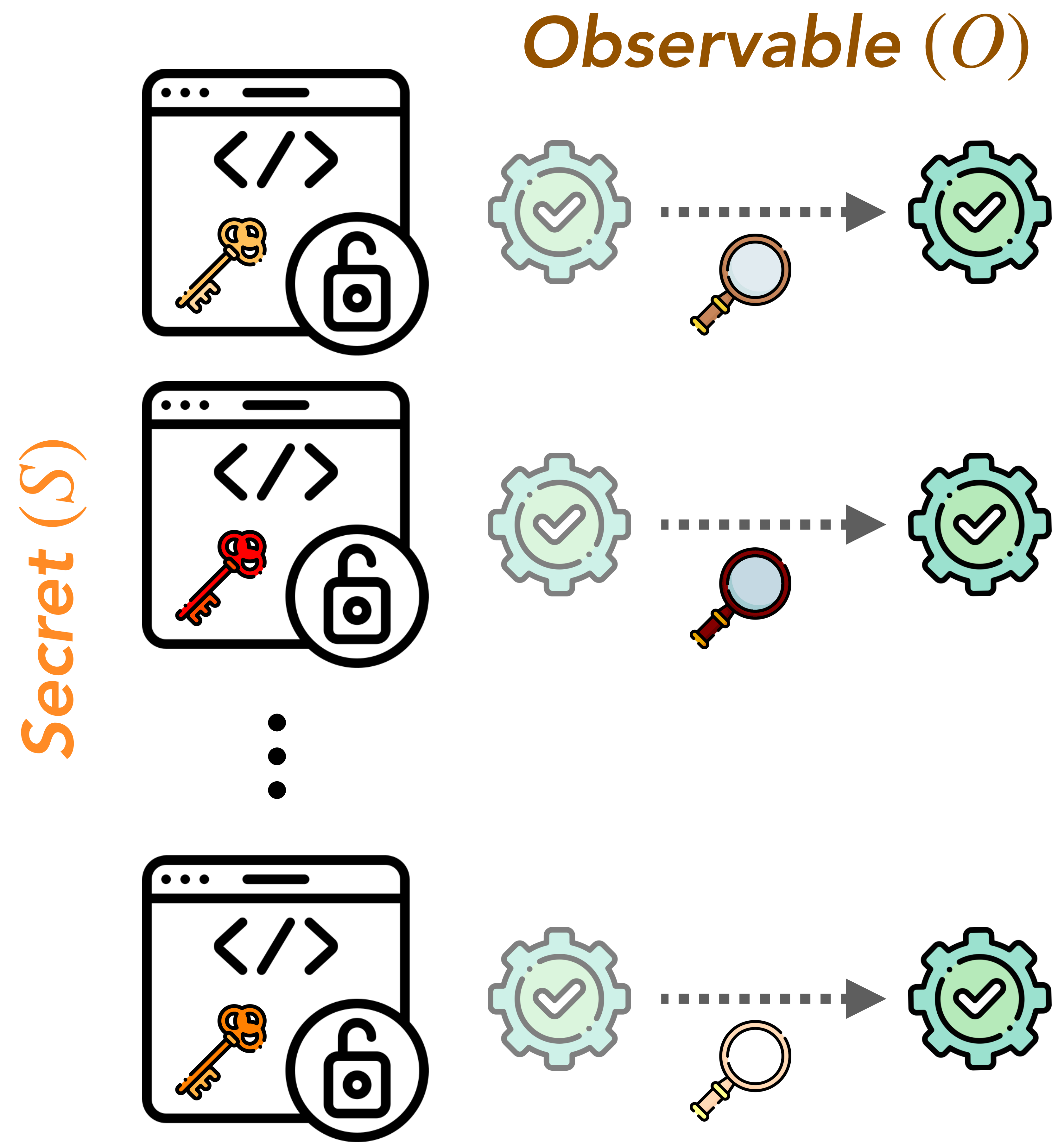
(Existing) Empirical Information Leakage Analysis



(Existing) Empirical Information Leakage Analysis



(Existing) Empirical Information Leakage Analysis



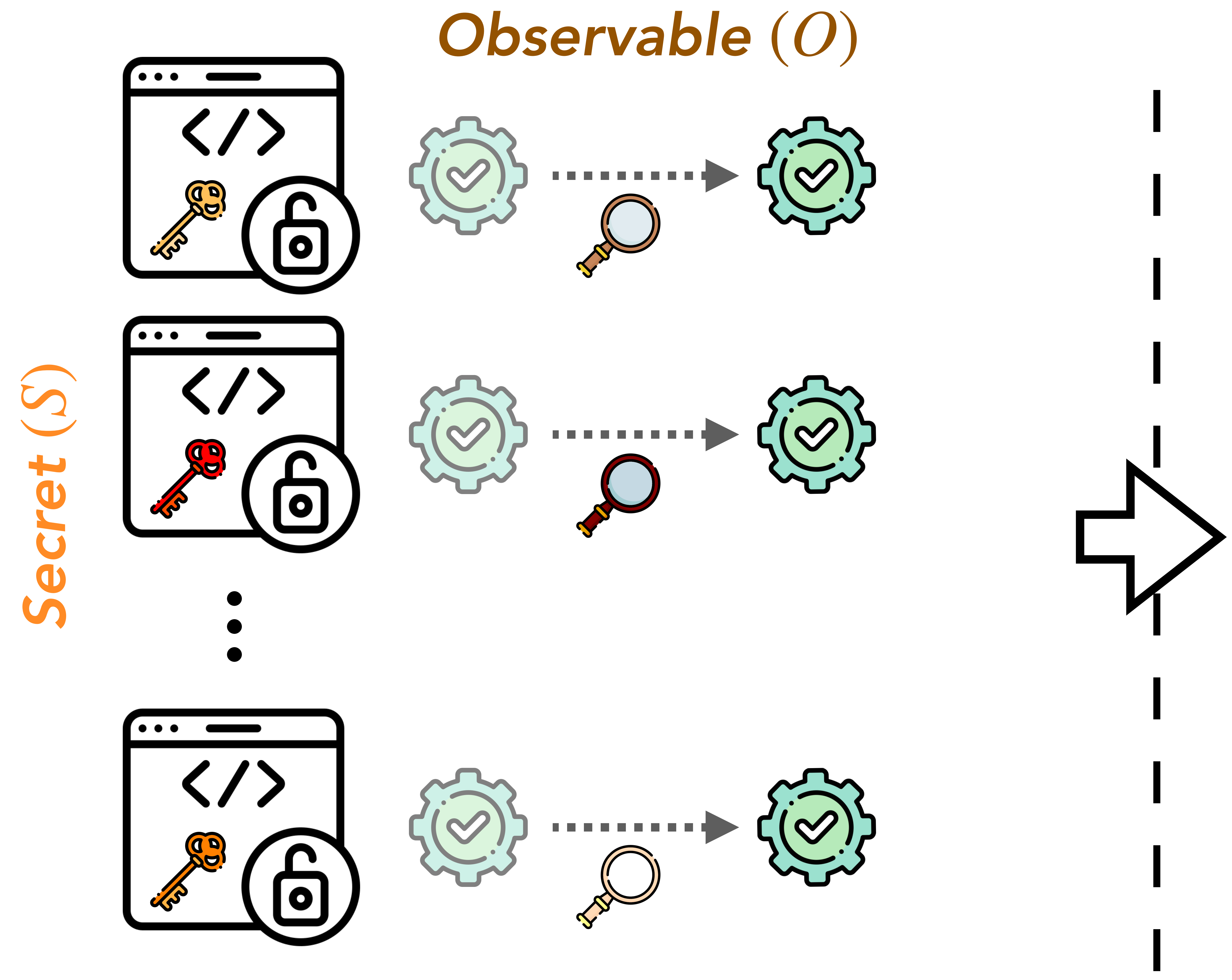
Observable (O)

Secret (S)

			...	
	100	0	...	30
	2	78	...	0
...	...	...	...	...
	13	50	...	50

***Sample* Joint Frequency Table**

(Existing) Empirical Information Leakage Analysis



Observable (O)

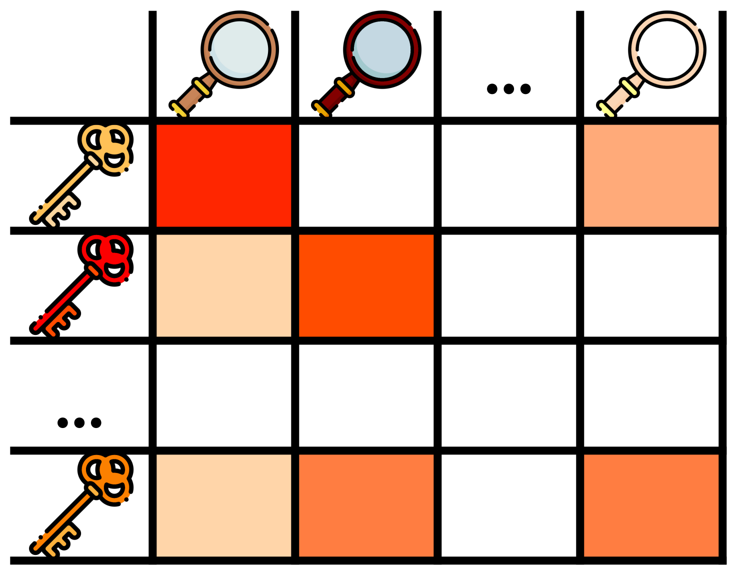
Secret (S)

	0.1	0.0	...	0.03
	0.002	0.078	...	0.0
...	...	...	...	...
	0.013	0.05	...	0.05

Empirical
Joint
Probability
Distribution

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)

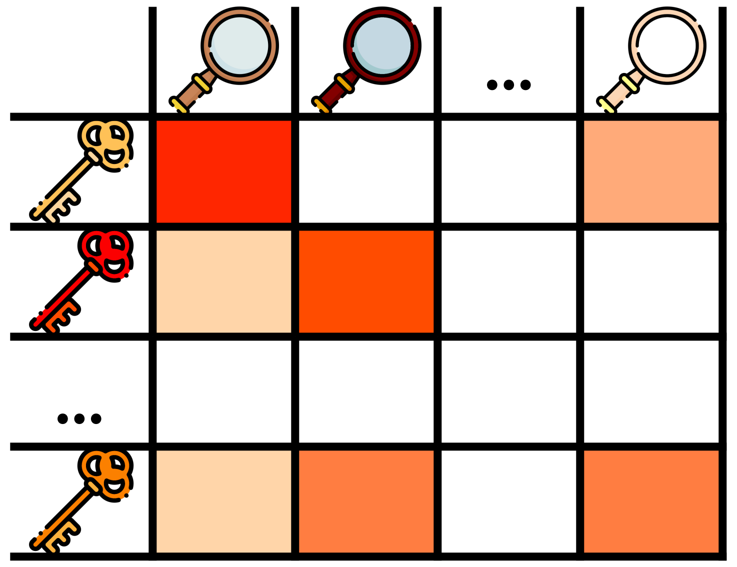


Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

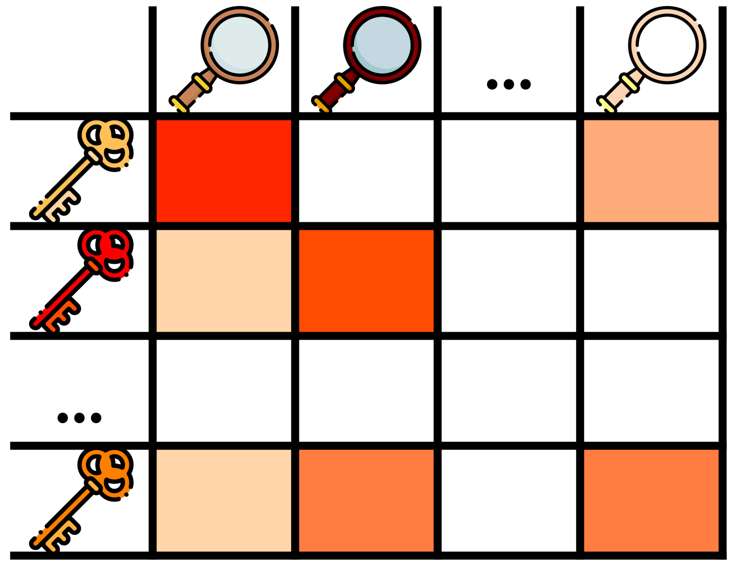
🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute ⇩

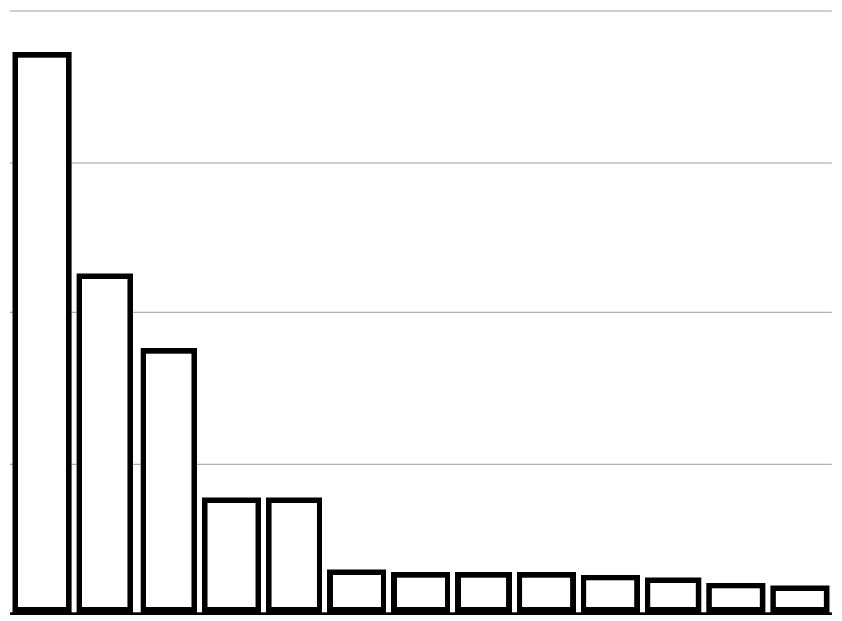
$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

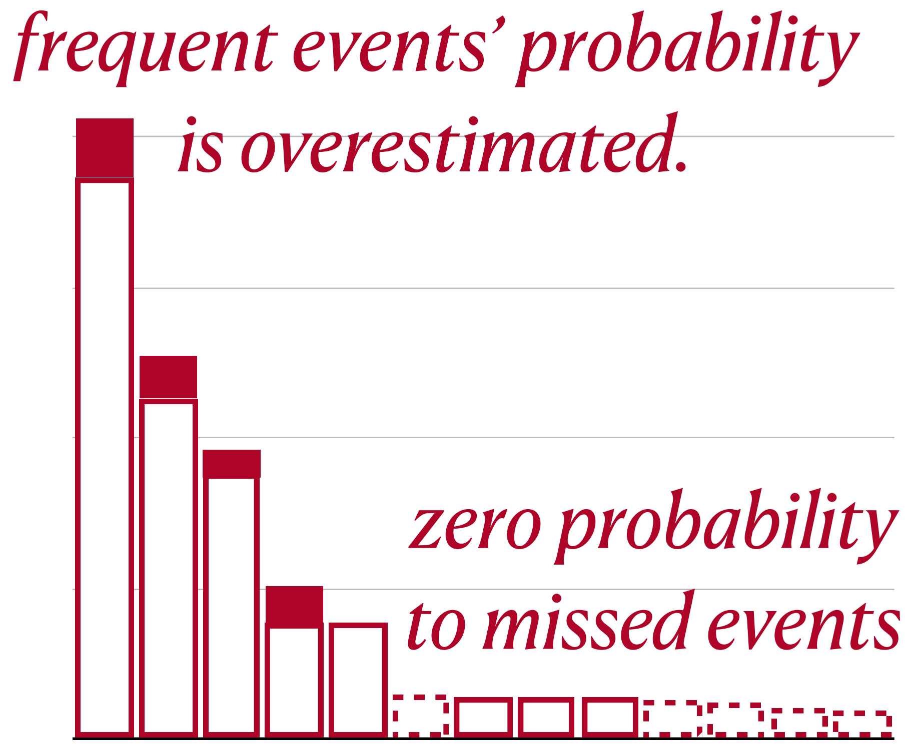
Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

Due to ***missing events*** *⟨🗝️, 🔍⟩ in the sample,*



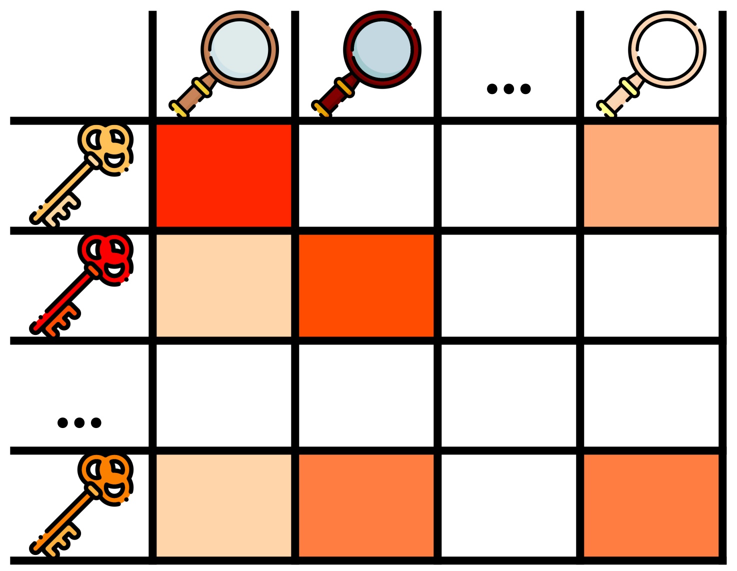
True Distribution



Empirical Dist.

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

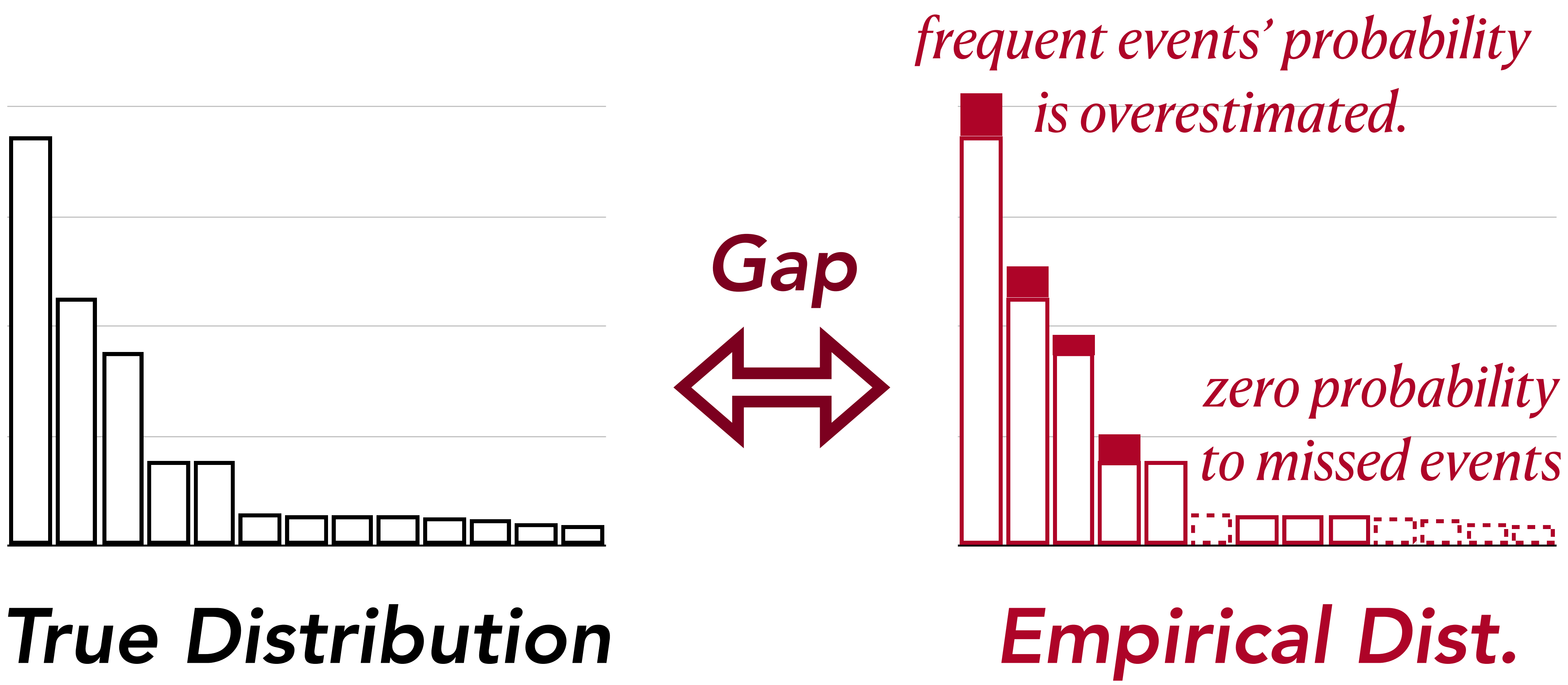
$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

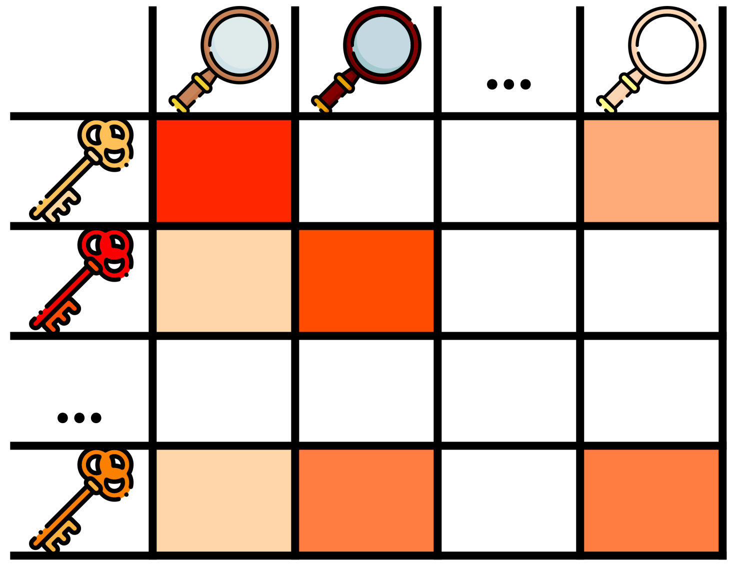
*Due to **missing events** ⟨🗝️, 🔍⟩ in the sample,*



$$I \ll \hat{I}_{emp}$$

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute ⇩

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

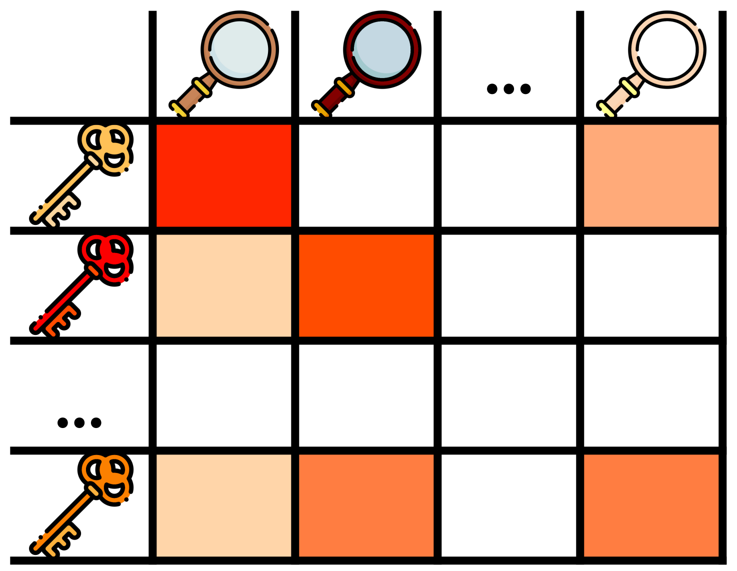
# of unique sec. in the sample

# of unique obs. in the sample

$$\hat{I}_{miller} = \hat{I}_{emp} \underbrace{- \frac{(m_S - 1)(m_O - 1)}{2n}}_{\text{Bias correction term}}$$

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

# of unique sec. in the sample   *# of unique obs. in the sample*

$$\hat{I}_{miller} = \hat{I}_{emp} - \boxed{\frac{(m_S - 1)(m_O - 1)}{2n}}$$

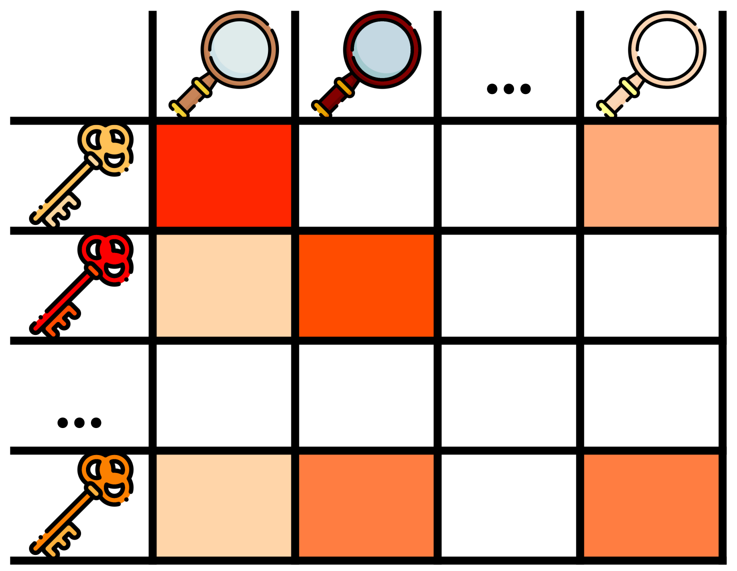
Bias correction term

*However, if the **space** of observables **is too large**, e.g.,*

memory access   time   program output

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

# of unique sec. in the sample   *# of unique obs. in the sample*

$$\hat{I}_{miller} = \hat{I}_{emp} - \boxed{\frac{(m_S - 1)(m_O - 1)}{2n}}$$

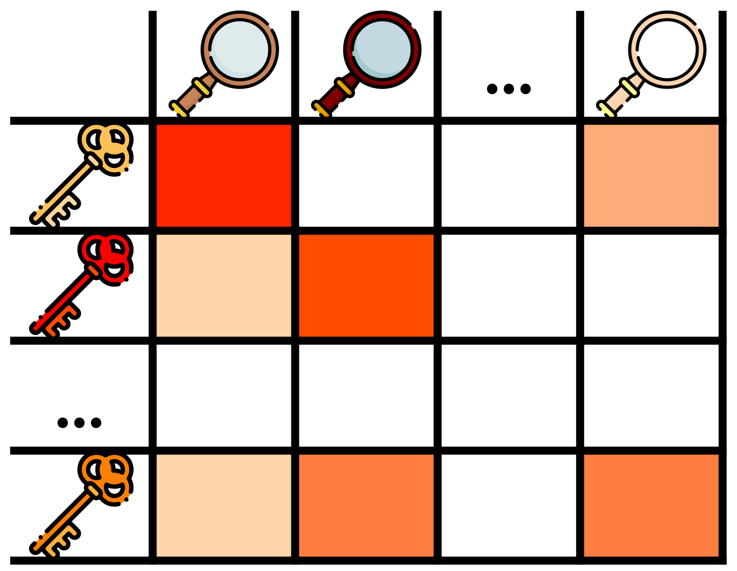
Bias correction term

*However, if the **space** of observables **is too large**, e.g.,*

*there may be too many **rare events** ⟨ , ⟩ in the sample.*

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute ⇩

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

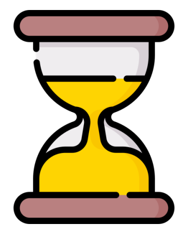
# of unique sec. in the sample   *# of unique obs. in the sample*

$$\hat{I}_{miller} = \hat{I}_{emp} - \boxed{\frac{(m_S - 1)(m_O - 1)}{2n}}$$

Bias correction term

*However, if the **space** of observables **is too large**, e.g.,*

memory access



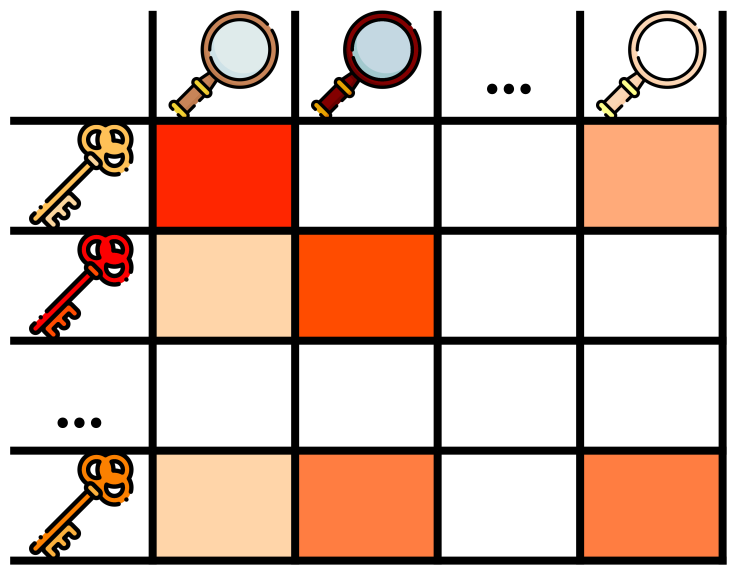
time

program output

*there may be too many **rare events** ⟨🔑, 🔍⟩ in the sample.*

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

🚨 Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

# of unique sec. in the sample *# of unique obs. in the sample*

$$\hat{I}_{miller} = \hat{I}_{emp} - \boxed{\frac{(m_S - 1)(m_O - 1)}{2n}}$$

underestimate ***Bias correction term***

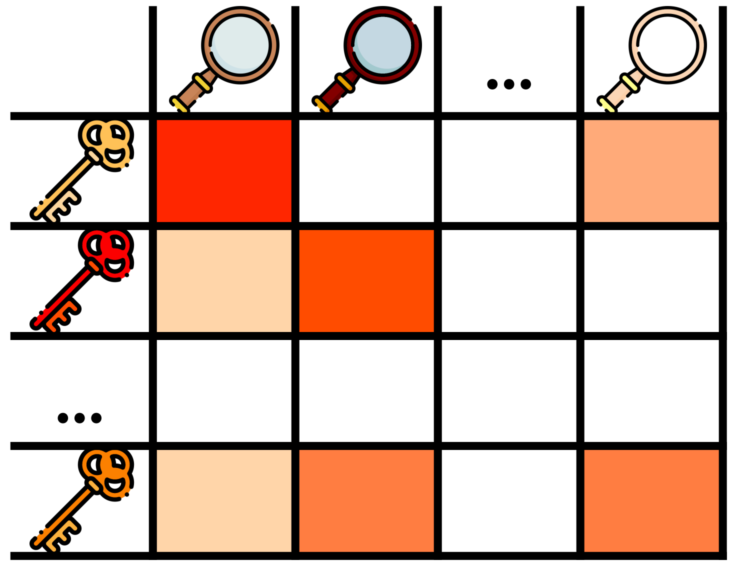
*However, if the **space** of observables **is too large**, e.g.,*

memory access time program output

*there may be too many **rare events** ⟨ , ⟩ in the sample.*

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Directly compute

$$\hat{I}_{emp} = \hat{H}_{emp}(S) - \hat{H}_{emp}(S \mid O)$$

Problem

Accuracy

It significantly ***overestimates MI*** if there are ***missing events***.

2. Miller MI Estimator (**Miller**)

The state-of-the-art estimator

# of unique sec. in the sample

# of unique obs. in the sample

$$\hat{I}_{miller} = \hat{I}_{emp} \boxed{- \frac{(m_S - 1)(m_O - 1)}{2n}}$$

underestimate

Bias correction term

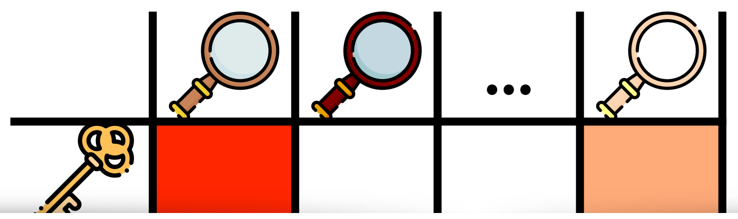
Problem

Safety

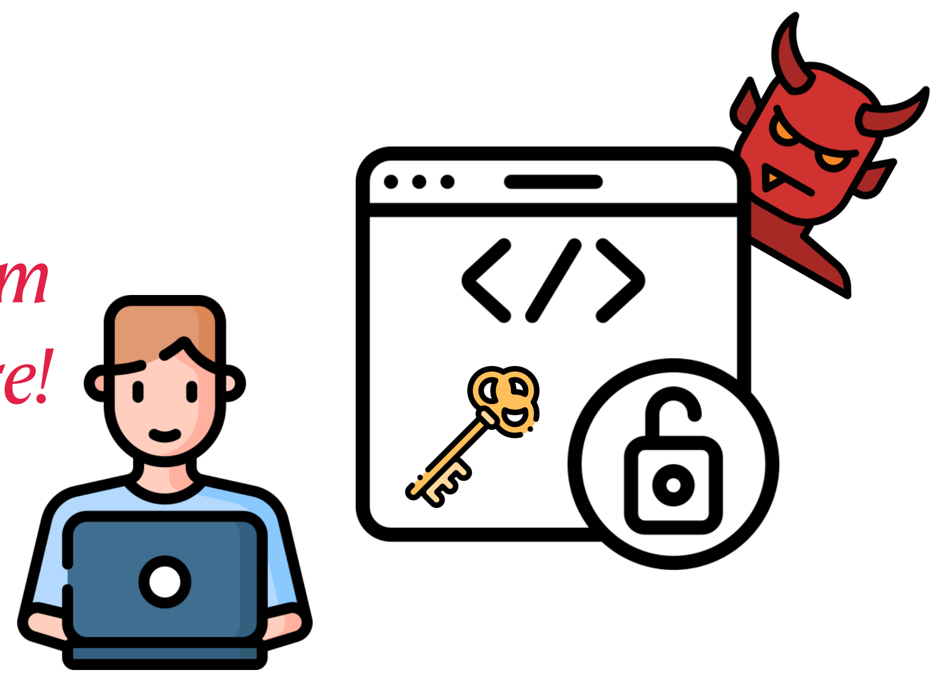
It ***underestimates MI*** if there are ***rare events*** *in the sample*.

(Existing) Empirical Information Leakage Analysis

1. Empirical MI Estimator (**Empirical**)



Underestimating the information leakage is especially ***harmful,*** since it leads to ***overconfidence in the privacy*** of the vulnerable software.



2. Miller MI Estimator (**Miller**)

<u>The state-of-the-art estimator</u>

\# *of unique sec. in the sample*   \# *of unique obs. in the sample*

$$\downarrow \hat{I}_{miller} = \hat{I}_{emp} - \boxed{\frac{(m_S - 1)(m_O - 1)}{2n}}$$

underestimate   ***Bias correction term***

🚨 Problem

Safety

It ***underestimates MI*** if there are ***rare events*** *in the sample.*

Research **Aim**

Empirical estimator

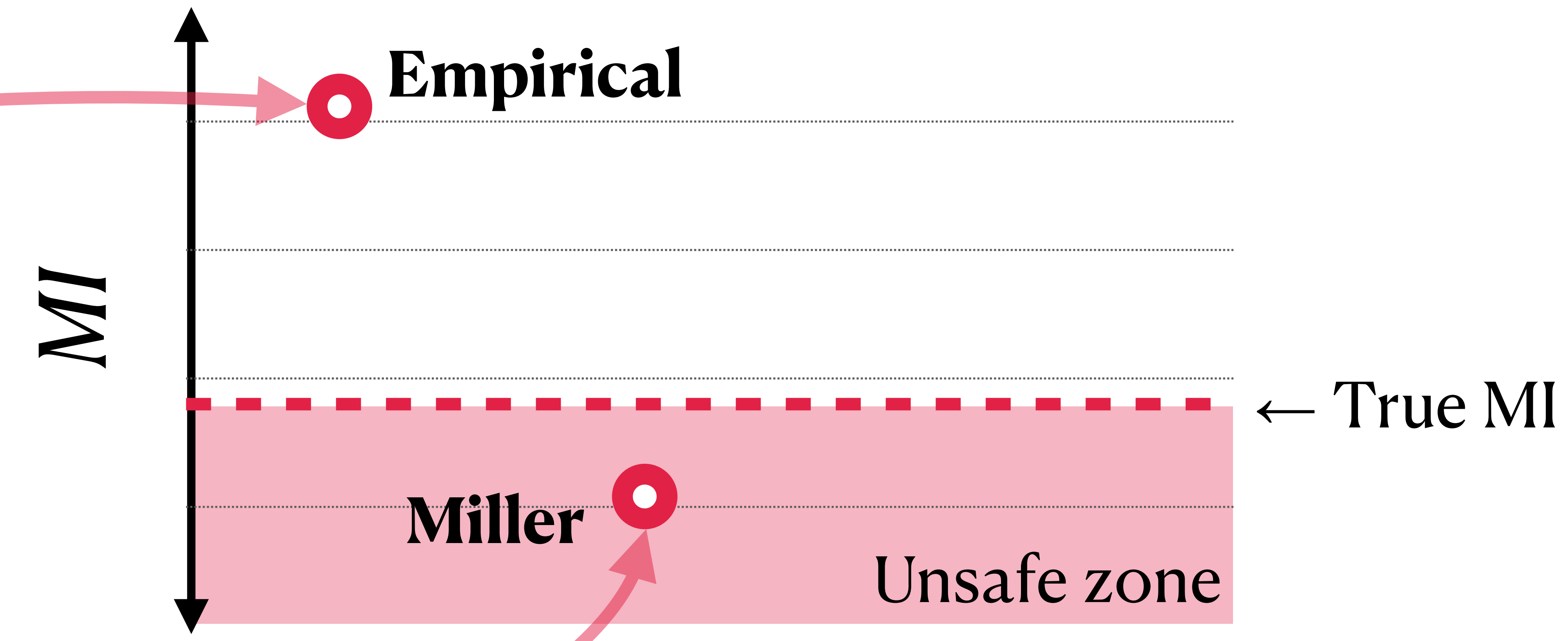
$$\hat{I}_{emp}(S;O) = \hat{H}_{emp}(X) - \hat{H}_{emp}(X \mid Y)$$

Inaccurate

Miller estimator

$$\hat{I}_{miller} = \hat{I}_{emp} - \frac{(m_S - 1)(m_O - 1)}{2n}$$

Unsafe w/ small samples



Research Aim

Empirical estimator

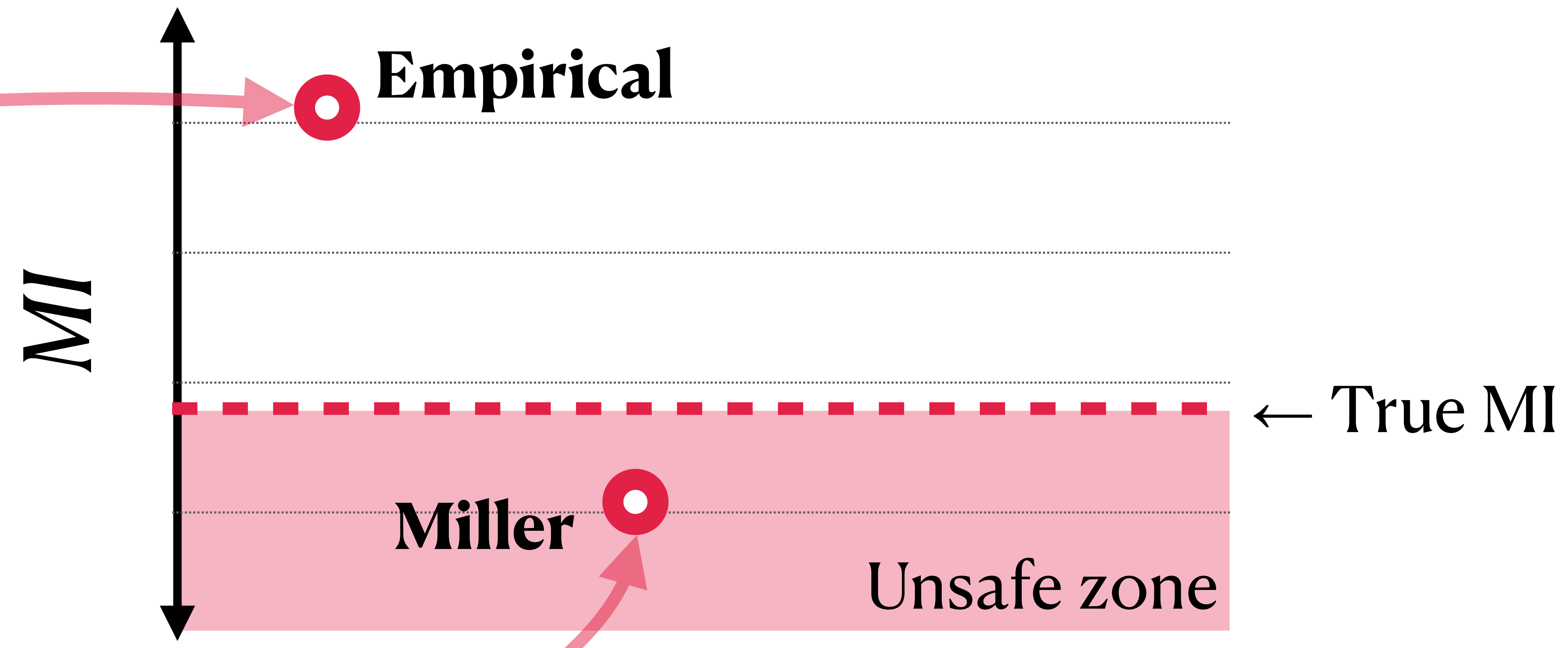
$$\hat{I}_{emp}(S;O) = \hat{H}_{emp}(X) - \hat{H}_{emp}(X \mid Y)$$

Inaccurate

Miller estimator

$$\hat{I}_{miller} = \hat{I}_{emp} - \frac{(m_S - 1)(m_O - 1)}{2n}$$

Unsafe w/ small samples



Existing estimators either produce ***inaccurate or unsafe*** *estimates due to mishandling* ***missing or rare events*****.**

Research Aim

Empirical estimator

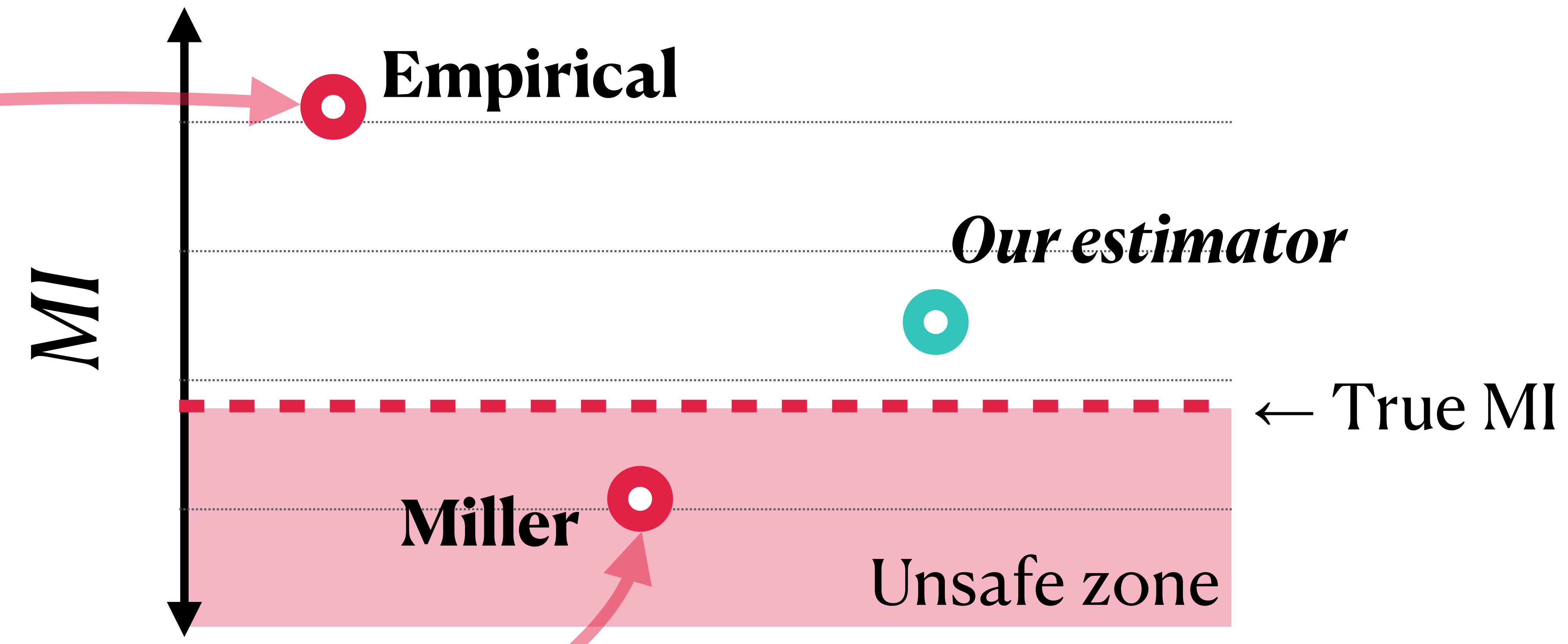
$$\hat{I}_{emp}(S;O) = \hat{H}_{emp}(X) - \hat{H}_{emp}(X \mid Y)$$

Inaccurate

Miller estimator

$$\hat{I}_{miller} = \hat{I}_{emp} - \frac{(m_S - 1)(m_O - 1)}{2n}$$

Unsafe w/ small samples



We developed an estimator that ***accurately*** *and* ***safely*** *estimates the leakage in the presence of* ***missing or rare events****.*

Chao's Multinomial Distribution (MD) Estimation

A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

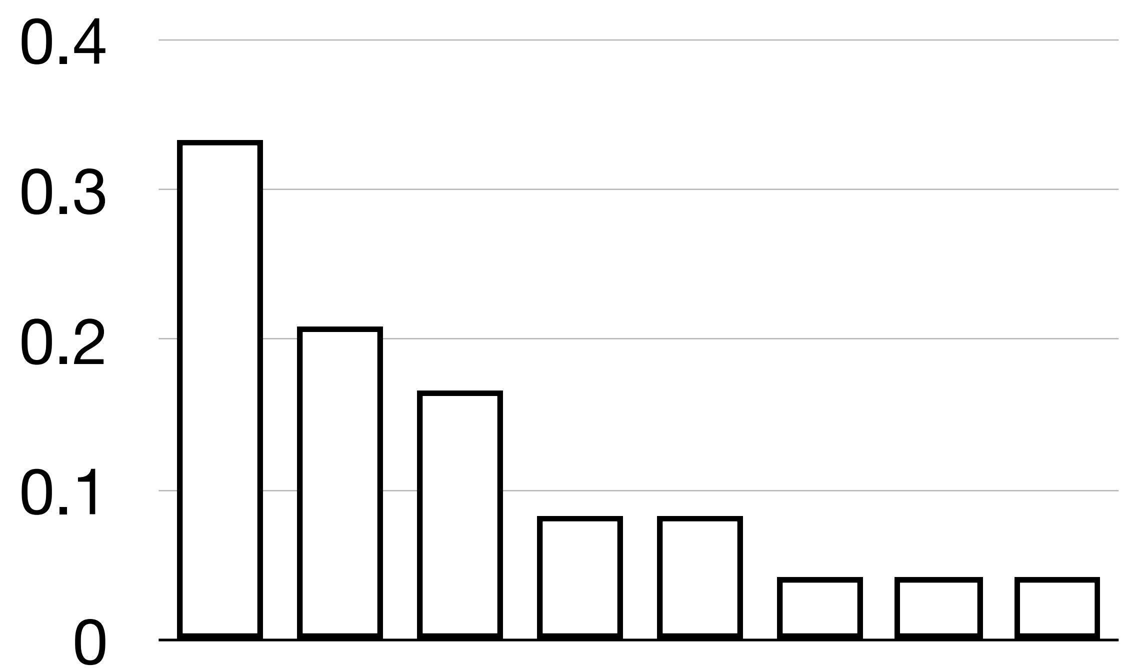
Chao's Multinomial Distribution (MD) Estimation

- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.

A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

Chao's Multinomial Distribution (MD) Estimation

- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.



Empirical Distribution

A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

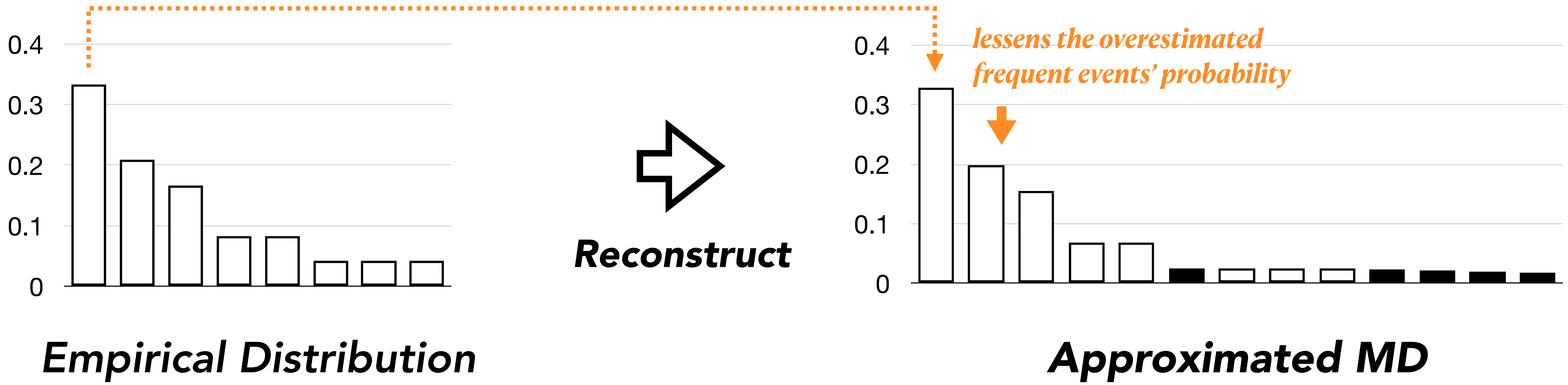
Chao's Multinomial Distribution (MD) Estimation

- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.

A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

Chao's Multinomial Distribution (MD) Estimation

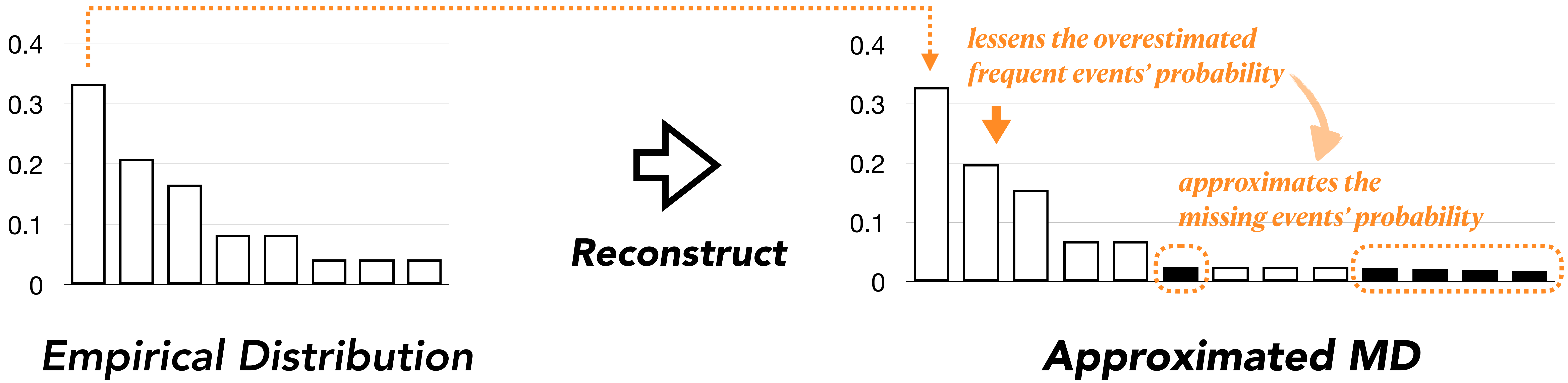
- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.



A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

Chao's Multinomial Distribution (MD) Estimation

- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.

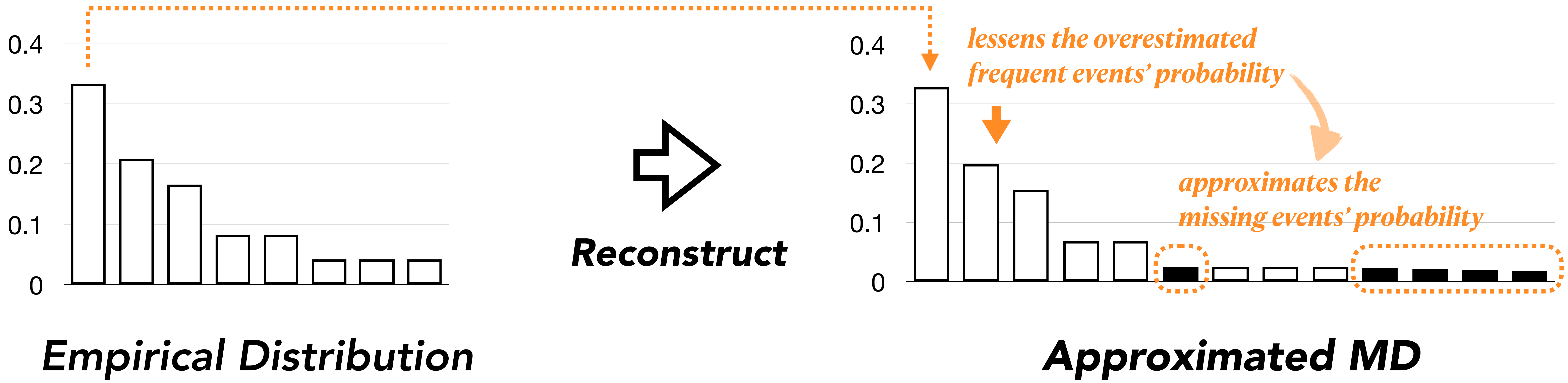


A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

Chao's Multinomial Distribution (MD) Estimation

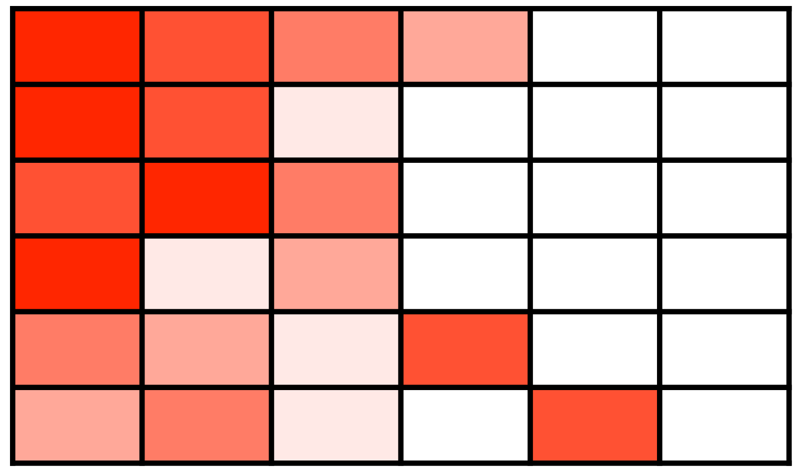
- Given samples from the unknown multinomial distribution (MD), it **reconstructs the underlying MD** by approximation.

➡ *Handle the missing/rare events problem* 😀



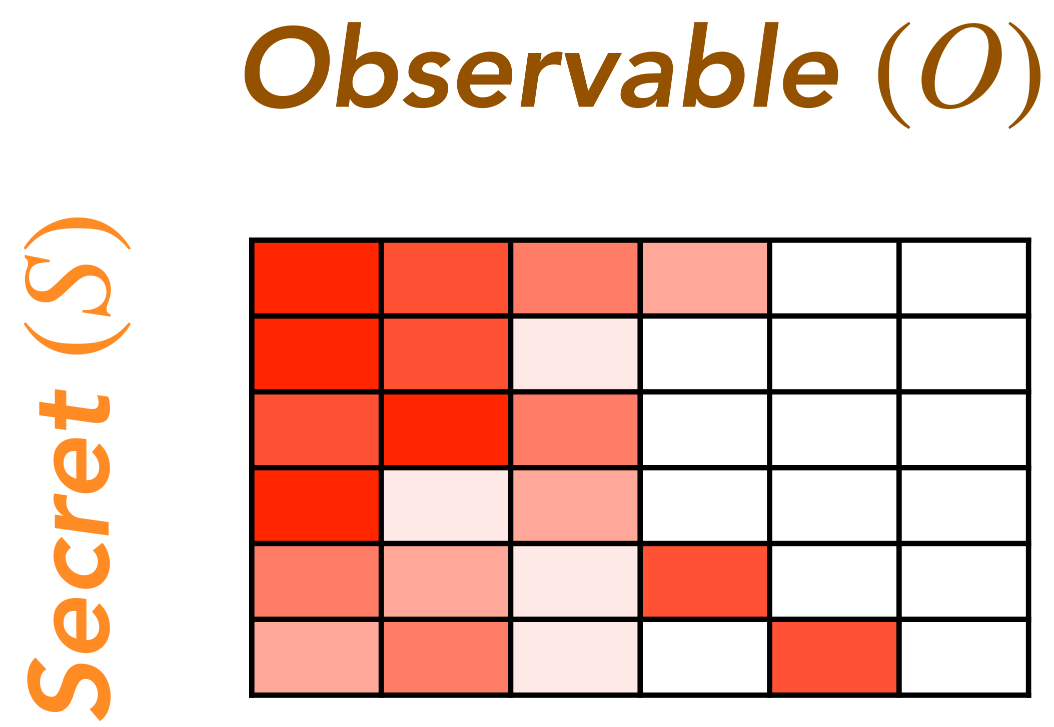
A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–201, 2015.

Challenge of apply MD estimation for MI estimation



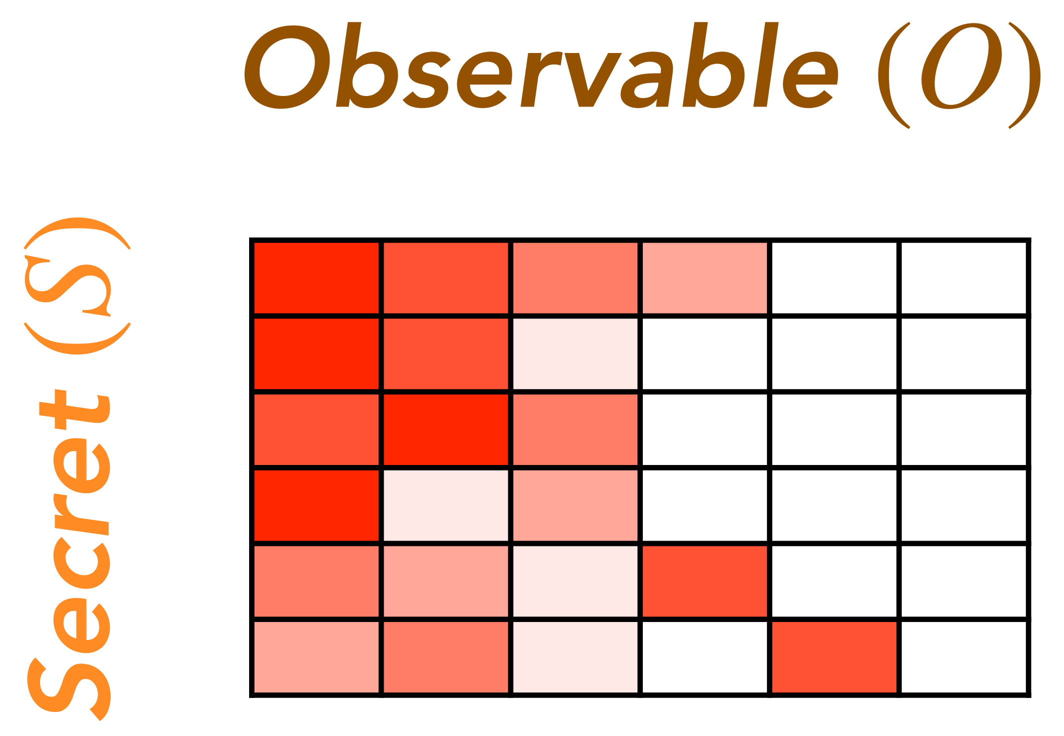
Empirical Joint
Probability Dist.

Challenge of apply MD estimation for MI estimation



Empirical Joint Probability Dist.

Challenge of apply MD estimation for MI estimation



Empirical Joint Probability Dist.

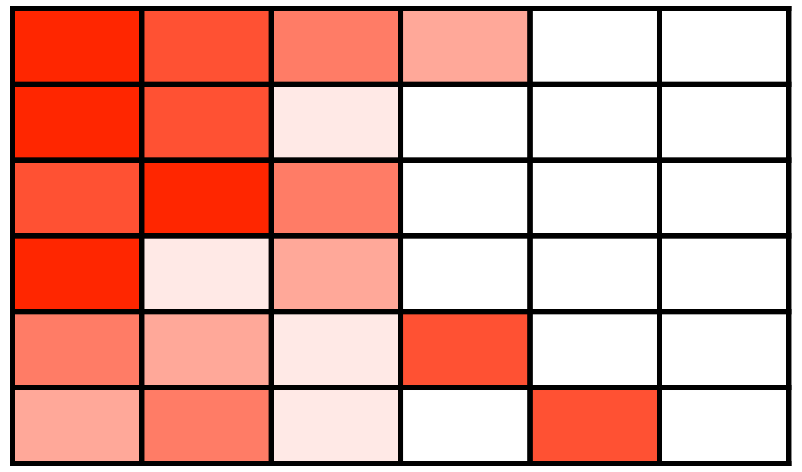
Challenge 1.

MD estimation is for ***a single random variable***, while **MI estimation** needs to handle ***two random variables.***

Our Approach to estimate MI

Challenge 1.

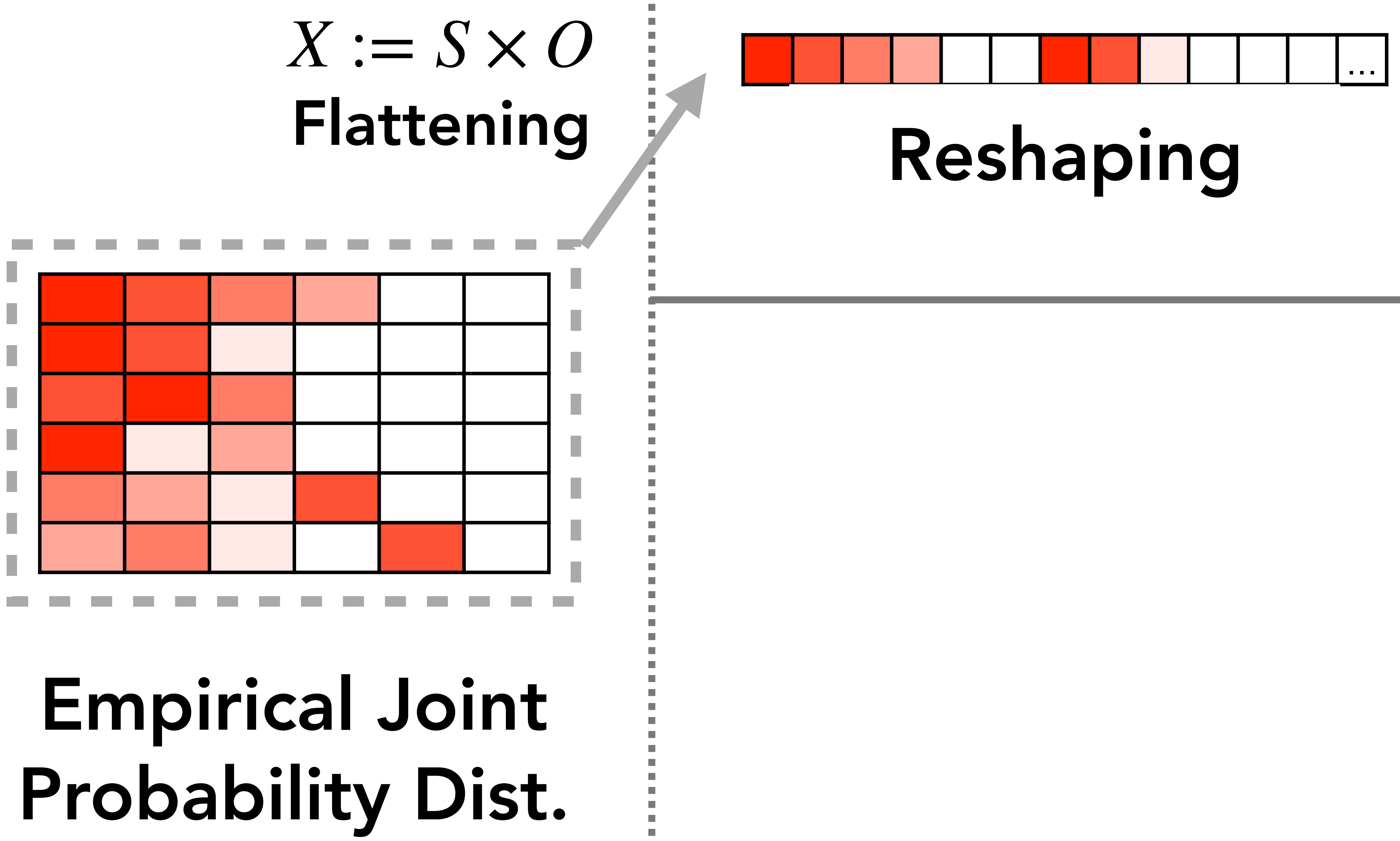
MD estimation is for ***a single random variable***, while **MI estimation** needs to handle ***two random variables.***



Empirical Joint Probability Dist.

Our Approach to estimate MI

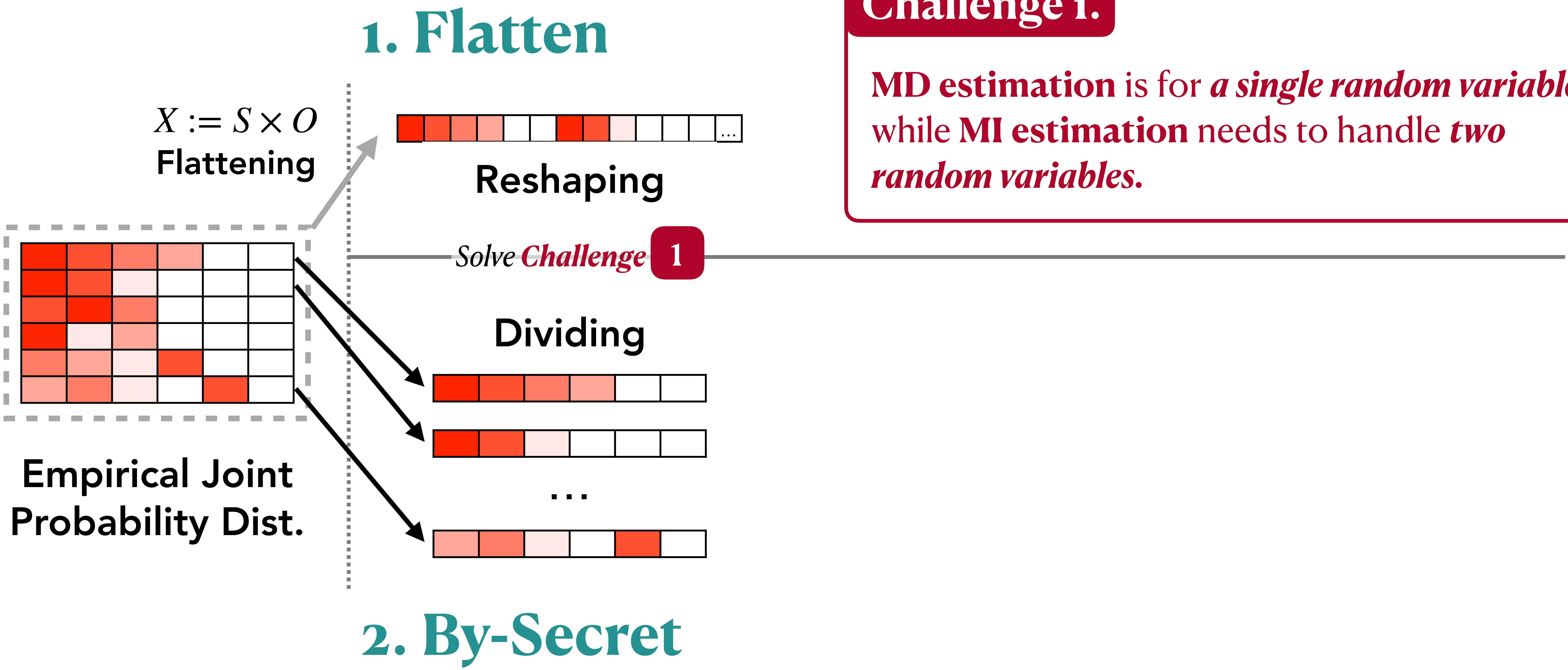
1. Flatten



Challenge 1.

MD estimation is for ***a single random variable***, while **MI estimation** needs to handle ***two random variables.***

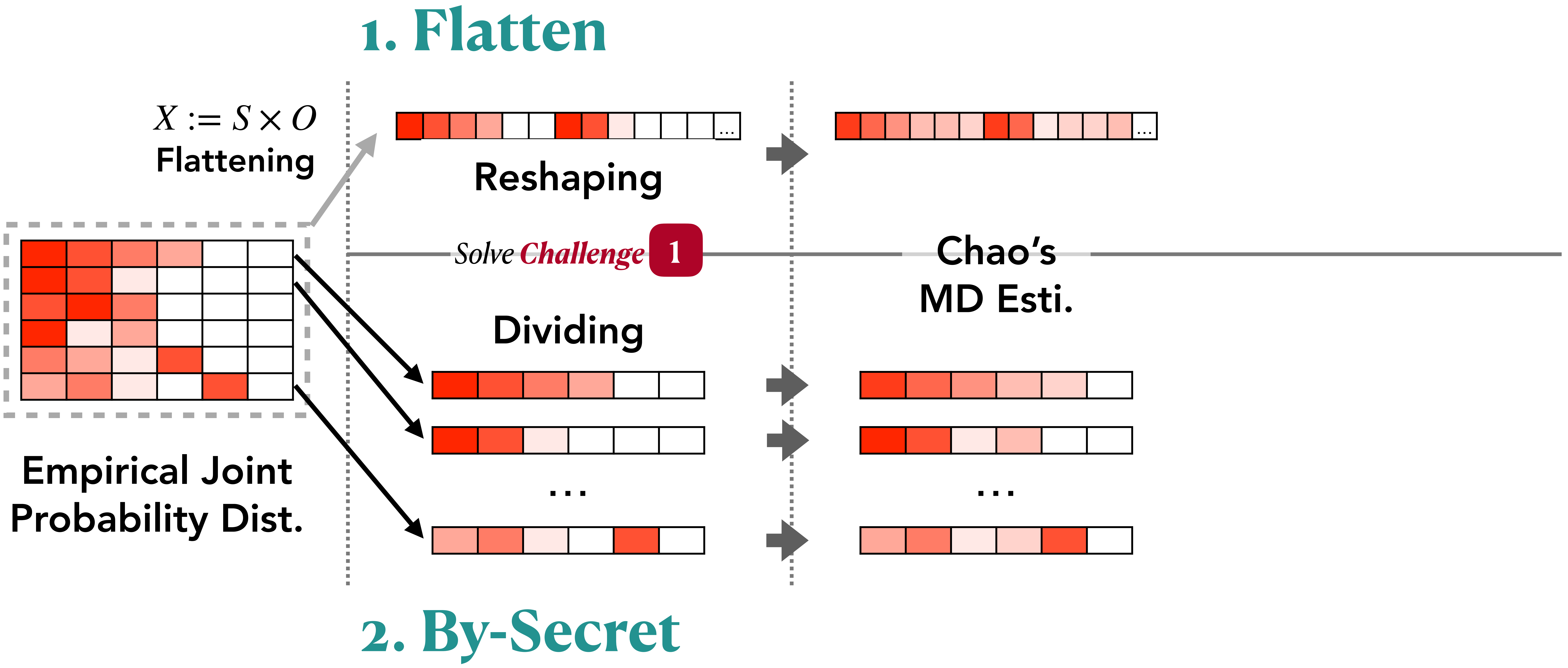
Our Approach to estimate MI



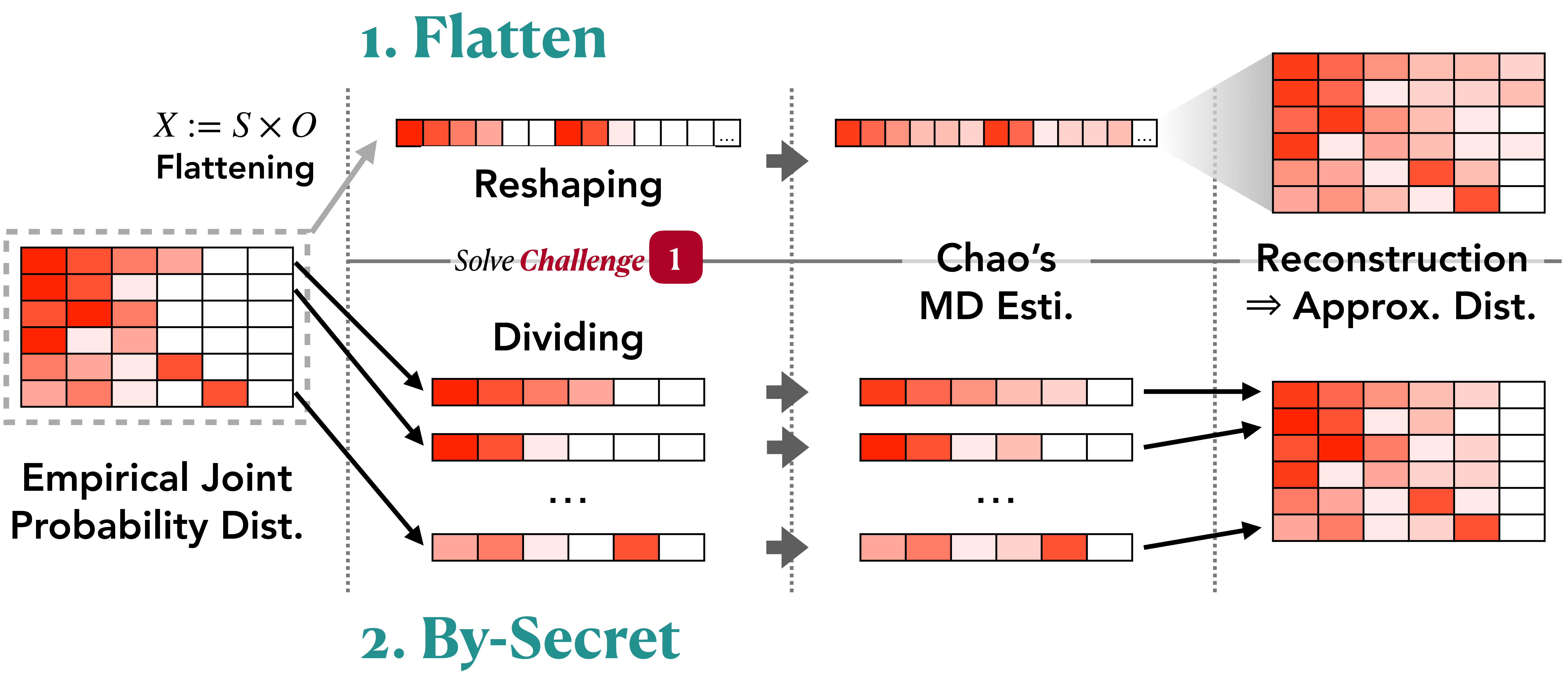
Challenge 1.

MD estimation is for ***a single random variable***, while **MI estimation** needs to handle ***two random variables.***

Our Approach to estimate MI



Our Approach to estimate MI

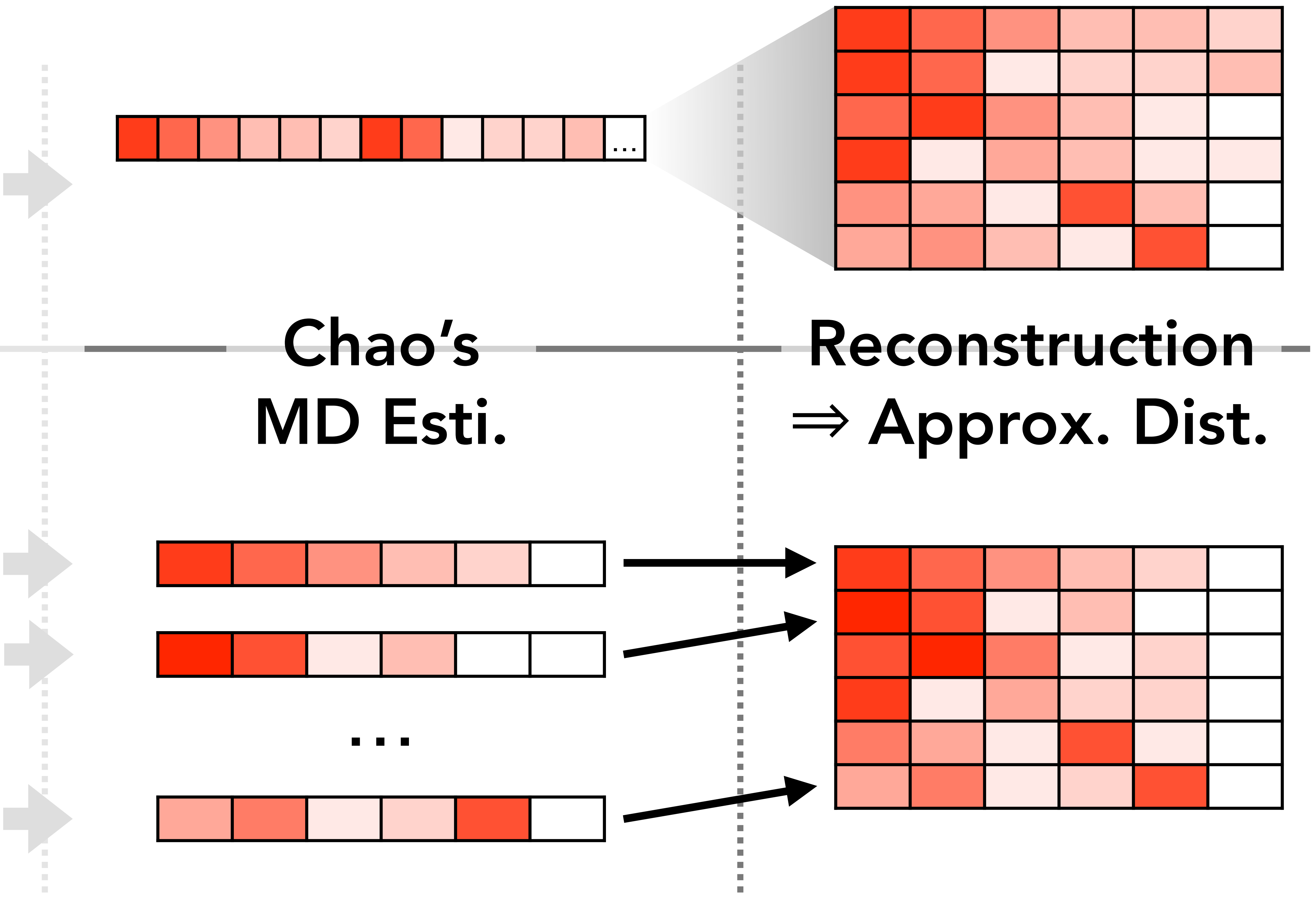


Our Approach to estimate MI

Challenge 2.

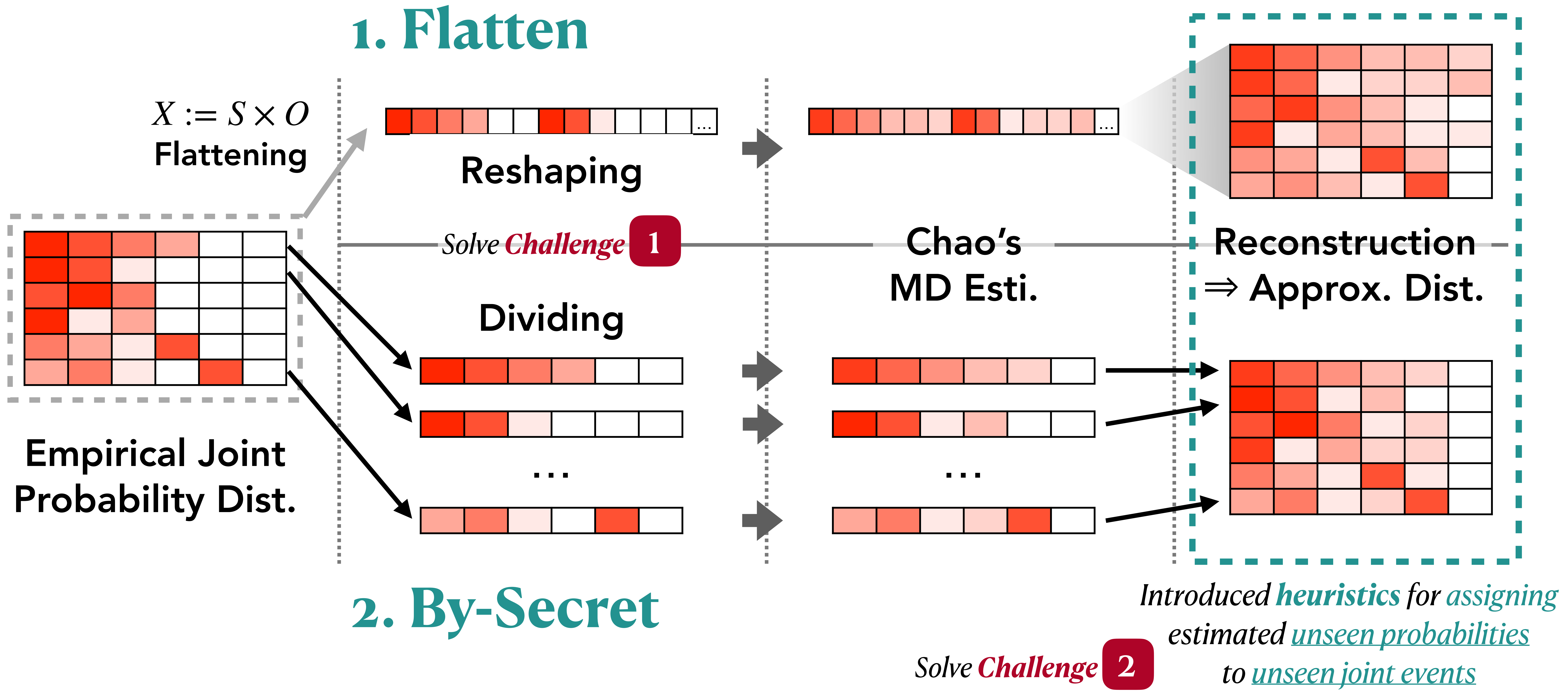
MD estimation does not provide **which missing event** has **which probability**.

It only estimates the ***shape of the distribution****, not the probability for each event.*

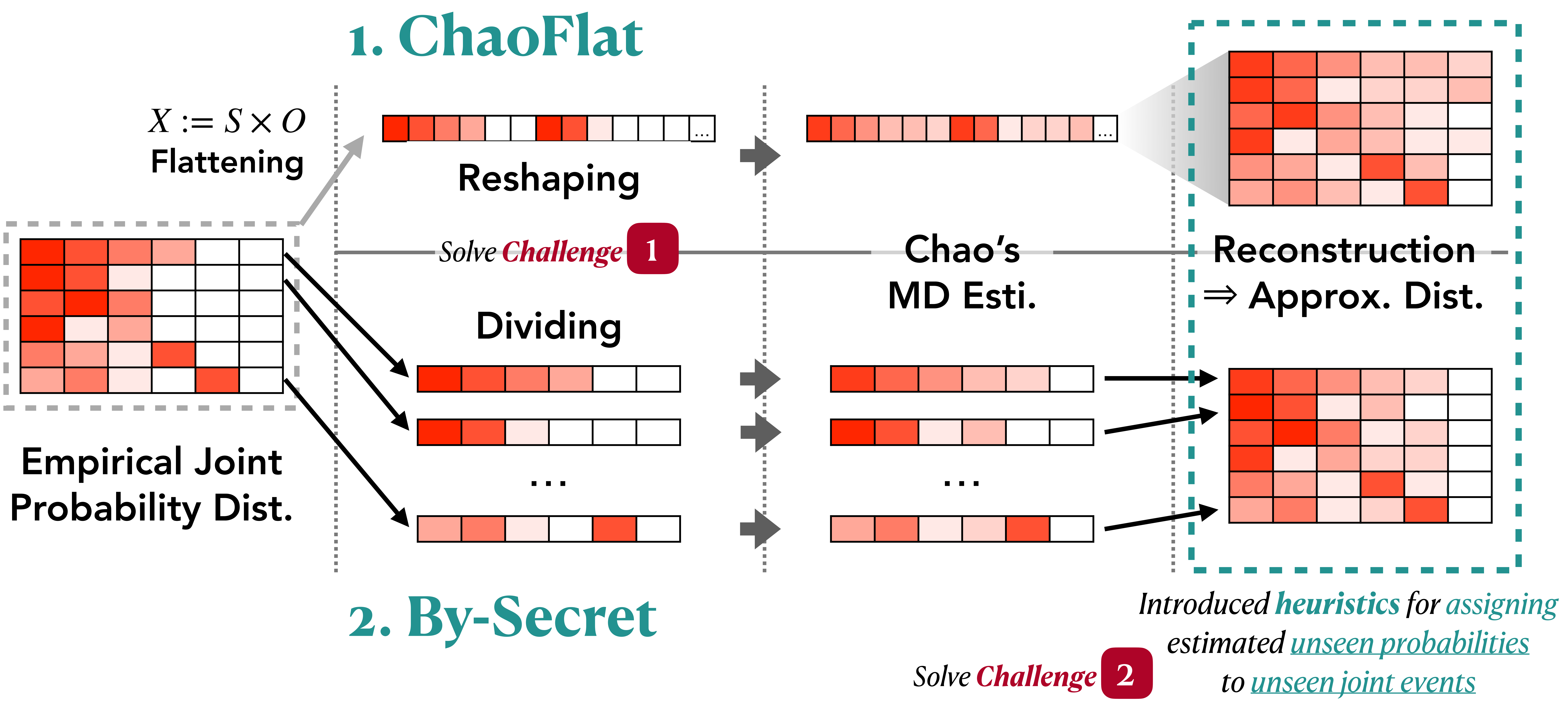


2. By-Secret

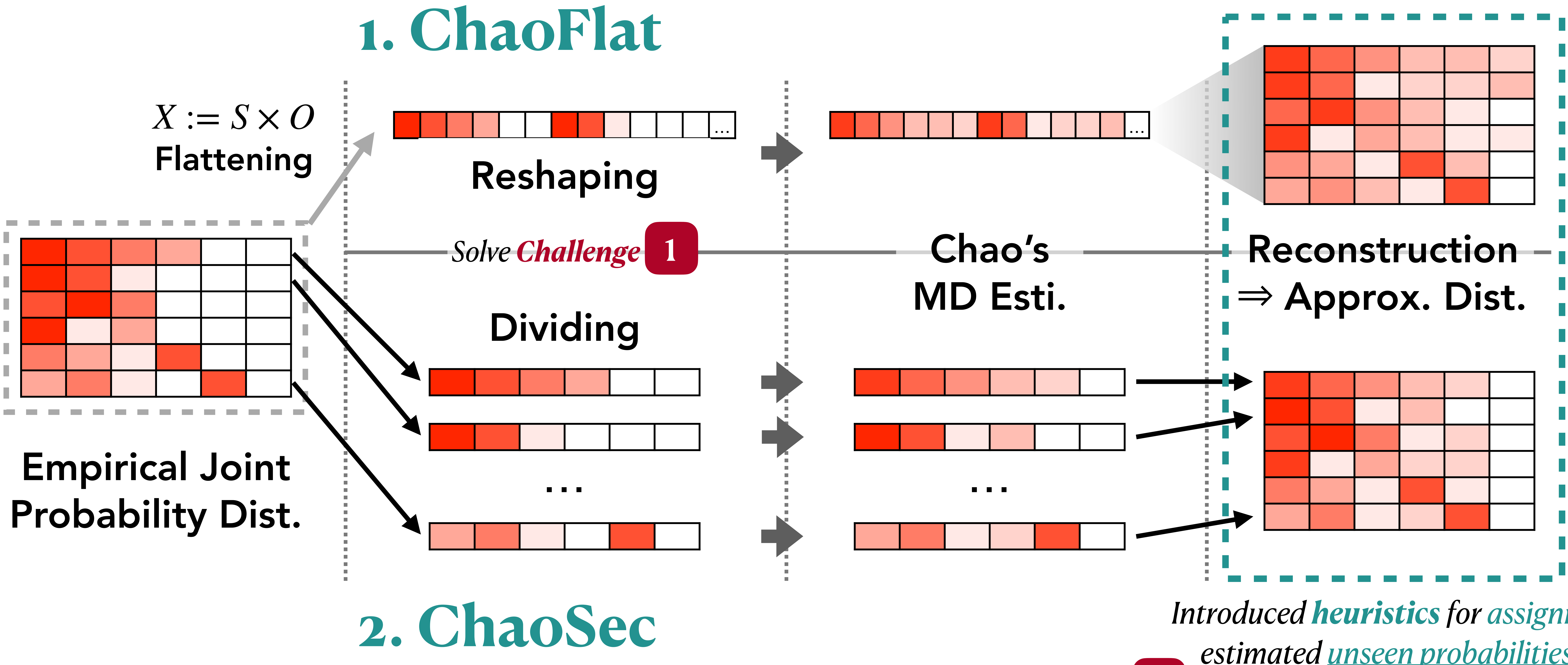
Our Approach to estimate MI



Our Approach to estimate MI



Our Approach to estimate MI



Introduced ***heuristics*** *for assigning estimated unseen probabilities to unseen joint events*

Solve ***Challenge*** **2**

Evaluation

Our estimator

ChaoFlat **ChaoSec**

VS

Baselines

Empirical **Miller**

Evaluation

Our estimator

ChaoFlat **ChaoSec**

VS

- Accuracy (*Mean Square Error)*
- Safety (*whether underestimate)*

Baselines

Empirical **Miller**

Evaluation

Our estimator

ChaoFlat **ChaoSec**

VS

- Accuracy (*Mean Square Error)*
- Safety (*whether underestimate)*

Baselines

Empirical **Miller**

Benchmark

1. Subject programs from previous study

Subject	$(\lvert\mathcal{X}\rvert, \lvert\mathcal{Y}\rvert)$	Variants (N)
ProbTerm	$(N+1, 10\text{–}20)$	$\{5, 7, 9, 12\}$
RandomWalk	$(500, 24\text{–}40)$	$\{3, 5, 7, 14\}$
Reservoir	$(2^N, 2^{N/2})$	$\{4, 6, 8, 10, 12\}$
SmartGrid	$(3^N, 12)$	$\{1, 2, 3, 4, 5\}$
Window	(N, N)	$\{20, 24, 28, 32\}$

- **Small size**
- Known ground-truth MI

Evaluation

Our estimator

ChaoFlat **ChaoSec**

VS

- Accuracy (*Mean Square Error)*
- Safety (*whether underestimate)*

Baselines

Empirical **Miller**

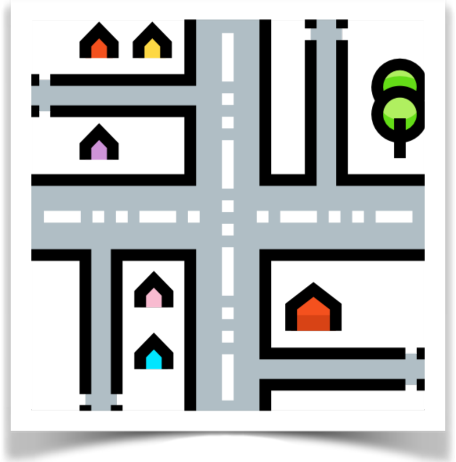
Benchmark

1. Subject programs from previous study

Subject	$(\lvert\mathcal{X}\rvert, \lvert\mathcal{Y}\rvert)$	Variants (N)
ProbTerm	$(N+1, 10\text{–}20)$	$\{5, 7, 9, 12\}$
RandomWalk	$(500, 24\text{–}40)$	$\{3, 5, 7, 14\}$
Reservoir	$(2^N, 2^{N/2})$	$\{4, 6, 8, 10, 12\}$
SmartGrid	$(3^N, 12)$	$\{1, 2, 3, 4, 5\}$
Window	(N, N)	$\{20, 24, 28, 32\}$

- **Small size**
- Known ground-truth MI

2. Practical scenarios with real-world examples

1) 

Location Privacy

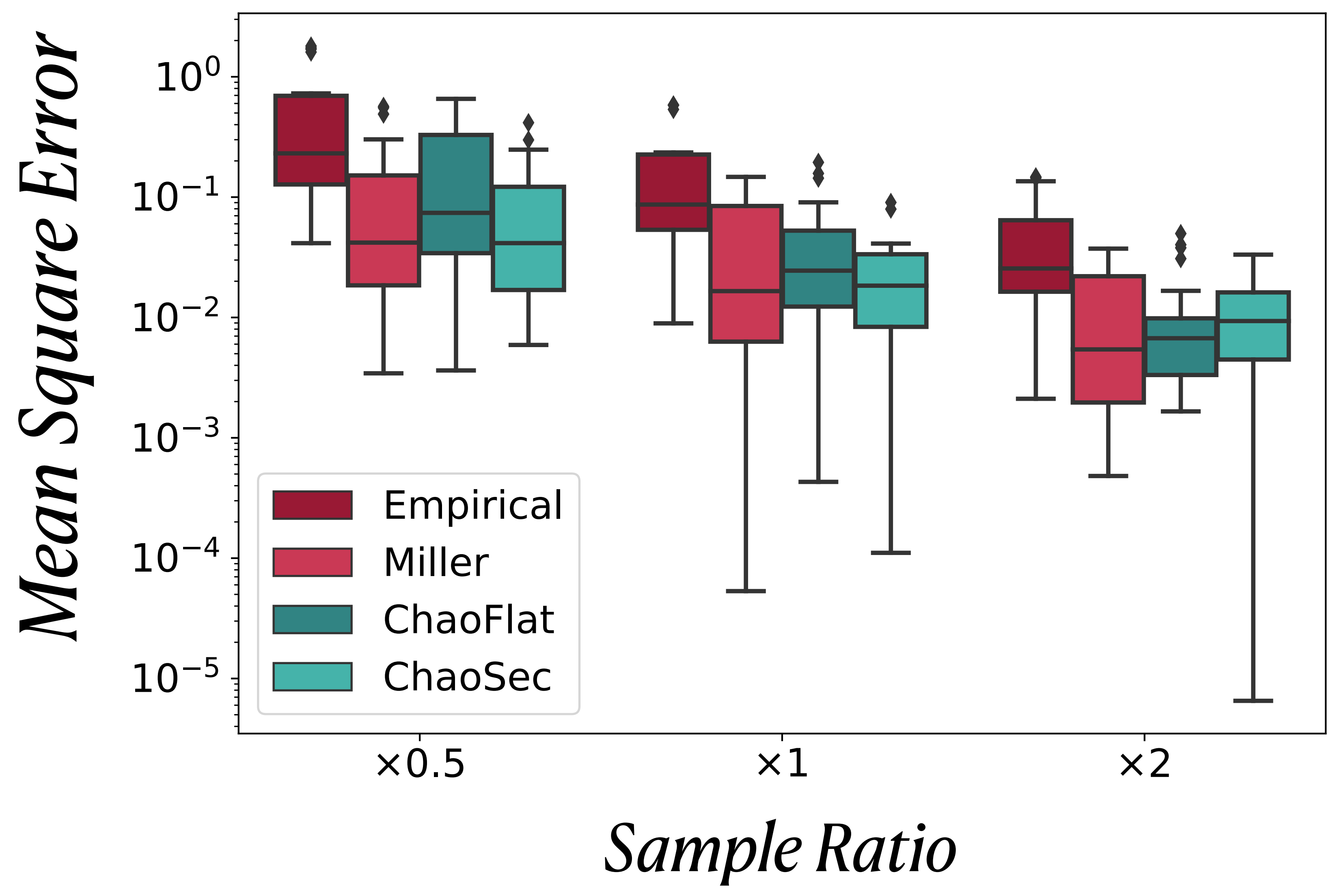
2) 

Passport Tracing

- **Domain of the joint event space ⟨ , ⟩ is substantially larger**
- Empirical ground truth

Result 1: Subject Programs from Prev. Study

— where the observable space is small —



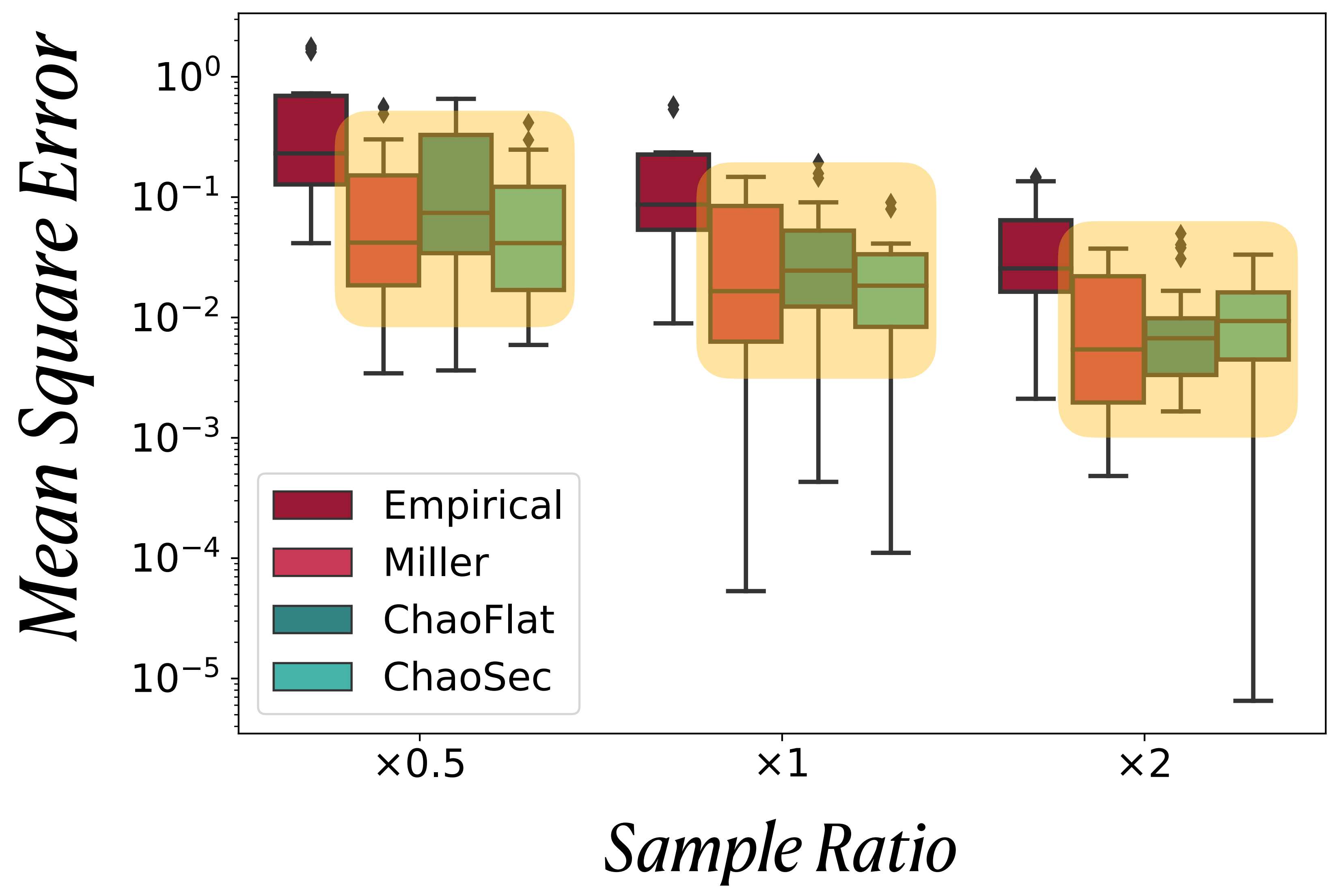
* *Sample Ratio of* $\times k$: $|\text{sample}| = |S| \cdot |O| \times k$

Accuracy

- MSE(**Empirical**) $\gg$ MSE(**Miller**), MSE(**ChaoFlat**), MSE(**ChaoSec**)
- No significant difference b/w **Miller**, **ChaoFlat**, **ChaoSec**

Result 1: Subject Programs from Prev. Study

— where the observable space is small —



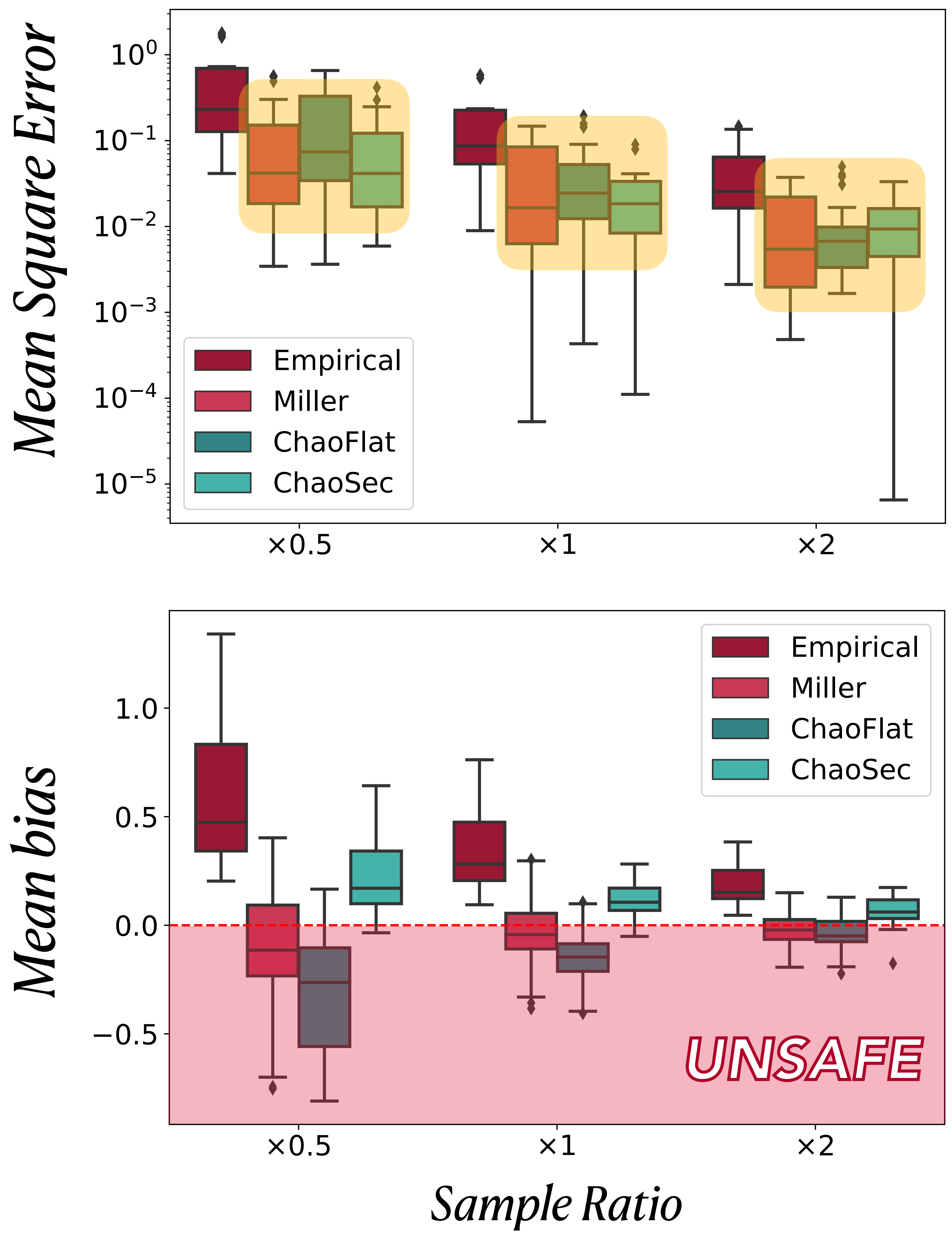
* *Sample Ratio of* $\times k$: $|\text{sample}| = |S| \cdot |O| \times k$

Accuracy

- MSE(**Empirical**)
  $\gg$ MSE(**Miller**), MSE(**ChaoFlat**), MSE(**ChaoSec**)
- No significant difference b/w **Miller, ChaoFlat, ChaoSec**

Result 1: Subject Programs from Prev. Study

— where the observable space is small —



Accuracy

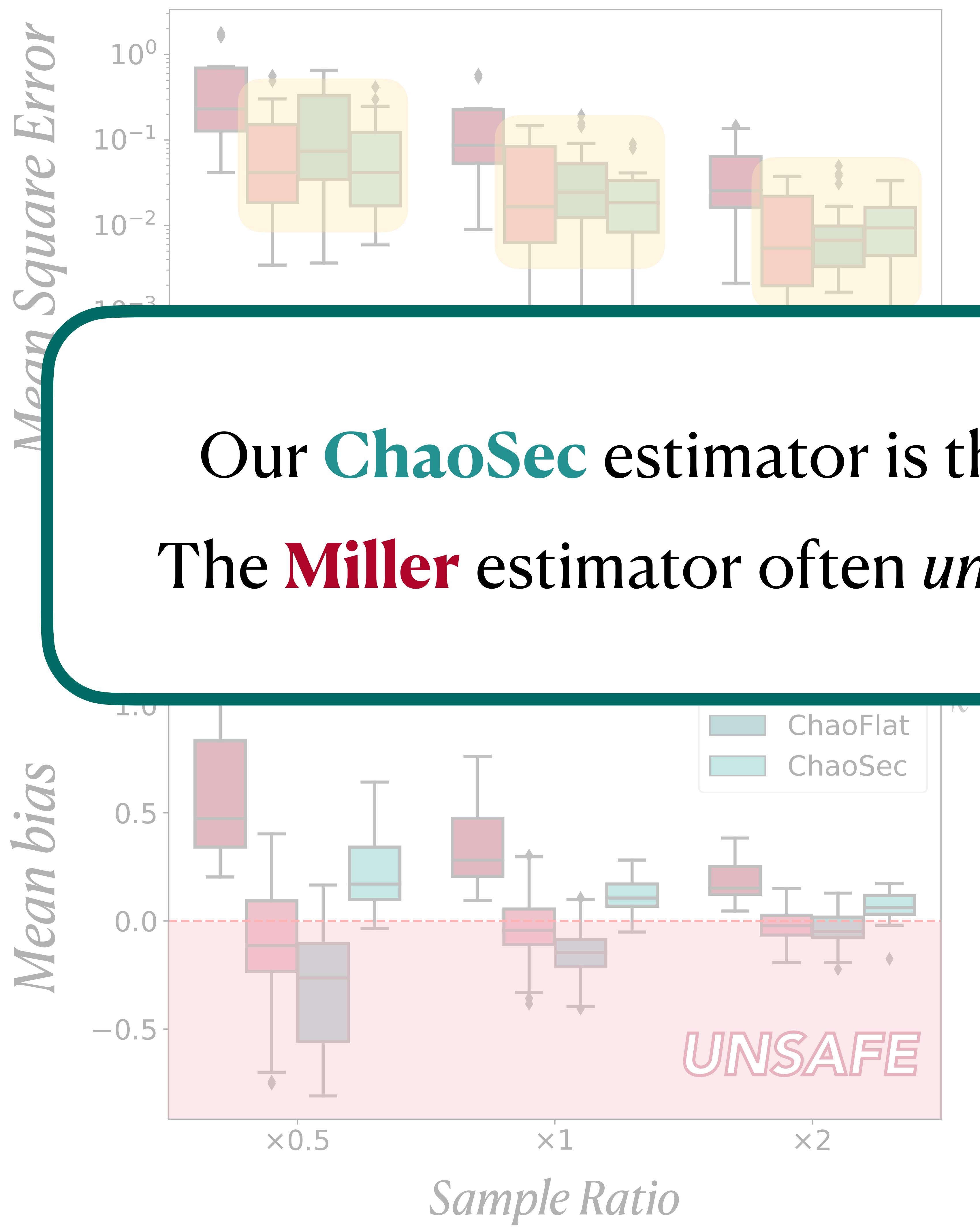
- MSE(**Empirical**)
  $\gg$ MSE(**Miller**), MSE(**ChaoFlat**), MSE(**ChaoSec**)
- No significant difference b/w **Miller, ChaoFlat, ChaoSec**

Safety

- **Miller** underestimates **57%** of the estimation.
- **ChaoSec** underestimates **8%** of the estimation.
- **ChaoFlat** underestimates **67%** of the estimation.

Result 1: Subject Programs from Prev. Study

— where the observable space is small —



Accuracy

- MSE(**Empirical**) ≫ MSE(**Miller**), MSE(**ChaoFlat**), MSE(**ChaoSec**)

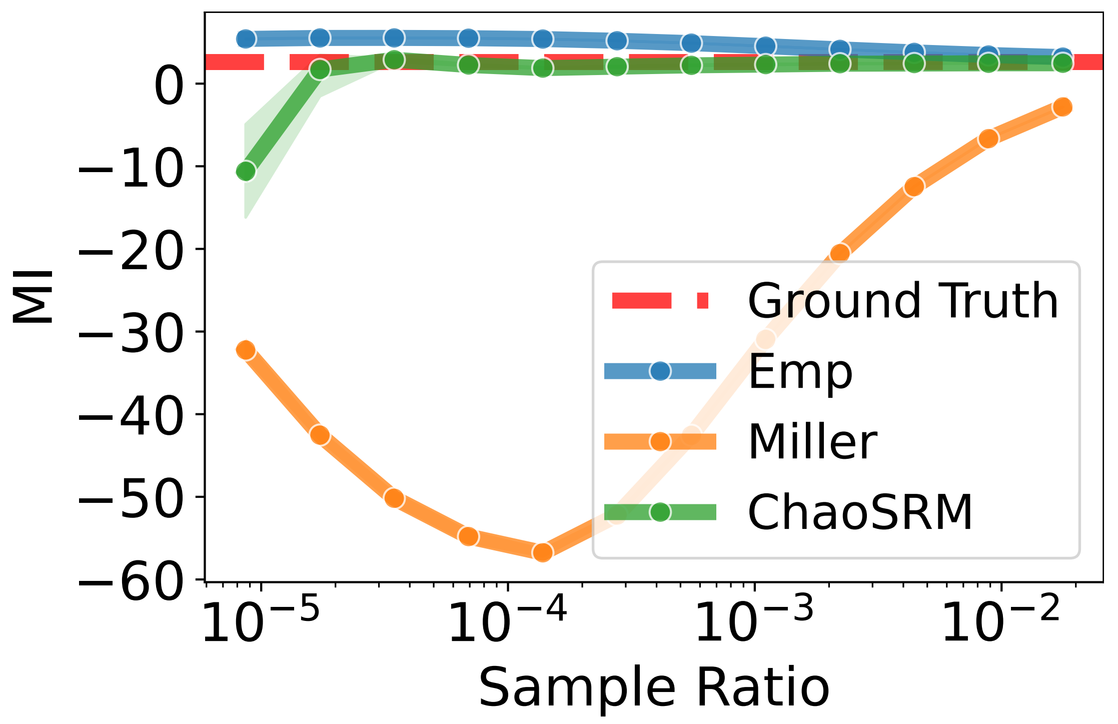
Our **ChaoSec** estimator is the best estimator in terms of both **safety** and **accuracy**.

The **Miller** estimator often *unsafely underestimates the MI* unless the sample size is large.

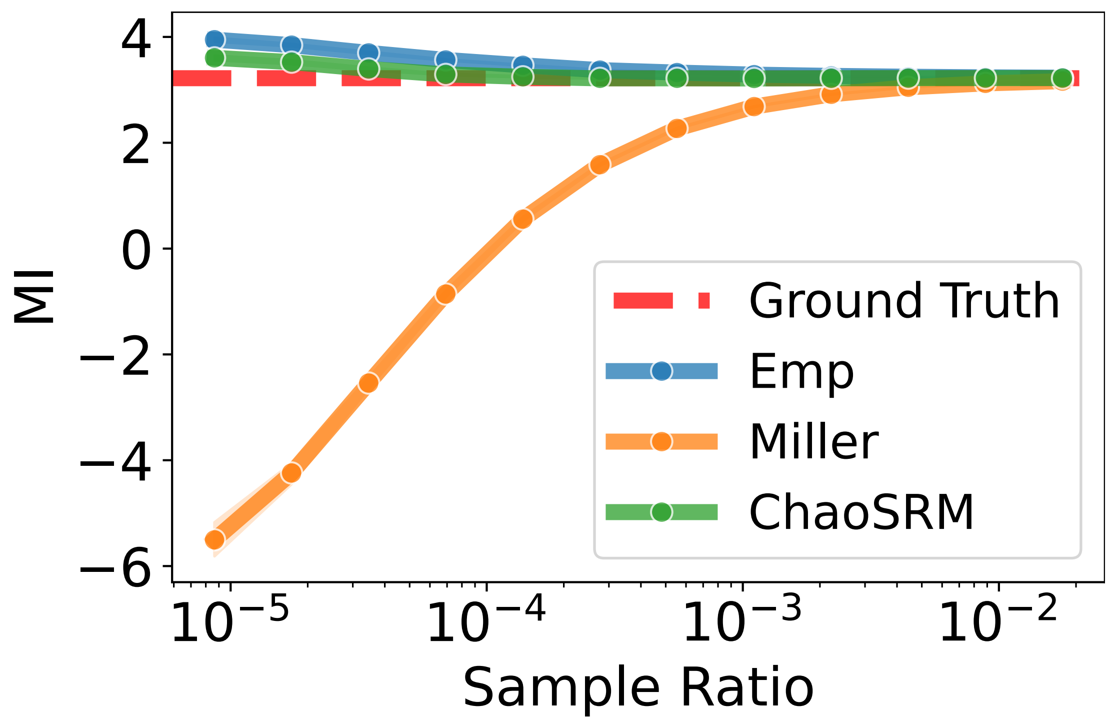
- **Miller** underestimates 57% of the estimation.
- **ChaoSec** underestimates 8% of the estimation.
- **ChaoFlat** underestimates 67% of the estimation.

Result 2: Practical Scenarios

Location Privacy

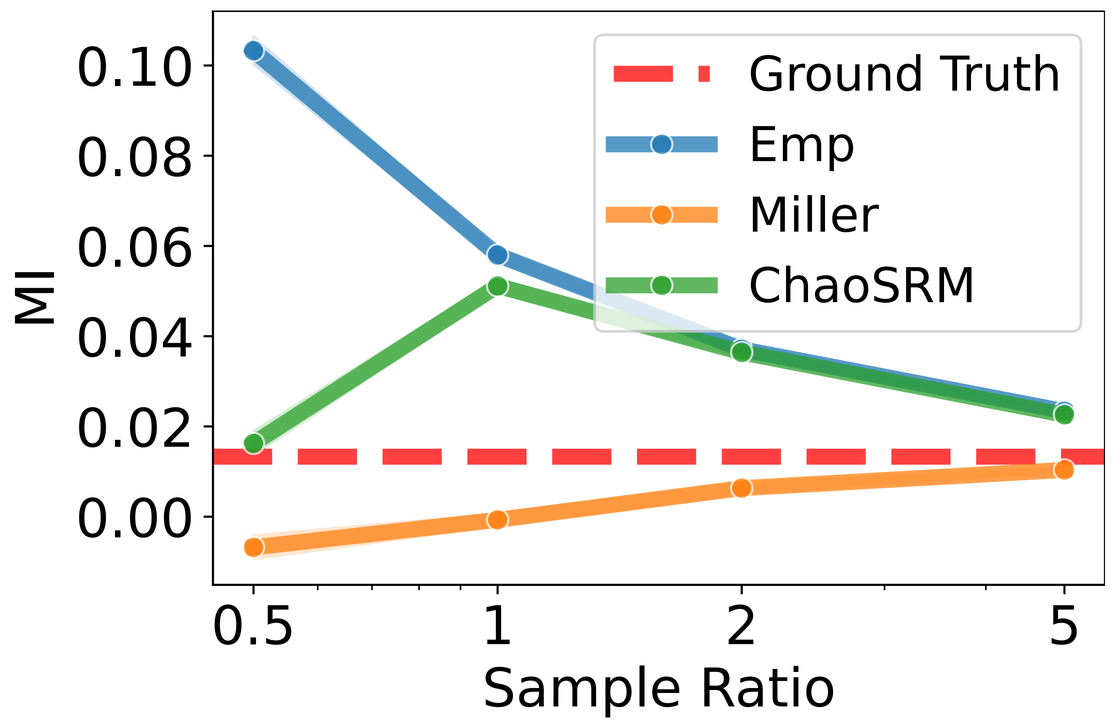


Planner Laplacian

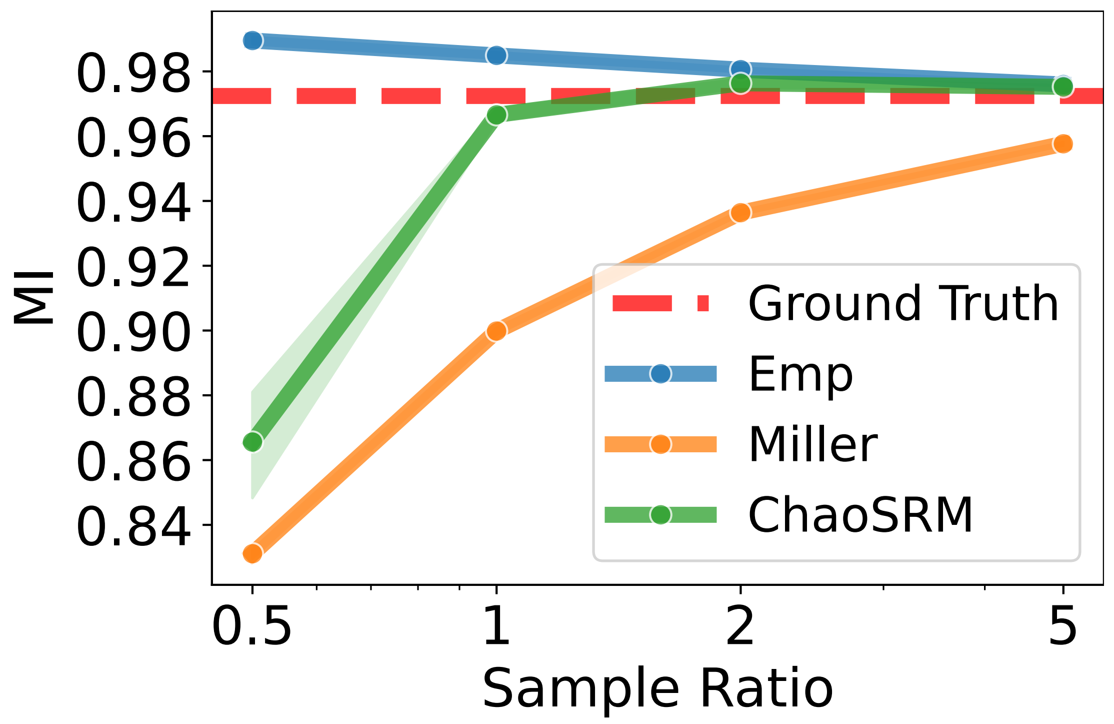


Oya et al.'s LPPM

Passport Tracing



British, Fixed



British, Unfixed

- The ***Domain of the joint event space*** ⟨🗝️,🔍⟩ ***is substantially larger*** than the previous subject programs.
- **Miller** estimator significantly underestimates (even < 0) due to the large bias correction term.

[*Accuracy*] **ChaoSec** $>$ **Empirical** $\gg$ **Miller** [*Safety*] **Empirical** $\approx$ **ChaoSec** $\gg$ **Miller**

Research **Aim**

Empirical estimator

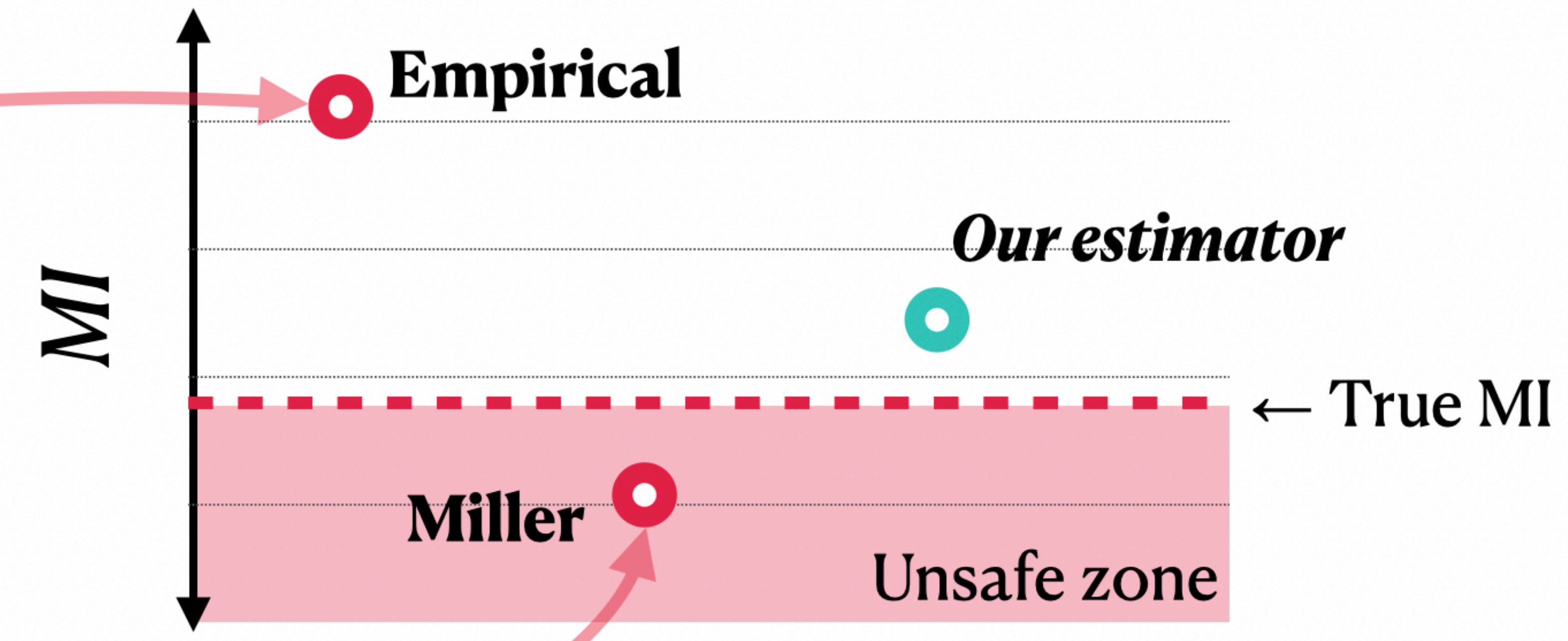
$$\hat{I}_{emp}(S;O) = \hat{H}_{emp}(X) - \hat{H}_{emp}(X \mid Y)$$

Inaccurate

Miller estimator

$$\hat{I}_{miller} = \hat{I}_{emp} - \frac{(m_S - 1)(m_O - 1)}{2n}$$

Unsafe w/ small samples



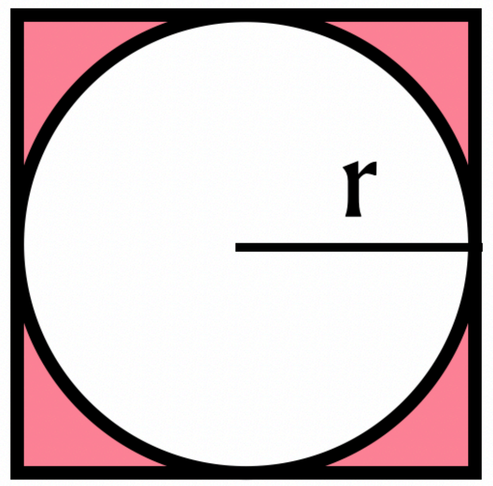
We developed an estimator that ***accurately*** *and* ***safely*** *estimates the mutual information in the presence of* ***missing or rare events***.

How Correct/Secure is our Software?

Q. What is the probability of a thrown 🎾 ball to the 🟥 square dropped not into the ⚪ circle?

1️⃣ Analytic methodology

If the problem can easily be ***mathematically*** *modeled, (e.g, area = circle)*

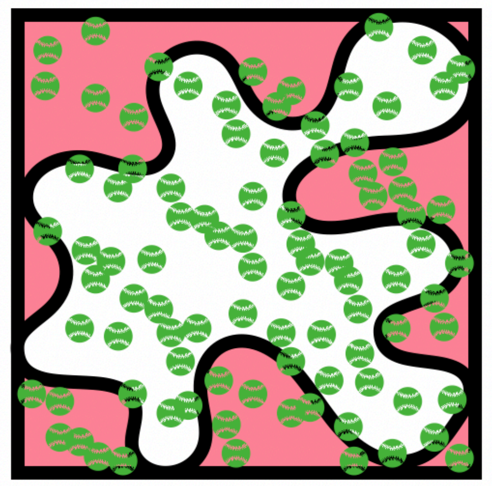


$$\Pr(\neg\text{in circle}) = \frac{Area(\text{Square}) - Area(\text{Circle})}{Area(\text{square})} = \frac{(2r)^2 - \pi r^2}{(2r)^2} = \frac{4-\pi}{4} \approx 0.2146...$$

✅ ***Precise result / Formal guarantees***

2️⃣ Empirical methodology

For example, the ***Monte Carlo method****, where we* ***simulate*** *the ball throwing*

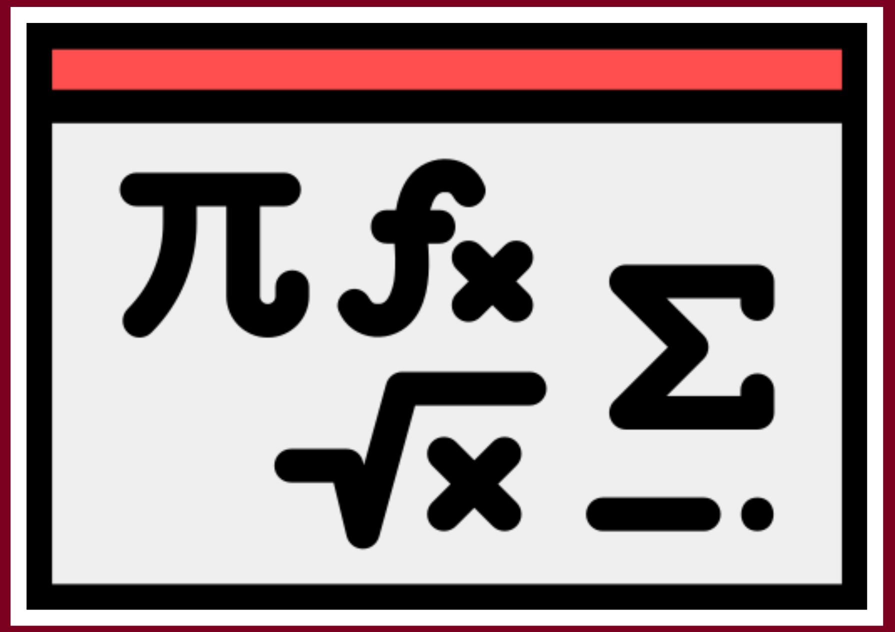


$$\hat{\Pr}(\neg\text{in area}) = \frac{\#\text{ of balls outside the area}}{\#\text{ of balls thrown}} = \frac{3577}{10000} = 0.3577$$

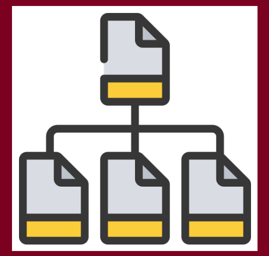
✅ ***Scalable****, i.e., can deal with complex problems*

Two Ways to answer How Correct/Secure is our Software?

Analytical Methods

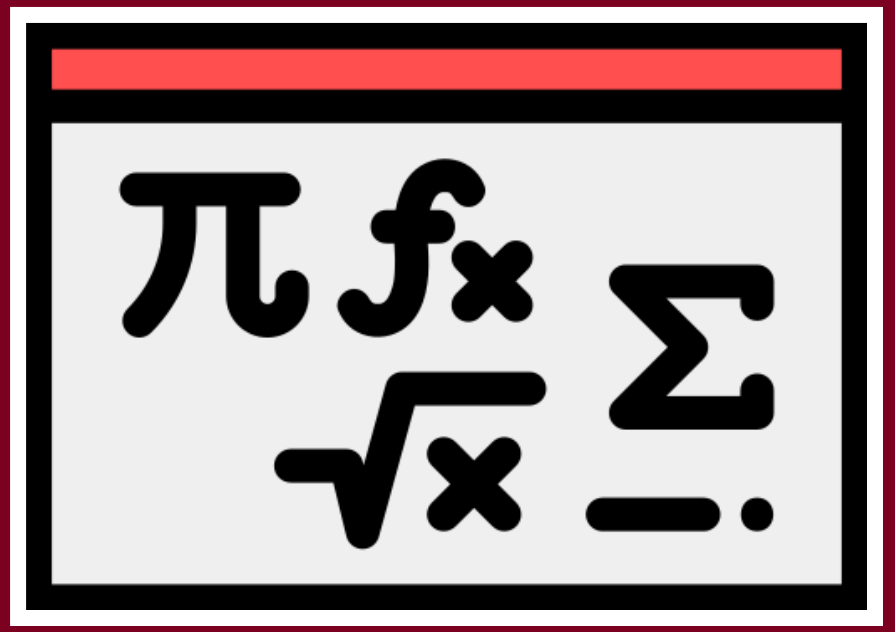


Mathematical proof can provide a ***formal guarantee***

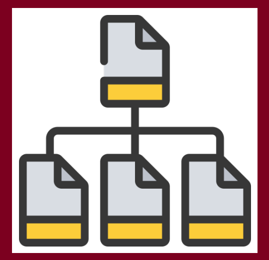


Scalability issues on modern software

Analytical Methods

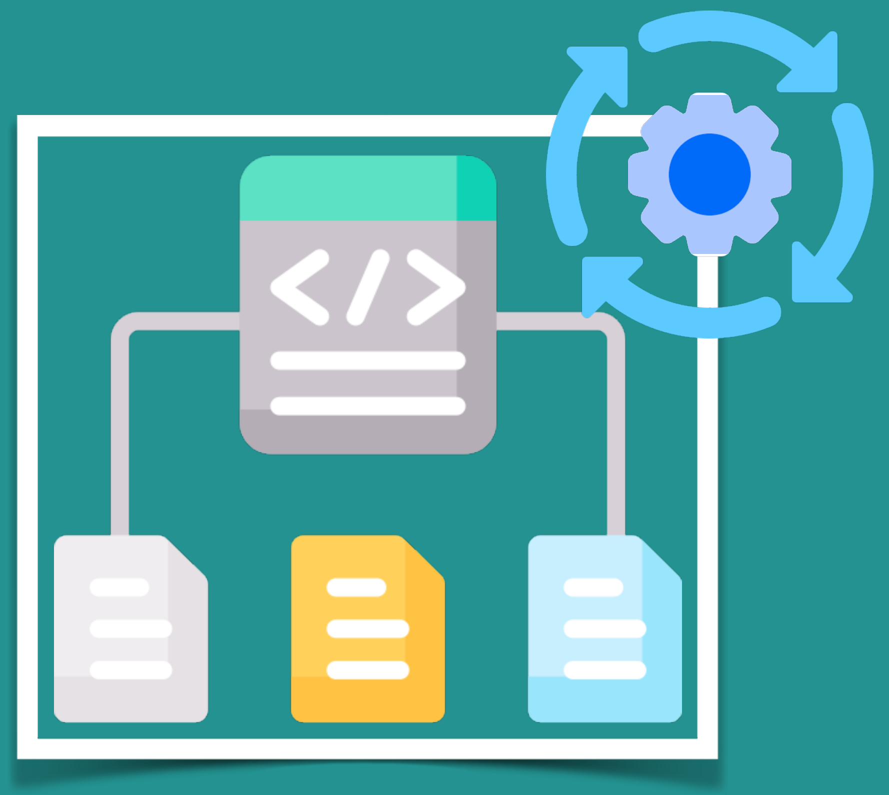


Mathematical proof can provide a ***formal guarantee***



Scalability issues on modern software

Empirical Methods

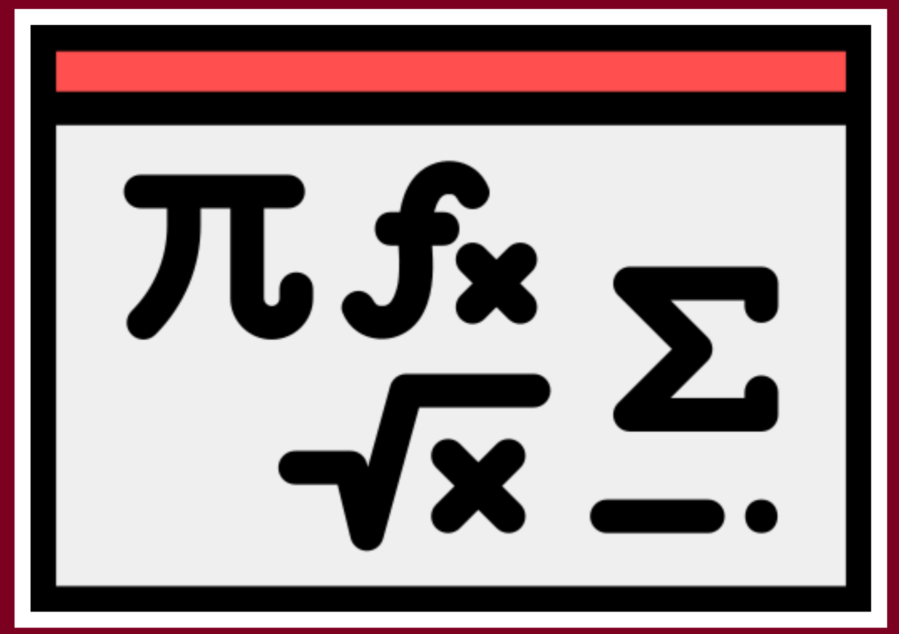


Test software by running it with various test executions

By actually running the software, it solves the ⚠️ scalability issue

There is always **unseen**
⇒ ***No guarantee***

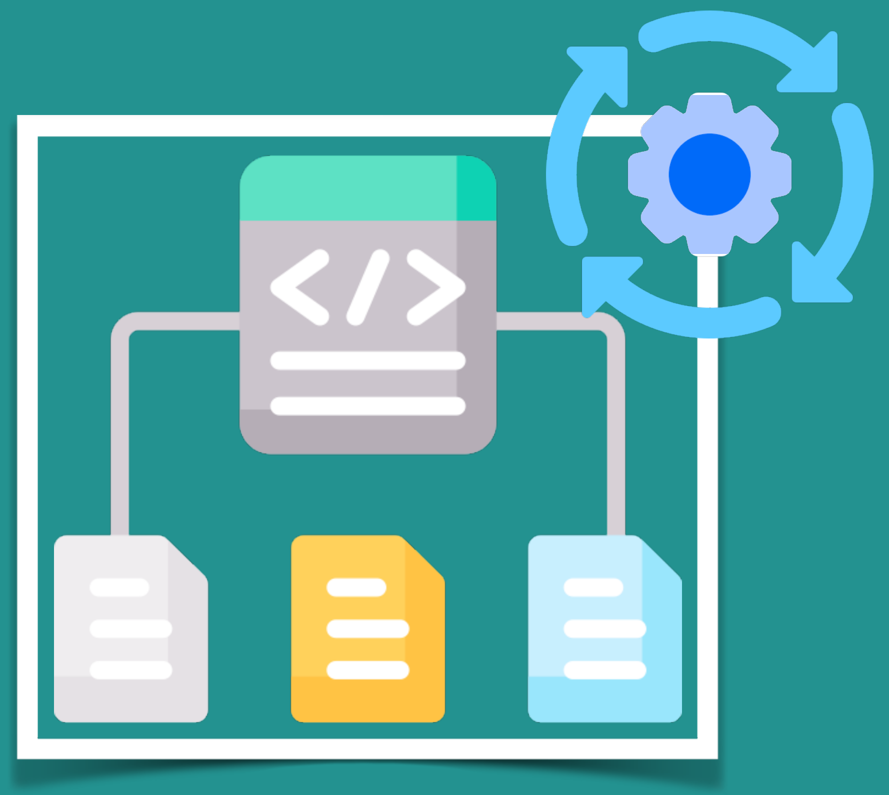
Analytical Methods



Mathematical proof can provide a ***formal guarantee***

Scalability issues on modern software

Empirical Methods



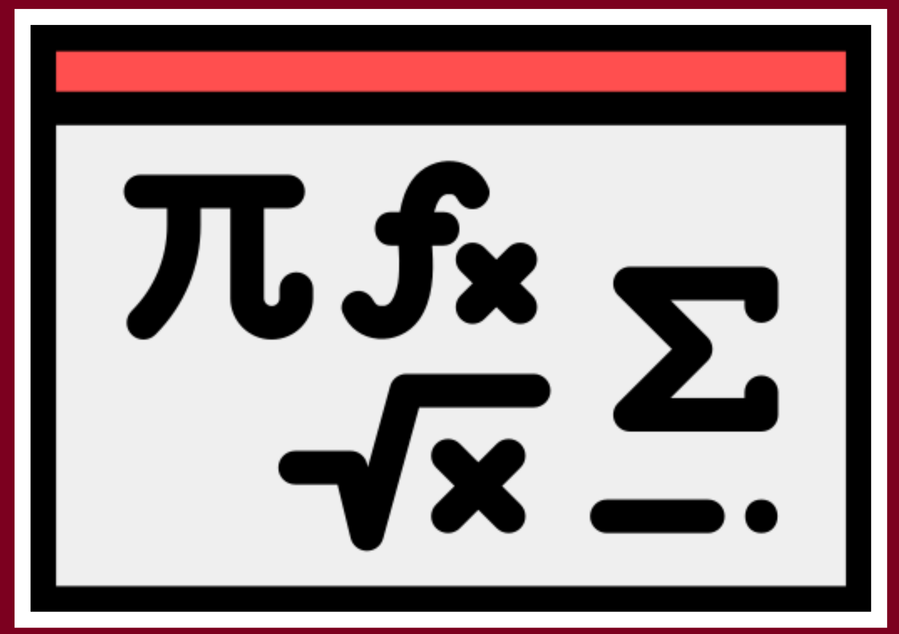
Test software by running it with various test executions

By actually running the software, it solves the ⚠ scalability issue

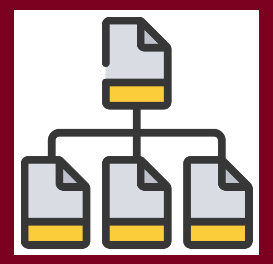
There is always unseen
⇒ ***No guarantee***

Statistics can solve this!

Analytical Methods

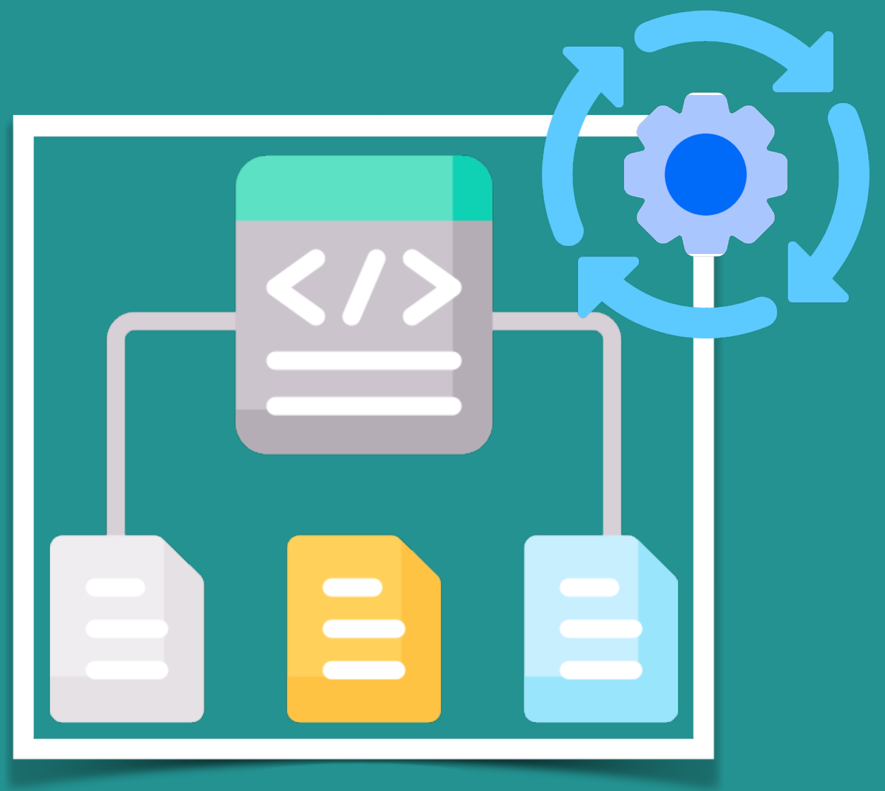


Mathematical proof can provide a ***formal guarantee***



⚠ Scalability issues on modern software

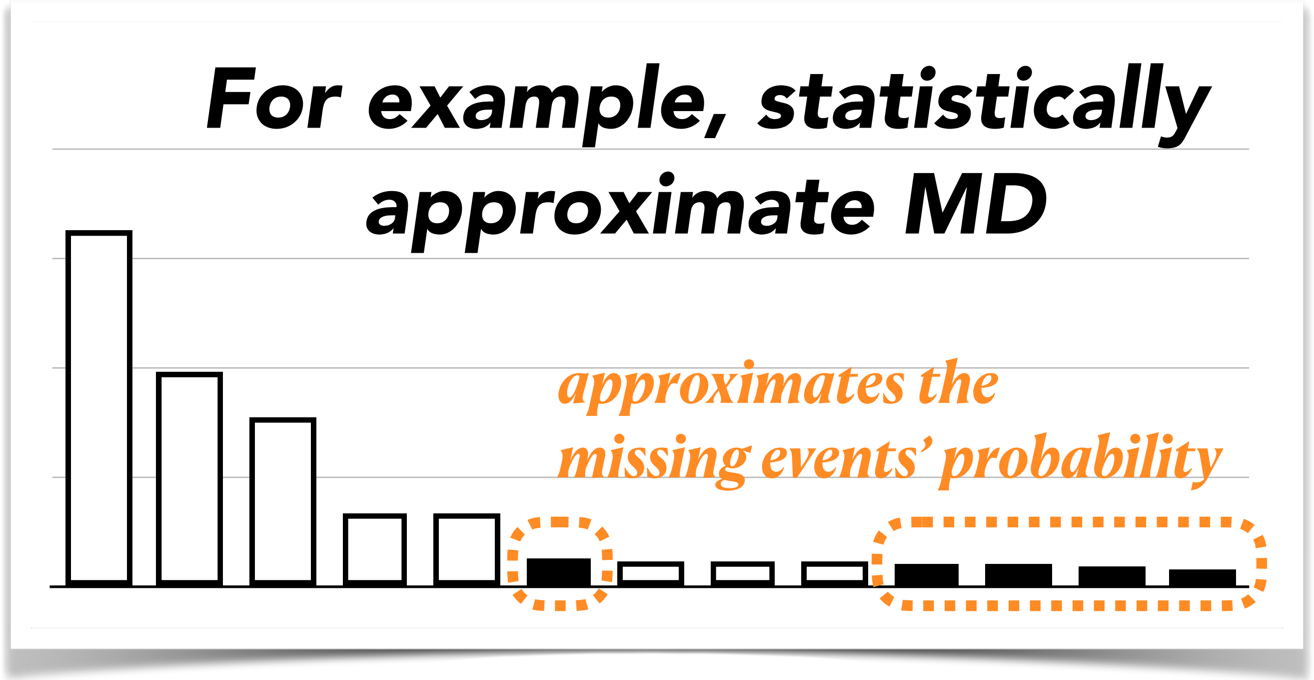
Empirical Methods



Test software by running it with various test executions

By actually running the software, it solves the ⚠ scalability issue

There is always unseen
⇒ ***No guarantee***



Statistics can solve this!

Accounting for Missing Events in Statistical Information Leakage Analysis

MAX PLANCK INSTITUTE
FOR SECURITY AND PRIVACY

Research Aim

Empirical estimator

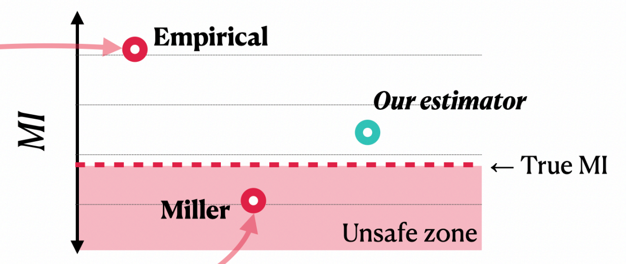
$\hat{I}_{emp}(S;O) = \hat{H}_{emp}(X) - \hat{H}_{emp}(X \mid Y)$

Inaccurate

Miller estimator

$\hat{I}_{miller} = \hat{I}_{emp} - \frac{(m_S - 1)(m_O - 1)}{2n}$

Unsafe w/ small samples

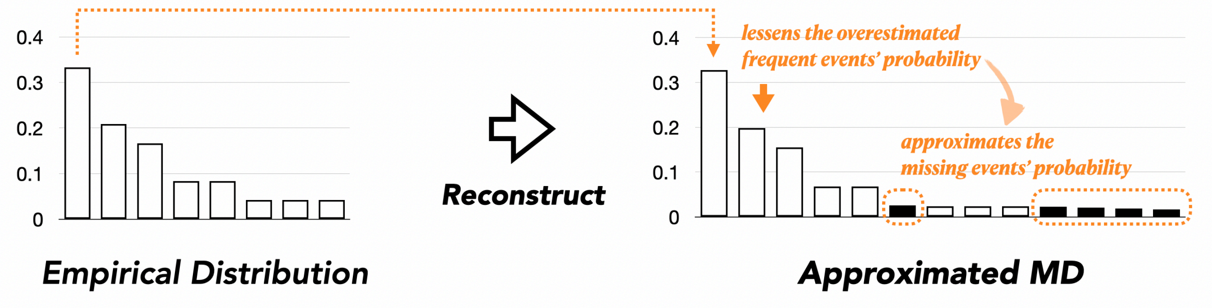


*We developed an estimator that **accurately** and **safely** estimates the mutual information in the presence of **missing or rare events**.*

Chao's Multinomial Distribution (MD) Estimation

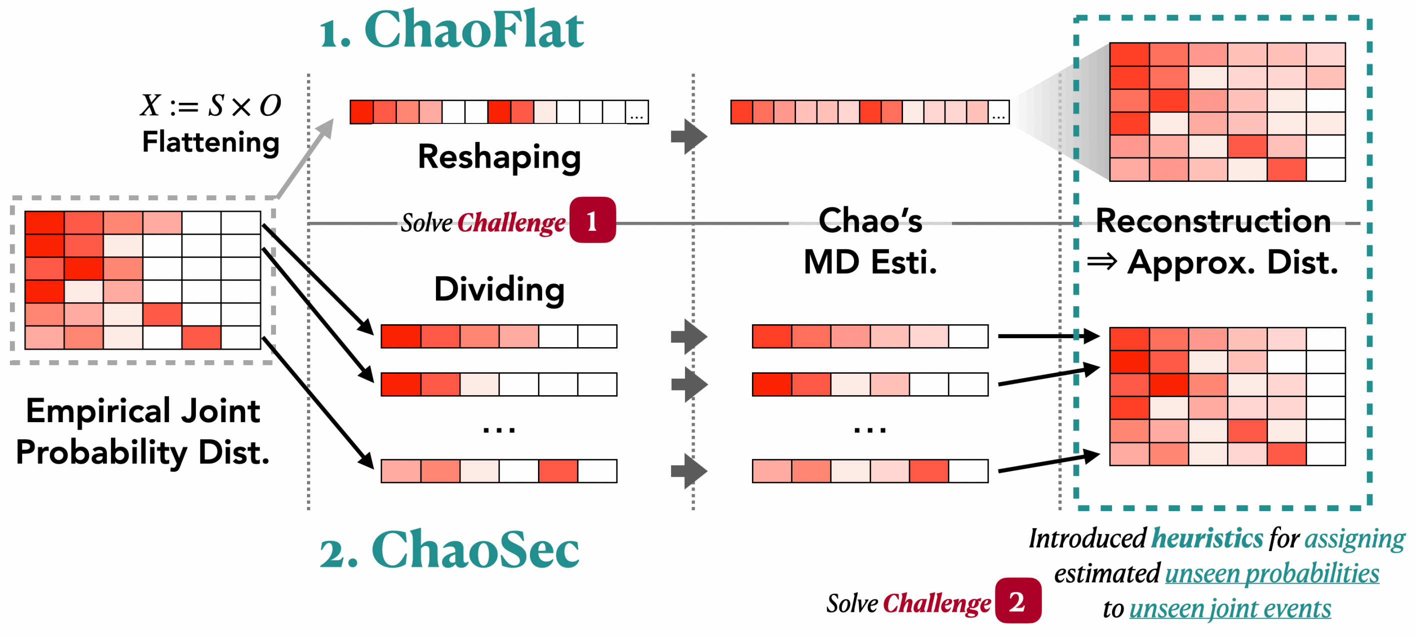
- Given a samples from the unknown multinomial distribution (MD), Chao's MD **reconstruct the underlying MD** by approximation.

➡ *Handle the missing/rare events problem* 😀

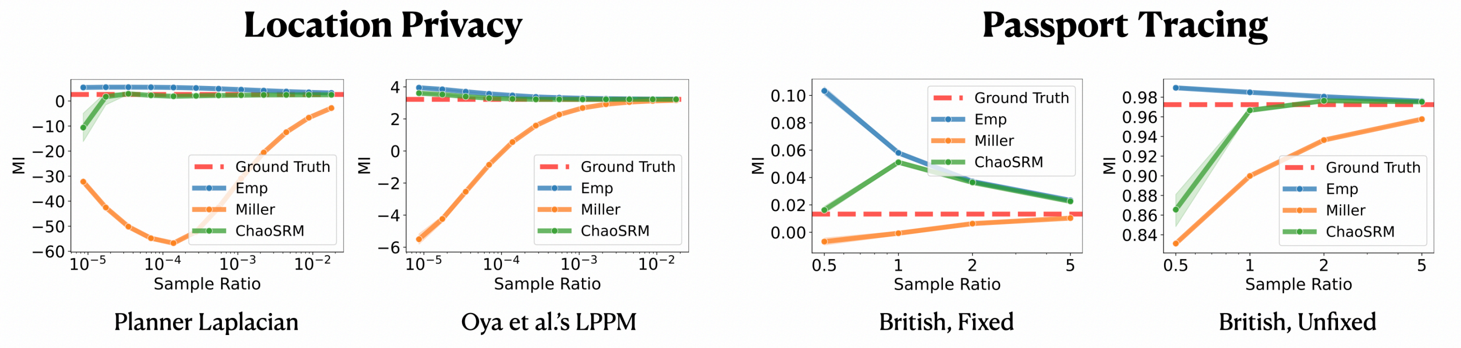


A. Chao et al, "Unveiling the species-rank abundance distribution by generalizing the good-turing sample coverage theory." Ecology, vol. 96 5, pp. 1189–1201, 2015.

Our Approach to estimate MI



Result 2: Practical Scenarios



- The *Domain of the joint event space* (🔑,🔎) ***is substantially larger*** than the previous subject programs.
- Miller estimator significantly underestimates (even < 0) due to the large bias correction term.

[Accuracy] ChaoSec > Empirical ≫ Miller *[Safety]* Empirical ≈ ChaoSec ≫ Miller

Dr. Seongmin Lee
MPI-SP Software Security
🏠 https://nimgnoeseel.github.io/

Shreyas Minocha
***Georgia Tech**
🏠 https://shreyasminocha.me/

Dr. Marcel Böhme
MPI-SP Software Security
🏠 https://mpi-softsec.github.io/

** This work has been done during his internship @ MPI-SP before joining Georgia Tech.*