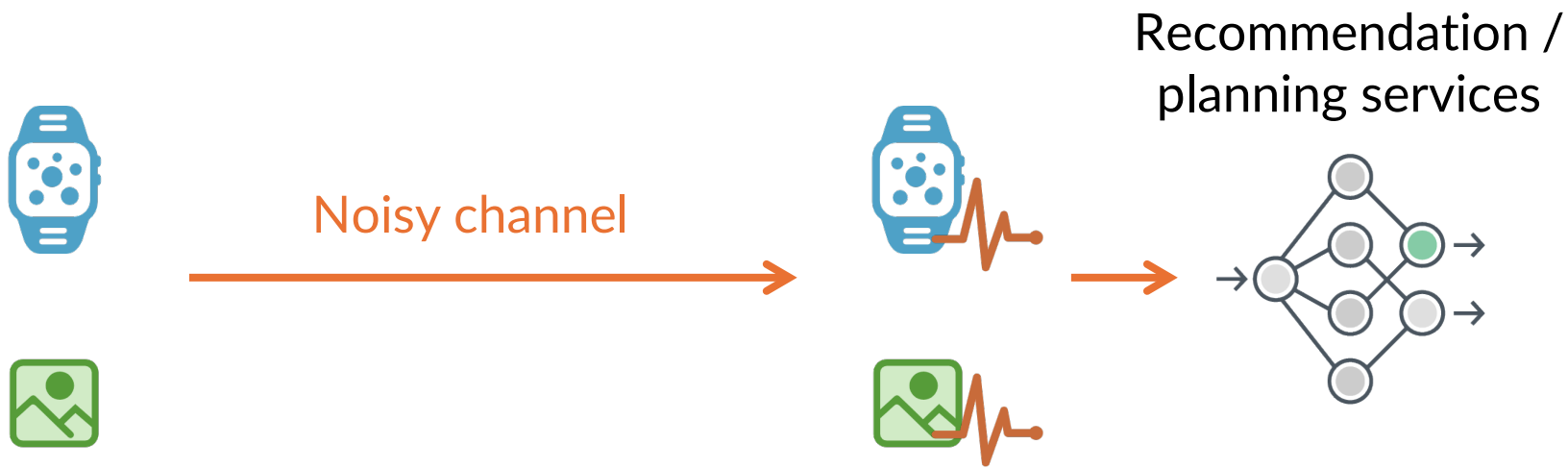
Quantifying Classifier Utility under Local Differential Privacy

Authors: Ye Zheng, Yidan Hu

RIT | Rochester Institute of Technology

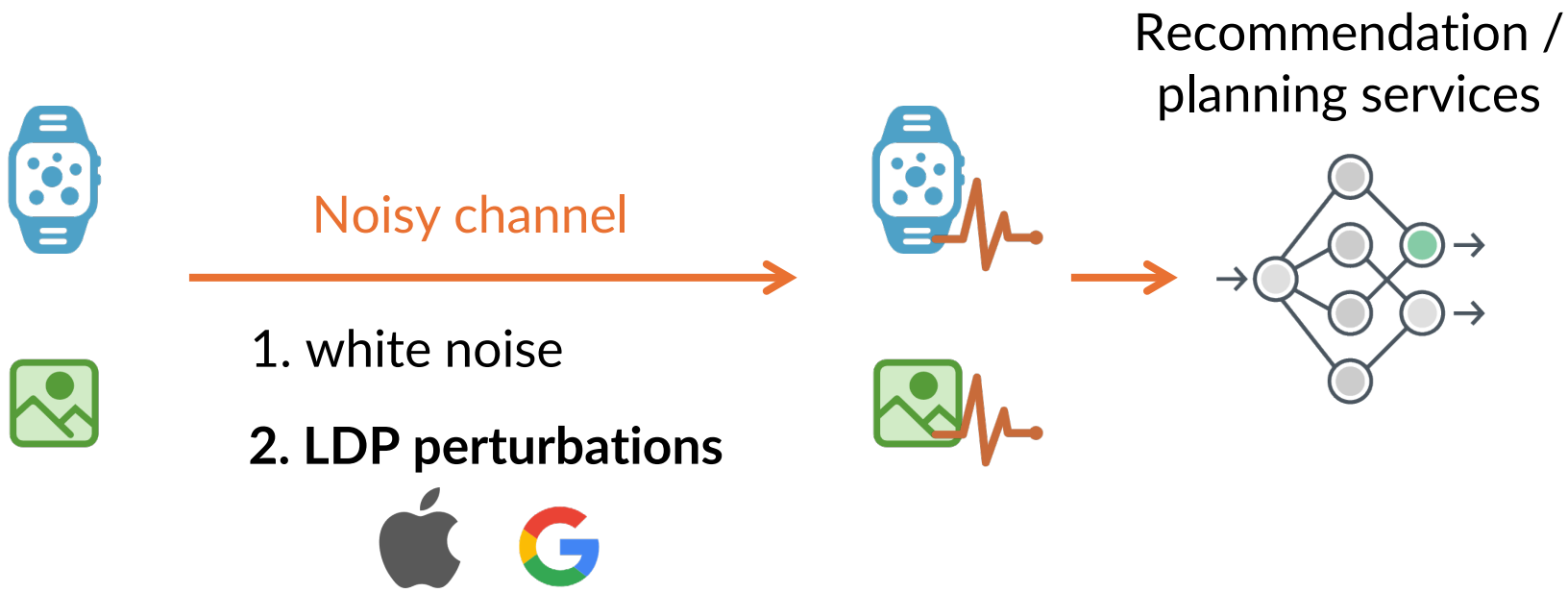
Classifier Utility under Noisy Inputs

- Input data for classifiers may be noisy



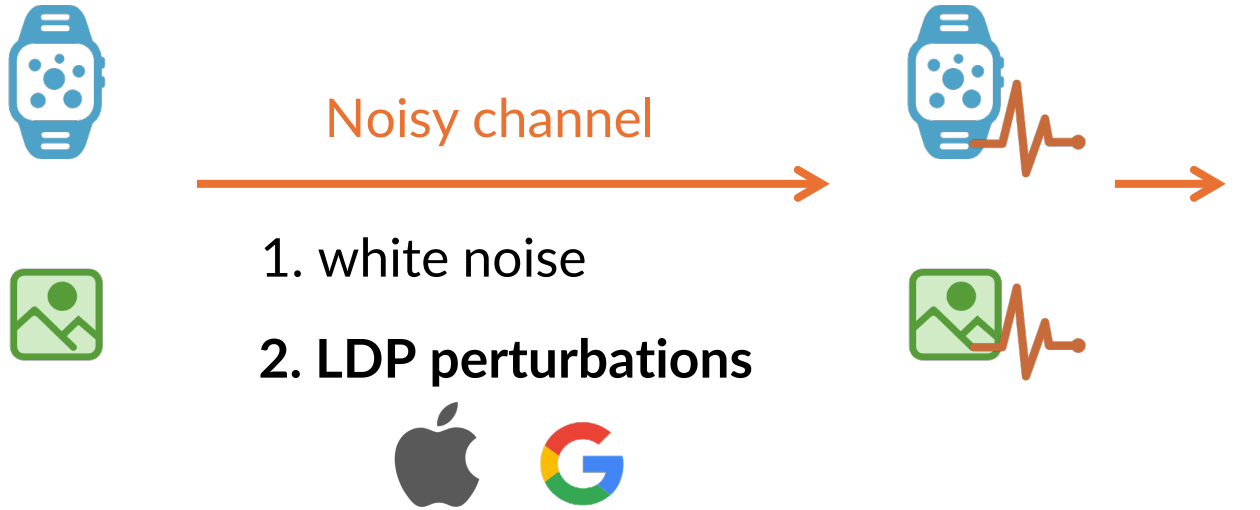
Classifier Utility under Noisy Inputs

- Input data for classifiers may be noisy



Classifier Utility under Noisy Inputs

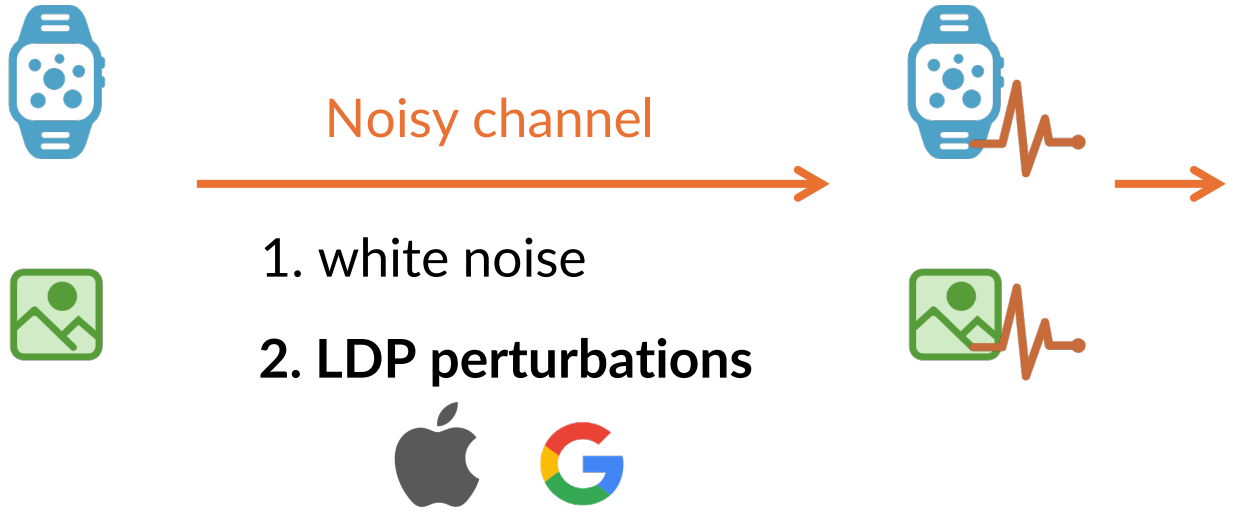
- Input data for classifiers may be noisy



What's the utility of the trained classifier under a noisy channel, e.g. LDP perturbation?

Classifier Utility under Noisy Inputs

- Input data for classifiers may be noisy



What's the utility of the trained classifier under a noisy channel, e.g. LDP perturbation?

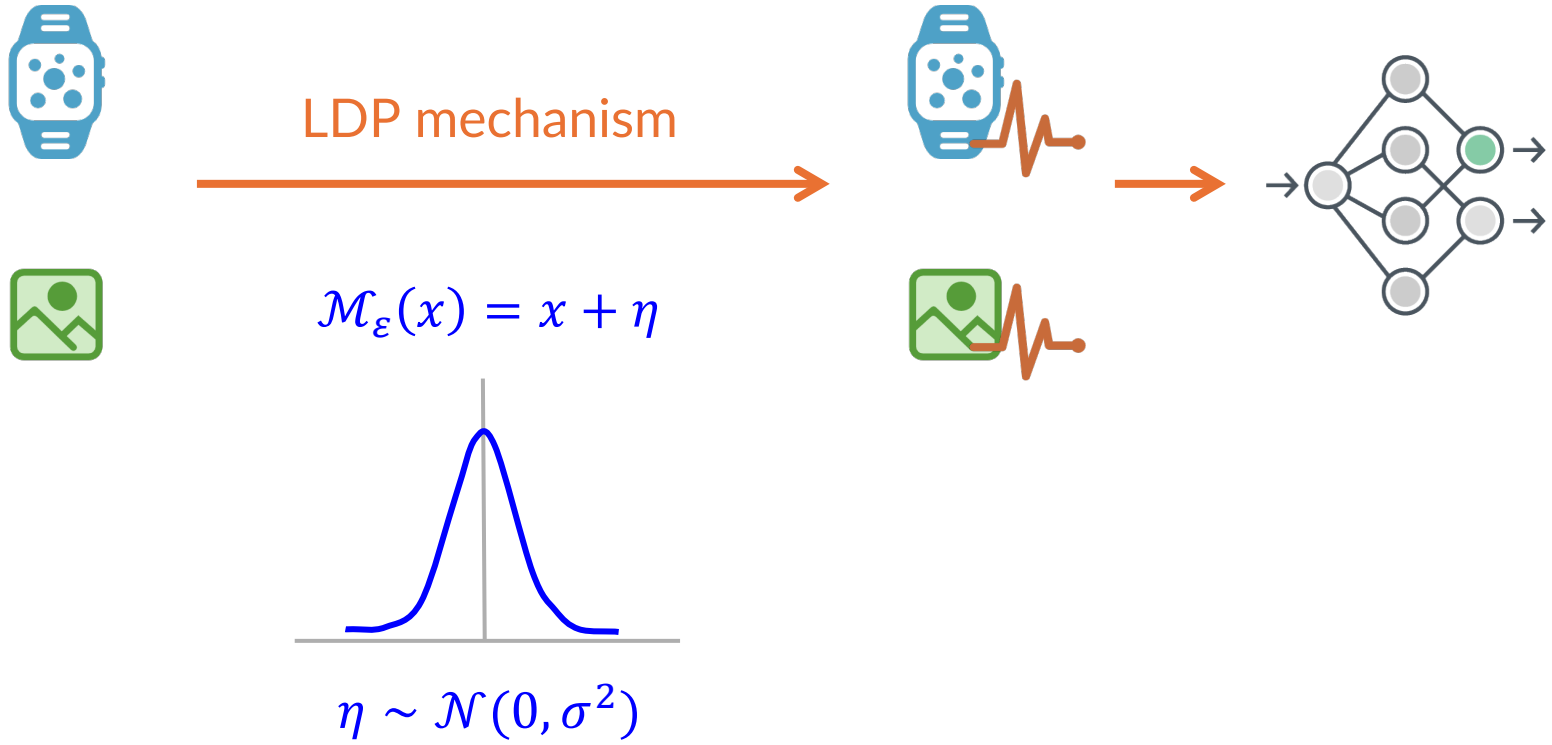
- Q: How can classifier designers/users know the classifier's accuracy under LDP-perturbed data?

For an LDP-friendly classifier

For a better privacy-utility balance

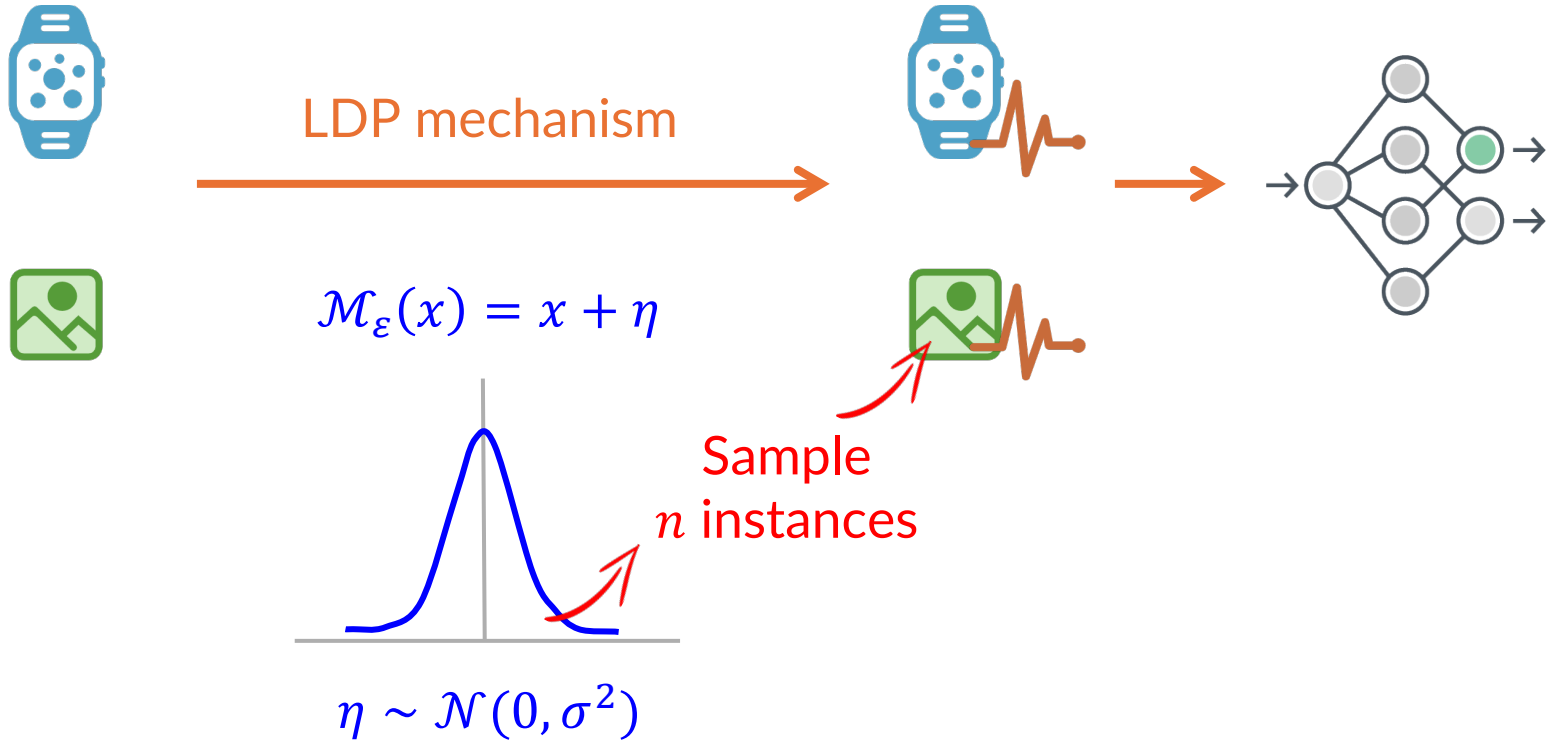
Empirical Classifier Utility under LDP-Data

- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help – cannot provide a classifier accuracy



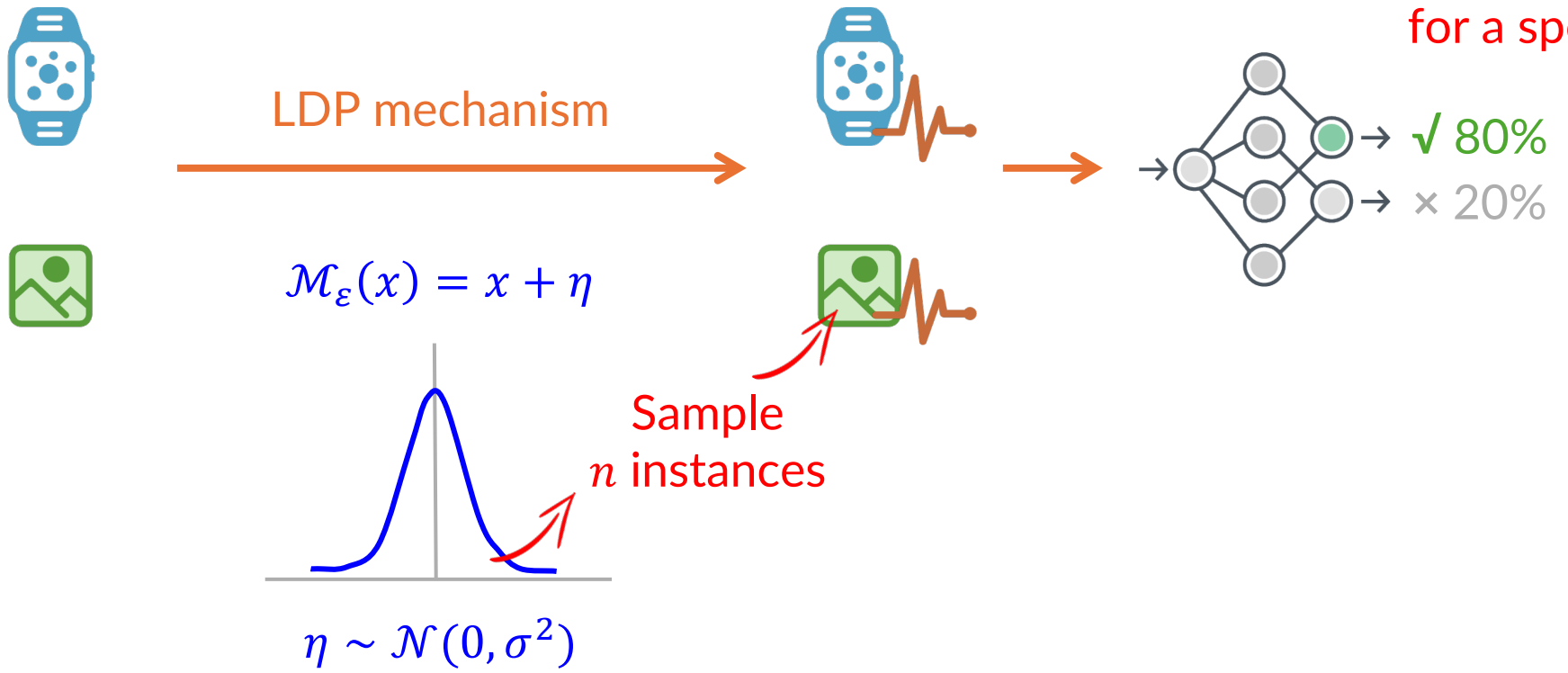
Empirical Classifier Utility under LDP-Data

- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help – cannot provide a classifier accuracy



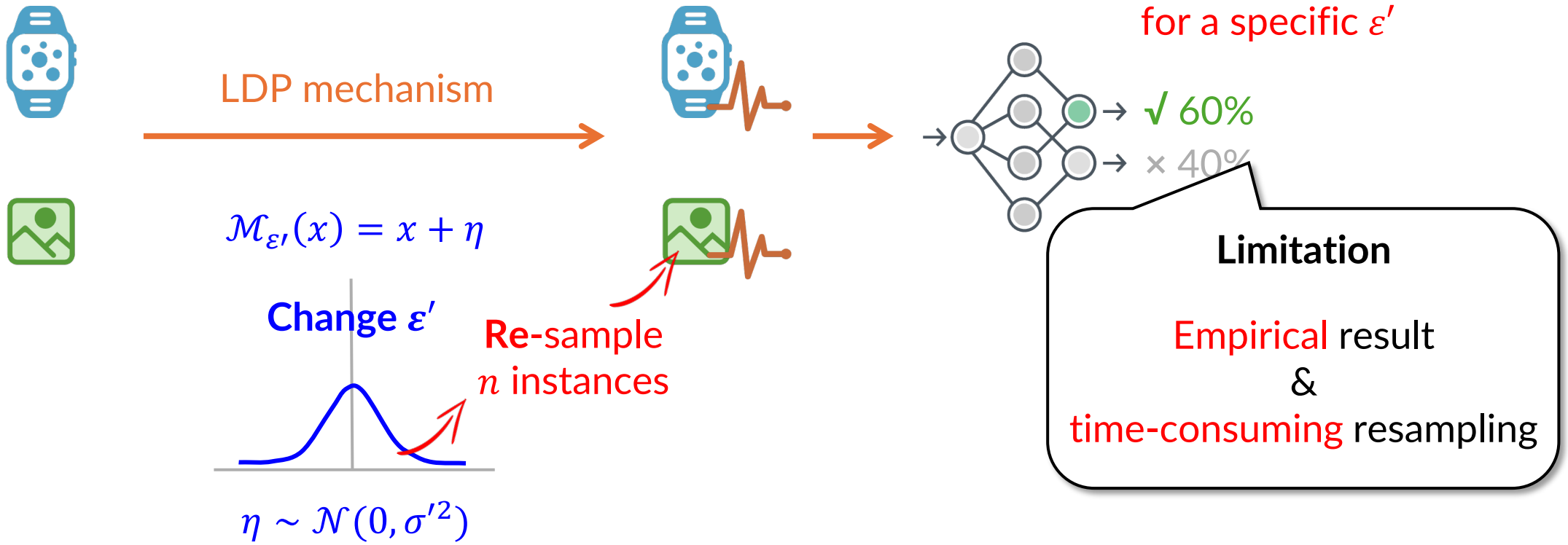
Empirical Classifier Utility under LDP-Data

- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help – cannot provide a classifier accuracy



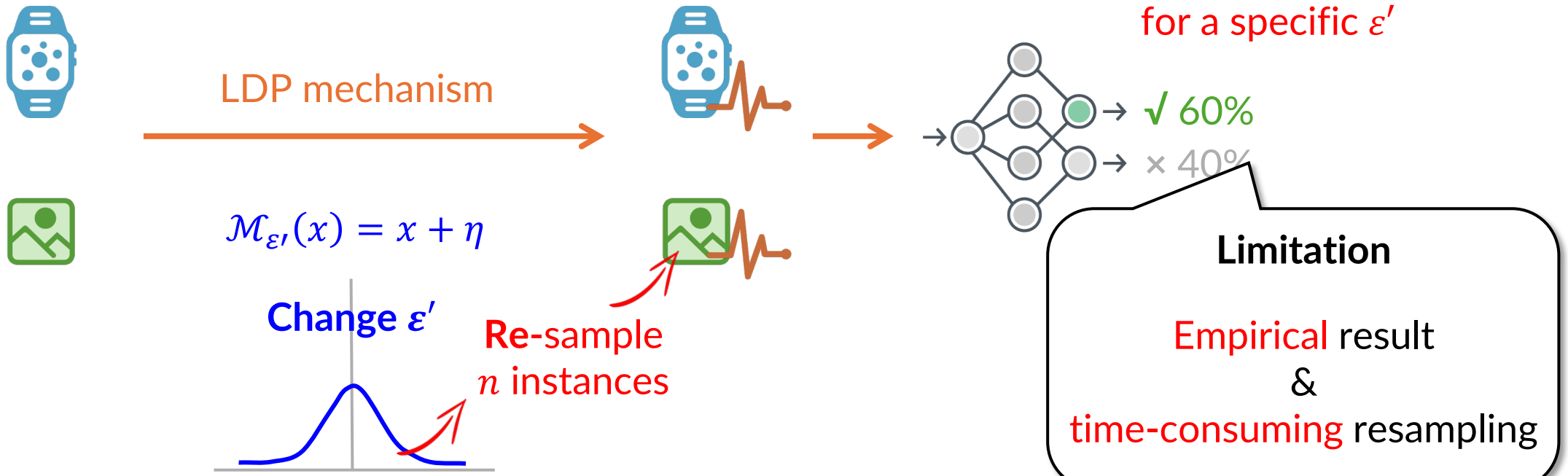
Empirical Classifier Utility under LDP-Data

- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help – cannot provide a classifier accuracy



Empirical Classifier Utility under LDP-Data

- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help – cannot provide a classifier accuracy

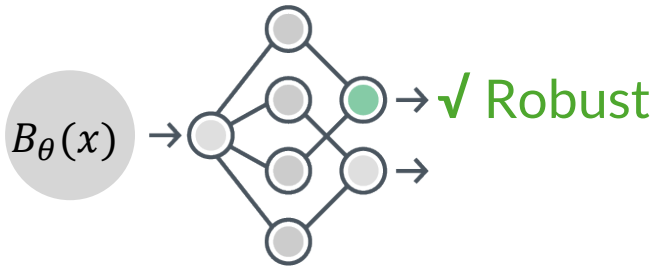


Lacks of an **analytical view** of classifier utility under LDP data

Empirical Utility → Analytical Utility

- Analytical approach: connecting LDP with robustness

$B_\theta(x)$ is robust

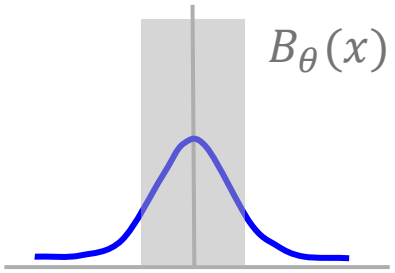


Robustness:
A θ-ball around x that doesn't change classification

Empirical Utility → Analytical Utility

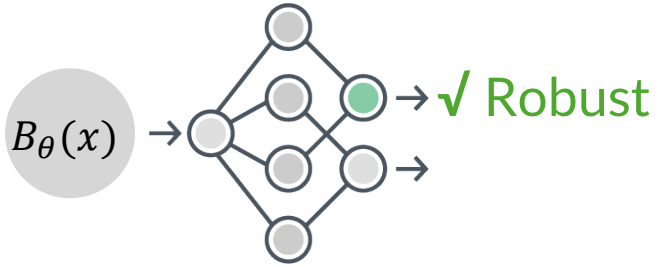
- Analytical approach: connecting LDP with robustness

$\mathcal{M}_\varepsilon(x)$ concentrates in $B_\theta(x)$



Concentration:
Probability of falling into a region
$\Pr[\mathcal{M}_\varepsilon(x) \in B_\theta(x)] \coloneqq p(\varepsilon, \theta)$

$B_\theta(x)$ is robust

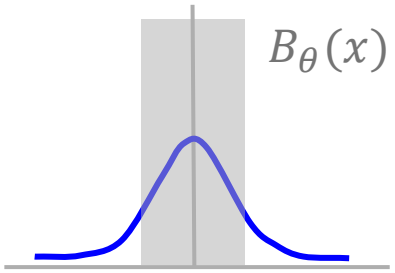


Robustness:
A θ-ball around x that doesn't change classification

Empirical Utility → Analytical Utility

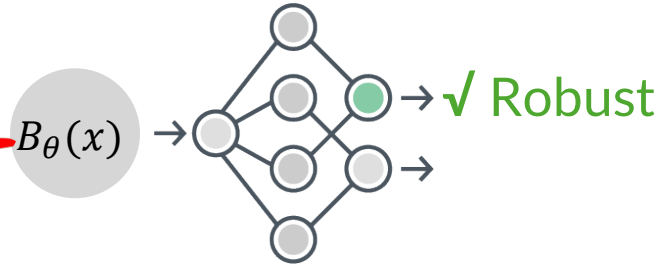
- Analytical approach: connecting LDP with robustness

$\mathcal{M}_\varepsilon(x)$ concentrates in $B_\theta(x)$



Concentration:
Probability of falling into a region
$\Pr[\mathcal{M}_\varepsilon(x) \in B_\theta(x)] \coloneqq p(\varepsilon, \theta)$

$B_\theta(x)$ is robust

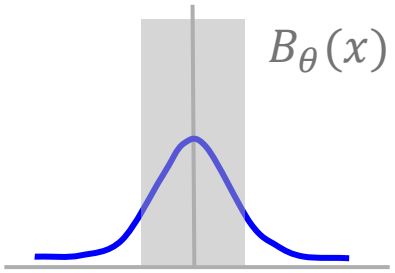


Robustness:
A θ-ball around x that doesn't change classification

Empirical Utility → Analytical Utility

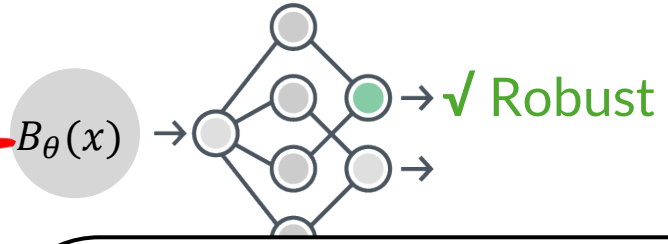
- Analytical approach: connecting LDP with robustness

$\mathcal{M}_\varepsilon(x)$ concentrates in $B_\theta(x)$



Concentration:
Probability of falling into a region
$\Pr[\mathcal{M}_\varepsilon(x) \in B_\theta(x)] \coloneqq p(\varepsilon, \theta)$

$B_\theta(x)$ is robust



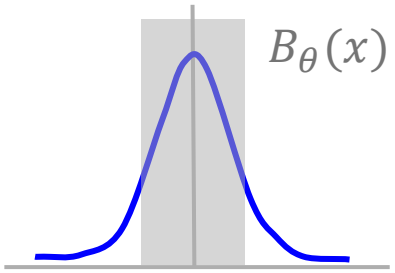
Analytical classifier utility

With probability at least $p(\varepsilon, \theta)$, the classifier preserves its correct classification under input $\mathcal{M}_\varepsilon(x)$.

Empirical Utility → Analytical Utility

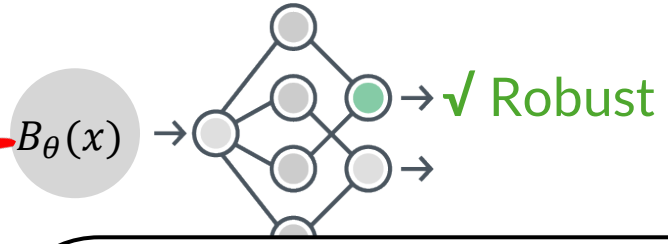
- Analytical approach: connecting LDP with robustness

$\mathcal{M}_{\varepsilon'}(x)$ concentrates in $B_\theta(x)$



Concentration:
Probability of falling into a region
$\Pr[\mathcal{M}_{\varepsilon'}(x) \in B_\theta(x)] \coloneqq p(\varepsilon', \theta)$

$B_\theta(x)$ is robust



Analytical classifier utility

With probability at least $p(\varepsilon', \theta)$, the classifier preserves its correct classification under input $\mathcal{M}_{\varepsilon'}(x)$.

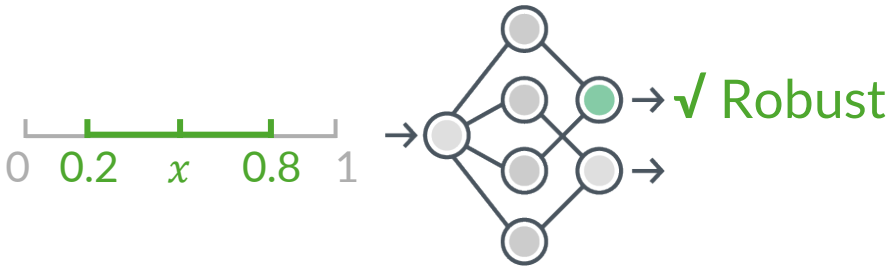
Analytical & systematic view for any ε without resampling

One-Dimension Example

- Classifier h: $[0,1] \to \{1, 2\}$ under Laplace mechanism $\mathcal{M}_{\mathrm{Lap}}(x)$

$B_{0.3}(x)$ is robust

Classifier:
(robustness analysis)

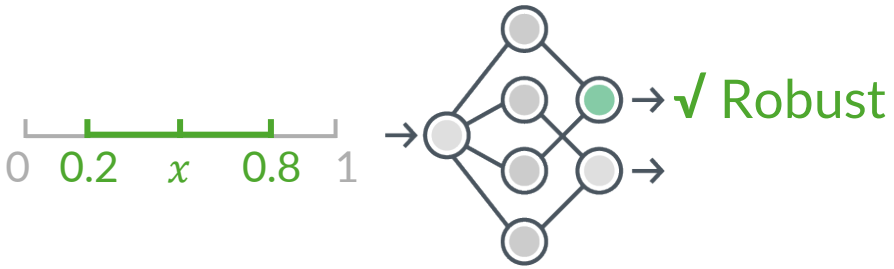


One-Dimension Example

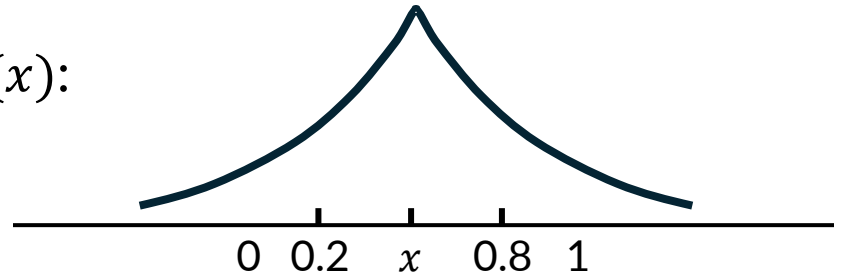
- Classifier h: $[0,1] \to \{1, 2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$

$B_{0.3}(x)$ is robust

Classifier:
(robustness analysis)

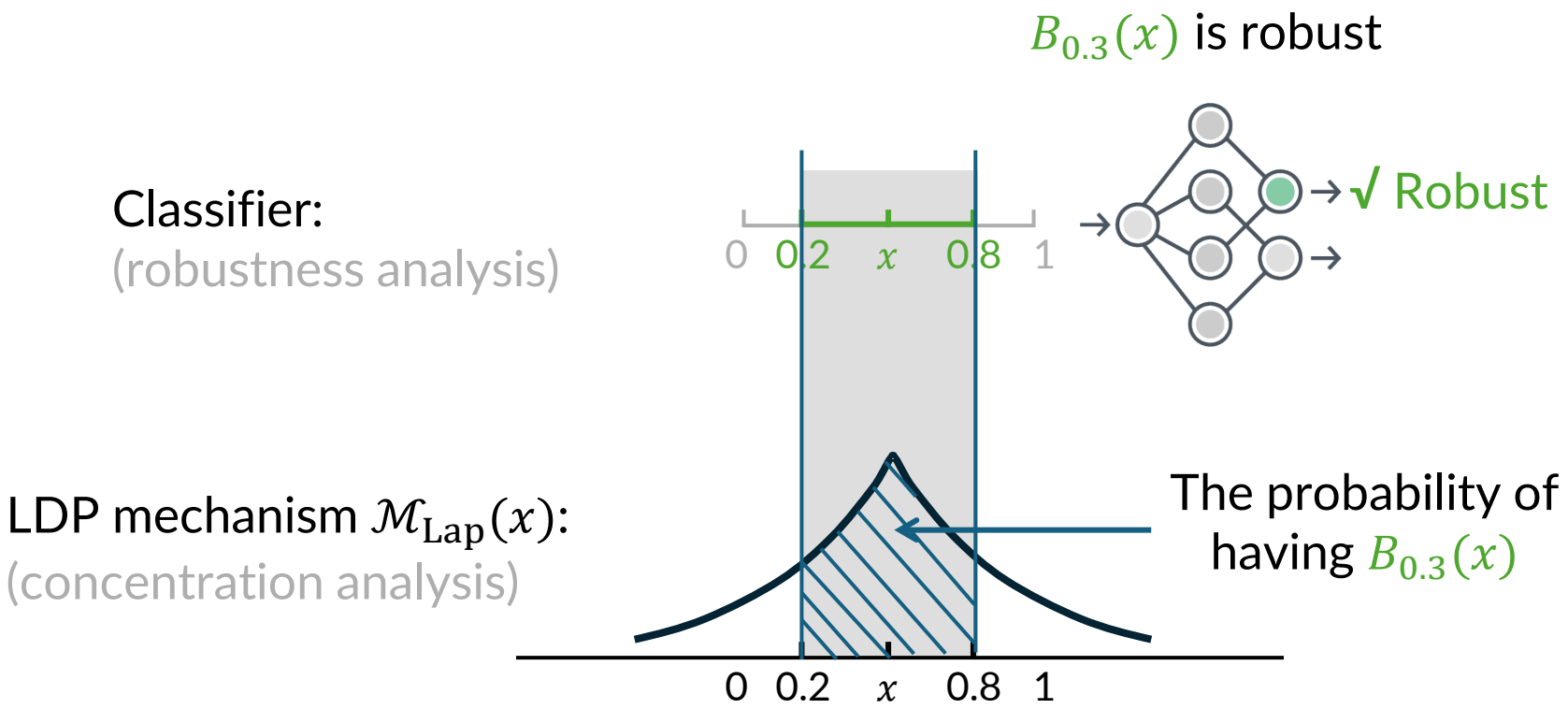


LDP mechanism $\mathcal{M}_{\text{Lap}}(x)$:



One-Dimension Example

- Classifier h: $[0,1] \to \{1, 2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$

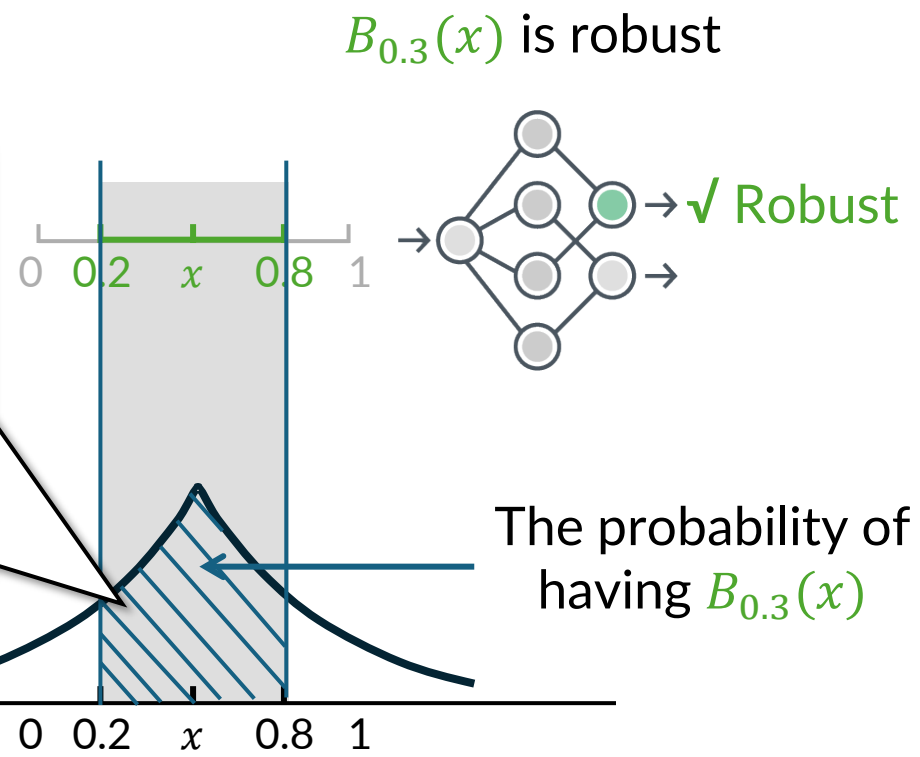


One-Dimension Example

- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\mathrm{Lap}}(x)$

Concentration analysis

$$p(\varepsilon,\theta) = \mathrm{cdf}_{\mathrm{Lap}}[\theta] - \mathrm{cdf}_{\mathrm{Lap}}[-\theta]$$
$$= 1 - e^{-\theta\varepsilon}$$



One-Dimension Example

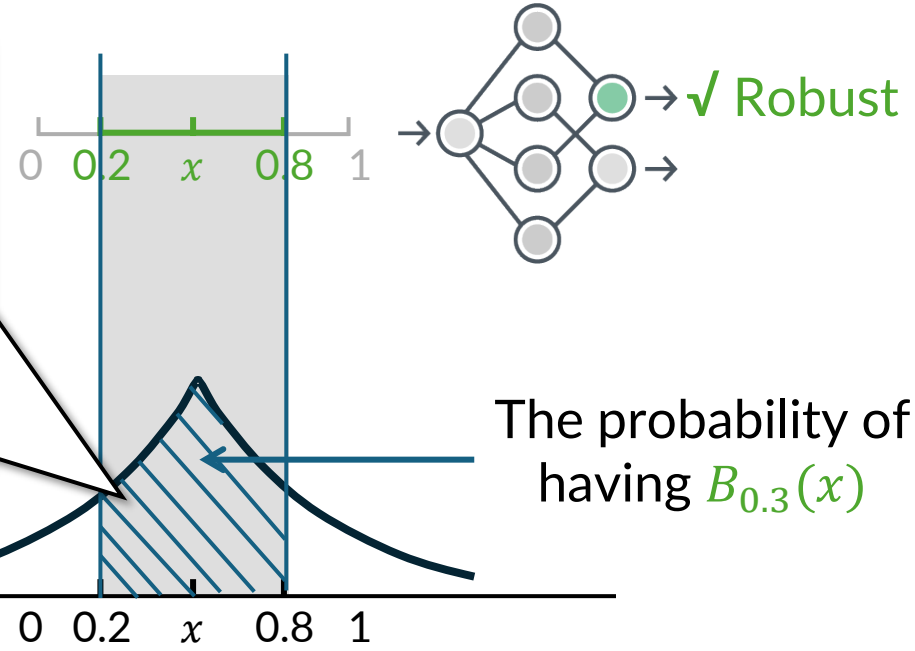
- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\mathrm{Lap}}(x)$

Concentration analysis

$$p(\varepsilon,\theta) = \mathrm{cdf}_{\mathrm{Lap}}[\theta] - \mathrm{cdf}_{\mathrm{Lap}}[-\theta]$$
$$= 1 - e^{-\theta\varepsilon}$$

$$p(\varepsilon = 2, \theta = 0.3) = 0.46$$

$B_{0.3}(x)$ is robust



One-Dimension Example

- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$

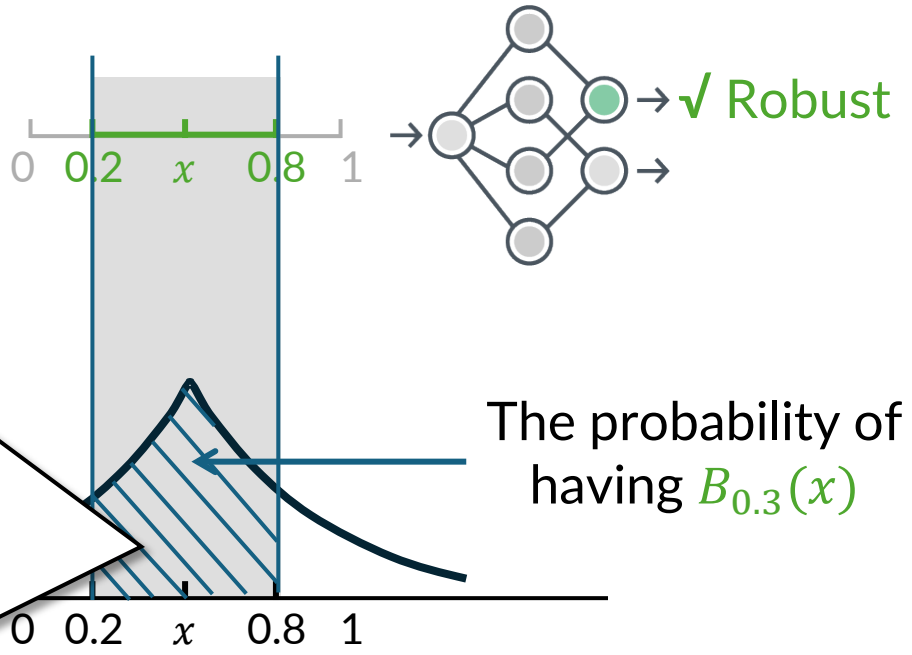
Concentration analysis

$$p(\varepsilon,\theta) = \text{cdf}_{\text{Lap}}[\theta] - \text{cdf}_{\text{Lap}}[-\theta]$$
$$= 1 - e^{-\theta\varepsilon}$$

$$p(\varepsilon = 2, \theta = 0.3) = 0.46$$

With probability at least $p(2,0.3) = 0.46$, the classifier preserves its correct classification under input $\mathcal{M}_{\varepsilon=2}(0.5)$.

$B_{0.3}(x)$ is robust



One-Dimension Example

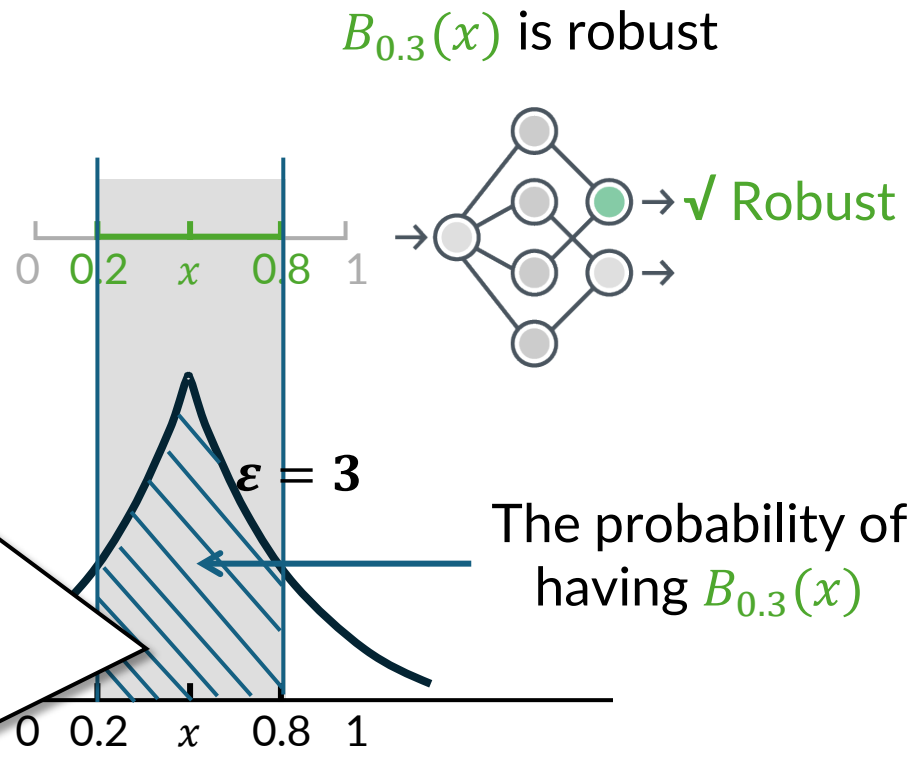
- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\mathrm{Lap}}(x)$

Concentration analysis

$$p(\varepsilon,\theta) = \mathrm{cdf}_{\mathrm{Lap}}[\theta] - \mathrm{cdf}_{\mathrm{Lap}}[-\theta]$$
$$= 1 - e^{-\theta\varepsilon}$$

For any ε:

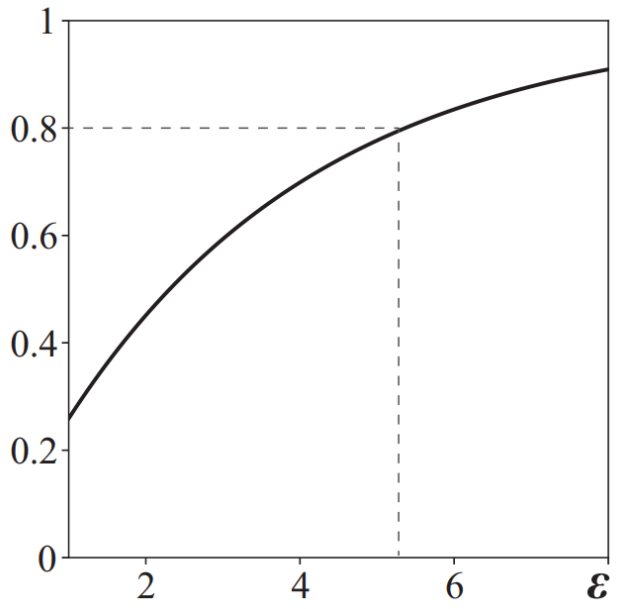
With probability at least $p(\varepsilon,\theta)$, the classifier preserves its correct classification under input $\mathcal{M}_{\varepsilon}(x)$.



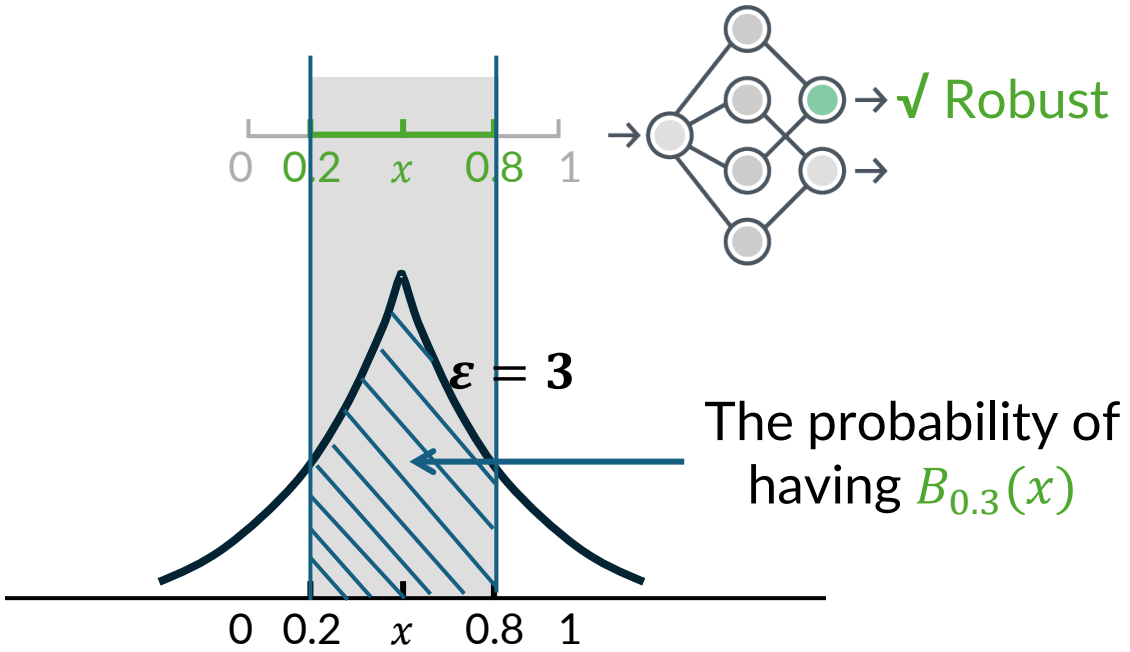
One-Dimension Example

- Classifier h: $[0,1] \to \{1, 2\}$ under Laplace mechanism $\mathcal{M}_{\mathrm{Lap}}(x)$

This classifier's utility $p(\varepsilon, 0.3)$



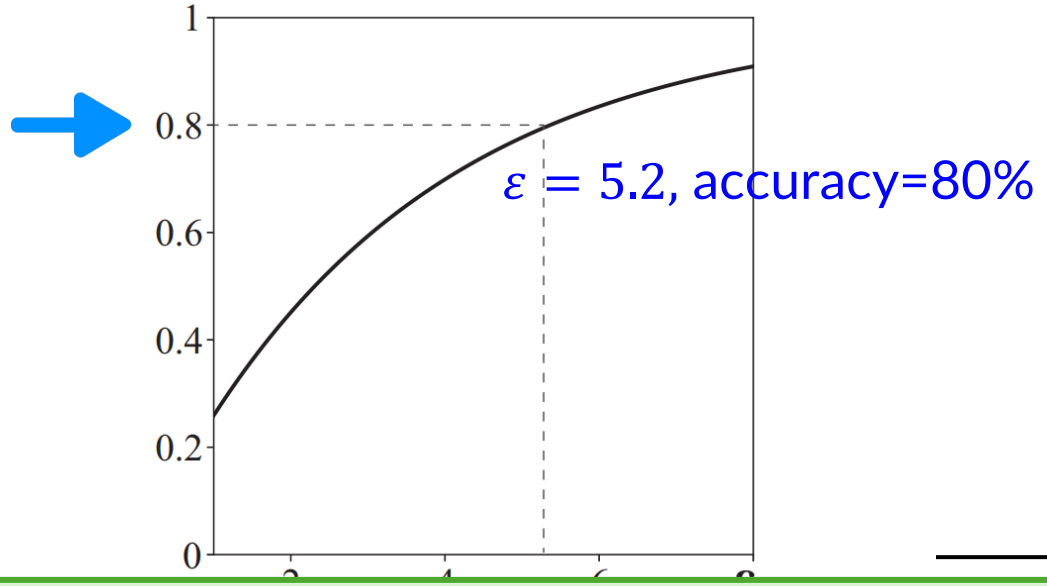
$B_{0.3}(x)$ is robust



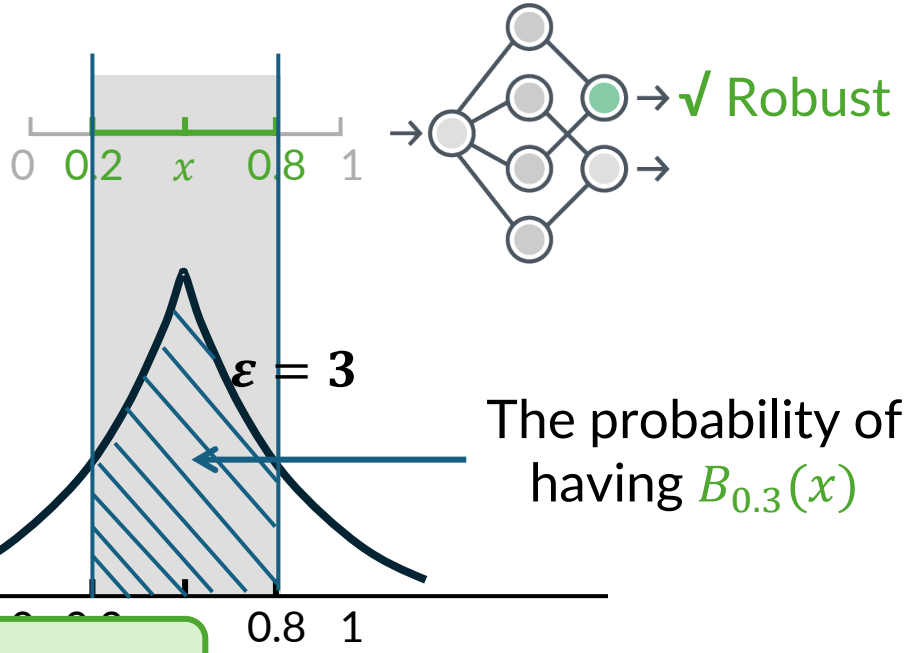
Fixed Classifier (θ)

- Classifier h: $[0,1] \to \{1, 2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$

This classifier's utility $p(\varepsilon, 0.3)$



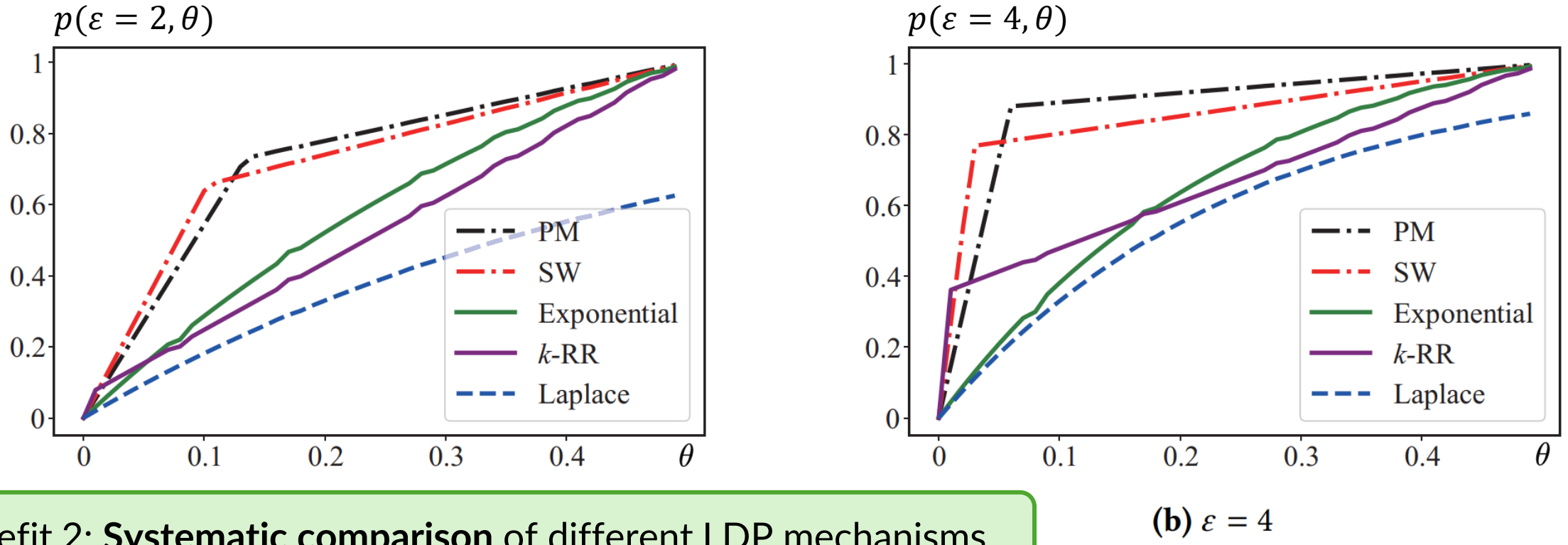
$B_{0.3}(x)$ is robust



Benefit 1: Choose the **best ε** for a desired classifier utility

Different $\mathcal{M}$ & Different Classifiers

- No universally optimal LDP mechanism for all ε and θ



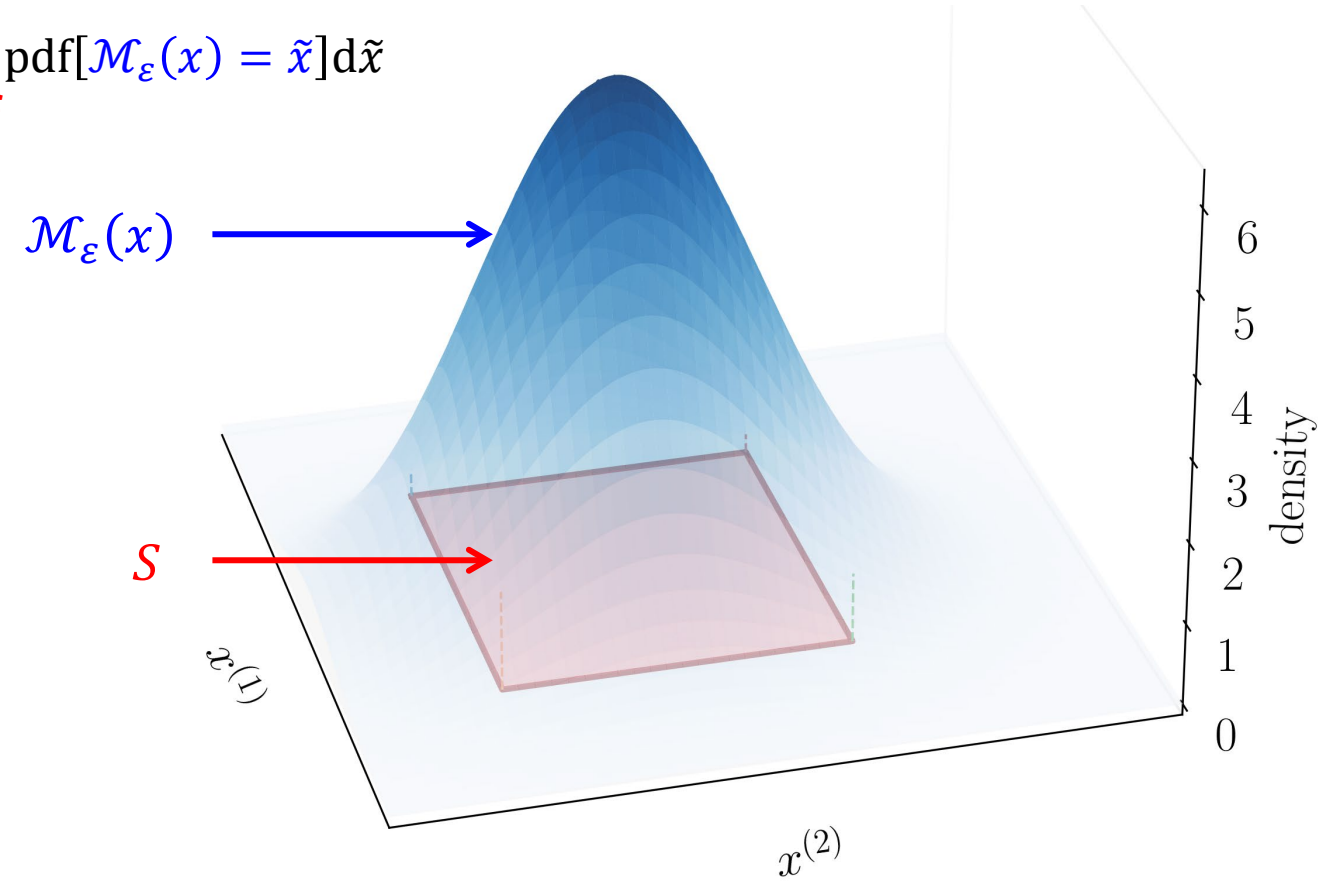
(b) $\varepsilon = 4$

Benefit 2: **Systematic comparison** of different LDP mechanisms

High-Dimensional Cases

- Concentration analysis on the robustness region S

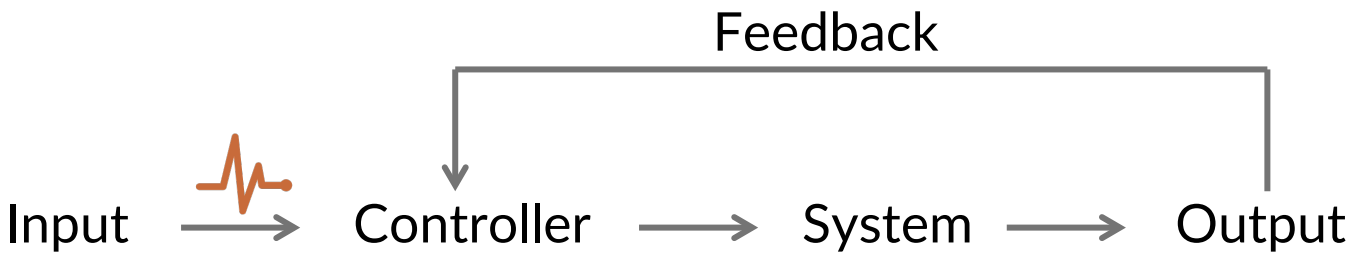
$$p(\varepsilon, S) = \int_S \mathrm{pdf}[\mathcal{M}_\varepsilon(x) = \tilde{x}]\mathrm{d}\tilde{x}$$



Beyond Classifier – Robustness in Other Systems

- **Communication systems**

- **Control systems**



Case Studies – Quality of The Analytical Utility

- Claim: Accurate (compared with the empirical utility) & efficient

Case Studies – Quality of The Analytical Utility

- Claim: Accurate (compared with the empirical utility) & efficient
- Classifiers:
  - low-dimensional: Logistic Regression, Random Forest
  - high-dimensional: Neural Network

Case Studies – Quality of The Analytical Utility

- Claim: Accurate (compared with the empirical utility) & efficient
- Classifiers:
  - low-dimensional: Logistic Regression, Random Forest
  - high-dimensional: Neural Network
- Datasets: Stroke Prediction (medical data), Bank Customer Attrition (financial data), MNIST-7×7 (image data)

2 sensitive dims

49 sensitive dims

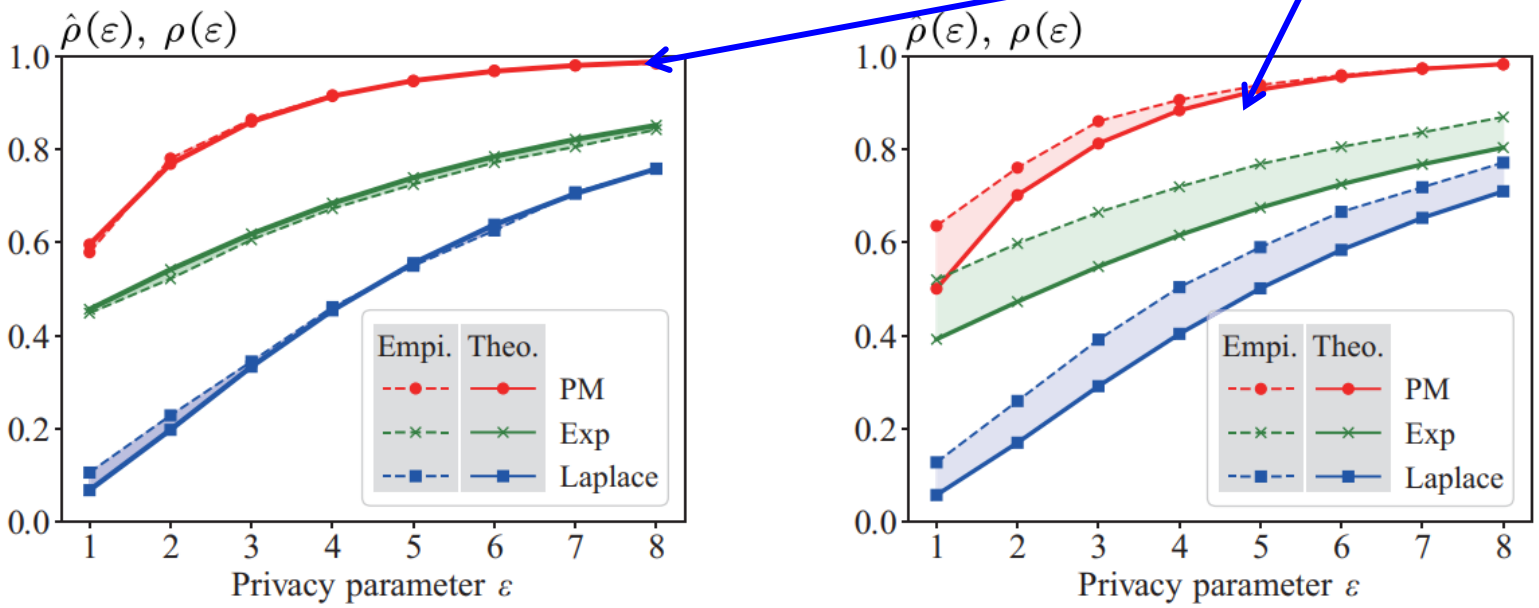
Case Studies – Quality of The Analytical Utility

- Claim: Accurate (compared with the empirical utility) & efficient
- Classifiers:
  - low-dimensional: Logistic Regression, Random Forest
  - high-dimensional: Neural Network
- Datasets: Stroke Prediction (medical data), Bank Customer Attrition (financial data), MNIST-7×7 (image data)
- Classifier utility:
  - analytical utility $p(\varepsilon, S)$: approximated S for black-box classifiers
  - empirical utility $\hat{p}(\varepsilon)$: 2000 samples from $\mathcal{M}_\varepsilon$ and then used for testing

Case Studies – Stoke Predication

- $x =$ the first record, with noisy "Age" and "BMI"

Accurate analytical utility



(a) Logistic Regression. **(b) Random Forest.**

Case Studies – Stoke Predication

- $x =$ the first record, with noisy "Age" and "BMI"

Time cost comparison (ms)

	PM	Exponential	Laplace
Empirical[a]	$6.56 + 1.38$	$859.83 + 1.53$	$11.29 + 1.53$
Theoretical[b]	0.24	0.94	0.30

←Negligible time cost

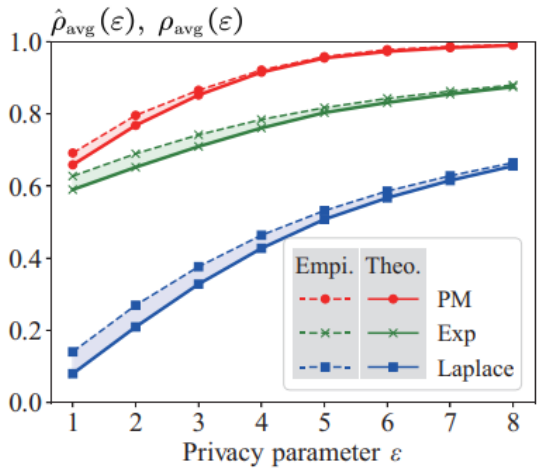
[a] Time of 2000 samples + inference.

[b] Time to compute $\rho(\varepsilon, S)$ only; computing S takes 5.80 ms but is a one-time cost amortized across all ε values.

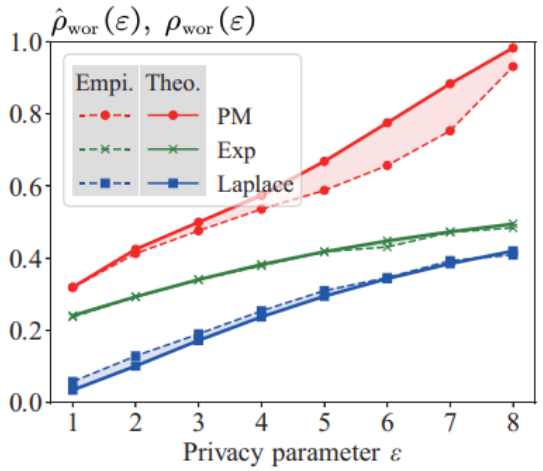
Case Studies – Stoke Predication

- **Average-case and worst-case utility**

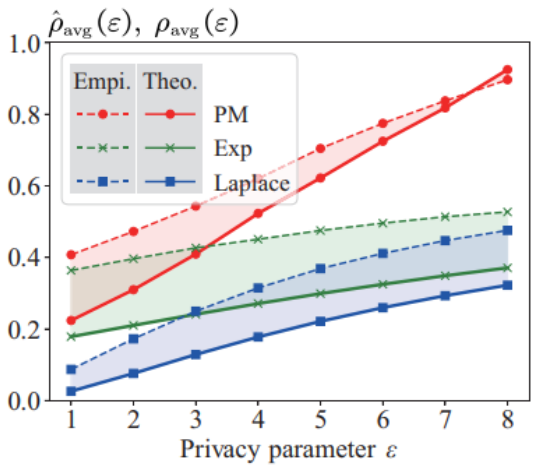
$$x \sim P_x \quad \rightarrow \begin{cases} p_{\text{avg}}(\varepsilon) = \mathrm{E}_{x \sim P_x}[p_x(\varepsilon, S)] \\ p_{\text{wor}}(\varepsilon) = \min\limits_{x \sim P_x}[p_x(\varepsilon, S)] \end{cases}$$



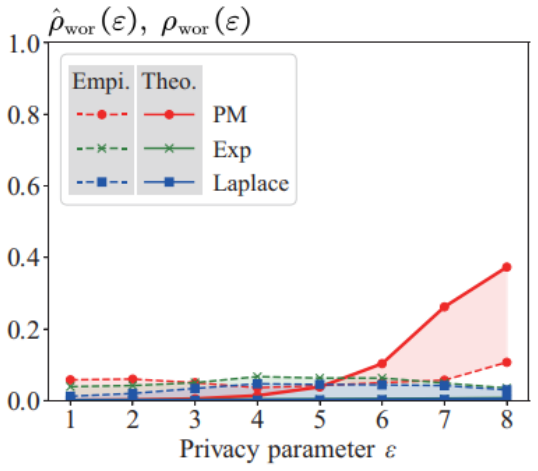
(a) LR: Average-case utility.



(b) LR: Worst-case utility.



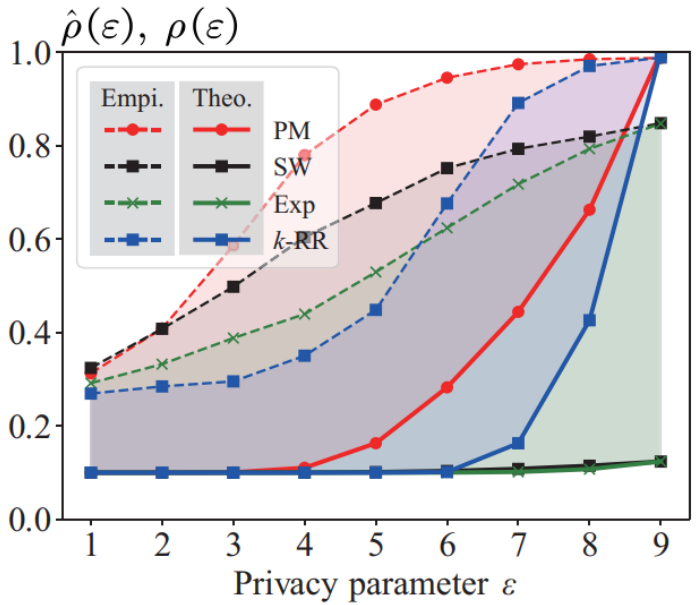
(a) RF: Average-case utility.



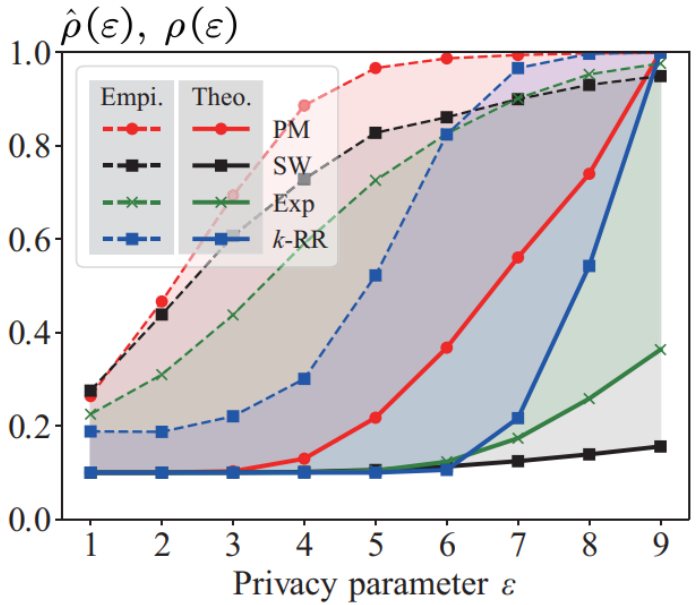
(b) RF: Worst-case utility.

Case Studies – Neural Networks (49-dim)

- Conservative analytical utility for high-dim classifiers
  - but high analytical utility → high empirical utility



1st image

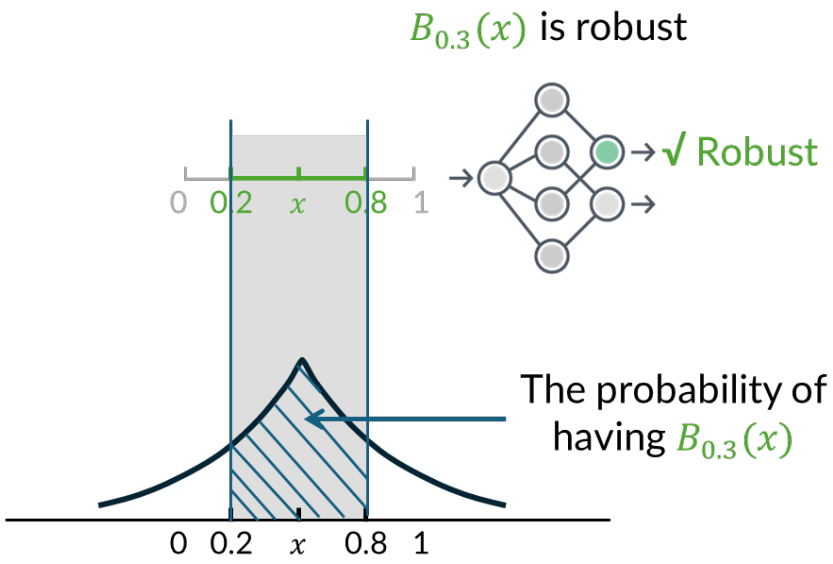


2nd image

Summary & Takeaway

- **RQ: Classifier utility under LDP-perturbed inputs**

- This paper:
  - **connects LDP with robustness**
  - **provides analytical classifier utility**
  - the analytical utility is accurate for low-dim classifiers
  - the analytical utility is also useful for high-dim classifiers



Quantifying Classifier Utility under LDP

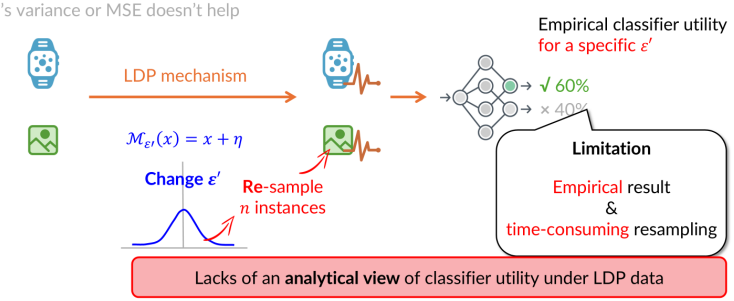
Classifier Utility under Noisy Inputs

- Input data for classifiers may be noisy

- Q: How can classifier designers/users know the classifier's accuracy under LDP-perturbed data?
  - For an LDP-friendly classifier
  - For a better privacy-utility balance

Empirical Classifier Utility under LDP-Data

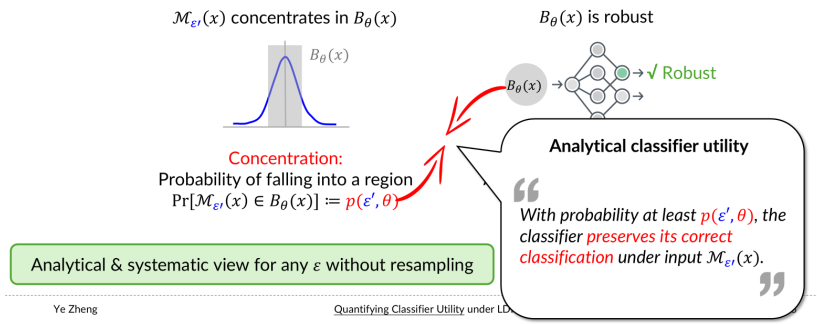
- Empirical approach: Sample and then test
  - $\mathcal{M}$'s variance or MSE doesn't help



Lacks of an **analytical view** of classifier utility under LDP data

Empirical Utility → Analytical Utility

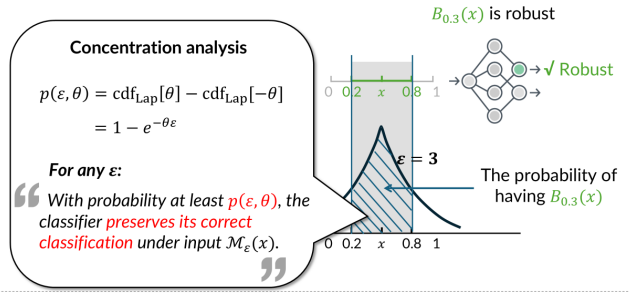
- Analytical approach: connecting LDP with robustness



Analytical & systematic view for any ε without resampling

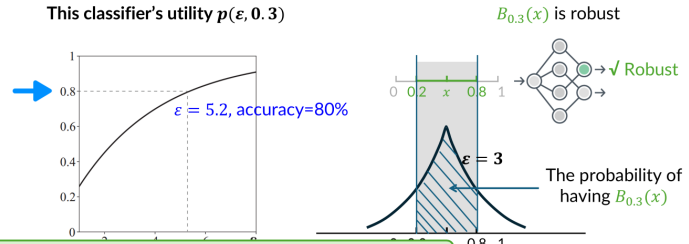
One-Dimension Example

- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$



Fixed Classifier (θ)

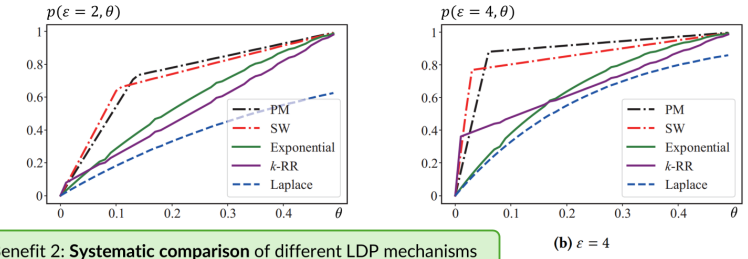
- Classifier h: $[0,1] \to \{1,2\}$ under Laplace mechanism $\mathcal{M}_{\text{Lap}}(x)$



Benefit 1: Choose the **best** ε for a desired classifier utility

Different $\mathcal{M}$ & Different Classifiers

- No universally optimal LDP mechanism for all ε and θ



Benefit 2: **Systematic comparison** of different LDP mechanisms

(b) $\varepsilon = 4$

Thank you!