Locally Differentially Private Frequency Estimation Exploiting Consistency

Tianhao Wang

Purdue University

Joint work with Milan Lopuhaä-Zwakenberg, Zitao Li, Boris Skoric, Ninghui Li


SAMSUNG

Privacy in Practice

- Local differential privacy is deployed
  - In Google Chrome browser, to collect browsing statistics
  - In Apple iOS and MacOS, to collect typing statistics
  - In Microsoft Windows, to collect telemetry data over time
  - In Alibaba, we built a system to collect user transaction info

- Different algorithms are proposed.
- They work for different tasks and different settings.
- They are all based on *Randomized Response*.

Randomized Response

- Survey technique for private questions
- Survey people:
  - "Do you have disease X?"
- Each person:
  - Flip a secret coin 
  - Answer truth if **head** (w.p. 0.5)
  - Answer randomly if **tail** (w.p. 0.5):
    - reply "yes"/"no" w.p. 0.5

$$\Pr[\text{disease} \to \text{yes}]$$
$$= \Pr[\text{disease} \to \text{yes} \wedge \textbf{head}]$$
$$+ \Pr[\text{disease} \to \text{yes} \wedge \textbf{tail}]$$
$$= \mathbf{0.5} \times 1 + \mathbf{0.5} \times 0.5 = 0.75$$

Similarly:

$$\Pr[\text{disease} \to \text{no}] = 0.25$$
$$\Pr[\text{no disease} \to \text{yes}] = 0.25$$
$$\Pr[\text{no disease} \to \text{no}] = 0.75$$

S L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. JASA. 1965.

Randomized Response

$\Pr[\text{disease} \rightarrow \text{yes}] = 0.75$
$\Pr[\text{disease} \rightarrow \text{no}] = 0.25$
$\Pr[\text{no disease} \rightarrow \text{no}] = 0.25$
$\Pr[\text{no disease} \rightarrow \text{yes}] = 0.75$

- To estimate the distribution:
- If n_{yes} out of n … see:
  … answers
- Inverting the a…
- It is the unbias… s

$$\mathrm{E}[\hat{n}_{\text{yes}}] = \frac{\ldots}{0.5} = n_{\text{yes}}$$

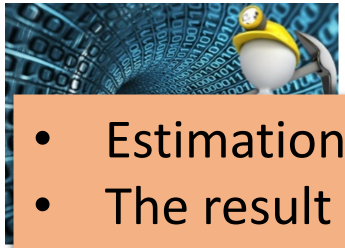
- Similar for the "no"

An algorithm A is ε -LDP if and only if for any v and v', and any valid output y,

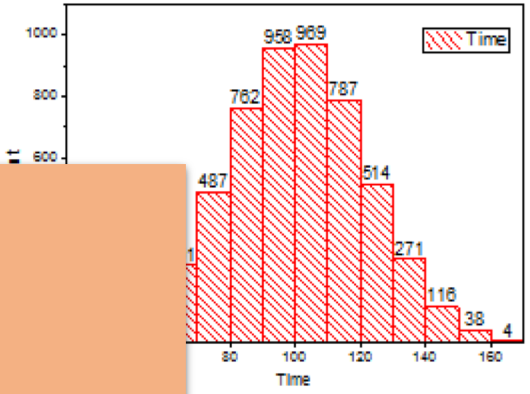
$$\frac{\Pr[\mathrm{A}(v)=y]}{\Pr[\mathrm{A}(v')=y]} \leq e^{\varepsilon}$$

Enumerating possibilities of v and v' taking disease or no disease, and y as yes or no, the binary randomized response is $ln3$-LDP.

Local Differential Privacy (LDP)



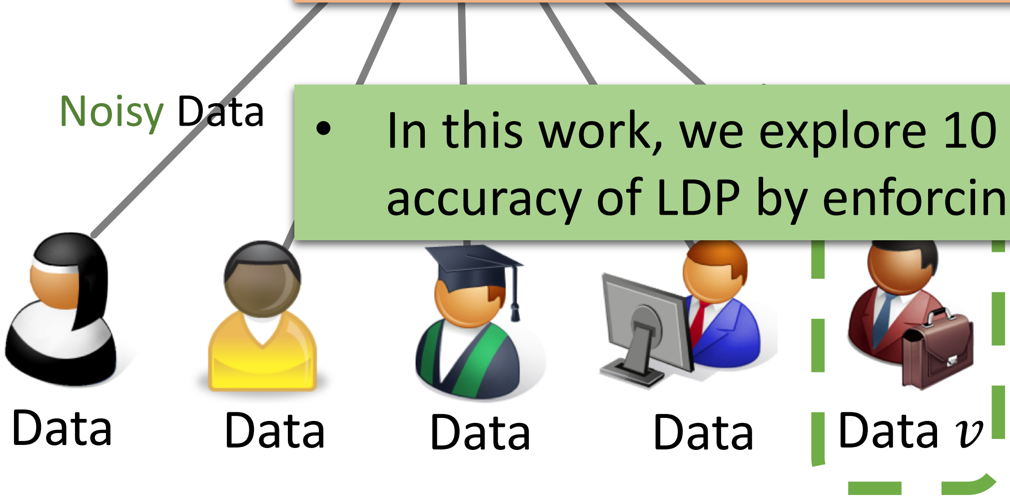
takes reports from all users and outputs



- Estimation function is done independent for each value v.
- The result is not consistent.
  - Some may be negative.
  - Sum may not be n (the original number of users).

Noisy Data

- In this work, we explore 10 different methods that improves the accuracy of LDP by enforcing consistency.



- $y = A(v)$ takes input value v and outputs y.

...or any v and v', and any valid output y,

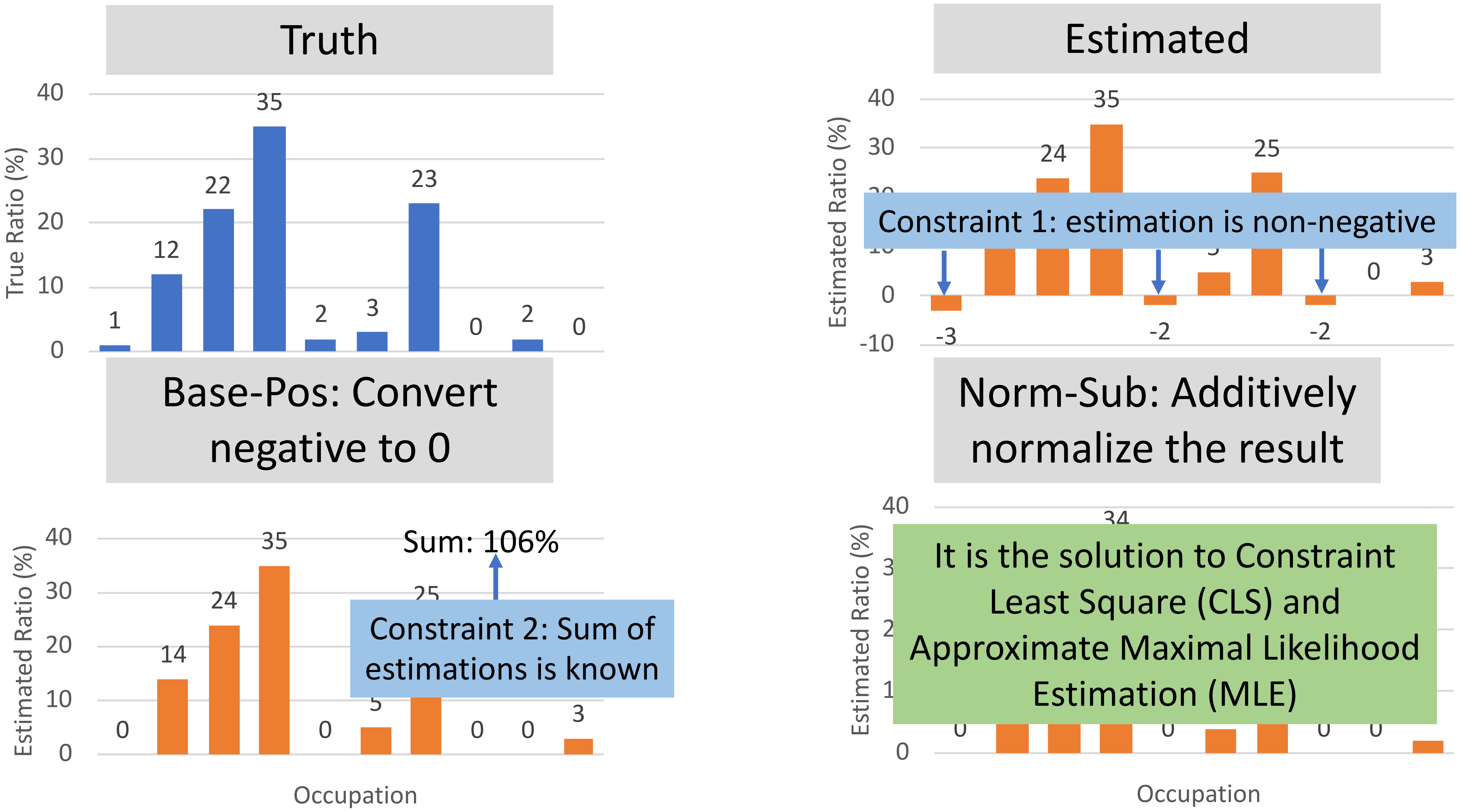
$$\frac{\Pr[A(v)=y]}{\Pr[A(v')=y]} \leq e^{\varepsilon}$$

Making Estimations Consistent

1) The estimated frequency of each value is non-negative.
2) The sum of the estimated frequencies is 1.

Group	Method	Description	Non-neg	Sum to 1	Complexity
Several Baselines	Base	Use existing estimation	No	No	N/A
	Base-Pos	Convert negative est. to 0	Yes	No	$O(d)$
	Post-Pos	Convert negative query result to 0	Yes	No	N/A
	Base-Cut	Convert est. below threshold T to 0	Yes	No	$O(d)$
Normalization-based Methods	Norm	Add δ to est.	No	Yes	$O(d)$
	Norm-Mul	Convert negative est. to 0, then multiply ϒ to positive est.	Yes	Yes	$O(d)$
	Norm-Cut	Convert negative and small positive est. below ϑ to 0	Yes	Almost	$O(d)$
	Norm-Sub	Convert negative est. to 0 while adding δ to positive est.	Yes	Yes	$O(d)$
MLE-based	MLE-Apx	Convert negative est. to 0, then add δ to positive est.	Yes	Yes	$O(d)$
Needs More Prior	Power	Fit Power-Law dist., then minimize expected squared error.	Yes	No	$O(\sqrt{n}d)$
	PowerNS	Apply Norm-Sub after Power	Yes	Yes	$O(\sqrt{n}d)$

Post-Processing: Toy Example



Analysis of the Estimation in LDP

- Estimation function
  - $\hat{n}_{\text{yes}} = \frac{I_{\text{yes}} - 0.25n}{0.5}$, more generally $\hat{n}_{\text{v}} = \frac{I_{\text{v}} - qn}{p - q}$

probability of $\text{A}(v)$ supporting v
(disease $\rightarrow$ yes)

probability of $\text{A}(v')$ supporting v
where $v' \neq v$
… $\rightarrow$ yes)

Takeaway: The noise of the LDP estimation approximately follows Gaussian distribution.

- Noise comes from … Binomials
  - $\text{Bin}(nv, p) + \text{Bin}(n - nv, q) = \text{Bin}\left(n, \frac{n_v}{n} p + \ldots\right)$

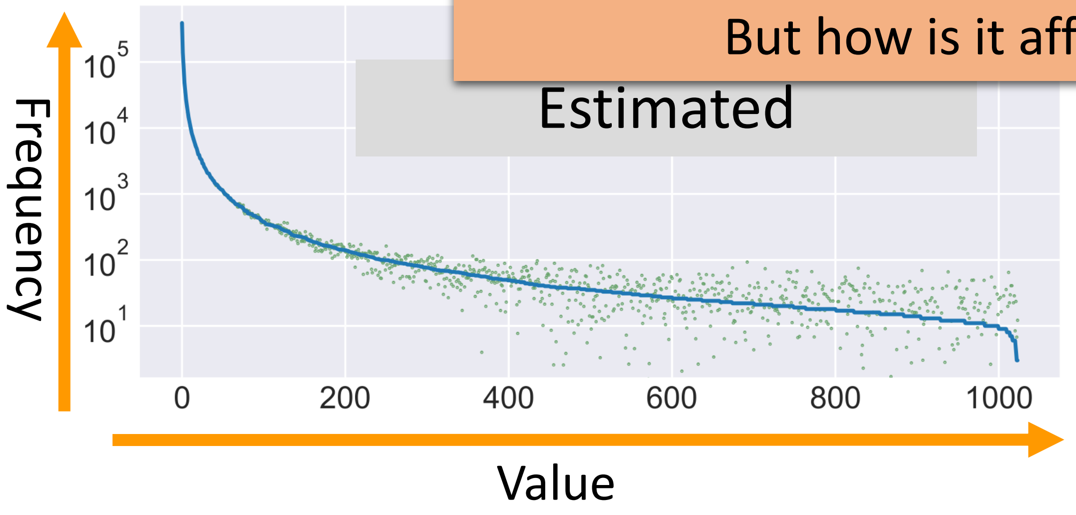
This makes the analysis easier (Norm-Sub is solution to MLE).

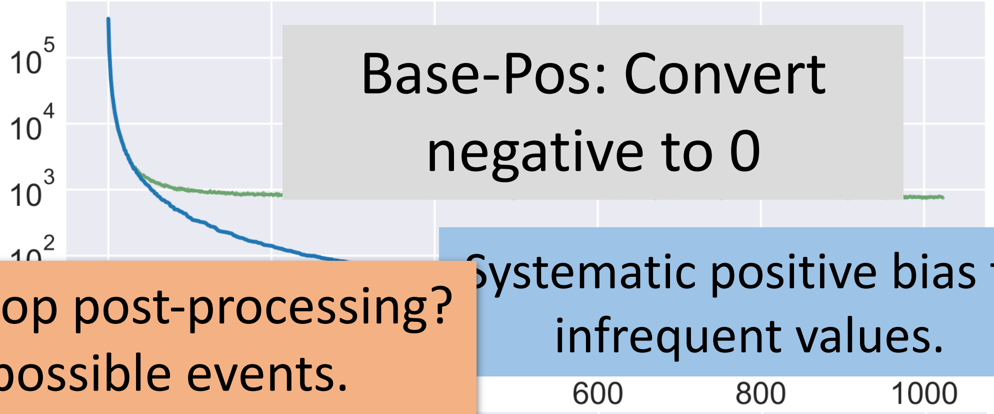
  - When n is large, noise $\approx N(p'n, \sqrt{np'(1 - p')})$ for $p' = \frac{n_v}{n} p + \frac{n - nv}{n} q$

J, Jia, and N. Gong. Calibrate: Frequency estimation and heavy hitter identification with local differential privacy via incorporating prior knowledge. *INFOCOM 2019*.

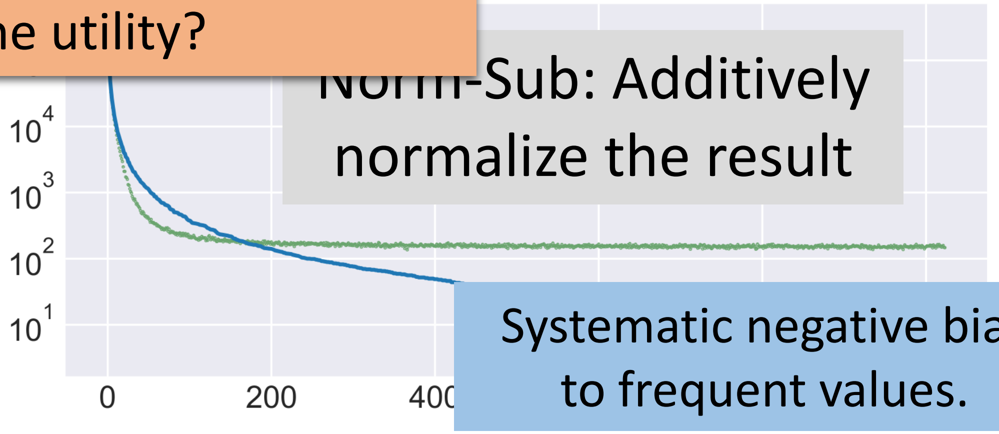
Empirical Understanding

- 1 million reports following Zipf's distribution (s=1.5) with 1024 values.
- 5000 runs (each dot is the mean)





Systematic positive bias to infrequent values.



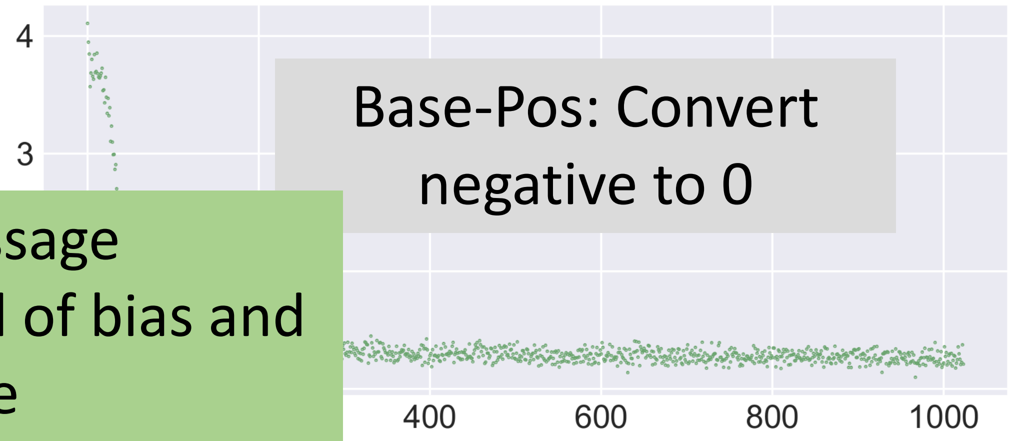
Systematic negative bias to frequent values.

Bias is a bad thing. Should we stop post-processing?
No, because it prevents impossible events.
But how is it affect the utility?

Empirical Understanding

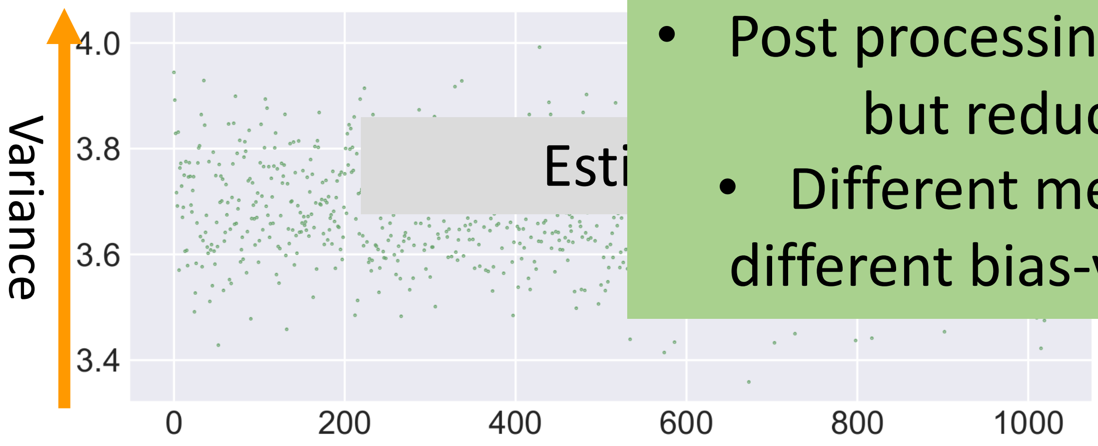
Variance is smaller for infrequent values.

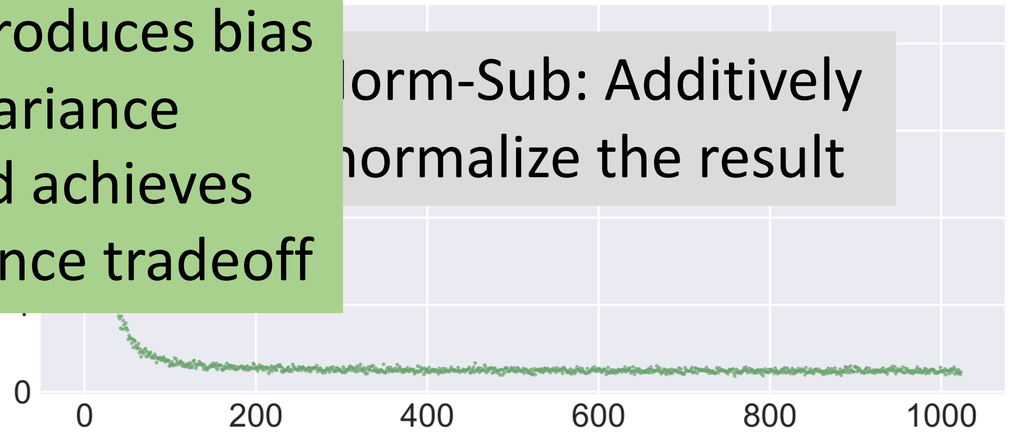
- 1 million reports following Zipf's distribution (s=1.5) with 1024 values.
- 5000 runs (each do



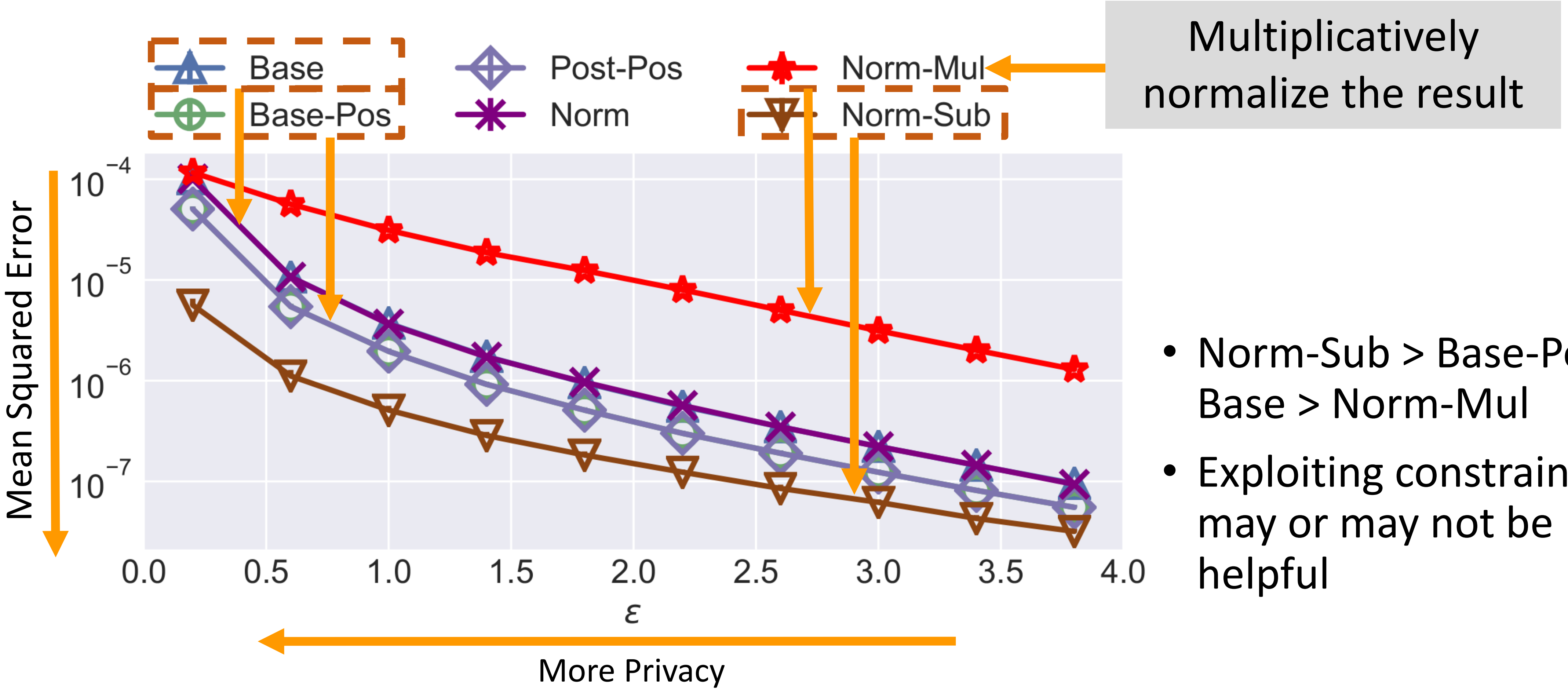
Takeaway Message

- Utility is composed of bias and variance
- Post processing introduces bias but reduces variance
- Different method achieves different bias-variance tradeoff



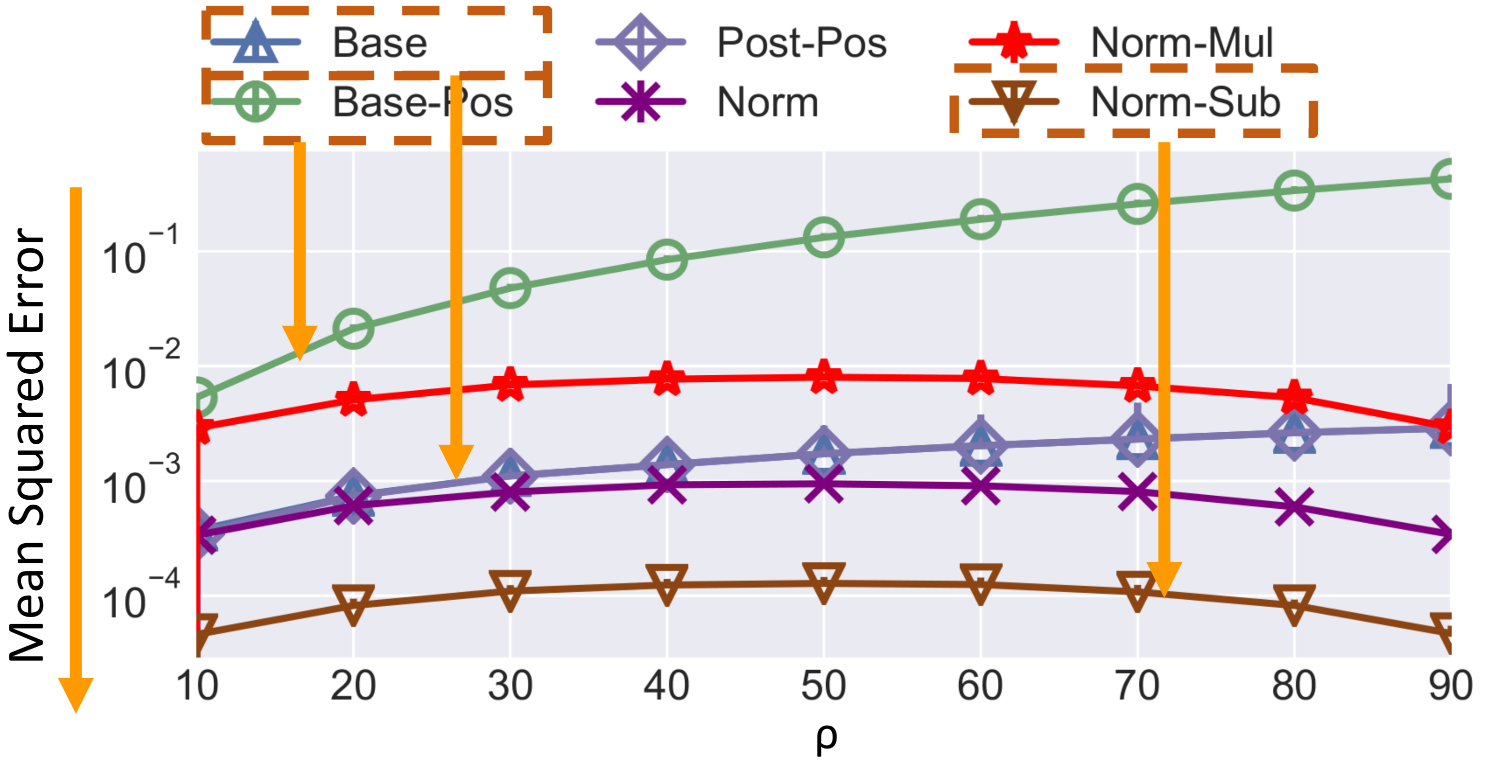


Comparison of Different Methods



- Norm-Sub > Base-Pos > Base > Norm-Mul
- Exploiting constraint may or may not be helpful

Comparison of Different Methods



- Uniformly sample ρ% elements from the domain.
- MSE of estimating a subset of values (set-value).

- Normalization-based methods works better.
- MSE is symmetric with ρ = 50 if the estimates sum up to 1.

Summary

- LDP noise follows Gaussian.
- Norm-Sub is the solution to MLE.
- Exploiting priors is helpful.
- Different method works for different tasks.

Method	Description
Base	Use existing estimation
Base-Pos	Convert negative est. to 0
Post-Pos	Convert negative query result to 0
Base-Cut	Convert est. below threshold T to 0
Norm	Add δ to est.
Norm-Mul	Convert negative est. to 0, then multiply ϒ to positive est.
Norm-Cut	Convert negative and small positive est. below ϑ to 0
Norm-Sub	Convert negative est. to 0 while adding δ to positive est.
MLE-Apx	Convert negative est. to 0, then add δ to positive est.
Power	Fit Power-Law dist., then minimize expected squared error.
PowerNS	Apply Norm-Sub after Power