Towards Practical Differentially Private Convex Optimization

ROGER IYENGAR
CARNEGIE MELLON UNIVERSITY

JOSEPH P. NEAR
UNIVERSITY OF VERMONT

DAWN SONG
UNIVERSITY OF CALIFORNIA, BERKELEY

OM THAKKAR
BOSTON UNIVERSITY

ABHRADEEP THAKURTA
UNIVERSITY OF CALIFORNIA, SANTA CRUZ

LUN WANG
UNIVERSITY OF CALIFORNIA, BERKELEY

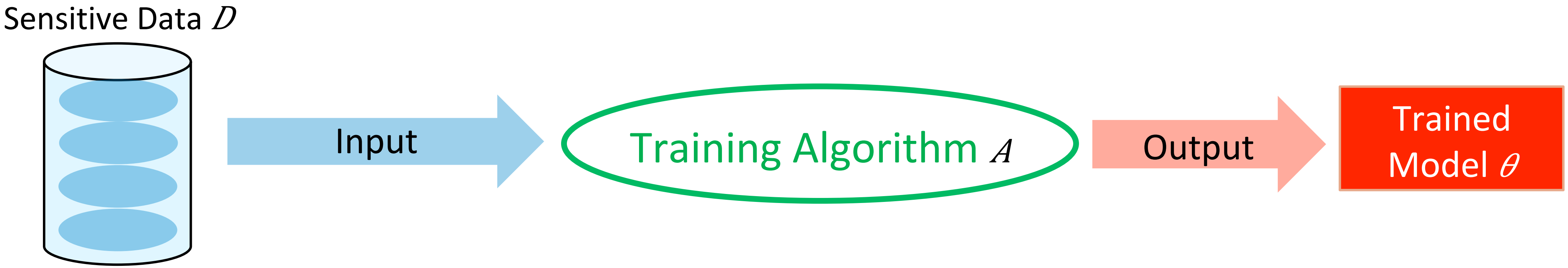
Contributions

- **New Algorithm for Differentially Private Convex Optimization: Approximate Minima Perturbation (AMP)**
  - Can leverage any off-the-shelf optimizer
  - Works for *all* convex loss functions
  - Has a competitive hyperparameter-free variant
- **Broad Empirical Study**
  - 6 state-of-the-art techniques
  - 2 models: Logistic Regression, and Huber SVM
  - 13 datasets: 9 public (4 high-dimensional), 4 real-world use cases
  - Open-source repo: https://github.com/sunblaze-ucb/dpml-benchmark

This Talk

- Why Privacy for Learning?
- Background
  - Differential Privacy (DP)
  - Convex Optimization
- Approximate Minima Perturbation (AMP)
- Broad Empirical Study

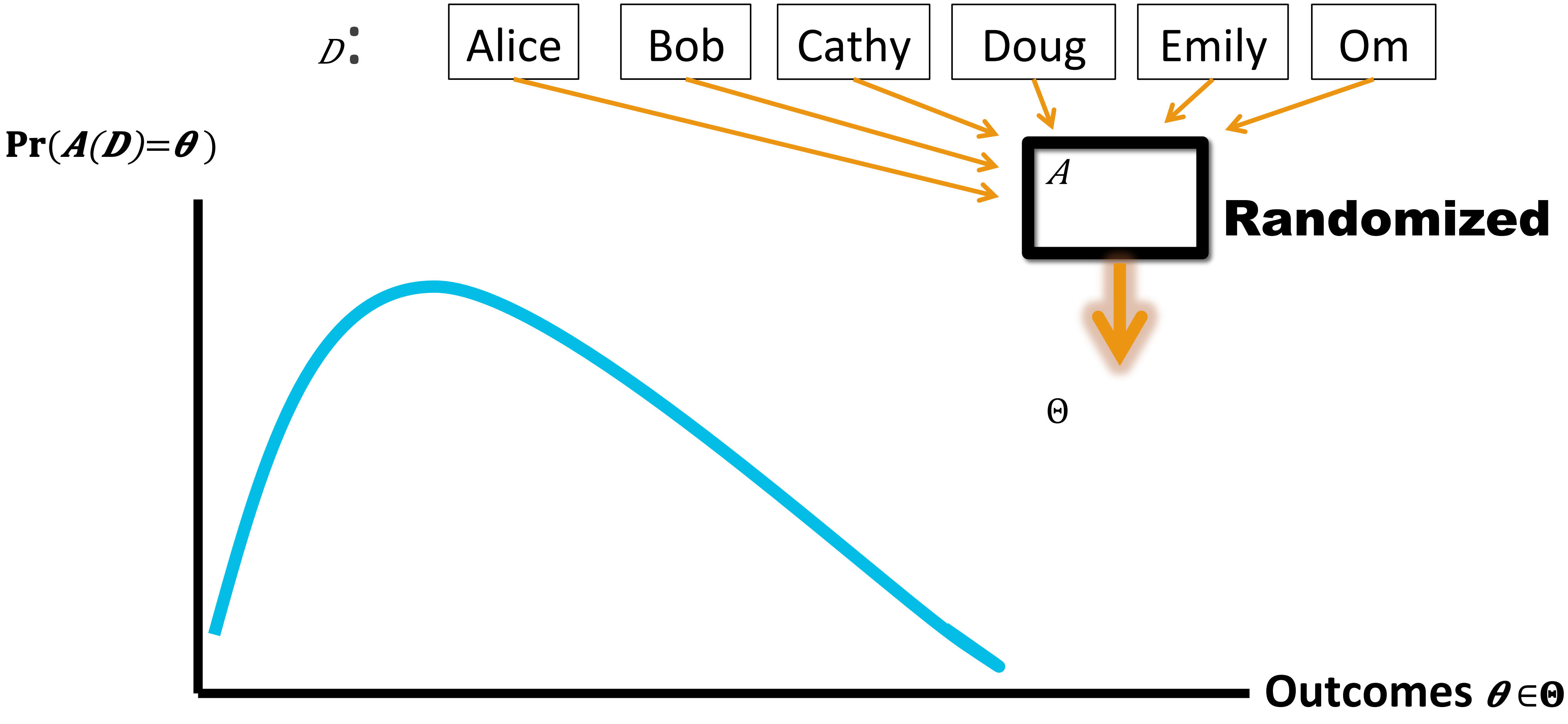
Why Privacy for Learning?



- Models can leak information about training data
  - Membership inference attacks [Shokri Stronati Song Shmatikov'17, Carlini Liu Kos Erlingsson Song'18, Melis Song Cristofaro Shmatikov'18]
  - Model inversion attacks [Fredrikson Jha Ristenpart'15, Wu Fredrikson Jha Naughton'16]

- Solution?

Differential Privacy [Dwork Mcsherry Nissim Smith '06]



D:

Alice Bob Cathy Doug Emily Om

A

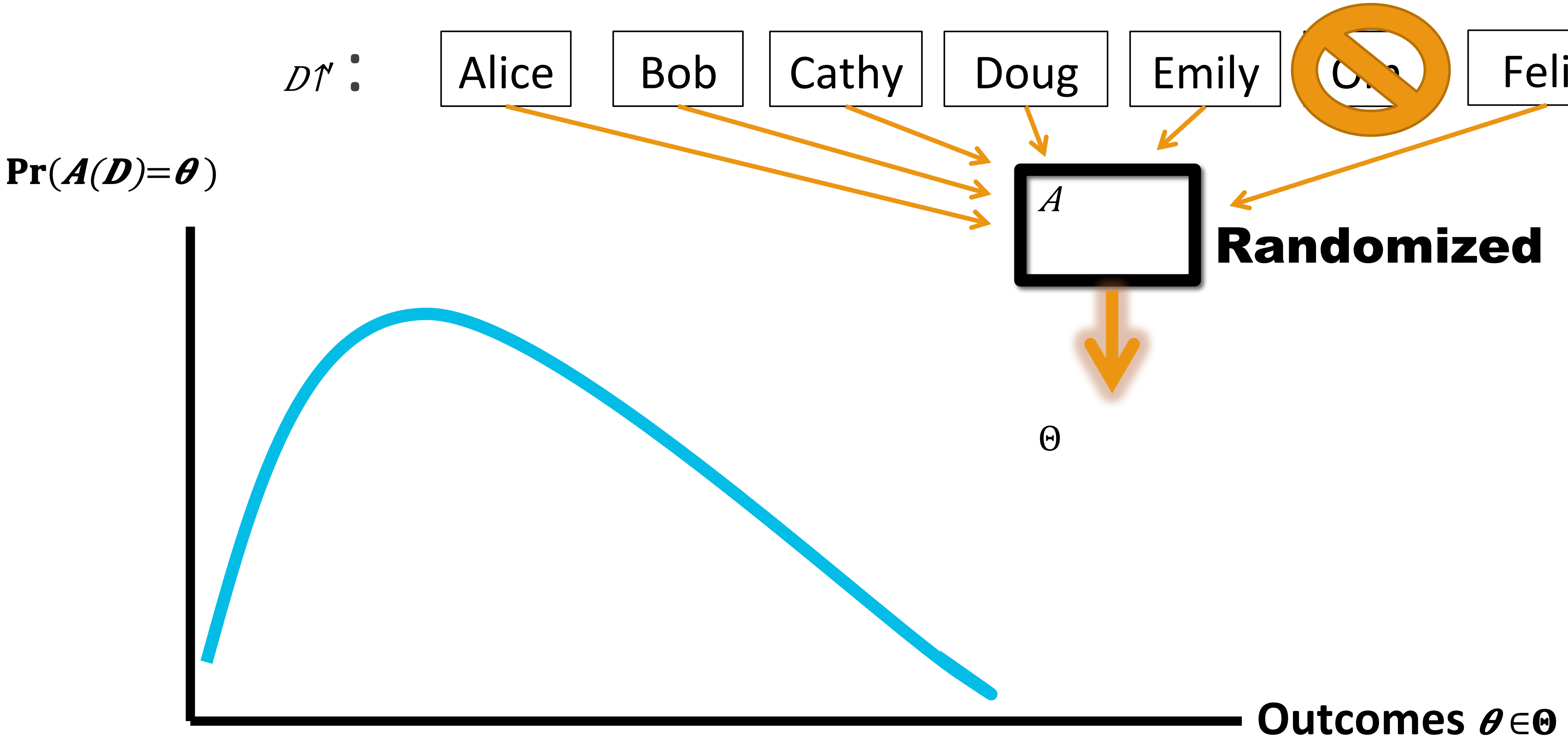
Randomized

Θ

$\mathbf{Pr}(\boldsymbol{A(D)}=\boldsymbol{\theta})$

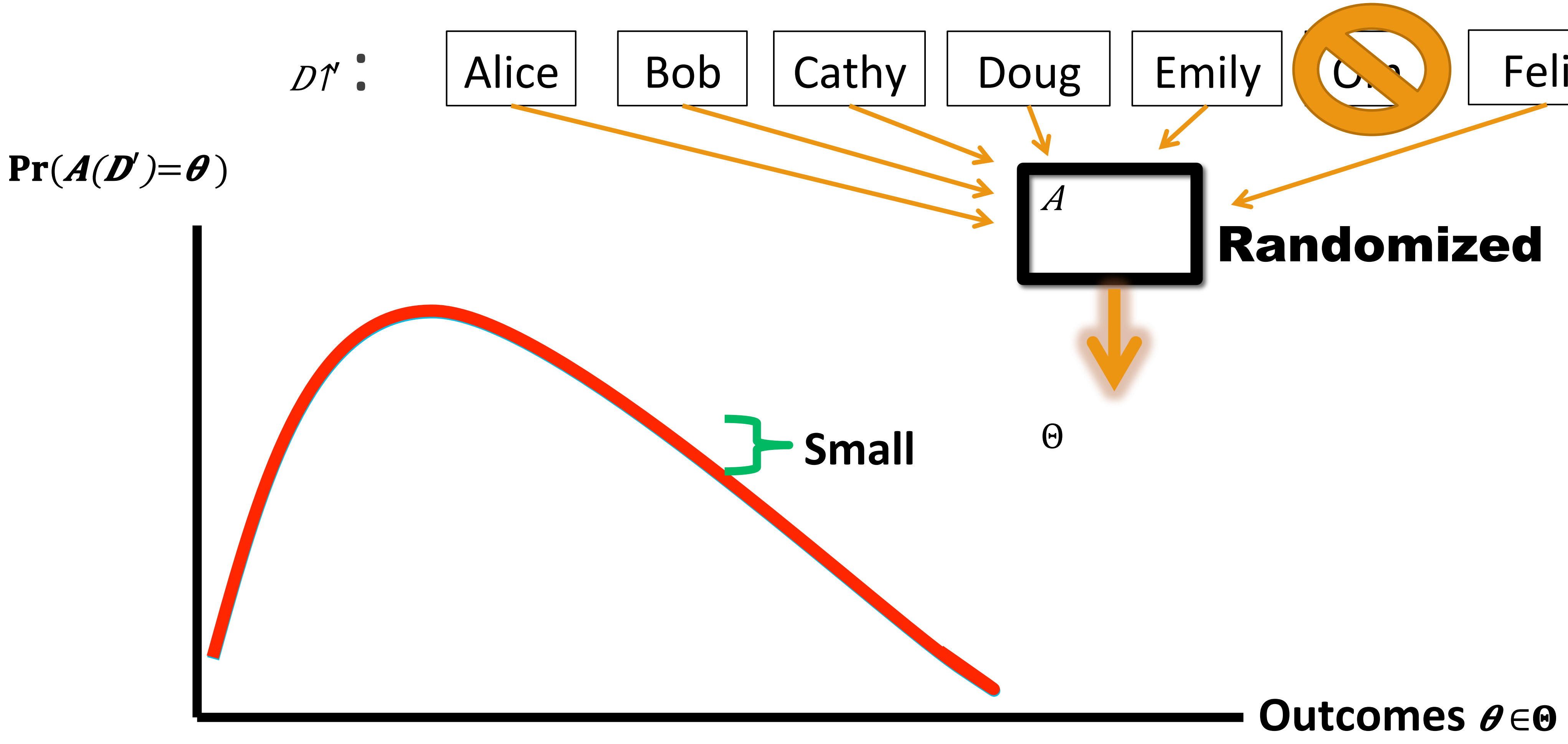
Outcomes $\boldsymbol{\theta}\in\boldsymbol{\Theta}$

Differential Privacy [Dwork Mcsherry Nissim Smith ‘06]



D' :
Alice
Bob
Cathy
Doug
Emily
Felix
A
Randomized
Θ
$\mathbf{Pr}(\boldsymbol{A}(\boldsymbol{D})=\boldsymbol{\theta})$
Outcomes $\boldsymbol{\theta} \in \boldsymbol{\Theta}$

Differential Privacy [Dwork Mcsherry Nissim Smith ‘06]



Differential Privacy [Dwork Mcsherry Nissim Smith '06]

- Privacy parameters: (ε,δ)
- A randomized algorithm $A:\mathcal{D}^n \to T$ is (ε,δ)-DP if
  - for all neighboring datasets $D,D' \in \mathcal{D}^n$, i.e., $dist(D,D')=1$
  - for all sets of outcomes $S \subseteq \Theta$, we have

$$\Pr(A(D)\in S) \le e^{\varepsilon} \Pr(A(D')\in S) + \delta$$

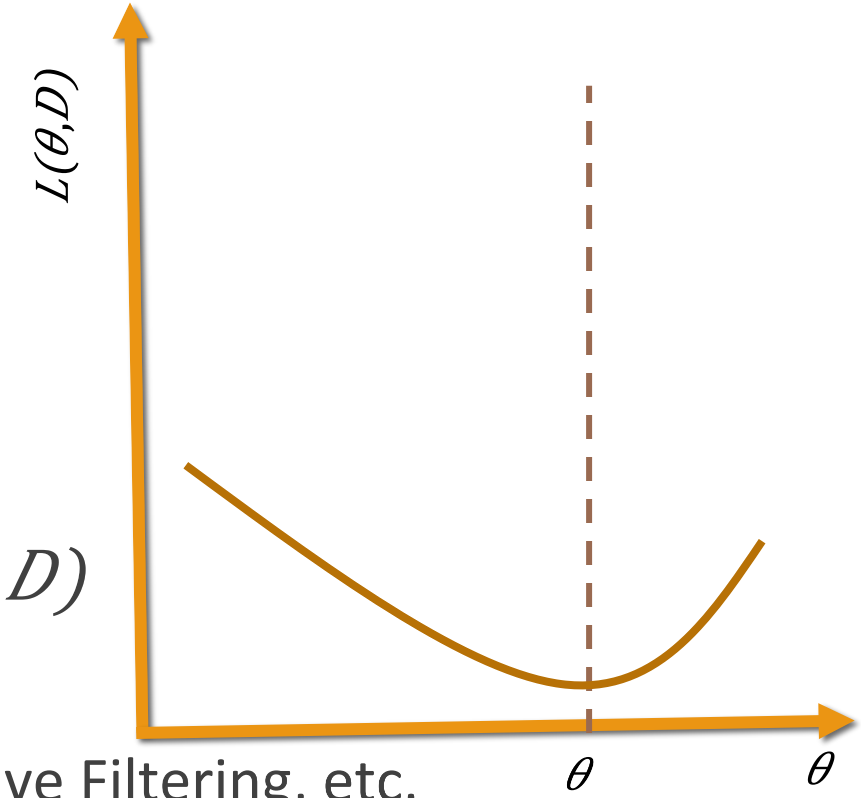
ε: Multiplicative change.
Typically, $\varepsilon=O(1)$

δ: Additive change.
Typically, $\delta=O(1/n^2)$

Convex Optimization

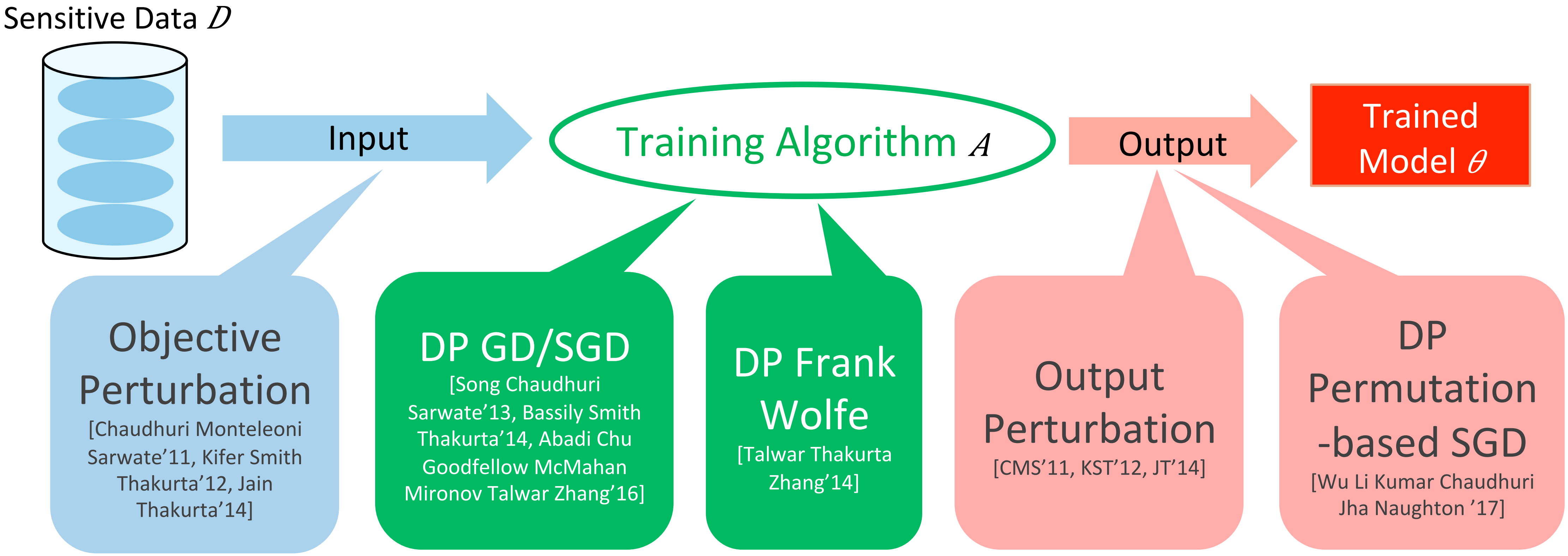
- Input:
  - Dataset $D \in \mathcal{D} \uparrow n$
  - Loss function $L(\theta,D)$, where
    - $\theta \in \mathbb{R} \uparrow p$ is a model
    - Loss L is convex in the first parameter θ
- Goal: Output model θ such that

$$\theta \in \min ┬ \theta \in \mathbb{R} \uparrow p \; □ L(\theta,D)$$



- Applications:
  - Machine Learning, Deep Learning, Collaborative Filtering, etc.

DP Convex Optimization - Prior Work



Sensitive Data D

Input

Training Algorithm A

Output

Trained Model θ

Objective Perturbation
[Chaudhuri Monteleoni Sarwate'11, Kifer Smith Thakurta'12, Jain Thakurta'14]

DP GD/SGD
[Song Chaudhuri Sarwate'13, Bassily Smith Thakurta'14, Abadi Chu Goodfellow McMahan Mironov Talwar Zhang'16]

DP Frank Wolfe
[Talwar Thakurta Zhang'14]

Output Perturbation
[CMS'11, KST'12, JT'14]

DP Permutation -based SGD
[Wu Li Kumar Chaudhuri Jha Naughton '17]

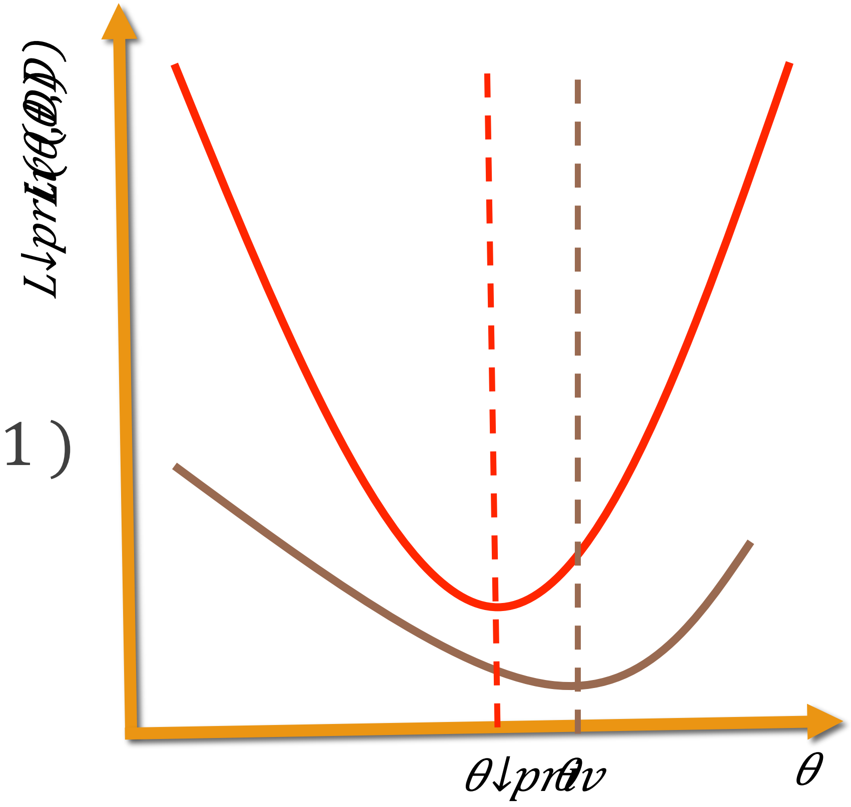
🟪 - Requires minima of loss

🟧 - Requires custom optimizer

Approximate Minima Perturbation (AMP)

- Input:
  - Dataset D, Loss function: $L(\theta,D)$
  - Privacy parameters: $b=(\epsilon, \delta)$
  - Gradient norm bound γ
- Algorithm (high-level):
  1. Split privacy budget into 2 parts b_1 and b_2
  2. Perturb loss: $L_{priv}(\theta,D)=L(\theta,D)+Reg(\theta,b_1)$

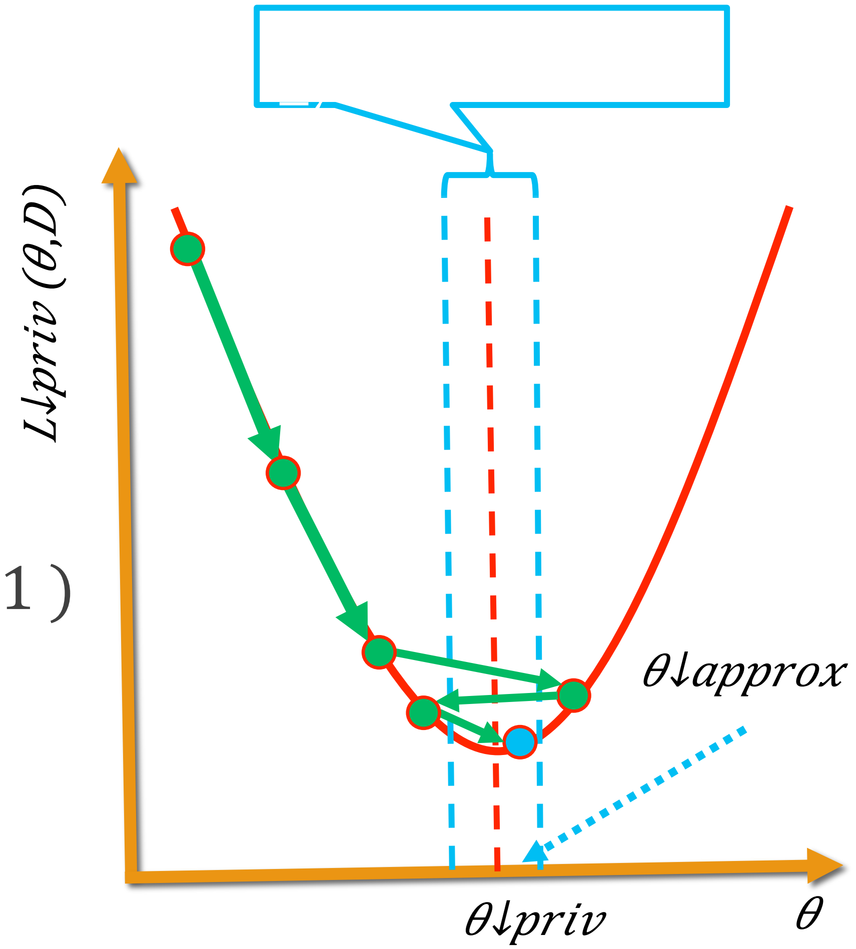
Similar to standard Objective Perturbation [KST'12]



Approximate Minima Perturbation (AMP)

- Input:
  - Dataset D, Loss function: $L(\theta,D)$
  - Privacy parameters: $b=(\epsilon, \delta)$
  - Gradient norm bound γ
- Algorithm (high-level):
  1. Split privacy budget into 2 parts b_1 and b_2
  2. Perturb loss: $L_{priv}(\theta,D)=L(\theta,D)+Reg(\theta,b_1)$
  3. Let $\theta_{approx}=\theta$ s.t. $\|\nabla L_{priv}(\theta,D)\|_2 \leq \gamma$
  4. Output $\theta_{approx}+Noise(b_2,\gamma)$

Similar to standard Objective Perturbation [KST'12]



Utility guarantees

- Let θ minimize $L(\theta;D)$, and the regularization parameter $\Lambda=\Theta(\xi\sqrt{p}/\epsilon n\|\theta\|)$.
- Objective Perturbation [KST'12]: If θ_{priv} is the output of obj. pert.:

$$\mathbb{E}(L(\theta_{priv};D)-L(\theta;D))=O(\xi\sqrt{p}\|\theta\|/\epsilon n).$$

- AMP (adapted from [KST'12]): For output θ_{AMP}:

$$\mathbb{E}(L(\theta_{AMP};D)-L(\theta;D))=O(\xi\sqrt{p}\|\theta\|/\epsilon n+\|\theta\|\gamma n).$$

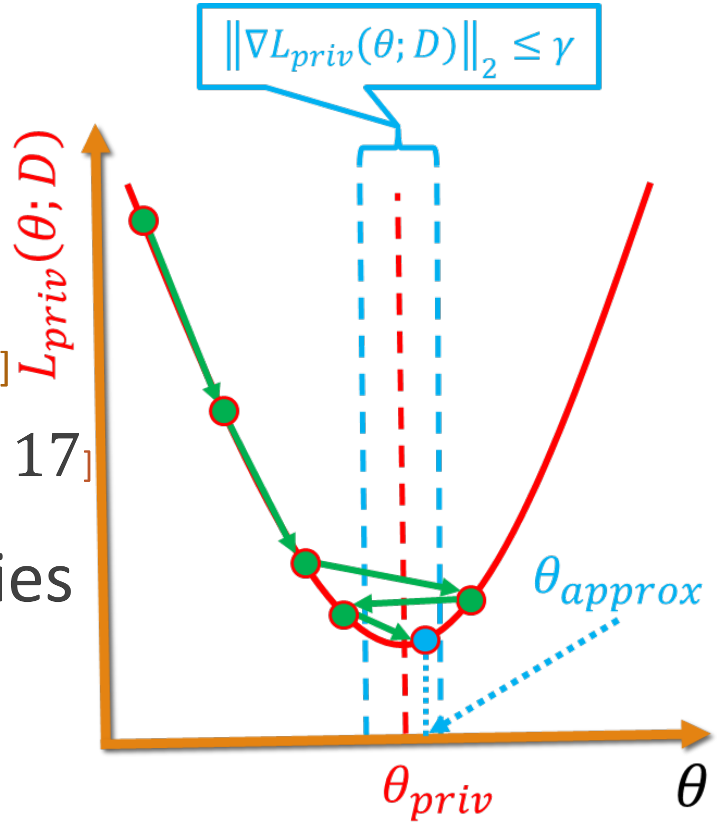
  - For $\gamma=O(1/n^2)$, the utility of AMP is asymptotically the same as that of Obj. Pert.
- Private PSGD [$\mathrm{WLK}^{+}17$]: For output θ_{PSGD}, and model space radius R:

$$\mathbb{E}(L(\theta_{PSGD};D)-L(\theta;D))=O(\xi\sqrt{p}R/\epsilon\sqrt{n}).$$

  - For $\gamma=O(1/n^2)$, the utility of AMP has a better dependence on n than Private PSGD.

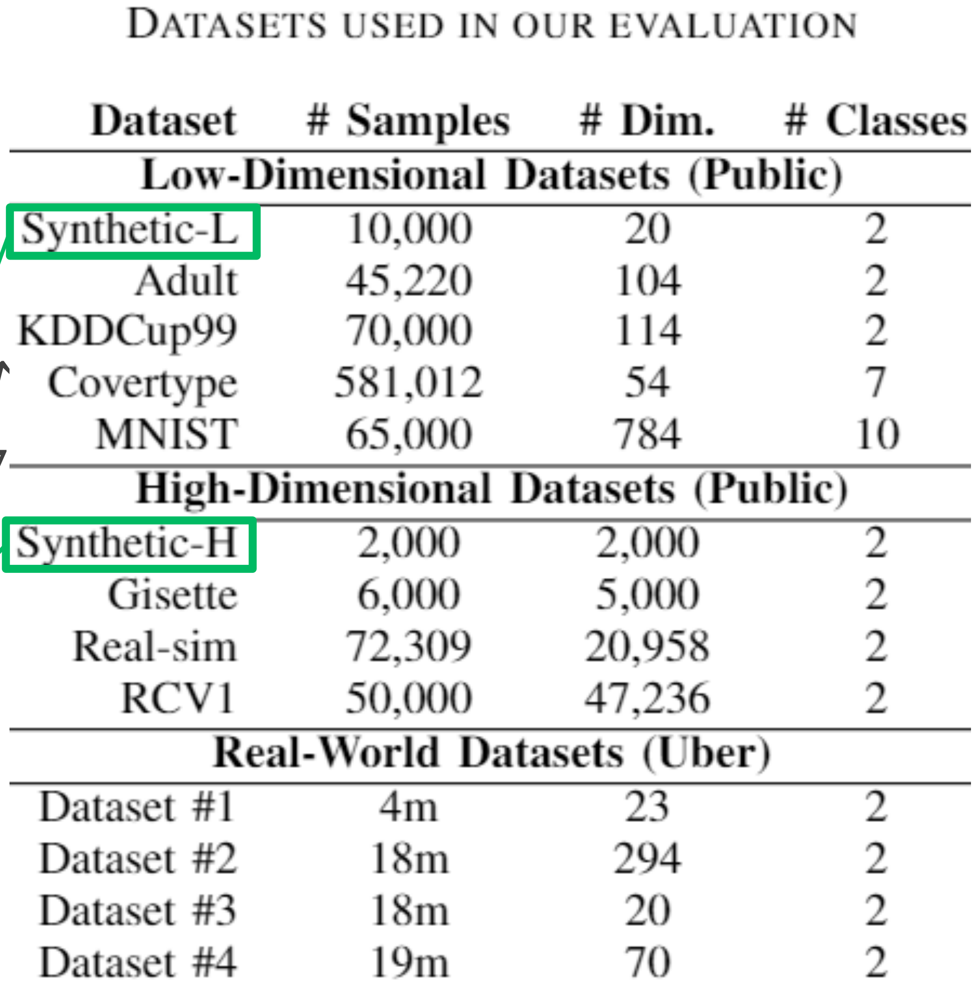
AMP - Takeaways

- Can leverage any off-the-shelf optimizer
- Works for all *standard* convex loss functions
- For $\gamma = O(1/n^2)$, the utility of AMP:
  - is asymptotically the same as Objective Perturbation [KST'12]
  - has a better dependence on n than Private PSGD $[WLK^+17]$
- $\gamma = 1/n^2$ achievable using standard Python libraries



Empirical Evaluation

- Algorithms evaluated:
  - Approximate Minima Perturbation (AMP)
  - Private SGD [BST↑ 14,ACG↑+ 17]
  - Private Frank-Wolfe (FW) [TTZ↑ 14]
  - Private Permutation-based SGD (PSGD) [WLK↑
  - Private Strongly-convex (SC) PSGD [WLK↑+ 17
  - Hyperparameter-free (HF) AMP
    - Splitting the privacy budget: We provide a schedule for low- and high-dim. data by evaluating AMP only on synthetic data
  - Non-private (NP) Baseline



DATASETS USED IN OUR EVALUATION

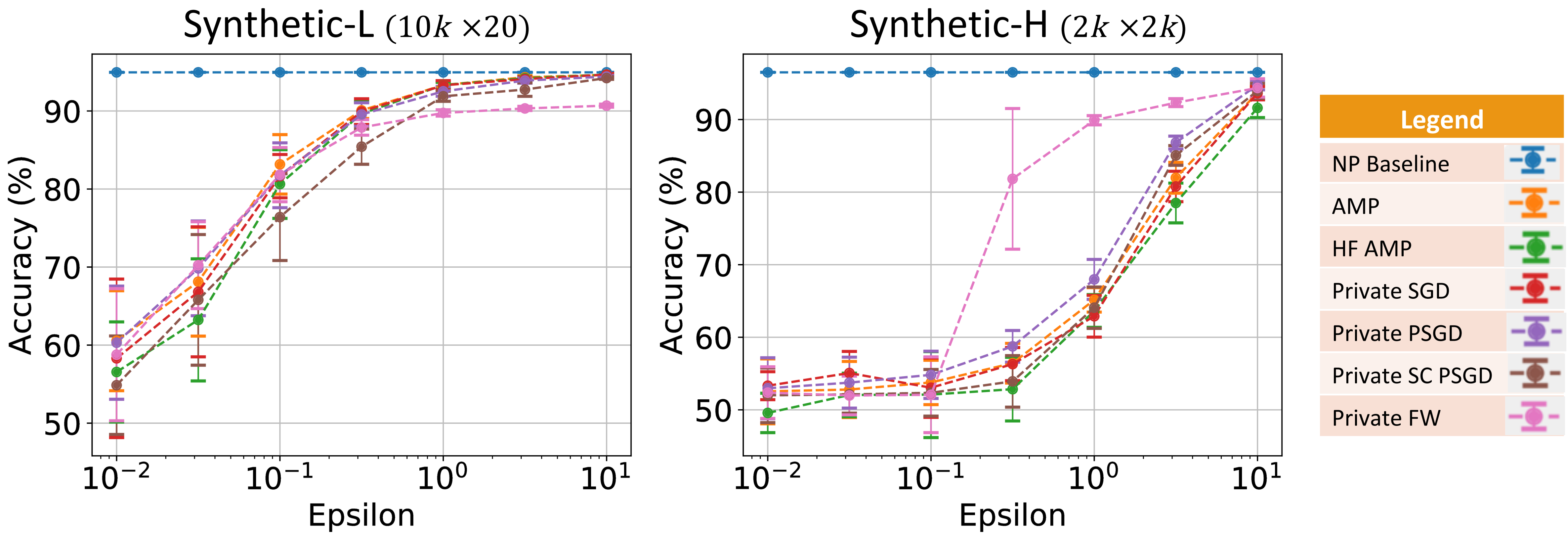
Dataset	# Samples	# Dim.	# Classes
Low-Dimensional Datasets (Public)			
Synthetic-L	10,000	20	2
Adult	45,220	104	2
KDDCup99	70,000	114	2
Covertype	581,012	54	7
MNIST	65,000	784	10
High-Dimensional Datasets (Public)			
Synthetic-H	2,000	2,000	2
Gisette	6,000	5,000	2
Real-sim	72,309	20,958	2
RCV1	50,000	47,236	2
Real-World Datasets (Uber)			
Dataset #1	4m	23	2
Dataset #2	18m	294	2
Dataset #3	18m	20	2
Dataset #4	19m	70	2

Empirical Evaluation

- Loss functions considered:
  - Logistic loss — This talk
  - Huber SVM
- Procedure:
  - 80/20 train/test random split
  - Fix $\delta=1/n^2$, and vary ϵ from 0.01 to 10
  - Measure accuracy of final tuned* private model over test set
  - Report the mean accuracy and std. dev. over 10 independent runs

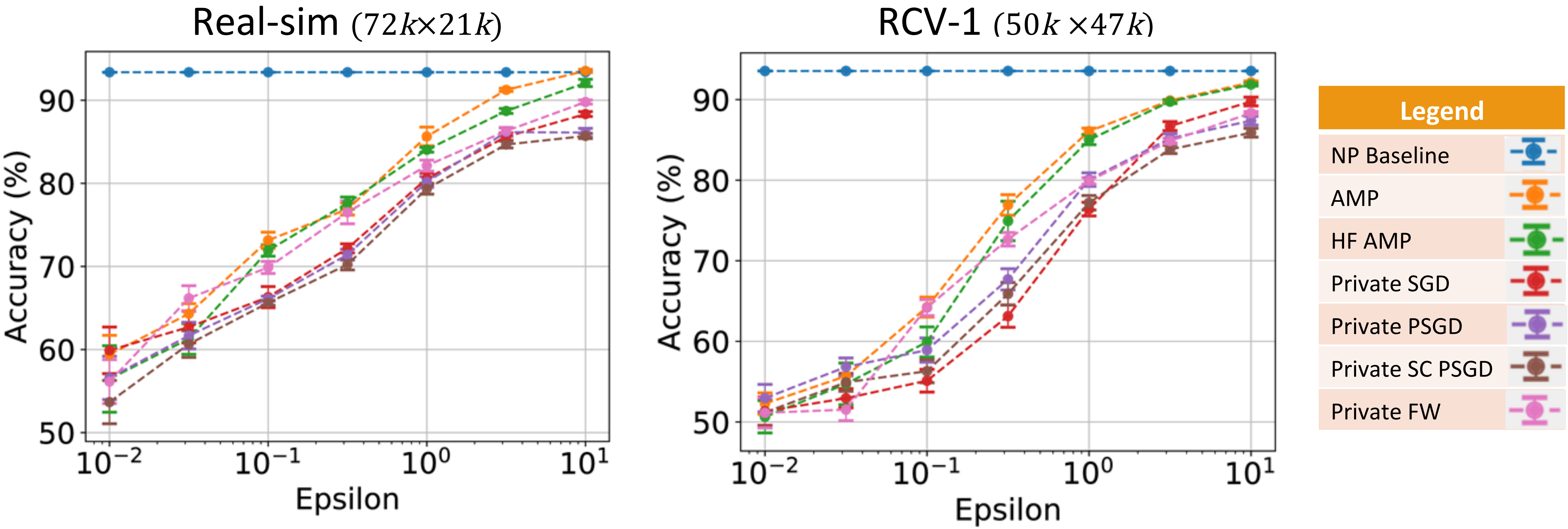
*Does not apply to Hyperparameter-free AMP.

Synthetic Datasets



- Synthetic-H is high-dimensional, but low-rank
- Private Frank-Wolfe performs the best on Synthetic-H

High-dimensional Datasets

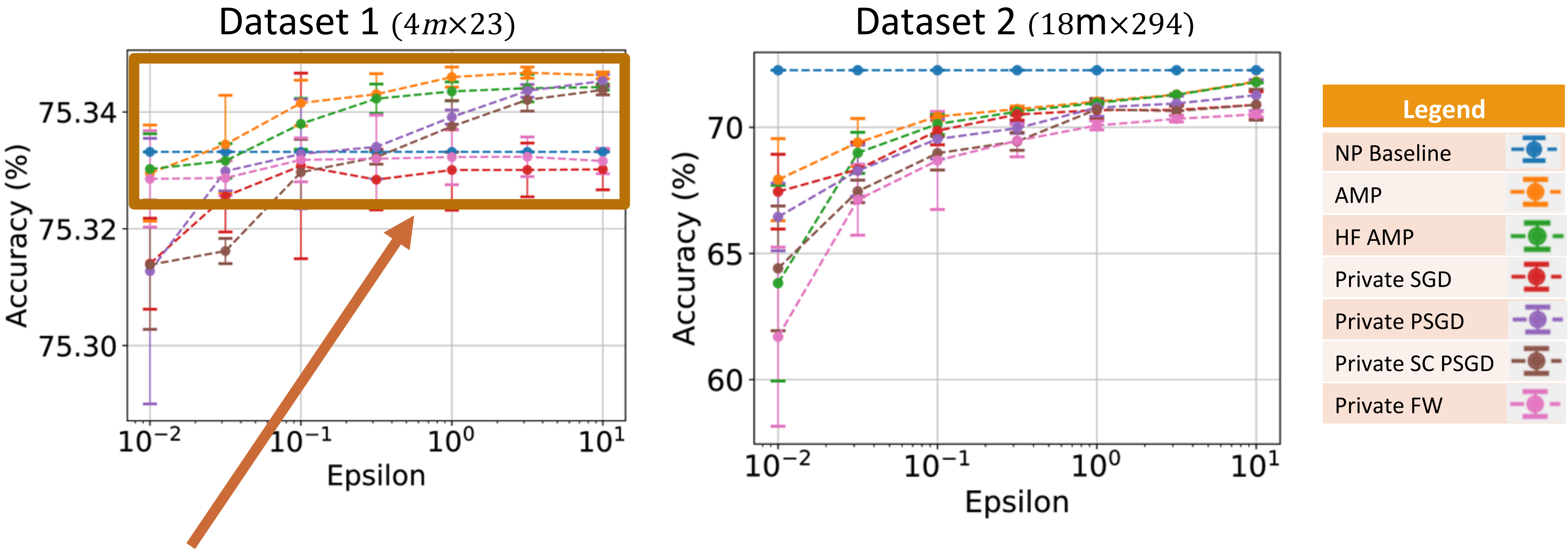


Real-sim $(72k\times 21k)$

RCV-1 $(50k\times 47k)$

- Both variants of AMP almost always provide the best performance

Real-world Use Cases (Uber)



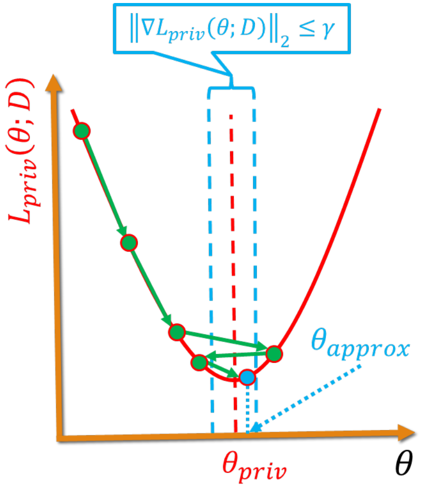
- DP as a regularizer [BST’14, Dwork Feldman Hardt Pitassi Reingold Roth ’15]
- Even for $\epsilon=10\uparrow-2$, accuracy of AMP is close to non-private baseline

Conclusions

- For large datasets, cost of privacy is low
  - Private model is within 4% accuracy of the non-private one for $\epsilon = 0.01$, and within 2% for $\epsilon = 0.1$

- AMP almost always provides the best accuracy, and is easily deployable in practice

- Hyperparameter-free AMP is competitive w.r.t. tuned state-of-the-art private algorithms



- Open-source repo: https://github.com/sunblaze-ucb/dpml-benchmark

Thank You!