Learning Privately with Labeled and Unlabeled Examples [SODA 2015]

Amos Beimel   Kobbi Nissim   Uri Stemmer

Dept. of Computer Science - Ben-Gurion University of the Negev -

CRCS Center for Research on Computation and Society

Differential Privacy

[DMNS06]

A (rand) algorithm $\mathcal{A}$ is (ϵ, δ) differentially private if for all neighboring databases S_1, S_2 and for all sets of outputs F:

$$\Pr[\mathcal{A}(S_1) \in F] \leq e^{\epsilon} \cdot \Pr[\mathcal{A}(S_2) \in F] + \delta$$

Private PAC Learner

[KLNRS08]

Operates on a labeled sample and:

1) Preserves differential privacy.

2) Outputs a good hypothesis w.h.p.

Motivation

The sample complexity of **private learners** may be **higher** than that of non-private ones.

In many cases: Unlabeled data is easy to come by, while Labeled data is expensive.

Goal: Reduce the labeled sample complexity.

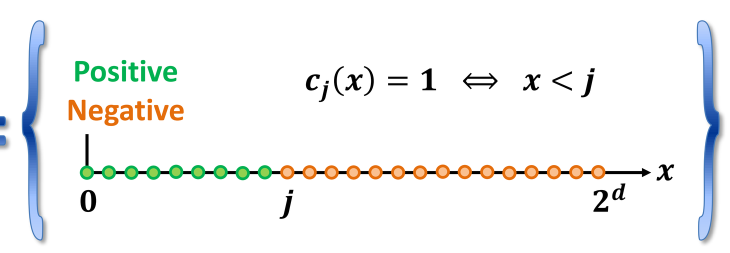
Main Theorem

Given a private learner with sample complexity n for a concept class C, it is possible to reduce the labeled sample complexity to $O(\mathrm{VC}(C))$ while maintaining the unlabeled sample complexity.

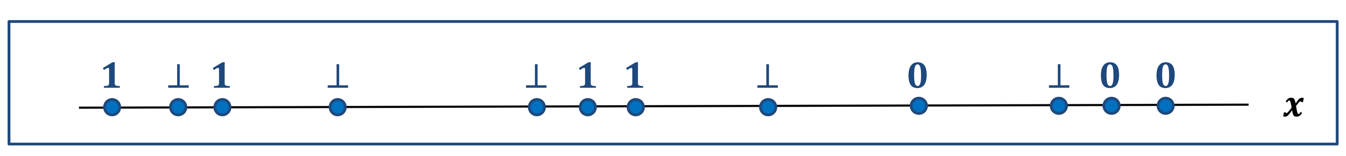
Examples (for non-active proper-learning)

	Previous Bounds		New Bounds	
	ϵ-d.p.	(ϵ, δ)-d.p.	ϵ-d.p.	(ϵ, δ)-d.p.
Points	$O(d)$	$O(1)$	$O(1)$; $O(d)$	
Intervals	$O(d)$	$2^{O(\log^* d)}$	$O(1)$; $O(d)$	$O(1)$; $2^{O(\log^* d)}$
ℓ-dim. Rect.	$O(\ell d)$	$\ell^3 \cdot 2^{O(\log^* d)}$	$O(\ell)$; $O(\ell d)$	$O(\ell)$; $\ell^3 \cdot 2^{O(\log^* d)}$

Example: Interval$_d$ =



Inputs: a base learner, and a partially labeled sample:



H:

1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	0	0	0	0	0	0	0	0	0	0

Choose $h \in H$ using the exponential mechanism, **relabel** the input using h, and execute the base learner on the relabeled database.

Algorithm LabelBoost

Goal: Reduce the labeled sample complexity of a given learner $\mathcal{A}$.

Inputs:
- Base learner $\mathcal{A}$ with sample complexity n.
- Database S of size n, only partially labeled.

1. Let H be the set of all dichotomies over S realized by the target concept class C.
2. Choose $h \in H$ using the exponential mechanism with the labeled portion of S.
3. Relabel S using h, and execute $\mathcal{A}$.

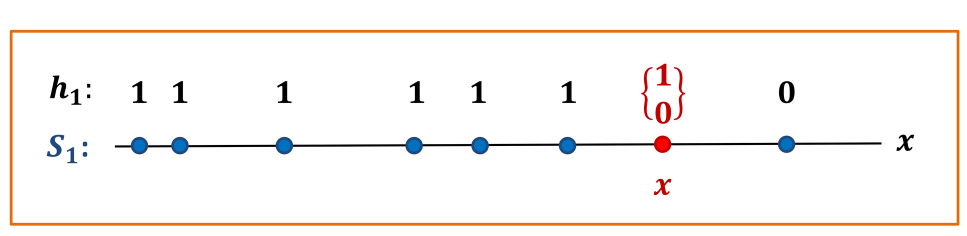
$\Rightarrow$ **Reduces the labeled sample complexity logarithmically!**

Privacy of LabelBoost

Difficulty: The set H strongly depends on the input points, so outputting an $h \in H$ may breach privacy.

Proof Intuition:

- Consider the executions on two neighboring databases S_1 and $S_2 = S_1 \cup \{(x, y)\}$
- Denote: H_1 and H_2 = the sets from step 2.
- For every $h_1 \in H_1$, there are either one or two dichotomies in H_2 that agree with h_1 on S_1.



- Assume that h_1 is chosen in the first execution, and that a "matching" dichotomy h_2 is chosen in the second execution. Then $\mathcal{A}$ is applied on neighboring databases and privacy is preserved.
- We show that h_1 and h_2 are chosen with similar probabilities.

Can Do More

We saw one step:
Reducing the labeled sample complexity from n to $O(\mathrm{VC}(C) \cdot \log n)$.

Using recursion:
Reduce the labeled sample complexity to $O(\mathrm{VC}(C))$ while maintaining the unlabeled sample complexity.

- The labeled sample complexity has no dependency in δ.

Active Learning Model

Input: pool of unlabeled examples.

Learner can query the labels of examples from the pool.

Privacy Question: Do the queries remain hidden?

Yes $\Rightarrow$ We show that the labeled sample complexity has no dependency on the privacy parameters of the learner.

No $\Rightarrow$ We show lower bounds on the labeled sample complexity.

Summary and Open Questions

- **The labeled sample complexity of private learners is characterized by the VC dimension.**
- A given private learner could be transformed into a private learner with reduced labeled sample complexity.

Efficient algorithms?

Efficient transformation?