Machine Learning with Differential Privacy

Elliot Creager
May 2019
AMMI @ AIMS Kigali

Learning Objectives

We want you to understand the following:

- DP+ML: Why?
- Basic tradeoffs and design choices at play
- DP+ML: How? (applying basic mechanisms)
- The Gaussian Mechanism and its applications:
  - Stochastic Gradient Descent
- Composition: how to analyze a sequence of mechanisms
- Private neural nets:
  - Implementation difficulties
  - Recent advances

Motivating Differential Privacy

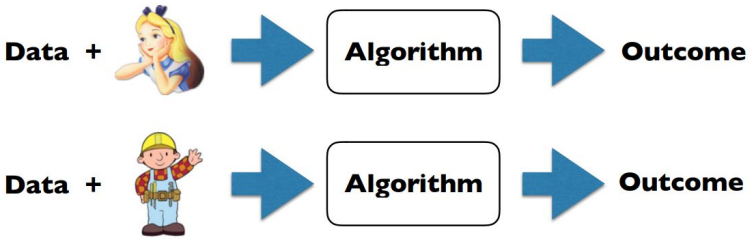
Anonymizing data is unsafe

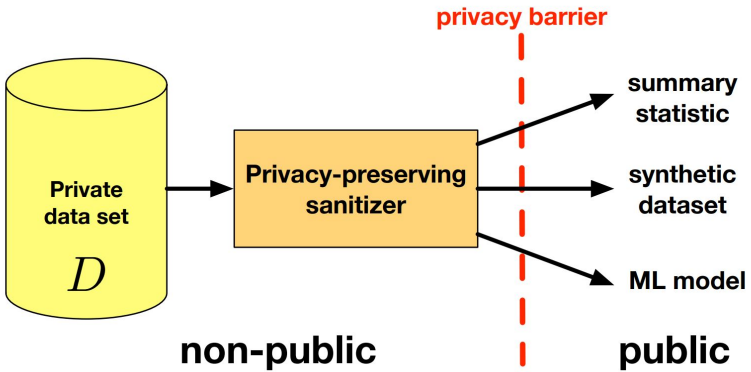
- Netflix challenge
- Narayanan & Shmatikov 2008 showed a handful of Netflix recommendation plus IMDB info reveals dataset membership and other ratings.

Publishing summary stats is unsafe

- Given an individual's gene sequence, querying a genome-wide association study reveals whether the individual is in the "case group" of the study [Dwork & Smith 2014]

What is Differential Privacy?





An individual's participation in a dataset/study doesn't change the outcome

Privacy ensured through calibrated randomness

[Chaudhuri & Sarwate 2017 NIPS tutorial]

Adversary

Prior Knowledge:

A's Genetic profile

A smokes

Case 1: Study

1.00										
.190	1.00									
.216	.251	1.00								
.186	.117	.047	1.00							
.154	.011	.170	.083	1.00						
.190	.140	.102	.095	.139	1.00					
.270	.215	.294	.248	.140	.141	1.00				
.101	.085	.170	.056	.234	.099	.175	1.00			
.239	.071	.163	.111	.161	.093	.199	.157	1.00		
.471	.117	.243	.094	.144	.123	.283	.216	.274	1.00	
.179	.202	.132	.094	.087	.159	.207	.108	.092	.294	1.00

Cancer

A has cancer

[Study violates A's privacy]

Case 2: Study

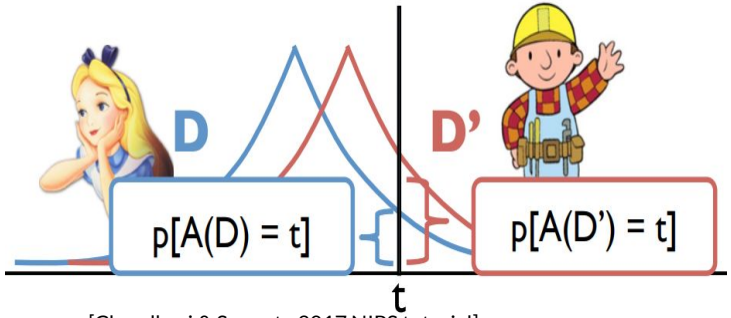
Smoking causes cancer

A probably has cancer

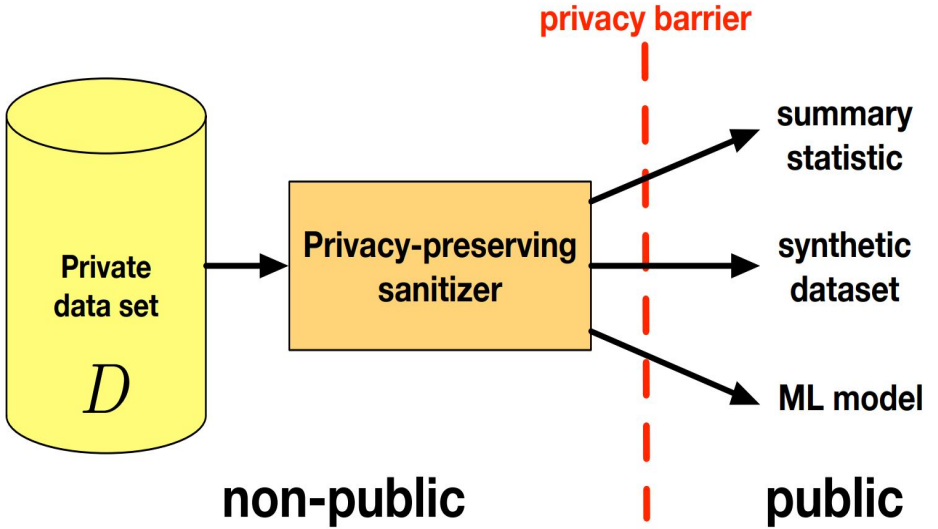
[Study does not violate privacy]

[Chaudhuri & Sarwate 2017 NIPS tutorial]

The Setting

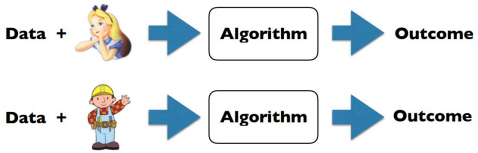


[Chaudhuri & Sarwate 2017 NIPS tutorial]



note: θ' is a specific outcome; $\Delta(\theta)$ denotes space of distributions over possible outcomes

The Definition

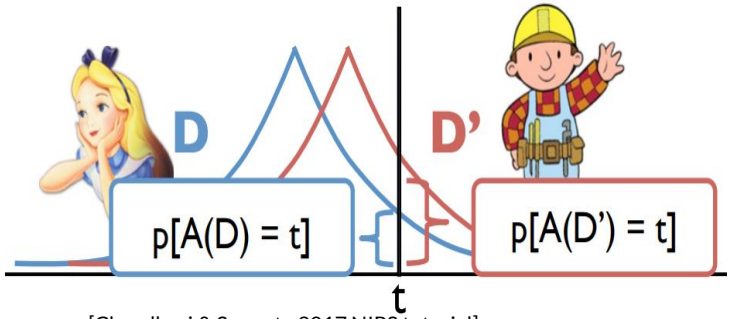


Algorithm A is (ϵ, δ)-differentially private iff $\forall$ "neighboring" datasets D, D' and $\forall$outcomes $t \in \text{Range}(A)$:

$$\mathbb{P}\left[A(D) \in t\right] \leq e^{\epsilon}\mathbb{P}\left[A(D') \in t\right] + \delta$$

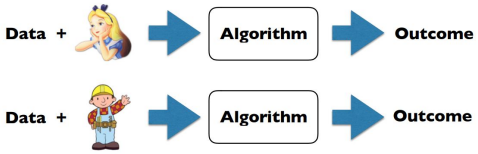
In other words:

$$\underbrace{\left|\log \frac{\mathbb{P}[A(D) \in t]}{\mathbb{P}[A(D') \in t]}\right|}_{\text{privacy loss}} \leq \epsilon \quad \text{with probability} \quad 1 - \delta$$



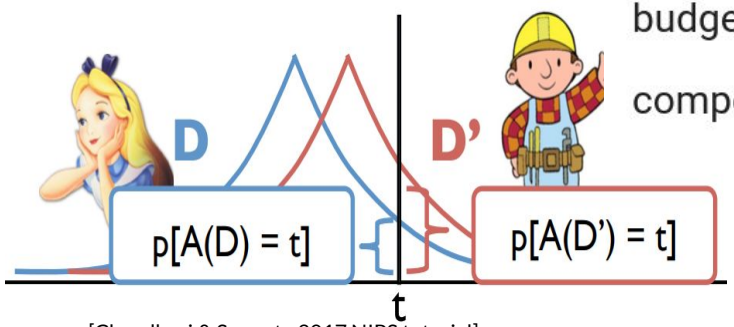
[Chaudhuri & Sarwate 2017 NIPS tutorial]

Properties of Differential Privacy



- Invariance to post-processing
  - Computation on algorithm outputs that *don't touch data* are "free"
- Composability

  If algorithms $\{A_1, A_2, \ldots, A_N\}$ are DP with respective privacy budgets $\{(\epsilon_1, \delta_1), (\epsilon_2, \delta_2), \ldots, (\epsilon_N, \delta_N)\}$, then their composition is $(\sum_{i=1}^{N} \epsilon_i, \sum_{i=1}^{N} \delta_i)$-DP.



[Chaudhuri & Sarwate 2017 NIPS tutorial]

Recall Some Basic Mechanisms

Laplace Mechanism - add Laplacian noise to fn output

Gaussian Mechanism - add Gaussian noise to fn output

Exponential Mechanism - sample fn output prop. to score

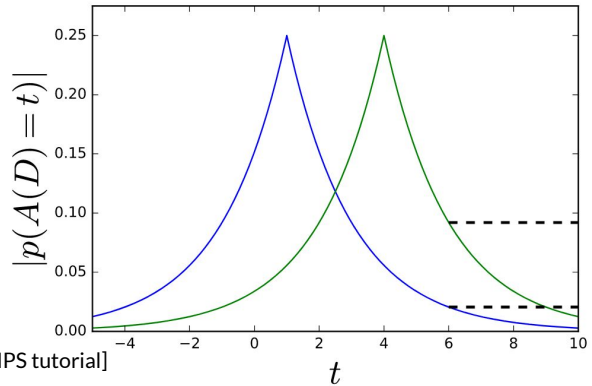
The Laplace Mechanism

Global Sensitivity of f is S(f) = $\max_{\text{dist}(D, D') = 1} |f(D) - f(D')|$

Output $f(D) + Z$, where

$$Z \sim \frac{S(f)}{\epsilon}\text{Lap}(0, 1)$$

ϵ-differentially private



[Chaudhuri & Sarwate 2017 NIPS tutorial]

Laplace distribution:

$$p(z|\mu, b) = \frac{1}{2b}\exp\left(-\frac{|z - \mu|}{b}\right)$$

The Laplace Mechanism

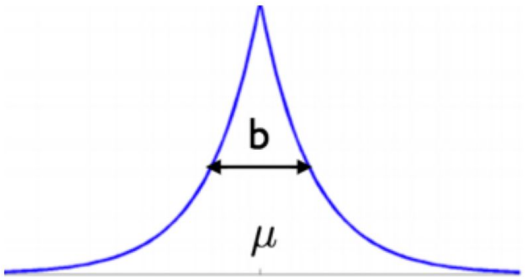
Example: compute private sample mean from n samples

f(D) = Mean(D), where each record is a scalar in [0,1]

Global Sensitivity of f = 1/n

Laplace Mechanism:

Output $f(D) + Z$, where $Z \sim \frac{1}{n\epsilon}\mathrm{Lap}(0,1)$



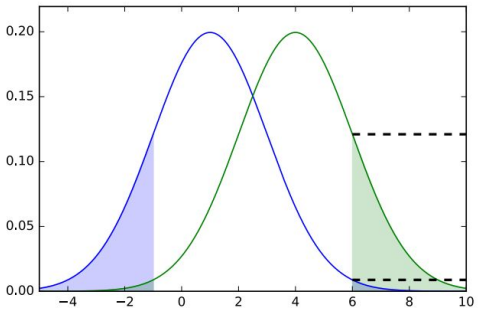
[Chaudhuri & Sarwate 2017 NIPS tutorial]

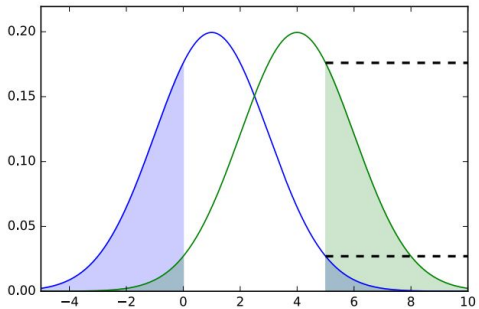
The Gaussian Mechanism

Global Sensitivity of f is S(f) = $\max_{D, D' \text{ “neighbouring”}} |f(D) - f(D')|_2$

Output $f(D) + Z$, where

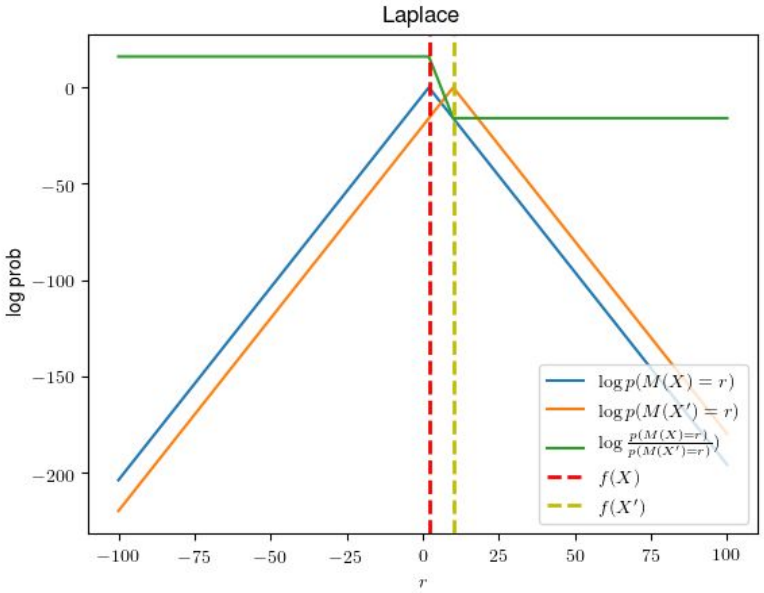
$$Z \sim \frac{S(f)}{\epsilon}\mathcal{N}(0, 2\ln(1.25/\delta)) \qquad (\epsilon, \delta)\text{-differentially private}$$

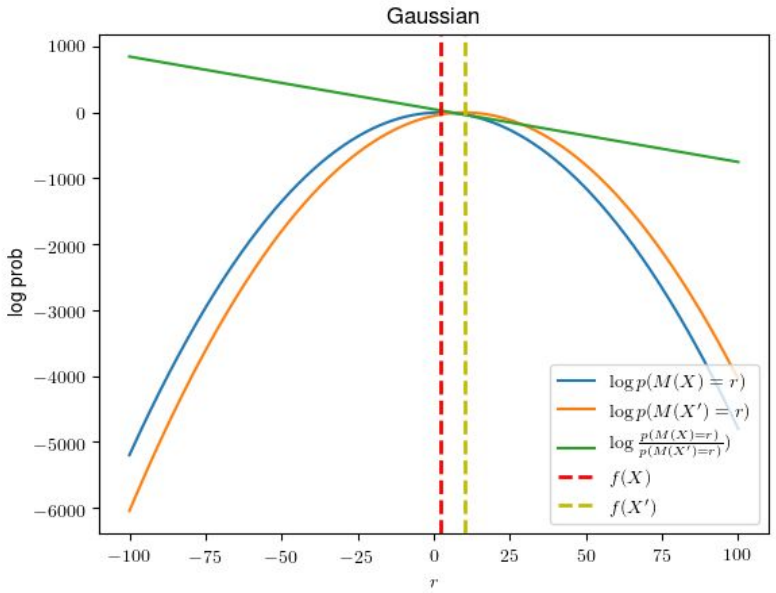




[Chaudhuri & Sarwate 2017 NIPS tutorial]

Why Approx. DP for GM?





GM cannot achieve (eps, 0) DP...

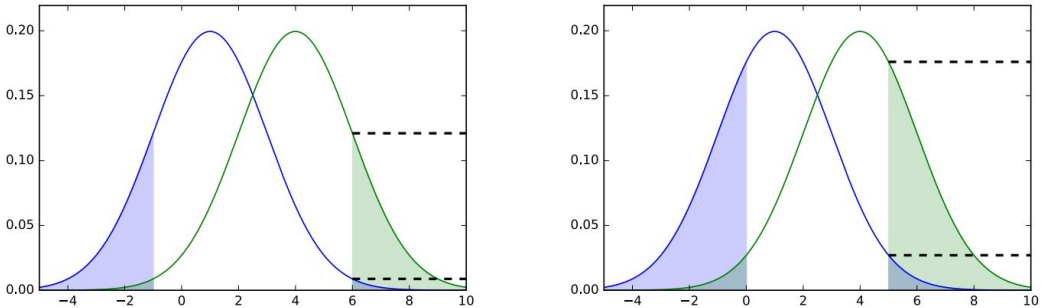
The Gaussian Mechanism

Global Sensitivity of f is S(f) = $\max_{D, D' \text{ “neighbouring”}} |f(D) - f(D')|_2$

See Dwork and Roth Appendix A for proof that GM is private...

Output $f(D) + Z$, where

$$Z \sim \frac{S(f)}{\epsilon}\mathcal{N}(0, 2\ln(1.25/\delta)) \qquad (\epsilon, \delta)\text{-differentially private}$$



[Chaudhuri & Sarwate 2017 NIPS tutorial]

The Exponential Mechanism

$$\Delta u = \max_{r \in R} \max_{x,y:||x-y| \leq 1} |u(x,r) - u(y,r)|$$

assuming x and y have same normalization term, we have

$$\log \frac{\exp(\frac{\epsilon u(x,r)}{2\Delta u})}{\exp(\frac{\epsilon u(y,r)}{2\Delta u})} = \epsilon(\frac{u(x,r) - u(y,r)}{2\Delta u}) \leq \frac{\epsilon}{2}$$

The actual mechanism saves have the privacy budget to account for changes in the normalization term

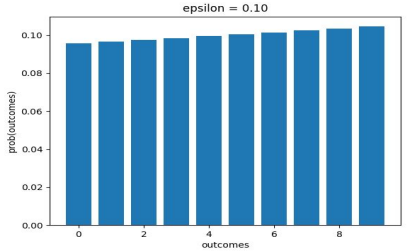
$$\mathbb{P}(M_\epsilon(x) = r) = \frac{\exp(\frac{\epsilon u(x,r)}{2\Delta u})}{\sum_{r' \in R} \exp(\frac{\epsilon u(x,r')}{2\Delta u})}$$

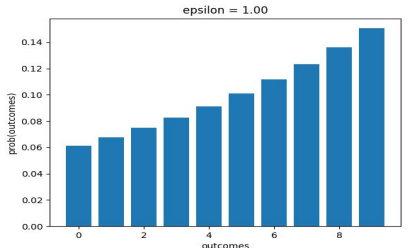
cf. Dwork and Roth section 3.4

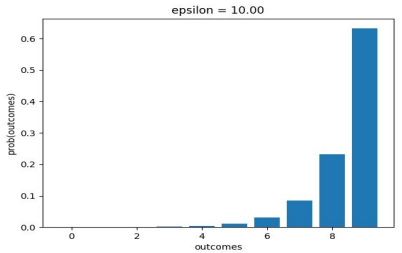
The Exponential Mechanism

$$\mathbb{P}(M_\epsilon(x) = r) \propto \exp\Big(\underbrace{\epsilon}_{\text{"temperature"}} \cdot \underbrace{\frac{u(x,r)}{2\Delta u}}_{\text{"logits"}}\Big)$$

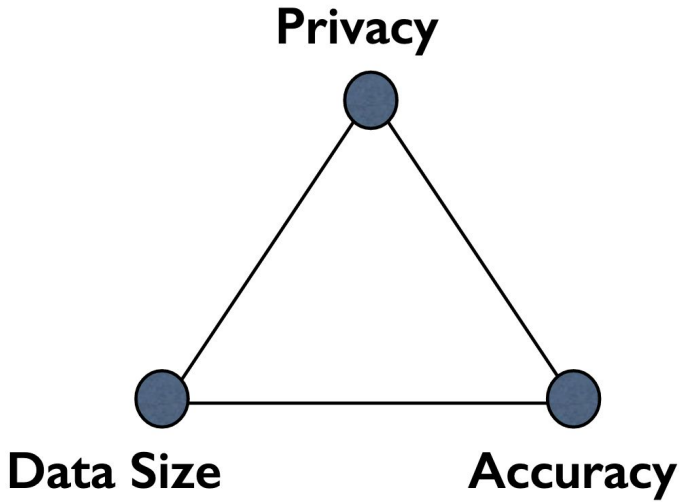
```python
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt

import torch
from torch.distributions.categorical import Categorical

scores = torch.arange(10.)
sensitivity = torch.tensor([5.])

for eps in (0.1, 1., 10.):
    eps = torch.tensor([eps])
    mech_distn = Categorical(logits=(eps * scores / (2. * sensitivity)))
    f, a = plt.subplots()
    a.bar(torch.arange(10), mech_distn.probs)
    a.set_title('epsilon = {:.2f}'.format(eps.item()))
    a.set_xlabel('outcomes')
    a.set_ylabel('prob(outcomes)')
    f.savefig('./probs{:.2f}.png'.format(eps.item()))
```





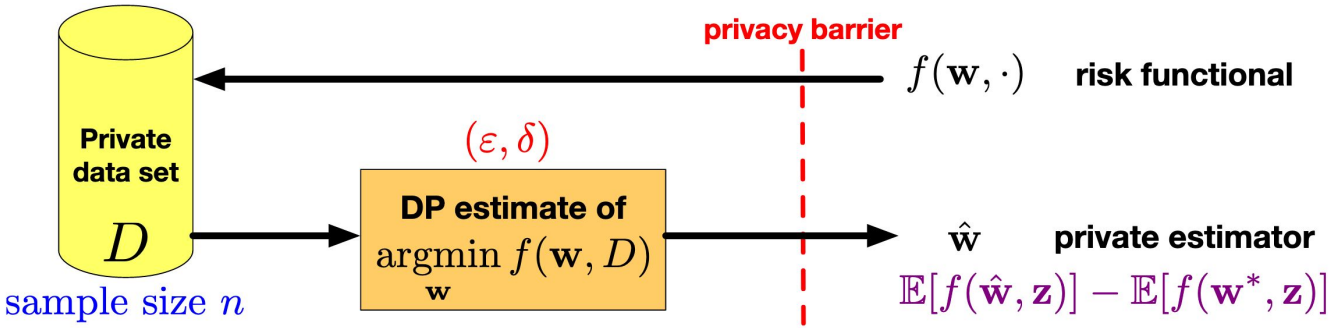


Tradeoffs in DP+ML



[Chaudhuri & Sarwate 2017 NIPS tutorial]

Tradeoffs in DP+ML



Statistical estimation: estimate a parameter or predictor using private data that has good expected performance on future data.

[Chaudhuri & Sarwate 2017 NIPS tutorial]

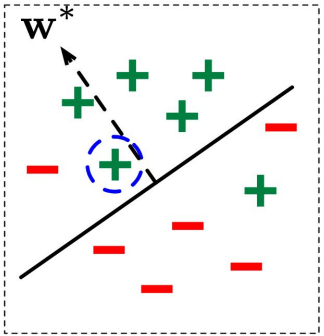
Goal: Good privacy-accuracy-sample size tradeoff

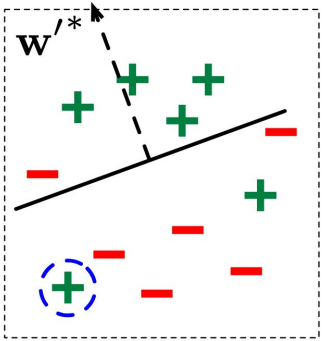
Private Empirical Risk Min.

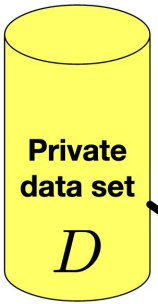
$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} \ell(\mathbf{w}, (\mathbf{x}_i, y_i)) + \lambda R(\mathbf{w})$$

- *Empirical Risk Minimization* (ERM) is a common paradigm for prediction problems.
- Produces a predictor **w** for a label/response *y* given a vector of features/covariates **x**.
- Typically use a convex loss function and regularizer to "prevent overfitting."

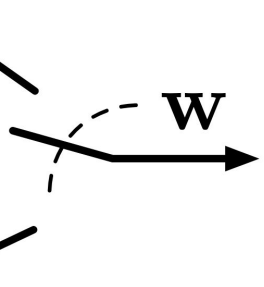
Private ERM







easy for adversary to tell the difference between D and D'



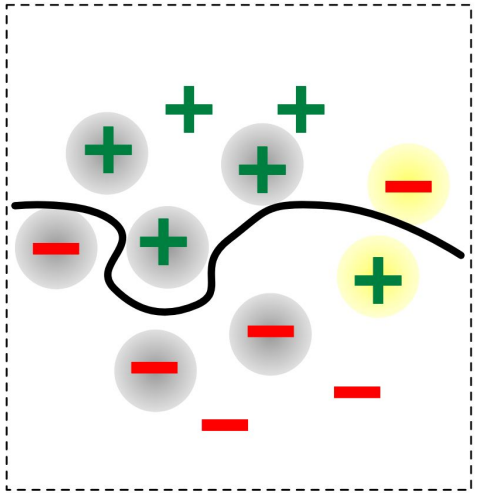
[CMS11, RBHT12]

[Chaudhuri & Sarwate 2017 NIPS tutorial]

Kernel Approaches Even Worse

- Kernel-based methods produce a classifier that is a function of the data points.
- Even adversary with black-box access to w could potentially learn those points.

$$\mathbf{w}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$



[Chaudhuri & Sarwate 2017 NIPS tutorial]

[CMS11]

Privacy & Learning Compatible

- Good learning algorithms *generalize* to the population distribution, not individuals.
- *Stable learning algorithms* generalize [BE02].
- Differential privacy can be interpreted as a form of stability that also implies generalization [DFH+15,BNS+16].
- Two parts of the same story:
  Privacy implies generalization asymptotically.
  Tradeoffs between privacy-accuracy-sample size for finite *n*.

[Chaudhuri & Sarwate 2017 NIPS tutorial]

Revisiting ERM

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \ell(\mathbf{w}, (\mathbf{x}_i, y_i)) + \lambda R(\mathbf{w})$$

- Learning using (convex) optimization uses three steps:

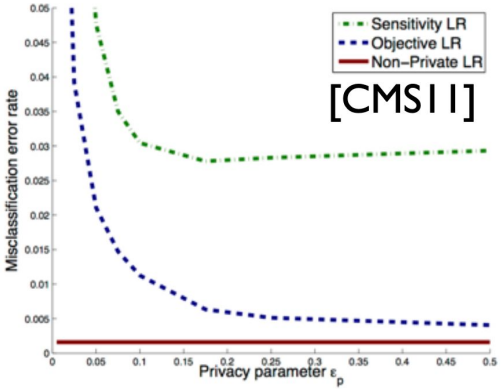
1. read in the data — **input perturbation**
2. form the objective function — **objective perturbation**
3. perform the minimization — **output perturbation**

We will discuss this one -->

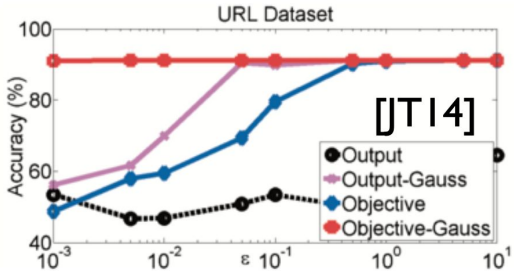
- We can try to introduce privacy in each step!

[Chaudhuri & Sarwate 2017 NIPS tutorial]

Typical Empirical Results



(a) Regularized logistic regression, KDDCup99

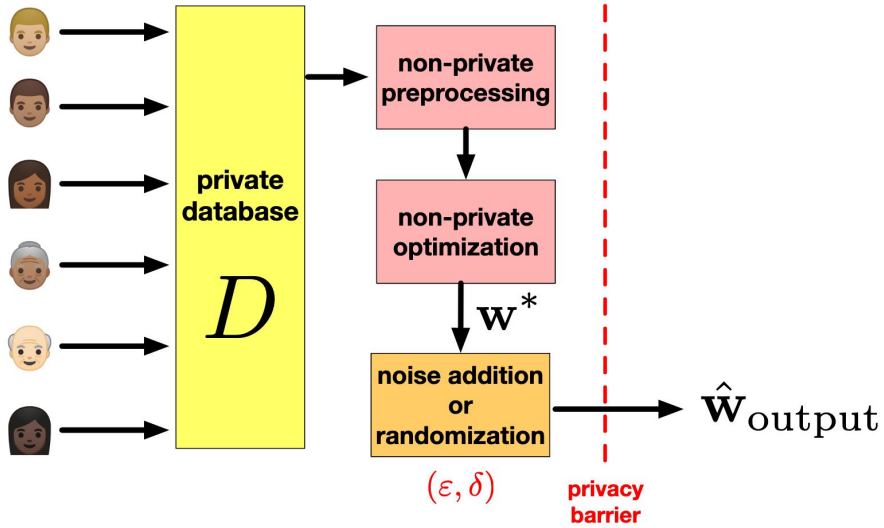


In general:

- **Objective** perturbation empirically outperforms output perturbation.
- **Gaussian** mechanism with (ε, δ) guarantees outperform **Laplace** - like mechanisms with ε-guarantees.
- **Loss vs. non-private methods** is very dataset-dependent.

[Chaudhuri & Sarwate 2017 NIPS tutorial]

Output Perturbation



- Compute the minimizer and add noise.
- Does not require re-engineering baseline algorithms

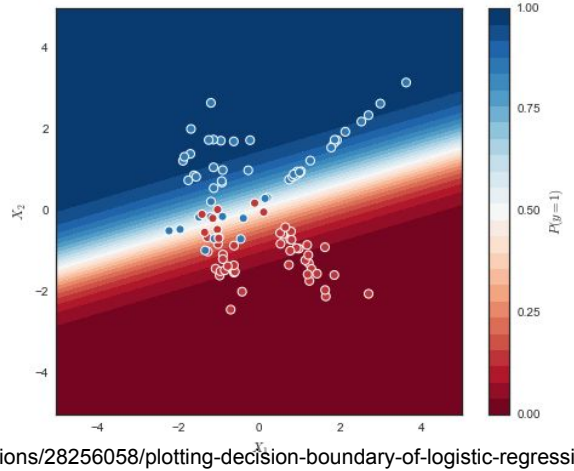
Noise depends on the sensitivity of the argmin.

[CMS11, RBHT12]

Private Logistic Regression

$$L(w) = \sum_{i=1}^{N} \text{CrossEntropy}(\sigma(w^T x_i), y_i) + \frac{1}{2}\lambda w^T w$$

What is $\Delta L(w)$, the sensitivity of the loss?



https://stackoverflow.com/questions/28256058/plotting-decision-boundary-of-logistic-regression

Private Logistic Regression

$$L(w) = \sum_{i=1}^{N} \underbrace{\text{CrossEntropy}(\sigma(w^T x_i), y_i)}_{\text{convex}} + \underbrace{\frac{1}{2}\lambda w^T w}_{\lambda-\text{strongly convex}}$$

$h(w)$ is convex iff $\nabla^2 h(w) \succcurlyeq 0$ (its Hessian is positive semi-definite)

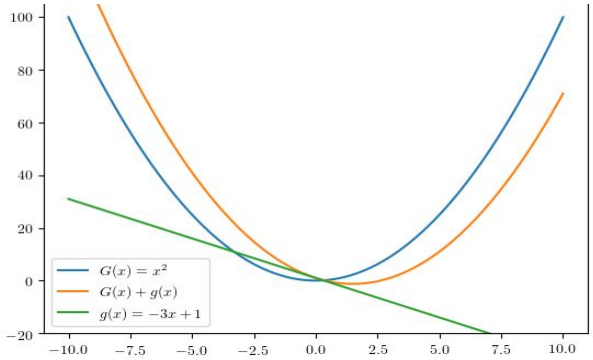
$h(w)$ is $\lambda-$strongly convex iff $(\nabla h(x) - \nabla h(y))^T (x - y) \geq \lambda ||x - y||_2^2$

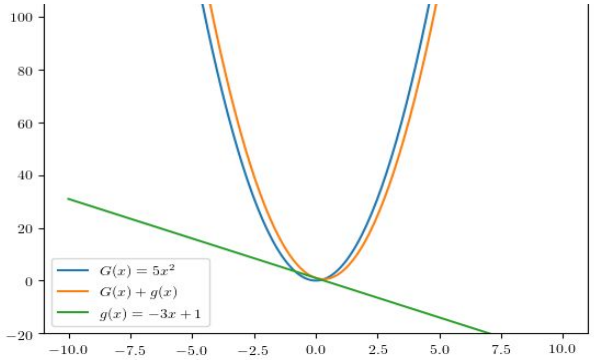
The sum of a convex function and a λ-strongly convex function is λ-strongly convex.

Private Logistic Regression

Lemma 7 *Let $G(\mathbf{f})$ and $g(\mathbf{f})$ be two vector-valued functions, which are continuous, and differentiable at all points. Moreover, let $G(\mathbf{f})$ and $G(\mathbf{f}) + g(\mathbf{f})$ be λ-strongly convex. If $\mathbf{f}_1 = \operatorname{argmin}_{\mathbf{f}} G(\mathbf{f})$ and $\mathbf{f}_2 = \operatorname{argmin}_{\mathbf{f}} G(\mathbf{f}) + g(\mathbf{f})$, then*

$$\|\mathbf{f}_1 - \mathbf{f}_2\| \leq \frac{1}{\lambda} \max_{\mathbf{f}} \|\nabla g(\mathbf{f})\| .$$





Private Logistic Regression

$h(w)$ is $\lambda-$strongly convex iff $(\nabla h(x) - \nabla h(y))^T (x - y) \geq \lambda ||x - y||_2^2$

The sum of a convex function and a λ-strongly convex function is λ-strongly convex.

$$L(w) = \frac{1}{N} \sum_{i \in \{1..N\}} \text{CrossEntropy}(\sigma(w^T x_i), y_i) + \frac{1}{2} \lambda w^T w$$

$$L'(w) = \left(\frac{1}{N} \sum_{i \in \{1..N\} \setminus j} \text{CrossEntropy}(\sigma(w^T x_i), y_i) \right) + \text{CrossEntropy}(\sigma(w^T x_{j'}), y_{j'}) + \frac{1}{2} \lambda w^T w$$

$$L'(w) = L(w) + g(w)$$

$$g(w) = \frac{1}{N} \left(\text{CrossEntropy}(\sigma(w^T x_{j'}), y_{j'}) - \text{CrossEntropy}(\sigma(w^T x_j), y_j) \right)$$

Assuming $||x|| \leq 1$, we have $\nabla g(w) \leq \frac{2}{N}$

$$\rightarrow \Delta L = \frac{2}{N\lambda}$$

Private Logistic Regression

Algorithm:

sensitivity $\Delta L = \frac{2}{N}$

1) solve $w^* = \arg\min_w L(w)$

2) draw η with $p(\eta) \propto \exp(-\frac{\Delta L}{\epsilon}||\eta||)$ (vector analogue of Laplace draw)

3) return $w = w^* + \eta$