CS562 PRESENTATION

VIKRAM DUVVUR
(VDUVVUR2)

SCALABLE PRIVATE LEARNING WITH PATE

Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, Úlfar Erlingsson

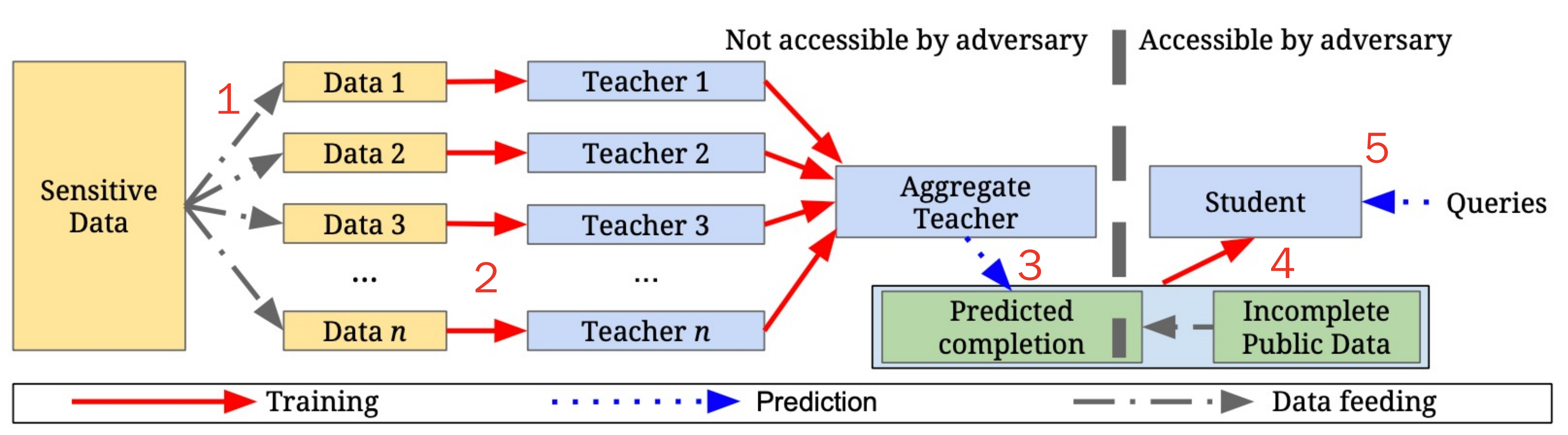
Problem Statement

- Protect private data
  - Sensitive medical data (HIPAA compliance)
  - Emails (Credit Card numbers, SSNs)
- Why is this so difficult?
  - Models unintentionally learn training data

- Authors propose improvement to Private Aggregation of Teacher Ensembles (PATE)

PATE Pipeline

1. Split private dataset into disjoint subsets
2. Train private teacher model for each subset
3. Predict labels using aggregate of predictions (with noise) on unlabeled public data
4. Train student model on newly labeled data
5. Use student model for predictions



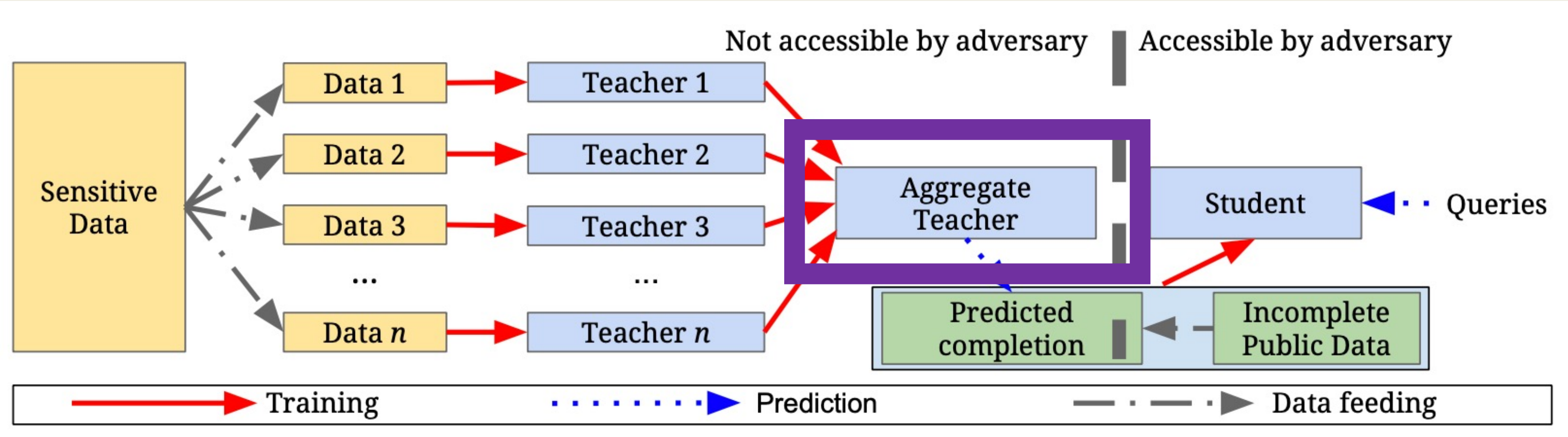
“Scalable” PATE

- Authors scale PATE to handle
  - Large numbers of output classes
  - Uncurated, imbalanced student training data

- Two ways to achieve this
  - Add less noise
  - Make student training data more selective

Three Proposed Aggregation Mechanisms

- Original Aggregator
  - Laplacian NoisyMax (LNMax)
- Gaussian NoisyMax (GNMax)
- Confident Aggregator (Confident-GNMax)
- Interactive Aggregator (Interactive-GNMax)



Privacy Budget

- Tradeoff between privacy and accuracy
- Upper bound on leakage
- ϵ = privacy budget
    - Maximum distance between a record on X and the same record on Y
    - ε = 0 means the output for the same query is the same on X and Y

- δ = probability of leakage (aka room for error)

Definition 2.4 (Differential Privacy). A randomized algorithm $\mathcal{M}$ with domain $\mathbb{N}^{|\mathcal{X}|}$ is (ε, δ)-differentially private if for all $\mathcal{S} \subseteq \text{Range}(\mathcal{M})$ and for all $x, y \in \mathbb{N}^{|\mathcal{X}|}$ such that $\|x - y\|_1 \leq 1$:

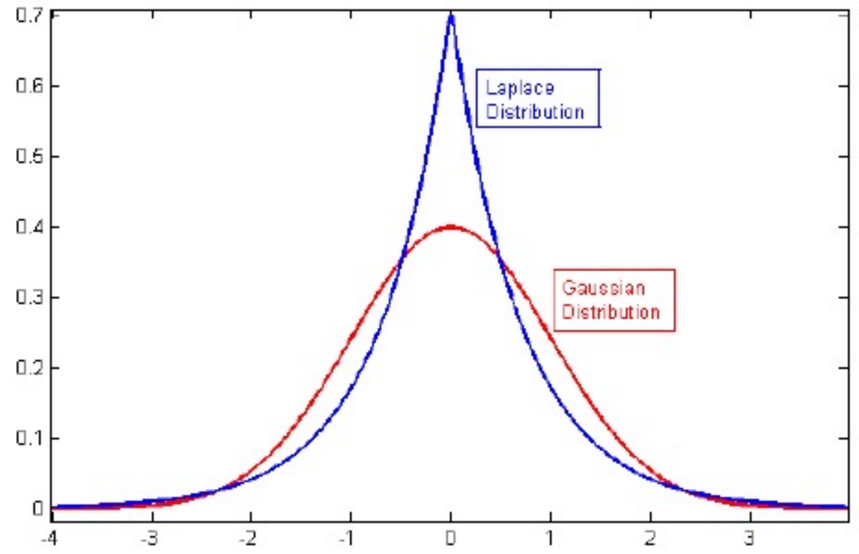
$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq \exp(\varepsilon) \Pr[\mathcal{M}(y) \in \mathcal{S}] + \delta,$$

Gaussian NoisyMax Aggregator

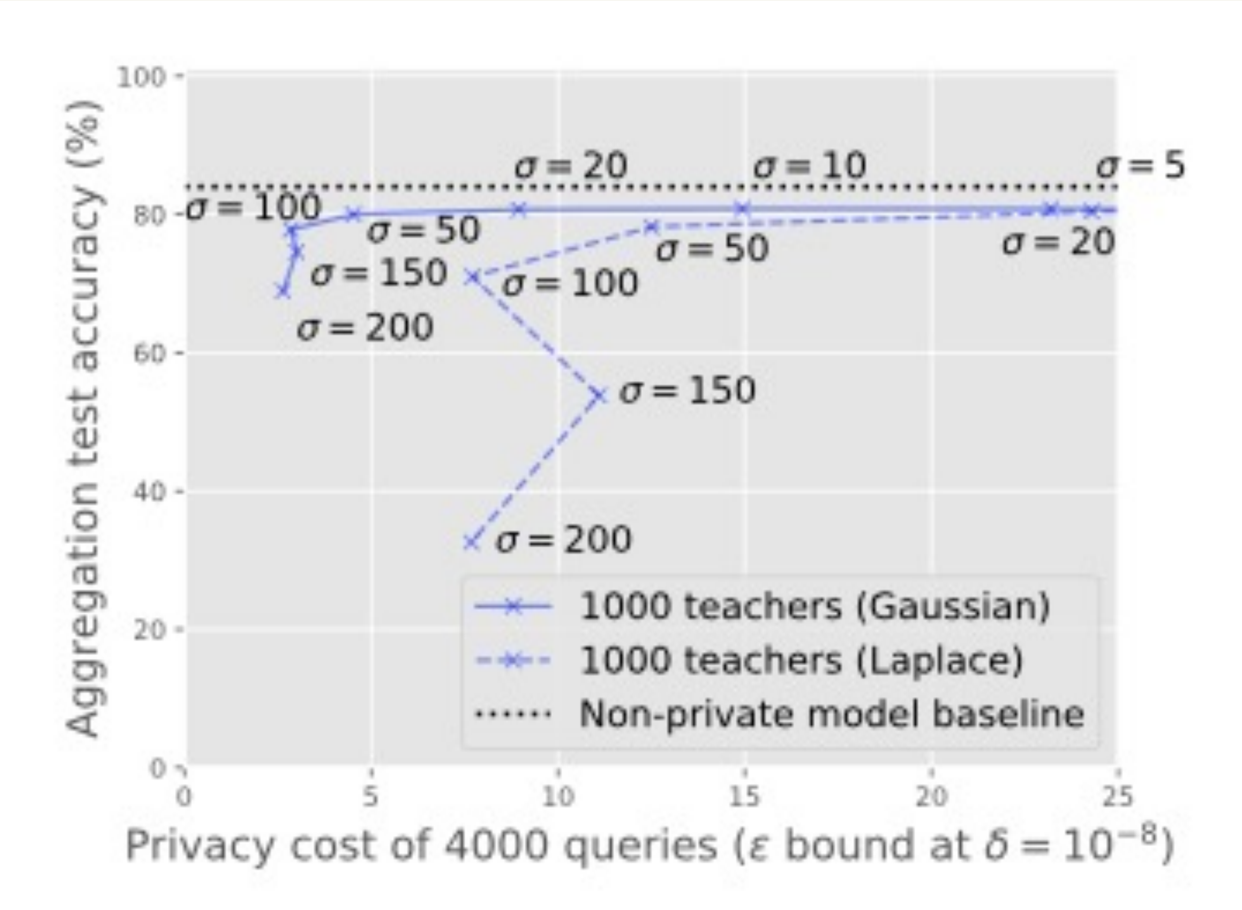
- ”Tail diminishes more rapidly than the Laplace distribution”

This section uses the following notation. For a sample x and classes 1 to m, let $f_j(x) \in [m]$ denote the j-th teacher model's prediction on x and $n_i(x)$ denote the vote count for the i-th class (i.e., $n_i(x) = |\{j : f_j(x) = i\}|$). We define a Gaussian NoisyMax (GNMax) aggregation mechanism as:

$$\mathcal{M}_\sigma(x) \triangleq \operatorname*{argmax}_i \left\{ n_i(x) + \mathcal{N}(0, \sigma^2) \right\},$$



Gaussian vs Laplacian



Confident Aggregator

- Check if teacher's max prediction score sum (with noise) add up to at least T
  - 0.6*teachers < T < 0.8*teachers
  - Use a high standard deviation for lower ε cost per query
- If they do, then use that training sample with a label from normal aggregation scheme and lower standard deviation
- Otherwise ignore sample

Confident Aggregator (Algorithm)

Algorithm 1 – Confident-GNMax Aggregator: given a query, consensus among teachers is first estimated in a privacy-preserving way to then only reveal confident teacher predictions.

Input: input x, threshold T, noise parameters σ_1 and σ_2

if $\max_i\{n_j(x)\} + \mathcal{N}(0, \sigma_1^2) \geq T$ **then** ▷ Privately check for consensus

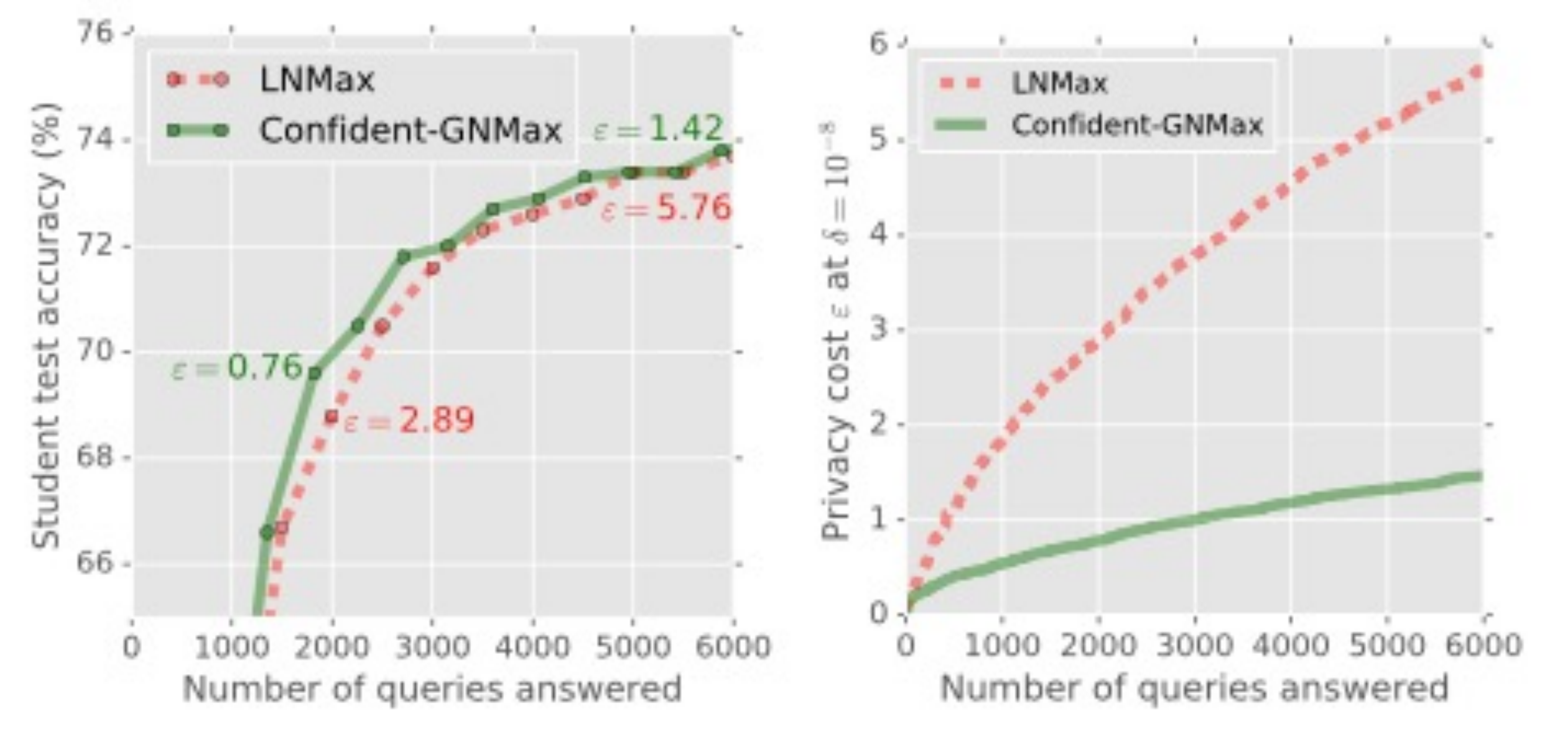
  return $\mathrm{argmax}_j \left\{n_j(x) + \mathcal{N}(0, \sigma_2^2)\right\}$ ▷ Run the usual max-of-Gaussian

else

  return $\perp$

end if

Confident Aggregator Stats



Interactive Aggregator

- Builds off confident aggregator
- Only use data if
  - Student predicts different answer teachers
  - Student predicts right answer confidently (> γ)
- Authors suggest using confident aggregator first, then interactive aggregator

Interactive Aggregator (Algorithm)

Algorithm 2 – Interactive-GNMax Aggregator: the protocol first compares student predictions to the teacher votes in a privacy-preserving way to then either (a) reinforce the student prediction for the given query or (b) provide the student with a new label predicted by the teachers.

Input: input x, confidence γ, threshold T, noise parameters σ_1 and σ_2, total number of teachers M

```
1: Ask the student to provide prediction scores p(x)
2: if max_j {n_j(x) − M p_j(x)} + N(0, σ_1^2) ≥ T then      ▷ Student does not agree with teachers
3:     return argmax_j {n_j(x) + N(0, σ_2^2)}                 ▷ Teachers provide new label
4: else if max{p_i(x)} > γ then                  ▷ Student agrees with teachers and is confident
5:     return arg max_j p_j(x)                                ▷ Reinforce student's prediction
6: else
7:     return ⊥                                               ▷ No output given for this label
8: end if
```

Empirical Evidence

Dataset	Aggregator	Queries answered	Privacy bound ε	Accuracy Student	Accuracy Baseline
MNIST	LNMax (Papernot et al., 2017)	100	2.04	98.0%	99.2%
	LNMax (Papernot et al., 2017)	1,000	8.03	98.1%	
	Confident-GNMax (T=200, σ_1=150, σ_2=40)	286	**1.97**	**98.5%**	
SVHN	LNMax (Papernot et al., 2017)	500	5.04	82.7%	92.8%
	LNMax (Papernot et al., 2017)	1,000	8.19	90.7%	
	Confident-GNMax (T=300, σ_1=200, σ_2=40)	3,098	**4.96**	**91.6%**	
Adult	LNMax (Papernot et al., 2017)	500	2.66	83.0%	85.0%
	Confident-GNMax (T=300, σ_1=200, σ_2=40)	524	**1.90**	**83.7%**	
Glyph	LNMax	4,000	4.3	72.4%	82.2%
	Confident-GNMax (T=1000, σ_1=500, σ_2=100)	10,762	2.03	**75.5%**	
	Interactive-GNMax, two rounds	4,341	**0.837**	73.2%	

Conclusion + Drawbacks

- Using PATE is an effective way to learn on private data while maintaining privacy guarantees
- By using two different aggregation schemes, we can get achieve near-baseline performance

- Drawbacks
    - Must have public version of dataset

Questions?

PLAUSIBLE DENIABILITY FOR PRIVACY-PRESERVING DATA SYNTHESIS

Vincent Bindschaedler, Reza Shokri, Carl A. Gunter

Problem Statement

- How do we release private datasets?
- Tradeoff between usability and privacy
- Issues that exist now
  - Imperfect deidentification methods
  - Requires domain knowledge
- Authors propose a generic theoretical framework for generating synthetic data in a privacy preserving manner
- Splits pipeline into two parts
  - Generative model
  - Privacy Test

Plausible Deniability

- A mechanism ensures plausible deniability if there at least k > 0 records that could have produced the same synthetic data with similar probability
- A data record is only published if it passes this privacy test, otherwise it is discarded
- Can be “attached” to any generative model
- Larger k and γ closer to 1 = stronger indistinguishability

Let $\mathcal{M}$ be a probabilistic generative model that given any data record d can generate synthetic records y with probability $\mathbb{P}\mathrm{r}\{y = \mathcal{M}(d)\}$. Let $k \geq 1$ be an integer and $\gamma \geq 1$ be a real number. Both k and γ are privacy parameters.

Definition 1 (Plausible Deniability). *For any dataset D with $|D| \geq k$, and any record y generated by a probabilistic generative model $\mathcal{M}$ such that $y = \mathcal{M}(d_1)$ for $d_1 \in D$, we state that y is releasable with (k, γ)-*plausible deniability, *if there exist at least $k - 1$ distinct records $d_2, ..., d_k \in D \setminus \{d_1\}$ such that*

$$\gamma^{-1} \leq \frac{\mathbb{P}\mathrm{r}\{y = \mathcal{M}(d_i)\}}{\mathbb{P}\mathrm{r}\{y = \mathcal{M}(d_j)\}} \leq \gamma, \qquad (1)$$

for any $i, j \in \{1, 2, \ldots, k\}$.

Example (k, γ)-PD Pipeline

- Inputs: Model (M), dataset (D), and privacy parameters k > 0 and $\gamma \geq 1$

- Steps:
  1. Sample a seed $d \in D$
  2. Generate synthetic record $y = M(d)$
  3. Run privacy test on (M, D, d, y, k, γ)
  4. If test passes, release data record otherwise toss it

Example Privacy Test

- Inputs: Model (M), dataset (D), records $d \in D$ and $y = M(d)$ and privacy parameters k and γ

- Steps:
  1. Find i such that
  $$\gamma^{-1-i} < \Pr\big(\mathrm{y} = M(d)\big) \leq \gamma^{-i}$$
  2. Find k', the number of records $d_a \in D$ such that
  $$\gamma^{-1-i} < \Pr\big(\mathrm{y} = M(d_{\mathrm{a}})\big) \leq \gamma^{-i}$$
  3. Pass test if $k' \geq k$

Example Privacy Test (with DP)

- Inputs: Model (M), dataset (D), records $d \in D$ and $y = M(d)$ and privacy parameters k and γ

- Steps:
  1. Let $\bar{k} = k + lap(\frac{1}{\epsilon_0})$
  2. Find i such that
     $$\gamma^{-1-i} < \Pr\big(\mathrm{y} = M(d)\big) \le \gamma^{-i}$$
  3. Find k', the number of records $d_a \in D$ such that
     $$\gamma^{-1-i} < \Pr\big(\mathrm{y} = M(d_\mathrm{a})\big) \le \gamma^{-i}$$
  4. Pass test if $k' \ge \bar{k}$

Relationship with DP

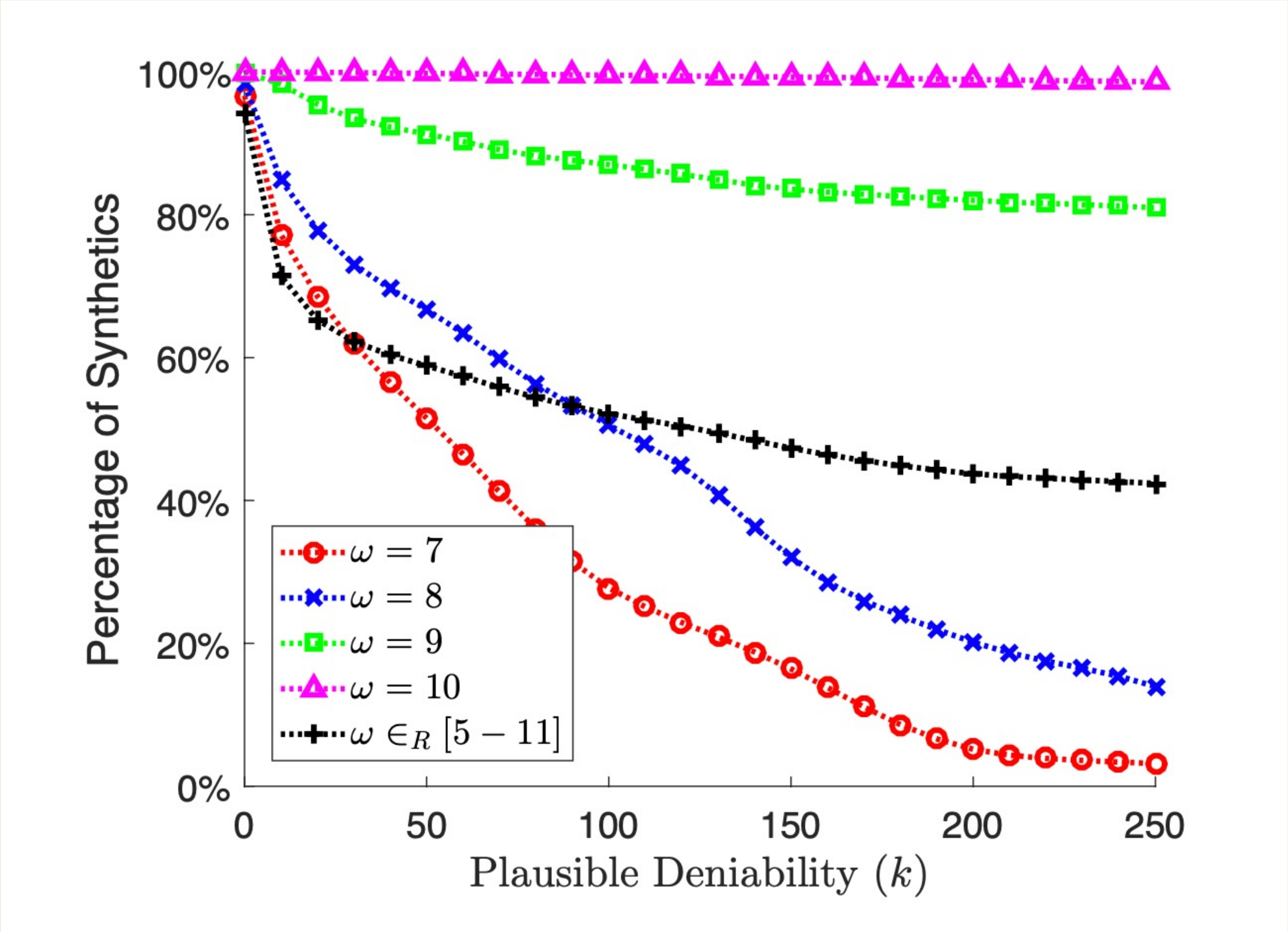
- If we use the example privacy test with differential privacy, we can obtain the following bound

Theorem 1 (Differential Privacy of $\mathcal{F}$). *Let $\mathcal{F}$ denote Mechanism 1 with the (randomized) Privacy Test 2 and parameters $k \geq 1$, $\gamma > 1$, and $\varepsilon_0 > 0$. For any neighboring datasets D and D' such that $|D|, |D'| \geq k$, any set of outcomes $Y \subseteq \mathcal{U}$, and any integer $1 \leq t < k$, we have:*

$$\mathbb{P}\mathrm{r}\{\mathcal{F}(D') \in Y\} \leq e^{\varepsilon}\mathbb{P}\mathrm{r}\{\mathcal{F}(D) \in Y\} + \delta \ ,$$

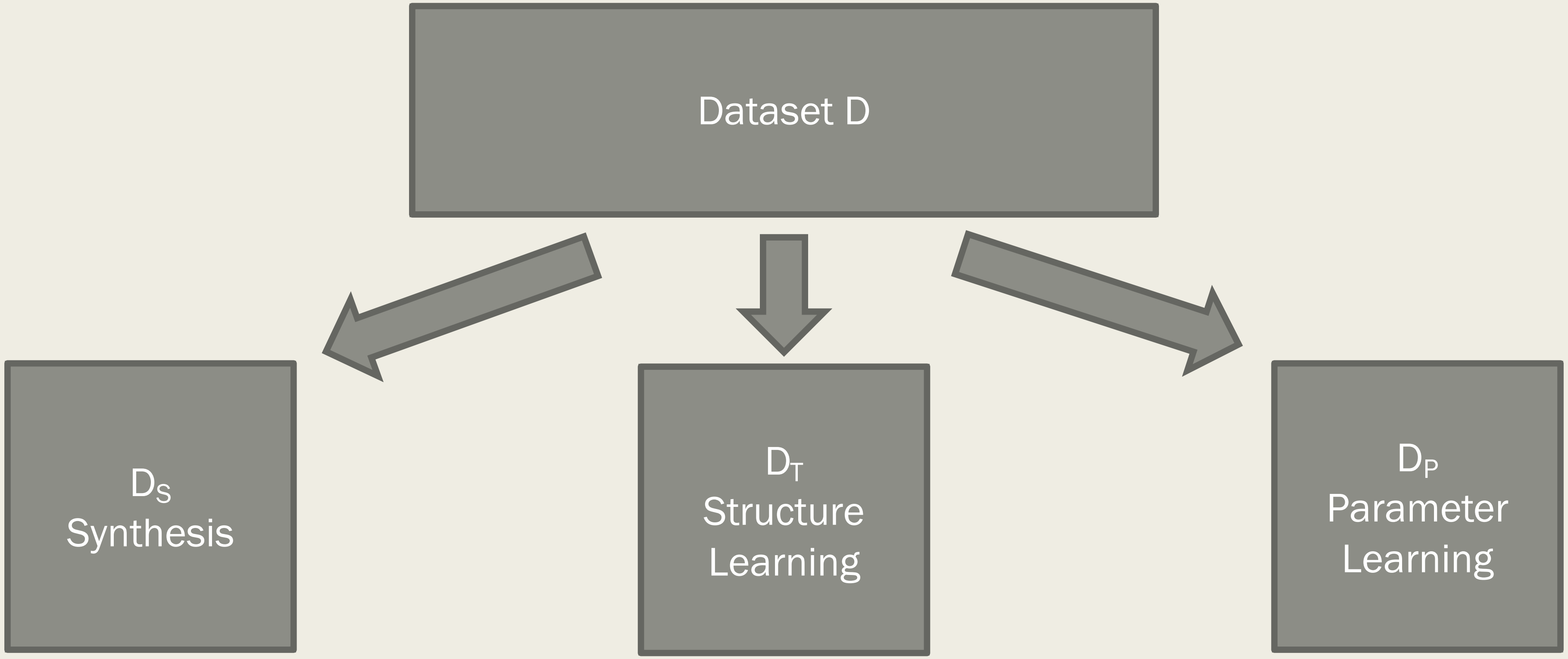
for $\delta = e^{-\varepsilon_0(k-t)}$ and $\varepsilon = \varepsilon_0 + \ln(1 + \frac{\gamma}{t})$.

Probability of Passing Privacy Test



Generative Model

- The proposed synthesizer is a probabilistic model that captures the joint distribution of attributes



Model

- Definitions
  - m = number of attributes
  - $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_\mathrm{m}\}$ = set of random variables associated with D
  - $\mathcal{G}$ = DAG where nodes are random variables and edges are dependencies
  - $P_\mathcal{G}(i)$ = set of parents for $\mathbf{x}_\mathrm{i}$

$$\mathbb{P}\mathrm{r}\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} = \prod_{i=1}^{m} \mathbb{P}\mathrm{r}\{\mathbf{x}_i \mid \{\mathbf{x}_j\}_{\forall j \in P_\mathcal{G}(i)}\}$$

Synthesis

- Creates new record by transforming a real data record (seed)
- Terms
  - ω = number of attributes for which new values are generated
  - σ = dependency order between random variables (topological order)
- Steps
  1. First fix $\{\sigma(1), \dots, \sigma(m-\omega)\}$ to be same as seed
  2. Resample $\sigma(i)$ for $i > m - \omega$ as

$$x'_{\sigma(i)} \sim \mathbb{P}\mathrm{r}\{\mathbf{x}_{\sigma(i)} \,|\, \{\mathbf{x}_{\sigma(j)} = x_{\sigma(j)}\}_{\forall j \in P_{\mathcal{G}}(i), j \leq m-\omega}, \\ \{\mathbf{x}_{\sigma(j)} = x'_{\sigma(j)}\}_{\forall j \in P_{\mathcal{G}}(i), j > m-\omega}\}$$

Creating Graph $\mathcal{G}$

- Use Correlation-based Feature Selection on D_T to build $\bar{\mathcal{G}}$, a dependency structure
- Add Laplacian noise to correlation to satisfy DP
- Modify Dirichlet distribution to use $\max(0, n + Lap\left(\frac{1}{\epsilon_p}\right))$ data records from D_P

Generator Differential Privacy

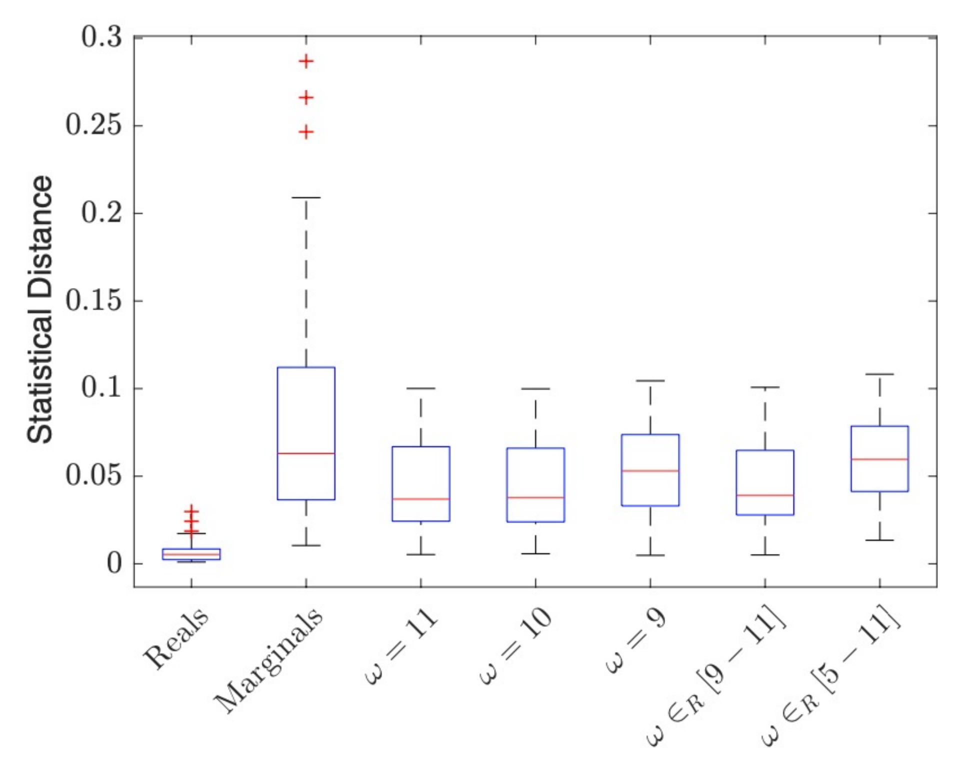
- Structural Learning (ϵ_L, δ_L)-differentially private
  - $\delta_L \ll \frac{1}{n_T}$

$$\epsilon_L = \epsilon_{n_T} + \epsilon_H \sqrt{2m(m+1)\ln(\delta_L^{-1})} + m(m+1)\epsilon_H(e^{\epsilon_H} - 1)$$

- Parameter Learning (ϵ_P, δ_P)-differentially private
  - $\delta_P \ll \frac{1}{n_p}$

$$\epsilon_P = \epsilon_p \sqrt{2m \ln(\delta_L^{-1})} + m\epsilon_p(e^{\epsilon_P} - 1)$$

How Good is the Generator with DP?



Pairwise distance to real records
(smaller is better)

	Accuracy		
	Tree	RF	Ada
Reals	77.8%	80.4%	79.3%
Marginals	57.9%	63.8%	69.2%
$\omega = 11$	72.4%	75.3%	78.0%
$\omega = 10$	72.3%	75.2%	78.1%
$\omega = 9$	72.4%	75.2%	77.5%
$\omega \in_R [9-11]$	72.3%	75.2%	78.1%
$\omega \in_R [5-11]$	72.1%	75.2%	78.1%

Classifiers trained on data

Distinguishing Game

	Reals	Marginals	$\omega =$ or $\in_R$				
			11	10	9	[9 – 11]	[5 – 11]
RF	50%	79.8%	62.3%	61.8%	63.0%	60.1%	61.4%
Tree	50%	73.2%	58.9%	58.6%	59.8%	57.9%	58.4%

Classifier's ability to distinguish real records from synthetics

Conclusion

- Introduced a new notion of "plausible deniability"
- Gives us a way to relate plausible deniability to differential privacy
- Proposed a mechanism to make any generative model have privacy guarantees
- Outlined a method to generate synthetic samples with differential privacy

Questions?