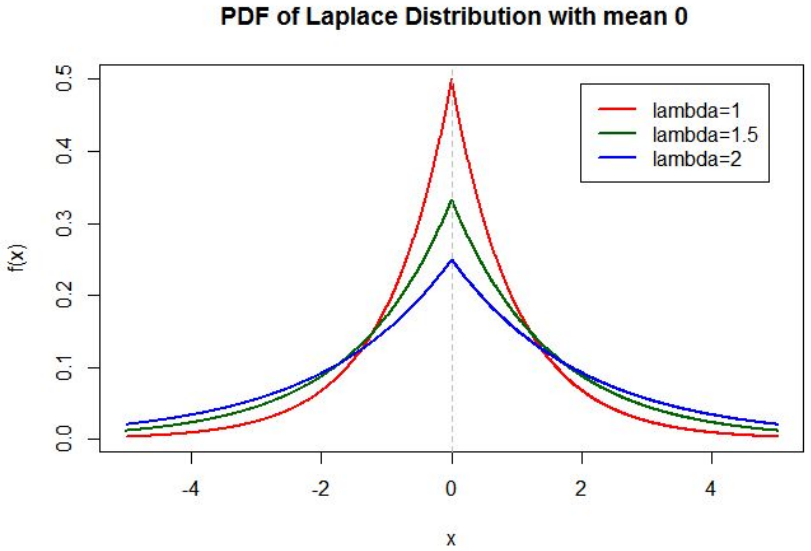
Incremental Generation of Differentially Private Yellow-Cab Datasets

By Ranger Kang, Rohan Kulkarni, Christopher Lee, Kevin Zhang

Columbia University Data Science Hackathon
February 11, 2023

Background

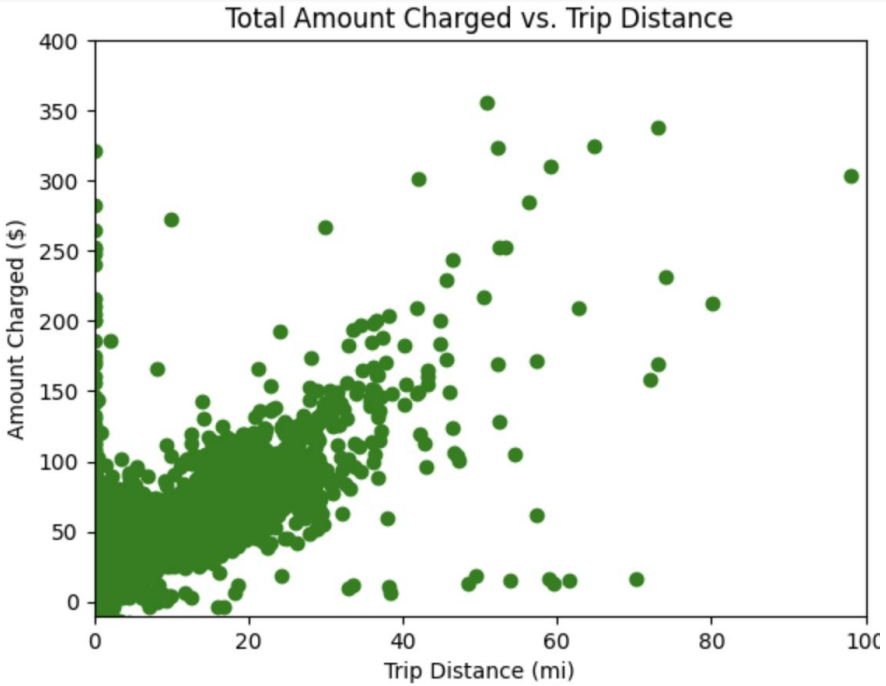
- Injecting privacy into datasets allows for the creation of differential privacy
- Increasing ε…
  - ensures the data is more true to the original! (accuracy increase)
  - but ensures the data is less private

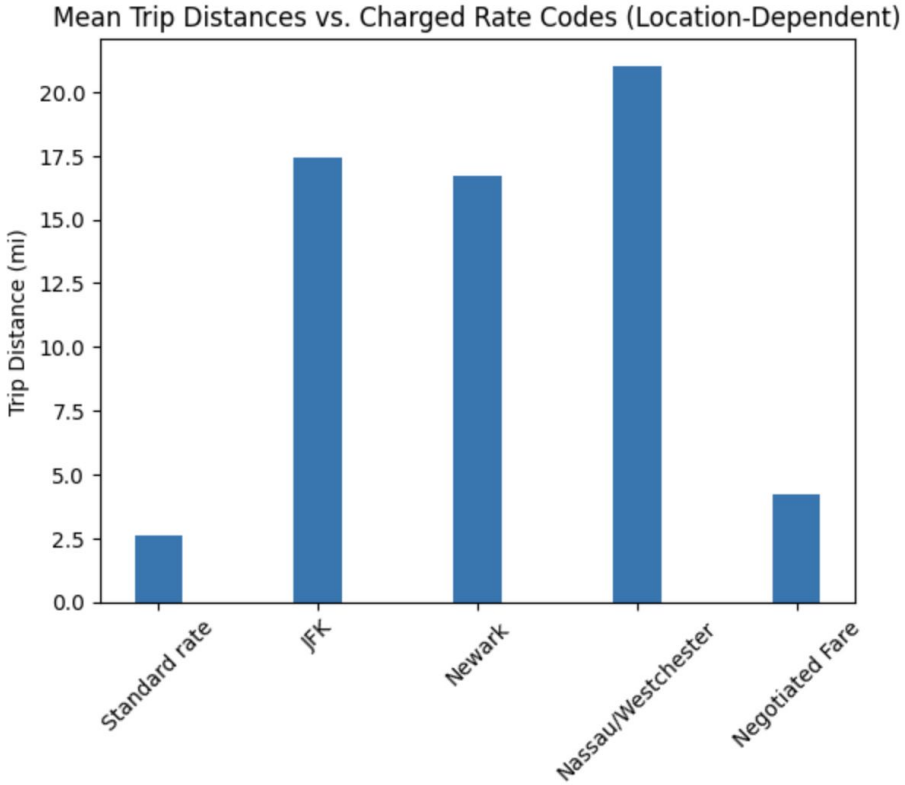


Background

- Goal: find an algorithm that provides a smooth, stable tradeoff between accuracy and privacy
- Overview: Created 4 ways of generating data on yellow cabs that differentially-privatizes the original dataset.
    - Methods incrementally build upon previous ones until we reach a **stable** dataset generating algorithm.

Exploratory Data Analysis (EDA)





Total amount → trip distance → charged rate codes!

What Columns in the Taxi Dataset Should Be Privatized?

PRIVATE	PRIVATE	UNMASKED
VendorID^{I}	Passenger_count$^{\Delta}$	Store_and_fwd_flag
tpep_pickup_datetime^{T}	Fare_amount*	Payment_type
tpep_dropoff_datetime^{T}	Extra*	MTA_tax
PULocationID^{T}	Tolls_amount*	Improvement_surcharge
DOLocationID^{T}	Total_amount*	Congestion_Surcharge
Trip_distance^{T}		
Airport_fee^{T}		
RateCodeID^{T}		
Tip_amount$^{\Delta}$		

* monetary

T time/location-based

I IDs

Δ conventionally private

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Methodologies

- 4 different ways of generating differentially private datasets
- Generating datasets based on maximizing accuracy for any given epsilon
    - Used [‘total_amounts’] column as our prediction label
- Want to also maintain privacy, privacy-accuracy tradeoff

$$\mathrm{Acc}(\mathcal{M}) := \frac{\text{number of correct tests}}{\text{total tests}}$$

Total Anonymity

Procedure:

1. Obtain the mean of each column
2. Repeat the mean of the column for *n* trials
3. Replace values in the dataset with sampled values

This is our “dummy” dataset. We do not expect it “preserve” our dataset well. However, this is a fantastically private dataset.

	VendorID	passenger_count	trip_distance	RatecodeID	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_a
0	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
1	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
2	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
3	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
4	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
...	...	...	...	...	...	...	...	...	...	...	...	
149995	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
149996	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
149997	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
149998	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3
149999	1.707967	1.388124	6.176475	1.415196	166.13058	163.662647	1.19406	12.789388	1.007348	0.4912	2.386165	0.3

MIN/MAX - Uniform Sampling

Procedure:

1. Obtain minimum and maximum values from relevant columns
2. Sample from uniform distribution
   a. Parameters: (min, max)
3. Replace values in the dataset with sampled values

This is less private than the last mechanism, as we now have values that follow a range including the mean, but the accuracy obviously decreases.

	VendorID	passenger_count	trip_distance	RatecodeID	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	toll
0	3.338731	3.179347	70790.765882	6.659970	123.799886	98.779460	2.016626	441.863140	4.637739	0.467898	32.206667	
1	5.760839	1.352552	69643.441152	66.401029	239.090405	204.341607	3.819824	92.958960	-1.775825	0.346257	70.225937	
2	5.273762	4.561921	83884.546171	42.175994	30.738391	91.347866	3.718687	611.835448	2.158248	0.041038	57.051800	-
3	3.607802	1.477879	39431.100628	11.858244	227.848935	16.498775	1.548194	-78.511412	-2.439366	0.297460	26.037130	
4	4.738844	3.972902	90717.477608	60.600414	13.413479	41.403228	0.349180	-113.531202	6.247777	-0.036001	51.407615	
...	...	...	...	...	...	...	...	...	...	...	...	...
149995	2.409924	3.502994	95284.554441	79.101721	102.987006	203.649592	3.258832	-100.488511	3.637518	-0.096018	26.464675	
149996	3.072872	4.197548	24889.005933	47.183698	201.867344	256.577914	3.847345	198.076672	8.668389	-0.356244	78.098224	
149997	3.695330	1.145630	20792.909170	95.009073	231.553976	171.271052	3.645579	144.756013	12.547464	-0.159230	21.141662	
149998	3.385442	1.154653	62267.263631	18.595748	76.913651	75.121029	1.564011	303.003031	8.929207	-0.050296	62.245953	
149999	1.184871	5.558528	81412.043214	53.777296	183.105617	248.331298	1.645137	-62.745505	4.620553	-0.064277	76.919088	

Gaussian Sampling

Consider the following theorem:

Theorem 2.7. *Define the Gaussian mechanism that operates on a statistic* θ *as* $M(S) = \theta(S) + \xi$, *where* $\xi \sim \mathcal{N}(0, \text{sens}(\theta)^2/\mu^2)$. *Then,* M *is* μ*-GDP.*

Proof of Theorem 2.7. Recognizing that $M(S), M(S')$ are normally distributed with means $\theta(S), \theta(S')$, respectively, and common variance $\sigma^2 = \text{sens}(\theta)^2/\mu^2$ we get

$$T\big(M(S), M(S')\big) = T\big(\mathcal{N}(\theta(S), \sigma^2), \mathcal{N}(\theta(S'), \sigma^2)\big) = G_{|\theta(S)-\theta(S')|/\sigma}.$$

By the definition of sensitivity, $|\theta(S) - \theta(S')|/\sigma \leqslant \text{sens}(\theta)/\sigma = \mu$. Therefore, we get

$$T\big(M(S), M(S')\big) = G_{|\theta(S)-\theta(S')|/\sigma} \geqslant G_{\mu}.$$

Conclusion: utilize Gaussian sampling which considers the chosen epsilon

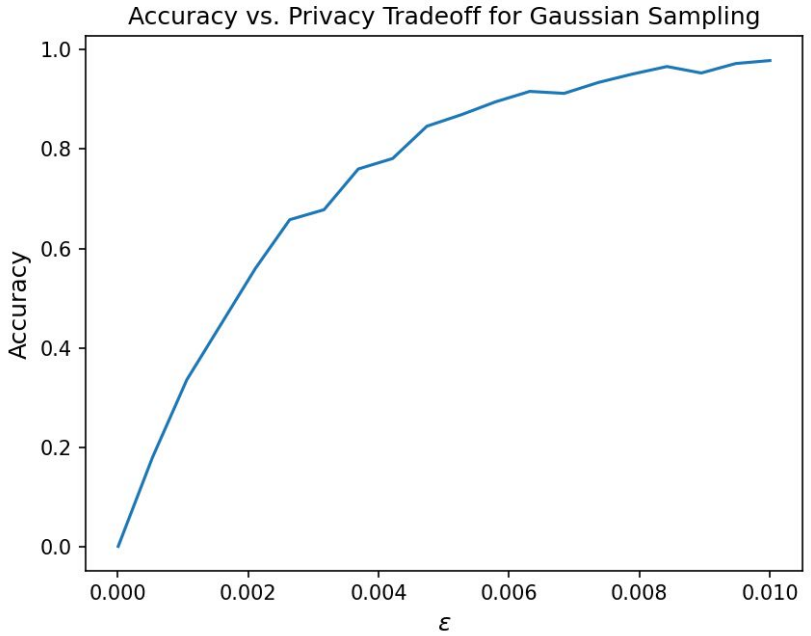
https://arxiv.org/pdf/1905.02383.pdf

Gaussian Sampling

$$\mathcal{M}_G(x, f, \epsilon) = f(x) + \mathcal{N}\left(\mu = 0, \sigma^2 = \frac{\Delta f^2}{\epsilon^2}\right) \qquad \Delta f = \max |f(x) - f(y)|$$

Procedure:

1. Find the mean-variance vectors of each column
2. Create Gaussian distributions with these parameters (μ_i, σ_i^2)
3. Sample from Gaussian distribution for *n* trials
4. Replace relevant values in data with sampled values



COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Gaussian Sampling

	VendorID	passenger_count	trip_distance	RatecodeID	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_
0	1.220705	1.552790	521.503841	6.862356	137.254470	229.260781	0.651561	23.042456	-0.229327	0.536002	-1.976728	0
1	0.623234	0.898858	493.002576	-0.627381	70.256473	223.099411	0.030060	-2.378968	1.564975	0.326175	5.871556	-0
2	1.839708	1.210543	404.127057	-2.546555	221.083206	171.007375	0.777893	-3.100851	3.016977	0.548094	7.883974	-1
3	2.390426	0.796215	1255.379235	-5.843365	148.881738	105.246186	1.650683	24.177653	2.128012	0.504584	3.539527	-0
4	1.656673	2.119950	-357.513163	-5.137519	23.916848	53.637246	0.873394	-4.828891	0.146799	0.624246	2.290696	-1
...	...	...	...	...	...	...	...	...	...	...	...	
149995	1.581105	2.466621	887.296574	7.195171	111.433979	204.395066	1.400382	-9.651370	1.204373	0.489740	7.358437	1
149996	1.835834	-0.933780	484.044977	-3.273024	73.466092	354.917680	1.312269	18.173013	4.024306	0.517934	5.569467	-0
149997	1.754836	1.423789	708.111718	2.212495	146.834112	172.913044	1.305010	-5.789702	0.783169	0.508863	0.892605	-1
149998	2.137768	3.200621	17.614606	3.997812	134.276708	103.088985	1.868122	20.200099	-0.047357	0.390066	-0.626092	-0
149999	1.389079	0.583692	176.089729	-2.964317	169.415168	65.703086	0.921389	10.998232	-0.166683	0.586900	6.370011	0

Some values are negative - and this is okay!

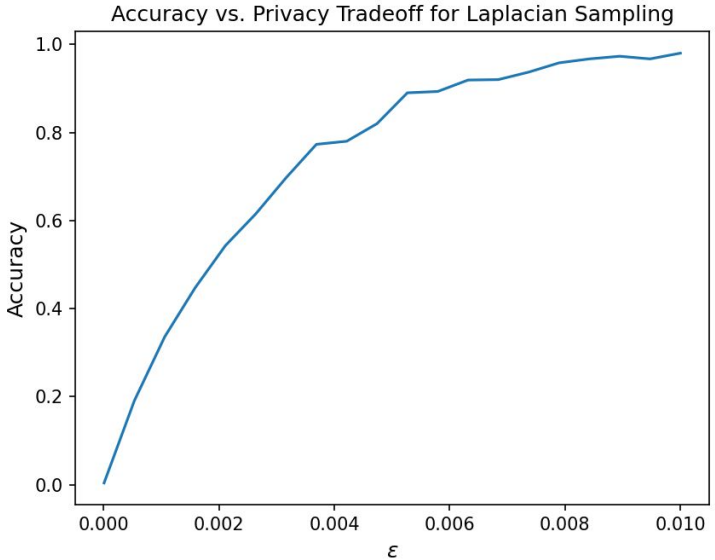
Laplacian Sampling

$$\mathcal{M}_{\mathrm{Lap}}(x, f, \epsilon) = f(x) + \mathrm{Lap}\left(\mu = 0, b = \frac{\Delta f}{\epsilon}\right)$$

$$\Delta f = \max |f(x) - f(y)|$$

Procedure:

1. Discover sensitivity of data for each column;choose epsilon
2. Create Laplacian distribution with corresponding *b* value
3. Replace respective values in the dataset with new sampled values



https://en.wikipedia.org/wiki/Additive_noise_mechanisms#Laplace_Mechanism

GUPT: Sampling and Aggregate

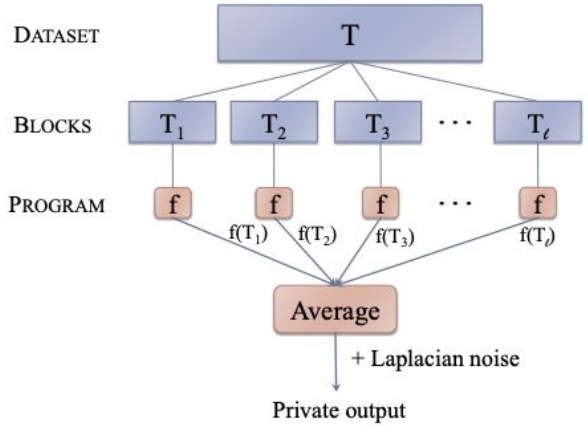
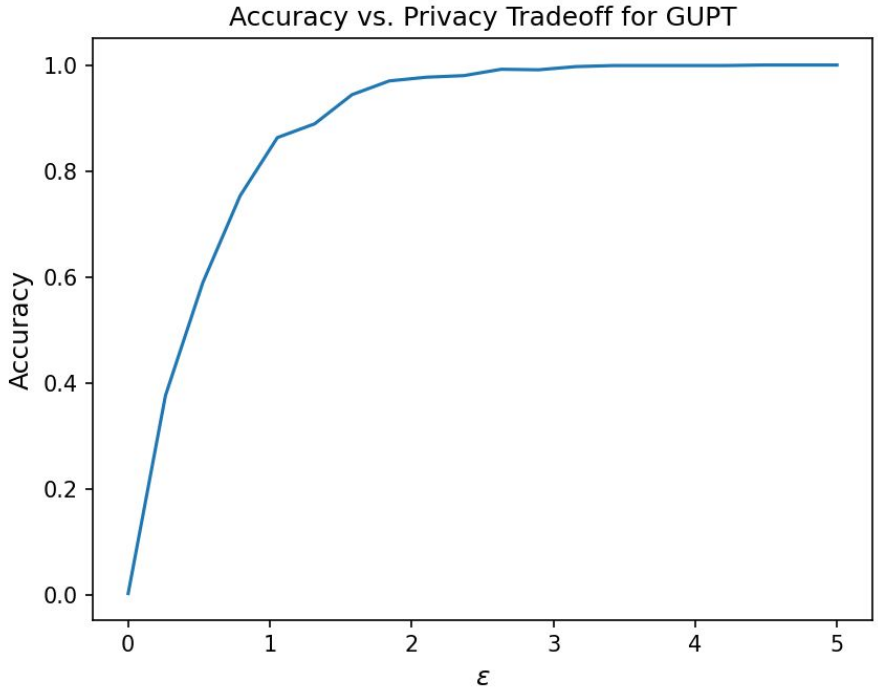


Figure 1: An instantiation of the Sample and Aggregate Framework [24].

Procedure:

1. Fix epsilon
2. Choose an (optimal) block size
3. Split data, sample and aggregate
4. Average results, add Laplacian noise

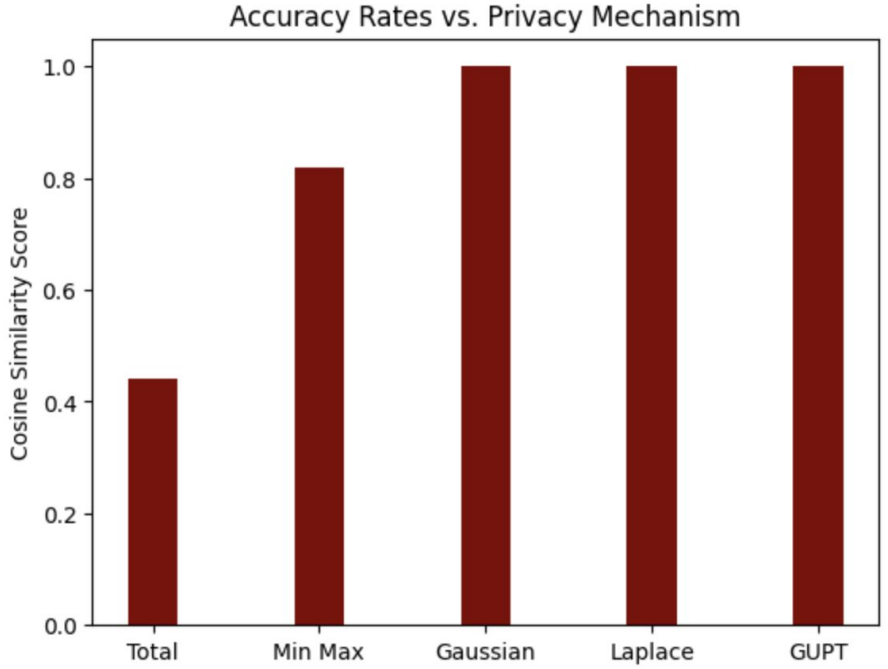


Comparing Mechanisms - Similarity Score

Similarity Score is defined by:

$$s := \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbf{v}(\mu, \sigma^2) \cdot \mathbf{u}(\mu, \sigma^2)}{\|\mathbf{v}(\mu, \sigma^2)\| \cdot \|\mathbf{u}(\mu, \sigma^2)\|}$$

1. Compute the mean-variance vectors of each of the columns
2. Compare these to the mean-variance vectors of the original dataset using cosine similarity
3. Normalize the score to the number of trials *n*



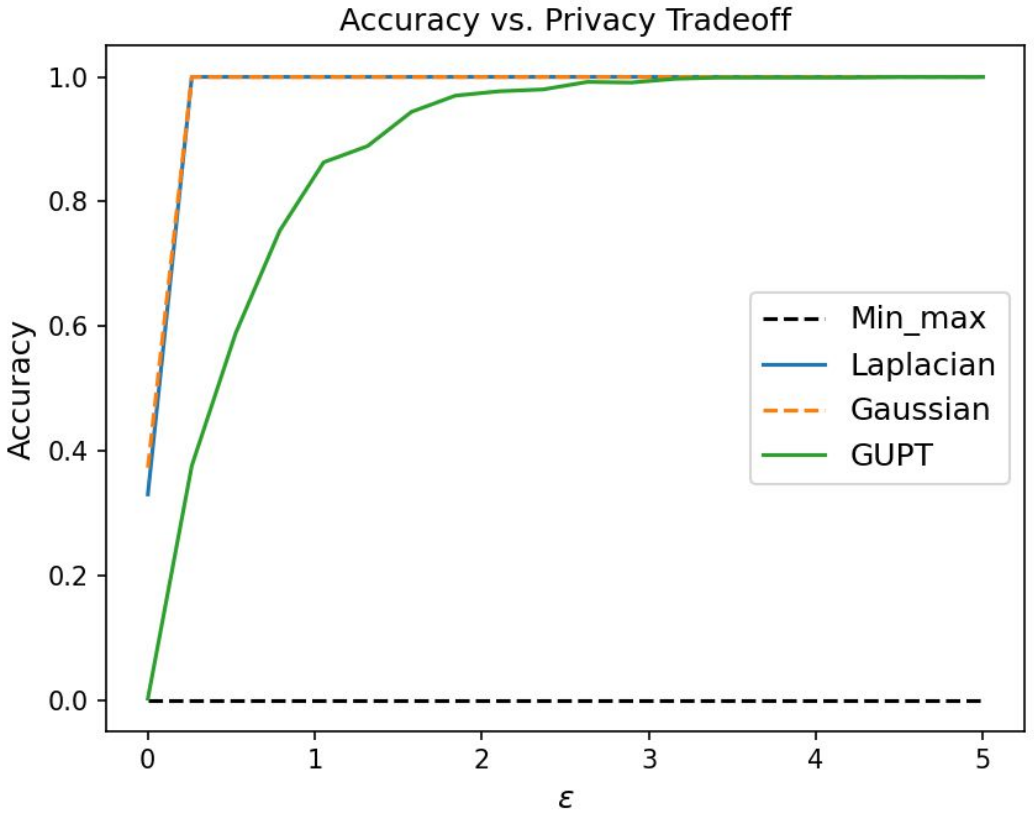
COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Summary

Conclusions:

1. GUPT creates a smoothly increasing function of epsilon to accuracy
2. for GUPT, minor differences in epsilon will not vastly affect accuracy

We utilize GUPT because it gives the most balanced tradeoff between privacy and accuracy.



Thank you! :)