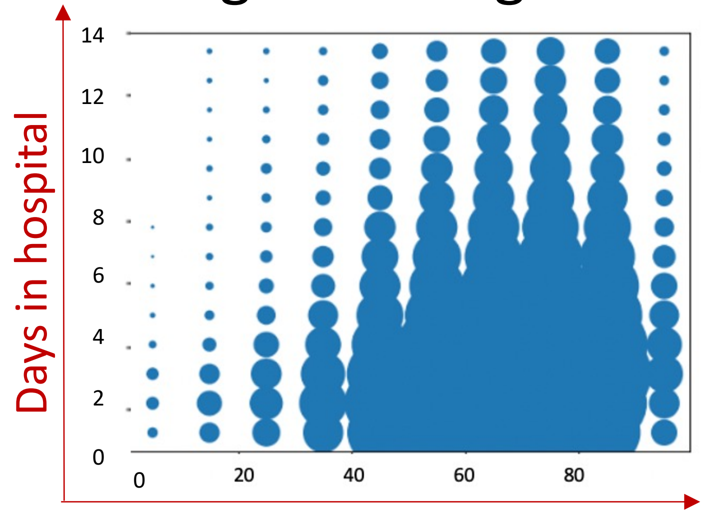
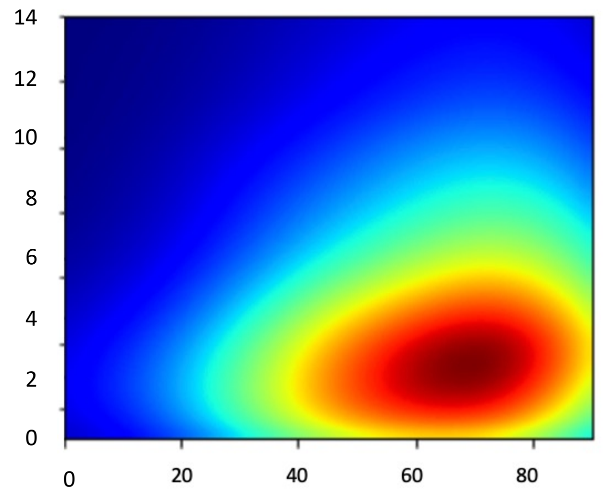
Private Kernel Density Estimation without the Curse of Dimensionality

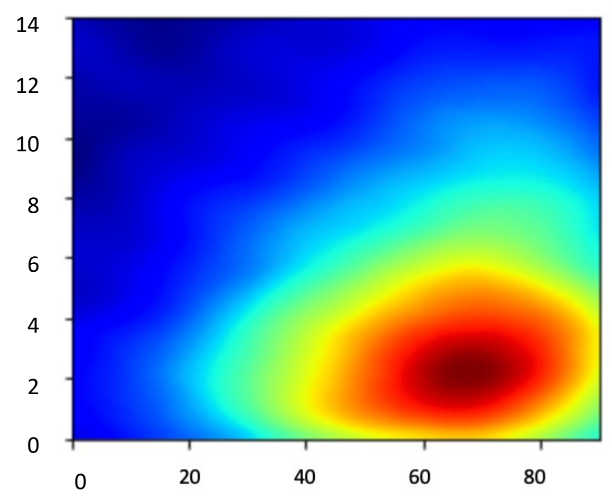
Original histogram



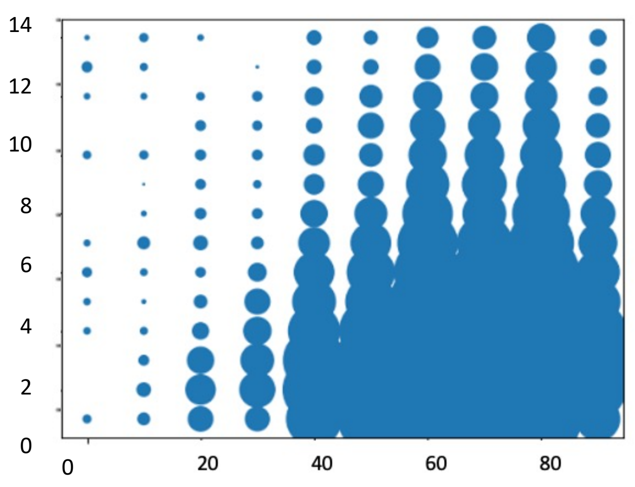
Original KDE



Differentially private KDE



Private synthetic histogram



The Gaussian KDE of a dataset $x_1, \dots, x_n \in \mathbb{R}^d$ is the function that maps $y \in \mathbb{R}^d \longrightarrow \frac{1}{n}\sum_{i=1}^{n} e^{-\|y-x_i\|_2^2}$

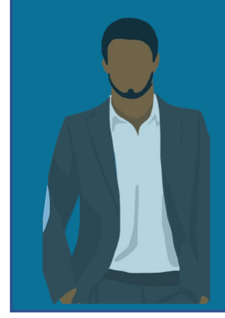
Differentially private Gaussian KDE:

Curator

- Has private dataset $x_1, \dots, x_n \in \mathbb{R}^d$
- Releases a function $\widetilde{K}: \mathbb{R}^d \to \mathbb{R}$
- $\widetilde{K}$ must be ε-DP w.r.t. the dataset
- $\widetilde{K}$ should approximate the Gaussian KDE

$\widetilde{K}$ ⟶

Client



- Receives $\widetilde{K}$
- For each query $y \in \mathbb{R}^d$, w.h.p.:

$$\widetilde{K}(y) \approx \frac{1}{n}\sum_{i=1}^{n} e^{-\|y-x_i\|_2^2}$$

Our results:

- High dimensions: ε-DP, error $\sim 1/\sqrt{n}$, runtime linear in d ⟶ *no curse of dimensionality*
- Low dimensions: ε-DP, error $\sim (\log n)^{O(d)}/n$, runtime exp. in d ⟶ *near-linear error decay if $d = O(1)$*

Tal Wagner, Yonatan Naamad, Nina Mishra

Yes, this is a "#betterposter", for #better or worse

The Technical Stuff:

Fast Private Kernel Density Estimation via Locality Sensitive Quantization

What is LSQ? Expressing a kernel on $\mathbb{R}^d$ with features that are *few, bounded,* and *sparse*.

Formally: $k(x,y)$ is (Q,R,S)-*LSQable* if there is a distribution $\mathcal{D}$ over pairs of functions $f,g: \mathbb{R}^d \to [-R,R]^Q$, such that for all $x,y \in \mathbb{R}^d$:

- $f(x)$ and $g(y)$ have $\leq S$ non-zeros
- $k(x,y) \approx \mathbb{E}_{(f,g)\sim\mathcal{D}}[f(x)^T g(y)]$

Theorem: LSQ ⇒ ε-DP KDE.
And, if Q, R, S are small, the mechanism has good utility and computational efficiency.

LSQ Constructions:

- Random Fourier Features (RFF) [Rahimi-Recht '07]
  - Leads to our high-dimensional result
- Fast Gauss Transform (FGT) [Greengard-Strain '91]
  - Leads to our low-dimensional result
- Locality Sensitive Hashing (LSH) [Indyk-Andoni '09]
  - Recovers prior results of [Coleman-Shrivastava '21]
  - LSQ extends LSH to more kernels (e.g., Gaussian)

Prior work:

	Method	Privacy	Error decay	Runtime in d	
Prior	[Several]	ε-DP	$\sim 1/\sqrt{n}$	$\exp(d)$	
	[HRW'13]	(ε,δ)-DP	$\sim 1/n$	$\exp(d)$	Unless query known ahead
	[CS'21]	ε-DP	$\sim 1/\sqrt{n}$	$O(d)$	LSH kernels, not Gaussian
Ours	LSQ-RFF	ε-DP	$\sim 1/\sqrt{n}$	$O(d)$	
	LSQ-FGT	ε-DP	$\sim (\lg n)^{O(d)}/n$ $\sim 1/n$ if $d = O(1)$	$\exp(d)$	

Does it work for other kernels?
Yes, but fineprint, see paper.

Paper, code, etc.: