Differentially Private Heatmaps

Badih Ghazi, Junfeng He, Kai Kohlhoff, Ravi Kumar
Pasin Manurangsi, Vidhya Navalpakkam, Nachiappan Valliappan

Google Research, Mountain View, CA
{badihghazi, ravi.k53}@gmail.com
{junfenghe,kohlhoff,pasin,vidhyan,nac}@google.com

Abstract

We consider the task of producing heatmaps from users' aggregated data while protecting their privacy. We give a differentially private (DP) algorithm for this task and demonstrate its advantages over previous algorithms on real-world datasets.

Our core algorithmic primitive is a DP procedure that takes in a set of distributions and produces an output that is close in Earth Mover's Distance to the average of the inputs. We prove theoretical bounds on the error of our algorithm under a certain sparsity assumption and that these are near-optimal.

1 Introduction

Recently, differential privacy (DP) (Dwork et al. 2006b,a) has emerged as a strong notion of user privacy for data aggregation and machine learning, with practical deployments including the 2022 US Census (Abowd 2018), in industry (Erlingsson, Pihur, and Korolova 2014; Shankland 2014; Greenberg 2016; Apple Differential Privacy Team 2017; Ding, Kulkarni, and Yekhanin 2017) and in popular machine learning libraries (Radebaugh and Erlingsson 2019; Testuggine and Mironov 2020). Over the last few years, DP algorithms have been developed for several analytic tasks involving aggregation of user data.

One of the basic data aggregation tools is a heatmap. Heatmaps are popular for visualizing aggregated data in two or higher dimensions. They are widely used in many fields including computer vision and image processing, spatial data analysis, bioinformatics, etc. Many of these applications involve protecting the privacy of user data. For example, heatmaps for gaze or gene microdata (Liu et al. 2019; Steil et al. 2019) would be based on data from individuals that would be considered private. Similarly, a heatmap of popular locations in a geographic area will be based on user location check-ins, which are sensitive. Motivated by such applications, in this paper, we present an efficient DP algorithm for computing heatmaps with provable guarantees, and evaluate it empirically.

At the core of our algorithm is a primitive solving the following basic task: how to privately aggregate sparse input vectors with a small error as measured by the Earth Mover's Distance (EMD)? While closely related to heatmaps, the EMD measure is of independent interest: it was originally proposed for computer vision tasks (Rubner, Tomasi, and Guibas 2000) since it matches perceptual similarity better than other measures such as ℓ_1, ℓ_2, or KL-divergence (Stricker and Orengo 1995; Levina and Bickel 2001; Wang and Guibas 2012). It is also well-suited for spatial data analysis since it takes the underlying metric space into account and considers "neighboring" bins. EMD is used in spatial analysis (Kranstauber, Smolla, and Safi 2017), human mobility (Isaacman et al. 2012), image retrieval (Rubner, Tomasi, and Guibas 1998; Puzicha et al. 1999), face recognition (Xu, Yan, and Luo 2008), visual tracking (Zhao, Yang, and Tao 2008), shape matching (Grauman and Darrell 2004), etc. For the task of sparse aggregation under EMD, we give an efficient algorithm with asymptotically tight error. We next describe our results in more detail.

1.1 Our Results

We consider the setting where each user i holds a probability distribution $\mathbf{p}_i$ over points in $[0,1)^2$, and the goal is to compute the heatmap of the average of these probabilities, i.e., $\frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i$. We give an ε-DP algorithm for this task, establish its theoretical guarantees, and provide empirical evaluations of its performance. (For definitions, see Section 2.)

Sparse Aggregation under EMD. At the heart of our approach is the study of aggregation under EMD[1], where we would like to output the estimate of $\frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i$ with the error measured in EMD. There are two main reasons why we consider EMD for the error measure. First, a bound on the EMD to the average distribution implies bounds on several metrics commonly used in evaluating heatmaps, including the KL-divergence, ℓ_1 distance, and EMD itself. Second, while it is possible to obtain DP aggregation algorithms with bounded EMD error, as we will discuss below, any DP aggregation algorithm must suffer errors under other metrics, including KL-divergence or ℓ_1 distance, that grow with the resolution[2], rendering them impractical when the number of users is small compared to the resolution.

[1] For a formal definition of EMD, please see Section 2.1.

[2] Specifically, it follows from previous work (Dwork et al. 2015) that, if we consider the ℓ_1 distance or KL-divergence for $\Delta \times \Delta$ grid and $n \leq O_\varepsilon(\Delta)$, then the error must be $\Omega(1)$.

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

When the distributions $\mathbf{p}_i$'s are arbitrary, we show that a simple ε-DP algorithm yields a guarantee of $O_\varepsilon(1/\sqrt{n})$ on EMD, and that this bound is essentially optimal. While this is already a reasonable bound, we can improve on it by exploiting a property that is commonly present in distributions used for aggregations: "sparsity" (Cormode et al. 2012a).

Following the literature on compressed sensing (Indyk and Price 2011; Backurs et al. 2016), we define our approximation guarantee for the *sparse EMD aggregation* problem with respect to the best k-sparse distribution[3] that approximates the average $\mathbf{a} := \frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i$ under EMD. More formally, we say that an output distribution $\hat{\mathbf{a}}$ is a (λ, κ)-*approximation* for sparse EMD aggregation if

$$\mathrm{EMD}\left(\hat{\mathbf{a}}, \mathbf{a}\right) \leq \lambda \cdot \min_{k\text{-sparse } \mathbf{a}'} \mathrm{EMD}\left(\mathbf{a}', \mathbf{a}\right) + \kappa,$$

where $\lambda, \kappa > 0$ denote the multiplicative approximation ratio and additive error respectively.

Our main algorithmic contribution is in showing that under such a sparse approximation notion, we can achieve an error of only $O_\varepsilon(\sqrt{k}/n)$ and that this is tight.[4]

Theorem 1.1 (Informal)**.** *There exists an ε-DP algorithm that, for any constant $\lambda \in (0, 1)$, can output a $(\lambda, O_\varepsilon(\sqrt{k}/n))$-approximation for sparse EMD aggregation w.p. 0.99. Furthermore, no ε-DP algorithm can output a $(\lambda, o_\varepsilon(\sqrt{k}/n))$-approximate solution w.p. 0.1.*

Due to a known connection between sparse EMD aggregation and k-median clustering on the plane (Indyk and Price 2011; Backurs et al. 2016), our result also yields an improved DP algorithm for the latter. Due to space constraints, we omit the formal statement of our k-median results.

Experimental results. We test our algorithm on both real-world location datasets and synthetic datasets. The results demonstrate its practicality even for moderate values of $\varepsilon \in [0.5, 5]$ and a number of users equal to 200. Furthermore, we compare our algorithm with simple baselines; under popular metrics for heatmaps, our results demonstrate significant improvements on these regimes of parameters.

1.2 Overview of Techniques

At a high level, our algorithm is largely inspired by the work of Indyk and Price (2011) on *compressed sensing* under EMD. Roughly speaking, in compressed sensing, there is an underlying vector $\mathbf{x}$ that is known to be well-approximated by a sparse vector; we have to provide a matrix $\mathbf{A}$ such that, when we observe the measurements $\mathbf{Ax}$, we can reconstruct $\mathbf{x}'$ that is close to $\mathbf{x}$ (under a certain metric). This can of course be done trivially by taking $\mathbf{A}$ to, e.g., be the identity matrix. Thus, the objective is to perform this recovery task using as few measurements (i.e., number of rows of $\mathbf{A}$) as possible. There is a rich literature on compressive sensing; most relevant to our work are the prior papers studying compressive sensing with EMD, in particular, Indyk and Price (2011) and Backurs et al. (2016).

Indyk and Price (2011) presented an elegant framework for reducing the compressed sensing problem under EMD to one under ℓ_1, which is well-studied (see, e.g., Berinde et al. 2008; Berinde, Indyk, and Ruzic 2008; Indyk and Ruzic 2008; Berinde and Indyk 2009). Their reduction centers around finding a linear transformation with certain properties. Once such a transformation is specified, the algorithm proceeds (roughly) as follows: transform the input $\mathbf{x}$, run the compressed sensing scheme for ℓ_1, and "invert" the transformation to get $\mathbf{x}'$. Note that the number of measurements required is that of the ℓ_1 compressed sensing scheme.

One can try to use the Indyk–Price scheme for DP aggregation by viewing the hidden vector $\mathbf{x}$ as the sum $\sum_{i=1}^{n}\mathbf{p}_i$, and then adding Laplace noise to each measurement to ensure privacy. Although they did not analyze their guarantees for noisy measurements, one can follow the robustness of known ℓ_1 compressed sensing schemes to analyze the error. Unfortunately, since the error will scale according to the ℓ_1 norm of the noise vector and the noise vector consists of $O(k \cdot \log(n/k))$ entries, this approach only provides an error guarantee of $O(k \cdot (\mathrm{poly}\log n)/n)$.

To overcome this, we observe that, while compressed sensing and DP aggregation seem similar, they have different goals: the former aims to minimize the *number* of measurements whereas the latter aims to minimize the *error* due to the noise added (irrespective of the number of measurements). With this in mind, we proceed by using the Indyk–Price framework but *without compressing*, i.e., we simply measure the entire transformation. Even with this, the noise added to achieve DP is still too large and makes the error dependent on $\log n$. As a final step, to get rid of this factor we carefully select a different noise magnitude for each measurement, which allows us to finally achieve the $O(\sqrt{k})$ error as desired. The details are presented in Section 3.

Our lower bound follows the packing framework of Hardt and Talwar (2010). Specifically, we construct a set of k-sparse distributions whose pairwise EMDs are at least $\Omega(1/\sqrt{k})$. The construction is based on an ℓ_1 packing of the $\sqrt{k} \times \sqrt{k}$ grid, which gives a set of size $2^{\Omega(k)}$. It then immediately follows from Hardt and Talwar (2010) that the error must be at least $\Omega_\varepsilon(\sqrt{k}/n)$ with probability 0.9.

1.3 Related Work and Discussion

In a concurrent and independent work, Bagdasaryan et al. (2022) also study the private heatmaps problem. However, our work differs from theirs in three aspects: (i) they do not formulate the problem in terms of EMD, (ii) their work does not provide any formal utility guarantees unlike ours, (iii) their emphasis is on communication efficiency in distributed/federated setting whereas our focus is more general.

Our DP EMD sparse aggregation algorithm bears high-level similarity to known algorithms for DP hierarchical histograms (see, e.g., Cormode et al. 2012b; Qardaji, Yang, and Li 2013): all algorithms may be viewed as traversing the grid in a top-down manner, starting with larger subgrids and moving on to smaller ones, where a noise is added to the "measurement" corresponding to each subgrid. The differences between the algorithms are in the amount of noise

[3] A distribution is k-*sparse* if it is non-zero on at most k points.

[4] Note that the output $\hat{\mathbf{a}}$ need *not* be k-sparse. This is the reason why the approximation ratio λ can be less than one.

added to each step and how the noisy measurement is used to reconstruct the final output. Our choices of the noise amount and the Indyk–Price reconstruction algorithm are crucial to achieve the optimal EMD error bound stated in Theorem 1.1.

There are also DP hierarchical histogram algorithms that do not fit into the above outline, such as the PrivTree algorithm (Zhang, Xiao, and Xie 2016). An advantage of our approach is that the only aggregation primitive required is the Laplace mechanism; therefore, while we focus on the *central* model of DP (where the analyzer can see the raw input and only the output is required to be DP), our algorithm extends naturally to distributed models that can implement the Laplace mechanism, including the secure aggregation model and the shuffle model (Balle et al. 2020; Ghazi et al. 2020). On the other hand, algorithms such as PrivTree that use more complicated primitives cannot be easily implemented in these models.

2 Notation and Preliminaries

For $N \in \mathbb{N} \cup \{0\}$, we write $[N]$ to denote $\{0, \dots, N\}$. Let G_Δ be the set of $(\Delta \times \Delta)$ grid points in $[0,1)^2$; specifically, $G_\Delta = \{(i/\Delta, j/\Delta) \mid i, j \in [\Delta - 1]\}$. For notational convenience, we assume throughout that $\Delta = 2^\ell$ for some $\ell \in \mathbb{N}$.

For an index set $\mathcal{I}$, we view $\mathbf{p} \in \mathbb{R}^{\mathcal{I}}$ as a vector indexed by $\mathcal{I}$ and we write $\mathbf{p}(i)$ to denote the value of its ith coordinate; this notation extends naturally to the set $S \subseteq \mathcal{I}$ of coordinates, for which we let $\mathbf{p}(S) := \sum_{i \in S} \mathbf{p}(i)$. Furthermore, we use $\mathbf{p}|_S$ to denote the restriction of $\mathbf{p}$ to S; more formally, $\mathbf{p}|_S(i) = \mathbf{p}(i)$ if $i \in S$ and $\mathbf{p}|_S(i) = 0$ otherwise. We also write $\mathbf{p}|_{\bar{S}}$ as a shorthand for $\mathbf{p} - \mathbf{p}|_S$, i.e., the restriction of $\mathbf{p}$ to the complement of S. We use $\mathrm{supp}(\mathbf{p})$ to denote the set of non-zero coordinates of vector $\mathbf{p}$. A vector is said to be k-*sparse* if its support is of size at most k. Recall that the ℓ_1-norm of a vector $\mathbf{p} \in \mathbb{R}^{\mathcal{I}}$ is $\|\mathbf{p}\|_1 := \sum_{i \in \mathcal{I}} |p(i)|$.

2.1 Earth Mover's Distance (EMD)

Given two non-negative vectors $\mathbf{p}, \mathbf{q} \in \mathbb{R}^{G_\Delta}_{\geq 0}$ such that $\|\mathbf{p}\|_1 = \|\mathbf{q}\|_1$, their *Earth Mover's Distance* (EMD) is

$$\mathrm{EMD}(\mathbf{p}, \mathbf{q}) := \min_\gamma \sum_{x \in G_\Delta} \sum_{y \in G_\Delta} \gamma(x, y) \cdot \|x - y\|_1,$$

where the minimum is over $\gamma \in \mathbb{R}^{G_\Delta \times G_\Delta}_{\geq 0}$ whose marginals are $\mathbf{p}$ and $\mathbf{q}$. (I.e., for all $x \in G_\Delta$, $\sum_{y \in G_\Delta} \gamma(x, y) = \mathbf{p}(x)$ and, for all $y \in G_\Delta$, $\sum_{x \in G_\Delta} \gamma(x, y) = \mathbf{q}(y)$.)

We define the *EMD norm* of a vector $\mathbf{w} \in \mathbb{R}^{G_\Delta}$ by

$$\|\mathbf{w}\|_{\mathrm{EMD}} := \min_{\substack{\mathbf{p}, \mathbf{q} \in \mathbb{R}^{G_\Delta}_{\geq 0} \\ \mathbf{p} - \mathbf{q} + \mathbf{r} = \mathbf{w}, \|\mathbf{p}\|_1 = \|\mathbf{q}\|_1}} \mathrm{EMD}(\mathbf{p}, \mathbf{q}) + \alpha \cdot \|\mathbf{r}\|,$$

where $\alpha = 2$ is the diameter of our space $[0,1) \times [0,1)$.

The following simple lemma will be useful when dealing with unnormalized vs normalized vectors.

Lemma 2.1. *Suppose that $\mathbf{s}, \hat{\mathbf{s}} \in \mathbb{R}^{G_\Delta}_{\geq 0}$ are such that $\|\mathbf{s}\|_1 = n$ and $\|\mathbf{s} - \hat{\mathbf{s}}\|_{\mathrm{EMD}} \leq n/2$. Let $\mathbf{a} = \mathbf{s}/\|\mathbf{s}\|_1$ and $\hat{\mathbf{a}} = \hat{\mathbf{s}}/\|\hat{\mathbf{s}}\|_1$. Then, we have $\|\mathbf{a} - \hat{\mathbf{a}}\|_{\mathrm{EMD}} \leq 4\|\mathbf{s} - \hat{\mathbf{s}}\|_{\mathrm{EMD}}/n$.*

Proof. Let $\zeta = \|\mathbf{s} - \hat{\mathbf{s}}\|_{\mathrm{EMD}}$; observe that $|\|\mathbf{s}\|_1 - \|\hat{\mathbf{s}}\|_1| \geq \zeta$. As a result, we have $\|\hat{\mathbf{s}}\|_1 \in [n - \zeta, n + \zeta]$. Thus,

$$\|\hat{\mathbf{s}}/n - \hat{\mathbf{a}}\|_{\mathrm{EMD}} \leq \|\hat{\mathbf{s}}\|_{\mathrm{EMD}} \cdot \left| \frac{1}{n} - \frac{1}{\|\hat{\mathbf{s}}\|_1} \right|$$

$$\leq (n + \zeta) \cdot \left| \frac{1}{n} - \frac{1}{n - \zeta} \right| \leq \frac{3\zeta}{n}.$$

As a result, from the triangle inequality, we have

$$\|\mathbf{a} - \hat{\mathbf{a}}\|_{\mathrm{EMD}} \leq \|\mathbf{a} - \hat{\mathbf{s}}/n\|_{\mathrm{EMD}} + \|\hat{\mathbf{s}}/n - \hat{\mathbf{a}}\|_{\mathrm{EMD}}$$
$$\leq \frac{\zeta}{n} + \frac{3\zeta}{n} = \frac{4\zeta}{n}. \qquad \square$$

2.2 Differential Privacy

Two input datasets $\mathbf{X}, \mathbf{X}'$ are *neighbors* if $\mathbf{X}'$ results from adding or removing a single user's data from $\mathbf{X}$. In our setting, each user i's data is a distribution $\mathbf{p}_i$ over G_Δ.

Definition 2.1 (Differential Privacy; Dwork et al. (2006b))**.** A mechanism $\mathcal{M}$ is said to be ε-*DP* iff, for every set O of outputs and every pair $\mathbf{X}, \mathbf{X}'$ of neighboring datasets, $\Pr[\mathcal{M}(\mathbf{X}) \in O] \leq e^\varepsilon \cdot \Pr[\mathcal{M}(\mathbf{X}') \in O]$.

For a vector-valued function f, its ℓ_1-*sensitivity*, denoted by $S_1(f)$, is defined as $\max_{\text{neighbors } \mathbf{X}, \mathbf{X}'} \|f(\mathbf{X}) - f(\mathbf{X}')\|_1$.

Definition 2.2 (Laplace Mechanism)**.** The *Laplace mechanism* with parameter $b > 0$ adds an independent noise drawn from the Laplace distribution $\mathrm{Lap}(b)$ to each coordinate of a vector-valued function f.

Lemma 2.2 (Dwork et al. (2006b))**.** *The Laplace mechanism with parameter $S_1(f)/\varepsilon$ is ε-DP.*

2.3 Heatmaps

Given $\mathbf{p} \in \mathbb{R}^{G_\Delta}_{\geq 0}$, its associated *heatmap* with Gaussian filter variance σ^2 is defined as

$$H^\sigma_{\mathbf{p}}(x, y) = \sum_{(x', y') \in G_\Delta} \frac{e^{-\frac{(x - x')^2 + (y - y')^2}{2\sigma^2}}}{Z(x', y')} \cdot p(x', y')$$

for all $(x, y) \in G_\Delta$, where $Z(x', y') := \sum_{(x', y') \in G_\Delta} e^{-\frac{(x - x')^2 + (y - y')^2}{2\sigma^2}}$ is the normalization factor.

In the heatmap aggregation problem over n users, each user i has a probability distribution $\mathbf{p}_i$ over G_Δ. The goal is to output an estimate of the aggregated heatmap $H^\sigma_{\mathbf{a}}$ where $\mathbf{a} = \frac{1}{n} \sum_{i \in [n]} \mathbf{p}_i$.

3 Algorithm

In this section, we describe our private sparse EMD aggregation algorithm and prove our main result.

Theorem 3.1. *For any $\varepsilon > 0$ and $\lambda \in (0, 1)$, there is an ε-DP algorithm that can, w.p. 0.99, output a $\left(\lambda, O\left(\frac{\sqrt{k}}{\lambda \varepsilon n}\right)\right)$-approximation for the k-sparse EMD aggregation problem.*

3.1 Pyramidal Transform

As alluded to in Section 1, we use a linear transformation from (Indyk and Price 2011). This linear transformation is the so-called *(scaled) pyramidal transform*, whose variant is also often used in (metric) embedding of EMD to ℓ_1 (Charikar 2002; Indyk and Thaper 2003). Roughly speaking, the transform represents a hierarchical partitioning of $[0,1)^2$ into subgrids, where a subgrid at a level is divided into four equal subgrids at the next level. The (scaled) pyramidal transform has one row corresponding to each subgrid;

Algorithm 1: DPSPARSEEMDAGG

1: **Input:** distributions $\mathbf{p}_1, \dots, \mathbf{p}_n$ on G_Δ
2: **Parameters:** $\varepsilon_1, \dots, \varepsilon_\ell > 0, w \in \mathbb{N}$
3: $\mathbf{s} \leftarrow \sum_{i=1}^n \mathbf{p}_i$
4: **for** $i = 0, \dots, \ell$ **do**
5: $\quad \nu_i \leftarrow \mathrm{Lap}(1/\varepsilon_i)^{\otimes m_i}$
6: $\quad \mathbf{y}'_i \leftarrow \frac{1}{2^i}(\mathbf{P}_i \mathbf{s} + \nu_i)$
7: **end for**
8: $\mathbf{y}' \leftarrow [\mathbf{y}'_0 \cdots \mathbf{y}'_\ell]$
9: $\hat{\mathbf{s}} \leftarrow$ RECONSTRUCT$(\mathbf{y}'; w)$
10: **return** $\hat{\mathbf{a}} := \hat{\mathbf{s}}/\|\hat{\mathbf{s}}\|$

Algorithm 2: RECONSTRUCT

1: **Input:** noisy measurements $\mathbf{y}' \in \mathbb{R}^{\bigcup_{i\in[\ell]} C_{2^i}}$
2: **Parameters:** $w \in \mathbb{N}$
3: $S_0 \leftarrow C_1$
4: **for** $i = 1, \dots, \ell$ **do**
5: $\quad T_i \leftarrow \mathrm{children}(S_{i-1})$
6: $\quad S_i \leftarrow$ the set of $\min\{w, |T_i|\}$ coordinates in T_i with maximum values in $\mathbf{y}'$
7: **end for**
8: $S \leftarrow \bigcup_{i\in[\ell]} S_i$ and $\hat{\mathbf{y}} \leftarrow \mathbf{y}'|_S$
9: **return** $\hat{\mathbf{s}} \leftarrow \arg\min_{\mathbf{s}' \geq \mathbf{0}} \|\hat{\mathbf{y}} - \mathbf{P}\mathbf{s}'\|_1$

the row is equal to the indicator vector of the subgrid scaled by its side length. These are formalized below.

Definition 3.1. For $i \in \mathbb{N} \cup \{0\}$, we let C_{2^i} denote the set of *level i grid cells* defined as $C_{2^i} := \{[a, a+2^{-i}) \times [b, b+2^{-i}) \mid (a,b) \in G_{2^i}\}$; let $m_i := |C_{2^i}|$.

For $i \in [\ell]$, the *level-i grid partition map* is defined as the matrix $\mathbf{P}_i \in \{0,1\}^{C_{2^i} \times G_\Delta}$ where $\mathbf{P}_i(c,p) = 1$ iff $p \in c$. The *(scaled) pyramidal transform* is the matrix $\mathbf{P} \in \mathbb{R}^{\bigcup_{i=0}^{\ell} C_{2^i} \times G_\Delta}$ defined by $\mathbf{P} := \left[\mathbf{P}_0^\top \; 2^{-1}\mathbf{P}_1^\top \dots 2^{-\ell}\mathbf{P}_\ell^\top\right]^\top$.

3.2 The Algorithm

Our algorithm for sparse EMD aggregation consists of two components. The first component (Algorithm 1) aggregates the input distributions (Line 3) and applies the pyramidal transform to the aggregate, adding different amounts of Laplace noise for different levels of the grid (Lines 5, 6). (The parameters $\varepsilon_1, \dots, \varepsilon_\ell$, which govern the amount of Laplace noise, will be specified in the next subsection.) The second component (Algorithm 2) takes these noisy measurements for every level of the grid and reconstructs the solution by first recovering the ℓ_1 solution (Line 8) and then the EMD solution using a linear program (Line 9).

We stress that our algorithm is similar to that of Indyk and Price (2011) except for two points: first, we add noise to the measurements and, second, we are not doing any "compression" in contrast to (Indyk and Price 2011), which takes a wide matrix $\mathbf{A}$ for ℓ_1 recovery and multiplies it with $\mathbf{P}\mathbf{s}$.

3.3 Analysis

Following the framework of Indyk and Price (2011), our analysis proceeds in two stages. We first show that the "recovered" $\hat{\mathbf{y}}$ is close, in the ℓ_1 metric, to the true value of $\mathbf{P}\mathbf{s}$. Then, we use the properties of $\mathbf{P}$ to argue that the output $\hat{\mathbf{s}}$ is close, in EMD, to $\mathbf{s}$. Since we are adding noise to our measurement, we need to extend the work of Indyk and Price (2011) to be robust to noise. Finally, we set the privacy parameters $\varepsilon_1, \dots, \varepsilon_\ell$ to finish our proof of Theorem 3.1.

Let us now briefly demystify the additive error bound $O_{\varepsilon,\lambda}(\sqrt{k})$ that we end up with for $\hat{\mathbf{s}}$ (which ultimately gives the $O_{\varepsilon,\lambda}(\sqrt{k}/n)$ error bound for the normalized $\hat{\mathbf{a}}$). We will select $w = O_\lambda(k)$ so as to have an additive error of $O_\varepsilon(\sqrt{w})$. At a high level, each noise $\frac{1}{2^i} \cdot \nu_i(t)$ added to a "queried" term $\mathbf{y}_i(t)$ for $t \in T_i$ permeates to an error of the same order. For simplicity, assume for the moment that $|\nu_i(t)| = O(1/\varepsilon_i)$. Now, notice that if we are at level $i < \log\sqrt{w}$, then $|T_i| = |C_{2^i}| = 2^{2i}$ and thus the total error contribution of this level is $O(2^i/\varepsilon_i)$. On the other hand, for a level $i \geq \log\sqrt{w}$, we will have $|T_i| = w$ and the error contribution is $O\left(\frac{w}{2^i\varepsilon_i}\right)$. Now, when $i = \log\sqrt{w} \pm O(1)$, these error terms are $O(\sqrt{w}/\varepsilon_i)$ and thus we should set $\varepsilon_i = \Omega(1)$ to get the desired bound. However, in terms of $|i - \log\sqrt{w}|$, these error terms become exponentially smaller, i.e., $O\left(\frac{\sqrt{w}}{2^{|i-\log\sqrt{w}|}\varepsilon_i}\right)$. This leads to the natural choices of ε_i we use, which is to make it proportional to $\gamma^{|i-\log\sqrt{w}|}$ for some constant $\gamma > 0.5$. This indeed leads to the desired $O_\varepsilon(\sqrt{w}) = O_{\varepsilon,\lambda}(\sqrt{k})$ bound.

Phase I: ℓ_1 Recovery. We will now analyze the ℓ_1 recovery guarantee of $\hat{\mathbf{y}}$. Our recovery algorithm, which is an adaptation of Indyk and Price (2011), does *not* work for general hidden vectors. However, it works well for those that follow a certain "tree-like structure", formalized below.

Definition 3.2 (Indyk and Price (2011))**.** For $i \geq 1$, a grid cell $c' \in C_{2^i}$ is said to be a *child* of grid cell $c \in C_{2^{i-1}}$ if $c \subseteq c'$. This forms a tree rooted at $[0,1) \times [0,1) \in C_0$ where every internal node has exactly four children. We let $\mathcal{T}_w$ denote the set of all trees such that the number of nodes at each level is at most w.

Let $\mathcal{M}_w$ denote the set of $\mathbf{y} = [\mathbf{y}_0 \cdots \mathbf{y}_\ell]$ where $\mathbf{y}_i \in \mathbb{R}_{\geq 0}^{C_{2^i}}$ such that

1. $\mathrm{supp}(\mathbf{y}) \subseteq T$ for some tree $T \in \mathcal{T}_w$.
2. For all $i \in [\ell - 1], p \in C_{2^i}$, the following holds: $\mathbf{y}(p) \geq 2 \cdot \mathbf{y}(\mathrm{children}(p))$.

Under the above notion, we can adapt the ℓ_1 recovery analysis of Indyk and Price (2011) in the no-noise case to our regime, where the noise shows up as an error:

Lemma 3.2. *Let $\mathbf{y}^* \in \arg\min_{\mathbf{y}\in\mathcal{M}_w} \|\mathbf{P}\mathbf{s} - \mathbf{y}\|_1$ where $\mathrm{supp}(\mathbf{y}^*) \subseteq T^*$ for some $T^* \in \mathcal{T}_w$; let T_i^* denote $T^* \cap C_{2^i}$ and $V_i = T_i^* \setminus S_i$ for all $i \in [\ell]$. Then, $\hat{\mathbf{y}}$ on Line 8 of* RECONSTRUCT *satisfies $\|\hat{\mathbf{y}} - \mathbf{P}\mathbf{s}\|_1 \leq 3\|\mathbf{y}^* - \mathbf{P}\mathbf{s}\|_1 + O\left(\sum_{i\in[\ell]} \frac{1}{2^i} \|\nu_i|_{V_i \cup S_i}\|_1\right)$.*

Proof. For every $q \in T^* \setminus S$, let $R(q)$ be the highest ancestor

of q that does not belong to S. We have

$$\|\mathbf{y}^*|_{\bar{S}}\|_1 = \sum_{q\in T^*\setminus S} \mathbf{y}^*(q) = \sum_{i\in[\ell]}\sum_{p\in V_i}\sum_{q\in R^{-1}(p)} \mathbf{y}^*(q)$$
$$\overset{(\diamondsuit)}{\le} \sum_{i\in[\ell]}\sum_{p\in V_i} 2\mathbf{y}^*(p) = 2\sum_{i\in[\ell]} \mathbf{y}^*(V_i), \tag{1}$$

where $(\diamondsuit)$ follows from the second property of $\mathcal{M}_w$.

Next, consider the algorithm at the ith iteration and $p \in V_i$. Since p was not picked, the following must hold for all $q \in S_i \setminus T_i^*$: $\mathbf{y}'(p) \le \mathbf{y}'(q)$. Observe also that from $|S_i| = \max\{w, |C_{2^i}|\}$ and $|T_i^*| \le \max\{w, |C_{2^i}|\}$, we also have $|S_i \setminus T_i^*| \ge |T_i^* \setminus S_i| = |V_i|$. Thus, we get

$$\mathbf{y}'(V_i) \le \mathbf{y}'(S_i \setminus T_i^*). \tag{2}$$

From this and (1), we can further derive

$$\begin{aligned}
&\|\mathbf{y}^*|_{\bar{S}}\|_1 \\
&\overset{(1)}{\le} 2\left(\sum_{i\in[\ell]}(\mathbf{y}^*(V_i) - \mathbf{Ps}(V_i)) + \mathbf{Ps}(S_i \setminus T_i^*)\right) \\
&\qquad + \left(\sum_{i\in[\ell]} \mathbf{Ps}(V_i) - \mathbf{Ps}((S_i \setminus T_i^*))\right) \\
&\overset{(\square)}{\le} 2\|\mathbf{y}^* - \mathbf{Ps}\|_1 + 2\left(\sum_{i\in[\ell]} \mathbf{Ps}(V_i) - \mathbf{Ps}((S_i \setminus T_i^*))\right) \\
&\overset{(\triangle)}{\le} 2\|\mathbf{y}^* - \mathbf{Ps}\|_1 + 2\left(\sum_{i\in[\ell]} \frac{1}{2^i}\left\|\nu_i|_{V_i\cup(S_i\setminus T_i^*)}\right\|_1\right) \\
&\qquad + 2\left(\sum_{i\in[\ell]} \mathbf{y}'(V_i) - \mathbf{y}'((S_i \setminus T_i^*))\right) \\
&\overset{(2)}{\le} 2\|\mathbf{y}^* - \mathbf{Ps}\|_1 + 2\left(\sum_{i\in[\ell]} \frac{1}{2^i}\left\|\nu_i|_{V_i\cup(S_i\setminus T_i^*)}\right\|_1\right),
\end{aligned} \tag{3}$$

where $(\square)$ follows from $\mathrm{supp}(\mathbf{y}^*) \subseteq T^* = \bigcup_{i\in[\ell]} T_i^*$ and $(\triangle)$ follows from how $\mathbf{y}'$ is calculated. Finally, from $\hat{\mathbf{y}} = \mathbf{y}'|_S$ and how each entry of $\mathbf{y}'$ is computed, we have

$$\begin{aligned}
&\|\hat{\mathbf{y}} - \mathbf{Ps}\|_1 = \|\mathbf{y}'|_S - \mathbf{Ps}|_S\|_1 + \|\mathbf{Ps}_{\bar{S}}\|_1 \\
&\le \left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{S_i}\|_1\right) + \|\mathbf{y}^*|_{\bar{S}}\|_1 + \|\mathbf{y}^*|_{\bar{S}} - \mathbf{Ps}|_{\bar{S}}\|_1 \\
&\overset{(3)}{\le} \left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{S_i}\|_1\right) \\
&\qquad + \left(2\|\mathbf{y}^* - \mathbf{Ps}\|_1 + 2\left(\sum_{i\in[\ell]} \frac{1}{2^i}\left\|\nu_i|_{V_i\cup(S_i\setminus T_i^*)}\right\|_1\right)\right) \\
&\qquad + \|\mathbf{y}^* - \mathbf{Ps}\|_1 \\
&\le 3\|\mathbf{y}^* - \mathbf{Ps}\|_1 + 3\left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{V_i\cup S_i}\|_1\right).
\end{aligned}$$

□

Phase II: From ℓ_1 to EMD. We now proceed to bound the EMD error. The main lemma is stated below.

Lemma 3.3. *Let the notation be as in Lemma 3.2. For any $\eta' \in (0,1)$, by setting $w = O(k/(\eta')^2)$, the output $\hat{\mathbf{s}}$ of* RECONSTRUCT *satisfies $\|\mathbf{s} - \mathbf{s}^*\|_{\mathrm{EMD}} \le \eta' \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s} - \mathbf{s}'\|_{\mathrm{EMD}} + O\left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{V_i\cup S_i}\|_1\right)$.*

Similar to the proof of Indyk and Price (2011), our proof of Lemma 3.3 converts the recovery guarantee under ℓ_1 metric to that under EMD; to do this, we need the following two statements from prior work.

Lemma 3.4 (Model-Alignment of EMD with $\mathcal{M}_w$ (Indyk and Price 2011))**.** *For any $\mathbf{x} \in \mathbb{R}^{G_\Delta}_{\ge 0}, k \in \mathbb{N}$ and $\eta \in (0,1)$, there exist $w = O(k/\eta^2)$ and $\mathbf{y}^* \in \mathcal{M}_w$ such that $\|\mathbf{y}^* - \mathbf{Ps}\|_1 \le \eta \cdot \min_{k\text{-sparse } \mathbf{x}'} \|\mathbf{x} - \mathbf{x}'\|_{\mathrm{EMD}}$.*

Lemma 3.5 (EMD-to-ℓ_1 Expansion (Indyk and Thaper 2003))**.** *For all $\mathbf{z} \in \mathbb{R}^{G_\Delta}$, $\|\mathbf{z}\|_{\mathrm{EMD}} \le \|\mathbf{Pz}\|_1$.*

Proof of Lemma 3.3. Recall that we use $\mathbf{s}$ to denote the true sum $\sum_{i=1}^n \mathbf{p}_i$. We set $\eta = \eta'/6$ and let $w = O(k/\eta^2) = O(k/(\eta')^2)$ be as in Lemma 3.4, which ensures that there exists $\mathbf{y}^* \in \mathcal{M}_w$ with

$$\|\mathbf{y}^* - \mathbf{Ps}\|_1 \le \eta \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s} - \mathbf{s}'\|_{\mathrm{EMD}}. \tag{4}$$

Thus, using Lemma 3.5, we can derive

$$\begin{aligned}
\|\mathbf{s} - \mathbf{s}^*\|_{\mathrm{EMD}} &\le \|\mathbf{P}(\mathbf{s} - \mathbf{s}^*)\|_1 \\
\text{(triangle inequality)} &\le \|\hat{\mathbf{y}} - \mathbf{Ps}\|_1 + \|\hat{\mathbf{y}} - \mathbf{Ps}^*\|_1 \\
\text{(how } \mathbf{s}^* \text{ is computed)} &\le 2\|\hat{\mathbf{y}} - \mathbf{Ps}\|_1 \\
\text{(Lemma 3.2)} &\le 6\|\mathbf{y}^* - \mathbf{Ps}\|_1 \\
&\qquad + O\left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{V_i\cup S_i}\|_1\right) \\
&\overset{(4)}{\le} \eta' \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s} - \mathbf{s}'\|_{\mathrm{EMD}} \\
&\qquad + O\left(\sum_{i\in[\ell]} \frac{1}{2^i}\|\nu_i|_{V_i\cup S_i}\|_1\right).
\end{aligned}$$

□

Finishing the Proof. We now select the privacy parameters and complete the proof of Theorem 3.1.

Proof of Theorem 3.1. Let $w = O(k/(\eta')^2)$ be as in Lemma 3.3 with $\eta' = \lambda/4$, and let $q = \lfloor \log_2 \sqrt{w} \rfloor$. Let $\gamma = 0.8$ be the "decay rate" for ε_i's, and let $Z = \sum_{i=0}^{\ell} \gamma^{|i-q|} \le O(1)$ be the normalization factor. We run Algorithm 1 with $\varepsilon_i = \gamma^{|i-q|} \cdot \varepsilon/Z$.

Privacy Analysis. We can view the ith iteration of the algorithm as releasing $2^i\mathbf{y}'_i = \mathbf{P}_i\mathbf{s} + \nu_i$. Since each $\mathbf{p}_i$ has ℓ_1 norm at most one, its sensitivity with respect to $\mathbf{P}_i\mathbf{s}$ is at most one; thus, Lemma 2.2 implies that the ith iteration is ε_i-DP. As a result, by basic composition theorem of DP, we can conclude that releasing all of $\mathbf{y}'_0, \ldots, \mathbf{y}'_\ell$ is $(\varepsilon_0 + \cdots + \varepsilon_\ell)$-DP. Since the reconstruction is simply a post-processing step, the post-processing property of DP ensures that Algorithm 1 is $(\varepsilon_0 + \cdots + \varepsilon_\ell)$-DP. Finally, observe that by definition of ε_i's, we have $\varepsilon_0 + \cdots + \varepsilon_\ell = \varepsilon$ as desired.

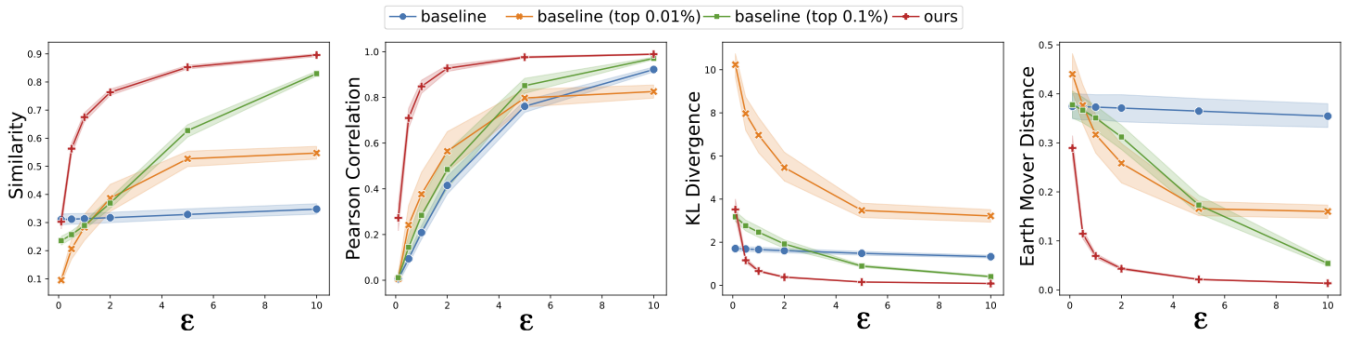

Figure 1: Metrics averaged over 60 runs when varying ε. Shaded areas indicate 95% confidence interval.

Utility Analysis. Applying Lemma 3.3, we can conclude that $\|\mathbf{s}-\mathbf{s}^*\|_{\text{EMD}} \le \eta' \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s}-\mathbf{s}'\|_{\text{EMD}} + \xi$, where $\xi = O\left(\sum_{i\in[\ell]} \frac{1}{2^i} \|\nu_i|_{V_i \cup S_i}\|_1\right)$. Recall that each of V_i, S_i's is of size at most $\max\{w, 2^{2i}\}$ (because of definition of $\mathcal{M}_w$ and the fact that $m_i = |C_{2^i}| = 2^{2i}$), and that each entry of ν_i is sampled from $\text{Lap}(1/\varepsilon_i)$. As a result, we have

$$\begin{aligned}\mathbb{E}[\xi] &\le O\left(\sum_{i\in[\ell]} \frac{1}{2^i} \cdot \max\{w, 2^{2i}\} \cdot \frac{1}{\varepsilon_i}\right) \\ &= O\left(\sum_{i\in[q]} \frac{2^i}{\gamma^{q-i}\varepsilon}\right) + O\left(\sum_{i\in\{q+1,\dots,\ell\}} \frac{k}{\lambda^2} \cdot \frac{1}{2^i \gamma^{i-q}\varepsilon}\right) \\ &= O\left(\frac{2^q}{\varepsilon}\right) + O\left(\frac{k}{\lambda^2} \cdot \frac{1}{2^q} \cdot \frac{1}{\varepsilon}\right) = O\left(\frac{\sqrt{k}}{\lambda\varepsilon}\right),\end{aligned}$$

where the last bound follows from our choice of $\gamma > 0.5$ and $q = \lfloor \log_2 \sqrt{w} \rfloor$. Hence, by Markov's inequality, w.p. 0.99, we have $\|\mathbf{s} - \mathbf{s}^*\|_{\text{EMD}} \le \eta' \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s} - \mathbf{s}'\|_{\text{EMD}} + 100\mathbb{E}[\xi] = \eta' \cdot \min_{k\text{-sparse } \mathbf{s}'} \|\mathbf{s}-\mathbf{s}'\|_{\text{EMD}} + O\left(\frac{\sqrt{k}}{\lambda\varepsilon}\right)$. Finally, applying Lemma 2.1 concludes the proof. □

4 Experiments

In this section, we study the performance of our algorithms on real-world datasets.

Implementation Details. We implement Algorithm 1 with a minor modification: we do not measure at the level $i < q = \lfloor \log \sqrt{w} \rfloor$. In other words, we start directly at the lowest level for which the number of grid cells is at most $\sqrt{w}$. It is possible to adjust the proof to show that, even with this modification, the error remains $O_\varepsilon(\sqrt{k})$. Apart from this, the algorithm is exactly the same as presented earlier. We note that the linear program at the end of Algorithm 2 can be formulated so that the number of variables is only $O(w\ell)$; the reason is that we only need one variable per cell that is left out at each stage. This allows us to solve it efficiently even when the resolution $\Delta = 2^\ell$ is large.

As for our parameters, we use the decay rate $\gamma = 1/\sqrt{2}$, which is obtained from minimizing the second error term in the proof of Theorem 3.1 as $\ell \to \infty$[5]. We use $w = 20$ in our experiments, which turns out to work well already for datasets we consider. We refrain from tuning parameters further since a privacy analysis of the tuning step has to be taken into account if we want to be completely rigorous. (See, e.g., (Liu and Talwar 2019) for a formal treatment.)

Datasets. We use two datasets available at snap.stanford.edu to generate the input distribution for users. The first dataset[6], called GOWALLA, consists of location check-ins by users of the location-based social network Gowalla. Each record consists of, among other things, an anonymized user id together with the latitude (lat) and longitude (lon) of the check-in and a timestamp. We filtered this dataset to consider only check-ins roughly in the continental US (i.e., lon $\in (-135, -60)$ and lat $\in (0, 50)$) for the month of January 2010; this resulted in 196,071 check-ins corresponding to 10,196 users. The second dataset[7], called BRIGHTKITE, also contains check-ins from a different and now defunct location-based social network Brightkite; each record is similar to GOWALLA. Once again, we filtered this dataset to consider only check-ins in the continental US for the months of November and December 2008; this resulted in 304,608 check-ins corresponding to 10,177 users.

For each of these datasets, we partition the whole area into a 300×300 grid. We then took the top 30 cells (in both datasets combined) that have the most check-ins. (Each of the 30 cells is mostly around some city like New York, Austin, etc, and has check-ins from at least 200 unique users). We then consider each cell, partition into $\Delta \times \Delta$ subgrids and snap each check-in to one of these subgrids.

Metrics. To evaluate the quality of an output heatmap $\hat{\mathbf{h}}$ compared to the true heatmap $\mathbf{h}$, we use the following commonly used metrics: Similarity, Pearson coefficient, KL-divergence, and EMD. (See, e.g., (Bylinskii et al. 2019) for detailed discussions of these metrics.) We note that the first two metrics should *increase* as $\hat{\mathbf{h}}, \mathbf{h}$ are more similar,

[5] When w (and thus q) is fixed, the second error term is proportional to $Z \cdot \sum_{i=0}^{\ell-q-1} \frac{1}{(2\gamma)^i}$ which converges to $\frac{1}{(1-\gamma)(1-0.5/\gamma)}$ as $\ell \to \infty$. The latter term is minimized when $\gamma = 1/\sqrt{2}$

[6] Available at http://snap.stanford.edu/data/loc-Gowalla.html

[7] Available at http://snap.stanford.edu/data/loc-Brightkite.html

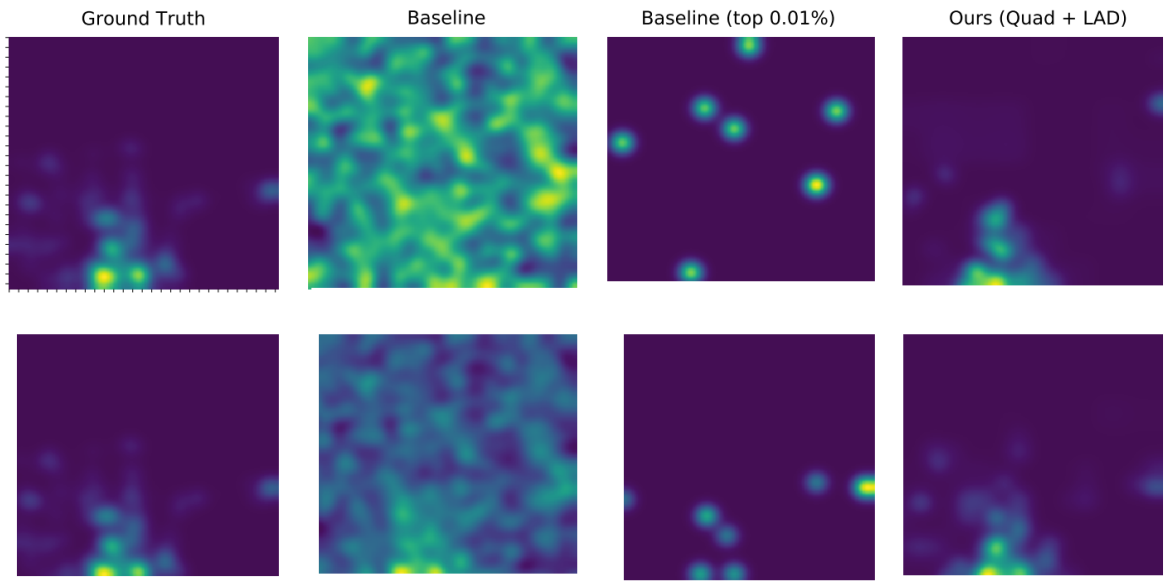

Figure 2: Example visualization of different algorithms for $\varepsilon = 1$ (top) and $\varepsilon = 5$ (bottom). The algorithms from left to right are: original heatmap (no privacy), baseline, baseline with top 0.01% and our algorithm.

whereas the latter two should *decrease*.

Baselines. We consider as a baseline an algorithm recently proposed in (Liu et al. 2019),[8] where we simply add Laplace noise to each subgrid cell of the sum $\mathbf{s}$, zero out any negative cells, and produce the heatmap from this noisy aggregate. We also consider a "thresholding" variant of this baseline that is more suited to sparse data: only keep top $t\%$ of the cell values after noising (and zero out the rest).

4.1 Results

In the first set of experiments, we fix $\Delta = 256$. For each $\varepsilon \in \{0.1, 0.5, 1, 2, 5, 10\}$, we run our algorithms together with the baseline and its variants on all 30 cells, with 2 trials for each cell. In each trial, we sample a set of 200 users and run all the algorithms; we then compute the distance metrics between the true heatmap and the estimated heatmap. The average of these metrics over the 60 runs is presented in Figure 1, together with the 95% confidence interval. As can be seen in the figure, the baseline has rather poor performance across all metrics, even for large $\varepsilon = 10$. We experiment with several values of t for the thresholding variant, which yields a significant improvement. Despite this, we still observe an advantage of our algorithm consistently across all metrics. These improvements are especially significant when ε is not too large or too small (i.e., $0.2 \leq \varepsilon \leq 5$).

In the second set of experiments, we study the effect of varying the number n of users. By fixing a single cell (with > 500 users) and ε, we sweep $n \in \{50, 100, 200, 300, 400, 500\}$ users. For each value of n, we run 10 trials and average their results. As predicted by theory, our algorithms and the original baseline perform better as n increases. However, the behavior of the thresholding variants of the baseline are less predictable, and sometimes the performance degrades with a larger number of users. It seems plausible that a larger number of users cause an increase in the sparsity, which after some point makes the simple thresholding approach unsuited for the data.

We also run another set of experiments where we fix a single cell and ε, and vary the resolution $\Delta \in \{64, 128, 256\}$. In agreement with theory, our algorithm's utility remains nearly constant for the entire range of Δ. On the other hand, the original baseline suffers across all metrics as Δ increases. The thresholding variants are more subtle; they occasionally improve as Δ increases, which might be attributed to the fact that when Δ is small, thresholding can zero out too many subgrid cells.

We include examples of the heatmaps from each approach in Figure 2.

5 Discussions and Future Directions

We present an algorithm for sparse distribution aggregation under the EMD metric, which in turn yields an algorithm for producing heatmaps. As discussed earlier, our algorithm extends naturally to distributed models that can implement the Laplace mechanism, including the secure aggregation model and the shuffle model (Balle et al. 2020; Ghazi et al. 2020). Unfortunately, this does not apply to the more stringent *local* DP model (Kasiviswanathan et al. 2008) and it remains an interesting open question to devise practical local DP heatmap/EMD aggregation algorithms for "moderate" number of users and privacy parameters.

[8] We remark that (Liu et al. 2019) also propose using the Gaussian mechanism. However, this algorithm does *not* satisfy ε-DP. Moreover, even when considering (ε, δ)-DP for moderate value of δ (e.g., $\delta = 10^{-3}$), the Gaussian mechanism will still add more noise in expectation than the Laplace mechanism.

References

Abowd, J. M. 2018. The US Census Bureau adopts differential privacy. In *KDD*, 2867–2867.

Apple Differential Privacy Team. 2017. Learning with privacy at scale. *Apple Machine Learning Journal*.

Backurs, A.; Indyk, P.; Razenshteyn, I. P.; and Woodruff, D. P. 2016. Nearly-optimal bounds for sparse recovery in generic norms, with applications to *k*-median sketching. In *SODA*, 318–337.

Bagdasaryan, E.; Kairouz, P.; Mellem, S.; Gascón, A.; Bonawitz, K.; Estrin, D.; and Gruteser, M. 2022. Towards Sparse Federated Analytics: Location Heatmaps under Distributed Differential Privacy with Secure Aggregation. *PoPETS*, 4: 162–182.

Balle, B.; Bell, J.; Gascón, A.; and Nissim, K. 2020. Private Summation in the Multi-Message Shuffle Model. In *CCS*, 657–676.

Berinde, R.; Gilbert, A. C.; Indyk, P.; Karloff, H.; and Strauss, M. J. 2008. Combining geometry and combinatorics: A unified approach to sparse signal recovery. In *Allerton*, 798–805.

Berinde, R.; and Indyk, P. 2009. Sequential sparse matching pursuit. In *Allerton*, 36–43.

Berinde, R.; Indyk, P.; and Ruzic, M. 2008. Practical near-optimal sparse recovery in the ℓ_1 norm. In *Allerton*, 198–205.

Bylinskii, Z.; Judd, T.; Oliva, A.; Torralba, A.; and Durand, F. 2019. What Do Different Evaluation Metrics Tell Us About Saliency Models? *PAMI*, 41(3): 740–757.

Charikar, M. 2002. Similarity estimation techniques from rounding algorithms. In *STOC*, 380–388.

Cormode, G.; Procopic, C.; Srivastava, D.; and Tran, T. T. L. 2012a. Differentially private summaries for sparse data. In *ICDT*, 299–311.

Cormode, G.; Procopiuc, C. M.; Srivastava, D.; Shen, E.; and Yu, T. 2012b. Differentially Private Spatial Decompositions. In *ICDE*, 20–31.

Ding, B.; Kulkarni, J.; and Yekhanin, S. 2017. Collecting telemetry data privately. In *NeurIPS*, 3571–3580.

Dwork, C.; Kenthapadi, K.; McSherry, F.; Mironov, I.; and Naor, M. 2006a. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, 486–503.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006b. Calibrating noise to sensitivity in private data analysis. In *TCC*, 265–284.

Dwork, C.; Smith, A. D.; Steinke, T.; Ullman, J. R.; and Vadhan, S. P. 2015. Robust Traceability from Trace Amounts. In Guruswami, V., ed., *FOCS*, 650–669.

Erlingsson, Ú.; Pihur, V.; and Korolova, A. 2014. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *CCS*, 1054–1067.

Ghazi, B.; Manurangsi, P.; Pagh, R.; and Velingker, A. 2020. Private Aggregation from Fewer Anonymous Messages. In *EUROCRYPT*, 798–827.

Grauman, K.; and Darrell, T. 2004. Fast contour matching using approximate Earth Mover's Distance. In *CVPR*, 220–227.

Greenberg, A. 2016. Apple's "differential privacy" is about collecting your data – but not your data. *Wired, June*, 13.

Hardt, M.; and Talwar, K. 2010. On the geometry of differential privacy. In *STOC*, 705–714.

Indyk, P.; and Price, E. 2011. K-median clustering, model-based compressive sensing, and sparse recovery for earth mover distance. In *STOC*, 627–636.

Indyk, P.; and Ruzic, M. 2008. Near-Optimal Sparse Recovery in the L1 Norm. In *FOCS*, 199–207.

Indyk, P.; and Thaper, N. 2003. Fast color image retrieval via embeddings. In *Workshop on Statistical and Computational Theories of Vision (at ICCV), 2003*.

Isaacman, S.; Becker, R.; Cáceres, R.; Martonosi, M.; Rowland, J.; Varshavsky, A.; and Willinger, W. 2012. Human mobility modeling at metropolitan scales. In *MobiSys*, 239–252.

Kasiviswanathan, S. P.; Lee, H. K.; Nissim, K.; Rashkodnikova, S.; and Smith, A. 2008. What can we Learn Privately? In *FOCS*, 531–540.

Kranstauber, B.; Smolla, M.; and Safi, K. 2017. Similarity in spatial utilization distributions measured by the earth mover's distance. *Methods in Ecology and Evolution*, 8(2): 155–160.

Levina, E.; and Bickel, P. 2001. The earth mover's distance is the Mallows distance: Some insights from statistics. In *ICCV*, 251–256.

Liu, A.; Xia, L.; Duchowski, A.; Bailey, R.; Holmqvist, K.; and Jain, E. 2019. Differential privacy for eye-tracking data. In *ETRA*, 1–10.

Liu, J.; and Talwar, K. 2019. Private selection from private candidates. In Charikar, M.; and Cohen, E., eds., *STOC*, 298–309.

Puzicha, J.; Rubner, Y.; Tomasi, C.; and Buhmann, J. M. 1999. Empirical evaluation of dissimilarity measures for color and texture. In *ICCV*, 1165–1173.

Qardaji, W. H.; Yang, W.; and Li, N. 2013. Understanding Hierarchical Methods for Differentially Private Histograms. *VLDB*, 6(14): 1954–1965.

Radebaugh, C.; and Erlingsson, U. 2019. Introducing TensorFlow Privacy: Learning with Differential Privacy for Training Data. https://blog.tensorflow.org/2019/03/introducing-tensorflow-privacy-learning.html. Accessed: 2023-03-05.

Rubner, Y.; Tomasi, C.; and Guibas, L. 1998. A metric for distributions with applications to image databases. In *ICCV*, 59–66.

Rubner, Y.; Tomasi, C.; and Guibas, L. J. 2000. The earth mover's distance as a metric for image retrieval. *IJCV*, 40(2): 99–121.

Shankland, S. 2014. How Google tricks itself to protect Chrome user privacy. *CNET, October*.

Steil, J.; Hagestedt, I.; Huang, M. X.; and Bulling, A. 2019. Privacy-aware eye tracking using differential privacy. In *ETRA*, 1–9.

Stricker, M.; and Orengo, M. 1995. Similarity of color images. In *SPIE*, 381–392.

Testuggine, D.; and Mironov, I. 2020. PyTorch Differential Privacy Series Part 1: DP-SGD Algorithm Explained. https://medium.com/pytorch/differential-privacy-series-part-1-dp-sgd-algorithm-explained-12512c3959a3. Accessed: 2023-03-05.

Wang, F.; and Guibas, L. J. 2012. Supervised earth mover's distance learning and its computer vision applications. In *ECCV*, 442–455.

Xu, D.; Yan, S.; and Luo, J. 2008. Face recognition using spatially constrained Earth Mover's Distance. *Trans. Image Processing*, 17(11): 2256–2260.

Zhang, J.; Xiao, X.; and Xie, X. 2016. PrivTree: A Differentially Private Algorithm for Hierarchical Decompositions. In *SIGMOD*, 155–170.

Zhao, Q.; Yang, Z.; and Tao, H. 2008. Differential Earth Mover's Distance with its applications to visual tracking. *PAMI*, 32: 274–287.