Dit is de titel van mijn afstudeerverslag

by

Tjibbe van der Ende

in partial fulfillment of the requirements for the degree of

Master of Science
in Software Engineering

at the Open University, faculty of Management, Science and Technology
Master Software Engineering
to be defended publicly on Day Month DD, YYYY at HH:00 PM.

Student number:	student number	
Course code:	IMA0002	
Thesis committee:	titles and name of the chairman (chairman),	Open University
	titles and name of the supervisor (supervisor),	Open University

Open Universiteit
www.ou.nl
de beste!

Contents

1
INTRODUCTION

1.1. RESEARCH QUESTIONS

Main question:
How can the nD-Laplace algorithm be applied in training privacy-preserving clustering algorithms on distributed n-dimensional data?

1. RQ1: How can 2D-Laplace be used to protect the data privacy of 2-dimensional data which is employed for training clustering algorithms?
2. RQ2: How can 3D-Laplace be extended to protect the data privacy of n-dimensional data which is employed for training clustering algorithms?
3. RQ3: What is the impact of different privacy budgets, dataset properties, and other clustering algorithms on the research conducted for research question 2?

2

LITERATURE REVIEW

This chapter lays out the theoretical foundation of this work. To review the past literature, it is first necessary to gather the required knowledge for it.

2.1. Differential privacy

2.1.1. Laplace algorithm

2.2. CLUSTERING

2.2.1. METHODS

2.2.2. EVALUATION METHODS

2.3. Literature review

3

nD-Laplace

3.1. 2D-Laplace

The theory for this subject is heavily inspired by the paper that was written by Andrés et al. [Andrés et al., 2012]. This notion of geo-indistinguishability was introduced to solve the issue of privacy and location data. It offers an alternative approach for differential privacy by adding noise to the location locally before sending it to a location-based system (LBS) like Google maps. This section starts with an introduction to mathematics for the planar and polar Laplace algorithm. For each of the different subsections, we visualize and explain open challenges and theoretic for applying them for clustering.

Math symbols

X Set of locations for a user. (R^2).

Z For every $x \in X$ a perturbed location $z \in Z$ is reported..

ϵ Defined as $\epsilon = l/r$.

θ Angle.

l Privacy level.

r Radius.

3.1.1. Planar and polar Laplace

The idea of planar Laplace is to generate an area around $x0 \in X$ according to the multivariate Laplace distribution. The mechanism of planar Laplace is a modification of the Laplace algorithm to support distance [Andrés et al., 2012]. This distance method $dist(x, x')$ is defined as the Euclidean distance between two points or sets. Recalling the definition of Laplace, this method $|x - x'|$ is replaced by the distance metric. Hence, the definition of the Probability Density Function (pdf) by Andrés et al. is:

$$\frac{\epsilon^2}{2 * \pi} e(-\epsilon d(x_0, x)) \tag{3.1}$$

Which is the likelihood a generated point $z \in Z$ is close to $x0$. The method works for Cartesian coordinates but was modified to support polar coordinates by including θ. So each point is reflected as (r, θ) and can be modified by using a slight modification to work for polar Laplace.

A point $z \in Z$ where $z = (r, \theta)$ is randomly generated using two separate methods.

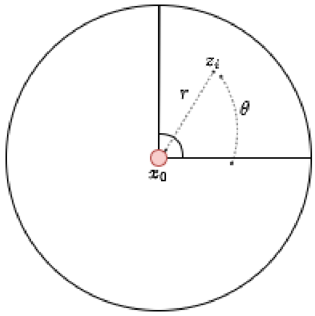

Figure 3.1: Representation of the generated $z = r\theta$ and original point $x0$.

Calculating r: This variable is described as $dist(x_0, z)$ and can be randomly drawn by inverting the CDF (**Link**) for the Laplace distribution:

$$C_\epsilon{}^{-1}(p) = -\frac{1}{\epsilon}(W_{-}1(\frac{p-1}{e}) + 1) \tag{3.2}$$

For this equation, $W_{-}1$ is a Lambert W function with -1 branch. The Lambert w function, also called the product logarithm is defined as $W(x)e^{W(x)} = x$ [Lehtonen, 2016]. The purpose of the Lambert w function is to invert the CDF of the Laplace distribution to generate random noise for one of the coordinates (r) using the random value of p.

Calculating θ: The other coordinate (θ) is defined as a random number $[0, 2\pi]$.

To visualize these methods it is necessary to convert the polar coordinates for $z = (r, \theta)$ back to a plane (x, y). This is described as step 4 of the planar Laplace algorithm [Andrés et al., 2012] and visualized using figure 1.

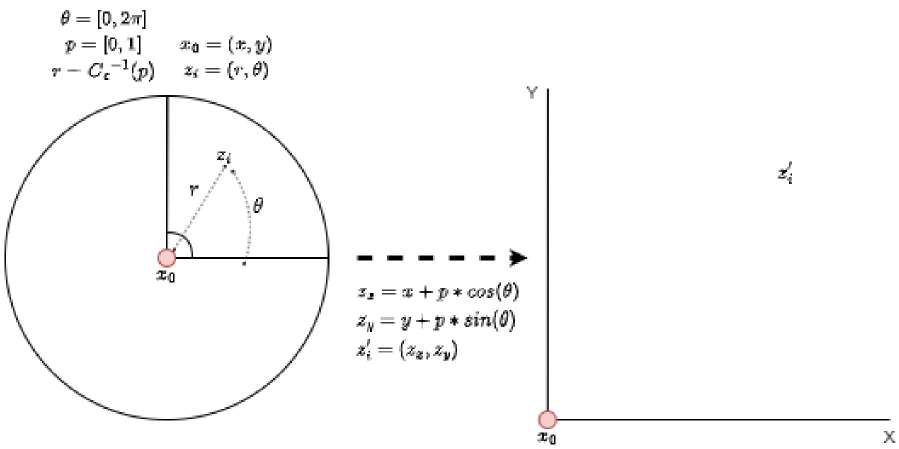

Figure 3.2: Representation of converting the perturbed point $z = (r, \theta)$ to a point z_x, z_y

We highlight the CDF function for assessing if the probability of a random point falls between 0 and r:

Can be moved to appendix

$$C\epsilon(r) = D_\epsilon, R(p)dp = \int_0^r 1 - (1 + \epsilon r)e^{-\epsilon r} \tag{3.3}$$

3.1.2. Truncation

Because we have a finite space, it can be possible the perturbed points are off-graph (outside the given domain). The solution was described in step 5 of the Laplacian mechanism for 2D space. This explains the idea of remapping to the closest admissible location in set A. For which $A \subset \mathbb{R}$, where A is the set of admissible locations [Andrés et al., 2012]. This is also described by chatzikokolakis et al, who also describes a method to do it. When a perturbed point z is located at the sea or in water, it is easily distinguishable as a fake location. They introduce a method to check this and efficiently remap to a nearby location.

Describe the method

Analyze other methods

3.1.3. Optimizing for clustering

The decision of the parameters for the algorithm is straightforward as it depends on the ϵ. This constant is calculated by defining the radius r and the desired level of privacy l and ϵ is calculated using l/r. The l is a predefined constant $l \in R^+$ but usually will be below 10. For geographical data, the r is straightforward and can be configured by using meters as a unit of measure. Therefore, $r = 200$ corresponds to a radius of 200m around point x_0. So, regarding clustering, it is a challenge to define a reasonable radius.
The ϵ can be considered the inverse unit of r [Andrés et al., 2012]. A radius can be defined per-use case based on how crowded a place is [Chatzikokolakis et al., 2015].

Give the algorithm

A drawn area as shown in **??** can be expressed as a perturbation area P_{area} [Yan et al., 2022]. This metric was formulated as:

$$P_{area} = \left\{ center = x_0, radius = \frac{1}{N} \times \sum_{i=1}^{N} r_i \right\} \tag{3.4}$$

The method loops through each perturbed point r on center x_0 (recall **??**) and calculates the Euclidean distance for an n amount of perturbation points. Although the method does not contribute to the Laplace algorithm, it is useful for visualization purposes.

3.2. 3D-Laplace

Is considered for research question 3

4
METHODOLOGY

To gain insights into the proposed methods for researching the appliance of (ND)-Laplace for cluster algorithms we conducted experiments. The experiment results are used to evaluate our method against other literature. In this chapter we explain:

1. Datasets
2. Environmental setup.
3. For each research question: Description of the different experiments.
4. For each research question: Results.

4.1. DATASETS

For this research, we will use a synthetic dataset for all three research questions.

Records	Centers	Dimensions	Standard deviation	Research
200	4	2	0.60	RQ 1
200	4	3	0.60	RQ 2
200	4	5	0.60	RQ 2

Research question 3 uses a "real-world" dataset to properly assess the different dataset properties that are the subject of this research question.

Describe datasets (RQ3)

4.2. ENVIRONMENTAL SETUP

Describe operation system / specs

Describe (python libraries)

4.3. METHODS

This section explains what methods/ algorithms we used and

4.3.1. EVALUATION

Describe evaluation methodology

4.3.2. RESEARCH QUESTION 1

We propose several solutions for open issues based on the theoretical framework.

CHOOSING R:

Based, on the idea of chatzikokolakis et al. to lower the size of the radius if the place is crowded, we can do the same with clustering. For this, we could use a metric like the standard division. This metric does exactly this, by providing the deviation from the mean:

This metric increases based on clutteredness of the data, which allows us to generate a radius r automatically regardless of domain. Therefore, we depend on the configurability of epsilon entirely on privacy level l. The generic standard deviation can be defined as:

$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{n}} \tag{4.1}$$

The σ being our diameter d, the radius r is then calculated as $\frac{d}{2}$.

TRUNCATION:

We explained the theory for truncation earlier in paragraph 3.1.2. The methods proposed work correctly for a geographic map where other (historic) locations for remapping are available.

However, it is difficult to apply this to data clustering. The number of data points is not known beforehand, so we may remap to a location that is too far away. This way we lose important clusters, which hurts the clustering. Also, the truncation threshold is so clear (the points are outside the known 2D domain), that we do not have to rely on historical data for remapping. Our algorithm can be much simpler by re-calculating the noise until it will be within the domain: This algorithm uses x_{min} and x_{max} to re-calculate the points

Algorithm 1 Truncation algorithm ($T(\min, \max, x_0, z)$) for clustering with planar Laplace

Ensure: z
  $x_1, y_1 \leftarrow x_{min}$
  $x_2, y_2 \leftarrow x_{max}$
  $z_x, z_y \leftarrow z$
  if $x_1 < z_x < x_2$ and $y_1 < z_y < y_2$ **then**
    return z
  else
    $x, y \leftarrow x_0$
    $z_2 \leftarrow LP(\epsilon, x, y)$ ▷ See formula 3.3.
    return $T(x_{min}, x_{max}, x_0, z_2)$ ▷ Rerun recursively
  end if

within the domain using respectively the minimum X/Y and maximum X/Y. An example of this is visualized:

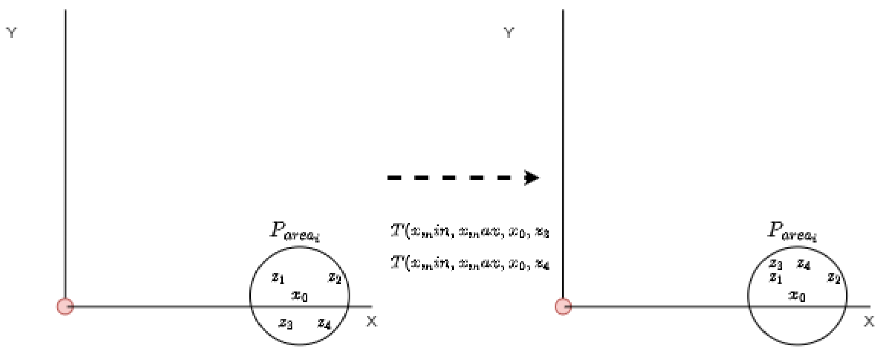

Figure 4.1: Representation of the remapping algorithm for clustering for points z_3 and z_4

Algorithm 2 Full algorithm for perturbing cluster data based on planar/2D-Laplace [Andrés et al., 2012]

Require: $x \in X$ ▷ 2D array of points
Require: $l \in R^+$
Ensure: $z \in Z$ ▷ 2D array of perturbed points
  $r = \frac{\sigma}{2}$ ▷ formula 4.1
  $\epsilon = \frac{l}{r}$ ▷ Calculating privacy budget [Andrés et al., 2012]
  $x_{min} \leftarrow min(X)$
  $x_{max} \leftarrow max(X)$
  $Z \leftarrow []$
  for $point_i \in X$ **do**
    $\theta \leftarrow [0, \pi 2]$ ▷ Random noise for θ
    $p \leftarrow [0, 1]$
    $z_i \leftarrow C_\epsilon{}^{-1}(p)$ ▷ formula 3.2
    $z_i \leftarrow T(x_{min}, x_{max}, point_i, z_i)$ ▷ algorithm 1.
    $x_{perturbed} \leftarrow point_{i_x} + (z_{i_x} * \cos(\theta))$ ▷ add noise to x-coordinate
    $y_{perturbed} \leftarrow point_{i_y} + (z_{i_y} * \sin(\theta))$ ▷ add noise to y-coordinate
    append $x_{perturbed}, y_{perturbed}$ to Z
  end for

ALGORITHM

The full algorithm is displayed here.

4.3.3. RESEARCH QUESTION 2

4.3.4. RESEARCH QUESTION 3

4.4. RESULTS

4.4.1. RESEARCH QUESTION 1

4.4.2. RESEARCH QUESTION 2

4.4.3. RESEARCH QUESTION 3

Bibliography

Miguel E. Andrés, Nicolás Emilio Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. *CoRR*, abs/1212.1984, 2012. 6, 7, 8, 11

Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Marco Stronati. Constructing elastic distinguishability metrics for location privacy. *Proceedings on Privacy Enhancing Technologies*, 2015(2):156–170, June 2015. ISSN 2299-0984. doi: 10.1515/popets-2015-0023. 8

Jussi Lehtonen. The Lambert W function in ecological and evolutionary models. *Methods in Ecology and Evolution*, 7(9):1110–1118, 2016. ISSN 2041-210X. doi: 10.1111/2041-210X.12568. 7

Yan Yan, Fei Xu, Adnan Mahmood, Zhuoyue Dong, and Quan Z. Sheng. Perturb and optimize users' location privacy using geo-indistinguishability and location semantics. *Scientific Reports*, 12(1):20445, November 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-24893-0. 8

Math symbols

X Set of locations for a user. (R^2). 6

Z For every $x \in X$ a perturbed location $z \in Z$ is reported.. 6

ϵ Defined as $\epsilon = l/r$. 6

θ Angle. 6

l Privacy level. 6

r Radius. 6