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

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

2.1.2. LOCAL DIFFERENTIAL PRIVACY

2.1.3. EVALUATION METHODS

Is also possible to evaluate and measure the impact of the noise between two distributions.

RE versus other methods

2.2. Clustering

2.2.1. Methods

Describe each clustering method with important parameters that could influence the outcome

2.2.2. Evaluation methods

Clustering comparison measures are important in cluster analysis for external validation by comparing clustering solutions to a "ground truth" clustering [Vinh et al.]. These external validity indices are a common way to assess the quality of unsupervised machine learning methods like clustering [Warrens and van der Hoef, 2022]. A method that could be used for this is the Rand Index Rand [1971]. It is a commonly applied method for comparing two different cluster algorithms Wagner and Wagner. An improvement of this method is adjusted for chance by considering the similarity of pairwise cluster comparisons Vinh et al.. Both the Rand Index (RI) and Adjusted Rand Index (ARI) [Hubert and Arabie, 1985] report a value between 0 and 1. Where 0 is for no-similarity and 1 for identical clusters. Alternatives for RI are the Fowles-Mallows Index and Mirkin Metric. However, these two methods have their disadvantages. Respectively, being sensitive to a few clusters and cluster sizes [Wagner and Wagner]. The ARI metric suffers from cluster size imbalance as well, so it only provides not a lot of information on smaller clusters [Warrens and van der Hoef, 2022]. Instead, they recommend using the cluster index metric that was proposed by Fränti et al. [Fränti et al., 2014].

Another popular group of methods is the information theoric-based measures [Vinh et al.]. This metric measures the information between centroids; the higher the value, the better [Vinh et al.]. Mutual Information (MI) is such metric, which calculates the probability of an element belonging to cluster C or C'. But, is not easy to interpret as it does not have a maximum value [Wagner and Wagner]. To this end, Normalized Mutual Information (NMI) can be used to report a value between 0 and 1 using the geometric mean [Strehl and Ghosh, 2002]. The metric exists also in an adjusted version as Adjusted Mutual Information (AMI). This works in the same way as for the Adjusted Rank Index (ARI) and is mostly needed if the number of data items is small in comparison to the number of clusters [Vinh et al.].

Comparison

Compare AMI / against other

Analyze existing literature and what they use

2.3. LITERATURE REVIEW

Mostly based on the preparation, and summarized here later

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 (GI) 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.

Geo-indistinguishability

As mentioned in the previous section, the GI method can be applied to preserve the privacy using a differential privacy method specific to spatial data. The formula to measure if an algorithm preserves ϵ-geo-indistinguishability can be expressed as [Andrés et al., 2012]:

$$K(x)(y) \leq e^{\epsilon * d(x,x')} * K(x')(y) \tag{3.1}$$

Where K is a probability method reporting $x, x' \in X$ as $z \in Z$. The idea of this algorithm looks a lot like that of differential privacy using the La Place method; but includes distance. The intuition for this is that it displays the distinguishability level between two secret locations/points x and x' [Chatzikokolakis et al., 2015]. An extension of this is called d_x-privacy and is a more general notation of distance-aware differential privacy. Their definition for GI is, therefore, d_2-privacy, but is essentially the same as the proof provided for GI.

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:

$$\frac{\epsilon^2}{2 * \pi} e(-\epsilon d(x_0, x)) \tag{3.2}$$

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 for calculating r and θ.

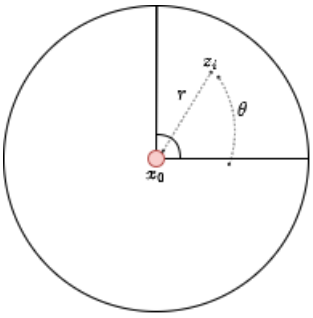

Figure 3.1: Representation of the generated $z = r\theta$ and original point $x0$.

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:

$$C_\epsilon{}^{-1}(p) = -\frac{1}{\epsilon}(W\_1(\frac{p-1}{e}) + 1) \tag{3.3}$$

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.

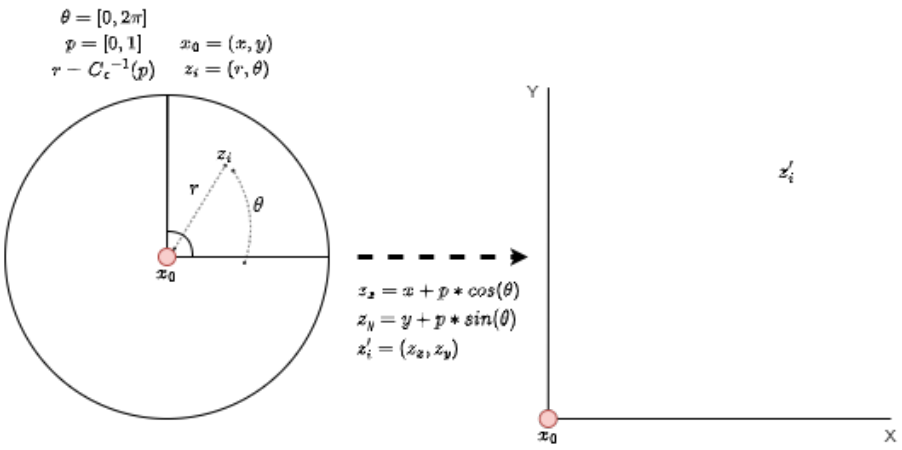

Figure 3.2: Representation of converting the perturbed point $z = (r, \theta)$ to a point z_x, z_y

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

For running the experiments we make use of 16GB ram memory and i7-10750H 2.6Ghz processor. The experiments are run using a Docker container which runs a pre-configured distribution of Linux Alpine. It includes a pre-installed Anaconda environment for python [1,2]. We run the container using the dev-container feature for visual-studio code [3]. This allows us to create a reproducible experiment environment.

[1] https://github.com/devcontainers/images/tree/main/src/anaconda
[2] tag: mcr.microsoft.com/devcontainers/anaconda:0-3
[3] https://code.visualstudio.com/docs/devcontainers/containers

4.2.1. Libraries & code versions

We use python version 3.9.13 with Jupyter notebook for creating a reproducible experimental environment. The packages for python are:

1. Scikit-learn: 1.0.*
2. Yellow-brick: 1.5
3. Numpy: 1.24.*
4. Pandas: 1.4.*
5. Seaborn: 0.11.*
6. Mathplotlib: 3.5.*

4.3. Methods

This section explains what methods/ algorithms we used and how we evaluate them.

4.3.1. Clustering methods

Exact parameters we used for the algorithms

Which algorithms we used

4.3.2. Evaluation

With differential privacy, it is a trade-off of utility versus privacy. Therefore, for the evaluation of the 2D/3D-Laplace algorithms, we compare both criteria to achieve a consensus between utility and privacy.

Utility

The utility of the cluster algorithm is decided by calculating the AMI between the baseline cluster algorithms. The clustering algorithm is trained using the plain data and functions as the ground truth [Sun et al., 2019; **?**]. Because of this, we are being able to calculate the AMI and compare the centroids between the non-private and privately trained clusters. To reduce the possible bias of results we executed them 10 times for multiple privacy budgets and report the average for each [Huang et al., 2021].

Explain why AMI and not another

The second way to measure utility is to calculate the error between the non-private and perturbed data [Huang et al., 2021; Sun et al., 2019; Xia et al., 2020]. There are several methods to do this (See 2.2.2), but we use the Average Estimated Error (AEE). As with AMI we run the calculations for multiple privacy budgets 10 times and report the average for each budget.

Explain why AEE and not RE

Privacy

The most important one here is the preserving of GI. This validates that we applied the algorithms in the right way and automatically inherit the strong privacy guarantees provided by GI (3.1). A disadvantage of this method is that it cannot be used to achieve a clear representation of privacy (it is either "yes" or "no"). Therefore, we also calculate the average Euclidean distance between the non-private and perturbed data.

4.3.3. RESEARCH QUESTION 1

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 distance information, 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:

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

This algorithm uses x_{min} and x_{max} to re-calculate the points within the domain using respectively the minimum X/Y and maximum X/Y. An example of this is visualized:

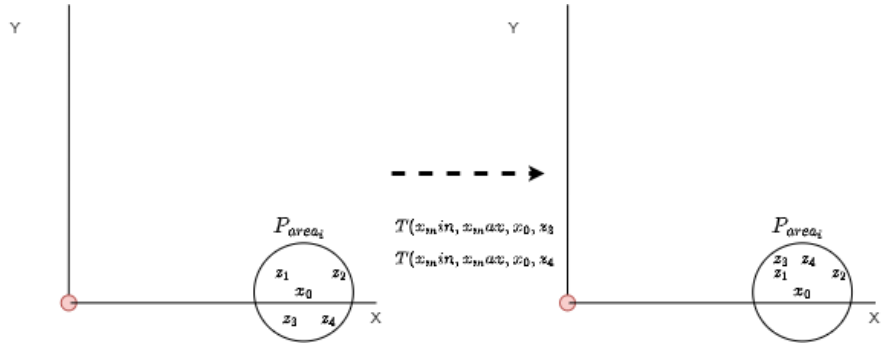

Figure 4.1: Representation of the remapping algorithm for clustering for points z_3 and z_4

PROBABILITY METRIC $K(x)(Z)$

Explain the probability metric K we used

ALGORITHM
The full algorithm for the perturbation:

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
  return Z

4.3.4. RESEARCH QUESTION 2

Starts after RQ1

4.3.5. RESEARCH QUESTION 3

Starts after RQ2

4.4. RESULTS

4.4.1. RESEARCH QUESTION 1

For research question 1 the results are 2-dimensional plotted using a line diagram.

UTILITY
PRIVACY

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, 13

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. 6, 8

Pasi Fränti, Mohammad Rezaei, and Qinpei Zhao. Centroid index: Cluster level similarity measure. *Pattern Recognition*, 47(9):3034–3045, September 2014. ISSN 00313203. doi: 10.1016/j.patcog.2014.03.017. 4

D. Huang, X. Yao, S. An, and S. Ren. Private distributed K-means clustering on interval data. In *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)*, pages 1–9, Los Alamitos, CA, USA, October 2021. IEEE Computer Society. doi: 10.1109/IPCCC51483.2021.9679364. 11

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2: 193–218, 1985. ISSN 0176-4268. 4

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

William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971. ISSN 0162-1459. 4

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, 2002. 4

Lin Sun, Jun Zhao, and Xiaojun Ye. Distributed Clustering in the Anonymized Space with Local Differential Privacy, June 2019. 11

Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. 4

Silke Wagner and Dorothea Wagner. Comparing Clusterings - An Overview. 4

Matthijs J. Warrens and Hanneke van der Hoef. Understanding the Adjusted Rand Index and Other Partition Comparison Indices Based on Counting Object Pairs. *Journal of Classification*, 39(3):487–509, November 2022. ISSN 1432-1343. doi: 10.1007/s00357-022-09413-z. 4

Chang Xia, Jingyu Hua, Wei Tong, and Sheng Zhong. Distributed K-Means clustering guaranteeing local differential privacy. *Computers & Security*, 90:101699, 2020. ISSN 0167-4048. 11

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

Glossary

Adjusted Mutual Information Comparable with Adjusted Rand Index this algorithm is modified to account to chance. This means it accounts for a higher MI for a higher amount of clusters between two cluster algorithms. Therefore, the calculations are strongly influenced by that of Adjusted Rand Index [**?**]. . 4, 6

Adjusted Rand Index The Rand Index is improved and adjusted for chance [**?**]. This algorithm takes also into consideration the number of clusters and can be used to also compare different cluster algorithms [**?**].. ii, 4, 6

Average Estimation Error This is the difference between an estimated value and the real value.. 6, 11

Bit Vector List or array to store several bits.. 6

Calinski-Harabasz Index This is a way to measure the similarity of clusters [**?**]. It tells how well the clusters are separated from each other and how well the points are grouped.. 6

Mutual Information This metric can be used to explain the amount of information about a random variable if compared to another random variable. Therefore, it can also be used to compare two cluster similarities.. ii, 4, 6

Normalized Mutual Information The normalized version is a scaled version of Mutual Information to always be a value between 0 (no correlation) and 1 (perfect correlation). This version of Mutual Information is not adjusted and therefore highly influenced by cluster amount [**?**]. So it suffers the same issue as with Mutual Information.. 4, 6

Rand Index Compares the similarity between two clusters by comparing all pairs. It can therefore be used to measure the performance between two clustering algorithms [**?**]. . 6

Acronyms

AEE Average Estimated Error. 6, 11, *Glossary:* Average Estimation Error

AMI Adjusted Mutual Information. 4, 6, 11, *Glossary:* Adjusted Mutual Information

AP Affinity Propogation. 6

ARI Adjusted Rank Index. 4, 6, *Glossary:* Adjusted Rand Index

BIRCH Balanced Iterative Reducing and Clustering using Hierarchies. 6

BV Bit Vector. 6

CHI Calinski-Harabasz Index. 6, *Glossary:* Calinski-Harabasz Index

DBSCAN Density-based spatial clustering of applications with noise. 6

DP Differential Privacy. 6

DPC Density Peaks Clustering. 6

GI Geo-indistinguishability. 6, 11

LDP Local Differential Privacy. 6

MI Mutual Information. 4, 6, *Glossary:* Mutual Information

NMI Normalized Mutual Information. 4, 6, *Glossary:* Normalized Mutual Information

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