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
de beste!
www.ou.nl

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

Explain general notion of privacy

MATH SYMBOLS

$K(x)(Z)$ Randomization method for $x \in X$ and output $z \in Z$..

$Pr(K(x_i) \in (Z))$ Probability of reporting $x \in X$ for $z \in Z$.

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

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

ϵ Privacy budget.

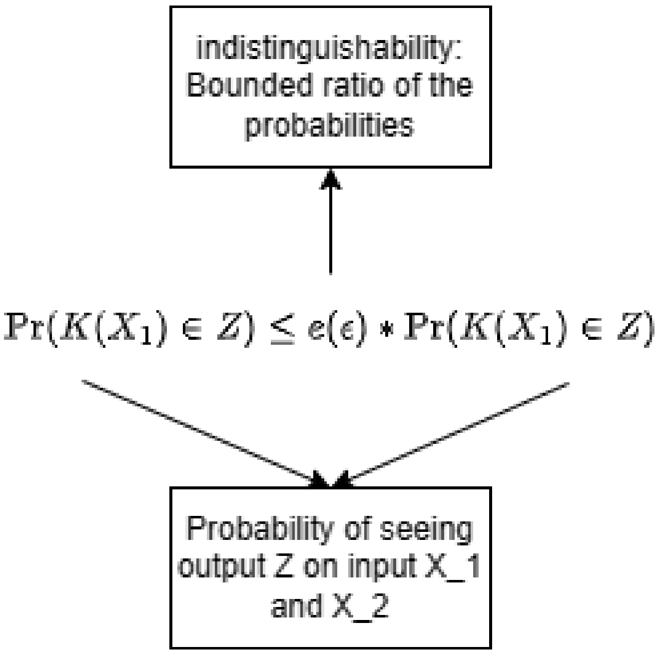

Figure 2.1: Randomization function K gives ϵ-differential privacy for all elements in D_1 and D_2 if they differ at most one element. [Dwork, 2006]

The privacy budget ϵ determines the amount of noise that is added.

2.1.1. LAPLACE ALGORITHM

One way to achieve ϵ-DP is using sampling noise from the Laplace or Gaussian distributions.

Explain gaussian / laplace distributions

The noise is then based on the sensitivity of a function f. This is the maximal possible change when adding or removing a single record [Dwork, 2006; Friedman and Schuster, 2010].

$$\Delta f = max_{D_1,D_2} ||f(D_1) - f(D_2)|| \tag{2.1}$$

Explain differential privacy implementation with La place distribution

2.1.2. LOCAL DIFFERENTIAL PRIVACY

2.1.3. GEO-INDISTINGUISHABILITY

2.1.4. ATTACKS ON PRIVACY

Membership inference attacks: An attack model that plays a big role in machine learning is a membership inference attack. With this attack, an adversary attempts to infer the original data point $x \in X$ from a given data point $z \in Z$. The adversary has access to a point z, the size of the dataset $|Z|$ and a distribution D where Z was drawn from [Yeom et al., 2018]. These attacks depend on the adversarial knowledge, which can be divided into white-box and black-box MIA's [Hu et al., 2022].

1. **White-box**: The attacker has all the data that is needed. Including target model parameters, the training dataset and even the architecture [Hu et al., 2022].

2. **Black-box**: The attacker has a limited amount of information, like training data distribution and the trained model [Hu et al., 2022].

An approach called *Binary classifier membership inference attacks* is used to separate members from non-members. [Hu et al., 2022]. This method evolves around the attacker generating a shadow model, with as goal to overfit [Shokri et al., 2017]. If the data is fed with real data the score is higher than similar data, which means the real data can be inferred [Jayaraman and Evans; Shokri et al., 2017] A white-box setting requires a lot of adversarial knowledge for training the shadow models. The black-box settings only take the prediction as input and decide if it is a (non-) member [Hu et al., 2022]. An unsupervised black-box MIA was introduced by Peng et al. and only considers that the attacker has access to the already trained model. They rescale the probabilities first using temperature scaling, to compensate for models that are overconfident [Peng et al.]. So instead of having a probability between two classes with for example 99% against 1% it will be more evenly distributed based on the training data. They then proceed in clustering the probabilities into two clusters using K-Means and label the higher confidence scores as members.

The above attacks do rely on the model to also provide the confidence or probabilities of the predictions. This is often not the case for the practical appliance of a model, and therefore Choquette-Choo et al. introduced a label-only attack. While the existing models exploit the probability output for MIA, they solely rely on labels. For this, they make use of the "HopSkipJump" attack; a so-called decision-based attack [Chen et al., 2020]. Choquette-Choo et al. consider a more semi-black-box approach, for which the attacker still requires access to a subset of the original training data and the trained model. Another paper that also uses "HopSkipJump" requires only the trained model and achieves higher accuracy by using an approach with random data [Li and Zhang, 2021].

Another take on this is prediction and confidence-based MIA which are both proposed by [Yeom et al., 2018]. They assume that an attacker knows the standard error and has access to the perturbation dataset. The algorithm is be-able to extract the truth label by minimizing the loss.

To conclude on this, there are many methods for MIA and in that regard; the unsupervised methods look the most promising. They require less setup and there is plenty of black-box approaches that score between 70% and 80% success rates. For our use case, however, it is harder to establish an MIA; as we focus mainly on clustering. Anyhow, it is possible if we consider a semi-supervised approach where we consider the cluster labels as ground truth (**??**) Differential privacy is proposed as a way of solving the inference attack for both white-box and black-box [Hu et al., 2022]. However, it is hard to find a way to protect privacy and utility as well, so it depends heavily on the privacy budget. In addition, it is hard to establish a member-inference attack on clustering as it requires a semi-supervised approach to be effective, as MIA requires classification **??**.

2.1.5. EVALUATION METHODS

It is possible to evaluate and measure the impact of the noise between two distributions by calculating the error between the non-private and private data [Xiong et al., 2020]. Two

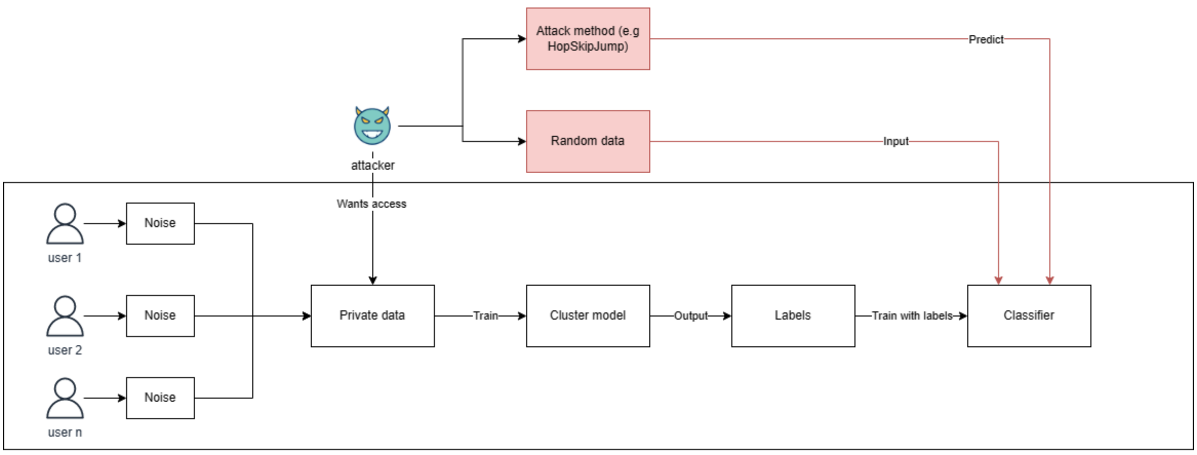

Figure 2.2: Semi-supervised membership inference attack considering a black-box approach [Chen et al., 2020; Li and Zhang, 2021]

metrics that are proposed by the same study are Mean Squared Error (MSE) and Mean Average Error (MAE). These metrics can be used to calculate the error between X and the perturbed dataset Z.

Just as it is possible to measure the utility, this can also be done with privacy. When performing a privacy algorithm, it can be proven whether a method meets the privacy requirements. These are metrics such as ϵ-differential-privacy (2.1) and ϵ-geo-indistinguishability (see next chapter 3.1). Although these methods give an idea of privacy, it can only be "yes" or "no". Furthermore, it can give a distorted image, since a chance of 70% also gives a "yes" according to the definition of geo-indistinguishability [Oya et al., 2017]. In other words, to gain more insight into the amount of privacy (such as with MSE or MAE), other metrics are needed.

For this reason, Oya et al. introduced a metric for geo-indistinguishability that makes it possible to give percentages in their study [Oya et al., 2017]: As an example, an adversary is given that guesses between two locations: $x \in X$ and $x' \in X$.

$$p_e(x, x', z) \le p_e^* = \frac{1}{1 + e^{e * d(x,x')}} \tag{2.2}$$

Where privacy level p_e^* is the lower bound of the probability of an adversary guessing correctly. The method is called ϵ-geo-indistinguishability as error. Based on this metric, it can be calculated that an adversary has an average of 90% chance to guess a location correctly. In that case, the algorithm would be ϵ-geo-indistinguishability, but in practice not.

2.2. CLUSTERING

2.2.1. METHODS

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

K-MEANS

Explain the working of the algorithm

. The most important parameter of the K-Means algorithm is the value of k. This value determines the number of clusters to consider and has a big influence on the results **?**. One of the oldest methods to do this is to use an "elbow" plot Kodinariya and Makwana [2013]. This method can be used to determine the best k by applying the algorithm multiple times and estimateing the best k.

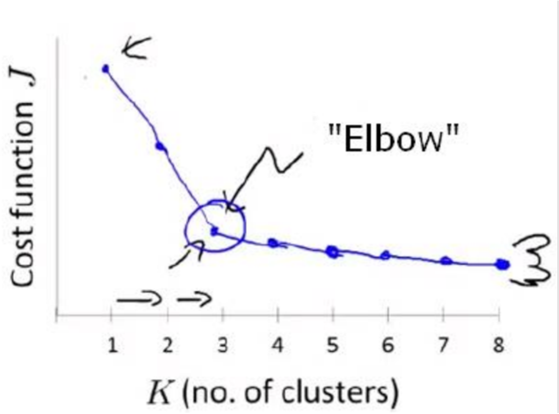

Figure 2.3: Illustration of determining k using the "elbow" method Kodinariya and Makwana [2013]

However, sometimes the "elbow" is hard to find

AFFINITY PROPAGATION

:

Explain the working of the algorithm

.

Explain most important parameters

DBSCAN

: [Bozdemir et al.]

Explain the working of the algorithm

.

Explain most important parameters

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.].

Besides the external validity measurements for clustering, it is also possible to use internal validation methods. These metrics focus entirely on the intrinsic dataset properties, instead of relying on an external baseline cluster algorithm Craenendonck and Blockeel. Assessing two important concepts of clustering: compactness and separation Hassani and Seidl [2017]. Both studies, consider three different metrics and measure both concepts at the same time Hassani and Seidl [2017]:

1. Calinski-Harabasz Index (CHI) [Caliński and Harabasz, 1974] is used to measure the cluster variance (well-separated clusters) and low variance within the clusters (tightly coupled data). A high score indicates better clustering.

2. Silhouette Index [Rousseeuw, 1987] this metric is similar, by also measuring cohesion within clusters and separation of clusters. However, this metric uses the pairwise distance Hassani and Seidl [2017]. A score of -1 indicates incorrect clustering and +1 for dense clusters Rousseeuw [1987].

3. Davies-Bouldin [Davies and Bouldin, 1979] uses the average distance between centroids. A lower score indicates good clustering.

K-Means scores relatively high for CHI [Craenendonck and Blockeel; Hassani and Seidl, 2017] and SI [Craenendonck and Blockeel]. The same applies to DBSCAN, which scores relatively high on SI and DB due to the sensitivity of noise Craenendonck and Blockeel.

Existing literature

Comparable studies with differential privacy use external validation [Sun et al., 2022; Xia et al., 2020]. Their experiment setup uses a so-called non-private cluster algorithm as external validation. This cluster algorithm is trained without the perturbed data and compared with the same clustering algorithm that is trained with perturbed data. Thus, the non-private variant functions as an external validation by providing the ground truth.

They compare the mutual information between a baseline cluster algorithm using AMI [Huang et al., 2021] or NMI [Sun et al., 2022; Xia et al., 2020]. Another study for evaluating Differential Privacy (DP) with Affinity Propogation (AP) uses both ARI and AMI. In addition to mutual information and rand index scores, it is also not uncommon to calculate the error between the two cluster algorithm's centroids [Huang et al., 2021; Xia et al., 2020]. These two studies used Relative Error (RE) for this.

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

θ Angle.

l Privacy level.

r Radius.

The other symbols can be found in section 2.1.

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]:

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

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.

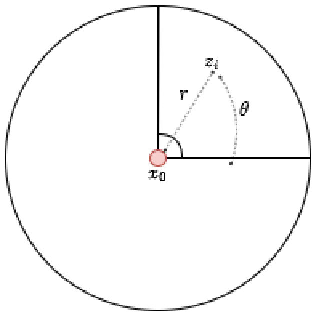

Figure 3.1: Representation of the generated $z = r\theta$ and original point $x0$.

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 **??**.

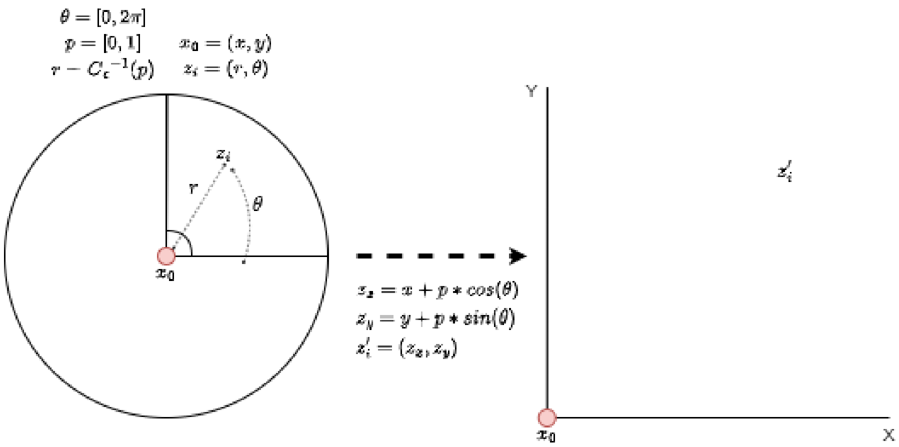

Figure 3.2: Representation of converting the perturbed point $z = (r,\theta)$ to a point z_x, z_y

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 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. In addition to this, we used the LSun dataset as a lot of comparable literature uses this as well for evaluation.

Describe LSun dataset

Records	Centers	Dimensions	Standard deviation	Research
50	4	2	0.60	RQ 1
50	4	3	0.60	RQ 2
50	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

For the three different algorithms: K-Means, AP and Density-based spatial clustering of applications with noise (DBSCAN) we analyzed the most important decisions regarding parameter selection. In this section, we give a short list and explanation of the different parameters we used throughout the experiments. For all three Sklearn was used, and for each of them we also provide the underlying formula.

K-Means

Work in progress

$$\sum_{i=0}^{n} \min_{\mu_j \in C}(||x_i - \mu_j||^2) \tag{4.1}$$

Parameter	Description	Value

Table 4.1: K-Means provided by the Scikit-learn package

Affinity Propagation

Work in progress

Parameter	Description	Value
Row 1 Data 1	Row 1 Data 2	Row 1 Data 3

Table 4.2: Affinity Propagation provided by the Scikit-learn package

DBSCAN

Work in progress

Parameter	Description	Value
Row 1 Data 1	Row 1 Data 2	Row 1 Data 3

Table 4.3: DBSCAN provided by the Scikit-learn package

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

Based on chapter 2.2.2, we can conclude that the corresponding literature mainly evaluates one clustering algorithm and not multiple ones. Because of this, we do not use any internal validation; because the different types of metrics are too easily influenced by the choice of clustering algorithm. Instead, we decided to pick up external validation methods similar to other studies as well. They did use either Rand Index or Mutual Information, but because both have different strengths we evaluate both.

It is not likely that we will use many data points for research questions 1 and 2. To compensate for this, the adjusted version is used for both Rand Index and Mutual Information. Since both Mutual Information and Rand Index have different characteristics in the type of clustering algorithm, we use both. 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].

Finally, the implementation for these metrics is provided by the Scikit-learn package. With the underlying formulas:

$$AMI(U,V) = \frac{MI(U,V) - E(MI(U,V))}{avg(H(U),H(V)) - E(MI(U,V))} \tag{4.2}$$

Adjusted Mutual Information formula [Hubert and Arabie, 1985; Vinh et al.]

$$RI = \frac{a+b}{C_2^n} \tag{4.3}$$

$$ARI = \frac{RI - E(RI)}{max(RI) - E(RI)} \tag{4.4}$$

(Adjusted) Rand Index formula [Hubert and Arabie, 1985; Rand, 1971]

Privacy

The most important one here is the preserving of GI according to the formula 3.1. This validates that we applied the algorithms in the right way and automatically inherit the strong privacy guarantees provided by GI. 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 analyze our method according to a popular attack: Membership inference attack.

For this purpose, we make use of a black-box Member inference attack, called "HopSkipJump" [Chen et al., 2020; Li and Zhang, 2021]. This attack is evaluated using a semi-supervised setup, as proposed in this figure: **??**. We will make use of a decision tree model for classification but can be replaced by any other classification model. Both the private and non-private trained models are evaluated based on the true positive rate (TPR) and false positive rate (FPR). Respectively meaning, the TPR is higher if the MIA is successful and likewise the FPR if the MIA is unsuccessful. We hypothesize that the private model leverages a higher FPR in comparison to the non-private variant.

4.3.3. Scaling

Because we use a distance metric, we need to apply some data standardization. For this purpose, we use standard scaling provided by the Scikit-learn package [4]. This is only for clustering, so it is applied after all the perturbation algorithms.

[4] https://scikit-learn.org/stable/modules/preprocessing.html

4.3.4. 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:

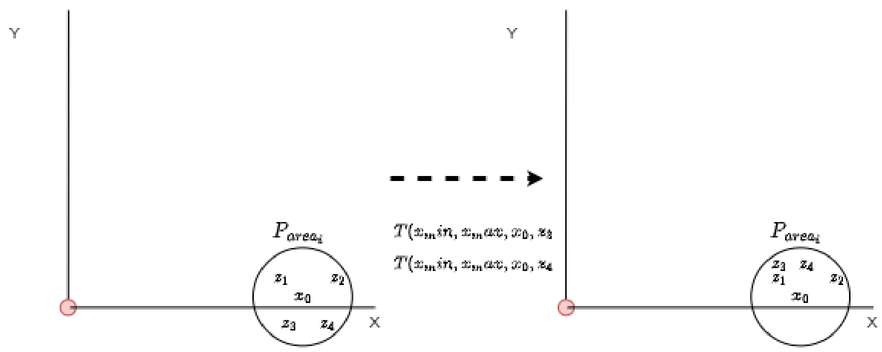

Figure 4.1: Representation of the remapping algorithm for clustering for points z_3 and z_4

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 ∈ X                                          ▷ 2D array of points
Require: l ∈ R+
Ensure: z ∈ Z                                  ▷ 2D array of perturbed points
  r = σ/2                                                     ▷ formula 4.1
  ε = l/r                  ▷ Calculating privacy budget [Andrés et al., 2012]
  x_min ← min(X)
  x_max ← max(X)
  Z ← []
  for point_i ∈ X do
      θ ← [0, π2]                                    ▷ Random noise for θ
      p ← [0, 1]
      z_i ← C_ε^{-1}(p)                                       ▷ formula 3.2
      z_i ← T(x_min, x_max, point_i, z_i)                     ▷ algorithm 1.
      x_perturbed ← point_{i_x} + (z_{i_x} * cos(θ))   ▷ add noise to x-coordinate
      y_perturbed ← point_{i_y} + (z_{i_y} * sin(θ))   ▷ add noise to y-coordinate
      append x_perturbed, y_perturbed to Z
  end for
  return Z
```

4.3.5. Research question 2

Starts after RQ1

4.3.6. 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

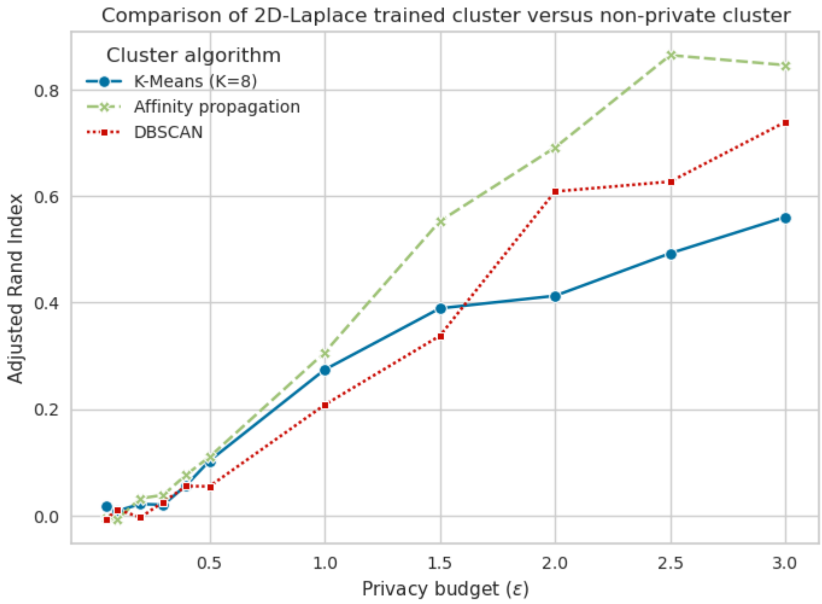

Figure 4.2: ARI evaluation for cluster algorithms 2D-Laplace for a dataset with shape (50, 2)

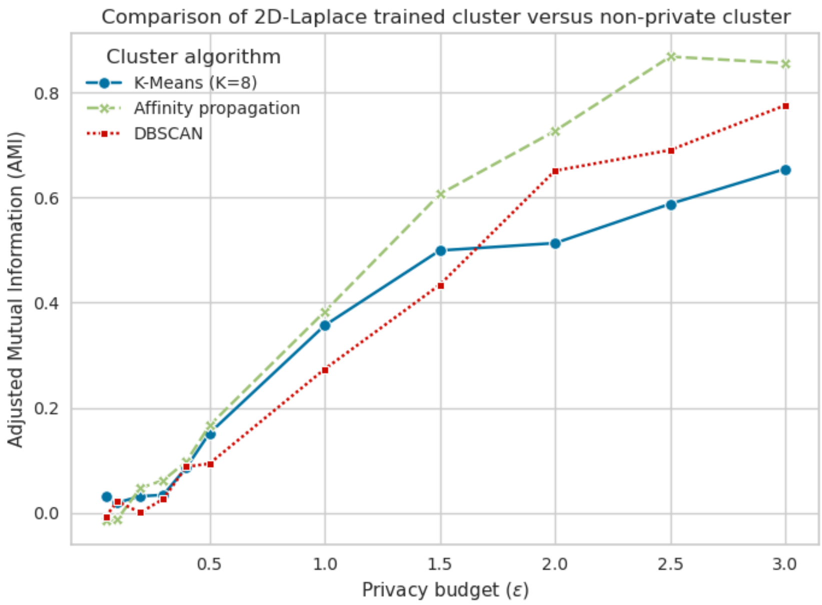

Figure 4.3: AMI evaluation for cluster algorithms trained with 2D-Laplace for a dataset with shape (50, 2)

Privacy

Table 4.4: Geo-indistinguishability as an error metric

epsilon	average p_e	distance
0.05	0.447944	3.854884
0.1	0.397897	3.761739
0.2	0.309593	2.899825
0.3	0.240543	3.234295
0.4	0.189171	2.508862
0.5	0.151567	2.461655
1	0.067893	1.282425
1.5	0.043456	1.178248
2	0.032263	0.900766
2	0.025261	0.591023
3	0.020199	0.645654

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. 9, 10, 11, 17

Beyza Bozdemir, Sébastien Canard, Orhan Ermis, Helen Möllering, Melek Önen, and Thomas Schneider. Privacy-preserving Density-based Clustering. 6

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974. ISSN 0090-3272. 7

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. 9, 11

Jianbo Chen, Michael I. Jordan, and Martin J. Wainwright. HopSkipJumpAttack: A Query-Efficient Decision-Based Attack, April 2020. 4, 5, 15

Toon Van Craenendonck and Hendrik Blockeel. Using Internal Validity Measures to Compare Clustering Algorithms. 7

David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979. ISSN 0162-8828. 7

Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II 33*, pages 1–12. Springer, 2006. ISBN 3-540-35907-9. 3

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. 7

Arik Friedman and Assaf Schuster. Data mining with differential privacy. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 493–502, Washington DC USA, July 2010. ACM. ISBN 978-1-4503-0055-1. doi: 10.1145/1835804.1835868. 3

Marwan Hassani and Thomas Seidl. Using internal evaluation measures to validate the quality of diverse stream clustering algorithms. *Vietnam Journal of Computer Science*, 4 (3):171–183, August 2017. ISSN 2196-8896. doi: 10.1007/s40595-016-0086-9. 7

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. Membership Inference Attacks on Machine Learning: A Survey, February 2022. 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. 7, 15

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2: 193–218, 1985. ISSN 0176-4268. 6, 15

Bargav Jayaraman and David Evans. Evaluating Differentially Private Machine Learning in Practice. 4

Trupti M Kodinariya and Prashant R Makwana. Review on determining number of Cluster in K-Means Clustering. *International Journal*, 1(6):90–95, 2013. 6

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. 10

Zheng Li and Yang Zhang. Membership Leakage in Label-Only Exposures, September 2021. 4, 5, 15

Simon Oya, Carmela Troncoso, and Fernando Pérez-González. Is Geo-Indistinguishability What You Are Looking for? In *Proceedings of the 2017 on Workshop on Privacy in the Electronic Society*, pages 137–140, Dallas Texas USA, October 2017. ACM. ISBN 978-1-4503-5175-1. doi: 10.1145/3139550.3139555. 5

Yuefeng Peng, Bo Zhao, and Hui Liu. Unsupervised Membership Inference Attacks Against Machine Learning Models. 4

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. 6, 15

Peter J Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. ISSN 0377-0427. 7

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks against Machine Learning Models, March 2017. 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. 7

Lin Sun, Guolou Ping, and Xiaojun Ye. PrivBV: Distance-aware encoding for distributed data with local differential privacy. *Tsinghua Science and Technology*, 27(2):412–421, April 2022. ISSN 1007-0214. doi: 10.26599/TST.2021.9010027. 7

Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. 6, 7, 15

Silke Wagner and Dorothea Wagner. Comparing Clusterings - An Overview. 6, 7

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. 6, 7

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. 7

Xingxing Xiong, Shubo Liu, Dan Li, Zhaohui Cai, and Xiaoguang Niu. A Comprehensive Survey on Local Differential Privacy. *Security and Communication Networks*, 2020:8829523, October 2020. ISSN 1939-0114. doi: 10.1155/2020/8829523. 4

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. 11

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 268–282, Oxford, July 2018. IEEE. ISBN 978-1-5386-6680-7. doi: 10.1109/CSF.2018.00027. 4

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 [?]. . 3, 7, 9

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 [?].. iii, 3, 7, 9

Average Estimation Error This is the difference between an estimated value and the real value.. 3, 9

Bit Vector List or array to store several bits.. 3, 9

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.. 3, 7, 9

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.. iii, 3, 7, 9

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.. 3, 7, 9

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 [?]. . 3, 9

Acronyms

AEE Average Estimated Error. 3, 9, *Glossary:* Average Estimation Error

AMI Adjusted Mutual Information. 3, 7, 9, *Glossary:* Adjusted Mutual Information

AP Affinity Propogation. 3, 7, 9, 14

ARI Adjusted Rank Index. 3, 7, 9, *Glossary:* Adjusted Rand Index

BIRCH Balanced Iterative Reducing and Clustering using Hierarchies. 3, 9

BV Bit Vector. 3, 9

CHI Calinski-Harabasz Index. 3, 7, 9, *Glossary:* Calinski-Harabasz Index

DBSCAN Density-based spatial clustering of applications with noise. 3, 9, 14

DP Differential Privacy. 3, 7, 9

DPC Density Peaks Clustering. 3, 9

GI Geo-indistinguishability. 3, 9, 15

LDP Local Differential Privacy. 3, 9

MI Mutual Information. 3, 7, 9, *Glossary:* Mutual Information

NMI Normalized Mutual Information. 3, 7, 9, *Glossary:* Normalized Mutual Information

MATH SYMBOLS

θ Angle. 3, 9

l Privacy level. 3, 9

r Radius. 3, 9

MATH SYMBOLS

$K(x)(Z)$ Randomization method for $x \in X$ and output $z \in Z$.. 3, 9

$Pr(K(x_i) \in (Z))$ Probability of reporting $x \in X$ for $z \in Z$. 3, 9

X Set of locations for a user. (R^2). 3, 9

Z For every $x \in X$ a perturbed location $z \in Z$ is reported.. 3, 9

ϵ Privacy budget. 3, 9