Anonymisation des données par apprentissage non supervisé

Sarah Zouinina

Numéro d'Ordre : D.U. . . .
EDSPIC : . . .

UNIVERSITÉ SORBONNE PARIS NORD -
UNIVERSITÉ ABDELMALEK ESSAADI

LIPN CNRS UMR 7030 - LTI

Thesis

Presented by

Sarah Zouinina

For the degree of

DOCTOR OF COMPUTER SCIENCE

Data Anonymisation through Unsupervised Learning

Reviewers:

Pr.	Omar EL BAQQALI	USMBA, Morocco
Pr.	Andonie RAZVAN	CWU, USA
Pr.	Abdelfettah SEDQUI	ENSATg, Morocco

Examiners:

Pr. Younès BENNANI	USPN, France	(Director)
Pr. Olivier BODINI	USPN, France	(Examiner)
Dr. Guénaël CABANES	USPN, France	(Examiner)
Pr. Abdelouahid LYHYAOUI	ENSATg, Morocco	(Director)
Dr. Parisa RASTIN	Ecole des Mines, France	(Examiner)
Dr. Nicoleta ROGOVSCHI	Université de Paris, France	(co-supervisor)

Numéro d'Ordre : D.U. . . .
EDSPIC : . . .

UNIVERSITÉ SORBONNE PARIS NORD -
UNIVERSITÉ ABDELMALEK ESSAADI

LIPN CNRS UMR 7030 - LTI

Thèse

Présentée par

Sarah Zouinina

Pour l'obtention du grade de

DOCTEUR DE L'UNIVERSITÉ SORBONNE PARIS NORD

Spécialité : Informatique

Anonymisation des Données par Apprentissage Non Supervisé

Rapporteurs:

Pr.	Omar EL BAQQALI	USMBA, Maroc
Pr.	Andonie RAZVAN	CWU, USA
Pr.	Abdelfettah SEDQUI	ENSATg, Maroc

Examinateurs:

Pr. Younès BENNANI	USPN, France	(Directeur)
Pr. Olivier BODINI	USPN, France	(Examiner)
Dr. Guénaël CABANES	USPN, France	(Examinateur)
Pr. Abdelouahid LYHYAOUI	ENSATg, Morocco	(Directeur)
Dr. Parisa RASTIN	Ecole des Mines, France	(Examinatrice)
Dr. Nicoleta ROGOVSCHI	Université de Paris, France	(co-encadrante)

"The Future is Private."

Mark Zuckerberg, CEO of Facebook

Abstract

Doctor of Philosophy

Data Anonymisation through Unsupervised Learning

by Sarah ZOUININA

Preserving the utility of anonymized data is one of the biggest limitation to the research field of Privacy Preserving Machine Learning. On the one hand, people claim a maximum level of privacy to protect their personal information from malicious intruders. And on the other hand, researchers, industries and governments demand a higher level of utility in order to develop products that are interesting and suitable to the specific needs of their customers. The research presented in this thesis tackles the privacy-utility trade-off by using unsupervised learning approaches. Firstly, the Multi-view Collaborative Self Organizing Maps as a way to cluster the data locally on each view of the data set, but collaborate by exchanging information about their findings. Secondly, the 1D Kernel Density Estimation, as a way to improve the utility of the anonymized data while respecting the distribution of each feature in the dataset. Lasty, a supervised learning layer using the Weighted Learning Vector Quantization is added in order to enhance the learning of the previously proposed approaches, and give more representative prototypes to pseudo-anonymize the data. The tests were done on more than six different datasets, and the results show an improvement in the accuracy of the models compared to the state of the art MDAV algorithm. The research presented give some interesting ways of using machine learning to achieve privacy preservation through multiview microaggregation

Résumé

Docteur d'Université

Anonymisation des Données par Apprentissage Non Supervisé

par Sarah ZOUININA

Depuis la mise en vigueur du Règlement Général sur la Protection des Données (RGPD), l'intérêt pour la protection et la sécurité des données a évolué. D'une part, les nombreux accidents de fuite de données. D'une autre part, l'évolution exponentielle des utilisateurs des appareils connectés dans le monde entier, ont fait de l'anonymisation des données une nécessité pour la sécurité des individus y figurant. Depuis les années 2000, plusieurs techniques d'anonymisation des données ont été proposées, certaines relèvent de la cryptographie, d'autres des statistiques et certaines se basaient sur la fouille des données. Les travaux présentés dans cette thèse, résument, comparent et développent des méthodes d'anonymisation des données en se basant sur l'Apprentissage Automatique. Les deux premières approches proposent d'utiliser l'apprentissage collaboratif comme un outil d'anonymisation des données. La troisième méthode utilise le clustering par densité des noyaux à une dimension pour protéger les données. La dernière solution proposée, améliore les performances des trois méthodes introduites précédemment en rajoutant une couche d'anonymisation supervisée. Les méthodes sont validées par des mesures d'utilité et de confidentialité.

Ce mémoire est structuré en quatre chapitres de poids relativement équivalents. Après une introduction rapide, le premier chapitre expose le contexte scientifique général de la thèse. Le chapitre deux, trois et quatre présentent les contributions effectives et discutent leur validation expérimentale sur plusieurs jeux de données.

Acknowledgements

Pursuing a PhD was not just another academic milestone, it was a life changing experience full of challenges, breakdowns, small successes, other breakdowns and so many breakthroughs. I have learned a lot and grew scientifically and mentally. All of this wouldn't have been possible without the precious help of my directors, Professor Abdelouahid Lyhyaoui and Professor Younès Bennani and my supervisor Dr. Nicoleta Rogovschi.

Pr. Lyhyaoui, I thank you for accepting to have me among your PhD students and for giving me the opportunity to study abroad, I deeply thank you for your trust and for your encouragements.

Pr. Bennani, I wish to express my sincere appreciation to your values, you have the substance of a genius, you always pushed me to do the right thing even when the road got tough. You made me go further professionally and personally. I thank you for your persistent help and support, it was invaluable.

Nicoleta, working with you was always a pleasure, I would like to thank you for your follow-ups, you have always been there for me no matter how life got busy.

I am also profoundly grateful for the hard work of my co-authors and their contribution to uplift the studies presented in this thesis. Thanks, Dr. Nistor Grozavu, Pr. Seeichi Ozawa, Dr. Basarab Matei, Dr. Issam Falih and Maha Ben-Fares.

I wish to thank the CNRST Morocco, the Ecole Doctorale Galilée and the project ANR Pro-text NR ANR-18-CE23-0024-01 for partially supporting this project financially.

The work presented in this thesis has been critically assessed by an outstanding committee to whom I am more than grateful: Pr.Omar El Baqqali, Pr. Andonie Razvan and Pr. ABdelfettah Sedqui.

I am truly grateful for all my friends and office colleagues with whom I shared many unforgettable moments, Ugo, Kaoutar, Mourad, Juanjo, Yohan, Davide, Enrico, Xin, Doha, Imad, Hasna, Noussayba and Houda. I would also give a special thanks to Fatima Ezzahraa, this last year was very special we went through so many moments of fun and hardships together, thank you for being a good friend.

I would love to thank a special person, Dr. Najlaa Zniber, Najlaa without your precious assistance, none of this could have been possible, I thank you from the bottom of my heart.

I wish to acknowledge the support and great love of my mother, my brother, Aymane; and my brother, Yasser. They kept me going on and this work would not have been possible without their precious input.

Contents

List of Figures

List of Tables

Avant-propos

De nos jours, les données sont collectées par tout objet connecté afin de les traiter, les explorer, les transformer et les apprendre. Afin de collecter les données sans violation de la sécurité des personnes y figurant, certaines règles liées notamment à la vie privée des personnes concernées doivent être respectées. Le processus de la confidentialisation des données s'intitule l'anonymisation des données. L'intérêt porté pour l'anonymisation des données a pour but d'assurer un bon compromis entre le niveau de protection des données et la qualité de cess données. L'anonymisation peut être définie comme: le processus de désidentification des données sensibles tout en préservant leur format et leur type [1] [2], généralement cette procédure est réalisée en masquant un ou plusieurs caractères afin de cacher certains aspects des données étudiées.

L'intérêt pour l'anonymisation des données a été principalement motivé par le désir des gouvernements et des institutions à *ouvrir* leurs données comme preuve de démocratie et de bonnes pratiques. *Open data ou données ouvertes* est un domaine où les données publiées doivent être anonymisées pour toujours avec un taux de réidentification très faible. Aussi, les données doivent-elles garantir une qualité suffisante pour les analyses [3].

Conscients de l'importance de l'équilibre entre confidentialité et utilité, de nombreuses approches ont été introduites pour s'attaquer à ce problème. Les premières approches étaient principalement basées sur de la randomisation qui consiste à ajouter du bruit aux données [4]. Cette technique s'est avérée inefficace car la reconstruction des données était très probable [5].

Le risque de violation de la vie privée des données par la randomisation a été dépassé par l'émergence du k-anonymat [6]. Cette méthode d'anonymisation qui consiste à regrouper les données de façon à produire une base anonymisée à au moins k enregistrements identiques. La création de groupes d'éléments k et leur remplacement par les représentants du groupe permet un bon compromis entre la perte d'informations et le risque potentiel d'identification des données [7].

L'objectif de cette thèse est de développer de nouvelles approches de l'anony-misation des données par apprentissage non supervisé et avec le minimum d'assistance humaine. Nous avons proposé deux nouvelles approches d'anonymisation des données basées sur un apprentissage topologique collaboratif. Ces deux méthodes ont été publiées dans des conférences internationales car elles offraient un bon compromis entre anonymat et utilité. Une autre approche est l'estimation de la densité du noyau orientée par les attributs, qui est une nouvelle méthode d'anonymisation des données utilisant le clustering 1D. Nous proposons également deux mesures de l'utilité des données, l'une utilisant la précision des données et l'autre la distance des Earth Movers. Nous avons également amélioré les trois méthodes d'anonymisation proposées en utilisant l'apprentissage non supervisé en incorporant les informations discriminantes et en ajoutant une nouvelle couche d'anonymisation, ce qui a permis d'obtenir une plus grande précision, ce qui signifie un niveau d'utilité plus élevé.

La structure de ma thèse est la suivante :

Chapitre 1. Revue de la littérature

Ce chapitre commence par une brève revue de la littérature sur l'anonymisation, il retrace l'histoire de l'anonym-isation des données, de la préservation de la vie privée, de l'exploration des données et des mesures d'anonymisation et d'utilité des données.

Chapitre 2. Clustering topologique collaboratif pour l'anonymisation des données

Dans ce chapitre, nous présenterons le Collaborative Topological Clustering pour l'anonymisation des données, nous illustrons la puissance des techniques proposées en utilisant les différents ensembles de données de la base de données UCI [8].

Chapitre 3. Anonymisation des données par densité de noyaux à une dimension

Dans ce chapitre, nous montrerons la troisième approche de l'anonisation que nous présentons ainsi que les mesures de l'utilité de séparation et de l'utilité structurelle qui sont deux méthodes de mesure de l'arbitrage entre vie privée et utilité.

Chapitre 4. L'impact de l'introduction d'informations discriminantes

Dans ce chapitre, nous revisitons les trois approches en ajoutant les informations discriminantes (labels) et en évaluant leur impact sur l'utilité des données anonymisées.

Chapitre 5. Conclusion & Travaux futurs

Dans cette partie, nous donnons les conclusions tirées de la thèse et nous suggérons les éventuels travaux futurs à effectuer.

To the soul of baba ..

Chapter 1

Introduction

Nowadays, data is used in every aspect of the human life. Data is collected by sensors, social networks, mobile applications and connected objects to treat it, explore it, transform it, learn it and learn from it. To have the most of the data collected without security breaching, some rules related especially to the privacy of the people on the dataset have to be respected. The process of preserving data privacy is called data anonymization which is a novice field in data science. Conscious of the pricey analysis provided by good quality data, researchers, studied data anonymization techniques with the purpose of assuring a good tradeoff between identity disclosure and information loss. So what is data anonymization? Data anonymization is *the process of de-identifying sensitive data while preserving its format and data type* [1] [2], generally this procedure is achieved by masking one or multiple characters in order to hide some aspects of the data studied. The growing interest in data anonymization was mainly motivated by the desire of government and institutions to *open* their *data* as a proof of democracy and good practices. *Open data* is a very promising study field and it is very challenging because the data released must be anonymized forever with very low re-identification rate and should ensure sufficient quality for the analytics [3].

1.1 Background

Aware of the importance of the balance between privacy and utility, many approaches were introduced to tackle this problem, the first approaches were mainly based on the randomization method which consists of adding noise to data [4]. This technique was proven to be inefficient since data reconstruction was feasible [5].

The risk of data privacy breach using randomization was overtaken by the emergence of the k- anonymization [6] technique. This group based anonymization method outputs a dataset containing at least k identical records and the anonymization is achieved by firstly removing the key-identifiers like the name and the address and secondly by *generalizing* and/or *suppressing* the pseudo-identifiers which are for example: the date of birth, the ZIP code, the gender and the age. The k value should be chosen in a way to preserve the information provided by the database. The method itself is interesting and was widely studied [9] [10], [11] [12], what gave a strong basis to further works

on anonymization. Since the *k*-anonymity is a group based method, clustering was considered as one of its strongest assets. Creating small groups of *k* elements and replacing the data by the group representatives gives a good trade-off between the information loss and the potential data identification risk [7]. However, the clustering methods presented are based on the *k*- means algorithm which is prone to local optima and may give biased results.

1.2 Motivation

Before the emergence of internet and its expanded use by computers and mobile phones, privacy as term was only referring to the physical existence and information about an individual. The privacy of a person was roughly meaning his home, documents, and his personal life. The traditional concept of privacy has the notion of secrecy and it is basically centered around shielding ourselves and our activities from outsiders. However, The shift that we knew in the 21^st century from the physical to the digital world completely changed the classical way of looking at privacy and thus the old laws are no longer applicable. People are willingly sharing private information to the public, the documents are intangible, conversation can be shared with hundreds of people, ones pictures and videos travel the globe with one click, credit card numbers and social security numbers are provided over the smart phones, opinions are freely discussed behind screens .. and the list goes on and on.

The information are shared over the network because we trust the receiver and we believe that the data won't be disclosed without our explicit permission. In an evolving technology driven society, setting clear boundaries and transparent frameworks for the release, the sharing, and the use of our information is a must. Laws like the General Data Protection Regulation 2016/679 (GDPR) came as an answer to the concerns of people who were worried about the illegal use of their privacy or personal data.

The 21^{st} century has known the biggest Privacy Breaches in human history, billion of records were compromised, their personal life was exposed in some cases and their financial life was at risk. One of the biggest privacy scandals happened on 2014 the *Yahoo* scandal where real names, email addresses, dates of birth and telephone numbers of more than 500 million users were compromised. Another giant data breach was the Target Stores attack where the Credit/debit card information of up to 110 million individuals was disclosed. On the one hand, considering those incidents, data should be anonymized as a preventive measure to protect individuals data.

On the other hand, there are Zettabytes of data out in the world that needs to be mined to provide better understanding of the world's most complex phenomenons, US Census Data for example contains information about every US household, it answers the who, the where; the age, the gender, the race, the income and the educational data, anonymizing it will give better insights into the daily habits of the American households and thus more customised services.

The need to anonymize data and the development of very performing machine learning techniques made us more curious about finding a way to anonymize data using unsupervised machine learning potential.

1.3 Aim and Objectives

The aim of the thesis is to develop new approaches of data anonymization using unsupervised learning automatically, and without extensive human assistance. In order to achieve this aim, we dressed the following outline:

- Test the classical approached proposed in the literature
- Propose an efficient method to pseudo-code data automatically and at once
- Quantify anonymity level
- Measure data utility

1.4 Contributions

In this thesis we proposed two new data anonymization approaches based on collaborative topological learning, both methods were published in international conferences since they gave a good anonymity-utility trade-off. Another approach is the attribute oriented kernel density estimation, which is a new method of anonymizing data using 1D clustering. We also propose two measures of data utility, one using the accuracy of data and the other using the Earth Movers's distance.

We also improved the three proposed anonymization methods using unsupervised learning by incorporating the discriminant information and adding a new layer of anonymization, this resulted in higher accuracy which means higher utility level.

1.5 Thesis Organization

The structure of my thesis is as follows:

Chapter 1. Literature Review

This chapter begins with a brief review of the anonymization literature, it dresses the history of data anonymization, privacy preserving data mining and anonymization - information quantification.

Chapter 2. Collaborative Topological Clustering for Data Anonymization

In this chapter we give details about how we used Collaborative Topological Clustering for data anonymization, we illustrate the power of the proposed techniques using the several datasets of the UCI database [8].

Chapter 3. Attribute-Oriented Data Anonymization

In this chapter we will show the third approch of anonmization that we present plus the separability utility and the structural utility measures that are two methods of measuring the privacy-utility trade-off.

Chapter 4. The impact of introduction of discriminative information

In this chapter we will revise the three approaches by adding the discriminative information (labels) and evaluating its impact on the utility of the data anonymized.

Chapter 5. Conclusion & Future Work

This chapter gives the conclusion part of the thesis. In this part we explain the concluding points and we suggest the possible future work to be done.

Chapter 2

State of the Art

Due to the saturation of cities with smartphones and sensors, the amount of information gathered about each individual is frightening. Humans are becoming walking data factories and third-parties are tempted to use personal data for malicious purposes. To protect individuals from the misuse of their precious information and to enable researchers to learn from data effectively, data anonymization is introduced with the purpose of finding balance between the level of anonymity and the amount of information loss (figure 2.1). The following presents a thorough analysis of state of the art approaches to Privacy Preservation, but first and foremost we should highlight the slight difference between Privacy Preserving Data Publishing which refers to the different frameworks to preserve data privacy and Privacy Preserving Data Mining which focuses on Data mining tools to achieve data privacy.

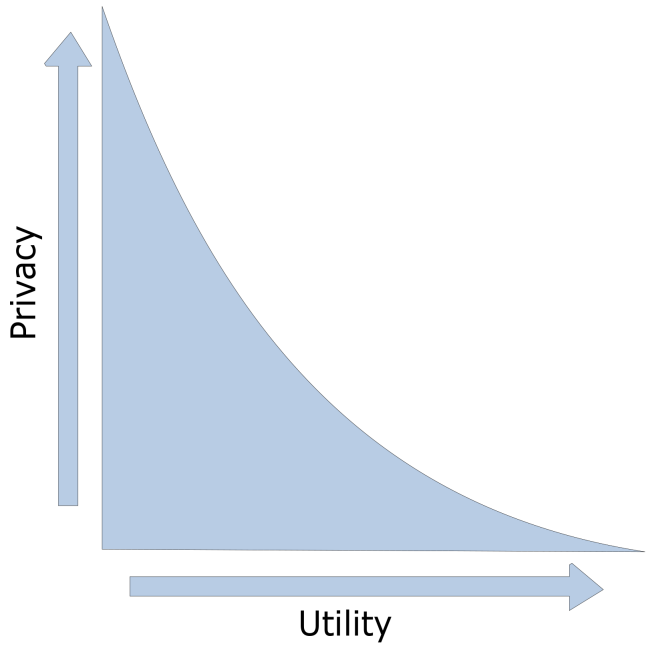
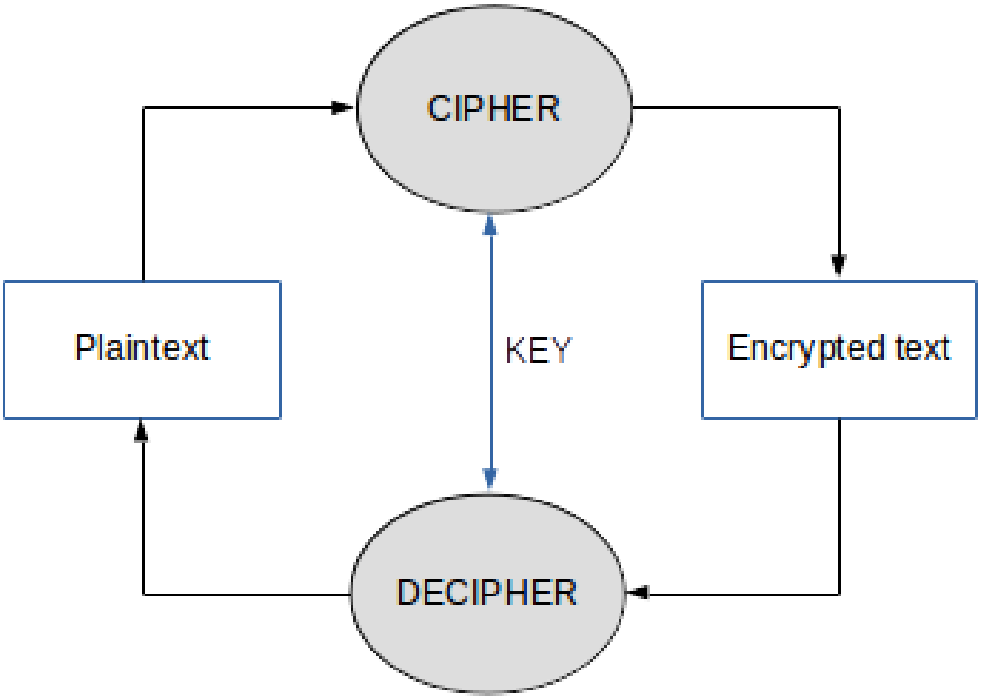


FIGURE 2.1: Tradeoff between the utility and the privacy

FIGURE 2.2: The Crypthography Process: *from plaintext to encrypted text*



2.1 Privacy Preserving Data Publishing

2.1.1 Randomization Methods

The first methods related to data anonymization were mainly based on cryptography or secret writing [13]. This is to say that, plain text is transformed into an encrypted text using a key that helps reconstructing the main information if needed. There are two basic forms of ciphers: transpositions and substitutions. The transpositions rearrange or shuffle the characters in the data and substitutions replace the blocks or characters of substitutes. Cryptography is a field of computer science that was first studied to share secrets [14] and now with emergence of cryptocurrencies and blockchain and the tendency to suppress third parties involvement [15] [16] [17] researches in the field are exploding. Cryptographic methods proved their efficiency on making data secret but the original information cannot be used by a wide public because every study or analyses of the database requires a prior knowledge of the encryption key as shown in figure 2.2.

As shown in the figure 2.2, it proposes to protect data by converting the plain text into a cipher text using a three layers encryption scheme. The three layers are: a Secret, an Authorized and a Public layer. At the first layer, data is encrypted using a digital mark that is accessible to the authorized parties only, in the second layer, the authorized people who are having the encryption key decrypt the data and extract knowledge using data mining

techniques. Lastly, in the public layer, self-information and the conclusions inferred from the data mining processes are published. The only drawback of this technique is that the quasi-sensitive information is encrypted too what limits the usefulness of the data published. [18].

The randomization methods can be described as a way of anonynimizing data by adding noise to elements and/ or multiplying by a constant in a way to hide data's original properties. Techniques of randomization using probabilistic theory were introduced by Liew [19] and used by Agrawal in his paper about privacy preserving data mining [4] which presented a decision tree classifier build over randomized dataset and presented a way to have classification accuracy similar to the accuracy of the original data. The random noise addition is described as follow: Let X be a set of records as $X = \{x_1, x_2, .., x_n\}$ for $x_i \in X$ we add a random noise $y_i \in Y$ drawn from some probabilistic distribution Y and $Y = \{y_1, y_2, .., y_n\}$. The new set of distorted records is denoted $Z = X + Y$. The original data might be approximated using the next formula $X = Z - Y$, the approximation obtained simulates the original data.

Kargupta et al. [20] questioned the utility of additive noise in privacy preservation. Based on the diverse techniques of additive noise filtering existing in signal processing literature, the authors show that additive noise is not well adapted to the privacy preserving problem. Inspired by the fact that multiplicative noise is more suitable for the economic modeling practices, Kim et al. [21] two main multiplicative noise schemes. The first one by generating normal random number with mean 1 and multiply them by the original data. The second is to transform original data by computing the logarithm, calculate the covariance, generate random numbers following a normal probability distribution with mean equals 0 and with a small variance equivalent to c times the first variance, the noise in then added to the data and then take the antilog. Attacks on these approaches were led by Liu et al. [22] who stated that the traditional perturbation methods perturb each data element independently without preserving the data similarity/distance between the elements of the same vector and presented a multidimensional projection in order to reduce the dimensionality of the data. Other perturbation techniques called *data swapping* [23] were also considered as randomization methods and helped preserving privacy while combined to other anonymization procedures [24].

2.1.2 Group Based Methods

Before we tackle this section some concepts must be clarified. As described in *"The Complete Book of Data Anonymization"* [1], the process of multidimensional data anonymization is not automatic and each problem should have its own specific design to help get the most of the data and minimize the risk of security breaching. Anonymization design follows approximatively 12 rules (described extensively by the authors). The main rules to keep in mind are:

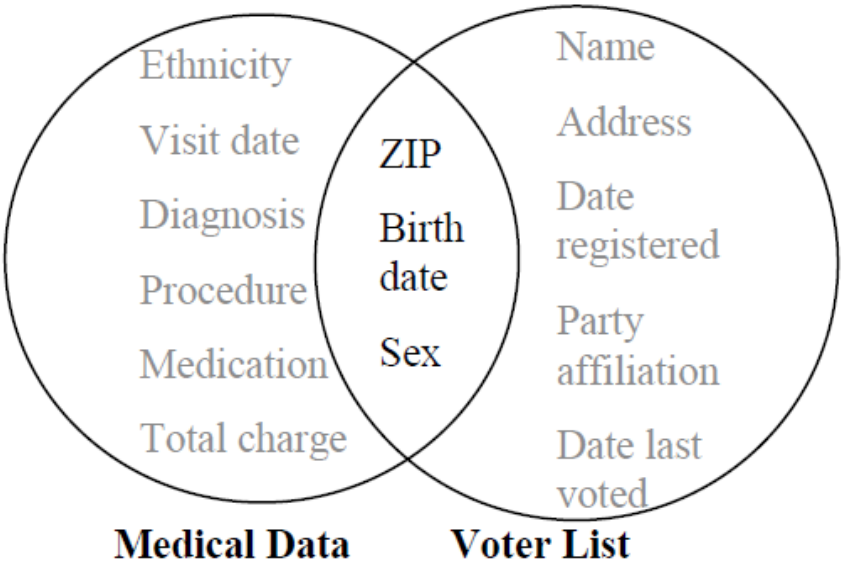
TABLE 2.1: Original data

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	21012	30	French	Hypertension
2	21054	29	French	Hypertension
3	21098	22	American	Obesity
4	21022	19	American	Obesity
5	23027	55	Russian	Cancer
6	23028	60	French	Hypertension
7	23022	45	American	Obesity
8	23022	45	American	Obesity
9	21012	37	Russian	Cancer
10	21021	38	Russian	Cancer
11	21022	34	Russian	Cancer
12	21011	35	Russian	Cancer

- The principle of classification: consists of defining clear boundaries between data types, that is to say (Explicit identifiers, Quasi identifiers, Sensitive data and Non sensitive data).
- The principle of threat modeling: that models the dangers of some environments, users or settings.
- The principle of correlation: which states that anonymized attributes should maintain the correlation between them.
- The principle of randomization: that helps preserving data statistical properties if perturbed randomly.

Defining clear boundaries for the different data types should take in account the properties of each problem and the goals of the further analysis. The first step to preprocess multidimensional data is to distinguish between the *explicit identifiers* which are the attributes responsible of immediately disclosing the identity of the people on the dataset like: name, address, ID number and passport number. The *quasi identifiers* that are generally composed of demographic and geographic information of the owner of the record, this kind of data might be traced if three or more quasi identifiers are combined [6] [3][25]. The third data category that is very important to the anonymization modeling design is *sensitive data*, sensitive data is the information that we want to protect from disclosure in contrary to *non sensitive data* that does not need any special treatment and can be used as it is. In general, explicit identifiers are omitted from the table and the other identifiers are processed to give good data anonymization quality.

FIGURE 2.3: Linking Attack for Data Re-identification



k-anonymization

At first, microdata was made public by only removing the explicit identifiers like the name and the social security number. This was proven to be inefficient by crossing to datasets anonymized in the same manner and the identity of a record owner and his health problem (sensitive data) were made public [6]. This attack was called a linking to re-identify data attack and was achieved by simply crossing the attributes of two anonymized datasets (Medical GIC data and Cambridge voter list) and combining their matching quasi-identifiers (ZIP code, Birth date and Sex) as shown in figure 2.3. In those dataset, six people had the same birth date of the governor of Massachusetts, three of them were man and only one of them has his 5 digit Zip code, the combination of those pseudo-identifiers disclosed the fact that this governor suffered from heart disease. This example brought up the intuition behind *k*-anonymization [26] [27] [6], we say that a database is *k*-anonymous if and only if at least *k* of its records are identical, the table 2.2 an example of 4-anonymous records table.

To achieve *k*-anonymity, techniques of *generalization* and/or *suppression* are used to reduce the granularity of the quasi-identifiers representation [24]. Generalization consists of enlarging the range of membership of the record's value, for example, the Zip Code is generalized by hiding the last number of the code (Zip Code in the table 2.2). In the suppression methods, the record is completely removed like the *Nationality* in the table 2.2 this reduces for sure the risk of identity disclosure if using public data, but it reduces also the quality of the information provided and the accuracy of applications of the transformed data. The first appearance of *k*-anonymity algorithm was

provided in [26] the approach was based on domain and value generalization hierarchies, a minimal generalization was obtained by testing the different generalizations possible and evaluating the levels of anonymity for each combination, the approach enforces a minimal suppression in order to get the required k-anonymization level with less tuple suppression. The problem with k-anonymity approaches is that the majority of the algorithms presented attempt to find a k anonymous table without evaluating its optimality, the works in [9] presented a generalization algorithm called k-optimize that finds an optimal k anonymization under two cost metrics. The experiences presented assume that quasi-identifiers are ordered and each item is given an index, the generalization of each element under a quasi identifying attribute is equivalent to the union of this element with least vale from each attribute domain. The algorithm then constructs a set enumeration tree based on the index ensemble, the tree contains all sub ensembles without repetition. The k-anonymity is evaluated at each node of the tree and a generalization cost is commuted at each step and compared to the previous cost. They also give a heuristic for tree pruning to reduce the computing time, a node is pruned if none of its descendants is optimal.

TABLE 2.2: 4 Anonymous data

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	210**	≤ 30	*	Hypertension
2	210**	≤ 30	*	Hypertension
3	210**	≤ 30	*	Obesity
4	210**	≤ 30	*	Obesity
5	2302*	> 40	*	Cancer
6	2302*	> 40	*	Hypertension
7	2302*	> 40	*	Obesity
8	2302*	> 40	*	Obesity
9	210**	3*	*	Cancer
10	210**	3*	*	Cancer
11	210**	3*	*	Cancer
12	210**	3*	*	Cancer

LeFevre et al. proposed another method to compute the k-minimal generalization with a bottom up aggregation along domain generalization hierarchies [10]. The algorithm creates generalization hierarchies for each subset of the quasi identifiers, the k-anonymity level is evaluated at each node of the hierarchy. The computations are simplified according to a lemma that states that if a node of the hierarchy satisfies the k-anonymity, all generalizations of this node satisfy the k-anonymity. This observation helps prune the tree and minimizes the computational time. Another generalization for k-anonymity technique, *Mondrian*, was also introduced by LeFevre et al. [11], the approach is based on the *multidimensional global recoding* and consisted of

spatially representing data elements and forming groups of k elements using median partitioning. The k-anonymity models presented in *incognito* and *mondrian* are more general than the first model presented by Samarati [24].

l-diversity

k-anonymization model was widely studied and two attacks on its robustness were carried on by [28], the first one is the *homogeneity attack*, as an illustration to this threat let us consider two neighbors Alice and Bob, Bob was once taken to the hospital, Alice knows his Age (37) and his ZIP code (21012), so she wants to discover his disease. After checking some publicly released hospital data as in figure 2.2, Alice realized that there are four identical records with the same attributes element of Bob and they all suffer from Cancer. The sensitive information is therefore revealed. The second one is the *background attack*, let us suppose that Bob wants to know the reason why Alice was hospitalized he knows her ZIP code (21022) and her Age (19), after checking the anonymized table he found two records matching Alice's case with different health problems Hypertension and Obesity, since Bob sees Alice every now and then he knows that she has an obesity problem he then discovers that she was taken to the hospital because of obesity.

Considering those two attacks, an other constraint was added to the k-anonymity optimization problem, this constraint enforces that every sensitive field associated to each equivalent class have to contain at least l distinct values, those values are considered *well-presented*, the approach is presented in table 2.3. The table is 4 anonymous and 3 diverse. In case of multiple sensitive attributes, applying l-diversity becomes challenging due to dimensionality curse.

Although l-diversity is an exploit over k-anonymity it might be considered very limited.

t- closeness

In the works of Li et al. [29] the limits of l-diversity were extensively discussed. Let's take for example a set of data where the sensitive attribute is HIV positiveness and only 1% of the data records owners are HIV positive and the other 99% are HIV negative, the two informations are not on the same level of sensitivity, in this case l diversity is impossible to accomplish because of the data skewness. Also if we had 50% 50% probability of having a disease every equivalent class would have 50% probability of having the disease, and this gives us plenty of free information. Another attack is the similarity attack: an equivalent class might be 3-diverse but the values of the sensitive data in the equivalence class are similar, for example: Disease: gastric ulcer, gastritis, stomach cancer, considering this equivalent class, we realize that the person we are investigating is having stomach problems. To sum up, information in l-diverse tables could be leaked because the approach does not

TABLE 2.3: 3 diverse data

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	2101*	≤ 40	*	Hypertension
4	2101*	≤ 40	*	Obesity
9	2101*	≤ 40	*	Cancer
10	2101*	≤ 40	*	Cancer
5	2302*	> 40	*	Cancer
6	2302*	> 40	*	Hypertension
7	2302*	> 40	*	Obesity
8	2302*	> 40	*	Obesity
2	2105*	≤ 40	*	Hypertension
3	2105*	≤ 40	*	Obesity
11	2105*	≤ 40	*	Cancer
12	2105*	≤ 40	*	Cancer

take in account the skewness of the distribution nor the different levels of sensitivity of sensitive data values.

The t-closeness model [29] follows an interesting thought experiment, we consider B_0 the belief a person had before observing the anonymized data (background knowledge), B_2 is the belief a person gets after investigating the anonymized data and the equivalence classes, as we have seen before l-diversity tries to minimize the distance between the two B_0 and B_2, the t-closeness model adds another intermediate state B_1 that believes that the observer watched the distribution of the sensitive data with QI's generalized to the maximum. The proposed model minimizes the distance between the B_1 and the B_2 states using the *Earth Mover's Distance*. In other words, we say that a data table is t-close if the distance between the whole distribution of sensitive data values and the distribution of data values in an equivalent class is equal to a threshold t. This approach outperforms the other two in case of numeric attributes [24].

2.2 Privacy Preserving Data Mining

In recent years, a question intrigued many researchers is : *How can we develop accurate models about aggregated data without accessing the information contained in the individual record?*.This could be done best by using machine learning's data mining methods. Data mining proved his effectiveness in knowledge discovery by providing some powerful tools related to machine learning. Mining data might breach the privacy of its components, models taking in consideration the power of data mining or using the power of data mining to anonymize data were introduced. In privacy preserving data mining we have two main categories: *Anonymize and mine* or *Mine and anonymize* in this section we will present the existing mine and anonymize methods using

unsupervised learning methods, works in anonymization using supervised learning models are extensively studied in [30] [31] [32] [33] [4] [34].

Before overviewing the PPDM methods we will give a quick reminder of ML categories:

- Supervised Learning: refers to a class of algorithms that teach a predictive model by feeding it input and correct output data (also called labelled data). The most known algorithms under this category are Regression and Classification.
- Unsupervised Learning: refers to the systems that analyzes and detect hidden patterns without knowing the correct answers or the label. It infers subtle relationships between unsorted data and it is mainly used in clustering, Dimensionality reduction and Anomaly detection,
- Reinforcement Learning: refers to this type of learning that determines how software agents ought to take actions in an environment in order to get a reward. These systems are mainly used in Games

Being a fast-expanding field, data mining presents some challenges such as scalability, efficiency, effectiveness and social impacts. The concern in collecting and using sensible data that may compromise privacy is one of those impacts and one that is being extensively researched

2.2.1 Supervised Learning

In order to classify data, we build a model called a classifier that can learn the input data and the response vector (the class or the labels) from a training set and that can identify the class label of unknown data from a testing set. In other words, the process of classification in supervised learning is a three-step approach problem:

1. **The Learning step**, or the training phase: let us consider a class label y for a given attribute vector $x = (x_1, x_2, \ldots, x_n)$. In the learning step, we aim to create a model that defines a function f where $y = f(x)$. The function f maps a tuple of attribute value to the respective class label.
2. **The classification step**, or the testing phase: Once we determine the function f, it can be a decision tree model or a classification rule, we can map any attribute vector x to its corresponding label.
3. **The validation step**: This step comes as a way to evaluate the classifier, we determine its accuracy by calculating the percentage of correct classifications obtained over the testing set.

Nearest neighbor classification with generalization has been investigated by [35]. The main purpose of generalizing exemplars (by merging them into hyper-rectangles) is to improve speed and accuracy as well as inducing classification rules, but not to handle anonymized data. Martin proposes building non-overlapping, non-nested generalized exemplars in order to induce

high accuracy. Zhang et al. discuss methods for building naive Bayes and decision tree classifiers over partially specified data. Partially specified records are defined as those that exhibit nonleaf values in the taxonomy trees of one or more attributes [36]. Therefore generalized records of anonymous data can be modeled as partially specified data. In their approach classifiers are built on a mixture of partially and fully specified data. Inan et al.[37] address the problem of classification over anonymized data. They proposed an approach that models generalized attributes of anonymized data as uncertain information, where each generalized value of an anonymized record is accompanied by statistics collected from records in the same equivalence class. They do not assume any probability distribution over the data. Instead, they propose collecting all necessary statistics during anonymization and releasing these together with the anonymized data. They show that releasing such statistics does not violate anonymity.

2.2.2 Unsupervised Learning

Clustering, or cluster analysis, is a process of grouping sets of object in groups without a prior knowledge of the corresponding classes. The objects belonging to the same cluster should be more similar than the objects from different clusters. Each cluster or group is considered as a class with no label and the process of clustering is often referred to as automatic classification.

The clustering problem is an unsupervised learning paradigm, it is used to detect patterns hidden in data and may reveal interesting insights about it. Those algorithms are commonly known to use similarity metrics that differ from an algorithm to another due to the following properties:

- **Partitioning criteria:** may induce the notion of hierarchy, it addresses the issue of if the clusters are on the same level or if one contains other ones.
- **Separation:** or the overlapping of the clusters. In the overlapping case, objects may belong to multiple clusters, whereas in the non-overlapping, clusters are well separated i.e. mutually exclusive.
- **Similarity measure:** the measure of similarity whether it is distance-based like the *k-means* clustering or connectivity-based like the hierarchical clustering.
- **Clustering space:** relates to the subspace clustering, whether the clusters may be searched within the entire data space (computationally heavy for large data)or within data subspaces (subspace clustering), where irrelevant attributes are suppressed by reducing dimensionality.

Many clustering algorithms have been introduced in order to achieve data anonymity, some of the most interesting are listed in the following:

1. ***k*-anonymization by clustering in attribute hierarchies:** proposed by li et al. [38], this algorithm measures the data distortion caused by

generalization using a *weighted hierarchical distance* calculated following the domain generalization hierarchies. The algorithm presented forms equivalence classes from the database by finding an equivalence class with record's size minus than k, it measures the distance between the found equivalence class and the other equivalence classes and fuses it with the nearest equivalence class to form a cluster of at least k element with minimal information distortion. This method gives good computational results but its very time consuming.

2. ***k*-member:** The main idea behind k-member clustering algorithm is to form clusters of at least k records in a way the information in each cluster is similar to the information in the other cluster. The approach presented [7], fixes the value of k, looks for the record and the cluster with the minimal information loss, add the record to the cluster and iterate until getting clusters with at least k members. Compared to *Mondrian* [11] the information loss is minimized but the execution time is huge.

3. **Clustering based greedy algorithm:** In privacy preserving literature, there is a dilemma between the utility of the data provided and the security of this data using k-anonymization. To capture the usefulness and protect the privacy of datasets, Loukides et al. [39] introduced measures taking in account the attribute and the tuples diversity and a clustering algorithm that is similar to the previous k-member clustering algorithms [7] but with the constraint of maximizing the dissimilarity of sensitive data values (privacy) and minimizing the similarity of the quasi-identifiers (usefulness).

4. **One pass *k*-means:** [40] is different than the k member and the greedy algorithm in two manners,It constructs all the clusters simultaneously and it is more resilient to outliers. This algorithm minimizes the information loss via a distance measure.

2.2.3 Microaggregation

Microaggregation is a technique for disclosure limitation aimed at protecting the privacy of data subjects in microdata releases. It has been used as an alternative to generalization and suppression to generate k-anonymous data sets, where the identity of each subject is hidden within a group of k subjects. Unlike generalization, microaggregation perturbs the data and this additional masking freedom allows improving data utility in several ways, such as increasing data granularity, reducing the impact of outliers and avoiding discretization of numerical data [41] microaggregation. Rather than publishing an original variable V_i for a given record, the average of the values of the group over which the record belongs is published. In order to minimize information loss, the groups should be as homogeneous as possible. The impact of microaggregation on the utility of anonymized data is quantified as the resulting accuracy of a machine learning model trained on a portion of microaggregated data and tested on the original data [42]. Microaggregation is measured in terms of *syntactic distortion*.

Achieving microaggregation might be done using machine learning models, like *clustering* and/or *classification*. LeFevre et al. [43] propose several algorithms for generating an anonymous data set that can be used effectively over pre-defined workloads. Workload characteristics taken into account by those algorithms include selection, projection, classification and regression. Additionally, LeFevre et al. consider cases in which the anonymized data recipient wants to build models over multiple different attributes. Microaggregation is a perturbative method that can be formulated mathematically as an optimization problem, where the goal is to find the clusters that minimize the global error. Each cluster is represented by a characteristic function χ_i where:

$$\begin{cases} \chi_i(x) = 1 \quad if \text{ x is assigned to the i}^{\text{th}} \text{ cluster} \\ \chi_i(x) = 0 \quad if \ not \end{cases}$$

k is the minimum number of elements to satisfy the privacy requirements, p_i are the clusters centers and d is a distance between the clusters prototypes and the records. The optimization problem minimizes the global error under three constraints. The optimization problem is written as follows:

$$\begin{aligned} \min SSE = & \sum_{i}^{c} \sum_{x \in X} \chi_i(x)(d(x, p_i))^2 \\ \text{s.t.} \quad & \sum_{i=1}^{c} \chi_i(x) = 1 \quad \forall x \in X \\ & 2k \geq \sum_{x \in X} \chi_i(x) \geq k \\ & \chi_i(x) \in \{0, 1\} \end{aligned}$$

One if the first introduced microaggregation algorithms is the MDAV algorithm [44], the algorithm is described in algorithm 1. The MDAV algorithm is criticized as an microaggregation algorithm that lacks flexibility to adapt the groups size to the distribution of the records what results in poor homogeneity of the microaggregated groups.

Algorithm 1 Maximum Distance to Average Vector- Generic MDAV

Inputs: D: dataset, k: integer
While $|D| \geq 3k$**:**

1. Compute the average record x of all records in D. The average record is computed attribute-wise.
2. Consider the most distant record x_r to the average record x using an appropriate distance.
3. Find the most distant record x_s from the record xr considered in the previous step
4. Form two clusters around x_r and x_s, respectively. One cluster contains x_r and the $k-1$ records closest to x_r. The other cluster contains x_s and the $k-1$ records closest to x_s.
5. Take as a new dataset D the previous dataset D minus the clusters formed around x_r and x_s in the last instance of Step 1d.

end While
if 3k-1 <$|D| \geq 2k$**:**

1. compute the average record x of the remaining records in D
2. find the most distant record x_r from x
3. form a cluster containing x_r and the $k-1$ records closest to x_r
4. form another cluster containing the rest of records.

else if (less than 2k records in D)

1. form a new cluster with the remaining records.

2.2.4 Comparison between the different privacy techniques

The table 2.4 shows a detailed comparison between the techniques used for data anonymization and their challenges

2.3 Privacy Metrics

Retaining information from an anonymized dataset while preserving its privacy is the most challenging part about data anonymization. Studying the different metrics used to quantify data anonymity vs utility is the purpose of this section, each of the proposed methods above has its own privacy preservation measure what makes dressing a general framework to the privacy preservation metrics difficult, especially that in the data anonymization field the use of heuristics is mostly frequent.

Techniques	Challenges
Slicing	• The attributes are grouped randomly which is not efficient • It's not clear how attribute disclosure is preserved • Utility of data is lost because of fake tuples
Cryptographic technique	• Difficult to apply for large databases • Difficult to scale when more events are involved • Non-sensitive data is also encrypted that can be useful for analytics
Differential privacy	• High computation complexity • No preservation of data truthfulness at the record level
K-anonymity	• Gives no consideration of the links between sensitive data • Not able to protect against attacks based on background knowledge • Not applicable for high-dimensional data
Generalization	• Causes loss of information • Not ready to protect attribute correlations • Each attribute is generalized separately • To climb up the hierarchy, each iteration needs to recognize the best generalization • Not applicable for large datasets
Top-down specialization	• Loss of privacy leads to its inadequacy in handling large-scale data sets

TABLE 2.4: Privacy techniques and challenges

2.3.1 Fung's Categorization

The question that was brought so many times is: how to evaluate anonymous data quality with respect to the original data quality? the question brings up another crucial question which is for what purpose this data is anonymized? we can classify the data anonymity metrics into 3 categories [45]: General purpose anonymization, Special purpose anonymization and Trade-off purpose anonymization.

General purpose anonymization The quantification of the anonymization level depends on the usage of the anonymous data, for general purpose anonymization the analysis that the data recipients will do to is completely unknown to the data provider. Since the methods used should provide a

general anonymization, the amount of information distortion is quantified by some 'similarity' measure between the anonymized and the original data.

1. **Minimal Distortion:** In the minimal distortion metric or MD, a penalty is charged to each instance of a value generalized or suppressed, let's consider the attribute *Nationality* in table 2.1, after anonymization, as shown in table 2.2 the attribute was completely suppressed so the minimal distortion will be 12 units of distortion since 12 elements were suppressed.

2. **Information Loss:** In order to quantify the information loss caused by a dataset to be hidden, the statistics computed on the perturbed dataset should not differ significantly from the ones obtained on original data, statistics might be achieved as detailed in [46] over:

   - Means and covariances on a small set of subdomains
   - Marginal values for tabulations of the data
   - At least one distributional characteristic.

   Computing information loss for microdata over **continuous data**, on the one hand, under an SDC framework might be using: Mean Square Error (MSE), Mean Absolute Error (MAE), Mean Variation (MV) : between covariance matrices, correlation matrices, correlation matrices between the p vars and the p factors obtained after PCA and factor score matrices. For **categorical data**, on the other hand, by determining the following:

   - Direct comparison of categorical values: by defining some kind of distance
   - Comparison of contingency tables (distance [41])
   - Entropy based measures [47].

   Captures the information loss of generalizing a specific value to a general value The first measure computes the information loss [6][26] based on the taxonomy tree designed for the model.

$$IL(e) = |e|.(\sum_i \frac{maxN_i - minN_i}{|N_i|} + \sum_j \frac{H(\wedge(\cup_{C_j}))}{H(T_{C_j})})$$

$$Total_{IL} = \sum_e IL(e)$$

   Where e refers to the equivalent class e, min_{N_i} and max_{N_i} are the minimum and maximum value of e, N_i to design the numerical values. $\wedge(\cup_{C_j})$ is the subtree of the inferior common incest of the tree, $H(T)$ refers to the size of the tree.

3. **Discernibility Metric:** The Discernibility Metric attempts to straightforwardly capture the desire to maintain discernibility between tuples. The metric attributes a penalty to each tuple based on the number of indistinguishable tuples in the transformed dataset. Each unsuppressed tuple is penalised by $|E|$. If a tuple is suppressed, then it is assigned a penalty of $|D|$, the size of the input dataset. This penalty reflects the fact that a suppressed tuple cannot be distinguished from any other tuple in the dataset. The metric can be mathematically stated as follows:

$$C_{DM}(g,k) = \sum_{\forall E \ st \ |E| \geq k} |E|^2 + \sum_{\forall E \ st \ |E| < k} |D||E|$$

As explained above, the metric tackles the notion of loss by charging a penalty to each record for being indistinguishable from other records with respect to the quasi identifiers.

Special purpose anonymization When the purpose of the data is known at the time of publication, it can be taken in account during the anonymization process. If so, why don't we just publish the result of the anonymization instead of the anonymized data, simply, at an algorithmic level, it is extremely committing to publish only the result of an algorithm and it is not practical for further studies.

1. **Classification Metric:** The second measure was crafted to optimize a k-anonymous dataset for training a classifier [48] [49]. This metric can be applied when tuples are assigned a categorical class label in an effort to produce anonymizations whose induced equivalence classes consist of tuples that are uniform with respect to the class label. This classification metric assigns no penalty to an unsuppressed tuple if it belongs to the majority class within its induced equivalence class. All other tuples are penalized a value of 1.

$$CM(T) = \frac{\sum_{allrows} penalty(row\ r)}{N}$$

Where N is the number of rows in a set. A row is penalized i.e $penalty(r) = 1$ if it is suppressed or if its class label is not the majority class label of its group, else it is equal to 0. A row r is penalized if it is suppressed or if its class label class(r) is not the majority class label.

$$Penalty = \begin{cases} 1 & \text{if } r \text{ is suppressed} \\ 1 & \text{if } class(r) \neq majority(G(r)) \\ 0 & \text{otherwise} \end{cases}$$

The *Classification Metric* penalizes impure groups that contain rows with different labels.

2. **Trade-off purpose anonymization:** Most privacy-preserving data mining methods apply a transformation which reduces the effectiveness of the underlying data when it is applied to data mining methods or algorithms. In fact, there is a natural trade-off between privacy and accuracy, though this trade-off is affected by the particular algorithm which is used for privacy preservation. The catch is if data anonymity is maximized, its utility is minimized and vice versa which renders the task of privacy preservation even harder. [45] proposed a measure to evaluate the trade-off between anonymity and utility

$$Score = \frac{InfoGain(v)}{AnonymLoss(v) + 1}$$

Where $InfoGain$ is the difference between entropy of classes before and after anonymization and $AnonymLoss$ is the difference between the anonymity level before and after the treatment. Preventing identity disclosure and incorporating privacy preserving techniques with unsupervised machine learning methods is the aim of the study we re conducting.

Trade-off Metrics The idea of trade-off metrics is to consider both the privacy and information requirements at every anonymization operation and to determine an optimal trade-off between the two requirements

2.3.2 Wagner's Categorization

The categorization of Fung in [45] is very interesting since it takes into account the purpose of the data privacy method but it is not complete. In their survey, published in 2018, Wagner et al. [50], studied the proposed anonymity metrics extensively, depending on the privacy domain, the adversary goals and capabilities and the data sources. They named over 70 privacy metric, in our case, we will focus only on the output measures, since they are the most accurate to our research interests. for reasons of simplicity we are going to follow the same categorization they proposed, since it is the most systematic and state of the art.

In the data privacy preserving literature we mean by a privacy metric each measure that describes in some way the level of privacy. It is mathematically convenient to qualify these measures as metrics even though they don't fulfill all the conditions of a mathematically approved metric which are (non-negativity, identity of indiscernibles, symmetry, and triangle inequality).

There are many cases when discussing a privacy metric, on the one hand, Andersson and Lundin [51] require that privacy metrics should be based on probabilities (e.g., the probability of an adversary identifying a given individual), they also argue that a privacy metric should reflect how evenly spread the guesses of an adversary are and how many individuals are completely indistinguishable.

On the other hand, Syverson [52] requires that privacy metrics should reflect how difficult it is for an adversary to succeed, undependably on the attributes of the dataset, he believes that those metrics refelect the resources needed for successful attacks on privacy instead of relying on cardinalities or probabilities. Bertino et al. [48] require that privacy metrics indicate the privacy level, the portion of sensitive data that is not hidden, and the data quality after application of the anonymity mechanism. In the next subsection we will introduce a taxonomy with eight properties each of which describing a different aspect of the privacy. It is very important to note that a single metric cannot capture the entire concept of privacy. We argue that is necessary to evaluate privacy using different metrics since the concept might overlap.

Uncertainty: Uncertainty metrics assume that high uncertainty in the adversary's estimate correlates with high privacy, because the adversary cannot base his guesses on information known with certainty. However, even guesses based on uncertain information can be correct, and thus individual users may suffer privacy losses even in scenarios with a highly uncertain adversary.

1. **Anonymity Set Size**: It can be denoted AS_u, it is the set of the users indistinguishable from u, the measure is as follows:

$$priv_{ASS} = |AS_u|$$

2. **Entropy:** Shanon's entropy is used in the context of privacy as the effective size of the anonymity set, in other words, it is the number of bits of additional information the adversary needs to identify a user. Let us consider an adversary who wants to identify which member of the anonymity set took a specific action The adversary would then estimate a probability $p(x)$ for each member x of the anonymity set AS_u which indicates the likelihood that x is the targeted user u, generally, each value $x_1, ..., x_n$ of the discrete random variable X represents a member of the anonymity set and $p(x_i)$ is the (estimated) probability of this member to be the target. The entropy of X is then:

$$priv_{ENT} \equiv H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

3. **Normalized Entropy (Degree of Anonymity):** Because the value range of entropy depends on the number of elements in the anonymity set, the absolute value cannot always be used to compare entropy values. This is why entropy is frequently normalized using Hartley entropy:

$$priv_{NE} \equiv \frac{H(X)}{H_0(X)}$$

4. **Cross-entropy /Likelihood:** Cross-entropy is derived from entropy, and the relative entropy D_{KL}, which indicates the distance between two

probability distributions Generally, cross-entropy measures the amount of information needed to identify an object in the data set if the original data are coded in terms of the model's distribution X, rather than their true distribution X^*.

$$priv_{CE} \equiv H(X^*) + D_{KL}(X^*||X)$$

Information Gain or Loss: Metrics that measure information gain or loss quantify the amount of information gained by the adversary, or the amount of privacy lost by users due to the disclosure of information.

1. **Relative Entropy, Kullback-Leibler divergence D_{KL}:** measures the distance between two probability distributions. In our case, the two distributions are the true distribution X^* and the adversary's estimate X

$$priv_{RLE} \equiv D_{KL}(X^*||X) = \sum_{x,x^*} p(x^*) \log_2 \frac{p(x^*)}{q(x)}$$

2. **Mutual Information:** quantifies how much information is shared between two random variables.It can be computed as the difference between entropy and conditional entropy. It is computed between the true distribution of data X^* and the adversary's (obfuscated) observations Y

$$priv_{MI} \equiv I(X^*;Y) = H(X^*) - H(X^*|Y) = \sum_{x^* \in X^*} \sum_{y \in Y} p(x^*, y) \log_2 \frac{p(x^*, y)}{p(x^*)p(y)}$$

3. **Pearson's Correlation Coefficient:** measures the degree of linear dependence between two random variables

$$priv_{PCC} \equiv \frac{cov(X^*, Y)}{\sigma_{X^*}.\sigma_Y}$$

Data Similarity: Data similarity metrics measure similarity either within a dataset, for example by forming equivalence classes, or between two sets of data, for example between a private dataset and its public, sanitized counterpart. These metrics abstract away from an adversary and focus on the properties of the data. For example, similarity can refer to the frequencies of data values, numerical similarity, or the (lack of) variation in published data.

1. **k-Anonymity:** is conceptually similar to the size of the anonymity set in the uncertainty property

$$priv_{KA} \equiv k, \quad where \quad \forall E : |E| \geq k$$

2. **l-Diversity:** the l-diversity principle requires l distinct values in each equivalence class. $H(S_E)$ is the entropy of the sensitive attribute frequencies.

$$priv_{LE} \equiv l, \quad where \quad \forall E : |H(S_E)| \geq \log(l)$$

3. **t-Closeness:** t-closeness modifies k-anonymity to bound the distribution of sensitive values. It states that the distribution SE of sensitive values in any equivalence class E must be close to their distribution S in the overall table. In particular, the distance between distributions $d(S, S_E)$, measured using the Earth Mover Distance metric, must be smaller than a threshold t.

$$priv_{TC} \equiv t, \quad where \quad \forall E : |d(S, S_E)| \leq t$$

4. **Normalized Variance:** measures the dispersion between the original data X^* and perturbed data Y

$$priv_{VAR} \equiv \frac{\sigma^2(X^* - Y)}{\sigma^2(X^*)}$$

Indistinguishability: Metrics based on indistinguishability, a classic notion in the security community, analyze whether the adversary is able to distinguish between two outcomes of a privacy mechanism. Privacy is high if the adversary cannot distinguish between any pair of outcomes. Metrics in this category are usually binary; they indicate whether two outcomes are indistinguishable or not, but do not quantify the privacy levels in-between.

1. **Cryptographic Games:** A challenge-response game, is set up in which the adversary selects the inputs for a protocol and is given the output and two alternative outcomes y_1 and y_2 after the protocol has been executed. The adversary then has to make an estimate, x, indicating whether y_1 or y_2 is the correct outcome x^*. The adversary has an advantage if they can do this with a probability that is non-negligibly greater than 0.5

$$priv_{VAR} \equiv \begin{cases} 1 & if \quad p(x = x^*) \leq \frac{1}{2} + \epsilon(k) \\ 0 & otherwise \end{cases}$$

2. **Differential Privacy:** Formally, differential privacy is defined using two data sets D_1 and D_2 that differ in at most a single row, in other words, the Hamming distance between the two data sets is at most 1. The level of privacy is measured by the ϵ, two queries differ at most with $\exp(\epsilon)$

Adversary's Success Probability: Metrics using the adversary's success probability to quantify privacy indicate how likely it is for the adversary to succeed in any one attempt, or how often they would succeed in a large number of attempts. Low success probabilities correlate with high privacy. While this assumption holds for an averaged population of users, an individual user may still suffer a loss of privacy even when the adversary's success probability is low.

1. **Adversary's Success Rate:** This metric measures the probability that the adversary is successful, or the percentage of successes in a large

number of attempts, the adversary is successful when he can find a record s' that is similar to the target record s with a similarity threshold of τ_s and an error threshold of τ_e

$$priv_{SRD} \equiv p(Sim(s, s') \geq \tau_s) \geq \tau_e$$

2. **Degrees of Anonymity:** In [53], they defined six degrees of anonymity for communications systems, even though this privacy metric is mainly related to the communications systems field we wanted to include it since it represents a basis to the privacy measurements. Let us consider $p(x)$ as the adversary's probability to identify the sender of a message the degrees are as follow:

$$priv_{DOA} \equiv \begin{cases} \text{absolute privacy}, & if\ p(x) = 0 \\ \text{beyond suspicion}, & if\ p(x) = \frac{1}{|X|} \\ \text{probable innocence}, & if\ p(x) \leq 0.5 \\ \text{possible innocence}, & if\ p(x) < 1 - \delta \\ \text{exposed}, & if\ p(x) \geq \tau \\ \text{provably exposed}, & if\ p(x) = 1 \end{cases}$$

Error: Error-based metrics measure how correct the adversary's estimate is, for example using the distance between the true outcome and the estimate. High correctness and small errors correlate with low privacy.

1. **Mean Squared Error:** The mean squared error describes the error between observations y by the adversary and the true outcome x^*.

$$priv_{MSE} \equiv \frac{1}{|X^*|} \sum_{x \in X^*} ||x^* - y||^2$$

Time: Time-based metrics either measure the time until the adversary's success, or the time until the adversary's confusion. In the first case, metrics assume that the adversary will succeed eventually, and so a longer time correlates with higher privacy. In the second case, metrics assume that the privacy mechanism will eventually confuse the adversary, and so a shorter time correlates with higher privacy.

1. **Time until Adversary's Success:** It assumes that the adversary will succeed eventually, and is therefore an example of a pessimistic metric. This metric relies on a definition of success, and varies depending on how success is defined in a scenario.

Accuracy or Precision: These metrics quantify how precise the adversary's estimate is without considering the estimate's correctness. More precise estimates correlate with lower privacy.

1. **Confidence Interval Width:** According to the confidence interval width, the amount of privacy at τ% confidence is given by the width of the confidence interval for the adversary's estimate $x \in [x_2, x_1]$ in which the true outcome x^* lies.

$$priv_{CIW} \equiv |x_2 - x_1| \quad where \quad p(x_1 \leq x < x_2) = \tau/100$$

2.4 Conclusion

In this chapter we reviewed the difference between Privacy Preserving Data Publishing (PPDP) and the Privacy Preserving Data Mining (PPDM), we dressed the different types of those two big categories and we introduced a literature review of the privacy metrics. The main goal of the following chapters is to introduce models that produce a protected output by using machine learning models.

Chapter 3

Collaborative Topological Clustering for Data Anonymization

In this chapter we use the topological structure of the Self Organizing Maps [54] and their ability to prone less to local optimas [55] to achieve anonymity with minimum information loss. We will use the SOM clustering model as it was proven to give good results on practical applications when the aim is to visualize and perform dimensionality reduction. The results of the clustering are enhanced using the collaborative learning process [56]. At the end of the topological learning, the "similar" data will be collected in clusters, corresponding to the sets of similar patterns. These clusters can be represented by a more concise information, such as their gravity center or different statistical moments since we believe that this information is easier to manipulate than the original one.

The two approaches we are presenting in the following, consists of anonymizing tabular data using multi-view topological collaborative clustering [57].

3.1 Related Works

3.1.1 Prototype based models in unsupervised learning

Unsupervised learning schemes are applied when the classes are unknown a priori, it aims to represent the vectorial data by typical representatives commonly called prototypes. The unsupervised analysis might be conducted to extract reveal the structure of data, pre-process large datasets for further analysis or to reduce its dimensionality and allow its vizualisation. Machine learning models that represent observations using prototypes falls into the category of Prototype-based models in machine learning.

Those models use two major appealing concepts: on the one hand, they use concepts of Hebbian Learning which refers to Donald O. Hebb who explained the biological neural weight adjustment mechanism. His work set the basic principles for the machine learning community since it describes how to convert a neuron inability to learn and enables it to develop cognition as a response to an external stimulation [58]. This is considered very intuitive since it mirrors the cognitive behaviour of the human being. On the other hand, it compares the observations with a reference set of prototypes using a distance or a similarity measure.

In brief, prototype based models in unsupervised learning are models of machine learning that accomplish unsupervised analysis by providing a clusters' representation using prototypes. In fact, a very basic scheme of unsupervised learning is the Vector Quantization (VQ) [59], a classical signal approximation method that forms a quantized approximation to the distribution of the input data vector $x \in \mathbb{R}^n$ where $i = 1, 2, .., k$ in a finite number of "codebook" vectors $w_i \in \mathbb{R}^n$, where $i = 1, 2, .., k$.

This approximation means to find the prototype w_c closest to $x \in \mathbb{R}^n$ using the eluclidean distance.

$$\|x - w_c\| = min\{\|x - w_i\|\}$$

or

$$c = argmin_i\{\|x - w_i\|\}$$

Hence, the aim of the VQ is formulated in terms of the cost function that guides the computation of the prototypes vectors i.e. the learning phase. Competitive learning is a type of VQ that assigns each datum to its closest prototype, called **'winner'**, the closeness is set in terms of a pre-defined distance measure, this scheme where the prototypes compete for updates is named the *winner takes all* (WTA) scheme. The idea behind competitive learning also comes from the signal processing literature. Firstly, we have several parallel "filters" that are initially tuned differently and possess pattern normalizing properties for the same input. Secondly, one of the filters randomly becomes the **'winner'** according to an input vector. Lastly, the so called **'winners'** i.e. best-matching filters suppress the other cells and remain the only activated filters, those filters and their neighbors will be updated during the learning. The competitive learning partitions the signal space in a way that the neighboring filters of the array get an equitable representation of the signal domain.

Competitive VQ corresponds to a stochastic gradient descent [60] where the convergence of the prototype vectors is guaranteed by employing a time-dependent learning rate that slowly approaches zero in the course of training. In spite of VQ's ability to provide prototypes, one should not confuse it with clustering. It is very important to highlight that VQ does not necessarily identifies the existing clusters within data. In the figure 3.1 (a) displays a single cluster with three prototypes, (b) shows two nearly overlapping clusters with a prototype each, (c) represents two separated clusters and two prototypes that represent neithe of the clusters and finally in (d) we can see two clusters where one is non identified since it is very small and does not contribute to the quantization. The main difficulty with competitive VQ is that the function might display many suboptimal local minima. For this reason, this can influence the initial prototype positions during the training. To illustrate this using an extreme example, we can imagine putting a prototype in an empty region, this will prevent it to be identified as a winner for any of those data points what will lead it to be considered as a *dead unit*. Maybe, the most popular machine learning model based on prototypes is the Self-organizing Map (SOM).

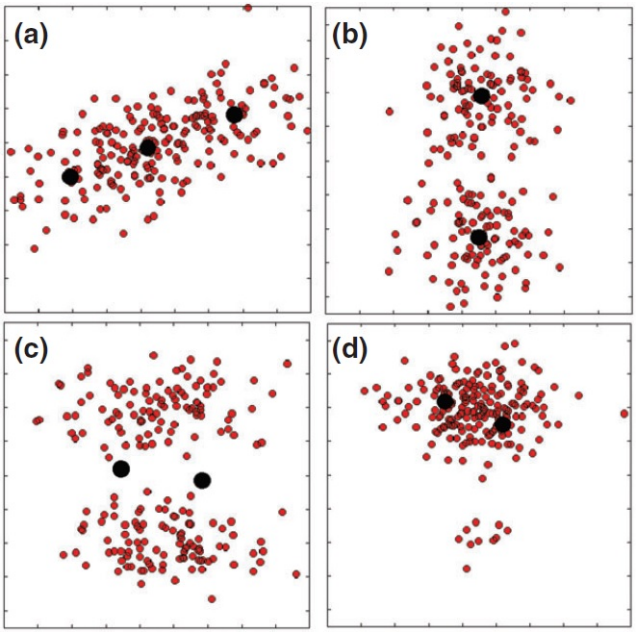


FIGURE 3.1: Representation of two-dimensional data points by prototypes. In each panel, 200 data points are displayed as red dots, and prototype positions marked by the black circles (figure credits : [60])

3.1.2 Self Organizing Maps

The SOM is a widely applicable algorithm that mimics the biological neurons, its aim is mainly to find a faithful topology-preserving representation of a given set of data. A SOM is characterized by a number of neurons that react to stimuli from the environment [60]. Their principal goal is to transform an incoming signal pattern of arbitrary dimension into a one or two-dimensional discrete map sometimes called lattice [61]. Consequently, Kohonen's SOMs [62],[59] allow for data visualization and compression. Together with posterior labeling or other post-processing techniques, the SOM can also be employed in classification or regression tasks with a low computational cost. Therefore, they might be seen as a k-means algorithm with topological constraints and better performance.

To form the SOM, the algorithm firstly, initializes a set of synaptic weights W, where for each neuron k, $w^{(k)} = [(w_1^{(k)}, w_2^{(k)}, \ldots, w_i^{(k)}, \ldots, w_n^{(k)}]^T$ where $k = 1, \ldots, C$ in the network by assigning them randomly generated small values. Once initialized we proceed by the further steps.

Competition For each input pattern (vector) $\boldsymbol{x} = [x_1, x_2, \ldots, x_j, \ldots, x_n]^T$, where n is the dimension of the input space. The neurons in the network compute their respective values of a discriminant function. This discriminant function minimizes the euclidean distance between the input learning set and the weight vector for each neuron in order to determine the winning

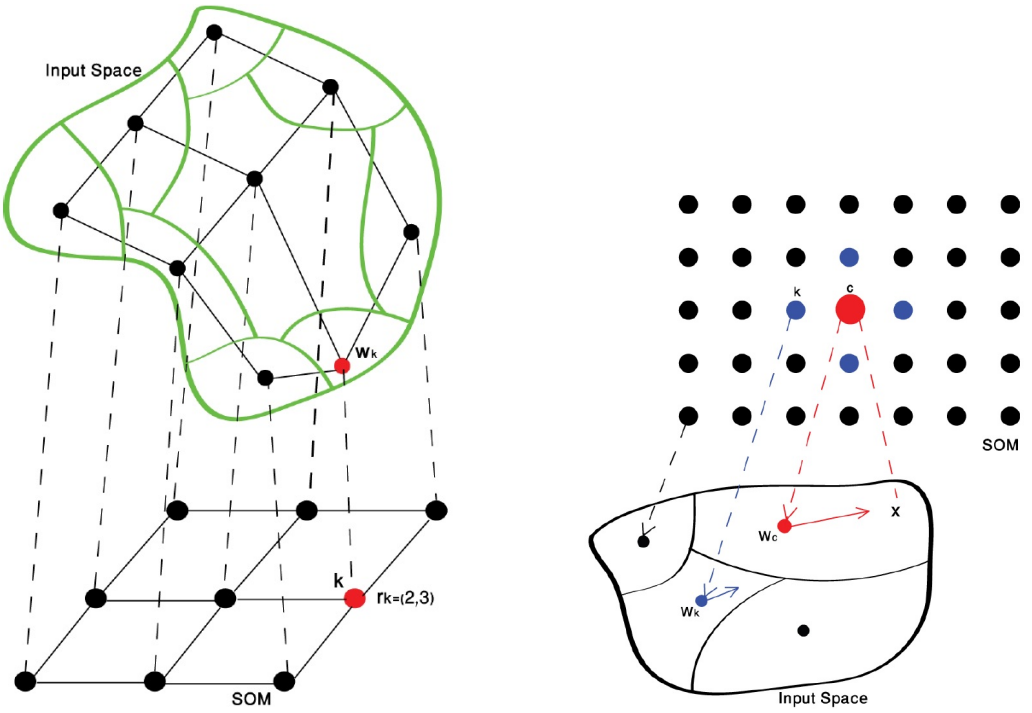


FIGURE 3.2: Architecture of SOM: Mapping the data space to the self-organizing map (left). SOM prototypes update (right). [63]

neuron as in equation 3.1.

$$c(\boldsymbol{x}) = argmin_k\{\left\|x - w^{(k)}\right\|\}, \quad k = 1, \ldots, C \tag{3.1}$$

where $c(\boldsymbol{x})$ refers to the index of the best matching unit, i.e. the winning neuron which is simply its position in the lattice.

Cooperation The winning neuron determines the spatial location of a topological neighborhood of excited neurons, thus it locates the center of a topological neighborhood of cooperating neurons. As commonly known from the neurobiological theory, *neurons that fire together, wire together*, therefore, the winning neuron tends to excite the immediate units in his neighborhood according to a neighborhood function that should satisfy two main requirements:

1. The function should attain her maximum and be similar at the position of the winning neuron what means that $d_{ji} = 0$.

2. The effect of learning should be proportional to the distance a node has from the winning unit and the amplitude of the neighborhood K_{ji} should shrink over time. For this reason a Gaussian is a good choice for a neighborhood function since it is translation invariant (i.e. independent of the location of winning neuron i) as shown in equation 3.2.

$$K_{j,i(x)} = \exp\left(\frac{-d^2_{(j,i)}}{2\sigma(t)^2}\right) \tag{3.2}$$

$$d^2_{(j,i)} = \|\boldsymbol{r}_j - \boldsymbol{r}_i\|^2 \tag{3.3}$$

where $\boldsymbol{r}_j$ is a discrete vector that defines the position of the excited neuron j and $\boldsymbol{r}_i$ is the one that determines the position of the winning neuron i.

$$\sigma(t) = \sigma_0 \exp\left(\frac{-t}{\lambda}\right) \tag{3.4}$$

$\sigma(t)$ is the temperature function modelling the neighborhood range, σ_0 is the initial temperature decaying over time, t is the current time and λ is a time constant chosen by the designer.

Synaptic Adaptation This last mechanism is the one responsible for the maps to be *self-organized*, it enables the excited neurons to increase their individual values of the discriminant function following the equation 3.5 in relation to the input pattern through suitable updates applied to their synaptic weights.

$$\boldsymbol{w}_j(t+1) = \boldsymbol{w}_j(t) + \eta(t)K_{j,i(x)}(t)(\boldsymbol{x}(t) - \boldsymbol{w}_j(t)) \tag{3.5}$$

The learning rate parameter η should also be time varying, as indicated in equation 3.6

$$\eta(t) = \eta_0 \exp\left(\frac{-t}{\lambda}\right) \tag{3.6}$$

t is the current time and λ is a time constant The algorithm keeps iterating until convergence. The SOM has three main properties as explained by Hykin in [61]:

- Approximation of the input space by the synaptic weight vector in the output space.
- Topological Ordering as the spatial location of a neuron in the lattice corresponds to a particular domain or feature of input patterns.
- Density estimation as the regions in the input space from which sample input vector are drawn with a high probability of occurrence are mapped onto larger domains of the output space.
- Feature selection since the SOM is able to select a set of best features for approximating the underlying distribution.

3.1.3 Multi-view Clustering

Multi-view learning refers to this area of machine learning where data can be represented using multiple distinct feature sets. It has attracted increasing attention in recent years since real world data applications employ examples that are described by multi sources streams, different feature sets or different

"views" as shown in figure 3.3. Multi-view learning has widespread applicability, to name few of its applications [64]:

- Multimedia content understanding needs to simultaneously analyze video and audio signals
- Web-page classification is achieved by describing a web page both by the document text itself and by the anchor text attached to hyperlinks pointing to this page.
- Web-image retrieval where an object is described by visual features from the image and by the text surrounding it.

There are five main categories of multi-view learning summarized by Yang in [65]

- **Co-training style algorithms:** This category of methods treats multi-view data by using co-training strategy. It bootstraps the clustering of different views by using the prior or learning knowledge from one another. By iteratively carrying out this strategy, the clustering results of all views tend to each other and this leads to the broadest consensus across all views.
- **Multi-kernel learning:** This category of methods uses predefined kernels corresponding to different views, and then combines these kernels either linearly or non-linearly in order to improve clustering performance.
- **Multi-view graph clustering:** This category of methods seeks to find a fusion graph (or network) across all views and then uses graph-cut algorithms or other clustering techniques on the fusion graph in order to produce the final result.
- **Multi-view subspace clustering:** This category learns a unified feature representation (to be input into a model for clustering) from all the feature subspaces of all views by assuming that all views share this representation.
- **Multi-task multi-view clustering:** This category treads each view with one task or multiple related tasks, transfers the inter-task knowledge to one another, and exploits multi-task and multi-view relationships in order to improve clustering performance.

Through analyzing the different categories described above, we observe that they mainly depend on either the consensus principle or the complementary principle to ensure their success. It is very important to note that the major difference between single-view and multi-view learning algorithms is that the latter demands redundant views of the same input data what gives the learning task abundant information to work with [66]. Multiple view generation not only aims to obtain the views of different attributes, but also involves the problem of ensuring that the views sufficiently represent the

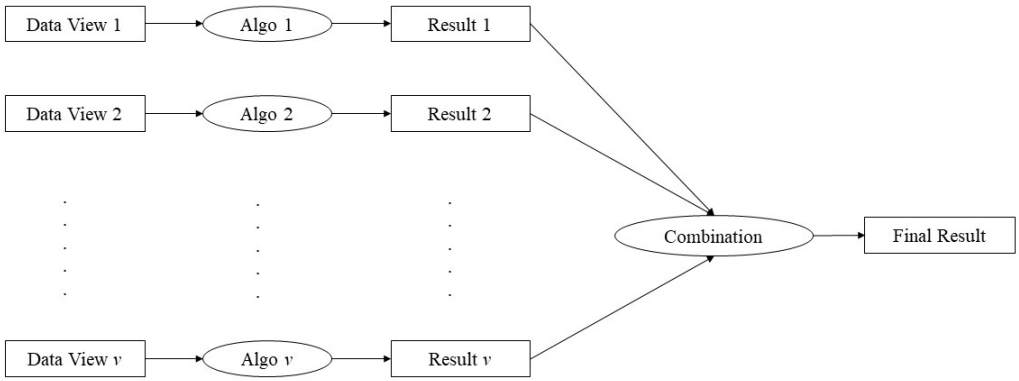


FIGURE 3.3: Multi-view learning Process

data and satisfy the assumptions required for learning. A simple way to convert from a single view to multiple views is to split the original feature set into different views at random. The principles of the multi-view learning are detailed below:

- **Consensus principle:** aims to maximize the agreement on multiple distinct views. For example, the co-training algorithm trains alternately to maximize the mutual agreement on two distinct views of the unlabeled data. By minimizing the error on labeled examples and maximizing the agreement on unlabeled examples, the co-training algorithm finally achieves one accurate classifier on each view.

- **Complementary principle:** states that in a multi-view setting, each view of the data may contain some knowledge that other views do not have; therefore, multiple views can be employed to comprehensively and accurately describe the data. In machine learning problems involving multi-view data, the complementary information underlying multiple views can be exploited to improve the learning performance by utilizing the complementary principle. However if the learning method is unable to cope appropriately with multiple views, these views may even degrade the performance of multi-view learning.

3.1.4 Collaborative Topological Learning

Collaborative clustering was first introduced in the works of Pedrycz [67] on fuzzy clustering. It can be defined by comparing it to the cooperative clustering and determining the main differences between both types. Contrarily to the cooperative clustering model, the collaborative model does not look for

obtaining better clustering results by combining individual solutions. The collaborative model looks instead to exchange information about the local data, or the current hypothesized local clustering, or the value of one algorithm's parameters between what they call "collaborators". Each local computation applied to a distinct data might benefit from the calculations done by the other data set. The architecture of such a model leads naturally to distributed computations [68].

Collaborative methods usually follow a two-step process:

1. Local step: Each algorithm will process the data it has access to and produce a clustering result; For example, information about the candidate structures hypothesized in the data sets and or the memberships of the instances in the data set.

2. Collaborative step: The algorithms share their results and try to confirm or improve their models with the goal of finding better clustering solutions.

Those two learning steps are then followed by an aggregation in order to reach a consensus between final results after collaboration. In this scenario, the collaborative method is a preliminary step to make the aggregation process easier.

Horizontal Collaborative Clustering: Overview

We distinguish two types of collaborative clustering, the horizontal collaborative clustering and the vertical collaborative clustering. In horizontal clustering, the patterns are the same and the feature spaces are different as shown in the figure 3.4.In horizontal clustering, the communication platform is based on through the partition matrix (Kernels in case of SOM). The confidentiality of data do not be breached: since we operate on the resulting information of a clustering model. The collaboration might be used in order to solve the privacy preservation problem encountered by some of the distributed clustering algorithms in their learning process. For horizontally distributed database, we might consider collaborators as semi-trusted third parties that return clusters' representatives without revealing real data distribution of each of the sites [69].

Horizontal collaboration can be applied to multi-view clustering, multi-expert clustering, clustering of high dimensional data, or multi-scale clustering [70].

Horizontal Collaborative Clustering: Mechanism

Below, we give details about the algorithmic foundation of the horizontal collaborative clustering and its main characteristics. Given P sets of data belonging to multiple sources, or figuring in different input spaces resulting from multi-view preprocessing. Considering that in horizontal clustering the number of elements in each subset is the same and it is equal to N and since each subset is described by the same patterns. The collaboration proposed

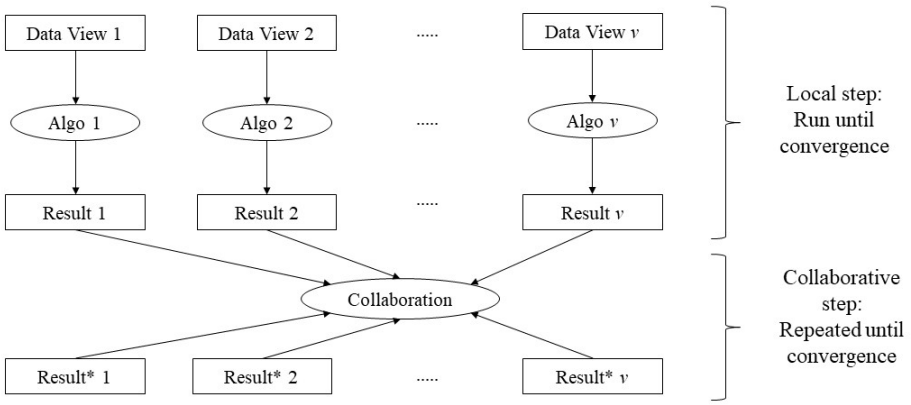


FIGURE 3.4: Horizontal collaborative learning process

between the two subsets is establish through a cofficient that determines the intensity of the collaboration. These cofficients determine the strength of the collaboration, (i.e. the higher the value of the coefficients, the stronger the collaboration between the subsets).

The main idea of the used collaborations is: the index of the neuron belonging to different maps should capture the same observations. This is to say, that if an observation from the k-th data set is projected on the j-th neuron in the k-map, then that same observation in the l-th data set will be projected on the same j neuron or one of its neighboring neurons of the l-th map. Based on the works of [56] [71], the classical SOM objective function is modified by adding a term that reflects the principle of collaboration. The objective function of SOM becomes 3.7.

$$R^{[k]}(\chi, w) = R^{[k]}_{SOM}(\chi, w) + (\lambda^{[l]}_{[k]})^2 R^{[k]}_{Col}(\chi, w) \tag{3.7}$$

with

$$R^{[k]}_{SOM}(\chi, w) = \sum_{i=1}^{N} \sum_{j=1}^{|w|} K^{[k]}_{j,\chi(x_i)} \|x^{[k]}_i - w^{[k]}_j\|^2 \tag{3.8}$$

$$R^{[k]}_{Col}(\chi, w) = \sum_{l=1, l\neq k}^{P} \sum_{i=1}^{N} \sum_{j=1}^{|w|} \left(K^{[k]}_{j,\chi(x_i)} - K^{[l]}_{j,\chi(x_i)}\right)^2 * D_{ij} \tag{3.9}$$

$$with \;\; D_{ij} = \|x^{[k]}_i - w^{[k]}_j\|^2 \tag{3.10}$$

where P represents the number of views, N - the number of observations,

$|w|$ is the number of prototype vectors from the k SOM (the number of neurons). $\chi(x_i)$ is the assignment function which allows to find the Best Matching Unit (BMU), it selects the neuron with the closest prototype from the data x_i using the Euclidean distance.

The value of the collaboration link λ is determined during the first phase of the collaboration step. This parameter allows to determine the importance of the collaboration between each two SOMs. Its value is in the interval $[1-10]$, 1 reflects the neutral link, when no importance to collaboration is given, and 10 the maximal collaboration within a map. Its value changes for each iteration during the collaboration step. In the case of the collaborative learning, as it is shown in the Algorithm 2, this value depends on topological similarity between both collaboration maps.

This function depends on the distance between two neurons and is defined as follows:

$$K_{j,i}^{[k]} = \exp\left(\frac{-d_{(j,i)}^2}{2\sigma(t)^2}\right) \tag{3.11}$$

$d_{(j,i)}$ represents the distance between two neurons i and j from the map, and it is defined as the length of the shortest path linking cells i and j on the SOM. $K_{j,i}^{[k]}$ is the neighborhood function on the $SOM[k]$ between two cells i and j. $\sigma(t)$ is the temperature which allows to control the size of the neighborhood influence of a cell on the map, it decreases with time. The nature of the neighborhood function $K_{j,i}^{[k]}$ is identical for all the maps, but its value changes from one map to another: it depends on the closest prototype to the observation that is not necessarily the same for all the SOM maps.

Linear Mixture of SOM Models

In [72], Kohonen introduced a novel method to analyze input patterns of SOMs. The technique can be described as follow: instead of representing inputs by the 'Best Matching Unit' i.e. the 'Winner neuron', they are described using the linear mixture of the reference vectors. This technique better approximates the input vector. It preserves better the information compared to the classical SOM learning process where only the BMU is used.

Let us consider each input as a Euclidean vector x of dimensionality n. The SOM matrix of prototypes is denoted as M of size (pxn) where p is the number of SOM's reference vectors. To get the coefficients of the models we minimize the following equation:

$$\min \left\| M'\alpha - x \right\|,$$

where α is a vector of non negative scalars α_i. The constraint of non negativeness is important when dealing with inputs consisting of statistical indicators because the negative of a sample has no meaning.

This technique extends the use of SOM by proving that the inputs can be represented by their linear mixture instead of the mere single neuron. For

Algorithm 2 The Topological Collaborative Multi-view Algorithm

Input: P views dataset $V[k]$
Output: P SOMs' optimized $\{w[k]\}_{k=1}^{P}$

Step 1 : Local Step:

1: **for** $k = 1$ to P **do**
2: Learn a SOM for view $V[k]$
3: $w^{[k]} \leftarrow \arg\min_{w} \left[R^{[k]}_{SOM}(\chi, w)\right]$
4: Compute DB index for $SOM[k]$
where $DB^{[k]}$ is the Davies Bouldin index computed using $w^{[k]}$
5: $DB^{[k]}_{Beforecollab} \leftarrow DB^{[k]}$
6: **end for**

Step 2 : Collaborative learning:

7: **for** $k = 1$ to P **do**
8: **for** $l = 1, l \neq k$ to P **do**
9: $$\lambda^{[l]}_{[k]}(t+1) \leftarrow \lambda^{[l]}_{[k]}(t) + \frac{\sum_{i=1}^{N}\sum_{j=1}^{|w|} K^{[k]}_{\sigma(j,\chi(x_i))}}{2\sum_{i=1}^{N}\sum_{j=1}^{|w|}\left(K^{[k]}_{\sigma(j,\chi(x_i))} - K^{[l]}_{\sigma(j,\chi(x_i))}\right)^2}$$
10: $$w^{[k]}_{jk}(t+1) \leftarrow w^{[k]}_{jk}(t) + \frac{\sum\limits_{i=1}^{N} K^{[k]}_{\sigma(j,\chi(x_i))} x^{[k]}_{ik} + \sum\limits_{l=1,l\neq k}^{P}\sum\limits_{i=1}^{N} \lambda^{[l]}_{[k]} L_{ij} x^{[k]}_{ik}}{\sum\limits_{i=1}^{N} K^{[k]}_{\sigma(j,\chi(x_i))} + \sum\limits_{l=1,l\neq k}^{P}\sum\limits_{i=1}^{N} \lambda^{[l]}_{[k]} L_{ij}}$$
11: $DB^{[k]}_{AfterCollab} \leftarrow DB^{[k]}$
12: **if** $DB^{[k]}_{AfterCollab} \geq DB^{[k]}_{BeforeCollab}$ **then**
13: $w^{[k]}_{jk}(t+1) \leftarrow w^{[k]}_{jk}(t)$
14: **end if**
15: **end for**
16: **end for**

the solution of the above objective function, there exist several ways. The most used and straightforward is the gradient-descent optimization. It's an iterative algorithm that can take into account the non-negativity constraint. The present fitting problem belongs to the quadratic optimization, for which numerous methods have been developed over the years. A one-pass solution is based on the Kuhn Tucker theorem [73]. It was implemented in a Matlab function called lsqnonneg as:

$$\alpha = lsqnonneg(M', x, \alpha(1))$$

This function returns the vector α that minimizes the norm $\|M' * \alpha - x\|$ subject to $x \geq 0$. Each element of the obtained vector is viewed as the coefficient of the prototype with the same index. The use of linear mixture of SOM models was proven to out-perform the BMU method as it preserves more information.

Notations

We use the *k*-anonymity notation i.e. data are organized as a table of rows (Records) and columns (Attributes) where each row is defined as a tuple, the tuples are not unique but attributes are. Each row is an ordered *m*- tuple of values $< a_1, a_2, .., a_j, .., a_m >$.

Notation 1 *Let $T\{A_1, A_2, .., A_m\}$ be a table with a finite number of tuples corresponding to attributes $\{A_1, A_2, .., A_m\}$. Given $T = \{A_1, A_2, .., A_m\}$, $\{A_l, .., A_k\} \subseteq \{A_1, A_2, .., A_m\}$*
For $t \in T$, $t[A_l, .., A_k]$ refers to the tuple of elements $x_l, .., x_k$ of $A_l, .., A_k$ in T.

Let us consider a table T of size $n \times m$, m is the number of attributes and n is the number of elements. The table is denoted $T = \{A_1, A_2, .., A_m\}$.

Definition 3.1.1 *k-anonymity*

$AT\{A_1, A_2, .., A_m\}$, is a table, OT is said to be k-anonymous if and only if each tuple in AT has at least k occurrences.

Definition 3.1.2 *The Davies Bouldin Index*

The DB index [74] is based on a similarity measure of clusters R_{ij} that is a fraction of the dispersion measure s_i and the cluster dissimilarity d_{ij} [75]. R_{ij} should satisfy the following:

$$
\begin{aligned}
&R_{ij} \geq 0 \\
&R_{ij} = R_{ji} \\
&R_{ij} = 0 \quad if s_i = s_j = 0 \\
&R_{ij} > R_{ik} \quad if s_j > s_k \; d_{ij} = d_{ik} \\
&R_{ij} > R_{ik} \quad if s_j = s_k \; d_{ij} < d_{ik} \\
&R_{ij} = \frac{s_i + s_j}{d_{ij}} \quad where \quad d_{ij} = dist(w_i, w_j) \; s_i = \frac{1}{\|c_i\|} \sum_{x \in c_i} d(x, c_i) \\
&DB = \frac{1}{n_c} \sum_{i=1}^{n_c} R_i \quad where \quad R_i = \max_{j=1,..,n_c} R_{ij}, \quad i = 1..n_c
\end{aligned}
$$

Where w_i are the prototypes of the neuron, n_c is the number of cells, c_i is the i^{th} cell.

Davies-Bouldin is a cluster validity index used to measure the "goodness" of a clustering result [74]. It takes in account the compactness and the separability of clusters and works best and foremost with hard clustering (when the clusters have no overlapping partitions).

Since the objective is to obtain clusters with minimum intra-cluster distances, small values for DB are interesting, the usage of this validity index is justified by our wiliness to evaluate how the elements of the same cluster are similar. Therefore, this index is minimized when looking for the best number of clusters [76].

3.2 A Topological k-Anonymity Model based on Collaborative Multi-view Clustering: *k*-TCA Algorithm

The proposed anonymization method uses the multi-view approach with the purpose of treating complex data and multisources data. This technique is also used to preserve the quality of the dataset to recode and prevent the dimensionality curse. The number of subsets to be used for collaboration is fixed by the user and it depends on the size of the data. The algorithm 2 use classical SOM and collaborative paradigm to form the maps by exchanging the topological information between the collaborated maps. In the pre-anonymization step shown in algorithm 3, the dataset is coded using the prototypes of the best matching units for each data point. At the end of this step, the output is a pre-anonymized dataset that will be fine-tuned using a SOM model where the map size is determined by the Kohonen heuristic [62].

Algorithm 3 The *k*- TCA Algorithm Protocol

Input : D dataset to anonymize
P number of views $V[p]$
Output: Anonymized dataset
k anonymity level

1 **Collaboration step:**

- Randomly generate P views $V[p]$
- Use the collaboration algorithm presented in Algorithm 2 with all $V[p]$

Pre-Anonymization:
For each $V[p]$, $p = 1$ to P :

- Find the BMU (Best Matching Unit) for each object in $V[p]$ using corresponding $w[p]$
- Code the dataset D using all code $V[p]$, output result in D'

Fine-tuning and anonymization:

- Build a global SOM using the pre-anonymized dataset D'
- Find the BMU for each object in D'
- Recode the dataset, output results in D'' and evaluate the k-anonymity level of D''

The model presented in this case uses the multi-view collaborative algorithm to do the first level of anonymization using the collaboration between multiple views of the data set to be anonymized. We call this first level of anonymization *the pre-anonymization step*, it takes advantage from the ability of Topological Collaborative SOMs to accomplish clustering without identity

breach. This step is common to both algorithms presented in this chapter, namely, algorithm 3 and algorithm 4.

3.3 *k*-Anonymization through Constrained Collaborative Clustering: *C*-TCA Algorithm

In the method presented in algorithm 2, the multi-view learning approach provides more flexibility to deal with different sources of data. It also allows to obtain more accurate clustering results and a better feature coding since it use each view of the data set alone which helps reduce the curse of dimensionality. The number of subsets used in collaboration is fixed by the user depending on the dimension of the data.

Algorithm 4 The *C*-TCA Algorithm Protocol

Input: D dataset to anonymize
P number of views $V[p]$
Output: Anonymized dataset
Multi-view Clustering step :

- Randomly generate P views $V[p]$
- Create a SOM for each view $V[p]$
- Use the collaboration algorithm presented in Algorithm 1 with all $V[p]$

Pre-Anonymization :
For each $V[p]$, $p = 1$ to P :

- Find the linear mixture of SOM models for each object in $V[p]$
- Code the dataset D using all code $V[p]$, output result in D'

Constrained Clustering and Anonymization :

- Build a global SOM using the pre-anonymized dataset D'
- Find the clusters with number of objects less than k
- Redistribute these elements on the other clusters in a way to have at least k element in each remaining neuron /cluster
- Find the new BMUs
- Recode the dataset D', output results in D'' and evaluate the quality of the anonymized data using accuracy

The algorithm 2 builds classical SOMs and uses the collaborative paradigm to exchange topological informations between collaborators. The Davies Bouldin index [74] is a clustering evaluation indicator that reflects the quality of the clustering, which is used here as a stopping criterion. If DB decreases it means that the collaboration is positive and if it increases, we stop the collaboration and use the initial map. Therefore, the collaboration allows us to obtain more homogeneous clusters by using the topological information from all the views.
The elements of each of the collaborating maps are coded using the linear mixture of the map's prototypes. This coding method gives better results than coding the elements with the best matching units because it preserves most of the information contained in each element and describes the element using a combination of all the SOM's models. The pre-anonymized parts are then reorganized in the same way as the the original data set. In order to guarantee a minimum k anonymity level, we learn a new constrained SOM map on the pre-anonymized data set i.e. each neuron contains at least k elements. The constrained map is created by using the following two steps : firstly an initial map is learned using classical SOM and secondly the elements from the neurons which don't respect the constraint of k cardinality are redistributed in the closest neurons. This process can modify the topology of the map, but helps designing groups of at least k elements in each neuron. We code the objects of each neuron using the best matching unit, this way we can get a k -anonymized data set.

3.4 Utility Measures

3.4.1 Earth Mover's distance as a measure of structural Utility preservation

The anonymized datasets are analysed using the Earth Mover's distance also known as the Wasserstein distance [77], this distance extends the notion of distance between two single elements to that of a distance between sets or distributions of elements. The Earth Mover's distance compare the probability distributions P and Q on a measurable space (Ω, Ψ) is defined as follows (We are using the distance of order 1):

$$W_1(P,Q) = \inf_{\mu} \left\{ \int_{\Omega\times\Omega} |x-y| \mathrm{d}\mu(x,y) \right\} \tag{3.12}$$

$$\mu : \text{prob. measure on } (\Omega\times\Omega, \Psi\otimes\Psi) \quad \text{with marginals } P, Q$$

where $\Omega \times \Omega$ is the product probability space. Notice that we may extend the definition so that P is a measure on a space (Ω, Ψ) and Q is a measure on a space (Ω', Ψ').

Let us examine how the above is applied in the case of discrete sample spaces. For generality, we assume that P is a measure on (Ω, Ψ) where $\Omega =$

$\{x_i\}_{i=1}^{n}$ and Q is a measure on (Ω', Ψ') where $\Omega' = \{y_i\}_{j=1}^{n'}$ - the two spaces are not required to have the same cardinality.

Then, the distance between P and Q becomes:

$$W_1(P,Q) = \inf_{\{\lambda_{i,j}\},i,j} \left\{ \sum_{i=1}^{n}\sum_{j=1}^{n'} \lambda_{i,j}|x_i - y_j| : \sum_{i=1}^{n} \lambda_{i,j} = q_j, \sum_{j=1}^{n'} \lambda_{i,j} = p_i, \lambda_{i,j} \geq 0 \right\} \tag{3.13}$$

Based on this definition, we believe that the best way to evaluate the utility of the anonymized dataset is to measure the distance between its distribution and the distribution of the original dataset attribute by attribute, this way, the distortion of the anonymized datasets can be easily identified. We then normalize all distances between 0 and 1, then we define the utility by $1 - W_1(P,Q)$. The smaller the distance W_1 is, the more the data utility is preserved.

3.4.2 Preserving combined utility

To choose the anonymization method which best addresses the separability structural utility Trade-off, we propose to combine the two types of utility structural and separability in a combined form while $\alpha = \frac{1}{2}$:

$$Comb - Utility = \alpha.Separability + (1-\alpha).Structural$$

To further evaluate the performance, we compute a measurement score by following [78]:

$$Score(A_i) = \sum_j \frac{CombUtility(A_i, D_j)}{\max_i CombUtility(A_i, D_j)} \tag{3.14}$$

where $CombUtility(A_i, D_j)$ refers to the combined Utility value of A_i method on the D_j dataset. This score gives an overall evaluation on all the datasets.

3.5 Experimental Results

3.5.1 Datasets

The methods were tested on several datasets provided by the UCI Machine Learning Repository [79]:

- The DrivFace database contains images sequences of subjects while driving in real scenarios. It is composed of 606 samples of 6400×480 pixels each, acquired over different days from 4 drivers (2 women and 2 men) with several facial features like glasses and beard.
- Ecoli & Yeast datasets contain protein localization sites. Each of the attributes used to classify the localization site of a protein is a score

(between 0 and 1) corresponding to a certain feature of the protein sequence. The higher the score is, the more possible the protein sequence has such feature.

- Glass dataset represents oxide content of the glass to determine its type. The study of classification of types of glass was motivated by criminological investigation. Since the glass left at the scene of the crime can be used as evidence...if it is correctly identified!

- Waveform describes 3 types of waves with an added noise. Each class is generated from a combination of 2 of 3 "base" waves and each instance is generated of added noise (mean 0, variance 1) in each attribute.

- Wine data is the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

3.5.2 First Level of Anonymization Validation

The impact of microaggregation on the utility of anonymized data are quantified as the resulting accuracy of a machine learning model [42]. To quantify the utility of the dataset for further study and since all the datasets used are labelled we thought that the best way to evaluate the proposed approaches is to use an external evaluation i.e. the classification. For this purpose, we designed a decision tree model and used it to see how the anonymized data was classified by this model. We then compared the accuracy of the results of both approaches to understand how much data *quality* have we traded for the sake of anonymization. The pre-anonymization step was crucial to create anonymized elements by views i.e. we didn't code the whole example by one model, instead, we coded each part of the example, depending on the view it belongs to, by the BMU in the case of Constrained TCA (algorithm 4), and by the linear mixture of the neighboring models in case of the *k*-TCA (algorithm 3). Table 3.1 illustrates the results of the two algorithms. To get more insights to the table we explain its elements and the experiences they represent in the following:

- **Original**: The initial accuracy of the raw data using the decision tree model.

- **BMU without Collab**: The accuracy of the dataset using the multi-view clustering without collaboration between the views and with a fixed size of the maps. The examples were coded using the BMUs.

- ***k*-TCA**: The accuracy of the dataset using the multi-view clustering with collaboration between the views and fixed map size. The examples were coded using the BMUs.

- ***k*-TCA-KH**: The accuracy of the dataset using the multi-view clustering with collaboration between the views and using the Kohonen Heuristic

to determine the size of the maps to use. The examples were coded using the BMUs.

- **Acc-Constrained-TCA-KH**: The accuracy of the dataset using the multi-view clustering with collaboration between the views and using the Kohonen Heuristic to determine the size of the maps to use. The examples were coded using the Linear Mixture of Models.

	DrivFace	Ecoli	Glass
Original	92.24	82.44	69.63
95% confidence interval	[89.86 , 94.62]	[77.84 , 87.04]	[61.76 , 77.50]
BMU without Collab	92.24	79.46	95.79
95% confidence interval	[90.77 , 93.71]	[75.14 , 83.78]	[93.15 , 98.43]
k-TCA	91.24	82.14	96.26
95% confidence interval	[89.29 , 93.19]	[81.40 , 87.64]	[93.60 , 98.92]
k-TCA-KH	96.21	80.06	77.60
Confidence interval 95%	[92.76, 98.34]	[72.39, 88.91]	[75.76, 80.43]
Acc-Constrained-TCA-KH	95.55	80.65	81.84
Confidence interval 95%	[90.82, 94.98]	[77.77, 88.31]	[79.85, 83.11]
	Waveform	**Wine**	**Yeast**
Original	76.88	88.76	83.63
95% confidence interval	[75.89 , 77.87]	[84.72 , 92.80]	[81.66 , 85.60]
BMU without Collab	81.98	89.89	86.05
95% confidence interval	[80.37 , 83.59]	[85.88 , 93.90]	[85.23 , 86.87]
k-TCA	81.94	88.76	84.30
95% confidence interval	[80.67 , 83.21]	[85.37 , 92.15]	[82.91 , 85.69]
k-TCA-KH	97.66	90.45	83.96
Confidence interval 95%	[95.56, 99.76]	[85.18, 95.72]	[80.01, 85.65]
Acc-Constrained-TCA-KH	93.92	89.78	87.73
Confidence interval 95%	[90.94, 96.9]	[86.53, 93.03]	[84.51, 90.69]

TABLE 3.1: Accuracy and confidence interval of the different tests on the Pre-anonymization step

	DrivFace	Ecoli	Glass	Waveform	Wine	Yeast
Before Collaboration	7.94	4.23	5.16	5.35	18.74	3.97
After Collaboration	7.56	4.16	3.70	5.28	16.71	3.94

TABLE 3.2: DB index before and after collaboration.

The first result we like to highlight is the collaboration effect. In the proposed experimental protocol, we used the DB index [74] as a stopping criterion for the collaboration between the maps. The Davies Bouldin index is a measure to evaluate the clustering quality. In our experiences, if the DB

index decreases we conclude that the collaboration is positive, we keep it, per contra, if it increases, we note that the collaboration was negative and in this case we proceed without collaboration. In the table 3.2 we show the results of the DB index on the different datasets. For all the datasets the collaboration was proven to be positive since the DB index increased. The accuracy after collaboration as presented in table 3.1 decreased slightly for some of the datasets. By considering the DB index as a clustering quality criterion we believe that we gain in the clustering quality. So the outputs of the pre-anonymization are well clustered what implies well anonymized with a slight decrease of the accuracy as a trade-off (1% for the DrivFace data, 0.04% for the WaveForm data, 1.12% for the Wine data & 1.75% for the Yeast data).

The second result is that the anonymized datasets, if used by the same model, gives better results than the initial accuracy using raw data. This can be explained by the process by which we anonymized the initial dataset, the process relies on clustering what implies that the different pattern of the datasets were discovered and all the noise was omitted. Let's take the Waveform data as an illustrative example. The used Waveform dataset is noisy, what explains that, at the start of the experiments, the accuracy was equal to 76.88%, after using the k-TCA (algorithm 3) increased by 5.06% after applying the Constrained TCA (algorithm 4) it increased by 20.98% (table 3.1). Same goes for the other datasets (DrivFace, Glass, Waveform, Wine, Yeast) where the accuracy obtained after applying the Constrained TCA increased significantly compared to the accuracy at the start of the experiments. Also in table 3.4, we illustrate the results of the Maximum Distance to Average algorithm (MDAV) [46]. MDAV represents the key attributes in a data set as points in the Euclidean space where k-anonymous microaggregation is the partitioning of points in cells of size k. The perturbed attributes are then characterized with a representative point at maximum distance of the average.

The algorithms we proposed outperform the MDAV algorithm as shown on the table 3.4.

3.5.3 Second Level of Anonymization Validation

The step of pre-anonymization is crucial to the rest of the experiments since it helps improving the quality of the fine tuning output. Table 3.3 illustrates the difference between the accuracy of the different steps of the process. We have the accuracy after fine tuning using the k-TCA (algorithm 3), the accuracy after fine tuning from the table 3.1 to measure the dataset's quality improvement and also the different k anonymity values that we got automatically.

From the table 3.3 we can say that the fine tuning and anonymization step helped enhance the quality of 4 out of 6 datasets with a slight decrease of 1.98% for the DrivFace dataset. The decrease in accuracy is minimal since the data anonymity level goes up to 10 which is considered a very good trade-off between the both factors (Anonymity & Utility).

	DrivFace	Ecoli	Glass	Waveform	Wine	Yeast
Original	92.24	82.44	69.63	76.88	88.76	83.63
Pre-Anonymization	91.24	82.14	96.26	81.94	88.76	84.30
After FineTuning	90.26	84.52	94.39	83.00	69.66	86.25
k	10	2	5	4	3	3

TABLE 3.3: Accuracy & k-anonymity level after Fine tuning as described in k-TCA, algorithm 3

	DrivFace	Ecoli	Glass	Waveform	Wine	Yeast
Original	92.24	82.44	69.63	76.88	88.76	83.63
MDAV	89.12	75.60	61.24	69.82	68.42	83.35
k-TCA	90.26	84.52	94.39	83.00	69.66	86.25
Constrained TCA	93.23	85.12	75.23	81.54	74.16	87.39

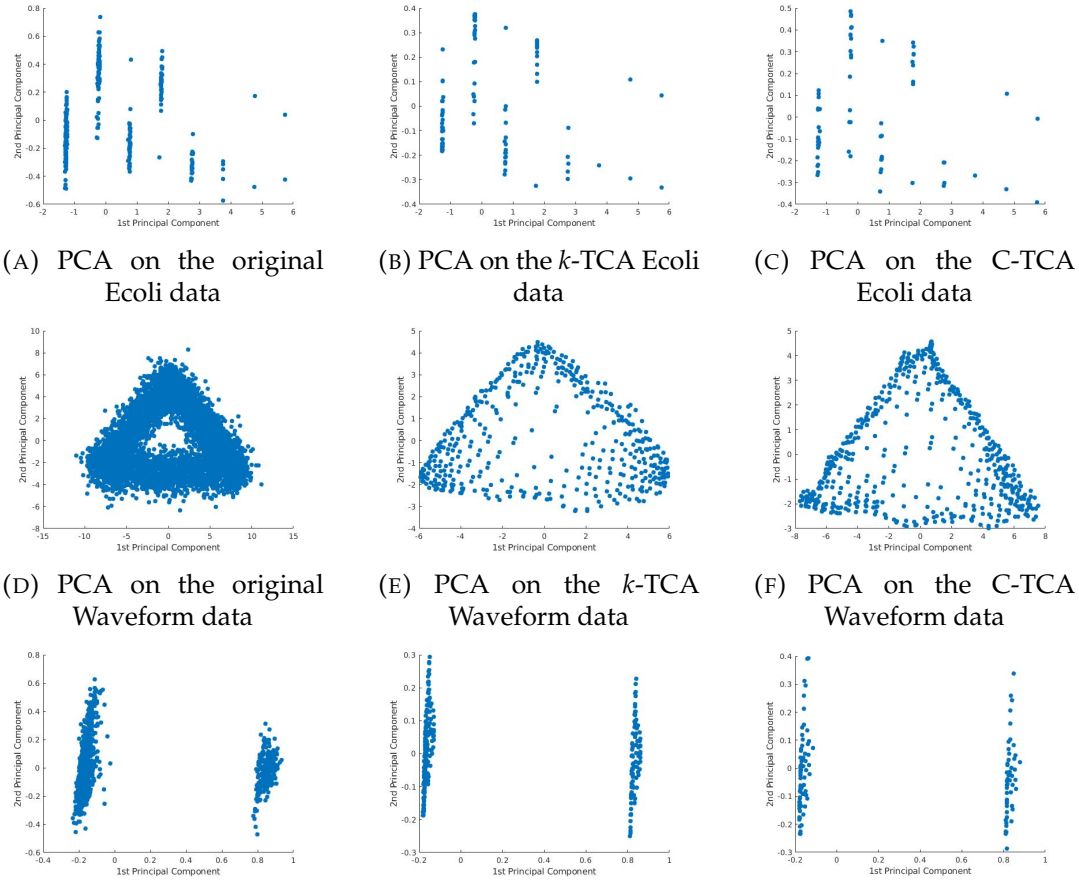
TABLE 3.4: Accuracy of the proposed algorithm compared to the MDAV algorithm with $k = 5$

In table 3.5 we varied the k anonymity level and explored the accuracy of the different datasets. To compare between the two methods we highlighted, in italic in table 3.5, the values of the accuracy corresponding to the k anonymity level found by the algorithm 3.3 and in bold, the values that might give better accuracy on k levels. The results shown in the table 3.5 illustrate the pertinence of the methods presented earlier and prove that there is a logical link between the two.

The figure 3.5 shows the projections representations of three of datasets used above. From top to bottom we find Ecoli, Waveform and Yeast datasets and from left to right we find the PCA (on the original dataset, on the results of the first anonymization and on the results of the second anonymization). The representation illustrates how the data doesn't lose its shape after anonymization, this means that the data anonymization methods respect initial data structure.

k	3	4	5	6	7	8	9	10
Drivface	**92.57**	**93.23**	**93.23**	**93.56**	**92.44**	**92.57**	**92.08**	*90.59*
Ecoli	83.33	**85.71**	**85.12**	82.74	**84.82**	80.36	58.93	58.93
Glass	**93.46**	**92.22**	*75.23*	69.16	44.86	50.94	50.94	50.94
Waveform	**81.38**	*81.28*	**81.54**	**81.58**	**81.64**	**81.9**	**81.87**	**81.74**
Wine	*67.42*	**69.66**	**74.16**	**70.79**	**69.66**	**70.79**	**70.79**	66.85
Yeast	*87.8*	87.53	87.39	86.39	**87.8**	**88**	**88.14**	**88.14**

TABLE 3.5: Accuracy of the datasets after fine tuning. Exploration of the different k levels as in Constrained TCA, algorithm 4



(A) PCA on the original Ecoli data

(B) PCA on the *k*-TCA Ecoli data

(C) PCA on the C-TCA Ecoli data

(D) PCA on the original Waveform data

(E) PCA on the *k*-TCA Waveform data

(F) PCA on the C-TCA Waveform data

(G) PCA on the original Yeast data

(H) PCA on the *k*-TCA Yeast data

(I) PCA on the C-TCA Yeast data

FIGURE 3.5: PCA on the anonymized datasets compared to the original data

Separability Utility preservation analysis

The proposed Separability Utility measure is determined by putting the dataset through a real-case scenario. The original and the anonymized tables were learned to a decision tree model with 10 folds cross validation and their Separability Utility was measured. The results given in table 3.6, dress a comparison between the separability utility measures of the original and the anonymized datasets. The loss in information's quality after anonymization is minimal. Melior, the separability utility was improved in most of the cases. This can be explained by the fact that the clustering gathers together elements with the same features and therefore omits non decisive attributes from data.

The method proposes a new data representation which makes it useful for data encryption as shown in table 3.6, the separability utility of the anonymized datasets is very good compared to its value before anonymization.

	Ecoli	Electrical	Glass
Original	82.1	99.5	69.2
95% CI	[73.1, 87.1]	[99.3, 99.7]	[61.3, 78.8]
k-TCA	84.5	99.9	90.25
95% CI	[77.8, 87.0]	[99.8, 99.9]	[87.4 , 93.1]
C-TCA	84.8	99.9	74.7
95% CI	[76.5, 93.0]	[99.8, 99.9]	[68.6, 89]
	Page Blocks	Waveform	Yeast
Original	96.6	74.8	83.4
95% CI	[95.9, 97.4]	[73.4, 77.6]	[75.5, 91.3]
k-TCA	90.25	83.0	86.4
95% CI	[89.9,90.6]	[82.4, 83.6]	[85.0, 87.8]
C-TCA	91.5	81.6	87.6
95% CI	[90.7,92.2]	[80.1,83.0]	[85.3,89.8]

TABLE 3.6: Impact of anonymization on Separability Utility (CI: confidence interval)

Structural Utility preservation analysis

We consider the probability distribution of three anonymization methods, *k*-anonymity using collaborative Multi-view Clustering (k-TCA), *k*-anonymity through constrained Clustering (Constrained TCA) and

	Ecoli	Electrical	Glass	Page Blocks	Waveform	Yeast
k-TCA	0.42	0.5	0.47	0.29	0.50	0.16
C-TCA	0.63	0.5	0.34	0.49	0.49	0.15

TABLE 3.7: Impact of anonymization on Structural Utility ($W_1(P, Q)$)

To quantify how much the anonymized datasets lost of their structural utility, we use the Earth Mover's Distance. This non-parametric distance is used to detect which of the anonymization methods is the most efficient in terms of its fidelity to the original dataset. The structural utility matrix of table 3.7 is used as an input to the Friedman statistical test, we use this test to propose a ranking of the three anonymization approaches in terms of the structural utility and thus differentiating between them. The test works as follows:

- Calculate the EMD between the original and anonymized datasets attribute by attribute.
- Take the median of each distance vector corresponding to each of the anonymi-zation methods.

	Ecoli	Electrical	Glass	Page Blocks	Waveform	Yeast	Score
k-TCA	0.63	0.74	0.71	0.60	0.66	0.51	**4.96**
C-TCA	0.74	0.75	0.54	0.70	0.65	0.51	**4.92**

TABLE 3.8: Combined separability and structural utility Comb-Utility

- Run the test on the matrix of medians i.e. the structural utility matrix.

The Constrained TCA is the one with the fewest points due to the usage of constrained clustering with a *k*-anonymity level of 5, we obtained an anonymized dataset with many overlayed points.

Preserving combined utility

Table 3.8 summarize the clustering results of the proposed approaches in terms of combined utility ($Comb - Utility$).

3.6 Discussion

In this chapter we covered in details the proposed approaches, *k*-TCA & Constrained TCA that we introduced for data anonymization. The results shown above prove the efficiency of the methods and illustrate its importance. The main contribution are summed up in the following points:

- The Multi-view clustering is a great way to deal with multisources data and high dimensional elements.
- The collaborative topological clustering improves the quality of the clustering what makes the model more accurate.
- The pre-anonymization using the Linear Mixture of SOMs gives better results than using BMUs.
- The accuracy of the datasets using the Linear Mixture of SOMs with Constrained SOM is more than the accuracy of the datasets with the *k*-TCA method.
- We found a good trade-off between the accuracy and anonymity levels.
- The Constrained TCA method gives the possibility to explore different levels of *k* anonymity and their respective accuracy's.

We are looking for other ways to anonymize data and we are experiencing 1D clustering as a way to anonymize data without loosing the information it is containing, also since the data is labelized we want to explore with weighted

vector quantization to give better approximation to the cells' representatives and reduce the information loss while preserving data utility [80].

Chapter 4

Attribute-Oriented Data Anonymization

Given the results obtained using the k-TCA and the C-TCA algorithms thoroughly discussed above in chapter 3, we wanted to explore with another type of clustering which is the density based clustering. It is a good path to explore as we wanted an efficient method of clustering that can be used to anonymize data while preserving the aspects and the utility of the original data. The problem noticed is that all the features of the dataset are grouped at once and in the same manner which can impact the quality of the output data. 1D clustering proceeding attribute by attribute is seen as a solution to this problem. That way, the characteristics of each attribute are preserved.

In the following, we will go through the different aspects of the density based clustering methods, then we are going to give fundamental notions of the Kernel Density Estimation in the univariate case and then we will explain in details the proposed algorithm and finally we will give the experimental results concerning the separability utility and the structural utility of the anonymized dataset.

4.1 Fundamental Concepts

4.1.1 Density based clustering

The general idea of density-based clustering methods is to continue growing the given cluster as long as the density or data points in the neighborhood exceed some threshold. A cluster is then distinguished mainly according to the probability distribution in the data. Regions with high densities of objects are recognized as clusters, and areas with sparse distributions of objects are boundaries to keep clusters divided from one another [81]. Density-based algorithms do not assume that the clusters should have specific shapes and can easily detect concave clusters if the parameters are well tuned. Hence, they can find arbitrary shaped clusters. Those algorithms propose a generalization of the ideas at the basis of the univariate procedure and thus shift the formulation from a space with any dimension to a one dimensional space. For this reason computation and visualization are both eased [82].

Examples of such density based methods include the DBSCAN algorithm (Density-Based Spatial Clustering of Applications with Noise [83]) which

uses a density-based notion of cluster and the key idea is that for each point of a cluster the neighborhood of a given radius has to contain at least a minimum number of points. or the OPTICS algorithm (Ordering points to identify the clustering structure) which adds to the first, a second threshold determining the minimum number of objects that must be in a neighborhood for the said neighborhood to be considered dense. [70].

One of the mostly used density based method that doesn't require any a priori knowledge concerning the data is the Kernel Density Estimation (KDE), it can be viewed as an attempt to estimate the probability density function from which the data was drawn. In short, this is done by putting a kernel over each point and summing them all up. This has the advantage to provide a continuous estimate to the probability density function. Although it can be sensitive to bandwidth. The clustering is done by simply selecting different level sets of the KDE. In [81], they discuss the ability of KDE to discoverdata's underlying group structure solely depending on the data's own characteristics and hereby requires no a priori information about the data; by searching the nearest local maxima of the density estimate to the corresponding data points in a given data space and this by employing an ascending gradient method. The KDE method and its mathematical foundations are detailed in subsection 4.1.2

4.1.2 Kernel Density Estimation

Estimating a probability density is very useful to investigate the properties of a given data set which can give great insights on data features as skewness and multimodality. Usually, parametric estimation models (e.g. Gaussian distribution with unknown expectation and variance) are chosen to perform estimation but it was proven that they showed some poor behaviour:

- The chosen density might be a poor model of the distribution that generates the data which results in poor predictions.
- The process that generates the data is multimodal, the aspects of the distribution can never be captured.

This poor behaviour is due to the fact the we don't have a good parametric model of the distribution probability function of the data. To circumvent this issue we will use a non-parametric approach to estimate the distribution.

The Histograms When the analytic form of the distribution function is not accessible we use non-parametric density estimation strategies. The oldest and most widely used density estimator is the histogram. To define the histogram the first step is to divide the entire range of values into a series of consecutive, non-overlapping intervals, called bins e.g.:

$$I_j = (x_0 + j \cdot h; x_0 + (j+1) \cdot h] \quad (j = \ldots, -1, 0, 1, \ldots). \tag{4.1}$$

Here the origin of the histogram is x_0 and the bin width is given by h. The form of the histogram highly depends on x_0 and h since they are considered

as tuning parameters. The second step consists in to count how many values fall into each interval I_j. More precisely we compute the frequencies, i.e. the number of samples in each fixed bin. Histograms give a rough sense of the density of the underlying distribution of the data, and often for density estimation: estimating the probability density function of the underlying variable. Let us assume that the density probability function f exists, the estimate of f by the histogram method is given by:

$$f_H(x) = \frac{1}{nh} \times \#\{i;\ X_i \in (x_0 + j \cdot h; x_0 + (j+1) \cdot h]\} \tag{4.2}$$

Here $\#E$ denotes the cardinality of the set E. Note that bins need not be of equal width. The histogram can be generalized by allowing the bin widths to vary. The histograms discontinuity causes extreme difficulty if derivatives of the estimates are required what made estimating density by simple histograms not enough to find modes of the distribution function and therefore the use of more sophisticated methods necessary [84].

The Kernel Estimator By definition the density probability function f_X of a random variable X, when it exists, is given by:

$$f_X(x) = \lim_{h \to 0} \frac{1}{2h} P(x - h < X \leq x + h) \tag{4.3}$$

We could estimate, for any given h, the vule of the probability $P(x - h < X \leq x + h)$ by the proportion of the sample data falling in the interval $(x - h, x + h)$. By choosing a small h we define the naive estimator of f_X by :

$$f_{N,X}(x) = \frac{1}{2nh} \times \#\{i;\ X_i \in (x - h; x + h]\} \tag{4.4}$$

The naive estimator could be also written as:

$$f_{N,X}(x) = \frac{1}{nh} \sum_{i=1}^{n} w\left(\frac{x - X_i}{h}\right), \tag{4.5}$$

where

$$w(x) = \begin{cases} \frac{1}{2}, & \text{if } \mid x \mid < 1 \\ 0, & \text{otherwise} \end{cases} \tag{4.6}$$

In contrast to the histogram, in the naive estimator's case, we shouldn't specify the origin x_0. The equation 4.5 shows that the constructed estimate is obtained by placing a "box" of width $2h$ and height $\frac{1}{2hn}$ on each observation. This approach is better than the histogram approach but as in the case of the histogram estimator the function estimate is not continuous and its smoothness highly depends on the bin width h value. To overcome those difficulties we use a kernel function $K(\cdot)$ instead of the weight function $w(\cdot)$.

Kernel name	K(x)
Normal	$\frac{1}{\sqrt{2\pi}} \exp \frac{-u^2}{2}$
Epanechnikov	$\frac{3}{4}(1-u^2) \quad \lvert u\rvert \leq 1$
Uniform (Box)	$\frac{1}{2} \quad \lvert u\rvert \leq 1$
Biweight	$\frac{15}{16}(1-u^2)^2 \quad \lvert u\rvert \leq 1$
Triweight	$\frac{35}{32}(1-u^2)^3 \quad \lvert u\rvert \leq 1$
Triangular	$(1-\lvert u\rvert) \quad \lvert u\rvert \leq 1$

TABLE 4.1: Kernel functions

The kernel function should satisfy the following properties:

$$\begin{aligned} &\int K(x)dx = 1 \\ &\int xK(x)dx = 0 \\ &\int x^2K(x)dx < \infty \\ &K(x) \geq 0 \\ &K(x) = K(-x) \end{aligned}$$

The kernel function is then symmetric, its PDF is continuous with mean 0 and it has a bounded variance. The popular univariate kernel functions are described in table 4.1, their shapes are illustrated in the figure 4.1

The kernel estimator of f_X writes:

$$f_{K,X}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{4.7}$$

The estimator depends on the bandwidth $h > 0$ which acts like a smoothing parameter, for a large bandwidth h, the estimate $f_{K,X}(x)$ tends to be varying very slowly, contrarily, if the bandwidth is small, the function is more wiggly.

Compared to the the naive estimator, which was defined as the sum of the boxes centered on the observation x, the kernel estimator is the sum of the *bumps* centered on the observation x, the kernel function $K(.)$ determines the shape of the *bump*. One of the most popular kernels is the *Gaussian Kernel* defined by:

$$K(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-x^2}{2\sigma^2}. \tag{4.8}$$

The choice of the bandwidth is very important to the accuracy of the KDE model, in figure 4.3, we show the impact of the bandwidth on the smoothness of the resulting estimation.

Thus, estimating the bandwidth is not trivial, and it was widely studied in the literature since it highly impacts the outcome of the model.

Bandwidth Selection The problem of choosing the bandwidth parameter is a crucial issue that occurs often in the context of KDE. for optimal bandwidth

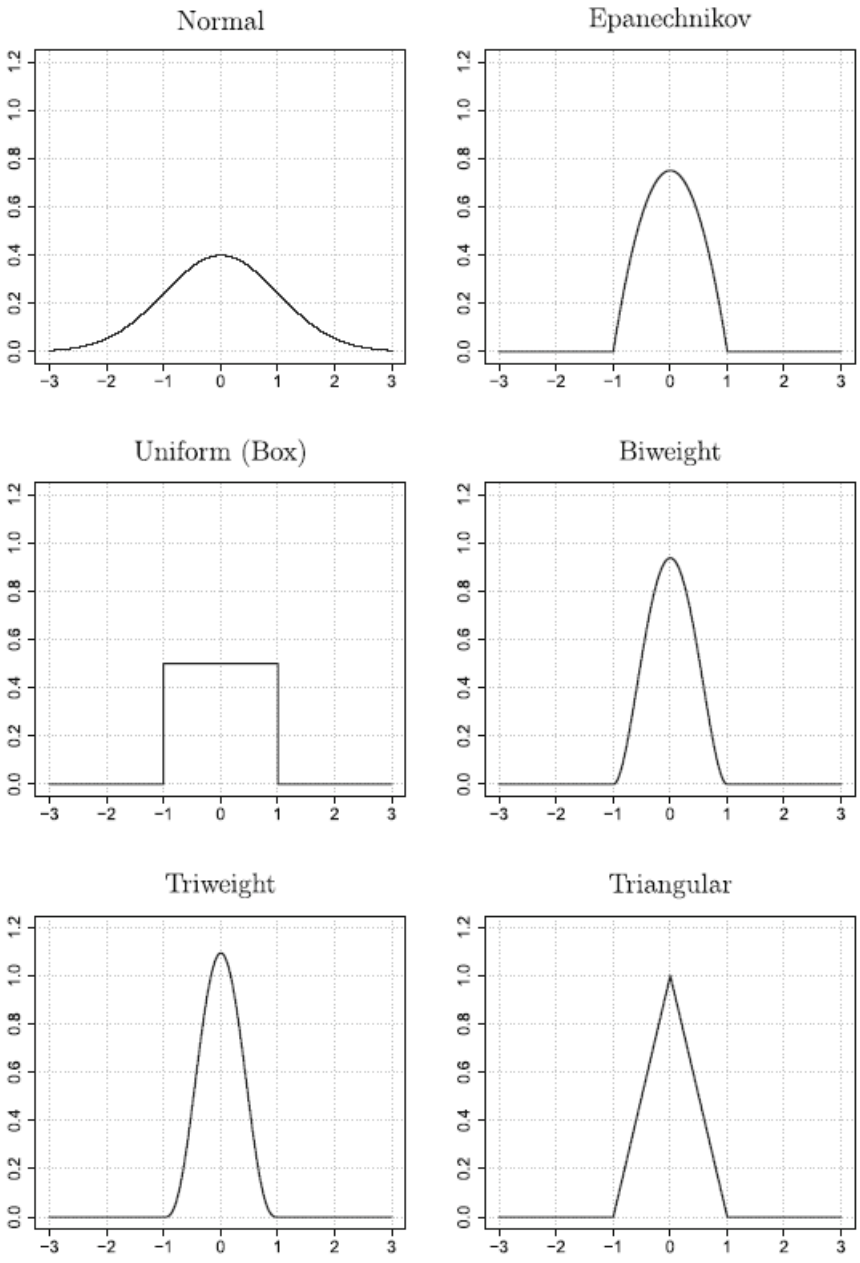


FIGURE 4.1: Kernel Function Plots, [85]

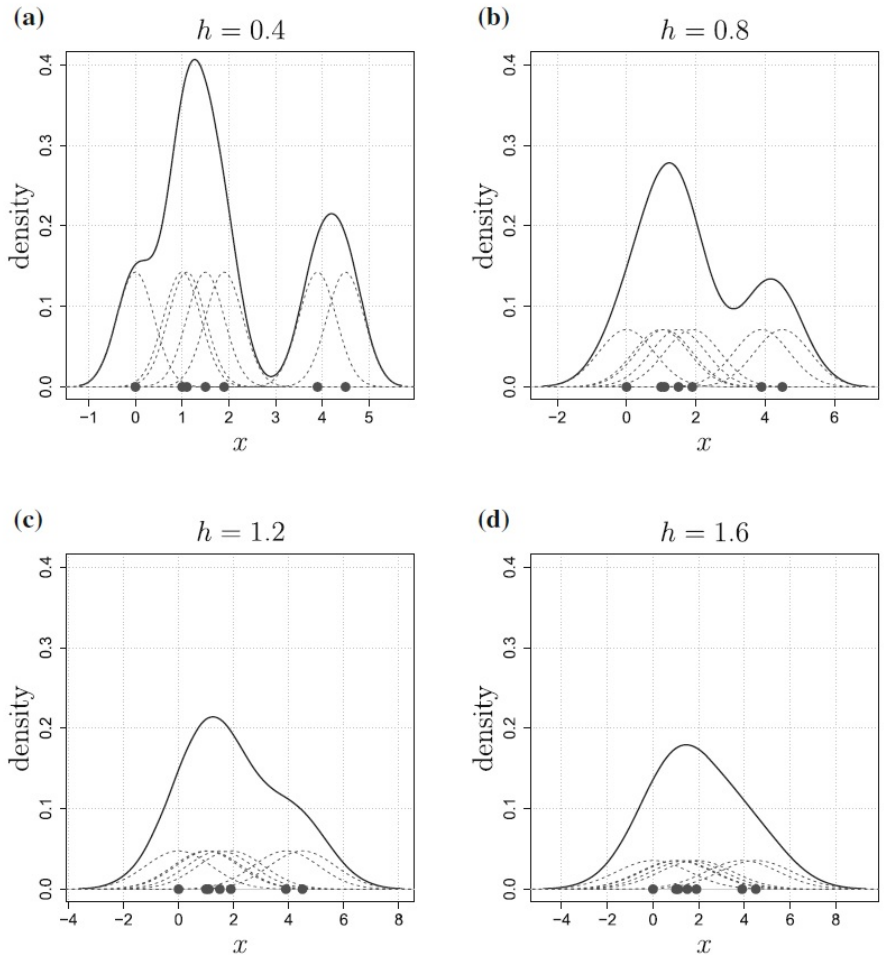


FIGURE 4.2: A toy example demonstrating the idea of the kernel density estimation with Gaussian kernels h refers to the different bandwidths, the optimal is $h = 0.8$. [85]

selection. The subject literature is abundant in many different solutions. Unfortunately, there does not exist one best method that can be applied universally. The accuracy of KDE depends very strongly on the bandwidth value. In the univariate case, the bandwidth is a scalar that controls the amount of smoothing. In the multivariate case, the bandwidth is a matrix and it controls both the amount and the orientation of smoothing. In our study we will be focusing on the univariate case, therefore we will give an overview only on the univariate bandwidth selectors. There are three main categories of these selectors:

- Methods using very simple and easy to compute mathematical formulas. They were developed to cover a wide range of situations, but do not guarantee that the result is close enough to the optimal (under certain criteria) bandwidth. They are often called the rules-of-thumb (ROT)

- Methods based on the notion of cross-validation (CV) that are based on a more precise mathematical footing. They require much more computational power, providing, however, the bandwidths that are more

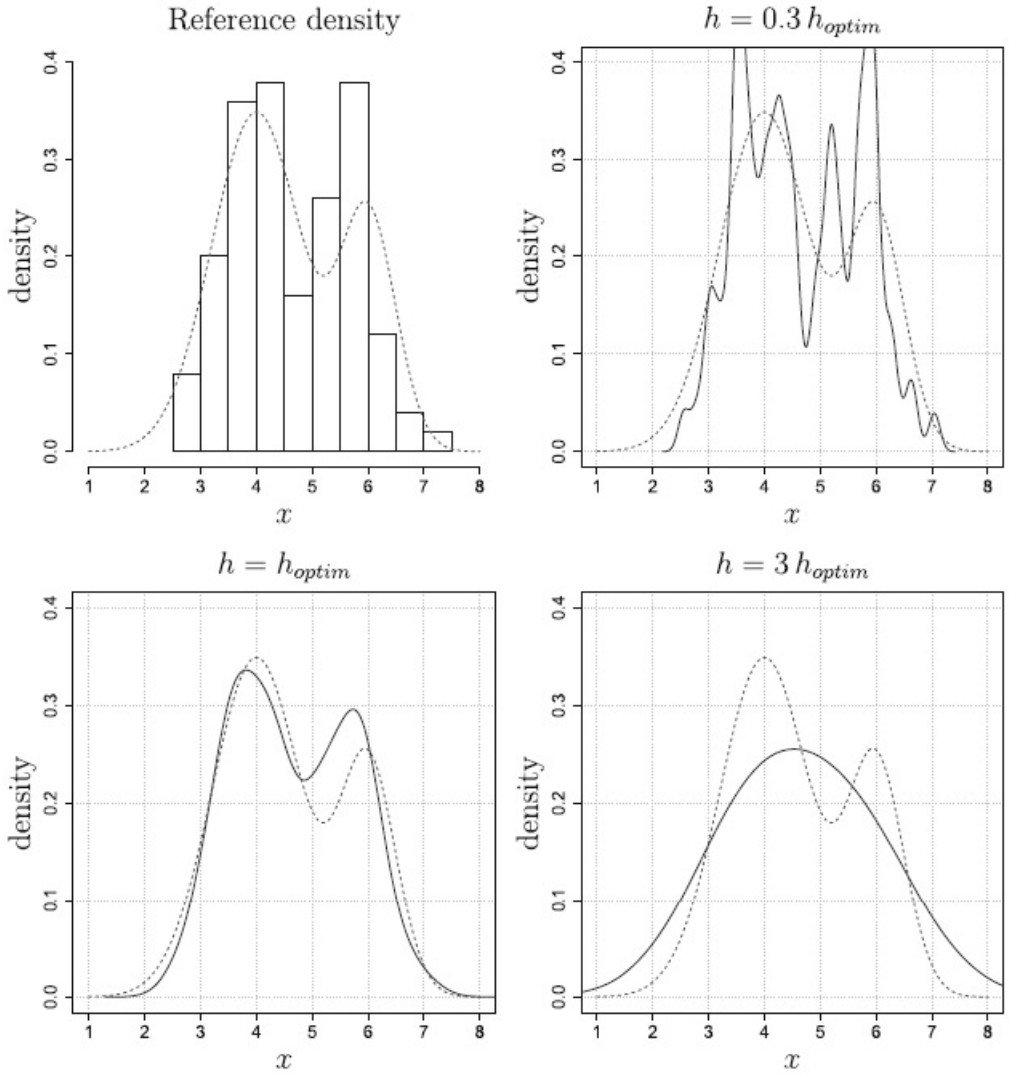


FIGURE 4.3: Three kernel density estimates with different bandwidths (too small (undersmoothing; small bias but large variability), optimal, and too big (oversmoothing; small variability but large bias)). The true density curve is plotted as the dashed line [85]

accurate for a wider class of density functions. Three classical variants of the CV are: least squares cross validation (LSCV), sometimes called unbiased cross validation (UCV), biased cross validation (BCV), and smoothed cross validation (SCV).

- Methods based on plugging in estimates of some unknown quantities that appear in formulas for the asymptotically optimal bandwidth. They are often called plugin (PI).

4.1.3 One Dimensional Clustering

There are two main approaches for clustering large sets of data, the subsampling and the approximation to a single representative vector. This techniques allow for adapting clustering algorithms suitable for small data sets to be applied on larger data sets [86].

- Subsampling: the subsample is small enough to fit into available memory and be clustered, those subsamples are clustered in parallel to achieve faster results. Once a clustering is obtained, the remaining data points can be assigned to the clusters with the closest centroid. The problem with this technique is that the subsample might not be a good representative of the data and therefore the clustering is considered inaccurate.

- Approximation to a single representative vector: needs one dimensional clustering in order to this can help with compression, or to speed up searching

As studied in the literature, 1D data can be very easy or difficult to cluster depending on the nature of the distribution it is following. For example, if it is linear with two clusters or more, multiple cut-off thresholds are needed to detect the groups within the dataset. If the data is non linear, we should fit it first to an appropriate distribution like the Gaussian and then find cut-off points based on the number of clusters you may have in the data.

The one dimensional clustering has many applications in retail market analysis, social networks analysis, microbiology, forecasting occurrence of rainfall, information systems, designing medical device, speech and language recognition, software packs for bioinformatics and other fields [87]. Many algorithms were improved by the researchers using the one dimensional clustering feature. We can cite the example of k-means largely detailed in the works of [87].

4.2 Experimentations

4.2.1 Datasets

Eight real-world datasets from the UCI machine learning repository are used in the experiment. The table below presents the main characteristics of these databases.

TABLE 4.2: Some Characteristics of Real-World Datasets

Datasets	#Instances	#Attributes	#Class
Ecoli	336	8	8
Electrical	10000	14	2
Glass	214	10	7
Page blocks	5473	10	5
Sat	4435	36	6
Spam	4601	57	2
Waveform	5000	21	3
Yeast	1484	8	10

4.2.2 Experimental Protocol

Based on the theoretical methods discussed in the previous section, we wanted to experiment with one dimensional clustering and Kernel Density Estimation as a way to deal with large data streams. The term curse of dimensionality refers to various phenomena and problems that arise when analyzing data in high-dimensional spaces that are usually absent in low-dimensional spaces (say, below four). In the context of KDE, the curse of dimensionality manifests itself as follows: an enormous amount of data is required to learn plausible probability density functions. In high dimensions data are extremely sparse and distance measure becomes meaningless. As argued above, the one dimensional clustering is an efficient method to deal with the dimensionality curse and to accomplish data compression. The data streams we are dealing with are supposed to be non linear so we have chosen to approximate them using the KDE and the Gaussian distribution.

The estimation of the probability density gives strong insights on the data features as skewness and multimodality. Usually, parametric estimation models (e.g. Gaussian distribution with unknown expectation and variance) are chosen to perform estimation but it was proven that they showed some poor behaviour especially if the process that generates the data is multimodal, then the aspects of the distribution can never be captured. Although when a good parametric model is not found, the non parametric models perform best. In the algorithm 5, we propose the anonymization procedure using KDE with 1D clustering. KDE helps identify the density of a data distribution. This can be useful in finding where a good number of data is grouped together and where it is not.

The KDE in one dimensional clustering allows for clusters to be detected using cut-off thresholds, in our work we use the local minimas and the local maximas as cut-off thresholds to determine the underlying clusters with the highest probability density. Local minimas of the resulting KDE estimation are used as the cluster regions and local maximas of the resulting KDE are used as the ptototypes (i.e. centers of the clusters).

Algorithm 5 1D Anonymization Approach

Input: D dataset to anonymize
Output: D' an anonymized dataset
1D Clustering step :
1: **for** each feature column of D **do**
2: Perform 1D Kernel Density Estimation.
3: Detect Cluster Regions by cutting at local minimas.
4: Encode the clusters using the clusters' centers i.e local maximas.
5: **end for**
Anonymization :
6: Reconstruct the dataset in D'.
7: Measure the Separability utility and the Structural utility of the anonymized data.

Local maxima are the peaks of the curves in the density plot. They can be easily discovered by checking the bins on either side. If it is the largest, it is a local maximum. If the curve is jaggy with too many maximas, we'll need to increase the number of neighbors sampled when computing density. The minimas are simply the bins of lowest density between two maximums. Local minima define where the clusters split. In the above example, we can see the data is split into four clusters on the left figure and three clusters on the right figure. We cut at the red markers. The green markers are our best estimates for the cluster centers. Each element is then recoded using the prototype of its corresponding cluster. In this way, we anonymize each attribute in data with respect to its peers characteristics which preserves the quality of information it is containing.

4.2.3 Experimental Results

Separability Utility preservation analysis

A kernel density estimation is intuitively seen as the a sum of "bumps". The size of the bump is the probability at the neighborhood of values around each data point. Each kernel has a bandwidth that determines the width of the bumps, the bigger the bandwidth the shorter and the wider the bump spreading out farther from the center. We estimate the bandwidth automatically depending on the size of the dataset.

The proposed Separability Utility measure is determined by putting the dataset through a real-case scenario. The original and the anonymized tables were learned to a decision tree model with 10 folds cross validation and their Separability Utility was measured. The results given in table 4.3, dress a comparison between the separability utility measures of the original and the anonymized datasets. The loss in information's quality after anonymization is minimal. Melior, the separability utility was improved in most of the cases. This can be explained by the fact that the clustering gathers together elements with the same features and therefore omits non decisive attributes from data.

TABLE 4.3: Impact of anonymization on Separability Utility (CI: confidence interval)

	Ecoli	Electrical	Glass	Sat
Original	82.1	99.5	69.2	85.05
95% CI	[73.1, 87.1]	[99.3, 99.7]	[61.3, 78.8]	[83.9, 86.2]
Attribute-oriented	80.1	98.8	70.05	84.9
95% CI	[77.4, 82.7]	[98.6, 99.1]	[58.6, 81.5]	[83.5, 86.4]
	Page Blocks	Waveform	Yeast	Spam
Original	96.6	75.6	81.2	92.2
95% CI	[95.9, 97.4]	[73.4, 77.6]	[75.5, 91.3]	[91.0, 93.3]
Attribute-oriented	95.5	75.5	83.4	92.1
95% CI	[95.2, 95.8]	[73.3, 77.6]	[78.6, 88.2]	[90.9, 93.2]

The method proposes a new data representation which makes it useful for data encryption as shown in table 4.3, the separability utility of the anonymized datasets is very good compared to its value before anonymization.

Structural Utility preservation analysis

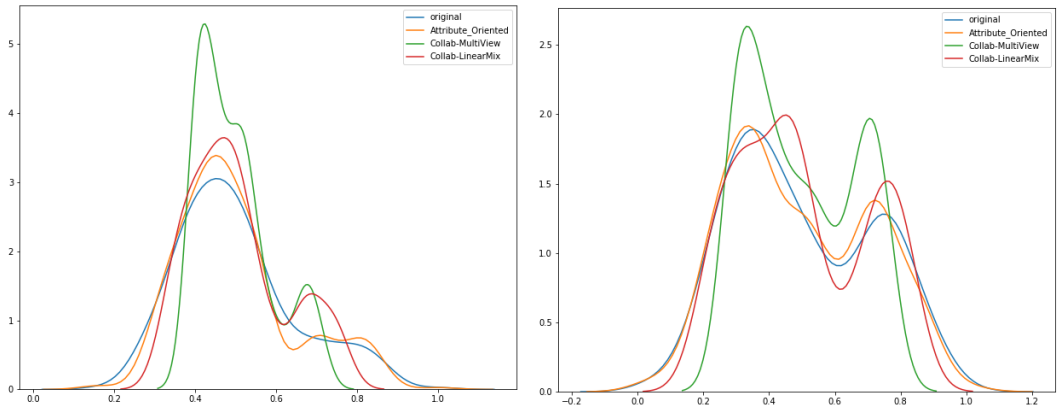


FIGURE 4.4: Probability Distribution of Attributes 2 and 6 of the Ecoli dataset using different approaches of data anonymization

The figure 4.4 shows the probability distribution of attributes 2 and 6 of the Ecoli dataset used in order to visualize the distribution of the anonymized datasets compared to the original data, this way, we can conclude that Attribute-oriented is the anonymization method that respects the original distribution of the attribute and thus, preserves the utility of the information its containing.

To quantify how much the anonymized datasets lost of their structural utility, we use the Earth Mover's Distance. This non-parametric distance is used to detect which of the anonymization methods is the most efficient in terms of its fidelity to the original dataset. The structural utility matrix of

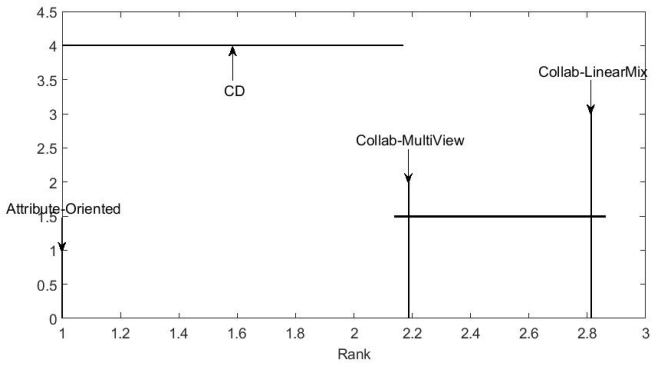
TABLE 4.4: Impact of anonymization on Structural Utility ($1 - W_1(P, Q)$)

	Ecoli	Electrical	Glass	Sat	PageBlocks	Waveform	Yeast	Spam
Attribute-oriented	1.00	1.00	1.00	0.97	0.99	1.00	1.00	1.00

table 4.4 is used as an input to the Friedman statistical test, we use this test to propose a ranking of the three anonymization approaches in terms of the structural utility and thus differentiating between them. The test works as follows:

- Calculate the EMD between the original and anonymized datasets attribute by attribute.
- Take the median of each distance vector corresponding to each of the anonymi-zation methods.
- Run the test on the matrix of medians i.e. the structural utility matrix.

FIGURE 4.5: Friedman test for comparing multiple approaches over multiple data sets



In figure 4.5, the critical diagram represents a projection of the average ranks methods on enumerated axis. The methods are ordered from left (the best) to right (the worst), in our case, the method of Attribute-oriented is the best and the worst is Collaborative SOM using the Linear Mixture of models. The Friedman test confirms what we implied from figure 4.4 and table 4.4. Attribute-oriented outperforms the other anonymization techniques since it respects the nature and the structure of each attribute.

To emphasis what was stated, we do PCA projections of the original and anonymized data with the three different methods as displayed in figure 4.6 and the ranking of the three anonymization methods is confirmed since the PCA projections show that the Attribute-oriented is the closest to the original

FIGURE 4.6: PCA of the anonymized waveform data

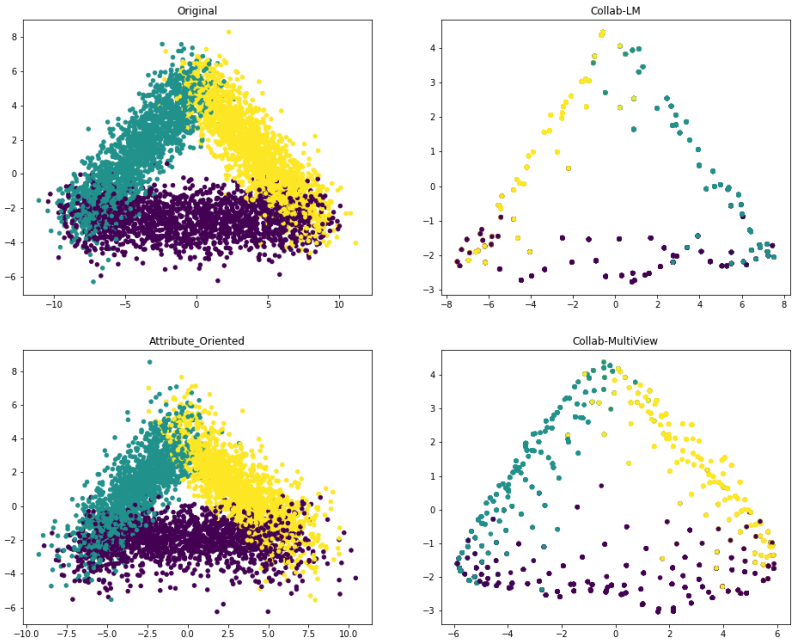


TABLE 4.5: Combined separability and structural utility Comb-Utility

	Ecoli	Electrical	Glass	Sat	PageBlocks	Waveform	Yeast	Spam	**Score**
Attribute-oriented	0.90	0.99	0.85	0.92	0.96	0.88	0.92	0.96	**7.79**
k-TCA	0.92	1.00	0.97	0.90	0.45	0.91	0.92	0.88	**7.35**
Constrained TCA	0.92	1.00	0.86	0.91	0.60	0.91	0.93	0.88	**7.40**

dataset. The Constrained TCA is the one with the fewest points due to the usage of constrained clustering with a k-anonymity level of 5, we obtained an anonymized dataset with many overlayed points. The figure shows that the best technique in terms of structural utility is Attribute-oriented and the worst one is Constrained TCA also the best technique in terms of privacy preservation is the Constrained TCA and the worst is Attribute-oriented.

From this combination, we conclude that the proposed $Attribute-oriented$ approach offers the best compromise. Indeed, it provides a good representation of the data allowing an efficient separability while respecting the distributional structure of the same data.

Table 4.5 summarize the clustering results of the proposed approaches in terms of combined utility ($Comb-Utility$). As it can be seen, our approach $Attribute-oriented$ generally performs best on all the datasets. To further evaluate the performance, we compute a measurement score by following

[78]:

$$Score(A_i) = \sum_j \frac{CombUtility(A_i, D_j)}{\max_i CombUtility(A_i, D_j)} \tag{4.9}$$

where $CombUtility(A_i, D_j)$ refers to the combined Utility value of A_i method on the D_j dataset. This score gives an overall evaluation on all the datasets, which shows our approach $Attribute - oriented$ outperforms the other methods substantially in most cases.

4.3 Discussion

The chapter gives a thourough analysis of the density based clustering methods, the KDE in particular. Attribute Oriented KDE can be seen under two perspectives: the first, as a performant non parametric model that combined with the one dimensional clustering can deal with the curse of dimensionality. The second is the ability of data compression to perform data anonymization or more specifically microaggregation.

Chapter 5

Incorporating discriminative power during anonymization process

After evaluating the different results of data anonymization using the methods in the previous works, we asked the question *What if data was labelled?* and *How the supervision can influence the obtained utility results?*. To answer these questions, in this chapter we will got through all the previously proposed approaches, and we added a second level of anonymization by incorporating the discriminative information and using Adaptive Weighting of Features to improve the quality of the anonymized data. This aims to improve the anonymized data quality without compromising its level of privacy.

5.1 Fundamental Concepts

5.1.1 Prototype based models in supervised learning

The subsection 3.1.1 have outlined a few prototype-based approaches to the unsupervised analysis high dimensional data. Usually, the aim of machine learning is to assign feature vectors to previously defined classes or categories. Supervised machine learning aims at extracting and representing information in terms of a hypothesis of the unknown classification rule or target function, which then can be generalized and applied to novel data in the working phase. Among the many machine learning frameworks specifically designed for supervised problems, prototype-based schemes constitute a family of very intuitive, easy to implement systems of great flexibility [60]. In the following subsection We will limit the discussion to classification problems and focus on Learning Vector Quantization (LVQ) family of algorithms that directly take on the basic ideas of competitive learning as presented in the chapter 3.

5.1.2 Learning Vector Quantization

Despite the fact that the Kohonen network is an unsupervised self organising learning paradigm, Kohonen does in fact make use of a supervised learning technique. This he describes as learning vector quantization. This is worth

mentioning because it amounts to a method for fine-tuning a trained feature map t o optimise its performance in altering circumstances. A typical situation may be that we wish to add new training vectors to improve the performance of individual neighbourhoods within the map.

The way this is achieved is by selecting training vectors (x) with known classification, and presenting them to the network to examine cases of misclassification. Again, a best-match comparison is performed at each node and the winner is noted (n,). The weight vector of the winning node is then modified [88].

LVQ is a pattern recognition model that takes advantage of the labels to improve the accuracy of the classification. The algorithms learns from a subset of patterns that best represent the training set.

The choice of the Learning Vector Quantization (LVQ) method was motivated by the simplicity and rapidity of convergence of the technique, since it is based on the hebbian learning. This is a prototype-based method that prepares a set of codebook vectors in the domain of the observed input data samples and uses them to classify unseen examples. Kohonen presented the self organizing maps as an unsupervised learning paradigm that he improves using a supervised learning technique, called the learning vector quantization. It is a method used for optimizing the performances of a trained map in a reward-punishment scheme.

Learning Vector Quantization was designed for classification problems that have existing data sets that can be used to supervise the learning by the system. LVQ is non-parametric, meaning that it does not rely on assumptions about that structure of the function that it is approximating. Euclidean distance is commonly used to measure the distance between real-valued vectors, although other distance measures may be used (such as dot product), and data specific distance measures may be required for non-scalar attributes. There should be sufficient training iterations to expose all the training data to the model multiple times. The learning rate is typically linearly decayed over the training period from an initial value until it is close to zero. Multiple passes of the LVQ training algorithm are suggested for more robust usage, where the first pass has a large learning rate to prepare the codebook vectors and the second pass has a low learning rate and runs for a long time (perhaps 10-times more iterations).

A typical situation is adding new training vectors to improve the performance of individual neighbourhoods within the map by selecting training vectors (x) with known classification. Learning them afterwards to the network to examine the cases of misclassification. A comparison is performed at each node of the map and the weight vector of the winning node is then modified following this criteria, the winner is noted m_c.

In the Learning Vector Quatization model, each class contains a set of fixed prototypes with the same dimension of the data to be classified. LVQ adaptively modifies the prototypes. In the learning algorithm, data is first clustered using a clustering method and the clusters' prototypes are moved using LVQ to perform classification. We chose to supervise the results of the clustering by moving the center clusters' using the wLVQ2 proposed in

Algorithm 6 Adaptive Weighting of Pattern Features During Learning

Initialization :
Initialize the matrix of weights W according to :

$$w_j^i = \begin{cases} 0, \ when \ \text{i}\neq\text{j} \\ 1, \ when \ \text{i}=\text{j} \end{cases}$$

The codewords **m** are chosen for each class using the k-means algorithm.
Learning Phase:

1. Present a learning example x.
2. Let $m_i \in C_i$ be the nearest codeword vector to x.
   - **if** $x \in C_i$, then go to 1
   - **else then**
     - let $m_j \in C_j$ be the second nearest codeword vector
     - **if** $x \in C_j$ then
       * a symmetrical window *win* is set around the mid-point of m_i and m_j.
       * **if** x falls within *win*, then
         Codewords Adaptation:
       * m_i is moved away from x according to the formula

$$m_i(t+1) = m_i(t) + \alpha(t)[Wx(t) - m_j(t)]$$

       * m_j is moved closer x according to the formula

$$m_j(t+1) = m_j(t) - \alpha(t)[Wx(t) - m_j(t)]$$

       * for the rest of the codewords

$$m_k(t+1) = m_k(t)$$

         Weighting Patterns features:
       * adapt w_k^k according to the formula:

$$w_k^k(t+1) = w_k^k(t) - \beta(t)x^k(t)(m_i^k(t) - m_j^k(t))$$

       * go to 1.

Where $\alpha(t)$ and $\beta(t)$ are the learning rates

algorithm 6 for each of the approaches. One of the interesting weighting

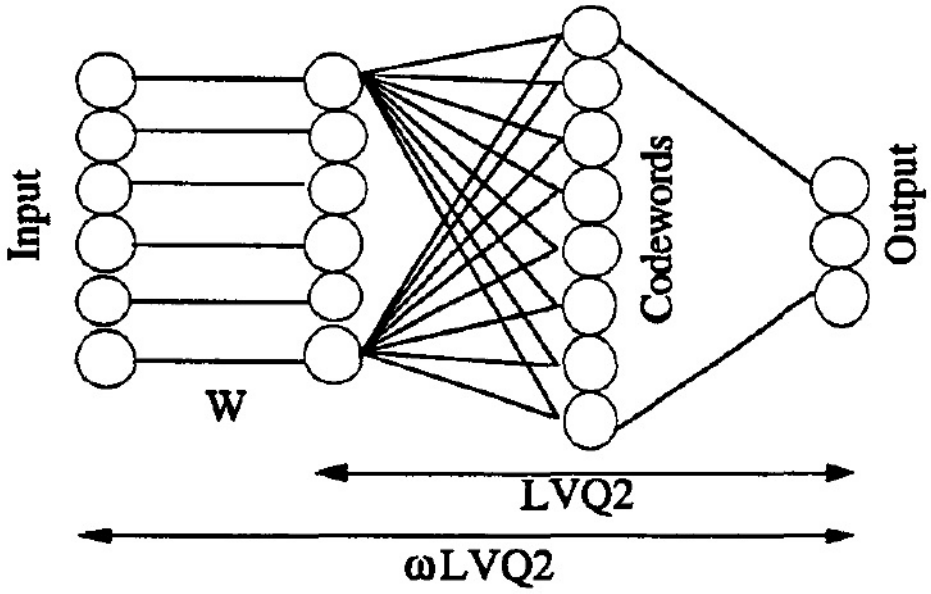


FIGURE 5.1: The wLVQ2 Architecture. (Picture credits: [89])

approach in the supervised learning is the wLVQ2 [89] since this upgraded version of the LVQ respects the characteristics of each features and adapts the weighting of each feature according to its participation to the discrimination. The system learns using two layers: the first layer calculates the weights of the features and then it is presented to the LVQ2 algorithm.

The cost function of this algorithm can be written as follows:

$$R_{wLVQ2}(x, m, W) = \begin{cases} \|Wx - m_j\|^2 - \|Wx - m_i\|^2, & If \quad C_k = C_j \\ 0, \ otherwise \end{cases}$$

Where $x \in C_k$ and W is the weighting coefficient matrix; m_i is the nearest codeword vector to Wx and m_j is the second nearest codeword vector to Wx. The wLVQ2 with the Collaborative Paradigm enhances the utility of the anonymized data by the k-TCA and the Constrained TCA (C-TCA) models, the use of wLVQ2 is done after the collaboration between cluster centers' to improve the results of the Collaboration at the pre-anonymization and the anonymization steps.

The experimental protocol of using wLVQ2 with Attribute-oriented data anonymization and Kernel Density Estimation, takes in account the labels of the dataset and improves the found prototypes and then represents the micro-clusters using them.

Algorithm 7 Supervised Attribute Oriented Anonymization Approach

Input: $X = x_{ij}, i = 1..n, j = 1..d$; a dataset to anonymize
Output: $X' = x'_{ij}$; an anonymized dataset

1D Clustering step :

1: **for** $x_{ij}, j = 1..d, \forall i$ **do**
2: Perform 1D Kernel Density Estimation.

$$f_{N,X}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_{ij}}{h}\right),$$

3: Microaggregation using the local maximas.
4: **end for**

Incorporating Discriminative Information wLVQ2 :

5: Algorithm 6
6: Coding the element with its nearest neighbor.
7: Measure the utility of the anonymized data.

5.2 Experimental Validation

5.2.1 Datasets

Six datasets from the UCI machine learning repository are used in the experiment. The table below presents the main characteristics of these databases.

TABLE 5.1: Some Characteristics of Datasets

Datasets	#Instances	#Attributes	#Class
Ecoli	336	8	8
Electrical	10000	14	2
Glass	214	10	7
Page blocks	5473	10	5
Waveform	5000	21	3
Yeast	1484	8	10

5.2.2 Quality Validity Indices

Cluster validity consists of techniques for finding a set of clusters that best fits natural partitions without any a priori class information. The outcome of the clustering process is validated by a cluster validity index. Internal validation measures reflect often the compactness, the connectivity and the separation of the cluster partitions. We choose to validate the results of the proposed methods using Silhouette Index and Davies Bouldin Index. The results are given in the tables 5.2 and 5.3

TABLE 5.2: Silhouette Index

	Ecoli	Electrical	Glass	Page Blocks	Yeast	Waveform
$k-TCA$	0.26	-0.05	0.42	-0.40	0.13	0.18
$k-TCA^{++}$	0.89	0.08	0.59	-0.49	0.84	0.24
$C-TCA$	0.24	-0.05	0.43	-0.34	0.07	0.13
$C-TCA^{++}$	0.84	0.08	0.45	-0.43	0.81	0.25
KDE	0.26	0.069	-0.19	-0.54	0.28	-0.27
KDE^{++}	0.99	0.99	0.57	0.98	0.99	1

As illustrated, the Attribute oriented microaggregation using wLVQ2 (++: Discriminative version of each approach, KDE^{++}, k-TCA^{++},C$-TCA^{++}$) outperforms by far the Attribute Oriented microaggregation in both Silhouette and Davies Bouldin indices.

TABLE 5.3: Davies Bouldin Index

	Ecoli	Electrical	Glass	Page Blocks	Yeast	Waveform
$k-TCA$	2.68	2.28	0.40	3.23	2.31	1.51
$k-TCA^{++}$	0.59	3.38	0.40	3.11	0.24	1.37
$C-TCA$	1.61	2.58	0.55	3.04	2.95	1.92
$C-TCA^{++}$	0.14	3.38	0.51	3.10	0.26	1.35
KDE	0.57	3.96	4.99	3.83	2.43	6.96
KDE^{++}	9.91E-08	0.02	1.32	0.52	4.20E-08	3.58E-06

5.2.3 Combined Utility Measure

Separability Utility

To measure the utility of the anonymized datasets we propose a test on the original and the anonymized data. The test consists of comparing the accuracy of a decision tree model with 10 folds cross validation before and after microaggregation to evaluate the practicality of the proposed anonymization. We call it separability utility since it measures the separability of the clusters. We give the results of this measure in table 5.4, we also provide a comparison between the separability utility measures of the original and the anonymized datasets.

The separability measure was improved after LVQ for 83% of the tests done on the datasets, this can be explained by the tendency of microaggregation to remove non decisive attributes from the dataset in order to gather together elements that are similar. The $^{++}$ in the name of the methods refers to discriminant version.

TABLE 5.4: Separability Utility

Datasets	Glass	Ecoli	Electrical	PageBlocks	Waveform	Yeast
Original	0.692	0.821	0.995	0.966	0.748	0.812
$k-TCA$	0.943	0.845	0.999	0.905	0.83	0.862
$k-TCA^{++}$	0.944	0.988	0.735	0.919	0.884	1
$C-TCA$	0.747	0.848	0.999	0.915	0.816	0.876
$C-TCA^{++}$	0.859	0.863	0.745	0.918	0.884	0.887
KDE	0.701	0.801	0.988	0.955	0.755	0.834
KDE^{++}	0.743	0.806	0.982	0.962	0.758	0.84

Structural Utility using the Earth Mover's Distance

We believe that measuring the distance between two distributions is the way to evaluate the difference between the datasets. The amount of utility lost in the process of anonymization can be see as the distance between the anonymized dataset and the original one.

The Earth Mover's distance (EMD) also known as the Wasserstein distance [77], extends the notion of distance between two single elements to that of a distance between sets or distributions of elements. It compares the probability distributions P and Q on a measurable space (Ω, Ψ) and is defined as follows (We are using the distance of order 1):

$$W_1(P,Q) = \inf_{\mu}\{\int_{\Omega\times\Omega} |x-y|\mathrm{d}\mu(x,y)\| \begin{array}{l}\mu \text{ :prob. measure on } (\Omega\times\Omega, \Psi\otimes\Psi)\\ \textit{with marginals} : P, Q\end{array}\} \tag{5.1}$$

where $\Omega \times \Omega$ is the product probability space. Notice that we may extend the definition so that P is a measure on a space (Ω, Ψ) and Q is a measure on a space (Ω', Ψ').

Let us examine how the above is applied in the case of discrete sample spaces. For generality, we assume that P is a measure on (Ω, Ψ) where $\Omega = \{x_i\}_{i=1}^{n}$ and Q is a measure on (Ω', Ψ') where $\Omega' = \{y_i\}_{j=1}^{n'}$ - the two spaces are not required to have the same cardinality.

Then, the distance between P and Q becomes:

$$W_1(P,Q) = \inf_{\{\lambda_{i,j}\},i,j}\left\{\sum_{i=1}^{n}\sum_{j=1}^{n'}\lambda_{i,j}|x_i-y_j| : \sum_{i=1}^{n}\lambda_{i,j} = q_j, \sum_{j=1}^{n'}\lambda_{i,j} = p_i, \lambda_{i,j} \geq 0\right\}$$

Preserving combined utility

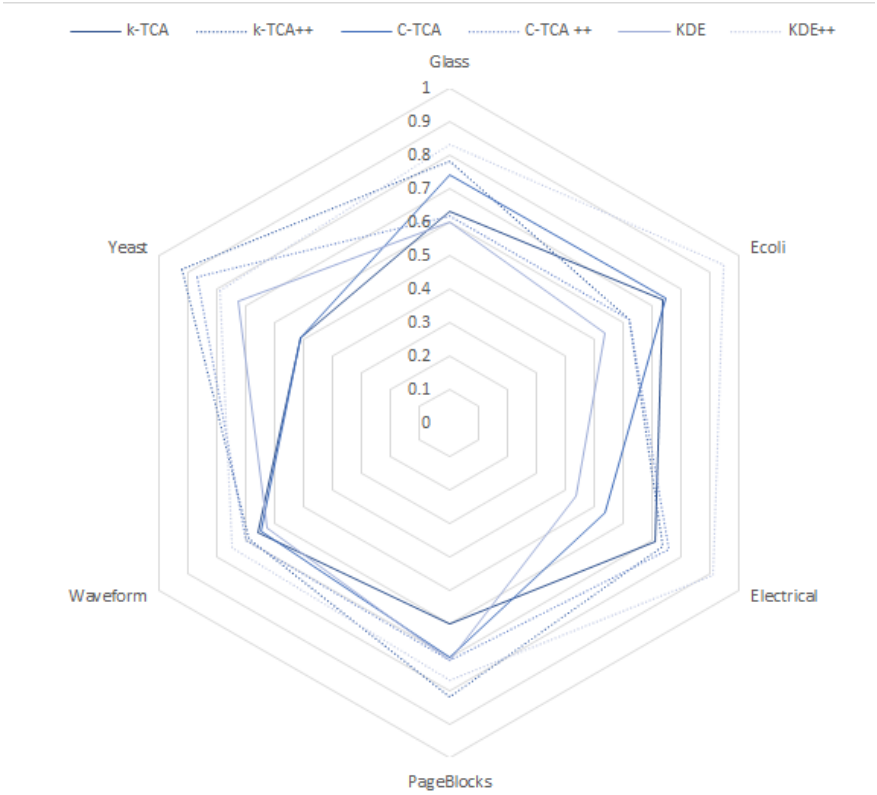


FIGURE 5.2: The combined utility of the six datasets using the six methods using the parameter $\alpha = 0.5$

As shown in the table 5.2, the introduction of the discriminant information improves the utility of the anonymized datasets for all of the methods proposed.

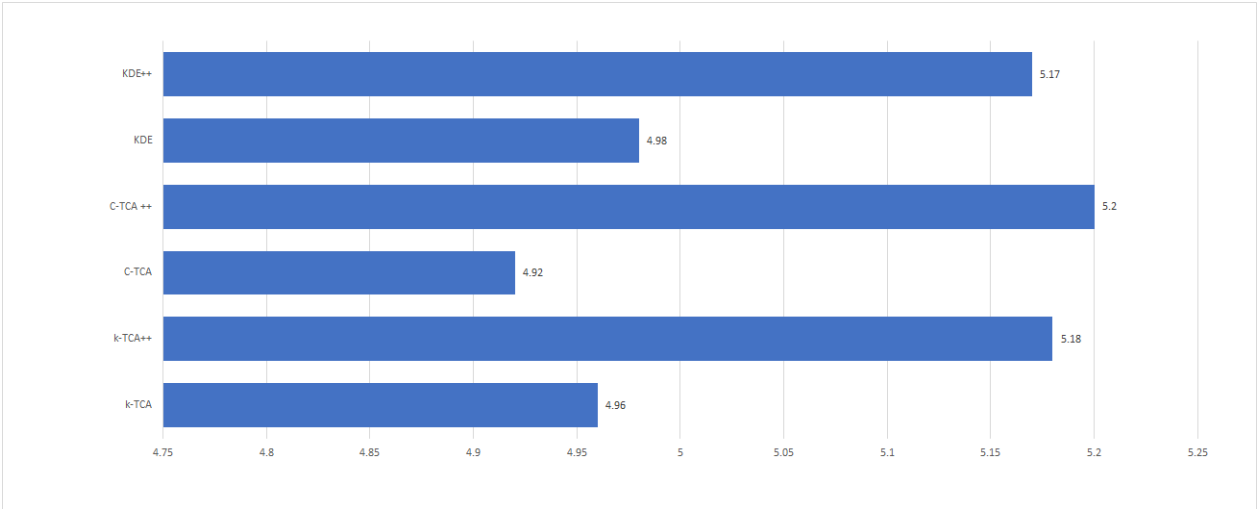


FIGURE 5.3: Score of the six proposed methods

5.3 Discussion

In this chapter we studied the impact of incorporating the discriminative information to improve data anonymization level and to preserve its usefulness. The anonymization is achieved in two levels process. The first, uses one of these three methods: k-TCA or Constrained TCA (C-TCA) or Attribute Oriented KDE, that we introduced for data anonymization through microaggregation approach. And the second, through the use of labels and the learning of the vectors weights adaptively using the weighted LVQ. The experimental investigation shown above prove the efficiency of the methods and illustrate its importance. The main contributions of the article are the addition of the supervised learning layer to improve utility of the model without compromising its anonymity. The separability utility reflects the usefulness of the data and the structural utility shows its level of anonymity. The combined utility is a weighted measure that combines both measures, we can change the weight of the utility tradeoff depending on wich side we want to emphasise on.

Chapter 6

Conclusion and Future Work

The broad purpose of this thesis was to use unsupervised machine learning to achieve data anonymization. After a thorough study of the state of the art, the main limitation of the existing approaches to achieve data anonymization with clustering was the lack of utility in anonymized data. We experimented with different clustering methods, prototype-based clustering, density based clustering and multi-view clustering. we even incorporated the discriminant information as a way to use the prototype based supervised learning to add another level of data security. Our focus was mainly on improving the trade-off between the utility and the anonymity of a protected dataset. Our main contributions:

- The first contribution was the use of collaborative clustering and multi-view clustering together with self-organizing maps to achieve data privacy. As we know that the Multi-view clustering is a great way to deal with multisource data and high dimensional elements. And the collaborative clustering improves the quality of the clustering and as a result, the level of utility of the output data. The results of the approach, k-TCA, were encouraging but there were some limits concerning the k level of anonymization; the level was given automatically by the map and depended completely on the quality of the clustering and how many neurons were captured by each cell. [Chapter 3]

- We then tweaked the approach and added the constrained of having at least k element by cluster. This method, Constrained TCA, outperformed the k-TCA in terms of accuracy, that we called later the Separability Utility, and gave the user the possibility to decide for the level of anonymity himself. In this approach we also used the linear mixture of SOM models as a way to recode the elements of the same cluster instead of just using the winning neuron (the best matching unit). This improved the accuracy of the model and gave better anonymity results. Also the trade-off between the accuracy level and the anonymity level was easier to measure and was acceptable compared to the k-TCA. The k-TCA and the constrained TCA are both a major contribution to the field of distributed collaborative anonymization, since the anonymization on those two is achieved in a distributed manner using the multi-view clustering and in a collaborative way as those views collaborate with each other to improve the clustering quality [Chapter 3]

- The improvement of the utility trade-off achieved by the multi-view clustering was encouraging to explore with one-dimensional clustering. we believe that approximating the distribution of a feature helps preserve its main characteristics and makes better assumptions about the information its containing. Also density based clustering is an intuitive way to do it. The Kernel Density Estimation along with one dimensional clustering are computationally lighter since the theoretical complexity of the 1D KDE is approximated to $O(n)$. This approach might be executed in parallel for all the variables and this inspired us to introduce a new measure of utility that we called the Structural utility since it uses the Earth Mover's Distance to measure the difference between the original data and the protected data.The closerthe lower the privacy. we also introduced a weighted combination of the separability utility (the accuracy of the anonymized dataset) and the structural utility to give the user the ability to express the trade-off in a simple manner. [Chapter 4]

- Since the datasets we were testing with, were all labelled, we wanted to explore more on the question of anonymizing data by adding another layer of data protection by incorporating the discriminant information. A prototype-based supervised learning method seemed like a good track considering the simplicity of its implementation along with its good performance. Particularly, we used a weighted version of the Learning Vector Quantization called wLVQ2, this method has the advantage of achieving weighted feature selection instead of blindly performing LVQ. This model, when added to the previously presented models, improved the results of the accuracy since we had more information about the feature of the dataset and only the relevant features were used to achieve anonymization.[Chapter 5]

The contributions of this thesis open several avenues for new research to be pursued:

- As we introduced the Collaborative Data Anonymization and we used a form of aggregation to output a protected dataset. This aggregation might be done using a secure aggregation protocol for Federated Learning and Diferentially Private Machine Learning. The problem of differential privacy is the same of data anonymization, we are still trying to improve the utility of the data anonymized while protecting it from privacy breaches. The noise added to attain a differentially private dataset is large and this damages the utility of the output data, thus some transformations should be applied to the query to reduce its sensitivity. It could be interesting to test if the collaboration paradigm along with microaggregation can help reducing the sensitivity of the queries.

- In this research we mainly proposed Privacy Preserving methods for data holders, this limits the right of individuals on their data. we believe that owners should have the right to manage and protect their data their own way. It would be interesting to think of models that

provide some tailored privacy preserving tools or protocols to give the user the ability to decide for the level of anonymity or protection that he wants.

- We also want to experiment with real world applications of data anonymization in the fields of IoT and healthcare data. Since the type of data collected in those fields is very sensitive and its protection is mandatory as we explained in the chapter 2.It will be interesting to experiment with real applications of the Privacy Preservation Microagregation and provide more measurements to assess its efficiency and the quality of the output data.

- For the collaborative clustering scheme, we would like to extend the usage of the SOMs in the local phase to consider different clustering model to each view of the dataset (to each source of the multisource data). We would like to achieve the Collaborative microaggregation by applying it in the distributed framework. The question will be how to achieve collaboration between different clustering models of different hyperparameters?

Chapter 7

Personal Publications

The publications supporting this thesis are:
International Journals

- Sarah Zouinina, Younès Bennani, Nicoleta Rogovschi, Abdelouahid Lyhyaoui: Data Anonymization through Collaborative Multi-view Microaggregation, Journal of Intelligent Systems, De Gruyter Poland Ltd. (2020)

- Sarah Zouinina, Younès Bennani, Maha Ben-Fares, Nicoleta Rogovschi, Abdelouahid Lyhyaoui: Preserving Utility during Attribute-oriented Data Anonymization Process. Australian Journal of Intelligent Information Processing Systems 16(3): 25-35 (2019)

- Nicoleta Rogovschi, Sarah Zouinina, Basarab Matei, Issam Falih, Nistor Grozavu, Seiichi Ozawa: t- Distributed Stochastic Neighbor Embedding Spectral Clustering using higher order approximations. Australian Journal of Intelligent Information Processing System 17(1): 78-86 (2019)

International Conferences:

- Sarah Zouinina, Nistor Grozavu, Younès Bennani, Abdelouahid Lyhyaoui, Nicoleta Rogovschi: Efficient k-Anonymization through Constrained Collaborative Clustering. SSCI 2018: 405-411

- Sarah Zouinina, Nistor Grozavu, Younès Bennani, Abdelouahid Lyhyaoui, Nicoleta Rogovschi: A Topological k-Anonymity Model Based on Collaborative Multi-view Clustering. ICANN (3) 2018: 817-827

- Sarah Zouinina, Younès Bennani, Nicoleta Rogovschi, Abdelouahid Lyhyaoui: A Two-Levels Data Anonymization Approach. In the Springer IFIP AICT series, AIAI (1) 2020: 85-95

- Ikram Chairi, Amina El Gonnouni, Sarah Zouinina, Abdelouahid Lyhyaoui: Prediction of Firm's Creditworthiness Risk using Feature Selection and Support Vector Machine, SIS 2017: 285-293

Bibliography

[1] N. Venkataramanan and A. Shriram, *Data Privacy: Principles and Practice*. Chapman & Hall/CRC, 2016.

[2] B. Raghunathan, *The Complete Book of Data Anonymization: From Planning to Implementation*. Boston, MA, USA: Auerbach Publications, 2013.

[3] K. El Emam and L. Arbuckle, *Anonymizing Health Data: Case Studies and Methods to Get You Started*, 1st. O'Reilly Media, Inc., 2013.

[4] R. Agrawal and R. Srikant, "Privacy-preserving data mining", in *ACM Sigmod Record*, ACM, vol. 29, 2000, pp. 439–450.

[5] Z. Huang, W. Du, and B. Chen, "Deriving private information from randomized data", in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, ACM, 2005, pp. 37–48.

[6] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 571–588, 2002. (visited on 04/28/2017).

[7] J.-W. Byun, A. Kamra, E. Bertino, and N. Li, "Efficient k-anonymization using clustering techniques", in *International Conference on Database Systems for Advanced Applications*, Springer, 2007, pp. 188–200.

[8] A. Asuncion and D. Newman, *UCI Machine Learning Repository*, 2007.

[9] R. J. Bayardo and R. Agrawal, "Data privacy through optimal k- anonymization", in *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, IEEE, 2005, pp. 217–228.

[10] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito: Efficient full-domain k-anonymity", in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, ACM, 2005, pp. 49–60.

[11] ——, "Mondrian multidimensional k-anonymity", in *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, IEEE, 2006, pp. 25–25. (visited on 04/28/2017).

[12] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity", *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 3, 2007.

[13] D. E. Robling Denning, *Cryptography and Data Security*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1982.

[14] A. Shamir, "How to share a secret", *Commun. ACM*, vol. 22, no. 11, Nov. 1979.

[15] M. Alharby and A. van Moorsel, "Blockchain-based smart contracts: A systematic mapping study", *CoRR*, vol. abs/1710.06372, 2017.

[16] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", 2009.

[17] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger", *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, 2014.

[18] A. Sharma, G. Singh, and S. Rehman, "A review of big data challenges and preserving privacy in big data", in *Advances in Data and Information Sciences*, M. L. Kolhe, S. Tiwari, M. C. Trivedi, and K. K. Mishra, Eds., Singapore: Springer Singapore, 2020, pp. 57–65.

[19] C. K. Liew, U. J. Choi, and C. J. Liew, "A data distortion by probability distribution", *ACM Trans. Database Syst.*, vol. 10, pp. 395–411, 1985.

[20] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "On the privacy preserving properties of random data perturbation techniques", in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, IEEE, 2003, pp. 99–106.

[21] J. J. Kim, J. J. Kim, W. E. Winkler, and W. E. Winkler, "Multiplicative noise for masking continuous data", Statistical Research Division, US Bureau of the Census, Washington D.C, Tech. Rep., 2003.

[22] K. Liu, H. Kargupta, and J. Ryan, "Random projection-based multiplicative data perturbation for privacy preserving distributed data mining", *IEEE Transactions on knowledge and Data Engineering*, vol. 18, no. 1, pp. 92–106, 2006.

[23] T. Dalenius and S. P. Reiss, "Data-swapping: A technique for disclosure control", *Journal of statistical planning and inference*, vol. 6, no. 1, pp. 73–85, 1982.

[24] C. C. Aggarwal and S. Y. Philip, "A general survey of privacy-preserving data mining models and algorithms", in *Privacy-preserving data mining*, Springer, 2008, pp. 11–52.

[25] B. Raghunathan, *The Complete Book of Data Anonymisation: From Planning to Implementation*. CRC Press, 2013.

[26] P. Samarati, "Protecting respondents identities in microdata release", *IEEE transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[27] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: K-anonymity and its enforcement through generalization and suppression", Technical report, SRI International, Tech. Rep., 1998.

[28] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity", *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 3, 2007.

[29] N. Li, T. Li, and S. Venkatasubramanian, "T-closeness: Privacy beyond k-anonymity and l-diversity", in *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, IEEE, pp. 106–115.

[30] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data", in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2002, pp. 639–644.

[31] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy preserving mining of association rules", *Information Systems*, vol. 29, no. 4, pp. 343–364, 2004.

[32] A. Friedman, R. Wolff, and A. Schuster, "Providing k-anonymity in data mining", *The VLDB Journal*, vol. 17, no. 4, pp. 789–804, 2008. (visited on 04/28/2017).

[33] Y. Lindell and B. Pinkas, "Privacy preserving data mining", in *Annual International Cryptology Conference*, Springer, 2000, pp. 36–54.

[34] J. Vaidya, M. Kantarcıoğlu, and C. Clifton, "Privacy-preserving naive bayes classification", *The VLDB Journal*, vol. 17, no. 4, pp. 879–898, 2008.

[35] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain, *Simultaneous Feature Selection and Clustering Using Mixture Models*.

[36] J. Zhang, D.-K. Kang, A. Silvescu, and V. Honavar, "Learning accurate and concise naive bayes classifiers from attribute value taxonomies and data", *Knowl. Inf. Syst.*, vol. 9, pp. 157–179, 2006.

[37] A. Inan, M.Kantarcioglu, and E. Bertino, "Using anonymized data for classification. In: Data Engineering", IEEE 25th International Conference, 2009, pp. 429–440.

[38] J. Li, R. C.-W. Wong, A. W.-C. Fu, and J. Pei, "Achieving k-anonymity by clustering in attribute hierarchical structures", in *International Conference on Data Warehousing and Knowledge Discovery*, Springer, 2006, pp. 405–416.

[39] G. Loukides and J. Shao, "Capturing data usefulness and privacy protection in k-anonymisation", in *Proceedings of the 2007 ACM symposium on Applied computing*, ACM, 2007, pp. 370–374.

[40] J.-L. Lin and M.-C. Wei, "An efficient clustering method for k-anonymization", in *Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society*, ser. PAIS '08, Nantes, France: ACM, 2008, pp. 46–50.

[41] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E. S. Nordholt, K. Spicer, and P.-P. De Wolf, *Statistical disclosure control*. John Wiley & Sons, 2012.

[42] A. Rodríguez-Hoyos, J. Estrada-Jiménez, D. Rebollo-Monedero, J. Parra-Arnau, and J. Forné, "Does *k*-anonymous microaggregation affect machine-learned macrotrends?", *IEEE Access*, vol. 6, pp. 28 258–28 277, 2018, ISSN: 2169-3536.

[43] D. J. D. K. LeFevre and R. Ramakrishnan, "Workload-aware anonymization", *in KDD'06*, pp. 277–286, 2006.

[44] J. Domingo-Ferrer and V. Torra, "Ordinal, continuous and heterogeneous k-anonymity through microaggregation", *Data Mining and Knowledge Discovery*, vol. 11, no. 2, pp. 195–212, 2005.

[45] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments", *ACM Computing Surveys*, vol. 42, no. 4, pp. 1–53,

[46] J. Domingo-Ferrer and V. Torra, "Disclosure control methods and information loss for microdata", *Confidentiality, disclosure, and data access: theory and practical applications for statistical agencies*, pp. 91–110, 2001.

[47] T. de Waal and L. Willenborg, "Information loss through global recoding and local suppression", *Netherlands Official Statistics*, vol. 14, pp. 17–20, 1999.

[48] E. Bertino, D. Lin, and W. Jiang, "A survey of quantification of privacy preserving data mining algorithms", in *Privacy-preserving data mining*, Springer, 2008, pp. 183–205.

[49] V. S. Iyengar, "Transforming data to satisfy privacy constraints", in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2002, pp. 279–288.

[50] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey", *ACM Computing Surveys (CSUR)*, vol. 51, no. 3, pp. 1–38, 2018.

[51] C. Andersson and R. Lundin, "On the fundamentals of anonymity metrics", in *IFIP International Summer School on the Future of Identity in the Information Society*, Springer, 2007, pp. 325–341.

[52] P. Syverson, "Why i'm not an entropist", in *International Workshop on Security Protocols*, Springer, 2009, pp. 213–230.

[53] M. K. Reiter and A. D. Rubin, "Crowds: Anonymity for web transactions", *ACM Trans. Inf. Syst. Secur.*, vol. 1, no. 1, pp. 66–92, Nov. 1998.

[54] T. Kohonen, *Self-organizing Maps*. Springer Berlin, 2001.

[55] F. Bação, V. Lobo, and M. Painho, "Self-organizing maps as substitutes for k-means clustering", in *Computational Science – ICCS 2005*, V. S. Sunderam, G. D. van Albada, P. M. A. Sloot, and J. Dongarra, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 476–483.

[56] N. Grozavu and Y. Bennani, "Topological Collaborative Clustering", *in LNCS Springer of ICONIP'10 : 17th International Conference on Neural Information Processing*, 2010.

[57] M. Ghassany, N. Grozavu, and Y. Bennani, "Collaborative multi-view clustering", in *The 2013 International Joint Conference on Neural Networks, IJCNN 2013, Dallas, TX, USA, August 4-9, 2013*, 2013, pp. 1–8.

[58] D. O. Hebb, *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

[59] T. Kohonen, M. R. Schroeder, and T. S. Huang, *Self-Organizing Maps*. Springer-Verlag, 2001.

[60] M. Biehl, B. Hammer, and T. Villmann, "Prototype-based models in machine learning", *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 7, no. 2, pp. 92–111, 2016.

[61] S. S. Haykin, *Neural networks and learning machines*, Third. Pearson Education, 2009.

[62] T. Kohonen, *Self-organizing Maps*. Berlin: Springer-Verlag Berlin, 1995.

[63] C. Hajjar and H. Hamdan, "Kohonen neural networks for interval-valued data clustering", *International Journal of Advanced Computer Science*, vol. 2, no. 11, pp. 412–419, 2012.

[64] S. Sun, "A survey of multi-view machine learning", *Neural computing and applications*, vol. 23, no. 7-8, pp. 2031–2038, 2013.

[65] Y. Yang and H. Wang, "Multi-view clustering: A survey", *Big Data Mining and Analytics*, vol. 1, no. 2, pp. 83–107, 2018.

[66] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning", *CoRR*, vol. abs/1304.5634, 2013.

[67] W. Pedrycz, "Fuzzy clustering with a knowledge-based guidance", *Pattern Recogn. Lett.*, vol. 25, no. 4, pp. 469–480, Mar. 2004, ISSN: 0167-8655.

[68] A. Cornuéjols, C. Wemmert, P. Gançarski, and Y. Bennani, "Collaborative clustering: Why, when, what and how", *Information Fusion*, vol. 39, pp. 81–95, 2018.

[69] M. Hai, S. Zhang, L. Zhu, and Y. Wang, "A survey of distributed clustering algorithms", in *2012 International Conference on Industrial Control and Electronics Engineering*, IEEE, 2012, pp. 1142–1145.

[70] J. Sublime, "Contributions to collaborative clustering and its potential applications on very high resolution satellite images", PhD thesis, University Paris-Saclay, 2016.

[71] N. Grozavu, M. Ghassany, and Y. Bennani, "Learning confidence exchange in collaborative clustering", in *The 2011 International Joint Conference on Neural Networks, IJCNN 2011, San Jose, California, USA, July 31 - August 5, 2011*, 2011, pp. 872–879.

[72] T. Kohonen, "Description of input patterns by linear mixtures of som models", in *Proceedings of WSOM*, vol. 7, 2007.

[73] W. Gentleman, "Solving least squares problems", *SIAM Review*, vol. 18, no. 3, pp. 518–520, 1976.

[74] D. Davies and D. Bouldin, "A cluster separation measure", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.

[75] F. Kovács, C. Legány, and A. Babos, "Cluster validity measurement techniques", in *6th International symposium of hungarian researchers on computational intelligence*, Citeseer, 2005.

[76] S. Saitta, B. Raphael, and I. F. Smith, "A bounded index for cluster validity", in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, Springer, 2007, pp. 174–187.

[77] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval", *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

[78] H. Zhao and Y. Fu, "Dual-regularized multi-view outlier detection", in *IJCAI*, 2015.

[79] D. Dheeru and E. Karra Taniskidou, *UCI machine learning repository*, 2017. [Online]. Available: http://archive.ics.uci.edu/ml.

[80] A. Sharma, G. Singh, and S. Rehman, "A review of big data challenges and preserving privacy in big data", in *Advances in Data and Information Sciences*, M. L. Kolhe, S. Tiwari, M. C. Trivedi, and K. K. Mishra, Eds., Singapore: Springer Singapore, 2020, pp. 57–65.

[81] W.-J. Wang, Y.-X. Tan, J.-H. Jiang, J.-Z. Lu, G.-L. Shen, and R.-Q. Yu, "Clustering based on kernel density estimation: Nearest local maximum searching algorithm", *Chemometrics and Intelligent Laboratory Systems*, vol. 72, pp. 1–8, Jun. 2004.

[82] L. C. Matioli, S. Santos, M. Kleina, and E. A. Leite, "A new algorithm for clustering based on kernel density estimation", *Journal of Applied Statistics*, vol. 45, no. 2, pp. 347–366, 2018.

[83] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications", *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 169–194, 1998.

[84] B. W. Silverman, "Using kernel density estimates to investigate multimodality", *Journal of the Royal Statistical Society*, pp. 97–99, 1981.

[85] A. Gramacki, *Nonparametric kernel density estimation and its computational aspects*. Springer, 2018.

[86] D. Littau and D. Boley, "Clustering very large data sets with principal direction divisive partitioning", in *Grouping Multidimensional Data*, Springer, 2006, pp. 99–126.

[87] H. Ismkhan, "A. 1d-c: A novel fast automatic heuristic to handle large-scale one-dimensional clustering", *Applied Soft Computing*, vol. 52, pp. 1200–1209, 2017.

[88] R. Beale and T. Jackson, *Neural Computing-an introduction*. CRC Press, 1990.

[89] Y. Bennani., "Adaptive weighting of pattern features during learning", in *IJCNN'99*, vol. 5, Piscataway, NJ, 1999, pp. 3008–13.