Contributions to Record Linkage for Disclosure Risk Assessment

by
Jordi Nin Guerrero

Advisor:
Dr. Vicenç Torra i Reventós

A dissertation submitted in partial fulfillment of the requirements for
the degree of Doctor in Artificial Intelligence at the
Departament de Ciències de la Computació
Universitat Autònoma de Barcelona

A la Mireia , que fa seves les meves il·lusions.

I a en Xavier, a qui esperem amb entusiasme.

I am only a child playing on the beach,
while vast oceans of truth lie undiscovered before me.

Isaac Newton (1642-1727)

Contents

Agraïments **xiii**

Abstract **xv**

1 Introduction **1**

1.1 Motivations 1

1.2 Contributions 3

1.3 Structure of the Document 4

2 Preliminaries **7**

2.1 Aggregation Functions 7

2.2 Time Series 12

2.3 Re-identification Methods 17

2.4 Microdata Protection Methods 21

2.5 Information Loss and Disclosure Risk 33

2.6 Data Sets Description 38

3 Microaggregation Analysis **45**

3.1 Attribute Selection in Multivariate Microaggregation 45

3.2 Modeling Projections in Microaggregation 58

3.3 Improving Microaggregation for Complex Record Anonymization . . . 62

4 Specific Disclosure Risk Measures **77**

4.1 Rank Swapping Record Linkage 77

4.2 Alignment Record Linkage 93

4.3 Projected Record Linkage 98

5 Record Linkage using Fuzzy Integrals **109**

5.1 An Alternative Disclosure Risk Scenario 109

5.2 Experiments 117

6 Time Series Protection **125**

6.1 Time Series Protection 125

6.2 Time Series Information Loss Measures 127

6.3 Time Series Disclosure Risk Measures 130

6.4 Final Trade-off Evaluation 135

6.5 Experiments 136

7 Conclusions and Future Directions **141**

7.1 Summary of Contributions 141

7.2 Conclusions 142

7.3 Future Directions 143

List of Figures

2.1 Record Linkage Schema. 18

2.2 Data set protection and release process. 22

2.3 Disclosure Risk Scenario. 23

3.1 Points which have been artificially generated to obtain databases with non-correlated attributes, in 2 dimensions (a), and 3 dimensions (b). . 50

3.2 Quantifier $Q(A) = Q(|A|/N)$ where $Q(x) = x$ and a represents the values of one record of the data set. 60

3.3 Graphical representation of the DR (a) and Score (b) of (PCP, Zscores and Sugeno) microaggregation using $a = 4$ and $k = 5, 15, 25$. 62

3.4 Mic1D-κ schema. 63

4.1 Graphic representation of disclosure risk. 79

4.2 Graphic representation of the p-distribution swap interval. 82

4.3 Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with standard rank swapping. 84

4.4 Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with rank swapping p-distribution. 85

4.5 Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with rank swapping p-buckets. 86

4.6 Graphic representation of the information loss (a), disclosure risk (b) and score (c) values for the three rank swapping methods when Census data set is protected. 89

4.7 Graphic representation of the information loss (a), disclosure risk (b) and score (c) values for the three rank swapping methods when EIA data set is protected. 89

4.8 Graphic representation of the number of links obtained with different record linkage techniques, applied to the Census data set protected with optimal microaggregation (a) and MDAV microaggregation (b) using $k = 50$. 96

4.9 Graphic representation of the number of links obtained with different record linkage techniques, applied to the EIA data set protected with optimal microaggregation (a) and MDAV microaggregation (b) using $k = 50$. 97

4.10 Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with PCP microaggregation with $v = 4$. 103

4.11 Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with Zscores microaggregation with $v = 4$. 103

4.12 Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with MDAV with $v = 4$. 104

5.1 Graphical representation of an artificial problem that satisfies Assumption 2: Data set A with attributes {Benefits, Start-up costs} and data set B with attribute {Business type}. In this figure, business types are represented using squares, ellipses, triangles, and so on. 111

5.2 Graphical representation of Q^e_α for $\alpha = 1/5, \ldots, 10/5$. 119

5.3 Graphical representation of Q^s_α for $\alpha = 0, 0.1, \ldots, 0.9$. 120

5.4 Graphical representation of Q^t_α for $\alpha = 0, 0.1, \ldots, 0.9$. 121

6.1 Graphical representation of distance function selection. 127

6.2 Graphical representation of the effects of time series normalization, (a) represents the original data without normalization, (b) represents normalized data with independent normalization, (c) represents normalized data with time series normalization. 132

List of Tables

2.1 Rank swapping example. 25

2.2 Optimal univariate microaggregation example. 28

2.3 Projection based microaggregation example. 32

2.4 MDAV microaggregation example. 33

2.5 Attributes of the Census data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 39

2.6 Attributes of the EIA data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 39

2.7 Attributes of the Abalone data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 40

2.8 Attributes of the Dermatology data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 41

2.9 Attributes of the Housing data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 42

2.10 Attributes of the Ionosphere data set. 42

2.11 Attributes of the Iris data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given. 43

2.12 Attributes of the Water Treatment data set. 43

2.13 Attributes of the WDBC data set. 44

3.1 Attribute selection of the Water Treatment data set. 51

3.2 Attribute selection of the EIA data set. 52

3.3 Scores of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 53

3.4 Scores of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 54

3.5 SSE and real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the Water-treatment data set. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 55

3.6 SSE and real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the EIA data set. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 56

3.7 Score of different microaggregation methods and parameterizations. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores or Sugeno) with $a = i$ and $k = j$. 61

3.8 Example of a microdata file used to illustrate the preprocessing block. . 64

3.9 Different groups of attributes known by the intruder. 69

3.10 SSE and real k' of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 70

3.11 SSE and real k' of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 71

3.12 Scores of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 75

3.13 Scores of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k. 76

4.1 Rank swapping example. 80

4.2 Correct links and average disclosure risk for Example 2.1 on record linkage of Section 2.4.1, computed with rank swapping (RS-RL), distance based (DB-RL) and probabilistic (P-RL) record linkage. 80

4.3 Score calculation for standard rank swapping(rs-p). IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure. 87

4.4 Score calculation for rank swapping p-distribution (rs p-d). IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure. 88

4.5 Score calculation for rank swapping p-buckets (rs p-b).IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure. 90

4.6 Average linkage values for rank swapping p-distribution.DB-RL stands for Distance Based Record Linkage, RS-RL stands for Rank Swapping Record Linkage. 92

4.7 SSE results for univariate microaggregation. 96

4.8 Score of different microaggregation methods and parameterizations when applied to Census data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$. 100

4.9 Score of different microaggregation methods and parameterizations when applied to EIA data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$. 101

4.10 New scores of the different microaggregation methods, applied to Census data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$. 105

4.11 New scores of the different microaggregation methods, applied to EIA data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$. 106

5.1 Data sets A and B for re-identification. 116

5.2 Data set A (and B) for re-identification. 116

5.3 Probabilities of having r correct links and of having more or equal than r links for 100 records. 122

5.4 Average number of re-identified records for the Abalone example. . . . 122

5.5 Average number of re-identified records for the Ionosphere example. . 123

5.6 Average number of re-identified records for the Census example. 123

5.7 Average number of re-identified records for the WDBC example. 123

6.1 Data extracted from Spanish National Statistics Institute. 131

6.2 Data normalized with the standard component-wise procedure. 132

6.3 Data normalized with the time series procedure. 132

6.4 Details of time series examples. 137

6.5 Score and its components in the forecasters data set. Forecaters.i.d-k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k. 138

6.6 Score and its components in the football data set. football.i.d.k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k. 139

6.7 Score and its components in the ibex35 data set. ibex35.i.d.k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k. 140

Agraïments

Uns dies abans de la defensa de la meva tesi doctoral és un bon moment per mirar enrere i valorar el llarg camí recorregut per arribar fins aquí. A dia d'avui puc dir sense por a equivocar-me que he tingut la gran sort de poder recórrer aquest camí acompanyat d'una gran quantitat de persones que m'han ajudat en molts aspectes fent aquesta travessia molt més senzilla.

Primer de tot, m'agradaria començar aquests agraïments per la persona que més m'ha ajudat en la realització d'aquesta tesi. Aquesta persona no és un altre que en Vicenç Torra. D'en Vicenç podria dir que és un gran director de tesi o podria destacar les seves grans qualitats científiques, totes dues coses són ben certes, però crec més important destacar les seves grans qualitats humanes. Per això prefereixo agrair-li la seva gran paciència i dedicació durant gairebé els últims quatre anys.

També vull donar les gràcies:

- a la gent de la UPC per la seva col·laboració en part de la feina feta en aquesta tesi, especialment a en Víctor Muntés, Norbert Martínez i Josep Lluís Larriba.
- a en Jordi i en Pau, dos alumnes de matemàtiques que han implementat part del codi font d'alguns dels mètodes de protecció de dades utilitzats en aquest treball.
- a en Mario per revisar l'anglès d'aquesta tesi en un temps record.
- als companys del IIIA per les estones agradables que m'han fet passar prenent cafè a la sala de descans.
- a en Dany pel seu somriure de benvinguda diari.

- a tots els meus amics. Per les bones estones que m'heu fet passar.

Finalment vull donar les gracies a la Mireia, la meva estimada dona i futura mare del meu fill. Si a algú he d'agrair el poder dedicar-me al que m'agrada és a ella, pel seu suport als moments difícils i per la seva gran paciència i recolzament. Sense ella res d'això tindria sentit.

Un boci d'aquesta tesi prové de l'ajuda de cadascú de vosaltres, si us plau trieu el que més us agradi.

MOLTES GRÀCIES A TOTS

Abstract

Every day, a large amount of data is collected by statistical agencies. This fact combined with the growth that the Internet has experimented during the recent years makes one wonders whether its confidential data is stored and distributed in a secure way.

In this framework, data protection methods have a great importance, becoming crucial to anonymize confidential attributes before releasing them in a private and secure manner. When a protection method is applied, a new and challenging problem arises. This problem is the evaluation of the privacy provided by such method. Re-identification techniques, as record linkage methods, are one of the most common techniques for evaluating the security of a protection method.

This thesis applies record linkage techniques to the calculation of the disclosure risk of a protection method. The aim of this application is to evaluate the security of a protection method in a real and fair way. The main contributions are:

- The definition of three specific record linkage techniques for evaluating two of the most common protection methods: rank swapping and microaggregation.
- The definition of an empirical disclosure risk measure for microaggregation.
- The development of new variants of rank swapping and microaggregation resistant to record linkage methods and disclosure risk measures defined in this thesis.
- The study of new disclosure risk scenarios. In particular, we have developed a record linkage method which applies aggregation functions to re-identify individuals when the intruder has no access to any of the original attributes of

the protected data. We have also developed a framework for the evaluation of protection methods when they are applied to time series data.

Chapter 1

Introduction

1.1 Motivations

Statistical Disclosure Control (SDC) is the discipline concerned with the anonymization of the statistical data containing confidential information about individual entities such as persons or enterprises. Normally, data anonymization is achieved by modifying data values. The aim of SDC is to prevent third parties working from this data to recognize individuals and disclosing confidential information about them. Here, we understand *third parties* as the data users outside the statistical agencies (*e.g.* policy makers, academic researchers and general public).

Typically, data published by statistical agencies can be classified as tabular data and microdata files. Tabular data contains aggregated values and their utility is limited. In contrast, microdata files (*i.e.* records which contain information about individuals) have much more utility due to their flexibility to allow the user to perform a wide range of data analysis (*i.e.* regressions). For this reason, third parties have increased their demand for statistical data according to this latter form. This issue motivates statistical agencies to increase the release of microdata files.

In both scenarios, statistical agencies have to be careful when releasing statistical data since they have an important responsibility towards the respondents. Moreover, international and local law seek to ensure that confidential data is managed in

a correct (and private) manner. They have to make (almost) impossible for third parties to acquire sensitive information about respondents from the released microdata file.

A closely related research line where privacy is involved is Privacy Preserving Data Mining (PPDM). PPDM tackles the problem of developing data mining techniques where the privacy of the individuals is preserved. In a very similar way to SDC, PPDM modifies individual data records in such a way that the results of a mining process are (almost) the same as those obtained when using the real data.

In both cases (SDC and PPDM) the privacy of the individuals through data protection methods should be ensured. These methods modify the original microdata file or data set[1], adding some noise in the original data. Of course, the aim of such methods is to preserve the statistical utility of the protected data as much as possible. This is equivalent to modify the information as little as possible. However, protected data have to be altered enough to obfuscate the identity of the respondents.

Protection methods solve in some way the problem of the privacy of the respondents. Nevertheless, an important and challenging problem arises: the evaluation of such methods. This evaluation has two clear components. On the one hand, the loss of statistical utility of the protected data (*information loss*) and on the other hand, the risk that third parties discover the identity of certain respondents (*disclosure risk*).

Information loss measures can be general or specific. General information loss measures roughly reflect the amount of information loss for a reasonable range of data uses. On the other hand, specific information loss measures evaluate the loss of statistical utility for a particular data analysis. Normally, the first kind of measures are used to compare protection methods and the second ones are used to evaluate in an accurate way the real effect of a protection method for a concrete statistical analysis.

Disclosure risk, the main topic of this thesis, evaluates the privacy of the respondents against possible malicious uses that third parties (sometimes called intruders) could do with the information released. Disclosure risk measures evaluate the number of respondents whose identity is revealed. Normally, these measures are computed in several scenarios where the intruder has partial knowledge of the original data. In order to compute the disclosure risk, general methods for re-identification are used.

[1] Microdata file is the term used in SDC to refer to the raw data, and data set is usually the term used in PPDM to refer to the same concept. In this thesis we will use both terms.

These methods find relationships (*i.e.* links) between the protected data and the partial knowledge which the intruder is assumed to have.

In the real world, the disclosure risk is bounded by the best re-identification method that an intruder is able to conceive. Finding this method is a challenging task as the intruder can exploit any weakness of the protection method or any extra information about the original data. Therefore, the computation of the real disclosure risk is a very hard issue since lots of considerations must be taken into account. This thesis is focused on this matter. The aim of this work is to provide a set of techniques for statistical agencies and data providers in general to determine the disclosure risk in the most accurate way.

1.2 Contributions

The research done in this thesis contributes in three different aspects.

Firstly, it contributes to the area of disclosure risk evaluation. We introduce several re-identification methods to compute the disclosure risk of different data protection methods. The new re-identification methods show that up to now the real disclosure risk of such protection methods was underestimated. These methods demonstrate that an intruder can increase the amount of correctly re-identified respondents by considering the protection method applied in the anonymization process. Therefore, the disclosure risk of these methods rises accordingly. We also define a different disclosure risk scenario where the intruder has no access to the original data. However, under some assumptions, we prove that it is still possible for the intruder to re-identify some of the respondents of such protected data set.

The second contribution is included in the area of data protection methods. We introduce several protection methods which solve the drawbacks presented in the disclosure risk evaluation. These new methods improve the privacy of the respondents. The methods showed in this thesis avoid that an intruder may exploit the knowledge of the protection method used. We also define a new measure to evaluate, in an empirical way, the anonymity level achieved using a specific configuration of a protection method and assuming that the intruder has access to the original values of a subset of the protected attributes.

Finally, we present a suite of techniques for time series anonymization and re-identification. The idea underlaying this approach is that data accumulation through consecutive statistical surveys enables to perform temporal analysis over such data (*e.g.* forecasting). However, this temporal information can be also used by the intruder to increase the disclosure risk of this new accumulated survey. Under this scenario, we also define new information loss measures which consider temporal analysis that third parties can perform in the accumulated data set.

1.3 Structure of the Document

This document is organized in three parts with five chapters: preliminaries and related work (Chapter 2), our contributions (Chapters 3 to 6) and, finally, conclusions and future directions (Chapter 7).

- **Chapter 2.** We explain some preliminaries needed later on. These preliminaries are divided in six sections:
  - **Aggregation functions.** We begin the preliminaries explaining some basic concepts about aggregation functions. Such description includes the definition of the OWA (Ordered Weighted Averaging) operator and some fuzzy integrals, in particular, the Choquet, Sugeno and twofold integrals.
  - **Time series.** We introduce some notions about time series as, for instance several time series distances and forecasting models.
  - **Re-identification methods.** We give a brief introduction of classical re-identification methods and explain in more detail record linkage (RL) methods. RL methods are specific cases of the re-identification methods.
  - **Microdata protection methods.** We show the general problem of data privacy, the re-identification scenario and we give two classifications of protection methods. We also explain in detail two specific data protection methods: rank swapping and microaggregation.
  - **Information loss and disclosure risk.** We present some information loss and disclosure risk measures and a framework for evaluating a data protection method.

  - **Data sets description.** We give an exhaustive description of the data sets used in the experiments performed in this thesis.

- **Chapter 3.** We explain some contributions about specific microaggregation disclosure risk measures. We also present two new variants of the generic microaggregation algorithm.

- **Chapter 4.** Three ad-hoc record linkage methods are presented. These methods consider the protection method applied on the original data, and due to this, they achieve a larger number of re-identifications than generic record linkage methods.

- **Chapter 5.** We study an alternative scenario for record linkage methods where attributes in the original and the protected data set are not the same.

- **Chapter 6.** We present some results about time series protection and re-identification. We also present some information loss measures for the evaluation of time series protection methods.

- **Chapter 7.** This thesis concludes with some conclusions and a description of future work.

Chapter 2

Preliminaries

In this chapter, we begin explaining some basics about aggregation functions including a description of OWA operators. Then, we introduce certain concepts about time series as the notation used in this dissertation, some distances and several time series forecasting models. Afterwards, we review re-identification methods and the two main existing approaches for standard record linkage: probabilistic and distance based record linkage. Then, we give a brief general description about microdata protection methods, which reviews the two protection methods studied in this work, rank swapping and microaggregation. Such methods are illustrated with a toy example. Finally, we present the standard way of computing the score of a protection method by combining its information loss and its disclosure risk.

2.1 Aggregation Functions

Aggregation functions [70] are numerical functions used for information fusion that combine N numerical values into a single one. These operators formally described below, typically satisfy unanimity (idempotency) and monotonicity.

Definition 1 *Let $X := \{x_1, \ldots, x_N\}$ be a set of information sources, and let $f(x_i)$ be a function that models the value supplied by the i-th information source x_i (for the sake of simplicity we often denote $f(x_i)$ by a_i), then a function $\mathbb{C} : \mathbb{R}^N \to \mathbb{R}$ is said to be an*

aggregation function if it satisfies:

1. $\mathbb{C}(a,\ldots,a) = a$ *(unanimity, also known as idempotency)*
2. $\mathbb{C}(a_1,\ldots,a_N) \leq \mathbb{C}(a'_1,\ldots,a'_N)$ *if* $a_i < a'_i$ *(monotonicity)*

At present, several aggregation functions exist in the literature (see *e.g.* [12, 70] for a review). Among them, the most well-known aggregation functions are the arithmetic mean and the weighted mean. They correspond, respectively, to the following functions:

1. $\mathbb{C}(a_1,\ldots,a_N) = \frac{\sum_i^N a_i}{N}$
2. $\mathbb{C}(a_1,\ldots,a_N) = \sum_i^N w_i a_i$

In the second definition, $\mathbf{w} = (w_1 \ldots w_N)$ stands for a weighting vector. That is, w_i are weights for sources x_i so that $w_i \geq 0$ and $\sum_i w_i = 1$. These values correspond to prior knowledge on the reliability of the sources. For example, when source x_i is twice as reliable as source x_j then we have that $w_i = 2w_j$.

Yager defined in [77] the so-called Ordered Weighted Averaging (OWA) operator that corresponds to a weighted linear combination of order statistics. At present there are different definitions for this operator based on the way the weights are defined. We recall a definition based on a non-decreasing function, as this is the most useful definition in our context.

Definition 2 *Let Q be a non-decreasing function in $[0,1]$ so that $Q(0) = 0$ and $Q(1) = 1$, then the mapping $OWA_Q : \mathbb{R}^N \to \mathbb{R}$ defined as follows is an OWA operator:*

$$OWA_Q(a_1,\ldots,a_N) = \sum_{i=1}^{N} \big(Q(i/N) - Q((i-1)/N)\big) a_{\sigma(i)}$$

where σ is a permutation of the values a_i such that $a_{\sigma(i)} \geq a_{\sigma(i+1)}$.

This operator has several properties. We underline the following ones:

i) For all Q, it holds that:

$$\min_i a_i \leq OWA_Q(a_1,\ldots,a_N) \leq \max_i a_i.$$

ii) The function Q permits to modulate the output. For example, when we consider the family of functions $Q_\alpha(x) = x^\alpha$, we have that large positive values of α lead to an OWA near to the minimum and, on the contrary, values of α near to zero lead to an OWA near to the maximum. Also, when a_i is fixed, OWA_{Q_α} is non-decreasing with respect to α. These conditions are formalized as:

- $\lim_{\alpha\to\infty} OWA_{Q_\alpha}(a_1,\ldots,a_N) = a_{\alpha(N)} = \min a_i$
- $\lim_{\alpha\to 0} OWA_{Q_\alpha}(a_1,\ldots,a_N) = a_{\alpha(1)} = \max a_i$
- if $\alpha_1 > \alpha_2$ then $OWA_{\alpha_1}(a_1,\ldots,a_N) < OWA_{\alpha_2}(a_1,\ldots,a_N)$

iii) The OWA operator is symmetric for all Q. That is, the order of the parameters is not relevant for the computation of the output. This can be formalized as follows:

$$OWA_Q(a_1,\ldots,a_N) = OWA_Q(a_{\pi(1)},\ldots,a_{\pi(N)})$$

for any permutation π.

Another relevant property of OWA operators is that they are equivalent to the so-called Choquet integrals [14] with respect to symmetric fuzzy measures. Choquet integrals are one family of the so-called fuzzy integrals [35], a set of functionals that can be used for information fusion. In short, given the function f that represents the information supplied by the sources in X, the fuzzy integral of f with respect to a fuzzy measure represents an aggregated value of those values in f. In such integrals, fuzzy measures play the role of weights in the weighted mean (*i.e.*, some prior knowledge on the reliability of the sources). The main difference between a fuzzy integral and a weighted mean is that in the weighted mean independence is assumed between the information sources. On the other hand, such independence is not formally required for fuzzy integrals, as fuzzy measures can accommodate dependencies between the sources.

Formally speaking, a fuzzy measure μ is a set function over X (*i.e.*, $\mu : 2^X \to [0,1]$) that satisfies the following constraints:

- $\mu(\emptyset) = 0$, $\mu(X) = 1$ (boundary conditions).
- if $A \subseteq B$ then $\mu(A) \leq \mu(B)$ (monotonicity conditions).

The OWA operator of f with respect to Q is equivalent to the Choquet integral of f with respect to the fuzzy measure μ defined as: $\mu(A) = Q(|A|/N)$ where $|\cdot|$ stands for the cardinality of a set. This equivalence establishes that the fuzzy measure associated with the OWA for a set A does not depend on the particular elements in A but only on its cardinality. That is, given two sets $A \neq B$ $(A, B \subseteq X)$ such that $|A| = |B|$ then $\mu(A) = \mu(B)$. For this reason, the measure is said to be symmetric and, consequently, any Choquet integral with respect to a measure of this form is also symmetric as this corresponds to the OWA operator.

Formally, the Choquet integral is defined as follows:

Definition 3 *Let μ be a fuzzy measure on X; then, the Choquet integral of a function $f : X \to \mathbb{R}^+$ with respect to the fuzzy measure μ is defined by*

$$(C)\int f\,d\mu = \sum_{i=1}^{N} [f(x_{\sigma(i)}) - f(x_{\sigma(i-1)})]\mu(A_{\sigma(i)})$$

where $f(x_{\sigma(i)})$ indicates that the indices have been permuted so that $0 \leq f(x_{\sigma(1)}) \leq \cdots \leq f(x_{\sigma(N)}) \leq 1$, and where $f(x_{\sigma(0)}) = 0$ and $A_{\sigma(i)} = \{x_{\sigma(i)}, \ldots, x_{\sigma(N)}\}$.

The property that a Choquet integral with respect to a symmetric fuzzy measure is symmetric also holds for other fuzzy integrals. In particular, it also holds for the Sugeno integral [59]. Formally, the Sugeno integral is defined as follows:

Definition 4 *Let μ be a fuzzy measure on X; then, the Sugeno integral of a function $f : X \to [0, 1]$ with respect to the fuzzy measure μ is defined by*

$$(S)\int f\,d\mu = \bigvee_{i=1}^{N} (f(x_{\sigma(i)}) \wedge \mu(A_{\sigma(i)}))$$

where $\vee$ stands for maximum, $\wedge$ stands for minimum, $f(x_{\sigma(i)})$ indicates that the indices have been permuted so that $0 \leq f(x_{\sigma(1)}) \leq \cdots \leq f(x_{\sigma(N)}) \leq 1$, and where $f(x_{\sigma(0)}) = 0$ and $A_{\sigma(i)} = \{x_{\sigma(i)}, \ldots, x_{\sigma(N)}\}$.

We give below the definition of the Sugeno integral with respect to a symmetric fuzzy measure representable, as above, in terms of a function Q. This expression is equivalent to the OWMax defined by Yager in [78].

Definition 5 *Let Q be a non-decreasing function in $[0,1]$ such that $Q(0) = 0$ and $Q(1) = 1$, then the mapping $SI_Q : \mathbb{R}^N \to \mathbb{R}$ defined as follows is a Sugeno integral with respect to the fuzzy measure $\mu(A) = Q(|A|/N)$:*

$$SI_Q(a_1, \ldots, a_N) = \bigvee_{i=1}^{N} (Q(i/N) \wedge a_{\sigma(i)})$$

where σ is a permutation such that $a_{\sigma(i)} \geq a_{\sigma(i+1)}$.

As stated above, this function is symmetric for all Q. Besides that, the function is an aggregation function (in the sense of Definition 1) and the output of the integral is modulated through the function Q.

The twofold integral [48, 65] is a generalization for both Choquet and Sugeno integrals. The twofold integral is a fuzzy integral that aggregates a function with respect to two fuzzy measures. The rationale of this generalization is that the semantics of both measures are different. In particular, the measure in the Choquet integral is seen as a 'probabilistic flavor' measure, and the measure used in the Sugeno integral is seen as a 'fuzzy flavor' measure. We use μ_C to denote the measure that corresponds to the one in the Choquet integral, and μ_S for the one in the Sugeno integral.

Definition 6 *Let μ_C and μ_S be two fuzzy measures on X, then the* twofold integral *of a function $f : X \to [0,1]$ with respect to the fuzzy measures μ_S and μ_C is defined by:*

$$TI_{\mu_S,\mu_C}(f) = \sum_{i=1}^{n} \Big(\big(\bigvee_{j=1}^{i} f(x_{s(j)}) \wedge \mu_S(A_{s(j)}) \big) \big(\mu_C(A_{s(i)}) - \mu_C(A_{s(i+1)}) \big) \Big)$$

where s in $f(x_{s(i)})$ indicates that the indices have been permuted so that $0 \leq f(x_{s(1)}) \leq \cdots \leq f(x_{s(n)}) \leq 1$, $A_{s(i)} = \{x_{s(i)}, \cdots, x_{s(n)}\}$, $A_{s(n+1)} = \emptyset$.

2.2 Time Series

Numerical time series are defined by pairs $\{(x_i, t_i)\}$ for $i = 1, \ldots, n$ where t_i corresponds to the temporal variable and x_i is the numerical variable that depends on time (dependent variable). Consequently, $t_{i+1} > t_i$. Income, stock prices and sport statistics are examples of time series, as they depend on time.

We can define in the same way ordinal or categorical time series replacing x_t by a categorical or ordinal variable. Weather forecast (*e.g.* sunny, cloudy, raining) and restaurant category (*e.g.* one Michelin star, two Michelin stars, three Michelin stars) are examples of categorical and ordinal time series respectively. In this thesis we will only consider numerical time series.

In this work, we will adopt the following assumptions: time series are discrete, the observations are made at fixed time intervals and all time series have the same initial time t_0. Under these assumptions, it is possible to simplify the notation disregarding the temporal variable. Therefore, from now on, our notation for a time series will be $(x_1, \ldots, x_n)$.

Certain time series statistics have been defined. In this work we will use the two most common ones: the *time series mean* and the *autocorrelation function*. The reason for this selection is that both statistics are involved in the ARMA and ARIMA processes [9], two well-known processes for time series modeling. Both statistics are defined as follows [11]:

- **Time series mean.** It is defined by

$$\mu = \frac{1}{n} \sum_{i=0}^{n} x_i$$

  where n corresponds to the number of elements of the time series.

- **Autocorrelation function (ACF).** It describes the correlation between the process at different times. It is defined by

$$R(j) = \frac{(x_i - \mu)(x_{i+j} - \mu)}{n}$$

  where n corresponds to the number of elements of the time series and i and

$i + j$ are the initial elements for computing the correlation. It is usual to use $i = 0$ with j being a given shift.

2.2.1 Time Series Distances

In the literature we can find a large number of distances for time series. See [15, 44, 41] for more details.

Here we only describe the distances used in this thesis for computing the disclosure risk of univariate microaggregation in Chapter 4 and for the definition of time series microaggregation in Chapter 6.

- **Euclidean distance (EU).** It is defined as

$$d_{EU}(x, v) = \sqrt[2]{\sum_{k=1}^{n} (x_k - v_k)^2}$$

- **Short time series distance (STS).** It was defined in [44] as the square root of the sum of the slope squared differences. Formally, it is defined as follows:

$$d_{STS}(x, v) = \sqrt[2]{\sum_{k=1}^{n} \left(\frac{v_{k+1} - v_k}{t_{k+1} - t_k} - \frac{x_{k+1} - x_k}{t_{k+1} - t_k} \right)^2}$$

- **Dynamic Time Wrapping (DTW).** Any two time series can be compared elementwise with the Euclidean distance. Nevertheless, this often leads to a large distance between two time series which are very similar but with some stretch along the dimension (*e.g.* shift on the time dimension). The key idea of the DTW distance [13, 47] is that any point of a time series can be (forward and/or backward) aligned with multiple points of the other time series that lie in a different dimensional position. This compensates possible stretches in both time series and therefore the distance is in some way more appropriate when we are interested in comparing the shapes of the time series.

  In the rest of this section we will present the DTW distance with some detail. We start with the notation. Let us consider two numerical time series $x = (x_1, \ldots, x_n)$ and $v = (v_1, \ldots, v_m)$, of length n and m respectively. Then, for aligning these two time series using the DTW distance, we proceed as follows.

Firstly, we construct a bi-dimensional $n \times m$ matrix where the element (i^{th}, j^{th}) contains the distance between the two points x_i and v_j. To compute the distance between these two points, the squared Euclidean distance is often used (*i.e.*, $d(x_i, v_j)=((x_i - v_j)^2)$). In this way, each matrix element (i,j) corresponds to the distance of the possible alignment between the points x_i and v_j.

A warping path $w = (w_1, \ldots, w_L)$, that represents a relation between x and v, is a route from element $(1,1)$ to element (n,m) formed by contiguous cells with some particular constraints. Formally, the following constraints are considered:

- **Boundary conditions.** $w_1=(1,1)$ and $w_L=(n,m)$. A warping path requires starting and finishing in opposite diagonal corners of the matrix.
- **Continuity.** Given a w_l such that $w_l=(i,j)$ for $i' - i \leq 1$ and $j' - j \leq 1$; then, $w_{l+1}=(i',j')$. This restricts the allowable steps to adjacent cells including diagonally adjacent cells.
- **Monotonicity.** Given $w_l=(i,j)$ then $w_{l+1}=(i',j')$, where $i' - i \geq 0$ and $j' - j \geq 0$, with at least one strict inequality. This forces W to progress over dimension and avoids cycles in the warping path.

There are many warping paths that satisfy the above restrictions and the number of warping paths grow exponentially with respect to their length. In our case, we are interested only in the optimal path w_{opt}, the one which minimizes the following warping cost

$$w_{opt} = min\left(\sum_{x_i, v_j \in W} d[x_i, v_j]\right)$$

where d is the distance between the two points x_i and v_j and W is the set of all possible paths.

Dynamic programming can be used to solve this problem because efficient algorithms exist. Its main drawback is its large computational cost. To decrease this cost, horizontal and/or vertical stretches are often restricted to have a maximum length. However, [53] shows that this limitation has a limited influence in the outcome of the method.

2.2.2 Time Series Forecasting

Forecasting is a process that uses a set of historical values to predict an outcome. It is commonly used in time series to predict future values of a given time series. Good surveys on forecasting are [5, 54]. We explain herein five well-known forecasting models widely used in real applications.

All forecasting models estimate future values using the previous elements of time series. For instance, given a time series $(x_1, \ldots, x_n)$, we can estimate the value x_{n+1}. In this case, $(x_1, \ldots, x_n)$ are independent values of the forecasting model, whereas x_{n+1} is the dependent one. This process can be repeated using x_{n+2} as the dependent value and adding the *estimated* x_{n+1} value to the independent ones.

Simple Exponential Smoothing Forecasting Model

This is a very popular model used to produce smoothed time series. Simple exponential smoothing ($SESF$) assigns exponentially decreasing weights as the observations get older. In other words, recent observations are given relatively more weight in forecasting than the older ones.

Double Exponential Smoothing Forecasting Model

The double exponential smoothing ($DESF$), also known as Holt exponential smoothing, is a refinement of the previous one adding a component to include any trend in the data. Simple exponential smoothing models work better with data with no trend or seasonality components. For this reason, when the data exhibits either an increasing or decreasing trend over time, simple exponential smoothing forecasts tend to fall behind observations. Double exponential smoothing is designed to address this type of time series by considering the trends existing in the data.

Linear Regression Forecasting Model

This is a regression model (RM) where a dependent variable y is expressed in terms of an independent variable x and a random term ϵ as follow

$$y = \beta_0 + \beta_1 x + \varepsilon$$

where β_0 is the intercept ('constant' term) and β_1 is the parameter of the independent variable. This model can be used for forecasting, using x as previous values of the variable and y the ones to be forecasted.

Multiple Linear Regression Forecasting Model

This is an extension of the linear regression model. In this case, there is a dependent variable y and several independent variables $x_i, i = 1, ..., p$, and a random term ϵ. The model ($MLRF$) is as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon$$

where β_0 is the intercept ('constant' term), the β_i are the respective parameters of independent variables, and p is the number of parameters to be estimated in the linear regression.

Polynomial Regression Forecasting Model

The lineal regression forecasting model (a first-order polynomial) can be extended to higher orders. The polynomial regression model (PRM) $y_i = \alpha_0 + \alpha_1 x_i + \alpha_2 x_i^2 + \cdots + \alpha_m x_i^m + \varepsilon_i$ $(i = 1, 2, \ldots, n)$ is a system of polynomial equations of order m with coefficients $\{\alpha_0, \ldots, \alpha_m\}$. This model can be expressed using a data matrix X, a target vector $\vec{y}$ and a parameter vector $\vec{\alpha}$. The i-th rows of X and $\vec{y}$ contain the x and y values for the i-th data sample. In this way, the model can be written as a system of linear equations:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^m \\ 1 & x_2 & x_2^2 & \dots & x_2^m \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^m \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

which, when using pure matrix notation is, as aforementioned,

$$Y = \mathbf{X}\vec{\alpha} + \varepsilon$$

Given $\mathbf{X}$ and Y, the vector of polynomial coefficients is determined using the following expression.

$$\widehat{\vec{\alpha}} = (\mathbf{X}^T\mathbf{X})^{-1}\,\mathbf{X}^T Y$$

2.3 Re-identification Methods

Re-identification methods are a specific class of data base techniques. These methods are designed to establish relationships among different entities or attributes stored in different data sources. Obtaining the relationships among entities or attributes makes sense at least in the following scenarios:

- **Schema matching [51].** It is a basic problem in many data applications. These methods take two schemas as input and produce a mapping between elements (attributes) of the two schemas that semantically correspond to each other.
- **Data integration [16].** It refers to the creation of an integrated view of several data sources apparently incompatible. The incompatibility arises due to different perceptions and requirements which often lead us to express similar information in dissimilar forms.
- **Data cleaning [52].** It deals with detecting and removing errors and inconsistencies from data in order to improve their quality. Data quality problems

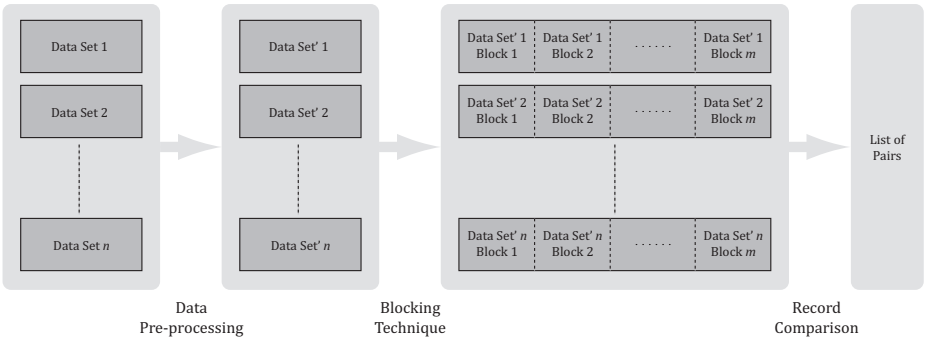


Figure 2.1: Record Linkage Schema.

presented in a single data set could be due to misspellings during data entry, missing information or other invalid data.

- **Object integration.** It refers to certain kind of applications whose aim is to establish relationships among objects, having similar properties or behaviors. A good example of this kind of application is ontology matching [6].

Record Linkage is one of the existing re-identification techniques. It is widely used for data cleaning [75] and integration of distributed and non-homogeneous data sets [69]. Typically, such data sets contain information about common individuals described using the same variables, that, frequently, do not match due to errors on the data. These errors can be accidently produced (*e.g.* typos or misspelling errors) or intentionally provoked (*e.g,* data protection). All the research done in this work is focused on this latter case.

We consider that the record linkage process is formed by different phases, as shown in Figure 2.1. To start the record linkage process, data sources are pre-processed in such a way that the attributes in the data sets are normalized [8, 74] separately to allow a simpler comparison among them in the following steps.

Once the pre-processing is done, record linkage should compare, in principle, all the records in the data sets under analysis in order to decide which records belong to the same individual. In practice, since the size of the data sets is usually very large, the comparison of all records becomes unfeasible. To avoid this comparison,

record linkage resorts to blocking methods [37, 39] that try to gather all records that present a potential resemblance. Typically, blocking methods are based on a common attribute without errors, but recently, more sophisticated methods have been developed [40].

Then, we proceed with the record linkage matching phase. During this step, only the records belonging to the same block are compared. There are several strategies to compare records. The most common ones are based on computing some conditional probabilities or distances. This is explained in Section 2.3.1 and 2.3.2.

Once the matching process delivers the result, it is necessary to analyze the list of matching pairs. This last step usually requires human intervention by means of expert individuals. As in our experiments the correct linkages are known in advance, we omit this last step.

2.3.1 Distance Based Record Linkage

Distance based record linkage consists of computing the distances between all the original and protected records. Then, the pair of records at minimum distance are considered as linked pairs (LP), whereas the remaining pairs are considered as not linked pairs (NP). In the context of data privacy the first use was [50] where it was applied to a microaggregation protection method based on the Euclidean distance.

Let $d(a,b)$ be a distance between a record in the original data set X and a record in the protected data set X'. Then, the distance based record linkage algorithm can be defined as in Algorithm 1.

Obviously, the application of this algorithm is only possible if such distance function can be calculated. Normally, this distance is defined in terms of a distance d_{attr_i} for each attribute $attr_i$ as follows:

$$d(a,b) = \sum_{i=1}^{n} d_{attr_i}(attr_i^A(a), attr_i^B(b))$$

The specific d_{attr_i} allows us tuning the method to obtain as many correct links as possible. A very common tuning approach is to weight different attributes in a different way depending on their importance. Nevertheless, in the original proposal

Algorithm 1: DB-RL

```
Data: X: original data set, X': protected data set
Result: LP: linked pairs
begin
    foreach a ∈ X do
        b' = arg_min_{b∈X'} d(a, b)
        LP = LP ∪ (a, b')
        foreach a ∈ X do
            NP = NP ∪ (a, b)
end
```

presented in [50], all attributes have the same weight.

2.3.2 Probabilistic Record Linkage

The probabilistic record linkage method was originally described in [30]. Later in [39], this method was tested over the 1985 census of Tampa, Florida. In that work, the matching algorithm was defined using the linear sum assignment model in order to define the linked pairs between the original and the protected data set. Afterwards, in [74] a new mathematical model based on the *Expectation-Maximization* (EM) algorithm was presented to compute the linked pairs. Formally, probabilistic record linkage is defined as follows.

For each pair of records (a, b) where a is an original record of the original data set X and b is a protected record of the protected data set X', we define a coincidence vector $\gamma(a,b) = (\gamma_1(a,b)\dots\gamma_n(a,b))$, where $\gamma_i(a,b)$ is defined as 1 if $attr_i(a) = attr_i(b)$ and as 0 if $attr_i(a) \neq attr_i(b)$. Note that, $attr_i$ values are the standardized values of the original and the protected data sets. Then an index is computed over this coincidence vector. Afterwards, by using such index, pairs are classified as either a linked pair (LP) or a non-linked pair (NP).

In this framework, indices are computed using conditional probabilities. Such probabilities are estimated using the EM algorithm. Then, the thresholds are computed from: (i) the probability of linking a pair that is an unmatched pair (a *false positive* or *false linkage*: $P(LP|\mathbf{U})$) and (ii) the probability of not linking a pair that is a match pair (a *false negative* or *false unlinkage*: $P(NP|\mathbf{M})$).

Although, from a computational point of view, probabilistic record linkage is a much more complex method compared to the distance based record linkage method, this approach is very interesting because the user has to provide only two probabilities as input: an upper bound of the probability of a false match and an upper bound of the probability of a false non-match. This is a clear advantage against the distance based record linkage.

2.4 Microdata Protection Methods

A data set X can be seen as a matrix with n rows (*records*) and k columns (*attributes*). Each row contains the values of the attributes for an individual. The attributes in a data set can be classified in three non-disjoint categories:

- **Identifiers.** They are attributes which unambiguously identify the individual, for example, the passport number.
- **Quasi-identifiers.** They are attributes which can identify the individual when some of those attributes are combined. For example, age, postal code or job cannot identify an individual, but the set of individuals working at the IIIA-CSIC, living in Tiana and being born in 1979, contains a single individual.
- **Confidential.** They are attributes which contain sensitive information about the individual. For example, salary.

When considering this classification, a data set X is defined as $X = id||X_{nc}||X_c$, where id are the identifiers, X_{nc} are the non-confidential quasi-identifier attributes, and X_c are the confidential attributes. Normally, before releasing a data set X with confidential attributes, a protection method ρ is applied, leading to a protected data set X'. Indeed, we will assume the following typical scenario: (i) identifier attributes in X are either removed or encrypted, therefore we will write $X = X_{nc}||X_c$; (ii) confidential attributes X_c are not modified, and so we have $X'_c = X_c$; (iii) the protection method itself is applied to non-confidential quasi-identifier attributes, in order to preserve the privacy of the individuals whose confidential data is being released. Therefore, we have $X'_{nc} = \rho(X_{nc})$. This scenario allows third parties to have precise information on confidential data without revealing to whom the confidential data

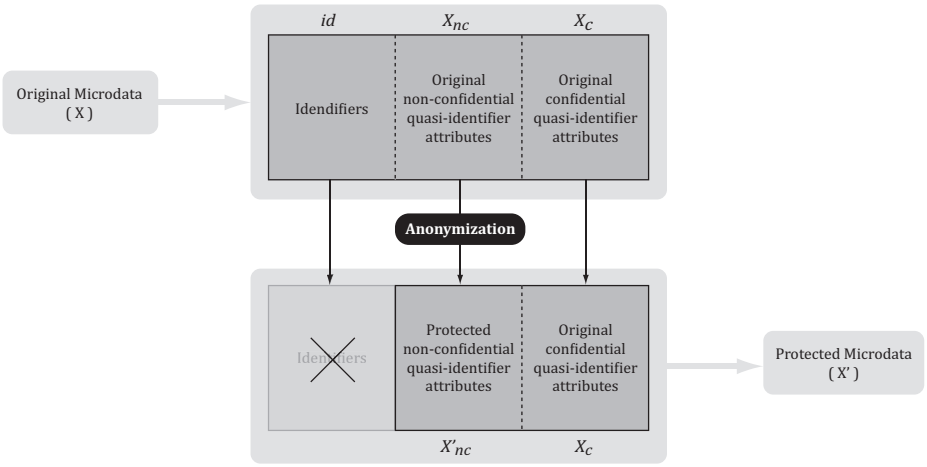


Figure 2.2: Data set protection and release process.

belongs to. Figure 2.2 depicts the process of protection and release of a microdata file, as formerly explained.

In this scenario, as shown in Figure 2.3, an intruder might try to re-identify individuals by obtaining the non-confidential quasi-identifier data (X_{nc}) together with identifiers (Id) from other data sources. By applying record linkage between the protected attributes (X'_{nc}) and the same attributes obtained from other data sources (X_{nc}), the intruder might be able to re-identify a percentage of the protected individuals together with their confidential data (X_c). This is what protection methods try to prevent. This scenario is similar to the scenario used in [73, 68].

Protection methods can be classified depending on their effect on original data into three different categories:

- **Perturbative.** The data set is distorted adding noise. In this way, in the original data set, the combinations of values which unambiguously identify an individual (or respondent) disappear and then, new combinations appear in the protected data set. This obfuscation makes difficult for an intruder to obtain the values of the original data set. A perturbative protection method has to ensure that the statistical information in the original data set is preserved on the protected one. The protection methods used in this thesis, Rank Swapping [45] and Microaggregation [19], are included in this category.

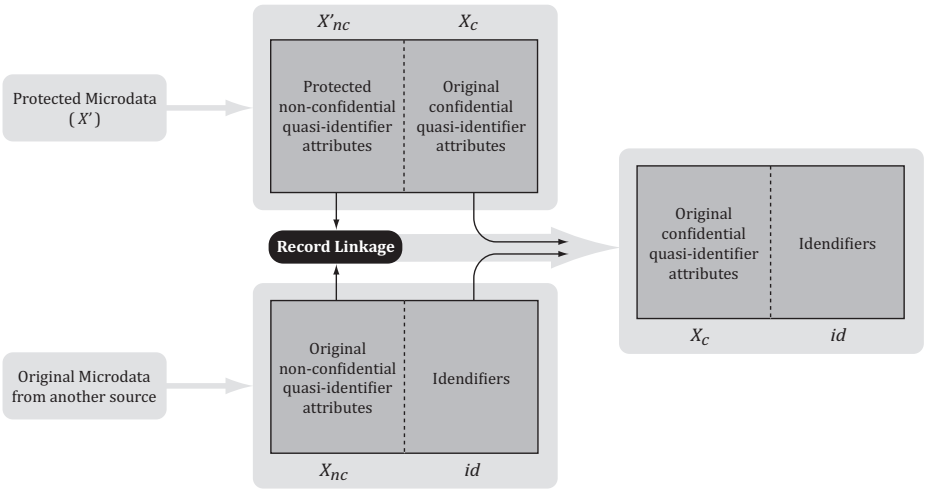


Figure 2.3: Disclosure Risk Scenario.

- **Non-perturbative.** Non-perturbative methods do not distort the original data set. They do partial suppressions or detail reductions on the original data set. These protection methods convert the combinations of values which unambiguously identify an individual into more general ones. Thus, the re-identification process is more difficult.

- **Synthetic Data Generators.** Synthetic data generators build a data model from the original data set and subsequently, a new (protected) data set is randomly generated constrained by the model computed. This approach is very promising for the statistical disclosure, although recent works as [68] show that it is possible to link synthetic data with the original data set.

Another dimension to classify protection methods is to consider the different type of data that protection methods can be used:

- **Numerical.** An attribute is considered numerical if arithmetic operations can be performed with it (*e.g.* age or income). Note that a numerical attribute does not necessarily have an infinite range, as in the case of age. When we are designing methods to protect numerical data, one has the advantage that arithmetic operations are possible, and the drawback that every combination of numerical values in the original data set is likely to be unique, which leads to

disclosure if no action is taken. This thesis focuses on protection methods for numerical data, even though rank swapping and microaggregation have been also defined for categorical data [45, 66] and some of the results presented herein can also be applied in that setting.

- **Categorical.** An attribute is considered categorical when it takes values over a finite set and standard arithmetic operations do not make sense. Ordinal and nominal scales can be distinguished among categorical attributes. In ordinal scales the order between values is relevant (*e.g.* academic degree), whereas in nominal scales it is not (*e.g.* hair color). In the former case, max and min operations are meaningful while in the latter case only pairwise comparison is possible. When we are designing methods to protect this kind of data, the inability to perform arithmetic operations is an inconvenient, but the finiteness of the value range is an interesting property that can be successfully exploited.

During the past few years, special efforts have been made to develop a wide range of protection methods. Good surveys about data protection methods can be found in the literature [1, 21]. Among all the proposed data protection methods, rank swapping and microaggregation are ones of the most used by the statistical agencies [31]. This wide application is due to they are very simple and have a low computational cost. This thesis is focused on the study of the disclosure risk of these methods. We also use the performance results of rank swapping and microaggregation as a baseline for the study of new protection methods developed herein. Now, we describe rank swapping and microaggregation in detail.

2.4.1 Rank Swapping

Rank swapping is a widely used microdata protection method, which was originally described only for ordinal attributes in [45]. However, in the comparisons made in [21], rank swapping was ranked among the best microdata protection methods for numerical attributes.

Rank swapping with parameter p and with respect to an attribute $attr_j$ (*i.e.*, the j-th column of the original data set X) can be defined as follows: firstly, the records of X are sorted in increasing order of the values x_{ij} of the considered attribute $attr_j$. For simplicity, we assume that the records are already sorted, that is $x_{ij} \leq x_{\ell j}$ for all

$1 \leq i < \ell \leq n$. Then, each value x_{ij} is swapped with another value $x_{\ell j}$, randomly and uniformly chosen from the limited range $i < \ell \leq i + p$. Finally, the sorting step is undone.

Generaly, rank swapping of a data set consists in running the algorithm explained above for each attribute to be protected, in a sequential way.

The parameter p is used to control the swap range. Normally, p is defined as a percent of the total number of records in X. Therefore, when p increases the difference between x_{ij} and $x_{\ell j}$ may increase accordingly. This fact makes re-identification more difficult, but of course the differences between the original and the protected data set are higher, decreasing in this way its statistical utility.

Original Data Set X				Protected Data Set X'			
$attr_1$	$attr_2$	$attr_3$	$attr_4$	$attr'_1$	$attr'_2$	$attr'_3$	$attr'_4$
8	9	1	3	10	10	3	5
6	7	10	2	5	5	8	1
10	3	4	1	8	4	2	2
7	1	2	6	9	2	4	4
9	4	6	4	7	3	5	6
2	2	8	8	4	1	10	10
1	10	3	9	3	9	1	7
4	8	7	10	2	6	9	8
5	5	5	5	6	7	6	3
3	6	9	7	1	8	7	9

Table 2.1: Rank swapping example.

Example 2.1 *Let us consider the data set shown in the left side of Table 2.1. Then we protect this data set using rank swapping with $p = 2$. The protected data set is shown in the right side of the same table. For the sake of simplicity, in the example original and protected files are identically sorted.*

2.4.2 Microaggregation

Recently, microaggregation [17] has emerged as one of the most promising protection methods. For example, [31] shows that microaggregation is used by many statistical agencies for data anonymization.

The basic implementation of microaggregation works as follows: given a data set of a attributes, microaggregation builds small clusters of at least k elements and replaces the original values by the centroid of the cluster to which the record belongs to. A certain level of privacy is ensured because k records have an identical protected value (*k-anonymity* [57, 61, 60]). Note that there are other ways to achieve k-anonymity; in some of them (just as it happens with basic microaggregation), the released data set enjoys k-anonymity as a whole (see [3], for example). In other solutions, the data holder chooses different subsets of attributes, and k-anonymity is ensured, independently, for each of these subsets of attributes (see [34]).

When the number of attributes is large, the basic microaggregation technique suffers from a low statistical utility (see for example [2]), especially if the attributes are not much correlated. This is so because in this case the distances between original records in the data set and the centroids are quite large. Therefore, much information on the original data is lost and is not included in the released (protected) data set.

To solve this drawback, the following natural strategy is applied by statistical agencies: the data set is split into smaller blocks of attributes, and microaggregation is applied separately to each block. In this way, the information loss is lower but at the cost of a loss in the achieved level of privacy. Indeed, the property of k-anonymity is not ensured now. For example, the k records which fall in the same cluster for the first block of attributes, can fall in different clusters for all the other blocks of attributes. So, the resulting protected records will not be equal and no k-anonymity is ensured. The simplest approach for microaggregation is when the size of the attribute blocks is equal to one, in other words, each attribute is protected independently. This corresponds to *Univariate Microaggregation* or *Individual Ranking Microaggregation*.

The goal of microaggregation methods is to minimize the total sum of distances between all the elements to be protected and the centroid of the cluster where an element belongs to, *i.e* minimize the total Sum of Square Errors (SSE). The rational of this process is to make the protected data as similar as possible to the original one. In any case, the methods should provide clusters with at least k elements. The optimal multivariate microaggregation has been proven as NP-Hard [49]. For this reason, heuristic methods can be found in the literature. On the other hand, several polynomial approaches for the optimal univariate microaggregation as [36] can be found in

the literature.

In this section we will explain several different algorithms that have been proposed (in more or less detail) for microaggregation. Firstly we will explain a deterministic and optimal algorithm for univariate microaggregation; it will also be used later on when we will explain two methods for projection based multivariate microaggregation: PCP microaggregation and Zscores microaggregation. Finally, we will describe one of the most used methods for heuristic microaggregation (specially for the multivariate case, although it can be applied to the univariate case as well): the MDAV (Maximum Distance to Average Vector) algorithm.

Optimal Univariate Microaggregation

In [36] optimal univariate microaggregation is defined as the univariate microaggregation which minimizes the Sum of Square Errors (SSE):

$$SSE = \sum_{i=1}^{C} \sum_{x_{ij} \in c_i} (x_{ij} - \bar{x}_i)^T (x_{ij} - \bar{x}_i) \tag{2.1}$$

where C is the total number of clusters, c_i is the i-th cluster and $\bar{x}_i$ is the centroid of c_i. The restriction is $|c_i| \geq k$, for all $i = 1, \ldots, C$.

In [19], the authors present two results for the optimal univariate microaggregation:

- **Result 1.** When elements are sorted according to an attribute, for any optimal partition, elements in each cluster are contiguous (non overlapping clusters exist)
- **Result 2.** All clusters of any optimal partition have between k and $2k-1$ elements.

This method for optimal univariate microaggregation is as follows:

Let $A = (a_1 \ldots a_n)$ be a vector of size n containing all the values for the attribute being protected. The values are sorted in ascending order so that if $i < j$ then $a_i \leq a_j$. Obviously, a_1 is the smallest element in A and a_n is the largest element in A. Let k

be an integer such that $1 \leq k < n$ (k is directly obtained from the microaggregation configuration).

Given A and k, a graph $G_{k,n}$ is defined as follows. Firstly, we define the nodes of G as the elements a_i in A plus one additional node g_0 (this node is later needed to apply the Dijkstra algorithm). Then, for each node g_i, we add to the graph the directed edges (g_i, g_j) for all j such that $i + k \leq j < i + 2k$. The edge (g_i, g_j) means that the values $(a_i, \ldots, a_j)$ might define one of the possible clusters. Then, the cost of the edge (g_i, g_j) is defined as the within-group sum of squared error for such cluster. That is, $SSE = \Sigma_{l=i}^{j}(a_l - \bar{a})^2$, where $\bar{a}$ is the average record of the cluster.

Given this graph, the optimal univariate microaggregation is defined by the shortest path algorithm between the nodes g_0 and g_n. This shortest path can be computed using the Dijkstra algorithm.

Original Data Set X		Protected Data Set X'	
$attr_1$	$attr_2$	$attr'_1$	$attr'_2$
1	4	2	5
2	15	2	15.5
3	5	2	5
6	17	6.5	17.5
7	6	6.5	5
8	18	8.5	17.5
9	16	8.5	15.5

Table 2.2: Optimal univariate microaggregation example.

Example 2.2 *Let us consider the data set shown in the left side of Table 2.2. Then, we protect this data set using the optimal microaggregation protection method with k = 2. The protected data set is shown in the right side of the same table. In this example the SSE value is equal to 0.14. Note that here k-anonymity is only preserved in records 1 and 3.*

Projection Based Microaggregation

The basic idea of two of the microaggregation methods that we analyze herein is to project $a > 1$ attributes (corresponding to some attributes of the records) into a

single one. In this way we reduce the multivariate microaggregation problem into the univariate one. The employed projection should maintain as much as possible the global statistical properties of the initial (non-projected) values. With this goal in mind, two projection methods seem particularly appealing; we explain them now.

Principal Component Projection

Formally Principal Component Projection (PCP in short) works as follows: let us assume that values of a attributes for n individuals are stored in a matrix X of dimension $n \times a$, where columns contain attributes and rows contain individuals. For the sake of simplicity, we will assume here that data is standardized (*i.e.*, the data has $\mu = 0$ and $\sigma = 1$, and so the covariance matrix is $S = 1/nX^T X$).

The first principal component is defined as the linear combination of the attributes which has the maximum variance. Therefore, this first principal component will be represented using a vector $z_1 = Xa_1$, for some vector a_1 with a components, to be found. Since the original values have $\mu = 0$, we have that z_1 also has $\mu = 0$, and its variance is

$$\frac{1}{n} z_1^T z_1 = \frac{1}{n} a_1^T X^T X a_1 = a_1^T S a_1 \tag{2.2}$$

Since S is positive-definite, the variance increases when the module of the vector a_1 does. For this reason, to find a concrete solution for the maximization of Expression (2.2), some constraint on the module of a_1 is needed; in this case, the search is limited to vectors a_1 with module 1 (*i.e.* $a_1^T a_1 = 1$). This is equivalent to maximize the following expression, where a Lagrange multiplier has been added to the variance:

$$M = a^T S a_1 - \lambda(a_1^T a_1 - 1) \tag{2.3}$$

To maximize Expression (2.3), the derivative with respect to the a_1 components must be made equal to 0.

$$\frac{\partial M}{\partial a_1} = 2Sa_1 - 2\lambda a_1 = 0 \tag{2.4}$$

The solution for such equation is $Sa_1 = \lambda a_1$, which implies that a_1 is an eigenvector

of the matrix S, and λ is its corresponding eigenvalue. To determine which eigenvalue of S is the right solution, Equation (2.4) is left-multiplied with a_1^T, leading to

$$a_1^T S a_1 = \lambda a_1^T a_1 = \lambda.$$

Summing up, λ is the variance of z_1. Since the goal is to maximize the variance, λ is the largest eigenvalue of the matrix S, and its associate eigenvector a_1 defines the coefficients of the projection (PCP). Therefore, the final projected value is

$$PCP = \sum_{i=1}^{a} a_i x_i.$$

Zscores Projection

As in the previous section, we assume that values of the a attributes for the n individuals are stored in a matrix X of dimension $n \times a$. Given a record (a row) $(x_1, x_2, \ldots, x_a)$ in X, the sum of Zscores Projection is defined as the single element

$$Z = \sum_{i=1}^{a} \frac{x_i - \mu_i}{\sigma_i}$$

where μ_i is the average and σ_i is the variance of the i-th attribute, computed by taking into consideration all the records in X.

Algorithm for Projected Microaggregation

The main problem when one tries to extend the optimal univariate solution to the case of multivariate microaggregation is how to sort multivariate data. One possibility, as we will see later, is to order the points with respect to their distance to the global centroid of the data. MDAV is an heuristic microaggregation method that takes this information into account.

A different possibility is to reduce the dimensionality of the problem, from more than one attribute to 1 attribute, by applying some projection method. In more detail, projected multivariate microaggregation is described in Algorithm 2, when applied

Algorithm 2: Projected Microaggregation

Data: X: original data set, k: integer
Result: X': protected data set

begin
  Split the data set X into r sub-data sets $\{X_i\}_{1\leq i\leq r}$, each one with a_i attributes of the n records, such that $\sum_{i=1}^{r} a_i = A$
  foreach $(X_i \in X)$ **do**
    Apply a projection algorithm to the attributes in X_i, which results in an univariate vector z_i with n components (one for each record)
    Sort the components of z_i in increasing order
    Apply to the sorted vector z_i the following variant of the univariate optimal microaggregation method explained in Section 2.4.2: use the algorithm defining the cost of the edges $\langle z_{i,s}, z_{i,t}\rangle$, with $s < t$, as the within-group sum of square error for the a_i-dimensional cluster in X_i which contains the original attributes of the records whose projected values are in the set $\{z_{i,s}, z_{i,s+1}, \ldots, z_{i,t}\}$
    For each cluster resulting from the previous step, compute the v_i-dimensional centroid and replace all the records in the cluster by the centroid
end

to a data set X with n records and A attributes.

Depending on the projection method which is applied to the attributes, we will obtain different methods of multivariate microaggregation. Due to the fact that they should preserve as much statistical properties of the data as possible (desirable in the scenario of statistical data protection), the PCP and Zscores projection methods seem to be the best choice. We call the resulting microaggregation algorithms *PCP microaggregation* and *Zscores microaggregation*.

Example 2.3 *Let us consider the same data set used in Example 2.2, the original data set shown in the left side of Table 2.3. Then, we protect this data set using the PCP and Zscores microaggregation with* $k = 2$. *The protected data sets are shown in the middle and right side of the same table. In this example, SSE value is equal to 1.02 and 0.65 for PCP and Zscores microaggregation respectively.*

Original Data Set X		PCP Prot. Data Set X'		Zscores Prot. Data Set X'	
$attr_1$	$attr_2$	$attr'_1$	$attr'_2$	$attr'_1$	$attr'_2$
1	4	3.67	5.0	2.0	4.5
2	15	4.0	16.0	4.5	10.5
3	5	3.67	5.0	2.0	4.5
6	17	4.0	16.0	7.67	17.0
7	6	3.67	5.0	4.5	10.5
8	18	8.5	17.0	7.67	17.0
9	16	8.5	17.0	7.67	17.0

Table 2.3: Projection based microaggregation example.

Algorithm 3: MDAV

Data: X: original data set, k: integer
Result: X': protected data set

```
1  begin
2      while (|X| > k) do
3          Compute the average record x̄ of all records in X
4          Consider the most distant record x_r to the average record x̄
5          Form a cluster around x_r. The cluster contains x_r together with the
           k − 1 closest records to x_r
6          Remove these records from data set X
7          if (|X| > k) then
8              Find the most distant record x_s from record x_r
9              Form a cluster around x_s. The cluster contains x_s together with
               the k − 1 closest records to x_s
10             Remove these records from data set X
11         Form a cluster with the remaining records
12 end
```

MDAV Microaggregation

The MDAV (Maximum Distance to Average Vector) algorithm [19, 42] is an heuristic algorithm for clustering records in a data set X so that each cluster is constrained to have at least k records. This algorithm can be used for univariate microaggregation and multivariate microaggregation. The MDAV algorithm is described in Algotihm 3.

MDAV generic algorithm can be instantiated for different data types, using appropriate definitions for distance and average. Normally, *the most distant record* and the *closest records* are computed using the Euclidean distance, and *the average record* is

defined as the arithmetic mean of the records. This same mean record is used to replace the original records when building the protected data set.

Original Data Set X		Protected Data Set X'	
$attr_1$	$attr_2$	$attr'_1$	$attr'_2$
1	4	1.5	4.5
2	15	1.5	12.33
3	5	5.33	4.5
6	17	5.33	17.5
7	6	5.33	12.33
8	18	8.5	17.5
9	16	8.5	12.33

Table 2.4: MDAV microaggregation example.

Example 2.4 *Let us consider the same data set used in Example 2.2 and 2.3, the original data set is shown in the left side of Table 2.4. Then, we protect this data set using the MDAV microaggregation protection method with* $k = 2$*. The protected data set is shown in the right side of the same table. In this example, SSE value is equal to 0.49.*

2.5 Information Loss and Disclosure Risk

The main objective of rank swapping and microaggregation, and in general of all protection methods, is to minimize both *disclosure risk* (DR) and *information loss* (IL) of the protected released data set. Disclosure risk measures the capacity of an intruder to obtain some information about the original data set from the protected one, and information loss measures the reduction of the statistical utility of the protected data set with respect to the original one.

However, when one of these parameters decreases the other one increases; finding the optimal combination of these two measures becomes a difficult and challenging task. Moreover, in some situations, an organization could be interested in releasing the data by fixing a desirable level for one of the parameters. For these two reasons, it becomes necessary to compute both measures in a very accurate manner before releasing the protected data set, ensuring an enough protection level and statistical

utility.

Some approaches are used to calculate the information loss. In [20] the authors calculate the average difference between some statistics computed on both the original and the protected microdata. A probabilistic variation of these measures was presented in [43] to ensure that the information loss value is always within the interval [0,1]. A different approach was presented in [7], where some measures (*accuracy*, *completeness* and *consistency*) are calculated over the protected data to evaluate the information loss.

In order to compute the disclosure risk, many works as *e.g.* [20, 58, 79] use the *record linkage* methods [20, 75, 76] explained before. Alternatively, other methods can be considered for evaluating the disclosure risk. For example, in [73], the authors define a framework for privacy protection where the intruder can only query the database by using propositional sentences. If the database answers these queries with enough level of generalization, it is difficult for the intruder to infer any confidential information about a specific individual. The measure of disclosure risk in this scenario is the percentage of individuals for which an intruder is able to discover the value of a confidential attribute.

Normally, information loss and disclosure risk are combined to obtain an overall value about a specific protection method, this value weighs the relationship between the information loss and disclosure risk. The best protection method is the one that optimizes the trade-off between both magnitudes. Consider the following extreme cases as examples of this trade-off:

- If masking consists of encrypting the original data, no disclosure is possible, but no information at all is released (maximum information loss, minimum disclosure risk).
- If no masking is performed and the original data are released, users can perform fully accurate computations, but disclosure of individual respondent data is complete (minimum information loss, maximum disclosure risk).

In order to compute this trade-off, one approach was presented in [20], where the authors combine both IL and DR in a *Score* using an arithmetic mean. Another approach is the R-U (risk-utility) maps [26, 27, 28], that show in a graphical way the

relationship between a numerical measure of statistical disclosure risk (R) and a numerical measure of data utility (U). Both measures, R and U, can be general or specific for a certain protection method.

Among all of these possibilities, we have selected the measures presented in [20]. The selection is based on the following reasons:

- These measures use the record linkage methods to compute the disclosure risk.
- A lot of protection methods have been evaluated using this score and therefore we can compare our results with many other works easily.
- These measures allow modifications in the IL and DR computation.

In the remaining of this section, we describe the five information loss measures used to calculate the overall information loss value and the three disclosure risk measures used to compute the overall disclosure risk value of the final score.

2.5.1 Information Loss Measures

Let n be the number of records in the original data set and n' the number of records in the masked data set. Let a be the number of attributes (assumed to be the same in both data sets). Then, we define X and X' as a $n \times a$ matrices representing the original and the masked data set: columns correspond to attributes and rows correspond to records.

- IL_1. We define the mean absolute error of a matrix X vs another matrix X' as the average of the absolute values of differences of corresponding components (records) in both matrices. We understand 'corresponding' as the map of each record (component) in X with the nearest record in X' using a a-dimensional Euclidean distance. Then, we define the mean variation of X vs X' as

$$IL_1 = X - X' = \frac{\sum_{j=1}^{a} \sum_{i=1}^{n} \frac{|x_{ij} - x'_{ij}|}{|x_{ij}|}}{na}$$

- IL_2. Let $\bar{X}$ and $\bar{X}'$ be the vectors of averages of attributes (rows) in X and X', we define the mean variation of these two vectors as

$$IL_2 = \bar{X} - \bar{X}' = \frac{\sum_{j=1}^{a} \frac{|\bar{x}_j - \bar{x}'_j|}{|\bar{x}_j|}}{a}$$

- IL_3. Let V and V' be the covariance matrices of X and X', we compute the mean variation of these two matrices as

$$IL_3 = V - V' = \frac{\sum_{j=1}^{a} \sum_{1 \le i \le j} \frac{|v_{ij} - v'_{ij}|}{|v_{ij}|}}{\frac{(a+1)a}{2}}$$

- IL_4. Let S and S' be the vectors of variances of attributes (rows) in X and X', these vectors are the diagonal of V and V' respectively; we compute the mean variation of these two vectors as

$$IL_4 = S - S' = \frac{\sum_{j=1}^{a} \frac{|v_{jj} - v'_{jj}|}{|v_{jj}|}}{a}$$

- IL_5. Let R and R' be the correlation matrices of X and X', we compute the mean variation of these two matrices as

$$IL_5 = R - R' = \frac{\sum_{j=1}^{a} \sum_{1 \le i \le j} |r_{ij} - r'_{ij}|}{\frac{(a-1)a}{2}}$$

The overall IL is computed using 100 times the average of the mean variations of all the measures explained before. That is,

$$IL = 100\big(0.2\,IL_1 + 0.2\,IL_2 + 0.2\,IL_3 + 0.2\,IL_4 + 0.2\,IL_5\big)$$

2.5.2 Disclosure Risk Measures.

Two types of disclosure risk measures are considered depending on the intention of the intruder.

Firstly, we suppose that an intruder has the protected information and knows some original attributes obtained from an external data source, this scenario is defined in Section 2.4. Here, the intruder is interested in linking the original and the protected data set (*i.e.* discover the values of some other attributes). This risk can be measured using record linkage. Two record linkage methods defined in Section 2.3 are used for this purpose:

- **Distance-based Linkage Disclosure (DLD).** This measure is computed over the number of attributes that the intruder is assumed to know that, for instance, from one to half of the attributes. The final value is calculated as the average percentage of linked records using distance based record linkage in each case.

- **Probabilistic Linkage Disclosure (PLD).** This measure is identical to DLD, but using the probabilistic record linkage instead of the distance based record linkage.

Secondly, we suppose that the intruder is not interested in knowing the exact original values or that he cannot obtain them. Alternatively, the intruder tries to get an approximation of the original values. *Interval Disclosure* (ID) is one of the approaches to model this scenario. The ID risk is computed as 100 times the average percentage of original values falling into an interval defined around the corresponding masked value. The interval is defined as a percentage, between 1 per cent and 10 per cent, of the values.

$$DR = 0.5\,\frac{DLD + PLD}{2} + 0.5\,ID$$

2.5.3 Score Computation

The score combining information loss measures with disclosure risk measures is then defined as follows:

$$score = 0.5\,IL + 0.5\,DR$$

2.6 Data Sets Description

In this section, we describe in detail the data sets used in the experiments performed in this thesis. We have considered seven data sets from the UCI repository [46] and the two reference data sets proposed in the European CASC project. Both groups of data sets have been widely used in many other works. For example, the CASC reference data sets have been used in the following works: [21, 68, 74], whereas some of the UCI data sets have been used in these other works: [62, 67]

2.6.1 CASC Data Sets

Here, we describe the two reference data sets proposed in the European CASC (Computational Aspects of Statistical Confidentiality) project [10].

The Census Data Set

The first data set, called Census, contains 1080 records consisting of 13 numerical attributes. It was extracted using the Data Extraction System of the U.S. Census Bureau [71]. A complete description about the details of the construction of this data set can be found in [25].

The data used to create this data set was extracted from the file-group 'March Questionnaire Supplement - Person Data Files' of the data source 'Current Population Survey of the year 1995'. Not all the records of this survey were selected. Records with zero or missing values for at least one of the 13 attributes were discarded to obtain the final 1080 records. Note that, 1080 is the largest integer less than 1200 which is a multiple of 2, 5, 8 and 9. Thus, the data set can be split or microaggregated into several groups of small size.

The attributes selected to build the Census data set are described in Table 2.5.

The EIA Data Set

The second data set, called EIA, was obtained from the U.S. Energy Information Authority [72]. It contains 4092 records consisting of 15 attributes, but only 10 attributes

id	Name	Description
a1	AFNLWGT	Final weight (2 implied decimal places)
a2	AGI	Adjusted gross income
a3	EMCONTRB	Employer contribution for health insurance
a4	ERNVAL	Business or farm net earnings in 19
a5	FEDTAX	Federal income tax liability
a6	FICA	Social security retirement payroll deduction
a7	INTVAL	Amount of interest income
a8	PEARNVAL	Total person earnings
a9	POTHVAL	Total other persons income
a10	PTOTVAL	Total person income
a11	STATETAX	State income tax liability
a12	TAXINC	Taxable income amount
a13	WSALVAL	Amount: Total wage & salary

Table 2.5: Attributes of the Census data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

are numerical. As we are only interested in numerical attributes, we have discarded the 5 non numerical attributes. In Table 2.6 we present the description of the EIA attributes.

id	Name	Description
a1	RESREVENUE	Revenue from sales to residential consumers
a2	RESSALES	Sales to residential consumers
a3	COMREVENUE	Revenue from sales to commercial consumers
a4	COMSALES	Sales to commercial consumers
a5	INDREVENUE	Revenue from sales to industrial consumers
a6	INDSALES	Sales to industrial consumers
a7	OTHREVENUE	Revenue from sales to other consumers
a8	OTHRSALES	Sales to other consumers
a9	TOTREVENUE	Revenue from sales to all consumers
a10	TOTSALES	Sales to all consumers

Table 2.6: Attributes of the EIA data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

2.6.2 UCI Data Sets

Now, we describe the seven data sets extracted from the UCI (University of California - Irvine) Machine Learning Repository [46]. As in this thesis we are only interested in numerical data, we have selected data sets from UCI repository described in terms of numerical attributes. Non-numerical attributes, if any, were discarded.

The Abalone Data Set

The Abalone data set was obtained from the Marine Research Laboratories of Taroona. Firstly, it was used to predict the age of abalones (a kind of mollusks) from physical measurements. It contains 4177 records consisting of 8 numerical attributes. In Table 2.7 we present the description of the Abalone attributes.

id	Name	Description
a1	SEX	Male (1.0), female (2.0) and infant (3.0)
a2	LENGHT	Longest shell measurement
a3	DIAMETER	Diameter perpendicular to length
a4	HEIGHT	Height with meat in shell
a5	WHOLEWEIGHT	Weight of the whole abalone
a6	SUCKEDWEIGHT	Weight of meat
a7	VISCERAWEIGHT	Gut weight (after bleeding)
a8	SHELLWEIGHT	Weight after being dried
a9	RINGS	Number of rings (+1.5 gives the age in years)

Table 2.7: Attributes of the Abalone data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

The Dermatology Data Set

The Dermatology data set was obtained from the School of Medicine of Gazi University (Turkey). The aim of this data set is to determine the type of Eryhemato-Squamous Disease. It contains 366 records consisting of 34 numerical attributes. In this thesis, we have only used 16 attributes. Attribute selection was done on the basis of the correlation coefficients. In particular, attributes with a low correlation

coefficient (less than 0.7) with all the other attributes were discarded. In Table 2.8 we present the description of the Dermatology attributes.

id	Name	Description
a1	POLPAP	Polygonal papules
a2	FOLPAP	Follicular papules
a3	ORAL	Oral mucosal involvement
a4	KNEEINVOL	Knee and elbow involvement
a5	SCALP	Scalp involvement
a6	MELANIN	Melanin incontinence
a7	EXO	Exocytosis
a8	FOCAL	Focal hypergranulosis
a9	FOLHORN	Follicular horn plug
a10	CLUBBING	Clubbing of the rete ridges
a11	ELONGATION	Elongation of the rete ridges
a12	THIN	Thinning of the suprapapillary epidermis
a13	VACUOL	Vacuolisation and damage of basal layer
a14	TOOTH	Saw-tooth appearance of retes
a15	PERI	Perifollicular parakeratosis
a16	INFILTRAT	Band-like infiltrat

Table 2.8: Attributes of the Dermatology data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

The Housing Data Set

The Housing data set was taken from the StatLib library which is maintained at Carnegie Mellon University. This data set concerns about housing values in suburbs of Boston. It contains 506 records consisting of 7 numerical attributes. In this thesis, we have only used 16 attributes. Attribute selection was done using the same criteria than in the Dermatology data set. In Table 2.9 we present the description of the Housing attributes.

The Ionosphere Data Set

The Ionosphere data set was taken from Johns Hopkins University. This data set was used to obtain a classification of radar returns from the ionosphere using neural net-

id	Name	Description
a1	INDUS	Proportion of non-retail business acres per town
a2	RM	Average number of rooms per dwelling
a3	AGE	Proportion of owner-occupied units built prior to 1940
a4	RAD	Index of accessibility to radial highways
a5	NOX	Nitric oxides concentration (parts per 10 million)
a6	TAX	Full-value property-tax rate per $10,000
a7	MEDV	Median value of owner-occupied homes in $1000's

Table 2.9: Attributes of the Housing data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

works. It contains 351 records consisting of 35 numerical attributes. In this thesis, we have only used 12 attributes. In Table 2.10 we present the identifier of the Ionosphere attributes, no description about the attributes was given in the UCI database.

id	a1, a2, a3, a4, a5, a6 a7, a8, a9, a10, a11, a12
name	V_5, V_7, V_9, V_{11}, V_{13}, V_{20} V_{15}, V_{17}, V_{19}, V_{21}, V_{23}, V_{30}

Table 2.10: Attributes of the Ionosphere data set.

The Iris Data Set

The Iris plant data set was collected in 1935 by E. Anderson in [4]. This is perhaps one of the best known data set to be found in the pattern recognition literature, it has been used in more than 100 articles, it was first time used in [32]. It contains 150 records consisting of 4 numerical attributes. In Table 2.11 we present the description of the Iris attributes.

The Water Treatment Data Set

The Water Treatment data set was extracted from the Unitat d'Enginyeria Química of the Universitat Autònoma de Barcelona. This data set concerns about faults in a urban waste water treatment plant. It contains 527 records consisting of 38 numerical

id	Name	Description
a1	SEPLEN	Sepal length in cm
a2	PETLEN	Petal length in cm
a3	SEPWID	Sepal width in cm
a4	PETWID	Petal width in cm

Table 2.11: Attributes of the Iris data set. In the first column, id stands for the attribute identifier used in this thesis, in the second column, Name stands for the identifier used in the source of the data set and in the third column a brief description of the attribute is given.

attributes. In this thesis, we have only used 12 attributes. In Table 2.12 we present the description of the Water Treatment attributes.

id	Name	Description
a1	PH-E	Input pH to plant
a2	DBO-E	Input biological demand of oxygen to plant
a3	SS-E	Input suspended solids to plant
a4	SSV-E	Input volatile supended solids to plant
a5	SED-E	Input sediments to plant
a6	COND-E	Input conductivity to plant
a7	DBO-D	Input biological demand of oxygen to secondary settler
a8	SSV-D	Input volatile supended solids to secondary settler
a19	DBO-S	Output biological demand of oxygen
a10	RD-DBO-S	Performance input biological demand of oxygen to sec. settler
a11	RD-DQO-S	Performance input chemical demand of oxygen to sec. settler
a12	PH-P	Input pH to primary settler
a13	DBO-P	Input biological demand of oxygen to primary settler
a14	SS-P	Input suspended solids to primary settler
a15	SSV-P	Input volatile supended solids to primary settler
a16	SED-P	Input sediments to primary settler
a17	COND-P	Input conductivity to primary settler
a18	PH-D	Input pH to secondary settler
a19	DQO-D	Input chemical demand of oxygen to secondary settler
a20	COND-D	Input conductivity to secondary settler
a21	SS-S	Output suspended solids
a22	SED-S	Output sediments
a23	COND-S	Input conductivity to secondary settler
a24	RD-DBO-G	Global performance input biological demand of oxygen
a25	RD-DQO-G	Global performance input chemical demand of oxygen

Table 2.12: Attributes of the Water Treatment data set.

The WDBC Data Set

The WDBC (Wisconsin Diagnostic Breast Cancer) data set was extracted from the General Surgery Department of the University of Wisconsin. This data set describes a digitized image of a fine needle aspirate (FNA) of a breast mass. Data describes characteristics of the cell nuclei present in the image. It contains 569 records consisting of 32 numerical attributes. In this thesis, we have only used 22 attributes. In Table 2.13 we present the identifier of the WDBC attributes, no description about the attributes was given in the UCI database.

id	a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12 a13, a14, a15, a16, a17, a18, a19, a20, a21, a22
name	V_2, V_4, V_6, V_8, V_{10}, V_{12}, V_{13}, V_{18}, V_{20}, V_{26}, V_{29}, V_{32} V_5, V_9, V_{15}, V_{16}, V_{19}, V_{22}, V_{25}, V_{27}, V_{28}, V_{30}

Table 2.13: Attributes of the WDBC data set.

Chapter 3

Microaggregation Analysis

This chapter is divided into three different parts. Firstly, we present some results about attribute selection in multivariate microaggregation. Secondly, we describe the application of aggregation functions to projected microaggregation. We show that our new microaggregation technique achieves a lower disclosure risk than classical projected microaggregation. Finally, we present a new microaggregation method to reduce the disclosure risk of multivariate microaggregation.

3.1 Attribute Selection in Multivariate Microaggregation

As we have said in the preliminaries, microaggregation is one of the most popular studied and used microdata protection methods. There are many factors studied in detail which influence the final result of applying microaggregation to a data set: the value of the parameter k, the specific microaggregation method, the number of blocks into which the data set is split (and the number of attributes in each block). In addition to these ones, there is another factor which should be considered and that, up to our knowledge, has not been carefully studied before: how to select which attributes will form each block.

In this section we study this issue in detail and show that the result (statistical utility

and privacy/anonymity levels) of applying microaggregation to a data set can significantly vary according to the grouping strategy. We concentrate on two grouping strategies. The first one, widely accepted by statistical agencies, is focused on the maximization of the statistical utility. That is, (highly) correlated attributes are grouped in the same block(s) so that the distance between the original elements and the protected ones is small. The second strategy, which we propose here for the first time, consists of scattering the groups of correlated attributes into different blocks. This strategy is defined with the goal of obtaining *correlated blocks* so that a higher level of anonymity can be maintained. For example, when two records are in the same cluster for one block, and the blocks of attributes (as a whole) are *correlated* to each other, then these two records are likely to fall in the same cluster for all the other blocks. This would lead to two identical protected records. In other words, the idea of this new strategy is to enjoy some anonymity (higher privacy) even in the case in which attributes are microaggregated by blocks (higher data utility).

We have tested these two strategies with real data sets. In order to see the differences between the two strategies more clearly, we have chosen data sets with strong correlations between some of the attributes. The results of the experiments support our intuitions: the first strategy leads to a lower information loss, but it is more vulnerable to privacy attacks; the second strategy suffers from a higher information loss, but it maintains a higher level of anonymity, and so the disclosure risk is lower. The consequence is that one strategy or the other can be followed, depending on the scenario and on the importance given to data utility and privacy.

3.1.1 Specific Measures for Microaggregation

Some microdata protection methods admit specific measures to evaluate their quality. This is the case of microaggregation, whose goal is to minimize the total Sum of Square Error SSE (defined in Equation 2.1 in Section 2.4.2). Since there are no optimal solutions in polynomial time to multivariate microaggregation and the methods used are heuristic, the actual value of SSE for a given method is a measure of its quality.

Regarding privacy, microaggregation provides, by definition, some level of anonymity. If the method is applied to all the attributes (a single block), then the ini-

tial parameter k indicates the achieved anonymity: for each protected record, there are at least k possible original records which can correspond to it. However, if the original data set is split into r blocks and the microaggregation method is applied to each block separately, then the final level of anonymity obviously decreases: two records which are in the same cluster for one block of attributes may be in different clusters for other blocks, which results in two different protected records.

A possible way of computing the real level of anonymity achieved by a microaggregation method is to consider the ratio between the total number n of records and the number of protected records which are different. This gives the average size of each 'global cluster' in the protected data set. We denote as k' this *real anonymity* measure

$$k' = \frac{n}{|\{x'|x' \in X'\}|}$$

In the (unrealistic) case where all the entries of the data set X are random and independent, counting the expected number of different protected records is equivalent to counting the expected number $s(m,n)$ of distinct elements in a sample of n elements extracted, with replacement, from a universe of m elements. In our case, the universe of m elements contains the $m = \lceil \frac{n}{k} \rceil^r$ different possible configurations for a protected record, where k is the initial anonymity parameter, and r is the number of blocks. The exact value of $s(m,n)$ is

$$s(m,n) = \frac{1}{m^n} \sum_{\ell=1}^{n} \binom{m}{\ell} \ell!\, \ell \sum_{\substack{i_1 + \ldots + i_\ell = n - \ell \\ i_j \geq 0}} \prod_{j=1}^{\ell} j^{i_j}$$

This value is quite hard to compute when m and n are large. Anyway, there are some tight bounds for $s(m,n)$ (see page 10 of [55], for example):

$$m(1 - e^{-m/n}) + 0.1839 \;\leq\; s(m,n) \;\leq\; m(1 - e^{-m/n}) + 0.3678. \tag{3.1}$$

The final value of k', in this unrealistic case of totally random entries, would be computed as $k' = n/s(m,n)$, taking $m = \lceil \frac{n}{k} \rceil^r$.

3.1.2 Strategies to Group Attributes in Microaggregation

To apply microaggregation to a data set X, we need to settle the method itself (*i.e.*, which variation we will apply), the parameter k, and the number of blocks the data set X is split into. However, these are not the only parameters to be considered when the number r of blocks is larger than 1. In this case, the way in which the attributes are grouped into blocks affects in an important way the results and the quality of the microaggregation.

It is standard practice to select the attributes on the basis of statistical utility. It is clear that if highly correlated attributes are considered, records similar with respect to one attribute will be similar with respect to another one. Due to this, if microaggregation is applied to correlated attributes, clusters will contain records that are similar with respect to all the attributes included in the cluster. Therefore, this approach results in microaggregation with low information loss.

Nevertheless, as usual, statistical utility and privacy are inversely related terms. Experiments in Section 3.1.4 show that, as expected, the disclosure risk of microaggregation in this case is higher than when correlated attributes are put into different blocks.

More specifically, we also study in Section 3.1.4 a different approach. Blocks are formed in such a way that the first attributes of all blocks are (highly) correlated, the second attributes of all blocks are (highly) correlated, and so on. In some way, we construct 'correlated blocks', instead of constructing blocks with correlated attributes. The goal of this new approach is to try to increase the resulting real anonymity k'. If two records A and B are in the same cluster for some blocks, this means that the first attribute values of these records are more or less close to each other, and the same for the second attribute of the block, etc. Then, when we consider another block, if the j-th attribute of this new block is (highly) correlated with the j-th attribute of the firstly considered block, records A and B will likely be close to each other as well, with respect to the attributes in the second block. Therefore, with some non-negligible probability, A and B will fall in the same cluster, again. Ideally, some records will fall inside the same clusters, for each block of attributes, and so the number of protected records which will be exactly equal will be higher, increasing in this way the real anonymity and the privacy level of the released data

set. Of course, the probability of maintaining a good level of anonymity decreases very quickly when the number r of blocks is high (remember the unrealistic but orientating formula for the expected size of the global clusters, stated in the example at the end of previous Section 3.1.1). But for small values of r, say $r = 2,3$, the difference between the two types of grouping strategies, in terms of the achieved real anonymity k', is appreciable, as we will see in our experiments in Section 3.1.4.

Before moving to these experiments involving real data sets, we want to illustrate the arguments explained above with two simple examples, where the two grouping strategies are easy to distinguish and lead to different results. In general, this will not be the case with real data sets, where it is not always easy to find enough (high) correlations between attributes, and so the differences between applying one grouping strategy or another may be slight.

3.1.3 Motivating Examples

We explain two unrealistic but illustrative ways to find examples of data sets for which the two grouping strategies are very different. In particular, the most popular strategy (first one) of grouping correlated attributes behaves worse than the second strategy (correlated blocks).

In the first example, the data set contains two different attributes, which are repeated (*i.e.*, we have four attributes in total), and so that the correlation between the two original attributes is zero. A simple way to artificially generate two completely uncorrelated attributes is to generate a random point (x, y) with two attributes, and then to include the four points $(x, y), (x, -y), (-x, y), (-x, -y)$ to the data set. Therefore, the total number of points will be a multiple of four. In our example, we have taken points (x, y) which are in the same circumference of radius 1; specifically, we have taken $(x, y) = (\cos\theta, \sin\theta)$, for $\theta = \pi/20, 3\pi/20, 5\pi/20, 7\pi/20, 9\pi/20$. The resulting 20 points, which are represented in Figure 3.1(a), form the two first attributes of the data set, which are then repeated to have four attributes $a1, a2, a3, a4$, such that $a1 = a3$, $a2 = a4$ and the correlation between $a1$ and $a2$ is zero. In this case, if we want to independently microaggregate two blocks of two attributes each, the two grouping strategies are clearly distinguishable.

In the first one (correlated attributes), we group $a1, a3$ on the one hand, and $a2, a4$

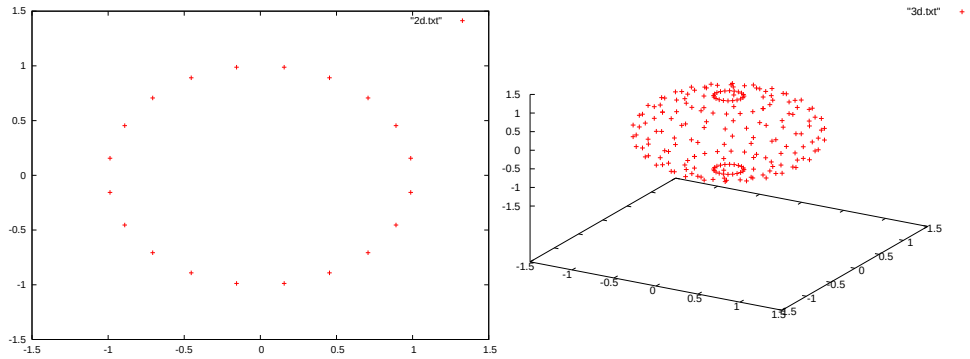


Figure 3.1: Points which have been artificially generated to obtain databases with non-correlated attributes, in 2 dimensions (a), and 3 dimensions (b).

on the other hand. We have applied the MDAV algorithm with $k = 2$. In this case, we have obtained no information loss, *i.e.* IL=0. However, the protected data set has real anonymity k' equal to 1. For this reason, the protected data set obtained using this attribute selection has a very high disclosure risk; for example the interval disclosure risk is maximum, ID=100, and the distance based linkage disclosure risk is DLD=75.00. If we compute the score, the measure explained in Section 2.5, we obtain Score=43.75.

Following the second strategy (correlated blocks), we group $a1, a2$ on the one hand, and $a3, a4$ on the other hand. We have applied the same microaggregation algorithm with the same parameterization than in the former case (MDAV with $k = 2$). Now, the information loss is equal to 26.38, quite higher than in the previous case. However, the disclosure risk is lower than in the case of correlated selection; for example, DLD=35.00 and ID=39.23. Summing up, the final score is 32.37, lower than in the correlated case. In other words, the trade-off between IL and DR is more in favour of the non-correlated case than of the correlated one.

The second example is in some way a generalization of the first one. Now the data set will contain three attributes which are repeated twice (nine attributes in total), such that the correlation between any two of the three initial attributes is zero. The way to generate three attributes with this property is the same as before: take a random point (x, y, z) and add to the data set the eight points $(x, y, z), (x, y, -z), (x, -y, z)$, $(x, -y, -z), (-x, y, z), (-x, -y, z), (-x, y, -z), (-x, -y, -z)$. Again, we have decided to

take points which are in the same sphere of radius 1; we have generated them as $(x, y, z) = (\cos\varphi\cos\theta, \cos\varphi\sin\theta, \sin\varphi)$, for $\theta, \varphi = \pi/20, 3\pi/20, 5\pi/20, 7\pi/20, 9\pi/20$. This gives us 25 initial points (x, y, z) and so 200 = 8·25 points in total, represented in Figure 3.1(b), which form the three first attributes of the data set. By repeating twice these three attributes, we obtain a data set with 200 records and nine attributes $a1, \ldots, a9$ such that $a1 = a4 = a7$, $a2 = a5 = a8$, $a3 = a6 = a9$, and such that the correlations between $a1$ and $a2$, between $a1$ and $a3$, and between $a2$ and $a3$, are zero. Suppose we want to microaggregate three blocks of three attributes each, with $k = 4$. Again, the two strategies lead to different results, which are very similar to the results obtained in the first example (two dimensions).

With the first strategy (correlated attributes), we group $(a1, a4, a7)$, $(a2, a5, a7)$ and $(a3, a6, a9)$. After applying the MDAV algorithm with $k = 4$, we obtain that the information loss is 0, but the disclosure risk is quite high; for example, the distance based linkage disclosure risk is DLD=55, and the interval disclosure risk is maximum, ID=100. The final value of the score in this case is 38.75.

The second strategy (correlated blocks) recommends to group $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$. We apply the same algorithm (MDAV) with $k = 4$, and now we obtain a non-negligible information loss, IL=31.52. However, the disclosure risk is lower, for example DLD=10 and ID=70.66, and the final score, 35.925, is better than the one obtained with the first strategy.

3.1.4 Experiments with Real Data Sets

id	Name	Description
$a1$	PH-E	Input pH to plant
$a2$	PH-P	Input pH to primary settler
$a3$	PH-D	Input pH to secondary settler
$a4$	DQO-E	Input chemical demand of oxygen to plant
$a5$	COND-P	Input conductivity to primary settler
$a6$	COND-D	Input conductivity to secondary settler
$a7$	DBO-S	Output biological demand of oxygen
$a8$	SS-S	Output suspended solids
$a9$	SED-S	Output sediments

Table 3.1: Attribute selection of the Water Treatment data set.

id	Name	Description
$a1$	UTILITYID	Unique utility identification number
$a2$	UTILNAME	Utility name
$a3$	YEAR	Reporting year for the data
$a4$	RESSALES	Sales to residential consumers
$a5$	COMREVENUE	Revenue from sales to commercial consumers
$a6$	COMSALES	Sales to commercial consumers
$a7$	MONTH	Reporting month for the data
$a8$	RESREVENUE	Revenue from sales to residential consumers
$a9$	INDREVENUE	Revenue from sales to industrial consumers

Table 3.2: Attribute selection of the EIA data set.

We have tested the two different strategies for attribute grouping with two real data set. The first one, denoted as Water Treatment data set, was extracted from the UCI repository [46], the second data set, called EIA, from the U.S. Energy Information Authority [72]. Both data sets are described in Section 2.6.

We have reduced both data sets to have only 9 attributes in order to form 3 blocks of 3 attributes each. In this scenario, it is easier to apply and compare the two attribute grouping strategies. Namely, if attributes $a1, a2, a3$ are highly correlated with each other, and the same happens for $a4, a5, a6$ on the one hand, and $a7, a8, a9$ on the other hand, the first strategy (correlated attributes) will lead to blocks $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$, whereas the second strategy (correlated blocks) will lead to blocks $(a1, a4, a7)$, $(a2, a5, a8)$ and $(a3, a6, a9)$.

In the case of the Water Treatment data set, there are many attributes (and possible groups) of highly correlated attributes. We have chosen the attributes presented in Table 3.1. In the case of the EIA data set we have chosen the attributes presented in Table 3.2

Tables 3.3 to 3.6 summarize the results of the experiments. We have applied to each data set the three microaggregation methods described in Section 2.4.2: MDAV, PCP and Zscores microaggregation. For each data set and method, we have tested five different parameterizations according to the initial value of k ($k = 5, 10, 15, 20, 25$ for the Water Treatment data set, and $k = 5, 25, 50, 75, 100$ for the EIA data set). Finally, we have run all these experiments for the two considered attribute grouping strategies: correlated attributes, where blocks are $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$, and non-correlated attributes (which corresponds to 'correlated blocks'), where blocks

	k	IL	DLD	PLD	ID	Score		k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	14.14	73.03	67.24	72.73	42.79	Mic.MDAV-k	5	31.75	8.16	39.87	45.79	33.32
	10	18.78	61.97	55.66	63.56	39.98		10	28.28	5.26	28.95	43.00	29.16
	15	17.34	49.74	43.95	56.99	34.63		15	35.60	2.50	18.82	41.89	30.94
	20	18.28	39.34	35.53	51.18	31.29		20	32.44	2.63	14.21	39.34	28.16
	25	21.68	32.37	29.08	48.59	30.67		25	36.74	1.71	12.89	30.85	27.91
Mic.PCP-k	0	18.36	40.39	30.39	60.82	33.23	Mic.PCP-k	5	50.41	8.95	2.11	36.63	35.74
	10	18.11	30.00	21.58	53.66	28.92		10	53.51	5.00	0.79	30.96	35.22
	15	21.67	23.82	20.39	50.54	29.00		15	56.28	4.21	1.32	30.37	36.42
	20	25.17	21.45	16.05	47.21	29.08		20	61.02	4.74	1.05	26.30	37.81
	25	23.25	19.08	13.68	49.34	28.05		25	62.48	3.82	0.26	25.61	38.15
Mic.Zscores-k	5	17.62	76.05	62.50	68.65	43.29	Mic.Zscores-k	5	98.33	10.13	3.16	42.96	61.57
	10	20.62	63.82	54.87	61.53	40.53		10	108.75	6.05	2.24	40.61	65.57
	15	20.99	54.08	47.76	56.42	37.33		15	113.74	5.39	1.71	40.18	67.80
	20	20.74	47.76	40.79	53.48	34.81		20	114.78	3.03	1.45	39.98	67.94
	25	24.30	43.95	34.47	54.04	35.46		25	113.71	3.29	1.05	37.60	66.80
	Correlated attributes							Non-correlated attributes					

Table 3.3: Scores of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

are $(a1, a4, a7)$, $(a2, a5, a8)$ and $(a3, a6, a9)$.

Firstly, we concentrate on the generic measures for the information loss and the disclosure risk (and so, the score). Table 3.3 shows the results obtained in the case of the Water Treatment data set. The differences between the two strategies are very evident, since the first one leads to much lower values of the information loss, whereas the second one leads to much lower values of the disclosure risk. For instance, by comparing the information loss of the Zscores microaggregation, correlated attributes selection obtains IL values between 17.62 and 24.30, whereas the non-correlated selection obtains values between 98.33 and 113.71. Regarding the three employed methods, MDAV has the best scores in the non-correlated scenario (27.91 is the best one, PCP and Zscores microaggregation always obtain scores over 35.00), whereas PCP microaggregation has the best scores in the correlated case (28.05 is the best one). The behaviour of Zscores microaggregation is quite surprising: it has quite good scores in the correlated case, but very bad scores (in particular, very high information loss) in the case of 'correlated blocks'.

Similar results are presented in Table 3.4, where the measures are computed for the

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	6.68	1.78	2.87	87.60	25.82
	25	11.83	0.78	0.68	79.55	25.99
	50	13.23	0.60	0.56	72.37	24.85
	75	15.56	0.53	0.55	72.16	25.95
	100	17.63	0.39	0.49	67.52	25.81
Mic.PCP-k	5	16.61	1.78	2.87	65.29	25.21
	25	18.33	0.78	0.68	61.62	24.76
	50	19.77	0.60	0.56	59.71	24.96
	75	21.16	0.53	0.55	59.67	25.63
	100	22.26	0.39	0.49	57.87	25.71
Mic.Zscores-k	5	12.58	6.17	8.36	75.09	26.88
	25	16.04	5.03	5.98	70.88	27.12
	50	16.84	4.92	5.60	69.36	27.07
	75	18.69	3.91	5.30	69.71	27.92
	100	18.86	3.48	4.63	67.78	27.39

Correlated attributes

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	10.05	1.61	2.43	83.30	26.36
	10	16.58	0.81	0.56	72.27	26.53
	15	21.86	0.62	0.49	67.16	27.86
	20	20.26	0.68	0.60	64.61	26.44
	25	25.14	0.60	0.48	61.29	28.02
Mic.PCP-k	5	19.37	0.62	0.54	57.35	24.17
	10	22.07	0.63	0.48	53.45	24.54
	15	22.25	0.64	0.48	52.41	24.37
	20	22.70	0.64	0.49	52.11	24.52
	25	23.07	0.66	0.50	50.93	24.42
Mic.Zscores-k	5	16.81	2.58	3.82	75.75	28.14
	10	17.06	1.92	2.55	75.21	27.89
	15	17.25	1.44	2.22	73.88	27.55
	20	17.83	1.25	2.14	73.71	27.77
	25	17.80	1.16	1.88	70.95	27.02

Non-correlated attributes

Table 3.4: Scores of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

EIA data set. Here, the comparison between correlated and non-correlated results is not as different as in the Water Treatment data set. In our opinion, this is so because the correlations among attributes are not so high. However, if one observes the information loss values presented in this table, it is easy to see that IL values are lower in the correlated case. See, for instance, the IL values in the MDAV microaggregation for the correlated selection. They are between 6.68 and 17.63. In contrast, for the non-correlated ones, IL values are between 10.05 and 25.14.

Regarding disclosure risk, we observe that non-correlated selection presents lower disclosure risk than correlated one. For instance, if one observes the values for the distance based and probabilistic record linkage (DLD and PLD) and interval disclosure (ID) for the Mic.PCP-5 configuration in the Water Treatment data set, it is clear that correlated selection has higher disclosure risk than non-correlated selection. In particular, DLD, PLD and ID values for the correlated case are 40.39, 30.39 and 60.82 respectively, whereas in the non-correlated case DLD, PLD and ID values are 8.95, 2.11 and 36.63.

Now, we consider the performance measures for microaggregation: the values of SSE

	k	SSE	k'		
r			1G	2G	3G
Mic.MDAV-k	5	28.18	5.28	1.02	1.00
	10	46.14	10.00	1.15	1.01
	15	72.03	15.20	1.38	1.01
	20	94.24	20.00	1.64	1.03
	25	114.56	25.33	2.11	1.09
Mic.PCP-k	5	28.59	5.35	1.04	1.00
	10	49.61	10.00	1.11	1.00
	15	71.99	15.20	1.30	1.01
	20	91.96	20.00	1.66	1.03
	25	110.91	25.33	2.09	1.04
Mic.Zscores-k	5	23.78	5.43	1.03	1.00
	10	49.05	10.00	1.16	1.01
	15	72.23	15.20	1.33	1.02
	20	93.10	20.00	1.62	1.03
	25	111.69	25.33	2.15	1.07

Correlated attributes

	k	SSE	k'		
r			1G	2G	3G
Mic.MDAV-k	5	69.51	5.00	1.16	1.01
	10	126.21	10.00	1.84	1.13
	15	173.96	15.20	2.66	1.38
	20	259.07	20.00	3.76	1.50
	25	247.58	25.33	4.87	1.87
Mic.PCP-k	5	93.67	5.07	1.04	1.00
	10	133.83	10.00	1.18	1.01
	15	170.12	15.20	1.41	1.02
	20	206.74	20.00	1.78	1.06
	25	229.50	25.33	2.16	1.12
Mic.Zscores-k	5	73.52	5.00	1.06	1.01
	10	115.77	10.00	1.35	1.05
	15	160.30	15.20	1.84	1.10
	20	197.20	20.00	2.59	1.26
	25	231.81	25.33	3.62	1.43

Non-correlated attributes

Table 3.5: SSE and real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the Water-treatment data set. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

and the real anonymity k'. We consider different situations where an intruder can have access to one (the first one), two (the first two ones) or the three blocks of protected data. 3.5 and 3.6 show the results for SSE and real anonymity k'.

Of course, if the intruder has access only to one block, then the real anonymity k' roughly coincides with the initial value of k. In fact, it is larger because for microaggregation with initial parameter k the number of records in a cluster is in the interval $[k, 2k)$. In the general case, the tables show that k' decreases rapidly with regards to the number of blocks considered. Also, as expected, k' is always larger when we consider correlated blocks. Note that the differences between the k' values of the two strategies are noticeable, specially, when only two blocks of attributes are considered, and when the initial anonymity value k is quite large. Furthermore, both strategies lead to higher values of k' than those which would be obtained in the 'unrealistic' totally random case introduced in the example in Section 3.1.3, as one should expect. For example, if we consider the Water Treatment data set (see Table 3.5) with two ($r = 2$) groups of attributes and $k = 25$, then the unrealistic case would

	k	SSE	k'		
r			1G	2G	3G
Mic.MDAV-k	5	28.74	5.12	1.10	1.03
	25	145.36	25.42	1.83	1.26
	50	219.45	50.52	3.20	1.57
	75	313.31	75.78	4.95	1.92
	100	397.48	102.30	7.54	2.57
Mic.PCP-k	5	141.29	5.27	1.02	1.00
	25	203.81	25.42	1.27	1.04
	50	330.19	50.52	1.91	1.15
	75	469.74	75.78	2.87	1.30
	200	649.49	102.30	4.28	1.53
Mic.Zscores-k	5	50.65	5.18	1.02	1.01
	25	140.70	25.26	1.44	1.06
	50	184.19	50.52	2.59	1.28
	75	287.95	75.78	4.45	1.58
	100	378.51	102.30	6.90	2.02

Correlated attributes

	k	SSE	k'		
r			1G	2G	3G
Mic.MDAV-k	5	45.18	5.01	1.15	1.06
	25	212.44	25.10	2.06	1.28
	50	361.38	50.52	4.43	1.81
	75	468.30	75.78	7.08	2.30
	100	569.66	102.30	10.94	3.17
Mic.PCP-k	5	124.83	5.14	1.02	1.01
	25	251.99	25.10	1.33	1.06
	50	369.73	50.52	2.16	1.20
	75	482.05	75.78	3.39	1.40
	100	608.82	102.30	5.00	1.73
Mic.Zscores-k	5	111.97	5.08	1.03	1.01
	25	212.47	25.10	1.65	1.12
	50	336.68	50.52	3.45	1.52
	75	439.99	75.78	6.13	2.18
	100	553.89	102.30	9.79	3.16

Non-correlated attributes

Table 3.6: SSE and real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the EIA data set. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

lead to a real anonymity k' between 1.915 and 1.918 (using the bounds for $s(m,n)$ given in Equation (3.1)), but the two realistic strategies lead to values around $k' = 2.1$ (for the first strategy) and values between 2.16 and 4.87 (for the second strategy).

SSE behaves more or less as the information loss: it is lower when the initial value of k is small, and it is lower in the correlated case than in the non-correlated case. The three microaggregation methods obtain very similar results for the SSE in both the correlated and non-correlated scenarios, so we cannot deduce from this experiment that any of them provides a better solution to the original microaggregation problem.

3.1.5 Attribute Selection Consequences

From the results obtained in the experiments, we can extract some consequences which are valid either for the microaggregation technique in general or for the specific strategies to group attributes in blocks.

The first of them is that the real anonymity that microaggregation provides, when the data set is split into blocks of attributes, decreases very quickly when the number of blocks increases, independently of the strategies for grouping attributes. For example, for standard values of the initial parameter k, less than 25, we observe that real anonymity is almost non-existent if the number of blocks is $r = 3$ (or more). Therefore, if k'-anonymity was the main motivation to choose microaggregation as a data protection method, one should either start with a large value for the initial k, or split the data set into only one or two blocks of attributes.

With respect to this, note, however, that microaggregation ranks among the best methods for data protection in [21] with respect to the trade-off between privacy and data utility. This is so, because even in the case that k-anonymity is not achieved, the perturbation added to the data might make re-identification difficult.

If we focus on the overall evaluation of the method taking into account all measures for information loss, disclosure risk, SSE and real anonymity, obtained by the two strategies, it is very difficult to conclude that one of them is better than the other. As expected, when blocks are formed by correlated attributes, we obtain better results in terms of the information loss and SSE. On the contrary, for 'correlated blocks', we obtain better results in real anonymity, and also in the disclosure risk. These aspects are more or less compensated when computing the final score for each case: the scores obtained by the two strategies are very similar.

The clear consequence of this analysis is that the strategy for grouping attributes is another degree of freedom for microaggregation that has to be considered with care. As shown in the simple (unrealistic) examples of Section 3.1.3, it might be even possible to have much better results if we use blocks with uncorrelated attributes.

Then, with real data, when choosing the value for k, one can take a small k if data utility is the main goal (at the cost of a lower level of privacy), and a larger k if privacy is the main concern. Analogously, one can microaggregate the whole data set as a single block, if privacy is considered to be more important than data utility; or one can form a higher number of blocks, if data utility is the most desired property of the protection. In the case of the grouping strategy selection, giving priority to data utility corresponds to choosing the first strategy, correlated attributes in the same block(s). This can be the case if the protected data is going to be released to a more or less reliable (or restricted) network. However, if the protected data is going to be

widely released, for example in the Internet, then maybe privacy is considered to be the main concern; in this case, the second strategy, 'correlated blocks', should be chosen, because it enjoys a higher anonymity level and a lower disclosure risk.

3.2 Modeling Projections in Microaggregation

As we have explained in the preliminaries, the main problem for extending optimal univariate microaggregation to the multivariate case is the sorting of multivariate data. One approach is to reduce the dimensionality of the problem. That is, to move from the case of several attributes into one attribute.

Projected microaggregation simplifies the multivariate microaggregation problem translating it into the univariate case. To do this, A attributes are summarized/represented into a single value in a projected axis. Normally, this summarization is done using the Principal Component Analysis or the sum of Zscores (both methods are described in the preliminaries). The aim of both methods is to establish an order among records to apply an optimal univariate microaggregation algorithm.

In order to summarize several attributes into a single value, aggregation functions can be used. In this section, we propose replacing the use of projection methods in microaggregation by the use of methods based on aggregation functions. We show that the trade-off between privacy and statistical utility achieved by microaggregation using the Sugeno integral (defined in Section 2.1) to summarize the attributes is equal, better in many cases, than the traditional projected microaggregation methods.

3.2.1 Algorithm Description

As we have explained before, projected microaggregation defines a sorting criterion over the multivariate data. Traditional projected microaggregation methods build a projected axis to establish an order among records. Here, we propose to do that using aggregation functions instead of building a projected axis. We propose to use aggregation functions over the records to be protected in order to compute a representative summarized value, and then, using such value, to sort the records in the

data set. Naturally, at that time, optimal univariate microaggregation methods can be applied. That is, we propose to use aggregation functions over the records to be protected in order to compute a representative summarized value and then using such value, to sort the records in the data set. Obviously, at that time, optimal univariate microaggregation methods can be applied.

This new approach has several advantages with regards to the traditional projected approach. We underline the following ones.

- In projected methods, we need to compute some parameters. For instance, the sum of Zscores calculates the average and the variance of all the attributes, PCP needs to solve an optimization problem. This is unnecessary using aggregation functions. Therefore, our new approach save execution time.
- Projected methods are not configurable. By using aggregation functions, one can define how data is sorted and, in some sense, protected.
- It is often the case that the projected values returned by a projection method are difficult to understand. Using aggregation functions one is able to understand the final summarized value for a concrete record.

Formally, the projected microaggregation is defined in Algorithm 4. Depending on the aggregation function used in this algorithm, we obtain different methods of modeling projection microaggregation. In this thesis, we use the *Sugeno microaggregation*. Such microaggregation uses the Sugeno integral with regards to the measure $\mu(A) = Q(|A|/N)$ for $A \subseteq X$ where $Q(x) = x$. A graphic representation of this measure is presented in Figure 3.2.

3.2.2 Experiments

We have protected two different data sets (Census and EIA data sets) with different instances of PCP, Zscores and Sugeno microaggregation methods. These data sets were proposed in the CASC project [10] as the reference data sets for comparing protection methods. Both data sets are described in Section 2.6.

Each of the three microaggregation methods has been applied with the following 9 parameterizations of the pairs (k,a): $k = 5, 15, 25$ for the minimal number of el-

Algorithm 4: Modeling Projection Microaggregation

Data: X: original data set, k: integer
Result: X': protected data set
1 **begin**
2 Split the data set X into r sub-data sets $\{X_i\}_{1\leq i\leq r}$, each one with a_i attributes of the n records and according to a partition $\{A_i\}_i$ of the attributes A
3 **foreach** *sub-data set* $X_i \in X$ **do**
4 Compute an aggregation function with the attributes A_i in X_i, which results in an univariate summarized vector p_i with n components (one for each record)
5 Sort the components of p_i in increasing order
6 Apply to the sorted vector p_i the univariate optimal microaggregation
7 For each cluster resulting from the previous step, compute the v_i-dimensional centroid and replace all the records in the cluster by the centroid
8 **end**

ements in the resulting clusters, and $a = 2,3,4$ for the number of attributes contained in each block of attributes to which microaggregation is applied. For example, Mic2.Zscores.15 refers to the Zscores microaggregation method applied to blocks of $a = 2$ attributes, with the constraint that resulting clusters must contain at least $k = 15$ records. When the total number of attributes is not a multiple of a (for example, this always happens with Census data set, since 13 is prime), the last non used attributes are non microaggregated and removed from the beginning.

For DLD, PLD and ID computation we have considered different cases, according to

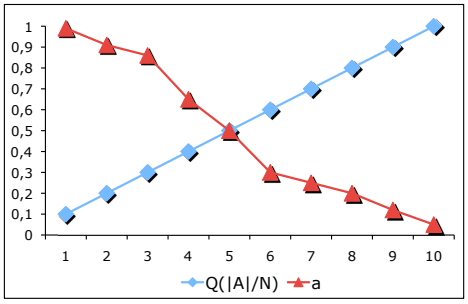


Figure 3.2: Quantifier $Q(A) = Q(|A|/N)$ where $Q(x) = x$ and a represents the values of one record of the data set.

	a	k	IL	DLD	PLD	ID	Score
Mic.a.PCP-k	2	5	13.90	2.94	6.91	70.04	25.69
	2	15	17.24	1.72	2.37	67.67	26.05
	2	25	19.98	1.42	1.58	67.21	27.17
	3	5	16.08	2.47	2.69	62.79	24.38
	3	15	17.76	1.49	1.21	59.41	24.07
	3	25	18.49	1.31	0.90	58.49	24.14
	4	5	18.25	4.23	4.81	73.22	28.56
	4	15	16.39	1.96	2.13	70.48	26.33
	4	25	17.27	1.93	1.91	69.66	26.53
Mic.a.Zscores-k	2	5	4.27	25.36	36.08	89.26	32.13
	2	15	5.08	21.92	34.37	87.85	31.54
	2	25	5.52	21.05	35.06	87.33	31.61
	3	5	13.24	6.40	9.26	72.31	26.66
	3	15	15.30	3.79	5.50	69.35	26.15
	3	25	15.73	3.21	5.02	68.65	26.06
	4	5	13.91	5.21	8.50	78.73	28.35
	4	15	21.79	2.71	4.40	77.41	31.14
	4	25	21.66	2.35	3.89	76.76	30.80
Mic.a.Sugeno-k	2	5	5.25	17.79	23.90	86.65	29.50
	2	15	6.24	14.14	19.11	85.08	28.55
	2	25	6.49	12.81	18.29	84.51	28.26
	3	5	17.22	3.78	6.89	65.52	26.32
	3	15	21.31	1.74	3.21	61.22	26.58
	3	25	20.08	1.47	2.65	60.83	25.76
	4	5	28.78	2.20	3.30	73.48	33.45
	4	15	35.97	0.71	1.03	70.86	35.92
	4	25	45.27	0.42	0.71	70.31	40.35

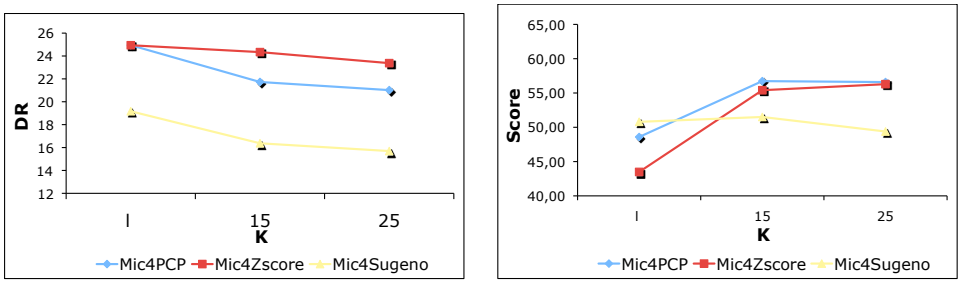
EIA data set

	a	k	IL	DLD	PLD	ID	Score
Mic.a.PCP-k	2	5	80.96	12.93	5.70	42.60	53.46
	2	15	92.94	8.46	2.94	35.64	56.81
	2	25	84.77	6.61	1.94	32.93	51.69
	3	5	57.72	10.15	5.71	43.48	41.71
	3	15	71.28	4.35	3.49	37.36	45.96
	3	25	72.49	4.07	2.65	35.51	45.96
	4	5	72.23	6.48	3.06	45.12	48.59
	4	15	91.74	3.43	2.04	40.73	56.74
	4	25	92.17	2.92	1.71	39.72	56.59
Mic.a.Zscores-k	2	5	81.57	16.78	7.85	48.27	55.93
	2	15	98.05	12.96	6.19	44.33	62.50
	2	25	100.92	12.85	4.83	42.90	63.40
	3	5	60.98	14.44	13.67	50.63	46.66
	3	15	75.21	9.38	10.46	45.71	51.51
	3	25	79.38	7.47	9.04	44.20	52.80
	4	5	62.04	11.71	7.04	40.50	43.49
	4	15	86.47	5.60	4.21	43.77	55.40
	4	25	89.20	4.40	3.38	42.86	56.29
Mic.a.Sugeno-k	2	5	73.44	9.63	6.00	40.75	48.86
	2	15	79.39	4.85	4.63	34.02	49.39
	2	25	73.43	3.72	4.76	32.96	46.02
	3	5	83.93	7.47	7.25	44.50	54.93
	3	15	122.52	3.55	5.52	39.47	72.26
	3	25	129.37	3.30	4.57	39.08	75.44
	4	5	82.43	3.15	0.51	36.51	50.80
	4	15	86.64	0.83	0.28	32.19	51.51
	4	25	83.08	0.60	0.37	30.90	49.39

Census data set

Table 3.7: Score of different microaggregation methods and parameterizations. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores or Sugeno) with $a = i$ and $k = j$.

the number of groups of attributes of the original record(s) to be linked, that the intruder knows. This number varies from 2 to the total number of attributes of each data set. The values in the table are the average of the obtained correct links for all these cases, for each parameterization of each microaggregation method. Figure 3.3.(a) and 3.3.(b) present in a graphical way disclosure risk (DR) and *score* for the microaggregation of the Census data set with $a = 4$ (the most protected configuration). We can observe that the Sugeno microaggregation algorithm obtains always the lowest DR and the best scores for $k = 15, 25$.



(a) (b)

Figure 3.3: Graphical representation of the DR (a) and Score (b) of (PCP, Zscores and Sugeno) microaggregation using $a = 4$ and $k = 5, 15, 25$.

In Table 3.7 we present the scores as well as the unaggregated components. It can be seen that for some cases, Sugeno microaggregation leads to the lowest score (and the same for its components). For example, the score obtained by Sugeno microaggregation method is 49.39 in the Census data set with $a = 4$ and $k = 25$, while using PCP and Zscores microaggregation, the values are around 56. It is similar for the IL and DR components (IL, DLD, PLD and ID values). The values for Sugeno microaggregation are 83.08, 0.60, 0.37 and 30.90, respectively better than for PCP and Zscores microaggregation (89.02, 4.40, 3.38 and 42.86 for Zscores microaggregation; 92.17, 2.92, 1.71, 39.72 and 39.72 for PCP microaggregation).

Another interesting result can be observed analyzing Table 3.7: our approach never obtains the worst results (neither score values nor its components) in any case. This fact indicates that the results of our new approach are more independent of the data set than projected microaggregation methods.

3.3 Improving Microaggregation for Complex Record Anonymization

As we have explained before, when records are complex, *i.e.*, the number of attributes of the data set is large, data sets are usually split into smaller blocks of attributes and microaggregation is applied to each block, successively and separately. In this way, information loss when collapsing several values to the centroid of their group is reduced, at the cost of losing the k-anonymity property when at least two attributes

of different blocks are known by the intruder.

In this section, we present a new microaggregation method called *one dimension microaggregation* (Mic1D-κ, for short). This method gathers all the values of the data set into a single sorted vector, independently of the attribute they belong to. Then, it microaggregates all the mixed values together. The experiments presented here show that, by using real data sets, our proposal obtains lower disclosure risk than previous approaches whereas the information loss is preserved.

3.3.1 One Dimension Microaggregation

As shown in Figure 3.4, the pre-processing data block of Mic1D-κ can be decomposed in several steps. Namely, vectorization, sorting, partitioning and normalization. Sorting, partitioning and normalization steps are repeated once. Then, we go into further details about these steps. We also illustrate this process by means of an illustrative example.

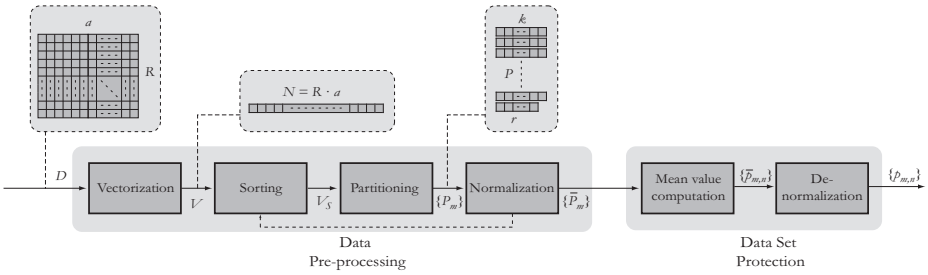


Figure 3.4: Mic1D-κ schema.

Vectorization

The vectorization step gathers all the values from the data set in a single vector, independently on the attribute they belong to. Thereby, we ignore the attribute semantics and therefore the possible correlation between two different attributes in the data set. In other words, we *desemantize* the microdata file. This process plays a central role in later discussion about the results achieved by Mic1D-κ.

Formally speaking, let $\mathscr{D}$ be the original data set to be protected. We denote by R the

number of records in $\mathscr{D}$. Each record consists of a numerical attributes. We assume that none of the records contains missing values. We denote by N the total number of values in $\mathscr{D}$. As a consequence, $N = R \cdot a$.

Let V be a vector of size N containing all the values in the data set. Mic1D-κ treats values in the data set as if they were completely independent. In other words, the concept of record and attribute is ignored and the N values in the data set are placed in V.

The effect of this step on a certain data set is depicted in the upper half of Figure 3.4.

Original data set $\mathscr{D}$			
Age	Height (cm)	Weight (kg)	Income (€)
23	159	52	12000
23	177	75	7000
55	173	79	50000
80	155	55	5000
30	180	70	30000

Table 3.8: Example of a microdata file used to illustrate the preprocessing block.

Example 3.1 *Let us consider the data set $\mathscr{D}$ shown in Table 3.8. According to the notation introduced we have $a = 4$, $R = 5$, and $N = 20$. The result of applying vectorization would be:*

V = [23 23 55 80 30 159 177 173 155 180 52 75 79 55 70 12000 7000 50000 5000 30000]

Sorting

Since the values in the vectorized data set belong to different source attributes, they present a pseudo-random aspect and it becomes very difficult to find the optimal partitions, *i.e.* partitions with SSE value as low as possible. In order to simplify this search, the whole vector is sorted. This way, by the result 1 of univariate microaggregation presented in Section 2.4.2, optimal partitions are contiguous and, therefore, the partitioning process in this new vector can be done easily, as we will see later.

Formally, V is sorted increasingly. Let us call V_s the sorted vector of size N containing

the sorted data and v_i the ith element of V_s, where $0 \leq i < N$.

Example 3.2 *Let us continue the process presented in Example 3.1, where we illustrate the pre-processing block applied to the data set shown in Table 3.8. The result of the sorting process applied to vector V would be:*

$$V_s = [\ \underset{v_0}{23}\ \underset{v_1}{23}\ \underset{v_2}{30}\ 52\ 55\ 55\ 70\ 75\ 79\ 80\ 155\ 159\ 173\ 177\ 180\ 5000\ 7000\ \underset{v_{17}}{12000}\ \underset{v_{18}}{30000}\ \underset{v_{19}}{50000}\ ]$$

Partitioning

Similarly to general microaggregation, in order to ensure a certain level of privacy (k-anonymity), Mic1D-κ splits the vectorized data set in several κ-partitions and it calculates the average value for each partition. By modifying the value of κ, Mic1D-κ allows us to adjust the trade-off between information loss (SSE) and disclosure risk. Note that if the vectorized data set was not sorted (previous step), κ would not have this property.

Formally, V_s is divided into smaller sub-vectors or partitions. We define κ where $1 < \kappa \leq N$ as the number of values per partition. Note that if κ is not a divisor of N, the last partition will contain a smaller number of values. Let P be the number of partitions containing κ values. We call r the number of values in the last partition where $0 \leq r < \kappa$. Therefore, $N = \kappa P + r$. We will suppose that $r > 0$, so we have $P + 1$ partitions (note that $r > 0$ if and only if κ does not divide N). We denote by P_m the mth partition.

Let $v_{m,n}$ be defined as the nth element of P_m:

$$\begin{cases} v_{m,n} := v_{m\kappa+n} & n = 0 \ldots \kappa - 1 \quad m = 0 \ldots P - 1 \\ v_{P,n} := v_{P\kappa+n} & n = 0 \ldots r - 1 \end{cases}$$

The upper half of Figure 3.4 shows the effect of this step on a certain data set.

Example 3.3 *The result of the partitioning process applied to vector V_s of Example 3.2 is presented below. We use $\kappa = 8$ (arbitrarily chosen) and thus, according to the notation introduced, the number of partitions containing 8 values is $P = 2$. Since $\kappa = 8$ does not divide $N = 20$, there are $r = 4$ values in the last partition.*

$$
\begin{array}{rcl}
P_0 & = & [\ \underset{v_{0,0}}{23}\ \ \underset{v_{0,1}}{23}\ \ 30\ \ 52\ \ 55\ \ 55\ \ \underset{v_{0,6}}{70}\ \ \underset{v_{0,7}}{75}\ ] \\
P_1 & = & [\ \underset{v_{1,0}}{79}\ \ \underset{v_{1,1}}{80}\ \ 155\ \ 159\ \ 173\ \ 177\ \ \underset{v_{1,6}}{180}\ \ \underset{v_{1,7}}{5000}\ ] \\
P_2 & = & [\ \underset{v_{2,0}}{7000}\ \ \underset{v_{2,1}}{12000}\ \ \underset{v_{2,2}}{30000}\ \ \underset{v_{2,3}}{50000}\ ]
\end{array}
$$

Normalization

Since the range of the values in the different attributes could differ significantly among them, it is necessary to normalize the data to a certain predefined range of values (see P_1 in Example 3.3).

There are many ways to normalize a data set. A possible solution would be to normalize each attribute independently before the application of the vectorization step. However, this normalization method could present problems with skewed attributes and therefore the attributes could not be merged in the sorting step. For this reason, we propose to normalize the data stored in each partition separately. Thereby, similar values are assign to the same partition and therefore the chances of avoiding the effect of skewness in the data are higher.

Formally, we denote the normalized values as $\bar{v}_{m,n}$ and the normalized partitions as $\bar{P}_m$. Let $\max_m$ and $\min_m$ be the maximum and the minimum values in the mth partition:

$$\max_m := \max_{0 \le i < k} \{v_{m,i}\} \qquad \min_m := \min_{0 \le i < k} \{v_{m,i}\}$$

The normalized values are then defined as:

$$
\begin{cases}
\bar{v}_{m,n} := & \frac{v_{m,n} - \min_m}{\max_m - \min_m} & \text{if } \max_m \neq \min_m \\
\bar{v}_{m,n} := & 0.5 & \text{if } \max_m = \min_m
\end{cases}
$$

where $0 \le m < P$ (or $0 \le m \le P$ if κ does not divide N,) and $0 \le n < \kappa$. Note that

$\max_m = \min_m$ means that all the values in the partition are the same. In this case, the normalized value is centered in the normalization range.

Example 3.4 *Below we present the result of normalizing the partitioned data set of Example 3.3.*

$$\bar{P}_0 = [\underset{\bar{v}_{0,0}}{0}\ \underset{\bar{v}_{0,1}}{0}\ 0.13\ 0.56\ 0.62\ 0.62\ \underset{\bar{v}_{0,6}}{0.90}\ \underset{\bar{v}_{0,7}}{1}\ ]$$

$$\bar{P}_1 = [\underset{\bar{v}_{1,0}}{0}\ \underset{\bar{v}_{1,1}}{0.0002}\ 0.015\ 0.016\ 0.019\ 0.020\ \underset{\bar{v}_{1,6}}{0.020}\ \underset{\bar{v}_{1,7}}{1}\ ]$$

$$\bar{P}_2 = [\underset{\bar{v}_{2,0}}{0}\ \underset{\bar{v}_{2,1}}{0.12}\ \underset{\bar{v}_{2,2}}{0.53}\ \underset{\bar{v}_{2,3}}{1}\ ]$$

Re-sorting and Re-normalization

One of the goals of the sorting process, apart from reducing the SSE value, is to desemantize the data set, *i.e.*, to merge values from different attributes in order to break completely the semantic and therefore make the re-identification process more difficult. If the range of values of a certain attribute differs significantly from the others, it is likely that it is not merged in previous steps. For instance, in Table 3.8, values referring to the income are not likely to be merged with other attributes.

In order to illustrate this problem, let us recall the expression of the sorted vector of data (V_s) of Example 3.2:

$$V_s = [\underset{v_0}{23}\ \underset{v_1}{23}\ \underset{v_2}{30}\ 52\ 55\ 55\ 70\ 75\ 79\ 80\ 155\ 159\ 173\ 177\ 180\ \underline{5000}\ \underline{7000}\ \underline{\underset{v_{17}}{12000}}\ \underline{\underset{v_{18}}{30000}}\ \underline{\underset{v_{19}}{50000}}\ ]$$

Values referring to the income are underlined, and we can therefore verify that they are not merged with other attributes.

In order to appropriately mingle all attributes, once data has been sorted and normalized, we repeat these two steps (sorting and normalization). Since the range of values have been homogenized by normalization, attributes are conveniently mixed in the second sorting step and thus the data set is correctly preprocessed.

Mean Value Computation

Once data is preprocessed, for each partition $\bar{P}_m$, the mean value of its components is computed:

$$\mu_m = \sum_{n=0}^{\kappa-1} \frac{\bar{v}_{m,n}}{\kappa} \qquad m = 0 \dots P-1 \qquad\qquad \mu_P = \sum_{n=0}^{r-1} \frac{\bar{v}_{P,n}}{r}$$

where the latter expression is applied to the last partition if $r > 0$, *i.e.*, if κ does not divide the total number of values in the data set.

The protected value $\bar{p}_{m,n}$ for $\bar{v}_{m,n}$ is then:

$$\begin{cases} \bar{p}_{m,n} = \mu_m & n = 0 \dots k-1 \quad m = 0 \dots P-1 \\ \bar{p}_{P,n} = \mu_P & n = 0 \dots r-1 \end{cases}$$

Finally, Mic1D-κ denormalizes the data into the original range, according to the normalization and re-normalization steps in the previous block. Then, the protected values are placed in the protected data set in the same place occupied by the corresponding $v_{m,n}$ in the original data set. In this way, we are undoing the sorting and vectorization steps.

3.3.2 Experimental Results

We have tested Mic1D-κ and compared our results with those obtained by the projected microaggregation (PCP, Zscores and Sugeno) and MDAV microaggregation, using the EIA and Water Treatment data sets (both described in Section 2.6). As shown in Section 3.1, when protecting a data set using multivariate microaggregation, the way in which the data is split to form blocks is highly relevant with regard to the degree of privacy achieved (k' value). As in Section 3.1, we have reduced both data sets to have 9 attributes, which we detail in Tables 3.1 and 3.2.

As before, in both data sets, attributes $a1, a2$ and $a3$ are highly correlated as well as attributes $a4, a5$ and $a6$ and attributes $a7, a8$ and $a9$. On the contrary, attributes of different blocks are non-correlated. For our experiments, when protecting data, we assume attributes to be split into three blocks of three attributes each. Also, we consider two situations when protecting the data sets: blocking correlated attributes and

Correlated	1G	$(a1, a2, a3)$, $(a4, a5, a6)$, $(a7, a8, a9)$
	2G	$(a1, a2, a5)$, $(a1, a3, a7)$, $(a2, a3, a6)$, $(a1, a4, a5)$, $(a2, a4, a6)$ $(a5, a6, a9)$, $(a6, a7, a8)$, $(a1, a8, a9)$, $(a2, a7, a9)$
	3G	$(a1, a4, a7)$, $(a1, a5, a8)$, $(a1, a6, a9)$, $(a2, a4, a7)$, $(a2, a5, a8)$ $(a2, a6, a9)$, $(a3, a4, a7)$, $(a3, a5, a8)$, $(a3, a6, a9)$
Non-correlated	1G	$(a1, a4, a7)$, $(a2, a5, a8)$, $(a3, a6, a9)$
	2G	$(a1, a4, a5)$, $(a1, a3, a7)$, $(a4, a7, a8)$, $(a1, a2, a5)$, $(a2, a4, a8)$ $(a5, a8, a9)$, $(a3, a6, a8)$, $(a1, a6, a9)$, $(a3, a4, a9)$
	3G	$(a1, a2, a3)$, $(a1, a5, a6)$, $(a1, a8, a9)$, $(a2, a3, a4)$, $(a4, a5, a6)$ $(a4, a8, a9)$, $(a2, a3, a7)$, $(a5, a6, a7)$, $(a7, a8, a9)$

Table 3.9: Different groups of attributes known by the intruder.

thus non-correlated blocks, *i.e.*, $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$; or blocking non-correlated attributes but correlated blocks, *i.e.*, $(a1, a4, a7)$, $(a2, a5, a8)$ and $(a3, a6, a9)$. Testing these two cases we study the impact of the choice of attributes for the microaggregation groups, based on their correlations, as we have done before in Section 3.1.

For each data set and attribute selection method, we apply all microaggregation methods using different configurations (*i.e.* different values of k). The selection of these values aims at covering a wide range of SSE values and, thus, studying scenarios with different *information loss* values. Namely, we protect the data sets with parameter k = 5, 25, 50, 75, 100 for the EIA data set, and k = 5, 10, 15, 20, 25 for the Water Treatment data set.

For Mic1D-κ, we use κ = 5000, 5500, 6000, 6500, 7000 for the EIA data set and κ = 300, 500, 800, 850, 900 for the Water Treatment data set. Note that, since Mic1D-κ *desemantizes* the data set, there is no point in considering different situations related to the correlation of the attributes and, therefore, we protect the data set just once for each parametrization. In order to make a fair comparison, we have chosen the values of κ in Mic1D-κ to obtain similar SSE values to those obtained by MDAV after protecting the data sets.

In order to compare the disclosure risk of microaggregation methods, we have performed two different kinds of measures, the k' measure and the DLD PLD measures. For the k' measure, we consider that a possible intruder knows the values of three

	k/κ	SSE	k'		
			1G	2G	3G
Mic.MDAV-k	5	28.74	5.05	1.98	1.03
	25	145.36	25.21	7.06	1.27
	50	219.44	50.52	13.86	1.57
	75	313.31	75.78	20.97	1.92
	100	397.48	102.30	29.05	2.57
Mic.PCP-k	5	141.28	5.18	1.95	1.00
	25	203.81	25.21	6.57	1.04
	50	330.19	50.52	12.74	1.15
	75	469.74	75.78	19.15	1.30
	100	649.49	102.30	26.14	1.53
Mic.Zscores-k	5	50.65	5.11	1.92	1.01
	25	140.70	25.16	6.64	1.06
	50	184.18	50.52	13.05	1.28
	75	287.95	75.78	19.88	1.58
	100	378.51	102.30	27.46	2.02
Mic.Sugeno-k	5	99.32	5.14	1.96	1.00
	25	181.07	25.10	6.64	1.04
	50	249.00	50.52	12.83	1.16
	75	358.39	75.78	19.46	1.36
	100	502.32	102.30	26.76	1.62
Mic1D-κ	5000	36.49	3.04	4.57	2.68
	5500	472.69	4.48	6.79	3.90
	6000	135.21	5.37	8.18	4.52
	6500	556.24	7.89	11.69	6.56
	7000	56.84	8.96	13.51	7.08

Correlated attributes

	k/κ	SSE	k'		
			1G	2G	3G
Mic.MDAV-k	5	45.18	5.01	2.03	1.06
	25	212.44	25.10	7.26	1.28
	50	361.38	50.52	14.59	1.81
	75	468.30	75.78	22.06	2.30
	100	569.66	102.30	30.67	3.17
Mic.PCP-k	5	124.83	5.15	1.94	1.01
	25	251.99	25.10	6.63	1.06
	50	369.73	50.52	12.96	1.20
	75	482.05	75.78	19.56	1.40
	100	608.82	102.30	26.74	1.73
Mic.Zscores-k	5	111.97	5.07	1.93	1.01
	25	212.47	25.10	6.92	1.12
	50	336.68	50.52	14.07	1.52
	75	439.99	75.78	21.96	2.18
	100	553.89	102.30	30.90	3.16
Mic.Sugeno-k	5	139.69	5.12	1.96	1.01
	25	276.15	25.10	6.72	1.10
	50	427.92	50.52	13.16	1.29
	75	594.42	75.78	20.00	1.59
	100	796.38	102.30	27.66	2.04
Mic1D-κ	5000	36.49	3.04	4.57	2.68
	5500	472.69	4.48	6.79	3.90
	6000	135.21	5.37	8.18	4.52
	6500	556.24	7.89	11.70	6.56
	7000	56.84	8.96	13.51	7.08

Non-correlated attributes

Table 3.10: SSE and real k' of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

	k/κ	SSE	k'		
			1G	2G	3G
Mic.MDAV-k	5	28.18	5.09	1.94	1.00
	10	46.14	10.00	3.14	1.01
	15	72.03	15.20	4.42	1.01
	20	94.24	20.00	5.75	1.04
	25	114.56	25.33	7.28	1.10
Mic.PCP-k	5	28.59	5.14	1.94	1.01
	10	49.61	10.00	3.10	1.00
	15	71.99	15.20	4.41	1.02
	20	91.96	20.00	5.70	1.03
	25	110.91	25.33	7.24	1.04
Mic.Zscores-k	5	23.78	5.14	1.94	1.01
	10	49.05	10.00	3.13	1.01
	15	72.23	15.20	4.43	1.03
	20	93.10	20.00	5.69	1.03
	25	111.69	25.33	7.23	1.07
Mic.Sugeno-k	5	26.59	5.07	2.03	1.01
	10	53.88	10.00	3.30	1.01
	15	84.86	15.20	4.65	1.03
	20	95.35	20.00	5.98	1.08
	25	121.94	25.33	7.51	1.11
Mic1D-κ	300	32.67	1.62	1.51	1.10
	500	65.89	3.25	3.39	1.76
	800	80.95	7.87	7.55	4.67
	850	132.13	9.65	10.03	6.65
	900	255.64	12.95	13.61	9.14
Correlated attributes					

	k/κ	SSE	k'		
			1G	2G	3G
Mic.MDAV-k	5	69.51	5.00	2.03	1.03
	10	126.21	10.00	3.55	1.16
	15	173.96	15.20	5.28	1.39
	20	247.58	20.00	7.00	1.53
	25	259.07	25.33	9.22	1.91
Mic.PCP-k	5	93.67	5.02	1.94	1.01
	10	133.83	10.00	3.14	1.02
	15	170.12	15.20	4.47	1.03
	20	206.74	20.00	5.75	1.06
	25	229.50	25.33	7.33	1.13
Mic.Zscores-k	5	73.52	5.02	1.97	1.02
	10	115.77	10.00	3.27	1.06
	15	160.30	15.20	4.75	1.10
	20	197.20	20.00	6.32	1.28
	25	231.81	25.33	8.21	1.46
Mic.Sugeno-k	5	102.76	5.09	2.07	1.03
	10	188.85	10.00	3.29	1.04
	15	216.44	15.20	4.66	1.07
	20	281.02	20.00	5.96	1.10
	25	294.89	25.33	7.53	1.15
Mic1D-κ	300	32.67	1.11	1.35	1.35
	500	65.89	2.78	2.58	2.63
	800	80.95	4.74	7.17	6.88
	850	132.13	6.54	9.77	8.67
	900	255.64	9.07	14.52	11.71
Non-correlated attributes					

Table 3.11: SSE and real k' of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

random attributes of the original data set. Different tests are performed assuming that the intruder knows different sets of three attributes. Depending on these attributes, by using multivariate microaggregation, the intruder will have information coming from one or more groups. Table 3.9 shows all the considered possibilities.

Firstly, we suppose that the three known attributes belong to the same microaggregated block (*e.g.* $(a1, a2, a3)$ in the correlated scenario or $(a1, a4, a7)$ in the non-correlated). Since the size of the three microagreggation blocks is 3, there are only three options to consider. We denote this case by 1G. Since the intruder only has access to data from one group, multivariate microaggregation ensures the k-anonymity property (this is the best possible scenario for multivariate microaggregation). However, note that, usually, the intruder cannot choose the attributes obtained from external sources and it might be difficult to obtain all the attributes for the same group. Secondly, we assume that the known attributes belong to two different microaggregated groups. There are many possible combinations of three attributes under this assumption, so nine of them were chosen randomly. We refer to this case as 2G. Finally, case 3G is defined analogously to 2G, and also nine possibilities of known attributes are considered. Note that, in both scenarios 2G and 3G, k-anonymity is not ensured by multivariate microaggregation. Note also that, if the intruder had more than three attributes, it would not be possible to consider 1G. We are considering the case were the intruder only has three attributes to study a scenario were multivariate microaggregation can still preserve k-anonymity.

The second column of Tables 3.10 and 3.11 presents the SSE values for all the parameterizations and situations described before. Note that the range of SSE covered by the two methods is similar, so this allows us to compare the disclosure risk of both methods fairly. For all these scenarios, we compute k' and the mean of all the k' values in each situation is presented in the third, fourth and fifth columns. Note that, whereas multivariate microaggregation is affected by the fact that the chosen attributes are correlated or not, this effect is not noticeable using Mic1D-κ. Specifically, when the attributes in a group are not correlated, the information loss (SSE) using multivariate microaggregation tends to be increased since we are trying to collapse the records in a single value, using three independent attributes or dimensions. Nevertheless, this effect can be neglected with Mic1D-κ since, thanks to the data preprocessing, the whole microaggregation process is performed on a single dimension (vector of values), the semantics of attributes are ignored and the effect caused by

attribute correlations is avoided.

These results show that, in this scenario, Mic1D-κ achieves lower disclosure risk levels (larger values of k') than those achieved by multivariate microaggregation for similar information loss (SSE), especially when the attributes chosen come from different microaggregated groups (2G and 3G), which is the most common case. When the intruder has access to the three attributes coming from a single microaggregated group, multivariate microaggregation configurations present k' values which are similar or, in some cases, even larger than those obtained by Mic1D-κ (comparing cases with similar SSE). This is normal since such methods preserve the k-anonymity in this case. However, in the remaining scenarios (2G and 3G), that represent most of the cases, Mic1D-κ achieves larger k' values than those obtained by multivariate microaggregation when similar SSE values are compared.

Table 3.12 and 3.13 show the score and its components. For DLD, PLD and ID computation we have considered different cases, according to the number of attributes of the original record(s), to be linked, that the intruder knows. This number varies from 1 to the total number of attributes of each data set. The values in the table are the average of the obtained correct links in all these cases, for each parameterization of each protection method. The first column of these tables presents the IL values for each configuration, the IL values are similar for all protection methods, except for the case of Zscores microaggregation using non-correlated attributes in the Water Treatment data set and Mic1D-κ in the EIA data set. These differences are due to the parameter selection, the selection has been done to ensure similar SSE values, and then in some configurations, it is possible to obtain very different IL values.

The second and the third columns of Table 3.12 and 3.13 show the DLD and PLD risk measures. As in Section 3.2, the lowest disclosure risk is achieved by Sugeno microaggregation, where the largest disclosure risk values have been obtained using P-RL and it is always lower than 15%. If we observe the disclosure risk obtained by Mic1D-κ in this scenario, we will see that, in some cases, the disclosure risk of Mic1D-κ is larger than multivariate microaggregation. This happens because we are averaging nine possible scenarios, in three of them the intruder knows attributes of only one group (the best situation for multivariate microaggregation), another three the intruder knows attributes of two groups and, finally, only in three situations the intruder knows attributes belonging to all groups. This averaging process favours

to multivariate microaggregation and makes the comparison between Mic1D-κ and multivariate microaggregation unfair. However, it is clear that in this kind of scenarios the disclosure risk of Mic1D-κ increases.

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	6.68	1.78	2.87	87.60	25.82
	25	11.83	0.78	0.68	79.55	25.99
	50	13.23	0.60	0.56	72.37	24.85
	75	15.56	0.53	0.55	72.16	25.96
	100	17.63	0.39	0.49	67.52	25.81
Mic.PCP-k	5	16.61	1.78	2.87	65.29	25.21
	25	18.33	0.78	0.68	61.62	24.75
	50	19.77	0.60	0.56	59.71	24.96
	75	21.16	0.53	0.55	59.67	25.63
	100	22.26	0.39	0.49	57.87	25.71
Mic.Zscores-k	5	12.58	6.17	8.36	75.09	26.88
	25	16.04	5.03	5.98	70.88	27.12
	50	16.84	4.92	5.60	69.36	27.08
	75	18.69	3.91	5.30	69.71	27.92
	100	18.86	3.48	4.63	67.78	27.39
Mic.Sugeno-k	5	14.83	2.56	5.22	67.86	25.35
	25	18.10	1.22	1.78	63.17	25.22
	50	18.28	1.01	1.23	62.17	24.96
	75	18.55	0.88	1.13	62.15	25.06
	100	18.88	0.63	0.75	60.49	24.74
Mic1D-κ	5000	60.25	0.00	0.00	67.44	46.99
	5500	88.10	0.00	0.00	70.38	61.65
	6000	93.13	0.00	0.00	74.19	65.11
	6500	104.57	0.00	0.00	61.47	67.65
	7000	133.38	0.00	0.00	59.97	81.68

Correlated attributes

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	10.05	1.61	2.43	83.30	26.36
	25	16.58	0.81	0.56	72.27	26.53
	50	21.86	0.62	0.49	67.16	27.86
	75	20.26	0.68	0.60	64.61	26.44
	100	25.14	0.60	0.48	61.29	28.03
Mic.PCP-k	5	19.37	0.62	0.54	57.35	24.17
	25	22.07	0.63	0.48	53.45	24.54
	50	22.25	0.64	0.48	52.41	24.37
	75	22.70	0.64	0.49	52.11	24.52
	100	23.07	0.66	0.50	50.93	24.41
Mic.Zscores-k	5	16.81	2.58	3.82	75.75	28.14
	25	17.06	1.92	2.55	75.21	27.89
	50	17.25	1.44	2.22	73.88	27.55
	75	17.83	1.25	2.14	73.71	27.77
	100	17.80	1.16	1.88	70.95	27.02
Mic.Sugeno-k	5	20.98	0.83	1.56	58.39	25.39
	25	25.25	0.33	0.57	53.77	26.18
	50	33.42	0.29	0.34	52.20	29.84
	75	43.41	0.23	0.33	51.65	34.69
	100	27.82	0.25	0.29	50.28	26.55
Mic1D-κ	5000	60.25	0.00	0.00	67.44	46.99
	5500	88.10	0.00	0.00	70.38	61.65
	6000	93.13	0.00	0.00	74.19	65.11
	6500	104.57	0.00	0.00	61.47	67.65
	7000	133.38	0.00	0.00	59.97	81.68

Non-correlated attributes

Table 3.12: Scores of different microaggregation methods and parameterizations using the EIA data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	14.14	73.03	67.24	72.73	42.79
	10	18.78	61.97	55.66	63.56	39.98
	15	17.34	49.74	43.95	56.99	34.63
	20	18.28	39.34	35.53	51.18	31.29
	25	21.68	32.37	29.08	48.59	30.67
Mic.PCP-k	5	18.36	40.39	30.39	60.82	33.23
	10	18.11	30.00	21.58	53.66	28.92
	15	21.67	23.82	20.39	50.54	29.00
	20	25.17	21.45	16.05	47.21	29.08
	25	23.25	19.08	13.68	49.34	28.06
Mic.Zscores-k	5	17.62	76.05	62.50	68.65	43.29
	10	20.62	63.82	54.87	61.53	40.53
	15	20.99	54.08	47.76	56.42	37.33
	20	20.74	47.76	40.79	53.48	34.81
	25	24.30	43.95	34.47	54.04	35.46
Mic.Sugeno-k	5	15.18	1.67	12.89	64.17	25.45
	10	24.56	1.23	10.35	56.49	27.85
	15	25.69	0.70	8.98	52.53	27.19
	20	32.30	0.88	8.71	48.14	29.38
	25	28.18	0.96	6.84	47.44	26.93
Mic1D-κ	300	29.63	62.63	67.13	81.54	51.42
	500	46.68	42.87	49.36	56.39	48.97
	800	82.99	24.44	12.46	39.36	55.95
	850	85.50	14.85	2.28	50.61	57.54
	900	87.70	9.80	1.93	43.72	56.25

Correlated attributes

	k	IL	DLD	PLD	ID	Score
Mic.MDAV-k	5	31.75	8.16	39.87	45.79	33.33
	10	28.28	5.26	28.95	43.00	29.17
	15	35.60	2.50	18.82	41.89	30.94
	20	32.44	2.63	14.21	39.34	28.16
	25	36.74	1.71	12.89	30.85	27.91
Mic.PCP-k	5	50.41	8.95	2.11	36.63	35.75
	10	53.51	5.00	0.79	30.96	35.22
	15	56.28	4.21	1.32	30.37	36.42
	20	61.02	4.74	1.05	26.30	37.81
	25	62.48	3.82	0.26	25.61	38.15
Mic.Zscores-k	5	98.33	10.13	3.16	42.96	61.57
	10	108.75	6.05	2.24	40.61	65.56
	15	113.74	5.39	1.71	40.18	67.80
	20	114.78	3.03	1.45	39.98	67.95
	25	113.71	3.29	1.05	37.60	66.80
Mic.Sugeno-k	5	36.15	4.97	0.82	44.25	29.86
	10	41.55	3.22	0.12	36.53	30.32
	15	48.41	1.73	0.23	31.45	32.31
	20	47.45	1.75	0.94	30.26	31.63
	25	51.53	1.11	0.53	28.62	33.12
Mic1D-κ	300	29.63	62.63	67.13	81.54	51.42
	500	46.68	42.87	49.36	56.39	48.97
	800	82.99	24.44	12.46	39.36	55.95
	850	85.50	14.85	2.28	50.61	57.54
	900	87.70	9.80	1.93	43.72	56.25

Non-correlated attributes

Table 3.13: Scores of different microaggregation methods and parameterizations using the Water Treatment data set. Mic.Method-k corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k.

Chapter 4

Specific Disclosure Risk Measures

As we stated in Chapter 2, only generic measures, as distance based or probabilistic record linkage, are considered when the disclosure risk of a protection method is computed. The use of these generic measures causes an underestimation of the resulting disclosure risk.

In this chapter we define specific record linkage methods which take into account the protection method applied to the protected data set. The direct consequence of these definitions is that we achieve a larger number of re-identifications than with generic record linkage methods. Therefore, under this scenario, disclosure risk is larger than believed up to now.

4.1 Rank Swapping Record Linkage

In this section, we describe a new record linkage method, specific for rank swapping. We call this method *rank swapping record linkage* (RS-RL for short). This method takes advantage of the fact that only a few values of the data set are eligible when doing rank swapping. By using this information, we can limit the pairs of records where record linkage method is applied and decrease in this way the probability of

finding incorrect links. The result causes an increase on the number of correct links, and therefore an increment in the disclosure risk of standard rank swapping.

Then, we propose two new protection methods, obtained as variants of rank swapping, which are called *rank swapping p-distribution* and *rank swapping p-buckets* and which are in some way immune to the effect of the new record linkage method. The main idea of these methods is that each attribute value can be potentially (maybe with very low probability, but never equal to 0) swapped with any other value. In this way, an intruder linking records will not be able to limit the swap interval with total confidence; this will lead to a higher number of incorrect linkages.

4.1.1 Algorithm Description

The idea is quite easy to understand: standard rank swapping swaps one original value with one of the p following values in the sorted table (recall the rank swapping description presented in Section 2.4.1). Therefore, if the protected values of the attribute are known, as in the scenario described in Section 2.4, it is possible to restrict the protected records into which a specific original record can have been mapped. Formally, the intruder must compare the original record x_i that he wants to link with only $2p$ records in the protected microdata file (note that a protected value can be either the source or the destination in the swap process). In other words, for every original attribute value x_{ij}, there is an efficiently computable set $B(x_{ij})$ of $2p$ protected records which may be the result of transforming the original record x_i.

Obviously, if more than one attribute is known, it is possible to repeat the process for each attribute. In particular, if the original record x_i is represented by $x_i = (x_{i1}, \ldots, x_{ic})$ for c attributes $attr_1, \ldots, attr_c$, then the matching protected record x'_ℓ will necessarily satisfy the condition

$$x'_\ell \in \cap_{1 \leq j \leq c} B(attr_i = x_{ij})$$

That is, the search of the linkage is reduced to the intersection of the sets of possible protected records. Of course, the more attributes are considered, the less records will be in this intersection, and therefore the probability of finding the correct record linkage will increase. This effect is illustrated in Figure 4.1. In particular, if some intersection (for some combination of the protected attributes) gives a unique pos-

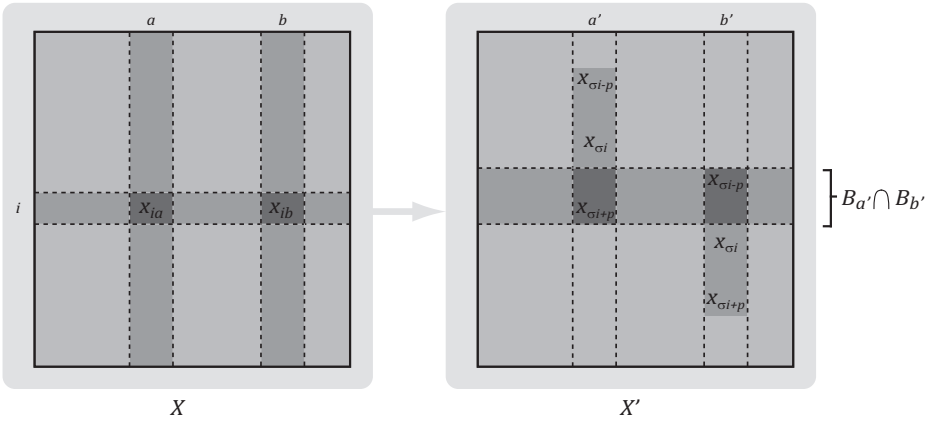


Figure 4.1: Graphic representation of disclosure risk.

sible record, we can be sure that this is the protected record which matches with the considered original record, because this linkage method does not introduce error probabilities. This is in contrast to the standard record linkage methods, where an original record is compared to all the protected records, possibly leading to incorrect linkages.

Example 4.1 *Let us illustrate this fact throughout the example described in Table 2.1 and reproduced here in Table 4.1 (this table is also used in Example 2.1 in Section 2.4.1). Consider the standard distance based record linkage method (with the Euclidean distance) applied to the original record* $(6,7,10,2)$. *When, the distances between this record and all the protected records are computed, the closest protected record results to be* $(6,7,6,3)$, *which is not the matching one. Therefore, this method leads to the incorrect linkage* $(6,7,10,2) \leftrightarrow (6,7,6,3)$. *In contrast, consider the new specific technique applied to the same record* $(6,7,10,2)$. *The set of possible protected values consistent with a 6 in the first original attribute is* $B(attr_1 = 6) = \{(4,1,10,10),(5,5,8,1),(6,7,6,3),(7,3,5,6),(8,4,2,2)\}$. *Analogously for the other three attributes, we obtain* $B(attr_2 = 7) = \{(5,5,8,1),(2,6,9,8),(6,7,6,3),(1,8,7,9),(3,9,1,7)\}$, $B(attr_3 = 10) = \{(5,5,8,1),(2,6,9,8),(4,1,10,10)\}$ *and* $B(attr_4 = 2) = \{(5,5,8,1),(8,4,2,2),(6,7,6,3),(9,2,4,4)\}$. *Therefore, as the intersection of the four sets is just the protected record (5,5,8,1), in this case we obtain the correct linkage. Note that if there had been more than one record in the intersection, the closest one to the considered original record would have been chosen.*

Note also that this new method has been defined assuming that the value of the parameter p is known. In situations where this value is kept secret by the owner of the original data set, then the method can be applied by first fixing an upper bound for the value of p (for example, the 20% of the number of entries of the database). Of course, the results of the method are optimal when the exact value of p is used. This assumption is quite realistic, for example, all available microdata files in the EUROSTAT web page [29] include a full description of the anonymization criteria that have been applied.

Original Data Set X				Protected Data Set X'			
$attr_1$	$attr_2$	$attr_3$	$attr_4$	$attr'_1$	$attr'_2$	$attr'_3$	$attr'_4$
8	9	1	3	10	10	3	5
6	7	10	2	5	5	8	1
10	3	4	1	8	4	2	2
7	1	2	6	9	2	4	4
9	4	6	4	7	3	5	6
2	2	8	8	4	1	10	10
1	10	3	9	3	9	1	7
4	8	7	10	2	6	9	8
5	5	5	5	6	7	6	3
3	6	9	7	1	8	7	9

Table 4.1: Rank swapping example.

In Table 4.2, we show the number of correctly linked records for the data in the Example 4.1 with the three considered record linkage methods, when different collections of attributes are assumed to be known by the intruder. We also show the average disclosure risk. Note that the record linkage process uses only the known attributes to

	$attr_1$	$attr_{1-2}$	$attr_{1-3}$	$attr_{1-4}$	average DR
RS-RL	0	2	7	8	42.5
DB-RL	0	2	4	1	17.5
P-RL	0	0	0	0	0.00

Table 4.2: Correct links and average disclosure risk for Example 2.1 on record linkage of Section 2.4.1, computed with rank swapping (RS-RL), distance based (DB-RL) and probabilistic (P-RL) record linkage.

compute the nearest record. From the comparison between the average disclosure risk using the new method and the DB-RL and P-RL showed in Table 4.2, it is clear that the real disclosure risk is much larger than the standard estimated using DB-RL or P-RL.

4.1.2 New Rank Swapping Methods

In this section, we present two variants of rank swapping which resist the record linkage method introduced in Section 4.1.1. The main idea in both variants is the same: to eliminate the fact that the swap interval is closed.

Rank Swapping p-Distribution

As we have explained in Section 2.4.1 the swap interval in the rank swapping is defined by the parameter p, this fact is exploited by the record linkage technique defined in Section 4.1.1 to increase the number of linked records. For this reason, rank swapping p-distribution defines the swap interval using a normal probability distribution defined by $\mu = \sigma = 0.5 \cdot p$. This modification makes possible that any value in the data set can be selected. Obviously very different values have lower probability to be elected than similar values, but never equals to zero. Therefore some values x_{ij} are swapped with values out of the standard interval $\ell \in [i+1, i+p]$; we can observe this in Figure 4.2 where the swap interval is defined by a normal probability distribution, in this case it is clear that the swap interval is an open interval. When this effect is propagated to all protected attributes, the RS-RL method becomes unsuitable. In the experiments presented in the next section we will show that the number of correct links obtained by an intruder decreases when more attributes are known.

Rank swapping p-distribution applied to an attribute $attr_j$ of an original microdata file X can be defined as follows: firstly, the table (microdata file) is sorted in increasing order of the values x_{ij} of the considered attribute $attr_j$. For simplicity, we assume that the records are already sorted, that is $x_{ij} \leq x_{\ell j}$ for all $1 \leq i < \ell \leq n$. Then, for each value x_{ij}, a random value r is computed by using the $N(0.5 \cdot p, 0.5 \cdot p)$ normal distribution, and the values x_{ij} and $x_{\ell j}$ are swapped, where $\ell = i + r$. Finally, the sorting step is undone.

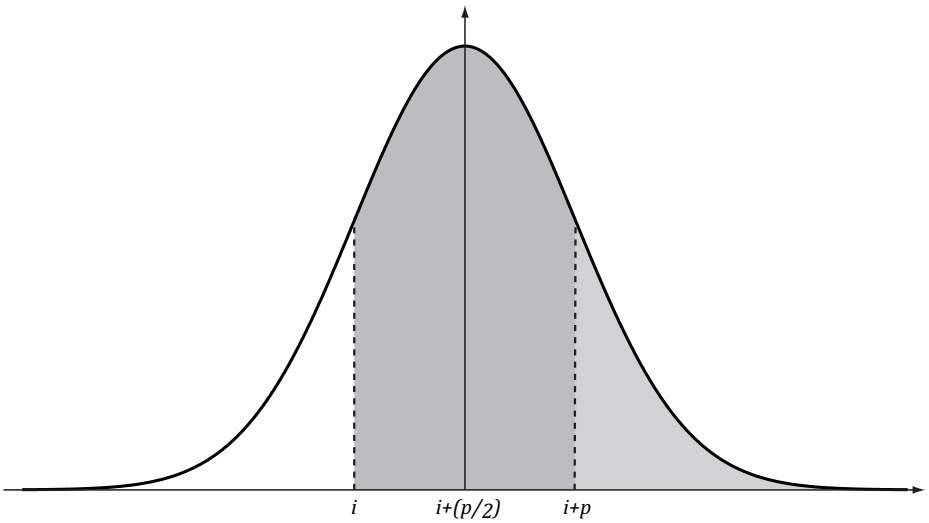


Figure 4.2: Graphic representation of the p-distribution swap interval.

The negative effect produced by swapping two different values is that the information loss of the protected files increases. We will show in Section 4.1.3 that the increment of information loss is compensated by a reduction of the disclosure risk, and therefore the scores obtained by this method are lower than the ones obtained by the standard rank swapping.

In Section 4.1.4 we present a possible specific record linkage for this new rank swapping p-distribution method. The experiments show that the performance of this specific record linkage is similar to distance based record linkage.

In the rest of this thesis, we will use *rs* α*-d* to denote the application of rank swapping p-distribution with $p = \alpha$.

Rank Swapping p-Buckets

The rank swapping p-buckets method pursues the same goal as rank swapping p-distribution. For this reason it also replaces the close swap interval of the rank swapping with an unlimited interval, but using now a different technique.

The idea of this method is to split the sorted original values of an attribute into

several buckets. Firstly, a probability function is used to choose a bucket for each value. Once the bucket is selected, the method works identically to the standard rank swapping, by using this bucket as the closed swap interval. Again, every original record will have some non-zero probability of being the correct link of every protected record. For this reason, the RS-RL method will be less effective here than when it is applied to the standard rank swapping.

Rank swapping p-bucket applied to an attribute $attr_j$ of an original microdata file X can be defined as follows: firstly, the table is sorted in increasing order of the values x_{ij} of the considered attribute $attr_j$. For simplicity, we assume that the records are already sorted, that is $x_{ij} \leq x_{\ell j}$ for all $1 \leq i < \ell \leq n$. The sorted values $\{x_{ij} | 1 \leq i \leq n\}$ are split into p buckets $B_1, \ldots, B_p$. For each value x_{ij}, which belongs to some bucket B_r, a bucket B_s is chosen, according to the probability distribution

$$\Pr[B_s \text{ is chosen}] = \frac{1}{2^{s-r+1}}.$$

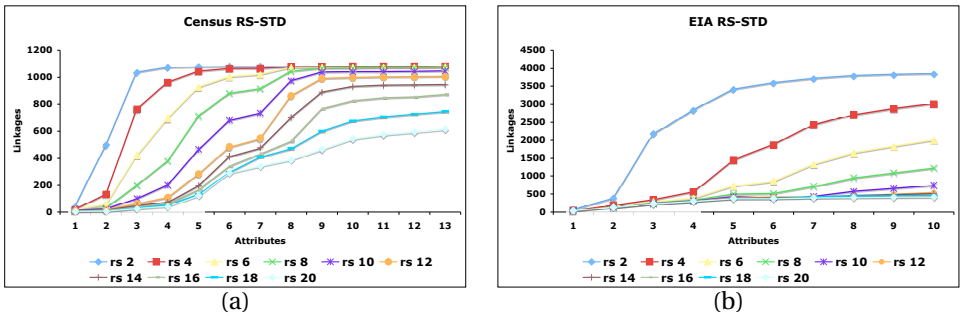
Then, a value $x_{\ell j}$ is randomly and uniformly chosen in the selected bucket B_s, and the values x_{ij} and $x_{\ell j}$ are swapped. Note that, if the same bucket $B_s = B_r$ is chosen, the condition $\ell > i$ must be imposed.

Note that, closer buckets to the original value are selected with higher probability than the far-off buckets and, therefore, the information loss of the protected microdata file is under control. Note that many other probability distributions could be used to define similar variants of rank swapping.

In the rest of the thesis, *rs α-b* will be used to denote a rank swapping p-bucket with $p = \alpha$.

4.1.3 Experimental RS-RL Results

As stated above, we have introduced the two variants of rank swapping to mitigate the effect of the specific record linkage method RS-RL. Of course, this is at the cost of increasing information loss, because some values may be swapped with values which are quite far. Our feeling was that this increment of the information loss had to be less significant than the saving in disclosure risk and, therefore, the new methods would obtain better general scores than standard rank swapping. The results



(a) (b)

Figure 4.3: Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with standard rank swapping.

described in this section confirm our feelings.

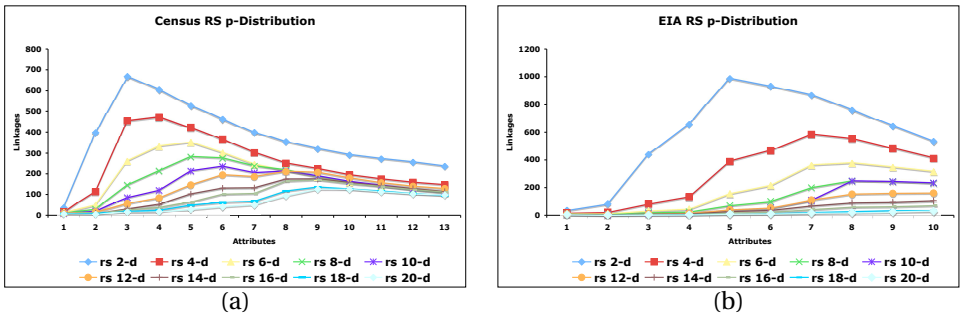
In short, this section describes the analysis of disclosure risk using our specific record linkage method and the comparison of the new rank swapping methods against standard ones. We start reviewing the data sets used in the experiments.

Disclosure Risk Analysis for Rank Swapping

In order to evaluate the specific record linkage method (RS-RL) introduced in Section 4.1.1, we have protected the Census and EIA data sets (defined in Section 2.6) by using different parameterizations of standard rank swapping ($p = 2 \ldots 20$), rank swapping p-distribution ($p = 2 \ldots 20$) and rank swapping p-buckets ($p = 75$, 50, 35, 30, 25, 20, 15,10, 5). For each protected data set, we have computed its disclosure risk using RS-RL. At this point, in order to study the worst case scenario, the parameter p is assumed to be known.

It is clear that the rank swapping p-distribution and the rank swapping p-buckets have an advantage with respect to standard rank swapping when RS-RL is used. For this reason, disclosure risk measures have been computed using a larger parameter. In particular, we used $2 \cdot p$ for the rank swapping p-distribution, and $p = 2 \cdot Bucket\ Size$ for the rank swapping p-buckets. For standard rank swapping, the parameter p was used to protect the data set.

Figure 4.3 shows, in a graphic way, the number of correct links that RS-RL obtains,



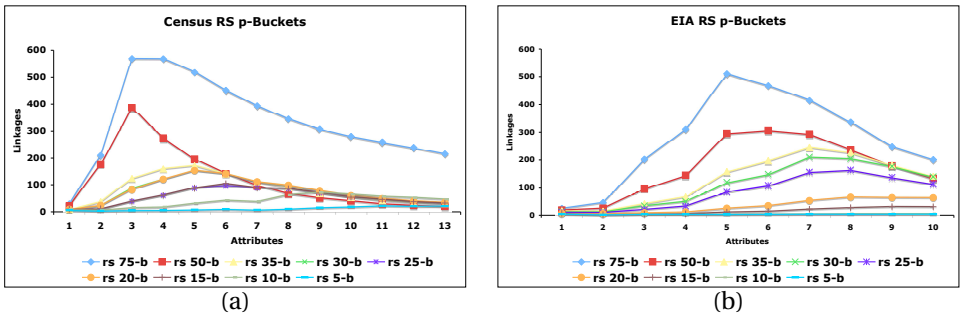
(a) (b)

Figure 4.4: Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with rank swapping p-distribution.

when applied to standard rank swapping, for both data sets, when different numbers of attributes are assumed to be known by the intruder (from one to all). It is easy to observe that the more attributes are known by the intruder, the more records are linked. Figures also show that for the five less protected data sets from Census, an intruder links more than 70% of the records when only half of the attributes are known. Another interesting result with the Census data set is that the intruder is always able to link more than 50% of the records if he knows all the attributes.

Similar results are obtained for the EIA data set. For the three less protected data sets, the intruder is able to link more than 50% of records when all the attributes are known. In EIA, the results of our specific record linkage are not so good because (i) the EIA data set has four times more records than Census data set, (ii) the EIA data set has less attributes than Census.

Figure 4.4 presents the results of RS-RL applied to the data sets protected using rank swapping p-distribution. The chart lines show that an intruder can make no use of knowing all the attributes when data is protected using rank swapping p-distribution. Therefore, the rank swapping record linkage becomes unsuitable now. This is so because when the intruder knows many attributes, he is forced to consider all the records, and not a small subset, to take advantage of his knowledge. For the Census data set, only in the less protected parameterization ($p = 2$) the intruder is able to link more than 50% of the records, and this happens when he knows three or four attributes: the knowledge of more attributes is not useful in this case. The rest of the cases are protected enough to avoid a large number of correct linkages for both



(a) (b)

Figure 4.5: Graphic representation of the results obtained by the rank swapping disclosure risk measure applied to the Census data set (a) and EIA data set (b), protected with rank swapping p-buckets.

the Census and EIA data sets.

The results obtained for rank swapping p-buckets, which are shown in Figure 4.5, are similar to the results of rank swapping p-distribution. Only in the less protected case of the Census data set the intruder is able to link more than 50% of the records, the remainder configurations are protected enough to avoid a large number of correct linkages. As it happens with the rank swapping p-distribution, the knowledge of all attributes is not useful for an intruder when using rank swapping record linkage.

New Rank Swapping Methods vs Standard Rank Swapping

In this section, we compare the overall behavior of the new methods with the standard one. We study the effects of the introduced modifications in information loss and standard disclosure risk measures.

The comparison is based on the score defined in Section 2.5. We have modified the score so that the disclosure risk measure takes into account the new rank swapping record linkage. Formally, we use the following disclosure risk measure $DR = 0.166 \cdot RSLD + 0.166 \cdot DLD + 0.166 \cdot PLD + 0.5 \cdot ID$ instead of $DR = 0.25 \cdot DLD + 0.25 \cdot PLD + 0.5 \cdot ID$. Here, $RSLD$ stands for Rank Swapping Linkage Disclosure, the average percentage of correctly linked records using rank swapping record linkage (RS-RL). Information loss measures are not changed and thus they are computed using the standard measures presented in Section 2.5.

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p	2	3.89	77.73	73.52	71.28	93.98	42.63
	4	6.54	66.65	58.40	42.92	83.09	36.67
	6	10.57	54.65	43.76	22.49	72.12	31.93
	8	16.54	41.28	32.13	11.74	62.11	29.16
	10	20.18	29.21	23.64	6.03	53.28	26.31
	12	23.46	19.87	18.96	3.46	47.17	24.77
	14	28.93	16.14	15.63	2.06	43.39	25.86
	16	35.16	13.81	13.59	1.29	40.78	27.97
	18	32.52	12.21	11.50	0.83	38.90	25.81
	20	35.12	10.88	10.87	0.59	37.33	26.55

Census data set

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p	2	4.24	43.27	21.71	16.85	93.10	28.06
	4	9.67	12.54	10.61	4.79	82.09	21.89
	6	14.63	7.69	7.40	2.03	72.21	21.42
	8	18.71	6.12	5.98	1.12	63.90	21.61
	10	22.87	5.60	5.19	0.69	57.09	22.37
	12	26.60	5.39	4.87	0.51	51.64	23.25
	14	29.42	5.28	4.55	0.32	47.49	23.91
	16	32.38	5.19	4.54	0.23	44.19	24.82
	18	34.22	5.20	4.54	0.22	41.42	25.28
	20	36.27	5.15	4.36	0.18	38.97	25.87

EIA data set

Table 4.3: Score calculation for standard rank swapping(rs-p). IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure.

Table 4.3 presents the scores as well as the original values before their aggregation, for both Census and EIA data sets protected using standard rank swapping. We can observe that the largest disclosure risk measure in all cases is $RSLD$. Therefore, it is clear that the new method increases the risk with respect to standard ones for the standard rank swapping. Another interesting result is that PLD is always lower than $RSLD$ and DLD. This is because all the values in all the attributes are swapped, and therefore the coincidence vectors for the correct links are always equal to zero, unless the swapped positions have the same value (which is possible only if the attributes have repeated values). It is possible to observe in Figures 4.6.(b) and 4.7.(b) that the standard rank swapping has the largest disclosure risk when the different parameterizations of the three methods are compared. This effect is much clearer in Figure 4.6.(b) (corresponding to the Census data set) than in Figure 4.7.(b) (corresponding to the EIA data set).

Tables 4.4 and 4.5 show the score values for rank swapping p-distribution and rank swapping p-buckets respectively. In both cases the largest disclosure risk measure is DLD. Therefore, when an intruder is interested in linking the original data set with the one protected using any of the two methods, the best way is just to consider all possible links.

In Section 4.1.4 we discuss a possible record linkage method specifically designed

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p-distr.	2	3.80	40.90	71.63	1.38	89.81	29.70
	4	7.42	28.34	53.10	0.68	71.48	24.83
	6	14.44	20.51	37.57	0.62	55.58	23.08
	8	17.31	15.74	25.98	0.54	44.44	20.90
	10	22.49	11.69	19.42	0.46	38.60	21.28
	12	31.04	9.10	16.17	0.41	35.05	24.26
	14	31.80	6.12	13.16	0.38	32.19	23.41
	16	33.53	4.63	11.77	0.32	30.42	23.61
	18	37.89	3.10	11.12	0.35	28.98	25.25
	20	43.92	2.15	9.59	0.31	27.05	27.65

Census data set

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p-distr.	2	5.24	10.74	18.78	13.94	90.76	22.26
	4	11.33	3.13	9.12	3.90	78.07	20.28
	6	16.83	1.27	6.41	1.61	67.20	20.50
	8	18.71	6.12	5.98	1.12	63.90	21.61
	10	26.11	0.39	4.81	0.55	52.05	22.24
	12	29.89	0.39	4.62	0.42	47.11	23.28
	14	32.01	0.30	4.51	0.29	43.30	23.67
	16	35.59	0.20	4.60	0.22	39.95	24.89
	18	37.69	0.10	4.69	0.18	37.52	25.52
	20	40.12	0.10	4.57	0.11	35.25	26.34

EIA data set

Table 4.4: Score calculation for rank swapping p-distribution (rs p-d). IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure.

for rank swapping p-distribution and p-buckets. Since the obtained results are essentially the same as with distance based record linkage we have not considered this method to compute the disclosure risk and the score.

Interval disclosure for standard rank swapping is higher than the one for the two new rank swapping methods. This is so because the rank swapping p-distribution and the p-buckets may do swaps between two far values avoiding the interval disclosure. This is not the case in standard rank swapping.

In general, when we compare the same parameterizations for the three different rank swapping methods (*e.g.* rs 2, rs 2-d and rs 75-b), information loss is higher for the rank swapping p-distribution and the rank swapping p-buckets. Nevertheless, disclosure risk is higher for standard rank swapping (in some cases more than 15%) for these parameterizations. In Figures 4.6.(a) and 4.7.(a) we can observe that some cases of standard rank swapping have lower information loss than the other rank swapping versions. However, these differences are rather small in most of the cases. See, for example, the IL in the EIA data set for the fourth parameterization of the three rank swapping methods (rs 8, rs 8-d and rs 30-b). It is clear that the values for IL are rather similar.

In relation to the overall scores, the best scores obtained for the standard rank swapping (see Table 4.3) are 24.77 for Census and 21.42 for EIA. In contrast to that, the

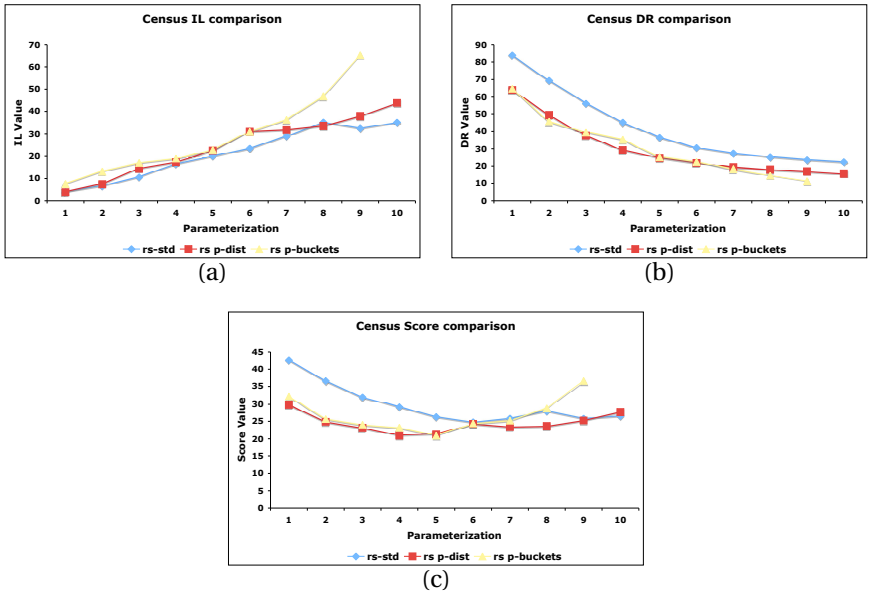


Figure 4.6: Graphic representation of the information loss (a), disclosure risk (b) and score (c) values for the three rank swapping methods when Census data set is protected.

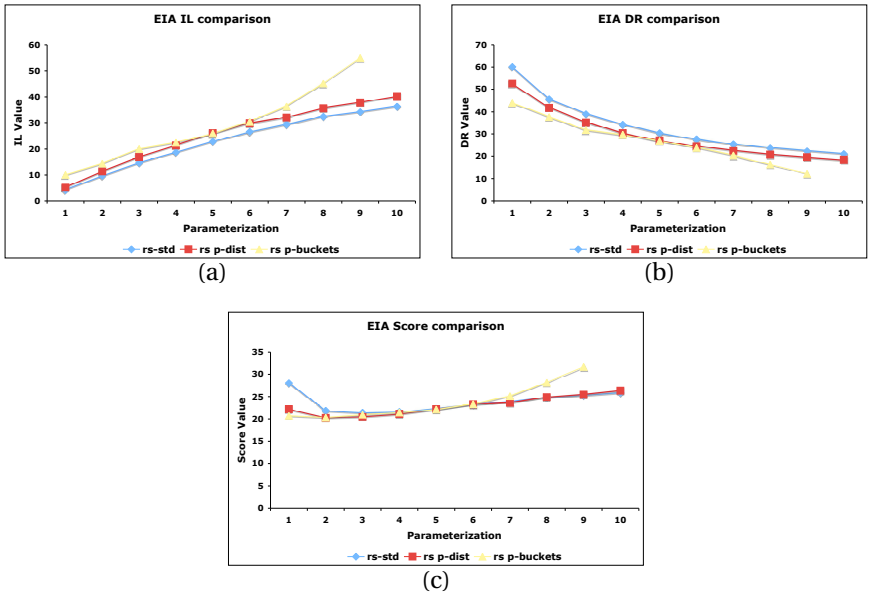


Figure 4.7: Graphic representation of the information loss (a), disclosure risk (b) and score (c) values for the three rank swapping methods when EIA data set is protected.

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p-buckets	75	7.51	36.24	58.08	44.50	83.02	32.18
	50	13.14	17.09	42.71	27.21	62.60	25.68
	35	17.03	9.96	29.93	13.36	61.29	24.00
	30	18.96	8.56	24.86	9.81	56.37	23.19
	25	22.79	5.26	20.18	0.40	41.60	20.92
	20	30.97	8.51	15.86	0.42	36.61	24.36
	15	36.29	5.36	11.59	0.44	30.82	25.18
	10	46.91	2.19	8.53	0.40	25.40	28.80
	5	65.29	0.63	6.54	0.31	20.18	36.68

Census data set

	p	IL	RSLD	DLD	PLD	ID	Score
rank swapping p-buckets	75	10.03	5.35	10.90	6.50	80.37	20.75
	50	14.35	2.83	7.53	3.17	70.70	20.32
	35	20.03	1.46	5.53	1.56	60.80	20.98
	30	22.49	1.11	5.42	1.16	56.92	21.47
	25	25.98	0.78	4.90	0.79	51.61	22.19
	20	30.46	0.28	4.65	0.54	45.88	23.35
	15	36.32	0.14	4.82	0.33	39.12	25.13
	10	45.17	0.09	4.74	0.17	31.04	28.18
	5	54.96	0.08	4.55	0.08	22.87	31.69

EIA data set

Table 4.5: Score calculation for rank swapping p-buckets (rs p-b).IL stands for Information Loss, RSLD stands for Rank Swapping Linkage Disclosure, DLD stands for Distance Linkage Disclosure, PLD stands for Probability Linkage Disclosure and ID stands for Interval Disclosure.

best scores obtained for the rank swapping p-distribution (see Table 4.4) are between 20.90 and 21.30 for the Census data set and between 20.20 and 21.00 for the EIA data set. The best score presented in Table 4.5 for the rank swapping p-buckets is 20.92 for the Census data set and 20.32 for the EIA data set. So, the new rank swapping methods lead to better scores and, thus, the trade-off between information loss and disclosure risk benefits the rank swapping methods introduced in this section. Even though the new methods have a small increment in the information loss, they gain an important reduction in disclosure risk. This effect is illustrated in Figures 4.6.(c) and 4.7.(c) where in most of the cases the standard rank swapping has the largest score.

4.1.4 Specific Record Linkage Methods for the New Variations of Rank Swapping

In Section 4.1.1 we have shown that standard rank swapping can be attacked using a specific record linkage method. These results have motivated the introduction of two variants of rank swapping, variations that cannot be attacked using the specific methods. Nevertheless, it is worth considering at this point whether other ad-hoc attacks might be developed for these new methods. In this section we present another specific record linkage method specially designed for rank swapping p-distribution

and p-buckets. We show that this new method, when applied to the same data sets used in Section 4.1.3 leads to similar results to the ones obtained with standard distance based record linkage method.

The difficulty of applying record linkage to data protected by rank swapping p-distribution and p-bucket is that the intruder cannot limit the swap interval. In other words, it is always possible that the correct linkage is not included in the intersection set $B(x_i) = \cap_{1 \leq j \leq c} B(x_{ij})$. Therefore, the use of $B(x_i)$ is sometimes useless.

To avoid this, but reducing at the same time the complexity of record linkage, we can consider proper subsets $B(x_{ij})$, and then compute their union. Of course, with a large probability, the correct link will be in $B(x_{ij})$ for at least one of the attributes j. In fact, the record will belong to most of the $B(x_{ij})$. Because of this, the record will be also in the union of the $B(x_{ij})$. That is, in $B(x_i) = \cup_{1 \leq j \leq c} B(x_{ij})$.

We will compute an *annotated* union that takes into account the number of sets $B(x_{ij})$ where each record is stored. Then, we assume that the intruder only compares the original record with the protected records which have the maximum cardinality in the annotated union set (*i.e.* records stored in the maximum number of $B(x_{ij})$ sets). In this approach the intruder is minimizing the far swaps in the protected attributes. So, the intruder is exploiting his knowledge of the rank swapping p-distribution and p-buckets because he knows that the probability of a far swap is near to 0.

As we did in Section 4.1.3, we suppose here that the parameter p is known. We have applied this variation of the rank swapping record linkage method to the rank swapping p-distribution. In order to apply the method in the worse case scenario, we also suppose that the intruder has the maximum number of attributes, *i.e.* the intruder knows the half of the protected attributes (this assumption is the same as in Section 2.5). At this point, the intruder needs to decide the size of the intervals $B(x_{ij})$. From the cumulative normal distribution, the intruder can deduce the confidence of the selected swap intervals. For example, if the swap intervals are $[i - 0.5p, i + 0.5p]$ the probability that the correct linkage is inside one of these intervals is equal to 68%. Otherwise, if the swap intervals are $[i - p, i + p]$; then, the same probability increases to 96%. Note that when the union is computed the probability to find the correct link inside increases with respect to the number of attributes known.

	p	DB-RL	RS-RL $[i-\frac{p}{2}, i+\frac{p}{2}]$	RS-RL $[x_i-p, x_i+p]$
rank swapping p-distr.	2	1048.2	752.8	1054.4
	4	938.9	461	907.4
	6	723.6	285.3	686.6
	8	502.5	190	487.1
	10	335.1	136.2	347.2
	12	219.5	96.4	241.7
	14	145.2	70.4	161.7
	16	95.7	48.9	109.7
	18	69.6	42.5	76.5
	20	47.8	33.7	54.3

Census data set

	p	DB-RL	RS-RL $[i-\frac{p}{2}, i+\frac{p}{2}]$	RS-RL $[x_i-p, x_i+p]$
rank swapping p-distr.	2	1267.2	771.5	1443.8
	4	330.1	256	419.2
	6	130.4	118.4	159.4
	8	60.4	58.9	72.9
	10	36	37.2	41.1
	12	22.4	23.9	28.4
	14	16.8	17	19.9
	16	11.5	11.3	13.1
	18	8.9	11.2	9.9
	20	7	8.3	9.4

EIA data set

Table 4.6: Average linkage values for rank swapping p-distribution.DB-RL stands for Distance Based Record Linkage, RS-RL stands for Rank Swapping Record Linkage.

Table 4.6 shows the results for the distance based record linkage and for the rank swapping record linkage with two different swap intervals. The table also includes the results obtained by the DB-RL for the same files and number of attributes.

The results show that, in general, when we use the interval $[i-p, i+p]$ the new method lead to results similar to the ones by DB-RL. There are only two cases were we can find a significant improvement, they correspond to the application of the method when $p = 2$ and $p = 4$ and using the EIA data set. RS-RL finds 1443 records while DB-RL finds 1267, and RS-RL finds 419 where DB-RL finds 330. Nevertheless, these cases are not a real threat to the protection method because they already correspond to the cases with a lower protection and high risk (note that in the case of the Census database, that is a smaller data set, almost all records are re-identified by DB-RL). Taking this into account, we have that the influence of the results shown here in the score would not be significant.

The table also shows the results for the interval $[i-p/2, i+p/2]$. This smaller interval reduces more the search than the other one. However, the results obtained by the new record linkage are much worse than the ones obtained by DB-RL.

Summing up, we have that the ad-hoc record linkage method has a similar behavior as DB-RL, except for a few cases that can not be considered a great threat to the protection mechanism.

4.2 Alignment Record Linkage

In this section, we propose a new record linkage method for univariate microaggregation based on finding the optimal alignment between the original and the protected sorted attributes. We show that this method, which uses a DTW distance to compute the optimal alignment, provides in many cases enough information to the intruder to decide if the link is correct or not. Note that, standard record linkage methods never ensure the correctness of the linkage.

The alignment record linkage uses the *Dynamic Time Warping* distance [47] to find the optimal alignment between the sorted original values and the sorted cluster centroids (*i.e.*, the protected values using microaggregation). In general, DTW is a distance to find an optimal match between two given sequences (*e.g.* time series or two sorted attributes) with certain restrictions (*e.g.* minimum or maximum number of elements of one sequence which can be *aligned* with one element of the other one). The sequences are 'warped' non-linearly in one dimension (*e.g.* time or a given order) to determine a measure of their similarity independent of certain non-linear variations in the dimension. In this scenario, optimality is understood as the shortest alignment between the two sequences in a given distance (in our case shorter with respect to the Euclidean distance). When the intruder computes an alignment for one attribute, he is limiting the number of possible correct links to a small set of records. When he knows more than one attribute, the intruder can combine (intersect) all the sets (one for each known attribute) to obtain all the possible correct links (this technique is very similar to the one applied to RS-RL). This intersection often results in a single possible link. When such situation happens (*i.e.*, the final set only contains a single link) the intruder is completely sure that the link is correct because it is the only possible one.

In the experiments described in Section 4.2.2, we show that this new record linkage method improves the performance of standard ones when applied to compute the disclosure risk of univariate microaggregation. Thus, the real risk of univariate microaggregation is underestimated when computed using standard record linkage methods.

4.2.1 Algorithm Description

Standard record linkage methods need to compare all the original records X with all the protected records X'. This process has two clear drawbacks. The first one is the high computational cost of the comparisons. The second one is that results obtained may not be as good as the ones that would be obtained using a more specific record linkage method.

If one takes into account the two results from [19] presented in Section 2.4.2 which hold for all optimal univariate microaggregation algorithms, it is clear (from Result 1) that it is unnecessary to compute all the comparisons because original values (once sorted) are put in contiguous clusters. Therefore, if we sort the original values and the protected cluster centroids, and we find the optimal alignment (using the DTW algorithm), we can define for an attribute j of the protected record x_i a set of original records $B(x_{ij})$ and limit the comparisons done by the standard record linkage process to the original records stored in this set $B(x_{ij})$. Note that the size of the set $B(x_{ij})$ is always between k and $2k-1$, and that the DTW can be constrained to have between k and $2k-1$ horizontal shifts, for each vertical shift (these values are directly obtained from Result 2 presented in Section 2.4.2).

As in RS-RL, when more than one attribute is known, it is possible to repeat the same process for each attribute. Let the protected record to be linked have c attributes $attr'_1,\ldots,attr'_c$ and be represented by $x'_i = (x'_{i1},\ldots,x'_{ic})$. Then, the corresponding original record x_ℓ will necessarily satisfy the condition

$$x_\ell \in \cap_{1\leq j\leq c} B(attr'_i = x'_{ij}).$$

That is, the alignment record linkage method can reduce the search to the intersection of the sets of possible original records. Of course, the more attributes are considered, the less records will be in this intersection, and therefore the probability of finding the correct linkage will be larger.

When the intersected set has only one record the intruder is sure that this is the correct linkage. Note that, probabilistic or distance based record linkage methods never satisfy this property and, therefore, the intruder never knows which links are the correct ones. So, he only has some heuristic information. In our method, in the rare situations in which the final intersected set has more than one possible protected

record, the closest one is chosen.

Non-optimal microaggregation methods (as MDAV) do not need to satisfy Results 1 and 2 (presented in Section 2.4.2). Nevertheless, MDAV microaggregation (univariate version) satisfies Result 1, and from this result, it is clear that the clusters are non-overlapping. As a consequence thereof, we are sure that each value is assigned to a cluster with a centroid greater or lower than itself or equal. As all the values of these clusters are contiguous, the value has to belong to one of these *contiguous* clusters. Therefore, we have to compute the intersection using all the values assigned to such clusters, instead of computing the optimal alignment using the DTW distance.

If one attribute j has more than k equal values, then it is possible that some of them are put in a non contiguous cluster, but the centroid of the corresponding cluster is never further than $Max(d(x_{ij}, x'_{lower}), d(x_{ij}, x'_{upper}))$ of the original value. Here, x_{ij} represents, as stated above, the value for the element for the given attribute, x'_{lower} is the first centroid of the attribute with a value smaller than x_{ij} and x'_{upper} is the first centroid of the attribute with a value larger than x_{ij}.

4.2.2 Experimental A-RL Results

As in Section 4.1, we have considered the reference microdata files proposed in the CASC project [10]. We have protected both data sets using (a) the optimal univariate microaggregation and (b) the MDAV heuristic algorithm (univariate version). The protection method has been applied using several different values for k ranging from 5 to 50.

To understand better the behavior of the two microaggregation approaches, we present in Table 4.7 the SSE values obtained for each k configuration using both microaggregation algorithms over the two data sets. As it is expected, the larger the k, the larger the SSE. Thus, the difference between the original and the protected data set increases with larger k. In principle, when the SSE increases, the statistical utility of the protected data set decreases but at the same time it is more difficult for an intruder to link the protected values with the original ones. Comparing SSE values using optimal and MDAV univariate microaggregation with the same k parameter, we observe that optimal univariate microaggregation has lower SSE (better statistical utility) than MDAV. However, as we will see later, MDAV behaves better with respect

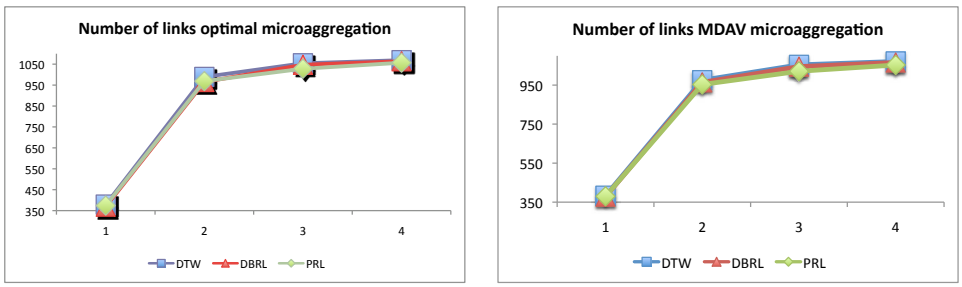
	k	Optimal	MDAV
univariate microaggregation	5	7.44	11.10
	10	25.34	29.20
	15	44.77	49.43
	20	65.45	69.92
	25	85.48	89.87
	30	104.57	108.73
	35	123.16	127.30
	40	141.36	145.34
	45	158.89	163.00
	50	176.82	180.68

Census data set

	k	Optimal	MDAV
univariate microaggregation	5	1.91	2.79
	10	9.87	10.91
	15	22.39	24.33
	20	37.59	39.78
	25	53.34	55.82
	30	69.23	71.44
	35	84.71	86.25
	40	98.16	100.33
	45	112.57	115.32
	50	128.20	131.25

EIA data set

Table 4.7: SSE results for univariate microaggregation.



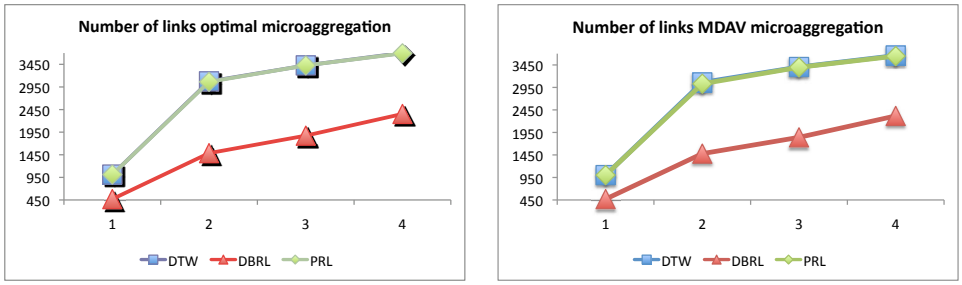
(a) (b)

Figure 4.8: Graphic representation of the number of links obtained with different record linkage techniques, applied to the Census data set protected with optimal microaggregation (a) and MDAV microaggregation (b) using $k = 50$.

to disclosure risk than optimal univariate microaggregation.

In these experiments, we are interested in the comparison between the new alignment record linkage method and standard ones. For this reason we compare four distinct scenarios corresponding to different sets of attributes of different size. That is, we assume intruders with different knowledges. In the worst (most dangerous) scenario, the intruder knows five attributes and in the best one, only two of them.

Figure 4.8 illustrates the number of correct links obtained for the census data set by the following three record linkage methods: alignment, distance based and probabilistic for optimal and MDAV microaggregation with $k = 50$. We can observe that the three methods obtain similar results, although our new method always presents



(a) (b)

Figure 4.9: Graphic representation of the number of links obtained with different record linkage techniques, applied to the EIA data set protected with optimal microaggregation (a) and MDAV microaggregation (b) using $k = 50$.

the best performance (larger number of correct links) and somewhat slightly worse results in P-RL. The best performance of our method is due to the fact that the new method limits the number of records to be compared in the linkage process. As this reduction never eliminates *good records*, the amount of false links is reduced. For example, using optimal microaggregation with $k = 50$ alignment record linkage obtains a number of correct links between 379 (the intruder only knows two attributes) and 1069 (the intruder knows five attributes), whereas probabilistic and distance based record linkage obtain between 374 and 1057, and between 369 and 1047, respectively, for the same scenarios.

Figure 4.8 illustrates the results for the EIA data set for optimal and MDAV microaggregation with $k = 50$. This figure shows clearly that the distance based record linkage obtains the worst results. As in the case of the Census data set, the alignment based record linkage method obtains the best results in all the scenarios considered. In all cases, the larger the number of attributes, the better is to use our new method. Similar results are obtained with respect to the k; the larger the k, the better is the performance of our method with respect to the others. For example, when using optimal microaggregation with $k = 50$, the alignment record linkage obtains a number of correct links beween 1012 (the intruder only knows two attributes) and 3703 (the intruder knows five attributes), whereas probabilistic and distance based record linkage obtain between 1011 and 3694, and between 469 and 2353, respectively, for the same scenarios.

From the comparison of these results we can conclude that alignment record linkage

obtains the best (or at least the same) result than the best of the standard record linkage method. However, such results are still more relevant since with such approach the intruder is completely sure in most of the cases that the links found are the correct ones. This is not possible using the standard re-identification methods (as only a probabilistic estimation of correctness can be given). So, in the light of these results, our method is better than the existing ones and can be exploited by any intruder.

In addition to these results, it is also worth to recall that the application of both DB-RL and P-RL need some parameters. P-RL needs probabilities of false match and false non-match, and DB-RL needs to assess the weights of attributes (of special relevance when data includes some bias). As such parameters are almost never needed in the alignment record linkage method (they are only needed in the few cases when the intersection of sets does not obtain singletons), the application of our approach is simpler. In our approach the single parameter needed (the number of horizontal shifts) can be directly extracted from the microaggregation parameter k.

In addition to that, if we compare the disclosure risk of both microaggregation methods, we find that the optimal microaggregation has a higher disclosure risk than MDAV. This seems to be related with the fact that the SSE is larger for MDAV than for optimal microaggregation. As the larger the SSE, the larger the difference between the original and the protected data set; it is natural that record linkage finds more difficulties in finding correct links with MDAV.

4.3 Projected Record Linkage

In this section we present a new record linkage technique, specific for multivariate microaggregation, which obtains more correct links than standard techniques. We have tested this new technique with MDAV microaggregation, PCP microaggregation and Zscores microaggregation.

4.3.1 Traditional Disclosure Risk Evaluation for Multivariate Microaggregation

In order to compare the microaggregation methods, we have protected the same microdata files as in the case of univariate microaggregation and rank swapping, with different instances of the three microaggregation methods, and then we have computed the resulting scores, after having applied the standard information loss and disclosure risk measures (detailed in Section 2.5).

Table 4.8 and 4.9 show the results of these experiments. Each of the three microaggregation methods has been applied with the following 9 parameterizations of the pairs (k,a): $k = 5, 15, 25$ for the minimal number of elements in the resulting clusters, and $a = 2, 3, 4$ for the number of attributes contained in each block of attributes to which microaggregation is applied. For example, Mic2.Zscores.15 refers to the Zscores microaggregation method applied to blocks of $a = 2$ attributes, with the constraint that resulting clusters must contain at least $k = 15$ records. When the total number of attributes is not a multiple of a (for example, this always happens with Census data set, since 13 is prime), the last non used attributes are non microaggregated and removed from the beginning.

For DLD, PLD and ID computation we have considered different cases, according to the number of groups of attributes of the original record(s), to be linked, that the intruder knows. This number varies from 2 to the total number of attributes of each data set. The values in the table are the average of the obtained correct links in all these cases, for each parameterization of each microaggregation method.

For the Census data set, the best scores are clearly obtained with MDAV; but for the larger EIA data set, both PCP microaggregation and (specially) Zscores microaggregation lead to better scores than MDAV. The lowest score using Zscores microaggregation is 16.55 (Mic3.Zscore.5) while the lowest score using MDAV is 29.15 (Mic3.MDAV.15). Since MDAV produces in (almost) all the cases less information loss (IL) than the other two methods, the difference has to come from the disclosure risk part. In effect, the standard methods (distance-based, probabilistic) for record linkage are less effective against the two projection based microaggregation than against MDAV. This difference in the disclosure risk seems to grow up with respect to the number of records of the data set, since it is much larger in the case of EIA than in

	v	k	IL	DLD	PLD	ID	DR	Score
Mic.v.PCP-k	2	5	80.96	12.93	5.70	42.60	25.96	53.46
	2	15	92.94	8.46	2.94	35.64	20.67	56.81
	2	25	84.77	6.61	1.94	32.93	18.6	51.69
	3	5	57.72	10.15	5.71	43.48	25.71	41.71
	3	15	71.28	4.35	3.49	37.36	20.64	45.96
	3	25	72.49	4.07	2.65	35.51	19.44	45.96
	4	5	72.23	6.48	3.06	45.12	24.94	48.59
	4	15	91.74	3.43	2.04	40.73	21.73	56.74
	4	25	92.17	2.92	1.71	39.72	21.02	56.59
Mic.v.Zscores-k	2	5	101.95	21.24	9.57	50.69	33.05	67.50
	2	15	121.76	16.17	6.11	46.14	28.64	75.20
	2	25	122.72	14.61	5.76	44.58	27.38	75.05
	3	5	90.72	14.97	11.48	51.89	32.56	61.64
	3	15	124.92	9.57	6.94	48.18	28.22	76.57
	3	25	128.25	9.23	5.86	46.73	27.14	77.69
	4	5	103.98	11.25	7.22	50.64	29.94	66.96
	4	15	136.65	6.53	3.75	46.74	25.94	81.29
	4	25	133.39	5.69	2.92	45.30	24.8	79.10
Mic.v.MDAV-k	2	5	19.30	69.06	49.22	74.77	66.95	43.13
	2	15	37.70	45.83	26.67	60.94	48.6	43.15
	2	25	47.16	28.56	16.81	51.93	37.31	42.23
	3	5	30.66	37.44	33.58	65.21	50.36	40.51
	3	15	42.76	22.75	19.38	54.79	37.93	40.34
	3	25	56.13	15.86	13.36	51.57	33.09	44.61
	4	5	34.67	31.9	24.35	61.37	44.75	39.71
	4	15	45.58	15.97	12.31	52.43	33.29	39.43
	4	25	54.6	11.2	7.08	45.09	27.12	40.86

Table 4.8: Score of different microaggregation methods and parameterizations when applied to Census data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$.

the case of Census (although they are different data sets, so it is impossible to formally conclude anything from this fact).

Summing up, the two projection based microaggregation methods can be a real alternative to MDAV in some situations, offering a better privacy level against (standard) re-identification attacks. However, as we will see in the next section, these conclusions are not completely right: we will show some new record linkage methods, specially designed for projection based microaggregation, which increase the real risk of re-identification (and so, the disclosure risk) of these methods. Maybe

	v	k	IL	DLD	PLD	ID	DR	Score
Mic.v.PCP-k	2	5	13.9	2.94	6.91	70.04	37.48	25.69
	2	15	17.24	1.72	2.37	67.67	34.86	26.05
	2	25	19.98	1.42	1.58	67.21	34.36	27.17
	3	5	16.08	2.47	2.69	62.79	32.68	24.38
	3	15	17.76	1.49	1.21	59.41	30.38	24.07
	3	25	18.49	1.31	0.9	58.49	29.8	24.14
	4	5	18.25	4.23	4.81	73.22	38.87	28.56
	4	15	16.39	1.96	2.13	70.48	36.26	26.33
	4	25	17.27	1.93	1.91	69.66	35.79	26.53
Mic.v.Zscores-k	2	5	4.11	33.88	44.25	28.14	33.6	18.86
	2	15	4.77	32.05	46.56	28.11	33.71	19.24
	2	25	4.95	31.15	49.39	28.08	34.18	19.56
	3	5	13.74	7.20	11.99	29.12	19.35	16.55
	3	15	15.82	4.66	8.46	29.63	18.09	16.95
	3	25	16.76	4.83	7.54	29.80	17.99	17.37
	4	5	20.06	6.87	11.85	32.92	21.14	20.60
	4	15	21.00	4.52	7.48	33.21	19.61	20.30
	4	25	27.50	3.96	6.84	36.28	20.84	24.17
Mic.v.MDAV-k	2	5	2.99	35.01	50.8	93.71	68.31	35.65
	2	15	5.49	20.02	31.49	86.5	56.13	30.814
	2	25	6.35	16.09	26.89	83.88	52.69	29.52
	3	5	7.64	21.47	34.53	85.52	56.76	32.2
	3	15	9.99	11.33	22.67	79.63	48.31	29.15
	3	25	11.12	9.6	18.32	77.63	45.8	28.46
	4	5	8.3	25.71	36.78	87.76	59.5	33.9
	4	15	19.16	12.66	21.31	81.57	49.28	34.22
	4	25	20.11	8.11	14.66	78.28	44.83	32.47

Table 4.9: Score of different microaggregation methods and parameterizations when applied to EIA data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$.

surprisingly, the new methods are also very effective when applied to MDAV. This will result in important changes to the real disclosure risk, and consequently the real score, of all these multivariate microaggregation methods.

4.3.2 The Projected Record Linkage Technique

Let X' be the result of applying a data protection method to a data set X, with n records and A attributes. Let y be an original record of X (obtained by an intruder,

Algorithm 5: Projected record linkage

Data: Y: external data set, X': protected data set
Result: LP: linked pairs

```
1 begin
2     foreach X'_i of X' do
3         Apply a projection to the a_i attributes in X'_i, for all the protected
          records. For example, the projection can be PCP or the Zscores one
          (intuitively, if X'_i has been obtained with PCP microaggregation,
          then PCP should be chosen as the projection method for record
          linkage)
4         Apply the same projection to the corresponding a_i attributes of the
          original record y
5     The result of the previous step is a projected original record ỹ and n
      projected protected records x̃', all of them with r values
6     Find the record x̃'_* which is closest to ỹ (for example, according to the
      Euclidean distance)
7     Let x'_* be the protected record whose projection was x̃'_*. Then the
      output link is y ↔ x'_* is added to LP
8 end
```

possibly from a different data set Y). The goal of the record linkage method is to find the record $x' \in X'$ which corresponds to the original y.

Projected record linkage technique (Pro-RL) is specifically designed for the case of microaggregation. Therefore, we can assume that X' is implicitly split into r blocks X'_i of a_i attributes, according to the blocks which have been considered to perform microaggregation. The algorithm of the projected record linkage method is defined in Algorithm 5.

In some way, the reasoning behind this strategy is that the results of projecting the original data, in X, and the protected data, in X', should be very similar, specially if the projection method applied in the record linkage algorithm has a similar statistical behaviour to the data protection method which transformed X into X'. The experiments that we have performed and explained in the following section, show that this intuition is right.

Although we have explained here a version of the projected record linkage technique which is specific for microaggregation, the idea can be easily extended to works with any data protection method. One can choose to project all the attributes of X into a single projected attribute, or to first split X in disjoint blocks of attributes, and then

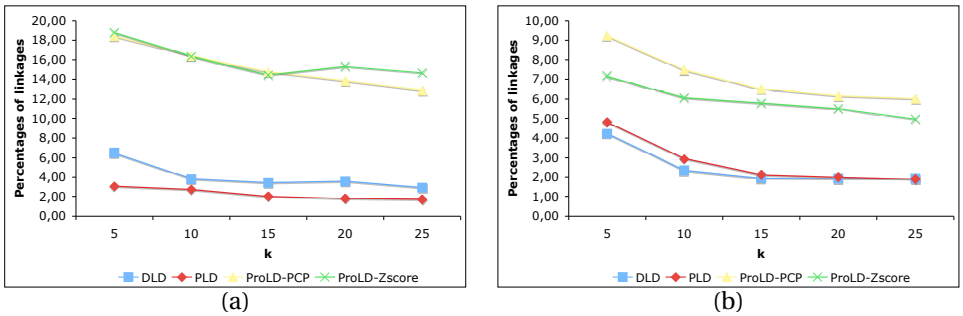


Figure 4.10: Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with PCP microaggregation with $v = 4$.

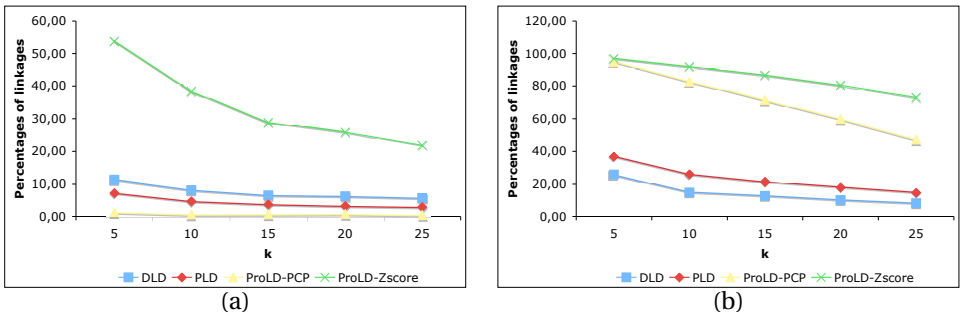


Figure 4.11: Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with Zscores microaggregation with $v = 4$.

to project the attributes in each block, separately. We have implemented and run this generic record linkage technique against other protection methods (rank swapping, noise addition) and, differently to what happens in the case of microaggregation, the results do not improve those obtained by standard record linkage techniques.

4.3.3 Consequences of Pro-RL in Multivariate Microaggregation

We have executed the two projected record linkage methods (using PCP and Zscores as the inherent projection mechanism) against all the protected data sets obtained in the experiments of Section 4.3.1; that is, the result of applying PCP microaggregation, Zscores microaggregation and MDAV, with different parameterizations, to the data

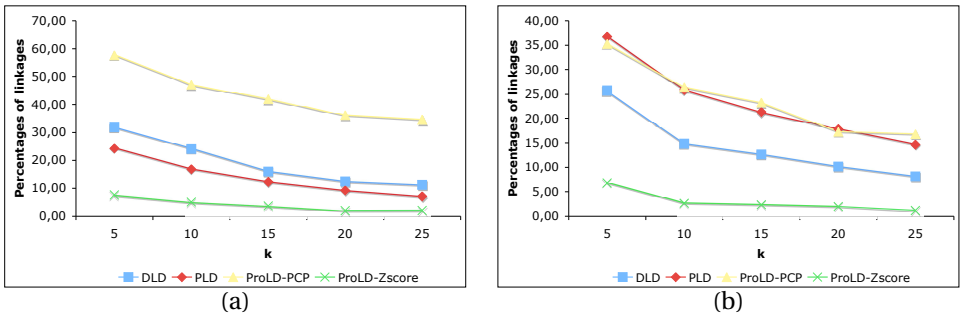


Figure 4.12: Percentage of correct links obtained with different record linkage techniques, applied to the Census data set (a) and EIA data set (b), protected with MDAV with $v = 4$.

sets Census and EIA.

Figures 4.10, 4.11 and 4.12 show the percentage of correct links obtained with different record linkage methods, when applied to both the Census and EIA data sets, protected with the three multivariate microaggregation schemes that we analyze in this section using the most secure configuration (using blocks of four attributes). The percentage is computed by taking into account different scenarios, where the intruder knows different amounts of groups of attributes of the original record(s) to be linked, from 2 to all the groups of attributes.

The results of these experiments can be summarized as follows:

- The new projected record linkage methods obtain, in almost all the cases, more correct links than the other (standard) record linkage methods. The difference is in some cases very significant. For instance, when the Census data set is protected using the Zscores microaggregation with $a = 4$ and $k = 25$ (the most protected configuration) DLD is equal to 5.69, a very small value, and ProjLD is equal to 21.85. A similar situation happens with MDAV, if we observe the configuration with $a = 3$ $k = 15$ (see Table 4.10), we observe an important increase of the disclosure risk: DLD is equal to 22.75 while ProjLD is equal to 60.03, much more than 50% of correctly linked records.

- Not surprisingly, the most effective record linkage method against PCP microaggregation is the projected one when PCP is used as the inherent projection, and the same happens with Zscores microaggregation and Zscores pro-

	v	k	IL	DLD	PLD	ProjLD	ID	ScoreOld	ScoreNew	ScoreMax
Mic.v.PCP-k	2	5	80.96	12.93	5.70	49.83	42.60	53.46	56.84	63.59
	2	15	92.94	8.46	2.94	45.67	35.64	56.81	60.14	66.80
	2	25	84.77	6.61	1.94	43.70	32.93	51.69	54.97	61.54
	3	5	57.72	10.15	5.71	27.59	43.48	41.71	43.35	46.63
	3	15	71.28	4.35	3.49	23.46	37.36	45.96	47.59	50.84
	3	25	72.49	4.07	2.65	20.90	35.51	45.96	47.42	50.35
	4	5	72.23	6.48	3.06	18.43	45.12	48.59	49.73	52.00
	4	15	91.74	3.43	2.04	14.77	40.73	56.74	57.74	59.74
	4	25	92.17	2.92	1.71	12.87	39.72	56.59	57.47	59.23
Mic.v.Zscores-k	2	5	101.95	21.24	9.57	99.59	50.69	67.50	74.52	88.55
	2	15	121.76	16.17	6.11	96.28	46.14	75.20	82.29	96.48
	2	25	122.72	14.61	5.76	90.65	44.58	75.05	81.75	95.17
	3	5	90.72	14.97	11.48	87.04	51.89	61.64	67.79	80.09
	3	15	124.92	9.57	6.94	67.41	48.18	76.57	81.50	91.36
	3	25	128.25	9.23	5.86	60.80	46.73	77.69	82.13	91.01
	4	5	103.98	11.25	7.22	53.75	50.64	66.96	70.67	78.09
	4	15	136.65	6.53	3.75	28.80	46.74	81.29	83.26	87.21
	4	25	133.39	5.69	2.92	21.85	45.30	79.10	80.56	83.48
Mic.v.MDAV-k	2	5	19.30	69.06	49.22	80.59	74.77	43.13	44.92	48.49
	2	15	37.70	45.83	26.67	69.48	60.94	43.15	45.92	51.46
	2	25	47.16	28.56	16.81	61.28	51.93	42.23	45.45	51.88
	3	5	30.66	37.44	33.58	75.06	65.21	40.51	43.81	50.40
	3	15	42.76	22.75	19.38	60.03	54.79	40.34	43.59	50.09
	3	25	56.13	15.86	13.36	53.49	51.57	44.61	47.85	54.33
	4	5	34.67	31.90	24.35	57.64	61.37	39.71	42.17	47.09
	4	15	45.58	15.97	12.31	41.99	52.43	39.43	41.75	46.40
	4	25	54.60	11.20	7.08	34.63	45.09	40.86	42.98	47.23

Table 4.10: New scores of the different microaggregation methods, applied to Census data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$.

jection. For instance, if we compare the ProjLD-PCP and ProjLD-Zscores in Figure 4.11 we observe that the best method is to apply Zscores projection in the record linkage when the data set is protected using Zscores microaggregation.

- Using PCP projection in the record linkage method to obtain correct links against Zscores microaagregation, or vice-versa, is not effective at all.
- When applied to MDAV, the projected record linkage method using PCP is very effective (more than any other method in the case of Census, and only over-

	v	k	IL	DLD	PLD	ProjLD	ID	ScoreOld	ScoreNew	ScoreMax
Mic.v.PCP-k	2	5	13.90	2.94	6.91	16.20	70.04	25.69	27.35	28.51
	2	15	17.24	1.72	2.37	9.17	67.67	26.05	26.98	27.83
	2	25	19.98	1.42	1.58	8.64	67.21	27.17	28.07	28.95
	3	5	16.08	2.47	2.69	7.17	62.79	24.38	24.97	25.53
	3	15	17.76	1.49	1.21	4.84	59.41	24.07	24.49	24.94
	3	25	18.49	1.31	0.90	3.85	58.49	24.14	24.46	24.83
	4	5	18.25	4.23	4.81	9.24	73.22	28.56	29.19	29.74
	4	15	16.39	1.96	2.13	6.52	70.48	26.33	26.90	27.45
	4	25	17.27	1.93	1.91	6.01	69.66	26.53	27.04	27.55
Mic.v.Zscores-k	2	5	4.11	33.88	44.25	98.08	28.14	18.86	26.88	33.61
	2	15	4.77	32.05	46.56	94.13	28.11	19.24	27.00	32.94
	2	25	4.95	31.15	49.39	88.99	28.08	19.56	26.79	31.74
	3	5	13.74	7.20	11.99	91.09	29.12	16.55	27.04	36.92
	3	15	15.82	4.66	8.46	79.03	29.63	16.95	26.25	35.07
	3	25	16.76	4.83	7.54	71.00	29.80	17.37	25.65	33.58
	4	5	20.06	6.87	11.85	96.85	32.92	20.60	31.85	42.47
	4	15	21.00	4.52	7.48	86.61	33.21	20.30	30.57	40.46
	4	25	27.50	3.96	6.84	73.17	36.28	24.17	32.82	41.11
Mic.v.MDAV-k	2	5	2.99	35.01	50.80	54.48	93.71	35.65	38.08	38.54
	2	15	5.49	20.02	31.49	36.66	86.50	30.81	32.89	33.53
	2	25	6.35	16.09	26.89	32.84	83.88	29.52	31.61	32.36
	3	5	7.64	21.47	34.53	35.40	85.52	32.20	33.94	34.05
	3	15	9.99	11.33	22.67	20.53	79.63	29.15	30.30	30.57
	3	25	11.12	9.60	18.32	15.35	77.63	28.46	29.18	29.55
	4	5	8.30	25.71	36.78	35.29	87.76	33.90	35.10	35.28
	4	15	19.16	12.66	21.31	23.29	81.57	34.22	35.55	35.79
	4	25	20.11	8.11	14.66	16.81	78.28	32.47	33.56	33.83

Table 4.11: New scores of the different microaggregation methods, applied to EIA data set. Mic.i.var.j corresponds to microaggregation using variation var (either PCP, Zscores of MDAV) with $v = i$ and $k = j$.

come by Probabilistic Record Linkage in some instances of EIA). However, using Zscores as the inherent projection leads to quite bad results. For example, if we observe the MDAV configuration with $a = 3$ and $k = 25$ in the Census data set (Table 4.10), we observe that ProjLD is equal to 53.49 while DLD and PLD are equal to 15.86 and 13.36. Here, the ProjLD is four times higher than standard record linkage disclosure risks.

Obviously, the fact that the projected record linkage technique has a higher success rate than the other record linkage techniques must have a direct impact in the real

disclosure risk (and so, in the score) of the studied multivariate microaggregation methods.

One possibility is to compute the disclosure risk, again, as the average of the Interval Disclosure risk (ID) and the Record Linkage risk, but now this last value is computed as the average of three values: Distance based Linkage Disclosure risk (DLD), Probabilistic Linkage Disclosure risk (PLD) and Projected Linkage Disclosure risk (ProjLD), which is the maximum percentage of correct links obtained by the projected record linkage technique, using either PCP or Zscores as the inherent projection. When the protection method is PCP microaggregation or MDAV, the maximum is obtained by using PCP as the inherent projection for record linkage; when the protection method is Zscores microaggregation, the maximum is obtained by using Zscores projection.

Another possibility, maybe more realistic, is to assume that the intruder who wants to break the privacy of the protection method knows which is the most successful strategy to find correct links. For example, after reading this section, he may know that projected record linkage with PCP as the inherent projection is the best known technique to attack PCP microaggregation. In this case, it is clear that he will always use this technique to find correct links between original and protected records. Therefore, considering other values to compute the linkage disclosure risk would make no sense in this situation; the real linkage disclosure risk would be defined as the maximum among all the linkage disclosure risk values: DLD, PLD, ProjLD.

Summing up, there would be two different alternatives to compute the new scores of these methods. In the first one, that we call *ScoreNew*, the disclosure risk is computed as $DR\_New = 0.5 \cdot (0.333 \cdot DLD + 0.333 \cdot PLD + 0.333 ProjLD) + 0.5 \cdot ID$. In the second one, that we call *ScoreMax*, the disclosure risk is computed as $DR\_Max = 0.5 \cdot MAX\{DLD, PLD, ProjLD\} + 0.5 \cdot ID$. As usual, the final score value is computed as the average of the information loss (IL) and the corresponding disclosure risk (DR).

Tables 4.10 and 4.11 show the new values of the scores, which strongly depend on the success rate (ProjLD) of the new projected record linkage technique. This can be easily verified by comparing the new ScoreNew and ScoreMax with the standard Score, computed by considering only standard (and generic) record linkage techniques. We consider the same parameterizations than in Section 4.3.1.

After looking at these tables, one can conclude that the real scores of multivariate microaggregation methods are not as good as one could think (see the classification of data protection methods in [21]). In particular, the percentage of correct links are now over the threshold of 50% in many cases, so it is not clear if these methods offer the desired level of privacy.

Chapter 5

Record Linkage using Fuzzy Integrals

Standard record linkage algorithms assume that both data sets are described using the same attributes, *i.e.* they assume that the two microdata files A and B are described using the same attributes. In this chapter, we will study the non-standard case when attributes are not the same. In this scenario, record linkage methods described in Section 2.3 and in Chapter 4 cannot be applied.

5.1 An Alternative Disclosure Risk Scenario

At present, there are several works in the literature dealing with scenarios in which data sets do not include common attributes. Most of the research corresponds to attribute matching or schema matching [18, 51, 52]. This is possible due to attribute matching is computationally simpler than record matching because the amount of redundant information existing in the data for attributes is larger than that for records.

[63] introduced a scenario in which data sets do not share the same attributes. Re-identification is still possible in such scenario when the attributes still represent similar information. This would be the case, for example, if we have the attribute

Income-tax in data set B while the data set A contains *Net-income*.

In general, re-identification can still be achieved in this context under the following assumptions:

Assumption 1 *A set of common individuals is shared by both files.*

Assumption 2 *Data in both microdata files contain, implicitly, similar structural information.*

According to [63], we can say that structural information of data is defined as any organization of the data that allows explicit representation of the relationship between individuals. This structural information is obtained from the data files through manipulation of such data (*e.g.* using clustering techniques or any other data analysis or data mining techniques). In other words, even though there are no common attributes, there is substantial correlation between some attributes in both data sets; or applying some clustering techniques we obtain the same clusters for both sets of records, this latter approach was considered in [23];

Figure 5.1 represents a case that satisfies this latter assumption. In this case, two data sets A and B are considered. Data set A describes a set of retailers in terms of the attributes {*Benefits, Start-up costs*}, whereas B describes the same retailers with the attribute *Business type*. In this case, some re-identifications are possible.

As different formalisms can be used for representing the structural information, different techniques are needed to extract such structural information. In this section, we will focus on structural information represented by means of numerical representatives (as in the example given above of *Income-tax* vs *Net-income*). Therefore, the following assumption is considered:

Assumption 3 *Structural information is expressed by means of numerical representatives for each individual.*

Here, we describe an approach for record linkage for the case that data sets do not share attributes and when the structural information is expressed using numerical representatives. Then, we show that under a few conditions aggregation functions

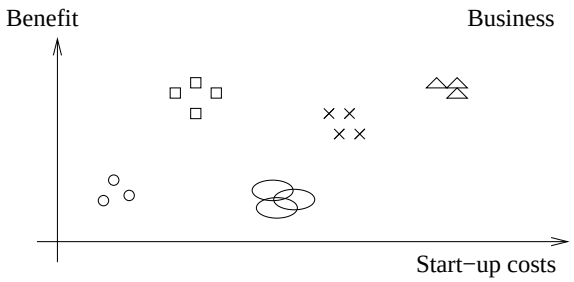


Figure 5.1: Graphical representation of an artificial problem that satisfies Assumption 2: Data set A with attributes {Benefits, Start-up costs} and data set B with attribute {Business type}. In this figure, business types are represented using squares, ellipses, triangles, and so on.

arise as the appropriate functions for building such representatives. Aggregation functions (described in Section 2.1) are functions that combine (aggregate) N values into a single one.

To tackle this problem, we consider the transformation of data sets A and B into two new data sets A' and B' in order that standard re-identification algorithms can be applied on this latter pair of microdata files (A', B').

To do so, we consider the construction of several representatives for each record a in A and each record b in B so that re-identification can be performed over such representatives. This process is detailed below:

- Firstly, we consider a set of functions f_i for building the representatives. In general, we consider that f_i is a function of both the record and of the whole data set A. Therefore, being a a record in A, $f_i(a, A)$ stands for a representative of the record. We denote by $\mathscr{F} = \{f_i\}$ for $i = 1, \ldots, k$ the set of considered functions.

- Then, we apply the functions in $\mathscr{F}$ to the records a in A to obtain a'. Formally speaking $a' := \mathscr{F}(a, A)$ where:

$$a' := \mathscr{F}(a, A) = (f_1(a, A), \ldots, f_k(a, A))$$

- Now, assuming that functions in $\mathscr{F}$ are also applicable to records b in B, we

Algorithm 6: Transform File

```
Data: A: data set, 𝓕: set of functions
Result: A': transformed data set
1 begin
2     foreach a ∈ A do
3         a':=new record(f_1(a, A), ..., f_k(a, A))
4         write(a',A')
5 end
```

define records b' in B in a similar way:

$$b' := \mathscr{F}(b,B) = (f_1(b,B),\ldots,f_k(b,B))$$

- Finally, we define files A' and B' in terms of the new records a' and b'. That is:

$$A' := \{\mathscr{F}(a,A)\}_{a\in A}$$

$$B' := \{\mathscr{F}(b,B)\}_{b\in B}$$

Therefore, given the set of functions $\mathscr{F} = \{f_i\}$ for $i = 1,\ldots,k$, and applying each f_i to every record in A and B, we obtain data sets A' and B'. This process is defined in Algorithm 6 in a procedural point of view.

Thus, data sets A' and B' are obtained as:

A':=`transformFile(`A`, `$\mathscr{F}$`);` B':=`transformFile(`B`, `$\mathscr{F}$`);`

With this construction, both data sets A' and B' contain the same number of records as A and B, and records in both microdata files are described using the same kind of representatives. Therefore, both data sets can be considered as described using the same attributes and, as such, standard re-identification algorithms can be applied to the pair (A', B').

At this point, it is clear that a crucial decision is the selection of functions in $\mathscr{F}$. This is reviewed in detail in next section.

5.1.1 Aggregation Functions for Building Representatives

For building the representatives, we have to select the functions in $\mathscr{F}$. Firstly, we show that aggregation functions are suitable functions for this purpose, and then, on the basis of the properties we require for f_i, we will illustrate that OWA is an appropriate selection.

So, we come back to the requirements for functions $f \in \mathscr{F}$:

i) The outcome of f applied to a record a should not depend on the values of the other records in A. This condition corresponds to the so-called *condition of independence of irrelevant alternatives*, and its inclusion excludes functions based *e.g.* on principal component analysis. Formally speaking, this condition implies that functions $f(a, A)$ do only depend on a and should not depend on the other values in A.

ii) When all the values of a record are equal, the representative is this value. This condition implies that all functions f satisfy unanimity (idempotency).

iii) The representatives should be monotonic with respect to their inputs. That is, given two records $a = (a_1, \ldots, a_N)$ and $a^* = (a_1^*, \ldots, a_N^*)$ so that $a_i \leq a_i^*$, the representatives of a should always be smaller than (or equal to) the representatives of a^*.

iv) When there is no prior knowledge on the attributes (if this is not the case, other methods might be used for linkage), no preference should be given to any of the attributes involved in the process. In other words, the order of the attributes is irrelevant. This is formally expressed saying that a permutation of the attributes does not affect the output:

$$f(a_1, \ldots, a_N) = f(a_{\pi(1)}, \ldots, a_{\pi(N)})$$

where π is a permutation of the indices. That is, f is a symmetric function.

v) The function should be easily extensible to an arbitrary number of parameters, so that the same procedure can be applied to files with an arbitrary number of attributes. In this way, we can apply $\mathscr{F}$ to both data sets A and B although the number of attributes in each one is different.

vi) This function should be parameterizable so that different representatives can be computed for the same record.

These requirements constrain functions in $\mathscr{F}$. In particular, the first condition implies that functions $f_i(a, A)$ can be defined in terms of another function f'_i that depends only on a. That is, $f_i(a, A) = f'_i(a)$. Then, conditions (ii) and (iii) imply that functions f'_i are aggregation functions as they should be idempotent and monotonic (see Definition 1 in Section 2.1). Therefore, the following holds:

Proposition 1 *Let the functions in $\mathscr{F}$ satisfy the condition of independence of irrelevant alternatives, idempotency and monotonicity. Then, the functions in $\mathscr{F}$ are aggregation functions.*

Additionally, when conditions (iv), (v) and (vi) are required for aggregation functions, we have that some of such operators are discarded. This is the case of the weighted mean (that is not symmetric and not easily extensible because it requires weights for each attribute) or the arithmetic mean (that is not parameterizable). The OWA operator and other fuzzy integrals with symmetric fuzzy measures are some of the few ones that are appropriate. They are symmetric and parameterizable (in terms of the function Q). In relation to the property of being extensible for an arbitrary number of parameters, we have that not all definitions for OWA operators are appropriate. For example, definitions based on weighting vectors (as the original definition in [77]) are not appropriate because additional arguments would require additional weights. Nevertheless, the definition given in Definition 2 is appropriate because the same function Q can be used for an arbitrary value of N.

Taking all this into account we state that we can use either OWA operators, Sugeno integrals or twofold integrals (all based on non-decreasing functions Q). This selection is valid as the following proposition holds:

Proposition 2 *The functions OWA_Q, SI_Q $TI_{Q,Q}$ and satisfy conditions (i)-(vi) for all non-decreasing functions Q.*

Additionally, as functions satisfying condition (v) above are applicable to an arbitrary number of parameters, they can also be applied to situations in which data contains missing values. In this case, instead of defining record a' as before, we would define:

$$a' := \mathscr{F}(a, A) = (f_1(\hat{a}, A), \ldots, f_k(\hat{a}, A))$$

where $\hat{a}$ is a projection of a over those attributes with non-missing values in a. For all this, the following holds:

Proposition 3 *The functions OWA_Q, SI_Q and $TI_{Q,Q}$ are applicable to records with missing data for all non-decreasing functions Q.*

5.1.2 Example

Now, we illustrate with an illustrative example the method we have proposed. In Section 5.2 we will describe several experiments with real data.

Let us consider the two data files A and B represented in Figure 5.1. Data set A consists of 10 records described in terms of 4 attributes. All attributes are numerical and numbers belong to the interval $[0,1]$. Data set B contains the same data included in A but the attributes have been permuted.

Standard re-identification algorithms cannot be applied to establish links between the records in A and B without knowing the correspondence between attributes in A and B. Nevertheless, in this case, we can apply the method described in this section. To do so, we need to define the set of functions $\mathscr{F}$. We will use here the OWA operator with $Q_\alpha(x) = x^\alpha$ with several values of α. In particular, we consider 10 different functions corresponding to Q_α with the following values of α:

$$\alpha = (1/5, 2/5, 3/5, 4/5, 5/5, 6/5, 7/5, 8/5, 9/5, 10/5)$$

By applying these aggregation functions, we obtain exactly the same records for both microdata files A and B presented in Figure 5.1. The records obtained are given in Table 5.2. Now, as both files contain exactly the same records, the re-identification is trivial.

Note that the first row in Table 5.2 is obtained by applying the OWA operator to the first row of Tables (a) and (b) of Table 5.1 using the function $Q_\alpha(x) = x^\alpha$ with

	$attr_1$	$attr_2$	$attr_3$	$attr_4$
r_1^A	0.2	0.4	0.2	0.4
r_2^A	0.1	0.2	0.1	0.2
r_3^A	0.5	0.6	0.5	0.1
r_4^A	0.8	0.4	0.4	0.7
r_5^A	0.9	0.2	0.0	0.0
r_6^A	0.2	0.2	0.3	0.9
r_7^A	0.5	0.3	0.2	1.0
r_8^A	0.0	0.1	0.5	1.0
r_9^A	1.0	0.0	0.9	0.2
r_{10}^A	0.5	1.0	1.0	0.8

(a)

	$attr'_1$	$attr'_2$	$attr'_3$	$attr'_4$
r_1^B	0.4	0.2	0.2	0.4
r_2^B	0.2	0.1	0.1	0.2
r_3^B	0.1	0.5	0.5	0.6
r_4^B	0.7	0.4	0.8	0.4
r_5^B	0.0	0.0	0.9	0.2
r_6^B	0.9	0.3	0.2	0.2
r_7^B	1.0	0.2	0.5	0.3
r_8^B	1.0	0.5	0.0	0.1
r_9^B	0.2	0.9	1.0	0.0
r_{10}^B	0.8	1.0	0.5	1.0

(b)

Table 5.1: Data sets A and B for re-identification.

	$Q_{1/5}$	$Q_{2/5}$	$Q_{3/5}$	$Q_{4/5}$	$Q_{5/5}$	$Q_{6/5}$	$Q_{7/5}$	$Q_{8/5}$	$Q_{9/5}$	$Q_{10/5}$
$r_1'^A = r_1'^B$	0.37	0.35	0.33	0.32	0.30	0.29	0.28	0.27	0.26	0.25
$r_2'^A = r_2'^B$	0.19	0.18	0.17	0.16	0.15	0.14	0.14	0.13	0.13	0.13
$r_3'^A = r_3'^B$	0.55	0.51	0.48	0.45	0.43	0.40	0.38	0.36	0.35	0.33
$r_4'^A = r_4'^B$	0.74	0.69	0.64	0.61	0.56	0.55	0.53	0.51	0.49	0.48
$r_5'^A = r_5'^B$	0.71	0.55	0.44	0.35	0.28	0.22	0.18	0.14	0.12	0.094
$r_6'^A = r_6'^B$	0.74	0.62	0.53	0.46	0.40	0.36	0.32	0.30	0.28	0.26
$r_7'^A = r_7'^B$	0.85	0.73	0.63	0.56	0.50	0.45	0.41	0.38	0.36	0.34
$r_8'^A = r_8'^B$	0.82	0.68	0.57	0.47	0.40	0.34	0.29	0.25	0.22	0.19
$r_9'^A = r_9'^B$	0.87	0.77	0.67	0.59	0.53	0.47	0.41	0.37	0.33	0.29
$r_{10}'^A = r_{10}'^B$	0.96	0.92	0.88	0.85	0.83	0.80	0.78	0.755	0.74	0.72

Table 5.2: Data set A (and B) for re-identification.

$\alpha = 1/5, \ldots, 10/5$. In particular, the first column in Table 5.2 corresponds to $\alpha = 1/5$, second column to $\alpha = 2/5$ and so on since the tenth column where $\alpha = 10/5$.

Therefore, the element in the i-th row, column Q_α in Table 5.2 corresponds to $OWA_{Q_\alpha}(r_i^A)$. Of course, $OWA_{Q_\alpha}(r_i^A)$ is equivalent to $OWA_{Q_\alpha}(r_i^B)$ in this example because r_i^B is a permutation of r_i^A and the OWA operator is symmetric.

This example can be considered as too simplistic. Nevertheless, this same situation arises in database integration with unlabeled attributes or with inconsistent labeled attributes. In a more general case, instead of having a permutation of exactly the same attributes, we might have attributes in one data set that are combinations of some attributes in the other database.

5.2 Experiments

The approach presented herein has been extensively tested with several microdata files, considering three types of aggregation functions (OWA, Sugeno integral and twofold integral) and considering three different quantifiers. We have used the seven data sets extracted from the UCI repository [46] and the Census data set extracted from the CASC project. All these data sets are described in Section 2.6.

5.2.1 Preprocessing

To test the re-identification algorithms the microdata files have been partitioned. Each data set was split into two new data sets in such a way that both data sets contained the same records but only some of the attributes. Attribute selection was done on the basis of the correlation coefficients. In particular, attributes with a low correlation coefficient over all the other attributes were discarded and pairs of attributes with a correlation coefficient of at least 0.7 were separated assigning one of each to a different microdata file.

Below we list the microdata files used in the experiments, and for each one the two sets of attributes considered (each set defines one microdata file). For example, in the case of the Iris Plants Database, that contains 150 records, one microdata file contains the 150 records but only the attributes *Sepal-length* and *Petal-length* and the other one (that also contains 150 records) contains the attributes *Sepal-width* and *Petal-width*.

- **Iris data set.** {a1, a2}, {a3, a4}
- **Abalone data set.** {a4, a5, a7}, {a2, a3, a6, a8}
- **Ionosphere data set.** {a1, a2, a3, a4, a5, a6}, {a7, a8, a9, a10, a11, a12}
- **Dermatology data set.** {a1, a2, a3, a4, a5, a6, a7, a8, a9}, {a10, a11, a12, a13, a14, a15, a16}
- **Housing data set.** {a1, a2, a3, a4}, {a5, a6, a7}
- **Water Treatment data set.** {a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11}, {a12, a13, a14, a15, a16, a17, a18, a19, a20, a21, a22, a23, a24, a25}

- **WDBC data set.** {a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12}, {a12, a13, a14, a15, a16, a17, a18, a19, a20, a21, a22}

- **Census data set.** {a1, a3, a8, a9, a10, a12, a13}, {a2, a4, a5, a6, a7, a11}

Before applying the re-identification algorithm, the data has been normalized. We have considered both ranging (denoted below by N_1) and standardization (N_2). Missing values have been replaced by zero (after normalization).

5.2.2 Tests

The procedure described in Section 5.1.1 has been applied to each pair of data set. For each pair, we have selected at random sets of 100 records and applied the re-identification algorithms. 10 executions have been applied and the average number of re-identifications has been computed.

Experiments have been done using the OWA operator as well as for the Sugeno integral and twofold integral with respect to a fuzzy measure of the form $\mu(A) = Q(|A|/N)$ (and also with respect to $\mu(B) = Q(|B|/N)$ in the case of the twofold integral). For the aggregation functions, three different families of non-decreasing functions were considered. The functions and the parameters used are the following ones:

1. $Q^e_\alpha(x) = x^\alpha$ for $\alpha = 1/5, 2/5, 3/5, \ldots, 10/5$

2. $Q^s_\alpha(x) = 1/(1 + e^{(\alpha - x)*10})$ for $\alpha = \{0, 0.1, \ldots 0.9\}$

3. $Q^t_\alpha(x) = \begin{cases} 0 & \text{if } x \leq \alpha \\ 1 & \text{if } x > \alpha \end{cases}$ for $\alpha = \{0, 0.1, \ldots 0.9\}$

Here, Q^e stands for exponent, Q^s for sigmoidal and Q^t for threshold. Figures 5.3, 5.2, and 5.4 give a graphical representation of these functions.

Once we have obtained microdata files with common attributes, we used both probabilistic and distance-based record linkage.

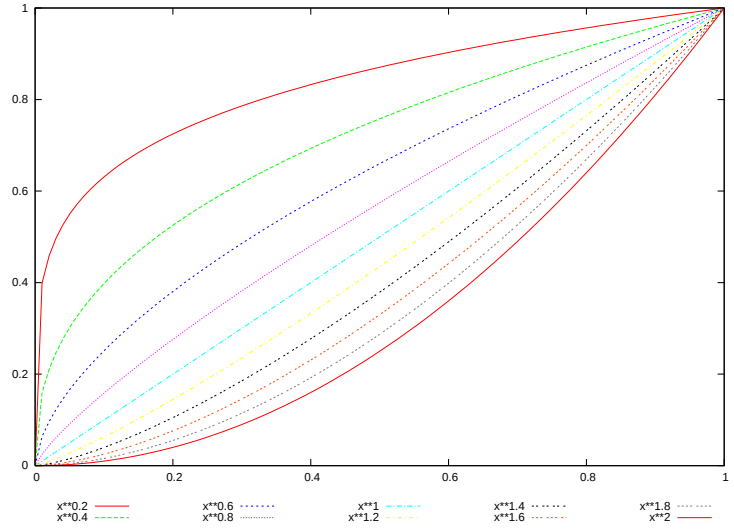


Figure 5.2: Graphical representation of Q^e_α for $\alpha = 1/5, \ldots, 10/5$.

5.2.3 Results

The results for the data sets with better performance are given in Tables 5.4, 5.5, 5.6, and 5.7. They correspond to the microdata files Abalone, Ionosphere, Census, and WDBC. Iris, Dermatology and Housing led to poor results. The bad performance of Iris and Housing was probably due to the reduced number of attributes, that did not permit to express structural information correctly. The file Water Treatment, not included herein, led to results similar to Census with around 10 re-identifications and a maximum of 16.

In the aforementioned tables, we give the average number of re-identifications obtained over 10 executions, considering in each execution the parameters described above: (i) either OWA, the Sugeno integral (denoted SI) or twofold integral (TI) with respect to a symmetric fuzzy measure; (ii) either distance-based record linkage (DBRL) or probabilistic one (P-RL); (iii) either ranging (denoted N_1) or standardization (N_2) as the normalization method and (iv) either Q^e, Q^s or Q^t as the non-decreasing functions Q that with OWA or SI define the set $\mathscr{F}$; to define the set $\mathscr{F}$ for TI we have selected the three best possible combinations of Q^e, Q^s and Q^t in each case.

The experiments show that, except for the data sets with poor performance, at least

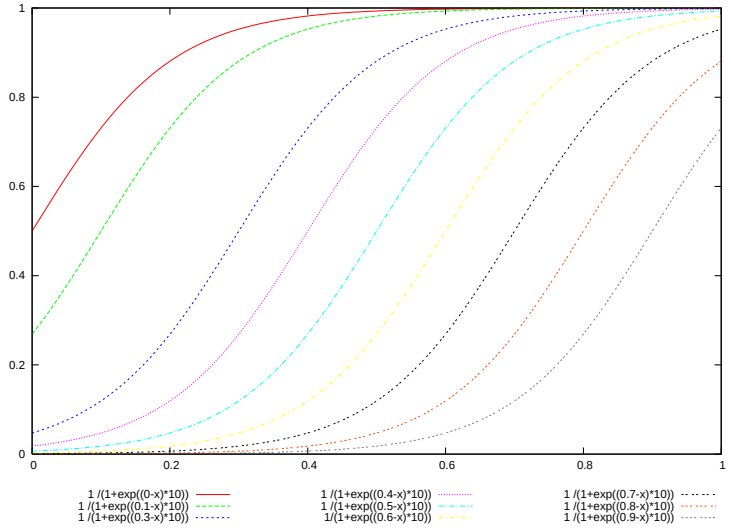


Figure 5.3: Graphical representation of Q_α^s for $\alpha = 0, 0.1, \ldots, 0.9$.

10% of the records were re-identified achieving averages of 18.2 or 22.6 for data sets WDBC and Ionosphere. The maximum percentage of records re-identified in an experiment was 26% in WDBC and 28% in Ionosphere. These values are not given in the tables, since tables only include the averages of 10 executions.

The evaluation of our approach is not straightforward as there are no systematic alternative approaches to deal with the same problem. Two simple methods were considered in [23] for the Census data set:

- The one-dimensional ranking based on first principal component: that permitted to correctly re-identify 5 out of 90 records.
- The one-dimensional ranking based on the sum of z-scores: by using this approach, 5 out of 90 records were correctly re-identified.

For the same problem, by using the approach described herein, we were able to correctly re-identify 12 records out of 100 and the better average over 10 runs was 10.4 (see Table 5.6).

An alternative way to assess the successfulness of the method is to consider the probability of random linkage. The probability of randomly obtaining r or more linkage

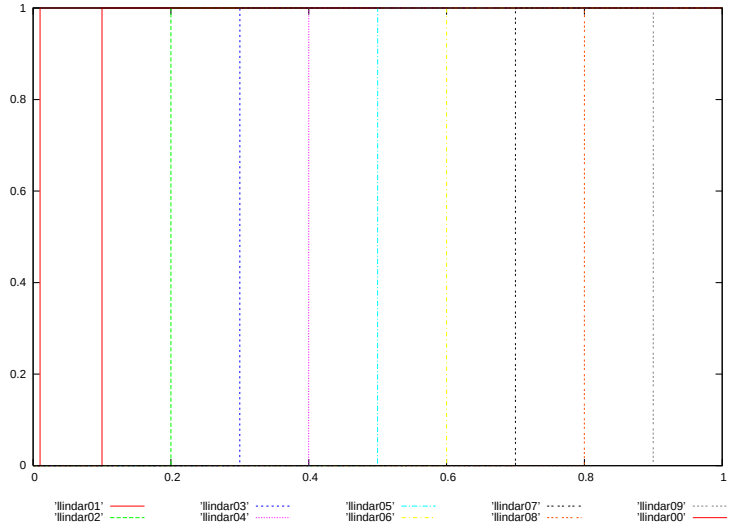


Figure 5.4: Graphical representation of Q_α^t for $\alpha = 0, 0.1, \ldots, 0.9$.

out of n is defined in the next proposition.

Proposition 4 *[64, 67] If A and B both contain n records corresponding to the same set of n individuals, the probability of correctly re-identifying exactly r individuals by a random strategy is*

$$\frac{\sum_{v=0}^{n-r} \frac{(-1)^v}{v!}}{r!} \tag{5.1}$$

Table 5.3 gives the probabilities for some values of r when the number of records is 100. It can be seen that the probability of obtaining between 15 and 30 records (as obtained in some of the experiments reported here) is almost zero. For example, the probability of re-identifying 26 records or more as in wdbc is $9.47 \cdot 10^{-28}$ and for 28 records as in Ionosphere is $1.24 \cdot 10^{-30}$.

Finally, it is possible to compare the results of our approach with the success rate of re-identification of standard record linkage when an original data set and a masked data set are compared. In [22], around 300 experiments are described and an average number of re-identifications of 26.12% was obtained for distance-based record linkage and 19.72% for probabilistic one. Here, the rate is smaller but considering that we use data sets not sharing attributes the performance is acceptable, specially since

r	prob. $\|links\| = r$	prob. $\|links\| \geq r$
0	0.36787944	1
1	0.36787944	0.63212056
2	0.18393972	0.26424112
3	0.06131324	0.08030140
5	0.00306566	0.00365985
10	1.0138E-7	1.1143E-7
15	2.8132E-13	3.0000E-13
20	1.5121E-19	1.5875E-19
25	2.3717E-26	2.4664E-26
26	9.1219E-28	9.4723E-28
28	1.2066E-30	1.2496E-30
30	1.3869E-33	1.4331E-33
50	1.2096E-65	1.2338E-65
100	1.071E-158	1.071E-158

Table 5.3: Probabilities of having r correct links and of having more or equal than r links for 100 records.

		OWA			SI			TI		
		Q^e	Q^s	Q^t	Q^e	Q^s	Q^t	Q^e Q^t	Q^s Q^t	Q^t Q^t
N_1	DB-RL	6.5	5.9	6.7	4.8	4.2	6.7	6.6	6.4	6.1
	P-RL	3.9	5.2	1.8	5.5	5.2	1.8	5.3	4.1	4.2
N_2	DB-RL	9.9	7.9	8.8	5.6	6.5	7.0	11.2	8.6	7.3
	P-RL	6.3	8.4	2.2	5.6	6.2	2.4	8.2	7.6	8.1

Table 5.4: Average number of re-identified records for the Abalone example.

the best performance for Ionosphere is 28 (larger than 26.12%) and the best average for the same problem for probabilistic record linkage is 22.2%, still larger than the result in [22] for probabilistic record linkage. Similar results were reported [24] with respect to re-identification of synthetic data.

The results permit to compare the different approaches experimented. In general, we can state that the use of the Choquet integral and twofold integral are more successful than that of the Sugeno integral. Also, we may add that the use of the quantifier Q^t leads to better results than the use of Q^e and Q^s. The results also show that distance-based record linkage is more suitable for numerical data. Finally, we have that the use of standardization is, in general, preferable over ranging.

		OWA			SI			TI		
		Q^e	Q^s	Q^t	Q^e	Q^s	Q^t	Q^s Q^t	Q^s Q^s	Q^t Q^s
N_1	DB-RL	14.4	21.8	21.9	11.6	20.3	21.9	19.3	19.1	17.7
	P-RL	12.9	22.2	3.9	10.8	20.7	3.9	22.4	22.6	17.7
N_2	DB-RL	5.7	7.9	8.6	6.4	6.9	8.0	8.2	7.3	7.1
	P-RL	4.2	7.5	1.3	4.9	6.2	1.6	6.1	4.3	5.8

Table 5.5: Average number of re-identified records for the Ionosphere example.

		OWA			SI			TI		
		Q^e	Q^s	Q^t	Q^e	Q^s	Q^t	Q^e Q^s	Q^s Q^e	Q^s Q^s
N_1	DB-RL	7.1	9.5	7.5	6.1	8.6	7.5	9.9	8.7	8.8
	P-RL	4.7	9.6	10.4	6.0	7.9	10.4	9.6	9.1	7.3
N_2	DB-RL	8.4	8.8	9.9	4.3	3.6	5.0	9.8	9.6	9.3
	P-RL	7.4	8.8	5.0	3.7	3.5	2.2	7.1	8.2	7.6

Table 5.6: Average number of re-identified records for the Census example.

		OWA			SI			TI		
		Q^e	Q^s	Q^t	Q^e	Q^s	Q^t	Q^s Q^t	Q^t Q^s	Q^t Q^t
N_1	DB-RL	5.0	7.0	4.4	5.5	5.8	4.4	6.8	5.8	5.1
	P-RL	4.4	7.1	8.0	6.3	5.8	8.0	7.8	6.2	4.3
N_2	DB-RL	10.8	15.8	18.2	3.3	4.6	5.1	17.7	18.2	16.4
	P-RL	10.5	14.8	16.2	3.3	4.7	4.6	16.4	14.1	12.3

Table 5.7: Average number of re-identified records for the WDBC example.

Chapter 6

Time Series Protection

In this chapter we present some results about time series protection and re-identification. We propose a complete framework to evaluate time series protection methods. We also present some empirical results to show how our framework works.

To the best of our knowledge, neither information loss nor disclosure risk measures are described for the case of time series protection. In this chapter we propose a group of information loss measures designed for time series protection evaluation. Such measures consider the main uses of time series, *e.g.* forecasting and autocorrelation analysis. We also propose the use of the record linkage methods, specially adapted to time series, as the most straightforward way to compute the disclosure risk. Finally, we propose to combine both IL and DR measures in a final score using the arithmetic mean.

6.1 Time Series Protection

A lot of effort has been made in the last few years to develop protection methods, see [1, 21] for a survey. Nevertheless, the research on protection methods focuses on the anonymization of numerical and categorical data.

However, in the real world, an increasing percentage of the released information has an implicit or explicit time component. This is the case of *e.g.*, income or stock prices.

Similarly, data accumulation through consecutive years (*e.g.*, economical data from companies or census data from individuals) can also be considered from this point of view. Standard protection methods have been designed for non-temporal attributes and they disregard many key questions regarding time series as *e.g.* time series normalization or preservation of time information. In general, methods ignore the standard uses specific for time series as *e.g.* forecasting or tendency analysis.

In this section, we present a method for time series protection. It is a method based on MDAV microaggregation. Recall that MDAV (and microaggregation in general) requires the definition of a distance on the data. For standard data, the usual distance is the Euclidean one. In the case of time series, several distances on time series can be considered. Here, we propose two different distances: short time series distance and Euclidean distance (both described in Section 2.2.1).

6.1.1 Time Series Microaggregation

To specialize the MDAV algorithm for time series we need to establish which distance and which average function will be used. We propose to implement the general MDAV algorithm described in Algorithm 3 (Section 2.4.2) with the following parameterizations:

- **Distance functions.** We propose the use of Euclidean and STS distances: $d_{EU}(x,v)$ and $d_{STS}(x,v)$ as defined in Section 2.2.1.
- **Average.** We propose to use a kind of arithmetic mean. Such mean has been defined component-wise. That is, given the set $X = \{x^j\}_{j=1,\ldots,J}$ with time series x^j for $j = 1,\ldots,J$, each one with x^j_k, we define the average series $\tilde{x}_k$ by $\tilde{x}_k = (1/J)\sum_{j=1,\ldots,J} x^j_k$.

With these definitions, the average record $\tilde{x}$ in the MDAV algorithm is the average of all records (time series) in X.

The two distance functions considered (Euclidean and STS distances) lead to different results when combined with the microaggregation algorithm. While the Euclidean distance makes clusters based on the distance between data components,

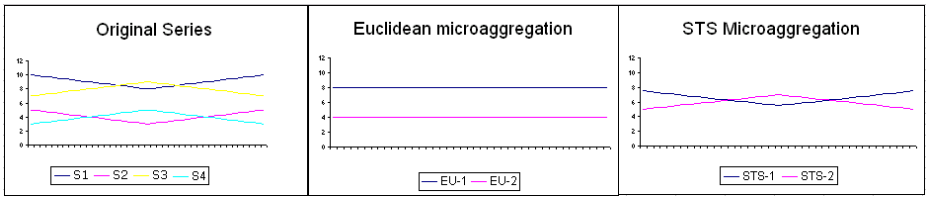


Figure 6.1: Graphical representation of distance function selection.

the STS distance makes clusters based on the shape of the time series. This is illustrated in the following example.

> **Example 6.1** *Figure 6.1 (left) represents 4 series to be microaggregated. The results of microaggregating these 4 series into 2 clusters using either Euclidean or STS distances are given, respectively, in middle and right chart of Figure 6.1. It can be observed that the Euclidean distance gathers together the nearest series even in the case that they have different shapes (and, thus, the outcomes are just lines but that mainly keep the original values). In contrast, the STS distance gathers series according to shapes (and, thus, the outcomes keep such shapes but not the original position of the series).*
> *In this example, we have used point-wise average for computing the representative of each cluster.*

According to this, in the step of selecting the distance function, we have the opportunity to model how the microaggregation procedure makes the clusters and decide which information is the most important to be kept in the final protected model.

In the following we will use eu-microaggregation to denote the microaggregation based on the Euclidean distance and sts-microaggregation to denote the microaggregation based on the STS distance.

6.2 Time Series Information Loss Measures

Strictly speaking, information loss depends on the data uses to be supported by the protected data. However, potential data uses are very diverse and it could be even hard to identify them all at the moment of data release. It is thus desirable for the data protector to be able to measure information loss in a generic way. Informa-

tion loss measures should reflect how much perturbation is added by a given protection method. The amount of information loss measured in this generic way should roughly correspond to the amount of information loss for a reasonable range of data uses. When one defines the measure from a set of components, we need such components to cover (almost) all the possible data uses of a generic user.

In our scenario for time series protection, information loss components have to cover a broad variety of uses, ranging from the statistical analysis to forecasting. For this reason, we divide the information loss components into three different categories:

- IL_1. Measures related to statistical analysis. Such measures, as the average or the autocorrelation function, cover part of the typical statistical analysis like ARMA or ARIMA processes [9].
- IL_2. Measures related to the differences among original and protected time series. It is clear that information loss increases if protected elements are 'far' (dissimilar) to the original ones.
- IL_3. Measures related to forecasting. As forecasting is one of the most common uses of time series, we can say that the statistical information is preserved when the forecast from protected time series is similar to the forecast using the original data.

6.2.1 Information Loss Computation

We have defined the general information loss in terms of the three different components described above IL_1, IL_2 and IL_3:

$$IL = \frac{IL_1 + IL_2 + IL_3}{3}$$

We formally define IL_1, IL_2 and IL_3 below. These three measures are calculated using the differences between values obtained from the original and the protected data. It is possible to define such differences in different ways (*e.g.* by using the mean square error or the mean absolute error). However, as we want to obtain a value in the $[0,1]$ interval, we define the IL_i measures as mean variations dividing

the differences by the largest value (original or protected) to ensure that the result is always inside the rank $[0,1]$. We denote γ for the original statistic value and γ' for the same statistic computed in the protected data set.

- IL_1. It is defined as the average of the difference between the time series means and the autocorrelation functions of both original and protected time series. Formally, IL_1 is computed using the formula

$$IL_1 = \frac{IL_{1.1} + IL_{1.2}}{2}$$

where $IL_{1.1}$ and $IL_{1.2}$ correspond, respectively, to

$$IL_{1.1} = \frac{\sum_{i=1}^{s} \frac{(|\mu_i| - |\mu_i'|)}{Max(|\mu_i|,|\mu_i'|)}}{s}$$

$$IL_{1.2} = \frac{1}{4} \sum_{i=0,n/4,n/2,3n/4} \left(\frac{\sum_{i=1}^{s} \frac{|(R_i| - |R_i')|)}{Max(|R_i|,|R_i'|)}}{s}\right)$$

where s is the number of series in the data set and n is the number of elements (length) of the time series.

- IL_2. It is defined in terms of the absolute differences between original and protected time series elements

$$IL_2 = \frac{\sum_{i=1}^{s \times n} \frac{|x_i - x_i'|}{Max(|x_i|,|x_i'|)}}{s \times n}$$

- IL_3. It is defined using the differences between different forecasting models for the $n+1$, $n+2$ and $n+3$ values

$$IL_3 = \frac{\sum_{m \in FM} \frac{\sum_{i=1}^{3} \frac{|x_{n+i} - x'_{n+i}|}{Max(x_{n+i}, x'_{n+i})}}{3}}{5}$$

where FM is the set containing all the forecasting models described in Section 2.2.2. Then, FM is defined as $FM = \{SESF, DESF, RF, MLRF, PRF\}$.

6.3 Time Series Disclosure Risk Measures

Section 6.2 discusses ways to measure the information loss caused by protection methods for time series. However, as we have explained in the preliminaries, the assessment of the quality of a protection method should not be restricted to its information loss but it should also include a measure of its disclosure risk.

Following the scenario described in Section 2.4, once the modified (protected) data set X' is released, everybody can see its content. This scenario also assumes that the intruder has access to some other data set $Y = Y_{id}||Y_{nc}$ which includes an identifier and some of the non-confidential quasi-identifier attributes of some of the individuals whose data is in X'. Then, according to this scenario, disclosure risk measures have to be in accordance with the difficulty for an intruder of linking the protected data X' with the original data Y. To do this, in this thesis we propose to modify the distance based record linkage method presented in Section 2.3.1 for time series re-identification.

We also measure the disclosure risk in the scenario where the intruder has no access to an external data set. In this case, we assume that the intruder tries to infer the original values from the protected ones. We model this situation using the *interval disclosure*. In this approach an interval is considered around each protected value. Then, when one original value falls within the interval defined around the corresponding protected value, we assume that the intruder obtains a value of enough quality to break the privacy of the data respondent.

6.3.1 Time Series Normalization

It is usual to normalize data sets before applying record linkage methods. This is so, to avoid the scale problems of raw data. The following two alternatives are usually considered:

- **Ranging.** Raw data is translated into the $[0,1]$ interval using this expression $x' = \frac{(x-\min(a))}{(\max(a)-\min(a))}$, where x is the original value and $\max(a)$ and $\min(a)$ are the maximum and minimum values for the corresponding attribute a.

	Index of prices								
	1993	1994	1995	1996	1997	1998	1999	2000	2001
Bread	106.5	110.3	114.9	117.9	119.3	121	122.2	124.1	129
Oil	102.7	119.8	147.8	178.7	130.8	116.2	133.6	123.5	114.4
Vegetables	95.6	101.9	110.8	116.4	114.2	119	124.6	126.4	133.9
Potatoes	101.1	133.6	162.8	123.8	121.3	140.4	149.8	148.6	177.6

Table 6.1: Data extracted from Spanish National Statistics Institute.

- **Standardization.** Raw data is normalized by translating the mean to be equal zero and the standard deviation to be equal one. That is, $x' = \frac{(x-\mu_a)}{S_a}$, where μ_a and S_a are, respectively, the mean and the standard deviation of the corresponding attribute a.

This kind of pre-processing, when applied independently for each component of the time series, causes the loss of the temporal information of the time series. For this reason, we apply another type of normalization using all the elements included into the time series. In this work we had used the following normalization

$$x'_i = \frac{(x_i - \mu_x)}{S_x}$$

where μ_x and S_x are the mean and the standard deviation of the elements of the corresponding time series.

Now, we illustrate with a clear example (that uses the index prices for some food products) the impact of the normalization of the time series, comparing the normalization by component (each component treated as an attribute) and the normalization of the time series as a whole. The example illustrates that the normalization by component distorts completely the shape of the time series.

	Index of prices								
	1993	1994	1995	1996	1997	1998	1999	2000	2001
Bread	1.00	0.26	0.08	0.02	0.31	0.20	0.00	0.02	0.23
Oil	0.65	0.56	0.71	1.00	1.00	0.00	0.41	0.00	0.00
Vegetables	0.00	0.00	0.00	0.00	0.00	0.12	0.09	0.12	0.31
Potatoes	0.50	1.00	1.00	0.12	0.43	1.00	1.00	1.00	1.00

Table 6.2: Data normalized with the standard component-wise procedure.

	Index of prices								
	1993	1994	1995	1996	1997	1998	1999	2000	2001
Bread	106.5	110.3	114.9	117.9	119.3	121	122.2	124.1	129
Oil	102.7	119.8	147.8	178.7	130.8	116.2	133.6	123.5	114.4
Vegetables	95.6	101.9	110.8	116.4	114.2	119	124.6	126.4	133.9
Potatoes	101.1	133.6	162.8	123.8	121.3	140.4	149.8	148.6	177.6

Table 6.3: Data normalized with the time series procedure.

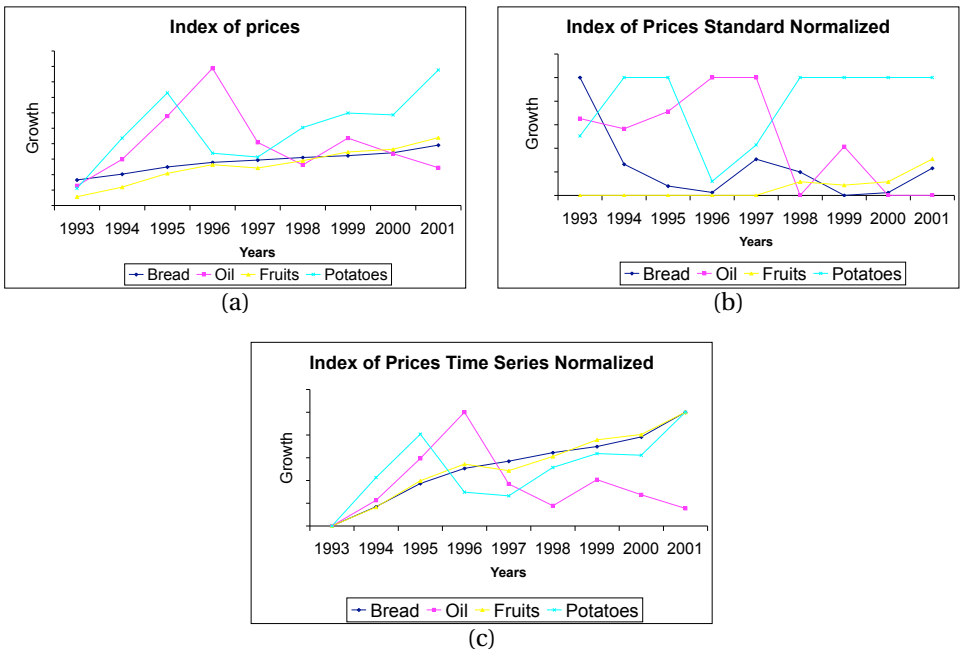


Figure 6.2: Graphical representation of the effects of time series normalization, (a) represents the original data without normalization, (b) represents normalized data with independent normalization, (c) represents normalized data with time series normalization.

Example 6.2 *Let us consider the price index of four different foods in nine years. We can observe in Table 6.1 the original raw values and their tendency in the period 1993 - 2001 and in Tables 6.2 and 6.3, respectively, the normalized data values after standard (component-wise) and time series (data altogether) normalization. Figure 6.2 shows that different normalizations produce different outcomes and that the standard component-wise normalization causes important divergences on the tendency of the time series between the original time series and the normalized one. For example, in the case of bread, when comparing charts (a) and (b), we observe that in the original data bread price tendency was to increase every year but that after normalization bread price has a decreasing tendency. This is a negative effect of the normalization over the data.*

To avoid this effect of component-wise normalization, we propose the use of specific normalization procedures for time series: normalization of all the series.

6.3.2 Time Series Re-identification

The time series record linkage presented in this section is based on the standard distance based record linkage. Recall that DB-RL method can be applied when a distance between pairs of records (one in the original data set and the other in the protected data set) can be defined. Then, every protected record is linked to the closest original one. When the data is numerical (DB-RL standard), it is usual to use the Euclidean distance (after normalizing the whole data set). In our case with time series, we use the normalization explained above and the distances presented in Section 2.2.1. That is, the Euclidean distance and the STS distance. Formally, time series record linkage is described in Algorithm 7, where $d_{ts}(a,b)$ is defined in terms of a given distance d_{x_i} for each time series x_i.

6.3.3 Time Series Interval Disclosure

When the intruder has no access to any external data source, he can try to approximate original values assuming that they are in a finite interval around the protected value. To measure the risk of this approach, we apply the Algorithm 8 where p is a parameter defined by the user and $|E|$ is the number of values in the entire data set.

Algorithm 7: Time Series Record Linkage

```
Data: X: original data set, X': protected data set
Result: LP: linked pairs
1 begin
2     Apply time series normalization to X and X'
3     foreach a ∈ X do
4         b' = arg_min_{b∈X'} d_ts(a, b)
5         LP = LP ∪ (a, b')
6         foreach a ∈ X do
7             NP = NP ∪ (a, b)
8 end
```

Algorithm 8: Time Series Interval Disclosure

```
Data: X: original data set, p: interval size
Result: c: percentage of elements revealed
1 begin
2     foreach record r ∈ X do
3         foreach time series t ∈ r do
4             foreach element x ∈ t do
5                 r = p × x'
6                 if (x ≥ x' − r) and (x ≤ x' + r)) then
7                     c = c + 1
8 end
```

Normally, the parameter p is defined using a percentage of difference of an element. For example, with $p = 10\%$, if the element is equal to 10, the corresponding interval will be $[9, 11]$.

6.3.4 The Computation of the DR Measures

Considering the two scenarios presented above, it is possible to compute the final disclosure risk as:

$$DR = \frac{DR_1 + DR_2}{2}$$

where DR_1 and DR_2 summarize the re-identification risk and the interval disclosure

risk respectively.

DR_1 is computed averaging the percentage of records correctly linked by the intruder using different time series distances. In our case we consider $EULD$ (Euclidean distance linkage disclosure) and $STSLD$ (Short time series linkage disclosure). Formally, we compute DR_1 using the formula

$$DR_1 = \frac{EULD + STSLD}{2}$$

where $EULD$ and $STSLD$ are the average percentage of records correctly linked using time series record linkage with Euclidean and STS distance when the intruder knows different numbers of time series (from 1 to all).

DR_2 is computed as the interval disclosure using different values for the parameter p, in our case p ranges from 1% to 10%

$$DR_2 = \frac{\sum_{p=0.01}^{0.1} ID_p}{10}$$

6.4 Final Trade-off Evaluation

As we said in the preliminaries, information loss and disclosure risk have to be combined to obtain a global value about the performance of a specific protection method. This value weighs the relationship between information loss and disclosure risk. To do this, we follow the definition of the *score* presented in Section 2.5. Then, the final evaluation of a time series protection method is as follows:

$$score = \frac{IL + DR}{2}$$

where IL is the overall information loss measure and DR is the overall disclosure risk measure.

6.5 Experiments

As stated above, we have introduced a new data protection method for time series, and we have also presented a framework to evaluate time series anonymization methods. In this section, we describe some experiments done with real data using the time series microaggregation protection method. These experiments show how our framework works.

6.5.1 Data Protection

To analyse empirically our framework and to evaluate the time series microaggregation method we have protected some real data sets that can be obtained freely from different data sources. Firstly, we have used a file from [38] (the so-called forecasters) with 3003 time series of different lengths (between 14 and 64 elements). We have re-sampled all time series to 10 elements to covert them into the same length. Secondly, we have used the Stock Exchange information of the thirty five most important Spanish companies. These companies are ranked in the so-called Ibex35 stock market. We have downloaded the information about prices from June, 21st 2005 to April, 28 th 2006 from [56]. And finally, we have used data information about all football teams of the nine most important European domestic leagues from [33]. As said above, the information about these three testbeds is publicly available. Data details are given in Table 6.4.

We have protected the original data with the time series microaggregation method described in Section 6.1. We have applied this method with $k \in \{2,3,6,9,12\}$.

We have applied the time series microaggregation method splitting the original time series into n masked ones to obtain a larger variety of tests. We detail now these conversions for each file.

- **Forecasters problem.** We have split the original time series into $n \in \{1,2\}$ time series. So, in this case we have two different data sets, one with one time series and the other with two time series.
- **Ibex35 problem.** We have split the original time series into $n \in \{2,4,20\}$ time series. So, in this case, we have three different data sets with $4,8$ and 40 time

Forecasters	Records	3003
	Number of time series	1
	Time series length	10
	Number of records	10
	Series description	Financial information
Ibex35	Records	35
	Number of time series	2
	Time series length	220
	Number of records	440
	Series description	Financial information, Volume transactions
football	Records	176
	Number of time series	8
	Time series length	25
	Number of records	200
	Series description	Years, FIFA points, Leage position, Goals for Goals against, Matchs win, Matchs dice, Matchs loose

Table 6.4: Details of time series examples.

series.

- **football problem.** In this case, no conversion is done because the original data set already consisted on eight time series.

6.5.2 Results

Tables 6.5, 6.6 and 6.7 present the score and its components for the forecaster, football and ibex35 data set respectively. Columns one to three present the IL_i components and column four shows the overall IL value. From these columns we can infer that IL increases when k increases. *E.g.* in the forecaster problem protected with eu-microaggregation, IL values range from 6.78 to 15.89. This behavior is consistent with the usual results for general microaggregation methods.

We can also infer from IL_1 (column one) that time series microaggregation preserve the time series mean and autocorrelation function. See, for example, the forecasters data set in Table 6.5, where for all the microaggregation configurations, IL_1 is always 0.00. It is known that general microaggregation preserves the average when applied to numerical attributes. Therefore, it is not surprising that time series microaggregation also preserves time series mean, when applied to time series.

	i	k	IL_1	IL_2	IL_3	IL	$EULD$	$STSLD$	ID	DR	$score$
forecasters.i.eu-k	1	2	0.00	6.33	6.78	4.37	42.79	33.37	40.32	39.20	21.79
	1	3	0.00	8.32	9.27	5.86	25.67	17.32	39.20	30.35	18.11
	1	6	0.00	11.00	12.56	7.85	10.62	6.03	37.63	22.98	15.41
	1	9	0.00	12.57	14.20	8.92	7.19	3.33	36.52	20.89	14.91
	1	12	0.00	13.63	15.89	9.84	5.49	2.56	35.80	19.92	14.88
	2	2	0.00	26.27	29.44	18.57	28.37	22.56	28.41	26.94	22.75
	2	3	0.00	27.76	31.84	19.87	15.55	11.72	26.43	20.04	19.95
	2	6	0.00	26.12	32.30	19.47	7.74	5.11	26.42	16.43	17.95
	2	9	0.00	25.09	31.74	18.95	6.44	3.50	26.97	15.97	17.46
	2	12	0.00	24.41	30.34	18.25	5.44	2.56	27.76	15.88	17.07
forecasters.i.sts-k	1	2	0.00	10.16	7.88	6.01	30.67	42.12	38.14	37.27	21.64
	1	3	0.00	12.72	10.10	7.61	17.65	25.61	36.60	29.11	18.36
	1	6	0.00	16.53	13.39	9.97	7.16	10.92	34.05	21.55	15.76
	1	9	0.00	18.51	15.46	11.32	4.83	7.49	32.70	19.43	15.38
	1	12	0.00	20.41	17.10	12.50	3.26	5.19	31.39	17.81	15.16
	2	2	0.00	23.69	27.63	17.11	20.63	30.82	29.63	27.68	22.39
	2	3	0.00	26.01	29.40	18.47	11.39	17.62	27.80	21.15	19.81
	2	6	0.00	28.68	31.31	19.99	5.81	8.04	26.05	16.49	18.24
	2	9	0.00	30.30	32.78	21.03	3.70	5.61	24.93	14.79	17.91
	2	12	0.00	31.69	34.17	21.95	2.86	4.56	24.19	13.95	17.95

Table 6.5: Score and its components in the forecasters data set. Forecaters.i.d-k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k.

Comparing eu-microaggregation and sts-microaggregation with the same k and number of series, it can be observed that (in general) IL is lower for the eu-microaggregation. However, in a few cases, sts-microaggregation obtains a lower IL. For instance, in the forecasters data set with two time series and $k = 2$, IL for eu-microaggregation is equal to 18.57 while for sts-microaggregation is equal to 17.11.

Columns five and six present the $EULD$ and $STSLD$. From these two columns it is clear that re-identification risk decreases when k increases. The same happens with ID and the overall DR (columns seven and eight). Then, we can say that parameter k is inversely proportional to disclosure risk.

In general, the greatest re-identification risk for a given microaggregation (eu and sts) occurs when the same distance is used in the time series record linkage. For instance, in the football data set configurations with $k = 6$, $EULD$ for eu-microaggregation is 65.54 while $STSLD$ is 54.26. In contrast, using sts-

	i	k	IL_1	IL_2	IL_3	IL	$EULD$	$STSLD$	ID	DR	$score$
football.i.eu-k	8	2	0.11	44.55	44.25	29.63	84.16	84.37	19.94	52.11	40.87
	8	3	0.15	45.38	45.51	30.35	78.91	75.14	19.73	48.38	39.36
	8	6	0.17	45.39	45.75	30.44	65.27	54.26	19.49	39.63	35.03
	8	9	0.40	45.44	46.37	30.74	54.97	34.38	19.61	32.14	31.44
	8	12	0.22	45.43	45.20	30.28	50.28	27.91	19.44	29.27	29.77
football.i.sts-k	8	2	0.10	46.66	45.41	30.72	71.66	83.59	19.74	48.68	39.70
	8	3	0.16	49.08	48.43	32.56	56.82	77.77	17.68	42.49	37.52
	8	6	0.25	50.84	50.13	33.74	31.75	58.17	15.79	30.37	32.06
	8	9	0.31	51.90	48.94	33.72	22.23	41.12	14.82	23.25	28.48
	8	12	0.34	52.63	49.79	34.25	14.35	33.95	14.74	19.44	26.85

Table 6.6: Score and its components in the football data set. football.i.d.k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k.

microaggregation the largest re-identification risk is $STSLD$ (58.17).

If one compares the ID of both microaggregation methods, in general sts-microaggregation achieves lower values. For instance, comparing in the football data set both microaggregation methods with $k = 12$, eu-microaggregation obtains 19.44 while sts-microaggregation only 14.74.

The last column of each table shows the overall score. It can be observed that the score is very data set dependent. However, (in general) with small values of k the best scores are obtained by sts-microaggregation (*e.g.* ibex35.20.eu.2 is equal to 31.62 and ibex35.20.sts.2 is equal to 27.67). On the other hand, with large values of k the best scores are obtained by eu-microaggregation (*e.g.* ibex35.20.eu.12 is equal to 21.02 and ibex35.20.sts.12 is equal to 23.17).

	i	k	IL_1	IL_2	IL_3	IL	$EULD$	$STSLD$	ID	DR	$score$
ibex35.i.eu-k	2	2	0.00	29.82	31.51	20.44	57.14	52.86	13.54	34.27	27.36
	2	3	0.00	36.82	40.05	25.62	35.71	35.71	10.57	23.14	24.38
	2	6	0.01	43.61	44.18	29.27	14.29	18.57	8.69	12.56	20.91
	2	9	0.02	44.40	52.81	32.41	5.71	10.00	7.47	7.67	20.04
	2	12	0.03	49.34	52.54	33.97	7.14	4.29	6.74	6.23	20.10
	4	2	0.00	31.85	32.12	21.32	60.00	60.71	12.50	36.43	28.88
	4	3	0.00	37.44	37.28	24.91	45.71	42.14	10.06	27.00	25.95
	4	6	0.01	42.61	44.96	29.19	21.43	22.86	8.83	15.48	22.34
	4	9	0.01	44.29	47.63	30.64	11.43	14.29	7.14	10.00	20.32
	4	12	0.01	49.90	53.63	34.51	8.57	7.14	6.78	7.32	20.92
	20	2	0.00	32.78	35.27	22.68	67.86	70.57	11.89	40.55	31.62
	20	3	0.00	37.71	41.09	26.27	48.71	48.14	9.76	29.10	27.68
	20	6	0.00	42.73	45.59	29.44	26.43	24.71	8.72	17.14	23.29
	20	9	0.00	44.97	49.81	31.59	12.29	16.71	6.65	10.58	21.08
	20	12	0.00	50.63	52.83	34.49	8.29	7.71	7.09	7.55	21.02
ibex35.i.sts-k	2	2	0.02	45.94	47.57	31.18	20.00	51.43	6.81	21.26	26.22
	2	3	0.04	47.87	49.53	32.48	15.71	28.57	6.13	14.14	23.31
	2	6	0.07	58.98	62.03	40.36	10.00	11.43	4.83	7.77	24.07
	2	9	0.18	57.09	64.54	40.60	1.43	2.86	3.56	2.85	21.73
	2	12	0.17	58.04	61.52	39.91	4.29	2.86	5.77	4.67	22.29
	4	2	0.02	45.61	45.72	30.45	20.71	52.86	7.94	22.36	26.41
	4	3	0.03	51.69	50.39	34.04	15.00	34.29	4.94	14.79	24.41
	4	6	0.07	56.50	57.91	38.16	10.00	16.43	5.08	9.15	23.65
	4	9	0.08	57.44	57.11	38.21	7.86	7.86	4.75	6.30	22.26
	4	12	0.13	57.88	57.72	38.58	5.71	4.29	5.16	5.08	21.83
	20	2	0.00	46.47	47.28	31.25	19.57	60.29	8.24	24.09	27.67
	2	3	0.01	51.55	53.66	35.08	14.43	43.43	5.55	17.24	26.16
	2	6	0.02	57.51	59.55	39.03	11.43	22.43	4.19	10.56	24.79
	2	9	0.03	57.06	60.42	39.17	7.86	13.86	4.70	7.78	23.47
	2	12	0.02	57.25	60.06	39.11	6.00	12.00	5.46	7.23	23.17

Table 6.7: Score and its components in the ibex35 data set. ibex35.i.d.k corresponds to microaggregation using distance d (Euclidean or STS) with i series and parameter k.

Chapter 7

Conclusions and Future Directions

Along the preceding chapters we have presented several contributions to disclosure risk assessment. Now, in the first section of this chapter we review these contributions. Afterwards, we will explain some conclusions obtained from the work presented in this thesis. Finally, in the last section, we sketch future research lines.

7.1 Summary of Contributions

In this thesis we have proposed different ways to calculate disclosure risk of a protection method. In what follows, we review each contribution shortly, summarizing its relevance.

- **Microaggregation contributions.** Firstly, we have defined an empirical disclosure risk measure for multivariate microaggregation and provided a theoretical limit for such measure. Secondly, we have described different techniques for attribute selection in microaggregation, studying in detail their consequences for the disclosure risk using real data sets. Finally, we have explained two new variants of microaggregation, the first one uses aggregation

functions to replace the traditional projected methods in projected microaggregation, whereas the second one solves the problem of attribute selection in multivariate microaggregation.

- **Ad-hoc methods for risk assessment.** We have defined three specific record linkage methods which take into account the protection method applied to the protected data set. The direct consequence of these definitions is that such methods achieve a larger number of re-identifications than generic record linkage ones. Therefore, the disclosure risk increases when using the analyzed methods. Another advantage of two of these new methods (namely, RS-RL and A-RL) is that an intruder using them is sure (in certain cases) that the linkages obtained are correct. This fact never happens with generic record linkage methods.

- **Record linkage using fuzzy integrals.** We studied a method for record linkage when data sets do not share attributes. An exhaustive testing has been carried out to evaluate its performance. Results show that the re-identification is still possible in this scenario.

- **Time Series.** We have presented a new framework for evaluating time series protection methods. We have introduced some information loss measures and disclosure risk measures for time series which cover all their common uses. We have also presented some results analysing an extension of microaggregation for time series.

7.2 Conclusions

In this thesis we have covered different aspects in the field of statistical disclosure control. Most of our attention has been devoted to the accuracy of disclosure risk assessment for certain well-known anonymization methods. From the results presented in this thesis, we can state that, protection methods have to be designed to ensure that an intruder cannot perform specific attacks to break the privacy of the respondents.

The privacy of the respondents is sometimes not regarded as important as information loss when evaluating protection methods. The main reason for this is that it

is often difficult for an intruder to obtain a file with the same (anonymized) quasi-identifiers from another data source. However, as we have shown in this thesis, in many cases re-identification is still possible when the intruder has access to a different set of quasi-identifiers. Therefore, in our opinion, a correct protection method evaluation has to weigh disclosure risk and information loss with the same importance. We would like to point out that disclosure risk measures have to cover as many as possible different disclosure risk scenarios.

A lot of research done in statistical disclosure control is related to protection methods that, in some way, ensure k-anonymity. However, as we have seen in Chapter 3, many times when such methods are used to protect real data by statistical agencies, the theoretical k-anonymity is not preserved. Particularly in microaggregation, the k-anonymity property is not preserved so that higher values of data utility can be obtained. If this is the case, privacy of the respondents is disregarded against the interest of the data users. Then, statistical agencies have to ensure with *a posteriori* measures that at least k'-anonymity (where $0 \leq k' \leq k$) is preserved.

Finally, we would like to highlight that intruders exploit whatever weakness they detect to achieve their goal. For this reason, statistical agencies must study countermeasures to avoid that intruders find and exploit such weaknesses. In Chapter 6, we have described some modifications for distance based record linkage to increase the number of links when the intruder fuses several data sets released at different times. Then, if the statistical agency knows in advance that a data set will be released repeatedly, it has to anonymize such data with a protection method prepared to avoid this attack.

7.3 Future Directions

Along the different topics explained in this dissertation, there are certain facets which are still open. Now, we sketch some ideas to continue our research.

- **Attribute disclosure risk evaluation.** This thesis is devoted to individual re-identification, however, in some occasions, the intruder is not able to infer which protected record belongs to one individual, but he is able to infer that a respondent belongs to a certain group of records. Then, if (almost) all the

records of this group have the same value for a given confidential attribute, the intruder is able to infer the confidential value for the respondent. In this case, even though the intruder does not obtain the correct linkage, he obtains the confidential value. We would like to develop protection methods which avoid both problems, *i.e.*, individual and attribute re-identification.

- **Probabilistic record linkage with conditional probabilities.** As we have explained in the preliminaries, probabilistic record linkage assumes that attributes in the data set are independent, this assumption makes easier the computation of indexes in the expectation-maximization algorithm. However, in the real world, attributes are not independent. For this reason we plan to develop a new record linkage method based on conditional probabilities. Our intuition says that this new record linkage will achieve a larger amount of correct linkages than traditional probabilistic record linkage.

- **Supervised re-identification.** We are interested in the study of alternative methods for record linkage based on supervised machine learning techniques as neural networks. For simplicity we will assume that only two data sets A and B are considered. Our idea will work as follows. Firstly, a model between the attributes of A and B is built. In this way, it is later possible to translate the values on the domain of A into values on the domain of B. Then, after such translation, re-identification is possible using classical record linkage. This is done using the new translated data set, say A', and the original data set B. We will consider that the construction of such model is done in a supervised way. That is, we will consider that there is a set of records of both data sets A and B for which we will know the correct re-identifications. Such records will be used to build the model between the two data sets.

Our Contributions

[1a] P. Medrano-Gracia, J. Pont-Tuset, J. Nin and V. Muntés-Mulero, Ordered data set vectorization for linear regression on data privacy. In *Modeling Decisions for Artificial Intelligence (MDAI)*, volume 4617 of *Lecture Notes in Artificial Intelligence*, pages 361-372. Springer, 2007.
(acceptance rate, 21.7%).

[2a] J. Nin and V. Torra, Towards the use of OWA operators for record linkage. In *Proceedings of the European Society on Fuzzy Logic and Technologies (EUSFLAT)*, ISBN: 84-7653-872-3, pages 34-39, 2005

[3a] J. Nin and V. Torra, Empirical analysis of database privacy using twofold integrals. In *Computational Intelligence and Security (CIS)*, volume 3801 of *Lecture Notes in Artificial Intelligence*, pages 1 - 8. Springer, 2005.
(acceptance rate, 19.89%).

[4a] J. Nin and V. Torra, New approach to re-identification problem using neural networks. In *Modeling Decisions for Artificial Intelligence (MDAI)*, volume 3885 of *Lecture Notes in Artificial Intelligence*, pages 356-267. Springer, 2006.
(acceptance rate, 31.9%).

[5a] J. Nin and V. Torra, Fuzzy measures and integrals in re-identification problems. In *Proceedings Of International Symposium on Nonlinear Theory and its Applications (NOLTA)*, CD-ROM, 2006.

[6a] J. Nin and V. Torra, Extending microaggregation procedures for time series protection. In *Rough Set and Soft Computation Society (RSCTC)*, volume 4259 of *Lecture Notes in Computer Science*, pages 899-908. Springer, 2006.
(acceptance rate, 27.4%).

[7a] J. Nin and V. Torra, Distance based re-identification for time series, Analysis of distances. In *Privacy in Statistical Databases (PSD)*, volume 4302 of *Lecture Notes in Computer Science*, pages 205-216. Springer, 2006.

[8a] J. Nin and V. Torra, Blocking anonymized data. In *Proceedings of the 4th International Summer School on Aggregation Operators (AGOP)*, ISBN: 978-90-382-1140-4, pages 83-87, 2007.

[9a] J. Nin, J. Pont-Tuset, P. Medrano-Gracia, J. Larriba-Pey and V. Muntés-Mulero, Anonymizing Data via polynomial regression. In *Proceedings of the Computer Science Spanish conference (CEDI)*. II Simposio sobre seguridad informática, ISBN: 978-84-9732-607-0, pages 19-26, 2007.

[10a] J. Nin, V. Muntés-Mulero, N. Martínez-Bazán and J. Larriba-Pey, Semantic blocking for record linkage. In *Proceedings of the International conference of catalan artificial intelligence society (CCIA)*, ISBN: 0922-6389, pages 141-149, 2007.

[11a] J. Nin, V. Muntés-Mulero, N. Martínez-Bazán and J. Larriba-Pey, On the use of semantic blocking techniques for data cleansing and integration. In *Proceedings of IEEE Eleventh International Database Engineering and Applications Symposium (IDEAS)*, ISBN: 0-7695-2947-X, pages 190-198, 2007.
(acceptance rate, 31.5%).

[12a] J. Nin, J. Pont-Tuset, P. Medrano-Gracia, J. Larriba-Pey and V. Muntés-Mulero, Increasing polynomial regression complexity for data anonymization. In *Proceedings of IEEE International Conference on Intelligent Pervasive Computing (IPC)*, ISBN: 0-7695-3006-0, pages 29-34, 2007.

[13a] J. Nin, J. Herranz and V. Torra, Rethinking rank swapping to decrease disclosure risk. *Data and Knowledge Engineering*, 64(1): 346-364. Elsevier, 2008.
SCI index (2006):1.367.

[14a] J. Nin, J. Herranz and V. Torra, On method-specific record linkage for risk assessment, In *Proceedings of the UNECE Work Sesison on Statistical Confidentiality*, 2007.

[15a] J. Nin, J. Herranz and V. Torra, Attribute selection in multivariate microaggregation. In *Post-Proceedings of 11th ACM International Conference on Extending Database Technology (EDBT)*, 2008.

[16a] J. Nin, J. Herranz and V. Torra, How to group attributes in multivariate microaggregation. *International Journal on Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS)*, 16(1):121-138, World Scientific publishing, 2008.
SCI index (2006):0.406.

[17a] J. Nin and V. Torra, Modeling projections in microaggregation. In *Proceedings of 12th International Conference on Information Processing and Management of Uncertainly in Knowledge-Based Systems (IPMU)*, in press, 2008.

[18a] J. Nin and V. Torra, Analysis of the univariate microaggregation disclosure risk, submitted to *New Generation Computing*.

[19a] J. Nin and V. Torra, Towards the evaluation of time series protection methods, submitted to *Information Science*.

[20a] J. Nin, J. Herranz and V. Torra, On the disclosure risk of multivariate microaggregation. submitted to *Data and Knowledge Engineering*.

[21a] J. Pont-Tuset, P. Medrano-Gracia, J. Nin, J. Larriba-Pey and V. Muntés-Mulero, ONN the use of neural networks for data privacy, In *Current Trends in Theory and Practice of Computer Science (SOFSEM)*, volume 4910 of *Lecture Notes in Computer Science*, pages 634-645. Springer, 2008.
(acceptance rate, 35.1%).

[22a] J. Pont-Tuset, J. Nin, P. Medrano-Gracia, V. Muntés-Mulero and J. Larriba-Pey, A new methodology for numerical microdata anonymization. In *Advances in Artificial Intelligence for Privacy Protection and Security*, World Scientific publishing, in press, 2008.

[23a] V. Torra and J. Nin, Record linkage for database integration using fuzzy integrals. *International Journal of Intelligent Systems*, 23(6):715-734, Wiley Editors, 2008.
SCI index (2006):0.429.

Other References

[1] N. R. Adam and J. C. Worthmann. Security-control methods for statistical databases: a comparative study. *ACM Computing Surveys*, 21(4):515–556, 1989.

[2] C. Aggarwal. On k-anonymity and the curse of dimensionality. In *Proceedings of the 31st International Conference on Very Large Databases*, pages 901–909, 2005.

[3] G. Aggarwal, T. Feder, K. Kenthapadi, S.Khuller, R. Panigrahy, D. Thomas, and A. Zhu. Achieving anonymity via clustering. In *Proceedings of the 25th ACM Symposium on Principles of Databases Systems*, pages 153–162, 2006.

[4] E. Anderson. The irises of the gaspe peninsula. *Bulletin of American Iris Society*, 59:2–5, 1935.

[5] J. Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*. Norwell, Massachusetts: Kluwer Academic Publishers, 2001.

[6] M. Atencia and M. Schorlemmer. A formal model for situated semantic alignment. In *Proceedings of the 6th International Conference in Agent and Multiagent Systems*, 2007.

[7] E. Bertino, I. N. Fovino, and L. P. Provenza. A framework for evaluating privacy preserving data mining algorithms. *Data Mining and Knowledge Discovery*, 11(2):121–154, 2005.

[8] M. Bilenko, S. Basu, and M. Sahami. Adaptive product normalization: Using online learning for record linkage in comparison shopping. In *Proceedings of the 5th IEEE International Conference on Data Mining*, pages 58–65, 2005.

[9] G. Box and G. Jenkins. *Time Series Analysis, Forecasting and Control*. Holden-Day, Incorporated, 1990.

[10] R. Brand, J. Domingo-Ferrer, and J. M. Mateo-Sanz. Reference datasets to test and compare sdc methods for protection of numerical microdata. Technical report, European Project IST-2000-25069 CASC, 2002.

[11] P. Brockwell and R. Davis. *Introduction to Time Series and Forecasting*. Springer Texts in Statistics, 2002.

[12] T. Calvo, G. Mayor, and R. Mesiar. *Aggregation Operators*. Physica-Verlag, 2002.

[13] P. Capitani and P. Ciaccia. Efficiently and accurately comparing real-valued data streams. In *Proceedings of the Sistemi Evoluti per Basi di Dati*, pages 161–168, 2005.

[14] G. Choquet. Theory of capacities. *Annales Academiae Scientiarum Fennicae*, 5:131–296, 1954.

[15] S. Chu, E. Keogh, D. Hart, and M. Pazzani. Iterative deepening dynamic time warping for time series. In *The 2nd SIAM International Conference on Data Mining*, 2002.

[16] S. M. Deen, R. R. Amin, and M. C. Taylor. Data integration in distributed databases. *IEEE Transactions on Software Engineering*, 13(7):860–864, 1987.

[17] D. Defays and P. Nanopoulos. Panels of enterprises and confidentiality: The small aggregates method. In *Proceedings of 92th Symposium on Design and Analysis of Longitudinal Surveys*, pages 195–204. Statistics Canada, Ottawa, 1993.

[18] H. H. Do and E. Rahm. COMA - A system for flexible combination of schema matching approaches. In *Proceedings of the 28th Very Large Databases Conference*, pages 610–621, 2002.

[19] J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):189–201, 2002.

[20] J. Domingo-Ferrer and V. Torra. *Disclosure control methods and information loss for microdata*, pages 91–110. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, 2001.

[21] J. Domingo-Ferrer and V. Torra. *A quantitative comparison of disclosure control methods for microdata*, pages 111–133. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, 2001.

[22] J. Domingo-Ferrer and V. Torra. Validating distance-based record linkage with probabilistic record linkage. In *Topics in Artificial Intelligence*, volume 2504 of *Lecture Notes in Computer Science*, pages 207–215, 2002.

[23] J. Domingo-Ferrer and V. Torra. Disclosure risk assessment in statistical microdata protection via advanced record linkage. *Statistics and Computing*, 13:343–354, 2003.

[24] J. Domingo-Ferrer, V. Torra, J. M. Mateo-Sanz, and F. Sebé. Empirical disclosure risk assessment of the IPSO synthetic data generators. In *Proceedings of the UNECE Work Sesison on Statistical Confidentiality*, 2005.

[25] J. Domingo-Ferrer, V. Torra, J. M. Mateo-Sanz, and F. Sebé. Systematic measures of re-identification risk based on the probabilistic links of the partially synthetic data back to the original microdata. Technical report, 2005.

[26] G. T. Duncan, S. E. Fienberg, R. Krishnan, R. Padman, and S. F. Roehrig. *Disclosure Limitation Methods and Information Loss for Tabular Data*, pages 91–110. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, 2001.

[27] G. T. Duncan, S. A. Keller-McNulty, and S. L. Stokes. Disclosure risk vs. data utility: The R-U confidentiality map. Technical Report 21, National Institute of Statistical Sciences, http://www.niss.org/, 2001.

[28] G. T. Duncan, S. A. Keller-McNulty, and S. L. Stokes. Database security and confidentiality: Examining disclosure risk vs data utility through the R-U confidentiality map. Technical Report 142, National Institute of Statistical Sciences, http://www.niss.org/, 2004.

[29] EUROSTAT. Statistical Office of the European Communities.

[30] I. P. Fellegi and A. B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.

[31] F. Felsö, J. Theeuwes, and G. Wagner. *Disclosure Limitation in Use: Results of a Survey*, pages 17–42. Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, 2001.

[32] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[33] Football statistics web. http://www.histora.org/.

[34] B. Fung, K. Wang, and P. Yu. Top-down specialization for information and privacy preservation. In *Proceedings of the 21st IEEE International Conference on Data Engineering*, pages 205–216, 2005.

[35] M. Grabisch, T. Murofushi, and M. Sugeno. *Fuzzy Measures and Integrals: Theory and Applications*. Physica-Verlag, 2000.

[36] S. L. Hansen and S. Mukherjee. A polynomial algorithm for optimal univariate microaggregation. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):1043–1044, 2003.

[37] M. A. Hernández and S. J. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Mining and Knowledge Discovery*, 2(1):9–37, 1998.

[38] Internation Institute of Forecasters. http://www.forecasters.org/.

[39] M. Jaro. Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. *Journal of the American Statistical Association*, 84:414–420, 1989.

[40] J. Kubica, A. Moore, D. Cohn, and J. Schneider. Finding underlying connections: A fast graph-based method for link analysis and collaboration queries. In *Proceedings of the International Conference on Machine Learning*, pages 392–399, 2003.

[41] T. W. Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38:1857–1874, 2005.

[42] J. M. Mateo-Sanz and J. Domingo-Ferrer. A method for data-oriented multivariate microaggregation. In *Statistical Data Protection for Official Publications of the European Communities*, pages 89–99, 1999.

[43] J. M. Mateo-Sanz, J. Domingo-Ferrer, and F. Sebé. Probabilistic information loss measures in confidentiality protection of continuous microdata. *Data Mining and Knowledge Discovery*, 11(2):181–193, 2005.

[44] C. Möller-Levet, F. Klawonn, K. Cho, and O. Wolkenhauer. Fuzzy clustering of short time series and unevenly distributed sampling points. In *International Symposium on Intelligent Data Analysis*, volume 2810 of *Lecture Notes in Computer Science*, pages 330–340. Springer, 2003.

[45] R. A. Moore. Controlled data swapping techniques for masking public use microdata sets. U. S. Bureau of the Census, 1996.

[46] P. Murphy and D. Aha. UCI Repository machine learning databases. *Irvine, CA: University of California, Department of Information and Computer Science*, 1994.

[47] C. Myers and L. Rabiner. A comparative study of several dynamic time-warping algorithms for connected word recognition. *The Bell System Technical Journal*, 60:1389–1409, 1981.

[48] Y. Narukawa and V. Torra. Twofold integral and multi-step choquet integral. *Kybernetika*, 40(1):39–50, 2004.

[49] A. Oganian and J. Domingo-Ferrer. On the complexity of optimal microaggregation for statistical disclosure control. *Statistical Journal United Nations Economic Commission for Europe*, 18(4):345–354, 2000.

[50] D. Pagliuca and G. Seri. Some results of individual ranking method on the system of enterprise accounts annual survey. Technical report, Esprit SDC Project, Deliverable MI-3/D2, 1999.

[51] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *The Very Large Database Journal*, 10(4):334–350, 2001.

[52] E. Rahm and H. H. Do. Data cleaning: problems and current approaches. *IEEE Bulletin of the Technical Committee on Data Engineering*, 23(4):3–13, 2000.

[53] C. A. Ratanamahatana and E. Keogh. Three myths about dynamic time warping data mining. In *SIAM International Conference on Data Mining*, 2005.

[54] N. Rescher. *Predicting the future: An introduction to the theory of forecasting*. State University of New York Press, 1998.

[55] R. Rivest. On estimating the size of a statistical audit. available at `http://people.csail.mit.edu/rivest/publications.html`, November 2006.

[56] Sabadell Bank. Stock exchange web, http://www.bsmarkets.com/.

[57] P. Samatari and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, SRI Intl. Tech. Rep., 1998.

[58] F. Sebé, J. Domingo-Ferrer, J. M. Mateo-Sanz, and V. Torra. Post-masking optimization of the tradeoff between information loss and disclosure risk in masked microdata sets. In *Inference Control in Statistical Databases*, volume 2316 of *Lecture Notes in Computer Science*, pages 187–196. Springer, 2002.

[59] M. Sugeno. *Theory of fuzzy integrals and its application*. PhD thesis, Tokyo Institute of Technology, 1974.

[60] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal Uncertainty Fuzziness Knowledge-Based Systems*, 10(5):571–588, 2002.

[61] L. Sweeney. k-anonymity: a model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, 2002.

[62] V. Torra. Re-identifying individuals using OWA operators. In *Proceedings of the 6th International Conference on Soft Computing*, 2000.

[63] V. Torra. Towards the re-identification of individuals in data files with non-common variables. In *Proceedings of the 14th European Conference on Artificial Intelligence*, pages 326–330, 2000.

[64] V. Torra. On the re-identification of individuals described by means of non-common variables: a first approach. In *Proceedings of the UNECE Work Sesison on Statistical Confidentiality*, 2001.

[65] V. Torra. Twofold integral: A choquet integral and sugeno integral generalization. *Butlletí de l'Associació Catalana d'Intel·ligència Artificial*, 29:13–19 (in

Catalan). Preliminary version: IIIA Research Report TR–2003–08 (in English)., 2003.

[66] V. Torra. Microaggregation for categorical variables: a median based approach. In *Privacy in Statistical Databases*, volume 3050 of *Lecture Notes in Computer Science*, pages 162–174. Springer, 2004.

[67] V. Torra. OWA operators in data modeling and reidentification. *IEEE Transactions on Fuzzy Systems*, 12(5):652–660, 2004.

[68] V. Torra, J. M. Abowd, and J. Domingo-Ferrer. Using mahalanobis distance-based record linkage for disclosure risk assessment. In *Privacy in Statistical Databases*, volume 4302 of *Lecture Notes in Computer Science*, pages 175–186. Springer, 2006.

[69] V. Torra and J. Domingo-Ferrer. Record linkage methods for multidatabase data mining. In *Information Fusion in Data Mining*, pages 101–132. Springer, 2003.

[70] V. Torra and Y. Narukawa. *Modeling decisions: Information Fusion and Aggregation Operators*. Springer, 2007.

[71] U.S. Census Bureau, Data Extraction System, http://www.census.gov/.

[72] U.S. Energy Information Authority, http://www.eia.doe.gov/.

[73] D. W. Wang, C. J. Liau, and T. S. Hsu. An epistemic framework for privacy protection in database linking. *Data and knowledge engineering*, 61:176–205, 2007.

[74] W. E. Winkler. Matching and record linkage. *Business Survey Methods*, pages 355–384, 1995.

[75] W. E. Winkler. Data cleaning methods. In *Proceedings of the ACM Workshop on Data Cleaning, Record Linkage and Object Identification*, 2003.

[76] W. E. Winkler. Re-identification methods for masked microdata. In *Privacy in Statistical Databases*, volume 3050 of *Lecture Notes in Computer Science*, pages 216–230. Springer, 2004.

[77] R. Yager. On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on System, Man, and Cybernetics*, 18:183–190, 1988.

[78] R. Yager. Applications and extensions of OWA aggregations. *International Journal Man-Machine Studies*, 37:103–122, 1992.

[79] W. E. Yancey, W. E. Winkler, and R. H. Creecy. Disclosure risk assessment in perturbative microdata protection. In *Inference Control in Statistical Databases*, volume 2316 of *Lecture Notes in Computer Science*, pages 135–152. Springer, 2002.