Differential Privacy Protection via Inexact Data Cloning

by

Zelpha Thomas

A Dissertation Presented in Partial Fulfillment
of the Requirement for the Degree
Doctor of Philosophy

Approved April 2023 by the
Graduate Supervisory Committee:

Daniel W. Bliss, Chair
Antonia Papandreou-Suppappola
Ayan Banerjee
Aviral Shrivastava

ARIZONA STATE UNIVERSITY

May 2023

ABSTRACT

With the advent of new advanced analysis tools and access to related published data, it is getting more difficult for data owners to suppress private information from published data while still providing useful information. This dual problem of providing useful, accurate information and protecting it at the same time has been challenging, especially in healthcare. The data owners lack an automated resource that provides layers of protection on a published dataset with validated statistical values for usability.

Differential privacy (DP) has gained a lot of attention in the past few years as a solution to the above-mentioned dual problem. DP is defined as a statistical anonymity model that can protect the data from adversarial observation while still providing intended usage. This dissertation introduces a novel DP protection mechanism called Inexact Data Cloning (IDC), which simultaneously protects and preserves information in published data while conveying source data intent. IDC preserves the privacy of the records by converting the raw data records into clonesets. The clonesets then pass through a classifier that removes potential compromising clonesets, filtering only good inexact cloneset. The mechanism of IDC is dependent on a set of privacy protection metrics called differential privacy protection metrics (DPPM), which represents the overall protection level. IDC uses two novel performance values, differential privacy protection score (DPPS) and clone classifier selection percentage (CCSP), to estimate the privacy level of protected data.

In support of using IDC as a viable data security product, a software tool chain prototype, differential privacy protection architecture (DPPA), was developed to utilize the IDC. DPPA used the engineering security mechanism of IDC. DPPA is a hub which facilitates a market for data DP security mechanisms. DPPA works by incorporating standalone IDC mechanisms and provides automation, IDC protected

published datasets and statistically verified IDC dataset diagnostic report. DPPA is currently doing functional, and operational benchmark processes that quantifies the DP protection of a given published dataset. The DPPA tool was recently used to test a couple of health datasets. The test results further validate the IDC mechanism as being feasible.

I dedicate this dissertation to my family and with special appreciation to those who are seeking new insights for leverage.

ACKNOWLEDGEMENTS

I would like to express my most worm gratitude and appreciation to my family for their support, insights and encouragements towards the completion of this effort. Thanks for your vigilance, mindfulness and guidance along this journey. I appreciate you a lot.

I would like to thank my committee members for their exemplary diligence and quest for clarity in documented works, teaching methodologies and presentation styles. I appreciated all of your contributions, many, many thanks to all of you. I appreciate the foot prints you left for me to follow.

I would like to thank BLISS Labs and WISCA for their guidance and support.

I would like to extend exceptional thanks to Dr. Daniel Bliss, my Committee Chair for providing research topic, research direction, general guidance, presentation contributions, patience and other support. Thanks for providing insights, encouragements and suggestions which directly contributed to the completion of this work. Thanks for providing Dr. Owen Ma and Dr. Arindam Dutta as support for this research.

I also would like to extend special thanks to Dr. Owen Ma and Dr. Arindam Dutta for their encouragements, reminders, patience, followups and general guidance. Their efforts helped me to complete this work. Thanks to both of you for making sure I say on the path to completion. Thanks for looking beyond and showing me the forward path. I really appreciate both of you.

I would like to thank my friends for their exemplary track records of similar efforts. Thanks for listing, sharing your ideas and all your recommendations.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

1.1 Introduction

For this research, differential privacy (DP) is used in the context of data security, whereby explicit or calculated privacy in a data record is protected via inexact data cloning (IDC). The nomenclature, inexact data cloning means inexact replicas of raw data that maintain raw data semantics, and protect selected raw data parameters while allowing useful computation on data.

A data breach of adversarial access or exposure of differential privacy can lead to fateful consequences for data businesses [1]. Data businesses that do not practice filtering of private identifiable data can be liable for breaking the law [2]. Differential privacy breach includes cases where the adversary rightfully obtained data and then performed calculations, statistical analysis, or some comparison means to extract private evidential information. This information can be about a record, an individual, a business, an organization, or an entity. This work uses biometrics data breach of type referential analysis to demonstrate data security via IDC. Biometrics is a set of features that contain universal, unique, and permanent human identifiable such as whole-body, gait, facial imagery, and anthropometry metadata. These features are used for human verification, recognition, or identification [3], as a consequence of which they require strict privacy protection. With the increase in new advanced data processing tools, data hackers can do more sophisticated analyses and extract private information from published data. It is necessary to devise novel DP tools to combat the ever-increasing analysis methods being provided to data hackers. New

DP tools will; for example, help data scientists from being exposed to individuals' data, suppress personal information in published data, and provide data variation in data used for testing, training, and evaluation in machine learning applications.

IDC is a methodology for protecting private parameters in published data. IDC was constructed to withstand data security attacks such as reverse engineering, two-way hashing functions, and referential analysis. Reverse engineering includes applying exploratory analysis to abstracted work for extracting auxiliary information from the work [4, 5]. This means, for example, taking the published answers to a sensitive survey, performing analysis on the answers, and identifying information about the participants of the survey. The identified information, in this case, is private differential information. When IDC is used, the private differential information will be an IC and not the real data.

The two-way hashing function includes a mathematical mechanism whereby the input to a hash function is reproduced using a subset of the hash output, along with extra information and no given reverse hash key. This two-way hashing function example is like a one-way permutation-based hash function that has an inversion [6]. A two-way hashing function, in this case, means using the output of the published data to identify the input, source, or private differentials of the published data. When IDC is used, the private differentials of the published data will be ICs and not real data.

Referential Analysis refers to the review of requests for records during a specific period to identify more requested records and request patterns [7]. Reference records are used to present information or to draw logical conclusions or analyses based on the received data. For this case, referential analysis is like using advanced analysis tools to perform analysis on published data for the extraction of private information or private differentials in the published data.

For added security, IDC uses a classifier to exclude selected compromising raw data substitutes.

1.2 Differential Privacy Defined

After reading several reviews on DP, it was noted and observed that in almost every case, the context of DP was given, and then constructs from the context were used to define DP. Below are seven observed cases formulated as salient definitions for DP. Per this document, DP means the state of having selective parameters in published data, unintendedly obtained by calculations, statistical analysis, recordings, or some comparison means, being concealed from adversarial observation. Simultaneously, the published data provides useful calculations to its intended. Other authors describe DP in respective contexts. This definition is similar to the definition by Barthe, Gilles et al. of [8]. In [8], DP is summarized as being a concept of confidentiality that protects the privacy of individuals while simultaneously allowing useful computations on their private data

Per Cynthia Dwork and Aaron Roth of [9], DP describes a promise made by a data holder, or curator, to a data subject. The data holder promises the data subject no adverse effects while allowing their data to be used in any study or analysis, no matter what other studies, data sets, or information sources are available. DP addresses the concept of learning nothing about an individual while learning useful information about a population containing the individual. In this reference, the author is assuring the data owners that their private information will remain private as only group information will be released and not individual information.

Per Almadhoun, Nour, Erman Ayday, and Özgür Ulusoy of [10], DP provides a mathematically rigorous approach to suppress the chance of extracting membership inference from a published dataset while sharing statistical information about

a dataset. This author is focused on growing privacy concerns about the sensitive information for participants of genome sequencing biomedical data collection. The conclusion was that since biomedical breakthroughs and discoveries require publishing genomic datasets, it is necessary to introduce a privacy-preserving mechanism to protect participants' identities in the biomedical dataset.

Per Kairouz, Peter, Sewoong Oh, and Pramod Viswanath of [11], DP is a formal construct to quantify to what extent individual privacy in a statistical database is preserved while providing useful aggregate information about the database. This paper is about extracting private information via sequential querying of a dataset. Emphasis is placed on the privacy degradation of the dataset as a function of total queries to the dataset. In conclusion, a privacy mechanism was constructed to produce an upper bound on the overall privacy level.

Per X. Li, C. Luo, P. Liu, and L. Wang of [12], DP privacy is a strict privacy protection framework proposed to address statistical database privacy leakage. Data leakage is prevented by adding noise to statistical results. This paper focuses on protecting DP related to the correlation in record attributes. The protection mechanism publishes the correlation of the records equivalence class, noise, instead of the correlations in the record's attributes. Correlations of a record's attributes can lead to the reidentification of the record owner.

Per Julian Steil et al. of [13], DP is about protecting gender and user re-identification of participants who published their gazed-based data collected while using an eye-tracking tool.

1.3 Problem Statement and Goals

Differential private data in published or raw data contains sensitive information that can be used adversely. This adverse manner can harm an individual, a company,

or an organization. DP, as it relates to this research, means the state of having sensitive parameters in published data being concealed from adversarial observations. The sensitive parameters may be obtained by calculations, statistical analysis, reverse engineering, or some comparison means. The published data, while being canceled from adversarial observation, is providing useful calculations to its intended users. This research uses inexact data cloning to facilitate privacy and useful calculations in published data. This facilitation involves replacing the raw published data with a raw data substitute called inexact clones. Raw data is expensive and difficult to obtain.

The problem is, with advanced statistical analysis tools, data owners are finding it difficult to conceal sensitive information or sensitive differential in published data. This research uses IDC on biometrics raw data of parameters weight, height, body mass index, waist, hip, and waist-to-hip ratio to demonstrate the concealing of sensitive information in published data. IDC is also used to solve the secondary problem of preserving calculated data which is dependent on other raw recode attributes.

The main goal of this research is to present IDC as a standardized mechanism that suppresses differential privacy data breaches while allowing useful calculations on published raw data substitutes. The secondary goal is to establish a benchmark for calculating data security performance via the IDC privacy protection metric systems, DPPS, and CCSP. The privacy protection rating would include using PDFs to validate IDC as valid raw data substitutes and using other statistics to validate protected and preserved parameters.

The rest of this document is outlined as follows. Chapter 2 is about DP protection mechanism approaches. The topics covered are differential privacy protection solution routes, noise adding as the de facto privacy mechanism, and four salient privacy mechanisms. Chapter 3 is about the metrics of DP. The topics covered are

contributions and depictions of salient privacy metrics, followed by components of the novel differential privacy protection metrics (DPPM). Chapter 4 is about the performances of IDC. These performances are proposed by this work and include DPP and CCSP. Chapter 5 presents IDC as a valid raw data substitute. Chapter 6 is about the summary and future works.

Chapter 2

DIFFERENTIAL PRIVACY PROTECTION APPROACHES

2.1 Differential Privacy Protection Solution Routes

The solution provided by this work protects DP using inexact clones. The solution: while protecting, preserves useful data computation by intended raw data users. After reading other works, it was noted that the solutions to DP were established in some form of noise-adding or noise-inclusion mechanism. Noise comes in different forms, and so noise is really a name given to a collection of actions that serve the purpose of hiding or protecting data sensitivity. The noises disguise or normalize the DP in the selected data. The noise can be included in forms such as heuristic, algorithmic, or specifications. It was concluded that the noise inclusion mechanism could be modeled on three main properties: the when, where, and what of the noise. This work used these three properties as solution routes for DP protection mechanism approaches.

2.1.1 Noise Inclusion as the de facto Privacy Mechanism in DP

In this research, we assume noise inclusion as the de facto privacy mechanism to protect DP or sensitivity in raw data [12, 13, 14, 15, 16, 17, 18]. From prior studies, we observed that the three main approaches of noise inclusion are when to add noise, where to add noise, and what noise to add, as shown in Figure 2.1. These noise inclusion approaches satisfactorily categorized the privacy mechanism we explored in this document.

For reference, the noise aims to convert the raw data into protected data while allowing useful calculations on the published, protected data.

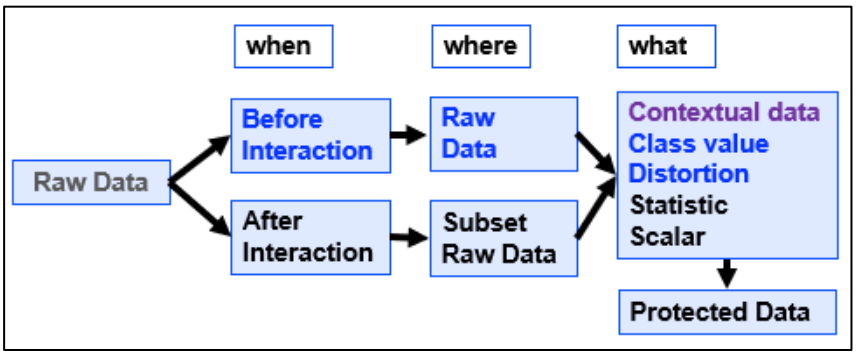


Figure 2.1: Three ordered noise inclusion approaches for raw data. When to add noise, where to add noise, and what noise to add.

For when to add noise to the raw data, the options are before interaction or after an interaction. An example of interaction is querying the data for a result. Regarding where to add noise and knowing when the noise is to be added, the options are either to the raw data or to a subset of the raw data. A subset of the raw data could include up to all records in the dataset. Regarding what noise to add, knowing when the noise is to be added and where to add noise, the salient options are equivalence class values, distortion values, statistic data, scalar data, and contextual data. This work uses contextual data.

Various DP protection approaches are classified based on their ordered noise inclusion properties. Table 2.1 shows salient privacy mechanism approaches and their respective ordered noise inclusion properties. The approaches are inexact data cloning, information entropy [12], algorithm heuristics, randomization [14], and statistical distribution [19]. The IDC approach is the novel approach introduced and described in this document.

The information entropy privacy mechanism approach proposed by X. Li, C. Luo, P. Liu, and L. Wang of [12], adds noise before interaction with the raw data. The raw data is grouped, and respective noise is added to each group. In this case, noise is added to the entire raw dataset using equivalence class values [12].

Table 2.1: Five Privacy Mechanism Approaches and Their Ordered Noise Inclusion Properties of When to Add Noise, Where to Add Noise and What Noise to Add.

Privacy Mechanism Approaches	When	Where	What
Inexact Data Cloning	Before Interaction	Raw Dataset	Contextual Data
Information Entropy	Before Interaction	Raw Dataset	Class Value
Algorithm Heuristics	Before Interaction	Raw Dataset	Distortion
Randomization	After Interaction	Query Affected Dataset	Statistic
Statistical Distribution	After Interaction	Query Affected Dataset	Scalar

The algorithm heuristics privacy mechanism approach used by software engineers, in general, adds noise before interaction with raw data. The noise can be added to the entire raw data set, to an entire row, to an entire column, or to parts of the data set as needed. Distortion, abstraction, or redaction are common noises used in the algorithm heuristics privacy approach.

The randomization privacy mechanism approach adds noise after interaction with the raw data. Interaction, in this case, comes in the form of a query to the dataset. Noise as appropriate statistics is added to the query response, resulting in a protected query response [14].

The statistical Distribution privacy mechanism approach adds noise after interaction with the raw data. The interaction can be a query or constantly changing historical data. The noise is added to the query-affected data set or the historical data using appropriate variables such as scalar or statistics [19, 15].

Finally, our proposed IDC privacy mechanism approach adds noise before interaction with the raw data. The noise is added to the entire raw dataset using contextual data. Contextual data means data that is dependent on the type of raw data. For

example, raw data could be biometrics, country locations, or cars. In this case, the noise has to be related to biometrics data, country locations, or cars respectively.

2.2 Information Entropy

Figure 2.2: Noise Inclusion Order Used by Information Entropy Privacy Mechanism to Produce Protected Data; Updated Data. 1: When to Add Noise, 2: Where to Add Noise 3: What Noise to Add.

The information entropy differential privacy protection mechanism (IEDPPM) approach was proposed by X. Li, C. Luo, P. Liu, and L. Wang of [12]. The IEDPPM approach is mainly used on databases. Statistical databases are databases that do not return actual replies/information to requests but instead return an approximate answer. Approximate answers are statistics about an equivalence class to which a record belongs and not statistics about the actual record. The following is an attempt to explain the IEDPPM given in [12].

In Figure 2.2, we show the noise inclusion order by the IEDPPM to produce protected data, which we refer to as updated data. The noise inclusion approach can be described as follows.

1. When the noise is to be added: before interaction with the raw data.

2. Where the noise is to be added: to the raw data.

3. What noise to be added: equivalence class value.

Figure 2.3 is a diagram showing example raw data, example equivalence class and value, information entropy noise adding mechanism, and resultant protected

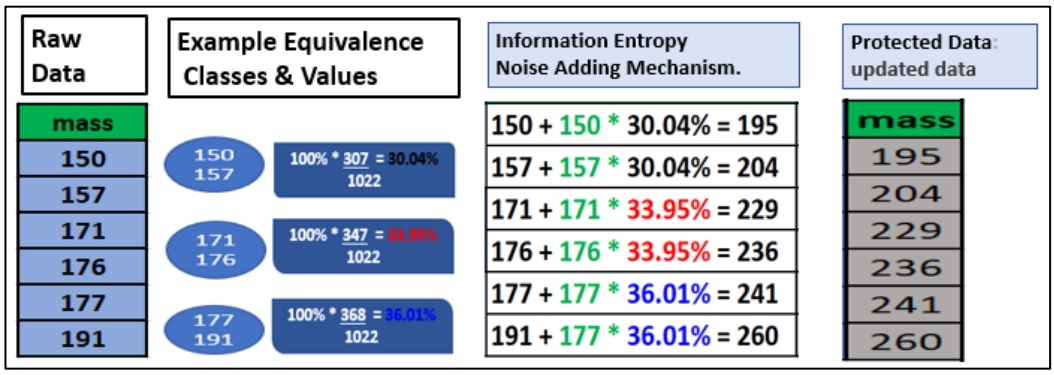


Figure 2.3: Diagram Showing Example: Raw Data, Example Equivalence Class and Value, Information Entropy Noise Adding Mechanism and Resultant Protected Data, Used in Information Entropy Differential Privacy Protection Mechanism.

data used in IEDPPM. The raw data column contains sensitive data. This raw data column is transformed into protected data, the rightmost column in the diagram. The transformation mechanism starts with creating values for equivalence classes as seen in the column, for example, equivalence class and values. The raw data is grouped into three equivalence classes. Class 1 has numbers 150 and 157. The equivalence class value for class 1 is 30.04%. The value is calculated by taking the percentage of the summed class; 307, divided by the summed column; 1022. A similar manner is done for classes 2 with values 171 and 176 resulting in an equivalence value of 33.95% and also for class 3 with values 177 and 191 resulting in an equivalence value of 36.91%. The column, information entropy noise adding mechanism, shows how noise is added to the raw data. For class 1, noise is calculated by multiplying the raw data record value; 150 by the equivalence class value; 30.04% giving a noise value of 45. The noise; 45 is added to the raw data; 150 gives a protected value of 195. All protected values in column Protected Data are calculated similarly. Below shows the calculations for the first two raw data entries.

$$150 + (150 * 30.04\%) \rightarrow 150 + 45 \rightarrow 195 \tag{2.1}$$

$$157 + (157 * 30.04\%) \rightarrow 157 + 47 \rightarrow 204 \tag{2.2}$$

In general, the properties of the information entropy privacy mechanism are as follows. 1) Raw data is used by statistical databases which provide approximate results to query requests instead of actual answers. 2) Raw data records are put into groups called equivalence classes or rough sets to generate noise percentage components. 3) The equivalent class % is used to calculate the noise. 4) The noise is added to the raw data to create the protected data. 5) This mechanism suppresses raw data record information by providing information of the record's equivalence class instead.

2.3 Randomization

Figure 2.4: Noise Inclusion Order Used by Randomization Privacy Mechanism to Produce Protected Data; Updated Data. 1: When to Add Noise, 2: Where to Add Noise 3: What Noise to Add.

The randomization differential privacy protection mechanism (RDPPM) approach was proposed by Cynthia Dwork of [14]. The randomization protection mechanism approach is mainly used on statistical databases. Statistical databases are databases that do not return actual replies/information to requests but instead return an approximate answer. This mechanism aims to generate static that will help in providing a satisfactory approximation. The statistic is not given and can be ad hoc. The following is an attempt to explain the randomization protection mechanism given in [14].

Figure 2.4 shows the noise inclusion order used by RDPPM to produce protected data, which we refer to as updated data. The noise inclusion approach can be described as follows. 1 indicates when the noise is to be added: post interaction with the raw data. Interaction is like a user query. 2 indicates where the noise is to be

added: to a subset of the raw data. 3 indicates what noise to be added: statistic. The statistic values plus the subset raw data, created the protected data, updated data.

Raw Dataset	Chosen Series of example Questions	True Statistic	Noise	Protected Data: updated data
Weight	-Size of dataset?	True: 6 > 4	> 4	> 4
150	-Count (weight < 170)?	True: (2/6) ~ 33%	33%	33%
157	-Max(weight) ?	True: 191 > 178	> 178	> 178
171	-Min(weight)?	True: 150 < 157	< 157	< 157
176	-Count(Integers)?	True: 100% > 80%	> 80%	> 80%
177	-Count (multiple of 10)?	True: 16.5% < 20%	< 20%	< 20%
191	-Contains dight 1?	True: 100% > 80%	> 80%	> 80%

Figure 2.5: Diagram Showing an Example: Raw Data, Chosen Series of Questions or Query Sets, Noise, True Statistic, and Resultant Protected Data, Used in Randomization Differential Privacy Protection Mechanism.

Figure 2.5 is a diagram showing an example: raw data, chosen series of questions or query sets, noise, true statistic, and resultant protected data used in RDPPM. The first column shows an example raw data set. The statistical database receives interactions or queries, or questions about the raw data. The second column shows example queries on the raw data. The third column shows the true statistic for these example questions. The next column is the noise column. The noises are extracts from the true statistics that are not precise answers but are valid responses to the example questions. For example, the first query requests the size of the dataset. The true answer is 6; there are 6 items in the weight column. The generated noise is greater than 4. The value 6 is greater than 4; therefore, the response of greater than 4 is a true statistic for 6; 6 is greater than 4. The right column shows the protected data; response from the statistical database to the example questions.

In general, the properties of RDPPM follow. 1) Raw data is used by statistical databases which provide queries, approximate results instead of actual raw data results. 2) Use proximity response as a privacy mechanism. 3) Suppresses private data

by providing statistics. 4) Pre-identify questions and formulate answers as needed.

2.4 General Algorithm Heuristic

The noise inclusion order used by general algorithm heuristic privacy mechanisms (GAHPM) varies. Three examples of GAHPM are redaction, obscuring, and distortion. The concept of when to add noise, where to add noise and what noise to add changes a needed. These privacy mechanisms are ad-hock, so they are mentioned to let the reader know of them as options.

Figure 2.6: Diagram Showing Example Illustration of Before and after Privacy Mechanism for General Algorithm Heuristics: Redaction, Obscuring, and Distortion.

Figure 2.6 is a diagram showing an example illustration of the before and after privacy mechanism for GAHPPM: redaction, obscuring, and distortion. Column 1 shows the state of the raw data before the privacy mechanism is applied. The sensitive information or attribute of interest is the weight column. This attribute needs protection. Column 2 contains the privacy mechanisms, redaction, obscuring, and distortion. The next column shows an example illustration of the protected attribute after applying each privacy mechanism. The redaction mechanism protects by leaving

blank the attribute name and attribute values. The obscuring mechanism protects by including the attribute name and blocking out the attribute values. The distortion mechanism protects by is including the attribute name and providing unrealistic attribute values.

In general, the properties of general algorithmic heuristic privacy mechanisms are as follows. 1) Raw data is a column of a table. The table resides in, for example, a relational database. 2) Raw data might be published data where sensitive information needs to be blocked out. 3) Raw data might be in a dependent table in a database. This means the dependent table cannot be removed from the database because another table depends on it. In this case, distortion of a column in the dependent table is needed to maintain the referential integrity of the dependent table to the database.

2.5 Inexact Data Cloning

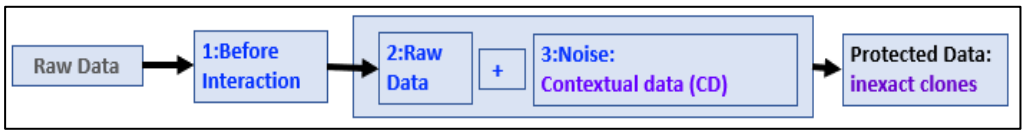


Figure 2.7: Noise Inclusion Order Used by Inexact Data Cloning Privacy Mechanism, to Produce Protected Data; Inexact Clones. 1: When to Add Noise, 2: Where to Add Noise 3: What Noise to Add.

In Figure 2.7 we show the inclusion noise order used by the IDC privacy mechanism to produce protected data, which we refer to as inexact clones. The noise inclusion approach can be described as follows. 1 indicates when the noise is to be added: before interaction with the raw data. An example interaction is like a query to the raw dataset. 2 indicates where the noise is to be added: to the raw data. 3 indicates what noise is to be added: noise that is appropriate to the DP context. After the contextual noise is added, protected data is produced. The novel name for the protected data is inexact clones.

Contextual data originates from domain knowledge of shifts. A shift in a particular area of interest, a specific body of knowledge, a domain, etc. Domain knowledge is information that is specific to a particular shift, for example, the limbs ratios of an average basketball player. Example shifts are body types of national basketball players, body types of national football players, body types of high school track team members, etc. Each shift has its statistical distribution and properties. For this work, the purpose of the contextual data is to facilitate the intra-correlation of parameters in shifts.

Figure 2.8 illustrates five shifts. In the illustration, the identity of the person in the first column is shielded using inexact clones. Each set of IC is from one of five shifts: Family Male, Family Female, Young Adult, Body Transformation, and Athlete Male.

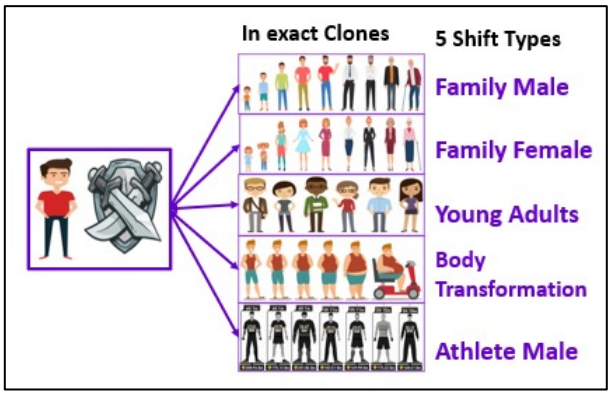


Figure 2.8: Diagram Illustrating Six Shifts Used to Create Inexact Clones.

The contextual data for the used biometric data is a scalar. When the contextual data is added to the raw data value the sum gives a new data value that becomes a member of the shift distribution. The new data: inexact clones, preserves correlations in the raw data. This work used three shifts, shift 1:taller, shift 2: shorter and shift 3: much shorter. For a noisy variable of type, CD_{ijk}, the first subscript corresponds to the shift used. The second subscript corresponds to the raw data parameter index.

The third subscript corresponds to the indexed IC that will use the generated shift value.

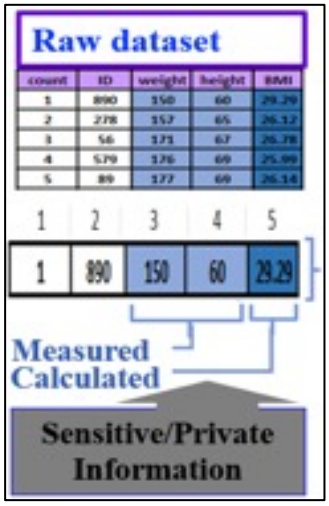


Figure 2.9: Example Raw Dataset Used by Privacy Mechanism, Inexact Data Cloning. Illustrating Measured/Private and Calculated/Preserved Parameters.

Figure 2.9 shows an example raw dataset used in the privacy mechanism of IDC. The raw dataset has five records and six columns: count, ID, weight, height, and body mass index (BMI). Record 1 is isolated for illustration purposes, showing the measured/private and calculated/preserved data. The weight and height are measured parameters and the BMI is a calculated parameter. Measured parameters are sensitive and need to be protected. The calculated parameters may or may not be private, however, they need to be preserved. For this data record case, the weight, height, and BMI are sensitive or private information. The BMI is calculated using the values of weight and height. The BMI of an IC is not sensitive since its dependent variables, weight, and height are in a protected form.

Figure 2.10 shows an example raw data record, IDC noise adding mechanism, clonesets, raw dataset, classifier, and classifier outputs used in inexact data cloning DP protection mechanism. The example raw data record has five columns. Columns 3 and 4 are measured data. Measured means data provided by the data owner.

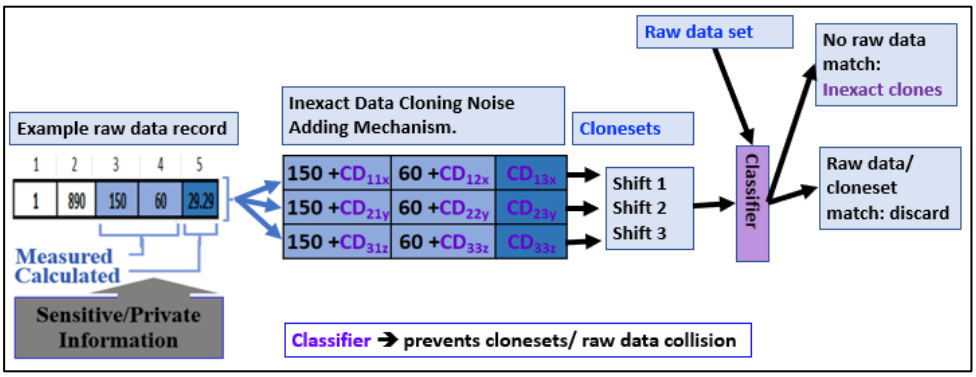


Figure 2.10: Diagram Showing Example Raw Data Record, Inexact Data Cloning Noise Adding Mechanism, Clonesets, Classifier and Classifier Outputs, Used in Inexact Data Cloning Differential Privacy Protection Mechanism.

Column 5 is calculated data; data created from measured data. Columns 3, 4, and 5 are sensitive, private information. The IDC noise-adding mechanism uses 3 shifts. After the noise is added, the shifts are combined to form a cloneset. The noise is contextual data added to raw data values to form the shift distribution. The clonesets and raw dataset go through a classifier. The classifier discards clonesets that satisfactorily match any raw data record and filters inexact clones as clonesets that do not satisfactorily match any raw data record. The word match is subjective. For this work, a raw data record has four measured attributes. A match means, of the four measured attributes, at least 2 matches the exact attributes of an inexact clone record. A match is the same as a collision, inexact clone, and raw data record collision. Collided cloneset records are discarded

In general, the properties of inexact data cloning privacy mechanism (IDCPM) include the following. 1) Create inexact copies; inexact clones, of raw datasets using templates, called shifts. 2) Maintain intra-correlations of the raw dataset in shifts. 3) Shield intent of calculated or preserved data. 4) Suppress real value of measured, sensitive or private data. 5) Hide record source using inexact clones. 6) Exceed raw dataset in numbers. 7) Provides a performance metric; differential privacy protection

score (DPPS) which is a level of data security. 8) Implements clone classifier selection percentage also called classifier's collision-free performance. This selection process introduces another level of data security for the combating of referential analysis attacks.

Chapter 3

METRICS OF PRIVACY

3.1 Salient Privacy Metric Contributions and Depictions

DP has been used in practice for a while. This section reviews a subset of previous contributions to the DP field. We highlight example applications to which DP has been applied and some methods for measuring its efficacy. We show in Figure 3.1 the selected works, organized by their application, and identify their particular protection metric. We also identify what this work contributes toward measuring DP efficacy and an example use case.

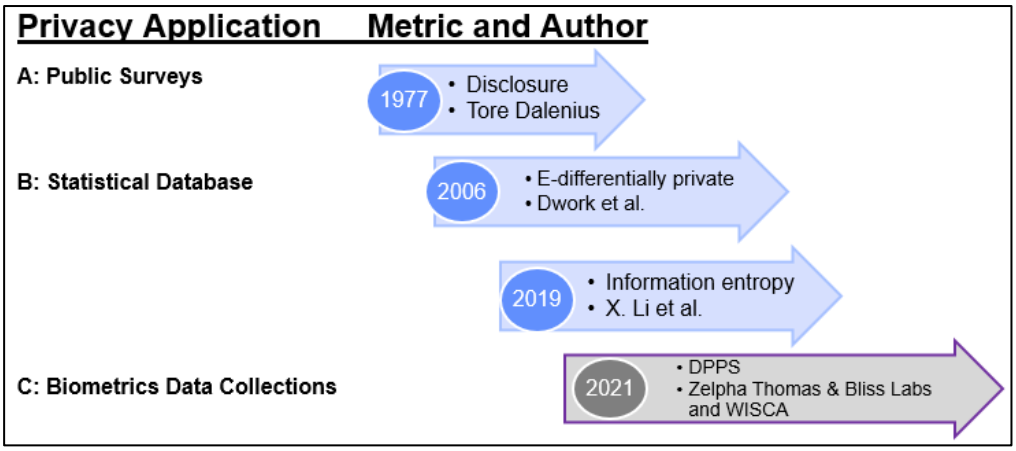


Figure 3.1: Diagram Showing Three Privacy Applications and Their Respective Metrics and Authors.

In 1977, Dalenius et al. used the metric of disclosure to quantify released information in public surveys [20]. This released information is a measure of DP. Dalenius et al. emphasized that controlling the disclosure of information from computerized databases prevents confidentiality violations of survey participants [21]. The work of Dalenius et al. was used as the foundation work for other DP research. The authors,

Duncan, George T., and Diane Lambert of [22] summarized that their framework of, "Disclosure-Limited Data Dissemination", is used as the foundation definition of disclosure proposed by Dalenius [20].

Dalenius in his 1977 paper, "Towards a methodology for statistical disclosure control", identifies the problem of statistical disclosure as unintended exposure of private data in released results of sample and census surveys. This problem was connected to public concern about invasion of privacy in the context of releasing statistics from a survey. The purpose of Dalenius's paper was to suggest a definition for statistical disclosure, present a theory for statistical disclosure, provide examples of statistical disclosure, and propose a methodology for statistical disclosure control.

To define the disclosure, Dalenius introduced a conceptual framework. The framework consists of a frame comprising of objects $\{O\}_F$. Data is associated with these objects, D, I, C, X,... Z. The data consist of responses to a survey and statistics, S, of the data is released to the public. Also released to the public are extra-objective data, E, which may be related to the source and additional information about the objects. Using the framework, the following is the salient part of Dalenius' suggestion for statistical disclosure definition. If the release of the statistics, S, makes it possible to determine the value DK more accurately, than is possible without access to S, a disclosure has taken place, more exactly a D-disclosure has taken place. This disclosure could have been an X, Y, or Z disclosure according to the objects in the data.

Dalenius's theory of statistical disclosure is based on understanding the disclosure phenomenon. The phenomenon includes identifying characteristics that reveal the complexity of the underlying release effort. The complexity includes developing a topology consisting of dimensions of subproblems. Dimensions would include 1) kinds of statistics, S, 2) measurement scale used to express S, 3) accessibility of disclosure,

4) accuracy of disclosure, 5) scope of the disclosure and 6) the disclosing entities. Combinations of these dimensions determine a disclosure classification, DK, and hence the theory behind statistical disclosure. Having 6 dimensions, Dalenius inferred about $2^6 = 64$ types of disclosures. Their detailed explanation was deferred as they were beyond the scope of his paper. Their existence and conception were only needed for mentioning.

According to Dalenius, the objective of a survey is expressed in terms of population and sample variables, $X, Y, \ldots Z$. To achieve this objective, the statistic S is released. Example released statistics can be classified into macro statistics and micro statistics. Macro statistics deals with information about a population, for example, a city, country, town, or a business, and is usually in the form of aggregate reports. For mentioning purposes, a few example disclosure terms associated with macro statistics reports are exact direct disclosure, exact indirect disclosure, and approximate direct disclosure. Micro statistics deals with information about individuals with their identifiable missing and are in forms such as census reports. The disclosure problem for micro statistics is as follows; the statistics released for an individual object, O_j does not contain the identifier I, of the individual, however, this does not mean that O_j cannot be identified.

Dalenius proposed a methodology for statistical disclosure control and integrated his definition, theory, and examples of statistical disclosure into the criterion problem and the techniques for control. The criterion problem focuses on a documented process of what can and cannot be disclosed per given specifics. In this sense, SDC is an attempt to formally prevent or avoid unintended disclosure. The criterion has two measures and is quoted as follows. 1) $M = M(Si, E)$, the amount of disclosure associated with the release of some statistics $Si(I = 1, 2, \ldots, k)$ and the extra-objective data E; and 2) $B = B(Si)$, the benefit associated with the statistics Si. It would

then be possible to use a criterion of the following type: Maximize B for $M = M0$. The technique for control focuses on general-purpose and special-purpose means. The general-purpose means focuses on training the statisticians and using sampling techniques to avoid unintended disclosure. The special-purpose means focuses on tailoring unintended disclosure for identified disclosures such as direct disclosure and approximate direct disclosure.

In 2006, Cynthia Dwork et al. used the metric of ε-differentially private to quantify the protection of a statistical database. ε-differential privacy is a measure of privacy loss [14]. Cynthia Dwork describes DP as a promise made by a data holder or a data creator to a data subject. The data holder promised the data contributor that the contributor will not be affected adversely or otherwise, by allowing your data to be used in any study or analysis. This also means no matter what other studies, datasets, or information sources are available [9]. ε-differential privacy is the metric used for measuring that promise [9].

In [14], Dwork describes ε-DP as the output of a function. The definition states that a randomized function K gives ε-DP if for all data set $D1$ and $D2$ differing on at most one element, and all set S is a subset of Range(K). This definition means a dataset with or without one participant will provide no more or less information. Dwork further describes a concrete interactive privacy mechanism that archives ε-DP. The mechanism involves the adding of random noise which is a function of the largest change a single participant could have on the output of a query function, the sensitivity of the function.

In [9], Dwork et al. explain ε-DP in the form of algorithmic foundations. They explain that DP is a definition and not an algorithm. They explain, given the task T, a given value ε; many differentially private algorithms can achieve T in an ε-differentially private manner. The smaller the more private is the task T.

In 2019 X. Li et al. used information entropy to measure privacy leakage from a statistical database [12]. They see DP as a strict privacy protection model proposed for privacy leakage from statistical databases. The protection mechanism adds noise to the statistical results through the noise mechanism to prevent the privacy of the data from being leaked [12].

X. Li et al. explain that even though information entropy is a protection measure, it is also a measure of uncertainty. The larger the information entropy the greater the uncertainty analogous to the adversary having to do more work. This scenario was used to introduce information entropy into DP to measure attribute senility plus other interests. To measure information entropy DP, one can consider two data sets, D and D'. There is at most one record difference between them: $|D - D'| \leq 1$. $H(P)$ represents the information entropy of the attribute H. Given the privacy algorithm, M, Range(M) represents the range of values of M. If the arbitrary output x (x is a member of Range(M) of the algorithm M on the data sets D and D' satisfies the following inequality: $\Pr(M(D) = x)/\Pr(M(D') = x) \leq \exp(\varepsilon H(P))$: then we say that the algorithm M satisfies the ε-information entropy differential privacy. This inequality is explained in more detail below. Per measuring privacy, X. Li et al. considered the distribution of attribute values and used information entropy to measure the sensitivity of attributes and propose the concept of information entropy difference privacy.

In this work, we use DPPS to quantify a measure of privacy when using inexact data clones (IDCs). Inexact data clones are inexact copies of raw data that maintain the intra-correlation of the attributes in raw data records. IDCs are used as substitutes for the raw data, and the raw data is never shared. DPPS is a measure of obscurity in IDC when the IDC is unintendedly used. This obscurity is measured using heuristics and statistical analysis explained in detail in this document.

DPPS depicts privacy as a level of obscurity from unintended observers. Obscurity aims to suppress raw data records re-identified given an inexact data clone record. ε-differential privacy measures privacy in terms of privacy loss [14]. This privacy loss is the difference between two datasets that differs only by one record. Information entropy quantifies privacy in terms of the assigned weight of a record's equivalence class.

A syntax for a standardized formula that describes privacy metrics is as follows:

$$\frac{\Pr(M(D) \in \Omega_1)}{\Pr(M(D') \in \Omega_1)} = A\,. \tag{3.1}$$

This formula was constructed after reviewing [23, 12, 24, 25, 14]. Each reference has a version of this constructed formula. Pr means probability. Probability acts as a standardizing or normalizing function. This means it is used so the similarities in the different formula versions can be identified and their comparisons can be done using the same weight. M is a privacy-generating algorithm that maps raw data into some representative form. D and D' depict two datasets that differ by one record. This is the same as $|D - D'| = 1$ record. The omega letter indicates that the output of $M(D)$ and the output of $M(D')$ are of the same class. A represents the value of differential privacy. In general, this equation means that A is the level of privacy loss or privacy gained when an individual is added to or subtracted from a group, where the only change is the individual being added or subtracted to the group.

For this research, M represents the process that generates inexact data clones from raw data. For the ε-differential privacy formula, M represents the randomization privacy mechanism that generates the noise of true statistics mentioned in section 2.5. For the information entropy formula, M represents the information entropy privacy mechanism mentioned in section 2.4.

The actual formulas for DPPS, ε-differential privacy, and information entropy can

be re-arranged algebraically to match the above-constructed formula. This formula rearrangement is used so as to show similarities and common methodologies used by DP authors in formula constructions. For this work, the rearranged formulas show clarity and provide top-down insights. A more detailed understanding of the sourced formulas can be seen in their respective references. These rearranged formulas are as follows.

The following is a representation of the arranged formula for DPPS:

$$\frac{\Pr(M(D) \in \Omega_1)}{\Pr(M(D') \in \Omega_1)} = A = p\% \,. \tag{3.2}$$

A equates to $p\%$. The $p\%$ means the percentage of obscurity. The p has a value between 0 and 100. This rearranged formula, semantically matches the equation for constructing the source component DPPM metric, record clone ratio. See section 3.2.4 for more information on the sourced formula for record clone ratio.

The following is a representation of the arranged formula for ε-differential privacy:

$$\frac{\Pr(M(D) \in \Omega_3)}{\Pr(M(D') \in \Omega_1)} = A = \exp(\varepsilon_1) \,. \tag{3.3}$$

The A equates to formula $\exp(\varepsilon_1)$. ε_1 represents privacy loss value or privacy protection. When the individual or record is added to the data set, that is the privacy they gained by the dataset. When the individual is removed from the dataset, that is the privacy loss of the individual. Privacy loss could be interchangeably used for privacy protection. Privacy is actually an entropy delta. The exponential form of ε_1 serves a normalization purpose. Privacy loss in this equation has an extremely small value. Taking its exponential allows for standardization, normalization, and comprehending of small values like quantitative privacy loss. The following is the sourced formula for ε-differential privacy

$$\Pr(M(D') \in \Omega_3) \leq \exp(\varepsilon_1) \Pr(M(D') \in) \Omega) \,. \tag{3.4}$$

[9] and [14] provides more information on this sourced formula.

The following is a representation of the arranged formula for information entropy

$$\frac{\Pr(M(D) \in \Omega_3)}{\Pr(M(D') \in \Omega_1)} = \exp\left(\varepsilon_2 H(x)\right). \tag{3.5}$$

A equates to the formula $\exp(\varepsilon_2 H(x))$. ε_2 is the equivalence class privacy contribution. This could be privacy loss or gain. For $H(x)$, x is the attribute to be protected or the attribute of interest. The value of equivalence is composite. $H(x)$ isolates the contribution of $H(x)$ to the ε_2 variable, thus providing the privacy loss/gain/contribution of the variable x. Privacy loss in this equation has an extremely small value. Taking its exponential allows for standardization, normalization, and comprehension of small values. The sourced and arranged formulas for information entropy privacy are the same. [12] provides more information on this sourced formula.

3.2 Components of Differential Privacy Protection Metrics

DPPM uses five components for constructing the performance of IDC. The five components include shift count, published dataset size, record clone ratio, and raw data clone set ratio. Shift Count (SC) means the total template used in IDC and serves the purpose of providing diversity in obscurity. Published Dataset Size (PDS) means the size of the published IDC and serves the purpose of curtailing the size of the cloneset size to prevent frivolous IC sizes. Record Attribute Count (RAC) means the minimum raw data record / inexact clone attribute match that warrants removing an inexact clone from a published cloneset. Record Clone Ratio (RCR) means raw data and IDC attribute match and serves the purpose of controlling the re-identification of raw data given an IC. Raw Data Cloneset Ration (RDCR) means the ratio of raw data to IC and serves the purpose of controlling raw data IC collision.

Altering each component of DPPM can change the protection of IC. DPPM values

are between 0 and 1 and are conceived as levels of protection. This section explains the cumulative DPPM values. Cumulative means getting a value for DPPM by adding one metric value at a time. Other values and names are explained in subsequent sections. The cumulative values are used to show differences when IDC protection is not used by a set of raw data. This cumulative form also provides direction on what component to adjust when more protection is needed in data using IDC.

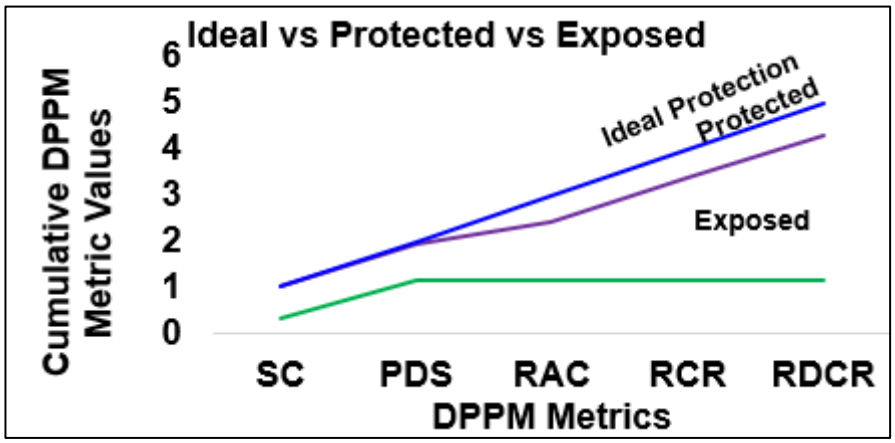


Figure 3.2: Diagram Showing Three Privacy Applications and Their Respective Metrics and Authors.

DPPM is computed through a weighted sum of the five-component metrics, shown in the following equation:

$$\text{DPPM} = \sum_{i}^{5} \text{Metric}_i w_i \,, \tag{3.6}$$

where $w_i = 1$ equally balances each metric's importance. Example calculations of these values will be demonstrated later in the document.

Figure 3.2 shows plots of the ideal, protected, and exposed datasets. Protection means data with IDC protection and no protection means data without IDC protection. The DPPM metrics and their respective value and cumulative value can be seen. The plot of the ideal protection is seen in blue, the protected data is seen in purple, and the unprotected data is in green. Per Figure 3.2 data that are protected

using IDC have a higher DPPM value than data that are not protected with IDC. This is shown where the purple plot follows a higher cumulative curve than the green plot. Unprotected data has zero RAC, RCR, and RDCR values. If one wanted to improve protection in the green plot; the RAC, RCR, and RDCR are areas that can be improved.

3.2.1 Shift Count

The first component of DPPM is the shift count (SC). The shift count protects by diverting the identity of the raw data or individual. It increasingly introduces new record variable options or record patterns for adversarial consideration during adversarial observations. The shift maintains the intra-correlation of the raw data by maintaining a satisfactory ratio of the raw data record. The more shifts, the more obscuring the IC will be for unintended users. The DPPM value for the Shift Count:

$$SC = \text{Total shifts used/Total shifts available for IDC}\,. \tag{3.7}$$

The shift provides distribution variety and hence IC obscurity. Figure 2.8 shows an example dataset where there are five shift types: family male, family female, young adult, body transformation, and athletic male. The interpretative value of shift count for Figure 2.8 is 5. There are 5 shifts available. If all the shifts are included when generating IDCs, then the SC would be 5/5. If only three of the shifts were used, then the DPPM SC value would be 3/5. The formula to calculate the DPPM SC is the total shifts used divided by the total available shifts. The purpose of the SC is to create a diversion for adversarial IC observation.

3.2.2 *Published Dataset Size*

The second component of DPPM is the published dataset size. PDS protects by controlling frivolity that might result in accidental loss of raw data intent. The value of the published dataset size varies from 1 to 100 million. The size of 100 million suits our benchmark purposes. The published dataset size is the size of the published IC. The value for DPPM PDS is based on a novel hill function shown below.

$$\frac{\max\{1, \mathrm{PDS}^{1/3}\} - 1}{3 + \mathrm{PDS}^{1/3}} \tag{3.8}$$

To accommodate large, published data sizes, DPPM transforms the large values of IC sizes into comparable values. The value of the hill function increases as PDS increases and then tapers, giving diminishing returns. IDC is a data privacy tool and the PDS DPPM component is to prevent a frivolous increase in published data size. The components of the hill function were chosen so that the function will provide credit for increased published dataset sizes but reduce the value of the credit as dataset size increased. This methodology fits our benchmark intent. The values in the function can change, but for now, gives a good foundation for this research. Figure 3.3 shows a plot for DPPM PDS Value versus Published IC Dataset size.

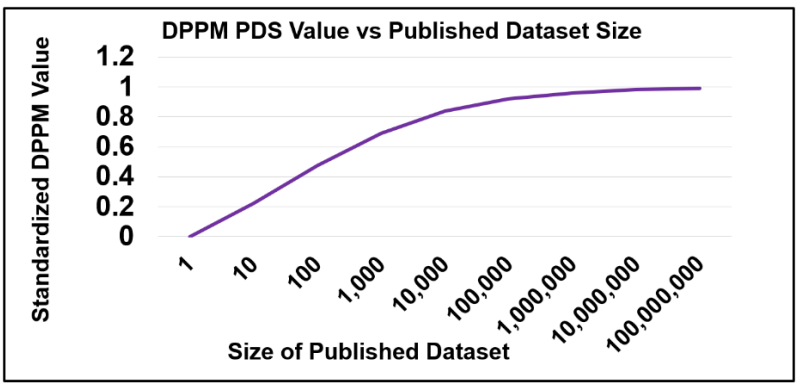


Figure 3.3: Plot Showing DPPM Value for Corresponding Published Dataset Size.

3.2.3 *Record Attribute Count*

The third DPPM component is recorded attribute count (RAC). RAC protects by minimizing partial matches of a raw data records in published cloneset records. The value of the record attribute count is equivalent to the minimum IC/raw data attributes match that disqualifies an IC from being published. If an IC resembles a raw data record too closely, then it must be discarded. We show an example in Figure 3.4 of how the RAC DPPM value is calculated. There are six protected attributes (organized by columns) and six examples (one per row). An x in the table denotes a match between the inexact clone and the original for a particular attribute. A threshold is set to retain a clone if at most one attribute matches. Therefore, all clones except for the first are discarded. The RAC column shows the RAC DPPM value assigned for each case. This value is calculated as

$$1 - (\text{total attribute match/total protecting attributes})\,. \tag{3.9}$$

For clarity, the total protecting attributes are all the parameters in the record. Each parameter is either a preserved parameter or a private parameter.

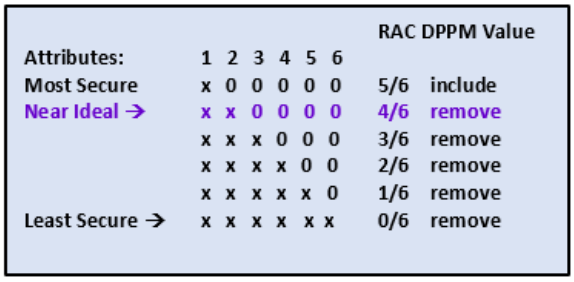


Figure 3.4: Diagram Showing Example Record Attribute Count Dppm Rac Values. X Indicates Raw Data Record Ic Collision. Only Protected Attributes Are Used.

For our work, the classifier does the removal process. The classifier implements a collision detection mechanism that discards clones that match or collide with raw data in a pre-decided way. The classifier controls raw data/cloneset collision.

3.2.4 *Record Clone Ratio*

The fourth DPPM component is the record clone ratio (RCR). RCR protects by providing ample inexact copies of itself to establish a record-level adversarial smoke screen. RCR is the number of created inexact replicas for a raw data record. These clones have not yet passed through the classifier. Per Figure 3.5, an example record clone ratio is 1 to 37. Per Figure 3.5, the RCR DPPM value is $(37-1)/37$. Generally, the normalized value is given by the clone ratio minus 1, all divided by the clone ratio. For this work, three shifts were used and nine inexact clones were generated from each shift. Three shifts were used as it serves the purpose of establishing a benchmark number of shifts. The same number of inexact clones were chosen from each shift as it serves our attempt of establishing a benchmark effort. Nine inexact clones were chosen as this would result in the largest RCR value which is less than 90% if only one shift was used. 90% is an exploratory benchmark percentage. In the end, the resultant benchmark effort used a record clone ratio of 1 to 27. This gives an RCR DPPM value of $(27-1)/27 = 26/27 = 0.96296$.

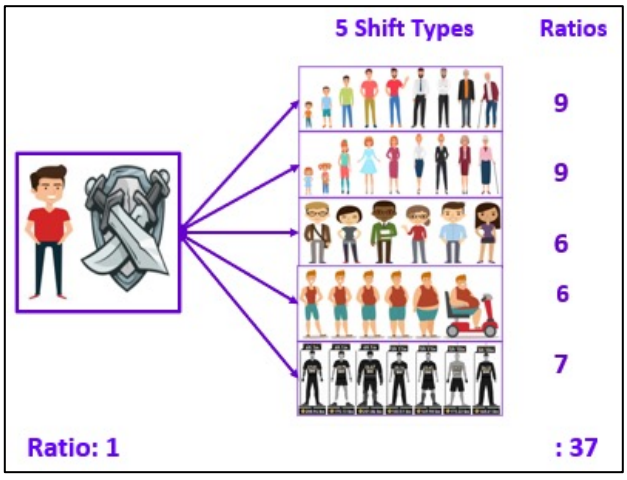


Figure 3.5: Diagram Showing Record Clone Ratio Value of 1: 37

3.2.5 Raw Data Cloneset Ratio

The last DPPM component is the raw data cloneset ratio (RDCR). RDCR protects by reducing the probability of identifying or selecting an adversarial record of interest. It increases the total records released for public observations. RDCR is the ratio of raw data count to published IC count. In DPPM, RDCR accounts for added security contributed by the classifier usage. Section 3.2.3 explains what a collision is and what to do when one occurs. The collision is where at least two same raw data/cloneset attributes match in values. The classifier aims to produce minimal-collision inexact clones. The DPPM value for RDCR is

$$1 - (\text{raw dataset size}/\text{published cloneset size})\,. \tag{3.10}$$

As the collisions decrease, the RDCR DPPM value increases. The published cloneset size is not always the same as the original cloneset. The used size is a statistic that is calculated using heuristics. We describe an example to demonstrate how to compute this value.

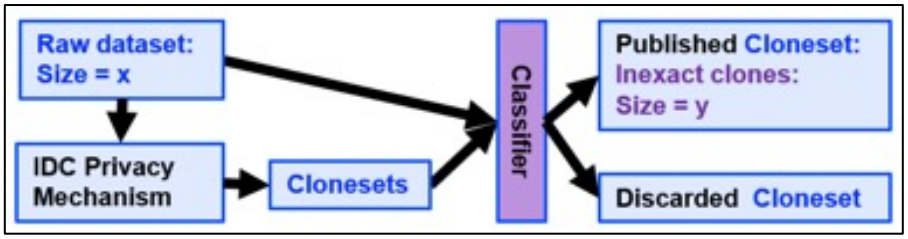


Figure 3.6: Diagram Showing the Origin of Raw Dataset Size and Published Cloneset Size.

The heuristic for calculating the RDCR DPPM value with example values is below.

1 Raw dataset size → A 6068 2 Cloneset size → B 163,836 3 Collision free statistic → C 52.81 % 4 Published cloneset size → D =B * C 86529 5 Collision free contribution → E = (1-(A/D) 0.93 6 RDCR DPPM value → 0.93

The raw data size is A. The cloneset size is B. The collision-free statistic is C. This value is further explained in section 4.2. In short, the IDC mechanism produced about 163,000 IC from about 6,000 raw data records. The classifier then produced about 86,000 collision-free IC from the 163,000 IC. This production was done 500 times in a batch size of 1. After 22 batches, the statistic of collision-free IC divided by total produced IC is assigned the value of C. The published cloneset size is D. The collision-free contribution is E, which results from $(1 - (A/D))$. For classifier summary, Figure 3.6 shows the use case of the classifier for IDC. The IDC process takes in the raw dataset and produces clonesets. For this work, each record in the raw dataset has exactly 27 clones. None of the 27 clones collide its source raw record but could collide with a different source raw record. To prevent collision with different source raw records, the classifier is implemented. The classifier takes in the cloneset and discards all inexact clones that match any raw data record in a specified form. The specified form is given by RAC in document section 3.2.3.

Chapter 4

PERFORMANCES OF IDC

The performances of IDC are expressed in terms of its DPPS score and its clone classifier selection percentage (CCSP). DPPS is based on the metric values of DPPM and CSSP is based on an estimated mean value and its margin of error. The raw dataset used for calculating DPPS and CSSP is presented below. The detailed calculations of DPPS and CCSP are then followed.

4.1 Raw Dataset Used to Evaluate IDC

The raw dataset used in this work is a biometrics data source from Kaggle.com. On Kaggle, the dataset name is ANSUR II. Other parameters are ANSUR II Survey Data, ANSUR 2 (2012), version 1 [26]. The extracted attributes used from the ANSURI II dataset were Weightlbs (weight: lb), Heightin: (height: in), waistcircumference (waist: mm), and buttoxcircumference (hip: mm). The dataset had 6,068 combined records. The PDS for this work is about 86,183 in value.

Three shifts were used, bigger, smaller, and much smaller. For each record, nine inexact clones were generated from each shift, resulting in 27 clones per record. This resulted in a closet of size 163,836. A note of importance: The raw data used by this work has six attributes. Two attributes are calculated or preserved attributes: body mass index (BMI) and waist to hip ratio (WHR). Four attributes are private: weight, height, waist, and hip. The clone values for preserved or calculated attributes can remain exposed, they do not have to be hidden. The calculated values; in this work; are dependent on the private attributes. The clone values of private attributes need to be hidden and are not to be exposed. An inexact clone that does not collide with

a raw data record is an inexact clone that has at most one attribute match with any raw data record.

To evaluate the performance of IDC, DPPS and CCSP are used. CCSP is explained in section 4.2 and DPPS is explained in section 4.3.

4.2 Clone Classifier Selection Percentage

The classifier is multifunctional. One of its functions is to remove any IC that might have an undesired resemblance to a raw data record. The resemblance is not needed as it might lead to the reidentification of a raw data record.

Figure 3.6 shows a use case of the classifier for IDC. The IDC process takes in the raw dataset and produces clonesets. The clonesets and the raw data are then passed through the classifier and clones that collide with a raw data record are discarded and those that do not collide are presented as collision-free inexact clones or classifier inexact clones or published cloneset.

For this work, each record in the raw dataset has exactly 27 clones. None of the 27 clonesets match its original record but could, be a different record. This is where the classifier weeds out clones that match some resemblance of another record. The criteria for weeding out clones that collide is presented in section 3.2.3. Weeding out heuristic is: "if x attributes of a raw data record match equivalent attribute in a clone, remove the clone from the cloneset to be published."

The performance of the classifier is interpretative. There are four numbers of importance, which in some way affect each other. These are, raw data size: A, cloneset size: B, collision-free clones size: C, and size of discarded clones: D. Per CCSP, the focus is placed on numbers B and C forming the percentage $100\% * C/B$. An interpretation is: given a cloneset, what percentage will be collision-free or what will be the CCSP? The continuation of this section outlines how this question is answered;

the estimation of the random variable CCSP. CCSP is used as the performance of the classifier and also the performance of IDC.

This work uses statistical analysis and heuristics measures via the classifier to calculate the CCSP performance. These measures are presented below.

The classifier implements a collision detection mechanism that discards IC that matches or collides with raw data in a pre-decided way. The interest is to standardize the percentage of collision-free IC produced by the classifier. This standardization was done using statistical analysis and heuristics. The heuristics and their supporting statistical analysis are presented below.

A run of IC dataset; 163K, was passed through a classifier, and about 52.81%; 86255, emerged as classifier IC dataset. The 52.81%, in this case, is the value for the random variable of interest, CCSP. A consistent percentage of the classifier IC dataset was needed to standardize the classifier's IC production performance. After several runs, Figure 4.1 illustrated a batch size of 500 runs statistically satisfying 52.81% as a candidate value for CCSP. The findings were the collision-free percentage was about 52.577% with a standard error of ± 0.000895 and a confidence level of 96%. Figure 4.2 shows the 95% confidence level finding is within two standard deviations of the batch mean. Further results on the data show the findings follow a normal distribution with 0.51 samples having a mean less than or equal to the batch means. The first standardization of the batch mean contains 0.684 of total samples, the second contains 0.96 of total samples and the third contains 0.99 of total samples.

Continuing the process of validating CCSP as an estimator, 22 batches were run and their findings were as follows. Figure 4.3 shows that CCSP approaches 52.584% as the cumulative batch count increases. Since the samples per cumulative batch were in the millions and the proportion of collision-free clones was increasing extremely slowly, 22 batches were considered sufficient to validate 52.584% as the CCSP estimate. This

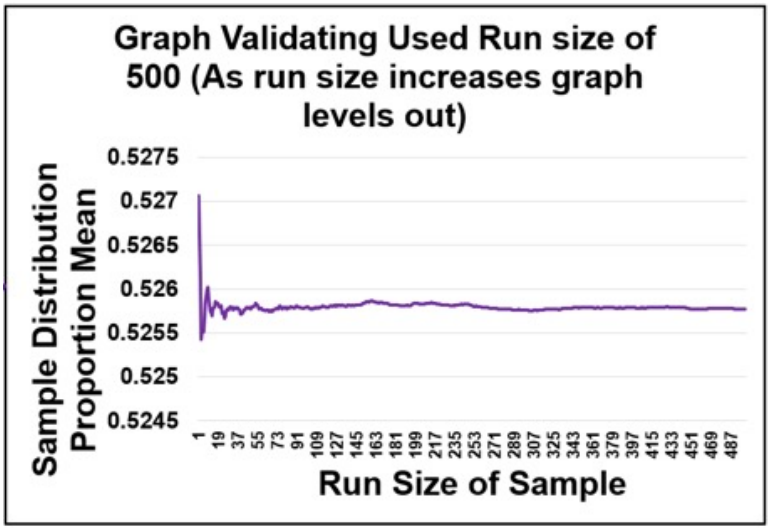


Figure 4.1: Graph Showing Sample Distribution Proposition Mean of 52.577 % Satisfied by the Run Size of 500.

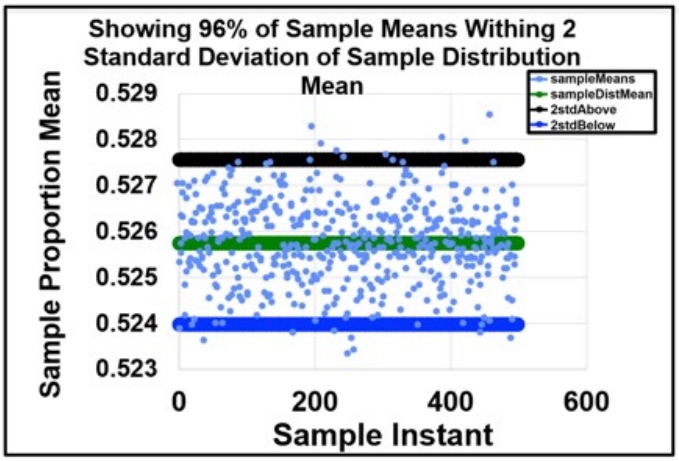


Figure 4.2: Graph Showing Samples in 500 Runs Follow a Normal Distribution Having 96% of the Sample Means Within Two Standard Deviations of the Sample Mean.

52.584% estimate had a standard error of ± 0.00005 and a confidence interval of 82%. Figure 4.4 shows the 82% confidence level finding is within 1 standard deviation of the mean for the 22 batches.

Figure 4.4 shows the 82% confidence level finding is with 1 standard deviation of the mean of the 22 batches. Further results on the data show the finding follows a normal distribution with 0.41 batches having a mean less than or equal to the batches'

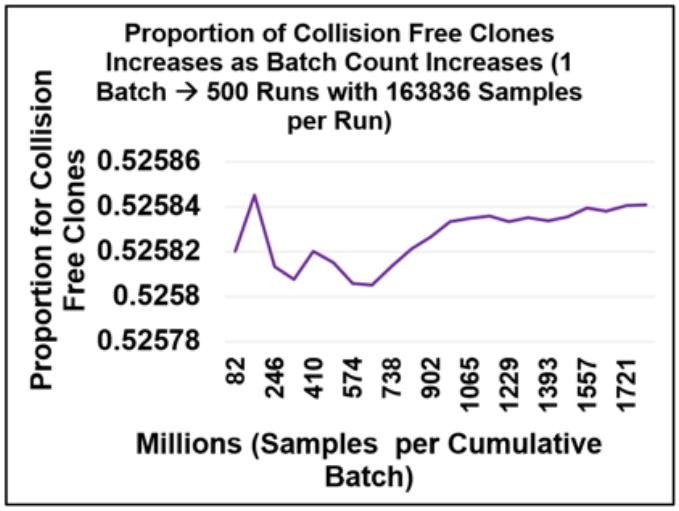


Figure 4.3: Plot Showing That as the Batch Count Increases the Proportion of Collision-free Clones Approaches 52.584%.

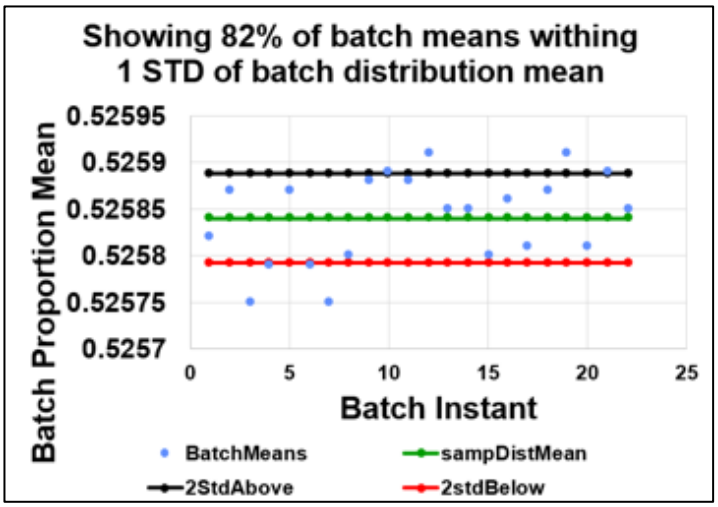


Figure 4.4: Graph Showing Batch Samples in Twenty-two Batch Runs Follow a Normal Distribution Having 82% of the Sample Means Within One Standard Deviation of the Batch Sample Mean.

mean. The first standard deviation of the batches mean contains 0.82 batches, the second and third contain all 22 batches.

In summary, the statistics and heuristics for the clone classifier find that, when clonesets are created, about 52.6% of them will be collision-free. This 52.6% is the value used as the clone classifier performance; CCSP and has a standard error of

± 0.0001. CCSP estimate is presented with a confidence level of 82% where each estimate is expected to be within 1 standard deviation of the actual CCSP. This work presents CCSP as a benchmark standardized metric for measuring data protection per data security. According to the heuristics of this work, this work has a CCSP value of 52.6%.

4.3 Differential Privacy Protection Score

DPPS has a description and a value. This work describes DPPS as a performance measure that provides a quantitative value for the protection level of a raw dataset. The demonstrated raw dataset is a biometrics dataset of approximately 6K records with 6 columns. DPPS is used as a performance measure for IDC. The percentage is calculated by dividing the IDC cumulative DPPM values by the ideal cumulative DPPM value;5, multiplied by 100 %. The demonstrated DPPS value for this work is 86.16%. The heuristic for calculating this 86.16 is presented later in this section.

The DPPM values are from the five DPPM metrics. These metrics and their heuristic values are as follows. 1) Shift Count: SC has a heuristic value of 1. The calculation is, shift count used; 3, divided by shift count available; 3, evaluating to 1.000. 2) Published Dataset Size: PDS has a heuristic value of 0.915. The calculation is from the constructed hill function mentioned in section 3.2.2 Equation 3.8 and which evaluates to 0.915. 3) Record Attribute Count: RAC has a heuristic value of 0.500. The calculation is from the formula, $X - (Y/Z)$. X is the constant 1. Y is the minimum raw data record private attributes not allowed to match in the classifier; for this work, 2. Z is the total raw data private attributes; for this work, 4. The formula $X - (Y/Z)$ is equal to $1 - (2/4)$ which evaluates to 0.500. 4) Record Clone Ratio: RCR has a heuristic value of 0.963. The calculation is from the formula, $(J - 1)/J$. This work creates 9 IC for each shift used, resulting in 27 IC for each raw data record.

J is the total IC for each raw data record before the classifier is run; for this work 27. The formula $(J-1)/J$ is equal to $(27-1)/27$ which equates to 0.963. 5) Raw Data Cloneset Ratio: RDCR has a heuristic value of 0.93. The calculation is from the formula $R-(H/K)$. R is the constant 1. H is the raw dataset records count; for this work, about 6000. K is the classifier IC count; for this work, about 86,000. The formula $R-(H/K)$ is equal to about $1-(6068/86183)$ and equates to 0.930. DPPM has some of Sum of 4.308. DPPS has a heuristic value of $100\% * (4.308/5)$ and equates to 86.16%.

Figure 3.2 shows the values for the above five mentioned DPPM metrics. Their actual and cumulative values are shown. Data that used DPPM protection are in purple and data that did not use the protection are in green. This table puts in place the basis for subsequent statistical analysis. Figure 3.2 shows the cumulative DPPM plots for data that is protected; in color purple and data that is not protected; in color green. Protected data has higher DPPM metric values. This group is intended to provide an initial perspective on data protection decision-making. Figure 3.2 shows the DPPM cumulative values of ideal, protected, and exposed data used in this work. The data in this table is used for statistical analysis in Figure 3.2. Figure 3.2 shows the DPPM protection level for the ideal versus the protected versus the exposed data. This graph is intended to provide more insight into one wanting to invest in doing further improvement to existing protected data.

In summary, DPPS is a composite percentage that represents the sum of each DPPM privacy mechanism contribution. The summation of all DPPM values, divided by 5, and multiplied by 100 percent, gives the value of DPPS. The evaluation of DPPS is a work in progress. It is to be interpreted as figurative obscurity produced by an IC dataset when the dataset is being observed by those it is not intended for. This work presents DPPS as a benchmark standardized metric for measuring data protection

per data security. According to the heuristics and statistical analysis of this work, this work has a DPPS value of 86.16%.

Chapter 5

COMPARING STATISTICS OF RAW DATA AND IDC DATA

Inexact data clones are inexact copies of raw data records. One can make inexact copies of data records and expect them to work just as well as the raw data.. Inexact clones are expected to do the job of the raw data even though they are not the raw data. Following are some expectations of inexact clones: protect intended parameters, preserve intended parameters, suppress raw data record reidentification, corroborate on parameters PDF, and corroborate on parameter statistics. The parameters used for illustration in this work are weight, height, BMI, waist, hip, and WHR. We aim to protect the parameters of weight, height, waist, and hip while protecting the BMI and WHR.

Protected means they are private and sensitive data and therefore cannot be published as an exact copy. To suffice their publication, their inexact copies which have their semantics is used instead. Parameters BMI and WHR are to be preserved. Preserved means carry the same calculations method. Preserved data are calculated data and in this work, calculated from the protected data. For example, the equation or formula for WHR = waist/him holds for raw data and inexact clones. The formulas for the calculations in both clones and raw data should be persistent. BMI is dependent on weight and height and WHR is dependent on waist and hip.

5.1 Probability Density Functions

The protection mechanism protects and preserves intended parameters. Where the protection mechanism protects, it suppresses raw data reidentification but still allows the intent of the raw data. The intent of the raw data means corroboration with

the raw data. Corroborates in this case means, have the same semantic intentions and functions but different semantics or appearance.

In this work, we compare the probability density functions (PDF) of attributes from two datasets; the raw dataset and the classifier IC dataset. Three equivalent attributes from each dataset are compared. These three attributes consist of one preserved attribute and two private attributes. The preserved attribute is dependent on the two included private attributes. For the other three attributes that were not demonstrated, they bear the same relationship, hence no need to include them. The graphical result indicated that the PDFs of the private parameters did not match and the PDFs of the preserved parameters matched.

This section compares the raw data and IC PDFs for the private attributes weight, height, and BMI. It shows that the private raw data attributes are protected by the IC attributes.

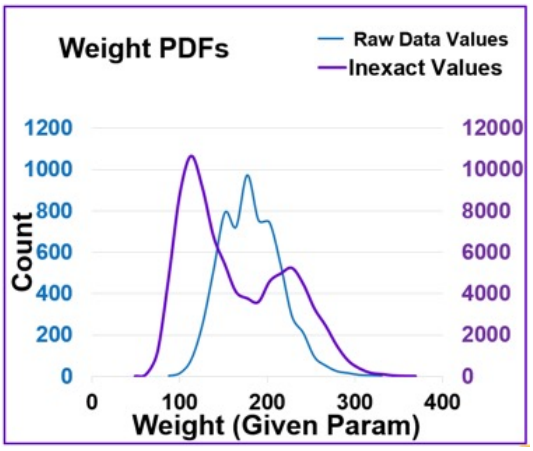


Figure 5.1: Probability Density Function (PDF) Plot of Weight Parameters of Raw Data Vs Inexact Clones.

Figure 5.1 shows the PDF plot of weight parameters for raw data vs inexact clones. The weight parameter is sensitive and is to be protected. It can be seen in the PDF that the weight parameter is protected as the plots do not match each other. The plots not matching, also means the values do not match accordingly,

hence one value is being protected. Parts of the shape in the plots bear some form of similarity. For example, the left side going up and the far right side going down. This similarity can be interpreted as underlying corroboration of data and not equality in data values. Not being equal means suppression of reidentification of private data is taking place. Similarity means some form of corroboration is taking place. This PDF shows protection of data and Section 5.2 shows corroboration of data.

The middle section of the IC PDF is below the middle section of the raw data PDF. This indicates for weight values in this region, the IC dataset had lesser than the raw dataset. A reason for the IC dataset having lesser data is because the classifier removed raw-data-compromising IC data records. This removal function of the classifier facilitates suppressing raw data reidentification via IC. It also indicates the attempt to protect the private parameter. To ensure that this PDF observation was no coincidence. The same finding was found in the other three private parameters, height, hip, and waist. Below is the analysis of the PDFs for the height parameter.

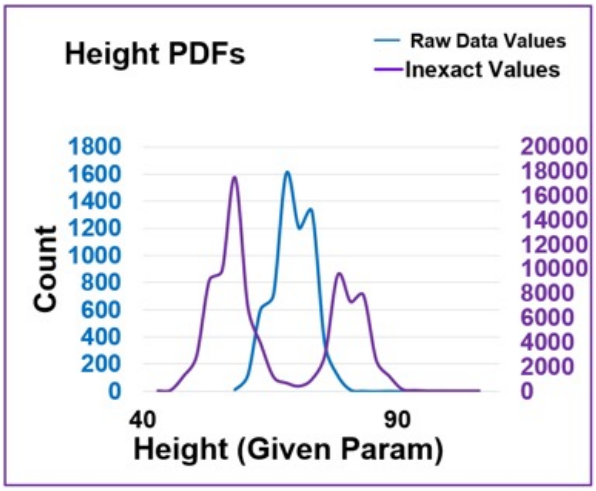


Figure 5.2: Probability Density Function Plot (PDF) of Height Parameters of Raw Data Vs Inexact Clones.

Figure 5.2 shows the PDF plot of height parameters for raw data vs inexact clones. The height parameter is sensitive and is to be protected. As in Figure 5.1, it

can be seen in Figure 5.2 that the height parameter is protected as the PDF plots do not match each other. The plots not matching, also means the values do not match accordingly and hence one distribution is protecting the other. The shape of the right section of the IC plot bears some similarities to the raw data plot. This similarity can be interpreted as underlying corroboration of data where the occurrence in the IC plot is a shifted version of the raw dataset that does not share the raw dataset values. The version is in a shifted mode because it is in a protecting mode, doing what it is supposed to do; protect.

Like in Figure 5.1, the middle section of the IC PDF is below the middle section of the raw data PDF. This indicates for height values in this region, the IC dataset had lesser than the raw dataset. Like Figure 5.1, the reason for the IC dataset having lesser data is because the classifier removed raw-data-compromising IC data records. This removal function of the classifier facilitates suppressing raw data reidentification via IC. It also indicates the attempt to protect the private parameter. According to the PDF results of Figure 5.1 and Figure 5.2, it is concluded that the IDC protects private parameters.

Where the protection mechanism preservers, it does not care about the replication or compromising of preserved raw data attribute values. This means that some of the raw data values can match the IC values. It is to be remembered that the classifier does not check for matching preserved parameter values.

Figure 5.3 shows the PDF plot of the BMI parameter of raw data vs inexact clones. The BMI is a preserved parameter. This means that its values are allowed to match those of the raw data. The plots show that they are almost identical. They have the same trend and some parts even overlap. This PDF graph shows that the IC dataset matches the raw dataset and hence the datasets are preserved using the same mechanism. In this case, they use the mechanism of the same formula. The BMIs of

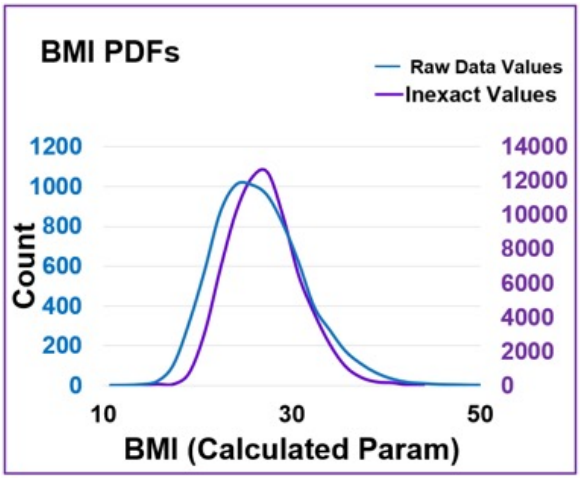


Figure 5.3: Probability Density Function Plot (PDF) of Body Mass Index (BMI) Parameter of Raw Data Vs Inexact Clones.

both datasets use the same formula. The PDF for the preserved WHR parameter; not shown here, also shows matching plots. According to the PDF results of Figure 5.3, it is concluded that the IDC preserves preserved parameters.

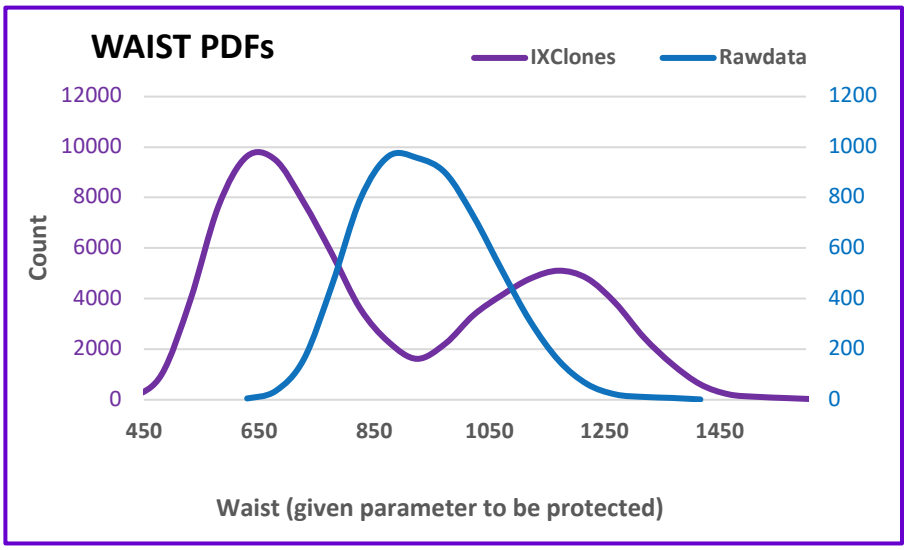


Figure 5.4: Probability Density Function Plot (PDF) of Waist Parameter of Raw Data Vs Inexact Clones.

Figure 5.4 and 5.5 are example protected attributes plots. Their explanations are similar to figures 5.1 and 5.2 .

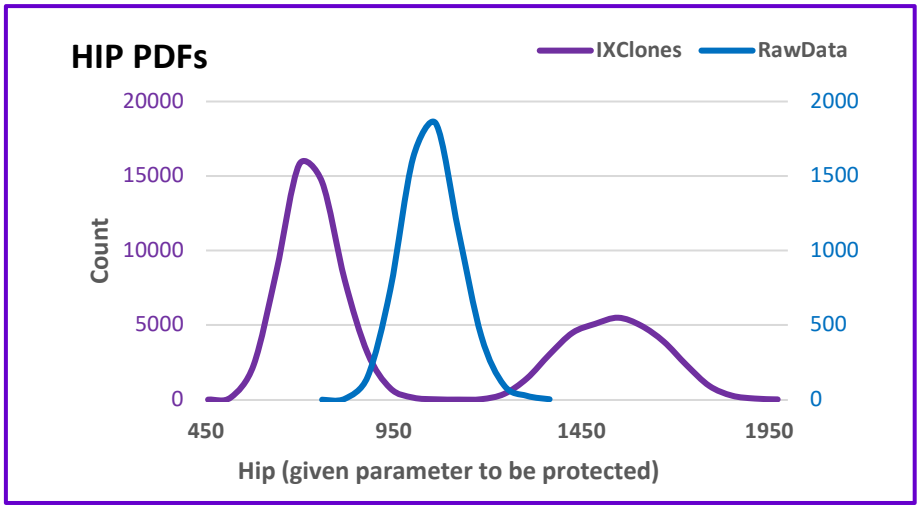


Figure 5.5: Probability Density Function Plot PDF) of Hip Parameter of Raw Data Vs Inexact Clones.

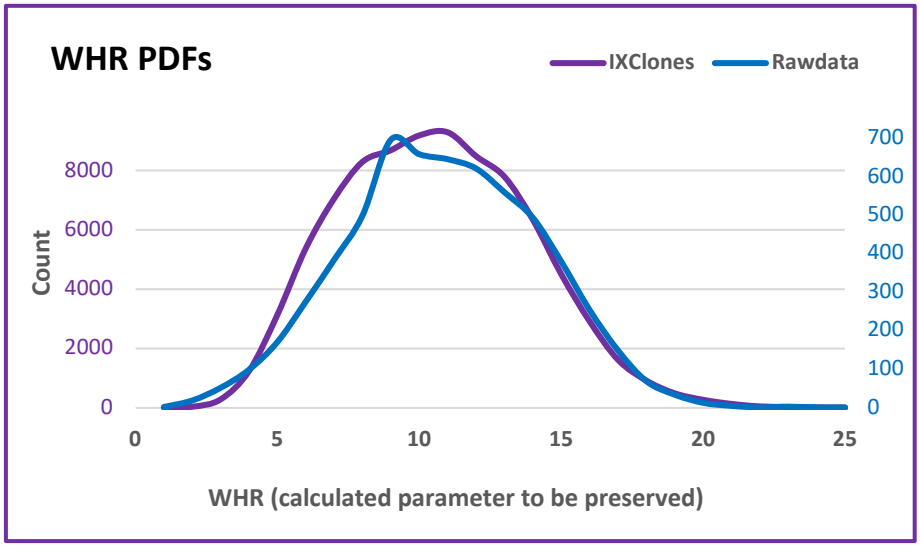


Figure 5.6: Probability Density Function Plot (PDF) of Waist-hip Ratio (WHR) Parameter of Raw Data Vs Inexact Clones.

Figure 5.6 is an example protected attribute plot. Its explanation is similar similar to figure 5.3 .

5.2 Comparing Raw Data vs Cloneset Statistics

The data record used in this work has an attribute count of 6: weight, height, BMI, waist, hip, and WHR.

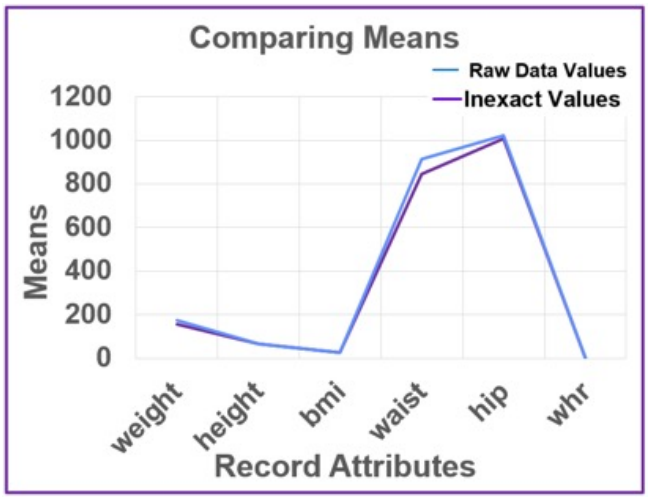


Figure 5.7: Diagram Showing Comparisons of Raw Data and Inexact Values via Attributes Using the Statistic of the Mean for Dataset BMIWHR

Figure 5.7 shows the comparisons of raw data and inexact clones values via attribute using the statistic of the mean. The mean is one of the first static that indicates some form of comparison. Since the means closely follow the same trend, then both sets of data carry similar properties and one corroborates the other. The plot shows that the values are very close. The calculated/preserved values: BMI and WRH are almost identical, and the given/protected values: weight, height, waist, and hip are very close. From the observation of Figure 5.7, one can say that the inexact clones satisfactorily represent the raw data record. With that said, one might want more distinct evidence that shows the difference between preserved and protected parameters during data corroboration. Figure 5.8 showing the kurtosis plots makes

an answer to this clearer.

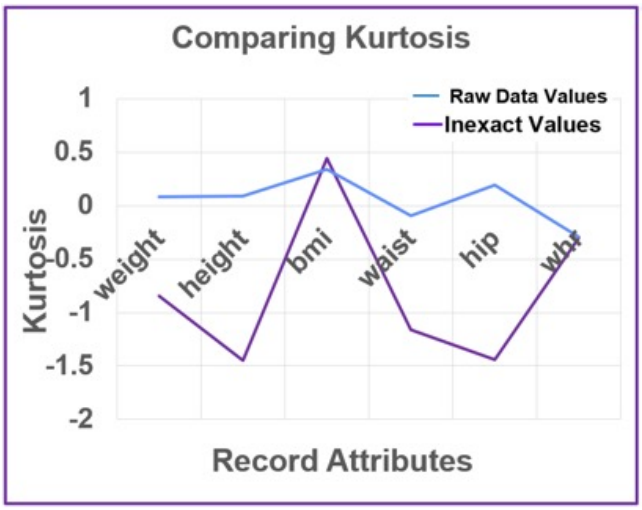


Figure 5.8: Diagram Showing Comparisons of Raw Data and Inexact Values via Attributes Using the Statistic of Kurtosis for Dataset BMIWHR.

Figure 5.8 shows a comparison of raw data and inexact clones values via attributes using the statistic of kurtosis. The kurtosis statistic is used in this work to show comparisons between preserved and protected attributes of different distributions. In Figure 5.8, the kurtosis plots show that corresponding preserved attributes have almost identical values. The BMI is a preserved attribute and its values are almost identical in the plots. The WHR is a preserved attribute and its values are also, almost identical in the plots. This shows that the BMI and WHR values of both distributions are similar and corroborate each other. The kurtosis plots of different distributions show that corresponding protected attributes should appear apparat in the plots. This appearing apart is seen in Figure 5.8 where the weights, heights, waists, and hips do not meet at a point but are satisfactorily apart from each other. This being apart also means that the weight, heights, waist, and hip values of both distributions are similar and corroborate each other. From the observation of Figure 5.8, one can say that the inexact clones satisfactorily represent the raw data records concerning isolating preserved and protected attributes.

It is difficult to reidentify protected attributes using referential analysis. Figure 5.8 shows that the inexact clone distribution protects sensitive parameters and preserves calculated data. Overall, the kurtosis plots show that one distribution corroborates the other in terms of protecting and preserving parameters.

Sometimes another statistic can be needed to provide a second opinion. The variance statistic can provide some insights to assist the findings of the kurtosis plots. The variance statistic will follow the kurtosis in terms of preserved parameters. For protected parameters, results vary, they can be close together or far apart. Figure 5.9 and Figure 5.10 show the protected parameters not having the same trend and the preserved parameter having almost identical values.

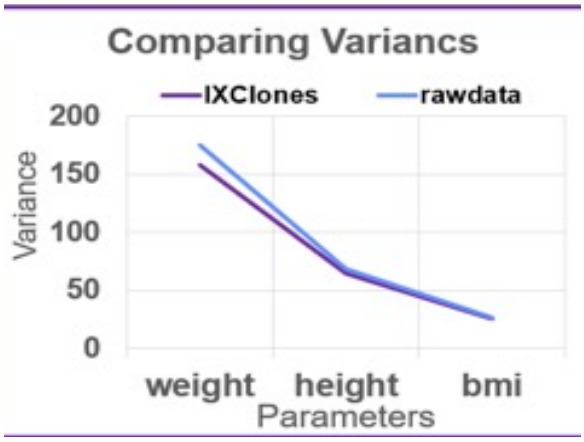


Figure 5.9: Preserved Parameter (BMI). Protected Parameters (Weight, Height). Diagram Showing Comparisons of Preserved Parameters Values and Protected Parameter Values for Dataset BMIWHR.

Figures 5.11, 5.12, 5.13, 5.14, 5.15 and 5.16 shows example plots in diagnostic report for dataset APEIndex. Dataset BMIWHR has similar plots in its diagnostic report. Dataset BMIWHR has six variables and dataset APE index has 3 variables. Although both dataset have different variable count their mean plots corroborate in validating both datasets as valid substitutes for their respective raw data. This is concluded because both mean plots show respective means following the same trend.

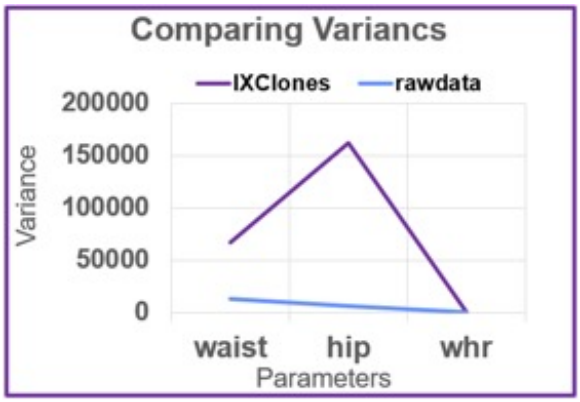


Figure 5.10: Preserved Parameter (Whr). Protected Parameters (Waist, Hip). Diagram Showing Comparisons of Preserved Parameters Values and Protected Parameter Values for Dataset BMIWHR.

For a in IDC dataset to be a good replica its means trend is to follow the same trend as the raw data' mean trend. Figure 5.11 shows the mean plots for dataset APEIndex and figure 5.7 shows the mean plots for dataset BMIWHR.

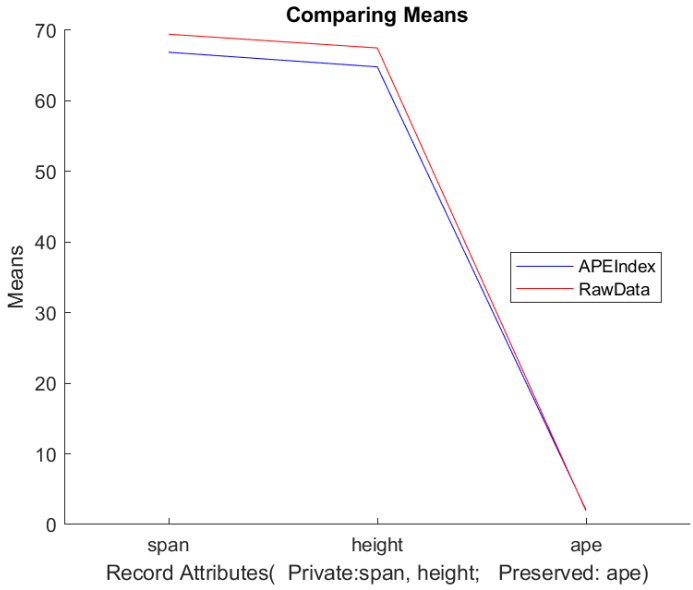


Figure 5.11: Diagram Showing Comparisons of Raw Data and Inexact Values via Attributes Using the Statistic of the Mean for Dataset APEIndex

The kurtosis plots of both datasets corroborate in validating both datasets as valid substitutes for their respective raw data. This is concluded because both kurtosis plots

show respective kurtosis patterns. Figure 5.8 is a reduced kurtosis whereby its normal section is removed. This form makes more visible, preserved and private variables. This plot sows the preserved variables are almost identical in kurtosis value and the protected variables are more part. Figure 5.12 is a regular kurtosis plot. It has the normal form in it and hence its values are all positive. Since the APEIndex has only one dependent variable; the preserved variable, we can say this kurtosis plot shows the preserved variables as the ones closest to each other. For a in IDC dataset to be a good replica its kurtosis plots is to show preserved variables being almost identical or being the closest together.

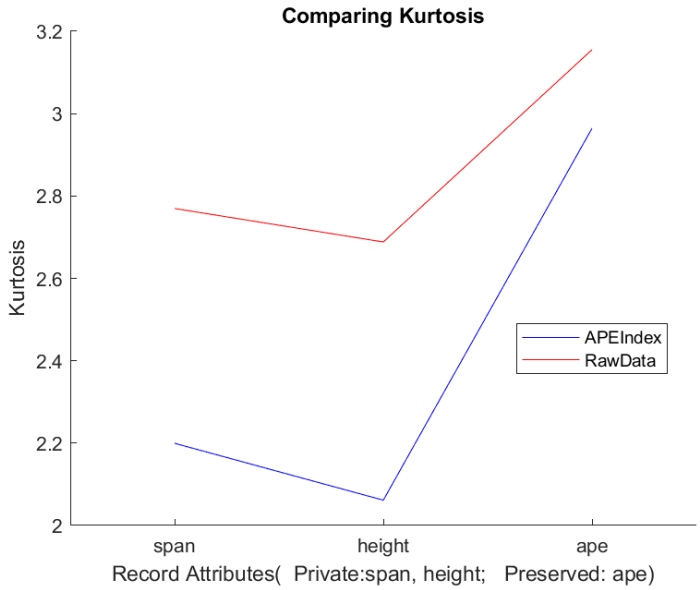


Figure 5.12: Diagram Showing Comparisons of Raw Data and Inexact Values via Attributes Using the Statistic of Kurtosis for Dataset APEIndex

Figure 5.13 is the variance plot for dataset APEIndex and figures 5.10 and 5.9 are the variance plots for dataset BMIWHR. In all three cases, the variance plots corroborate in validating both datasets as valid substitutes for their respective raw data. This is concluded because all variance plots show the preserved variables as being the closest together. The variance plots show the private variables as being

further apart. For a in IDC dataset to be a good replica its variances patterns are to be as described.

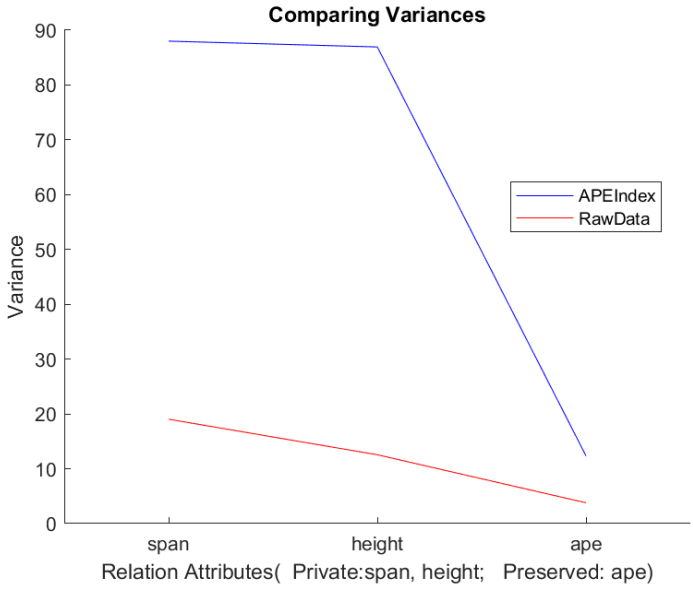


Figure 5.13: Preserved Parameter (Ape). Protected Parameters (Span, Height). Diagram Showing Comparisons of Preserved Parameters Values and Protected Parameter Values for Dataset APEIndex

Figures 5.14, 5.15 and 5.16 are verification plots for dataset APEIndex. For ease, only the BMI relation plots will be referenced from the BMIWHR dataset. figures 5.1, 5.2 and 5.3 are verification plots for the BMIWHR dataset.

Figure 5.14, shows a probability plot verifying the APEIndex data variable span as a private variable. Figure 5.1, shows a count plot verifying the BMIWHR data variable weight as a private variable.

Figure 5.15, shows a probability plot verifying the APEIndex data variable height as a private variable. Figure 5.2, shows a count plot verifying the BMIWHR data variable height as a private variable.

Figure 5.16, shows a probability plot verifying the APEIndex data variable ape as a preserved variable. Figure 5.3, shows a count plot verifying the BMIWHR data

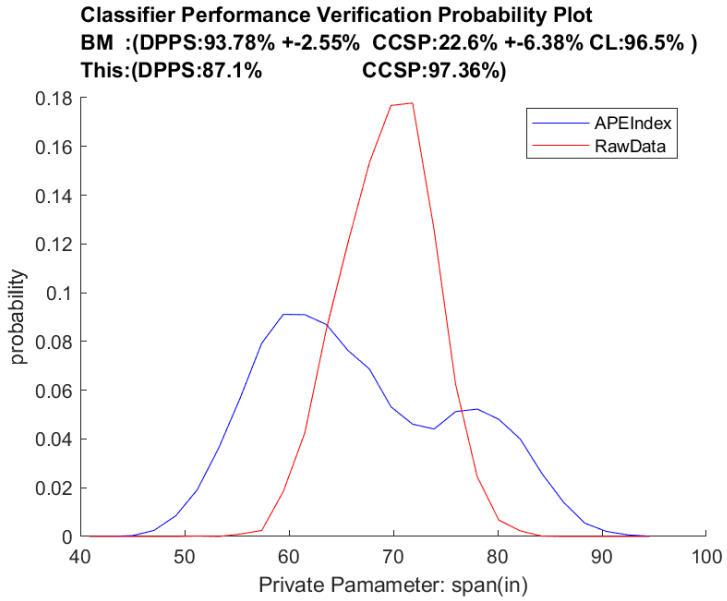


Figure 5.14: Diagram Showing Probability Plots of Attribute Span in Dataset APEIndex

variable BMI as a preserved variable. The count plot and the probability plot can be compared to be the same as their sample size is very large. This can be verified where corresponding charts follow the same trend. This means, without labels, the plots on each graph could be substituted for each other. One could say the count charts and the probability charts are ratios of each other.

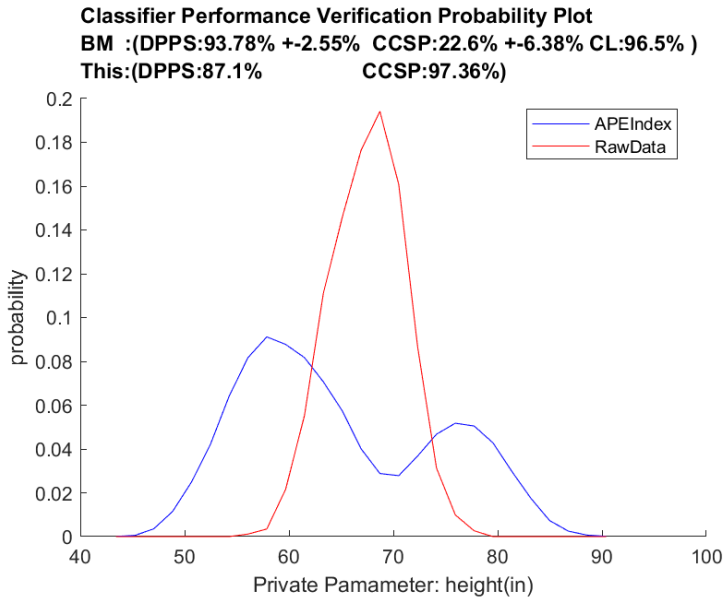


Figure 5.15: Diagram showing probability plots of attribute height in dataset APEIndex

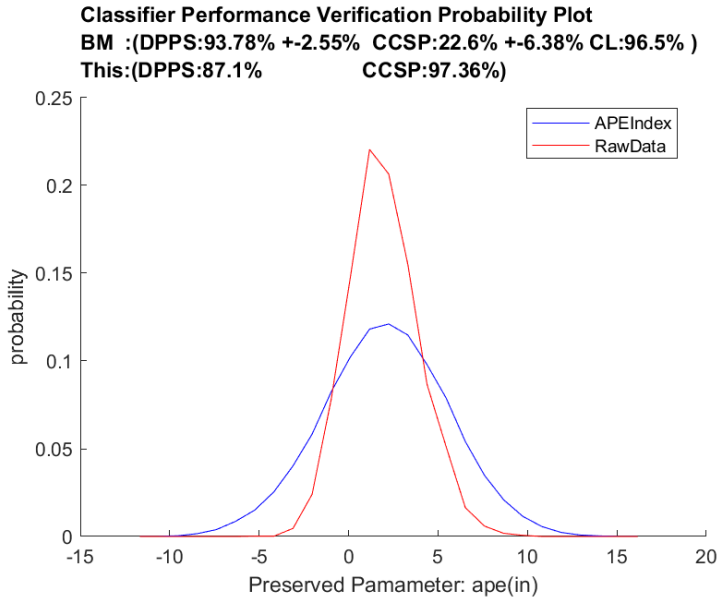


Figure 5.16: Diagram Showing Probability Plots of Attribute Ape in Dataset APEIndex

Chapter 6

DIFFERENTIAL PRIVACY PROTECTION ARCHITECTURE

6.1 Introduction

DPPA; per this dissertation, is a set of specifications which when implemented produces an umbrella infrastructure. The infrastructure facilitates IDC data differential privacy security products and their respective services. The umbrella infrastructure has a hub component which takes in datasets currently not in DPPA and annex them into the DPPA IDC security platform. Platform integration is via DPPA engineering services in the form of DPPA Annexation Requirements, Annexation, Integration, Statistical Validations of Metrics, Diagnostic Report Validations, Automation Response to Variability and DPPA Roster Debut. The implementation of DPPA was inspired from the existence of IDC security products that were standalone with disjoint and manual operations; lacking cohesive services. DPPA provides automation which produce published IDC datasets with their corresponding diagnostic reports.

DPPA was implemented as a proof of concept for its automation capabilities. The automation goal is to facilitate an environment that repeatedly produce IDC datasets and their respective diagnostic reports. For further automation capabilities, automation response to variability per raw dataset size and maxMatch were implemented. maxMatch is the raw dataset column replication count in published data. This means each published record, can replicate maximum, maxMatch columns of any raw data record. Our realized instant of DPPA is supervised; meaning our rendition is used as a testing or prototype facility that accepts standalone IDC security products. Theses standalone products are functional with no cohesive automation and no diagnostic

reports.

The word umbrella targets the uniting of three main areas, dataset domain, DPP security mechanism and security mechanism versioning. For dataset domain; DPPA can accept datasets from different domains. Example dataset domain include finance, marketing, warehouse, and human biometrics. This work used data from the domain of human biometrics. For DPP security mechanism; the security mechanism can be of different types. Example security mechanisms are IDC, Information Entropy, and randomization. Our implemented DPPA instant accepts only IDC. This exclusive implementation suits our prototype aim. For versioning; this means you can have the same security mechanism with different specializations. Some specializations are raw dataset size, dataset relation count, and relation equations. Our current IDC version specialized in raw dataset sizes of 1K up to 6,000 $\pm$ 200. This size restriction is a precaution since our test datasets ranges between near 1K up to near 6K. Another version of IDC could take raw dataset sizes that are between 6,000 and 20,000 or between 20,000 up to big data. To handle these larger dataset sizes, code optimization; resulting in a different version of IDC would most likely be necessary.

DPPA was implemented using software. The immediate section below explains some general software properties of DPPA. These properties are called extended specifications. The rest of this chapter outlines DPPA architectural layout of services, automation verification services and automation validation services, this work as a benchmark, limitations of DPPA and use cases of DPPA.

6.1.1 Extended Specifications

We implemented the specifications of DPPA using software. During our implementation, we deferred Software implementation dexterity as a priority, and hence we are mentioning some implementation properties which we think can bring clarity and

understanding to DPPA implementations in general. We created IDC as a theory. DPPA incorporates IDC. We used software to realize DPPA as a software tool chain platform which showcases the IDC theory as a product. Since DPPA is a set of specifications, its software implementation is independent of programming language and software functionalities. We therefore defined and incorporate the following software engineering properties as extended DPPA specification. The aim of this specification extension is to maintain general software implementations properties that we found useful for software comparisons. These properties are: modular, scalable, portable, application interaction, device access, execution system, and archiving. We implemented only those extensions that suited our prototyping needs. The rest of this section further explains each property, its specification significance, and our sufficient renditions of it.

Creating modules is about sectioning DPPA into parts or sub areas. For example codes, operations, files, or storage can be modularized. Modules are significant as they make orchestrating, searching and organizing easy and more efficient. From an architectural layout perspective, we modularized our DPPA Automation Engine into four sub engines or sub modules; IDC mechanism, classifier, figstoImage and reportGen. From a software engineering perspective, our DPPA Automation Engine module contains 14 directories and 16 files before execution. After execution, the module contains 14 directories and 42 files. We next give a summary of how we use the files and directory as modules. We used 6 modules. File1, this is the raw dataset from which published data is produced. This file was our 6K human biometrics sample raw dataset. This file is also called the incoming dataset file. File2, this is an Excel file with File1's data characterization index. The index is an integer which serves as File1's pass to DPPA processing. Our file 2 had the number 1003 in it. For start we just wanted a big value; 1003 is a random number that sufficed. Characterization indexes are unique

to DPPA. File3, this is the DPPA master execution file; an orchestration file. The content of this file is further explained in the Integration section in the Automation Verification Services sub chapter. File4, this is the DP security execution file. This file creates protected published data and other data for performance verifications. This file creates our IDC dataset. File5, this file generates diagnostic reports. This is DPPA report generation engine. Most plots, charts and notes used in the report are generated by dependent files. For example, we call a function to generate the plots and save them. File5 then add these saved plots to the report. File6, this is our repository. Our repository file is a folder. It is our DPPA processing directory. This arrangement suited our purpose. More importantly, it serves as a major motivating factor to implement the specifications of DPPA. Our modules arrangement suited our purpose. A minimal functional DPPA needs these five modules in some form.

Scalable software are dynamic in some areas. This means their functionalities include automatic adjustment as a response. This also means the software responds to variability. For example, when the size of our raw dataset changes, DPPA automation engine scales to facilitate the change. DPPA also scales to accommodate a new maxMatch value. Our test raw dataset sizes were near 1K up to near 6K. We used two raw datasets. Dataset BMIWHR had maxMatch values 0 up to 4 and dataset APEIndex had maxMatch value 0 up to 3. Our DPPA scaling focus was on raw dataset row size and maxMatch value. This scaling arrangement suited our purpose. A minimal functional DPPA needs these two scaling variables.

Portable software can be executed on different software platforms and or different operating systems. Portability is important especially when we need to meet the needs of customers with different operating systems. The portable property of DPPA is tied to the operating systems in which it can operate. Our DPPA currently operates on the Linux operating system and the windows operating system. A minimal functional

DPPA needs to operate on one of these two operating systems; windows and Linux. We strongly recommend operations on both as one will experience the difference of what each system has to offer. One operating system might be more friendly to legacy versions of programming languages.

We consider the application interaction status property as the methods for initiating processing of the DPPA master execution file. This is significant as it might be necessary to run DPPA via remote, gestures, audio, touch screen, GUI or command line. We currently interact with DPPA via MATLAB command line and MATLAB GUI. A minimal functional DPPA needs to allow interaction via MATLAB command line. This command line interaction form was sufficient for our prototype needs.

Per device access, this property tells us what devices can be used to access DPPA execution file. This property is significant as customers can choose from a variety of platforms, for example: mobile devices, laptops, desktops, network, and terminals. Our DPPA accessing devices were laptop and desktop. A minimal functional DPPA needs to accommodate device access of type laptop and desktop. These access types were sufficient for our prototype needs.

Per execution system type, this property addresses the locations of DPPA's modules while operating as a cohesive functional unit. This executing system type is important as customers might want to run a particular DPPA module on a different computer. For example one might want to run the classifier unit or each encryption or IDC unit, on a different computer or cluster. A possible reason is that, the code for each unit is optimized to exploit certain hardware functionalities. A case is were the code for the classifier unit might be optimized for speed while the code for the encryption unit might be optimized for memory. Where the code units run on different machines and the DPPA master file run on a different unit, would be an example of a distributed execution system. Some other execution system types are, central,

online, network or any combination; hybrid. Our DPPA used a centralized system. All our modules and execution units run on the same machine. A minimal functional DPPA needs to allow all execution unit to run on the same machine. This execution system style was sufficient for our prototype needs.

The archiving property is tied to the accessing of DPPA resources and records. Archiving is about the provisioning for storage and access to past records. This property is important as DPPA might be expected and most likely will be expected to keep a copy of all the published datasets. Another reason is that customers may request copies of past published datasets. An example case is where published datasets are used for testing, training and evaluations by graduate students of a class. The same datasets might be repeatedly requested for different groups of graduate in the same class. Our DPPA input, processing and output files are archived in the same DPPA execution folder. Our DPPA folder has modules arranged in a directory tree consisting of respective folders. This arrangement suited our purpose. Per provisioning for archiving storage, this property is referring to ample, huge, extensive, or extremely large storage facility. This could require the usage of a data center. Per access of records, these records are to follow these properties: repeatability, versioning, dataset domain and dataset characterization. For us the repeatability property requires that the stored record can be satisfactorily regenerated on demand. All regenerated records should be comparable. We used versioning as being able to request the same dataset with different DPPM values or different protection preference. We used dataset domain as having the raw dataset belonging to different disciplines such as biology, finance, geography and so on. We used dataset characterization as having datasets with varied number of column and also varied relations among the columns. A minimal functional DPPA needs to allow archiving in at least one central directory. This one central directory archiving was sufficient for our prototype needs.

6.2 DPPA's Architectural Layout of Services, Requirements and Automation Path

Figure 6.1 shows DPPA's architectural layout of services, requirements and automation path. The pre-automation path consists of the requirements and services. An engineering purpose of DPPA is to orchestrate and prototype the automation of IDC published datasets and their respective diagnostic reports. In this layout we give an overview of DPPA operations in fulfilling this purpose. We used the order of the key as our explanation guide.

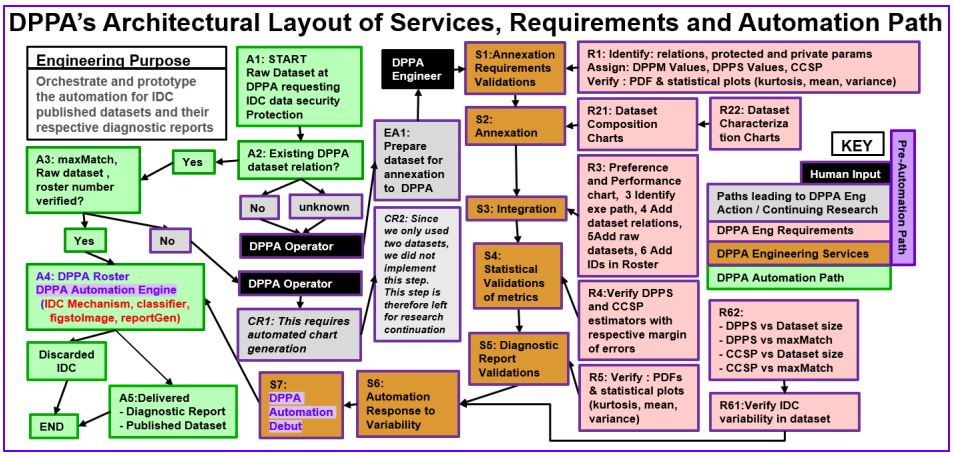


Figure 6.1: DPPA's Architectural Layout of Services, Requirements and Automation Path

When referencing to figure 6.1 , the square boxes are nodes. The writings inside the nodes are labels. Most labels starts with a letter followed by a number. The key node shows two main nodes, one with a green background; a green node and the other with a purple background; purple node. The green node has label, 'DPPA Automation Path', indicating all green nodes are on the DPPA automation path. The purple node has label, 'Pre-Automation Path', indicating all nodes beside the purple node are on the DPPA pre-automation path. The nodes on the pre-automation path has background color black, gray, rose and orange with respective names; Human Input, 'Paths leading to DPPA Eng Action/ Continuing Research', 'DPPA Eng Requirements' and DPPA

Engineering Services. The automation path shows the processing of a raw dataset from the time it enters the DPPA umbrella for encryption to the point when its IDC published dataset and its diagnostic reports are produced. The pre-automation path explains all the services that are applied to get a non-DPPA raw dataset into DPPA for future automation path usage. For the rest of this section, we will first cover the nodes on the automation path followed by the nodes on the pre automation path.

In figure 6.1 , the green nodes represent DPPA Automation Path. The labels in these nodes with the letter A followed by a respective number. A represents automation path. For reference, one purpose of DPPA is to automate the standalone and disjoint processes of existing IDC security mechanisms. After an IDC security mechanism is annexed into DPPA, incoming datasets requesting its encryption mechanism will follow the automation route to generate published datasets and respective diagnostic reports. DPPA's automation path is as follows . Node A1 is the start node. This node indicates the step when a raw dataset enters DPPA and is requesting IDC data security protection. The dataset is acknowledge and goes on to the next automation node A2. Node A2 test if the relations of the dataset exist in DPPA. Exist in DPPA means the dataset has at least completed service S1. If the relationship does not exist, the request goes to a DPPA operator. If the relations exist in DPPA then the dataset continues on the automation path along the yes node to node A3. Node A3 validates the input to the DPPA roster: maxMatch, raw dataset and roster ID. If the dataset's information is not in the roster the request is passed to a DPPA operator. If this test passes the dataset is moved on to node A4 the DPPA roster, DPPA automation engine and classifier. Node A4 takes in IDC encryption requests consisting of raw dataset, maxMatch and roster number. Respective inputs have already been verified in the master roster file and the dataset is now ready for automation processing. The DPPA automation engine takes the A4 input processes

them. The DPPA automation engine has four sub engines; IDC mechanism, classifier, figstoImage and reportGen. The sub engines operates in sequence. The first processing takes place in the IDC mechanism engine which generates diagnostic datasets and IDC clones; called cloneset. The second processing takes place in the classifier engine, which generates diagnostic datasets, published dataset and discarded dataset. The third processing takes place in the figstoImage engine, which generates diagnostic datasets and converts figures to images. The forth processing takes place in the reportGen engine which generated diagnostic reports. The flagship purpose of the DPPA automation engine is to produce diagnostic reports and published datasets. Discarded IDC is a bi-product dataset of the automation engine. When processing is completed by the automation engine, A4 passes a deliverable to node A5. Node A5 receives deliverable from node A4. These deliverable are requests made by node A1. These deliverable consist of a diagnostic report and an IDC published dataset.

We classified the nodes on DPPA's pre automation path as requirements and services. The nodes with orange backgrounds provides services and those with rose backgrounds contains the requirements or specifications for the services.

In figure 6.1 , the black nodes represent Human inputs. The Human Inputs are roles performed by DPPA Engineers or DPPA Operators. DPPA Engineers are usually holders of advance engineering knowledge that can convert DPPA engineering requirements into DPPA services. All DPPA engineers are DPPA Operators. DPPA Operators are skilled workers that can successfully manage a DPPA operation. The DPPA engineer do verifications and validations of specifications.

In figure 6.1 , the gray nodes represent 'Path leading to DPPA Engineering Action or Research Continuation' (DEARC). The DEARC represents alternatives paths taken, when automation path is forwarded for engineering action or continuing research. There are two paths that have gray backgrounds. The one that leads to the

Data Engineer shows the path completed by this research work, node EA1. EA1 is reached when a dataset relation is being encrypted for the first time. This is how all dataset relations gets to be in DPPA. The other path leads to continuing research; a path that is not completed by this work, nodes CR1 and CR2. Our completed work has two IDC instances, BMIWHR and APEIndes. Since we had two, it was not necessary to check for existing DPPA dataset relation. We however included this checking option as continuing research. This check involves writing a characterization search engine. This was not necessary for our prototype.

In figure 6.1 , the rose nodes represent DPPA Engineering Requirements and have labels starting with an R followed by a respective number. R represents roles or skill sets. The purpose of a role is to do verification or provide evidence of some kind. All roles will be explained in details later in this chapter. For now we will give a brief summary of each role. The DPPA Engineering Requirements consists of specifications for completing an engineering tasks. These specifications are arranged into modules for easy identification and pipelining. The intent is that one module would represent a role of a DPPA engineer. A DPPA engineer can have multi roles. Currently there are eight distinct roles.

Role R1 consist of the following: (Identify: relations, protected and private parameters; Assign: DPPM values, DPPS values, CCSP values; Verify : statistical plots, PDFs, kurtosis, means, variances). Role R21 consist of the following: (Create Dataset Composition Charts). Role R22 consist of the following:(Create Dataset Characterization Charts). Role R3 consist of the following: (Index Chart, Performance chart, Identify execution path, Add dataset relations, Add raw datasets and Add IDs in Roster). Role 41 consist of the following: (Verify DPPS and CCSP estimators and their respective margin of errors) Role R51 consists of the following: (Perform for metrics verification PDFs and statistical plots of type kurtosis, mean,

variance). Role R61 consists of the following: (Verify IDC variability in dataset). Role R62 consists of the following: (Perform for automation clarifications: DPPS vs Dataset size, DPPS vs maxMatch, CCSP vs Dataset size, CCSP vs maxMatch).

In figure 6.1 , the orange nodes represent DPPA Engineering Services and have labels starting with an S followed by a respective number. S represents appointment services. Appointees of appointment services do verifications and validations of specifications. All services will be explained in details later in this chapter. For now we will give a brief summary of each service. DPPA Engineering Services are like engineering positions, job positions, milestones, a software task or a pipeline step. Engineering services can be seen as a noun or as a adjective. All DPPA Engineers are certify to provide at least one DPPA Engineering service. Currently there are seven distinct services. The services are arranged in sequential order forming a pipeline. It is recommend to carry out services in the sequential order of S1, S2, S3, S4, S5, S6 and lastly S7.

S1 is about annexation requirements validation. S1 verifies and validates R1. S2 is about annexation. S2 verifies and validates R21 and R22. S3 is about Integration. S3 verifies and validates R31. S4 is about statistical validations of metrics. S4 verifies and validates R4. S5 is about diagnostic report validation. S5 verifies and validates R51. S6 is about variability testing. S6 verifies and validates R61 and R62. S7 is about DPPA automation debut. S7 grants DPPA automation engine access to the DPPA roster items. This service makes a dataset relations available for public encryption for the first time. When all services are completed the dataset's relations are activated for automation engine access.

6.3 Automation Verification Services

A general purpose of DPPA is to extend the services offered by stand alone DP security mechanisms by introducing cohesive services and automation. DPPA provides specifications that allows a stand alone service to be converted to a DPPA service. There are seven specifications groups; collectively called automation verification services. The automation verification service has two sections; appointments and roles. The appointments are called DPPA engineering services and are occupied by DPPA engineers. The roles are called DPPA engineering Requirements and consists of specifications. The seven groups works in sequence and forms a pipeline does the stand alone conversion process. Specification groups are also synonyms. Each specification group is DPPA Engineer position or milestone. The seven automation verification services are as follows: annexation requirement validations, annexation, integration, statistical validation of metrics diagnostic report validations automation response to variability and DPPA roster debut. The seven appointment services and their respective roles are explained below using our BMIWHR dataset.

6.3.1 Annexation Requirements Validations

The annexation requirement validation process is the initial step in validating a raw dataset and its components as meeting the DPPA IDC conversion requirements. When the dataset meets all the annexation requirements specifications, it is classified having DPPA candidacy 1; meaning it is ready for DPPA annexation process. Annexation is not necessarily immediate after candidacy 1. The candidacy 1 process is outlined in the below table using dataset BMIWHR as example.

In the table, the first column shows the row number, the second column shows the requirement specifications, the third column shows verifications for respective

requirements and the last column shows the verifications status of the requirement.

Row	Requirement Specifications	Verifications	Verified
A	Assigned unique name	BMIWHR	yes
B	Metrics identification and measurement	DPPM, DPPS, CCSP	yes
C	Performance validation identification and measures	PDF, Kurtosis, Mean, Variance	yes
D	Dataset characterization and preference	Relations, protected, private	yes

Table 6.1: DPPA Product Annexation Requirements, Showing Verifications for DPPA Dataset Candidacy.

Row A requires an assigned unique name. Our first dataset name BMIWHR, was unique to the DPPA roster and met the unique name verification requirement.

Row B requires metrics identification and measurement. All datasets in an annexation requirement verification process, by default are assigned default metric values of DPPM, DPPS and CCSP. Our datasets therefore met this verification requirement.

Row C requires performance validation identification and measurements. All datasets in an annexation requirement verification process, by default are assigned default performance validation measures of PDF, kurtosis, Mean and Variance. Our datasets therefore met this verification requirement.

Row D requires dataset characterization and dataset preference. Characterization is the formal identification of all dataset columns. Preference is a column property. A column is either protected or preserved. Protected columns are private columns. Customers do not want their private column to be exposed. Customers do not care so

much about their preserved columns. preserved columns are calculated columns. This requirement is essentially the identification of dependent and independent variables in a dataset. Dataset BMIWHR had private variables weight, height, waist and hip. Its preserved variables were BMI and WHR. Dataset BMIWHR had two relations. All our preserved variables are functions of our private variables, hence the existence of relations among our dataset attributes. Our datasets met this dataset requirement.

Our dataset is verified for all rows. Therefore dataset BMIWHR passed annexation requirement validations. Our datasets now has DPPA candidacy 1 and is ready for Annexation processing.

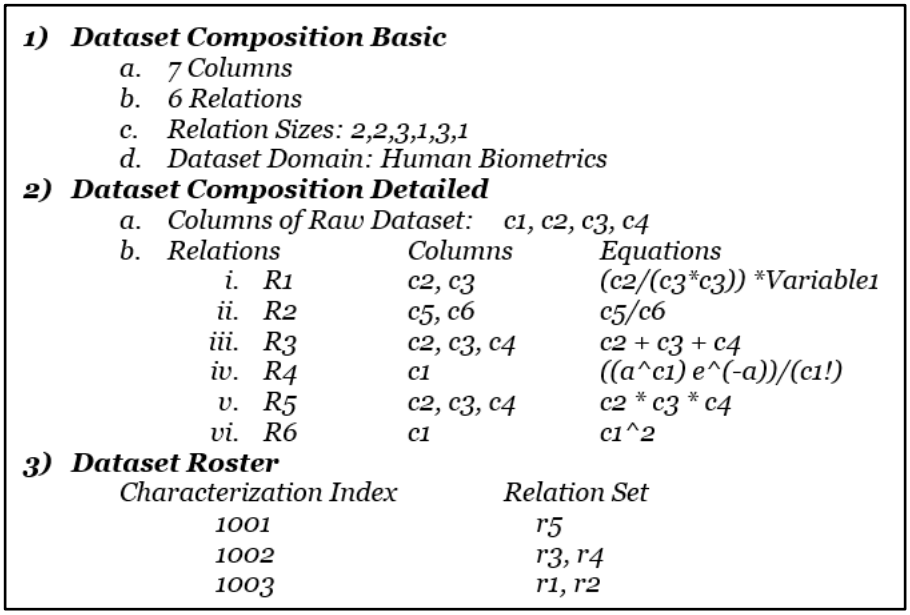
6.3.2 Annexation

The annexation Process is the second step in validating a raw dataset and its components as meeting the DPPA IDC conversion requirements. When a characterization chart and a composition chart is presented for a dataset of candidacy 1, the dataset is promoted to DPPA candidacy 2; meaning it is ready for DPPA integration processing. Integration is not necessarily immediate after candidacy 2.

Dataset composition and dataset characterization are orchestration mechanisms that DPPA uses for raw data column correlations maintenance in IDC published datasets. One dataset composition can have multiple characterizations.

We defined as specifications for a basic dataset composition to consist of sample data column count, relation count, relation sizes, and dataset domain. The detailed composition consists of the column names, relation names, relation columns and the equation for each relation. The column count varies between 1 and n. N is a number which manually scales upward by DP security engineering requests. Relation count is the total relations to be considered for protection processing. Relation size is the total columns in each relation. Each dataset has infinite relations. This is

because the equation for the relation dictates the relation. Each relation has a unique equation. Equations are written using column names. A column can be repeated many times in an equation. Besides the basic and detailed composition, the data composition consists of a roster. Rosters are particulate to datasets type. The roster provides unique combinations of the relations in the detailed composition. The unique combinations are a relation set and are identified by characterization index. These indexes are unique to the dataset and DPPA. The below figure shows the basic, detailed and roster contents of dataset BMIWHR's component chart. The component chart for dataset APEIndex is in the Intra-variability section of this chapter.



1) Dataset Composition Basic

a. *7 Columns*
b. *6 Relations*
c. *Relation Sizes: 2,2,3,1,3,1*
d. *Dataset Domain: Human Biometrics*

2) Dataset Composition Detailed

a. *Columns of Raw Dataset: c1, c2, c3, c4*
b. *Relations*

	Columns	Equations
i. R1	c2, c3	(c2/(c3*c3)) *Variable1
ii. R2	c5, c6	c5/c6
iii. R3	c2, c3, c4	c2 + c3 + c4
iv. R4	c1	((a^c1) e^(-a))/(c1!)
v. R5	c2, c3, c4	c2 * c3 * c4
vi. R6	c1	c1^2

3) Dataset Roster

Characterization Index	Relation Set
1001	r5
1002	r3, r4
1003	r1, r2

Figure 6.2: Outlined Basic, Detailed and Roster Content of the Dataset Composition Chart for Dataset BMIWHR.

The basic composition has 7 columns, 6 relations with relation sizes of 2, 2, 3, 1, 3, and 1. The columns are C1, C2, C3, C4, C5, C6 and C7. The relations are r1 to r6 with respective columns as c2, c3; c5, c6, c2, c3, c4; c1; c2, c3, c4; c1. The respective equations are 1) $(c2/(c3 * c3)) * Variable1$, 2) $c5/c6$, 3) $c1 + c2 + c3$, 4) $((a^c1)e^(-a))/(c1!)$, 5) $c2*c3*c4$, 6) $c1^2$. The dataset roster has three characterization

indexes: 1001, 1002 and 1003. The relation set for respective indexes are r5; r3 and r4; and r1, r2. When a dataset component chart is extended, at least one of its main areas is altered.

We define specifications for a characterize dataset to be as follows: characterization index, dataset type, total column, column indexes and column names, total relations and relation: index, name and equation. The below figure shows the dataset characterization chart for dataset BMIWHR. The characterization chart for dataset APEIndex is in the Intra-variability section of this chapter.

1) Characterization Index		*1003*
2) Dataset Type:		*Human Biometrics*
3) Total Columns:		*7*
4) Column Indexes & Names		
1. Index		
2. Weight		
3. Height		
4. BMI		
5. Waist		
6. Hip		
7. WHR		
5) Total Relations:	*2*	
6) Relation Index	**Relation Name**	**Equation**
i.	*BMI*	*(c2/(c3*c3)) *x*
ii.	*WHR*	*c5/c6*

Figure 6.3: Outlined Dataset Characterization of a Raw Dataset Which Used IDC BMIWHR Security Mechanism.

The above figure shows our dataset having characterization index of 1003 and is of dataset domain type: human biometrics. The data set had 7 columns. The first column is the index column followed by columns weight, height, BMI, waist, hip and WHR. The dataset had two relations: BMI and WHR. The equation for BMI is $(c2/(c3*c3))*Variable1$ and the equation for WHR is $c5/c6$. Dataset characterization is important as it plays a role in early analysis. It provides clear information about what aspects of the data component chart is being used for plots generation. All dataset characterizations are derived form a dataset composition chart. The compo-

sition charts and characterization charts for dataset BMIWHR indicated the datasets are at the DPPA candidacy 2 stage and is ready for integration processing.

6.3.3 Integration

The integration Process is the third step in validating a raw dataset and its components as meeting the DPPA IDC conversion requirements. When a characterization preference chart, a performance chart, an execution path, search location for dataset relations, example raw dataset and dataset roster ID is presented, is presented for a dataset of candidacy 2, the dataset is promoted to DPPA candidacy 3; meaning it is ready for statistical validations of metrics. It also means the detest has completed its all six integration steps. Statistical validations of metrics is not necessarily immediate after candidacy 3. The integration process for dataset BMIWHR is as follows. It is very technical and is a best effort to provide file content and folder structure that is constantly being optimized.

Dataset Preference and Chart File Addition: Dataset characterization file addition, is the creation and path availability of Module File2. File2 contains the roster index which corresponds to the processing of module File1. If the roster index of File2 is in the master roster, then module File1 will be processed by the roster index's correspondence. Our File2 equivalent was in our DPPA execution directory. Its context was number 1003. 1003 is a valid roster index value. From the master roster, this index corresponds to a preference chart. The preference chart of an index cannot be changed. The preference chart contains the protection preferences of the variables. Preference values are private, protected and not applicable. The preference chart is like the data characterization chart, except the preference chart has a preference column for the variables. The preference chart is used in corroboration with performance analysis of plots generated for the diagnostic report. Below is the preference chart

used by dataset BMIWHR. This chart's corroboration role is explained as follows.

1)	**Characterization Index**		*1003*
2)	**Dataset Type:**		*Human Biometrics*
3)	**Total Columns:**		*7*
4)	**Columns:**		
	Index	**Name**	**Preference**
	1.	*Index*	*Na*
	2.	*Weight*	*Private*
	3.	*Height*	*Private*
	4.	*BMI*	*Preserved*
	5.	*Waist*	*Private*
	6.	*Hip*	*Private*
	7.	*WHR*	*Preserved*
5)	**Total Relations:**		*2*
6)	**Relation Index**	**Relation Name**	**Equation**
	i.	*BMI*	*(c2/(c3*c3)) *x*
	ii.	*WHR*	*c5/c6*

Figure 6.4: The Parameter Preference Chart Used by Dataset BMIWHR

There are three preference types: NA, private and protected. NA means not applicable. This is usually used when a column has no effect on the rest of columns. For example, an index column; like in our dataset. Column 1 is used for row indexing and is no used in our reports. The private preference type is applied to variables whose value are not to be exposed in the published report. The private assignment is usually assigned at dataset characterization design time. For the equations, private variables are usually independent variables. The preserved variables are usually dependent variables. This distinction is important as they are used as points of verification whenever a variable is validated via a plot. For a preserved variable, it is not detrimental if its value is exposed. This index chart indicated our dataset BMIWHR met step 1 of the integration process.

Performance Chart Addition: The below figure shows the security product performance chart for dataset BMIWHR. This chart is used in the administrative diagnostic report.

Product performance chart differs for each product. Dataset BMIWHR's per-

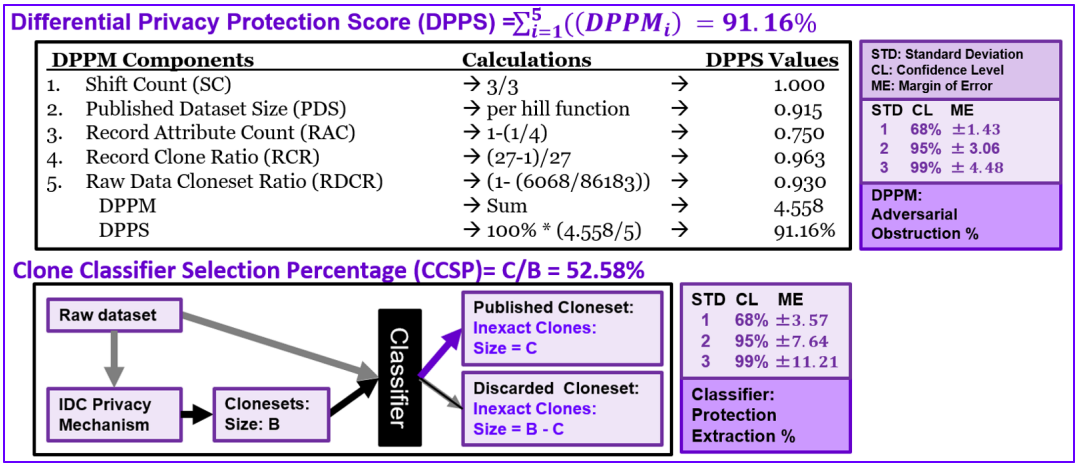


Figure 6.5: The Performance Chart Used by Dataset BMIWHR

formance chart has two sections. The first section shows its performance metric measurement DPPS and its accompanied equation, variables, and values. The other section shows its second performance metrics measurement; CCSP, with its equations, variables, and values. The DPPM components are explained in detailed in chapter 3. For clarity, these performance metric values can change, thus requiring a different dataset version. This new version would require new characterization indexes for its respective data characterizations. The DPPS score is tied to an IDC version. For our work, we started out with 6K raw data records. We generated 163K per published BMIWHR IDC records. We than filter via the classifier 86K published data records. This shows were our 86K/163K gave a comparable 52.58 %. Our 52.58 CCSP resulted in a DPPS of 91.16%. Both CCSP and DPPS are estimators with receptive margin of errors. DPPS has margin of errors with respective standard deviation and confidence level as follows. Margin of error: 1.43,3.06 and 4.48, standard deviation: 1,2 and 3,and confidence level: 68%, 95% and 99%. CCSP has margin of errors with respective standard deviation and confidence level as follows. Margin of error: 3.57,7.64, and 11.21, standard deviation: 1,2 and 3,and confidence level: 68%, 95% and 99%. At this point the purpose of this chart is just to show that it exist. The values on

the chart will be used by later steps. This performance chart indicated our dataset BMIWHR met step 2 of the integration process.

Execution Path Identification: Securing the execution path of DPPA is dependent on its portability, interaction status, access device and execution system type. Our most dependable was execution path was via the Linux operating system, using the MATLAB GUI programming interface, on a desktop from a centrally located folder. The below figure shows our execution path on this machine. It shows the before and after execution paths. We later transfer this execution path to a windows machine using a different MATLAB version. Even though the windows machine was twice as fast, it was less reliable. The windows machine would update unexpected while the code was running in the background.

Figure 6.6: Before and after Directory Tree for IDC Data Execution Path

The figure shows 14 directories and 16 files before execution and 14 directories and 42 files after execution. This might look like lots of files, but remember that one figure in the diagnostic report is also a file. At this point, we are interested in

the existence of the directory tree and not so much the names or content of the files. This is because our focus is on the automation path. The execution directory trees indicated our dataset BMIWHR met step3 of the integration process.

Add Dataset Relation: A relation is a mathematical formula. If we have had six hundred relations or even 30, we would be moved to create a relation search engine. We only had three relations and so we did not implement this search tool. We Left this implementation for continuing research. All new dataset automatically meet this step. This therefore indicated that our dataset BMIWHR met step 4 of the integration process.

Raw Dataset Addition: Raw dataset addition means making available the location and raw dataset which is to be protected. That is, making File1 available in its execution path. The below figure shows our raw dataset available, before and after execution. The presence of this figure indicated that our dataset BMIWHR met step 5 of the integration process.

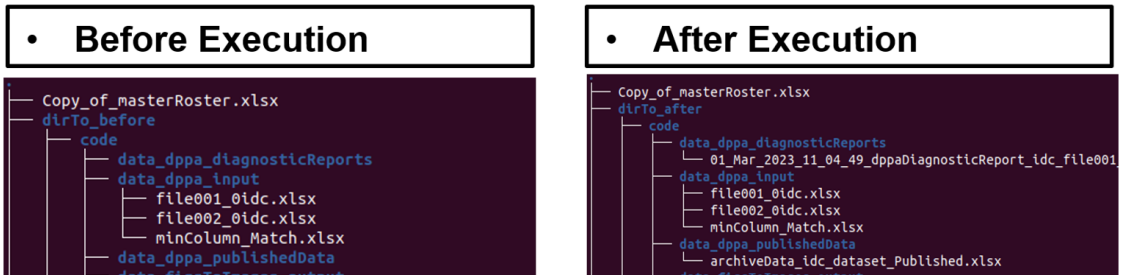


Figure 6.7: Before and after Directory Tree for Location of Raw Dataset

Roster Addition: The ideal master roster is an object that contains dataset characterizations, dataset component charts and local roster indexes of datasets type. The master roster is to be stored in the repository. Roster addition is the adding of a dataset type's: component chart, dataset characterizations and its local roster index to DPPA's master roster. The master roster implementation is not a part of this presented work. The figure below show our Module File2 with its characteriza-

tion index. This characterization index is added to the master roster file as an ID or a roster number for the dataset. The figure below shows our master roster file

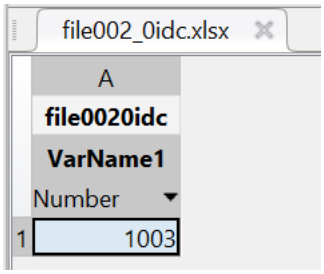


Figure 6.8: Characterization Index for Dataset APEIndex

with the characterization index added. If the index in File2 does not match an index in the master roster, DPPA will defer the processing of inputted raw dataset file; File1. Roster addition is one of the integration step for a DPPA security product annexation. Before a roster adding, it is first verified that the information is not yet in DPPA master roster. Roster addition might have manual parts. Our master roster is an Excel file and we manually added characterization indexes to it. We used the DPPA's execution folder as our master roster location. We manually added the component chart file and characterization file to the DPPA execution folder. The below figure shows our master roster file. It has 4 columns. Each column is added at different times of the integration process.

	rosterID	CopyOnlyDirectory	DatasetExecutionFile	AutomationEnginAccess
	Number	Text	Text	Text
1	roster ID	Copy Only directory	Dataset Execution File	Automation Engin Access
2	1000	na	na	
3	1001	na	na	
4	1003	1003_IDC_for_copy_only	IDC_init_f004v004.m	runDPPA/file002_0idc.xlsx
5	1004	1004_XYZ_for_copy_only	XYZ_init_f004v004.m	runDPPA/file002_0xyz.xlsx
6	1005	1005_NDK_for_copy_only	NDK_init_f004v004.m	runDPPA/file002_0ndk.xlsx
7	1156	na	na	

Figure 6.9: This Shows the Content of the Master Roster File

The presence of our master roster file and the characterization index file; file002 indicated that our dataset BMIWHR met step 6 of the integration process. At this point dataset BMIWHR has met all the requirement specification for the DPPA Integration process. Our dataset BMIWHR now has DPPA candidacy 3 and is ready for statistical validation of metrics processing.

6.3.4 Statistical Validation of Metrics

The statistical validation of metrics process is the fourth step in the validating a raw dataset and its components as meeting DPPA IDC conversion requirements. When the key chart, sample size selection plot, with its four accompanied CCSP statistic validating plots are presented for a dataset of candidacy 3, the dataset is promoted to DPPA candidacy 4; meaning it is ready for diagnostic report validations. It also means the dataset has completed and pass its statistical validation of metric, step. Diagnostic report validation is not necessarily immediate after candidacy 4. Figure 6.10 is the key chart used by datasets BMIWHR and APEIndex for their sample size selection plot reference. Figures 6.11 and 6.12 are the sample size selection plots for datasets BMIWHR and APEIndex respectively. Figures 4.1, 4.2, 4.3 and 4.4 are the four CCSP statistic validating plots for dataset BMIWHR. Figures 6.25, 6.26, 6.27 and 6.28 are the four CCSP statistic validating plots for dataset APEIndex. The above mentioned figures indicate datasets BMIWHR and APEIndex have completed and pass its statistical validation of metric step and is now at candidacy 4 and is ready for Diagnostic report validations.

The following is the statistical validation of metric, process for datasets BMIWHR and APEIndex. Our findings are that, the plots in figures 6.11 and 6.12 verify and validate our sample sizes are ample to represent datasets BMIWHR and APEIndex respectively; in determining the CCSP estimator. The following is our process for this

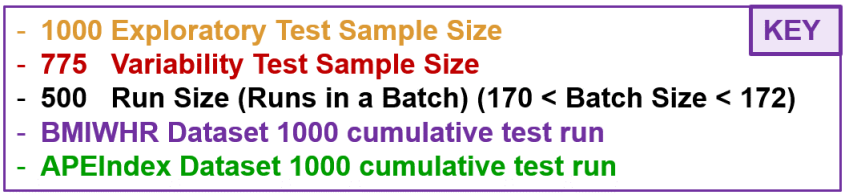


Figure 6.10: Key Chart Used by Datasets BMIWHRr and APEIndex for Sample Size Selection Plots References.

conclusion. Figure 6.10 shows a chart with the key for the sample size selection plots (SSSP) of datasets BMIWHR and APEIndex. Figures 6.11 and 6.12 are SSSP for datasets BMIWHR and APEIndex respectively. The SSSP are multi plot consisting of four plots each.

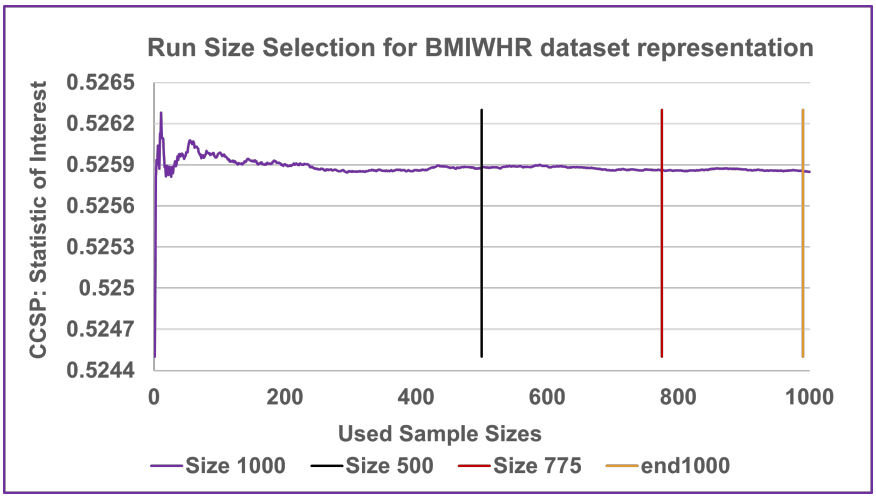


Figure 6.11: Sample Size Selection Plot for Dataset BMIWHR

The first row of the key chart is orange and correspond to the orange plots in the SSSP. The orange plots are vertical lines with sample size value of 1000. The second row of the key chart is red and correspond to the red plots in the SSSP. The red plots are vertical lines with sample size value of 775. The third row of the key chart is black and correspond to the black plots in the SSSP. The black plots are vertical lines with sample size value of 500. Run size is the same as sample size. The fourth row of the key chart is purple and correspond to the purple plot of dataset BMIWHR. The fifth

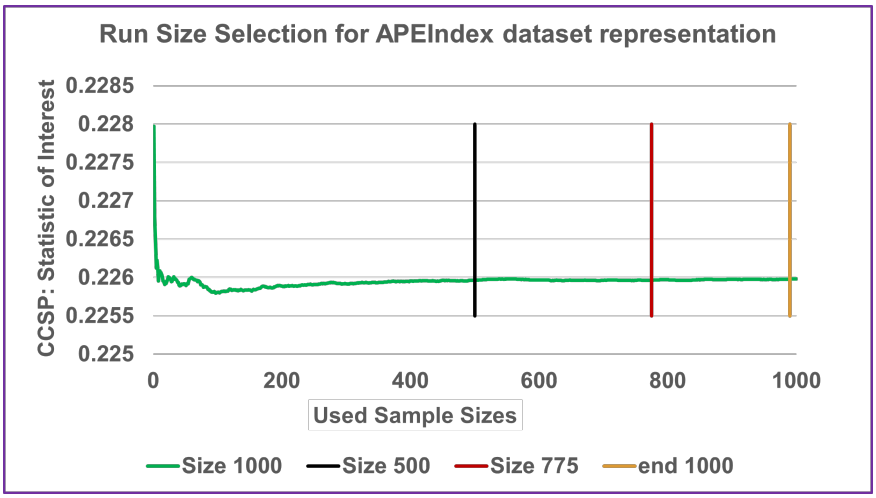


Figure 6.12: Sample Size Selection Plot for Dataset APEIndex

row of the key chart is green and correspond to the green plot of dataset APEIndex. Plots purple and green plots are tapers to the right. The tapering plots consist of 1000 cumulative samples each. The tapering plots intersects all the vertical plots at the same respective CCSP value. For the BMIWHR dataset, points of intersection have CCSP value of 0.5285 for all points. For APEIndex dataset, points of intersection have CCSP value of 0.226 for all points.

The tapering plots show that they level out as the sample sizes increases. Visual leveling starts to appear from sample value in range of 250 to 300. From this we say a valid minimal sample size is about 250. From this we say any sample size above 250 is valid for estimating the CCSP statistic. This is because the CCSP value remained the same after the sample size passed 250. Fro more accuracy, we choose 500 as our sample size. The black plots indicates sample sizes of 500. In validating CCSP as a statistic, we used batch sizes with sample size of 500. We used 500 as one batch takes a longer time to run. We used about 171 batches for our final CCSP estimators. For variability testing, we used sample size 775. Variability test takes less time to run; they used 1 batch of size 775. The formulation and calculations of DPPS and CCSP

as valid statistical metrics for dataset BMIWHR are in chapter 5. The formulation and calculation for DPPS and CCSP as valid statistical metrics for dataset APEIndex are in later sections of chapter 6. The minimal sample size used for CCSP was 500.

The process of validating the CCSP statistic for datasets BMIWHR and APEIndex is in chapter 4 and later in chapter 6 respectively. Each dataset has four CCSP statistic validating plots which has contents for the validation.

6.3.5 Diagnostic Report Validations

The Diagnostic report validation process is the fifth step in validating a raw dataset and its components as meeting the DPPA IDC conversion requirements. When a diagnostics report is presented for a dataset of candidacy 4, the dataset is promoted to candidacy 5; meaning it is ready for automation response to variability testing. This variability testing is not necessarily immediate. Diagnostic reports for datasets BMIWHR and APEIndex were presented and therefore completed their diagnostic report validation step. This means both datasets has candidacy level 5 and is ready for automation response to variability step. In the followings we give a general description of our diagnostic report design and also a description of a diagnostic report for dataset BMIWHR.

DPPA diagnostic report design has its significance. For prototyping we focused on two report versions: the View version and the Detailed version. The detailed version has the contents of the View's version followed by an administration chapter. The report layout is dependent on the security product type. This work focused on the IDC security product. The general layout of the detailed report is as follows. The report starts with a timestamp page, followed by a cover page, table of contents page and four chapters. Report length is dependent on the total columns in the raw dataset. The table of contents contains the chapter headings and their subsection

headings. Chapter 1 has information about the raw dataset. The information includes dataset preference chart, extracted diagnostics, and the detailed list of count plots in chapter 2. Chapter 2 has information about the classifier's performance count plots. Chapter 3 has information about administrative charts and the classifier's performance probability plots. Chapter 4 is the salutation section. Salutations are intended to provide information about the security product origin. The salutation has a time-date stamp, followed by the DPPA product origin. The rest of this chapter contains a description of a diagnostic report for a dataset with preserved and private variables.

This is a description of a detailed diagnostic report for dataset BMIWHR dataset. The first page of the report is our timestamp page. The first line shows the time and data when the report was generated. The second line shows the name of the raw data file. The third line shows the name of the diagnostic report. The nomenclature of the diagnostic report starts with the word, 'dppaDiagnosticReport_' immediately followed by the name of the raw data file name followed by the report's file extension, 'pdf'. The second page is the cover page.

Cover Page: The cover page has title, 'A DPPA Diagnostic Report:', followed by the report's version. In this case, this is a detailed report. The subtitle contains the security product's name. In this case, 'of an Inexact Data Cloning (IDC) Security Report'. The subtitle is followed by a picture showing three plots arranged in vertical order. Besides their aesthetics and content appreciation purposes; a DPPA engineer would identify them as follows. The first two plots shows IDC protection for private variables. The last plot shows IDC protection for preserved variables. . The subtitle for all DPPA reports will match its respective security product name. The image is followed by my name, Zelpha Thomas, the PhD Candidate and our supporting Research Lab at Arizona State University, Bliss Labs. This work is produced through

Bliss Labs. The date when the report was generated is the last content on the cover page. The cover page is followed by the table of contents. The table of contents list their four chapters with their sub headings. Table of contents is followed by chapter 1.

Plot Chart Properties: The charts in this section show two plots on the same axis. The plot in blue is the DPPA security product plot and the plot in red is the raw data plot. The charts are divided into two types, validation, and verification. All verification plots are on the left side of the page. There are two types of verification plots: count and probability. Probability plots are for administrative purposes and are found in chapter 3. All probability plots have their corresponding count plots. Count plots are in chapter 2. Validation plots are on the right side of the page. Each verification plot has its corresponding validation plots. There are two types of parameters in verification plots, private and preserved. Private means the published data value must not resemble or compromise the raw data value. Preserved means the published data value may or may not resemble the raw data value. Below, we explain our definition and application of verification and validation. Our verification plots aim to provide visual evidence that one dataset: DPPA security product dataset, is a comparable representation of another: raw dataset. After the evidence is provided, it is interpreted and validated. For us, interpretation is the identification of plot characteristics which matches engineering constructs. A construct is an area of respective significance. Validation is the presentation of measurements that corroborates the identified characteristics in the verified plots. Verification plots are representations of histograms or frequency distribution plots. The vertical axes on verification plots are in unites of integer counts or probabilities. The horizontal axis on verification plots is in units of integer representing bucket sizes.

Chapter 1: Datasets Information. The heading for chapter 1 is ‘Dataset Informa-

tion'. Chapter 1 has four sections: Dataset Preference Chart; Extracted Diagnostics 1: DPPS and CCSP Metrics; Extracted Diagnostics 2: Related Variables; Lists of Count Plots: IDC Data vs Raw Data. These sections are explained below.

Section 1.1 is entitled, Dataset Preference Chart. For this section, the area of emphasis on the preference chart is item 4; Columns. This line item tells the preferences of variables per the raw data; private or preserved. Published data are to maintain raw data preferences. Further information about preference charts is explained in the Integration section of this chapter. Figure 6.4 is the preference chart for dataset BMIWHR.

Section 1.2 is entitled, Extracted Diagnostics 1: DPPS and CCSP Metrics. This section has diagnostics extracted from the IDC execution. The diagnosis is related to the DPPS and CCSP Metrics. The calculated values are extracted form the last IDC generation execution run. The benchmark values are the archived DPPM benchmark values. The BM values are indicators which tells if the last execution falls within the DPPM benchmark range. A 1 indicates within range and a 0 indicates out of range. Further metric descriptions and their corresponding values are explained in the 'Components of DPPM' section of this work; outlined as section 3.2 in the table of contents.

Section 1.3 is entitled, Extracted Diagnostics 2: Related Variables. This section shows another set of diagnostics extracted from the IDC execution. The diagnostics are about salient variables. For clarity, their explanations are as follows. Raw dataset size indicates the size of the dataset to be encrypted or protected. All generated clones total indicates is the number of IDC records generated and send to the classifier. After the inexact clones passed through the classifier, some are discarded because they have compromising columns. The rest are published. All published inexact clones indicates the number of clones passes through the classifier and were deemed good to

be published; uncompromising. Shift count available indicates the number of noise pattern our IDC mechanism has. We had three shift types. Shift count used indicates the number of noise pattern used in the published dataset. Published dataset size indicates the total count of good IDC data records extracted by the classifier. This is the same as the size of all inexact clones in item 3. Min matching attributes indicates the minimum compromising matching column count between IDC record column and raw data column. MinMatch is further explained in section Record Attribute Count of chapter 3. Total protected attributes indicates the total private variables in a record column. Clones per human subject indicates the number of inexact clones to one raw data record before the inexact clones enter the classifier. .

Section 1.4 is entitled, List of Count Plots: IDC Data Vs Raw Data. This is the count plots for IDC data vs raw data. Software engineers might have use for this concise information. It is ordered via relations. This section list the verification and validation plots for each relations used by the published dataset. These are the plots in chapter 2. List 1 shows the performance plots (verification and validation) for relation: BMI. List 2 shows the performance plots (verification and validation) for relation WHR. BMI has variables weight, height and BMI. WHR has variables waist, hip and WHR. Each variable has a verification plot. Each verification plot has its validation plot in the form of mean, kurtosis and variance.

Chapter 2: Classifier Performance Count Plots. The heading for chapter 2 is, Classifier Performance Count Plots. The word classifier is used in this section as this is where the security value of the classifier is most visible via plots. The sections in chapter 2 is dependent on the total relations used by the published data. Each section is about a unique relation.

Section 2.1 is entitled, Performance Plots (Verification and Validations) for Relation: BMI. This section has verification plots for private variables weight, height

and preserved variable bmi. Verification plots are to the left. The variables shares validation plots. The validations plots are to the right and are of statistic mean, kurtosis and variance.

Section 2.2 is entitled, Performance Plots (Verification and Validations) for Relation: WHR. This section has verification plots for private variables waist, hip, and preserved variable whr. Verification plots are to the left. The variables shares validation plots. The validations plots are to the right and are of statistic mean, kurtosis and variance.

Chapter 3: Administration Charts and Classifier Performance Probability Plots. The heading for Chapter 3 is, Administrative Charts and Classifier Performance Probability Plots. Chapter 3 is present only in the Detailed diagnostic reports. This chapter is called the administrative chapter since it serves the purpose of providing additional perspective of data plots and raw dataset origins. This information can be used for referencing by DPPA engineers. Chapter 3 has five sections. The first three sections are always given and are; Dataset Composition Chart, Dataset Characterization Chart, IDC Benchmark Performance Chart. All other sections that follows are dependent on the total relations in an IDC record columns. Dataset BMIWHR has two relations. The next two sections of chapter 3 are BMI: Classifier Performance Probability Plots and WHR: Classifier Performance Probability Plots. The classifier performance probability plots of chapter 3, provide intuition form a probability perspective whereas the plots in chapter 2 provides the same information form a count perspective. Each section of chapter 3 is explain below.

Section 3.1 is entitled, Dataset Composition Chart. This chart shows the detailed and basic compositions relating to the raw dataset origin. The area of interest is item 2; dataset composition detailed and item 3, dataset roster. Further information about the dataset composition chart is explained in the Annexation section of this

chapter. Figure 6.2 is the composition chart for dataset BMIWHR.

Section 3.2 is entitled, Dataset Characterization Chart. The main area of interest is items 4, it lists the column names and their indexes. This item provides hard connection between the detailed and viewed report versions. Further information about the dataset characterization chart is explained in the Annexation section of this chapter. Figure 6.3 is the characterization chart for dataset BMIWHR. .

Section 3.3 is entitled IDC Benchmark Performance Chart. The chart has two parts. The first section shows the components and calculation of the DPPS. The second section shows the components and calculation of the CCSP. These sections show internal aspects of the DPPA benchmarks. All DP security product have their respective benchmark. These benchmarks are called DPPA benchmarks. The IDC Benchmark Performance Chart shows the DPPS and CCSP benchmark values. The value of DPPS is 91.16% and the value of CCSP is 52.58%. Further information about the IDC Benchmark Performance Chart is explained in the Integration section of this chapter. Figure 6.5 is the IDC performance chart for dataset BMIWHR.

Section 3.4 and Section 3.5 are similar but use different relations. Section 3.4 is entitled, BMI Classifier Performance Probability Plots and Section 3.5 is entitled, WHR: Classifier Performance Probability Plot. Both sections show verification charts in terms of probability plots. The validation plots for these charts are in chapter 2 on the right-hand side. Repeated for reference; the verification plots in chapter 3 represents probability plots. The vertical axes on these verification plots are in unites of probabilities. The horizontal axis on the verification plots are in units of integer representing bucket sizes values. The probability charts show two plots on the same axis. The plot in blue is the IDC plot and the plot in red is the raw data plot. The charts are divided into two types, verification and validation. There are two types of verification plots, private and protected. Each verification plot has its corresponding

validation plots: mean, kurtosis and variance. These verification charts express the probability distribution of the bucket size for a raw dataset value vs an IDC dataset value.

Chapter 4: Salutation. The heading for chapter 4 is salutation. The most recent salutation version is as follows. "Time: 13-Oct-2022 04:17:09 This differential privacy dataset diagnostic report was produced using specifications of DPPA. This report was produced by Zelpha Thomas in conjunction with research lab: Bliss Labs of Arizona State University."

6.3.6 Automation Response to Variability

Our datasets completed chapter 6.4; Automation Validation Services, hence this requirement is fulfilled by both our datasets. This means our dataset has candidacy 6 and is ready for DPPA roster Debut for automation.

6.3.7 DPPA Roster Debut for Automation

DPPA Roster debut for automation means that the dataset has a corresponding entry in column, AutomationEngineAccess for the roster file. Figure 6.9 shows an example of the roster file. All our datasets have an entry in that column. With this entry a dataset relations is now available for usage along the DPPA Automation Path. This means our dataset has candidacy 7 and is ready for public usage in the DPPA Roster.

6.4 Automation Validation Services

The DPPA requirements charts: Dataset Preference chart, Data Composition chart, Dataset Characterization chart, are built by DPPA engineers. These charts are built or extracted from the dataset before DPPA integration. DPPA on receiving

these charts, treats datasets from the chart's domain and the charts as mathematical constructs, independent of raw dataset domain. This concept of mathematical constructs is the indication that DPPA has unite datasets from different domains and thus is technically independent of dataset domain. This also means, once the raw dataset charts are constructed and DPPA sees the mathematical constructs, dataset domain differentiation is cosmetic. With that said, we moved our interest to the behavior of DPPA under variability. For us, variability has a few meanings. For now, it means systematically changing a component of a mathematical construct and observing changes in the construct's domain.

Creating DPPA as a maiden product, we wanted to showcase it's ample stability, while outlying some of its limitations. For stability, the goal was to see if the dependent variables; DPPS and CCSP would hold under variability. Hold means behave as we expected. We placed, 'behaving as we expected', as an equivalent to passing a software engineering variability test. We classified our software variability testing into two types: inter variability and intra-variability.

Per our variability test outcomes, our dependent variables behaved as we expected and thus, we claim DPPA holds under variability. The test results showed that DPPA was a satisfactorily stable product. The following actions were therefore taken. Variability charts were constructed and used as DPPA product stability verification charts. Product stability verification chart are indicator charts which shows that DPPA holds under new security product integration. We use our test results to satisfy our establishment of DPPA as a prototype. Our DPPA prototype serves as a benchmark for DP security products benchmark suite augmentation. The verification of these claims lie in the repeatable outcomes of the variability tests.

The next two subsections explain our two variability test findings. Our software variability testing is similar to combining aspect of software functional testing, per-

formance testing and smoke testing. Functional testing verifies the required output of the software system. Performance testing evaluates the behavior of the software upon varying system or requirement components. Our smoke test used a subset of our prominent requirement tests to verify system assurance.

6.4.1 Intra-variability

For us, intra-variability is the changing of one component of a performance metric while performing successful DPPA runs and noting its outcome. For clarity, we called the component that is being changed, the driver variable; usually it is being called the independent variable. Intra-variability is also our novel way of saying all variables of a successful DPPA runs are kept constant except for one selected variable from the performance metrics equation. The one selected variable is the driver variable and variables that are kept constant are the controlled variables. Successful means that the security product has been integrated into DPPA and has already generated verified diagnostic reports.

We generated data for intra variability testing using two distinct scaling drivers; the raw dataset size and the maxMatch value. When we vary the raw dataset size, we kept the maxMatch size constant; and viceversa. This work has two performance metrics: DPPS and CCSP. The purpose of the variability test was to note the effects of raw dataset variation on DPPS and CCSP. The purpose continued where the variability test was to note the effects of maxMatch variation on DPPS and CCSP.

Our intra variability testing, uses three variable types; dependent, independent and controlled. DPPS and CCSP are treated as dependent variables. We selected DPPM components RDCR(Raw Dataset Size) and RAC (maxMatch) as independent variables and where necessary SC, PDS, RCR, DPPS, CCSP as controlled variables. We keep the values of the controlled variables constant, vary the value of the indepen-

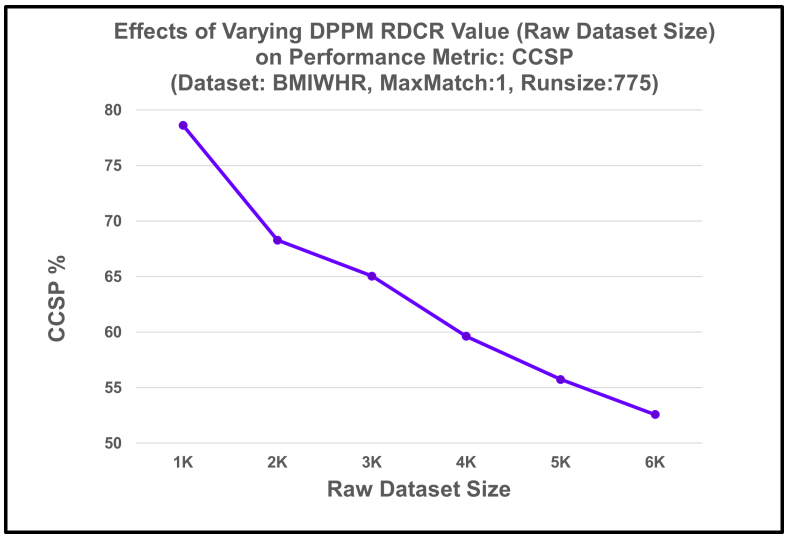


Figure 6.13: Chart Showing Plot of Raw Dataset Size Versus CCSP Metric Performance

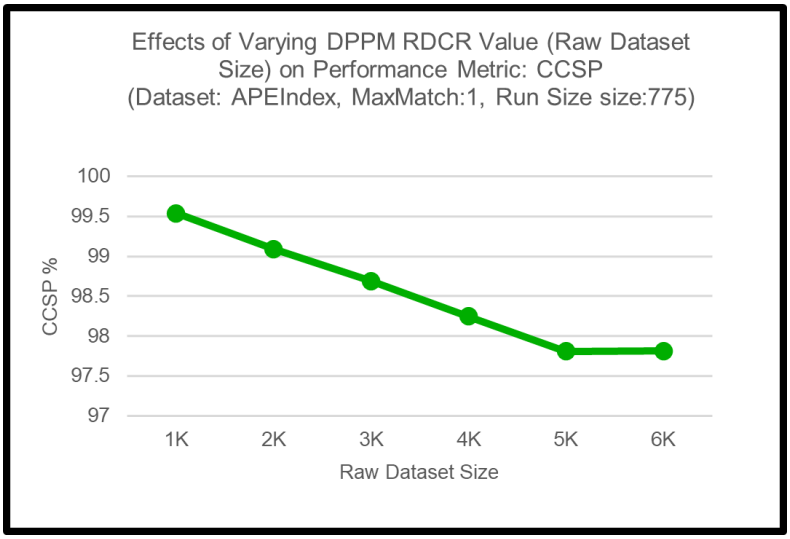


Figure 6.14: Runsize Chart for Dataset APEINdex Characterization Chart.

dent and then evaluate and or measure the value of the dependent variables. The rest of this section outline the intra variability testing using independent variable DPPM RDCR followed by using independent variable DPPM RAC. RDCR varies the raw dataset size and RAC varies the privacy threshold of the classifier; maxMatch.

This section outlines the effects of varying RDCR(Raw Dataset Size) on Perfor-

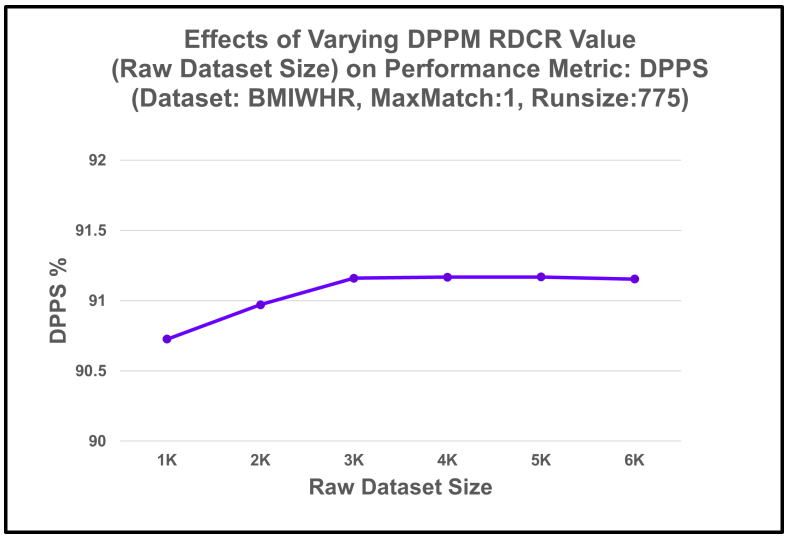


Figure 6.15: Chart Showing Plot of Raw Dataset Size Versus DPPS Metric Performance

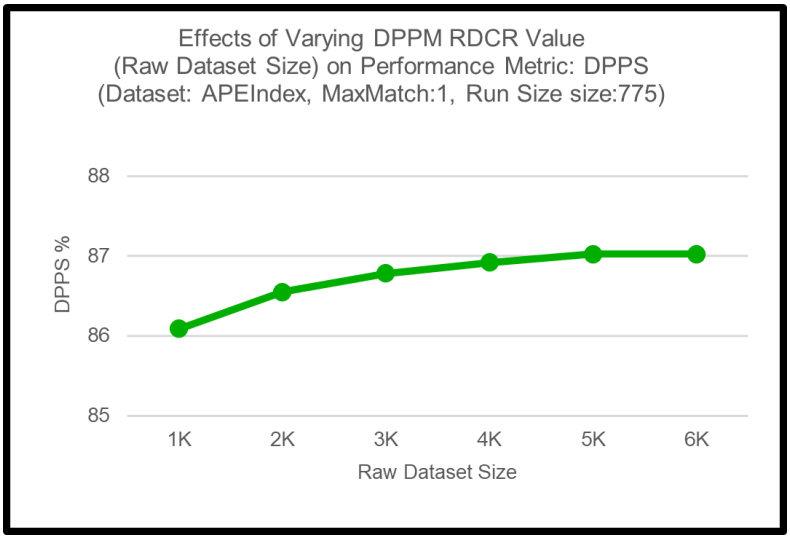


Figure 6.16: Runsize Chart for Dataset APEINdex Characterization Chart.

mance Metric, CCSP and DPPS for datasets BMIWHR and APEIndex. We called this test case, Test Case 1. Test case 1 used raw dataset size as independent variable. Case Description: This case tests the performance of DPPA under raw dataset size variability. Purpose: The purpose verifies if DPPA performs consistently whenever it encounters a new raw dataset size change. Ample raw dataset size change by the or-

der of 1K. Raw dataset sizes used: 1K, 2K, 3K, 4K, 5K and 6K. For each dataset size, 775 IDC samples were run in DPPA and their average DPPS and CCSP scores were collected, plotted and compared to their respective performance chart. For each run, the maxMatch value was 1, DPPS shift count was 3/3, DPPM RAC was 27, DPPS PDS and DPPS RDCR were calculated using respective averages. Expectation 1: As the dataset size increases the CCSP decreases for both datasets. Figures 6.13 and 6.14 show that this expectation was met. Expectation 2: As the dataset size increases the DPPS increase at a small rate and then levels off for both datasets. Figures 6.15 and 6.16 show this expectation was met. Expectation 3: For all 6K results each matches the statistic on their respective performance chart. The 6K results in figures 6.13 and 6.15 match their respective statistic in performance chart figure 6.5. This match show that this expectation was met for dataset BMIWHR. The 6K results in figures 6.14 and 6.16 match their respective statistic in performance chart figure 6.24. This match show that this expectation was met for dataset APEIndex.

Findings: The expectations were met. Test 1 status Pass. Conclusion: Dependent variables DPPS and CCSP behaves as expected under the variation of independent variable, raw dataset size. This result indicates that DPPA is stable under intra variation of raw data size.

This section outlines the effects of varying RAC (maxMatch) on Performance Metric, CCSP and DPPS for datasets BMIWHR and APEIndex. We called this test case, Test Case 2.

Test case 2 used maxMatch, as independent variable. Case Description: This case tests the performance of DPPA under DPPM RAC variability. DPPM RAC is the same as maxMatch or minMatch + 1. Purpose: The purpose verifies if DPPA performs consistently whenever it encounters a new RAC value. RAC value controls the restriction of the classifier. For dataset BMIWHR, maxMatch values were 0, 1,2,

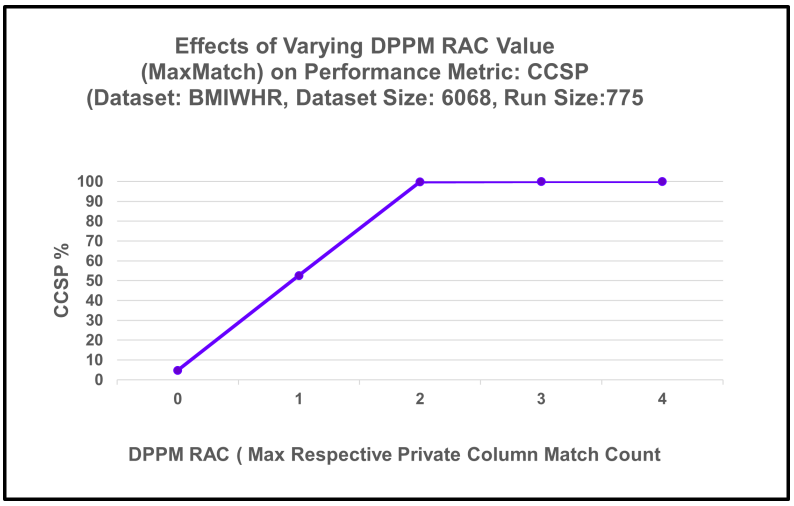


Figure 6.17: Chart Showing Plot of DPPM RAC Versus CPPS Metric Performance

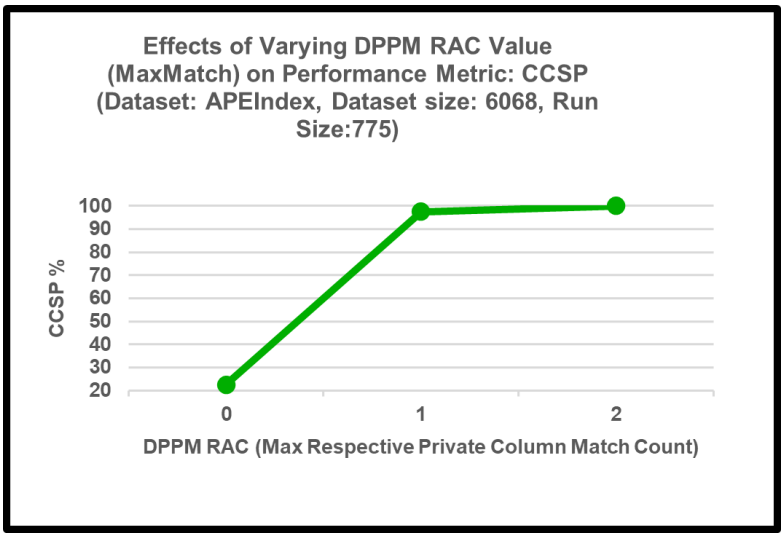


Figure 6.18: Runsize Chart for Dataset APEINdex Characterization Chart.

3 and 4. For dataset APEIndex, maxMatch values were 0, 1 and 2. For each dataset size, 775 IDC samples were run in DPPA and their average DPPS and CCSP scores were collected, plotted and compared to their respective performance chart. For each run, the raw dataset size was 6K, DPPS shift count was 3/3, DPPM RAC was 27, DPPS PDS and DPPS RDCR were calculated using respective averages.

Expectation 1: As the maxMatch size increases the CCSP increases up to 100% for both datasets. Figures 6.17 and 6.18 show that this expectation was met. Expectation

2: As maxMatch size increases the DPPS eventually decreases towards the same value for all datasets. Figures 6.19 and 6.20 show this expectation was met. Expectation 3: For all maxMatch value of 1 or 0, each matches the statistic on their respective performance chart. The results in figures 6.17 and 6.19 match their respective statistic in performance chart figure 6.5. This match show that this expectation was met for dataset BMIWHR. The results in figures 6.18 and 6.20 match their respective statistic in performance chart figure 6.24. This match show that this expectation was met for dataset APEIndex.

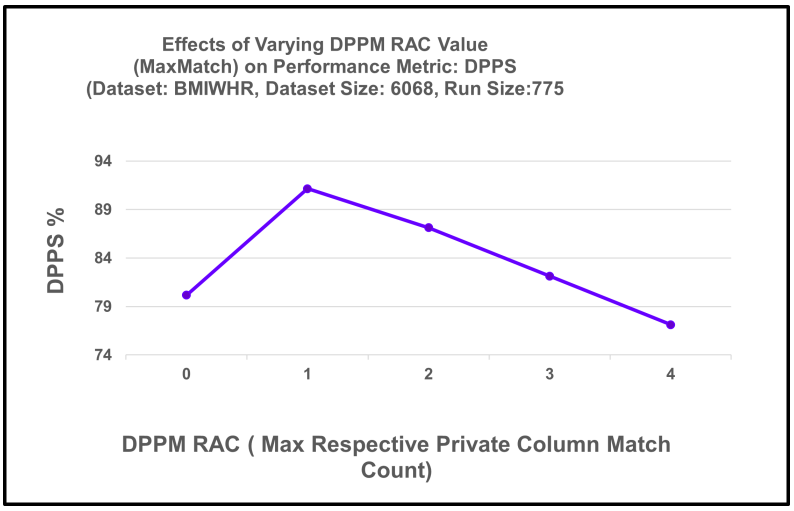


Figure 6.19: Chart Showing Plot of DPPM RAC Versus DPPS Metric Performance

Findings: The observations matched the expectations. Status: Pass. Conclusion: Dependent variables DPPS and CCSP behaves as expected under the variation of independent variable, DPPM RAC. This result indicates that DPPA is stable under intra variation of DPPM RAC.

Per the conclusions of test cases 1 and 2, DPPA is stable under intra-variability.

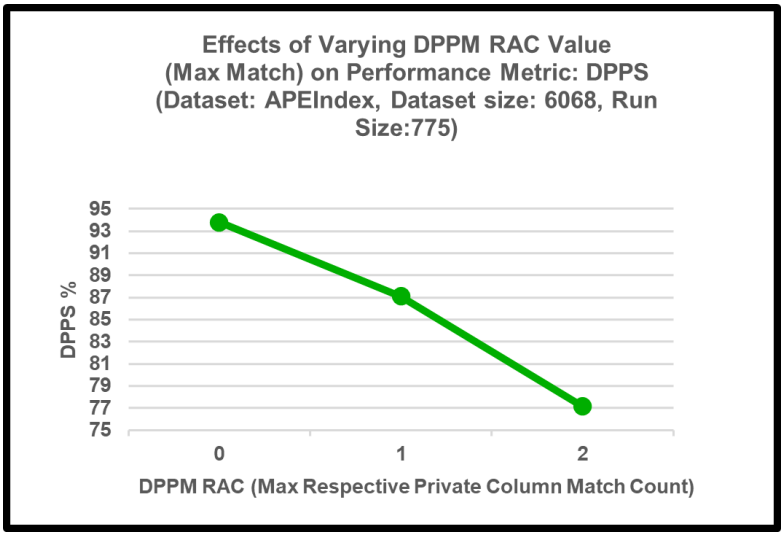


Figure 6.20: Runsize Chart for Dataset APEINdex Characterization Chart.

6.4.2 Inter Variability

The purpose of inter variability testing is to verify and validate that DPPA is stable under the introduction of a new IDC dataset. We have incorporated a new dataset, APEIndex into DPPA which verified and validated this inter variability case. The APEIndex dataset is sometimes called the xyz dataset. The following is the pre-automation process used to incorporate dataset APEIndex into DPPA. The process is outlined as steps.

We first create a story for the new dataset. Example customer story and request: The customer is in the business of indoor recreational rock climbing. The business idea is to make the boulders on the rock-climbing surface interactive, whereby they adjust to the climbers ape index. We use the definition of climbers arm wingspan minus climbers height to calculate climbers ape index. The customer provided a raw dataset of 6K records and 4 columns. The first column was for indexing, second was wingspan, third height, and fourth ape index. The customer stated that inexact replicated data was needed for machine learning software development and testing.

The customer also requests to have the height and wingspan private, and the ape index preserved. The customer wanted the processed data realistically represent the raw dataset and also large enough to accurately reflect ape index variability in the raw dataset. The above request would be send for data engineering processing. The source of the APEIndex dataset is Kaggle.com from ANSURI II dataset. The used attributes were span: mm used as wingspan: in and stature: mm used as height: in. The Data Engineer proposed solution was as follows: Create APEIndex security code using MATLAB platform. The raw dataset layout is like figure 2-9 having weight, height and BMI replaced with span, stature, and ape index respectively. The security mechanism for dataset APEIndex is of type IDC. Use the IDC DPPS and CCSP metrics. Create the following APEIndex dataset signatures: charts; composition, characterization, preference and performance. Generate ApeIndex diagnostic report and create APEIndex variability plot.

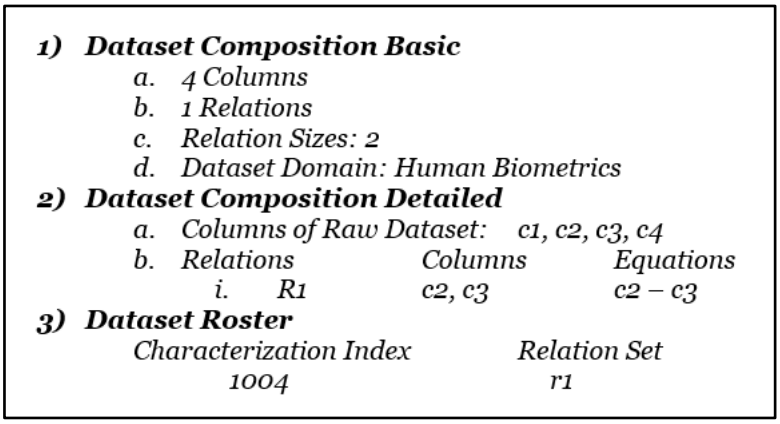


Figure 6.21: APEIndex Dataset Composition Chart.

APEIndex Dataset Composition Chart. Figure 6.21 shows the dataset composition chart for dataset APEIndex. The basic section shows 4 columns, 1 relation that has a size of 2. The dataset domain is human biometrics. The detailed section shows four columns: c1, c2, c3, c4 and c5. Relation 1 uses columns 2 and 3. The equation

is c2 – c3. The dataset roster shows characterization index of 1004 for relation set r1.

1) Characterization Index		**1004**
2) Dataset Type:		**Human Biometrics**
3) Total Columns:		**4**
4) Column Indexes & Names		
1. *Index*		
2. *Span*		
3. *Stature*		
4. *ApeIndex*		
5) Total Relations:		**1**
6) Relation Index	**Relation Name**	**Equation**
i.	*ApeIndex*	*c2 – c3*

Figure 6.22: APEIndex Characterization Chart.

Dataset characterization chart: Figure 6.22 shows the dataset characterization chart for dataset APEIndex. The characterization index is 1004; dataset type is human biometrics; total columns are 4; column index from 1 to 4 with respective names of index, span, stature and ape index. There is one relation: ape index having equation c2 – c3.

1) Characterization Index			**1004**
2) Dataset Type:			**Human Biometrics**
3) Total Columns:			**4**
4) Column			
Indexes	**Names**	**Preference**	
1.	*Index*	*Na*	
2.	*Span*	*Private*	
3.	*Stature*	*Private*	
4.	*ApeIndex*	*Preserved*	
5) Total Relations:			**1**
6) Relation Index	**Relation Name**		**Equation**
i.	*ApeIndex*		*c2 – c3*

Figure 6.23: The Parameter Performance Chart Used by APEIndex Dataset

Dataset preference chart: Figure 6.23 shows the dataset characterization chart for dataset APEIndex. The characterization index is 1004; dataset type is human biometrics; total columns are 4; column index from 1 to 4 with respective names of index, span, stature and apexindex. There is one relation: ape index having equation

c2 – c3. The perimeter performance preference are as follows: index, not applicable; span, private; stature, private; apeindex, preserved.

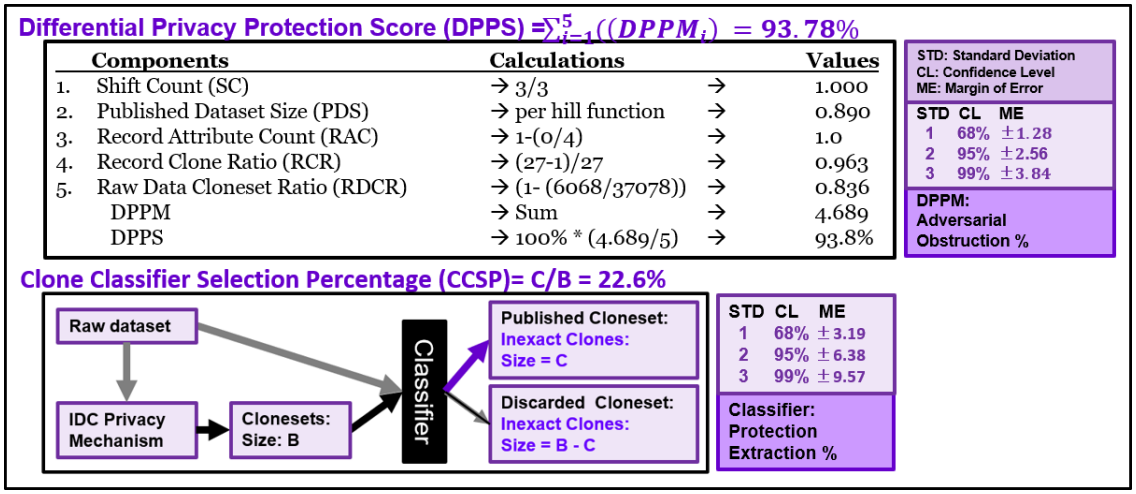


Figure 6.24: Performance Chart Used by Dataset APEIndex

Dataset performance chart: Figure 6.24 shows the performance chart for dataset APEIdex.

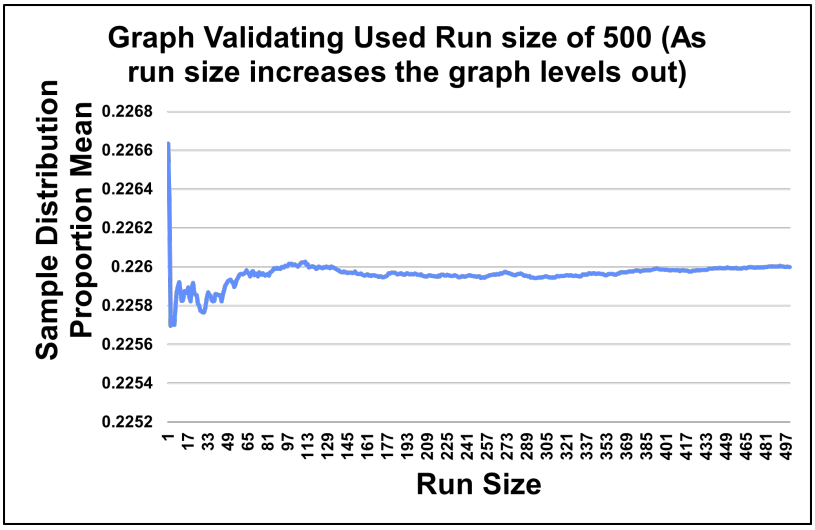


Figure 6.25: Graph Showing Sample Distribution Proportion Mean of 22.6% Satisfied by the Run Size of 500 for Dataset APEIndex.

Figure 6.25 verifies and validate a sample size of 500 is ample. The explanations are as follows. Data: biometric records. Raw data size (RDS): 6,068. Ssample size:

163,836. Collision free clone: 37,027. Statistic: proportion of classifier extracted inexact clones . Run Size: 500. Statistic of 1 batch: 0.2260%. Proportion std Error: 1.87%. Normal Distribution Check, (Per Empirical rule and parameters) mean, mode, median: (0.226, 0.226, 0.226). samples <= mean : 50.40%. samples in 1STD: 73.4%. samples in 2STD: 94.60%. samples in 3STD: 99.40%.

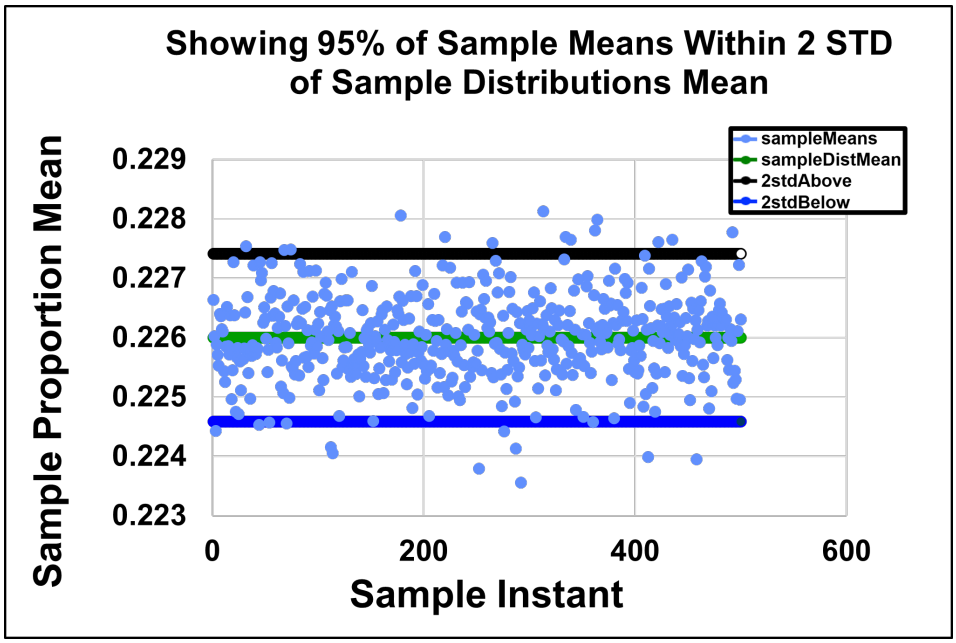


Figure 6.26: Chart Showing Run Sample Findings Are Normal Hence Are Usable as Batch Size to Calculate APEINdex CCSP Statistic.

Figure 6.26 validate figure 6.25 as following a normal distribution. The explanations are as follows. Data: biometric records. Raw data size (RDS): 6,068. Clone/sample size: 163,836. Collision free clone: 37,027. Statistic: proportion of classifier extracted inexact clones . Run Size: 500 . Statistic of 1 batch: 0.2260. Proportion standard error: 1.87%. Normal Distribution Check. (Per Empirical rule and parameters). mean, mode, median: (0.226, 0.226, 0.226) samples <= mean : 50.40%. samples in 1STD: 73.4%. samples in 2STD: 94.60%. samples in 3STD: 99.40%. Confidence level: 94.6%% = 100

Figure 6.27 uses the samples or runs in figure figure 6.25 as one batch. Figure

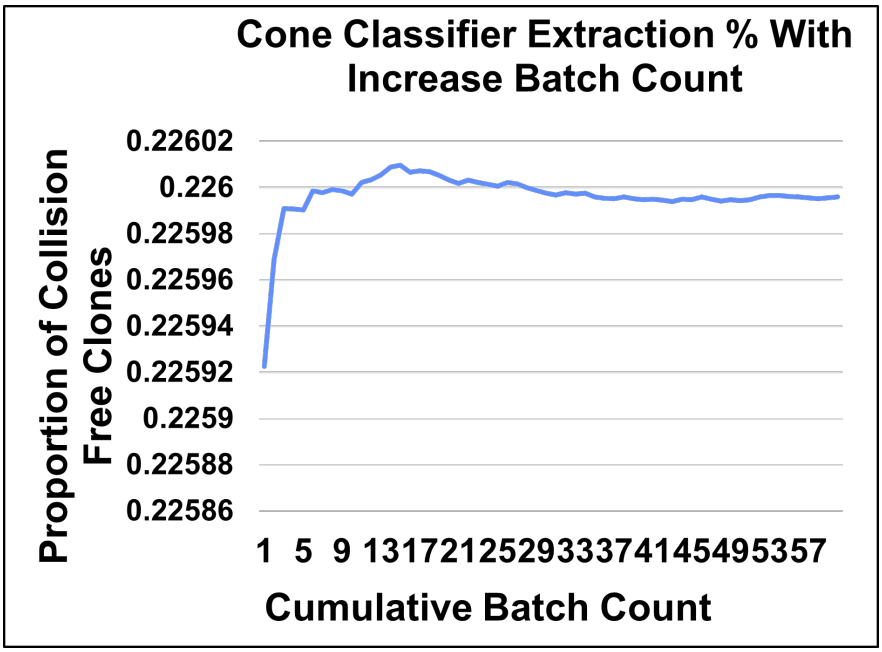


Figure 6.27: Chart Showing Batch Size Used to Calculate APEINdex CCSP Statistic.

6.27 shows that the used batch size is ample. The explanations are as follows. Data: biometric records. Raw data size (RDS): 6,068. Clone/sample size: 163,836. Collision free clone: 37,027. Statistic: proportion of classifier extracted inexact clones . Batches: 57 ; 500 Runs per Batch. Statistic of 60 batches: 0.2260. Proportion standard error: 5.4%. Normal Distribution Check. (Per Empirical rule and parameters). mean, mode, median: (0.226, 0.226, 0.226) samples <= mean : 56.67%. samples in 1STD: 68.33%. samples in 2STD: 96.67%. samples in 3STD: 99.99%.

Figure 6.28 validates the APEIndex CCSP statistic. The explanations are as follows. Data: biometric records. Raw data size (RDS): 6,068. Clone/sample size: 163,836. Collision free clones: 37,027. Statistic: proportion of classifier extracted inexact clones . Inexact clones per batch: 37,027. Statistic of 60 batches: 0.226. Proportion standard error:5.4%. Clone Classifier Performance. Collision Free 22.6%. Prop Std Error 5.4% . Confidence level 96.7%. Confidence level: 96.7% = 100% * (1-(2/60)).

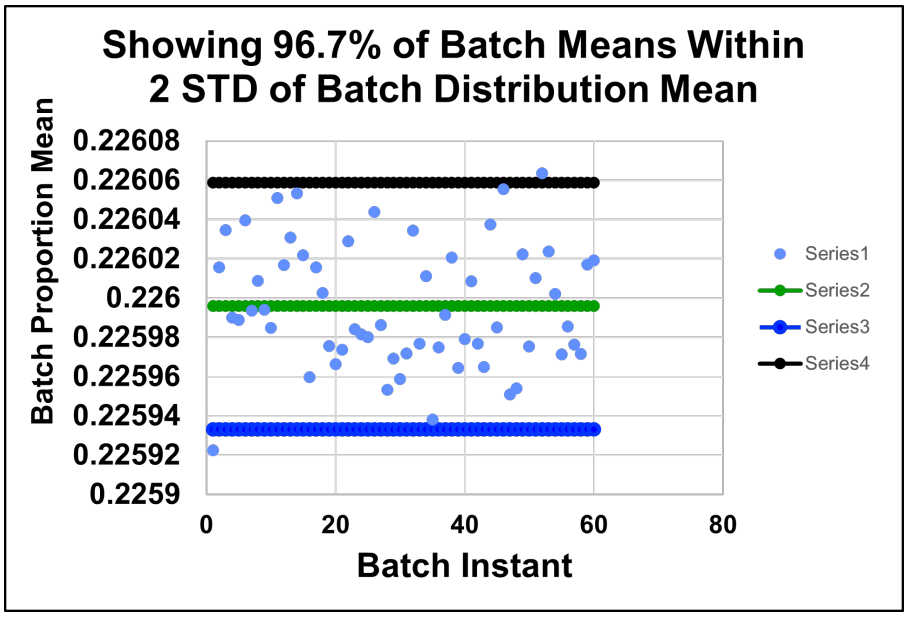


Figure 6.28: Chart Showing APEINdex CCSP Statistic Validation

Figures 5.14, 5.15 and 5.16 were used to verify the published APEIndex dataset as valid representation of its raw dataset. Figures 5.11, 5.12 and 5.13 were used to validate the published APEIndex dataset as valid representation of its raw dataset.

Data Engineer implementation report. We implemented the proposed solution for dataset APEIndex incorporation into DPPA. The implementation results were successful and DP security product for APEIndex dataset was officially became the second DP security product in DPPA.

6.5 The Benchmark

In this document, we have been using the words prototype and benchmark. We find it necessary to explain how these words fit into our context, thereby classifying DPPA as a prototype and a benchmark. We will also explain DPPA as holding a suite of benchmarks. The benchmark proposes a new way to quantify the security of a published dataset by providing the percentages for its DPPS Value and CCSP value.

6.6 Limitations of DPPA

Below are some limitations of DPPA. One-way encryption engine. The data published by DPPA is intended for semantics and not for syntax or data origin reidentification. Therefore, we can say that DPPA has properties of a one-way encryption engine. This limitation is a part of DPPA's planned restrictions. It needs mentioning to clarify the intended purpose of DPPA published data.

Demonstrated using only two products/datasets. As our DPPA grows we are expecting more products/ datasets. To date we only have two datasets. This limitation is a part of DPPA's planned restrictions. Using only two datasets was sufficient to bring our prototype int o fruition.

Deferred software dexterity such as GUI and data storage organization. Per software engineering, this work is about business logic, decision processes, customer requirement, and services orchestration. This accounts for the specialization of software middle-ware. GUI and data storage organization accounts for the specialization of front end and back end respectively.

Demonstrated data from only one domains. We consider using two different datasets from the same domain to be ample for our prototyping needs.

Apply level 1 inexact cloning mechanism. For this work, level 1 inexact cloning mechanism means all records received the same type of change. For example, to be granular, the raw records had two distinct groups: male and female. Usually, males are taller than females. We deferred using this level of granularity in our inexact cloning application process. Our randomization process compensates for granular distinction. Our choice still allows for data protection and realistic records. Additional protection comes from our classifier and our replication process was sufficient.

Automation is limited to DPP data generation and report generation. The au-

tomation we provided is ample for our middle ware purposes.

Only a subset of DPPM controlled variables were used to verify the stability of DPPA under intra variability. DPPA has DPPS and CCSP as its metrics. The selected driver variable raw data size varies DPPS and CCSP. The selected driver variable DPPM RAC varies DPPS and CCSP. These selected driver variables are easy to prove and thus can easily provide understanding to our stability verification. We have other controlled variables: SC, PDS, RCR that we could also vary. However, we found using our main two variables: raw data size and DPPM RAC to be adequate. It is our methodology that counts. We want to share our methodology. Had we showcase all our controlled variables as driver variables or independent variables, they would follow the same methodology. They would have their respective test case showing their distinctive observation charts.

6.7 Use Cases of DPPA

We have successfully demonstrated incorporating two security products into the DPPA family of services. These security products perform IDC security mechanism. Via our automation, they generate published data and diagnostic reports. As our DPPA grows we are expecting more products. DPPA is a benchmark, since to our knowledge we have not seen any implementations with DPPS and CCSP with percentages. More importantly, we have not seen any implementation that are flexible with protecting private and preserved variable as we do. We protect or preserve one variable while keeping the statistic of another. we called DPPA a prototype since it is a proof of concept.

Following our same trend, DPPA can be implemented into a data security business.

Chapter 7

CONCLUSION

7.1 Summary

For the initial section of our work, we propose a novel DP protection mechanism called IDC which preserves and protects the privacy of raw data records by creating and publishing similar copies, inexact clones (IC). The inexact clones are created in such a way that it preserves certain statistics of the raw data while maintaining privacy. Moreover, we can make a decision about which parameters and statistics to preserve while creating and publishing ICs.

IDC aims to create data that are valid and representative of the raw data. In this work, we introduce five differential privacy protection metrics (DPPM) that cumulatively define and quantify the privacy protection level of the IDC mechanism. They are Shift Count: SC, Published Dataset Size: (PDS), Record Attribute Count: (RAC), record clone ratio: (RCR), and raw data cloneset ratio: (RDCR). Each component ranges from 0 to 1 and their cumulative value forms the differential privacy protection score (DPPS) performance of IDC. IDC has another performance, a metric called clone classifier selection percentage (CCSP) which denotes the percentage of published inexact clones selected out of all the created cloneset. The classifier ensures that there is no major collision between the published clones and the raw data.

We test the IDC mechanism on a large open-source biometric data set of about 6K subjects, with records of their height, weight, hip, waist, BMI, and WHR. We aim to protect the first four parameters (weight, height, hip, and waist) and preserve the two derived parameters BMI and WHR. We show that we can achieve a DPPS of

86.16% with a CCSP of 52.6%. We further show that we can preserve the statistics of the BMI and WHR, and scramble the distributions of the other four parameters, thus ensuring privacy.

We are confident that our work has a competitive advantage in the area of data security, where providing deterministic quantitative data security protection metric values are needed. This need arises when customers need assurance of their data security promises. The deterministic methods are advanced statistically backed heuristics that produces quantitative values for DPPS and CCSP. DPPS gives; for example, the data scientist an operational obscurity level of their published data. Obscurity in this case means the extent to which sensitive data are hidden. Operational in this case means the data might be constantly viewed or used by many audiences. Our methodology is scalable for data sizes and adjusted to raw data column duplicates in published data.

For the final section of our research, we proposed and implemented a prototype DPPA which incorporated IDC as a service. Chapter 6 contains the findings of the final section of our research. DPPA was created to facilitate a market for a class of DP security products. IDC is the first security product incorporated into DPPA. DPPA provides to IDC, automation for dataset creation and diagnostic report generation. IDC contributed its engineering security mechanism to DPPA. The IDC mechanism is heavily dependent on domain knowledge. By domain knowledge we mean, using raw dataset relations for the creation of comparable raw dataset substitutes. IDC changed, restructured, and modified raw data records to produce new inexact records which maintain the signature or characteristics or relations of the raw datasets. IDC hides and protects the identities in the raw dataset. DPPA is created as a prototype for benchmark purposes and thus its independent components are restrictive. The restrictions were needed to facilitate the usage of various independent mathematical

variables working towards a common goal, dependent variables. These dependent variables are DPPS and CCSP.

After analyzing the design of IDC and its integration into DPPA, we claimed that DPPA and IDC can accept datasets from different domains. Example domains are like finance, medical, biometrics and sports. We stand by this claim and thus move to explain its transformation. We are interested in the mathematical signature of the relations of a raw dataset. This means a customer can give us their dataset and its mathematical signature without telling the domain of the raw dataset. In this case we can assign a domain name via number assignment. We can see that after receiving the mathematical signature, the dataset domain is not so much the focus. Another case is that engineering science can generate the signature of a dataset; however, we are not going that route. We are depending on DPPA engineers to gain domain knowledge via corroboration with raw dataset experts. The engineers will then use this knowledge to create mathematical constructs of the raw dataset. These constructs will be stored in the DPPA repository. DPPA engineers that provide mathematical constructs are like independent engineering contractors. Hence the domain knowledge dependency is understood.

7.2 Research Continuation

The work completed by this effort is part of an end-to-end package that aims to establish an automated benchmark process for validating the security protection of a given dataset. Since this is a new effort, every part of the completed work can be improved. There are three main interests of this work, function, automation, and optimization. The function of interest is to create and publish datasets of inexact clones. This function is completed and automated. The next interest is to optimize the classifier functionality. At present, the classifier discards about 50 percent of

all produced IC records. Optimization can include repositioning these discarded IC records. With that said, this optimization suggestion is considered secondary. The primary optimization is the automation of Chapter 5. This automation is complete and is validated by the generation of diagnostic reports. Automation of Chapter 5 has provide one end-to-end proof case for the operational use of this work. Besides having confidence in the DPPS and CCSP score, printing out a diagnostic sheet showing PDFs and comparing statistics graphs has provide assurance and confidence that this work is can be realized commercially.

A research continuation recommendation is the optimization of the code for exploitation of advances in current computer hardware technologies. Having a faster machine will help to increase batch sizes so that our metrics can improve in accuracy. With faster executions, we could increase our sample sizes to 1000 and thus get more accurate metrics. Other possible research continuation are as follows. We improve and investigate various ways to generate clonesets: 1) Experiment to see a) what is an optimal shift count or b) what is an optimal attribute count, before the classifier's performance starts to degrade. 2) Investigate the classifier's collision process, there might be a simpler way. 3) Write specifications for preserving and protecting parameters for specialized datasets. 4) Build a catalog for shifts. A catalog would be necessary for long-term business. The customer's dataset will always change and so we would aim to accommodate the customers. 5) write a dataset relation search engine to accommodate automation of dataset relations new to DPPA.

This document in its current state has room for reconstruction. The diagnostic report and the presentation for this work forms the cohesive driver for high level understanding of this research. Chapter 6 is an annexation to chapters 1,2,3,4 and 5. Where time was given, Chapter 6 had priority for updates. Since the latter part of this research was based mainly on automation and generation of diagnostic reports,

emphasis was given to areas where these could be demonstrated; presentation and diagnostic reports. There is still room for improvement in associated research areas. We recommend research continuation especially in the area of classifier optimization for advance hardware functionalities utilization. We wanted to include more plots that could allow us to track granular accuracy. Enabling us to increase our speed of testing, would have allow us to explore current and more, optimization options. This research was quite a journey and it has taught us a lot.

REFERENCES

[1] "Proactive investors: Faang report: Facebook may be fined $1.63b by eu over data breach; netflix to let you choose ending of your show," Oct 01 2018.

[2] L. A. Daming, "How to stay within the law when using biometric information: Ensure compliance with biometric privacy statutes and data-breach laws," *HRNews*, Apr 03 2018.

[3] M. Jeong, C. Lee, J. Kim, J.-Y. Choi, K.-A. Toh, and J. Kim, "Changeable biometrics for appearance based face recognition," in *2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pp. 1–5, 2006.

[4] E. Chikofsky and J. Cross, "Reverse engineering and design recovery: A taxonomy," *IEEE Software*, vol. 7, no. 1, pp. 13–17, 1990.

[5] J. Sadiq and T. Waheed, "Reverse engineering & design recovery: An evaluation of design recovery techniques," in *International Conference on Computer Networks and Information Technology*, pp. 325–332, 2011.

[6] J. H. Kim, D. Simon, and P. Tetali, "Limits on the efficiency of one-way permutation-based hash functions," in *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*, pp. 535–542, 1999.

[7] The Law Dictionary, "Reference analysis."

[8] G. Barthe, B. Köpf, F. Olmedo, and S. Zanella-Béguelin, "Probabilistic relational reasoning for differential privacy," *ACM Trans. Program. Lang. Syst.*, vol. 35, no. 3, p. 1–49, 2013.

[9] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, p. 211–407, aug 2014.

[10] N. Almadhoun, E. Ayday, and O. Ulusoy, "Differential privacy under dependent tuples—the case of genomic privacy," *Bioinformatics*, vol. 36, pp. 1696–1703, 11 2019.

[11] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 4037–4049, 2017.

[12] X. Li, C. Luo, P. Liu, and L.-e. Wang, "Information entropy differential privacy: A differential privacy protection data method based on rough set theory," in *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pp. 918–923, 2019.

[13] J. Steil, I. Hagestedt, M. X. Huang, and A. Bulling, "Privacy-aware eye tracking using differential privacy," in *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, ETRA '19, (Denver, Colorado), Association for Computing Machinery, 2019.

[14] C. Dwork, "Differential privacy," in *Automata, Languages and Programming* (M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, eds.), (Berlin, Heidelberg), pp. 1–12, Springer Berlin Heidelberg, 2006.

[15] W. Huang, S. Zhou, Y. Liao, and H. Chen, "An efficient differential privacy logistic classification mechanism," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10620–10626, 2019.

[16] D. Sadhya and S. K. Singh, "Privacy preservation for soft biometrics based multimodal recognition system," *Computers & Security*, vol. 58, pp. 160–179, 2016.

[17] Y. Han, S. Li, Y. Cao, Q. Ma, and M. Yoshikawa, "Voice-indistinguishability: Protecting voiceprint in privacy-preserving speech data release," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, (Los Alamitos, CA, USA), pp. 1–6, IEEE Computer Society, jul 2020.

[18] Y. Yao, Z. Wang, and P. Zhou, "Privacy-preserving and energy efficient task offloading for collaborative mobile computing in iot: An admm approach," *Computers & Security*, vol. 96, pp. 1–10, 2020.

[19] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pp. 263–275, 2017.

[20] T. Dalenius, "Towards a methodology for statistical disclosure control," *Statistik Tidskrift*, vol. 15, p. 429–444, 1977.

[21] S. E. Fienberg and J. McIntyre, "Data swapping: Variations on a theme by dalenius and reiss," *Journal of Official Statistics*, vol. 21, no. 2, p. 309–323, 2005.

[22] G. T. Duncan and D. Lambert, "Disclosure-limited data dissemination," *Journal of the American Statistical Association*, vol. 81, no. 393, pp. 10–18, 1986.

[23] A. A. de Amorim, M. Gaboardi, J. Hsu, and S.-y. Katsumata, "Probabilistic relational reasoning via metrics," *arXiv*, 2018.

[24] A. Friedman and A. Schuster, "Data mining with differential privacy," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, (Washington, DC, USA), p. 493–502, Association for Computing Machinery, 2010.

[25] R. Hall, A. Rinaldo, and L. Wasserman, "Differential privacy for functions and functional data," *J. Mach. Learn. Res.*, vol. 14, p. 703–727, feb 2013.

[26] D. US Army Natick Soldier Research and E. Center, "Ansur ii." Distributed by the University of Michigan, 2012.

APPENDIX A

LIST OF ACRONYMS

BMI: Body Mass Index
CCSP: Clone Classifier Selection Percentage
DP: Differential Privacy /Differentially Private
DPPA: Differential Privacy Protection Architecture
DPPM: Differential Privacy Protection Metrics/Mechanisms
DPPS: Differential Privacy Protection Score
GAHPPM: General Algorithm Heuristic Privacy Mechanism
IC: Inexact Clones
IEDPPM: Information Entropy Differential Privacy Protection Mechanism
IDC: Inexact Data Cloning, Inexact Data Clones
PDF: Probability Density Function, Portable Document Format
PPP: Parameter Protection Preference
PPC: Protected Parameter Count
PMESCS: Privacy Mechanism Execution Specification check Sheet
PDS: Published Dataset Size
RAC: Record Attribute Count
RCR: Record Clone Ratio
RDCR: Raw Data Cloneset Ratio
SC: Shift Count
ASU: Arizona State University
BLISS: Bliss Laboratory of Information, Signals, and Systems

BIOGRAPHICAL SKETCH

Zelpha; a native of Jamaica, graduated from Shortwood Teachers' College in Kingston as a Secondary Education Teacher for Physics, Chemistry and Biology. Zelpha taught Mathematics and Physics at Hillel Academy in Cherry Gardens; Kingston, before graduating from Brigham Young University with a BS in Computer Science. Zelpha received Masters in Industrial Engineering and Computer Engineering from University of Florida. Zelpha completed her Ph.D. degree in Computer Engineering; specializing in Differential Privacy Protection, at Arizona State University with BLISS Labs and WISCA. Some of Zelpha's engineering work experiences include: Software Engineering for online retailing, Software Testing and Data Extraction as an Integration Engineer and metrics implementation for complex systems performances. Some of Zelpha's other experiences include industrial logistics for food delivery, passenger transportations and warehouse operations.