UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Tècnica Superior d'Enginyeria
de Telecomunicació de Barcelona

Karlsruher Institut für Technologie

Utility of Semantic Privacy Notions for Correlated Data

Master Thesis
submitted to the Faculty of Informatics
Karlsruhe Institute of Technology
by

Martin Lange

In partial fulfillment
of the requirements for the master in
INFORMATICS / COMPUTER SCIENCE

Advisors: Patricia Guerra-Balboa and Dr. Javier Parra-Arnau
Reviewer: Prof. Dr. Thorsten Strufe
Second Reviewer: Prof. Dr. Jörn Müller-Quade

Barcelona, 27/06/2024

Karlsruher Institut für Technologie

Contents

List of Figures

List of Tables

Abstract

In research on data privacy, differential privacy (DP) is the method of choice to measure privacy leakage while conducting data analysis. A DP algorithm is an algorithm that limits the privacy leakage when its result is released publicly. However, privacy attacks are possible against DP algorithms if the underlying data is correlated – even if the privacy leakage according to DP is low. As a response, researchers have proposed various alternative semantic privacy notions that measure privacy leakage while taking correlation into account. However, increasingly strict privacy measurements call into question whether this allows for sufficient utility for data analysis algorithms. The theoretical limits on their utility remain unclear.

Therefore, this thesis investigates the utility of algorithms that limit privacy leakage according to Bayesian differential privacy (BDP), a common extension of DP. We directly prove statements about the utility of BDP algorithms, as well as how much the privacy leakage of a DP algorithm increases when measuring it with BDP. We present novel theoretical results for various correlation models of the data: arbitrary correlation, genomic data, multivariate Gaussian correlation and data following a Markov chain. We find that BDP algorithms cannot provide good utility when the correlation model is completely arbitrary. However, we also show that the increase in measured privacy leakage from DP to BDP is highly dependent on the specific correlation model. These results enable the creation of BDP algorithms with high utility for correlation models where the increase in privacy leakage is relatively small. For example, high utility BDP algorithms on data from a multivariate Gaussian correlation model are possible if the strength of correlation is sufficiently weak – even if all records in the data are correlated with each other. Additionally, we empirically test the utility of BDP algorithms on real-world data sets and find improvements over the state of the art.

1 Introduction

Data analysis is becoming increasingly important and has enabled many breakthroughs in different fields such as social science and health. However, data analysis also enables serious privacy violations [42]. For example, Sweeney [46] found in 2000 that the majority of the US citizens can be uniquely identified in a data set if it includes their ZIP code, their gender and their date of birth. At the time, such information was publicly available in hospital discharge data [46].

Since then, *differential privacy* (DP) [15] has emerged as the standard privacy metric in data analysis [29]. It is a notion of *statistical disclosure control* [23], which aims to release global statistics about a data set, while keeping the records of individuals private. DP is a *semantic privacy notion* [29] that measures the *privacy leakage* ε of an algorithm processing the dataset, whose output is released publicly. To limit the privacy leakage, the output of a non-trivial algorithm must be randomized so that an adversary cannot confidently identify the original input data. More specifically, a DP algorithm must generate outputs with "similar" probability for neighboring data sets. Two data sets are neighbors if they only differ in one record. The similarity in the output distributions is measured by the privacy leakage $\varepsilon > 0$. The smaller ε, the higher the similarity and the stronger the privacy guarantees. Therefore, since the probability of the output is similar (up to ε), an adversary cannot distinguish (up to ε) whether the input database contains a targeted record, independent of the auxiliary knowledge. More formal interpretations of the privacy guarantees of DP have been proven in [25].

When conducting data analysis with DP algorithms, we need to consider the *privacy-utility trade-off* [23]: Offering strong privacy protections for individuals comes at the cost of utility of the data analysis: E.g., the strongest privacy protection would be to release nothing at all, offering no utility. On the other hand, high utility compromises privacy: E.g., if the we simply release the original data set to maximize utility to data analysts, privacy is destroyed.

DP offers several benefits such us being independent of the adversary's prior background knowledge: A DP algorithm limits the *posterior-to-posterior* change in the belief of *any* adversary. The adversary's beliefs after seeing the output of the algorithm are roughly equal to their beliefs after seeing the output for a neighboring data set [24]. Nonetheless, the guarantee of limited posterior-to-posterior belief change is weaker than a limit on *prior-to-posterior* change: Here, one would limit the change in beliefs of the adversary before and after seeing the output of the algorithm. Figure 1 visualizes the difference between posterior-to-posterior and prior-to-posterior. However, the use of a privacy notion offering a bound on all prior-to-posterior belief changes is understood to offer no utility [29]. This follows from an impossibility result of Dwork and Naor [13]. Therefore, DP has been the standard privacy metric able to produce a useful privacy-utility trade-off.

However, researchers have discovered that DP algorithms can leak private information if the underlying data is correlated [26, 51, 56]. Correlation among data records is a common attribute of real world data sets, e.g., consider friendships in a social network [31] or the genome of family members [1]. Kifer et al. [26] were first to observe unexpected inference of private information under DP. They demonstrate that an adversary can detect the

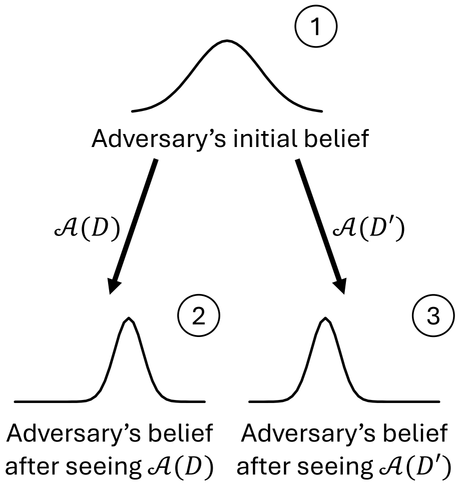

1
Adversary's initial belief
$\mathcal{A}(D)$
$\mathcal{A}(D')$
2
3
Adversary's belief after seeing $\mathcal{A}(D)$
Adversary's belief after seeing $\mathcal{A}(D')$

Figure 1: Difference between posterior-to-posterior and prior-to-posterior. The belief of the adversary about a private piece of information, e.g., the income of a person, is represented as a distribution over all possible incomes. This belief changes when the adversary sees the output of algorithm $\mathcal{A}$ with input data D, or when they see the output of $\mathcal{A}$ with a different input D'. The difference between (2) and (3) is the *posterior-to-posterior* change in the adversary's belief. The difference between (1) and (2) – or between (1) and (3) – is called the *prior-to-posterior* change.

presence of a friendship in a social network. This has led some to claim that DP makes an implicit assumption of independent data [10, 26, 32]. This is not the case - its mathematical definition holds both for correlated and independent data [49]. However, this mathematical definition does not always imply the privacy guarantees and interpretation that researchers expect of DP when data correlations are present.

In response, researchers have proposed various new semantic privacy notions to explicitly address the problem of correlation and data privacy. The most notable ones comprise Bayesian differential privacy [55], dependent differential privacy [32], and blowfish privacy [21]. Similarly to DP, Bayesian differential privacy (BDP) also requires that an algorithm publishes "similar" results if a single record – the so-called *target record* – in the data set changes its value. Here, similarity is measured by the so called *Bayesian differential privacy leakage* (BDPL), analogous to the DP privacy leakage. However, unlike DP, the values of the remaining records are not fixed in this comparison. Instead, they are drawn from the conditional probability distribution of the remaining data set, given the target record. Thus, the effect of correlation from the targeted record on the remaining data set is considered. In this thesis, we focus on BDP in favor of the other two privacy notions for two reasons: (1) The definition of BDP does not restrict the correlation that is possible between records, unlike dependent differential privacy and blowfish. (2) Various other privacy notions [11, 30, 50, 54] published in smaller conferences or journals are variations of BDP. Thus, by focusing on BDP the results of our research will apply to a number of semantic privacy notions.

Yang et al. [55] show that a BDP algorithm limits the difference in prior-to-posterior odds for the adversary's belief of the value of a specific record. While BDP does not limit all prior-to-posterior changes, it does limit changes in belief about the value of records arising from the results of the algorithm and the correlation between records. Given the aforementioned results that show the complexity of retaining global statistics under a prior-to-posterior metric [13, 29] it is questionable whether BDP algorithms can provide sufficient utility in general. It is an important question whether a BDP algorithm can provide sufficient utility, as algorithms with poor utility will not be adopted, even if they offer high privacy. The theoretical bounds on the utility of BDP algorithms have not yet been investigated to the best of our knowledge.

Consequently, this thesis will investigate the following research questions: What utility can BDP algorithms achieve when we make no assumptions about the correlation within the data? Additionally, what utility can be achieved when a certain correlation model is assumed (i.e., genomic data, data with a multivariate Gaussian correlation model, or data following a Markov chain)? To answer these questions, we employ theoretical proofs, simulations with synthetic data and an experiment on real-world data. In our proofs, utility will be measured using the metric (α, β)-*accuracy* which calculates the maximum probability β of an error of magnitude α [5]. In many cases, we prove the accuracy of a BDP algorithm by reducing it to the accuracy of a DP algorithm. This involves proving the relationship between DP and BDP, i.e., proving an upper bound on how much the BDPL increases compared to the DP privacy leakage. Additionally, we run Monte Carlo simulations to estimate the BDPL empirically and indicate whether these upper bounds are tight. Finally, the experiment tests whether these bounds improve utility of BDP algorithms on real-world data.

The thesis makes the following main contributions:

- We prove the *general bound* that a DP algorithm with privacy leakage ε has a BDPL of at most $m\varepsilon$, where m is the number of correlated records. We show that this bound on the BDPL is tight for arbitrary correlation models, and how the bound impacts the (α, β)-accuracy of DP algorithms used as BDP algorithms. Additionally, we prove a lower bound of (α, β)-accuracy for *any* BDP algorithm processing data with arbitrary correlation.

- We prove that the general bound is nearly tight for genomic data: A counting query with DP privacy leakage ε has a BDPL of at least $(m-1)\varepsilon$, where m is the size of the largest family in the data. Here, we also prove a lower bound of (α, β)-accuracy for any BDP counting query under genomic correlation.

- We prove the *Gaussian bound* on the BDPL of certain DP algorithms when data has a multivariate Gaussian correlation model. We show that this improves upon the state of the art when all records are "weakly" correlated, and how it impacts the (α, β)-accuracy of DP algorithms used for BDP. Our simulations suggest that the bound can be further improved.

- We prove the *Markov chain bound* on the BDPL of DP algorithms when data follows a Markov chain. This improves on the state of the art when the transition probabilities of the Markov chain are sufficiently similar. Again, we show the (α, β)-accuracy of DP algorithms when using this bound to provide BDP guarantees. Simulations indicate that a better bound is possible.

- We test the utility of BDP algorithms on real-world data sets that correspond to the correlation models we have investigated in our theoretical proofs. We find that using the Gaussian bound and the Markov chain bound yields utility improvements compared to the state of the art.

Overall, this thesis shows when privacy can be protected while keeping useful global statistics, and when both objectives cannot be met at the same time. These novel results expand the research community's understanding of the conflict between privacy and utility under correlation. The improved bounds on the BDPL of DP algorithms enable the safe application of more existing DP algorithms to situations with correlation. This is not limited to the algorithms presented in this thesis.

The remaining parts of this thesis are organized as follows: We first discuss important background information on correlation and different privacy notions such as DP and BDP. This is followed by the problem statement, and how we address it with our methodology. In the main body of the thesis, we investigate the utility of BDP algorithms in four different situations: (1) data with arbitrary correlation, (2) genomic data, (3) data with a multivariate Gaussian correlation model, and (4) data following a Markov chain. Subsequently, we test the utility of BDP algorithms on real-world data sets. Finally, we present the conclusions from our work. In addition, a glossary of commonly used terms and abbreviations is found after the bibliography.

Notation	Description
$\mathcal{X}$	Domain of a single record $x \in \mathcal{X}$.
$D = (x_1, \dots, x_n)^\top \in \mathcal{X}^n$	Data tuple with n records.
$d_H(D, D') = \sum_{i \in [n]} \mathbb{1}\{x_i \neq x'_i\}$	Hamming distance of data tuples $D, D' \in \mathcal{X}^n$.
$\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathcal{X}^n$	Random vector. Data tuple D is drawn from the distribution of $\mathbf{X}$.
$[n]$	Set $\{1, \dots, n\}$ for $n \in \mathbb{N}$.
$\mathbf{X}_K = (X_{i_1}, \dots, X_{i_k})^\top$	Random vector of a subset $K = \{i_1, \dots, i_k\} \subseteq [n]$ of the random variables $X_1, \dots, X_n$.
$\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$	Randomized algorithm with input from domain $\mathcal{X}^n$ and output from codomain $\mathcal{Y}$.
Y	Random variable representing output of algorithm $\mathcal{A}$.

Table 1: Notation

2 Background

This section introduces privacy notions and the concepts of correlation that will be important throughout the thesis.

First, the notation. In this thesis, $\mathcal{X}$ denotes the domain of a single record $x \in \mathcal{X}$. A data tuple with n records is $D = (x_1, \dots, x_n)^\top \in \mathcal{X}^n$ and the hamming distance $d_H(D, D')$ between two data tuples of equal size n is the number of records that differ between them. As we are dealing with correlation between records in D, we must represent the distribution that D is drawn from. Here, $\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathcal{X}^n$ is a random vector and D is drawn from the distribution of $\mathbf{X}$. It will often be useful to refer to a subset of the random variables $X_1, \dots, X_n$. Let $[n]$ denote the set $[n] := \{1, \dots, n\}$, and let $K \subseteq [n]$ be a set of indices with cardinality $k = |K|$. Then, random vector $\mathbf{X}_K \in \mathcal{X}^k$ is defined as $\mathbf{X}_K := (X_{i_1}, \dots, X_{i_k})^\top$ for indices $\{i_1, \dots, i_k\} = K$. Throughout the thesis, $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is a randomized algorithm that takes a data tuple $D \in \mathcal{X}^n$ as input and returns an output from the codomain $\mathcal{Y}$. The output of $\mathcal{A}$ is also represented by the random variable $Y \in \mathcal{Y}$ that depends on the randomization of $\mathcal{A}$ and the random vector $\mathbf{X}$. An overview over the notation is given in Table 1.

2.1 Differential Privacy

This section introduces differential privacy, its intuitive meaning, how to achieve DP algorithms, and a theoretical measure of the accuracy of a DP algorithm. First, we define differential privacy:

Definition 2.1 (Differential Privacy [15]). A randomized algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is called *ε-differentially private*, if and only if for all measurable sets $S \subseteq \mathcal{Y}$, for any target index

$i \in [n]$, for any target values $x_i, x_i' \in \mathcal{X}$, and for any remaining values $\mathbf{x} \in \mathcal{X}^{n-1}$, we have

$$\Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}, X_i = x_i] \leq e^{\varepsilon} \Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}, X_i = x_i']. \tag{1}$$

Intuitively, this means the likelihood of a specific output of an ε-DP algorithm only changes a little (at most by $\exp(\varepsilon)$) if a single record in the data tuple has a different value and all other records stay the same. The idea is to make it difficult for an adversary to distinguish between the two scenarios when observing the result of the ε-DP algorithm. A more concrete interpretation of the meaning of DP is given by Wasserman and Zhou in terms of hypothesis testing [53]: A statistical test between hypotheses $H_0 : X_i = x_i$ and $H_1 : X_i = x_i'$ that uses the output of an ε-DP algorithm has its informativeness bounded by a function of the privacy leakage ε. It is important to note that this result assumes that all records are independently and identically distributed. We will later see that breaking this assumption of independence enables privacy attacks against DP algorithms.

The level of privacy-protection of an ε-DP algorithm depends on two factors:

1. The neighborhood definition encapsulates which information can change while the probabilities of the output must remain similar up to e^{ε}. In other words, it defines which situations are indistinguishable for the adversary. Definition 2.1 corresponds to DP with a *bounded* neighborhood, i.e., two situations with data tuples of equal size are compared. Other variations include the *unbounded* neighborhood notion in which the sizes of the data tuples differ by one, i.e., one record has been added or removed. However, we focus on a bounded neighborhood due to its broad applicability and relevance as well as its close relation to BDP, whose definition will follow.

   A second dimension of the neighborhood notion is *user-level* or *event-level* DP [14]. In event-level DP, we aim to hide the presence or absence of a single event, so the neighborhood notion only compares situations in which the data tuples differ by a single event (e.g., a single location of a user). For user-level DP, the neighborhood notion compares situations in which the data tuples differ by all events related to a single user (e.g., all locations of a user). Definition 2.1 does not make explicit whether it is user-level or event-level DP. Rather, this depends on whether a single record $x \in \mathcal{X}$ from data tuple D contains all information of a single user (user-level DP), or only a part of it (event-level DP).

   A survey on the different neighborhood definitions can be found in [12].

2. The privacy leakage ε controls the privacy-utility trade-off. A smaller ε means that the output distributions for different input data tuples are "closer together", and thus the participants enjoy higher privacy as it is more difficult for an adversary to distinguish the different data tuples. Likewise, a bigger ε allows larger differences in distribution, making it easier to distinguish them and therefore reducing privacy. We will see how ε has a reversed influence on utility at the end of this section.

It is not directly clear what values of ε are sufficiently small for "good" privacy protection. In [28], the authors find that the value of ε to prevent a disclosure risk of a record that is processed by an ε-DP algorithm depends on the domain of the record, and the query that is computed. In their analysis of different situations, they find appropriate values of ε between 0 and 5, with $\varepsilon = 5$ once resulting in a 97% risk of disclosure. In practice, even algorithms with considerably higher privacy leakages $5 < \varepsilon < 20$ are considered to "provide robust privacy protection in a variety of settings" according to the National Institute of Standards and Technology [36, p. 1]. These algorithms have been shown to resist certain specific privacy attacks [8], but have not been proven to give theoretical guarantees of privacy protection.

There are a variety of methods to limit the privacy leakage of an originally deterministic (non-private) function f [15]. Here, we introduce the Laplace mechanism, one of the earliest and most common methods. It relies on the sensitivity of the function, and on the Laplace distribution.

Definition 2.2 (Sensitivity [15])**.** Let $f : \mathcal{X}^n \to \mathbb{R}$ be a function. Let $D, D' \in \mathcal{X}^n$ be bounded neighbors, i.e., they have hamming distance $d_H(D, D') = 1$. The *sensitivity* of function f is defined as

$$\Delta f := \sup_{d_H(D,D')=1} ||f(D) - f(D')||_1. \tag{2}$$

Definition 2.3 (Laplace Distribution [27])**.** We say random vector $Z \in \mathbb{R}$ follows the *Laplace distribution with location* $\mu \in \mathbb{R}$ *and scale* $b > 0$ if the probability density of Z at point $x \in \mathbb{R}$ is

$$p_Z(x) = \frac{1}{2b} \exp\left(\frac{-|x - \mu|}{b}\right). \tag{3}$$

The following definition and theorem from [15, p. 32] introduce the Laplace mechanism.

Definition 2.4 (Laplace Mechanism [15])**.** Let $f : \mathcal{X}^n \to \mathbb{R}$ be a function and let privacy parameter $\varepsilon > 0$ be positive. Let $Z \in \mathbb{R}$ be a random variable that follows the Laplace distribution with location 0 and scale $\Delta f/\varepsilon$. Then we call randomized algorithm

$$\mathcal{M}_{\varepsilon,f} : \mathcal{X}^n \to \mathbb{R}, D \mapsto f(D) + Z \tag{4}$$

the *Laplace mechanism*.

Theorem 2.1 ([15])**.** The Laplace mechanism $\mathcal{M}_{\varepsilon,f}$ is ε-DP.

As we can see, the Laplace mechanism consists of adding noise to the output of the deterministic function f to limit its privacy leakage. The noise is sampled from a Laplace distribution with scale $\Delta f/\varepsilon$.

Adding noise to the output of a function f undoubtedly has an impact on utility. This impact can be quantified by (α, β)-accuracy, a well-established metric of utility [5, 32].

Definition 2.5 ((α, β)-Accuracy [5])**.** A randomized algorithm $\mathcal{A}$ is (α, β)*-accurate with respect to function* f if for all data tuples $D \in \mathcal{X}^n$ we have

$$\Pr[|\mathcal{A}(D) - f(D)| \geq \alpha] \leq \beta. \tag{5}$$

In different words, a randomized algorithm $\mathcal{A}$ is (α, β)-accurate if an error of magnitude α has a probability of at most β. Thus, the smaller α and/or β, the better the accuracy of algorithm $\mathcal{A}$. We can calculate the (α, β)-accuracy of an algorithm that has achieved DP through the Laplace Mechanism by applying the following corollary from [15, p. 34]:

Corollary 2.2 ([15])**.** Let $\mathcal{M}_{\varepsilon,f}$ be the Laplace mechanism. Let $\beta \in (0, 1]$ be a probability. Then $\mathcal{M}_{\varepsilon,f}$ is (α, β)-accurate with respect to f with

$$\alpha = \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon}. \tag{6}$$

Note that the error α in Equation (6) is the smallest possible error α with probability β for the Laplace mechanism: There is no smaller $\alpha' < \alpha$ so that algorithm $\mathcal{M}_{\varepsilon,f}$ is (α', β)-accurate. This is apparent from the proof of Corollary 2.2 in [15, p. 34].

As an example of an application of Corollary 2.2, consider a sum query

$$f : \{0, 1\}^n \to \mathbb{R}, D = (x_1, \ldots, x_n)^\top \mapsto \sum_{i \in [n]} x_i. \tag{7}$$

This query has sensitivity $\Delta f = 1$. When we use the Laplace mechanism with privacy leakage $\varepsilon = 2$, we get a randomized algorithm $\mathcal{M}_{2,f}$. Corollary 2.2 tells us that with probability $\beta = 1\%$ we can expect an error of magnitude

$$\alpha = \ln\left(\frac{1}{0.01}\right) \cdot \frac{1}{2} \approx 2.3 \tag{8}$$

or greater. In other words, with probability 99% the output $\mathcal{M}_{2,f}(D)$ lies within a ± 2.3 range of the true result $f(D)$. If we increase the privacy leakage to $\varepsilon = 4$, this range halves to ± 1.2. In the second case, we have higher utility because the result of the algorithm will likely be closer to the true result. We see that the utility of the Laplace mechanism depends inversely on the privacy leakage ε. This is the other side of the privacy-utility trade-off introduced at the beginning of this section.

2.2 d-Privacy

The privacy notion d-privacy is a generalisation of DP that encapsulates the neighborhood notion and privacy leakage ε into a single parameter d which measures distance between data tuples [9]. It is useful because many variations of DP can be reformulated as d-privacy with a specific distance metric d.

Definition 2.6 (d-privacy [9])**.** Let $d : \mathcal{X}^n \to \mathbb{R}$ be a metric. A randomized algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is called *d-private* if and only if for all data tuples $D, D' \in \mathcal{X}^n$ and all measurable sets $S \subseteq \mathcal{Y}$ we have

$$\Pr[Y \in S \mid \mathbf{X} = D] \leq e^{d(D,D')} \Pr[Y \in S \mid \mathbf{X} = D']. \tag{9}$$

It is made difficult for an adversary to distinguish situations in which data tuples D and D' are "close together" according to metric d. However, if the two data tuples are very different, the output distribution can change more radically and it becomes easier for the adversary to distinguish. Note that the definition of d-privacy becomes equivalent to DP if metric $d = \varepsilon d_H$ is the hamming distance, multiplied by the privacy leakage $\varepsilon > 0$.

2.3 Correlation

We have seen that some interpretations of the privacy protections of a DP algorithm depend on the independence of the records in the data tuple [53]. Before reviewing how correlation weakens certain privacy guarantees by DP algorithms, we must define the concept in a mathematical fashion. Correlation is the counterpart of independence. We use the following definition of independent random variables [39, p. 157].

Definition 2.7 (Mutual Independence [39])**.** The real random variables $X_1, \dots, X_n$ are called *mutually independent* if and only if, for all $x_1, \dots, x_n \in \mathbb{R}$

$$\Pr[X_1 \leq x_1, \dots, X_n \leq x_n] = \prod_{i=1}^{n} \Pr[X_i \leq x_i]. \tag{10}$$

Two random variables X_1, X_2 are called *dependent* if they are not independent.

Originally, the term *correlation* only referred to *linear* dependence [39], which is quantified below by the Pearson correlation coefficient. However, recent research on differential privacy and correlation uses the terms dependence and correlation interchangeably [32, 55]. We adopt the same convention and use the term *linear correlation* for linear dependence to avoid confusion.

Sometimes, we also have the situation that only a subset of the random variables $X_1, \dots, X_n$ is independent. This can be expressed by group independence [40, p. 187].

Definition 2.8 (Group Independence [40])**.** We say that random variables $X_1, \dots, X_m$ are *independent* of the random variables $X_{m+1}, \dots, X_n$ if for all $x_1, \dots, x_m, x_{m+1}, \dots, x_n \in \mathbb{R}$

$$\begin{aligned}\Pr[X_1 \leq x_1, \dots, X_m \leq x_m, X_{m+1} \leq x_{m+1}, \dots, X_n \leq x_n]&\\ = \Pr[X_1 \leq x_1, \dots, X_m \leq x_m] \Pr[X_{m+1} \leq x_{m+1}, \dots, X_n \leq x_n].&\end{aligned} \tag{11}$$

Here, the random variables *within* the group $X_1, \dots, X_m$ may be correlated, but they are all independent of the group $X_{m+1}, \dots, X_n$.

An intuitive understanding of independence and correlation is given by the following property. If random variables X_1 and X_2 are independent, then for any measurable set $S \subseteq \mathbb{R}$ and for all $x \in \mathbb{R}$ we have

$$\Pr[X_1 \in S \mid X_2 = x] = \Pr[X_1 \in S] \tag{12}$$

according to [39]. However, this equality does not hold for correlated X_1 and X_2. Thus, correlation occurs when information about X_2 changes the probability distribution of X_1 and vice versa.

In this thesis, we are also interested in the *strength* of the correlation. The Pearson correlation coefficient measures the extent of linear correlation [39, p. 158].

Definition 2.9 (Pearson Correlation Coefficient [39])**.** Let X_1, X_2 be two random variables with means $\mu_1, \mu_2 \in \mathbb{R}$, variances $0 < \sigma_1^2, \sigma_2^2 < \infty$ and covariance $Cov(X_1, X_2) = \mathbb{E}[(X_1 - \mu_1)(X_2 - \mu_2)]$. We call

$$\rho_{X_1,X_2} = \frac{Cov(X_1, X_2)}{\sigma_1 \sigma_2} = \frac{\mathbb{E}X_1X_2 - \mathbb{E}X_1\mathbb{E}X_2}{\sqrt{(\mathbb{E}X_1^2 - \mathbb{E}[X_1]^2)(\mathbb{E}X_2^2 - \mathbb{E}[X_2]^2)}} \in [-1, 1] \tag{13}$$

the *Pearson correlation coefficient* of X_1 and X_2.

The larger the absolute value of ρ_{X_1,X_2} the stronger the linear relationship between the two random variables. E.g., the random variables Z and $Y := 5 \cdot Z$ have Pearson correlation coefficient 1. While independence of X_1 and X_2 implies zero linear correlation $\rho_{X_1,X_2} = 0$, it is important to note that the reverse does not hold: Pearson correlation coefficient $\rho_{X_1,X_2} = 0$ does not generally imply that X_1 and X_2 are independent; the two random variables may be correlated without any linear effect [39, p. 158].

2.4 Impact of Correlation on DP

Kifer et al. [26] show that the privacy guarantees that are often colloquially attributed to ε-DP algorithms do not formerly hold if records in the data are correlated with each other. They define the goal of a private algorithm as “Can it limit the ability of an attacker to infer that an individual has participated?” [26, p. 194]. The authors find that whether an ε-DP algorithm fulfills this goal depends on the records in the data being independent. We have adapted the following social network example from [26] to demonstrate a situation in which a DP algorithm does not fulfill this guarantee. Here, the presence of a single edge (i.e., a friendship between two users) is private and its participation in the network is supposed to be concealed.

Example 1. Caroline uses a social network. Each user is represented as a node, and each friendship is an edge between two nodes. Caroline becomes friends with Peter and subsequently, Caroline introduces her friends to Peter. As a result, multiple other friendships (i.e., edges) form between Peter and friends of Caroline (see Figure 2). Thus, these new edges are correlated with the edge between Caroline and Peter. The social network allows analysts to submit counting queries of the number of edges between groups of users. These queries are answered in a differentially private manner so that the presence of a single edge cannot be distinguished. I.e., in this case two databases are neighbors if they differ in a single edge. This is a variant of DP known as edge-differential privacy [20]. An adversary queries the network for the number of edges between (a) Peter and (b) Caroline and her two friends. The adversary can easily distinguish the scenario in which Peter and Caroline are friends (query outputs ’3’ with noise) from the scenario in which they are not (query outputs ’0’ with noise) because more than one edge is affected.

Caroline may consider her friendship to Peter private information and her privacy has been violated by the adversary. Here, DP does not properly capture the privacy leakage of the data analysis because more than one edge is affected by the participation of a

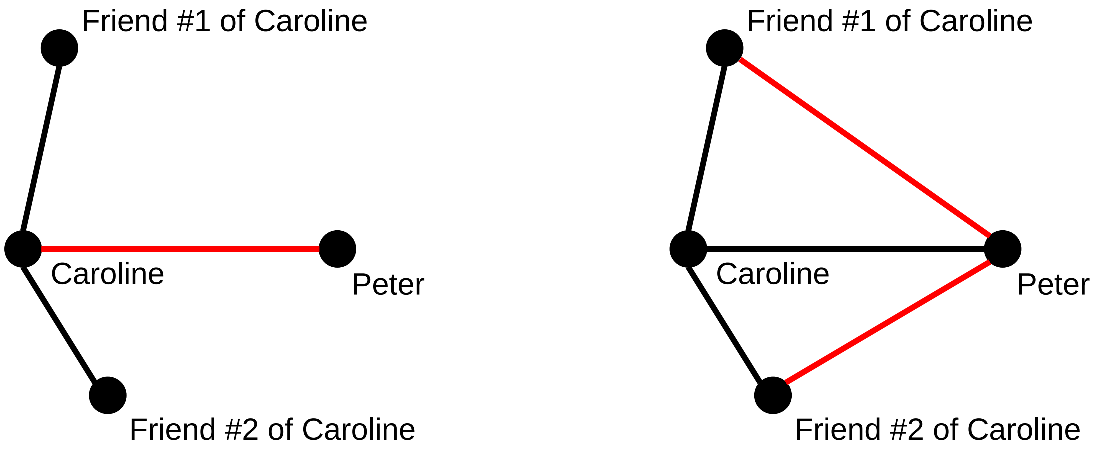


Figure 2: A social network represented as a graph. An edge is red if it is new. First, Caroline becomes friends with Peter (left network). Then, Caroline introduces her friends to Peter and they also become friends (right network).

single edge. Indeed, multiple studies have shown successful privacy attacks against DP algorithms when correlations are present [22, 26, 32].

Kifer et al. also discuss the difficulty of properly measuring the privacy leakage related to the evidence of participation. They restate the impossibility result by Dwork and Naor [13] that protecting against all inferential privacy threats and still providing useful global statistics is not achievable. This is exemplified by *Free-lunch privacy* [26]:

Definition 2.10 (Free-Lunch Privacy [26])**.** A randomized algorithm $\mathcal{A}$ is called *ε-free-lunch private*, if for any two data sets $D_1, D_2 \in \mathcal{X}^n$ and any measurable set S, we have

$$\Pr[\mathcal{A}(D_1) \in S] \leq e^{\varepsilon} \cdot \Pr[\mathcal{A}(D_2) \in S]. \tag{14}$$

For a sufficiently small ε, this notion ensures the indistinguishability between *any* two data tuples. An ε-free-lunch private algorithm would protect against any privacy attack from any adversary. However, this also means that no utility can be gained from an ε-free-lunch private algorithm because almost all information is obscured by noise [26].

2.5 Bayesian Differential Privacy

The privacy notion Bayesian differential privacy [55] also addresses the shortcomings of DP on correlated data. However, instead of always requiring indistinguishability, it depends on the adversary's prior knowledge of the data tuple. BDP is defined in two steps: First, we define the Bayesian differential privacy leakage (BDPL), a measurement of the maximum privacy violation an algorithm can cause. Subsequently, a BDP algorithm is defined as an algorithm whose BDPL is limited.

Definition 2.11 (Bayesian Differential Privacy Leakage [55])**.** Let random vector $\mathbf{X} = (X_1, \ldots, X_n)^\top \in \mathcal{X}^n$ follow distribution π. Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ be a randomized algorithm whose input data is drawn from π. Let $i \in [n]$ be the index of the targeted record by the adversary and let $K \subseteq [n] \setminus \{i\}$ be the indices of records that are known to the adversary. Then the *adversary-specific Bayesian differential privacy leakage* of $\mathcal{A}$ is

$$BDPL_{(K,i)} = \sup_{x_i, x_i', \mathbf{x}_K, S} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']}. \tag{15}$$

If the data domain is equivalent to the real numbers (i.e., $\mathcal{X}^n = \mathbb{R}^n$), Yang et al. [55] define a slight modification where the difference between x_i and x_i' is limited by a real number $M \in \mathbb{R}$:

$$BDPL_{(K,i)} = \sup_{|x_i - x_i'| \leq M, \mathbf{x}_K, S} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} \tag{16}$$

If the output Y of algorithm $\mathcal{A}$ has a continuous density p_Y, then Equation (15) is equivalent to the following formulation with the density:

$$BDPL_{(K,i)} = \sup_{x_i, x_i', \mathbf{x}_K, s} \ln \frac{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i)}{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i')} \tag{17}$$

In any case, the *Bayesian differential privacy leakage* of algorithm $\mathcal{A}$ is

$$BDPL = \sup_{K,i} BDPL_{(K,i)}. \tag{18}$$

The BDPL has a similar role to the privacy leakage ε in DP: It measures the extent of a possible privacy violation by comparing the difference in the output probabilities of algorithm $\mathcal{A}$. A lower BDPL corresponds to higher privacy because an adversary will be less likely to differentiate the two situations compared in Equation (15). The BDPL is directly used to define ε-BDP:

Definition 2.12 (Bayesian Differential Privacy [55])**.** Algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is an *ε-Bayesian differentially private* algorithm if and only if $BDPL \leq \varepsilon$.

The authors show that BDPL can be reformulated as the change between the prior odds of the adversary's belief of the value of a record, and the posterior odds [55, p. 752]:

$$BDPL_{(K,i)} = \sup_{x_i, x_i', \mathbf{x}_K, S} \left(\ln \frac{\Pr[X_i = x_i \mid Y \in S, \mathbf{X}_K = \mathbf{x}_K]}{\Pr[X_i = x_i' \mid Y \in S, \mathbf{X}_K = \mathbf{x}_K]} - \ln \frac{\Pr[X_i = x_i \mid \mathbf{X}_K = \mathbf{x}_K]}{\Pr[X_i = x_i' \mid \mathbf{X}_K = \mathbf{x}_K]} \right) \tag{19}$$

The left ratio shows the odds of record i having value x_i versus value x_i' *after* the output Y is known to the adversary (posterior belief). The right ratio shows these odds *before* the output is known (prior belief). However, in both cases the adversary can use their knowledge of the records $\mathbf{x}_K$.

2.5.1 Utility

Yang et al. [55] find that utility of a BDP algorithm is poor under certain circumstances. They assume that correlation behaves according to a so-called Gaussian correlation model and that a sum query is made private with Laplacian noise with scale λ. Then, the Bayesian differential privacy leakage grows linearly in n: $BDPL = n/\lambda$. The authors conclude that utility is too low under these assumptions because the scale λ of the noise would have to be n times larger to provide sufficient privacy guarantees. Consequently, they expand their Gaussian correlation model by adding uncertain prior knowledge of the values of all records. In their experiments, the authors show that the BDPL of a Laplace sum query is reduced if the adversary has higher a priori certainty of all records. This improves utility as the scale of noise λ no longer needs to be multiplied by n, but by a smaller factor.

Note that their Gaussian correlation model is not to be confused with the multivariate Gaussian correlation model we investigate in Section 6. Yang et al. assume that the conditional distributions of the random variables is Gaussian; the joint distribution is not necessarily a Gaussian. This is different to our assumption in Section 6: We directly assume a joint Gaussian distribution.

2.5.2 Relationship between DP and BDP

Yang et al. prove that ε-BDP always implies ε-DP. Conversely, they also show that ε-DP implies ε-BDP if all records are independently sampled. However, they do not prove any implication from DP to BDP if there is correlation present.

Zhao et al. [57] consider the situation when data follows a Markov chain. They show that ε-DP implies $q(\varepsilon)$-BDP with

$$q(\varepsilon) = \varepsilon + 6 \ln \max_{x,y,y'} \frac{\Pr[X_2 = x \mid X_1 = y]}{\Pr[X_2 = x \mid X_1 = y']}. \tag{20}$$

The BDPL depends on the privacy loss ε of the DP algorithm, and the maximum ratio between different transition probabilities to the same state in the Markov chain. This result should be met with caution because the workshop paper by Zhao et al. has not been peer-reviewed and the proof is not available. However, we have chosen to include it here because it applies to the same assumptions that we make in Section 7 on Markov chains. We will also prove an upper bound on the BDPL, and show that it improves on the result of Zhao et al. in certain situations.

3 Problem Statement

This section presents the problem statement and how we aim to solve it. The main research questions are "What utility can BDP algorithms achieve when we make no assumptions about the correlation within the data? What utility can be achieved when a certain correlation model is assumed?" These questions are important for two main reasons:

(1) The theoretical utility guarantees of BDP algorithms are unclear in many situations.

(2) BDP may be subject to the impossibility result by Dwork and Naor [13] that states that a privacy notion which limits prior-to-posterior changes in any beliefs of the adversary cannot provide utility. Yang et al. [55] have shown that BDP does limit the change in prior odds to posterior odds for the value of single records in the data tuple. Thus, it needs to be investigated whether the impossibility result applies here, or under which situations.

3.1 Methodology

Here, we explain our methodology to investigate our research questions. It consists of theoretical proofs, simulations and a utility experiment.

3.1.1 Theoretical Proofs

In this thesis, we mainly answer the research questions via novel theoretical proofs. This is advantageous to a purely empirical study: In experiments, utility can only be tested for a limited number of algorithms and data sets. In contrast, a valid theoretical proof covers any situation that falls under the assumptions of the proof.

We quantify utility with (α, β)-accuracy. Utility is improved when accuracy is improved, i.e., if the error α or its probability β is reduced. Our proofs answer the research questions from two perspectives:

(1) We show that there exists a BDP algorithm that fulfills (α, β)-accuracy for certain values of α and β. This is achieved by proving the *relationship between the DP privacy leakage and the BDPL*. I.e., we prove that an ε-DP algorithm is also a $q(\varepsilon)$-BDP algorithm for a certain function q. This allows us to reduce the accuracy of certain BDP algorithms to DP algorithms. Then, we can use accuracy results of DP algorithms to prove accuracy results of BDP algorithms.

This method is applied to arbitrary correlation models in Section 4, genomic data in Section 5, a multivariate Gaussian correlation model in Section 6, and to data following a Markov chain in Section 7.

(2) We prove that *any* BDP algorithm must have some minimum error α with minimum probability β. This shows that BDP algorithms cannot give high accuracy guarantees in certain situations. It involves proving that Definition 2.12 of BDP stands in contradiction to (α, β)-accuracy for certain correlation models. We apply this method to arbitrary correlation models and genomic data.

3.1.2 Simulations

In this subsection, we explain why and how we use Monte Carlo simulations to estimate the BDPL of DP algorithms. Our theoretical analysis involves proofs of the upper bound on the BDPL $q(\varepsilon)$ of an ε-DP algorithm for various correlation models. Initially, we do not know how tight these upper bounds are with respect to the real BDPL, i.e., it may be possible that ε-DP algorithms are actually $q'(\varepsilon)$-BDP, with $q'(\varepsilon) < q(\varepsilon)$. In our proofs, we use the function $q(\varepsilon)$ to calculate the accuracy of ε-BDP algorithms compared to ε-DP algorithms. If the increase in privacy leakage when moving from DP to BDP is actually lower than $q(\varepsilon)$, then improved utility guarantees of ε-BDP algorithms are still possible. Thus, the tightness of these bounds is of interest to our research questions: If they are not tight, then higher utility BDP algorithms remain achievable.

Thus, for each possibly untight bound, we perform a Monte Carlo simulation [34, p. 123] to estimate the output distribution of an ε-DP algorithm $\mathcal{A}$. Then, we can calculate its empirical BDPL, giving us an intuition about the tightness of our bounds. Specifically, we estimate the distribution of the outputs of algorithm $\mathcal{A}$ in the two situations that are compared by BDPL (see Definition 2.11). The estimation evaluates the adversary-specific BDPL for 1 000 adversaries. In each of the 1 000 cases, we do the following:

1. Pick a random adversary: A random number generator chooses a target index $i \in [n]$ and its possible values $x_i, x_i' \in \mathcal{X}$, a number of known records $k \in [n-1]$, their indices $K \subseteq [n]$ and their concrete values $\mathbf{x}_K \in \mathcal{X}^k$.
2. The values of the remaining unknown records are randomly drawn from the data distribution, with condition $\mathbf{X}_K = \mathbf{x}_K$ and $X_i = x_i$ or $X_i = x_i'$ respectively. This is repeated 100 000 times in both cases, for x_i and x_i' respectively. We end up with 100 000 possible data tuples $D \in \mathcal{X}^n$ in each case.
3. For each data tuple D, we sample one possible output of the DP algorithm $\mathcal{A}$.
4. Now, there are 100 000 possible outputs of $\mathcal{A}$. To estimate the BDPL, we have to estimate the probability density of the output distribution of $\mathcal{A}$. We do this twice, once for $X_i = x_i$ and once for $X_i = x_i'$. We apply state of the art kernel density estimation (KDE) [4] using the Python implementation fastKDE [38].
5. We need to evaluate the probability densities at certain points. For this, we pick 1 000 random samples t between the 0.5% quantile and 99.5% quantile of the outputs of $\mathcal{A}$. We do not evaluate the tails of the KDE because the results here are very unstable: The low density means only few outputs are sampled here and the KDE cannot give a good estimation.
6. For each sample t, we evaluate the density for $X_i = x_i$ and the density for $X_i = x_i'$ and take the ratio of the two results. The natural logarithm of that ratio is the BDPL for this specific point t.
7. We now have 1 000 estimations of the adversary-specific BDPL of this adversary. Since the adversary-specific BDPL is a supremum, we take the maximum of these estimates as the empirical adversary-specific BDPL of this adversary.

8. Finally, we have empirical estimates for the adversary-specific BDPL of 1 000 different adversaries. We sort these estimates from minimum to maximum and pick the 99th percentile as our estimate for the total BDPL for this n and ρ. We use the 99th percentile instead of the maximum to make the result more robust: During testing, the estimate of an adversary-specific BDPL would rarely be exceptionally high. These rare results were not repeatable: Estimating the adversary-specific BDPL for the exact same adversary with the same data distribution and the same known values would yield a much lower adversary-specific BDPL. In contrast, the adversary-specific BDPL estimates around the 99th percentile were repeatable.

This estimation represents a lower bound for the maximum possible BDPL. The actual BDPL is a supremum over an infinite number of possibilities, from which we can only sample a finite subset. Additionally, to ensure robust results we did not sample the tails of the probability densities and chose the 99th percentile as the estimate for the maximum.

3.1.3 Utility Experiment

While we have previously highlighted the advantages of theory compared to empirical results, it is nonetheless important to measure the effect of our findings on the utility of BDP algorithms on real-world data. This allows us to verify that our results improve utility of BDP algorithms in actual applications.

For our utility experiment, we choose data sets that correspond to the correlation models analysed in our proofs (i.e., genomic data, data with a multivariate Gaussian correlation model, or data from a Markov chain). Then, we use our proofs on the relationship between DP and BDP to use DP algorithms for counting queries or sum queries as BDP algorithms. Here, we measure utility by calculating the *mean squared error* of the output from the BDP algorithm to the output from the deterministic counting or sum query.

4 Arbitrary Correlation Model

We begin by presenting our results when data has an arbitrary or unknown correlation model. The British statistician George E. P. Box wrote in 1978 “all models are wrong” [6, p. 202], describing the fact that any statistical model can only be an approximation of the real world. Even conceding this limitation, it may not be clear whether a given statistical model is an *appropriate* approximation for the distribution and correlation of a data tuple [45, p. 214]. In these cases, it would be beneficial to still use BDP to hide the influence of possible correlation, however without any assumptions or with only very limited assumptions about the correlation. Unfortunately, this is a futile attempt: We present an impossibility result that BDP algorithms cannot guarantee privacy against arbitrary correlation models, and provide utility at the same time.

First, Theorem 4.2 proves the *general bound*: An ε-DP algorithm is $m\varepsilon$-BDP where m is the number of correlated records in the data tuple. For arbitrary correlation, we have to assume that the entire data tuple may be correlated, i.e., $m = n$. Thus, a satisfactory privacy-utility trade-off via a DP algorithm under arbitrary correlation is not viable for sufficiently large data tuples. Here, we also explore the idea of limiting the Pearson correlation coefficient ρ of the data distribution. We hypothesized this could consistently reduce the BDPL of an ε-DP algorithm below the general bound $m\varepsilon$. However, Theorem 4.3 shows a counterexample where the general bound is tight, even though ρ is bounded.

Second, we prove in Corollary 4.8 that *any* ε-BDP algorithm – not just an adapted DP algorithm – must have bad utility if it protects data with arbitrary correlations. Finally, in Section 4.3 we come to the conclusion that more assumptions have to be made about the correlation – such as a model of the distribution of the data – to give satisfactory utility guarantees of BDP algorithms. After all, the full quote by Box reads “all models are wrong, *but some are useful*” [6, p. 202] (emphasis mine). We will explore such models in the subsequent sections on Genomic data, data with a multivariate Gaussian correlation model, and data following Markov chains.

4.1 Relationship between DP and BDP

First, we introduce and prove the general bound of the BDPL of an ε-DP algorithm. We will refer back to this result many times in the thesis, to compare improved bounds in specific situations. The general bound states that if we have an ε-DP algorithm $\mathcal{A}$ that takes data drawn from a distribution with at most m correlated random variables, then that algorithm $\mathcal{A}$ is $m\varepsilon$-BDP. Multiplying the DP privacy leakage by the number of correlated records to determine the maximum privacy leakage against attacks exploiting correlations has been a common practice [10, 32]. However, to the best of our knowledge, this method has not previously been proven to fulfill the privacy notion BDP. We address this knowledge gap. Our proof is more involved than previous results on this subject because Chen et al. [10] do not formally prove any privacy guarantees, and Liu et al. [32] only do so when considering linear correlation. In contrast, our proof holds for any correlation model.

We formally define what we mean by m correlated random variables. We split the random variables $X_1, \ldots, X_n$ into mutually independent subsets with maximum size m:

Definition 4.1. We say the random vector $\mathbf{X} = (X_1, \dots, X_n)^\top$ has at most $m \leq n$ *correlated random variables* if all of the following conditions are fulfilled.

- There exist disjoint sets of indices $C_1, \dots, C_r$ that make up $[n] = \bigcup_{l=1}^{r} C_l$.
- Each set C_l has maximum cardinality $m \geq |C_l|$ for any $l \in [r]$.
- For any $l \in [r]$, the random variables $\{X_j \mid j \in C_l\}$ are independent of the remaining random variables $\{X_j \mid j \in [n] \setminus C_l\}$.

Now, we prove the bound in two steps: First, we show in Lemma 4.1 that the ratio considered in the definition of BDPL can always be upper bounded by including more random variables in the condition. This will be useful to prove Theorem 4.2, where we limit the change of a single random variable X_i by changing all of its correlated random variables $\mathbf{X}_C$.

Lemma 4.1. Let $Y \in \mathcal{Y}$ be a random variable and let $\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathcal{X}^n$ be a random vector. Let $i \in [n]$ be an index, set $K \subseteq [n] \setminus \{i\}$ comprise the known indices, and set $\tilde{C} \subseteq [n] \setminus K$ with index $i \in \tilde{C}$ comprise the correlated indices. Then, we have

$$\sup_{S, \mathbf{x}_K, x_i, x_i'} \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} \leq \sup_{S, \mathbf{x}_K, \mathbf{x}_{\tilde{C}}, \mathbf{x}'_{\tilde{C}}} \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]}. \tag{21}$$

Proof. We prove the lemma by showing that for any values of $x_i, x_i' \in \mathcal{X}$ and $\mathbf{x}_K \in \mathcal{X}^{|K|}$, we can find $\mathbf{x}_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ so that the numerator of eq. (21) is greater and we can find $\mathbf{x}'_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ so that the denominator is smaller. We show this via proving a contradiction.

Assume that for all $\mathbf{x}_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$, we have

$$\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] > \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]. \tag{22}$$

Now, we bring this to a contradiction, thereby proving the opposite of eq. (22). Let index set $\tilde{C}_{-i} := \tilde{C} \setminus \{i\}$ include all indices of $\tilde{C}$ except for i. Then, we have

$$\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]$$

$$= \int_{\mathcal{X}^{|\tilde{C}_{-i}|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i, \mathbf{X}_{\tilde{C}_{-i}} = \mathbf{x}_{\tilde{C}_{-i}}] \, p_{\mathbf{X}_{\tilde{C}_{-i}}}(\mathbf{x}_{\tilde{C}_{-i}} | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i) \, \mathrm{d}\mathbf{x}_{\tilde{C}_{-i}} \tag{23}$$

$$= \int_{\mathcal{X}^{|\tilde{C}_{-i}|}} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = (x_i, \mathbf{x}_{\tilde{C}_{-i}})] \, p_{\mathbf{X}_{\tilde{C}_{-i}}}(\mathbf{x}_{\tilde{C}_{-i}} \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i) \, \mathrm{d}\mathbf{x}_{\tilde{C}_{-i}} \tag{24}$$

$$< \int_{\mathcal{X}^{|\tilde{C}_{-i}|}} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] \, p_{\mathbf{X}_{\tilde{C}_{-i}}}(\mathbf{x}_{\tilde{C}_{-i}} \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i) \, \mathrm{d}\mathbf{x}_{\tilde{C}_{-i}} \tag{25}$$

$$= \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] \int_{\mathcal{X}^{|\tilde{C}_{-i}|}} p_{\mathbf{X}_{\tilde{C}_{-i}}}(\mathbf{x}_{\tilde{C}_{-i}} \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i) \, \mathrm{d}\mathbf{x}_{\tilde{C}_{-i}} \tag{26}$$

$$= \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]. \tag{27}$$

First, we use the law of total probability to introduce all remaining random variables with indices in set $\tilde{C}$ in eq. (23). Random variable X_i is already included in the condition, so only indices $\tilde{C}_{-i}$ need to be added. In eq. (24), we combine the two conditions $X_i = x_i$ and $\mathbf{X}_{\tilde{C}_{-i}} = \mathbf{x}_{\tilde{C}_{-i}}$ into one condition $\mathbf{X}_{\tilde{C}} = (x_i, \mathbf{x}_{\tilde{C}_{-i}})$. Notice that this is the same condition, just stated differently. Then, we use eq. (22) – which applies to all $\mathbf{x}_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ – in eq. (25). Now, the first probability can be pulled out of the integral in eq. (26) as it no longer depends on the value $\mathbf{x}_{\tilde{C}_{-i}}$. The final eq. (27) follows because a probability density integrated over its entire domain is always 1. Overall, we have shown that the initial probability from the left-hand side of eq. (23) is strictly smaller than itself – a contradiction. Thus, the opposite of our assumption in eq. (22) must be true and there must exist a vector $\mathbf{x}_{\tilde{C}}$ so that

$$\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] \leq \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]. \tag{28}$$

Analogously, we can show that there exists a vector $\mathbf{x}'_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ with the opposite property

$$\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x'_i] \geq \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]. \tag{29}$$

Thus, with eq. (28) and eq. (29) we have

$$\sup_{S,\mathbf{x}_K,x_i,x'_i} \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x'_i]} \leq \sup_{S,\mathbf{x}_K,\mathbf{x}_{\tilde{C}},\mathbf{x}'_{\tilde{C}}} \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]}. \tag{30}$$

For any values of $\mathbf{x}_K$ and x_i, x'_i included in the left supremum, we can always find values $\mathbf{x}_{\tilde{C}}$ and $\mathbf{x}'_{\tilde{C}}$ so that the ratio becomes greater or equal, and these values $\mathbf{x}_{\tilde{C}}, \mathbf{x}'_{\tilde{C}}$ are included in the supremum on the right-hand side. □

Using Lemma 4.1, we prove the general bound in Theorem 4.2.

Theorem 4.2 (The General Bound)**.** Let $n \in \mathbb{N}$ be the size of the data tuple. Let $\mathbf{X} = (X_1, \ldots, X_n)^\top$ be a random vector with at most $m \leq n$ correlated random variables that follows a distribution π. Then, any ε-DP algorithm $\mathcal{A}$ with input data drawn from distribution π is $m\varepsilon$-BDP.

In the following proof of the general bound, we will limit the adversary-specific BDPL of any adversary by $m\varepsilon$, thereby showing that algorithm $\mathcal{A}$ is $m\varepsilon$-BDP. For an arbitrary adversary, the idea is that we can always increase the BDPL by not just changing the target X_i, but by also changing all of its correlated random variables in C_l. To show this, we use Lemma 4.1. Then, we can prove that the BDPL must be bounded by $m\varepsilon$ because of the independence of $\mathbf{X}_{C_l}$ from the rest of the random variables, and because changing m records $\mathbf{X}_{C_l}$ results in a DP privacy loss of $m\varepsilon$.

Proof. Consider any adversary (K, i) with $i \in [n]$ and $K \subseteq [n] \setminus \{i\}$. We begin by applying Lemma 4.1 and show that an upper bound for the adversary-specific BDPL can be found by conditioning on all of the random variables that are correlated with target X_i. First, we must identify the random variables correlated with X_i. There is an $l \in [r]$ so that we have target index $i \in C_l$, because $C_1, \dots, C_r$ partition the entire set $[n]$. Thus, C_l contains the index i and all indices of random variables potentially correlated with X_i. Let set $\tilde{C} := C_l \setminus K$ be the indices of random variables correlated with X_i that are not already included in K, and let random variable Y denote the output of algorithm $\mathcal{A}$. Then, the adversary-specific BDPL can be upper bounded directly with Lemma 4.1 as follows:

$$\begin{aligned} BDPL_{(K,i)} &= \sup_{S, \mathbf{x}_K, x_i, x_i'} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} && (31) \\ &\leq \sup_{S, \mathbf{x}_K, \mathbf{x}_{\tilde{C}}, \mathbf{x}'_{\tilde{C}}} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]} && (32) \end{aligned}$$

Now, we will show that eq. (32) is further limited by $m\varepsilon$. We achieve this by limiting the multiplicative difference between the numerator and the denominator. Let the set $U = [n] \setminus (K \cup \tilde{C})$ include all remaining indices. Let the known values $\mathbf{x}_K \in \mathcal{X}^{|K|}$ be arbitrary, and let the correlated values $\mathbf{x}_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ and $\mathbf{x}'_{\tilde{C}} \in \mathcal{X}^{|\tilde{C}|}$ be arbitrary. We have

$$\begin{aligned} &\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}] \\ &= \int_{\mathcal{X}^{|U|}} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}, \mathbf{X}_U = \mathbf{x}_U]\, p_{\mathbf{X}_U}(\mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}})\, \mathrm{d}\mathbf{x}_U && (33) \\ &\leq \int_{\mathcal{X}^{|U|}} e^{m\varepsilon} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}, \mathbf{X}_U = \mathbf{x}_U]\, p_{\mathbf{X}_U}(\mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}})\, \mathrm{d}\mathbf{x}_U && (34) \\ &= e^{m\varepsilon} \int_{\mathcal{X}^{|U|}} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}, \mathbf{X}_U = \mathbf{x}_U]\, p_{\mathbf{X}_U}(\mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K)\, \mathrm{d}\mathbf{x}_U && (35) \\ &= e^{m\varepsilon} \int_{\mathcal{X}^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}, \mathbf{X}_U = \mathbf{x}_U]\, p_{\mathbf{X}_U}(\mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}})\, \mathrm{d}\mathbf{x}_U && (36) \\ &= e^{m\varepsilon} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]. && (37) \end{aligned}$$

First, we use the law of total probability in eq. (33) to include the remaining random variables. In eq. (34), we use the group privacy property of DP [15, p. 20] to show that the probability distribution of the output changes at most by $e^{m\varepsilon}$ if at most m entries in the data tuple change. Here, it is important that $|\tilde{C}| \leq m$ has at most cardinality m. Then, we apply the independence of random variables $\{X_j \mid j \in \tilde{C}\}$ and the unknown random variables $\{X_j \mid j \in U\}$ in eq. (35) to remove a condition. These two sets are independent because they are subsets of two independent sets of random variables: $\{X_j \mid j \in C_l\}$ and $\{X_j \mid j \in [n] \setminus C_l\}$. Similarly, in eq. (36) we add the condition $\mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}$ because of the independence. Note that the value of the correlated random variables has now changed from $\mathbf{x}_{\tilde{C}}$ to $\mathbf{x}'_{\tilde{C}}$, but this has not had an effect on the probability density $p_{\mathbf{X}_U}$ because of the independence. Finally, we apply the law of total probability in reverse in eq. (37).

Now, all is ready to limit the total BDPL for algorithm $\mathcal{A}$:

$$
\begin{aligned}
BDPL &= \sup_{K,i} BDPL_{(K,i)} && (38)\\
&\leq \sup_{K,i} \sup_{S,\mathbf{x}_K,\mathbf{x}_{\tilde{C}},\mathbf{x}'_{\tilde{C}}} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}_{\tilde{C}}]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]} && (39)\\
&\leq \sup_{K,i} \sup_{S,\mathbf{x}_K,\mathbf{x}_{\tilde{C}},\mathbf{x}'_{\tilde{C}}} \ln \frac{e^{m\varepsilon} \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_{\tilde{C}} = \mathbf{x}'_{\tilde{C}}]} && (40)\\
&= \sup_{K,i} \sup_{S,\mathbf{x}_K,\mathbf{x}_{\tilde{C}},\mathbf{x}'_{\tilde{C}}} \ln e^{m\varepsilon} = m\varepsilon. && (41)
\end{aligned}
$$

First, we use eq. (32) in eq. (39). Then, we apply eq. (37) in the numerator of eq. (40). Subsequently, the ratio is shortened and the suprema are redundant as the bound no longer depends on those values.

We have bounded the BDPL of algorithm $\mathcal{A}$ to $m\varepsilon$. Thus, algorithm $\mathcal{A}$ is $m\varepsilon$-BDP. □

Now, we have the general upper bound on the BDPL of an ε-DP algorithm as $m\varepsilon$, where m is the number of correlated records. We will see in Corollary 4.4 that this causes an m-fold increase in the error α when using the Laplace mechanism and moving from ε-DP to ε-BDP. This proportional increase in the error can be detrimental to utility, especially if all $n = m$ records are correlated.

The general bound appears to imply that changing a single record has a completely deterministic effect on the remaining $m - 1$ correlated records. However, the *strength* of the correlation in a real world scenario may not be that pronounced. Therefore, we hypothesized that limiting the Pearson correlation coefficient ρ between records would result in a smaller bound for the BDPL, and therefore also better utility guarantees when moving from DP to BDP. However, Theorem 4.3 shows a counterexample where the general bound is tight, even if $\rho > 0$ is limited to be arbitrarily small. This proves that it does not suffice to limit ρ to create high-utility BDP algorithms from DP algorithms.

The following counterexample concerns data tuples of size two where the two records are correlated. Thus, the general upper bound of the BDPL of an ε-DP algorithm is 2ε. Theorem 4.3 applies to algorithms $\mathcal{A}$ that make "full use" of their available privacy budget ε, i.e., see Equation (42). An example of such an ε-DP algorithm on the domain $\{0,1\}^2$ would be a sum query with Laplacian noise of scale $\frac{1}{\varepsilon}$.

Theorem 4.3. Let $\mathbf{X} = (X_1, X_2)^\top$ be a random vector. Let $\mathcal{A} : \{0,1\}^2 \to \mathcal{Y}$ be an ε-DP algorithm with input data drawn from the distribution of $\mathbf{X}$. Let $S \subseteq \mathcal{Y}$ be a measurable set so that $\mathcal{A}$ reaches the maximum allowed privacy leakage, i.e.,

$$
\Pr[\mathcal{A}(0,0) \in S] = e^{\varepsilon} \cdot \Pr[\mathcal{A}(0,1) \in S] = e^{\varepsilon} \cdot \Pr[\mathcal{A}(1,0) \in S] = e^{2\varepsilon} \cdot \Pr[\mathcal{A}(1,1) \in S]. \quad (42)
$$

Then, for all $\hat{\rho} > 0$ and for all $\delta > 0$ there exists a probability distribution π with the following property: If random vector $\mathbf{X}$ follows distribution π, then the Pearson correlation coefficient $\rho_{X_1,X_2} \leq \hat{\rho}$ of random variables X_1 and X_2 is bounded, and the Bayesian differential privacy leakage is at least $BDPL \geq 2\varepsilon - \delta$.

	$X_1 = 0$	$X_1 = 1$	
$X_2 = 0$	β	$1 - \alpha - \beta$	$1 - \alpha$
$X_2 = 1$	0	α	α
	β	$1 - \beta$	1

Table 2: Joint probability distribution of random variables X_1 and X_2.

Proof. Let $r \in \mathbb{N}$ be arbitrary. Table 2 shows a valid probability distribution π for $\mathbf{X} = (X_1, X_2)^\top$ with $\alpha := \frac{r-1}{r}$ and $\beta := \frac{1}{r^2}$. In order to prove the theorem, it suffices to show that the correlation coefficient $\rho_{X_1,X_2} \to 0$ tends to zero and the Bayesian differential privacy leakage $BDPL \to 2\varepsilon$ tends to the general upper bound for $r \to \infty$. We begin with the Pearson correlation coefficient. The variances of X_1 and X_2 are denoted by $\sigma_{X_1}^2 = \beta(1-\beta)$ and $\sigma_{X_2}^2 = \alpha(1-\alpha)$ respectively.

$$\rho_{X_1,X_2} = \frac{\mathbb{E}[X_1X_2] - \mathbb{E}[X_1]\mathbb{E}[X_2]}{\sigma_{X_1}\sigma_{X_2}} = \frac{\alpha - \alpha(1-\beta)}{\sqrt{\beta(1-\beta)\alpha(1-\alpha)}} = \frac{\alpha\beta}{\sqrt{\beta(1-\beta)\alpha(1-\alpha)}} \tag{43}$$

$$= \sqrt{\frac{\alpha\beta}{(1-\beta)(1-\alpha)}} = \sqrt{\frac{\frac{r-1}{r^3}}{\frac{r^2-1}{r^3}}} = \sqrt{\frac{r-1}{r^2-1}} \xrightarrow{r\to\infty} 0 \tag{44}$$

Let random variable Y represent the output of $\mathcal{A}$. With Definition 2.11 of BDPL, the following holds as well:

$$BDPL \geq \ln \frac{\Pr[Y \in S \mid X_1 = 0]}{\Pr[Y \in S \mid X_1 = 1]} \tag{45}$$

$$= \ln \frac{\sum_{x_2 \in \{0,1\}} \Pr[Y \in S \mid X_1 = 0, X_2 = x_2] \cdot \Pr[X_2 = x_2 \mid X_1 = 0]}{\sum_{x_2 \in \{0,1\}} \Pr[Y \in S \mid X_1 = 1, X_2 = x_2] \cdot \Pr[X_2 = x_2 \mid X_1 = 1]} \tag{46}$$

$$= \ln \frac{\Pr[\mathcal{A}(0,0) \in S] \cdot \frac{\Pr_X[X_2=0,X_1=0]}{\Pr_X[X_1=0]} + \Pr[\mathcal{A}(0,1) \in S] \cdot \frac{\Pr_X[X_2=1,X_1=0]}{\Pr_X[X_1=0]}}{\Pr[\mathcal{A}(1,0) \in S] \cdot \frac{\Pr_X[X_2=0,X_1=1]}{\Pr_X[X_1=1]} + \Pr[\mathcal{A}(1,1) \in S] \cdot \frac{\Pr_X[X_2=1,X_1=1]}{\Pr_X[X_1=1]}} \tag{47}$$

$$= \ln \frac{e^{2\varepsilon} \cdot \Pr[\mathcal{A}(1,1) \in S] \cdot \frac{\beta}{\beta} + e^{\varepsilon} \cdot \Pr[\mathcal{A}(1,1) \in S] \cdot \frac{0}{\beta}}{e^{\varepsilon} \cdot \Pr[\mathcal{A}(1,1) \in S] \cdot \frac{1-\alpha-\beta}{1-\beta} + \Pr[\mathcal{A}(1,1) \in S] \cdot \frac{\alpha}{1-\beta}} \tag{48}$$

$$= \ln \frac{e^{2\varepsilon}(1-\beta)}{e^{\varepsilon} \cdot (1 - \alpha - \beta) + \alpha} \tag{49}$$

$$= 2\varepsilon + \ln \frac{1 - \frac{1}{r^2}}{e^{\varepsilon}(1 - \frac{r-1}{r} - \frac{1}{r^2}) + \frac{r-1}{r}} \tag{50}$$

$$= 2\varepsilon + \ln \frac{r^2 - 1}{e^{\varepsilon}(r-1) + r^2 - r} \xrightarrow{r\to\infty} 2\varepsilon + \ln 1 = 2\varepsilon \tag{51}$$

First, we use that the BDPL is defined as the supremum over all adversary-specific BDPLs. Thus, any adversary-specific BDPL is a lower bound for the total BDPL. In eq. (45) we choose the adversary that targets record $i = 1$ and does not know any other records. In eq. (46), we use the law of total probability. Then, we explicitly write down the terms of the sum in eq. (47). Conditional probabilities have been rewritten with the rule

$$\Pr(A \mid B) = \frac{\Pr(A \cap B)}{\Pr(B)}. \tag{52}$$

In eq. (48), we use the probabilities from Table 2 and use eq. (42). Subsequently, we simplify the ratio in eq. (49) by dividing numerator and denominator by the probability $\Pr[\mathcal{A}(1,1) \in S]$ and multiplying both with $1 - \beta$. Overall, we see that the BDPL is greater than a term that approaches 2ε for $r \to \infty$. We know that the BDPL for an ε-DP algorithm is limited by 2ε because of the general bound $m\varepsilon$: Here, two records are correlated and we have $m = 2$. Thus, the BDPL also approaches 2ε because there is a lower and an upper bound that both approach 2ε. □

Theorem 4.3 shows that for arbitrary correlation models, the general bound from Theorem 4.2 is tight, even if we limit Pearson correlation coefficient ρ. Therefore, we will explore additional assumptions on the correlation in later sections on genomic data, data with a multivariate Gaussian correlation model, and data following a Markov chain. This will enable us to improve upon the general bound, in contrast to only limiting ρ.

4.2 Accuracy

Before we explore specific correlation models, we investigate the concrete accuracy statements that can be made for BDP algorithms under arbitrary correlation. First, we do so for DP algorithms that are used as BDP algorithms, and subsequently we investigate the accuracy of *any* BDP algorithm.

The general bound on the BDPL of an ε-DP algorithm enables us to use DP algorithms as BDP algorithms. In cases where there exist utility guarantees of DP algorithms, we can use these results for the resulting BDP algorithms. Corollary 4.4 shows the increase in the error α when going from ε-DP to ε-BDP for algorithms using the Laplace mechanism.

Corollary 4.4. Let $n \in \mathbb{N}$ be the size of the data tuple. Let $\mathbf{X} = (X_1, \ldots, X_n)^\top$ be a random vector with at most $m \leq n$ correlated random variables that follows distribution π. Let function $f : \mathcal{X}^n \to \mathbb{R}$ be deterministic with finite sensitivity $\Delta f < \infty$. Then, if the Laplace mechanism $\mathcal{M}_{\varepsilon,f}$ fulfills (α, β)-accuracy, there exists an ε-BDP algorithm $\mathcal{B}$ whose input is drawn from π and that is $(m\alpha, \beta)$-accurate w.r.t. f.

Proof. From Corollary 2.2, we know that the (α, β)-accuracy of $\mathcal{M}_{\varepsilon,f}$ with respect to f is

$$\alpha = \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon} \tag{53}$$

for any $\beta \in (0, 1]$ because algorithm $\mathcal{M}_{\varepsilon,f}$ uses the Laplace mechanism.

The idea is to also use the Laplace mechanism for algorithm $\mathcal{B}$, but to use an adjusted DP privacy parameter ε' so that $\mathcal{B}$ is ε-BDP. We will see that this results in $(m\alpha, \beta)$-accuracy.

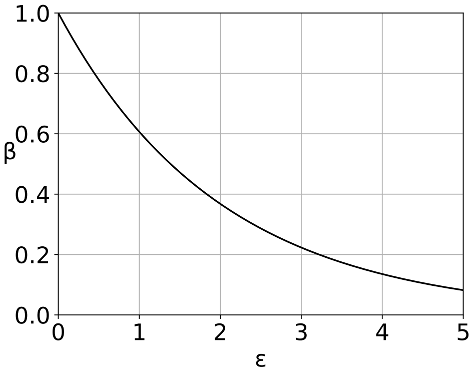


Figure 3: Probability of error $\alpha = n/2$ of ε-BDP counting query using Laplace mechanism under an arbitrary correlation model. The x-axis shows the BDPL ε, the y-axis the probability β of an error greater than $\alpha = n/2$.

With the general bound from Theorem 4.2, we know that the algorithm $\mathcal{B}$ is $m\varepsilon'$-BDP. Thus, we have

$$m\varepsilon' = \varepsilon \;\Leftrightarrow\; \varepsilon' = \frac{1}{m}\varepsilon. \tag{54}$$

Now, we can calculate the accuracy of algorithm $\mathcal{B}$ because it also uses the Laplace mechanism and we now know the used DP privacy leakage $\varepsilon' = \varepsilon/m$. Algorithm $\mathcal{B}$ is (α', β)-accurate with

$$\alpha' = \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon'} = \ln\left(\frac{1}{\beta}\right) \cdot \frac{m\Delta f}{\varepsilon} = m\alpha. \tag{55}$$

Thus, algorithm $\mathcal{B}$ is $(m\alpha, \beta)$-accurate. □

This result shows that the error α of the Laplace mechanism increases proportionally with the number of correlated records when moving from ε-DP to ε-BDP, and while making no assumption about the distribution of the records. In a situation where the number of correlated records cannot be limited, we have to assume the worst-case scenario that all $n = m$ records may be correlated. This can make a BDP algorithm have little utility: For example, consider a counting query g, i.e., a query that counts the number of records which fulfill some predicate. Function g must return a number between 0 and n and has sensitivity $\Delta g = 1$. Figure 3 shows the probability β of an error that exceeds $\alpha = n/2$ if we use the Laplace mechanism $\mathcal{M}_{\varepsilon/n,g}$ to create an ε-BDP algorithm. The probability of this large error – half the range of the function – is 60% for $\varepsilon = 1$. Even for a high privacy leakage $\varepsilon = 5$, it is nearly 10%.

We see that ε-BDP algorithms using the Laplace mechanism do not offer a good privacy-utility trade-off when all n records are correlated with an arbitrary model of correlation. This does not rule out that it may be possible to create "native" BDP algorithms with a better privacy-utility trade-off, without using existing DP algorithms or the Laplace mechanism. However, we will now show that *any* BDP algorithm that protects against arbitrary correlation cannot offer good utility.

We show that BDP is reduced to free-lunch privacy (see Theorem 4.5) when the correlation model can be arbitrarily chosen. For free-lunch privacy it is known that good utility is impossible [26]. However, this impossibility result has not yet been stated in terms of (α, β)-accuracy, which we use to quantify utility in this thesis. Therefore, we expand on the impossibility result by proving Theorem 4.6 and Corollary 4.7 which provide a lower bound for the error α of a free-lunch private algorithm. Then, we can derive a lower bound of α for BDP algorithms under arbitrary correlation in Corollary 4.8.

Theorem 4.5. Let $D, D' \in \mathcal{X}^n$ be any two tuples. Then, there exists a probability distribution π over the data tuples $\mathcal{X}^n$ so that both D and D' have non-zero probability and any ε-BDP algorithm $\mathcal{A} : \tilde{\mathcal{X}}^n \to \mathbb{R}$ is ε-free-lunch private, where $\tilde{\mathcal{X}}^n \subseteq \mathcal{X}^n$ denotes the set of data tuples with non-zero probability.

Proof. Let $\mathbf{X} = (X_1, \dots, X_n)^\top$ be a random vector. We will now construct a probability distribution π with $\mathbf{X} \sim \pi$ so that algorithm $\mathcal{A}$ is ε-free-lunch private. We differentiate between two cases: Either D, D' are the same tuple, or they are two differing tuples.

Case 1: The tuples $D = D'$ are identical.

In this case, it suffices to choose a probability distribution π where D has probability 1. Then, $\tilde{\mathcal{X}}^n = \{D\}$ will only include D and the algorithm is trivially ε-free-lunch private.

We choose the following probability distribution. For any $\tilde{D} \in \mathcal{X}^n$ we have

$$\Pr[\mathbf{X} = \tilde{D}] := \mathbb{1}\{\tilde{D} = D\}. \tag{56}$$

In order for $\mathcal{A}$ to be ε-free-lunch private, the following inequality must hold for any $D_1, D_2 \in \tilde{\mathcal{X}}^n$ and any measurable set $S \subseteq \mathbb{R}$:

$$\Pr[\mathcal{A}(D_1) \in S] \leq e^\varepsilon \Pr[\mathcal{A}(D_2) \in S] \tag{57}$$

This inequality is trivially fulfilled since we have $D_1 = D_2 = D$. Thus, algorithm $\mathcal{A}$ is ε-free-lunch private.

Case 2: The tuples $D \neq D'$ are not identical.

The idea of the proof in this case is to choose a probability distribution π so that the values of all other random variables are determined by just one random variable X_i. Then, D and D' become "neighbors" in the definition of BDP, i.e., they must be indistinguishable up to a factor of e^ε. This is equivalent to the definition of ε-free lunch privacy.

More precisely, we represent the data tuples D, D' by $D = (x_1, \dots, x_n)^\top$ and $D' = (x'_1, \dots, x'_n)^\top$. We choose any index $i \in [n]$ for which the value x_i differs from $x'_i \neq x_i$. There must be at least one such index because $D \neq D'$. This record x_i will be the record that is correlated with the entire remaining data tuple in our chosen probability distribution. To simplify notation, we name the remainders of the tuples $\mathbf{x}_{-i}$ and $\mathbf{x}'_{-i}$ respectively, with $\mathbf{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)^\top$ and $\mathbf{x}'_{-i} = (x'_1, \dots, x'_{i-1}, x'_{i+1}, \dots, x'_n)^\top$.

We construct a probability distribution π with strong correlation, so that all values of the data set are determined by the single random variable X_i:

$$\begin{aligned} \Pr[X_i = x_i] &:= \Pr[X_i = x'_i] := 1/2 && (58)\\ \Pr[\mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}_{-i} \mid X_i = x_i] &:= 1 && (59)\\ \Pr[\mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}'_{-i} \mid X_i = x'_i] &:= 1 && (60) \end{aligned}$$

For any other tuple $\tilde{D} \in \mathcal{X}^n \setminus \{D, D'\}$, we set the probability to zero:

$$\Pr[\mathbf{X} = \tilde{D}] := 0 \tag{61}$$

It follows that $\tilde{\mathcal{X}^n} = \{D, D'\}$ only includes D and D' because no other tuples have non-zero probability.

Now, let $\mathcal{A} : \tilde{\mathcal{X}^n} \to \mathbb{R}$ be any ε-BDP algorithm. This means that the adversary-specific BDPL for any (K, j) is bounded by ε, for any $j \in [n]$ and $K \subseteq [n] \setminus \{j\}$. In particular, the BDPL of the adversary that targets index $j = i$ with no known records $K = \emptyset$ is bounded by ε. We rewrite this fact to fulfill the definition of free-lunch privacy. Let Y be the random variable that represents the output of algorithm $\mathcal{A}$. For any measurable set $S \subseteq \mathbb{R}$ we have

$$\begin{aligned} \varepsilon &\geq \sup_{\tilde{S}, \tilde{x}_i, \tilde{x}'_i} \ln \frac{\Pr[Y \in \tilde{S} \mid X_i = \tilde{x}_i]}{\Pr[Y \in \tilde{S} \mid X_i = \tilde{x}'_i]} && (62)\\ &\geq \ln \frac{\Pr[Y \in S \mid X_i = x_i]}{\Pr[Y \in S \mid X_i = x'_i]} && (63)\\ &= \ln \frac{\sum_{\hat{\mathbf{x}} \in \mathcal{X}^{n-1}} \Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \hat{\mathbf{x}}, X_i = x_i] \Pr[\mathbf{X}_{[n]\setminus\{i\}} = \hat{\mathbf{x}} \mid X_i = x_i]}{\sum_{\hat{\mathbf{x}} \in \mathcal{X}^{n-1}} \Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \hat{\mathbf{x}}, X_i = x_i] \Pr[\mathbf{X}_{[n]\setminus\{i\}} = \hat{\mathbf{x}} \mid X_i = x'_i]} && (64)\\ &= \ln \frac{\Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}_{-i}, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_{[n]\setminus\{i\}} = \mathbf{x}'_{-i}, X_i = x'_i]} = \ln \frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]}. && (65) \end{aligned}$$

Equation (62) is the adversary-specific BDPL (see Definition 2.11) of an adversary that targets index i and knows none of the records (i.e., $K = \emptyset$). We have a lower bound of this adversary-specific BDPL in eq. (63) by choosing set S and target values x_i and x'_i. In eq. (64) we use the law of total probability both in the numerator and the denominator. Only a single term of the sum remains in eq. (65): It has probability 1 according to the chosen distribution in eqs. (59) and (60). These terms correspond exactly to the situation in which the input of algorithm $\mathcal{A}$ is D and the situation in which it is D'.

The reverse bound (i.e., D and D' switch places)

$$\varepsilon \geq \ln \frac{\Pr[\mathcal{A}(D') \in S]}{\Pr[\mathcal{A}(D) \in S]} \tag{66}$$

can be proven analogously. Thus, for all choices of two tuples from the domain $\tilde{\mathcal{X}^n} = \{D, D'\}$ we have shown that the inequality required by ε-free lunch privacy is fulfilled. It follows that $\mathcal{A}$ is ε-free-lunch private. □

Theorem 4.6 will present a lower bound on the error α of any randomized algorithm $\mathcal{A}$ w.r.t. any deterministic function f, if $\mathcal{A}$ limits the difference between the output distributions of two different input tuples D, D'. The larger the difference between $f(D)$ and $f(D')$, the worse the (α, β)-accuracy of $\mathcal{A}$ w.r.t. f must be. Corollary 4.7 will subsequently show that this means an ε-free lunch private algorithm must have a large error α.

Theorem 4.6. Let $\mathcal{A} : \mathcal{Z} \to \mathbb{R}$ be a randomized algorithm with domain $\mathcal{Z}$. Let $\varepsilon > 0$ be real. Let $0 \leq \beta < \frac{1}{e^\varepsilon+1}$ be real and let $f : \mathcal{Z} \to \mathbb{R}$ be a deterministic function. Let $\mathcal{A}$ fulfill the following inequality for two $D, D' \in \mathcal{Z}$ and all measurable sets $S \subseteq \mathbb{R}$:

$$\frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]} \leq e^\varepsilon. \tag{67}$$

If algorithm $\mathcal{A}$ is (α, β)-accurate w.r.t. f for an error $\alpha \geq 0$, then we must have

$$\alpha > \frac{1}{2}|f(D) - f(D')|. \tag{68}$$

The basic idea of the following proof is that algorithm $\mathcal{A}$ is forced by eq. (67) to provide statistically similarly likely results for both input D and input D'. Therefore, it is possible to show that a low error α would contradict eq. (67).

Proof. We assume that $\mathcal{A}$ fulfills a "better" (α, β)-accuracy and show that this ends in a contradiction. Let $\alpha \geq 0$ and $\beta \in [0, 1]$ be arbitrary with $\alpha \leq \frac{1}{2}|f(D) - f(D')|$ and $\beta < \frac{1}{e^\varepsilon+1}$. We assume that $\mathcal{A}$ is (α, β)-accurate with respect to f. The (α, β)-accuracy assumption means that for all data tuples $\tilde{D} \in \mathcal{Z}$ we have

$$\Pr[|f(\tilde{D}) - \mathcal{A}(\tilde{D})| \geq \alpha] \leq \beta. \tag{69}$$

We show that eq. (69) does not hold for D', which is a contradiction.

$$\begin{aligned}
\Pr[|f(D') - \mathcal{A}(D')| \geq \alpha] &= \Pr[\mathcal{A}(D') \in \mathbb{R} \setminus (f(D') - \alpha, f(D') + \alpha)] && (70)\\
&\geq \Pr[\mathcal{A}(D') \in (f(D) - \alpha, f(D) + \alpha)] && (71)\\
&\geq e^{-\varepsilon} \cdot \Pr[\mathcal{A}(D) \in (f(D) - \alpha, f(D) + \alpha)] && (72)\\
&= e^{-\varepsilon} \cdot (1 - \Pr[\mathcal{A}(D) \in \mathbb{R} \setminus (f(D) - \alpha, f(D) + \alpha)]) && (73)\\
&= e^{-\varepsilon} \cdot (1 - \Pr[|f(D) - \mathcal{A}(D)| \geq \alpha]) && (74)\\
&\geq e^{-\varepsilon} \cdot (1 - \beta) && (75)\\
&> e^{-\varepsilon} \cdot (\frac{e^\varepsilon + 1}{e^\varepsilon + 1} - \frac{1}{e^\varepsilon + 1}) && (76)\\
&= \frac{1}{e^\varepsilon + 1} && (77)\\
&> \beta && (78)
\end{aligned}$$

Observe that the first equality in eq. (70) holds: The distance between $\mathcal{A}(D')$ and $f(D')$ is greater or equal to α if and only if $\mathcal{A}(D')$ is any real number outside of an α-range around $f(D')$. Equation (71) follows because the probability of a subset is always smaller or equal

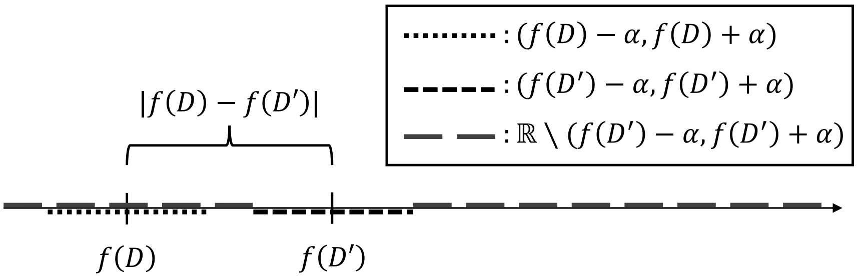

$: (f(D)-\alpha, f(D)+\alpha)$
$: (f(D')-\alpha, f(D')+\alpha)$
$: \mathbb{R} \setminus (f(D')-\alpha, f(D')+\alpha)$
$|f(D)-f(D')|$
$f(D)$
$f(D')$

Figure 4: Visualization of sets in eqs. (70) and (71).

to the probability of the greater set. Here, this subset relationship $(f(D)-\alpha, f(D)+\alpha) \subseteq \mathbb{R} \setminus (f(D')-\alpha, f(D')+\alpha)$ holds because α is smaller or equal to half the distance between $f(D)$ and $f(D')$ (see Figure 4). This probability can once again be limited in eq. (72) by using the fact that $\mathcal{A}$ is forced to provide statistically similar result for D and D' (eq. (67)). The probability is replaced by its complementary probability in eq. (73). The transformation to eq. (74) uses the same technique as eq. (70) (see above). Finally, we use that $\mathcal{A}$ is (α, β)-accurate in eq. (75) and subsequently replace β by its upper bound in eq. (76). Overall, it follows that $\Pr[|f(D') - \mathcal{A}(D')| \geq \alpha]$ is strictly greater than β. This is in contradiction to (α, β)-accuracy.

We have shown that this (α, β)-accuracy is impossible under the given assumptions, as it leads to a contradiction. The additional assumption we made was $\alpha \leq \frac{1}{2}|f(D) - f(D')|$, which therefore must have been false. Thus, we have

$$\alpha > \frac{1}{2}|f(D) - f(D')|. \tag{79}$$

□

Now, we can use Theorem 4.6 to show a lower bound for the error α of any ε-free lunch private algorithm w.r.t. any function f. This lower bound will be comparatively high because we can choose D and D' to maximize the distance $|f(D) - f(D')|$.

Corollary 4.7. Let $\mathcal{A} : \mathcal{Z} \to \mathbb{R}$ be an ε-free-lunch private algorithm with domain $\mathcal{Z}$. Let $0 \leq \beta < \frac{1}{e^\varepsilon + 1}$ be a real number and let $f : \mathcal{Z} \to \mathbb{R}$ be a deterministic function with finite sensitivity $\Delta f < \infty$. If algorithm $\mathcal{A}$ is (α, β)-accurate w.r.t. f for an error $\alpha \geq 0$, then

$$\alpha > \frac{1}{2} \max_{\tilde{D}, \tilde{D}'} |f(\tilde{D}) - f(\tilde{D}')|. \tag{80}$$

We will prove Corollary 4.7 by choosing two tuples D, D' that cause maximally different outputs in f. Subsequently, we can apply Theorem 4.6 because $\mathcal{A}$ is forced by free-lunch privacy to provide statistically similar results for *any* data sets.

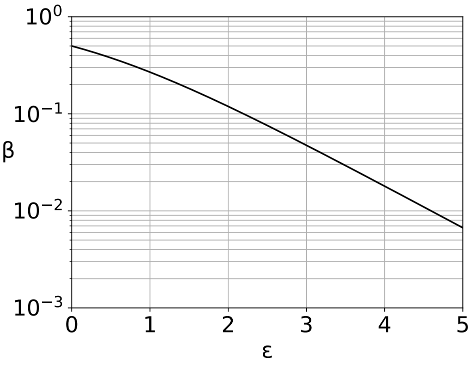


Figure 5: Minimum β for the (α, β)-accuracy of an ε-free-lunch private algorithm.

Proof. Let $D, D' \in \mathcal{Z}$ be arbitrary with $|f(D) - f(D')| = \max_{\tilde{D},\tilde{D}'} |f(\tilde{D}) - f(\tilde{D}')|$. Algorithm $\mathcal{A}$ is ε-free lunch private, so for any measurable $S \subseteq \mathbb{R}$ we have

$$\frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]} \leq e^{\varepsilon}. \tag{81}$$

The corollary now follows immediately with Theorem 4.6. □

We have found a lower bound for the error α of any ε-free lunch private algorithm $\mathcal{A}$. However, it is not directly obvious that this lower bound results in *poor* utility. Here, we present the case that it does. We must consider what the result from Corollary 4.7 means: The probability of an error greater than $\alpha = \frac{1}{2} \max_{\tilde{D},\tilde{D}'} |f(\tilde{D}) - f(\tilde{D}')|$ is at least $\beta = \frac{1}{e^{\varepsilon}+1}$. If algorithm $\mathcal{A}$ is supposed to approximate f, then an error as great as α is very undesirable: Even the largest differences in the output of f can no longer be differentiated when using the private algorithm $\mathcal{A}$. However, such an error may be acceptable if its probability β is very small. Figure 5 shows that for privacy leakage $\varepsilon \in (0, 2)$, the lower bound for β exceeds 10%, and even for high privacy leakage $\varepsilon \in (2, 4)$, β is above 1%. Therefore, we interpret this result as poor utility of any ε-free lunch private algorithm that gives reasonable privacy guarantees.

Now, we use Corollary 4.7 and the equivalence of BDP algorithms and free-lunch private algorithms from Theorem 4.5 to show that the accuracy of BDP algorithm must also be poor if the data distribution can be arbitrary.

Corollary 4.8. Let $f : \mathcal{X}^n \to \mathbb{R}$ be a deterministic function with finite sensitivity $\Delta f < \infty$. Let $\varepsilon > 0$ be real and let $0 \leq \beta < \frac{1}{e^{\varepsilon}+1}$ be real. Then, there exists a probability distribution π over the data tuples $\mathcal{X}^n$ so that for any ε-BDP algorithm $\mathcal{A} : \tilde{\mathcal{X}}^n \to \mathbb{R}$ whose input tuples are drawn from π, the following holds: If algorithm $\mathcal{A}$ is (α, β)-accurate w.r.t. f for an $\alpha \geq 0$, then we have

$$\alpha > \frac{1}{2} \max_{\tilde{D},\tilde{D}'} |f(\tilde{D}) - f(\tilde{D}')|. \tag{82}$$

Here, $\tilde{\mathcal{X}}^n \subset \mathcal{X}^n$ denotes the set of data tuples with non-zero probability.

Proof. Choose tuples $D, D' \in \mathcal{X}^n$ so that they maximize the difference in outputs of f, i.e., $|f(D) - f(D')| = \max_{\tilde{D},\tilde{D}'} |f(\tilde{D}) - f(\tilde{D}')|$. Then, it follows from Theorem 4.5 that there exists a probability distribution over $\mathcal{X}^n$ so that tuples $D, D' \in \tilde{\mathcal{X}}^n$ are among the tuples with non-zero probability and so that any ε-BDP algorithm with domain $\tilde{\mathcal{X}}^n$ is ε-free-lunch private – including $\mathcal{A}$. Thus, Corollary 4.7 applies to $\mathcal{A}$. It follows the stated lower bound of (α, β)-accuracy w.r.t. any deterministic function, in particular f. □

4.3 Discussion

This section has shown that the BDPL of an ε-DP algorithm can be bounded by $m\varepsilon$, where m is the number of correlated records. Under arbitrary correlation models, this bound is tight, even if the Pearson correlation coefficient is limited. Additionally, we have proven that the utility of *any* ε-BDP algorithm must be poor if correlation can be completely arbitrary, with no limit to the number of correlated records m. These negative results indicate that it is not be possible to give good utility guarantees for BDP algorithms under arbitrary correlation, unless only a sufficiently small number $m < n$ of records are correlated. These results motivate the following sections on utility where specific models of the data distribution are assumed (i.e., genomic data, data with a multivariate Gaussian correlation model, and data following a Markov chain). Indeed, in Section 6 we find that limiting Pearson correlation coefficient ρ *can* result in decent utility guarantees if the data has a multivariate Gaussian correlation model – even if all n records are correlated.

5 Genomic Correlation Model

Genomic data is a prime candidate for applications of BDP for three reasons: (1) It is often considered highly sensitive and thereby private information [1]. (2) It is collected and analyzed of millions of people [37]. (3) The correlation of genetic material between family members is very well understood, and can – for some parts of the human genome – be broken down to the fundamental laws of inheritance by Gregor Mendel [18]. In this section explore the impact of this correlation type on the privacy leakage, proving Corollary 5.2 which shows that the general bound of the BDPL of an ε-DP counting query on genomic data is nearly tight. Besides, in Section 5.2 we investigate the consequences of this result for the utility of a counting query. We investigate both the utility of the Laplace mechanism, and the utility of *any* BDP algorithm in this situation. In Section 5.3, we come to the conclusion that the maximum size of the largest family in the data must be sufficiently small so that acceptable utility is achieved.

The following stochastic model follows Mendel's laws of inheritance [18] and can be derived from the model by Wang and Xu [52], who use independent Bernoulli variables to represent inheritance. While Wang and Xu only looked at the probabilistic relationship between two parents and their child, we expand the model to include any familial relationship. An *allele* is a singular piece of genetic information at a specific position in the genetic code; humans have two alleles at each position because they inherit one from each parent [47]. Figure 6 visualizes this phenomenon.

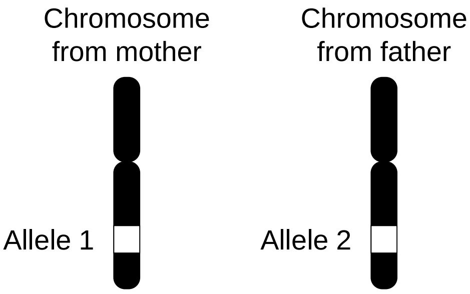


Figure 6: An allele pair at a specific position in the genetic code.

Definition 5.1 (Mendelian Inheritance)**.** Let $\mathbf{X} = (X_1, \ldots, X_n)^\top \in \mathcal{X}^n$ be a random vector. Let set $E \subseteq \{(i,j) \mid i,j \in [n]\}$ be a set of edges. We say random vector $\mathbf{X}$ is *Mendelian distributed* with pedigree graph $G = ([n], E)$ if all of the following requirements hold. Let $i, j, k \in [n]$ be arbitrary.

- The data domain consists of n *allele pairs*, each one corresponding to a different person at the same genetic position. For each person, either both alleles are major (homozygous-major: BB), they are mixed (heterozygotes: Bb), or both are minor (homozygous-minor: bb). I.e., the domain is $\mathcal{X} = \{BB, Bb, bb\}$.
- We call X_j an ancestor of X_i if there is a path from j to i in the set of edges E. In this case, we also call X_i a descendent of X_j.

		X_k		
		BB	Bb	bb
X_j	BB	$(1, 0, 0)$	$(0.5, 0.5, 0)$	$(0, 1, 0)$
	Bb	$(0.5, 0.5, 0)$	$(0.25, 0.5, 0.25)$	$(0, 0.5, 0.5)$
	bb	$(0, 1, 0)$	$(0, 0.5, 0.5)$	$(0, 0, 1)$

Table 3: Probability distribution of X_i on $\{BB, Bb, bb\}$ given the alleles of their parents.

- We call X_j a parent of X_i if X_j is an ancestor of X_i, and the path between them in E is exactly one edge long. Every random variable X_i has at most two parents.
- If $j \neq k$ and X_j and X_k are parents of X_i, then the probability distribution $\Pr[X_i = x_i \mid X_j = x_j, X_k = x_k]$ is given by Table 3. Each cell shows the probability distribution of X_i on $\{BB, Bb, bb\}$ given the values of X_k and X_j.
- Random variable X_i is independent of X_j if they do not share a common ancestor.
- Random variable X_i is conditionally independent of any random variable that is not a descendent of X_i, given their parents X_j and X_k. I.e., let $A \subseteq [n]$ be the set of the indices of all descendants of X_i. Then, for any subset $B \subseteq [n] \setminus A$ of indices that are not descendants of X_i, and for all $x_i, x_j, x_k \in \mathcal{X}$ and all $\mathbf{x}_B \in \mathcal{X}^{|B|}$ we have

$$\Pr[X_i = x_i \mid X_j = x_j, X_k = x_k, \mathbf{X}_B = \mathbf{x}_B] = \Pr[X_i = x_i \mid X_j = x_j, X_k = x_k]. \quad (83)$$

Under this stochastic model, the members of a family (i.e., a connected component in the pedigree graph G) can all be correlated with each other. E.g., in a family with two parents and $m-2$ children, each child will – of course – be correlated with both parents, but also with every other child because their genetic code depends on the same parents. Thus, if we assume that the largest family in the data tuple has m members (i.e., the largest connected component in G has size m), then the general bound becomes $m\varepsilon$.

5.1 Relationship between DP and BDP

In this section, we will show that the general bound of $m\varepsilon$ is (nearly) tight for genomic data. More specifically, Corollary 5.2 will prove that we can construct a pedigree graph G so that the BDPL of a counting query is at least $(m-1)\varepsilon$. It is worth mentioning that this does not involve a particularly complicated or unrealistic pedigree graph. Indeed, such a BDPL occurs when we have a family with two parents and $m-2$ children because certain changes in the genetic code of the parents can completely alter the respective genetic code of the children under Mendelian inheritance.

First, we prove Theorem 5.1, which shows that two specific data tuples become "neighbors" under the definition of BDPL for certain pedigree graphs G. This result will be essential for Corollary 5.2, where we show that this neighborhood causes a high BDPL for a counting query with the Laplace mechanism. Additionally, we use this result in

Corollary 5.3 to show a general lower bound on the error α of any BDP counting query on genomic data.

Theorem 5.1. Let $n \in \mathbb{N}$ be the size of the data tuple, and let $3 \leq m \leq n$ be the number of correlated records. Let $D = (BB, bb, Bb^{m-2}, bb^{n-m})$ and $D' = (BB, BB, BB^{m-2}, bb^{n-m})$ be two data tuples, where $x^i := (x, \ldots, x) \in \mathcal{X}^i$ denotes a repeating tuple for any $i \in \mathbb{N}$ and any $x \in \mathcal{X}$. Let $S \subseteq \mathcal{Y}$ be a measurable set. Then, there exists a set of edges $E \subseteq \{(i,j) \mid i,j \in [n]\}$, so that the BDPL of any algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ whose input tuples are drawn from a Mendelian distribution with pedigree graph $G = ([n], E)$ with m correlated records, is lower bounded by

$$BDPL \geq \ln \frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]}. \tag{84}$$

Proof. First, we will construct the pedigree graph $G = ([n], E)$ and show that this results in D and D' being "neighboring" data sets from the perspective of BDP. The idea is to have a single family with two parents and $m-2$ children, whose records will be completely determined by the records of the parents. We call set $F = [m]$ the indices of the family. The indices of the parents are 1 and 2, while the indices of the children are $C = \{3, \ldots, m\}$. Thus, we choose the set of edges as

$$E = \{(1,i), (2,i) \mid i \in C\}. \tag{85}$$

We call the remaining indices $R = [n] \setminus F$. Let random vector $\mathbf{X} = (X_1, \ldots, X_n)^\top$ follow a Mendelian distribution with pedigree graph $G = ([n], E)$.

We are trying to find a lower bound for the BDPL where $D = (BB, bb, Bb^{m-2}, bb^{n-m})$ and $D' = (BB, BB, BB^{m-2}, bb^{n-m})$ are neighbors. Observe that in data tuple D, the allele pairs of the parents are BB and bb, respectively. Thus, all $m-2$ children must have allele pair Bb according to the probability distribution from Table 3. Indeed, these are the alleles of the children (indices 3 to m) in data tuple D. For D', it is similar: The allele pairs of the parents are both BB. Thus, all children must have allele pair BB.

We see that a change in the allele pair of a single parent – from bb to BB – can cause the allele pairs of all $m-2$ children to change. We use this fact to construct the lower bound for the BDPL. Consider the adversary (K, i) who knows indices $K = \{1\} \cup R$, i.e., they know the record of one parent and any remaining records not part of the family, and target the second parent $i = 2$. The records of the children in set C are unknown to the adversary. Then, we can construct the following lower bound for this adversary-specific BDPL. Let random variable Y represent the output of algorithm $\mathcal{A}$, and let $S \subseteq \mathcal{Y}$ be any measurable set.

$$BDPL_{(K,2)} = \sup_{\mathbf{x}_K, x_2, x_2', \tilde{S}} \ln \frac{\Pr[Y \in \tilde{S} \mid \mathbf{X}_K = \mathbf{x}_K, X_2 = x_2]}{\Pr[Y \in \tilde{S} \mid \mathbf{X}_K = \mathbf{x}_K, X_2 = x_2']} \tag{86}$$

$$\geq \ln \frac{\Pr[Y \in S \mid X_1 = BB, \mathbf{X}_R = bb^{n-m}, X_2 = bb]}{\Pr[Y \in S \mid X_1 = BB, \mathbf{X}_R = bb^{n-m}, X_2 = BB]} \tag{87}$$

The lower bound holds because the BDPL is a supremum and we simply choose the values for x_i, x_i' and $\mathbf{x}_K$ that correspond to data tuples D and D'. We now treat the numerator and the denominator separately to avoid cluttered notation. We begin with the denominator:

$$\begin{aligned}
&\Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}] \\
&= \sum_{\mathbf{x}_C \in \mathcal{X}^{m-2}} \Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = \mathbf{x}_C] \\
&\quad \cdot \Pr[\mathbf{X}_C = \mathbf{x}_C \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}] && (88) \\
&= \sum_{\mathbf{x}_C \in \mathcal{X}^{m-2}} \Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = \mathbf{x}_C] \\
&\quad \cdot \Pr[\mathbf{X}_C = \mathbf{x}_C \mid X_1 = BB, X_2 = BB] && (89) \\
&= \sum_{\mathbf{x}_C \in \mathcal{X}^{m-2}} \Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = \mathbf{x}_C] \\
&\quad \cdot \prod_{j \in C} \Pr[X_j = x_j \mid X_1 = BB, X_2 = BB] && (90) \\
&= \Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = BB^{m-2}] \\
&\quad \cdot \prod_{j \in C} \Pr[X_j = BB \mid X_1 = BB, X_2 = BB] && (91) \\
&= \Pr[Y \in S \mid X_1 = BB, X_2 = BB, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = BB^{m-2}] && (92) \\
&= \Pr[\mathcal{A}(D') \in S] && (93)
\end{aligned}$$

We use the law of total probability in eq. (88) to introduce the alleles of the children in C. It is possible to remove the condition on the remaining random variables $\mathbf{X}_K$ in eq. (89) because they are independent of $\mathbf{X}_C$ since there are no paths between any index in R and any index in C in the set of edges E. In eq. (90), we simply calculate the joint probability of the children as a large product because they are all conditionally independent given the alleles of their common parents. Here, we denote the alleles of the children as $\mathbf{x}_C = (x_3, \dots, x_m)^\top$. For the alleles of the parents $X_1 = BB$ and $X_2 = BB$, only $X_j = BB$ has probability 1, all other options have probability 0 (see Table 3). Thus, this is the only option that remains in eq. (91). Since all of the remaining conditional probabilities are 1, we arrive at eq. (92). Now, the entire input of $\mathcal{A}$ is defined in the condition on the probability of $Y \in S$. Thus, we can rewrite it to eq. (93).

An analogous equality can be shown for the numerator with $X_2 = bb$.

$$\begin{aligned}
&\Pr[Y \in S \mid X_1 = BB, X_2 = bb, \mathbf{X}_R = bb^{n-m}] \\
&= \Pr[Y \in S \mid X_1 = BB, X_2 = bb, \mathbf{X}_R = bb^{n-m}, \mathbf{X}_C = Bb^{m-2}] && (94) \\
&= \Pr[\mathcal{A}(D) \in S] && (95)
\end{aligned}$$

With these two results, we can calculate a lower bound for the BDPL in this situation and prove the theorem.

$$\begin{aligned}
BDPL &\geq BDPL_{(K,2)} && (96) \\
&\geq \ln \frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]} && (97)
\end{aligned}$$

□

Now, we use this result to show the minimum BDPL of the Laplace mechanism applied to a counting query. A counting query is a reasonable application for genomic data since the frequency of certain genetic code in the population is of interest [2].

Corollary 5.2. Let $n \in \mathbb{N}$ be the size of the data tuple, and let $3 \leq m \leq n$ be the number of correlated records. For any data tuple $\tilde{D} \in \mathcal{X}^n$, let query $g : \mathcal{X}^n \to \mathbb{N}$ be defined by

$$g(\tilde{D}) = \sum_{i \in [n]} \mathbb{1}\{\tilde{D}_i = BB\}. \tag{98}$$

Then, there exists a set of edges $E \subseteq \{(i,j) \mid i,j \in [n]\}$, so that the BDPL of the Laplace mechanism $\mathcal{M}_{\varepsilon,g}$ whose input tuples are drawn from a Mendelian distribution with pedigree graph $G = ([n], E)$ with m correlated records, is at least

$$BDPL \geq (m-1)\varepsilon. \tag{99}$$

Proof. Let algorithm $\mathcal{M}_{\varepsilon,g}$ be the Laplace mechanism applied to g with privacy leakage ε, and let random variable Y denote the output of $\mathcal{A}$. We use the previous result from Theorem 5.1 and explicitly calculate the lower bound of the BDPL for set $S = (-\infty, 0]$. This choice of set S makes the following calculations simpler because the probability $\Pr[Y \in S]$ simply becomes the cumulative distribution function $\Pr[Y \leq 0]$.

We first determine the output distribution of the algorithm $\mathcal{M}_{\varepsilon,g}$, and subsequently we can calculate the lower bound on the BDPL using this distribution. The Laplace mechanism uses the Laplace distribution. If a random variable Z is Laplace distributed with location $\mu \in \mathbb{R}$ and scale $b > 0$, then the cumulative distribution function at $z \in \mathbb{R}$ with $\mu \geq z$ is

$$\Pr[Z \leq z] = \frac{1}{2} \exp\left(\frac{z-\mu}{b}\right) \tag{100}$$

according to [27, p. 21]. The sensitivity of counting query g is $\Delta g = 1$. The Laplace mechanism adds Laplacian noise Z with location zero and scale $b = \frac{\Delta g}{\varepsilon} = \frac{1}{\varepsilon}$ to the output of g. Thus, for any $\tilde{D} \in \mathcal{X}^n$ we have

$$\begin{aligned}
\Pr[\mathcal{M}_{\varepsilon,g}(\tilde{D}) \in S] = \Pr[\mathcal{M}_{\varepsilon,g}(\tilde{D}) \leq 0] = \Pr[g(\tilde{D}) + Z \leq 0] &= \Pr[Z \leq -g(\tilde{D})] && (101)\\
&= \frac{1}{2} \exp\left(\frac{-g(\tilde{D}) - 0}{\frac{1}{\varepsilon}}\right) && (102)\\
&= \frac{1}{2} \exp(-\varepsilon g(\tilde{D})) && (103)
\end{aligned}$$

In eq. (102) we use the definition of the cumulative distribution function from eq. (100).

We can now use this equality in the lower bound of the BDPL. We apply Theorem 5.1, so for data tuples $D = (BB, bb, Bb^{m-2}, bb^{n-m})$ and $D' = (BB, BB, BB^{m-2}, bb^{n-m})$ we have

$$\begin{aligned} BDPL &\geq \ln \frac{\Pr[\mathcal{M}_{\varepsilon,g}(D) \in S]}{\Pr[\mathcal{M}_{\varepsilon,g}(D') \in S]} && (104)\\ &= \ln \frac{\frac{1}{2}\exp(-\varepsilon g(D))}{\frac{1}{2}\exp(-\varepsilon g(D'))} && (105)\\ &= \ln \frac{\exp(-\varepsilon)}{\exp(-\varepsilon m)} && (106)\\ &= \ln \exp((m-1)\varepsilon) && (107)\\ &= (m-1)\varepsilon. && (108) \end{aligned}$$

In eq. (105), we use the equality from eq. (103). Then, we can apply counting query g to D and D' respectively and find $g(D) = 1$ and $g(D') = m$. We use these two results in eq. (106). After two more straight-forward equivalences, we arrive at the lower bound for the BDPL which we set out to prove. □

5.2 Accuracy

When dealing with genomic data, we have seen that the BDPL of the Laplace mechanism applied to a counting query can (nearly) reach the general upper bound of the BDPL, for certain pedigree graphs. Now we investigate how this affects the utility of *any* BDP algorithm with respect to the deterministic counting query. More specifically, Corollary 5.3 proves a lower bound on the error α, using Theorem 4.6 from Section 4 on arbitrary correlation. Afterwards, we compare this result to the actual (α, β)-accuracy of a BDP algorithm using the Laplace mechanism in Figure 7.

Corollary 5.3. Let $n \in \mathbb{N}$ be the size of the data tuple, and let $3 \leq m \leq n$ be the number of correlated records. For any data tuple $\tilde{D} \in \mathcal{X}^n$, let query $g : \mathcal{X}^n \to \mathbb{N}$ be defined by

$$g(D) = \sum_{i \in [n]} \mathbb{1}\{D_i = BB\}. \tag{109}$$

Let $\varepsilon > 0$ be real and let $0 \leq \beta < \frac{1}{e^{\varepsilon}+1}$ be real. Then, there exists a set of edges $E \subseteq \{(i,j) \mid i,j \in [n]\}$, so that for any ε-BDP algorithm $\mathcal{A}$ whose input tuples are drawn from a Mendelian distribution with pedigree graph $G = ([n], E)$ with m correlated records, the following holds: If algorithm $\mathcal{A}$ is (α, β)-accurate with respect to g for an error $\alpha \geq 0$, we have

$$\alpha > \frac{m-1}{2}. \tag{110}$$

Proof. First, we use Theorem 5.1 to show that in this situation, there exists a set of edges E so that the difference in distribution between outputs for data tuples $D = (BB, bb, Bb^{m-2}, bb^{n-m})$ and $D' = (BB, BB, BB^{m-2}, bb^{n-m})$ of any ε-BDP algorithm $\mathcal{A}$ must be limited by ε. Then, we apply Theorem 4.6 from the previous section to show that this causes a minimum bound on the (α, β)-accuracy.

According to Theorem 5.1, for any ε-BDP algorithm $\mathcal{A}$ and measurable $S \subseteq \mathbb{R}$ we have

$$BDPL \geq \ln \frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]} \tag{111}$$

$$\Leftrightarrow \qquad \varepsilon \geq \ln \frac{\Pr[\mathcal{A}(D) \in S]}{\Pr[\mathcal{A}(D') \in S]} \tag{112}$$

The second inequality follows because of inequality $\varepsilon \geq BDPL$ according to Definition 2.12, the definition of ε-BDP.

Now we have fulfilled the prerequisites to apply Theorem 4.6 with data tuples D and D'. Recall that – in a nutshell – Theorem 4.6 states that the error α must exceed $\frac{1}{2}|f(D) - f(D')|$ for any deterministic function f and for certain probabilities β when the difference in output distribution for D and D' is limited by e^{ε}. Since Theorem 4.6 applies to any deterministic function $f : \mathcal{X}^n \to \mathbb{R}$, it must also apply to g. Additionally, for the distance between $g(D)$ and $g(D')$ we have

$$|g(D) - g(D')| = |1 - m| = m - 1. \tag{113}$$

So when Theorem 4.6 is applied to this situation, it states: If algorithm $\mathcal{A}$ is (α, β)-accurate with respect to g for a probability $\beta < \frac{1}{e^{\varepsilon}+1}$, then we have

$$\alpha > \frac{1}{2}|g(D) - g(D')| = \frac{m-1}{2}. \tag{114}$$

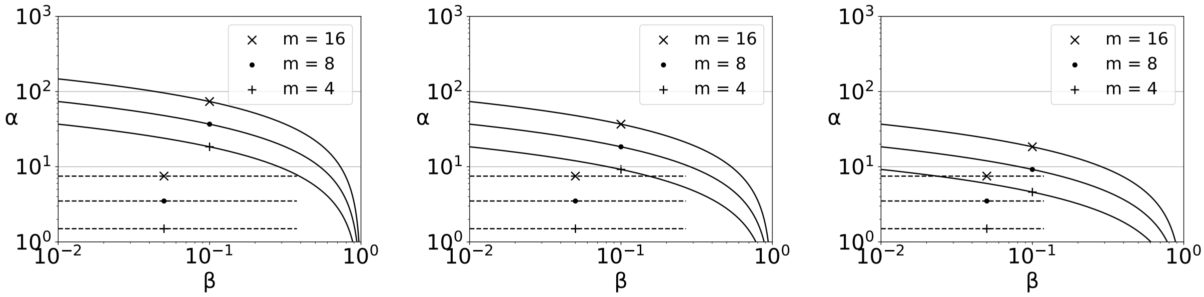
□

Now, we have a lower bound of the accuracy of any ε-BDP algorithm approximating counting query g. We can compare this to the accuracy of an actual ε-BDP algorithm approximating g – the Laplace mechanism. We have proven an accuracy statement of the Laplace mechanism using the general bound in Corollary 4.4 in the section on arbitrary correlation. Figure 7 compares the two results: The theoretical lower bound of the (α, β)-accuracy, and the actual (α, β)-accuracy of the Laplace mechanism. This is shown for different BDPLs ε and different numbers of correlated records m. The figure is interpreted in the following discussion.

5.3 Discussion

This section has shown that the general bound $m\varepsilon$ of the BDPL of an ε-DP algorithm is (nearly) tight for Genomic data when the pedigree graph is unrestricted except for limiting the size of the largest family to m. Whether an acceptable utility-privacy trade-off can be achieved thus depends on the parameter m of the number of correlated records.

If we make no assumptions about the pedigree graph of the data, than in the worst case we must expect that all n records are correlated and we have $m = n$. In this case, the error of the Laplace mechanism for $\varepsilon = 2$ exceeds the size of the data tuple n with probabilities as high as $\beta = 10\%$ (see solid lines in Figure 7). Even the general best-case errors (dashed lines) reach up to 50% of the size of the data tuple. The utility degrades further when



(a) $\varepsilon = 0.5$ (b) $\varepsilon = 1$ (c) $\varepsilon = 2$

Figure 7: Accuracy of an ε-BDP counting query on genomic data. The y-axis shows the respective value of error α for a given probability β on the x-axis. The (α, β)-accuracy of the Laplace mechanism applied to a counting query g is shown as a solid line. The lower bound of α for a BDP counting query on genomic data is shown as dashed. The results are displayed for different privacy leakages ε and number of correlated records m.

decreasing the privacy leakage ε. Thus, we conclude that the maximum size of families in the pedigree graph must be limited in this situation to achieve good accuracy. How severely the maximum size must be limited would depend on the specific application and the error tolerance of the data analyst.

It may still be possible to find an improved bound for the BDPL of an ε-DP algorithm on genomic data if the structure of the pedigree graph is further restricted. The restriction would need to be designed so that our tightness proof (see Theorem 5.1 and Corollary 5.2) no longer applies. E.g., one could limit the size of *core families* (families consisting only of parents and their direct children) to l, and the size of entire (multi-generational) families to $m > l$. Here, the general bound would still be $m\varepsilon$, but we hypothesize that an improved bound closer to $l\varepsilon$ may be possible because the correlation between a person's allele pair and their ancestor's allele pair should weaken with every generation between them. Finding the exact bound and proving it is a possible avenue for future work.

6 Multivariate Gaussian Correlation Model

A Gaussian distribution is an appropriate model for a wide variety of phenomena [39, p. 14]; for example, the distribution of heights in the population follows a bell curve [7]. When we are dealing with data tuple of n records, and each record is drawn from a Gaussian distribution, then we can model the joint distribution of all records as a multivariate Gaussian distribution. This model also able to capture the linear correlation between any records [43, p. 28].

In this section, we explore the relationship between DP and BDP when the data is drawn from a multivariate Gaussian distribution. We find a new upper bound for the BDPL in Proposition 6.4, dubbed the *Gaussian bound*. When the Pearson correlation coefficient ρ is sufficiently small, the Gaussian bound improves on the general bound of $m\varepsilon$, where m is the number of correlated records (see Figure 8). However, we will see that the Gaussian bound is not tight. Therefore, we run simulations to compare the theoretical and empirical BDPL for different strengths of correlation. The results can be seen in Figure 9. Finally, we investigate the utility and show under which circumstances ε-BDP algorithms can achieve similar accuracy to ε-DP algorithms (see Corollary 6.5).

Definition 6.1 (Multivariate Gaussian Distribution [43])**.** Let $\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathbb{R}^n$ be a random vector, let vector $\mu \in \mathbb{R}^n$ be real and let matrix $\Sigma \in \mathbb{R}^{n \times n}$ be symmetric and positive definite. We say $\mathbf{X}$ follows the *multivariate Gaussian distribution with mean μ and covariance* Σ if the probability density of $\mathbf{X}$ for any point $\mathbf{x} \in \mathbb{R}^n$ is

$$p_{\mathbf{X}}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp(-\frac{1}{2}(\mathbf{x} - \mu)^\top \Sigma^{-1} (\mathbf{x} - \mu)) \tag{115}$$

where $|\Sigma|$ denotes the determinant of Σ. We denote that $\mathbf{X}$ follows the multivariate Gaussian distribution by $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$.

More specifically, this section concerns tuples $D = (x_1, \dots, x_n) \in \mathbb{R}^n$ that are drawn from a multivariate Gaussian distribution where each record corresponds to one component of the random vector $\mathbf{X}$. Note that the multivariate Gaussian distribution has the infinite domain $\mathcal{X}^n := \mathbb{R}^n$, so any real number is theoretically possible for a record x_i.

A multivariate Gaussian distribution is defined by its mean vector μ and covariance matrix Σ. These two parameters correspond to the expected value $\mathbb{E}[\mathbf{X}] = \mu$ and the actual covariance $\mathbb{E}[(\mathbf{X} - \mathbb{E}\mathbf{X})(\mathbf{X} - \mathbb{E}\mathbf{X})^\top] = \Sigma$ of random vector $\mathbf{X}$. This also means that the covariance between single random variables X_i and X_j for $i \neq j$ is found in entry Σ_{ij} of the covariance matrix [43, p. 28].

The Gaussian bound will apply to a certain class of algorithms: Those that are both DP and fulfill d-privacy when using the ℓ_1-distance for d. An example of such an algorithm is a sum query that clips the input to a specified range, computes the sum, and adds Laplacian noise. To prove this, we will first establish a relationship in Proposition 6.1 and Theorem 6.2 between d-privacy (see Definition 2.6) and an analogous form of BDP which we call d-BDP. Subsequently, we use this result in Proposition 6.4 to investigate the relationship between conventional BDP and algorithms that are both DP and d-private.

6.1 Relationship between d-Privacy and d-BDP

This section concerns algorithms that are d-private, a generalisation of DP where the allowed privacy leakage depends on the distance between data tuples. See Definition 2.6. We define an equivalent generalisation of BDP that depends on the distance between value x and x' of the target record.

Definition 6.2 (Target Dependent BDPL)**.** Let random vector $\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathcal{X}^n$ follow distribution π. Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ be a randomized algorithm whose input data is drawn from π. Let $i \in [n]$ be the index of the target record by the adversary, and let $K \subseteq [n] \setminus \{i\}$ be the indices of records that are known to the adversary. Then, the *adversary-specific target dependent BDPL* of $\mathcal{A}$ w.r.t. adversary (K, i) for any target values $x, x' \in \mathcal{X}$ is

$$BDPL_{(K,i)}(x, x') = \sup_{\mathbf{x}_K, s} \ln \frac{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x)}{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x')}. \tag{116}$$

The *target dependent BDPL* of $\mathcal{A}$ is

$$BDPL(x, x') = \sup_{i,K} BDPL_{(K,i)}(x, x'). \tag{117}$$

Definition 6.3 (d-Bayesian Differential Privacy)**.** Let d be a distance metric on $\mathcal{X}$. Algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is *d-Bayesian differentially private* if and only if for all values $x, x' \in \mathcal{X}$ we have $BDPL(x, x') \leq d(x, x')$.

This definition is very similar to the original definition of BDP, Definition 2.12. The only difference is that d-BDP does not take the supremum over x, x'. Privacy notion d-BDP is equivalent to BDP for distance metric $d(x, x') = \varepsilon$ if $x \neq x'$ and $d(x, x') = 0$ otherwise.

The target dependent BDPL of an algorithm is defined as the maximum over all adversary-specific $BDPL_{(K,i)}(x, x')$, for any adversary (K, i). Therefore, in order to obtain a bound on BDPL, we divide the problem into two steps: First, we derive the adversary-specific target dependent BDPL in Proposition 6.1. Then, we use this result to derive a bound on the target dependent BDPL of a d-private algorithm for any adversary in Theorem 6.2.

Proposition 6.1. Let $\varepsilon > 0$ be the privacy parameter and $n \geq 2$ be the size of the data tuple. Let $\mathcal{A}$ with data domain $\mathbb{R}^n$ be a d-private algorithm with distance metric $d(D, D') = \varepsilon || D - D' ||_1$ for any tuples $D, D' \in \mathbb{R}^n$. Further, $0 \leq k \leq n-2$ and $u \in \mathbb{N}$ are the number of known and unknown records by the adversary, respectively, with $n = k + u + 1$. Let $K = \{n-k, \dots, n-1\}$ be the set of known indices for the adversary. Finally, the data tuples are drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ with

$$\mu = \begin{pmatrix} \mu_U \\ \mu_{K,n} \end{pmatrix}, \tag{118}$$

$$\Sigma = \begin{pmatrix} \Sigma_U & \Sigma_{U;K,n} \\ \Sigma_{U;K,n}^\top & \Sigma_{K,n} \end{pmatrix} \tag{119}$$

for mean $\mu_U \in \mathbb{R}^u$ and $\mu_{K,n} \in \mathbb{R}^{k+1}$ and covariance $\Sigma_U \in \mathbb{R}^{u\times u}$, $\Sigma_{K,n} \in \mathbb{R}^{(k+1)\times(k+1)}$ and $\Sigma_{U;K,n} \in \mathbb{R}^{u\times(k+1)}$. Let the matrix $\Sigma_{K,n}$ be invertible. Then, the adversary-specific target dependent BDPL of $\mathcal{A}$ for any target values $x_n, x'_n \in \mathbb{R}$ is limited by

$$BDPL_{(K,n)}(x_n, x'_n) \leq (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0},1)^\top||_1 + 1)d(x_n, x'_n) \tag{120}$$

where the notation $(\mathbf{0},1)^\top$ denotes the vector $(0,\dots,0,1)^\top \in \mathbb{R}^{k+1}$.

Proof. Let Y be the random variable that represents the output of algorithm $\mathcal{A}$. Let random vector $\mathbf{X} = (X_1,\dots,X_n)^\top$ follow the multivariate Gaussian distribution $\mathbf{X} \sim \mathcal{N}(\mu,\Sigma)$. Consider the definition of the adversary-specific target dependent BDPL of (K,n). From eq. (116) we have

$$BDPL_{(K,n)}(x_n, x'_n) = \sup_{\mathbf{x}_K, s} \ln \frac{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)}{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x'_n)}. \tag{121}$$

Let $s \in \mathbb{R}$ and known values $\mathbf{x}_K \in \mathbb{R}^k$ be arbitrary. We will calculate the adversary-specific target dependent BDPL by rewriting the density $p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)$ in terms of $p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x'_n)$, where the value of X_n has changed. First, we apply the law of total probability to rewrite the probability density.

$$p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n) = \int_{\mathbb{R}^u} p_Y(s \mid \mathbf{X}_U = \mathbf{x}_U, \mathbf{X}_K = \mathbf{x}_K, X_n = x_n) \cdot p_{\mathbf{X}_U}(\mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)\,\mathrm{d}\mathbf{x}_U \tag{122}$$

To complete the proof, we must rewrite the latter two probability densities with their respective counterpart that has condition $X_n = x'_n$ instead of $X_n = x_n$. Later, we achieve this with integration by substitution.

We will now analyze $p_{\mathbf{X}_U}(x_u \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)$ to see which substitution is useful. Since $\mathbf{X}$ is jointly Gaussian distributed, this conditional distribution is also Gaussian distributed [41, p. 40]. To simplify notation, we combine the random vector $\mathbf{X}_K$ and random variable X_n into one vector $\mathbf{X}_{K\cup\{n\}}$ with $\mathbf{x}_{K,n} = (\mathbf{x}_K, x_n)^\top$ and $\mathbf{x}'_{K,n} = (\mathbf{x}_K, x'_n)^\top$. Now we can express the conditional distribution with the formulas from [41, p. 40]. Let $\tilde{\mathbf{x}}_U \in \mathbb{R}^u$ be arbitrary. We have

$$p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U \mid \mathbf{X}_{K\cup\{n\}} = \mathbf{x}_{K,n}) = \mathcal{N}(\hat{\mu}_U, \hat{\Sigma}_U) \text{ with}$$

$$\hat{\mu}_U = \mu_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mu_{K,n}), \tag{123}$$

$$\hat{\Sigma}_U = \Sigma_U - \Sigma_{U;K,n}\Sigma_{K,n}^{-1}\Sigma_{U;K,n}^\top. \tag{124}$$

It is important to note that only the conditional mean $\hat{\mu}_U$ depends on the specific value $\mathbf{x}_{K,n}$, the conditional covariance $\hat{\Sigma}_U$ remains fixed. Therefore, the two distributions (conditioned on $\mathbf{x}_{K,n}$ and $\mathbf{x}'_{K,n} \in \mathbb{R}^{k+1}$ respectively) only differ by a translation. To demonstrate this, here is the conditional distribution for $\mathbf{x}'_{K,n}$:

$$
\begin{aligned}
p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U \mid \mathbf{X}_{K\cup\{n\}} = \mathbf{x}'_{K,n}) = \mathcal{N}(\hat{\mu}'_U, \hat{\Sigma}'_U) \text{ with}& \\
\hat{\mu}'_U &= \mu_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}'_{K,n} - \mu_{K,n}) && (125)\\
&= \hat{\mu}_U - \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n}), && (126)\\
\hat{\Sigma}'_U &= \Sigma_U - \Sigma_{U;K,n}\Sigma_{K,n}^{-1}\Sigma_{U;K,n}^{\top} && (127)\\
&= \hat{\Sigma}_U && (128)
\end{aligned}
$$

With eq. (126) we can derive the following equivalence between the probability densities of the unknown random variables, conditioned on $X_n = x_n$ or $X_n = x'_n$ respectively. We plug in a particular value into the first probability density, to show that it then becomes equal to the probability density in eq. (131).

$$
\begin{aligned}
&p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n}) \mid \mathbf{X}_{K\cup\{n\}} = \mathbf{x}_{K,n})\\
&\quad= \frac{1}{\sqrt{(2\pi)^u|\hat{\Sigma}_U|}} \exp\Big(-\frac{1}{2}(\tilde{\mathbf{x}}_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n}) - \hat{\mu}_U)^{\top}\hat{\Sigma}_U^{-1}\\
&\quad\quad (\tilde{\mathbf{x}}_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n}) - \hat{\mu}_U)\Big) && (129)\\
&\quad= \frac{1}{\sqrt{(2\pi)^u|\hat{\Sigma}'_U|}} \exp\Big(-\frac{1}{2}(\tilde{\mathbf{x}}_U - \hat{\mu}'_U)^{\top}\hat{\Sigma}_U'^{-1}(\tilde{\mathbf{x}}_U - \hat{\mu}'_U)\Big) && (130)\\
&\quad= p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U \mid \mathbf{X}_{K\cup\{n\}} = \mathbf{x}'_{K,n}) && (131)
\end{aligned}
$$

First, in eq. (129) we expand the definition of the multivariate Gaussian probability density from Definition 6.1 with mean $\hat{\mu}_U$ and covariance matrix $\hat{\Sigma}_U$. Then we use eq. (126) in eq. (130). The resulting equation now contains the probability density with mean $\hat{\mu}'_U$ and covariance matrix $\hat{\Sigma}'_U$, which corresponds to the density with condition $\mathbf{X}_{K\cup\{n\}} = \mathbf{x}'_{K,n}$. We use this fact in eq. (131). Therefore, we have shown that the condition of the probability density $p_{\mathbf{X}_U}$ can simply be changed by additively shifting the input for the density.

As a result, we substitute $\mathbf{x}_U$ with $\tilde{\mathbf{x}}_U + \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n})$ in the integral from eq. (122): It allows us to change density $p_{\mathbf{X}_U}$ from being conditioned on $X_n = x_n$ to being conditioned on $X_n = x'_n$. To shorten notation of the substitution from here on out, we define $\gamma := \Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n})$.

We also have to consider the effect of this substitution on the other probability density p_Y in the integral. Note that the entire random vector $\mathbf{X}$ is defined in the probability density in eq. (132), so random variable Y only depends on the randomization of algorithm $\mathcal{A}$. Therefore, we can use the fact that algorithm $\mathcal{A}$ is d-private with the ℓ_1-distance.

$$
\begin{aligned}
&p_Y(s \mid \mathbf{X}_U = \tilde{\mathbf{x}}_U + \gamma, \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)\\
&\qquad \le \exp((||\gamma||_1 + |x_n - x'_n|)\varepsilon) \cdot p_Y(s \mid \mathbf{X}_U = \tilde{\mathbf{x}}_U, \mathbf{X}_K = \mathbf{x}_K, X_n = x'_n) \qquad (132)
\end{aligned}
$$

The equivalence from eq. (131) and the inequality from eq. (132) now come together to reformulate the density $p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x_n)$ in terms of $p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_n = x'_n)$

where the targeted random variable has changed from value x_n to x'_n. We once more employ the law of total probability to introduce the unknown values $\mathbf{x}_U$.

$$p_Y(s \mid \mathbf{X}_K = x_K, X_n = x_n)$$

$$= \int_{\mathbb{R}^u} p_Y(s \mid \mathbf{X}_U = \mathbf{x}_U, \mathbf{X}_K = x_K, X_n = x_n) \cdot p_{\mathbf{X}_U}(\mathbf{x}_U \mid \mathbf{X}_K = x_K, X_n = x_n)\, \mathrm{d}\mathbf{x}_U \tag{133}$$

$$= \int_{\mathbb{R}^u} p_Y(s \mid \mathbf{X}_U = \tilde{\mathbf{x}}_U + \gamma, \mathbf{X}_K = x_K, X_n = x_n) \cdot p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U + \gamma \mid \mathbf{X}_K = x_K, X_n = x_n)\, \mathrm{d}\tilde{\mathbf{x}}_U \tag{134}$$

$$= \int_{\mathbb{R}^u} p_Y(s \mid \mathbf{X}_U = \tilde{\mathbf{x}}_U + \gamma, \mathbf{X}_K = x_K, X_n = x_n) \cdot p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U \mid \mathbf{X}_K = x_K, X_n = x'_n)\, \mathrm{d}\tilde{\mathbf{x}}_U \tag{135}$$

$$\leq \exp((||\gamma||_1 + |x_n - x'_n|)\varepsilon) \cdot$$

$$\int_{\mathbb{R}^u} p_Y(s \mid \mathbf{X}_U = \tilde{\mathbf{x}}_U, \mathbf{X}_K = x_K, X_n = x'_n) \cdot p_{\mathbf{X}_U}(\tilde{\mathbf{x}}_U \mid \mathbf{X}_K = x_K, X_n = x'_n)\, \mathrm{d}\tilde{\mathbf{x}}_U \tag{136}$$

$$= \exp((||\gamma||_1 + |x_n - x'_n|)\varepsilon) \cdot p_Y(s \mid \mathbf{X}_K = x_K, X_n = x'_n) \tag{137}$$

We substitute $\mathbf{x}_U$ for $\tilde{\mathbf{x}}_U + \gamma$ in eq. (134) using the change of variable theorem for multiple integrals [44, p. 310]. The substitution is linear so the domain $\mathbb{R}^u$ over which we integrate does not change, and the determinant of the Jacobian of the substitution is simply 1. Then, we apply eq. (131) in eq. (135). Similarly, we use the inequality from eq. (132) in eq. (136). Finally, the law of total probability is once again employed in eq. (137).

Now, we can formulate the upper bound of the adversary-specific target dependent BDPL for (K, n) by applying eq. (137) in eq. (121). Let $x_n, x'_n \in \mathbb{R}$ be arbitrary.

$$\begin{aligned}
BDPL_{(K,n)}(x_n, x'_n) &\leq (||\gamma||_1 + |x_n - x'_n|)\varepsilon \\
&= (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{x}_{K,n} - \mathbf{x}'_{K,n})||_1 + |x_n - x'_n|)\varepsilon \\
&= (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}((\mathbf{x}_K, x_n)^\top - (\mathbf{x}_K, x'_n)^\top)||_1 + |x_n - x'_n|)\varepsilon \\
&= (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0}, x_n - x'_n)^\top||_1 + |x_n - x'_n|)\varepsilon \\
&= (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0}, 1)^\top||_1 + 1)|x_n - x'_n|\varepsilon \\
&= (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0}, 1)^\top||_1 + 1)d(x_n, x'_n)
\end{aligned}$$

□

Proposition 6.1 has given an upper bound on the adversary-specific target dependent BDPL. Now, we will prove an upper bound for the target dependent BDPL for a d-private algorithm, given that the maximum Pearson correlation coefficient between any two random variables X_i and X_j is bounded by $\rho < \frac{1}{n-2}$. This means correlation is restricted to be very small.

Definition 6.4. For $\rho \in [0, 1]$ and $n \in \mathbb{N}$, we call the matrix $\Sigma_\rho \in \mathbb{R}^{n \times n}$ a *limited covariance matrix* if

1. The matrix Σ_ρ is symmetric and positive definite.

2. The diagonal of Σ_ρ is constant. I.e., there is a variance $\sigma^2 > 0$ so that $\Sigma_{\rho,ii} = \sigma^2$ for all $i \in [n]$.
3. Any pairwise correlation is limited by ρ. I.e., for all $i \neq j$ we have $|\Sigma_{\rho,ij}| \leq \rho\sigma^2$.

Theorem 6.2. Let $\varepsilon > 0$ be the privacy parameter, $n \geq 3$ the size of the data tuple, and $\rho < \frac{1}{n-2}$ the maximum correlation coefficient. Let $\mathcal{A}$ with data domain $\mathbb{R}^n$ be a d-private algorithm with distance metric $d(D, D') = \varepsilon||D - D'||_1$ for any tuples $D, D' \in \mathbb{R}^n$. Let the tuples be drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma_\rho)$ with mean $\mu \in \mathbb{R}^n$ and limited covariance matrix $\Sigma_\rho \in \mathbb{R}^{n\times n}$. Then, for any $x, x' \in \mathbb{R}$ we have

$$BDPL(x, x') \leq \left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1 \right) d(x, x'). \tag{138}$$

Proof. To prove an upper bound for the target dependent BDPL, we must bound the adversary-specific target dependent BDPL of every possible adversary (K, i) with $i \in [n]$ and $K \subseteq [n] \setminus \{i\}$. The remaining indices besides K and i make up the unknown indices $U = [n] \setminus (K \cup \{i\})$. Proposition 6.1 requires that there is at least one unknown index in U. Thus, we differentiate between two cases.

Case 1: There are no unknown indices; we have $U = \emptyset$.

Here, the set of known indices $K = [n] \setminus \{i\}$ must contain every index except the target. Thus, the entire data set is made up of the known values $\mathbf{x}_K$ and the target value $x_i \in \mathbb{R}$ or $x_i' \in \mathbb{R}$. We define $D = (\mathbf{x}_K, x_i)^\top$ and $D' = (\mathbf{x}_K, x_i')^\top$. We can calculate the target dependent BDPL of this adversary by using the fact that $\mathcal{A}$ is d-private.

$$\begin{aligned}
BDPL_{(K,i)}(x_i, x_i') &= \sup_{\mathbf{x}_K, s} \ln \frac{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i)}{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i')} && (139)\\
&= \sup_{\mathbf{x}_K, s} \ln \frac{p_Y(s \mid \mathbf{X} = D)}{p_Y(s \mid \mathbf{X} = D')} && (140)\\
&\leq \sup_{\mathbf{x}_K, s} \ln \frac{e^{\varepsilon||D-D'||_1} p_Y(s \mid \mathbf{X} = D')}{p_Y(s \mid \mathbf{X} = D')} && (141)\\
&= \ln e^{\varepsilon|x_i - x_i'|} = \varepsilon|x_i - x_i'| = d(x_i, x_i') && (142)\\
&\leq \left(\frac{n^2}{4(\frac{1}{\rho} - n + 3)} + 1 \right) d(x_i, x_i') && (143)
\end{aligned}$$

Case 2: There is at least one unknown record in $U \neq \emptyset$.

Let $k = |K|$ be the number of known records and $u = |U|$ the number of unknown records. W.l.o.g. we have $K = \{n - k, \dots, n - 1\}$ and $i = n$. Otherwise, we simply reorder the components of random vector $\mathbf{X}$ so that the statements about K and i apply.

Choose $\Sigma_U \in \mathbb{R}^{u\times u}$, $\Sigma_{U;K,n} \in \mathbb{R}^{u\times(k+1)}$ and $\Sigma_{K,n} \in \mathbb{R}^{(k+1)\times(k+1)}$ so that the following holds:

$$\Sigma_\rho = \begin{pmatrix} \Sigma_U & \Sigma_{U;K,n} \\ \Sigma_{U;K,n}^\top & \Sigma_{K,n} \end{pmatrix} \tag{144}$$

In order to use Proposition 6.1, we require that the covariance matrix of the known records $\Sigma_{K,n}$ is invertible. We show that this is the case: A matrix is invertible if none of its eigenvalues are zero. We will find a lower bound for the eigenvalues of $\Sigma_{K,n}$ that is strictly positive. Conveniently, we can later also use this fact to bound the entries of $\Sigma_{K,n}^{-1}$. To find the bound on the eigenvalues, we must refer to individual cells of $\Sigma_{K,n}$.

$$\Sigma_{K,n} =: (a_{jl})_{j,l\in[k+1]} \tag{145}$$

Now, we can bound the eigenvalues with the Gershgorin circle theorem [17] and the fact that a symmetric matrix only has real eigenvalues [19, p. 1]. According to the Gershgorin circle theorem, every eigenvalue of a real matrix such as $\Sigma_{K,n}$ is contained in a closed disc on the complex number plane with center a_{jj} and radius $\sum_{l\neq j} |a_{jl}|$ for $j \in [k+1]$. Since the eigenvalues of the matrix must be real, we are only concerned with the real part of this disc, i.e., the interval $[a_{jj} - \sum_{l\neq j} |a_{jl}|, a_{jj} + \sum_{l\neq j} |a_{jl}|]$. We can construct a lower bound of the smallest eigenvalue λ_- of $\Sigma_{K,n}$ by finding the lowest border of these intervals.

$$\begin{aligned}
\lambda_- &\geq \min_j a_{jj} - \sum_{l\neq j} |a_{jl}| && (146)\\
&\geq \min_j \sigma^2 - \sum_{l\neq j} |\rho\sigma^2| && (147)\\
&= \sigma^2 - k\rho\sigma^2 && (148)\\
&= (1-k\rho)\sigma^2 && (149)\\
&> (1-(n-2)\cdot\frac{1}{n-2})\sigma^2 = 0 && (150)
\end{aligned}$$

In eq. (147), we use the fact that every random variable has the same variance σ^2 and the correlation between any two random variables in $\mathbf{X}$ is in the interval $[-\rho, \rho]$. See Definition 6.4 on the limited covariance matrix. Then we show in eq. (150) that the bound is positive: The number of known records k must be $n-2$ or smaller because there is one targeted record and at least one unknown record in $U \neq \emptyset$. Additionally, we use that the maximum correlation ρ is bounded with $\rho < \frac{1}{n-2}$. Thus, we have shown that each eigenvalue of $\Sigma_{K,n}$ is strictly positive and the matrix is therefore invertible.

Now, nothing is standing in our way of applying Proposition 6.1. Thus, we have an upper bound of the adversary-specific target dependent BDPL for any $x_i, x_i' \in \mathbb{R}$ with

$$BDPL_{(K,i)}(x_i, x_i') \leq (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0},1)||_1 + 1)d(x_i, x_i'). \tag{151}$$

This bound still depends on the specific matrices $\Sigma_{U;K,n}$ and $\Sigma_{K,n}$ for this adversary. Our goal is to find a bound for the total target dependent BDPL, irrespective of the specific adversary. To find a general upper bound for this expression that does not depend on the specific matrices at hand, we name the cells of both matrices similar to our naming convention for $\Sigma_{K,n}$.

$$\Sigma_{U;K,n} =: (b_{jl})_{j\in[u],l\in[k+1]} \tag{152}$$
$$\Sigma_{K,n}^{-1} =: (\alpha_{jl})_{j,l\in[k+1]} \tag{153}$$

To find a more universal bound for eq. (151), we bound the entries of $\Sigma_{U;K,n}$ and $\Sigma_{K,n}^{-1}$. Let us begin with the first matrix. It only contains covariances between random variables from U and random variables from K or n. Thus, we use the fact the covariance is bounded by $\rho\sigma^2$ because the maximum correlation between two random variables is ρ. For all indices $j, l \in [u]$ we obtain the result

$$|b_{jl}| \leq \rho\sigma^2. \tag{154}$$

Now, we bound the entries of the inverse matrix. We first show that the entries of a matrix A^{-1} must be smaller than the inverse of the smallest absolute eigenvalue λ_- of A. Consider the 2-norm of any symmetric matrix $A \in \mathbb{R}^{d\times d}$, defined as

$$||A||_2 := \sup\{||Av||_2 \mid ||v||_2 = 1\}. \tag{155}$$

The 2-norm of A is equal to its maximum singular value (i.e., the largest absolute eigenvalue for a symmetric matrix) [48, p. 47]. Thus, we can use the 2-norm to bound the entries of a symmetric matrix by its largest absolute eigenvalue $|\lambda_+|$. Let $e_j \in \mathbb{R}^d$ be the vector with 1 at position j and 0 elsewhere. For every entry a_{ij} we have

$$|a_{ij}| = \sqrt{a_{ij}^2} \leq \sqrt{\sum_{k\in[m]} a_{kj}^2} = ||Ae_j||_2 \leq ||A||_2 = |\lambda_+|. \tag{156}$$

Additionally, the eigenvalues of an inverse A^{-1} can be determined knowing the eigenvalues of A. Let $\lambda \in \mathbb{R}$ be any eigenvalue of A; λ cannot be zero because A is invertible.

$$\begin{aligned} && Ax &= \lambda x \\ \Leftrightarrow && A^{-1}Ax &= \lambda A^{-1}x \\ \Leftrightarrow && x &= \lambda A^{-1}x \\ \Leftrightarrow && \frac{1}{\lambda}x &= A^{-1}x \end{aligned}$$

Thus, the eigenvalues of A^{-1} are the inverses of the eigenvalues of A. Putting it all together, the entries of $\Sigma_{K,n}^{-1}$ are smaller or equal to the inverse of the smallest absolute eigenvalue of $\Sigma_{K,n}$. In eq. (149), we have shown that all eigenvalues of $\Sigma_{K,n}$ are positive and larger than $(1-k\rho)\sigma^2$. Therefore, the smallest *absolute* eigenvalue of $\Sigma_{K,n}$ is also larger than this bound. Now, these two facts (the entries of a matrix can be bounded by the largest absolute eigenvalue, and the eigenvalues of A^{-1} are the inverses of the eigenvalues of A) are brought together to bound the entries of $\Sigma_{K,n}^{-1}$:

$$\alpha_{jl} \leq \frac{1}{\lambda_-} \leq \frac{1}{(1-k\rho)\sigma^2} \tag{157}$$

With the bounds for the entries of $\Sigma_{U;K,n}$ and $\Sigma_{K,n}^{-1}$ in hand, we can find a general upper bound for the adversary-specific target dependent BDPL for any $x_i, x_i' \in \mathbb{R}$.

$$\begin{aligned}
BDPL_{(K,i)}(x_i, x_i') &\leq (||\Sigma_{U;K,n}\Sigma_{K,n}^{-1}(\mathbf{0},1)^\top||_1 + 1)d(x_i, x_i') && (158)\\
&= (||\Sigma_{U;K,n}(\alpha_{1,k+1},\dots,\alpha_{k+1,k+1})^\top||_1 + 1)d(x_i, x_i') && (159)\\
&= \left(||(\sum_{l=1}^{k+1}\alpha_{l,k+1}\cdot b_{1,l},\dots,\sum_{l=1}^{k+1}\alpha_{l,k+1}\cdot b_{u,l})^\top||_1 + 1\right)d(x_i, x_i') && (160)\\
&= \left(\sum_{j=1}^{u}|\sum_{l=1}^{k+1}\alpha_{l,k+1}\cdot b_{j,l}| + 1\right)d(x_i, x_i') && (161)\\
&\leq \left(\sum_{j=1}^{u}|\sum_{l=1}^{k+1}\frac{\rho\sigma^2}{(1-k\rho)\sigma^2}| + 1\right)d(x_i, x_i') && (162)\\
&= \left(\frac{u(k+1)\rho}{1-k\rho} + 1\right)d(x_i, x_i') && (163)\\
&= \left(\frac{u(k+1)}{\frac{1}{\rho}-k} + 1\right)d(x_i, x_i') && (164)\\
&\leq \left(\frac{(\frac{n}{2})^2}{\frac{1}{\rho}-n+2} + 1\right)d(x_i, x_i') && (165)\\
&= \left(\frac{n^2}{4(\frac{1}{\rho}-n+2)} + 1\right)d(x_i, x_i') && (166)
\end{aligned}$$

We first use the inequality from Proposition 6.1 in eq. (158). Then, we multiply $\Sigma_{K,n}^{-1}$ and $(\mathbf{0},1)^\top$; only the last column of $\Sigma_{K,n}^{-1}$ remains in eq. (159). The result is multiplied with $\Sigma_{U;K,n}$ in eq. (160). Afterwards, in eq. (161) we apply the ℓ_1-distance to the remaining vector. The bounds for the entries α and b are used in eq. (162). Keep in mind that the denominator is always positive because $k \leq n-2$ and $\rho < \frac{1}{n-2}$. Then the two sums can be simplified by multiplying by their cardinality in eq. (163) since the entries of the sum no longer depend on j or l. Finally, we bound the numerator and denominator in eq. (165): The numerator $u(k+1)$ is smaller or equal to $(n/2)^2$ because u and $k+1$ are positive and together form $n = u + k + 1$. Thus, their product is maximal if they meet at the exact midpoint to n. As mentioned previously, the denominator is always positive. It therefore becomes minimal (and the entire expression maximal) if k becomes maximal. This is the case for $k = n-2$.

In both cases we were able to bound adversary-specific target dependent BDPL as in eq. (138). Thus, the proof is complete. □

6.2 Relationship between DP and BDP

So far, the proofs have only covered the relationship between d-privacy and d-BDP. However, we are mainly interested in the relationship between DP and BDP. Therefore, Definition 6.5 and Lemma 6.3 show how any d-private algorithm with the ℓ_1-distance can be made DP by clipping the input data.

Definition 6.5. For real values $a, b \in \mathbb{R}$ with $a < b$, we call $c_{a,b} : \mathbb{R}^n \to \mathbb{R}^n$ a *clipping function* if for all $D \in \mathbb{R}^n$ and all $i \in [n]$ we have

$$c_{a,b}(D)_i = \max(a, \min(b, D_i)). \tag{167}$$

Let $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}$ be a d-private algorithm. Then we call $\mathcal{A}_{a,b} : \mathbb{R}^n \to \mathbb{R}$ the *clipped version of* $\mathcal{A}$ if for every input $D \in \mathbb{R}^n$, the algorithm $\mathcal{A}_{a,b}$ gives the output $\mathcal{A}(c_{a,b}(D))$.

Lemma 6.3. Let $\mathcal{A}_{a,b}$ be the clipped version of a d-private algorithm $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}$ with $d(D, D') = \varepsilon||D - D'||_1$ for any tuples $D, D' \in \mathbb{R}^n$. Then, $\mathcal{A}_{a,b}$ is d-private and $(b-a)\varepsilon$-DP.

Proof. We begin by showing that $\mathcal{A}_{a,b}$ is d-private. Let $D_1, D_2 \in \mathbb{R}^n$ be arbitrary and $S \subseteq \mathbb{R}$ be any measurable set. We have

$$\begin{aligned}
\Pr[\mathcal{A}_{a,b}(D_1) \in S] &= \Pr[\mathcal{A}(c_{a,b}(D_1)) \in S] && (168)\\
&\leq e^{\varepsilon||c_{a,b}(D_1) - c_{a,b}(D_2)||_1} \Pr[\mathcal{A}(c_{a,b}(D_2)) \in S] && (169)\\
&\leq e^{\varepsilon||D_1 - D_2||_1} \Pr[\mathcal{A}(c_{a,b}(D_2)) \in S] && (170)\\
&= e^{\varepsilon||D_1 - D_2||_1} \Pr[\mathcal{A}_{a,b}(D_2) \in S]. && (171)
\end{aligned}$$

In eq. (169) we use that $\mathcal{A}$ is d-private. Then, we apply the fact that the ℓ_1-distance of two clipped data sets is smaller or equal to the ℓ_1-distance of the original data sets in eq. (170). Finally, eq. (171) shows that $\mathcal{A}_{a,b}$ is d-private.

Now, we will show that $\mathcal{A}_{a,b}$ is also DP. Let $D, D' \in \mathbb{R}^n$ be arbitrary neighboring data sets, i.e., there only exists a single index $i \in [n]$ with $D_i \neq D'_i$. For any measurable set $S \subseteq \mathcal{Y}$ we have

$$\begin{aligned}
\Pr[\mathcal{A}_{a,b}(D) \in S] &= \Pr[\mathcal{A}(c_{a,b}(D)) \in S] && (172)\\
&\leq e^{\varepsilon||c_{a,b}(D) - c_{a,b}(D')||_1} \Pr[\mathcal{A}(c_{a,b}(D')) \in S] && (173)\\
&= e^{\varepsilon \sum_{j \in [n]} |\max(a,\min(b,D_j)) - \max(a,\min(b,D'_j))|} \Pr[\mathcal{A}(c_{a,b}(D')) \in S] && (174)\\
&= e^{\varepsilon |\max(a,\min(b,D_i)) - \max(a,\min(b,D'_i))|} \Pr[\mathcal{A}(c_{a,b}(D')) \in S] && (175)\\
&\leq e^{\varepsilon(b-a)} \Pr[\mathcal{A}(c_{a,b}(D')) \in S] = e^{\varepsilon(b-a)} \Pr[\mathcal{A}_{a,b}(D') \in S]. && (176)
\end{aligned}$$

Once again, we use that $\mathcal{A}$ is d-private in eq. (173). We expand the definition of the ℓ_1-distance and of the clipping function $c_{a,b}$ in eq. (174). Then, we use for eq. (175) that D and D' only differ for index i. Finally, we can bound the difference between the two entries because they are clipped to the interval $[a, b]$. We have shown that $\mathcal{A}_{a,b}$ is $(b-a)\varepsilon$-DP. □

With Lemma 6.3 and Theorem 6.2, we can directly show that this class of DP-algorithms has a limited BDPL. Note that we are now directly establishing a relationship to BDP, not to d-BDP. Proposition 6.4 is the Gaussian bound.

Proposition 6.4 (The Gaussian Bound)**.** Let $\varepsilon > 0$ be real, $n \geq 3$ the size of the data tuple, and $\rho < \frac{1}{n-2}$ the maximum correlation coefficient. Let $\mathcal{A}_{a,b}$ be the clipped version of a d-private algorithm $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}$ with $d(D, D') = \varepsilon||D - D'||_1$. Let the data tuples be drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma_\rho)$ with mean $\mu \in \mathbb{R}^n$ and limited covariance $\Sigma_\rho \in \mathbb{R}^{n\times n}$. Then, the clipped algorithm $\mathcal{A}_{a,b}$ is

$$\left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1\right)(b-a)\varepsilon\text{-BDP}. \tag{177}$$

Proof. We know that $\mathcal{A}_{a,b}$ is a d-private query for $d(D, D') = \varepsilon||D - D'||_1$ because of Lemma 6.3. Thus, we can apply Theorem 6.2 to find a universal bound of the target dependent BDPL in this situation. Therefore, the idea is to show how the BDPL is bounded by the target dependent BDPL, and to then apply Theorem 6.2.

$$\begin{aligned}
BDPL &= \sup_{K,i} BDPL_{(K,i)} && (178)\\
&= \sup_{K,i}\left(\sup_{\mathbf{x}_K, s, |x_i - x_i'| \le b-a} \ln \frac{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i)}{p_Y(s \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i')}\right) && (179)\\
&= \sup_{K,i}\left(\sup_{|x_i - x_i'| \le b-a} BDPL_{(K,i)}(x_i, x_i')\right) && (180)\\
&= \sup_{|x_i - x_i'| \le b-a}\left(\sup_{K,i} BDPL_{(K,i)}(x_i, x_i')\right) && (181)\\
&= \sup_{|x_i - x_i'| \le b-a} BDPL(x_i, x_i') && (182)\\
&\le \sup_{|x_i - x_i'| \le b-a}\left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1\right) d(x_i, x_i') && (183)\\
&= \sup_{|x_i - x_i'| \le b-a}\left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1\right) |x_i - x_i'|\varepsilon && (184)\\
&= \left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1\right)(b-a)\varepsilon. && (185)
\end{aligned}$$

In eqs. (178) and (179) we expand the definition of BDPL. Then, in eq. (180) we use that the adversary-specific *target dependent* BDPL is defined equivalently, except it does not take the supremum over x_i, x_i'. We switch the order of the suprema in eq. (181) to subsequently plug in the definition of the general target dependent BDPL. Then, we use Theorem 6.2 to derive eq. (183). Finally, the last supremum can be resolved by writing out $d(x_i, x_i')$ and bounding it with $(b-a)\varepsilon$. It follows that clipped algorithm $\mathcal{A}$ is BDP since the BDPL is limited. □

6.3 Comparison to State of the Art

We now have the Gaussian bound on the BDPL of certain DP algorithms. It applies when the Pearson correlation coefficient ρ is small, regardless of the number m of correlated records. The question remains when the Gaussian bound improves on the general bound for the BDPL. For a fair comparison, we assume that all $n = m$ records are correlated with each other. This is possible if every entry of Σ is positive. Thus, if algorithm $\mathcal{A}_{a,b}$ is $(b-a)\varepsilon$-DP, it is $n(b-a)\varepsilon$-BDP according to the general bound. The following remark shows that the Gaussian bound improves on the general bound if the Pearson correlation coefficient ρ is restricted to be small, i.e., in the order of $\rho \approx \frac{1}{n}$.

Remark. For number of records $n \geq 3$, range boundaries $a, b \in \mathbb{R}$ with $a < b$, privacy parameter $\varepsilon > 0$ and maximum correlation coefficient $0 < \rho < \frac{1}{n-2}$ we have

$$\begin{aligned}
&\left(\frac{n^2}{4(\frac{1}{\rho}-n+2)}+1\right)(b-a)\varepsilon \leq n(b-a)\varepsilon \\
\Leftrightarrow \quad & \frac{n^2}{4(\frac{1}{\rho}-n+2)}+1 \leq n \\
\Leftrightarrow \quad & \frac{1}{4}n^2 \leq (n-1)(\frac{1}{\rho}-n+2) \\
\Leftrightarrow \quad & \frac{1}{4}n^2 \leq (n-1)\frac{1}{\rho}-n^2+3n-2 \\
\Leftrightarrow \quad & \frac{5}{4}n^2-3n+2 \leq (n-1)\frac{1}{\rho} \\
\Leftrightarrow \quad & \rho \leq \frac{n-1}{\frac{5}{4}n^2-3n+2}
\end{aligned} \tag{186}$$

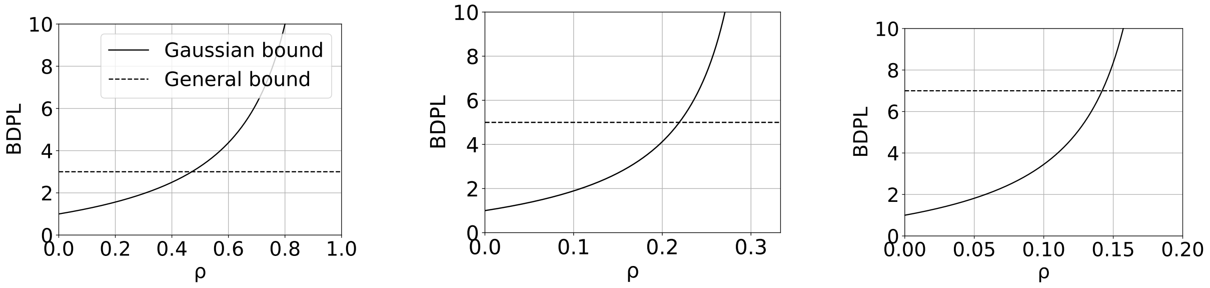
This remark is qualitatively exemplified by Figure 8. It shows the general upper bound (dashed) for clipped algorithms $\mathcal{A}_{a,b}$ that are 1-DP, and for multiple tuple sizes n. The Gaussian bound is shown as a solid black line. The x-axis is the maximum correlation coefficient ρ, while the y-axis shows the BDPL. One can see that the Gaussian bound improves for small correlation coefficients, while the general bound is still smaller for larger correlations. The higher the number of records n, the lower the maximum correlation ρ so that the Gaussian bound improves upon the general bound.

6.4 Empirical BDPL

We have seen that the Gaussian bound is an improvement when correlation is small, however it does not appear to be a tight bound because it goes above the general bound for higher correlation (see Figure 8). Therefore, this raises the question how tight the Gaussian bound is compared to the maximum possible BDPL in realistic situations. To investigate this, we simulate data sets from a multivariate Gaussian distribution to empirically measure the BDPL for the Laplace mechanism applied to a sum query that clips records to the range $[0, 10]$.

6.4.1 Simulation

First, we discuss the methodology of the simulation to empirically estimate the BDPL. The simulation estimates the BDPL for DP privacy leakage $\varepsilon = 1$, data tuples of sizes $n \in \{3, 5, 7\}$ and for 10 maximum Pearson correlation coefficients $\rho \in (0, \frac{1}{n-2})$. This upper bound for ρ is chosen because the Gaussian bound from Proposition 6.4 only applies for



(a) $n = 3$ (b) $n = 5$ (c) $n = 7$

Figure 8: General bound compared to Gaussian bound. The x-axis shows the maximum Pearson correlation coefficient ρ, plotted for the range in which the Gaussian bound applies, i.e., $\rho \in (0, \frac{1}{n-2})$. The y-axis is the BDPL. The solid line shows the Gaussian bound from Proposition 6.4, the dashed line is the general bound $n\varepsilon$ of a 1-DP algorithm $\mathcal{A}_{a,b}$.

$\rho < \frac{1}{n-2}$. To estimate the maximum possible BDPL for each ρ, we use the Monte Carlo method as described in Section 3.1.2 for each choice of n and ρ. For each n and ρ, we test 1 000 different multivariate Gaussian distributions, each with a randomly chosen mean μ and limited covariance matrix Σ_ρ.

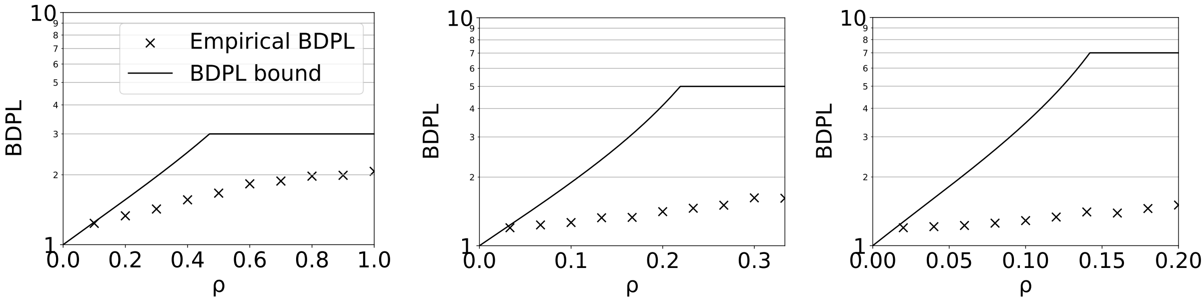
6.4.2 Results With Clipping

The results for the maximum empirical BDPL can be seen in Figure 9. For reference, the figure also includes the minimum of the Gaussian bound and the general bound. Interpreting the result, it appears likely that the bound is not tight overall. For very small Pearson correlation coefficients ρ (e.g., $\rho = 0.1$ for data tuple size $n = 3$) the bound does appear to be tight. However, the distance between the theoretical and empirical results increases as we increase the size of the data tuple n or the maximum Pearson correlation coefficient ρ.

6.4.3 Results Without Clipping

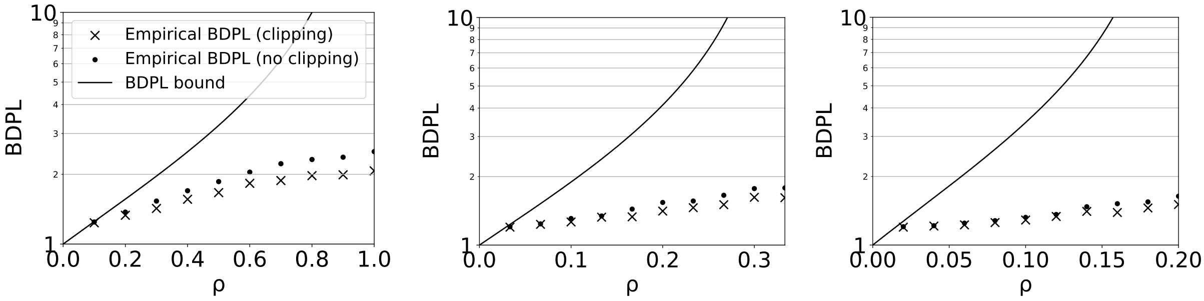
One possible explanation for the gap between the Gaussian bound and the empirical results in Figure 9 is the clipping that is applied to the data set. The Gaussian bound from Theorem 6.2 still applies for d-private algorithms, even if no clipping is used to make them DP. A possible hypothesis is that only the addition of clipping makes the Gaussian bound not be tight. The following results indicate that this is not the case.

We use the same setup from Section 6.4.1, except that the sum query does not clip the values. Now, the general bound does not apply because the algorithm is only d-private, not DP. Figure 10 shows the results for the maximum empirical target-dependent BDPL. The empirical target-dependent BDPL is marginally greater than with clipping, and the gap slightly widens as ρ increases. However, there remains a comparatively large difference between the theoretical upper bound and the empirical target dependent BDPL.



(a) $n = 3$ (b) $n = 5$ (c) $n = 7$

Figure 9: Empirical BDPL compared to upper bound for Gaussian distributed data. The x-axis shows the maximum allowed Pearson correlation coefficient ρ within the multivariate Gaussian distribution, plotted for $\rho \in (0, \frac{1}{n-2})$. The y-axis is the BDPL. The solid line is the minimum of the general bound and the Gaussian bound.



(a) $n = 3$ (b) $n = 5$ (c) $n = 7$

Figure 10: Empirical target dependent BDPL compared to upper bound for Gaussian distributed data. The x-axis shows the maximum allowed Pearson correlation coefficient ρ within the multivariate Gaussian distribution, plotted for $\rho \in (0, \frac{1}{n-2})$. The y-axis is the target dependent BDPL, evaluated for $x_i = 0$ and $x'_i = 10$. The solid line shows the upper bound for the target dependent BDPL from Theorem 6.2. The empirical target dependent BDPL for a sum query without clipping is marked by the dots. For reference, the previous results with clipping are included as crosses.

6.5 Accuracy

While the Gaussian bound on the BDPL is most likely not tight, the bound is an improvement and will be useful for calculating improved utility guarantees of certain BDP algorithms when correlation is sufficiently small. More specifically, we can determine how small correlation has to be so that the error of an ε-BDP algorithm only increases by a fixed factor $h \in \mathbb{R}$ compared to the error of the ε-DP Laplace mechanism. If one only uses the general bound, the error would grow with the size of the data set n as proven by Corollary 4.4, so this shows an improvement.

Corollary 6.5. Let $h > 1$ be a real number, let $n \geq 3$ be the size of the data tuple, let $\varepsilon > 0$ be real and let the maximum correlation parameter be

$$\rho \leq \frac{1}{\frac{n^2}{4(h-1)} + n - 2}. \tag{187}$$

Let $f_{a,b}$ be a deterministic clipped sum query so that for any data tuple $D \in \mathbb{R}^n$ we have

$$f_{a,b}(D) = \sum_{i \in [n]} c_{a,b}(D)_i. \tag{188}$$

Let the data tuples be drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma_\rho)$ with mean $\mu \in \mathbb{R}^n$ and limited covariance $\Sigma_\rho \in \mathbb{R}^{n \times n}$. Let $\mathcal{M}_{\varepsilon, f_{a,b}}$ be the Laplace mechanism. Then, there exists an ε-BDP algorithm $\mathcal{B}_{a,b}$ with the following property: If $\mathcal{M}_{\varepsilon, f_{a,b}}$ is (α, β)-accurate with respect to $f_{a,b}$, then $\mathcal{B}_{a,b}$ is $(h\alpha, \beta)$-accurate with respect to $f_{a,b}$.

Proof. The idea of this proof is to construct algorithm $\mathcal{B}_{a,b}$ with the Laplace mechanism as well, but to choose a carefully selected privacy leakage $\varepsilon' < \varepsilon$ so that algorithm $\mathcal{B}_{a,b}$ is (1) ε-BDP and (2) $(h\alpha, \beta)$-accurate.

First, we determine the accuracy of algorithm $\mathcal{M}_{\varepsilon, f_{a,b}}$. With Corollary 2.2, we know that the (α, β)-accuracy of the Laplace mechanism for a given probability $\beta \in (0, 1]$ and privacy parameter ε is

$$\alpha = \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon} = \ln\left(\frac{1}{\beta}\right) \cdot \frac{b-a}{\varepsilon}. \tag{189}$$

So this is the (α, β)-accuracy of $\mathcal{M}_{\varepsilon, f_{a,b}}$.

We have to show that there exists an ε-BDP algorithm $\mathcal{B}_{a,b}$ which is $(h\alpha, \beta)$-accurate. We choose $\mathcal{B}_{a,b}$ as the Laplace mechanism applied to $f_{a,b}$ with an adjusted privacy parameter $\varepsilon' > 0$. Thus, $\mathcal{B}_{a,b}$ will be ε'-DP. Observe that $\mathcal{B}_{a,b}$ is d-private with $d(D, D') = \frac{\varepsilon'}{b-a} ||D - D'||_1$. Therefore, we can use Proposition 6.4 to show that $\mathcal{B}_{a,b}$ is BDP. We must choose ε' in a way that ensures that the BDPL is limited to ε, so that we have ε-BDP. With Proposition 6.4, $\mathcal{B}_{a,b}$ is

$$(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1)\varepsilon'\text{-BDP}. \tag{190}$$

Therefore, to achieve ε-BDP, we must have

$$(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1)\varepsilon' = \varepsilon \quad \Leftrightarrow \quad \varepsilon' = \varepsilon \cdot (\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1)^{-1}. \tag{191}$$

Now, we can calculate the accuracy of $\mathcal{B}_{a,b}$ because it also uses the Laplace mechanism. Then, we find an upper bound for this accuracy. Algorithm $\mathcal{B}_{a,b}$ is (α', β)-accurate, with

$$\begin{align}
\alpha' &= \ln(\frac{1}{\beta}) \cdot \frac{\Delta f}{\varepsilon'} = \ln(\frac{1}{\beta}) \cdot \frac{b-a}{\varepsilon} \cdot (\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1) \tag{192}\\
&\leq \ln(\frac{1}{\beta}) \cdot \frac{b-a}{\varepsilon} \cdot (\frac{n^2}{4(\frac{n^2}{4(h-1)} + n - 2 - n + 2)} + 1) \tag{193}\\
&= \ln(\frac{1}{\beta}) \cdot \frac{b-a}{\varepsilon} \cdot (\frac{n^2}{\frac{n^2}{(h-1)}} + 1) = \ln(\frac{1}{\beta}) \cdot \frac{b-a}{\varepsilon} \cdot (h - 1 + 1) \tag{194}\\
&= \ln(\frac{1}{\beta}) \cdot \frac{b-a}{\varepsilon} \cdot h = h\alpha. \tag{195}
\end{align}$$

In eq. (193) we use the upper bound of ρ from eq. (187). Finally, we find that $h\alpha \geq \alpha'$. Since $\mathcal{B}_{a,b}$ is (α', β)-accurate, it is therefore also $(h\alpha, \beta)$-accurate. □

6.6 Discussion

This section has shown that reasonable utility guarantees for BDP sum queries are possible if correlation in the multivariate Gaussian distribution is very small. More concretely, the accuracy of a BDP sum queries only deteriorates with a fixed multiplicative factor compared to DP sum queries if the Pearson correlation coefficient ρ is in the order $\rho \approx \frac{1}{n^2}$. Additionally, the Gaussian bound on the BDPL for Gaussian distributed data improves on the general bound when the Pearson correlation coefficient is smaller than $\rho \approx \frac{1}{n}$. Thus, the results of this section enable the privacy protection of weakly correlated data drawn from a multivariate Gaussian distribution, while retaining utility.

The aforementioned proofs cannot offer an improvement when the correlation exceeds $\rho \approx \frac{1}{n}$. In these cases, one would have to fall back to the general bound. However, our simulations indicate that neither the Gaussian bound, nor the general bound, are tight. Thus, we hypothesize that higher utility BDP algorithms for data with a multivariate Gaussian correlation model are possible. Also, it begs the question which steps in the proofs have not given an exact upper bound. They are of interest for future research to further improve the bound on the BDPL. The identified steps concern Theorem 6.2.

- In eq. (156), the entries of the inverse matrix $\Sigma_{K,n}^{-1}$ are bounded by the maximal eigenvalue of the matrix. This is not necessarily tight, especially because it is applied to an entire column of the matrix in eq. (162). It may be possible to find a better bound for the sum of the entries of a column in a matrix.

- The minimum eigenvalue of a symmetric matrix is bounded by the lower bound of the Gershgorin discs on the real number line in eq. (157). This is not tight when the smallest eigenvalue lies inside such a disc, rather than on its border.

- The numerator and denominator are bounded individually in eq. (165). It is possible to obtain a tighter bound by forming the derivative of the entire function with respect to the number of known records k. However, we have performed a preliminary investigation and the result suggests that the bound is not substantially improved.

7 Markov Chain Correlation Model

The states of a system in subsequent time steps are often highly correlated. The state of the current time step depends on the state in the previous time step. E.g., the location of a user at time step i is highly correlated with their location at time step $i-1$. This phenomenon is modeled by Markov chains [3].

In this section, we study the impact of time-series correlation on the privacy leakage and utility of BDP algorithms. The main results are Theorem 7.3 and Corollary 7.4: They provide a new bound on the BDPL of any ε-DP algorithm when data follows a Markov chain. We call it the *Markov chain bound*. For certain Markov chains, this result improves on the general bound and the bound from the workshop paper by Zhao et al. [57] (see Section 7.2). In Corollary 7.5, we show that BDP algorithms with only moderate losses in accuracy compared to their DP counter part are possible when the difference between large and small transition probabilities in the Markov chain is limited. Finally, we discuss in Section 7.5 how this result may further be improved in the future, because simulations once again suggest that the bound is not tight (see Figure 15).

We use the definition of a Markov chain by Behrends [3]. Strictly speaking, it refers to finite, time-homogeneous Markov chains, i.e., Markov chains with finite state spaces whose transition probabilities remain constant for every time step i.

Definition 7.1 (Markov Chain [3])**.** Let $s \in \mathbb{N}$ be the number of states and let $\mathbf{X} = (X_1, \ldots, X_n)^\top$ be a random vector. We say $\mathbf{X}$ is a *Markov chain* with transition matrix $P \in \mathbb{R}^{s \times s}$ and initial distribution $w \in \mathbb{R}^s$ if all of the following holds.

1. For all states $x, y \in [s]$ and all indices $i \in [n-1]$ we have $\Pr[X_{i+1} = x | X_i = y] = P_{y,x}$.
2. For all states $x \in [s]$ we have $\Pr[X_1 = x] = w_x$.
3. The Markov property: The probability distribution of the next state, given the current state, is conditionally independent of all other previous states. More precisely, for all indices $i \in [n-1]$ and for all states $x_1, \ldots, x_i, x_{i+1} \in [s]$ we have

$$\Pr[X_{i+1} = x_{i+1} \mid X_1 = x_1, \ldots, X_i = x_i] = \Pr[X_{i+1} = x_{i+1} \mid X_i = x_i]. \tag{196}$$

7.1 Relationship between DP and BDP

This subsection will prove the new Markov chain bound for the BDPL of any ε-DP algorithm whose data follows a Markov chain. The most important result comes in Theorem 7.3. It shows that the change in the conditional probability of the unknown records for condition $X_i = x_i$ and condition $X_i = x_i'$ can be bounded. This bound depends on the maximum difference between the highest and smallest transition probability in the Markov chain. This makes intuitive sense – if all transition probabilities are close together, changing random variable X_i from state x_i to state x_i' will not have much of an effect on the remaining time steps of the Markov chain. However, if the transition probabilities are far apart, it could have a large effect for many time steps. Subsequently, we use this result in Corollary 7.4 to bound the BDPL of a DP algorithm on data following a Markov chain – the Markov chain bound.

Before we can prove Theorem 7.3, we first require Lemma 7.1 and Lemma 7.2. The first lemma allows us to remove redundant conditions from probabilities, which will be useful in many later proofs of this section. The second lemma provides bounds on many probabilities that appear while proving Theorem 7.3.

Lemma 7.1. Let random vector $\mathbf{X}$ be a Markov chain. Let $i, j \in [n]$ be indices and $I \subseteq [n] \setminus \{i, j\}$ be a set of indices so that there exists an index $l \in I$ that is "in between" i and j, i.e., we have $i > l > j$ or $i < l < j$. Then, for all states $x_i, x_j \in [s]$ and for all state tuples $\mathbf{x}_I \in [s]^{|I|}$ we have

$$\Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I,\, X_j = x_j] = \Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I]. \tag{197}$$

Proof. In order to prove this lemma, we require that the Markov property from Definition 7.1 holds not only when the full history is known, but also when only the *partial* history is known. We will prove that this *generalized Markov property* follows directly from the Markov property before proving the statements of the lemma. Let index $h \in [n-1]$ be arbitrary, and let set $J \subseteq \{1, \dots, h-1\}$ only contain indices smaller than h. Let set $L = [h-1] \setminus J$ be the complement of J. Then, for any states $x, y \in [s]$ and any state tuple $\mathbf{x}_J \in [s]^{|J|}$ we have

$$\begin{aligned}
&\Pr[X_{h+1} = x \mid X_h = y, \mathbf{X}_J = \mathbf{x}_J] \\
&\quad = \sum_{\mathbf{x}_L \in [s]^{|L|}} \Pr[X_{h+1} = x | X_h = y, \mathbf{X}_J = \mathbf{x}_J, \mathbf{X}_L = \mathbf{x}_L] \cdot \Pr[\mathbf{X}_L = \mathbf{x}_L | X_h = y, \mathbf{X}_J = \mathbf{x}_J] && (198) \\
&\quad = \sum_{\mathbf{x}_L \in [s]^{|L|}} \Pr[X_{h+1} = x \mid X_h = y] \cdot \Pr[\mathbf{X}_L = \mathbf{x}_L \mid X_h = y, \mathbf{X}_J = \mathbf{x}_J] && (199) \\
&\quad = \Pr[X_{h+1} = x \mid X_h = y] \sum_{\mathbf{x}_L \in [s]^{|L|}} \Pr[\mathbf{X}_L = \mathbf{x}_L \mid X_h = y, \mathbf{X}_J = \mathbf{x}_J] && (200) \\
&\quad = \Pr[X_{h+1} = x \mid X_h = y] && (201)
\end{aligned}$$

We use the law of total probability to introduce all remaining indices L of the history in eq. (198). Then, in eq. (199) we apply the Markov property. Now, the first probability can be pulled out of the sum in eq. (200) because it no longer depends on $\mathbf{x}_L$. The remaining sum adds up to 1 and therefore we only retain the condition of the direct predecessor $X_h = y$ in eq. (201).

Now we use this generalized Markov property to prove the lemma. The lemma applies in two cases; either $i > l > j$ or $i < l < j$. We differentiate between these two cases and begin with the first to prove eq. (197).

Case 1: There exists an index $l \in I$ with $i > l > j$.

We choose index $l \in I$ with $i > l > j$ so that no other index in I lies between i and l. This means that the set of indices between l and i – defined as $M = \{l+1, \dots, i-1\}$ – is disjoint with I. If M is empty, then eq. (197) follows immediately from the generalized Markov property because index l is the direct predecessor of index i.

Otherwise, if $M \neq \emptyset$ is not empty, the idea is to use the law of total probability to also include the random vector $\mathbf{X}_M$ in the condition of the probability. Then, we can use the generalized Markov property and conditional independence to remove the condition on random variable X_j. To do this, we first only look at the conditional probability $\Pr[\mathbf{X}_M = \mathbf{x}_M \mid \mathbf{X}_I = \mathbf{x}_I, X_j = x_j]$ and show that the last condition is superfluous. Let $\mathbf{x}_M = (x_{l+1}, \dots, x_{i-1})^\top \in [s]^{|M|}$ be an arbitrary vector of states. We have

$$\begin{aligned}
&\Pr[\mathbf{X}_M = \mathbf{x}_M | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \\
&\quad = \Pr[X_{l+1} = x_{l+1}, \dots, X_{i-1} = x_{i-1} | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \qquad (202)\\
&\quad = \Pr[X_{l+1} = x_{l+1}, \dots, X_{i-2} = x_{i-2} | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \cdot \\
&\qquad \Pr[X_{i-1} = x_{i-1} | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j, X_{l+1} = x_{l+1}, \dots, X_{i-2} = x_{i-2}] \qquad (203)\\
&\quad = \dots \\
&\quad = \Pr[X_{l+1} = x_{l+1} | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \cdot \\
&\qquad \prod_{m=l+2}^{i-1} \Pr[X_m = x_m | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j, X_{l+1} = x_{l+1}, \dots, X_{m-1} = x_{m-1}] \qquad (204)\\
&\quad = \Pr[X_{l+1} = x_{l+1} | \mathbf{X}_I = \mathbf{x}_I] \cdot \prod_{m=l+2}^{i-1} \Pr[X_m = x_m | \mathbf{X}_I = \mathbf{x}_I, X_{l+1} = x_{l+1}, \dots, X_{m-1} = x_{m-1}] \\
&\qquad (205)\\
&\quad = \Pr[\mathbf{X}_M = \mathbf{x}_M | \mathbf{X}_I = \mathbf{x}_I]. \qquad (206)
\end{aligned}$$

First, we expand the components of random vector $\mathbf{X}_M$ in eq. (202). Then, in eq. (203), we rewrite the joint probability of $\mathbf{X}_M$ as the two parts X_{i-1} and $\mathbf{X}_{M \setminus \{i-1\}}$. This uses the fact that a joint probability can be rewritten as

$$\Pr[A \cap B \mid C] = \Pr[A \mid C] \cdot \Pr[B \mid C, A]. \tag{207}$$

Here, event A corresponds to conditions $X_{l+1} = x_{l+1}, \dots, X_{i-2} = x_{i-2}$, event B to condition $X_{i-1} = x_{i-1}$ and event C to conditions $\mathbf{X}_I = \mathbf{x}_I, X_j = x_j$. This step is repeatedly used to fully split $\mathbf{X}_M$ into its components and derive eq. (204). Then, we use the generalized Markov property from eq. (201) for eq. (205): Random variable X_{l+1} is conditionally independent of X_j given the direct predecessor X_l with index $l \in I$. Similarly, random variable X_m is conditionally independent of X_j, given the direct predecessor X_{m-1}. Now, the joint probability can be written together again, by repeatedly applying eq. (207) in reverse in eq. (206). We have shown that the condition $X_j = x_j$ can be removed without changing the probability $\Pr[\mathbf{X}_M = \mathbf{x}_M \mid \mathbf{X}_I = \mathbf{x}_I]$.

Now, we use the result from eq. (206) to directly prove eq. (197) of the lemma by applying the law of total probability. For any state $x_i \in [s]$ we have

$$\begin{aligned}
&\Pr[X_i = x_i | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \\
&\quad = \sum_{\mathbf{x}_M \in [s]^{|M|}} \Pr[X_i = x_i | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j, \mathbf{X}_M = \mathbf{x}_M] \cdot \Pr[\mathbf{X}_M = \mathbf{x}_M | \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] \\
&\qquad (208)\\
&\quad = \sum_{\mathbf{x}_M \in [s]^{|M|}} \Pr[X_i = x_i | \mathbf{X}_I = \mathbf{x}_I, \mathbf{X}_M = \mathbf{x}_M] \cdot \Pr[\mathbf{X}_M = \mathbf{x}_M | \mathbf{X}_I = \mathbf{x}_I] \qquad (209)
\end{aligned}$$

$$= \Pr[X_i = x_i | \mathbf{X}_I = \mathbf{x}_I]. \tag{210}$$

In eq. (209), we use the fact that X_i is conditionally independent of X_j given index $i-1 \in M$ (see generalized Markov property in eq. (201)). Also, we apply eq. (206) to the second probability. Then, we derive eq. (210) by once again using the law of total probability, this time in reverse. Now, eq. (197) from the lemma has been proven for the case $i > l > j$.

Case 2: There exists an index $l \in I$ with $i < l < j$.

Here, Equation (197) from the lemma follows from the first case we have just proven:

$$\Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I, X_j = x_j] = \frac{\Pr[X_j = x_j \mid \mathbf{X}_I = \mathbf{x}_I, X_i = x_i] \cdot \Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I]}{\Pr[X_j = x_j \mid \mathbf{X}_I = \mathbf{x}_I]} \tag{211}$$

$$= \frac{\Pr[X_j = x_j \mid \mathbf{X}_I = \mathbf{x}_I] \cdot \Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I]}{\Pr[X_j = x_j \mid \mathbf{X}_I = \mathbf{x}_I]} \tag{212}$$

$$= \Pr[X_i = x_i \mid \mathbf{X}_I = \mathbf{x}_I] \tag{213}$$

We use Bayes' theorem in eq. (211). The first probability of the numerator is now in the situation of Case 1, since it is the probability of random variable X_j, conditioned on $\mathbf{X}_I$ and X_i with $j > l > i$. Equation (197) has already been proven for that case, so we apply it to derive eq. (212). Finally, eq. (213) follows directly by simplifying and we have proven eq. (197) for the second possible case.

□

Now we come to the second lemma before we can prove Theorem 7.3. Lemma 7.2 will allow us to put a bound on probabilities or ratios of probabilities. In Theorem 7.3, this result will be essential to bound the difference of the conditional probabilities on all unknown samples of the adversary when conditioning on $X_i = x_i$ versus $X_i = x_i'$.

Lemma 7.2 applies to Markov chains in which all transition probabilities in P are greater than zero. Additionally, the initial distribution w must be an eigenvector of eigenvalue 1 of the transition matrix P. This distribution w may at first appear like a strong assumption, however it arises naturally: An eigenvector of P with eigenvalue 1 represents the *equilibrium distribution* of the Markov Chain, and such an eigenvector always exists if all entries of P are strictly positive [3, p. 50]. It is called the equilibrium distribution because the distribution of the states does not change as the Markov chain progresses from time-step to time-step.

Lemma 7.2. Let $\mathbf{X} = (X_1, \ldots, X_n)^\top$ be a Markov chain with transition probabilities $P \in \mathbb{R}^{s \times s}$ and initial distribution $w \in \mathbb{R}^s$ with the following properties:

- Every cell of P is positive, i.e., for all $x, y \in [s]$ we have $P_{x,y} > 0$.
- Vector w is an eigenvector of P to the eigenvalue 1, i.e., $w^\top P = w^\top$.

Then, for any index $i \in [n]$ and any state $x \in [s]$ we have

$$\min_{k,l \in [s]} P_{k,l} \leq \Pr[X_i = x] \leq \max_{k,l \in [s]} P_{k,l}. \tag{214}$$

Furthermore, for all indices $i, j \in [n]$ with $i > j$ and all states $x, x', y, y' \in [s]$ we have

$$\frac{\Pr[X_i = x \mid X_j = y]}{\Pr[X_i = x \mid X_j = y']} \leq \frac{\max_{k,l \in [s]} P_{k,l}}{\min_{k,l \in [s]} P_{k,l}} \tag{215}$$

and

$$\frac{\Pr[X_i = x \mid X_j = y]}{\Pr[X_i = x' \mid X_j = y]} \leq \frac{\max_{k,l \in [s]} P_{k,l}}{\min_{k,l \in [s]} P_{k,l}}. \tag{216}$$

Proof. We begin by proving eq. (214). First, we show that w – as an eigenvector of P – not only contains the prior probabilities of X_1, but of any random variable X_i for $i \in [n]$. I.e., the equality $\Pr[X_i = x] = w_x$ holds for any state $x \in [s]$ because w is the equilibrium distribution. Then we can bound the entries of w to prove eq. (214). We use induction to prove the equality $\Pr[X_i = x] = w_x$.

$$\begin{aligned} \Pr[X_i = x] &= \sum_{y \in [s]} \Pr[X_i = x \mid X_{i-1} = y] \cdot \Pr[X_{i-1} = y] && (217) \\ &= \sum_{y \in [s]} P_{y,x} w_y && (218) \\ &= (w^\top P)_x && (219) \\ &= w_x && (220) \end{aligned}$$

We apply the law of total probability in eq. (217). Then, we use the transition matrix P and the induction hypothesis ($\Pr[X_{i-1} = x] = w_x$) to replace the probabilities with entries of P and w in eq. (218). Then, we rewrite the sum as a matrix-vector product in eq. (219). Finally, we take advantage of the fact that w is an eigenvector of P in eq. (220). With the basis $\Pr[X_1 = x] = w_x$ (which is the definition of w, see Definition 7.1) and this derivation, we prove by induction that the entries of w are equal to the prior probabilities of any random variable X_i.

Now we can bound the entries of w, thereby also bounding the probabilities $\Pr[X_i = x]$. We prove the upper bound $w_x \leq \max_{k,l \in [s]} P_{k,l}$ by contradiction; the lower bound $w_x \geq \min_{k,l \in [s]} P_{k,l}$ follows analogously. Assume that there exists a state $y \in [s]$ so that its prior probability in w is greater than any transition probability, i.e., $w_y > \max_{k,l \in [s]} P_{k,l}$. This leads to a contradiction as follows.

$$\begin{aligned} w_y &= (w^\top P)_y && (221) \\ &= \sum_{k \in [s]} P_{k,y} \cdot w_k && (222) \\ \Leftrightarrow (1 - P_{y,y}) w_y &= \sum_{k \neq y} P_{k,y} \cdot w_k && (223) \\ \Leftrightarrow 1 - P_{y,y} &= \sum_{k \neq y} \frac{P_{k,y}}{w_y} \cdot w_k && (224) \\ &< \sum_{k \neq y} w_k && (225) \end{aligned}$$

$$
\begin{aligned}
&= 1 - w_y && (226)\\
\Leftrightarrow\ & 1 + w_y < 1 + P_{y,y} && (227)\\
\Leftrightarrow\ & w_y < P_{y,y} && (228)
\end{aligned}
$$

First, we use that w is an eigenvector in eq. (221) and subsequently rewrite the matrix-vector multiplication. In eq. (225), we apply the assumption that w_y is greater than any transition probability, so $P_{k,y}/w_y$ must be strictly smaller than one. Equation (226) follows because the entries of w must sum to one as w is a probability distribution (see Definition 7.1). Finally, we arrive at the statement $w_y < P_{y,y}$ which is contradictory to our assumption that w_y is bigger than every $P_{k,y}$. Thus, this assumption was false and probability w_y must be smaller or equal to $\max_{k,l\in[s]} P_{kl}$ for any state $y \in [s]$. Together with the lower bound that can be shown analogously, this proves the first statement of the lemma, Equation (214).

Now, we prove Equation (215). Let indices $i, j \in [n]$ be arbitrary with $i > j$ and let states $x, y, y' \in [s]$ be arbitrary. If random variables X_i and X_j are direct neighbors, i.e., $i = j + 1$, then the probabilities are transition probabilities from matrix P and the bound follows immediately with

$$
\frac{\Pr[X_i = x \mid X_j = y]}{\Pr[X_i = x \mid X_j = y']} = \frac{P_{y,x}}{P_{y',x}} \leq \frac{\max_{k,l\in[s]} P_{k,l}}{\min_{k,l\in[s]} P_{k,l}}. \tag{229}
$$

In the other case (i.e., $i > j + 1$), we once more use the law of total probability to make this tractable. We focus on rewriting the numerator $\Pr[X_i = x \mid X_j = y]$ in terms of the denominator. Let states $x, y, y' \in [s]$ be arbitrary. Then we have

$$
\begin{aligned}
&\Pr[X_i = x \mid X_j = y]\\
&= \sum_{z\in[s]} \Pr[X_i = x \mid X_j = y, X_{j+1} = z] \cdot \Pr[X_{j+1} = z \mid X_j = y] && (230)\\
&= \sum_{z\in[s]} \Pr[X_i = x \mid X_{j+1} = z] \cdot \Pr[X_{j+1} = z \mid X_j = y] && (231)\\
&= \sum_{z\in[s]} \Pr[X_i = x \mid X_j = y', X_{j+1} = z] \cdot \Pr[X_{j+1} = z \mid X_j = y'] \cdot \frac{\Pr[X_{j+1} = z \mid X_j = y]}{\Pr[X_{j+1} = z \mid X_j = y']} && (232)\\
&= \sum_{z\in[s]} \Pr[X_i = x \mid X_j = y', X_{j+1} = z] \cdot \Pr[X_{j+1} = z \mid X_j = y'] \cdot \frac{P_{y,z}}{P_{y',z}} && (233)\\
&= \sum_{z\in[s]} \Pr[X_i = x \mid X_j = y', X_{j+1} = z] \cdot \Pr[X_{j+1} = z \mid X_j = y'] \cdot \frac{\max_{k,l\in[s]} P_{k,l}}{\min_{k,l\in[s]} P_{k,l}} && (234)\\
&\leq \Pr[X_i = x \mid X_j = y'] \cdot \frac{\max_{k,l\in[s]} P_{k,l}}{\min_{k,l\in[s]} P_{k,l}}. && (235)
\end{aligned}
$$

We use Lemma 7.1 to remove $X_j = x_j$ from the condition of the first probability in eq. (231) because $j + 1$ is closer to i. Similarly, we add the condition $X_j = y'$ in eq. (232)

by applying Lemma 7.1 in reverse and adding an independent condition. Here, we also multiply by one $(\Pr[X_{j+1} = z \mid X_j = y']/\Pr[X_{j+1} = z \mid X_j = y'])$ which introduces $\Pr[X_{j+1} = z \mid X_j = y']$ into the equation. The probabilities in the ratio are replaced by the corresponding cells in P in eq. (233) because they are transition probabilities. Then, the ratio is bounded in eq. (234). Finally, we can apply the law of total probability a second time in eq. (235), but this time conditioned on $X_j = x'_j$. If one divides the last equation by $\Pr[X_i = x_i \mid X_j = x'_j]$ (which by assumption is greater than zero), one obtains Equation (215), which we set out to prove.

Finally, we prove the third and last equation from the lemma, Equation (216), in a similar fashion to the one before. Let states $x, x', y \in [s]$ be arbitrary. If random variables X_i and X_j are direct neighbors (i.e., $j = i - 1$), the ratio can once again be bounded straightforwardly as it only contains probabilities from P:

$$\frac{\Pr[X_i = x \mid X_{i-1} = y]}{\Pr[X_i = x' \mid X_{i-1} = y]} = \frac{P_{y,x}}{P_{y,x'}} \leq \frac{\max_{k,l \in [s]} P_{k,l}}{\min_{k,l \in [s]} P_{k,l}} \tag{236}$$

Otherwise for $j < i - 1$ we once more use the law of total probability to rewrite the probability in the numerator from $X_i = x_i$ to $X_i = x'_i$.

$$\begin{aligned}
&\Pr[X_i = x | X_j = y] \\
&= \sum_{z \in [s]} \Pr[X_i = x | X_j = y, X_{i-1} = z] \cdot \Pr[X_{i-1} = z | X_j = y] && (237)\\
&= \sum_{z \in [s]} \Pr[X_i = x | X_{i-1} = z] \cdot \Pr[X_{i-1} = z | X_j = y] && (238)\\
&= \sum_{z \in [s]} \Pr[X_i = x' | X_{i-1} = z] \cdot \frac{\Pr[X_i = x | X_{i-1} = z]}{\Pr[X_i = x' | X_{i-1} = z]} \cdot \Pr[X_{i-1} = z | X_j = y] && (239)\\
&= \sum_{z \in [s]} \Pr[X_i = x' | X_{i-1} = z] \cdot \Pr[X_{i-1} = z | X_j = y] \cdot \frac{P_{z,x}}{P_{z,x'}} && (240)\\
&\leq \sum_{z \in [s]} \Pr[X_i = x' | X_{i-1} = z] \cdot \Pr[X_{i-1} = z | X_j = y] \cdot \frac{\max_{kl} P_{k,l}}{\min_{kl} P_{k,l}} && (241)\\
&= \sum_{z \in [s]} \Pr[X_i = x' | X_j = y, X_{i-1} = z] \cdot \Pr[X_{i-1} = z | X_j = y] \cdot \frac{\max_{kl} P_{k,l}}{\min_{kl} P_{k,l}} && (242)\\
&= \Pr[X_i = x' | X_j = y] \cdot \frac{\max_{kl} P_{k,l}}{\min_{kl} P_{k,l}} && (243)
\end{aligned}$$

Lemma 7.1 is used to drop $X_j = y$ from the condition in eq. (238). Then, we multiply by one $(\Pr[X_i = x' \mid X_{i-1} = z]/\Pr[X_i = x' \mid X_{i-1} = z])$ to replace $\Pr[X_i = x \mid X_{i-1} = z]$ with $\Pr[X_i = x' \mid X_{i-1} = z]$ in eq. (239). The ratio only contains transition probabilities from P, and thus can be bounded as in eq. (241). We reintroduce the condition $X_j = y$ in eq. (242) using Lemma 7.1 to make it possible to apply the law of total probability in reverse in eq. (243). If we divide the last equation by $\Pr[X_i = x' \mid X_j = y]$ (which by assumption is greater than zero), we obtain Equation (216). □

With Lemma 7.1 and Lemma 7.2 in hand, we are ready to prove Theorem 7.3. It bounds the difference of the conditional probabilities of all unknown samples of the adversary when conditioning on $X_i = x_i$ versus when conditioning on $X_i = x_i'$. Subsequently, this result is applied in Corollary 7.4 to bound the BDPL of any ε-DP algorithm $\mathcal{A}$. The conditional probability that is treated in Theorem 7.3 appears while expanding Definition 2.11 of BDPL using the law of total probability.

Theorem 7.3. Let random vector $\mathbf{X} = (X_1, \dots, X_n)^\top$ be a Markov chain with transition probabilities $P \in \mathbb{R}^{s \times s}$ and initial distribution $w \in \mathbb{R}^s$ with the following properties:

- Every cell of P is positive, i.e., for all $k, l \in [s]$ we have $P_{k,l} > 0$.
- Vector w is an eigenvector of P to the eigenvalue 1, i.e., $w^\top P = w^\top$.

Let $i \in [n]$ be the target index and let sets $U \subseteq [n] \setminus \{i\}$ and $K \subseteq [n] \setminus \{i\}$ be disjoint, with $[n] = U \cup K \cup \{i\}$ and at least one index in $|U| \geq 1$. Then, for any unknown states $\mathbf{x}_U \in [s]^{|U|}$, known states $\mathbf{x}_K \in [s]^{|K|}$ and target states $x_i, x_i' \in [s]$ we have

$$\frac{\Pr[\mathbf{X}_U = \mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[\mathbf{X}_U = \mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} \right)^4 . \tag{244}$$

Proof. The main part of this proof is a case analysis distinguishing between different positions of i in $[n]$ and different sets K, and bounding the ratio of Equation (244) in each case. When deriving these bounds, we will often utilize Lemma 7.2 to bound probabilities without any condition, or with the value of one random variable as a condition. However, we also encounter probabilities that still contain two conditions. To bound these probabilities, we first have to show that a probability with two conditions can be rewritten into two probabilities with one condition each (given we are dealing with a Markov chain). Let indices $h, i, j \in [n]$ be arbitrary with $h < i < j$. Let states $x_h, x_i, x_j \in [s]$ be arbitrary.

$$\Pr[X_i = x_i \mid X_h = x_h, X_j = x_j] = \frac{\Pr[X_j = x_j \mid X_h = x_h, X_i = x_i] \cdot \Pr[X_i = x_i \mid X_h = x_h]}{\Pr[X_j = x_j \mid X_h = x_h]} \tag{245}$$

$$= \frac{\Pr[X_j = x_j \mid X_i = x_i] \cdot \Pr[X_i = x_i \mid X_h = x_h]}{\Pr[X_j = x_j \mid X_h = x_h]} \tag{246}$$

First, we apply Bayes' theorem in eq. (245). Then, we use Lemma 7.1 in eq. (246) to remove condition $X_h = x_h$ because index i is closer to j than index h is to j. Now, the ratio of two such probabilities with two conditions can be expressed and bounded as follows for any state $x_i' \in [s]$:

$$\frac{\Pr[X_i = x_i | X_h = x_h, X_j = x_j]}{\Pr[X_i = x_i' | X_h = x_h, X_j = x_j]}$$

$$= \frac{\Pr[X_j = x_j | X_i = x_i] \cdot \Pr[X_i = x_i | X_h = x_h]}{\Pr[X_j = x_j | X_h = x_h]} \cdot \frac{\Pr[X_j = x_j | X_h = x_h]}{\Pr[X_j = x_j | X_i = x_i'] \cdot \Pr[X_i = x_i' | X_h = x_h]} \tag{247}$$

$$= \frac{\Pr[X_j = x_j | X_i = x_i] \cdot \Pr[X_i = x_i | X_h = x_h]}{\Pr[X_j = x_j | X_i = x_i'] \cdot \Pr[X_i = x_i' | X_h = x_h]} \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} \right)^2 \tag{248}$$

In eq. (247), we use eq. (246) to rewrite both the numerator and the denominator. Then, we shorten the ratio by $\Pr[X_j = x_j \mid X_h = x_h]$ in eq. (248). Finally, Lemma 7.2 can be applied twice to derive the bound.

Now, we can begin with the main part of the proof: The case analysis for different target indices $i \in [n]$ and sets $K \subseteq [n] \setminus \{i\}$. A similar pattern emerges in each case: We always repeatedly apply Lemma 7.1 to only retain those conditions that are absolutely necessary (i.e., only the indices closest to i stay in the condition, all others are conditionally independent). Then, we bound all factors with Lemma 7.2 or Equation (248).

- **Case 1:** The set $K \neq \emptyset$ contains at least one index.

  We apply Bayes' theorem to rewrite the main probability that appears in Equation (244). Let states $\mathbf{x}_U \in [s]^{|U|}$ and $\mathbf{x}_K \in [s]^{|K|}$ and $x_i \in [s]$ be arbitrary:

  $$\begin{aligned}&\Pr[\mathbf{X}_U = \mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] \\ &\qquad = \frac{\Pr[X_i = x_i \mid \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U \mid \mathbf{X}_K = \mathbf{x}_K]}{\Pr[X_i = x_i \mid \mathbf{X}_K = \mathbf{x}_K]} \end{aligned} \tag{249}$$

  We use this result to show the following equivalence for the left hand side of Equation (244) for any state $x_i' \in [s]$:

  $$\frac{\Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} = \frac{\Pr[X_i = x_i | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[X_i = x_i' | \mathbf{X}_K = \mathbf{x}_K]}{\Pr[X_i = x_i' | \mathbf{X}_K = \mathbf{x}_K, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[X_i = x_i | \mathbf{X}_K = \mathbf{x}_K]} \tag{250}$$

  This ratio must now be bounded by $(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}})^4$ in order to prove Equation (244). We will accomplish this in each subcase of Case 1.

  - **Case 1.1:** Target index $i = 1$ is the first index.

    Let k_1 be the smallest index in K. This index $k_1 > i$ must be bigger than $i = 1$ because i is the first index and i can by assumption not be included in K. Equation (250) is equivalent to

    $$= \frac{\Pr[X_1 = x_1 \mid X_2 = x_2] \cdot \Pr[X_1 = x_1' \mid X_{k_1} = x_{k_1}]}{\Pr[X_1 = x_1' \mid X_2 = x_2] \cdot \Pr[X_1 = x_1 \mid X_{k_1} = x_{k_1}]} \tag{251}$$

    $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^2 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4. \tag{252}$$

    In eq. (251) we use Lemma 7.1 to remove any condition except for the one closest to i. For the first probability, the closest index to $i = 1$ in the condition must be index 2 because U and K make up the rest of indices $[n] \setminus \{1\}$. For the second probability, the closest index to i must be k_1. Subsequently, we bound the ratio by applying Lemma 7.2 twice in eq. (252). Observe that the ratio $\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}$ must at least be 1, so increasing the exponent to 4 can also only increase the upper bound.

  - **Case 1.2:** Index $i \in [n]$ is arbitrary with $1 < i < n$. We must further differentiate how the indices in set K are located around i.

- - **Case 1.2.1:** The set $K \subseteq [n] \setminus \{i\}$ only includes indices smaller than i. I.e., there exists an index $k_1 \in K$ with $k_1 < i$ so that for all indices $k \in K$ we have $k \leq k_1$.

    Then, Equation (250) is equivalent to

    $$= \frac{\Pr[X_i = x_i \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i' \mid X_{k_1} = x_{k_1}]}{\Pr[X_i = x_i' \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i \mid X_{k_1} = x_{k_1}]} \tag{253}$$

    $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^3 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4. \tag{254}$$

    We use Lemma 7.1 in eq. (253) to remove any redundant conditions. Then, we employ the bound from eq. (248) and Lemma 7.2 in eq. (254) and once again increase the exponent to 4, which corresponds to the final result we wish to prove among all cases.

  - **Case 1.2.2:** The set $K \subseteq [n] \setminus \{i\}$ includes both indices smaller and larger than index i. I.e., there exist indices $k_1, k_2 \in K$ with $k_1 < i$ and $k_2 > i$ so that for all indices $k \in K$ we have $k \leq k_1$ or $k \geq k_2$.

    Then, Equation (250) is equivalent to

    $$= \frac{\Pr[X_i = x_i | X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i' | X_{k_1} = x_{k_1}, X_{k_2} = x_{k_2}]}{\Pr[X_i = x_i' | X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i | X_{k_1} = x_{k_1}, X_{k_2} = x_{k_2}]} \tag{255}$$

    $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4. \tag{256}$$

    We use Lemma 7.1 to remove redundant conditions and subsequently employ Equation (248) twice to find the upper bound.

  - **Case 1.2.3:** The set K only include indices bigger than i. I.e., there exists an index $k_2 \in K$ with $k_2 > i$ so that for all indices $k \in K$ we have $k \geq k_2$.

    Then, Equation (250) is equivalent to

    $$= \frac{\Pr[X_i = x_i \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i' \mid X_{k_2} = x_{k_2}]}{\Pr[X_i = x_i' \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i \mid X_{k_2} = x_{k_2}]} \tag{257}$$

    $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^3 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4. \tag{258}$$

- **Case 1.3:** Target index $i = n$ is the last index.

  Let k_+ be the largest index in K. This index k_+ must be strictly smaller than index i because $i = n$ is the last index and i cannot be included in the set of known indices K. Then, Equation (250) is equivalent to

$$= \frac{\Pr[X_n = x_n \mid X_{n-1} = x_{n-1}] \cdot \Pr[X_n = x'_n \mid X_{k_+} = x_{k+}]}{\Pr[X_n = x'_n \mid X_{n-1} = x_{n-1}] \cdot \Pr[X_n = x_n \mid X_{k_+} = x_{k+}]} \tag{259}$$

$$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^2 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \tag{260}$$

- **Case 2:** The set $K = \emptyset$ is empty.

  The set of unknown indices $U = [n] \setminus \{i\}$ must include every index except for i. We apply Bayes' theorem to once more rewrite the main probability that appears in Equation (244). Let states $\mathbf{x}_U \in [s]^{n-1}$ and $x_i \in [s]$ be arbitrary:

  $$\Pr[\mathbf{X}_U = \mathbf{x}_U \mid X_i = x_i] = \frac{\Pr[X_i = x_i \mid \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U]}{\Pr[X_i = x_i]} \tag{261}$$

  We use this result to show the following equivalence for the left hand side of Equation (244) for any state $x'_i \in [s]$:

  $$\frac{\Pr[\mathbf{X}_U = \mathbf{x}_U \mid X_i = x_i]}{\Pr[\mathbf{X}_U = \mathbf{x}_U \mid X_i = x'_i]} = \frac{\Pr[X_i = x_i \mid \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[X_i = x'_i]}{\Pr[X_i = x'_i \mid \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[X_i = x_i]} \tag{262}$$

  Just like in Case 1, this ratio must be bounded by $(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}})^4$ in order to prove Equation (244). We will accomplish this in each of the following subcases of Case 2.

  - **Case 2.1:** Target index $i = 1$ is the first index. Equation (262) is equal to

  $$= \frac{\Pr[X_1 = x_1 \mid X_2 = x_2] \cdot \Pr[X_1 = x'_1]}{\Pr[X_1 = x'_1 \mid X_2 = x_2] \cdot \Pr[X_1 = x_1]} \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^2 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 . \tag{263}$$

  - **Case 2.2:** Target index $i \in [n]$ is arbitrary with $1 < i < n$. Then, Equation (262) is equivalent to

  $$= \frac{\Pr[X_i = x_i \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x'_i]}{\Pr[X_i = x'_i \mid X_{i-1} = x_{i-1}, X_{i+1} = x_{i+1}] \cdot \Pr[X_i = x_i]} \tag{264}$$

  $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^3 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 . \tag{265}$$

  - **Case 2.3:** Target index $i = n$ is the last index. Equation (262) is equivalent to

  $$= \frac{\Pr[X_n = x_n \mid X_{n-1} = x_{n-1}] \cdot \Pr[X_n = x'_n]}{\Pr[X_n = x'_n \mid X_{n-1} = x_{n-1}] \cdot \Pr[X_n = x_n]} \tag{266}$$

  $$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^2 \leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 . \tag{267}$$

The case analysis is complete and all cases have been bounded by $\left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4$. □

With Theorem 7.3, it becomes possible to directly prove a bound on the BDPL of any ε-DP algorithm that processes data following a Markov chain. Corollary 7.4 applies to Markov chains with strictly positive transition probabilities that are initialized with their equilibrium distribution w. It is the Markov chain bound.

Corollary 7.4 (The Markov Chain Bound)**.** Let $s \in \mathbb{N}$ be the number of states. Let $\mathcal{A} : [s]^n \to \mathcal{Y}$ be an ε-DP algorithm. Let the data tuples follow a Markov chain with transition matrix $P \in \mathbb{R}^{s \times s}$ and initial distribution $w \in \mathbb{R}^s$ with the following properties:

- Every cell of P is positive, i.e., for all $k, l \in [s]$ we have $P_{k,l} > 0$.
- Vector w is an eigenvector of P to the eigenvalue 1, i.e., $w^\top P = w^\top$.

Then, $\mathcal{A}$ is an $(\varepsilon + 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}})$-BDP algorithm.

Proof. In order to show that $\mathcal{A}$ is a BDP algorithm, we have to upper bound the BDPL. The BDPL is the supremum over the adversary-specific BDPLs. So we have to consider every possible adversary (K, i) with target index $i \in [n]$ and known indices $K \subseteq [n] \setminus \{i\}$, and bound $BDPL_{(K,i)}$. Let set $U = [n] \setminus (K \cup \{i\})$ be the indices that are unknown to the adversary. We differentiate between two cases because Theorem 7.3 can only be applied when there is at least one unknown index, i.e., $U \neq \emptyset$.

Case 1: There are no unknown indices $U = \emptyset$.

While Theorem 7.3 cannot be applied in this case, it is fortunately easy to reduce the BDPL of this adversary to the DP privacy leakage: In this case, the adversary knows every index $K = [n] \setminus \{i\}$ except the target index and the adversary-specific $BDPL_{(K,i)}$ becomes the same as the DP privacy leakage [55, p. 751]. Thus, we have

$$BDPL_{(K,i)} = \varepsilon \leq \varepsilon + 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}. \tag{268}$$

Case 2: There is at least one unknown index in $U \neq \emptyset$ for the adversary.

Here, we will be able to apply Theorem 7.3. Let random variable Y represent the output of $\mathcal{A}$, let set $S \subseteq \mathcal{Y}$ be measurable and let states $\mathbf{x}_K \in [s]^{|K|}$ and $x_i, x_i' \in [s]$ be arbitrary. First, we use the law of total probability to rewrite the probability of any result of $\mathcal{A}$ given $X_i = x_i$. Subsequently, we employ the fact that algorithm $\mathcal{A}$ is DP, as well as the result from Theorem 7.3, to bound the difference between this probability and the probability where we condition on the changed target variable $X_i = x_i'$. Finally, we will use this bound to limit the BDPL.

$$\begin{aligned}
&\Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] \\
&= \sum_{\mathbf{x}_U \in [s]^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i] &(269)\\
&= \sum_{\mathbf{x}_U \in [s]^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i'] \cdot \\
&\quad \frac{\Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} &(270)
\end{aligned}$$

$$\leq \sum_{\mathbf{x}_U \in [s]^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i'] \cdot \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \tag{271}$$

$$= \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \sum_{\mathbf{x}_U \in [s]^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i, \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i'] \tag{272}$$

$$\leq \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \cdot e^{\varepsilon} \sum_{\mathbf{x}_U \in [s]^{|U|}} \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i', \mathbf{X}_U = \mathbf{x}_U] \cdot \Pr[\mathbf{X}_U = \mathbf{x}_U | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i'] \tag{273}$$

$$= \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \cdot e^{\varepsilon} \cdot \Pr[Y \in S | \mathbf{X}_K = \mathbf{x}_K, X_i = x_i'] \tag{274}$$

In eq. (270), we effectively multiply by 1 to introduce the condition $X_i = x_i'$ into the equation. The ratio can then be upper bounded in the following eq. (271) by employing Theorem 7.3. Additionally, we use that $\mathcal{A}$ is ε-DP in eq. (273). Finally, the law of total probability is applied in reverse in eq. (274).

Now, we use this result in the definition of $BDPL_{(K,i)}$, thereby bounding the BDPL of every possible adversary with at least one unknown index in U:

$$BDPL_{(K,i)} = \sup_{x_i, x_i', \mathbf{x}_K, S} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} \tag{275}$$

$$\leq \sup_{x_i, x_i', \mathbf{x}_K, S} \ln \frac{\left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \cdot e^{\varepsilon} \cdot \Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']} \tag{276}$$

$$= \sup_{x_i, x_i', \mathbf{x}_K, S} \ln \left(\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}\right)^4 \cdot e^{\varepsilon} = 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} + \varepsilon \tag{277}$$

We use the result from eq. (274) in eq. (276). The final bound follows with straight-forward simplifications.

Since we have bounded the $BDPL_{(K,i)}$ of every possible adversary (K, i), we also bound the total BDPL

$$BDPL = \sup_{K,i} BDPL_{(K,i)} \leq 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} + \varepsilon. \tag{278}$$

□

7.2 Comparison to State of the Art

The Markov chain bound from Corollary 7.4 must be compared to the existing bounds. These comprise of the general bound and the bound by Zhao et al. [57]. The result by Zhao et al. comes with the caveat that it has not been peer-reviewed as it was published in a workshop, and the proof is not available. However, we nonetheless include it in the comparison as it applies under the same conditions as our Markov chain bound.

First, a short overview of all three bounds is given, and subsequently they are compared. We will introduce parameters γ and ω to simplify the notation of the bounds comprising of transition probabilities.

1. All random variables $X_1, \ldots, X_n$ are correlated, so the general bound becomes

$$BDPL \leq n\varepsilon. \tag{279}$$

   The Markov property from Definition 7.1 may lead one to believe that not all random variables are correlated and the general bound should be lower. However, the Markov property only implies *conditional* independence given the direct predecessor state, not complete independence.

2. The bound by Zhao et al. [57, p. 7] applies when all transition probabilities $P_{x,y}$ are greater than zero. It is

$$BDPL \leq \varepsilon + 6 \ln \max_{x,x',y \in [s]} \frac{P_{x,y}}{P_{x',y}} =: \varepsilon + 6 \ln \omega. \tag{280}$$

   Here, the ratio ω is the maximum divided by the minimum transition probability *that lead to the same state $y \in [s]$*. The logarithm of ω is multiplied by 6.

3. In Corollary 7.4, we have proven the Markov chain bound that also requires positive transition probabilities. This bound is

$$BDPL \leq \varepsilon + 4 \ln \frac{\max_{x,y} P_{x,y}}{\min_{x,y} P_{x,y}} =: \varepsilon + 4 \ln \gamma. \tag{281}$$

   In contrast to Zhao's bound, γ is the maximum ratio of any two transition probabilities. Observe that for any specific Markov chain, we must have $\gamma \geq \omega$. However, in the Markov chain bound, the logarithm of γ is only multiplied by 4.

We begin by comparing the Markov chain bound to the previous result for Markov chains by Zhao et al. [57].

Remark. For any privacy leakage $\varepsilon > 0$, any $\omega \geq 1$, and any $\gamma \geq \omega$ we have

$$\varepsilon + 4 \ln \gamma \leq \varepsilon + 6 \ln \omega \ \Leftrightarrow\ \ln \gamma \leq \frac{3}{2} \ln \omega \ \Leftrightarrow\ \gamma \leq \omega^{\frac{3}{2}}. \tag{282}$$

As we can see, it depends on the relationship between γ and ω whether Zhao's bound or the Markov chain bound is smaller. This is exemplified by Figure 12, which shows two possible Markov chains. Zhao's bound is smaller for the left Markov chain, while the Markov chain bound becomes smaller for right one. Figure 11 also shows the situation in more quantitative terms. For values of γ and ω in the dotted region, the Markov chain bound is smaller. E.g., for $\omega = 10$, the ratio γ can be up to three times larger and the Markov chain bound is still an improvement. For $\omega = 100$, ratio γ can be up to $\gamma = 1\,000$ for an improvement.

Now, we also compare the Markov chain bound to the general bound $n\varepsilon$.

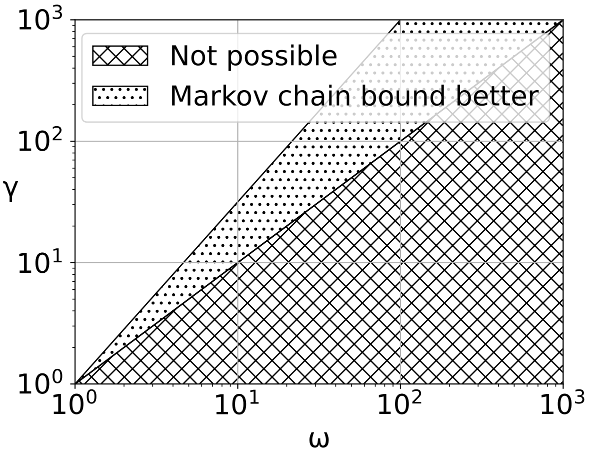


Figure 11: Comparison of Markov chain bound to Zhao's bound. The x-axis shows the values of ratio ω while the y-axis shows ratio γ. Both axes are logarithmic. Our Markov chain bound improves upon Zhao's bound for values of γ and ω in the dotted region. Values in the grid are not possible, because we must have $\gamma \geq \omega$.

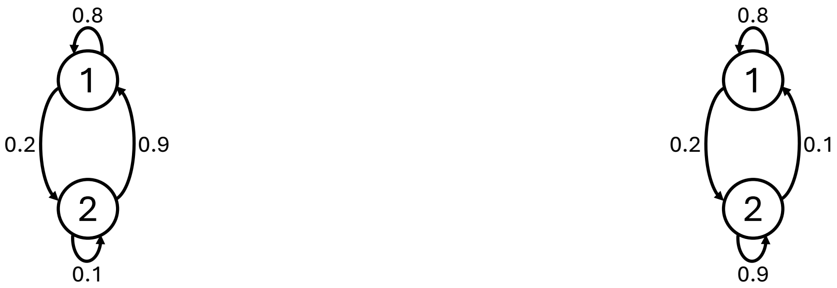


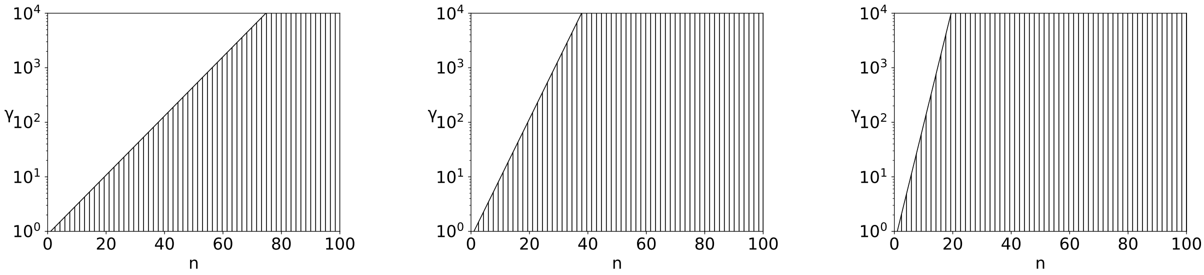
Figure 12: Two examples of a Markov chain. Each transition probability is drawn next to the corresponding edge. On the left we see a Markov chain with $\gamma = 9$ and $\omega = 2$. The bound by Zhao et al. is smaller. On the right we see a Markov chain with $\gamma = 9$ and $\omega = 8$. Our Markov chain bound is smaller.

Remark. For number of records $n \in \mathbb{N}$, privacy parameter $\varepsilon > 0$, and maximum transition probability ratio $\gamma \geq 1$ we have

$$\varepsilon + 4 \ln \gamma < n\varepsilon \;\Leftrightarrow\; 4 \ln \gamma < (n-1)\varepsilon \tag{283}$$

$$\Leftrightarrow\; \ln \gamma < \frac{n-1}{4}\varepsilon \;\Leftrightarrow\; \gamma < \exp\left(\frac{n-1}{4}\varepsilon\right) \tag{284}$$

Figure 13 shows that the Markov chain bound improves upon the general bound in many situations. E.g., for an ε-DP algorithm with $\varepsilon = 0.5$ and data tuple of size $n = 80$, the Markov chain bound is an improvement even if the largest transition probability is 10 000 times larger than the smallest. However, while we often see an improvement, this is not always the case. E.g., for $\varepsilon = 0.5$, $n = 20$ and transition probability ratio γ of 100, the general bound would be smaller. This – along with the comparison to the bound by Zhao et al. – shows that the Markov chain bound is not always a tight bound.



(a) $\varepsilon = 0.5$ (b) $\varepsilon = 1$ (c) $\varepsilon = 2$

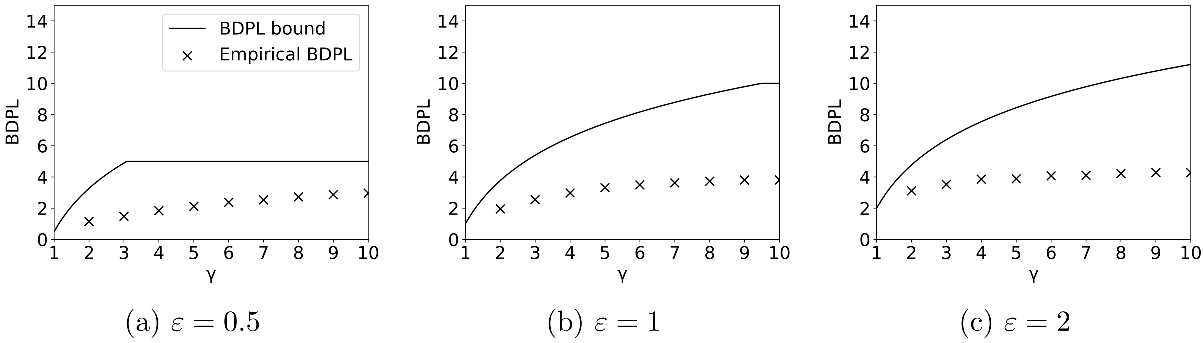
Figure 13: Comparison of Markov chain bound to general bound. The x-axis represents the size of the data tuple, while the logarithmic y-axis shows the ratio γ. The Markov chain bound of the BDPL for an ε-DP algorithm improves upon the general bound for values of n and γ in the shaded area.

7.3 Empirical BDPL

The Markov chain bound is an improvement in many situations. However, it is not tight since it does exceed either the general bound and/or Zhao's bound in certain other situations (see Figures 11 and 13). Thus, this raises the question how tight the Markov chain bound is compared to the maximum possible BDPL in reality. To this end, we run Monte Carlo simulations to estimate the maximum BDPL.

7.3.1 Simulation

First, we discuss the methodology of the simulation to empirically estimate the BDPL. We estimate the BDPL for DP privacy leakages $\varepsilon \in \{0.5, 1, 2\}$, data tuples of sizes $n = 10$ and for transition probability ratios $\gamma \in \{1, \ldots, 10\}$. To estimate the maximum possible BDPL for each γ, we use the Monte Carlo method as described in Section 3.1.2 for each choice of γ and ε. We estimate the BDPL of the specific Markov chain shown in Figure 14.



(a) $\varepsilon = 0.5$

(b) $\varepsilon = 1$

(c) $\varepsilon = 2$

Figure 15: Empirical BDPL compared to upper bound for Markov chain data. The x-axis shows the ratio γ between the largest and smallest transition probability within the Markov chain. The y-axis is the BDPL. The solid line shows the upper bound for the BDPL; the minimum of the Markov chain bound and the general bound. The empirical BDPL of an ε-DP counting query is marked by the crosses. The data has size $n = 10$.

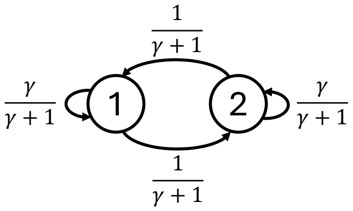


Figure 14: Markov chain used in the simulations. For any $\gamma \geq 1$, this Markov chain has maximum transition probability ratio γ.

It would also be possible to generate different random Markov chains, however preliminary simulations showed that this resulted in a lower estimation of the BDPL. The Markov chain in Figure 14 appears to maximize the BDPL for a given maximum ratio γ.

7.3.2 Results

The results of the simulation can be seen in Figure 15. They are compared to the minimum of the Markov chain bound and the general bound $n\varepsilon$. The empirical BDPL does increase with a higher transition probability ratio γ, however the results still suggest that the given bound is not tight. For higher values of ε, the increase in γ appears to have less of an effect on the empirical BDPL.

7.4 Accuracy

While the Markov chain bound is not tight, it still enables us to give improved utility guarantees for the Laplace mechanism, when the ratio γ is sufficiently small.

Corollary 7.5. Let $h > 1$ be a real number, let $n \in \mathbb{N}$ be the size of the data tuple, and let $\varepsilon > 0$ be real. Let $f : [s]^n \to \mathbb{R}$ be a deterministic query with finite sensitivity $\Delta f < \infty$. Let $\mathcal{M}_{\varepsilon,f}$ be the Laplace mechanism. Let the data tuples follow a Markov chain with initial distribution $w \in \mathbb{R}^s$ and transition probabilities $P \in \mathbb{R}^{s \times s}$ with positive entries $P_{k,l} > 0$ for any $k, l \in [s]$, eigenvector $w^\top P = w^\top$ and

$$\frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} \leq \exp\left(\frac{h-1}{4h}\varepsilon\right). \tag{285}$$

Then, there exists an ε-BDP algorithm $\mathcal{B}$ with the following property: If $\mathcal{M}_{\varepsilon,f}$ is (α, β)-accurate with respect to f, then $\mathcal{B}$ is $(h\alpha, \beta)$-accurate with respect to f.

Proof. By applying Corollary 2.2, we know that $\mathcal{M}_{\varepsilon,f}$ is (α, β)-accurate with respect to f with $\beta \in (0, 1]$ and $\alpha = \ln(\frac{1}{\beta}) \cdot \frac{\Delta f}{\varepsilon}$ because $\mathcal{M}_{\varepsilon,f}$ is the Laplace mechanism.

We have to show that there exists an ε-BDP algorithm that is $(h\alpha, \beta)$-accurate with respect to f. We accomplish this by also applying the Laplace mechanism to f – however with a smaller DP privacy leakage ε' – so that the resulting ε'-DP algorithm $\mathcal{B}$ is ε-BDP according to Corollary 7.4.

We choose the DP privacy leakage ε' as follows, and ensure that it is greater than zero.

$$\begin{align}
\varepsilon' &:= \varepsilon - 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}} \tag{286}\\
&\geq \varepsilon - 4 \ln \exp\left(\frac{h-1}{4h}\varepsilon\right) = \varepsilon - 4\left(\frac{h-1}{4h}\varepsilon\right) = \varepsilon - \left(1 - \frac{1}{h}\right)\varepsilon \tag{287}\\
&= \frac{1}{h}\varepsilon \tag{288}\\
&> 0 \tag{289}
\end{align}$$

Per Corollary 7.4, algorithm $\mathcal{B}$ is $(\varepsilon' + 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}})$-BDP, i.e., ε-BDP.

We can calculate the accuracy of $\mathcal{B}$ because it is also using the Laplace mechanism. Algorithm $\mathcal{B}$ is (α', β)-accurate, with

$$\begin{align}
\alpha' = \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon'} &= \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon - 4 \ln \frac{\max_{kl} P_{kl}}{\min_{kl} P_{kl}}} \tag{290}\\
&\leq \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\frac{1}{h}\varepsilon} \tag{291}\\
&= \ln\left(\frac{1}{\beta}\right) \cdot \frac{\Delta f}{\varepsilon} \cdot h = h\alpha. \tag{292}
\end{align}$$

In eq. (291) we apply eq. (288).

Since algorithm $\mathcal{B}$ is (α', β)-accurate, it is also $(h\alpha, \beta)$-accurate because $\alpha' \leq h\alpha$.

□

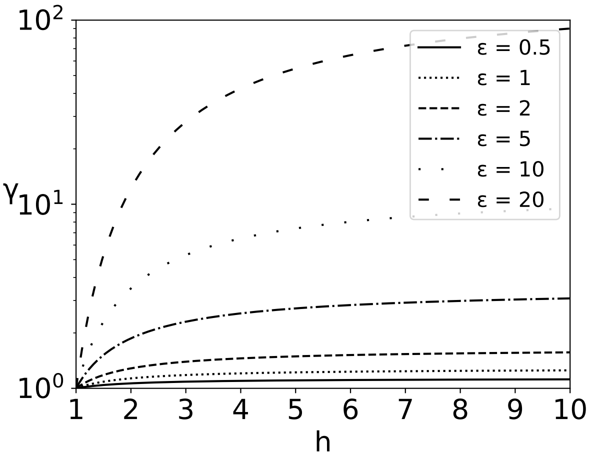


Figure 16: Relative accuracy of an ε-BDP algorithm to an ε-DP algorithm for Markov chain data. The x-axis shows different factors h while the logarithmic y-axis shows the transition probability ratio γ. Each line indicates the maximum transition probability ratio γ of a Markov chain so that the error of an ε-BDP algorithm $\mathcal{B}$ is at most h times greater than the error of an ε-DP algorithm $\mathcal{M}_{\varepsilon,f}$.

The statement of Corollary 7.5 is visualized in Figure 16. Let algorithm $\mathcal{M}_{\varepsilon,f}$ be ε-DP with the Laplace mechanism, and let $\mathcal{M}_{\varepsilon,f}$ be (α, β)-accurate with respect to f. Then Figure 16 shows the maximum transition probability ratio γ for which we know exists an ε-BDP algorithm $\mathcal{B}$ with $(h\alpha, \beta)$-accuracy. The figure shows that in order to provide similar utility guarantees to DP, either the BDPL bound ε has to be larger than 5, or the ratio γ between different transition probabilities must be smaller than 3.

7.5 Discussion

In this section, we have derived the Markov chain bound on the BDPL of any ε-DP algorithm on data that follows a Markov chain. We have shown that it improves upon previous results by Zhao et al. and the general bound under certain conditions. The Markov chain bound has allowed us to provide new utility guarantees in Corollary 7.5 for algorithms utilizing the Laplace mechanism. However, when one wishes to give reasonable privacy guarantees with BDP (i.e., $\varepsilon \leq 5$), then the utility improvements are limited to situations in which the largest transition probability is at most three times larger than the smallest. This is visualized for different BDP privacy leakages ε in Figure 16.

However, empirical simulations suggest that the Markov chain bound is not tight, and better utility guarantees may be possible if one can further refine the upper bound on the BDPL of DP algorithms processing data that follows a Markov chain. While it is out of the scope of this thesis, we nonetheless identify the following parts in our proofs that may be viable for improvement in the future.

1. In Theorem 7.3, we have bounded the difference in the conditional probability when changing from condition $X_i = x_i$ to $X_i = x'_i$. The bound is γ^4. The only equation in our case analysis that required the bound γ^4 – in place of γ^3 or γ^2 like the other

cases – is eq. (256). Finding a tighter bound for eq. (256) may be possible because it actually only consists of two probability ratios, not four.

2. In Corollary 7.4, we bound the BDPL by applying the bound from Theorem 7.3 to any conditional probability for the unknown random variables (see eq. (271)). Even if Theorem 7.3 is tight for some instantiations $\mathbf{x}_U$ of the unknown random variables $\mathbf{X}_U$, it will not be tight for all of them. Here, an improved corollary could use different (tighter) bounds for different classes of instantiations $\mathbf{x}_U$. However, we suspect that this would require making more assumptions about the algorithm $\mathcal{A}$, as the probability distribution of the output Y of algorithm $\mathcal{A}$ determines how these multiple bounds would be weighted in the final result.

8 Utility Experiment

In this section, we present our utility experiment using real-world data sets. The general idea is to test the utility of BDP algorithms that are created by taking DP algorithms and using the applicable bounds to calculate their maximum BDPL (i.e., the general bound from Theorem 4.2, the Gaussian bound from Proposition 6.4, the Markov chain bound from Corollary 7.4, or the bound by Zhao et al. [57]). We use data sets that correspond to genomic data, a multivariate Gaussian correlation model, or a Markov chain, respectively.

8.1 Objective

In general, our objective is to empirically investigate the utility improvements by our bounds over the state of the art bounds in real-world situations. Here, the state of the art is the general bound. While we were first to prove the general bound for BDP, it had already been common practice to multiply the DP privacy leakage by the number m of correlated records to estimate the maximum privacy loss due to correlation attacks [10]. For Markov chain data, the state of the art is extended by the bound by Zhao et al. [57].

For genomic data, we have shown that the general bound is nearly tight. Thus, we cannot investigate an improvement over the general bound in this situation. Here, our objective is to instead investigate the effect of parameter m on the utility of BDP algorithms created with the general bound.

For data with a multivariate Gaussian correlation model, we have found the Gaussian bound which improves over the general bound when the Pearson correlation coefficient ρ is limited by $\approx \frac{1}{n}$. Therefore, the Gaussian bound offers an improvement either when the data set is small, or the correlation between all records is weak. Specifically, the BDPL of an ε-DP and d-private algorithm is

$$BDPL \leq \left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1 \right) \varepsilon. \tag{293}$$

In the utility experiment, we investigate the utility improvement over the general bound for a real world height data set.

For Markov chain data, we have also found a new bound, the Markov chain bound. The BDPL of any ε-DP algorithm is bounded by

$$BDPL \leq \varepsilon + 4 \ln \max_{x,x',y,y'} \frac{P_{y,x}}{P_{y',x'}}. \tag{294}$$

This bound must also be compared to the bound by Zhao et al.:

$$BDPL \leq \varepsilon + 6 \ln \max_{x,y,y'} \frac{P_{y,x}}{P_{y',x}}. \tag{295}$$

We found the new Markov chain bound improves on the general bound and Zhao's bound for certain Markov chains in Section 7.2. We compare the effect of the three different bounds on utility for real-world activity data.

This section also gives us the opportunity to compare the benefits and drawbacks of our three BDPL bounds between each other. We have identified three important differences:

- The Markov chain bound is a sum of ε and a fixed ratio, while the other bounds are multiplicative in ε. This can be a benefit when working with Markov chain data because larger DP privacy leakages ε do not increase the BDPL as strongly. However, it can also be a drawback for smaller DP privacy leakages. Here, the general bound $m\varepsilon$ can become arbitrarily small for a fixed m by using a sufficiently small DP privacy leakage ε. This is similar with the Gaussian bound. In contrast, the minimum Markov chain bound is $BDPL \leq 4 \ln \max_{x,x',y,y'} \frac{P_{y,x}}{P_{y',x'}}$, even when DP privacy leakage ε is set to zero. We will see this disadvantage exemplified in the utility experiment.

- Another notable difference is the dependence on the size of the data set n. The Gaussian bound and the general bound both increase with the size of the data set, while the Markov chain bound stays constant. This is of course an advantage of the Markov chain bound. We test the effect of this advantage on utility by experimenting with Markov chain data of different sizes n and comparing the performance of the general bound and the Markov chain bound.

- Finally, we also compare how restrictive the assumptions of the bounds are. The general bound is applicable to any data distribution and any DP algorithm. The other two bounds make assumptions about the general distribution and its specific parameters. In detail, the Gaussian bound requires a multivariate Gaussian distribution with equal variances and small Pearson correlation coefficients. It also requires from the algorithm that it is both DP and d-private with the ℓ_1-distance. The Markov chain bound requires a Markov chain starting in its equilibrium distribution and with no transition probabilities that are zero. In contrast to the Gaussian bound, it applies to any DP algorithm. Overall, the general bound clearly has an advantage in this category. We will also see in the experiments that it can be applied regardless of the type of data.

8.2 Data Sets and Queries

Here, we present the data sets and the queries we use in our experiment. We also explain which correlation model fits to each data set. We work with three types of data:

- **1000Genome phase 3 data:** This data set contains the alleles of 2 504 persons for over 88 million positions in the genetic code. It originates from a major study on human genetic variation [2]. In our experiment, we use the first 1 000 recorded allele pairs of chromosome 22. Thus, we have 1 000 data sets, each with 2 504 records.

  Here, the data comes from a Genomic distribution. The authors state that "individuals and available first-degree relatives (generally, adult offspring) were genotyped" [2, p. 1], but it is not known *how many* of the participants are directly related to each other, i.e., the exact pedigree graph G is not given. In Section 8.4.1, we show results for different assumptions about G.

On this data, we execute a counting query that counts the occurrences of the allele pair homozygous-major, i.e., BB.

- **Galton height data:** A historical data set of heights of individuals [16]. Each of the 897 records contains the height of a person, their gender, an identifier for their family, both their parents heights, and the number of children in their family. The data is collected from 205 different families. For our experiment, each of the 897 records is turned into a single data set of size three, containing the heights of the parents and the height of the child. We end up with a total of 897 data sets, each containing three records.

  We assume that the data sets are drawn from a multivariate Gaussian distribution with three correlated records, because height closely follows a Gaussian distribution [7] and because the height of a child is correlated with the heights of their parents [33]. In a multivariate Gaussian distribution, all correlation between variables is linear [43, p. 29], and linear models have been used in the literature to model the inheritance of height [33].

  We execute a sum query on each data set. The records are clipped to the range of 60 to 80 inches (about 1.5 to 2 meters).

- **Activity data:** Each record in this data set contains the number of steps an individual has taken in a five minute interval [35]. It is a time-series of consecutive intervals from October to November 2012. We prepare the data by replacing each record with 0 steps by state “inactive”, and each one with more than 0 steps by state “active”. We model this data set as a Markov chain with the two states active and inactive. For our experiment, we prepare two versions of this data set:

  - *Data for single days:* We split the data set into many data sets, each corresponding to the activity states of a single day. We end up with 61 unique data sets, each with $n = 288$ records.
  - *Complete data:* The data of all days is combined into a single data set with $n = 17\,568$ records.

  On these data sets, we execute a counting query for the number of active states.

Overall, we end up with the following groups of data sets:

1. A total of 1 000 genomic data sets, each containing 2 504 records.
2. A collection of 897 height data sets, each with three records.
3. A total of 61 activity data sets, each comprising 288 records.
4. One activity data set with 17 568 records.

8.3 Experiment

The experiment measures utility for BDP algorithms with maximum BDPL $\varepsilon \in (0, 20]$. This includes the range of privacy leakages $\varepsilon \in (0, 5)$ that enable sound theoretical privacy

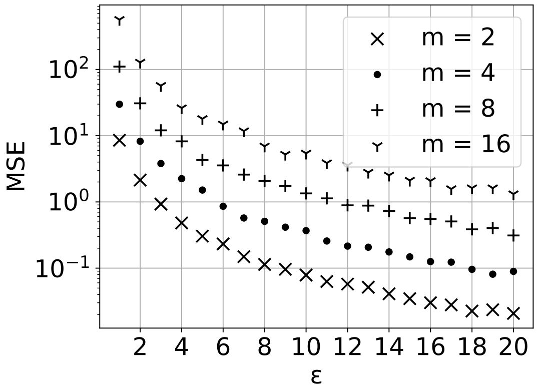


Figure 17: MSE of BDP counting query on genomic data for different maximum family sizes m. The x-axis shows the privacy leakage ε of the BDP query. The y-axis shows the MSE compared to the deterministic counting query.

guarantees, and the higher range of $\varepsilon \in [5, 20)]$ that empirically show resilience to certain privacy attacks. For each group of data sets, we identify the BDPL bounds that apply in this case. For each applicable BDPL bound, we use the Laplace mechanism to create an algorithm that fulfills ε-BDP according to the BDPL bound. In our experiments, we quantify utility by measuring the *mean squared error* (MSE) of a BDP algorithm to the deterministic ground-truth query. The error is averaged over all data sets in a group. If the group only contains one data set, we execute the BDP algorithm 1 000 times on the same data set and take the MSE of all of these results.

8.4 Results

Here, we present the results of the experiment for each type of data. To put the size of the MSE into perspective, we also mention the average result of the deterministic query for each group of data sets. In this section, we refer to the DP privacy leakage by τ, to avoid confusion with the maximum BDPL ε.

8.4.1 Genomic Data

For genomic data, we have shown that the general bound of BDPL is nearly tight. Thus, we use the general bound along with the Laplace mechanism to create a BDP counting query for different assumptions about the maximum number of correlated records m. Here, m corresponds to the size of the largest family in the data. For these data sets, the deterministic counting query has an average output of 40.29 occurrences of allele pair BB. The utility results for the 1 000 genomic data sets are seen in Figure 17. The MSE increases when the number of correlated records increases.

8.4.2 Height Data

As explained in Section 8.2, we assume that the height data is drawn from a multivariate Gaussian distribution. We require the maximum Pearson correlation coefficient ρ for the

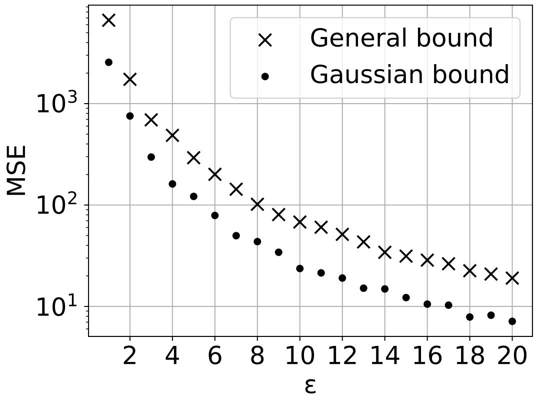


Figure 18: MSE of BDP sum query on height data with $n = 3$. The x-axis shows the privacy leakage ε of the BDP query. The y-axis shows the MSE compared to the deterministic sum query on a logarithmic scale.

Gaussian bound. The empirical Pearson correlation coefficient in our height data sets is 0.275 between child and father, 0.202 between child and mother, and 0.074 between father and mother. Thus, we have a maximum Pearson correlation coefficient of $\rho = 0.275$. For $n = 3$, Proposition 6.4 tells us that the Laplace mechanism $\mathcal{A}_{\tau,f}$ applied to a sum query f is ε-BDP, with

$$\varepsilon = \left(\frac{n^2}{4(\frac{1}{\rho} - n + 2)} + 1\right)\tau = \left(\frac{9}{4(\frac{1}{0.275} - 3 + 2)} + 1\right)\tau \approx 1.853\tau. \tag{296}$$

In comparison, the general bound tells us that we have $\varepsilon = m\tau = 3\tau$ for algorithm $\mathcal{A}_{\tau,f}$.

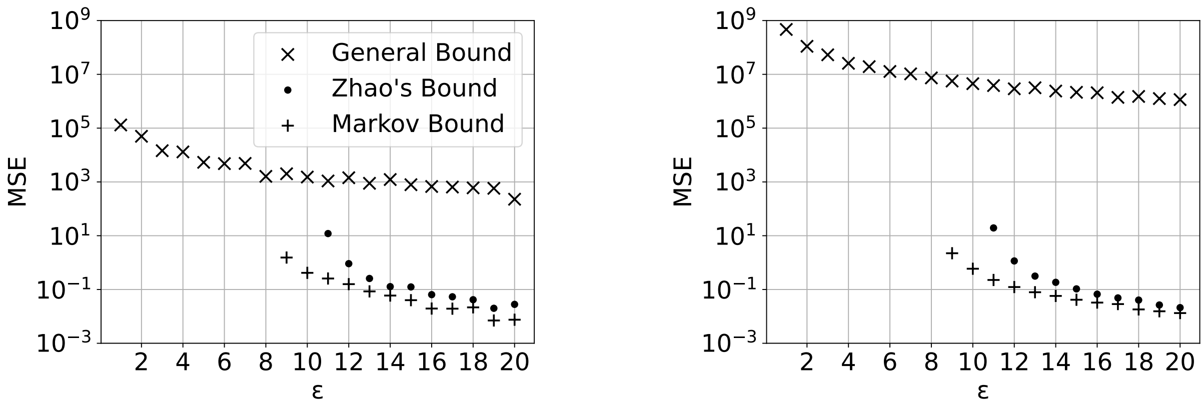
The deterministic sum query has an average output of $200 = 2 \cdot 10^2$ inches for each family. The results for the Gaussian bound and the general bound for the 897 height data sets are seen in Figure 18. We see that the algorithm using the Gaussian bound performs better.

8.4.3 Activity Data

Here, we assume that data comes from a Markov chain. Thus, we can apply the general bound, our Markov chain bound, or the bound by Zhao et al. For the latter two, we require the transition probabilities of the Markov chain. We calculate them empirically from the data set containing all 17 568 records and receive

$$\Pr[X_{i+1} = \text{inactive} \mid X_i = \text{inactive}] = 0.882 \tag{297}$$
$$\Pr[X_{i+1} = \text{active} \mid X_i = \text{inactive}] = 0.118 \tag{298}$$
$$\Pr[X_{i+1} = \text{inactive} \mid X_i = \text{active}] = 0.305 \tag{299}$$
$$\Pr[X_{i+1} = \text{active} \mid X_i = \text{active}] = 0.695. \tag{300}$$

Thus, with the general bound the maximum BDPL of a τ-DP algorithm becomes $\varepsilon = n\tau$. Our Markov chain bound gives $\varepsilon = \tau + 4\ln\frac{0.882}{0.118} = \tau + 8.05$ and the bound by Zhao et al. is $\varepsilon = \tau + 6\ln\frac{0.695}{0.118} = \tau + 10.64$.



(a) Single day data sets of size $n = 288$.

(b) Complete data set of size $n = 17\,568$.

Figure 19: MSE of BDP counting query on activity data. The x-axis shows the privacy leakage ε of the BDP query. The y-axis shows the MSE compared to the deterministic sum query.

The utility results for the 61 daily activity data sets with $n = 288$ each are seen in Figure 19a. Here, the average deterministic output is $80 = 8 \cdot 10^1$ occurrences of the active state. We see that our Markov chain bound only provides results for $\varepsilon \geq 9$ and the bound by Zhao et al. only for $\varepsilon \geq 11$. The reason for this becomes clear when looking at the bounds in this situation. Since the DP privacy leakage $\tau > 0$ must be greater than zero, our Markov chain bound for the BDPL is at least 8.05 and Zhao's bound is at least 10.64. In contrast, the general bound can provide an ε-BDP algorithm for any $\varepsilon > 0$. However, it performs worse than the Markov chain bound and Zhao's bound for $\varepsilon \geq 9$ and $\varepsilon \geq 11$ respectively. Our Markov chain bound outperforms Zhao's bound for any privacy leakage in this situation, and it outperforms the general bound for $\varepsilon \geq 9$.

The results for the complete activity data with $n = 17\,568$ records are seen in Figure 19b. The deterministic result is $4250 \approx 4 \cdot 10^3$ occurrences of the active state. The results look similar to Figure 19a, however the error of the algorithm using the general bound has increased sharply while algorithms with the Markov chain bound or Zhao's bound remain stable. This is because the general bound scales with the size of the data set n, while the other two bounds are independent of n.

8.4.4 Discussion

The results show that our newly proven Gaussian bound and Markov chain bound can improve on the utility of state of the art bounds, on real-world data when BDP algorithms are created by using DP algorithms. Additionally, we have seen that the utility of the general bound is very sensitive to the number of correlated records m.

9 Conclusion

In this thesis, we have investigated the utility of Bayesian differential privacy (BDP) for correlated data. We focused on Bayesian differential privacy because it applies to any correlation model and we found that many other semantic privacy notions for correlated data are variations of BDP. Our investigation provides both formal and empirical results.

We have investigated the relationship between DP and BDP. Although similar results existed for other semantic privacy notions, we are first to prove that ε-DP implies $m\varepsilon$-BDP where m is the number of correlated records. This result is tight if arbitrary correlation within the data is possible. We also improved on this result when restricting the correlation model to a multivariate Gaussian distribution with Pearson correlation coefficient below $\rho \approx \frac{1}{n}$, or for Markov chain time-series when the difference between transition probabilities is limited. Simulations of DP algorithms under these conditions have indicated that these two results can be refined further. We have suggested multiple possible locations in our proofs where more analysis may lead to an improved bound in the future.

The proofs on the relationship between DP and BDP enabled us to derive (α, β)-accuracy bounds. We have found that algorithms fulfilling BDP can offer high utility in the following situations: (a) When the number of correlated records m in the data tuple of size $n > m$ is sufficiently small. How small m must be depends on the specific task to be solved, and the subjective error-tolerance of the data analyst. (b) When the data has a multivariate Gaussian correlation model and the strength of correlation is very small, i.e., the Pearson correlation coefficient can be bounded to $\rho \approx \frac{1}{n^2}$. In this case, all records may be correlated and utility guarantees similar to DP algorithms are still possible for sum queries. (c) When the data is a time-series of subsequent states following a Markov chain and the transition probabilities are sufficiently similar. Again, in this situation all n records are correlated.

We have also found situations in which it was possible to prove that the utility of *any* ε-BDP algorithm must be poor, assuming that we give decent privacy guarantees $\varepsilon \in (0, 5)$: (a) When we make no assumptions about the correlation model and up to all n records may be correlated. (b) When performing a counting query on genomic data where all persons in the underlying data may be part of the same core family.

Additionally, we have tested the utility of BDP algorithms on real-world data. The BDP algorithms were created from DP algorithms using the proven relationship between DP and BDP in the different situations. This experiment has shown that our new Gaussian bound and Markov chain bound can improve utility over the state of the art in practical applications. Additionally, we have seen that the utility of algorithms created with the general bound is very sensitive to the number of correlated records m.

Overall, this thesis contributes new knowledge in which situations BDP algorithms can perform with high utility, and in which they cannot. This helps practitioners decide when limiting the BDPL is a viable method to protect against privacy attacks that exploit correlation between records. We also improve the understanding of the relationship between the privacy notions DP and BDP. This enables the use of existing DP algorithms as BDP algorithms. We intend to publish the results in an international conference to be decided.

References

[1] N. Almadhoun, E. Ayday, and Ö. Ulusoy. “Differential privacy under dependent tuples—the case of genomic privacy”. In: *Bioinformatics* 36.6 (Nov. 2019). Ed. by J. Hancock, pp. 1696–1703. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btz837.

[2] A. Auton and G. R. Abecasis. “A global reference for human genetic variation”. In: *Nature* 526.7571 (Sept. 2015), pp. 68–74. ISSN: 1476-4687. DOI: 10.1038/nature15393.

[3] E. Behrends. *Introduction to Markov Chains*. Vieweg+Teubner Verlag, 2000. ISBN: 9783322901576. DOI: 10.1007/978-3-322-90157-6.

[4] A. Bernacchia and S. Pigolotti. “Self-Consistent Method for Density Estimation”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 73.3 (Apr. 2011), pp. 407–422. ISSN: 1467-9868. DOI: 10.1111/j.1467-9868.2011.00772.x.

[5] A. Blum, K. Ligett, and A. Roth. “A learning theory approach to noninteractive database privacy”. In: *Journal of the ACM* 60.2 (Apr. 2013), pp. 1–25. ISSN: 1557-735X. DOI: 10.1145/2450142.2450148.

[6] G. Box. “Robustness in the Strategy of Scientific Model Building”. In: *Robustness in Statistics*. Elsevier, 1979, pp. 201–236. ISBN: 9780124381506. DOI: 10.1016/b978-0-12-438150-6.50018-2.

[7] J. Brainard and D. E. Burmaster. “Bivariate Distributions for Height and Weight of Men and Women in the United States”. In: *Risk Analysis* 12.2 (June 1992), pp. 267–275. ISSN: 1539-6924. DOI: 10.1111/j.1539-6924.1992.tb00674.x.

[8] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramèr. “Membership Inference Attacks From First Principles”. In: *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2022. DOI: 10.1109/sp46214.2022.9833649.

[9] K. Chatzikokolakis, M. E. Andrés, N. E. Bordenabe, and C. Palamidessi. “Broadening the Scope of Differential Privacy Using Metrics”. In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 82–102. ISBN: 9783642390777. DOI: 10.1007/978-3-642-39077-7_5.

[10] R. Chen, B. C. M. Fung, P. S. Yu, and B. C. Desai. “Correlated network data publication via differential privacy”. In: *The VLDB Journal* 23.4 (Nov. 2013), pp. 653–676. ISSN: 0949-877X. DOI: 10.1007/s00778-013-0344-8.

[11] K. M. Chong and A. Malip. “May the privacy be with us: Correlated differential privacy in location data for ITS”. In: *Computer Networks* 241 (Mar. 2024), p. 110214. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2024.110214.

[12] D. Desfontaines and B. Pejó. “SoK: Differential privacies”. In: *Proceedings on Privacy Enhancing Technologies* 2020.2 (Apr. 2020), pp. 288–313. ISSN: 2299-0984. DOI: 10.2478/popets-2020-0028.

[13] C. Dwork and M. Naor. “On the Difficulties of Disclosure Prevention in Statistical Databases or The Case for Differential Privacy”. In: *Journal of Privacy and Confidentiality* 2.1 (Sept. 2010). ISSN: 2575-8527. DOI: 10.29012/jpc.v2i1.585.

[14] C. Dwork, M. Naor, T. Pitassi, and G. N. Rothblum. “Differential privacy under continual observation”. In: *Proceedings of the forty-second ACM symposium on Theory of computing*. STOC’10. ACM, June 2010. DOI: 10.1145/1806689.1806787.

[15] C. Dwork and A. Roth. "The Algorithmic Foundations of Differential Privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (2013), pp. 211–407. ISSN: 1551-3068. DOI: 10.1561/0400000042.

[16] F. Galton. *Galton height data.* 2017. DOI: 10.7910/DVN/T0HSJ1.

[17] S. A. Gershgorin. "Über die Abgrenzung der Eigenwerte einer Matrix". In: *Izvestija Rossijskoj akademii nauk. Serija matematičeskaja* 6 (1931), pp. 749–754.

[18] D. Gordon, S. Heath, and J. Ott. "True Pedigree Errors More Frequent Than Apparent Errors for Single Nucleotide Polymorphisms". In: *Human Heredity* 49.2 (1999), pp. 65–70. ISSN: 1423-0062. DOI: 10.1159/000022846.

[19] T. Hawkins. "Cauchy and the spectral theory of matrices". In: *Historia Mathematica* 2.1 (Feb. 1975), pp. 1–29. ISSN: 0315-0860. DOI: 10.1016/0315-0860(75)90032-4.

[20] M. Hay, C. Li, G. Miklau, and D. Jensen. "Accurate Estimation of the Degree Distribution of Private Networks". In: *2009 Ninth IEEE International Conference on Data Mining.* IEEE, Dec. 2009. DOI: 10.1109/icdm.2009.11.

[21] X. He, A. Machanavajjhala, and B. Ding. "Blowfish privacy: tuning privacy-utility trade-offs using policies". In: SIGMOD/PODS'14 (June 2014). DOI: 10.1145/2588555.2588581.

[22] T. Humphries, S. Oya, L. Tulloch, M. Rafuse, I. Goldberg, U. Hengartner, and F. Kerschbaum. "Investigating Membership Inference Attacks under Data Dependencies". In: *2023 IEEE 36th Computer Security Foundations Symposium (CSF).* IEEE, July 2023. DOI: 10.1109/csf57540.2023.00013.

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

[24] S. P. Kasiviswanathan and A. Smith. "On the "Semantics" of Differential Privacy: A Bayesian Formulation". In: *Journal of Privacy and Confidentiality* 6.1 (June 2014). ISSN: 2575-8527. DOI: 10.29012/jpc.v6i1.634.

[25] D. Kifer, J. M. Abowd, R. Ashmead, R. Cumings-Menon, P. Leclerc, A. Machanavajjhala, W. Sexton, and P. Zhuravlev. "Bayesian and frequentist semantics for common variations of differential privacy: Applications to the 2020 census". In: *arXiv preprint arXiv:2209.03310* (2022).

[26] D. Kifer and A. Machanavajjhala. "No Free Lunch in Data Privacy". In: *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data.* SIGMOD '11. Athens, Greece: Association for Computing Machinery, 2011, pp. 193–204. DOI: 10.1145/1989323.1989345.

[27] S. Kotz, T. J. Kozubowski, and K. Podgórski. *The Laplace Distribution and Generalizations.* Birkhäuser Boston, 2001. ISBN: 9781461201731. DOI: 10.1007/978-1-4612-0173-1.

[28] J. Lee and C. Clifton. "How Much Is Enough? Choosing Epsilon for Differential Privacy". In: *Information Security.* Springer Berlin Heidelberg, 2011, pp. 325–340. ISBN: 9783642248610. DOI: 10.1007/978-3-642-24861-0_22.

[29] N. Li, M. Lyu, D. Su, and W. Yang. *Differential Privacy: From Theory to Practice.* Springer International Publishing, 2017. ISBN: 9783031023507. DOI: 10.1007/978-3-031-02350-7.

[30] Y. Li, X. Ren, S. Yang, and X. Yang. "Impact of Prior Knowledge and Data Correlation on Privacy Leakage: A Unified Analysis". In: *IEEE Transactions on Infor-*

mation Forensics and Security 14.9 (Sept. 2019), pp. 2342–2357. ISSN: 1556-6021. DOI: 10.1109/tifs.2019.2895970.

[31] D. Liben-Nowell and J. Kleinberg. "The link prediction problem for social networks". In: *Proceedings of the twelfth international conference on Information and knowledge management*. ACM, Nov. 2003. DOI: 10.1145/956863.956972.

[32] C. Liu, S. Chakraborty, and P. Mittal. "Dependence Makes You Vulnerable: Differential Privacy Under Dependent Tuples". In: NDSS 2016 (2016). DOI: 10.14722/ndss.2016.23279.

[33] Z. C. Luo, K. Albertsson-Wikland, and J. Karlberg. "Target Height as Predicted by Parental Heights in a Population-Based Study". In: *Pediatric Research* 44.4 (Oct. 1998), pp. 563–571. ISSN: 1530-0447. DOI: 10.1203/00006450-199810000-00016.

[34] S. Mahadevan. "Monte carlo simulation". In: *Mechanical Engineering-New York and Basel-Marcel Dekker-* (1997), pp. 123–146.

[35] S. Malik. *Activity Data*. https://www.kaggle.com/datasets/shambhavimalik/activity-data/data. Accessed: 2024-06-17.

[36] J. Near and D. Darais. *Differential Privacy: Future Work & Open Challenges*. https://www.nist.gov/blogs/cybersecurity-insights/differential-privacy-future-work-open-challenges. Accessed: 2024-06-11. Jan. 2022.

[37] A. C. Need and D. B. Goldstein. "Next generation disparities in human genomics: concerns and remedies". In: *Trends in Genetics* 25.11 (Nov. 2009), pp. 489–494. ISSN: 0168-9525. DOI: 10.1016/j.tig.2009.09.012.

[38] T. A. O'Brien, K. Kashinath, N. R. Cavanaugh, W. D. Collins, and J. P. O'Brien. "A fast and objective multidimensional kernel density estimation method: fastKDE". In: *Computational Statistics and Data Analysis* 101 (Sept. 2016), pp. 148–160. ISSN: 0167-9473. DOI: 10.1016/j.csda.2016.02.014.

[39] V. M. Panaretos. *Statistics for Mathematicians*. Springer International Publishing, 2016. ISBN: 9783319283418. DOI: 10.1007/978-3-319-28341-8.

[40] A. Papoulis. *Random variables and stochastic processes*. McGraw Hill, 1965.

[41] K. B. Petersen, M. S. Pedersen, et al. "The matrix cookbook". In: *Technical University of Denmark* 7.15 (2008), p. 510.

[42] M. Rigaki and S. Garcia. "A Survey of Privacy Attacks in Machine Learning". In: *ACM Computing Surveys* 56.4 (Nov. 2023), pp. 1–34. ISSN: 1557-7341. DOI: 10.1145/3624010.

[43] J. Shao. *Mathematical Statistics*. Springer New York, 2003. ISBN: 9780387217185. DOI: 10.1007/b97553.

[44] J. Shurman. *Calculus and Analysis in Euclidean Space*. Springer International Publishing, 2016. ISBN: 9783319493145. DOI: 10.1007/978-3-319-49314-5.

[45] M. Sunnåker and J. Stelling. "Model Extension and Model Selection". In: *Studies in Mechanobiology, Tissue Engineering and Biomaterials*. Springer International Publishing, Oct. 2015, pp. 213–241. ISBN: 9783319212968. DOI: 10.1007/978-3-319-21296-8_9.

[46] L. Sweeney. "Simple demographics often identify people uniquely". In: *Health (San Francisco)* 671.2000 (2000), pp. 1–34.

[47] The Editors of Nature Education. *Allele*. https://www.nature.com/scitable/definition/allele-48/. Accessed: 2024-06-23. 2014.

[48] J. A. Trop. "Topics in Sparse Approximation". PhD thesis. University of Texas, Aug. 2004.

[49] M. C. Tschantz, S. Sen, and A. Datta. "SoK: Differential Privacy as a Causal Property". In: (May 2020). DOI: 10.1109/sp40000.2020.00012.

[50] H. Wang and Z. Xu. "CTS-DP: Publishing correlated time-series data via differential privacy". In: *Knowledge-Based Systems* 122 (Apr. 2017), pp. 167–179. ISSN: 0950-7051. DOI: 10.1016/j.knosys.2017.02.004.

[51] H. Wang, Z. Xu, S. Jia, Y. Xia, and X. Zhang. "Why current differential privacy schemes are inapplicable for correlated data publishing?" In: *World Wide Web* 24.1 (June 2020), pp. 1–23. DOI: 10.1007/s11280-020-00825-8.

[52] M. Wang and S. Xu. "Statistics of Mendelian segregation—A mixture model". In: *Journal of Animal Breeding and Genetics* 136.5 (Apr. 2019), pp. 341–350. ISSN: 1439-0388. DOI: 10.1111/jbg.12394.

[53] L. Wasserman and S. Zhou. "A Statistical Framework for Differential Privacy". In: *Journal of the American Statistical Association* 105.489 (Mar. 2010), pp. 375–389. ISSN: 1537-274X. DOI: 10.1198/jasa.2009.tm08651.

[54] X. Wu, T. Wu, M. Khan, Q. Ni, and W. Dou. "Game Theory Based Correlated Privacy Preserving Analysis in Big Data". In: *IEEE Transactions on Big Data* (2017), pp. 1–1. ISSN: 2332-7790. DOI: 10.1109/tbdata.2017.2701817.

[55] B. Yang, I. Sato, and H. Nakagawa. "Bayesian Differential Privacy on Correlated Data". In: SIGMOD/PODS'15 (May 2015). DOI: 10.1145/2723372.2747643.

[56] T. Zhang, T. Zhu, R. Liu, and W. Zhou. "Correlated data in differential privacy: Definition and analysis". In: *Concurrency and Computation: Practice and Experience* 34.16 (Sept. 2020). DOI: 10.1002/cpe.6015.

[57] J. Zhao, J. Zhang, and H. V. Poor. "Dependent Differential Privacy for Correlated Data". In: *2017 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2017. DOI: 10.1109/glocomw.2017.8269219.

Glossary

Table 4 is a collection of terms that occur multiple times throughout the thesis and may require explanation. If the term is explained in more detail in the thesis, then this is referenced in the glossary.

Term	**Description**
(α, β)-accuracy	A utility metric of an algorithm $\mathcal{A}$ with respect to a ground truth function f. It states that an error greater than α occurs with a probability less than β. See Definition 2.5.
Adversary (K, i)	An adversary as defined by *Bayesian differential privacy leakage*. The given adversary knows the values of the data points with indices in set K, tries to infer the value of data point with index i, and does not know the remaining data points with indices $U = [n] \setminus (K \cup \{i\})$. See Definition 2.11.
Allele	A singular piece of genetic information. Humans have a pair of alleles at each position because they inherit one allele from each parent. See Section 5.
Bayesian differential privacy	A privacy notion that takes correlation between data points into account. It limits the maximum *Bayesian differential privacy leakage* of an algorithm $\mathcal{A}$. See Definition 2.12.
Bayesian differential privacy leakage	The logarithm of the maximal multiplicative difference in the output distribution of an algorithm $\mathcal{A}$ when the value of a single record changes. Also takes correlation between data points into account. A smaller Bayesian differential privacy leakage indicates stronger privacy protection. See Definition 2.11.
Bayes' theorem	A theorem in probability theory. For any events A and B we have $\Pr(A \mid B) = \Pr(B \mid A) \Pr(A) / \Pr(B)$.
BDP	See *Bayesian differential privacy.*
BDPL	See *Bayesian differential privacy leakage.*
Counting query	A function that calculates the number of entries in a data tuple that fulfill a certain predicate.
Data tuple D	A tuple with n records $D = (x_1, \ldots, x_n)^\top \in \mathcal{X}^n$. See Section 2. Has the same semantic meaning as a data set. However, a tuple with a fixed order was notationally more convenient in this thesis.
Differential privacy	A privacy notion that limits the logarithm of the maximal multiplicative difference in the output distribution of an algorithm $\mathcal{A}$ when a single record changes and all other records stay the same. See Definition 2.1.
DP	See *differential privacy.*

d-privacy	A generalisation of *differential privacy* where the privacy leakage depends on the distance d between two data tuples. See Definition 2.6.
Free-lunch privacy	A privacy notion introduced by Kifer et al. [26] to demonstrate that perfect privacy and utility is impossible. A free-lunch private algorithm must have indistinguishable outputs for any two data tuples D, D'. See Definition 2.10.
Gaussian bound	The bound of the BDPL of an ε-DP and d-private algorithm that applies when data has a multivariate Gaussian correlation model. See Proposition 6.4.
General bound	The bound of the BDPL of an ε-DP algorithm that applies in any case. If the maximum number of correlated records in the data is m, then we have $BDPL \leq m\varepsilon$. See Theorem 4.2.
Known indices K	See *adversary* (K, i).
Laplace mechanism $\mathcal{M}_{\varepsilon,f}$	A mechanism that can take a function f and render it ε-*differentially private* by adding Laplacian noise to its output. See Definition 2.4 and Theorem 2.1.
Law of total probability	A theorem in probability theory. For any event A and finite or countably infinite set of events $B = \{B_1, \ldots, B_n\}$ with B_i and B_j disjoint for any $i, j \in [n]$ and $\Pr(B_1 \vee \cdots \vee B_n) = 1$ we have $\Pr(A) = \sum_{i=1}^{n} \Pr(A \mid B_i) \cdot \Pr(B_i)$.
Markov chain bound	The bound of the BDPL of an ε-DP algorithm that applies when data follows a Markov chain. See Corollary 7.4.
Pearson correlation coefficient	A scalar value to measure linear correlation between two random variables. See Definition 2.9.
Sum query	A function that calculates the sum of all entries of a data tuple.
Target index i	See *adversary* (K, i).
Unknown indices U	See *adversary* (K, i).

Table 4: Glossary