Can Swapping be Differentially Private? A Refreshment Stirred, not Shaken

James Bailie, Ruobin Gong and Xiao-Li Meng*

May 4, 2023

This is currently a working paper. Comments and discussion are welcome; constructive criticism will be appreciated.

Abstract

This paper presents a formal privacy analysis of data swapping, a family of statistical disclosure control (SDC) methods which were used in the 1990, 2000 and 2010 US Decennial Census disclosure avoidance systems (DAS). Like all swapping algorithms, the method we examine has invariants – statistics calculated from the confidential database which remain unchanged. We prove that our swapping method satisfies the classic notion of pure differential privacy (ϵ-DP) when conditioning on these invariants. To support this privacy analysis, we provide a framework which unifies many different types of DP while simultaneously explicating the nuances that differentiate these types. This framework additionally supplies a DP definition for the TopDown algorithm (TDA) which also has invariants and was used as the SDC method for the 2020 Census Redistricting Data (P.L. 94-171) Summary and the Demographic and Housing Characteristics Files. To form a comparison with the privacy of the TDA, we compute the budget (along with the other DP components) in the counterfactual scenario that our swapping method was used for the 2020 Decennial Census. By examining swapping in the light of formal privacy, this paper aims to reap the benefits of DP - formal privacy guarantees and algorithmic transparency - without sacrificing the advantages of traditional SDC. This examination also reveals an array of subtleties and traps in using DP for theoretically benchmarking privacy protection methods in general. Using swapping as a demonstration, our optimistic hope is to inspire formal and rigorous framing and analysis of other SDC techniques in the future, as well as to promote nuanced assessments of DP implementations which go beyond discussion of the privacy loss budget ϵ.

Keywords: Differential Privacy, Statistical Disclosure Control, Data Swapping, US Census, Confidentiality.

*jamesbailie@g.harvard.edu, ruobin.gong@rutgers.edu, meng@stat.harvard.edu

1 Connecting Data Swapping with Differential Privacy

1.1 Explicating invariant constraints in differential privacy definitions

In 2018, the United States Census Bureau (USCB) announced an overhaul of their disclosure avoidance system (DAS) [Abowd, 2018]. The DAS for the Decennial Censuses of 1990, 2000 and 2010 principally consisted of a *data swapping* method [McKenna, 2018] which interchanges the reported values of sensitive variables in a randomly selected subset of records [Dalenius and Reiss, 1982, Fienberg and McIntyre, 2004]. In contrast, the USCB declared that the 2020 DAS would be redesigned from the ground up with the primary goal of satisfying a mathematical definition of privacy. This definition, the USCB decided, must be some type of *differential privacy* (DP) [Dwork et al., 2006b] – a large family of technical standards which aim to quantify privacy by measuring the change in the output statistic due to a unit change in the input data.[1]

However, there were other priorities for the 2020 Census, some of which appear to complicate a straightforward adoption of DP. In particular, state population counts are legislatively required to be published exactly. On the other hand, DP typically requires that all published statistics are infused with random noise. The USCB's TopDown algorithm (TDA) [Abowd et al., 2022] – used to protect the 2020 Census Redistricting Data (P.L. 94-171) Summary File, the Demographic and Housing Characteristics (DHC) File and the Demographic Profile[2] – sidesteps this conflict by first applying a DP method to add noise into the 2020 Census data and then removing this noise from a set of key statistics, called the *invariants*, via a complex optimization procedure. The invariants for the 2020 Census include not only the state population totals but also the counts of households at the lowest level of Census geography, amongst other statistics [Abowd et al., 2022, Section 5.2]. More generally, invariants refer to any summaries of data that must be released without subjecting them to any modification, differentially private or not.

To date, the privacy analysis of the TDA has focused solely on its first step, when privacy noise is added to the Census data [Abowd et al., 2022]. Yet any rigorous analysis must encompass the entire TDA procedure and assess the privacy impacts of both the noise infused in the first

[1] See [Desfontaines and Pejó, 2022] for a survey of the numerous differential privacy definitions.

[2] The USCB produces multiple data products from each Decennial Census. For 2020, the three principal data products are the Redistricting Data (P.L. 94-171) Summary File (published August 2021), the Demographic and Housing Characteristics (DHC) File (to be published May 2023) and the Detailed Demographic and Housing Characteristics Files (to be published starting from September 2023) [U.S. Census Bureau].

Both the Redistricting Data and DHC Files use versions of the TopDown algorithm, with the DHC's updated version "tuned" to improve accuracy. Related tabulations across these Files will be consistent. Additionally, "noisy measurement files" for the Redistricting Data and DHC will be published by the USCB. These files are the output of the first step (noise infusion) of the TDA, before the second step (termed "post-processing" by the USCB) is applied [US Census Bureau, 2023a].

In contrast, the Detailed DHC Files will use a new privacy mechanism, the SafeTab-P algorithm [Tumult Labs, 2022, US Census Bureau, 2023f]. The Detailed DHC Files will not be internally consistent nor agree with related tabulations in the Redistricting Data and DHC Files.

To be clear, in this paper we are interested in the assessing the privacy afforded by the two steps of the TDA in combination – which together produce the final versions of the Redistricting Data and DHC Files – and not simply the privacy afforded by the first step – which produces the noisy measurement files.

step and the noise removed in the second step. This second step is not readily addressable by the standard definitions of DP in the current literature, because these definitions do not explicate the permissible counterfactual data universes which are essential for meaningfully defining the DP operations (e.g., the action of alternating a single membership or attribute). Such explication is paramount for handling invariants, much like the explication of conditioning in statistical inference – that is, constraining the possible states by the known or assumed information – is the first step towards properly account for known information. Indeed the process of infusing invariants into DP definitions parallels that of defining and applying a sampling distribution with respect to a sub-population instead of the entire population. The overall mathematical notion of the distribution is the same. The difference is to what state space (e.g., a population) it is applied to.

At the same time, just as defining conditional distributions brings complications and subtleties (such as conditioning on a probability-zero event), defining invariant-infused DP reveals a host of hidden complexities and assumptions implicit in conventional DP definitions, as we explore in Section 3. By making these nuances explicit, we make the notion of DP more applicable and meaningful in the presence of invariants, because any theoretical guarantee of privacy that does not take into account known information (such as invariants) is not a practically relevant one, as the 2020 Census demonstrates. Further, this explication of DP also naturally leads to an understanding of the formal privacy of data swapping confidentiality methods.

1.2 Goals of this paper

This paper presents a formal privacy analysis of data swapping [Dalenius and Reiss, 1982, Fienberg and McIntyre, 2004]. Swapping refers to a general concept and thus there are a broad class of statistical disclosure control (SDC) methods which can be designated as data swapping, including the 1990, 2000 and 2010 US Census DAS [McKenna, 2018]. In order to conduct a rigorous privacy analysis, we focus on a single type of swapping algorithm. Since the details of the USCB's swapping methods have never been made fully public due to confidentiality concerns, our algorithm is designed with the dual mandate of being congenial to a formal privacy analysis while also aligning as much as possible with the public knowledge of the 2010 US Census's swapping method.

We show that this swapping algorithm is differentially private conditioning on the invariants that it produces, and we do so by providing an explicit expression for its privacy loss budget ϵ. This achieves one of the major goal of this paper: a theoretical analysis of the privacy guarantees that swapping affords.

As a prerequisite for such an analysis, we first provide a unifying notion of a *differential privacy definition* as a bound on the derivative of the data-release mechanism. By interpreting differentiation in a very general sense, this abstract definition encompasses many of the different types of DP, while simultaneously making explicit the various nuances which differentiate these types. Although this framework is an important tool on its own (see Bailie et al. [2023+]), for our specific purposes it

demonstrates how invariants can be incorporated into DP, without changing the definition but simply by explicating an existing DP component – the *data universe*. While this perspective results in a suitable privacy definition for swapping, it also enables us to formally describe a privacy definition for the USCB's TopDown algorithm. Such a privacy definition, which encompasses both the noise infusion and noise removal steps of TDA, was previously missing from the literature. Finally, this framework gives necessary context for understanding the privacy budget ϵ of a differential privacy mechanism.

Another major goal of this paper is to establish swapping methods on an equal and formal footing with other invariant-respecting DP mechanisms, in order to compare and elucidate their similarities and differences. Swapping mechanisms have received criticism since they have been shown theoretically to introduce bias into the published data [Drechsler and Reiter, 2010]. But the level of this bias depends on implementation parameters which are typically kept secret. Only with public transparency of the 2010 USCB's swapping algorithm – as enabled by a formal privacy analysis – can the extent of this bias be quantified. This would provide practical considerations for the 2010 Census data, above and beyond what the current theoretical understanding can provide.

With formal privacy analysis serving as the theoretical support, a focal comparison presented in this paper is between the TDA and the counterfactual scenario in which the USCB uses our swapping algorithm to protect the 2020 Census. The swapping algorithm we examine, which mimics the 2010 Census DAS, satisfies a type of differential privacy we call $(\boldsymbol{c}_{\text{Swap}}, \epsilon)$-DP which is the same in form as the privacy definition of the TDA. Thus, this analysis illustrates the evolution over time of the privacy guarantees provided in the US Decennial Censuses.

In a nutshell, this paper continues an existing line of research [Rinott et al., 2018, Bailie and Chien, 2019, Sadeghi and Chien, 2023] examining non-DP statistical disclosure control (SDC) techniques – which are typically regarded as ad-hoc – under the light of formal privacy. Hence our fourth and broadest goal is to demonstrate how seemingly ad-hoc SDC techniques can (and should) be framed and analyzed, formally and rigorously. Since both DP and traditional SDC each have their own unique advantages [Slavković and Seeman, 2023], combining these two somewhat conflicting fields bestows opportunities to reap the best of both worlds. On the one hand, DP supplies a formal, mathematical description of the level and substance of privacy provided by a confidentiality method. DP also allows for complete transparency of the method without any degradation of these privacy guarantees. Recasting swapping algorithms, or other SDC techniques, as formally private provides strong guarantees that the details of these algorithms – which have traditionally been kept secret – can safely be made public. This transparency is an important prerequisite for any valid statistical analysis of privacy-protected data [Gong, 2022]. For example, our work suggests that once the formal privacy guarantee of the 2010 Census DAS is explicitly stated, the details of its implementation can be published without privacy risk, allowing for the first time statisticians, economists and social scientists to appropriately account for the DAS in their analyses. On the other hand, swapping carries its own advantages, including facial validity and logical consistency which are important to data users [boyd and Sarathy, 2022, Hotz and Salvo, 2022, Ruggles et al.,

2019], and which DP methods, such as the 2020 DAS, cannot achieve without partially destroying statistical transparency.

2 A Preview of the Main Results

The swapping algorithm we examine can be briefly summarised as follows. The variable set is partitioned into two non-empty categories: the *swapping variables* $\boldsymbol{V}_{\text{Swap}}$ and the *holding variables* $\boldsymbol{V}_{\text{Hold}}$. Each record is independently selected with probability given by a parameter p called the *swap rate*. The swapping variables $\boldsymbol{V}_{\text{Swap}}$ of the selected records are then randomly shuffled. More specifically, a derangement σ (i.e. a permutation with no fixed points) over the selected records' indices is sampled uniformly at random. The new $\boldsymbol{V}_{\text{Swap}}$ of a selected record i is given by the $\boldsymbol{V}_{\text{Swap}}$ of the $\sigma(i)$-th selected record.

Sometimes, swapping is restricted to records which share the same values on a (possibly empty) subset of the holding variables, called the *matching variables* $\boldsymbol{V}_{\text{Match}}$. Also referred to as the *swap key* [McKenna, 2018, Abowd and Hawes, 2023], the matching variables are often important characteristics of the data population, as they define strata so that swapping is restricted within these strata. Whenever $\boldsymbol{V}_{\text{Match}}$ is nonempty, the above procedure is repeated independently within each category of $\boldsymbol{V}_{\text{Match}}$.

Like the 2020 DAS, swapping maintains some statistics as invariant. Specifically, there are two contingency tables which remain unchanged by swapping: 1) $\boldsymbol{V}_{\text{Match}} \times \boldsymbol{V}_{\text{Swap}}$: the cross-classification of the matching variables by the swapping variables; and 2) $\boldsymbol{V}_{\text{Hold}}$: the cross-classification of all the holding variables. We denote these two invariant tables by $\boldsymbol{c}_{\text{Swap}}$. The interior of the contingency table $(\boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}) \times \boldsymbol{V}_{\text{Swap}}$ is perturbed by the swapping algorithm.

The following result is a simplification of our main results give in Section 5 for general p, where the notation $(\boldsymbol{c}_{\text{Swap}}, d_{\text{HamS}}, \epsilon)$-DP denotes pure ϵ-DP mechanism conditioning on the invariants defined by $\boldsymbol{c}_{\text{Swap}}$ and with respect to the Hamming distance d_{HamS}. (We explain in detail this notation in Section 3 and argue its necessity in Section 4.)

A Simplified Main Result *The above swapping mechanism is* $(\boldsymbol{c}_{\text{Swap}}, d_{\text{HamS}}, \epsilon)$*-differentially private, where*

$$\epsilon \leq \ln\left[(b+1)(1-p)p^{-1}\right], \quad \text{when} \quad 0 < p \leq 0.5,$$

where b *is the number of records in the largest category of* $\boldsymbol{V}_{\text{Match}}$.

In intuitive terms, this result says that the swapping algorithm we consider satisfies ϵ-differential privacy, when conditioning on the swapping invariants $\boldsymbol{c}_{\text{Swap}}$. What does this mean? In DP, ϵ can informally be viewed as a measure of the difference in output between any two neighbouring input datasets. In this case, we measure change in input datasets via the Hamming distance d_{HamS} (disregarding ordering of records). By conditioning on the swapping invariants, we restrict our

consideration to input datasets which have the same values on $\boldsymbol{c}_{\text{Swap}}$. More exactly, the data universe $\mathscr{D}$ is now a function of the confidential dataset $\boldsymbol{X}^*$:

$$\mathscr{D}(\boldsymbol{X}^*) = \{\boldsymbol{X} : \boldsymbol{c}_{\text{Swap}}(\boldsymbol{X}) = \boldsymbol{c}_{\text{Swap}}(\boldsymbol{X}^*)\},$$

and we measure the difference in output between any two $\boldsymbol{X}, \boldsymbol{X}' \in \mathscr{D}(\boldsymbol{X}^*)$. In order to maintain the robustness property of DP, we must analyse every data universe $\mathscr{D}(\boldsymbol{X}^*)$, over all the possible confidential datasets $\boldsymbol{X}^*$. While classic DP compares all neighboring datasets in $\bigcup_{\boldsymbol{X}^*} \mathscr{D}(\boldsymbol{X}^*)$ simultaneously, invariant-respecting DP analyzes each $\mathscr{D}(\boldsymbol{X}^*)$ separately. Rather than being ad-hoc, this restriction is a necessary precondition for defining differential privacy with invariants (whether for swapping, the TDA, or other methods [Gong and Meng, 2020, Gao et al., 2022, Dharangutte et al., 2023]), as illustrated by Propositions 4.3 and 4.4. If we were required to compare datasets across $\mathscr{D}(\boldsymbol{X}^*)$, then we could no longer release invariants exactly. Moreover, as explained in Section 3, restriction of the data universe is a typical procedure in many statistical analyses.

Informal Definition A mechanism T satisfies $(\boldsymbol{c}, d_{\mathcal{X}}, \epsilon)$-*differential privacy* if, for all $\boldsymbol{X}^*$, all possible outputs t and all datasets $\boldsymbol{X}, \boldsymbol{X}' \in \mathscr{D}(\boldsymbol{X}') = \{\boldsymbol{X} : \boldsymbol{c}(\boldsymbol{X}) = \boldsymbol{c}(\boldsymbol{X}^*)\}$, the following inequality is satisfied:

$$\Pr(T(\boldsymbol{X}) = t) \leq \exp\left[\epsilon d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}')\right] \Pr(T(\boldsymbol{X}') = t).$$

Here the divergence $d_{\mathcal{X}}$ typically encodes the notion of neighbouring datasets. Datasets $\boldsymbol{X}, \boldsymbol{X}'$ with $d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}') = 1$ are called neighbours in the DP literature. Neighbours represent *unit changes* against which changes in the output of T are measured.

This discussion highlights some of the important components underlying any differential privacy definition. Other components, as discussed in the Section 3, include the choice of divergence $d_{\mathcal{T}}$ on the output space: pure DP uses the multiplicative distance, while other types of DP such as (ϵ, δ)-DP and ρ-zero concentrated DP (zCDP) use different divergences. Revealing these nuances allows the definition of differential privacy to be understood as a broad standard that can be used to investigate and compare a wide range of confidentiality methods. Importantly, many existing implementations of differential privacy rely on convenient interpretations of these nuances. By casting DP in a light which illuminates these nuances, we hope to improve the clarity and rigor in assessing DP implementations. (We provide a limited discussion of suggested improvements in Section 4 with the aspiration this sparks more extensive conversation on this important subject.)

Figure 2.1 is a graphical depiction of the swap rate (p) to privacy loss budget (ϵ) conversion, a result discussed in detail in Section 5. As Theorem 5.6 makes precise, the relationship between the swap rate and the nominal ϵ achieved by swapping depends on b, the size of the largest stratum delineated by $\boldsymbol{V}_{\text{Match}}$. Three observations are worth noting. First, for each b, there exists a smallest ϵ, call it ϵ_b, below which no swap rate $p \in (0, 1)$ can attain. Marked by the outlined diamonds in the Figure, we see that the larger the b, the larger the ϵ_b: for example when $b = 10$, ϵ_b is 1.15 (at $p = 76\%$), whereas when $b = 10^6$, ϵ_b is 6.91 (at $p = 99.9\%$). Second, for every b and every

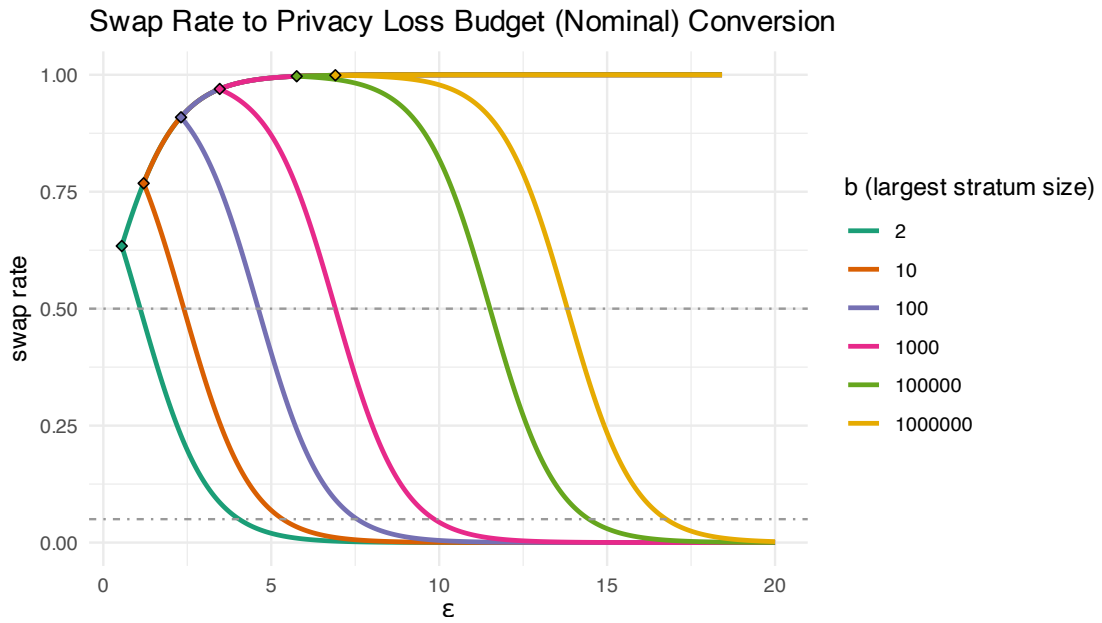

Figure 2.1: Conversion between the swap rate (p) and the nominal privacy loss budget (ϵ) at different values of b, the size of the largest stratum delineated by $\boldsymbol{V}_{\text{Match}}$ (from 2 to 1 million, color coded). Outlined diamonds indicate the smallest ϵ attainable for each b. Grey dotted lines correspond to swap rates of 5% and 50% respectively. The ϵ values are nominal in that the privacy guarantee they afford shall be understood in the context of $\boldsymbol{c}_{\text{Swap}}$ (and hence b).

attainable ϵ at that b (except for $\epsilon = \epsilon_b$), two different swap rates can achieve that ϵ, the higher one of the two often being very close to 100%. For example at $b = 10$, a swap rate of either 33.2% or 95.2% achieves the nominal ϵ of 3. The reason behind this is that, under the swapping scheme we consider, swaps are derangements and have no fixed points. Thus an overly aggressive swap rate may inadvertently preserve the statistical information, akin to a randomised response mechanism with a high probability of flipping the binary confidential answer.

Third and most importantly, we emphasize that the ϵ values visualized in Figure 2.1 are *nominal* in the sense that the privacy guarantee they afford must be understood with respect to the full context as outlined by the privacy definition. An aspect of this context is b, the size of the largest stratum of $\boldsymbol{V}_{\text{Match}}$, and as a result, even the same ϵ value across different b's shall not be equated to be the same privacy guarantee. Indeed, the reader may have noticed that the ordering of the b curves in the Figure suggests a seemingly peculiar fact that, for a larger b, a higher p is needed to achieve the same ϵ. In Section 5.3, we provide a numerical demonstration of our swapping mechanism using the 1940 Decennial Census full count data, and in Section 6.2 a "what-if" analysis on the counterfactual application of swapping to the 2020 Census, to further illustrate the contextual nature of the privacy guarantee.

As part of the comparative analysis of our swapping algorithm and the 2020 DAS, we show in Section 6.1 that the formal privacy of the 2020 DAS requires relaxations of ρ-zero concentrated

DP (or approximate-(ϵ, δ)-DP) to allow for invariants. Thus, our swapping mechanism satisfies a type of DP which is similar to that of the 2020 DAS. The within-system privacy evaluation (i.e. the privacy budget ϵ) of the 2020 redistricting data and DHC files is more than double that of our swapping algorithm (with a 2-4% swap rate as was purportedly used in the 2010 Census DAS [boyd and Sarathy, 2022]).[3] However, as we will discuss in Section 6, there are a number of important caveats to this statement. Firstly, the output of the swapping algorithm can be used to produce all of the Census data publications (with no increase in privacy budget), whereas additional data products – such as the Detailed DHC Files – will necessarily increase the total privacy loss in the 2020 Census. More importantly, swapping necessitates more invariants than were used by the TDA. Hence, the privacy budgets of our swapping algorithm and the 2020 DAS cannot be compared directly, since they satisfy different types of DP. This reflects the broader point that privacy budgets are contextual and their interpretation depends on a range of factors, including the sensitivity of the collected microdata, and of the published statistics, as well as the exact privacy definition used, as discussed in Section 4.

Our formal assessment of swapping showcases the *relative* privacy guarantee it affords. It does not, as no disclosure mechanism can, offer *absolute* privacy guarantees. While a determination of swapping's privacy budget ϵ provides a within-system privacy evaluation, across-system evaluations – comparing the disclosure risk of ϵ-DP mechanisms with different sets of invariants – is inherently subjective and contextual. We leave this as an important topic for future research.

Paper Organization Section 3 develops the abstract differential privacy definition from an intuition of bounding the derivative of the output statistics per unit of input data. It describes the three components of any differential privacy definition: the data universe $\mathcal{D}$, the divergence $d_{\mathcal{X}}$ on the input data and the divergence $d_{\mathcal{T}}$ on the output. Section 4 argues that any privacy evaluation of a mechanism must be made with regard to these three components, as context for understanding the mechanism's privacy loss ϵ. Section 5 derives the necessary invariants of swapping; formally defines our swapping algorithm; presents the differential privacy analysis of this algorithm; and demonstrates its use on the 1940 US Decennial Census full count data. Section 6 uses the framework of Section 3 to provide differential privacy definitions for the 2020 TopDown algorithm and conducts a counterfactual thought experiment of applying swapping to the 2020 Decennial Census. Section 7 ends the paper with a discussion on some criticisms and extensions of data swapping.

[3] We have not attempted to verify the accuracy of boyd and Sarathy [2022]'s claim that the swap rate was between 2-4% for the 2010 Census.

3 Explicating the Nuances of Differential Privacy

3.1 Set up and intuition

A data custodian is interested in releasing a privacy-protected (i.e. *sanitised*) statistic $T \in \mathcal{T}$ based on a dataset $\boldsymbol{X} \in \mathcal{X}$. The dataset $\boldsymbol{X}$ is observed by the data custodian and is some representation of a population. In this Section, we focus on the case where the population consists of a collection of *individuals* whose privacy must be respected when publishing T. This generalises to settings where the population units are, for example, businesses or to settings where multiple types of units require privacy protection, for example individuals and households.

Typically $\mathcal{T} \subset \mathbb{R}^d$, although we do not rule out the possibility that the output space $\mathcal{T}$ is more complex. For example, T may itself be a (synthetic) dataset. To be clear, T – like any statistic – is not a fixed value. Rather T is a *function* of the data $\boldsymbol{X} \in \mathcal{X}$ which transforms $\boldsymbol{X}$ into some value in the output space $\mathcal{T}$. We allow the output of T to depend not just on $\boldsymbol{X}$ but also upon some auxiliary randomness, say a uniform random variable $U \sim \text{Uniform}[0,1]$.[4] U provides the noise which is used by T to protect the privacy of $\boldsymbol{X}$.

Thus, the statistic $T(\cdot, \cdot)$ is a function $\mathcal{X} \times [0,1] \to \mathcal{T}$, which, in the differential privacy literature, is typically called the *privacy mechanism*. We instead refer to T as the *data-release mechanism* to emphasize that – in addition to privacy protection – T may encompass many other data processing steps (such as cleaning, coding, imputation, etc.) from data collection through to data publication. Indeed, understanding the starting point of T – and hence how the data $\boldsymbol{X}$ *represents* individuals in the population – is crucial to understanding what is or, more importantly, is not protected by T, as explored further in Section 7.1.

We stress the duality of T as simultaneously a statistic from the data user's perspective, and a privacy mechanism from the privacy analyst's perspective. This duality lies at the heart of the fundamental tension of this field: the tradeoff between privacy and utility. T must be designed to balance these two competing interests.

By convention in differential privacy, the dataset $\boldsymbol{X}$ is considered fixed and is not modelled, so that the randomness in $T(\boldsymbol{X}, U)$ is induced solely by U. Therefore, the *data* $\boldsymbol{X}$, as the object of an attacker's inference, plays the role of the *parameter* in privacy analysis. An immediate and critical consequence of this recognition is that any distribution placed on $\boldsymbol{X}$ can be viewed either as a posited generative model for $\boldsymbol{X}$ or as a prior distribution for $\boldsymbol{X}$, or a mixture of both. We emphasise this crucial observation by denoting the law of $T(\boldsymbol{X}, U)$ by $P_{\boldsymbol{X}}(T(\boldsymbol{X}, U) \in \cdot)$ or, if T is clear from the context, $P_{\boldsymbol{X}}$.[5] When the dependence of T on U (or on both U and $\boldsymbol{X}$) is apparent from the context, we write $T(\boldsymbol{X})$ (or just T) for simplicity.

[4] Ignoring computational issues, a single uniform random variable U is sufficient for all practical applications, since U can generate countably many (independent) random variables of arbitrary distribution via the inverse-CDF method. .

[5] We require the technical condition that, for all $\boldsymbol{X} \in \mathcal{X}$, the function $T(\boldsymbol{X}, \cdot)$ is measurable with respect to a given σ-algebra on $\mathcal{T}$ and the Borel σ-algebra $[0,1]$, so that $P_{\boldsymbol{X}}$ is well-defined.

Differential privacy is a condition on the data-release mechanism T. Loosely, it is the requirement that if the data $\boldsymbol{X}$ change slightly, then the output $T(\boldsymbol{X})$ – or more precisely, the distribution $P_{\boldsymbol{X}}(T \in \cdot)$ of the output – also changes slightly. Succinctly, DP requires that T is robust to changes in the data input $\boldsymbol{X}$.

Intuitive Definition: A data-release mechanism T satisfies *differential privacy* (DP) if the 'derivative' of the map $\boldsymbol{X} \mapsto P_{\boldsymbol{X}}(T \in \cdot)$ is bounded within $[-\epsilon, \epsilon]$, for all datasets $\boldsymbol{X}$ in the data universe $\mathcal{D}$.

Here the derivative is understood in a loose sense as the small change in output $P_{\boldsymbol{X}}(T \in \cdot)$ per small change in input $\boldsymbol{X}$. The parameter $\epsilon > 0$ is a measure of the 'degree' of privacy and the minimum possible ϵ can be intuitively thought of as the *privacy loss* of T. A small ϵ implies that T is stable to perturbations in $\boldsymbol{X}$; a large ϵ means that a change in $\boldsymbol{X}$ can be influential. Ideally, a data curator would choose a value of ϵ – which in this context is called the *privacy loss budget* – based on the sensitivity of the data and the broader social context [Nissenbaum, 2010] and design a mechanism T with privacy loss at most ϵ. However, in practice, choosing the budget ϵ is an opaque process, complicated by various nuances underlying differential privacy. For example, what are the practical implications on real-world privacy protections from increasing the budget by one unit? To answer this question, one must understand the scale of the privacy budget. In the standard formulation of DP (pure ϵ-DP), the privacy loss is measured on the log scale, since under pure DP, ϵ is a bound on the log-likelihood ratio $\log P_{\boldsymbol{X}} - \log P_{\boldsymbol{X}'}$, not the likelihood ratio $\frac{P_{\boldsymbol{X}}}{P_{\boldsymbol{X}'}}$. Thus, it might be more accurate in this case to call ϵ the log-budget. Indeed, when interpreting the privacy loss, the privacy semantics of ϵ-DP are all expressed in terms of $\exp(\epsilon)$, rather than ϵ (see e.g. Kifer et al. [2022], Wasserman and Zhou [2010], Gong et al. [2023+]).

The scale of the privacy loss ϵ is implicit in the 'derivative' $\frac{d}{d\boldsymbol{X}} P_{\boldsymbol{X}}$. In fact, by writing $\frac{d}{d\boldsymbol{X}} P_{\boldsymbol{X}}$, one must have set (consciously or unconsciously) the metric against which changes in $P_{\boldsymbol{X}}$ are measured – indeed this is a requirement to define a differential $dP_{\boldsymbol{X}}$. The metric used in $dP_{\boldsymbol{X}}$ for pure ϵ-DP happens to be on the log scale. The privacy budget ϵ inherits this scale, which complicates its interpretation, since a change in budget from (for example) $\epsilon = 10.3$ to $\epsilon = 17.9$ is very different to a change from $\exp(10.3) \approx 30,000$ to $\exp(17.9) \approx 59$ million. Other types of DP use different measures of change, which in turn imply different scales for their privacy loss budgets (see Sections 3.6 and 4.2).

As for the differential $dP_{\boldsymbol{X}}$, we likewise need a measure of change between datasets $\boldsymbol{X}$ and $\boldsymbol{X}'$ for the differential $d\boldsymbol{X}$. More fundamentally, we must first define the space of realizable datasets $\mathcal{D}$, which is called the *data universe*. As we shall now see, even such an apparently simple concept as the space of possible datasets $\mathcal{D}$ gives rise to its own nuances. Yet to make precise the definition of differential privacy, requires that these complications are explored. This is the goal of the next Section.

3.2 The data universe $\mathcal{D}$

The differential privacy guarantee given in the intuitive definition above is afforded to datasets in a given data universe $\mathcal{D}$. That is, we bound the derivative $\frac{d}{d\boldsymbol{X}}P_{\boldsymbol{X}}$ only at each $\boldsymbol{X} \in \mathcal{D}$. The data universe $\mathcal{D}$ may be equal to $\mathcal{X}$ (the collection of all datasets that are structurally and mechanically possible) but more generally $\mathcal{D}$ is a subset of $\mathcal{X}$.

Sometimes the data curator may partially base their choice of $\mathcal{D}$ on the observed, confidential dataset, which we denote by $\boldsymbol{X}^*$. The most prominent example of this situation is the 2020 US Census. There, $\mathcal{D}$ was restricted to datasets which share the same state population totals (amongst other quantities) as $\boldsymbol{X}^*$. To accommodate this possibility, we take $\mathcal{D}$ to be the output of a *set-valued* function $\mathscr{D}(\cdot) : \mathcal{X} \to 2^{\mathcal{X}}$. In this way, there are multiple data universes $\{\mathscr{D}(\boldsymbol{X})\}_{\boldsymbol{X}\in\mathcal{X}}$, with each universe $\mathcal{D} = \mathscr{D}(\boldsymbol{X})$ associated to the event that the data custodian observes the confidential dataset to be $\boldsymbol{X}$.

DP requires that, for every data universe $\mathcal{D} \in \{\mathscr{D}(\boldsymbol{X})\}_{\boldsymbol{X}\in\mathcal{X}}$, the derivative $\frac{d}{d\boldsymbol{X}}P_{\boldsymbol{X}}$ is bounded at every $\boldsymbol{X}$ in the space $\mathcal{D}$ – but not necessarily in the space $\mathcal{X}$. This distinction is crucial since $\boldsymbol{X}$ typically has very high dimensional, so that there are multiple derivatives – one for each dimension. Restricting $\mathcal{X}$ to $\mathcal{D}$ restricts not just the datasets $\boldsymbol{X}$ which are protected, but also the protections afforded to them (i.e. in which directions the derivative $\frac{d}{d\boldsymbol{X}}P_{\boldsymbol{X}}$ is bounded). So restricting $\mathcal{D}$ strictly weakens the privacy protection in two senses: 1) it limits the counterfactual datasets that are protected and 2) it reduces the protection afforded to each of the protected datasets.

While restricting the data universe via $\mathscr{D}$ leads to a reduction in actual privacy protection, this complication is necessary in many real-world applications of DP. In addition to other examples from the DP literature, we will demonstrate in Section 6.1 that $\mathscr{D}$ is required to describe the formal privacy protections afforded to the 2020 US Census response. (See also Section 4.3 for a more general discussion of its necessity.) Furthermore, this practice is typical in statistical disclosure control and data analysis more broadly. Top-coding – where one sets a maximum limit on a continuous variable, usually after looking at the raw data – is one common example. More generally, data-dependent categorization of a continuous variable entails a restriction of the data universe.

An important class of data universe functions $\mathscr{D}$ encodes *invariants*: exact quantities calculated from the confidential data. Due to legal and policy mandates or other guidance that the data curator must observe, invariants are published as-is. From the perspective of data utility, invariants are thus restrictions on the output of a mechanism. Conversely, from the perspective of data privacy, invariants are restrictions on the input, or more exactly, the data universe $\mathcal{D}$.

In this work, we encode the invariants as a deterministic function $\boldsymbol{c} : \mathcal{X} \to \mathbb{R}^l$ of the database.[6] The

[6] In some applications (such as the 2020 Decennial Census), there are also inequality invariants [Abowd et al., 2022]. As an example of such an invariant, TDA requires that the reported number of group quarters in any geographical unit is at most the number of persons in that unit. More generally, an inequality invariant is of the form $f(\boldsymbol{X}) \leq 0$

invariant-compliant data universe function $\mathscr{D}_{\boldsymbol{c}}$ is defined as

$$\mathscr{D}_{\boldsymbol{c}}(\boldsymbol{X}) = \Big\{\boldsymbol{X}' \in \mathcal{X} : \boldsymbol{c}(\boldsymbol{X}') = \boldsymbol{c}(\boldsymbol{X})\Big\}. \tag{3.1}$$

Note that an invariant function $\boldsymbol{c}$ defines an equivalence relation $\sim$ over its domain $\mathcal{X}$. Specifically, two datasets $\boldsymbol{X} \sim \boldsymbol{X}'$ if and only if $\boldsymbol{c}(\boldsymbol{X}) = \boldsymbol{c}(\boldsymbol{X}')$. Hence, the data universe function (3.1) induces a *partition* of $\mathcal{X}$ indexed by the image of the invariant function c.

Example 3.1. Let the dataset be an contingency table of $m \times n$ records taking non-negative integer values: $\mathcal{X} = (\mathbb{N}^+)^{m\times n}$. Suppose the function $\boldsymbol{c} : (\mathbb{N}^+)^{m\times n} \to (\mathbb{N}^+)^{m+n}$ tabulates the column- and row-margins:

$$\boldsymbol{c}(\boldsymbol{X}) = \left(\sum_{i=1}^{m} x_{i1}, \ldots, \sum_{i=1}^{m} x_{in}, \sum_{j=1}^{n} x_{1j}, \ldots, \sum_{j=1}^{n} x_{mj}\right).$$

If the data curator treats the column- and row-margins of the confidential dataset as invariant, it would be equivalent to employing the data universe function $\mathscr{D}$ as defined in (3.1) using the $\boldsymbol{c}$ function above.

3.3 Measuring change via divergences

We now return to the question of measuring change in $\boldsymbol{X}$ and in $P_{\boldsymbol{X}}$, so that we can make the intuitive definition precise. We measure change very generally via the notion of divergence:

Definition 3.2. A *divergence* d on a set S is a function $S \times S \to [0, \infty]$ satisfying $d(x, y) = 0$ if $x = y$.

Divergences generalise the concept of metrics. We use the term divergence to highlight that, although they measure distance in some abstract sense, we generally do not require them to be a metric.

To define DP, we need to equip the space $\mathcal{X}$ with a divergence $d_{\mathcal{X}}$. To measure change in $P_{\boldsymbol{X}}$, define $\mathcal{L}(\mathcal{T})$ as the set of probability distributions on the measurable space $\mathcal{T}$ and equip the space $\mathcal{L}(\mathcal{T})$ with a divergence $d_{\mathcal{T}}$. Now we can understand the 'derivative' of T at $\boldsymbol{X}$ (in the direction towards $\boldsymbol{X}'$) as the ratio $\frac{d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'})}{d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}')}$. DP is the requirement that this ratio is bounded between $[-\epsilon, \epsilon]$.

3.4 Formalising the intuition

Definition 3.3. Let $\mathcal{X}$ be the space of input datasets and let the measurable space $(\mathcal{T}, \mathscr{F})$ be the space of outputs. A *differential privacy definition* is a tuple $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ where

for some function $f : \mathcal{X} \to \mathbb{R}$. Such an invariant can be incorporated in our framework by defining

$$c(\boldsymbol{X}) = \begin{cases} 1 & \text{if } f(\boldsymbol{X}) \leq 0, \\ 0 & \text{otherwise.} \end{cases}$$

1. $\mathscr{D} : \mathcal{X} \to 2^{\mathcal{X}}$ is the data universe function which associates the observed dataset $\boldsymbol{X}$ with a data universe $\mathcal{D} = \mathscr{D}(\boldsymbol{X}) \subset \mathcal{X}$;
2. $d_{\mathcal{X}}$ is a divergence on $\mathcal{X}$; and
3. $d_{\mathcal{T}}$ is a divergence on the space $\mathcal{L}(\mathcal{T})$ of probability distributions on $\mathcal{T}$.

There are many definitions of differential privacy, each corresponding to different choices of $\mathscr{D}, d_{\mathcal{X}}$ and $d_{\mathcal{T}}$.

The choice of $d_{\mathcal{T}}$ has received much attention in the literature. In Section 3.6, we demonstrate how different choices of $d_{\mathcal{T}}$ correspond to pure-DP [Dwork et al., 2006b], approximate and probabilistic DP [Dwork et al., 2006a, Meiser, 2018], Rényi DP [Mironov, 2017] and concentrated DP [Dwork and Rothblum, 2016, Bun and Steinke, 2016a]. Deciding upon the choices for $\mathscr{D}$ and $d_{\mathcal{X}}$ – which in practice are no less important than $d_{\mathcal{T}}$ – have received comparatively little attention.

Definition 3.4. A data-release mechanism $T : \mathcal{X} \times [0,1] \to \mathcal{T}$ *satisfies* a differential privacy definition $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with privacy budget $\epsilon_{\mathcal{D}} \geq 0$ if

$$d_{\mathcal{T}}\big[P_{\boldsymbol{X}}(T \in \cdot), P_{\boldsymbol{X}'}(T \in \cdot)\big] \leq \epsilon_{\mathcal{D}} d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}'), \tag{3.2}$$

for all $\boldsymbol{X}, \boldsymbol{X}'$ in every data universe $\mathcal{D} \in \operatorname{Im} \mathscr{D}$.[7]

We allow the privacy budget $\epsilon_{\mathcal{D}}$ to vary with the data universe $\mathcal{D}$. That is, $\epsilon_{\mathcal{D}}$ is a function $\operatorname{Im} \mathscr{D} \to \mathbb{R}^{\geq 0}$.

DP is Lipschitz continuity Differential privacy is the requirement of *bounded change* in T's output $P_{\boldsymbol{X}}$ per unit change in input $\boldsymbol{X}$. As stated above, this is intuitively like bounding the derivative of T. Since the notion of derivatives does not readily generalise beyond Euclidean space, this intuition – that DP is a bound on the derivative – is not technically true. However, we can formalise this intuition using Lipschitz continuity, which is equivalent to bounded derivatives in $\mathbb{R}^n$:[8] When $d_{\mathcal{X}}$ and $d_{\mathcal{T}}$ are metrics, equation (3.2) is exactly the condition that the function

$$\begin{aligned} \mathcal{D} &\to \mathcal{L}(\mathcal{T}), \\ \boldsymbol{X} &\mapsto P_{\boldsymbol{X}}(T \in \cdot), \end{aligned}$$

is Lipschitz continuous with Lipschitz constant $\epsilon_{\mathcal{D}}$. Thus, differential privacy is precisely the requirement that T is Lipschitz continuous.

[7] So that the probability measures $P_{\boldsymbol{X}}(T)$ exist, we additionally require the (weak) technical condition that $T(\boldsymbol{X}, \cdot)$ is measurable for all $\boldsymbol{X} \in \mathcal{X}$.

To resolve the edge case where $\epsilon = 0$ but $d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}') = \infty$, we define $0 \times \infty = \infty$. This means DP never controls the difference between $P_{\boldsymbol{X}}$ and $P_{\boldsymbol{X}'}$ when $\boldsymbol{X}$ and $\boldsymbol{X}'$ are incomparable (i.e. $d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}') = \infty$), even in the case of complete privacy ($\epsilon = 0$).

[8] For open $D \subset \mathbb{R}^n$, a differentiable function $D \to \mathbb{R}^m$ has bounded derivatives if and only if it is Lipschitz continuous.

3.5 The divergence $d_{\mathcal{X}}$

Typically the divergence $d_{\mathcal{X}}$ is built from a relation r on $\mathcal{X}$:

$$d_r(\boldsymbol{X}, \boldsymbol{X}') = \begin{cases} 0 & \text{if } \boldsymbol{X} = \boldsymbol{X}', \\ 1 & \text{else if } \boldsymbol{X} \overset{r}{\sim} \boldsymbol{X}', \\ \infty & \text{otherwise.} \end{cases} \tag{3.3}$$

The relation r captures some notion of "neighbouring" datasets: $\boldsymbol{X} \overset{r}{\sim} \boldsymbol{X}'$ if $\boldsymbol{X}$ and $\boldsymbol{X}'$ are neighbours. There are multiple different relations r used in the differential privacy literature but they are all formalisations of the following intuitive definition: Datasets $\boldsymbol{X}$ and $\boldsymbol{X}'$ are neighbours – i.e. $\boldsymbol{X} \overset{r}{\sim} \boldsymbol{X}'$ – if they differ only on a single unit. However, what is exactly meant by a "unit" also varies across the DP literature. The most common examples of units are persons, but units are often families or businesses. In the case where a person repeatedly interacts with a service (such as social media), the units are sometimes the data generated by a single interaction, or the data generated by a person over a single day. In Section 4, we will argue that the choice of the privacy unit is critical to any privacy assessment.

In any case, privacy units must be the building blocks of every dataset $\boldsymbol{X}$. That is, in order to define $d_{\mathcal{X}}$ via a neighbour relation r, it is required that $\mathcal{X} \subset \bigcup_{n=0}^{\infty} \mathcal{R}^n$, where $\mathcal{R}$ is the set of all theoretically-possible units and $\mathcal{R}^n$ is the n-fold cartesian product of $\mathcal{R}$. Thus, every dataset $\boldsymbol{X} \in \mathcal{X}$ is a vector (or multi-set, if we can disregard ordering), with each element of the vector being a single privacy unit.

Once the privacy unit has been fixed, there are two common choices for r: A) $\boldsymbol{X} \overset{r_b}{\sim} \boldsymbol{X}'$ if $\boldsymbol{X}$ and $\boldsymbol{X}'$ have the same number of units but take different values on exactly one unit. That is, $\boldsymbol{X} \overset{r_b}{\sim} \boldsymbol{X}'$ if $|\boldsymbol{X}| = |\boldsymbol{X}'| = n$ and $\frac{1}{2}|\boldsymbol{X} \ominus \boldsymbol{X}'| = 1$, where $\ominus$ is the symmetric set difference. B) $\boldsymbol{X} \overset{r_u}{\sim} \boldsymbol{X}'$ if $\boldsymbol{X}'$ can be formed by adding or subtracting a unit from $\boldsymbol{X}$. That is, $\boldsymbol{X} \overset{r_u}{\sim} \boldsymbol{X}'$ if $|\boldsymbol{X} \ominus \boldsymbol{X}'| = 1$.[9] In the literature, r_b is referred to as *bounded* DP and r_u *unbounded* DP since r_u – unlike r_b – relates datasets of differing length.

A divergence d_r built from a relation r as in (3.3) can always be sharpened to a metric d_r^*. Here $d_r^*(\boldsymbol{X}, \boldsymbol{X}')$ is defined as the length of a shortest path between $\boldsymbol{X}$ and $\boldsymbol{X}'$ in the graph on $\mathcal{X}$ with

[9] In the situations where the ordering of the data is meaningful (such as under the shuffle model of DP [Erlingsson et al., 2019]), $\boldsymbol{X} \overset{r_b}{\sim} \boldsymbol{X}'$ if $|\boldsymbol{X}| = |\boldsymbol{X}'| = n$ and $\sum_{i=1}^{n} \mathbb{1}\{X_i \neq X_i'\} = 1$; and $\boldsymbol{X} r_u \boldsymbol{X}'$ if $|\boldsymbol{X}| = |\boldsymbol{X}'| \pm 1$ and there exists some $j \in \{1, \ldots, \max(|\boldsymbol{X}|, |\boldsymbol{X}'|)\}$ such that $\boldsymbol{X} = \boldsymbol{X}'_{-j}$ or $\boldsymbol{X}_{-j} = \boldsymbol{X}'$. (Here we use the notation $\boldsymbol{v}_{-j}$ to denote the vector $(v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_n)$ where the j-th element of $\boldsymbol{v}$ has been removed.)

edges given by r. For example the extension of d_{r_b} is the Hamming distance on unordered datasets:[10]

$$d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}') = \begin{cases} \frac{1}{2}|\boldsymbol{X} \ominus \boldsymbol{X}'| & \text{if } |\boldsymbol{X}| = |\boldsymbol{X}|, \\ \infty & \text{otherwise} \end{cases} \tag{3.4}$$

and the extension of d_{r_u} is the symmetric difference distance:

$$d^u_{\text{SymDiff}}(\boldsymbol{X}, \boldsymbol{X}') = \left|\boldsymbol{X} \ominus \boldsymbol{X}'\right|. \tag{3.5}$$

The superscript ^{u} emphasizes that these distances are defined with respect to a choice of the privacy unit u. Each choice of privacy unit u defines a different version of the Hamming distance d^u_{HamS} (and different $d_{r_u}, d_{r_b}, d^u_{\text{SymDiff}}$), since the unit defines the elements of the *multi-set* $\boldsymbol{X}$ and hence the operation $\ominus$.

Under mild assumptions, the privacy definitions $(\mathcal{X}, d_r, d_{\mathcal{L}})$ and $(\mathcal{X}, d^*_r, d_{\mathcal{L}})$ are equivalent[11] if and only if $d_{\mathcal{L}}$ is a metric [Bailie et al., 2023+].

3.6 The divergence $d_{\mathcal{T}}$

Different variants of DP (pure, approximate, zCDP, Rényi, etc.) correspond to different choices of $d_{\mathcal{T}}$. These variants refer to families of DP definitions, since they leave the other two components $\mathscr{D}$ and $d_{\mathcal{X}}$ unspecified.

In classical (pure) ϵ-DP [Dwork et al., 2006b], the divergence $d_{\mathcal{T}}$ is the *multiplicative distance* $\textsc{Mult}(P, Q)$, defined for two distributions P and Q on the same probability space $(\Omega, \mathcal{F})$ as:

$$\textsc{Mult}(P, Q) = \sup\left\{\left|\ln \frac{P(S)}{Q(S)}\right| : S \in \mathcal{F}\right\},$$

where $\frac{0}{0} := 1$.

For approximate (ϵ, δ)-DP [Dwork et al., 2006a], the divergence $d_{\mathcal{T}}$ is the *δ-approximate multiplicative divergence* $\textsc{Mult}^\delta(P, Q)$:

$$\textsc{Mult}^\delta(P, Q) = \sup\left\{\ln \frac{[P(S) - \delta]^+}{Q(S)}, \ln \frac{[Q(S) - \delta]^+}{P(S)}, 0\right\}_{S \in \mathcal{F}},$$

[10] When the ordering of the data is meaningful, the sharpening $d^*_{r_b}$ of the bounded divergence d_{r_b} is the Hamming distance

$$d^u_{\text{Ham}}(\boldsymbol{X}, \boldsymbol{X}') = \begin{cases} \sum_{i=1}^n \mathbb{1}\{X_i \neq X'_i\} = 1 & \text{if } |\boldsymbol{X}| = |\boldsymbol{X}'| = n, \\ \infty & \text{otherwise,} \end{cases}$$

and the sharpening $d^*_{r_u}$ of the unbounded divergence d_{r_u} is given by

$$d^{u*}_{r_u}(\boldsymbol{X}, \boldsymbol{X}') = \min\{|I| + |J| : I \subset \{1, \ldots, |\boldsymbol{X}|\}, J \subset \{1, \ldots, |\boldsymbol{X}'|\}, \boldsymbol{X}_{-I} = \boldsymbol{X}_{-J}\}.$$

[11] Two privacy definitions are equivalent if any mechanism which satisfies one definition satisfies the other with the same privacy budget ϵ. See Section 3.8 for a formal definition.

where

$$[x]^+ = \begin{cases} x & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

For ρ-zero concentrated differential privacy (ρ-zCDP) [Bun and Steinke, 2016a], the divergence $d_{\mathcal{T}}$ is the *normalised Rényi metric* D_{nor}:

$$D_{\text{nor}}(P, Q) = \sup_{\alpha > 1} \frac{1}{\sqrt{\alpha}} \max \left[\sqrt{D_\alpha(P||Q)}, \sqrt{D_\alpha(Q||P)} \right],$$

where D_α is the Rényi divergence of order α:

$$D_\alpha(P||Q) = \frac{1}{\alpha - 1} \ln \int \left[\frac{dP}{dQ} \right]^\alpha dQ,$$

if P is absolutely continuous with respect to Q (where $\frac{dP}{dQ}$ is the Radon-Nikodym derivative of P with respect to Q) and $D_\alpha(P||Q) = \infty$ otherwise.

Note that we re-parametrise ρ so that D_{nor} is a metric [Bailie et al., 2023+]. The parameter ρ in Bun and Steinke [2016a] is equivalent to ρ^2 in our formulation of zCDP.

3.7 Post-processing and composition

A common requirement for differential privacy definitions is closure under post-processing: Any transformation of a differentially private output is also differentially private. More formally, if T satisfies $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ then – for any (possibly randomized) function f – the post-processed mechanism $f \circ T$ also satisfies $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with the same budget.

A second requirement is closure under composition: if T_1 and T_2 satisfy $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with budgets ϵ_1 and ϵ_2, then their composition $\boldsymbol{T} = (T_1, T_2)$ also satisfies $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$, with budget equal to some function $\epsilon(\epsilon_1, \epsilon_2)$. Both these requirements are properties of the divergence $d_{\mathcal{T}}$ only. For example, MULT and D_{nor} satisfy both post-processing and composition with $\epsilon(\epsilon_1, \epsilon_2) = \epsilon_1 + \epsilon_2$ and $\rho(\rho_1, \rho_2) = \sqrt{\rho_1^2 + \rho_2^2}$ respectively. More details can be found in Bailie et al. [2023+].

It is worth noting that, because we are merely explicating the details of different DP definitions – rather than changing them – all their properties found elsewhere in the literature should be preserved under the framework developed in this Section. This includes post-processing and composition, along with other formal privacy properties.

3.8 Equivalence of differential privacy definitions

Fix the input data space $\mathcal{X}$ and the output space $\mathcal{T}$. Write $\mathcal{S}(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}}, \epsilon_{\mathcal{D}})$ for the set of mechanisms T which satisfy $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with privacy budget $\epsilon_{\mathcal{D}}$.

Definition 3.5. A differential privacy $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ *implies* another definition $(\mathscr{D}, d'_{\mathcal{X}}, d'_{\mathcal{T}})$ with accounting $f : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$ if $\mathcal{S}\big(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}}, \epsilon_{\mathcal{D}}\big) \subset \mathcal{S}\big(\mathscr{D}, d'_{\mathcal{X}}, d'_{\mathcal{T}}, f(\epsilon_{\mathcal{D}})\big)$ for all $\epsilon_{\mathcal{D}} \geq 0$.

Equivalently, $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ implies $(\mathscr{D}, d'_{\mathcal{X}}, d'_{\mathcal{T}})$ if all mechanisms T satisfying $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with privacy budget $\epsilon_{\mathcal{D}}$ also satisfy $(\mathscr{D}, d'_{\mathcal{X}}, d'_{\mathcal{T}})$ with budget $f(\epsilon_{\mathcal{D}})$.

Definition 3.6. Two differential privacy definitions are *weakly equivalent* if one implies the other and visa versa (with possibly different accounting functions). Two definitions are *strictly equivalent* (or simply *equivalent*) if they are equivalent with accounting the identity function – that is, if

$$\mathcal{S}\big(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}}, \epsilon_{\mathcal{D}}\big) = \mathcal{S}\big(\mathscr{D}, d'_{\mathcal{X}}, d'_{\mathcal{T}}, \epsilon_{\mathcal{D}}\big)$$

for all $\epsilon_{\mathcal{D}} \geq 0$.

4 The Contextual Nature of the Privacy Loss Budget

Traditionally, privacy evaluations have primarily focused on the budget ϵ, using it as a comprehensive measure of the protection provided by a mechanism. In this Section, we argue that ϵ is solely a *within-system* evaluation of privacy. It can only be interpreted within the context of the relevant differential privacy definition $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$. Between-system evaluations – i.e. comparisons between the privacy of two different mechanisms – cannot be based on their budgets without regard to their underlying privacy definitions.

When spelt out in this way, our argument might sound obvious. Such naïve evaluations are akin to ignoring the difference between dollars and pounds when comparing the (financial) budgets of American and British companies. Nevertheless, there is a tendency when implementing and analysing DP systems to report their budgets, without placing these budgets in the context of the chosen definition $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$. In particular for this paper, contextual thinking is crucial when we compare data swapping with the 2020 Census TopDown algorithm, since their privacy definitions differ on all three components – $\mathscr{D}, d_{\mathcal{X}}$ and $d_{\mathcal{T}}$.

As we will show in this Section, $\mathscr{D}, d_{\mathcal{X}}$ and $d_{\mathcal{T}}$ are all critical for understanding both the substance and the extent of the privacy guarantees. In comparison, the privacy budget ϵ only contributes to understanding the extent of the privacy protection. In this sense, ϵ is essentially meaningless without an accompanying description of $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ to explain the substance of what is being protected. This suggests that ϵ may be, in spite of its prominence, the least important factor in an assessment of a DP implementation.

Perhaps one explanation for the literature's focus on the privacy budget is the legitimate difficulty in understand how the components $\mathscr{D}, d_{\mathcal{X}}$ and $d_{\mathcal{T}}$ connect to real-world privacy considerations. Indeed, much more research is needed in order to qualitatively understand the privacy afforded (or lost) by a DP definition. Further, we need quantitative methods for between-system evaluations

which can rank the strengths of different DP definitions: Are there (non-trivial) criteria which allow us to compare one definition against another? When does strengthening one component cancel out a weakening in another component?

Fortunately, these considerations have recently begun to see more attention. For example, Dwork et al. [2019] makes the case that:

> When meaningfully implemented, DP supports deep data-driven insights with minimal worst-case privacy loss. When not meaningfully implemented, DP delivers privacy mostly in name.

The rub is, of course, understanding what is, and what is not, a meaningfully implementation of DP, as determined by the societal and cultural context of the data release. Nevertheless, this quote hints at another possible (less charitable) explanation for the prominence placed on ϵ: An exclusive focus on the privacy loss budget allows a data custodian to hide important limitations to the actual privacy provided by their implementation of DP. Blanco-Justicia et al. [2022] makes an argument for this explanation, although we are unsure of the extent of this practice. (And indeed it is at least better than keeping the budget secret – or using an absurdly large budget – as the rubber stamp of "differential privacy" on its own is truly meaningless for assessing privacy.)

The rest of this Section consists of some preliminary thoughts on these considerations, some of which have already been raised before by others. We hope that the abstract DP definition $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ from Section 3 provides a coherent framework to organise these thoughts, thereby sparking future work to resolve the questions raised above and improve our ability to assess and compare DP implementations.

While this Section raise a number of complications, the upshot is that it reveals other levers – beyond the budget ϵ – which can be used to trade off privacy and utility for DP mechanisms. This trade-off lies at the heart of any data release and therefore is a fundamental question of SDC. Existing research from the formal privacy perspective has focused on adjusting ϵ as DP's sole measure of privacy [Abowd and Schmutte, 2019]. This Section illustrates that the three components $\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}}$ can also be used – and in fact are already being used in an ad-hoc fashion – to manage this trade-off. Our secondary hope for this Section is therefore to inspire principled privacy-utility trade-offs which use all four components.

4.1 Impact of the data divergence $d_{\mathcal{X}}$ on privacy protection

Consider the typical choice for $d_{\mathcal{X}}$: Set a neighbour relation r on $\mathcal{X}$ and build $d_{\mathcal{X}}$ from r as in (3.3). Defining r requires choosing the privacy unit, which is critically important in assessing the privacy protection provided by a mechanism T. Under this setup, DP protects units in the sense that it bounds the change in $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'})$ for a single change in a single unit. Intuitively, when $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'})$ is small, an attacker cannot distinguish changes in a single unit; equivalently, if $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}) = \epsilon$ for

neighbours $\boldsymbol{X}, \boldsymbol{X}'$, then a single unit is "ϵ-indistinguishable". However, for $\boldsymbol{X}$ and $\boldsymbol{X}'$ which differ by more than a single unit (i.e. $\boldsymbol{X}, \boldsymbol{X}'$ are not neighbours), the DP condition (3.2) is vacuous as $d_r(\boldsymbol{X}, \boldsymbol{X}') = \infty$. This does not mean that DP provides no bound on $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'})$ for $\boldsymbol{X}, \boldsymbol{X}'$ which differ on more than one unit. In many cases, indistinguishability of single units implies protection for multiple units but with a decrease in the level of indistinguishability. (This property is called group privacy. Group privacy with linear decrease in the level of indistinguishability is equivalent to $d_{\mathcal{T}}$ being a metric [Bailie et al., 2023+].)

The converse of this observation hints at a method for artificially shrinking the privacy loss budget: By decreasing the size of the privacy unit, one can decrease ϵ without changing the mechanism T. This means we can add exactly the same amount of noise into the data while reporting that we have increased the level of privacy protection! More generally, inflating the divergence $d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}')$ between datasets $\boldsymbol{X}, \boldsymbol{X}'$ will reduce the budget ϵ. This highlights the necessity of understanding the privacy unit and $d_{\mathcal{X}}$ when assessing the actual privacy afforded by a mechanism.

As an example which is particularly relevant for this paper, using individuals as privacy units provides less privacy protection than using households, as household records are as least as large as individual records. Furthermore, if the privacy unit is an individual, then the privacy protection afforded to a household's characteristics decreases as the number of individuals in that household increases. For this reason, if one wants to protect the privacy of households, the privacy unit must be at least as large as the (theoretically possible) largest household.

The choice of privacy unit in commercial implementations of DP is particularly important. In such settings, a user typically generates data during each interaction with the commercial service (for example, tweeting, liking a post, or even typing out a message or emoji on a phone [Tang et al., 2017]). Since the number of interactions a person can have is theoretically unbounded, it is common to choose the privacy unit as the data record generated by a single interaction, or the data generated by one user over a single day – rather than the user's entire data over time – so that the sensitivity of each unit is controlled [Kenthapadi and Tran, 2018]. Yet this implies the privacy protection for an individual degrades as the individual's interactions increase. Since the user data of an individual is generally correlated, an attacker can infer individual characteristics with high accuracy even if each single data record is DP-protected.

The following Proposition will prove useful in comparing DP definitions which use different divergences $d_{\mathcal{X}}$ on $\mathcal{X}$. Whenever the privacy units of $d'_{\mathcal{X}}$ are nested inside those of $d_{\mathcal{X}}$, the Proposition's assumption – $d_{\mathcal{X}} \leq d'_{\mathcal{X}}$ – is satisfied.

Proposition 4.1. *Let $d_{\mathcal{X}}$ and $d'_{\mathcal{X}}$ be divergences on $\mathcal{X}$ such that $d_{\mathcal{X}} \leq d'_{\mathcal{X}}$. Then $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ implies $(\mathscr{D}, d'_{\mathcal{X}}, d_{\mathcal{T}})$ with accounting function the identity.*

Proof. Let T be a mechanism satisfying $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with privacy budget ϵ. Then

$$d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}) \leq \epsilon_{\mathcal{D}} d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}') \leq \epsilon_{\mathcal{D}} d'_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}'). \qquad \square$$

4.2 Impact of the output divergence $d_{\mathcal{T}}$ on privacy protection

Out of the three components, $d_{\mathcal{T}}$'s importance is the most widely recognised in the literature, as evidenced by the use of different letters to denote the privacy budget for different $d_{\mathcal{T}}$. For example, ρ is used in place of ϵ to denote the privacy budget when $d_{\mathcal{T}} = D_{\text{nor}}$ (zero-concentrated DP). Understanding what $d_{\mathcal{T}}$ means is crucial for putting the privacy budget in context. Just as we can inflate $d_{\mathcal{X}}$, we can artificially shrink $d_{\mathcal{T}}$ (for example by dividing it by some large number) to achieve nominal improvements in the privacy budget.

Perhaps more importantly, $d_{\mathcal{T}}$ formalizes what is meant by the notion of indistinguishability between neighbouring datasets $\boldsymbol{X}, \boldsymbol{X}'$. Small values of $d_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{X}')$ should mean that an attacker has difficulty in distinguishing the single unit change between $\boldsymbol{X}$ and $\boldsymbol{X}'$. Yet to make this precise, we need to know how the attacker will infer this unit – i.e. we need to assume the attacker's statistical inference framework. The choice of framework is not without controversy [Gong et al., 2023+]. (Existing research has measured indistinguishability as bounds on frequentist hypothesis testing, Bayesian posterior-to-posterior and prior-to-posterior semantics.) See Kifer et al. [2022] for a survey of the semantics of indistinguishability under common choices of $d_{\mathcal{T}}$ and Wasserman and Zhou [2010], Bailie and Gong [2023+], Gong et al. [2023+] for $d_{\mathcal{T}} = \textsc{Mult}$.

Proposition 4.2. *Let $d_{\mathcal{T}}$ and $d'_{\mathcal{T}}$ be divergences on $\mathcal{L}(\mathcal{T})$ such that $d_{\mathcal{T}} \geq d'_{\mathcal{T}}$. Then $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ implies $(\mathscr{D}, d_{\mathcal{X}}, d'_{\mathcal{T}})$ with accounting function the identity.*

The proof of this Proposition is analogous to that of Proposition 4.2. Since $\textsc{Mult} > \textsc{Mult}^{\delta}$ for any $\delta > 0$, this Proposition shows that pure ϵ-DP is stronger than approximate (ϵ, δ)-DP. ϵ-DP also dominates zCDP because $D_{\text{nor}} \leq 2^{-0.5}\textsc{Mult}$ [Bun and Steinke, 2016b, Proposition 3.3].

4.3 Impact of the data universe $\mathscr{D}$ on privacy protection

The data universe function $\mathscr{D}$ is a reminder that the range of possible datasets over which the privacy guarantee applies is a matter of the data curator's choice. Therefore, the interpretation of the privacy guarantee must be contextually situated within that choice as well. In Definition 3.4, the privacy guarantee acknowledges the data universe $\mathcal{D}$ explicitly via the subscript to the privacy loss budget: $\epsilon_{\mathcal{D}}$. In the following two Propositions, we will see that the interpretation of the value of ϵ cannot be isolated from $\mathcal{D}$, and indeed this complicates the comparison of privacy loss budgets across different applications. For these two results, fix a data space $\mathcal{X}$ and invariants $\boldsymbol{c} : \mathcal{X} \to \mathbb{R}^l$. Let the invariant-compliant universe $\mathscr{D}_{\boldsymbol{c}}$ be defined as in equation (3.1).

Proposition 4.3. *For any $d_{\mathcal{X}}$ and $d_{\mathcal{T}}$, the mechanism $T(\boldsymbol{X}) = \boldsymbol{c}(\boldsymbol{X})$ that releases the invariants exactly satisfies $(\mathscr{D}_{\boldsymbol{c}}, d_{\mathcal{X}}, d_{\mathcal{T}})$ with privacy budget $\epsilon_{\mathcal{D}} = 0$.*

Now suppose $d_{\mathcal{T}}(P, Q) = \infty$ if $d_{\text{TV}}(P, Q) = 1$.[12] Let $\mathscr{D}$ be a data universe function such that there

[12] We write d_{TV} to denote the total variation distance. $d_{\text{TV}}(P, Q) = 1$ means that the probability measures P and Q have no common support. The assumption $d_{\text{TV}}(P, Q) = 1 \Rightarrow d_{\mathcal{T}}(P, Q) = \infty$ is satisfied by all common choices of $d_{\mathcal{T}}$.

exists datasets $\boldsymbol{X}_1, \boldsymbol{X}_2$ *in some data universe* $\mathcal{D}_0 \in \operatorname{Im} \mathscr{D}$ *with* $d_{\mathcal{X}}(\boldsymbol{X}_1, \boldsymbol{X}_2) < \infty$ *and* $\boldsymbol{c}(\boldsymbol{X}_1) \neq \boldsymbol{c}(\boldsymbol{X}_2)$. *Then* T *does not satisfy* $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ *for any* $\epsilon_{\mathcal{D}_0} < \infty$.

The results of Proposition 4.3 also hold if $\mathscr{D}$ is any data universe function with $\boldsymbol{c}$ constant within all $\mathcal{D} \in \operatorname{Im} \mathscr{D}$.

Proof. T is constant within data universes $\mathcal{D}$. Therefore $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}) = 0$ for all $\boldsymbol{X}, \boldsymbol{X}' \in \mathcal{D}$. This proves the first half of the Proposition. To prove the second half, observe that $d_{\mathcal{T}}(P_{\boldsymbol{X}_1}, P_{\boldsymbol{X}_2}) = \infty$ but $d_{\mathcal{X}}(\boldsymbol{X}_1, \boldsymbol{X}_2) < \infty$. □

The following result is the converse of Proposition 4.3.

Proposition 4.4. *Suppose that a mechanism* T *varies within some universe* $\mathcal{D}_0 \in \operatorname{Im} \mathscr{D}_{\boldsymbol{c}}$ *in the sense that there exists* $\boldsymbol{X}, \boldsymbol{X}' \in \mathcal{D}_0$ *with* $d_{\mathcal{X}}(\boldsymbol{X}, \boldsymbol{X}') < \infty$ *but* $P_{\boldsymbol{X}} \neq P_{\boldsymbol{X}'}$.

When $d_{\mathcal{T}}$ *is a metric,* T *satisfies* $(\mathscr{D}_{\boldsymbol{c}}, d_{\mathcal{X}}, d_{\mathcal{T}})$ *only if* $\epsilon_{\mathcal{D}_0} > 0$.

Proof. This Proposition relies on the metric axiom $d_{\mathcal{T}}(P, Q) > 0$ if $P \neq Q$. This implies $d_{\mathcal{T}}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}) > 0$. □

These two Propositions demonstrate that for privacy with invariants, it is necessary and sufficient to restrict the data space via $\mathscr{D}_{\boldsymbol{c}}$. Necessity follows from the second half of Proposition 4.3: if there are datasets $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathcal{D}_0$ with different values on the invariants, then releasing the invariants exactly would require $\epsilon_{\mathcal{D}} = \infty$. Sufficiency is described in two parts: 1) the invariants can be released exactly without privacy loss (by the first half of Proposition 4.3); but 2) any additional information (not logically equivalent to the invariants) cannot be released without incurring privacy loss (by Proposition 4.4).

As a concrete example of how the meaning of ϵ changes with $\mathscr{D}$, consider evaluating the same mechanism T against two definitions $(\mathscr{D}_{\boldsymbol{c}}, d_{\mathcal{X}}, d_{\mathcal{T}})$ and $(\mathscr{D}_{\boldsymbol{c}'}, d_{\mathcal{X}}, d_{\mathcal{T}})$ which differ only on their invariants. Suppose the first of invariants are nested within the second; that is, $\boldsymbol{c}'$ is strictly more constraining than $\boldsymbol{c}$. (For example, $\boldsymbol{c}$ are population counts at the county level and $\boldsymbol{c}'$ are counts at the block level.) Then T's budget ϵ under $\mathscr{D}_{\boldsymbol{c}}$ may be strictly larger than T's budget under $\mathscr{D}_{\boldsymbol{c}'}$ (and in fact can not be smaller). We have already alluded to this result when discussing Figure 2.1, and will encounter it again in concrete terms in Section 6.2. As we repeatedly emphasize, it is dangerous to think that the $\boldsymbol{c}$-release is indeed afforded with less privacy protection than the $\boldsymbol{c}'$-release because there is privacy leakage due to specifying additional invariants, which is not captured by the within-system privacy evaluation ϵ. Indeed in the extreme example where $\boldsymbol{c}$ is an injective function so that the universes $\mathcal{D}$ are singletons, there is no privacy protection afforded by $(\mathscr{D}_{\boldsymbol{c}}, d_{\mathcal{X}}, d_{\mathcal{T}})$ regardless of the choices of $d_{\mathcal{X}}, d_{\mathcal{T}}$ and $\epsilon_{\mathcal{D}}$. This point is crucial to understanding the comparative analysis between our swapping algorithm and the 2020 TDA as presented in Section 6.

Proposition 4.5. *Suppose that* $\mathscr{D}$ *and* $\mathscr{D}'$ *are nested in the sense that* $\mathscr{D}'(\boldsymbol{X}) \subset \mathscr{D}(\boldsymbol{X})$ *for all* $\boldsymbol{X} \in \mathcal{X}$. *Then* $(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ *implies* $(\mathscr{D}', d_{\mathcal{X}}, d_{\mathcal{T}})$.

5 Data Swapping and Differential Privacy

5.1 What invariants does swapping preserve?

The goal of this Section is to show that data swapping satisfies DP where $\mathscr{D}$ is induced by some invariants $\boldsymbol{c}$. Specifically, we are interested in examining swapping under the following differential privacy definition.

Definition 5.1. Let $\boldsymbol{c} : \mathcal{X} \to \mathbb{R}^l$ be some invariants. Write $(\boldsymbol{c}, d_{\mathcal{X}}, \epsilon_{\mathcal{D}})$-DP as shorthand for the differential privacy definition $(\mathscr{D}_{\boldsymbol{c}}, d_{\mathcal{X}}, \textsc{Mult})$ with privacy budget $\epsilon_{\mathcal{D}}$.

In this Section, we will determin the invariants $\boldsymbol{c}$ of swapping by examining what swapping does, and does not, change in the data. Swapping is, very loosely, a synthetic data generation mechanism. Given a dataset $\boldsymbol{X}$ as input, swapping produces a sanitised (i.e. privacy protected) version $\boldsymbol{Z}$ of $\boldsymbol{X}$. Both $\boldsymbol{X}$ and $\boldsymbol{Z}$ have the same variables – we denote the set of variables as $\boldsymbol{V}$ – and the same number of records.

As introduced in Section 2, the *swapping variables* $\boldsymbol{V}_{\text{Swap}}$ and the *holding variables* $\boldsymbol{V}_{\text{Hold}}$ form a non-empty partition of the variable set. In practice, variables in $\boldsymbol{V}_{\text{Swap}}$ are typically sensitive variables [Fienberg and McIntyre, 2004], and variables in $\boldsymbol{V}_{\text{Hold}}$ are typically quasi-identifiers. Pairs of records are randomly selected and their swapping variables are interchanged. The two categories are 'duals' of each other, in the sense that they can be interchanged without modifying the behaviour of the swapping algorithm. Both categories must be non-empty; otherwise swapping is vacuous in the sense that it will not change the data, except perhaps by re-ordering rows.[13] Matching variables $\boldsymbol{V}_{\text{Match}} \subset \boldsymbol{V}_{\text{Hold}}$ (also referred to as the *swap key* [McKenna, 2018, Abowd and Hawes, 2023]) define strata in the database in which the swapping operation is restricted and are often key characteristics about the underlying population. Note that it is permissible for $\boldsymbol{V}_{\text{Match}}$ to be empty, so that any two records can be swapped without matching on any characteristics. Typically, however, some non-trivial matching variables are defined for data utility reasons, which we illustrated below.

Example 5.2. Simplification of the disclosure avoidance system (DAS) for the 2010 US Decennial Census:[14] In the 2010 Census, the swapping units are individual households. (We will show later that this implies the privacy unit for the 2010 DAS are households.) This means a record in the data $\boldsymbol{X}$ correspond to a household and each swap interchanges $\boldsymbol{V}_{\text{Swap}}$ between two households. The matching variables $\boldsymbol{V}_{\text{Match}}$ include both the number of voting age persons and the total number of persons in the household. $\boldsymbol{V}_{\text{Match}}$ also includes a geographic variable V_g, either the Census tract, county or state of the household. $\boldsymbol{V}_{\text{Swap}}$ are the geographic variables nested underneath V_g. (See US Census Bureau [2021] for a description of the geographic hierarchy of the Decennial Census.) For

[13] A swap is vacuous if and only if the two records share the same values of $\boldsymbol{V}_{\text{Swap}}$ or the same values of $\boldsymbol{V}_{\text{Hold}}$. Thus, to guarantee no vacuous swaps one must choose records which disagree on both $\boldsymbol{V}_{\text{Swap}}$ and $\boldsymbol{V}_{\text{Hold}}$.

[14] This example is based on publicly-available information about the 2010 DAS. This information is incomplete as some implementation details have been deemed confidential due to concerns that they may allow the privacy protection to be undone.

example if V_g is the county, then $\boldsymbol{V}_{\text{Swap}}$ is the block and tract of the household. All other variables are holding variables – in particular, the household and person characteristics. One can imagine the 2010 DAS as digging up pairs of houses of the same size in the same geographic area and swapping their locations but not changing the house and its occupants.

In the 2010 DAS, each household is assigned a risk score based on the USCB's assessment of how unique the household is within its neighbourhood. These risk scores are used to compute each household's probability of being swapped. Every household has a non-zero swap probability. Selected households are then swapped with one of their neighbours.

After swapping, a record $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}})$ becomes $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}'_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}})$ – i.e. only the values of the swapping variables change. Thus, any statistic generated by only $\boldsymbol{V}_{\text{Hold}}$ is preserved by swapping, since the pairs $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}})$ of each record are themselves preserved by swapping. Moreover, since $\boldsymbol{V}_{\text{Match}}$ is equal between swapped records, any statistic generated by only $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Swap}})$ is also preserved by swapping. Thus, only statistics generated by $(\boldsymbol{V}_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}})$ are not preserved by swapping. In the case of discrete variables (which is our primary concern), we can be more exact in describing the invariants of swapping:

Proposition 5.3. *Suppose that all variables in $\boldsymbol{X}$ are discrete.*[15] *Without loss of generality, we may assume that each of $\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}$ are singletons (otherwise, cross-classify each set of variables into a single variable). Denote the categories of the matching variable $\boldsymbol{V}_{\text{Match}}$ by $j = 1, \dots, \mathcal{J}$. Similarly let $k = 1, \dots, \mathcal{K}$ and $l = 1, \dots, \mathcal{L}$ be the categories of $\boldsymbol{V}_{\text{Swap}}$ and $\boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}$ respectively.*

The dataset $\boldsymbol{X}$ can be represented as a 3-dimensional contingency table $H(\boldsymbol{X}) = [n^{\boldsymbol{X}}_{jkl}]$ of counts in each combination of categories j, k, l. (We will omit the superscript $^{\boldsymbol{X}}$ when it is clear from the context.) No interior cell count n_{jkl} is preserved under swapping. But all margins – except $n_{\cdot kl} = \sum_j n_{jkl}$ – are invariant under swapping.

To simplify the notation, whenever a subscript is omitted, it means the corresponding position has been summed over, such as

$$n_j = \sum_{k,l} n_{jkl}, \quad n_k = \sum_{j,l} n_{jkl}, \quad n_{jk} = \sum_{l} n_{jkl}, \quad n_{lk} = \sum_{j} n_{jkl}.$$

This will of course cause ambiguity when specific values are used, such as n_1, in which case we will use the full notation $n_{1\cdot\cdot}$ or $n_{\cdot 1\cdot}$ for the first two, for example.

Proof. Swapping pairs a record a in categories jkl with a record b in $jk'l'$. It moves a to $jk'l$ and b to jkl'. The matching category j is the same in a and b by construction.[16] After the swap n_{jkl} and

[15] Swapping only requires that the matching variable is discrete. $\boldsymbol{V}_{\text{Swap}}$ and $\boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}$ may be continuous. However, for simplicity we focus on the main motivating example of releasing contingency tables, in which case all variables must be discrete. All of the variables in both the 2010 and 2020 US Decennial Censuses are discrete.

[16] We do allow the possibility that $k = k'$ or $l = l'$ but in either of these cases the swap is vacuous in the sense that the contingency table $[n_{jkl}]$ does not change.

$n_{jk'l'}$ decrease by one, and $n_{jk'l}$ and $n_{jkl'}$ increase by one. Hence, the margins $n_{jl} = \sum_k n_{jkl}$ and $n_{jk} = \sum_l n_{jkl}$ are invariant under swapping. □

Example 5.2 (continued). In the 2010 US Census DAS, the number of adults, children and households in each block are invariant. (This is the n_{jk} margin.) The counts of all the person and household characteristics inside each V_g are also invariant. (This is the n_{jl} margin.) For example, if V_g is the county, then the aggregate characteristics at the county level remain unchanged by swapping, but these aggregates at the block and tract level are perturbed.

Definition 5.4. Suppose that all variables in $\boldsymbol{X}$ are discrete. Using the notation in Proposition 5.3, define the *swapping invariants* $\boldsymbol{c}_{\text{Swap}}(\boldsymbol{X})$ for a given choice of $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}})$ as the margins n_{jl} and n_{jk}:

$$\boldsymbol{c}_{\text{Swap}}(\boldsymbol{X}) = \begin{bmatrix} n_{1\cdot 1} \\ n_{1\cdot 2} \\ \vdots \\ n_{\mathcal{J}\cdot\mathcal{L}} \\ n_{11\cdot} \\ n_{12\cdot} \\ \vdots \\ n_{\mathcal{J}\mathcal{K}\cdot} \end{bmatrix}.$$

Example 5.5. In the 2020 TDA, there are two invariants (ignoring group quarters and structural zeroes for simplicity): 1) the number of people in each state; and 2) the number of households in each block [U.S. Census Bureau, 2021c]. We cannot design a swapping algorithm which preserves these – and only these – invariants. In other words, the 2020 US Census invariants do not correspond to any swapping invariants $\boldsymbol{c}_{\text{Swap}}$, regardless of the choice of $(\boldsymbol{V}_{\text{Match}}, \boldsymbol{V}_{\text{Swap}}, \boldsymbol{V}_{\text{Hold}})$. Why? Swapping always preserves the one-dimensional marginals: n_j, n_k and n_l; but the 2020 US Census DAS does not. For example, the number of females in the US is not invariant in the 2020 Redistricting Data Summary File but it must necessarily be invariant under any swapping algorithm. If we wanted to remove some of the swapping invariants, we could apply additional privacy protections (e.g. noise infusion) before or after swapping. (These and other extensions to data swapping are discussed further in Section 7.2.)

5.2 Swapping satisfies pure ϵ-DP conditioning on its invariants

From herein, follow the assumptions of Proposition 5.3: Assume that all variables in $\boldsymbol{X}$ are discrete; and that there is one matching variable, one swapping variable and one non-matching holding variable, with categories j, k and l respectively.

In this Section, we will design a specific data swapping algorithm – termed the *Permutation Algorithm* to distinguish it from other data swapping methods – which satisfies $(\boldsymbol{c}_{\text{Swap}}, d^u_{\text{HamS}}, \epsilon_{\mathcal{D}})$-DP.

Here the privacy unit u of d^u_{HamS} is given by the level of the data swapping, which can vary between implementations of the algorithm. We do not claim that this is the swapping algorithm used by the 2010 DAS but we do believe it reflects the essential features of the 2010 DAS. However, certain aspects of the Permutation Algorithm were made with the specific goal of satisfying DP. For example, DP cannot be satisfied if the number of swaps is fixed.[17] To see this, suppose $\boldsymbol{X}, \boldsymbol{X}'$ differ by a single swap and the same number of swaps m are applied to both $\boldsymbol{X}$ and $\boldsymbol{X}'$ to produce $\boldsymbol{Z}$ and $\boldsymbol{Z}'$ respectively. It is possible that $m+1$ swaps are necessary to produce $\boldsymbol{Z}$ from $\boldsymbol{X}'$. In this case, $\Pr(\boldsymbol{Z}|\boldsymbol{X}) > 0$ but $\Pr(\boldsymbol{Z}|\boldsymbol{X}') = 0$. More generally, a necessary condition for a swapping procedure to be DP (with $d_{\mathcal{T}}$ a metric) is that every orbit space is a superset of its data universe: $G \cdot \boldsymbol{X} \supset \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X})$, where G is the set of all possible swaps that are permissible by the swapping procedure.[18]

To avoid this complication, the Permutation Algorithm randomly permutes the swapping variables $\boldsymbol{V}_{\text{Swap}}$ of records in the same matching category j. (A permutation is simply multiple swaps done one after the other.) Since we do not want to permute every record, each record is selected for permutation independently with probability p. To avoid the case where the random permutation leaves a record fixed, we sample uniformly at random from the set of all derangements.[19]

Pseudocode for the Permutation Algorithm is provided in Algorithm 5.1. The output is a contingency table $H(\boldsymbol{Z}) = [n^{\boldsymbol{Z}}_{jkl}]$ (i.e. a 3-way tensor) computed on the swapped dataset $\boldsymbol{Z}$.[20]

Theorem 5.6. *Let*

$$b = \max\{0, n_j \mid \text{there are at least two different records in stratum } j\}.$$

Then Algorithm 5.1 is $(\boldsymbol{c}_{\text{Swap}}, d^u_{\text{HamS}}, \epsilon_{\mathcal{D}})$*-DP where* d^u_{HamS} *is the Hamming distance defined in* (3.4)*,* $\epsilon_{\mathcal{D}} = 0$ *if* $b = 0$ *and otherwise*

$$\epsilon_{\mathcal{D}} = \begin{cases} \ln(b+1) - \ln o & \text{if } 0 < p \leq 0.5 \\ \max\left\{ \ln o, \ln(b+1) - \ln o \right\} & \text{if } 0.5 < p < 1, \end{cases} \tag{5.1}$$

with $o = p/(1-p)$*. On the other hand, for* $p \in \{0, 1\}$ *and for some* $\mathcal{D}$ *with* $b > 0$*, Algorithm 5.1 does not satisfy* $(\boldsymbol{c}_{\text{Swap}}, d^u_{\text{HamS}}, \epsilon_{\mathcal{D}})$*-DP for any finite* $\epsilon_{\mathcal{D}}$*.*

The units for the Permutation Algorithm's privacy definition $(\boldsymbol{c}_{\text{Swap}}, d^u_{\text{HamS}}, \epsilon_{\mathcal{D}})$ are given by the type of records which are being swapped. For example, if Algorithm 5.1 swaps person-level records,

[17] To be clear, based on the available public information, we do not believe the 2010 DAS fixes the number of swaps, although we have not been able to confirm this.

[18] The other direction $G \cdot \boldsymbol{X} \subset \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X})$ is a consequence of Proposition 5.3.

[19] A derangement is a permutation with no fixed points – i.e. a function $\sigma : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $\sigma(j) \neq j$ for all j.

[20] $H(\boldsymbol{Z}) = [n^{\boldsymbol{Z}}_{jkl}]$ can also be expressed as a collection of $\mathcal{J}$ matrices $H_j(\boldsymbol{Z}) = [n^{\boldsymbol{Z}}_{jkl}], j = 1, \ldots, \mathcal{J}$, each of which has dimension $\mathcal{K} \times \mathcal{L}$. The contingency table $H(\boldsymbol{Z})$ fully determines $\boldsymbol{Z}$ up to re-ordering of the rows of $\boldsymbol{Z}$. The Permutation Algorithm could output the swapped dataset $\boldsymbol{Z}$ directly, without changing any of its privacy guarantees. For an explanation, see (A.2).

Algorithm 5.1: The Permutation Algorithm

Input: Dataset $\boldsymbol{X}$

1: **for** $j = 1, \ldots, \mathcal{J}$ **do**
2: **if** $n_j = 0$ or $n_j = 1$ **then**
3: **continue**
4: **end if**
5: **for** record i with category j **do**
6: Select i with probability p
7: **end for**
8: **if** 0 records selected **then**
9: **continue**
10: **else if** exactly 1 record selected **then**
11: **go to** line 5
12: **end if**
13: Sample uniformly at random a derangement σ of the selected records.
14: /* *Permute the swapping variable of the selected records according to* σ*:* */
15: Save copy $\boldsymbol{X}_0 \leftarrow \boldsymbol{X}$ before permutation
16: Let $k^{\boldsymbol{X}}(i)$ be the value of the swapping variable of record i in dataset $\boldsymbol{X}$.
17: **for all** selected records i **do**
18: Set $k^{\boldsymbol{X}}(i) \leftarrow k^{\boldsymbol{X}_0}(\sigma(i))$
19: **end for**
20: **end for**
21: Set $\boldsymbol{Z} \leftarrow \boldsymbol{X}$ to be the swapped dataset.
22: **return** contingency table $[n^{\boldsymbol{Z}}_{jkl}]$

then the privacy unit for d^u_{HamS} is persons. If instead it swaps household-level records (as was done in the 2010 Census DAS and in our applications below), then the privacy unit of d^u_{HamS} are households.

The proof of Theorem 5.6 is in Appendix A. A broad sketch for the case $0 < p \leq 0.5$ (i.e. $o \leq 1$) is given here: We need to show that, for fixed datasets $\boldsymbol{X}, \boldsymbol{X}', \boldsymbol{Z}$ in the same data universe $\mathcal{D}$,

$$\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{Z}) \leq \exp(m\epsilon) \Pr(\sigma'(\boldsymbol{X}') = \boldsymbol{Z}),$$

where $m = d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}')$ and the probability is over the random sampling of σ and σ' in Algorithm 5.1. We can show that there exists a derangement ρ of m records such that $\boldsymbol{X} = \rho(\boldsymbol{X}')$. There is a bijection between the possible σ and σ' given by $\sigma' = \sigma \circ \rho$. If m_σ is the number of records deranged by σ, we have

$$m_\sigma - m \leq m_{\sigma'} \leq m_\sigma + m. \tag{5.2}$$

This gives a bound on $\Pr(\sigma)/Pr(\sigma')$ in terms of $o^{m_\sigma - m_{\sigma'}}$ and the ratio between the number of derangements of $m_{\sigma'}$ and of m_σ. For $o \leq 1$, this can be bounded by $o^{-m}(b+1)^m$ using the inequality (5.2). The result for $0 < p \leq 0.5$ then follows with some algebraic simplification.

Table 5.1: A comparison of two-way tabulations of dwelling ownership by county based on the 1940 Census full count for the state of Massachusetts (left) and one instantiation of Algorithm 5.1 at $p = 50\%$ (right). Total dwellings per county, as well as total owned versus rented units per state, are invariant. All invariants induced by the Algorithm are not shown.

county	owned	rented	total	owned (swapped)	rented (swapped)	total (swapped)
Barnstable	7461	3825	11286	5907	5379	11286
Berkshire	14736	18417	33153	13770	19383	33153
Bristol	33747	63931	97678	35537	62141	97678
Dukes	1207	534	1741	946	795	1741
Essex	53936	81300	135236	52631	82605	135236
Franklin	7433	6442	13875	6337	7538	13875
Hampden	30597	58166	88763	32267	56496	88763
Hampshire	9427	8630	18057	8145	9912	18057
Middlesex	104144	147687	251831	100372	151459	251831
Nantucket	593	432	1025	471	554	1025
Norfolk	44885	40285	85170	38566	46604	85170
Plymouth	24857	23882	48739	21549	27190	48739
Suffolk	49656	176553	226209	67357	158852	226209
Worcester	53126	78535	131661	51950	79711	131661
total	435805	708619	1144424	435805	708619	1144424

5.3 Numerical demonstration: 1940 Census full count data

We demonstrate Algorithm 5.1 using the 1940 Decennial Census full count data.[21] For the 1940 Census, the smallest geography level is county, hence swapping is performed among household units across counties within each state, where each household's county indicator is set to be $\boldsymbol{V}_{\text{Swap}}$. The matching variables (or swap key) $\boldsymbol{V}_{\text{Match}}$ are the number of persons per household and the household's state. Our analysis is focused on the ownership status of household dwellings, an indicator variable taking value of either owned (including on loan) or rented. This is our $\boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}$. The invariants $\boldsymbol{c}_{\text{Swap}}$ induced by this swapping scheme include 1) the total number of owned versus rented dwellings at each of the household sizes at the state level, and 2) the total number of dwellings at each of the household sizes at the county level. In our notation, these are the n_{jl}'s and the n_{jk}'s, respectively.

We restrict our illustration to the state of Massachusetts. Table 5.1 compares the two-way tabulations of dwelling ownership by county based on the original data and one instantiation of the swapping mechanism using a high swap rate of $p = 50\%$. The row margin of either table is the county-level total dwellings and is invariant due to $n_k = \sum_j n_{jk}$. The column margin is the total number of owned versus rented dwellings in Massachusetts and is invariant due to $n_l = \sum_j n_{jl}$.

Table 5.2 supplies the conversion between different swap rates to the privacy loss budget ϵ. Under the

[21] The data is obtained from IPUMS USA Ancestry Full Count Database [Ruggles et al., 2021].

Table 5.2: Conversion of swap rate to ϵ (PLB). Under this swapping scheme, the largest stratum size is $b = 264,331$, the number of all two-person households of Massachusetts.

swap rate	0.01	0.05	0.10	0.50
ϵ	17.08	15.43	14.68	12.48

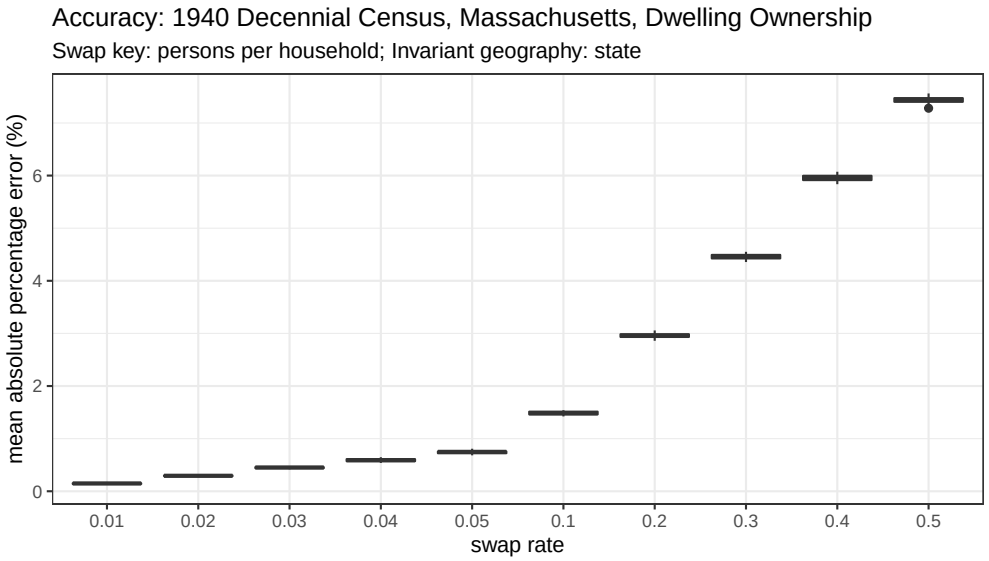

Figure 5.1: Mean absolute percentage error (MAPE) in the two-way tabulation of dwelling ownership by county induced by Algorithm 5.1 applied to the 1940 Census full count data of Massachusetts, at different swap rates from 1% to 50%. Each boxplot reflects 20 independent runs of Algorithm 5.1 at that swap rate.

current swapping scheme, the largest category size delineated by $\boldsymbol{V}_{\text{Match}}$ is $b = 264,331$, consisting of all two-person households of Massachusetts. Therefore by (5.1), we see that a low swap rate of 1% corresponds to an ϵ of 17.08, whereas a high swap rate of 50% corresponds to an ϵ of 12.48. It is worth noting that since $\boldsymbol{c}_{\text{Swap}}$ is fixed in this analysis, the different values of ϵ presented in this Table can be directly interpreted as privacy guarantees of different quantified strengths.

We also examine the accuracy of the two-way tabulation as a function of swap rate. Figure 5.1 shows the mean absolute percentage error (MAPE) in the two-way tabulation induced by swapping at different swap rates from 1% to 50%.[22] The variability across runs is small: each boxplot reflects 20 independent runs of Algorithm 5.1.

[22]The mean absolute percentage error of a swapped table from its true table is defined as the cell-wise average of the ratio between their absolute differences and the true table values. The MAPE in Figure 5.1 is with respect to the contingency table of county by dwelling ownership in Massachusetts and is defined in the notation of Section 5.1 as

$$\frac{1}{\mathcal{K}\mathcal{L}} \sum_{k,l} \frac{|n_{kl}^{\boldsymbol{X}} - n_{kl}^{\boldsymbol{Z}}|}{n_{kl}^{\boldsymbol{X}}},$$

where $\boldsymbol{X}$ is the true table, $\boldsymbol{Z}$ is the swapped table, k is the county indicator and l whether the house was rented or owned.

The accuracy assessment we demonstrate here is highly limited. The analysis above assesses only cell-wise departures of the swapped two-way marginal table from its confidential counterpart. It does not capture potential loss of data utility in terms of multivariate relational structures. It is well understood in the literature that swapping erodes the correlation between $\boldsymbol{V}_{\text{Swap}}$ and $\boldsymbol{V}_{\text{Hold}} - \boldsymbol{V}_{\text{Match}}$; see e.g. Slavković and Lee [2010]. For the current example, this means the county-wide characteristics of household dwellings (other than their size) are not preserved, but other multivariate relationships are. While an in-depth investigation into the utility of swapping is out of scope for this paper, we return to the subject of data utility in Section 7 to discuss the implication this work may have on that line of inquiry.

6 Implications for the US Decennial Censuses

One of the main motivations for developing a framework that unifies different DP definitions (as in Section 3) is to facilitate a comparison between swapping and the differentially private DAS for the 2020 Decennial Census. This Section is dedicated to this comparison.

A main component of the 2020 DAS is the TopDown algorithm [Abowd et al., 2022], used for the production of privacy-protected 2020 Census Redistricting Data (P.L. 94-171) Summary and the Demographic and Housing Characteristics (DHC) Files.[23] In addition, it has been announced that the SafeTab algorithm [Tumult Labs, 2022] will be used to produce the privacy-protected Detailed DHC Files and has been implemented in the Detailed DHC-A Proof of Concept [US Census Bureau, 2023f].[24]

Table 6.1 provides an overview of the comparison we make in this Section. It lays out the choices of $d_{\mathcal{T}}$, $d_{\mathcal{X}}$, privacy unit, invariants, and privacy loss budget that pertain to the TopDown algorithm, the proposed SafeTab algorithm, for comparison against a hypothetical application of swapping to the 2020 Decennial Census. Sections 6.1 and 6.2 respectively lay out the details that lead up to the elements reported in this Table.

[23] The P.L. 94-171 dataset was released on August 12, 2021 [U.S. Census Bureau, 2021a]. The Demographic Profile and DHC are scheduled for release in May 2023 [U.S. Census Bureau, 2022].

[24] The Detailed DHC-A Proof of Concept was released on January 31, 2023. The Detailed DHC-A is planned for release in August 2023. At the time of writing, the release dates for the Detailed DHC-B File and the Supplemental DHC File are to be determined [U.S. Census Bureau, 2022].

	$d_{\mathcal{T}}$	$d_{\mathcal{X}}$ (Unit)	Invariants	Privacy Loss Budget
TopDown	D_{nor}	d^{p}_{HamS} (person)[25]	Population (state) Total housing units (block) Occupied group quarters (block)[27] Structural zeros[29]	PL & DHC: $\rho = 15.29$[26] $\epsilon = 52.83$ ($\delta = 10^{-10}$)[28] See Table 6.2
SafeTab	D_{nor}	d^{p}_{HamS} (person)	None[30]	DDHC-A: $\rho = 19.776$[31] DDHC-B & S-DHC: *TBD.*
Swapping	MULT	d^{h}_{HamS} (household)	Varies but greater than TDA;[33] see Section 6.2	ϵ between 9.37-19.38[32] See Table 6.3

Table 6.1: A comparison of the DP definitions of the TopDown algorithm [Abowd et al., 2022], the SafeTab algorithm [Tumult Labs, 2022], and hypothetically applying swapping (Algorithm 5.1) to the 2020 Decennial Census.

6.1 Privacy analysis of the TopDown algorithm (TDA)

This Section provides a privacy definition for the TopDown algorithm. We prove (in Theorem 6.1) that TDA satisfies zero concentrated DP (zCDP) [Bun and Steinke, 2016a] when conditioning on the invariants it induces. We also show that it is necessary to restrict the data universe according to these invariants: TDA cannot satisfy ρ-zCDP for any finite ρ without conditioning on these invariants.

TDA is summarised in Algorithm 6.1. Briefly, it is a two step procedure: The first step (called the "measurement phase" in Abowd et al. [2022]) produces Noisy Measurement Files (NMF) $\boldsymbol{T}_p(\boldsymbol{X}_p)$ and $\boldsymbol{T}_h(\boldsymbol{X}_h)$. The NMF are privacy-protected versions of tabular summaries $\boldsymbol{Q}_p(\boldsymbol{X}_p)$ (at the person level)

[25] See Section 6.1 for a discussion regarding the choice of privacy units in the TopDown algorithm.

[26] To avoid confusion, we use the parametrisation of ρ common in the literature, which is equal to ρ^2 under the parametrisation of zCDP using D_{nor} as given in Section 3.6.

[27] Counts for each type of occupied group quarters (e.g. correctional facilities, university housing, military quarters, etc.) were held invariant at the block level when producing the redistricting data and DHC files with TDA.

[28] Using the conversion $\epsilon = \rho + 2\sqrt{-\rho \ln \delta}$ given in Bun and Steinke [2016a] and adopted by the USCB.

[29] The complete description of the 2020 Census TopDown invariants can be found in Section 5.2 of Abowd et al. [2022] (for the PL file) and Population Reference Bureau and U.S. Census Bureau's 2020 Census Data Products and Dissemination Team [2023] (for the DHC and Demographic Profile).

Note that structural zeros are classified as edit constraints, not invariants, in Abowd et al. [2022]. However, we recognize that structural zeros, such as what record values are deemed impossible, are contextual in nature as well, hence we classify them as invariants under the current framework. The distinction is between restricting $\mathcal{X}$ (the set of all theoretically-possible datasets) or restricting the data universe $\mathcal{D}$. If a function is constant on $\mathcal{X}$ then it can be considered as an edit constraint; otherwise it needs to be encoded as an invariant. So that $\mathcal{X}$ remains constant over time (as much as possible), we advocate for considering structural zeros as invariants.

[30] As far as we are aware at the time of writing but production settings for this algorithm have not yet been released.

[31] This is the privacy budget allocation reported in the Census Bureau's Detailed DHC-A Proof of Concept [US Census Bureau, 2023f]. It should be taken as provisional and may not be the privacy loss budget used in producing the actual DDHC-A File. Detailed DHC-B and Supplemental DHC proofs of concept have not yet been released and no provisional budgets have been announced by the USCB, as of April 2023.

[32] The exact privacy loss budget ϵ depends on the swapping rate p and the swap key $\boldsymbol{V}_{\text{Match}}$.

[33] Depending on the swap key $\boldsymbol{V}_{\text{Match}}$ and the swapping variables $\boldsymbol{V}_{\text{Swap}}$, the invariants are all (multivariate) household characteristics at either the state, county or block group and (optionally) the household size at a geography one level lower.

and $\boldsymbol{Q}_h(\boldsymbol{X}_h)$ (at the household level)[34] respectively. Here $\boldsymbol{X}_p$ and $\boldsymbol{X}_h$ are the Census Edited Files – the final version of the confidential Census data (after data cleaning and missing data imputation). $\boldsymbol{Q}_p(\boldsymbol{X}_p)$ and $\boldsymbol{Q}_h(\boldsymbol{X}_h)$ vary with the implementation of TDA, but (roughly) they are the statistics (without privacy noise) that the US Census Bureau would like to publish. For example, when releasing the redistricting data files, $\boldsymbol{Q}_p(\boldsymbol{X}_p)$ and $\boldsymbol{Q}_h(\boldsymbol{X}_h)$ are equal to these files, but aggregated directly from the Census microdata without any privacy protection. However, to improve accuracy, the USCB adds additional queries to $\boldsymbol{Q}_p, \boldsymbol{Q}_h$ beyond the counts in the redistricting data or DHC files.

Based on the theory developed in Canonne et al. [2022], Abowd et al. [2022] prove that the mechanism $\boldsymbol{T}_p$ satisfies $(\mathcal{X}, d^p_{r_b}, D_{\text{nor}})$ and $\boldsymbol{T}_h$ satisfies $(\mathcal{X}, d^h_{r_b}, D_{\text{nor}})$, where D_{nor} is the normalised Rényi metric (which was defined in Section 3.6 and corresponds to zero concentrated differential privacy) and $d^u_{r_b}$ is the bounded divergence (defined in Section 3.5) with persons – for $\boldsymbol{T}_p$ – and households – for $\boldsymbol{T}_h$ – as the privacy units u. Discrete Gaussian noise is added to $\boldsymbol{Q}_p(\boldsymbol{X}_p)$ and $\boldsymbol{Q}_h(\boldsymbol{X}_h)$ to produce the NMFs $\boldsymbol{T}_p(\boldsymbol{X}_p)$ and $\boldsymbol{T}_h(\boldsymbol{X}_h)$. The scale parameters $\boldsymbol{D}_p, \boldsymbol{D}_h$ (analogous to the covariance matrices of multivariate Gaussian distributions) of the privacy noise are diagonal matrices. The entries along the diagonal control the total privacy budget ρ_{TDA} of the mechanisms $\boldsymbol{T}_p$ and $\boldsymbol{T}_h$, as well as the allocation of that budget across the different cells in the tables $\boldsymbol{Q}_p(\boldsymbol{X}_p)$ and $\boldsymbol{Q}_h(\boldsymbol{X}_h)$. The privacy budgets ρ_{TDA} for both $\boldsymbol{T}_p$ and $\boldsymbol{T}_h$ as used in the production of the Census redistricting data (PL file) and DHC file are presented in Table 6.2.

		ρ[35]	ϵ (with $\delta = 10^{-10}$)[36]
PL	Household	0.07	2.70
	Person	2.56	17.90
DHC	Household	7.70	34.33
	Person	4.96	26.34
Total		15.29	52.83

Table 6.2: The privacy loss budgets of the mechanisms $\boldsymbol{T}_p$ (person) and $\boldsymbol{T}_h$ (household) used in the first step of the TDA to produce the 2020 Census Redistricting Data (P.L. 94-171) Summary and the Demographic and Housing Characteristics Files. Source: US Census Bureau [2023c].

In the second step (called the "estimation phase" in Abowd et al. [2022]), Privacy-Protected Microdata Files (PPMF) $\boldsymbol{Z}_p$ and $\boldsymbol{Z}_h$ are produced by solving a complex optimisation problem.[37] The PPMF $\boldsymbol{Z}_p$ and $\boldsymbol{Z}_h$ agree with the Census Edited File $\boldsymbol{X}_p, \boldsymbol{X}_h$ on the invariants $\boldsymbol{c}_{\text{TDA}}$. The invariants used in the production of the PL and DHC files are given in Table 6.1. In addition, the PPMF $\boldsymbol{Z}_p$ and $\boldsymbol{Z}_h$ for the DHC are consistent with related statistics in the PL file [US Census Bureau, 2023d]. When producing the DHC, the PL file $\boldsymbol{P}$ is passed as input into the TDA and a constraint

[34] In this Section, we will include group quarters as households for the purposes of conciseness.

[35] To avoid confusion, we use the parametrisation of ρ common in the literature, which is equal to ρ^2 under the parametrisation of zCDP using D_{nor} as given in Section 3.6.

[36] Using the conversion $\epsilon = \rho + 2\sqrt{-\rho \ln \delta}$ given in Bun and Steinke [2016a] and adopted by the USCB.

[37] The PPMF is also called the Microdata Detail File by Abowd et al. [2022].

$\boldsymbol{H}(\boldsymbol{Z}_p, \boldsymbol{Z}_h) = \boldsymbol{P}$ is added to the optimization problem to enforce consistency between the DHC and PL. (The input $\boldsymbol{P}$ is not used by the TDA in production of the PL file.)

The PL and DHC files are produced by summarising the PPMF datasets $\boldsymbol{Z}_p$ and $\boldsymbol{Z}_h$ into tables. In addition to the PL and DHC files, the USCB is releasing the NMF $\boldsymbol{T}_p(\boldsymbol{X}_p)$ and $\boldsymbol{T}_h(\boldsymbol{X}_h)$ produced for the PL and DHC files [US Census Bureau, 2023e], and the PPMF $\boldsymbol{Z}_p$ and $\boldsymbol{Z}_h$ for the DHC file [US Census Bureau, 2023b].

Algorithm 6.1: Overview of the TopDown Algorithm [Abowd et al., 2022], focusing on aspects salient to privacy analysis.

Input:

Census Edited Files $\boldsymbol{X}_p, \boldsymbol{X}_h$ at the person and household levels
Person queries $\boldsymbol{Q}_p$
Household queries $\boldsymbol{Q}_h$
Privacy noise scales $\boldsymbol{D}_p$ and $\boldsymbol{D}_h$
Constraints $\boldsymbol{c}_{\text{TDA}}$ (including invariants, edit constraints and structural zeroes)
(Optional) previously released statistics $\boldsymbol{P}$, as aggregated from a microdata file (where the aggregation was achieved using a function $\boldsymbol{H}$)

Step 1: Noise Infusion
  Sample discrete Gaussian noise [Canonne et al., 2022]:
    $\boldsymbol{W}_p \sim \mathcal{N}_{\mathbb{Z}}(\boldsymbol{0}, \boldsymbol{D}_p)$
    $\boldsymbol{W}_h \sim \mathcal{N}_{\mathbb{Z}}(\boldsymbol{0}, \boldsymbol{D}_h)$
  Compute Noisy Measurement Files:
    $\boldsymbol{T}_p(\boldsymbol{X}_p) \leftarrow \boldsymbol{Q}_p(\boldsymbol{X}_p) + \boldsymbol{W}_p$
    $\boldsymbol{T}_h(\boldsymbol{X}_h) \leftarrow \boldsymbol{Q}_h(\boldsymbol{X}_h) + \boldsymbol{W}_h$
Step 2: Post-Processing
  Compute Privacy-Protected Microdata Files $\boldsymbol{Z}_p, \boldsymbol{Z}_h$ as a solution to the optimisation problem:
    Minimize loss l between $[\boldsymbol{T}_p(\boldsymbol{X}_p), \boldsymbol{T}_h(\boldsymbol{X}_h)]$ and $[\boldsymbol{Q}_p(\boldsymbol{Z}_p), \boldsymbol{Q}_h(\boldsymbol{Z}_h)]$
      subject to constraints $\boldsymbol{c}_{\text{TDA}}(\boldsymbol{Z}_p, \boldsymbol{Z}_h) = \boldsymbol{c}_{\text{TDA}}(\boldsymbol{X}_p, \boldsymbol{X}_h)$ and $\boldsymbol{H}(\boldsymbol{Z}_p, \boldsymbol{Z}_h) = \boldsymbol{P}$.

Output:

Privacy-Protected Microdata Files $\boldsymbol{Z}_p, \boldsymbol{Z}_h$, and
Noisy Measurement Files $\boldsymbol{T}_p(\boldsymbol{X}_p), \boldsymbol{T}_h(\boldsymbol{X}_h)$ at the person and household levels.

Theorem 6.1. *Let $\boldsymbol{c}_{\text{TDA}}$ be the invariants of TDA (given in Table 6.1) and let $\mathscr{D}_{\boldsymbol{c}_{\text{TDA}}}$ be the induced data universe function (as defined in (3.1)). Then TDA satisfies the differential privacy definition $(\mathscr{D}_{\boldsymbol{c}_{\text{TDA}}}, d^p_{\text{HamS}}, D_{\text{nor}})$ with privacy budget $\rho_{\text{TDA}} = 2.63$ (for the Census Redistricting Summary File) and $\rho_{\text{TDA}} = 15.29$ (for the DHC), where d^p_{HamS} is the symmetric Hamming distance d_{HamS} with persons as privacy units.*

In the opposite direction, let $\boldsymbol{c}'$ be any proper subset of TDA's invariants (which varies on $\mathcal{X}$ – i.e. $\boldsymbol{c}'$ is not what the USCB call an edit constraint). Then TDA does not satisfy $(\mathscr{D}_{\boldsymbol{c}'}, d_{\mathcal{X}}, D_{\text{nor}})$ with any finite budget ρ.

The second half of this Theorem can be generalized from $(\mathscr{D}_{\boldsymbol{c}'}, d_{\mathcal{X}}, D_{\text{nor}})$ to any privacy definition

$(\mathscr{D}, d_{\mathcal{X}}, d_{\mathcal{T}})$ satisfying the assumptions of Proposition 4.3. In the first half of the Theorem, we cannot replace D_{nor} with the multiplicative distance MULT since the NMF $\boldsymbol{T}_p$ and $\boldsymbol{T}_h$ are protected with Gaussian noise. Gaussian noise infusion does not satisfy $(\mathcal{X}, d_{\mathcal{X}}, \text{MULT})$ for any finite ϵ (assuming that $\mathcal{X}$ and $d_{\mathcal{X}}$ are not trivial, so that $(\mathcal{X}, d_{\mathcal{X}}, \text{MULT})$ is not a vacuous definition).

Proof. Fix the privacy definition $\mathcal{DP} = (\mathscr{D}_{\boldsymbol{c}_{\text{TDA}}}, d^p_{\text{HamS}}, D_{\text{nor}})$. First, we analyze the TDA for producing the PL file. The household mechanism $\boldsymbol{T}_h$ satisfies $(\mathcal{X}, d^h_{r_b}, D_{\text{nor}})$ (see Abowd et al. [2022]), which is equivalent to $(\mathcal{X}, d^h_{\text{HamS}}, D_{\text{nor}})$ [Bailie et al., 2023+]. Hence $\boldsymbol{T}_h$ satisfies $\mathcal{DP}$ by Propositions 4.1 and 4.5 with $\rho = 0.07$. We can similarly conclude that $\boldsymbol{T}_p$ satisfies $\mathcal{DP}$ with $\rho = 2.56$. Then by composition, the mechanism $\boldsymbol{T}_{ph} = [\boldsymbol{T}_p, \boldsymbol{T}_h]$ has privacy budget $\rho = 0.07 + 2.56 = 2.63$. Proposition 4.3 implies the mechanism $\boldsymbol{T}_{\boldsymbol{c}_{\text{TDA}}}$ which releases the invariants $\boldsymbol{c}_{\text{TDA}}(\boldsymbol{X}_p, \boldsymbol{X}_h)$ has $\rho = 0$.

The composed mechanism $\boldsymbol{T} = [\boldsymbol{T}_{ph}, \boldsymbol{T}_{\boldsymbol{c}_{\text{TDA}}}]$ has budget $\rho = 2.63$. The second step of the TDA is post-processing on $\boldsymbol{T}$ and hence has the same budget.

The argument for producing the DHC file is almost analogous. The composed mechanism $\boldsymbol{T} = [\boldsymbol{T}_{ph}, \boldsymbol{T}_{\boldsymbol{c}_{\text{TDA}}}]$ has budget $\rho = 7.70 + 4.96 = 12.66$. Now the second step of the TDA also uses the PL file $\boldsymbol{P}$. Hence, this second step is post-processing on the composed mechanism $[\boldsymbol{T}, \boldsymbol{P}]$. This composed mechanism has budget $\rho = 12.66 + 2.63 = 15.29$.

The second half of the Theorem follows from Proposition 4.3. □

The second step of the TDA requires access to both the NMF $[\boldsymbol{T}_p(\boldsymbol{X}_p), \boldsymbol{T}_h(\boldsymbol{X}_h)]$ and the invariant statistics $\boldsymbol{c}_{\text{TDA}}(\boldsymbol{X}_p, \boldsymbol{X}_h)$ computed on the Census Edited File. Under the privacy definition $(\mathcal{X}, d_{\mathcal{X}}, d_{\mathcal{T}})$, the invariant statistics $\boldsymbol{c}_{\text{TDA}}(\boldsymbol{X}_p, \boldsymbol{X}_h)$ cannot be released with finite budget. So the second step of the TDA is not post-processing (in the sense given in Section 3.7) under $(\mathcal{X}, d_{\mathcal{X}}, d_{\mathcal{T}})$ – it is only post-processing when conditioning on the invariants. In the proof of Theorem 6.1, we must use the privacy definition $(\mathscr{D}_{\boldsymbol{c}_{\text{TDA}}}, d_{\mathcal{X}}, D_{\text{nor}})$ in order to use the post-processing property of DP. The second half of Theorem 6.1 shows that any argument which relies on TDA's second step being post-processing must necessarily use a privacy definition which conditions on the invariants $\boldsymbol{c}_{\text{TDA}}$.

It is also necessary to use persons as privacy units in TDA's privacy definition. While the household mechanism $\boldsymbol{T}_h$ satisfies $(\mathcal{X}, d^h_{\text{HamS}}, D_{\text{nor}})$ (where d^h_{HamS} is the symmetric Hamming distance with households as privacy units), the sensitivity of the person-level query $\boldsymbol{Q}_p$ due to a single change in a household record is very large. (In the Census Edited File, the maximum possible household size is 99,999 [Population Reference Bureau and U.S. Census Bureau's 2020 Census Data Products and Dissemination Team, 2023].) This means $\boldsymbol{T}_p$ satisfies $(\mathcal{X}, d^h_{\text{HamS}}, D_{\text{nor}})$ only with a very large amplification in privacy budget. Hence, the budget of the TDA would also have to be substantially increased in order to satisfy $(\mathscr{D}_{\boldsymbol{c}_{\text{TDA}}}, d^h_{\text{HamS}}, D_{\text{nor}})$.

6.2 What if the 2020 Census used swapping?

In this Section we ask the counterfactual question: what if swapping, as we formulate in Algorithm 5.1, is applied to the 2020 Decennial Census? In particular, what would the privacy guarantee look like under different choices of swapping schemes and swap rates?

Table 6.3 shows the total nominal ϵ that would be achieved by applying swapping to the 2020 Decennial Census for a variety of possible parameter choices. For the purpose of illustration, we stipulate the swapping variable $\boldsymbol{V}_{\text{Swap}}$ to be the block, tract, or county membership of each household, and the matching variable $\boldsymbol{V}_{\text{Match}}$ to be the geography one level higher than $\boldsymbol{V}_{\text{Swap}}$, either alone or crossed with the household size variable. From the top to bottom rows of Table 6.3, the $\boldsymbol{V}_{\text{Swap}}$ levels are ordered according to increasing granularity of geography. Within each level of $\boldsymbol{V}_{\text{Swap}}$, the two $\boldsymbol{V}_{\text{Match}}$ levels are nested, in the sense that the swapping scheme represented in the latter row (i.e. crossed with household size) induces a logically stronger and more constrained set of invariants than the former one. These $\boldsymbol{V}_{\text{Match}} \times \boldsymbol{V}_{\text{Swap}}$ level combinations result in largest strata of varying sizes, as can be seen from b ranging from as large as 13.68 million (the total number of households in California) to as small as $11,691$ (the total number of 3-person households in a Florida block group).[38] Varying the swap rate between a low level (5%) and a high level (50%), the nominal ϵ achieved ranges between 9.37 to 19.38.

If we entertain the assumption that the 2010 Census DAS implemented swapping in the same way as framed in Algorithm 5.1, we could also obtain a crude sketch of the privacy guarantee that it would afford. It has been suggested that the 2010 DAS utilized swap keys which include household size as well as household voting age population and some geography (either tract, county or state) [U.S. Census Bureau, 2021b]. As we are unable to locate 2010 Census data products that allows for the precise calculation of b pertaining to this particular swapping scheme, the swap key we consider here is coarser as it does not accounting for the household count of voting age persons. However, setting $\boldsymbol{V}_{\text{Match}}$ to be "state $\times$ household size" would imply $b = 3.65$ million (Table 6.3), which serves as an upper bound for the actual b for the 2010 Census. Combined with a purported swap rate p between $2\% - 4\%$ [boyd and Sarathy, 2022] we arrive at (an overestimate of) the nominal ϵ to be between 18.29 and 19. We emphasize that this ϵ does not necessarily reflect the privacy budget of the 2010 DAS, but rather the budget of Algorithm 5.1 when we choose its parameters to reflect what we know about the 2010 DAS.

This analysis reaffirms a counterintuitive observation we previously made with Figure 2.1 as we state the main results in Section 2. When the swap rate p is fixed, the more invariants induced by the swapping mechanism, the smaller the nominal ϵ it achieves. As Table 6.3 shows, when swaps are performed freely across counties in a state, even a high swap rate of 50% renders a nominal ϵ that is much larger than that pertaining to swaps among households of the same size within a

[38] At the time of writing, the 2020 Census DHC Files have not been released, hence we calculate the largest strata sizes (b) based on the Census Bureau's 2010 demonstration Privacy-Protected Microdata File (PPMF) for DHC via IPUMS [Van Riper et al., 2020].

[39] The Florida block group identified in lines 5 and 6 of Table 6.3 has GIS join match code G12011909112001.

Table 6.3: The total nominal ϵ achievable by applying swapping to the 2020 Decennial Census for a variety of $\boldsymbol{V}_{\text{Match}}$, $\boldsymbol{V}_{\text{Swap}}$, and swap rate choices. The largest stratum under each setting are obtained from the 2010 Census data.

$\boldsymbol{V}_{\text{Match}}$	$\boldsymbol{V}_{\text{Swap}}$	b	total ϵ $p = 5\%$	total ϵ $p = 50\%$	Largest stratum
state	county	13680081	19.38	16.43	California
state × household size	county	3653802	18.06	15.11	California, 3-household
county	tract	3445076	18.00	15.05	LA County
county × household size	tract	853003	16.60	13.66	LA County, 3-household
block group	block	21535	12.92	9.98	a FL block group[39]
block group × household size	block	11691	12.31	9.37	a FL block group, 3-household

block group at a low swap rate of 5% ($\epsilon = 16.43$ and 12.31 respectively). If these nominal ϵ's are taken at face value, one may be tempted to conclude that swapping schemes with finer invariants should be preferred from a privacy standpoint. Furthermore, one may find it convenient to also recognize that finer invariants are desirable from a data utility standpoint, for the obvious reason that more exact statistics about the confidential are made known. However, such a conclusion – that finer invariants should benefit both utility *and* privacy – would be dangerously mistaken, for it overlooks the privacy leakage, in an ordinary sense of the phrase, due to the invariants alone. This again highlights the importance of interpreting ϵ within its context, and the necessity of treating the invariants as an integral part of the privacy guarantee.

It remains difficult to conduct a direct comparison between the privacy guarantees of the above thought experiment and the actual construction of the suite of privacy-protected 2020 Decennial Census data products. There are several reasons for this. First, the 2020 Census data products, such as the redistricting data files generated by the TopDown Algorithm, carry a set of invariants that cannot be induced by the swapping algorithm under examination. That is, the invariants induced by TopDown does not accord to any choice of $\boldsymbol{V}_{\text{Swap}}$, $\boldsymbol{V}_{\text{Match}}$, and $\boldsymbol{V}_{\text{Hold}}$ (as shown in Example 5.5). Therefore, the two methods' classes of differential privacy guarantees are not nested, rendering their comparison inconvenient. (Although swapping almost always has stricter invariants for most variables, it does not necessarily have TDA's group quarter invariants.) In Section 7.2, we discuss possible modifications to result in a more flexible range of invariants for swapping, in order to bring the two schemes closer.

Second, swapping and the TopDown algorithm differ in their choices of both data divergence $d_{\mathcal{X}}$ and output divergence $d_{\mathcal{T}}$. Notably, they have different privacy units. Swapping as implemented in this Section defines households as its privacy unit, whereas the TDA uses persons. As Section 4.1 explains, household distance is a stronger notion than individual distance, since if the record of a single household changes part of its value, the multiple persons residing in a same household may all change their values. In addition, swapping utilizes the multiplicative distance Mult as its output divergence, which is stronger than D_{nor} as employed by the TopDown algorithm (see Proposition 4.2). These differences further complicate a comparison between swapping and the

suite of 2020 Census privacy mechanisms.

Note that if swapping were to be applied to the 2020 Decennial Census with a fixed parameter setting ($\boldsymbol{V}_{\text{Match}}$, $\boldsymbol{V}_{\text{Swap}}$, and p), the nominal ϵ reported in Table 6.3 would be the *total* privacy loss budget that pertain to all data products derived from the the swapped dataset $\boldsymbol{Z}$, including the P.L. 94-171 summary file, the DHC, and the Detailed DHC files, for both persons and household product types. This is because swapping is performed on the full microdata file, and hence produces a synthetic version of it from which all data products are produced. Therefore, when comparing the ϵ values in Table 6.3 with those reported for TopDown and SafeTab in Table 6.1, it should be understood that the PLB of swapping will not increase with the release of additional data products; yet (at the time of writing) the privacy loss budget for the 2020 DAS must necessarily continue to grow with additional data releases. This characteristic of swapping leads to an additional desirable property that is not enjoyed by mechanisms based on output noise infusion: the *logical consistency* of multiple data products resulting from swapping is automatically preserved without the need for post-processing. We return to this matter in Section 7.2.

7 Discussion

As the title of this work suggests, we have taken a "stirred, not shaken" approach to the two central subjects under study: data swapping and differential privacy. The goal is neither to revamp these notions nor to rob them of their essence. Quite the contrary, by examining swapping through the lens of formal DP and providing theoretical characterization of its privacy guarantees, we seek to unite and reap the benefits of both worlds [Slavković and Seeman, 2023], including the facial validity and backward compatibility of the data products produced by swapping, as well as algorithmic transparency that pertain to formal DP methods [Gong, 2022].

Nevertheless, to compare two SDC approaches that were previously thought to be distinct requires a supporting common ground, which this work establishes by spelling out the details of one in full, and piecing together the known facts about the other. The result is a rich lesson on both approaches. In what follows, we first reiterate the take-home messages from our investigation in a pragmatic (and invariably tongue-in-cheek) manner, then discuss implications on existing debates concerning swapping and traditional SDC.

7.1 How to reduce privacy loss without adding more noise: a perverse guide

As Section 4 explains in detail, ϵ as the privacy loss budget is contextual in nature. While that is true for every differentially private mechanism, our analysis of swapping makes a particularly vivid case study: If the value of a privacy loss budget is taken nominally and out of context, we risk running down a slippery slope. As we have shown, there are ways to spend apparently less privacy loss budget, all the while without adding more noise to the privacy-protected data product. In this

Section we review some of these ways to "cheat". Needless to say, our intention is not to encourage such behavior, but rather on the contrary to expose the inherent weaknesses that are open for exploitation in what seems to be an objective and mathematically absolute framework of privacy protection. Our warnings may be particularly applicable to commercial implementations of privacy protection, where conflicts of interest are commonplace. For example, when the data custodian and data user are the same entity (such as an internet platform company), it can be easily tempted to cut some differentially-private corners due to a desire to improve data utility while maintaining prima-facie privacy protection to assauge its data contributors.

The first way to spend less privacy loss budget without adding more noise is to add more invariants. Theorem 5.6 reveals that the PLB ϵ of the Permutation Algorithm is determined by two things: the swap rate p and the largest stratum size b. To decrease the nominal value of ϵ, one can either increase p (up to a point; see Figure 2.1) or decrease b. When the dataset has a fixed size, the simplest way to decrease b is to define the matching variables $\boldsymbol{V}_{\text{Match}}$ at a finer resolution, resulting in smaller sizes of strata within which swapping is confined. As Section 5.3 shows, the various $\boldsymbol{V}_{\text{Match}}$ choices at different levels of geography, with or without crossing with the household size variable, result in b ranging from as small as 11.7 thousand to as large as 13.7 million, and a nominal ϵ from 12.31 to 19.38 (respectively) at $p = 5\%$.

Decreasing invariants and increasing b should increase the number of candidates that a unit may swap with, thereby intuitively increasing privacy. But increasing b actually increases the nominal privacy budget ϵ, even though high resolution invariants can be revealing of the unit's information in and of themselves. For an extreme example, suppose we define the swap key $\boldsymbol{V}_{\text{Match}}$ in such a way that all records with the same key are duplicates of each other. Then, applying swapping accomplishes nothing regardless of p! Indeed, under this set-up, $d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}') = 0$ for all datasets $\boldsymbol{X}, \boldsymbol{X}'$ in the observed universe $\mathcal{D} = \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X}^*)$ and hence the privacy definition $(\mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}, d^u_{\text{HamS}}, d_{\mathcal{T}})$ is vacuous at $\mathcal{D}$. This is reflected in Theorem 5.6 which shows that the budget $\epsilon_{\mathcal{D}} = 0$ when there are no non-duplicates with the same swap key (i.e. when $b = 0$). The sheer amount of invariants is sufficient for the reconstruction of the entire micro dataset, whereas we can still obtain an apparent privacy guarantee "for free". This apparent conflict between the mathematics and our intuition is evidence that privacy accounting by the privacy loss budget alone is inadequate.

A second way to achieve a reduction of nominal PLB for free is to redefine privacy units at a finer granularity. The intuition behind this maneuver has been recognized, and to some extent utilized, in the literature of differential privacy mechanism design for complex data structures. For example for network data, the choice of neighbours is particularly important – are neighbours defined by removing a node or an edge from the network (i.e. are privacy units edges or nodes?) [Raskhodnikova and Smith, 2016]? For business databases, does a company constitute a unit, or should units be employees, or both [Haney et al., 2017, Schmutte, 2016, He et al., 2014]? Or should they be the company's transactions? Similarly for large personal databases in commercial settings, should an individual constitute a unit, or should each of their interaction with the platform (such a post or a "like") be privacy units, or should units be the set of a user's interactions within a given time period

(e.g. a single day) [Kenthapadi and Tran, 2018, Messing et al., 2020, Desfontaines, 2023]? Finally, when publishing social statistics, do households deserve privacy protection above and beyond the protection afforded to their individual members [Machanavajjhala, 2022]?

As Sections 4.1 and 6.2 explain, with all else being equal, a differentially private mechanism with a larger notion of privacy unit packs more weight in its privacy loss budget, and offers a stronger privacy guarantee at the same nominal budget than a mechanism with a smaller notion of privacy unit. Substituting a coarser privacy unit with a finer one shrinks the privacy loss budget.

The differential privacy definition as we spell out in full in Definition 3.3 points to additional ways to "cheat" beyond what's discussed above. Put simply, every component of a differential privacy definition – $d_{\mathcal{X}}$, $d_{\mathcal{T}}$ and $\mathscr{D}$ – can be gamed. For example, another way to gain PLB out of thin air is to artificially introduce an output divergence $d_{\mathcal{T}}$ that systematically assesses two distributions to be closer. Technically speaking, the relaxation from ϵ-DP to (ϵ, δ)-approximate DP can be understood as a maneuver of this type, due to the fact that $\textsc{Mult}^{\delta}$ is strictly smaller than $\textsc{Mult}$ provided that $\delta > 0$.

We finish this Section by briefly remarking on a component which has received little attention in this paper: the data space $\mathcal{X}$. We have assumed throughout that $\mathcal{X}$ has remained fixed and have attempted to compare privacy definitions defined on the same space $\mathcal{X}$. In reality, the data custodian is free to choose $\mathcal{X}$ along with their choice of privacy definition. Given the rest of this paper, it should come as no surprise that we view the choice of $\mathcal{X}$ as a critical, yet critically overlooked, nuance in DP and SDC more generally. However, we withhold this matter for a future discussion; and leave as an exercise for the reader the question of how one might use $\mathcal{X}$ to perversely lower the privacy loss budget.

7.2 Implications on current debates concerning swapping

The aim of the current paper is to present an objective analysis of a class of swapping mechanisms through the lens of formal privacy. It is not an endorsement of existing implementations of swapping in the 2010 Census data products or other previously published official data products, not the least because we cannot possibly ascertain the extent to which the existing implementations conforms to our formulation.

We are aware that from some perspectives, what this paper accomplishes may be an inconvenient truth. We want to be clear that we are not advocating to reverse the progress that the Census Bureau and the formal privacy research community have made to advance statistical disclosure control. On the contrary, our intention is to lay down a rigorous explication of differential privacy and to examine existing SDC methods through this formal lens. We believe that this approach ultimately facilitates the modernization of SDC in a way that is helpful to data custodians, responsible to data contributors, and respectful to data users.

A concrete benefit of the new perspective we provide is that it sheds light on debates concerning swapping. In what follows, we review and provide our comments on three current discussions. We will argue that most of these contentious issues are tangential to the fundamental nature of swapping and DP noise infusion as mechanisms for data privacy protection. Our work provides a level playing field that allows for a much needed, informed, and fair comparison.

The use of reconstruction attacks to assist privacy mechanism design One of the Census Bureau's main arguments for revamping its swapping-based SDC for Decennial Census data products is that swapping does not provide adequate protection against reconstruction attacks [Abowd and Hawes, 2023]. Generally speaking, reconstruction attacks work by pulling together many aggregate statistics about a confidential database and recreating the possible individual records under their guidance. The larger the number of aggregate statistics and the more accurate they are, the more heavily they constrain the possible configurations of the underlying microdata. As a result, it is easier to create reconstructed databases with a high chance to lead to the reidentification of units via linkage to external data sources. This is the idea behind the Bureau's own suite of simulated reconstruction attacks against the 2010 Census which resulted in high rates of reidentification [Abowd, 2018, Hawes, 2022].

The weakness of 2010 Census swapping mechanism as exposed by the reconstruction attacks is without a doubt alarming and calls for action. Equipped with a formal framework to understand and explicate the differential privacy guarantees associated with swapping, we can delineate the causes of its susceptibility to reconstruction. We now know that the reason swapping does not stand up against this particular style of reconstruction attacks is not because it is not differentially private. Because the swapping algorithm that we examine closely mimics the 2010 DAS, it will likely suffer from similar susceptibility. A fix for this problem would thus need to result from an examination of the elements of the algorithmic specification of swapping.

A commonly held belief why the Bureau's simulated reconstruction attack on the 2010 Census swapped data leads to high reidentification is that the 2010 swapping scheme induced too many invariants. As Abowd and Hawes [2023] discuss, these invariants are particularly harmful because of two facts: 1) total and voting age populations at the block level constitute information at very fine granularity, and 2) the existence of a high fraction of unique persons within blocks (57%) further facilitates reidentification via record linkage. As our analysis of Section 6.2 shows, the granularity of invariants has a larger numerical impact on the privacy loss budget ϵ, more so than the swap rate for a given set of invariants. Therefore, what may need the most urgent revision are the invariants. We expand on this in greater detail in our next discussion point.

The important, despite obvious, message here is that a reconstruction attack on swapping is a reconstruction attacks on a differentially private mechanism, and all consequences of it should be understood as such. The vulnerability of a privacy-protected data product against reconstruction attacks is the combined result of all aspects of its privacy mechanism's design, encompassing pa-

rameter choices that go far beyond the privacy loss budget.

On the plurality (and the limited choice) of invariants An important criticism of swapping as implemented in the 2010 and prior Decennial Censuses is that it induces too many invariants. The invariants severely constrain the permissible data universes, impacting the disclosure risk of the resulting data product in unpredictable ways. A key question is thus to understand the actual privacy guarantees in the presence of a plurality of invariants.

One salient consequence of the plurality of invariants is that it can be impossible to, via swapping, simultaneously maintain a low degree of data disruption and control the risk of identification for population uniques. The Census Bureau reconstruction attacks experiments show that they had to significantly increase the rate of swapping to arrive at what is deemed as an acceptable level of protection for the population uniques [Abowd and Hawes, 2023]. Specifically, the population uniques can be revealed as part of the invariants. As an extreme example, any swap-key stratum with only duplicate records would result in a exact reconstruction of that stratum.

In order to design a tailored solution that balances privacy and accuracy targets, we believe the data custodian should be equipped with the necessary methodology to control invariants in a flexible and precise manner. To this end, swapping – as implemented in the previous Decennial Censuses as well as in this work – is insufficient. Nevertheless, several tangible solutions present themselves as immediately open for exploration, which we discuss now.

The Census Bureau's comparative analysis between swapping and TopDown considered methods to override the hard invariants due to swapping. One such method is pre-swap perturbation [Hawes and Rodriguez, 2021, p. 23], which infuses noise into the confidential record prior to applying swapping. Another possible method is the probabilistic matching of units. That is, instead of using $\boldsymbol{V}_{\text{Match}}$ to form hard strata that confine swapping, allow, with a small probability, for units across different strata to be swapped. The probability can made inversely proportional to some distance metric on the strata. As a demonstration, take the 1940 Census full count example from Section 5.3, where $\boldsymbol{V}_{\text{Match}}$ is the state indicator and size of the household and $\boldsymbol{V}_{\text{Swap}}$ is the county indicator. Suppose for some $\alpha > 0$, a household chosen for swapping would have a $(1-\alpha)\%$ chance of being swapped with another household of the same size, but an $\alpha\%$ chance of being swapped with a differently-sized household. Doing so retains the county-wide household counts as invariant, but the county-wide total populations are no longer invariant. We leave it to future work to investigate the theoretical guarantees of pre-swap perturbation and probabilistic matching, only noting here that compared to classic swapping, both induce strictly more auxiliary randomness into the data product. Therefore, it would be reasonable to expect the resulting algorithms to enjoy formal privacy guarantees while supplying more flexible choices of invariants.

Finally, we note that the problem of invariant choice does not pertain to swapping alone. Whether the data custodian implements swapping or another privacy protection mechanism, the choice of invariants is unavoidable. The production settings of the TopDown algorithm employed invariants

as specified by Table 6.1, but the list of invariants was arrived at through an iterative process. USCB were previously considering block-level population invariants; see e.g. Ashmead et al. [2019], Kifer [2019]. This illustrates that invariant choice is often a part of many privacy mechanism designs and parameter choices.

Data utility under transparent privacy Our last point of discussion re-emphasizes another important motivation for this work, which was only mentioned briefly in the Introduction. By casting swapping as formal DP, we can theoretically allow its algorithmic specification to be made public. As the main SDC method for the Decennial Census of the previous three decades, a peek into the technical specification of swapping can bring tremendous utility to data users and privacy researchers alike.

Data users who conduct statistical modeling with official data products criticize swapping because it negatively affects the quality of downstream data analyses. It has been well understood in the literature that swapping inflicts the most utility damage to the relationships between swapping and holding variables. Mitra and Reiter [2006] and Drechsler and Reiter [2010] demonstrate that even low swap rates (e.g. 5%) can substantially reduce the effective coverage of confidence intervals for the regression coefficient between such variables.

We surmise that the deterioration in coverage is in part due to performing a naïve regression analysis on processed data, without accounting for the privacy mechanism itself. As Gong [2022] demonstrates, performing naïve regression analyses on data protected via DP noise-infusion results in similar types of coverage deterioration, and further that this deterioration can be restored once the privatization process is statistically modeled (at the expense of wider, though valid, intervals). However, the analyst cannot possibly be blamed for performing the naïve analyses on swapped data when the swapping procedure is not public. Unfortunately, swapping by tradition has not been a transparent SDC technique. The explicit statement of swapping's privacy guarantees provides theoretical justification to publish the implementation details of the swapping procedure. This would allow the swapping mechanism to be appropriately accounted for via statistical modeling.

Note that the justification for transparency relies on the privacy guarantee being public. In the case of swapping, publishing the privacy guarantee necessitates the release of its invariants. However, as we repeatedly emphasize throughout this work, there is danger of privacy leakage associated with the knowledge of invariants in and of themselves. For example, a plurality of invariants supply the adversary with confidence in their efforts to reconstruct the microdata and reidentify individual records. Therefore, careful deliberation has to be practised in weighing the cost of making public the invariants against the benefits of algorithmic transparency this allows.

Swapping also carries unique utility advantages compared to noise infusion, including maintaining the *facial validity* of the privacy-protected data product and the *logical consistency* between multiple data products derived from the same swapped database – all without having to resort to complex post-processing. The vast majority of formal DP methods based on noise injection rely

on optimization-based post-processing to restore facial validity and logical consistency [e.g. Barak et al., 2007, Hay et al., 2010]. Optimization-based post-processing can be procedurally transparent but in most cases it destroys the probabilistic transparency of the resulting two-step privacy mechanism. Probabilistic transparency is a stronger property than procedural transparency, but it is crucial – indeed necessary – to enable principled statistical analysis on the privacy-protected data product [Gong, 2022]. The recent proposal by Dharangutte et al. [2023] does away the need for post-processing in additive noise infusion to maintain both facial validity and logical consistency. However, it relies on MCMC sampling and hence is non-trivial to implement for large-scale data products. In contrast, swapping achieves both with ease, as it generates a synthetic version of the microdata through a recombination of empirically observed values, which in turn serves as the basis to derive all data products.

Acknowledgements

We thank Cory McCartan for his assistance with the 2010 US Census data, as well as for many stimulating discussions; and Xiaodong Yang, Nathan Cheng and Souhardya Sengupta for their help in proving Lemma A.4. All errors in the paper are purely our own.

References

J. Abowd, R. Ashmead, R. Cumings-Menon, S. Garfinkel, M. Heineck, C. Heiss, R. Johns, D. Kifer, P. Leclerc, A. Machanavajjhala, B. Moran, W. Sexton, M. Spence, and P. Zhuravlev. The 2020 Census disclosure avoidance system TopDown algorithm. *Harvard Data Science Review*, (Special Issue 2), June 2022. doi: 10.1162/99608f92.529e3cb9.

J. M. Abowd. The U.S. Census Bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '18*, pages 2867–2867, London, United Kingdom, 2018. ACM Press. ISBN 978-1-4503-5552-0. doi: 10.1145/3219819.3226070.

J. M. Abowd and M. B. Hawes. Confidentiality protection in the 2020 US Census of Population and Housing. *Annual Review of Statistics and Its Application*, 10:119–144, 2023.

J. M. Abowd and I. M. Schmutte. An economic analysis of privacy protection and statistical accuracy as social choices. *American Economic Review*, 109(1):171–202, Jan. 2019. ISSN 0002-8282. doi: 10.1257/aer.20170627.

R. Ashmead, D. Kifer, P. Leclerc, A. Machanavajjhala, and W. Sexton. Effective privacy after adjusting for invariants with applications to the 2020 Census. Technical report, 2019.

J. Bailie and C.-H. Chien. ABS perturbation methodology through the lens of differential privacy. In *Work Session on Statistical Data Confidentiality, UN Economic Commission for Europe*, page 13, Oct. 2019.

J. Bailie and R. Gong. Differential privacy: General inferential limits via intervals of measures. *Under submission*, 2023+.

J. Bailie, R. Gong, and X.-L. Meng. On the uniqueness of differential privacy. *in preparation*, 2023+.

B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 273–282, 2007.

A. Blanco-Justicia, D. Sánchez, J. Domingo-Ferrer, and K. Muralidhar. A critical review on the use (and misuse) of differential privacy in machine learning. *ACM Computing Surveys*, 55(8): 160:1–160:16, Dec. 2022. ISSN 0360-0300. doi: 10.1145/3547139.

d. boyd and J. Sarathy. Differential perspectives: Epistemic disconnects surrounding the U.S. Census Bureau's use of differential privacy. *Harvard Data Science Review*, (Special Issue 2), June 2022. doi: 10.1162/99608f92.66882f0e.

M. Bun and T. Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In M. Hirt and A. Smith, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 635–658, Berlin, Heidelberg, 2016a. Springer. ISBN 978-3-662-53641-4. doi: 10.1007/978-3-662-53641-4_24.

M. Bun and T. Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds, May 2016b.

C. Canonne, G. Kamath, and T. Steinke. The discrete Gaussian for differential privacy. *Journal of Privacy and Confidentiality*, 12(1), July 2022. ISSN 2575-8527. doi: 10.29012/jpc.784.

T. Dalenius and S. P. Reiss. Data-swapping: A technique for disclosure control. *Journal of Statistical Planning and Inference*, 6(1):73–85, Jan. 1982. ISSN 0378-3758. doi: 10.1016/0378-3758(82)90058-1.

D. Desfontaines. A list of real-world uses of differential privacy. https://desfontain.es/privacy/real-world-differential-privacy.html, Mar. 2023.

D. Desfontaines and B. Pejó. SoK: Differential privacies, June 2022.

P. Dharangutte, J. Gao, R. Gong, and F.-Y. Yu. Integer subspace differential privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-23)*, 2023.

J. Drechsler and J. P. Reiter. Sampling with synthesis: A new approach for releasing public use census microdata. *Journal of the American Statistical Association*, 105(492):1347–1357, Dec. 2010. ISSN 0162-1459. doi: 10.1198/jasa.2010.ap09480.

C. Dwork and G. N. Rothblum. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887*, 2016.

C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In S. Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, Lecture Notes in Computer Science, pages 486–503, Berlin, Heidelberg, 2006a. Springer. ISBN 978-3-540-34547-3. doi: 10.1007/11761679_29.

C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006b.

C. Dwork, N. Kohli, and D. Mulligan. Differential privacy in practice: Expose your epsilons! *Journal of Privacy and Confidentiality*, 9(2), Oct. 2019. ISSN 2575-8527. doi: 10.29012/jpc.689.

Ú. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, K. Talwar, and A. Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '19, pages 2468–2479. Society for Industrial and Applied Mathematics, Jan. 2019.

S. Fienberg and J. McIntyre. Data swapping: Variations on a theme by Dalenius and Reiss. In *Privacy in Statistical Databases*, 2004. doi: 10.1007/978-3-540-25955-8_2.

J. Gao, R. Gong, and F.-Y. Yu. Subspace differential privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3986–3995, June 2022. doi: 10.1609/aaai.v36i4.20315.

R. Gong. Transparent privacy is principled privacy. *Harvard Data Science Review*, (Special Issue 2), June 2022. doi: 10.1162/99608f92.b5d3faaa.

R. Gong and X.-L. Meng. Congenial differential privacy under mandated disclosure. In *Proceedings of the 2020 ACM-IMS on Foundations of Data Science Conference*, FODS '20, pages 59–70, 2020.

R. Gong, X.-L. Meng, and J. Bailie. Congenial differential privacy under mandated disclosure: Painting a deeper picture. *In preparation*, 2023+.

S. Haney, A. Machanavajjhala, J. M. Abowd, M. Graham, M. Kutzbach, and L. Vilhuber. Utility cost of formal privacy for releasing national employer-employee statistics. In *Proceedings of the 2017 ACM International Conference on Management of Data - SIGMOD '17*, pages 1339–1354, Chicago, Illinois, USA, 2017. ACM Press. doi: 10.1145/3035918.3035940.

M. Hawes. Reconstruction and re-identification of the demographic and housing characteristics file (DHC), 2022. https://www2.census.gov/about/partners/cac/sac/meetings/2022-09/presentation-reconstruction-and-re-dentification-of-dhc-file.pdf.

M. Hawes and R. Rodriguez. Determining the privacy-loss budget research into alternatives to differential privacy, 2021.

M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially private histograms through consistency. *Proceedings of the VLDB Endowment*, 3(1), 2010.

X. He, A. Machanavajjhala, and B. Ding. Blowfish privacy: Tuning privacy-utility trade-offs using policies. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1447–1458, 2014.

V. J. Hotz and J. Salvo. A chronicle of the application of differential privacy to the 2020 Census. *Harvard Data Science Review*, (Special Issue 2), June 2022. ISSN 2644-2353, 2688-8513. doi: 10.1162/99608f92.ff891fe5.

K. Kenthapadi and T. T. L. Tran. PriPeARL: A framework for privacy-preserving analytics and reporting at LinkedIn. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, pages 2183–2191, New York, NY, USA, Oct. 2018. Association for Computing Machinery. ISBN 978-1-4503-6014-2. doi: 10.1145/3269206.3272031.

D. Kifer. Design principles of the TopDown algorithm. Presentation, June 2019.

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. Technical Report arXiv:2209.03310, Sept. 2022.

A. Machanavajjhala. Candidate differential privacy algorithms for 2020 Decennial Census group II products. Workshop on the Analysis of Census Noisy Measurement Files and Differential Privacy, Apr. 2022. http://dimacs.rutgers.edu/events/details?eID=2038 [Accessed May 2023].

L. McKenna. Disclosure avoidance techniques used for the 1970 through 2010 Decennial Censuses of Population and Housing. Working paper, The Research and Methodology Directorate - US Census Bureau, Nov. 2018. URL https://www.census.gov/content/dam/Census/library/working-papers/2018/adrm/Disclosure%20Avoidance%20for%20the%201970-2010%20Censuses.pdf.

S. Meiser. Approximate and probabilistic differential privacy definitions. Technical Report 2018/277, 2018.

S. Messing, C. DeGregorio, B. Hillenbrand, G. King, S. Mahanti, Z. Mukerjee, C. Nayak, N. Persily, B. State, and A. Wilkins. Urls-v3.pdf. In *Facebook Privacy-Protected Full URLs Data Set*. Harvard Dataverse, 2020. doi: 10.7910/DVN/TDOAPG/DGSAMS. URL https://doi.org/10.7910/DVN/TDOAPG/DGSAMS.

I. Mironov. Rényi differential privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, Aug. 2017. doi: 10.1109/CSF.2017.11.

R. Mitra and J. P. Reiter. Adjusting survey weights when altering identifying design variables via synthetic data. In *Privacy in Statistical Databases: CENEX-SDC Project International Conference, PSD 2006, Rome, Italy, December 13-15, 2006. Proceedings*, pages 177–188. Springer, 2006.

H. F. Nissenbaum. *Privacy in Context: Technology, Policy, and the Integrity of Social Life*. Stanford Law Books, Stanford, Calif, 2010. ISBN 978-0-8047-5236-7 978-0-8047-5237-4.

Population Reference Bureau and U.S. Census Bureau's 2020 Census Data Products and Dissemination Team. Disclosure avoidance and the 2020 Census: How the TopDown algorithm works. 2020 Census Briefs C2020BR-04, Mar. 2023. https://www2.census.gov/library/publications/decennial/2020/census-briefs/c2020br-04.pdf [Accessed: 04-25-2023].

S. Raskhodnikova and A. Smith. Differentially private analysis of graphs. In M.-Y. Kao, editor, *Encyclopedia of Algorithms*, pages 543–547. Springer, New York, NY, 2016. ISBN 978-1-4939-2864-4. doi: 10.1007/978-1-4939-2864-4_549.

Y. Rinott, C. M. O'Keefe, N. Shlomo, and C. Skinner. Confidentiality and differential privacy in the dissemination of frequency tables. *Statistical Science*, 33(3):358–385, Aug. 2018. ISSN 0883-4237. doi: 10.1214/17-STS641.

S. Ruggles, C. Fitch, D. Magnuson, and J. Schroeder. Differential privacy and Census data: Implications for social and economic research. *AEA Papers and Proceedings*, 109:403–408, May 2019. ISSN 2574-0768. doi: 10.1257/pandp.20191107.

S. Ruggles, C. A. Fitch, R. Goeken, J. D. Hacker, M. A. Nelson, E. Roberts, M. Schouweiler, and M. Sobek. IPUMS ancestry full count data: Version 3.0 [dataset]. Minneapolis, MN: IPUMS, 2021. https://doi.org/10.18128/D014.V3.0.

P. Sadeghi and C.-H. Chien. On the connection between the ABS perturbation methodology and differential privacy, Mar. 2023.

I. M. Schmutte. Differentially private publication of data on wages and job mobility. *Statistical Journal of the IAOS*, 32(1):81–92, Jan. 2016. ISSN 1874-7655. doi: 10.3233/SJI-160962.

A. Slavković and J. Seeman. Statistical data privacy: A song of privacy and utility. *Annual Review of Statistics and Its Application*, 10(1):189–218, 2023. doi: 10.1146/annurev-statistics-033121-112921.

A. B. Slavković and J. Lee. Synthetic two-way contingency tables that preserve conditional frequencies. *Statistical Methodology*, 7(3):225–239, 2010.

J. Tang, A. Korolova, X. Bai, X. Wang, and X. Wang. Privacy loss in Apple's implementation of differential privacy on MacOS 10.12, Sept. 2017.

Tumult Labs. SafeTab: Dp algorithms for 2020 Census Detailed DHC Race & Ethnicity. Technical report, Mar. 2022.

U.S. Census Bureau. 2010 and 2020 Census data product release dates. https://www2.census.gov/programs-surveys/decennial/2020/program-management/2010_20_data_product_release_dates.pdf.

U.S. Census Bureau. 2020 Census: Redistricting File (Public Law 94-171) dataset (aug 12, 2021), 2021a. https://www.census.gov/data/datasets/2020/dec/2020-census-redistricting-summary-file-dataset.html.

U.S. Census Bureau. Comparing differential privacy with older disclosure avoidance methods. https://www.census.gov/content/dam/Census/library/factsheets/2021/comparing-differential-privacy-with-older-disclosure-avoidance-methods.pdf, 2021b.

U.S. Census Bureau. Disclosure avoidance for the 2020 Census: An introduction. Technical report, U.S. Government Publishing Office, Washington, D.C., Nov. 2021c.

US Census Bureau. Guidance for geography users: Hierarchy diagrams. https://www.census.gov/programs-surveys/geography/guidance/hierarchy.html, Oct. 2021.

U.S. Census Bureau. Next 2020 Census data products to be released in 2023 (April 27, 2022), 2022. https://www.census.gov/newsroom/press-releases/2022/2020-census-data-products-schedule-2023.html.

US Census Bureau. Developing the DAS: Demonstration data and progress metrics. https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance/2020-das-development.html, Apr. 2023a.

US Census Bureau. About 2020 Census data products. https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/release/about-2020-data-products.html, Apr. 2023b.

US Census Bureau. 2023-04-03 Privacy-loss budget allocations. Technical report, Apr. 2023c. https://www2.census.gov/programs-surveys/decennial/2020/program-management/data-product-planning/2010-demonstration-data-products/04-Demonstration_Data_Products_Suite/2023-04-03/2023-04-03_Privacy-Loss_Budget_Allocations.pdf [Accessed: 04-25-2023].

US Census Bureau. Factsheet on disclosure avoidance for the 2010 demonstration data products suite – Redistricting and Demographic and Housing Characteristics File – production settings (2023-04-03). Technical report, Apr. 2023d. https://www2.census.gov/programs-surveys/decennial/2020/program-management/data-product-planning/2010-demonstration-data-products/04-Demonstration_Data_Products_Suite/2023-04-03/2023-04-03_Factsheet.pdf [accessed 2023-04-25].

US Census Bureau. Release dates set for next 2020 Census data products; New reader-friendly disclosure avoidance briefs. https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance/newsletters/release-dates-and-da-briefs.html, Mar. 2023e.

US Census Bureau. Released for feedback: Detailed DHC-A proof of concept. https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance/newsletters/released-today-detailed-dhc-a-proof-of-concept.html, Jan. 2023f.

D. Van Riper, T. Kugler, and J. Schroeder. IPUMS NHGIS privacy-protected 2010 Census demonstration data, version 2021-06-08 [database]. Minneapolis, MN: IPUMS, 2020.

L. Wasserman and S. Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010.

A Proof of Theorem 5.6

In this Appendix, we prove that Algorithm 5.1 (the Permutation Algorithm) satisfies $(\boldsymbol{c}_{\text{Swap}}, d^u_{\text{HamS}}, \epsilon_{\mathcal{D}})$-DP for $\epsilon_{\mathcal{D}}$ given in Theorem 5.6. Assume throughout this Appendix the conditions of Theorem 5.6: that all variables in $\boldsymbol{X}$ are discrete; and that there is one matching variable, one swapping variable and one non-matching holding variable, with categories j, k and l respectively. Recall that $(J_i^{\boldsymbol{X}}, K_i^{\boldsymbol{X}}, L_i^{\boldsymbol{X}})$ are the i-th record's categories for the three variables, so that we can write $\boldsymbol{X}$ as

$$\boldsymbol{X} = \Big[(J_1^{\boldsymbol{X}}, K_1^{\boldsymbol{X}}, L_1^{\boldsymbol{X}}), \ldots, (J_N^{\boldsymbol{X}}, K_N^{\boldsymbol{X}}, L_N^{\boldsymbol{X}})\Big],$$

where $N = |\boldsymbol{X}|$ is the number of records in $\boldsymbol{X}$.

Let $\ell_1^u(\boldsymbol{X}, \boldsymbol{X}')$ be the ℓ_1-distance on the interior cells of the fully-saturated contingency table

$$\ell_1^u(\boldsymbol{X}, \boldsymbol{X}') := \sum_{j,k,l} \Big| n_{jkl}^{\boldsymbol{X}} - n_{jkl}^{\boldsymbol{X}'} \Big|, \tag{A.1}$$

where $n_{jkl}^{\boldsymbol{X}} = \sum_i 1_{J_i^{\boldsymbol{X}}=j} 1_{K_i^{\boldsymbol{X}}=k} 1_{L_i^{\boldsymbol{X}}=l}$.

Lemma A.1. *The ℓ_1^u-distance equals the symmetric-difference distance d^u_{SymDiff} defined in (3.5). Further, $\ell_1^u(\boldsymbol{X}, \boldsymbol{X}') = 2d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}')$ if $|\boldsymbol{X}| = |\boldsymbol{X}'|$.*

Lemma A.2. MULT *is a metric on the space of a.e. equal random variables (over the same probability space $\mathcal{T}$).*

Proof. It is easy to see that MULT is symmetric and $\text{MULT}(X, Y) = 0$ if and only if $X = Y$ a.e. All that remains is to verify the triangle inequality. Let $\{E_n\} \subset \mathcal{F}$ such that

$$\left| \ln \frac{\Pr(X \in E_n)}{\Pr(Z \in E_n)} \right| \to \text{MULT}(X, Z),$$

as $n \to \infty$. Then

$$\begin{aligned} \left| \ln \frac{\Pr(X \in E_n)}{\Pr(Z \in E_n)} \right| &\leq |\ln[\Pr(X \in E_n)] - \ln[\Pr(Y \in E_n)]| + |\ln[\Pr(Y \in E_n)] - \ln[\Pr(Z \in E_n)]| \\ &\leq \text{MULT}(X, Y) + \text{MULT}(Y, Z). \end{aligned}$$

□

Define $T(\boldsymbol{X}, U)$ to be the output of Algorithm 5.1 given the dataset $\boldsymbol{X}$ and random seed U as input. Following Section 3.1, T can be interpreted as a data-release mechanism. Let $P_{\boldsymbol{X}}$ be the probability distribution of $T(\boldsymbol{X}, U)$ induced by U.

Lemma A.3. *If $\boldsymbol{X}$ and $\boldsymbol{X}'$ differ only by reordering of rows – i.e. $d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}') = 0$, then* $\text{MULT}\left(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}\right) = 0$.

Proof. The contingency table $[n^{\boldsymbol{Z}}_{jkl}]$ is invariant to reordering of rows of $\boldsymbol{Z}$. Thus $P_{\boldsymbol{X}} = P_{\boldsymbol{X}'}$. □

Lemma A.4. *Fix some data universe $\mathcal{D} \in \operatorname{Im} \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}$ and some $\boldsymbol{X}, \boldsymbol{X}' \in \mathcal{D}$ with $d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}') = m > 0$. Then there exists a permutation σ which fixes $|\boldsymbol{X}| - m$ records such that $\sigma(\boldsymbol{X}) = \boldsymbol{X}'$ (up to re-ordering of records).*

We use the notation $\sigma(\boldsymbol{X})$ as shorthand to mean that we permute the swapping variable of records in $\boldsymbol{X}$ according to σ. For example, if $\boldsymbol{X} = [(J_i, K_i, L_i)]_{i=1}^N$ then $\sigma(\boldsymbol{X}) = [(J_i, K_{\sigma(i)}, L_i)]_{i=1}^N$.

Proof. We first establish some notation. For a matrix $\boldsymbol{M}$, define $\text{sub}(\boldsymbol{M})$ to be the matrix where the zero rows and columns of $\boldsymbol{M}$ have been removed. For example, if

$$\boldsymbol{M} = \begin{bmatrix} a & 0 & b \\ 0 & 0 & 0 \\ c & 0 & d \end{bmatrix}$$

for non-zero a, b, c, d then

$$\text{sub}(\boldsymbol{M}) = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

We have that $m < \infty$ since the invariants $\boldsymbol{c}_{\text{Swap}}$ imply that all datasets in $\mathcal{D}$ have the same number of records. Hence $\boldsymbol{X} \ominus \boldsymbol{X}'$ contain $2m$ records, with m records from $\boldsymbol{X}$ and m records from $\boldsymbol{X}'$. Denote the records in $\boldsymbol{X} \ominus \boldsymbol{X}'$ which come from $\boldsymbol{X}$ by $\boldsymbol{X}_0$ and the records from $\boldsymbol{X}'$ by $\boldsymbol{X}'_0$, so that $\boldsymbol{X} \ominus \boldsymbol{X}'$ is the disjoint union of $\boldsymbol{X}_0$ and $\boldsymbol{X}'_0$.

Without loss of generality, we may assume that there is a single matching category ($\mathcal{J} = 1$). (If there is more than one matching category, apply the following argument to each category separately.) Then the datasets $\boldsymbol{X}$ (disregarding the order of the records) can be represented as the matrix $[n_{kl}^{\boldsymbol{X}}]$.

We will need the following result ($*$): For any $\boldsymbol{X}'', \boldsymbol{X}''' \in \mathcal{X}$, the matrix $\boldsymbol{X}'' - \boldsymbol{X}''' = [n_{kl}^{\boldsymbol{X}''} - n_{kl}^{\boldsymbol{X}'''}]$ has zero row- and column-sums if and only if $\boldsymbol{X}'' \in \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X}''')$. Moreover, $\boldsymbol{X}'' \in \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X}''')$ implies $\boldsymbol{X}_0'' \in \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}(\boldsymbol{X}_0''')$ and $\boldsymbol{X}'' - \boldsymbol{X}''' = \boldsymbol{X}_0'' - \boldsymbol{X}_0'''$.

By the above results, the marginal counts of $\boldsymbol{X}_0$ and $\boldsymbol{X}_0'$ agree: $n_k^{\boldsymbol{X}_0} = n_k^{\boldsymbol{X}_0'}$ and $n_l^{\boldsymbol{X}_0} = n_l^{\boldsymbol{X}_0'}$ for all k and l. But the interior cells disagree: if $n_{kl}^{\boldsymbol{X}_0} > 0$ then $n_{kl}^{\boldsymbol{X}_0'} = 0$ (and visa versa, swapping $\boldsymbol{X}_0$ and $\boldsymbol{X}_0'$). Further $\boldsymbol{X}_0 - \boldsymbol{X}_0'$ has positive entries which sum to m and negative entries which sum to $-m$, and zero row- and column-sums. Permuting a record (K_i, L_i) will decrease $n_{K_i L_i}$ by 1 and increase $n_{K_{\sigma(i)}, L_i}$ by 1.

There is another key observation: By construction of $\boldsymbol{X}_0$ and $\boldsymbol{X}_0'$, if we can permute $\boldsymbol{X}_0$ to produce $\boldsymbol{X}_0'$ then we can use the same permutation to produce $\boldsymbol{X}'$ from $\boldsymbol{X}$. Critically, permutations of $\boldsymbol{X}_0$ can only use m records (since there are only m records in $\boldsymbol{X}_0$) and indeed must use m records to produce $\boldsymbol{X}_0'$ (since there are no records in common between $\boldsymbol{X}_0$ and $\boldsymbol{X}_0'$). Therefore we have reduced the problem: we need to find a permutation σ (regardless of the number of records it fixes) such that $\sigma(\boldsymbol{X}_0) = \sigma(\boldsymbol{X}_0')$.

We prove the result by induction on $d_{\text{HamS}}^u(\boldsymbol{X}, \boldsymbol{X}') = d_{\text{HamS}}^u(\boldsymbol{X}_0, \boldsymbol{X}_0') = m$. There are two base cases: The case $m = 1$ is vacuous since $d_{\text{HamS}}^u(\boldsymbol{X}, \boldsymbol{X}') = 1$ implies that $\boldsymbol{X}, \boldsymbol{X}'$ are not in the same data universe. Why? If $\ell_1^u(\boldsymbol{X}, \boldsymbol{X}') = 2$ then $\boldsymbol{X} - \boldsymbol{X}'$ only has one or two non-zero cells. But this implies $\boldsymbol{X} - \boldsymbol{X}'$ has a row or column with non-zero sum.

For the second base case ($m = 2$), result ($*$) implies that $\boldsymbol{M} = \boldsymbol{X}_0 - \boldsymbol{X}_0'$ has

$$\text{sub}(\boldsymbol{M}) = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

(up to re-ordering of rows and columns). Therefore $\boldsymbol{X}_0$ and $\boldsymbol{X}_0'$ differ by a single swap σ: if k, k', l, l' are indices such that $M_{kl} = M_{k'l'} = 1$ then define $\boldsymbol{X}_1$ by swapping the records (k, l) and (k', l') in $\boldsymbol{X}_0$. We have $\boldsymbol{X}_1 = \boldsymbol{X}_0'$ as desired.

This completes the base cases. Now we will prove the induction step. By ($*$), we can always re-order the rows and columns of $\boldsymbol{M} = \boldsymbol{X}_0 - \boldsymbol{X}_0'$ such that the 2×2 top-left submatrix looks like

$$\boldsymbol{M}_{1:2,1:2} = \begin{bmatrix} a & b \\ -c & d \end{bmatrix},$$

with $a, d, c > 0$. Define $\boldsymbol{X}_1$ by swapping the records $(1, 1)$ and $(2, 2)$ in $\boldsymbol{X}_0$. Then the top-left

submatrix of $\boldsymbol{M}' = \boldsymbol{X}_1 - \boldsymbol{X}_0'$ looks like

$$\boldsymbol{M}'_{1:2,1:2} = \begin{bmatrix} a-1 & b+1 \\ -c+1 & d-1 \end{bmatrix},$$

and the rest of $\boldsymbol{M}'$ is the same as $\boldsymbol{M}$. If $b < 0$ then $\ell_1^u(\boldsymbol{X}_1, \boldsymbol{X}_0') = \ell_1(\boldsymbol{M}') = \ell_1(\boldsymbol{M}) - 4$. If $b \geq 0$ then $\ell_1(\boldsymbol{M}') = \ell_1(\boldsymbol{M}) - 2$. In both cases, we can use the induction hypothesis to give us a permutation σ_1 of $\boldsymbol{X}_1$ which produces $\boldsymbol{X}_0'$. Define the permutation σ as the composition of σ_1 with the swap of $(1,1)$ and $(2,2)$. □

Proof of Theorem 5.6. Fix $\boldsymbol{X}, \boldsymbol{X}'$ in the same data universe $\mathcal{D} \in \operatorname{Im} \mathscr{D}_{\boldsymbol{c}_{\text{Swap}}}$. Recall

$$b = \max\{0, n_j \mid \text{there are at least two different records in stratum } j\}.$$

If $b = 0$, then $\boldsymbol{X}, \boldsymbol{X}'$ only differ by reordering of rows and hence $\epsilon_{\mathcal{D}} = 0$ by Lemma A.3. Having taken care of the case $b = 0$, from herein we may assume $b \geq 2$. (The case $b = 1$ is not possible.)

Suppose that $d^u_{\text{HamS}}(\boldsymbol{X}, \boldsymbol{X}') = \Delta$. By Lemma A.4, there exists a derangement ρ of Δ records such that $\rho(\boldsymbol{X}') = \boldsymbol{X}$ (up to re-ordering of records).

We need to prove that $\textsc{Mult}(P_{\boldsymbol{X}}, P_{\boldsymbol{X}'}) \leq \Delta\epsilon$ or equivalently

$$\Pr\left[H(\sigma(\boldsymbol{X})) = H(\boldsymbol{Z})\right] \leq \exp(\Delta\epsilon) \Pr\left[H(\sigma(\boldsymbol{X}')) = H(\boldsymbol{Z})\right],$$

for all $\boldsymbol{Z} \in \mathcal{X}$, where the probability is over the random permutation σ of $\boldsymbol{V}_{\text{Swap}}$ sampled by Algorithm 5.1.[40] Interpreting equality of datasets by disregarding the ordering of records (as we will do for the remainder of the proof), this is equivalent to

$$\Pr\left[\sigma(\boldsymbol{X}) = \boldsymbol{Z}\right] \leq \exp(\Delta\epsilon) \Pr\left[\sigma(\boldsymbol{X}') = \boldsymbol{Z}\right]. \tag{A.2}$$

Suppose $p = 0$. Then σ must be the identity. Thus $\Pr(\sigma(\boldsymbol{X}') = \boldsymbol{X}) = 0$ but $\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{X}) = 1$, which implies (A.3) cannot be satisfied for any $m > 0$. Then $\epsilon_{\mathcal{D}}$ cannot be finite for any choice of $\mathcal{D}$.

Now suppose that $p = 1$. We will show that there exists some data universes $\mathcal{D}$ such that $\epsilon_{\mathcal{D}}$ cannot be finite. For example, suppose that $\boldsymbol{X}$ and $\boldsymbol{X}'$ differ by a single swap between the first and second records and a derangement ρ' on the other records:

$$\boldsymbol{X} = \left[(j,k,l), (j,k',l'), \boldsymbol{X}^*\right],$$
$$\boldsymbol{X}' = \left[(j,k',l), (j,k,l'), \rho(\boldsymbol{X}^*)\right],$$

[40] Note that in the pseudocode for Algorithm 5.1, σ denotes a derangement of the selected records while here σ is a *permutation* which fixes the $\boldsymbol{V}_{\text{Swap}}$ of the unselected records.

where $\boldsymbol{X}^* = \{(J_i, K_i, L_i), i = 3, \ldots, N\}$, and $k \neq k'$ and $l \neq l'$. Suppose

$$n^{\boldsymbol{X}}_{jk} = n^{\boldsymbol{X}}_{jl} = n^{\boldsymbol{X}}_{jk'} = n^{\boldsymbol{X}}_{jl'} = 1.$$

Then $n^{\boldsymbol{X}}_{jkl} = n^{\boldsymbol{X}}_{jk'l'} = 1$ and $n^{\boldsymbol{X}}_{jk''l} = n^{\boldsymbol{X}}_{jkl''} = 0$ for all k'' and l''. Since no records can be fixed by σ when $p = 1$, we have $\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{X}) = 0$ but $\Pr(\sigma(\boldsymbol{X}') = \boldsymbol{X}) > 0$. So we cannot satisfy (A.3) for this data universe $\mathcal{D}$ with any finite $\epsilon_{\mathcal{D}}$.

The rest of the proof examines the case where $0 < p < 1$. Now Algorithm 5.1 allows for permutations which are not derangements. Since $\boldsymbol{X}$ and $\boldsymbol{X}'$ themselves differ by a permutation ρ, we can permute $\boldsymbol{X}$ to produce $\boldsymbol{Z}$ if and only if we can permute $\boldsymbol{X}'$ to produce $\boldsymbol{Z}$. (That is, the orbit spaces (under the action of permutation) of $\boldsymbol{X}$ and $\boldsymbol{X}'$ are both equal to $\mathcal{D}$. This result is given by Lemma A.4.) Thus, either $\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{Z})$ and $\Pr(\sigma(\boldsymbol{X}') = \boldsymbol{Z})$ are both zero, or they are both non-zero. We need only focus on the case where both probabilities are non-zero.

Because we perform random selection and swamping independently for each j, we can decompose the overall $\sigma = \{\sigma_1, \ldots, \sigma_J\}$, where σ_j will leave any unit i with matching category $J_i \neq j$ untouched. Write $\boldsymbol{X}_j$ for the records of $\boldsymbol{X}$ with $J_i = j$. Simplifying

$$\frac{\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{Z})}{\Pr(\sigma(\boldsymbol{X}') = \boldsymbol{Z})} = \frac{\prod_{j=1}^{J} \Pr(\sigma_j(\boldsymbol{X}_j) = \boldsymbol{Z}_j)}{\prod_{j=1}^{J} \Pr(\sigma_j(\boldsymbol{X}'_j) = \boldsymbol{Z}_j)}.$$

Our goal (A.2) is equivalent to proving

$$\frac{\Pr(\sigma_j(\boldsymbol{X}_j) = \boldsymbol{Z}_j)}{\Pr(\sigma_j(\boldsymbol{X}'_j) = \boldsymbol{Z}_j)} \leq \exp(\Delta_j \epsilon), \tag{A.3}$$

for all j where $\Delta_j = d^u_{\text{HamS}}(\boldsymbol{X}_j, \boldsymbol{X}'_j)$.

Fix some j. For notation simplicity, whenever it is not essential to indicate the role of j, we will drop the subscript j for the rest of the proof (until the end when we need to optimize over j). (This is the same as assuming $\boldsymbol{V}_{\text{Match}}$ is empty.)

Let $G_{\boldsymbol{X} \to \boldsymbol{Z}} = \{\text{permutation } g : g(\boldsymbol{X}) = \boldsymbol{Z}\}$. We use the notation g instead of σ to emphasise that g is not random, while the permutation σ chosen by Algorithm 5.1 is random.

There is a bijection between $G_{\boldsymbol{X} \to \boldsymbol{Z}}$ and $G_{\boldsymbol{X}' \to \boldsymbol{Z}}$ given by $g \mapsto g \circ \rho$. Since

$$\Pr(\sigma(\boldsymbol{X}) = \boldsymbol{Z}) = \sum_{g \in G_{\boldsymbol{X} \to \boldsymbol{Z}}} \Pr(\sigma = g),$$

we will prove (A.3) by showing

$$\Pr(\sigma = g) \leq \exp(\Delta\epsilon) \Pr(\sigma = g \circ \rho),$$

for all $g \in G_{\boldsymbol{X} \to \boldsymbol{Z}}$. (Note that this may not obtain the best possible bound for specific $\boldsymbol{X}$ and $\boldsymbol{X}'$,

but it is mathematically easier to bound $\Pr(\sigma = g)/\Pr(\sigma = g \circ \rho)$ than bound the desired ratio

$$\frac{\sum_{g \in G_{\boldsymbol{X} \to \boldsymbol{Z}}} \Pr(\sigma = g)}{\sum_{g \in G_{\boldsymbol{X} \to \boldsymbol{Z}}} \Pr(\sigma = g \circ \rho)}$$

directly. Yet in the case where $G_{\boldsymbol{X} \to \boldsymbol{Z}}$ and $G_{\boldsymbol{X}' \to \boldsymbol{Z}}$ are singletons, this approach gives tight bounds.)

Let m_g be the number of records (in category j) which were deranged by g and let $d(m)$ denote the m-th derangement number (i.e. the number of derangements of size m):

$$\begin{aligned} d(m) &= m! \sum_{k=0}^{m} \frac{(-1)^k}{k!} \\ &= m d(m-1) + (-1)^m \qquad \text{for } m \geq 0. \end{aligned} \tag{A.4}$$

Fix $g \in G_{\boldsymbol{X} \to \boldsymbol{Z}}$ and $g' = g \circ \rho$. We now compute $\Pr(\sigma = g)$. The permutation g is sampled in Algorithm 5.1 via a two-step procedure. Firstly records are independent selected for derangement with probability p. Suppose that g deranges records $\{i_1, \ldots, i_{m_g}\}$. Since we disallow the possibility of selecting only one record,

$$\Pr(\text{the selected records are } \{i_1, \ldots, i_{m_g}\}) = \frac{p^{m_g}(1-p)^{n-m_g}}{1 - \Pr(\text{exactly 1 record selected})}.$$

Secondly we sample uniformly from the set of all derangements of m_g records. We sample g with probability $[d(m_g)]^{-1}$. Thus,

$$\Pr(\sigma = g) = \frac{p^{m_g}(1-p)^{n-m_g}}{[1 - \Pr(\text{exactly 1 record selected})]d(m_g)},$$

and so

$$\frac{\Pr(\sigma = g)}{\Pr(\sigma = g')} = o^{\delta} \frac{d(m_g - \delta)}{d(m_g)}, \tag{A.5}$$

where $o = p/(1-p)$ and $\delta = m_g - m_{g'}$.

Our aim is now to bound the RHS of (A.5) by $\exp(\Delta\epsilon)$. Since g' and g differ only by the permutation ρ (which fixes $n - \Delta$ records), we must have $m_g - \Delta \leq m_{g'} \leq m_g + \Delta$. Therefore, there are at most $2\Delta + 1$ possible cases:

$$\begin{aligned} \delta \in S &= \left\{\delta \in \mathbb{Z} \mid -\Delta \leq \delta \leq \Delta \text{ and } \left(m_g - \delta = 0 \text{ or } 2 \leq m_g - \delta \leq n\right)\right\} \\ &= \left\{\delta \in \mathbb{Z} \mid \max(-\Delta, m_g - n) \leq \delta \leq \min(\Delta, m_g) \text{ and } \delta \neq m_g - 1\right\}. \end{aligned}$$

Suppose $0 < p \leq 0.5$. Since $d(m)$ is non-decreasing (except at $m = 1$ which is not realisable by g or g') and $\frac{1-p}{p} \geq 1$, the RHS of (A.5) is maximised when $m_{g'_j} = n_j$ and $m_{g_j} = n_j - \Delta_j$ (i.e. $\delta = -\Delta_j$),

in which case

$$\begin{aligned}\frac{\Pr(\sigma = g)}{\Pr(\sigma = g')} &= o^{-\Delta} \prod_{j=1}^{J} \frac{d(n_j)}{d(n_j - \Delta_j)} \\ &\leq o^{-\Delta} \prod_{j=1}^{J} (n_j + 1)^{\Delta_j} \\ &\leq o^{-\Delta} (b+1)^{\Delta} \\ &= \exp(\Delta\epsilon),\end{aligned}$$

for $\epsilon = \ln(b+1) - \ln o$. The second line uses Lemma A.5.

Now suppose $0.5 < p < 1$. In the case of $\delta_j = \Delta_j$, the ratio (A.5) is maximised at o^{Δ_j} when $m_g = \Delta_j$. Moreover, o^{Δ_j} also dominates $o^{\delta_j} \frac{d(m_{g_j} - \delta_j)}{d(m_{g_j})}$ for all $0 \leq \delta_j < \Delta_j$ and all possible m_{g_j}. Thus,

$$\begin{aligned}\frac{\Pr(\sigma = g)}{\Pr(\sigma = g')} &\leq \prod_{j=1}^{J} \max\left\{ o^{\Delta_j}, o^{\delta_j} \frac{d(m_{g_j} - \delta_j)}{d(m_{g_j})} : \delta_j \in S_j \text{ and } \delta_j < 0 \right\} \\ &\leq \prod_{j=1}^{J} \max\left\{ o^{\Delta_j}, o^{\delta_j} (m_{g_j} - \delta_j + 1)^{-\delta_j} : \delta_j \in S_j \text{ and } \delta_j < 0 \right\} \\ &\leq \prod_{j=1}^{J} \max\left\{ o^{\Delta_j}, o^{-\delta_j} (n_j + 1)^{\delta_j} : 0 < \delta_j \leq \Delta_j \right\} \\ &\leq \max\left\{ o^{\Delta}, o^{-\delta} (b+1)^{\delta} : 0 < \delta \leq \Delta \right\}.\end{aligned}$$

If $o^{-1}(b+1) \geq 1$ then $o^{-\delta}(b+1)^{\delta}$ is maximised at $\delta = \Delta$. Otherwise $o^{-\delta}(b+1)^{\delta} < 1 < o^{\Delta}$. Hence

$$\frac{\Pr(\sigma = g)}{\Pr(\sigma = g')} \leq \exp(\Delta\epsilon),$$

for $\epsilon = \max\big(\ln o, \ln(b+1) - \ln o\big)$. □

Lemma A.5. *For any $m \in \mathbb{N}$ and any $a \in \mathbb{N}$ satisfying $0 \leq a \leq m$ and $a \neq m - 1$,*

$$\frac{d(m)}{d(m-a)} \leq (m+1)^a.$$

Proof. We use induction on m. The base cases $m = 0, 1, 2$ are straightforward to verify since $d(0) = d(2) = 1$ and $d(1) = 0$. For the induction step, we can assume $m \geq 3$ so that $d(m-1) \geq 1$

and hence

$$\begin{aligned}\frac{d(m)}{d(m-a)} &= \frac{d(m)}{d(m-1)}\frac{d(m-1)}{d(m-a)} \\ &\leq \frac{d(m)}{d(m-1)} m^{a-1}\end{aligned}$$

by the induction hypothesis. The result then follows by the identity (A.4):

$$\begin{aligned}\frac{d(m)}{d(m-1)} &= \frac{md(m-1)+(-1)^m}{d(m-1)} \\ &\leq m+1.\end{aligned}$$

□