SafeTab-P: Disclosure Avoidance for the 2020 Census Detailed Demographic and Housing Characteristics File A (Detailed DHC-A)

Sam Haney[1], Skye Berghel[1], Bayard Carlson[1], Ryan Cumings-Menon[2], Luke Hartman[1], Michael Hay[1], Ashwin Machanavajjhala[1], Gerome Miklau[1], Amritha Pai[1], Simran Rajpal[1], David Pujol[1], William Sexton[1], Ruchit Shrestha[1], and Daniel Simmons-Marengo[1]

[1]Tumult Labs
[2]U.S. Census Bureau

August 23, 2024

Abstract

This article describes the disclosure avoidance algorithm that the U.S. Census Bureau used to protect the Detailed Demographic and Housing Characteristics File A (Detailed DHC-A) of the 2020 Census. The tabulations contain statistics (counts) of demographic characteristics of the entire population of the United States, crossed with detailed races and ethnicities at varying levels of geography. The article describes the SafeTab-P algorithm, which is based on adding noise drawn to statistics of interest from a discrete Gaussian distribution. A key innovation in SafeTab-P is the ability to adaptively choose how many statistics and at what granularity to release them, depending on the size of a population group. We prove that the algorithm satisfies a well-studied variant of differential privacy, called zero-concentrated differential privacy (zCDP). We then describe how the algorithm was implemented on Tumult Analytics and briefly outline the parameterization and tuning of the algorithm.

Contents

1 Introduction

Every 10 years, the U.S. Census Bureau carries out its constitutional mandate to conduct a census of the U.S. population. The 2020 Census is the latest of such efforts, which aims to completely enumerate every person living in the United States. The Herculean task of "counting everyone once, only once, and in the right place" for the 2020 Census involves 35 operations (e.g., Address Canvassing, Nonresponse Followup, Redistricting Data Program) [2]. Each of these operations is responsible for a number of systems handling various aspects of the entire census undertaking, ranging from data collection and processing to the dissemination of data products to the American people. The Disclosure Avoidance System (DAS) is one small component of this vast operational ecosystem. The DAS is technically a subsystem of the Decennial Response Processing System and it belongs to the Data Products and Dissemination (DPD) operation. The purpose of the DAS is to ensure that data shared with the public does not violate the confidentiality of individual respondents to the census. The DAS executes its duties after census responses have been collected and (eventually) processed into a database known as the Census Edited File (CEF). Then, the Decennial Tabulation System and Center for Enterprise Dissemination Services and Consumer Innovation, both of which are also systems of the DPD operation, finish preparing the data products for public consumption. For the 2020 Census, the DAS was overhauled and rebuilt from the ground up, compared to previous censuses, to modernize its privacy protection mechanisms. The modernization effort is intended to provide individuals with the best possible protection against privacy threats, known or unknown, given today's technological landscape by deploying a DAS redesigned on cutting-edge scientific research.

DPD releases a variety of different data products for the 2020 Census, including the Census Redistricting Data (P.L. 94-171) Summary File, Demographic and Housing Characteristics File (DHC), Demographic Profile, Detailed Demographic and Housing Characteristics File A (Detailed DHC-A), Detailed DHC-B, and Supplemental DHC (S-DHC). Each of these products are distinct and present their own unique set of challenges with regard to disclosure avoidance. Hence, the DAS deploys a variety of privacy algorithms to optimize protection and accuracy across the various data products. We note that some products share similar enough challenges to utilize the same algorithm. Despite algorithmic differences, all statistical disclosure limitation techniques fit into the same overarching privacy framework known as *differential privacy*. The differential privacy framework calls for the design of algorithms that satisfy mathematically provable guarantees regarding the data publication process. See Section 4.1.1 for further details. In this paper, we use "differential privacy" loosely to mean pure differential privacy or one of its several variants. In particular, one should assume, by default, that we are referring to zero-concentrated differential privacy unless otherwise specified. As with pure differential privacy, variants such as zero-concentrated differential privacy are based on mathematically rigorous privacy definitions.

The focus of this paper, SafeTab-P, is one of the several privacy algorithms deployed by the DAS. SafeTab-P is designed specifically to provide differential privacy guarantees for the production and release of the Detailed DHC-A (see Section 2 for a description of the data product and privacy release problem).

The main goals of this article are threefold:

1. Describe the SafeTab-P algorithm and how it meets the requirements of the Detailed DHC-A.

2. Prove the privacy properties of the SafeTab-P algorithm.

3. Describe the parameters in the SafeTab-P algorithm and how they impact privacy-accuracy tradeoffs.

On the first point, we aim to provide a technical pseudocode description of the algorithm that articulates the functional mechanics of how privacy protection is executed on the Census data to produce the Detailed DHC-A tabular summaries (see Section 3). We will also highlight salient differences between our pseudocode abstraction of SafeTab-P and the actual implementation of the algorithm [1] (see Section 5). The SafeTab-P algorithm is open-sourced to the public and this article may, in some regard, be viewed as a helpful companion guide to the code. It provides meaningful context for those interested in understanding the implemented algorithm. However, this article is not a code specification document, per se. That is, a deep-dive into the code architecture (e.g., module interactions and class descriptions) is out of scope.

On the second point, we will provide relevant background material on the differential privacy framework and explain why SafeTab-P fits into this framework (see Section 4). Since this paper primarily emphasizes the privacy properties of SafeTab-P, it is not intended as a user guide to the data product. Guidance for users can be found in the Detailed DHC-A technical documentation produced by the Census Bureau [1].

Finally, as the third point indicates, we discuss the parameters in SafeTab-P that impact the privacy-accuracy tradeoff of the algorithm. We cover parameter tuning, including a brief look at related data accuracy considerations (see Section 6). Parameter tuning requires correctly identifying the universe of parameters and understanding the trade-offs inherent in adjusting these parameters.

2 Problem & Desiderata

The Detailed DHC-A consists of statistics (counts) of demographic characteristics for all persons in the United States and Puerto Rico crossed with detailed races and ethnicities at varying levels of geography. The demographic characteristics include population counts and sex by age statistics for 2996 race and ethnicity characteristic iterations (defined in Section 2.2). Despite the name, the Detailed Demographic and *Housing* Characteristic File A product does not provide any data on households. Household data for detailed race and ethnicity groups and American Indian and Alaska Native tribes and villages is available in the Detailed DHC-B. The Census Bureau separated these products "to better facilitate developing disclosure protections for these complex data" [1]. In this section, we define relevant concepts, outline the precise statistics to be released, and then formulate the differentially private algorithm design problem.

2.1 Geography

Every person resides in exactly one Census block that determines their geographic location. Census blocks are the most granular form of *geographic entities*. All other geographic entities (e.g., L.A. county, the state of CA, and the United States) are aggregations of Census blocks. Geographic entities are divided into *geographic summary levels*. A geographic summary level is a set of nonoverlapping geographic entities, such as the set of all states, or the set of all counties. Detailed DHC-A produces statistics for the following geographic summary levels:

- Nation

- State or State equivalent

- County or County equivalent

- Census Tract

- Place

- American Indian, Alaska Native, and Native Hawaiian (AIANNH) areas.

Henceforth, we tend to write State (County) without including the "or State (County) equivalent" qualifier although one should assume the qualifier when applicable. Washington, D.C. is an example of a State equivalent.

2.2 Race and Ethnicity

The U.S. Census Bureau collects detailed race and ethnicity data from individuals in accordance with the 1997 Federal Register Notice "Revisions to the Standards for the Classification of Federal Data on Race and Ethnicity" released by the Office of Management and Budget (OMB).

Every person is associated with one or more *race codes* and, per OMB guidelines, a single *ethnicity code*. The maximum number of race codes, called the *race multiplicity*, that a person can be associated with is limited to 8 by the census data collection procedures. A *race group* is a set of race codes. Similarly, an ethnicity group is a set of ethnicity codes.

Detailed race or ethnicity groups are the most disaggregated racial or ethnic group classifications for which the Census Bureau publishes data. Examples of Detailed racial or ethnic groups include Basque, Dutch, Guatemalan, Puerto Rican, Ethiopian, Nigerian, Mongolian, Thai, Apache, and Navajo Nation. The OMB-specified *Major* race categories are aggregated race groupings that represent the minimum allowable race classification system for which census data may be published. Major race grouping include White, Black or African American, American Indian or Alaska Native, Asian, Native Hawaiian or Other Pacific Islander, and Some Other Race. The aggregated ethnic equivalent of the Major race groups is a coarse binary classification (Hispanic or Latino, Not Hispanic or Latino). *Regional* race or ethnicity groups provide an intermediate level of aggregation between Detailed and Major. Examples of Regional racial or ethnic groups include European, Central American, Caribbean, Sub-Saharan African, Alaska Native, and American Indian. Note that, while Regional groups are often distinguished by geographic location (Europe, Sub-Saharan Africa, Caribbean, etc.), the Regional concept pertains to a classification of race and ethnicity; it does not pertain to geography. Subject-matter experts at the Census Bureau determine which race or ethnicity groups are Detailed and which groups are Regional. For the purposes of SafeTab-P, we take these classifications as given exogenous factors. The universe specification of valid race and ethnicity groups as well as the classification into Detailed or Regional groups began well before data collection for the 2020 Census [10]. Appendix G of [1] provides a complete enumeration of the detailed race and ethnicity groups.

An individual person is in a race group *Alone* if all race codes associated with that individual are contained in the race group. For example, if an individual self-identifies with the single race code for Navajo Nation and no other race codes, said individual would belong to the Detailed race group Navajo Nation Alone. Alternatively, an individual may report multiple race codes (e.g., British, Scottish, and Dutch) that aggregate into the same Regional group (e.g., European Alone). A person is in a race group *Alone or in Any Combination* if some race code associated with that person is contained in the race group. This concept pertains to individuals that self-identify with a single race (e.g., British) or with multiple races (e.g., British and Thai). In both examples, the individual belongs to the Detailed British race group Alone or in Any Combination. The individual also belongs to the Regional European race group Alone or in Any Combination.

Since all individuals are only associated with a single ethnicity code, all ethnicity groups are Alone groups.

A *characteristic iteration* is the combination of a race (or ethnicity) group, along with the specification of either Alone or Alone or in Any Combination (e.g., Latin American Indian Alone or in Any Combination is a characteristic iteration). One person may be associated with multiple characteristic iterations. Like geographical entities, characteristic iterations are also divided into *characteristic iteration levels*. We have already seen the defining aspect of these iteration levels: namely, the concepts of *Detailed* and *Regional* race groups. The Detailed characteristic iteration level consists of the set of characteristic iterations for all Detailed race and ethnicity groups Alone as well as for all Detailed race groups Alone or in Any Combination (e.g., Chinese Alone, Chinese Alone or in Any Combination, Celtic Alone, and Celtic Alone or in Any Combination). The Regional characteristic iteration level consists of the set of characteristic iterations for all Regional race and ethnicity groups Alone as well as for all Regional race groups Alone or in Any Combination (e.g., Middle Eastern or North African Alone, Middle Eastern or North African Alone or in Any Combination, Polynesian Alone, and Polynesian Alone or in Any Combination). We intentionally omit the notion of a Major race and ethnicity characteristic iteration level, as no statistics for this level will be produced by the SafeTab-P algorithm for the Detailed DHC-A.

2.3 Population Groups

A *population group* is a pair (g, c), where g is a geographic entity (e.g., the state of NC or L.A. County) and c is a race or ethnicity characteristic iteration (e.g., Latin American Indian Alone or in Any Combination). Population groups are divided into *population group levels*. We will often identify a population group level by specifying a (geography level, characteristic iteration level) pair. However, each population group level is really a set of population groups, where each population group's geographic entity belongs to the specified geography level and its characteristic iteration belongs to the specified characteristic iteration level. More formally, the Detailed DHC-A requires the publication of statistics for the following population group levels:

- (Nation, Detailed) $\equiv \{(g, c) : g$ is the Nation, c is a Detailed characteristic iteration$\}$

- (State, Detailed) $\equiv \{(g, c) : g$ is a State, c is a Detailed characteristic iteration$\}$

- (County, Detailed) $\equiv \{(g, c) : g$ is a County), c is a Detailed characteristic iteration$\}$

- (Tract, Detailed) $\equiv \{(g, c) : g$ is a Census Tract, c is a Detailed characteristic iteration$\}$

- (Place, Detailed) $\equiv \{(g, c) : g$ is a Place, c is a Detailed characteristic iteration$\}$

- (AIANNH, Detailed) $\equiv \{(g, c) : g$ is an AIANNH area, c is a Detailed characteristic iteration$\}$

- (Nation, Regional) $\equiv \{(g, c) : g$ is the Nation, c is a Regional characteristic iteration$\}$

- (State, Regional) $\equiv \{(g, c) : g$ is a State, c is a Regional characteristic iteration$\}$

- (County, Regional) $\equiv \{(g, c) : g$ is a County, c is a Regional characteristic iteration$\}$

- (Tract, Regional) $\equiv \{(g, c) : g$ is a Census Tract, c is a Regional characteristic iteration$\}$

- (Place, Regional) $\equiv \{(g, c) : g$ is a Place, c is a Regional characteristic iteration$\}$

In practice, some Detailed or Regional characteristic iterations may be omitted from the tabulations in a geography level. In other words, the above population group levels are subsets of the designated sets. This is done in accordance with specifications for the Detailed DHC-A provided by the Census Bureau. There is no concise representation for the exact level sets. The population group level (AIANNH, Regional) is intentionally omitted from the Detailed DHC-A.

One person may belong to multiple population groups in the set that comprises a population group level. For example, an individual who resides in Texas and reports Kenyan and Ghanaian races would belong to the (Texas, Kenyan Alone or in Any Combination) and the (Texas, Ghanaian Alone or in Any Combination) population groups which are both contained in the population group level identified by (State, Detailed). A person who resides in Schuyler County, NY and only reports a single race of Dutch would still belong to the (Schuyler County, NY, Dutch Alone) and the (Schuyler County, NY, Dutch Alone or in Any Combination) population groups which are both contained in the level identified by (County, Detailed). One person may be connected with population groups across multiple population group levels. For example, the Dutch individual residing in Schuyler County, NY would additionally belong to the (NY, Dutch Alone) and (NY, Dutch Alone or in Any Combination) population groups in the (State, Detailed) level, the (Schuyler County, NY, European Alone) and (Schuyler County, NY, European Alone or in Any Combination) population groups in the (County, Regional) level, etc. It is possible for an individual to not belong to any population groups in a particular level. Specifically, because AIANNH areas do not cover the United States, an individual who resides outside all designated AIANNH areas does not belong to any population groups in the (AIANNH, Detailed) level.

Because the geographic entities in a geography level are disjoint (a person cannot reside in both Schuyler County, NY and Fairfax County, VA), an individual's characteristic iterations primarily determine the number of population groups the individual belongs to in each level. A person associated with the maximum of 8 race codes and a single ethnicity code could belong to at most 9 Detailed characteristic iterations (8 Alone or in Any Combination race groups and 1 Alone ethnic group) and, similarly, at most 9 Regional characteristic iterations. Thus, for any given population group level, the maximum number of population groups an individual may contribute to is 9.

2.4 Detailed Demographic and Housing Characteristics File A

We are now prepared to define the Detailed DHC-A. The product aims to tabulate statistics by population groups.

The following statistical tables are released for each eligible population group as part of Detailed DHC-A.[2]

- A total population table (T01001) associated with each population group. See Table 1.

- Sex by age counts for a subset of population groups. The sex by age tables come in three different variants (T02001, T02002, T02003) for reasons explained later. See Tables 2, 3, 4.

T01001 Total Population
Universe: Total Population
Total

Table 1: The T01001 table contains counts of persons for all eligible population groups.

---

[2]Population groups may be deemed ineligible to receive certain statistics for a variety of reasons discussed throughout the document, such as groups pre-specified to only receive T01001 counts and groups that may fail to meet certain population thresholds).

T02001 Sex by Age(4)
Universe: Total Population
Total
    Male
        Under 18 years
        18 to 44 years
        45 to 64 years
        65 years and over
    Female
        Under 18 years
        18 to 44 years
        45 to 64 years
        65 years and over

Table 2: The T02001 Sex by Age (4) table contains counts of persons by sex and age in 4 bins for each eligible population group.

T02002 Sex by Age(9)
Universe: Total Population
Total
    Male
        Under 5 years
        5 to 17 years
        18 to 24 years
        25 to 34 years
        35 to 44 years
        45 to 54 years
        55 to 64 years
        65 to 74 years
        75 years and over
    Female
        Under 5 years
        5 to 17 years
        18 to 24 years
        25 to 34 years
        35 to 44 years
        45 to 54 years
        55 to 64 years
        65 to 74 years
        75 years and over

Table 3: The T02002 Sex by Age (9) table contains counts of persons by sex and age in 9 bins for each eligible population group.

T02003 Sex by Age(23)
Universe: Total Population
Total
    Male
        Under 5 years
        5 to 9 years
        10 to 14 years
        15 to 17 years
        18 and 19 years
        20 years
        21 years
        22 to 24 years
        25 to 29 years
        30 to 34 years
        35 to 39 years
        40 to 44 years
        45 to 49 years
        50 to 54 years
        55 to 59 years
        60 to 61 years
        62 to 64 years
        65 and 66 years
        67 to 69 years
        70 to 74 years
        75 to 79 years
        80 to 84 years
        85 years and over
    Female
        Under 5 years
        5 to 9 years
        10 to 14 years
        15 to 17 years
        18 and 19 years
        20 years
        21 years
        22 to 24 years
        25 to 29 years
        30 to 34 years
        35 to 39 years
        40 to 44 years
        45 to 49 years
        50 to 54 years
        55 to 59 years
        60 to 61 years
        62 to 64 years
        65 and 66 years
        67 to 69 years
        70 to 74 years
        75 to 79 years
        80 to 84 years
        85 years and over

Table 4: The T02003 Sex by Age(23) table contains counts of persons by sex and age in 23 bins for each eligible population group.

Some population groups are pre-determined to only receive T01001 statistics. These groups will be called *TotalOnly* population groups throughout the document. The TotalOnly population groups are only tabulated at the Nation and State geography levels. That is, TotalOnly is a subset of the set union of (Nation, Detailed), (Nation, Regional), (State, Detailed), (State, Regional). The categorization of population groups as TotalOnly is exogenous to the SafeTab-P program. As with other population group categorizations, this determination is made by subject-matter experts outside the DAS. The Detailed DHC-A technical documentation explains, "detailed groups with a national population less than 50 in the 2010 Census were preset to only receive nation and state level total population counts" [1]. Importantly, the classification was not based on collected 2020 Census responses and thus, does not impact the confidentiality protections afforded by the SafeTab-P program.

2.5   Private Release Problem

The release of statistical data products by the U.S. Census Bureau about persons and households is regulated under Title 13 and any release of statistics about persons in the United States must be afforded strong privacy protections [3]. Moreover, it has been demonstrated that legacy statistical disclosure limitation (SDL) techniques are vulnerable to attacks that can reconstruct the sensitive person records from aggregate statistics [6]. Hence, the U.S. Census Bureau decided to release many of the 2020 Census data products, including Detailed DHC-A, using algorithms that satisfy modern privacy definitions like differential privacy [4].

In particular, we describe a disclosure avoidance technique for the Detailed DHC-A that was designed to satisfy the following desiderata:

- *Privacy:* The algorithm must ensure end-to-end zero-concentrated differential privacy with respect to (the addition/removal of) every person in the United States and Puerto Rico.

- *Population Groups:* The algorithm must release statistics for a predefined set of race and ethnicity characteristic iterations and the following geographies: nation, states, counties, Census tracts, places, and all areas designated as AIANNH areas.

- *Adaptivity:* The algorithm may adaptively choose the granularity at which sex by age statistics are released. For instance, for population groups with a few people, the sex by age histogram may only have 4 buckets of age (as seen in Table 2, while for population groups with many people a more detailed histogram may be released (such as the 23 age bins shown in Table 4).

- *Accuracy:* The algorithm must achieve pre-specified accuracy levels for population groups in terms of the margin of error (MOE) in output counts. Different population groups may have different MOEs specified (described later in the paper in Table 7). The MOE discussed in the paper captures error induced by disclosure avoidance induced and does not capture other sources of error such as under/over counting in the census.

- *Integrality:* The output statistics must be integers.

- *Minimal Consistency:* The algorithm is not required, in general, to ensure consistency. That is, different counts output by the system need not be consistent with each other (e.g., the number of people of a certain characteristic iteration in the United States need not equal the sum of the population counts for the same characteristic iteration across all states). We also note that no consistent is enforced with other data products such as the DHC. However, some

postprocessing of outputs is done to address specific demographic reasonableness concerns. These postprocessing steps are discussed in Section 5.3.

In the rest of the paper, we describe the SafeTab-P differential privacy algorithm, discuss implementation and parameter tuning, and analyze bounds on the privacy loss achievable while satisfying the constraints mentioned above.

3   SafeTab-P Algorithm

SafeTab-P is a privacy algorithm for releasing detailed race and ethnicity statistics from the 2020 Census. The algorithm must accommodate the release of tabulations for total counts by detailed race and ethnicity and tabulations for sex by age counts by detailed race and ethnicity for various geographic summary levels. The algorithm acts on a private dataframe derived from the 2020 Census. The algorithm does not re-use noisy estimates or privacy-loss budget from other 2020 data products such as the DHC. In this section, we will describe the algorithm as applied to the United States. Puerto Rico is discussed in Section 5.4.2.

3.1   Input Data Description

The 2020 CEF is a relational database consisting of multiple person and household attributes spread across several linked dataframes. Many of these attributes are irrelevant to the tabulations in the Detailed DHC-A. As such, we assume a simplified, reduced-form data representation that is sufficient for our purposes. That is, we imagine deriving a single private dataframe from the 2020 CEF that consists of a row for each person in the United States with the following attributes: BlockID, RaceEth, Sex, Age.

BlockID is a single attribute that geolocates a person record to a unique Census block. As previously discussed, all geographic entities are aggregations of blocks. Thus, we assume that BlockID implicitly encodes each record's unique tract, county, and state. BlockID also encodes whether a record belongs to an AIANNH area and, if so, uniquely identifies the area. We note that all records are vacuously included in the Nation geographic entity.

RaceEth is a single attribute that encodes up to eight race codes and an ethnicity code. That is, one person's RaceEth attribute may indicate the person is Andorran and Dominican while another person's RaceEth attribute indicates Chinese, Japanese, Tongan, and Not Hispanic or Latino. We assume this RaceEth conceptualization combined with Census Bureau specifications fully determines the characteristic iterations of an individual.

Sex is recorded as Male or Female in the 2020 Census.

Age is recorded in single-year increments (e.g., 12, 30, 82).

3.2   The Algorithm Description

SafeTab-P must produce tabulations for population groups. As previously defined, a population group is a geographic entity (e.g. a specific county) and a characteristic iteration code (see Section 2 for more details). Population groups are split into sets called population group levels (specified by

a geography level and an iteration level) with distinct privacy-loss budgets. Records are associated with the population groups of a population group level via transformations that map their BlockID to geographic entities and their RaceEth attribute to characteristic iterations. In our simplified model, we assume there exists a master list of all population groups divided into population group levels. This master list may include *empty* population groups (combinations of geographic entities and characteristic iterations with no associated records in the private dataframe). We assume the following model for population groups:

- SafeTab-P produces tabulations for population group levels $\mathcal{P}_1, \ldots, \mathcal{P}_\omega$. That is, it should produce a tabulation for each population group $P \in \mathcal{P}_i$ for $1 \leq i \leq \omega$. For example, $\mathcal{P}_i$ may be the level (State, Detailed) consisting of population groups, such as (Iowa, Albanian Alone) and (Kansas, German Alone or in any Combination).

- SafeTab-P is provided privacy-loss budgets for each population group level $\rho_1, \ldots, \rho_\omega$ with ρ_i corresponding to the budget for population group level $\mathcal{P}_i$. Privacy-loss budgets are described in greater detail in Section 4.1. For now, we note that each ρ_i is a positive real-valued number.

- For each $\mathcal{P}_i$, we assume we have a function $g_i : \mathcal{I} \to 2^{\mathcal{P}_i}$, where $\mathcal{I}$ is the domain of records in the private dataframe. That is, g_i maps a record r to the subset of population groups at level i to which it belongs (i.e., $g_i(r) \subset \mathcal{P}_i$). For example, suppose i corresponds to the (State, Regional) level, the record r's BlockID uniquely identifies the state of residence as Idaho and its RaceEth attribute encodes Nigerian, Beninese, Tongan, and Not Hispanic or Latino. Then $g_i(r)$ would associate the record with the following population groups: (Idaho, Sub-Saharan African, Alone or in any Combination) and (Idaho, Polynesian Alone or in any Combination).

- We assume the stability of g_i, denoted by $\Delta(g_i)$, is known. The stability is defined as the maximum number of population groups a record could belong to in a level. Formally, $\Delta(g_i) = \max_{r \in \mathcal{I}} |g_i(r)|$ [11]. Importantly, this value defines what could be the maximum based on any hypothetically possible record, rather than defining what is the maximum based on the collected 2020 Census data. In other words, the stability is a data-independent value. As described earlier, this value is $\Delta(g_i) = 9$ for all population group levels tabulated in Detailed DHC-A.

The main algorithm is presented in Algorithm 1. This algorithm proceeds by looping over the population group levels. For each population group level, we apply g_i to the dataframe to map each record to the set of population groups it is associated with. Then, for each population group in the level, we call the tabulation function TABULATEPOPULATIONGROUP, passing in a dataframe containing just the records in that population group.

The pseudocode for the procedure TABULATEPOPULATIONGROUP is given in Algorithm 2. This code tabulates a single population group. Population groups are characterized based on the tabulation we would like to compute. In particular, we assume we are given a set TotalOnly of population groups for which only a T01001 total count of the population group should be tabulated. We check whether the given group is a member of this set. If it is, we call the NOISYCOUNT function on the population group, which tabulates a noisy total count for the group. Otherwise, we use a two stage algorithm. The computations in both stages require noise infusion to ensure the entire algorithmic process is covered by the differential privacy guarantee. We first compute a noisy count of the group using NOISYCOUNT but using only a fraction (denoted γ) of the available

Notation	Description
ω	The number of population group levels
$\mathcal{P}_i$	Population group level i
ρ_i	The privacy-loss budget allocated to population group level i
g_i	A function mapping records to the set of population groups in $\mathcal{P}_i$ to which the record belongs
$\Delta(g_i)$	$\max_{r \in \mathcal{I}} \lvert g_i(r) \rvert$

Table 5: A summary of the notation used in Section 3

Algorithm 1 The main SafeTab-P algorithm.

Input: df: A private dataframe with attributes [BlockID, RaceEth, Sex, Age] and one row for each person in the United States.

Input: $\{\rho_i\}_{i \in [1,\omega]}$: Privacy loss parameters for each population group level $i \in [1, \omega]$.

Input: γ: The fraction of the privacy-loss budget to be used in Stage 1 of the two stage tabulation algorithm.

1: **procedure** SAFETAB-P(df, $\{\rho_i\}$, γ)
2:     **for** $i \in [1, \omega]$ **do**
3:         $df_i \leftarrow df$.flatmap(g_i); $\triangleright$ df_i has schema [PopGroup, Sex, Age]
4:         $s \leftarrow \Delta(g_i)$ $\triangleright$ 1 row in df may result in $\leq s$ rows in df_i
5:         **for** $P \in \mathcal{P}_i$ **do**
6:             $df_P \leftarrow df_i$.filter(PopGroup $== P$)
7:             TABULATEPOPULATIONGROUP(df_p, P, ρ_i/s, γ)
8:         **end for**
9:     **end for**
10: **end procedure**

---

privacy-loss budget. Next, we compare this noisy count against a set of given thresholds, denoted Θ_1, Θ_2, and Θ_3. Depending on which thresholds the noisy count exceeds, we compute sex by age noisy counts with a varying degree of age bin sizes. Age bins are coarser for smaller noisy counts. These sex by age counts are also computed by NOISYCOUNT using the remaining privacy-loss budget.

The pseudocode for the procedure NOISYCOUNT is given in Algorithm 3. This procedure computes the number of rows in the dataframe and adds noise from a discrete Gaussian distribution.

The notation used in this section and the algorithm pseudocode is summarized in Table 5.

Algorithm 2 Subroutine of SafeTab-P to tabulate statistics for a single population group.

Input: df: A private dataframe with attributes [PopGroup, Sex, Age]. This dataframe should contain the records in the population group.
Input: P: The population group.
Input: ρ: Privacy-loss budget for this subroutine.
Input: γ: Fraction of ρ used in the adaptive algorithm

 1: **procedure** TABULATEPOPULATIONGROUP(df, P, ρ, γ)
 2:     **if** $P \in$ TotalOnly **then**
 3:         *// For TotalOnly population groups, only report noisy total counts*
 4:         **Output** NOISYCOUNT(df.count(), ρ)
 5:
 6:     **else**
 7:         *// For the rest of the population groups, adaptively choose the statistics released*
 8:         *// based on the noisy total count of the population group.*
 9:         *// Step 1: Compute the noisy total count using $\gamma\rho$ privacy-loss budget*
10:         total $\leftarrow$ NOISYCOUNT(df.count(), $\gamma\rho$)   ▷ Compute noisy total
11:
12:         *// Step 2: Release statistics based on the noisy count with $(1-\gamma)\rho$ privacy-loss budget*
13:         **if** total $< \Theta_1$ **then**
14:             **Output** NOISYCOUNT(df.count(), $(1-\gamma)\rho$)   ▷ Output the total
15:         **else if** total $< \Theta_2$ **then** df_group $\leftarrow$ df.map(Age $\rightarrow$ Age4).groupby(Sex, Age4)
16:             **Output** NOISYCOUNT(df_group.count(), $(1-\gamma)\rho$)   ▷ Sex X Age4 marginal
17:         **else if** total $< \Theta_3$ **then** df_group $\leftarrow$ df.map(Age $\rightarrow$ Age9).groupby(Sex, Age9)
18:             **Output** NOISYCOUNT(df_group.count(), $(1-\gamma)\rho$)   ▷ Sex X Age9 marginal
19:         **else** df_group $\leftarrow$ df.map(Age $\rightarrow$ Age23).groupby(Sex, Age23)
20:             **Output** NOISYCOUNT(df_group.count(), $(1-\gamma)\rho$)   ▷ Sex X Age23 marginal
21:         **end if**
22:     **end if**
23: **end procedure**

Algorithm 3 The discrete Gaussian mechanism for vectors.

Input: a: An n dimensional vector of integers.
Input: ρ: A privacy-loss parameter.
 1: **procedure** NOISYCOUNT(a, ρ)
 2:     $y \leftarrow \mathcal{N}_{\mathbb{Z}}^n \left(\frac{1}{2\rho} \right)$
 3:     **return** $a + y$
 4: **end procedure**

---

4 SafeTab-P Privacy Analysis

The algorithms employed by the Census Bureau's DAS to preserve the privacy of individual census respondents must adhere to the privacy standards imposed by zero-concentrated differential privacy (zCDP). In this section, we show that the SafeTab-P algorithm presented in Section 3 does satisfy the requirements of zCDP. Before presenting the proof, we first provide necessary background on zCDP.

4.1 Privacy Preliminaries

We provide the formal mathematical definition of zCDP and then describe relevant privacy properties that zCDP ensures.

4.1.1 Privacy definitions

Definition 1 (Neighboring Databases). Let x, x' be databases represented as multisets of tuples. We say that x and x' are *neighbors* if their symmetric difference is 1.

We define zCDP, which bounds the *Rényi divergence* between the distributions of a mechanism run on neighboring databases.

Definition 2. The *Rényi divergence of order* α between distribution P and distribution Q, denoted $D_\alpha(P\|Q)$, is defined as

$$D_\alpha(P\|Q) = \frac{1}{\alpha - 1} \log \left(\underset{x \sim P}{\mathbb{E}} \left[\left(\frac{P(x)}{Q(x)} \right)^{\alpha - 1} \right] \right) \tag{1}$$

Definition 3. (zCDP [7]) An algorithm $M : \mathcal{X} \to \mathcal{Y}$ satisfies ρ-zCDP if for all neighboring $x, x' \in \mathcal{X}$ and for all $\alpha \in (1, \infty)$,

$$D_\alpha(M(x)\|M(x')) \leq \alpha\rho. \tag{2}$$

The value ρ is often called the privacy-loss budget of an algorithm. Definition 3 is sometimes called "unbounded" ρ-zCDP. There is an alternative definition known as "bounded" ρ-zCDP. The difference between (unbounded) zCDP (Definition 3) and bounded zCDP (to be defined in Definition 5) is the definition of neighboring databases. Informally, neighboring databases under unbounded zCDP are obtained by adding/removing a record, while neighbors under bounded zCDP are obtained by changing one record[3]. This difference in definitions results in a difference in the semantics of the privacy guarantee. Briefly, unbounded zCDP protects the presence of any record in the database, while bounded zCDP protects the value of each record in the database. One notable difference is that bounded zCDP does not protect the total number of records in the database.

The formal definition of bounded neighbors is the following:

Definition 4 (Bounded Neighboring Databases). Let x, x' be databases represented as multisets of tuples. We say that x and x' are *bounded neighbors* if they contain the same number of tuples and are identical except for a single tuple.

Using this new definition of neighbors, we can give the definition of bounded zCDP.

---

[3]The word "bounded" comes from the fact that all neighboring databases are the same size.

Definition 5 (Bounded zCDP). An algorithm $M : \mathcal{X} \to \mathcal{Y}$ satisfies bounded ρ-zCDP if for all bounded neighboring databases $x, x' \in \mathcal{X}$, for all output $y \in \mathcal{Y}$, and for all $\alpha \in (1, \infty)$,

$$D_\alpha(M(x)\|M(x')) \leq \alpha\rho. \tag{3}$$

The Tumult-developed libraries (see Section 5.2) used to implement SafeTab-P provide unbounded zCDP guarantees. We provide an additional proof for bounded zCDP to allow for consistent comparison across other 2020 Census data releases such as the DHC that utilize bounded zCDP. A common neighbors definition allows for composition analysis across different releases rather than treating each release in isolation. The Census Bureau interest in bounded neighbors stems from a desire to model attackers that know the true database size given that several 2020 Census data products release the exact total U.S. population (according to the CEF) without noise infusion.

Unless otherwise specified, we assume unbounded ρ-zCDP, by default, in this paper.

4.1.2 Composition Theorems

One of the most useful and important properties of privacy definitions is their behavior under composition. In this section, we state composition results for zCDP.

There are two types of composition we are interested in – sequential composition and parallel composition. We first state the sequential composition results.

Lemma 1. *(Adaptive sequential composition of zCDP [7]) Let $M_1 : \mathcal{X} \to \mathcal{Y}$ and $M_2 : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be mechanisms satisfying ρ_1-zCDP and ρ_2-zCDP respectively. Let $M_3(x) = M_2(x, M_1(x))$. Then M_3 satisfies $(\rho_1 + \rho_2)$-zCDP.*

Next, we state and prove a generalized parallel composition lemma for our privacy definition. To the best of our knowledge, these results are novel. The standard statement of parallel composition is a special case of our generalization.

Let the *maximum degree* of a set family $F = \{S_i\}$, $S_i \subseteq S$ be the maximum number of sets containing any fixed element of S. That is,

$$degree(F) = max_{s \in S}|\{S_i \in F | s \in S_i\}| \tag{4}$$

Lemma 2. *Let $F = \{S_1, ..., S_k\}$ be a family of subsets of the input domain with maximum degree z. Let $M_1, \ldots, M_k$ each provide ρ-zCDP. Then the mechanism $M(x) = (M_1(x \cap S_1), \ldots, M_k(x \cap S_k))$ provides $(z \cdot \rho)$-zCDP.*

The proof of Lemma 2 requires the following property on the Rényi divergence, given in Lemma 2.2 of [7].

Lemma 3. *(Lemma 2.2 of [7])*

Let P and Q be distributions on $\Omega \times \Theta$. Let P_Ω and Q_Ω denote the marginal distributions on Ω. Likewise, let P_Θ and Q_Θ denote the marginal distributions on Θ. For $x \in \Omega$, let P_Θ^x and Q_Θ^x denote the conditional distributions on Θ conditioned on the first coordinate. Then

$$D_\alpha(P\|Q) \leq D_\alpha(P_\Omega\|Q_\Omega) + \max_{x \in \Omega} D_\alpha(P_\Theta^x\|Q_\Theta^x) \tag{5}$$

When P and Q are product distributions, then this becomes the following.

$$D_\alpha(P\|Q) = D_\alpha(P_\Omega\|Q_\Omega) + D_\alpha(P_\Theta\|Q_\Theta) \tag{6}$$

With this, we can prove Lemma 2.

Proof of Lemma 2. Suppose x and x' are neighbors and let r be the (only) record in their symmetric difference. Let $i_1, \ldots, i_j$ be the indices of the sets in F containing r. $j \leq z$ since the maximum degree of F is z.

$$D_\alpha(M(x)\|M(x')) = \sum_{i=1}^{k} D_\alpha(M_i(x \cap S_i)\|M_i(x' \cap S_i)) \tag{7}$$

$$= \sum_{i \in \{i_1, \ldots, i_j\}} D_\alpha(M_i(x \cap S_i)\|M_i(x' \cap S_i)) \tag{8}$$

$$\leq \sum_{i \in \{i_1, \ldots, i_j\}} \alpha \cdot \rho \tag{9}$$

$$\leq \alpha \cdot (z \cdot \rho). \tag{10}$$

$\square$

4.1.3 Postprocessing

Zero-concentrated differential privacy is closed under postprocessing, meaning that the privacy guarantee cannot be weakened by manipulating the outputs of a zCDP mechanism without reference to the protected inputs.

Lemma 4. *(Postprocessing for zCDP [7]) Let $M : \mathcal{X} \to \mathcal{Y}$ and $f : \mathcal{Y} \to \mathcal{Z}$ be randomized algorithms. Suppose M satisfies ρ-zCDP. Then $f \circ M : \mathcal{X} \to \mathcal{Z}$ satisfies ρ-zCDP.*

4.1.4 Base Mechanism

Definition 6 (L2 Sensitivity). Given a vector function $q : \mathcal{X} \to \mathbb{Z}^n$, the sensitivity of q is $\sup_{x,x'} \|q(x) - q(x')\|_2$ where x and x' are neighboring databases and $\|\cdot\|_2$ is the Euclidean norm.

There is an equivalent notion of bounded sensitivity.

Definition 7 (Bounded L2 Sensitivity). Given a vector function $q : \mathcal{X} \to \mathbb{Z}^n$, the sensitivity of q is $\sup_{x,x'} \|q(x) - q(x')\|_2$ where x and x' are bounded-neighboring databases and $\|\cdot\|_2$ is the Euclidean norm.

Definition 8. The discrete Gaussian distribution $\mathcal{N}_\mathbb{Z}(\sigma^2)$ centered at 0 is

$$\forall x \in \mathbb{Z}, \quad \Pr[X = x] = \frac{e^{-x^2/2\sigma^2}}{\sum_{y \in \mathbb{Z}} e^{-y^2/2\sigma^2}}. \tag{11}$$

Lemma 5. *[8] Let $q : \mathcal{X} \to \mathbb{Z}^n$ with L2 sensitivity Δ. Then outputting $\text{NOISYCOUNT}(q(x), \rho)$ from Algorithm 3 satisfies $\Delta^2 \rho$-zCDP.*

Lemma 6. *[8] Let $q : \mathcal{X} \to \mathbb{Z}^n$ with bounded L2 sensitivity Δ. Then outputting $\text{NOISYCOUNT}(q(x), \rho)$ from Algorithm 3 satisfies bounded $\Delta^2 \rho$-zCDP.*

4.2 Privacy Analysis

With the provided background, we are ready to prove that SafeTab-P satisfies zCDP.

Theorem 1. *Let $\rho_{total} = \sum_{i=1}^{\omega} \rho_i$. Algorithm 1 satisfies (unbounded) ρ_{total}-zCDP.*

Proof. The proof of Theorem 1 follows via the combination of composition rules along with the fact that the NOISYCOUNT procedure satisfies zCDP, per Lemma 5, with respect to its inputs. In other words, we can think of NOISYCOUNT as a basic building block from which we construct SafeTab-P.

First, we construct the TABULATEPOPULATIONGROUP procedure. That is, we claim that the procedure TABULATEPOPULATIONGROUP in Algorithm 2 satisfies ρ-zCDP with respect to the input dataframe, where ρ is the privacy parameter input to the procedure. Note that TABULATEPOPULATIONGROUP actually uses one of two algorithms, depending on whether the population group is in the set TotalOnly. We consider each of these algorithms.

Case 1: $P \in$ TotalOnly. In this case, the procedure simply calls NOISYCOUNT, which has L2 sensitivity 1, and satisfies ρ-zCDP.

Case 2: $P \notin$ TotalOnly. In this case, the procedure can be decomposed into two parts. First, we call NOISYCOUNT, which has L2 sensitivity 1, with a budget of $\gamma\rho$. Then, we use the result to group the data by sex and age, we make a call to NOISYCOUNT, over the vector of all the sex by age categories, with a budget of $(1-\gamma)\rho$. The composition of the calls on all the sex by age groups satisfies $(1-\gamma)\rho$ by Lemma 2. The (adaptive) composition of the two parts has total privacy loss ρ by Lemma 1.

Next, we claim that the ith loop of the **for** loop on line 2 of Algorithm 1 satisfies ρ_i-zCDP. By the definition of s, any particular record can appear in the input (df_P) of at most s calls to TABULATEPOPULATIONGROUP. Therefore, by Lemma 2, the total privacy loss of the loop is s times the privacy loss of TABULATEPOPULATIONGROUP, i.e. $s \cdot \frac{\rho_i}{s} = \rho_i$.

Finally, the overall algorithm satisfies $(\sum_{i=1}^{\omega} \rho_i)$-zCDP by Lemma 1. $\qquad\square$

Theorem 2. *Let $\rho_{total} = \sum_{i=1}^{\omega} \rho_i$. Algorithm 1 satisfies bounded $2\rho_{total}$-zCDP.*

Proof. The proof of Theorem 2 follows by tracking privacy loss across the joint mechanisms along with the fact that the NOISYCOUNT procedure satisfies bounded zCDP, per Lemma 6, with respect to its inputs. In other words, we can think of NOISYCOUNT as a basic building block from which we construct SafeTab-P. Under bounded zCDP, it is sufficient to bound the changes due to the removal of one record *and* addition of another. As a result, we reason specifically about the effects of adding one record and removing another record.

First, we construct the TABULATEPOPULATIONGROUP procedure. That is, we claim that the procedure TABULATEPOPULATIONGROUP in Algorithm 2 satisfies bounded 2ρ-zCDP with respect to the input dataframe, where ρ is the privacy parameter input to the procedure.

Note that TABULATEPOPULATIONGROUP actually uses one of two algorithms, depending on whether the population group is in the set TotalOnly. Likewise, we also need to consider if both the added and deleted record belong to the same population group or different population groups. For each of these cases, we bound the Rényi divergence of the mechanism's output distribution for bounded neighboring databases and demonstrate under which conditions it is maximized.

Case 1: Assume $P \in$ TotalOnly and both records are in P. In this case, the procedure simply calls NOISYCOUNT. Since a record is both added and removed from this count, the distribution of this count does not change and as such the Rényi divergence is 0.

Case 2: Assume $P \in$ TotalOnly and either the added record or removed record is in P but not both. In this case, the procedure simply calls NOISYCOUNT. This count changes by 1 and, by Lemma 6, has a maximum Rényi divergence of $\alpha\rho$.

Case 3: Assume $P \notin$ TotalOnly and both records are in P. In this case, the procedure can be decomposed into two parts. First, we call NOISYCOUNT with a budget of $\gamma\rho$. Then, we use the result to do a GroupBy on the data by sex and age, and we make a call to NOISYCOUNT with a budget of $(1 - \gamma)\rho$ to compute the counts for all sex by age categories. The first output of NOISYCOUNT has a Rényi divergence of 0 since the count does not change when both records are in the same population group. By Lemma 3, the Rényi divergence of the combined outputs of NOISYCOUNT is bounded by the Rényi divergence of the first output of NOISYCOUNT plus the Rényi divergence of the second output of NOISYCOUNT when conditioned on the result of the first call that maximizes the total Rényi divergence. This bound holds for any of the sex by age marginals. In this case, one category increases by 1 and the other decreases by 1. Therefore, the outputs of NOISYCOUNT have a Rényi divergence of $\alpha 2(1 - \gamma)\rho$ by Lemma 6. Then, by Lemma 3, the (adaptive) composition of the two parts has a total divergence of $\alpha 2(1 - \gamma)\rho$.

Case 4: Assume $P \notin$ TotalOnly and either the added record or removed record is in P but not both. In this case, the procedure can be decomposed into two parts. First, we call NOISYCOUNT with a budget of $\gamma\rho$. Then, we use the result to do a GroupBy on the data by sex and age, and make a call to NOISYCOUNT, over the vector of all sex by age categories, with a budget of $(1 - \gamma)\rho$. The first output of NOISYCOUNT has a Rényi divergence of $\alpha\gamma\rho$ by Lemma 6 since the count changes by 1 when only one of the records is in the population group. By Lemma 3, the Rényi divergence of the combined outputs of NOISYCOUNT is bounded by the Rényi divergence of the first output of NOISYCOUNT plus the Rényi divergence of the second output of NOISYCOUNT when conditioned on result of the first call that maximizes the total Rényi divergence. In this case, one sex by age category either increases by 1 or decreases by 1. Therefore, the outputs of NOISYCOUNT have a Rényi divergence of $\alpha(1 - \gamma)\rho$ by Lemma 6. Then, by Lemma 3, the (adaptive) composition of the two parts has total Rényi divergence of $\alpha\rho$.

Next, we claim that the ith loop of the **for** loop on line 2 of Algorithm 1 satisfies $2\rho_i$-zCDP. By the definition of s, any particular record can appear in the input (df_P) of at most s calls to TABULATEPOPULATIONGROUP. For each of the s calls for the added record, the removed record can either overlap in their population group or be part of a different population group. When they overlap (Cases 1,3), the maximum Rényi divergence is of $\alpha 2(1 - \gamma)\rho_i/s$. This occurs when the population group is not in TotalOnly (Case 3). When they do not overlap (Cases 2,4), the Rényi divergence is the same for population groups in TotalOnly and not in TotalOnly. For these, each output of TABULATEPOPULATIONGROUP has a Rényi divergence of $\alpha\rho_i/s$. However, since both the added and removed records are part of different population groups, the output of TABULATEPOPULATIONGROUP for each of those population groups has a Rényi divergence of $\alpha\rho_i/s$ resulting in a combined Rényi divergence of $\alpha 2\rho_i/s$ by Lemma 3. Since $\alpha 2\rho_i/s > \alpha 2(1 - \gamma)\rho_i/s$, the maximum Rényi divergence occurs when none of the population group for either the added or removed record overlap. Since this can happen at most s times the total Rényi divergence is $s(\alpha 2\rho_i/s) = \alpha 2\rho_i$ by Lemma 3. This is the case that maximizes the Rényi divergence over the entire **for** loop and, therefore it satisfies bounded $2\rho_i$-zCDP.

Finally, the overall algorithm satisfies $(\sum_{i=1}^{\omega} 2\rho_i)$-zCDP by Lemma 1.

$\square$

5 Implementation of SafeTab-P

The algorithm presented in Section 3 is a simplified version of the implemented SafeTab-P program. In this section, we describe some of the differences between the implementation and the simplified algorithm. We focus on differences that could affect the privacy calculus and describe why the implementation is equivalent to the simplified algorithm.

5.1 Input Validation

Input validation is an important step before deploying a differentially private algorithm. SafeTab-P performs extensive validation of its input data to ensure that provided data are in the expected formats and internally consistent. Some validation occurs on data that are considered public (like the list of all geographic entities for which data is to be tabulated) and therefore, do not affect the privacy guarantees. However, we also validate the private input files that contain the data of individual census respondents.

Validation failures are only visible to the trusted curator running the program, and on failure, no part of the differentially private program is run. Validation failures are made available to the trusted curator, so they can correct any errors in the provided input files before executing the differentially private program. Failures in validation are not released publicly and therefore, do not contribute to any privacy loss.

5.2 Tumult Analytics

Rather than directly calculating stability and sampling from noise distributions, SafeTab-P is implemented using Tumult Analytics[5], a framework for implementing differentially private queries.

A key benefit of using Tumult Analytics is that all access to the sensitive data is mediated through a Tumult Analytics `session`. The `session` tracks all the transformations and measurements performed on the sensitive data and is able to correctly compute the total privacy loss of the computation on the sensitive data. In SafeTab-P, we construct an Analytics `session` with:

- The total privacy-loss budget for the pipeline (calculated as the sum of all population-group budgets ρ_i).

- The private dataset.

- The public datasets (information on all race and ethnicity codes, characteristic iterations, and geographic entities).

- The neighboring definition (privacy is with respect to the addition/removal of one record from the private dataset).

- The privacy definition to be satisfied (e.g., zCDP).

We then implement all data transformations (like mapping a record to its characteristic iterations) and queries within the framework. Tumult Analytics tracks the stability throughout the transformations and applies an appropriate amount of noise to the final queries, guaranteeing that the outputs are differentially private and no more than the total budget is expended.

The use of Tumult Analytics means that, while the simplified algorithm above only describes ρ-zCDP, SafeTab-P also supports "pure" ϵ-Differential Privacy with noise drawn from a double-sided geometric distribution. The user-provided privacy-loss budget values are interpreted as

either ϵ's or ρ's depending on which privacy definition the user selects in SafeTab-P's configuration. The production of Detailed DHC-A utilizes zCDP. Thus, the presentation of the algorithm in this document focus on zCDP.

The use of Tumult Analytics also allows (and necessitates) some other deviations from the simplified algorithm. Rather than counting each population group sequentially, in a for-loop, we use Analytics' `groupby` feature to tabulate many population groups at once. Analytics requires users of the `groupby` feature to specify all groups to tabulate in advance in the form of a `KeySet` object. In the case of SafeTab-P, we construct `KeySet`s containing the combinations of possible geographic areas, characteristic iterations, and (if applicable) sex values and age buckets. We build these `KeySet`s using separate input specification files (rather than relying on observed groups present in the CEF). `KeySet`s do not use the confidential CEF data to ensure that noisy statistics are produced for every valid population group. Thus, we do not reveal whether population groups are empty via their presence or absence in the output data. We note the simplified algorithm also allowed for the possibility that df_P (filtered dataframe that only contain records associated with population group P) is empty.

A key innovation in SafeTab-P is that statistics are released adaptively based on the total count of each population group. This is a challenging feature to implement in a manner that provably ensures differential privacy – the privacy accounting across population groups that only output a Total count versus groups that output sex by age counts does not follow from adding up the privacy losses for each group (as in simple composition) but follows from a more nuanced generalized parallel composition (see Lemma 2). To ensure that we receive this tighter accounting, we use the Tumult Analytics `partition_and_create` operator to make the structure of the computation legible to the automatic privacy accountant.

5.3 Postprocessing for Addressing Demographic Reasonableness Concerns

After the differentially private algorithm has completed, we perform several additional postprocessing steps. Because these steps are purely postprocessing, they cannot affect the differential privacy guarantees per Lemma 4. These postprocessing steps are designed to address specific data quality consistency concerns that arise when adding noise to tabular statistics. All postprocessing was implemented under the direction of subject-matter experts at the Census Bureau. These steps do not exhaustively address all possible demographic reasonableness concerns.

5.3.1 Marginals

A population group that is not in the $TotalOnly$ set can receive either a total count or a sex by age breakdown of varying granularity. For those that receive a sex by age breakdown, we calculate internally-consistent sex marginals (counts broken down by sex, but not age) and a total by summing the relevant noisy counts. This approach provides consistency within a sex by age table for a given population group but does nothing to provide consistency across different population groups.

5.3.2 Suppression

One demographic reasonableness concern associated with generating noisy counts is that of creating positive population group counts for groups that do not exist in the true data. We suppress population groups that received small noisy counts to combat this concern. The goal of the suppression is to reduce the probability that a true zero count is released as a positive noisy count.

Beyond accomplishing that goal, suppression has two other consequences:

- Groups with a true count larger than zero may be suppressed, including true counts larger than the suppression threshold.

- Suppression introduces a positive bias in released counts, especially for groups whose true count is at or below the suppression threshold. The bias varies depending on the magnitude of the true count relative to the suppression threshold with smaller counts exhibiting greater amounts of bias, as shown in Figure 2.

When analyzing suppression, we consider only the ρ-zCDP version of SafeTab-P as described in Section 3 (which uses discrete Gaussian noise).

Suppression algorithm description.

- Run SafeTab-P to produce noisy counts.

- For every sub-state population group receiving only a total count, filter out those population groups whose noisy count is less than the suppression threshold T (we consider a single threshold here for simplicity, but there may be different thresholds for different classes of population groups such as separate thresholds for detailed and regional population groups).

- Release the remaining noisy counts.

We only perform suppression on sub-state population groups that do not receive a sex by age breakdown since eligibility for sex by age requires passing a population threshold in the main SafeTab-P algorithm that is presumably higher than the postprocessing suppression threshold. Hence, population groups that do receive sex by age breakdowns are unlikely to have a true count of zero.

Computing the probability of zero count suppression. Rather than directly choosing a suppression threshold, T, we allow the SafeTab-P user to select a probability that a zero count is suppressed, p. We then derive a threshold T from p. In the following, we let $invcdf(\sigma, x)$ denote the inverse cumulative distribution function (CDF) of the discrete Gaussian distribution [8] with mean 0 and scale σ. As alluded to earlier, suppression is only applied at sub-state geography levels and thus, thresholds are only required for population groups that undergo the two stage adaptive process. The thresholds are set as

$$T = invcdf\left(\sqrt{\frac{9}{2(1-\gamma)\rho}}, p\right) \tag{12}$$

where ρ is the privacy-loss budget allocated to the population group level and γ is the fraction of the budget reserved for the adaptive step and 9 is the stability value $\Delta(g_i)$ regardless of the index i. In Table 6, we give threshold values for various values of ρ for a 99.99% target zero count suppression rate.

Computing the probability of nonzero count suppression. One side effect of suppressing true zeros (i.e., groups whose true count is zero) with high probability is that small population groups will also be suppressed. In Figure 1, we plot the probability that a population group will be suppressed, given that true zeros are suppressed with 99.99% probability. We plot the true count

Privacy loss (ρ)	Threshold (T)
0.008	93
0.159	21
0.543	11

Table 6: Suppression thresholds for various values of ρ to give a 99.99% chance that a true zero count will be suppressed. We set $\gamma = 0.1$ in all cases.

of the population group as a fraction of the suppression threshold T and therefore, this plot gives the correct probabilities regardless of the privacy loss (as long as T represents the threshold for which zero counts are suppressed with probability 99.99%.)
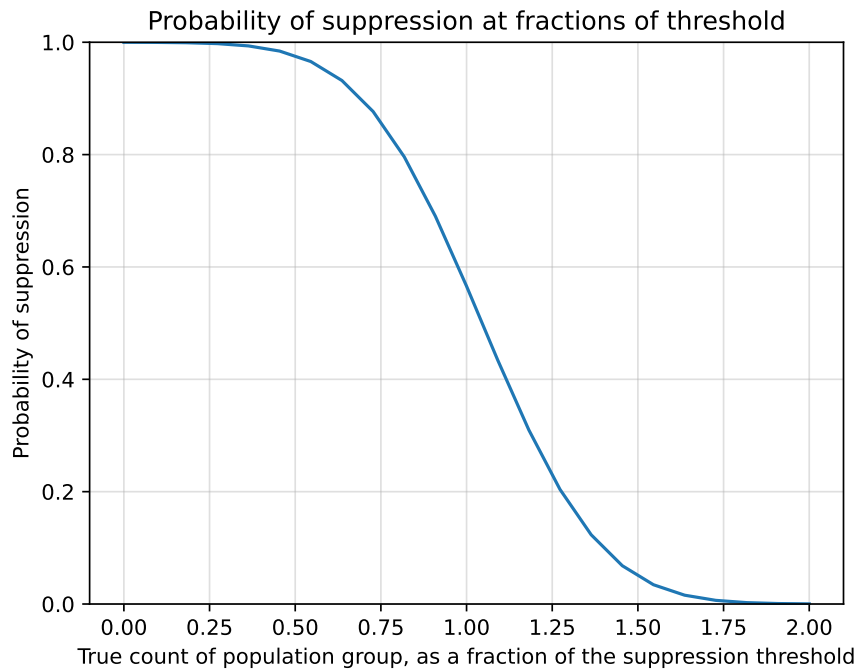

Figure 1: The probability of suppression as a function of the true size of the population group, when true zeros are suppressed with probability 99.99%. The true size of the population group is expressed as a fraction of the suppression threshold. For example, the "0.5" indicates a population group with true size $T/2$, where T is the suppression threshold.

Bias in released counts. Another side effect of suppression is that released counts have a positive bias. The cause is easy to understand for population groups whose true count is below the suppression threshold: noisy estimates for these counts will only be released if the algorithm adds sufficient positive noise. However, noisy estimates for population groups whose true count is larger than the suppression threshold also have a positive bias. These groups are suppressed if we add sufficient negative noise.

We can calculate the bias by calculating the expected noise added to a population group, given the population group is published. If we let n be the true size of the population group, we can use the following formula to calculate the expected noise:

$$\mathbb{E}[X|n + X > T] = \frac{\sigma \cdot \phi(\frac{T-n}{\sigma})}{1 - \Phi(\frac{T-n}{\sigma})}, \tag{13}$$

where $\phi(\cdot)$ is the probability density function of the standard discrete Gaussian distribution, $\Phi(\cdot)$ is the cumulative distribution function of the standard discrete Gaussian distribution, X is a random variable representing the amount of noise added to the population group count, and σ is the scale of the noise.

In Figure 2, we plot the expected noise added to a population group count, given the population group is released. As in the last section, we give the true size of the population group and the bias as fractions of the suppression threshold. This means the results hold regardless of the scale of the noise, as long as the suppression threshold, T, represents the threshold for which zero counts are suppressed with probability 99.99%.
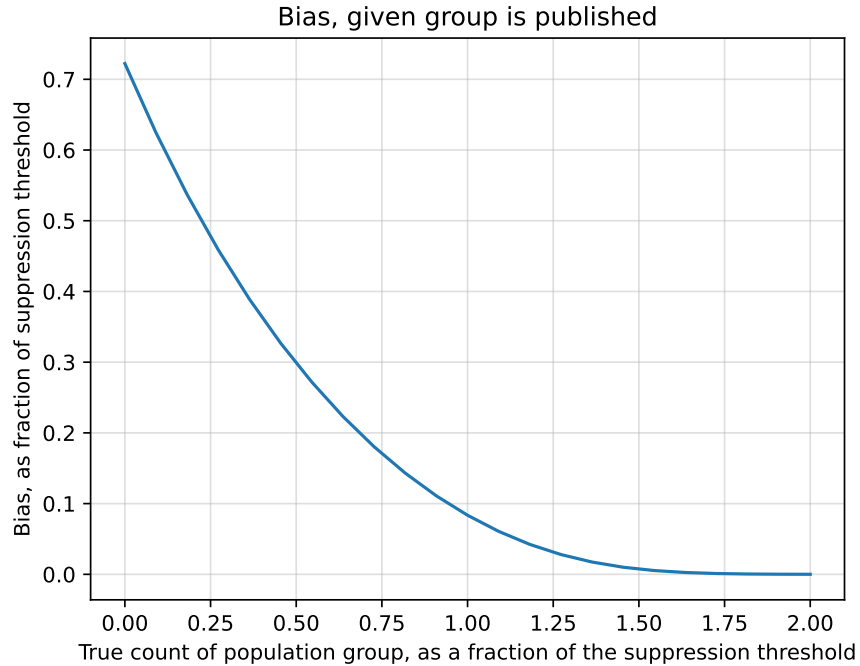

Bias, given group is published

Figure 2: The bias in reported population group size compared to the true size of the population group, when true zeros are suppressed with probability 99.99%. Both the size of the population group and the bias are expressed as fractions of the suppression threshold. For example, the "0.5" on the x-axis indicates a population group with size $T/2$, where T is the suppression threshold and "0.5" on the y-axis indicates a bias of $T/2$.

5.3.3 Coterminous Geographies

Sometimes, two or more geographic entities in different geographic summary levels share the same geographic boundaries (i.e., they are aggregated from identical collections of Census blocks). For example, Washington D.C. is tabulated as a state, county, and place. These geographic entities are called *coterminous*. Another coterminous example is a county containing a single Census tract. A characteristic iteration receives different independent noisy measurements for each geographic summary level of a given coterminous area. However, its counts should be identical at each summary level. It is also possible that a characteristic iteration is suppressed at one summary level of

---
Algorithm 4 The coterminous geographies postprocessing algorithm.

Input: df: The output of the main SafeTab-P algorithm
Input: $\mathcal{C}$: A list of all sets of coterminous population groups.
Input: R: An ordering of geographic summary levels.
 1: **procedure** COTERMINOUS GEOGRAPHIES(df, $\mathcal{C}$, R)
 2:     **for** $C \in \mathcal{C}$ **do**
 3:         **for** $r \in R$ **do**
 4:             $P \leftarrow$ The population group in C whose geographic entity is at summary level r.
 5:             **if** df contains data for P **then**
 6:                 Let M be the set of counts in df for P
 7:                 **for** $P' \in C - \{P\}$ **do**
 8:                     Remove all counts for P' from df
 9:                     Add M to df for P'.
10:                 **end for**
11:             **end if**
12:         **end for**
13:     **end for**
14: **end procedure**
---

a coterminous area but not suppressed in the other summary levels. Some geographic entities at different summary levels that do not share the same geographic boundaries should still be statistically equivalent. For example, if a county contains one water-only tract and one nonwater tract, characteristic iterations should have the same counts in the nonwater tract as in the county. We abuse terminology by calling statistically equivalent geographic areas coterminous. Working in consultation with subject-matter experts at the Census Bureau, we created a postprocessing step (implemented as a standalone program) that corrects inconsistencies in coterminous geographic entities. We assume as input a hierarchy over geographic summary levels. For each set of coterminous population groups (e.g., {(D.C. (as a State), Maori Alone), (D.C. (as a County), Maori Alone), (D.C. (as a Place), Maori Alone)}), we find the population group with the hierarchically highest geographic summary level that has not been suppressed, called the *donor* population group. We then overwrite the data of the other population groups in the set with the data of the donor population group. A pseudocode description of this procedure is presented in Algorithm 4.

5.3.4 Tabulation System Suppression

Additional demographic reasonableness corrections are addressed outside the DAS. In particular, the Decennial Tabulation System also performs suppression postprocessing. Again, per Lemma 4, this does not impact the privacy analysis. The suppression conducted by non-DAS systems is out of scope for this paper but includes further enforcement of nonnegativity as well as suppression in cases where noisy counts for Alone characteristic iterations appear to be greater than the corresponding noisy counts for Alone or in Any Combination characteristic iterations.

5.4 Other Implementation Details

We note a few other implementation differences that are primarily driven by the specification requirements of the system and data wrangling aspects of the code. These include the function

mapping race/ethnicity codes to characteristic iterations, rules for what statistics are tabulated for different population groups, and the handling of Puerto Rico.

5.4.1 Mapping Race/Ethnicity Codes to Characteristic Iterations

The pseudocode in Section 3 abstracts the process of mapping a person's input record into their corresponding population group as a function g_i. In practice, this process requires joining against several specification input files and some subtle logic (to determine whether a user qualifies for an Alone characteristic iteration in addition to an Alone or in Any Combination characteristic iteration). However, the end result is functionally equivalent to the g_i abstraction - each record is mapped to a number of geographic entities and characteristic iterations. The stability factor of the implemented transformations, the equivalent of $\Delta(g_i)$, is automatically tracked by Analytics rather than being computed by hand.

The pseudocode also ignores the logic associated with pre-processing a specified universe of geographic entities and iteration codes into population group levels $\mathcal{P}_i$. That is, the master list of all population groups divided into population group levels is constructed through a combination of specification files rather than being handed directly to the system.

5.4.2 Puerto Rico

The SafeTab-P algorithm presented in Algorithm 1 describes an input dataframe consisting of records of every person in the United States. However, the same algorithm is applied to a dataframe consisting of records from Puerto Rico. In implementation, SafeTab-P tabulates data for the United States and Puerto Rico in two separate passes.

6 Parameters and Tuning

Between the pseudocode representation of SafeTab-P and the selected implementation details presented earlier, we have alluded to a number of parameters that must be set before executing a run of the SafeTab-P program. Parameters are adjustable factors that must be fixed to fully define the nature of the program (e.g., the noise distribution employed in NOISYCOUNT, the privacy-loss budgets for population group levels, and population thresholds). With regard to SafeTab-P specifically (but any differentially private algorithm generally), parameter selection is a matter of policy. The Census Bureau's Data Stewardship Executive Policy (DSEP) committee, in consultation with subject-matter experts as well as internal and external privacy experts, approved all available parameters for the Detailed DHC-A. Parameter selection necessitates trade-offs, as many of these parameters are dependent on each other. To illustrate, we consider a fundamental relationship between the privacy loss parameters and their corresponding margins of error with discrete Gaussian noise distributions.

6.1 Error bounds

We derive error bounds for Algorithm 1 with discrete Gaussian noise. We begin by stating a portion of Proposition 25 from [8].

Proposition 1 (Proposition 25 in [8]). *For all $m \in \mathbb{Z}$ with $m \geq 1$ and for all $\sigma \in \mathbb{R}$ with $\sigma > 0$,* $\Pr[X \geq m]_{X \leftarrow \mathcal{N}_{\mathbb{Z}}(\sigma^2)} \leq \Pr[X \geq m-1]_{X \leftarrow \mathcal{N}(\sigma^2)}.$

The following corollary is immediate.

Corollary 1. *For all $x, \sigma \in \mathbb{R}$ with $x \geq 1$ and $\sigma > 0$, $\Pr[X > x]_{X \leftarrow \mathcal{N}_{\mathbb{Z}}(\sigma^2)} \leq \Pr[X > \lfloor x \rfloor]_{X \leftarrow \mathcal{N}(\sigma^2)}$.*

Figure 2 of [8] provides an intuitive visualization of these tail bounds. It follows that $X \in [-\lfloor 1.96\sigma \rfloor, \lfloor 1.96\sigma \rfloor]$ with probability at least 95%. That is, the 95% margin of error (MOE), defined as half the width of the 95% confidence interval, is given by $\lfloor 1.96\sigma \rfloor$.

Hence, for a population group in level i in the TotalOnly set, the MOE in the directly computed total estimate from line 4 in Algorithm 2 is $\left\lfloor 1.96\sqrt{\frac{s}{2\rho_i}} \right\rfloor$ ($s = \Delta(g_i) = 9$ and ρ_i is the privacy-loss budget for level i). For the population groups in level i not in the TotalOnly set, the MOE in a single sex by age group in Algorithm 2 is $\left\lfloor 1.96\sqrt{\frac{s}{2(1-\gamma)\rho_i}} \right\rfloor$ ($s = \Delta(g_i) = 9$, ρ_i is the privacy-loss budget for level i, and γ is the fraction of the budget used in Step 1 of the adaptive process).

Corollary 2. *The NoisyCount procedure implemented with discrete Gaussian noise and run with $\rho = \frac{1.92}{\lfloor MOE \rfloor^2}$ has a 95% MOE of at most MOE.*

This relationship dictates that adjusting a desired MOE to achieve improved data utility comes at the cost of additional privacy loss. Likewise, adjusting privacy-loss parameters to provide better privacy protection comes at the cost of wider MOE in the estimated statistics.

6.2 Parameter Identification, Trade-offs, and Outcomes

Fully identifying the set of possible parameters, let alone navigating the trade-offs associated with them, was no small feat. Before delving into specific parameters, we will overview some of the methods involved in navigating associated trade-offs. Firstly, parameters tend to impact some combination of these three aspects: data accessibility, data accuracy, and privacy. Data accessibility, in a nutshell, refers to the volume of tabular statistics released. Data accuracy is primarily measured by margins of error on the tabular statistics. Privacy is measured by the privacy-loss parameters of the algorithm. For example, excluding population group level i would reduce data accessibility but improve privacy since the privacy loss ρ_i is no longer necessary. To aid in the understanding of these trade-offs, we relied on a combination of tangible tools and theoretical analyses (such as the suppression analysis demonstrated in Section 5.3.2).

Before development on SafeTab-P began, we created the SafeTEx tool to provide hands-on experience with some of the trade-offs. We will provide a brief summary of this tool. Then, we will highlight specific parameters and the critical decisions made by the DSEP committee as a consequence of either interacting with the tool, reviewing theoretical analysis, or subject-matter expert guidance.

6.3 Parameter Tuning using the SafeTEx Tool

SafeTEx is an easy-to-use interactive decision support tool implemented in the Microsoft Excel program and developed to facilitate conversations between subject-matter experts, disclosure avoidance scientists, and ultimately, the DSEP committee.

The tool allowed users to interactively specify:

- The set of geography levels and characteristic iteration levels that constitute the universe of population groups for which statistics are tabulated.

- The maximum number of races a person is associated with to an integer in the range 1-8.

- A target error threshold.

- A privacy-loss budget ρ and the fraction of the privacy-loss budget reserved for the Step 1 of the adaptive procedure γ.

Based on these parameters, the SafeTEx tools computed thresholds on the size of population groups at which different statistics levels (total, sex by age(4), sex by age(9), sex by age(23) can be released at the target error. The computations were performed using analytical formulae for expected error of noise mechanisms employed in the SafeTab-P algorithm.

We describe in the next section some of the key parameters considered and the decision process used to set these parameters.

6.3.1 Parameter Selection

Population group levels, TotalOnly population groups, and population thresholds: As a reminder, a population group level is defined by a geographic summary level, such as Nation, State, and County, and an iteration level (i.e., Detailed or Regional). The SafeTEx tool helped subject-matter experts grasp the impact of adding or removing levels would have on the privacy-loss budget, but they had to weigh that against the value of having publishable statistics at each given level. They also gathered feedback from the user community on these topics and ultimately settled on the levels referenced in Table 7. Furthermore, SafeTEx helped subject-matter experts reason concerning the relation between absolute expect error and population sizes. These relative error comparisons were instrumental in determining population thresholds for the adaptive algorithm. We leave the research details of these decision-making efforts as a topic for a future paper. The subject-matter experts also made a determination as to which groups should be pre-defined as TotalOnly. The SafeTEx tooling did not directly address this concept.

Noise distribution: As mentioned in the implementation details, SafeTab-P can either be instantiated with discrete Gaussian noise or geometric noise. Other noise distributions were not considered because of a desire to preserve integrality in the outputs without needing to postprocess the results to achieve it. We analytically compared the privacy and accuracy of two noise distributions and observed that the discrete Gaussian mechanism displayed roughly 7% less privacy loss, compared to the geometric mechanism at the same accuracy targets. We analyzed both mechanisms under another alternative to pure differential privacy known as approximate differential privacy to ensure consistency in the assessment of privacy. Details of this analysis can be found in [9]. As an outcome of this analysis, the Census Bureau decided to use discrete Gaussian noise. Hence, the focus of this paper on that mechanism.

Race Multiplicity: The stability $\Delta(g_i)$ of the flatmap transformation g_i mapping individuals to population groups in level i is a significant factor in the noise scale required to satisfy zCDP. The data collection process restricts individuals to a maximum of 8 detailed race codes and 1 ethnicity code, which translates to a flatmap stability of 9 for any given population group level. This is because an individual with 8 unique race codes can be associated with at most 8 Alone or in Any Combination characteristic iterations for a level plus the one ethnic characteristic iteration. Higher stability equates to higher variance noise, all else held equal, so an option to improve the noise variance would be to reduce the stability by setting a lower cap on the number of race codes processed for each individual. For example, if individuals were restricted to 3 race codes instead of 8, the stability would drop from 9 to 4 resulting in a 33% decrease in MOE when holding the privacy loss constant. However, the restriction would also introduce another form of bias into the statistics and potential artificially reduce the set of population groups with true positive counts. The DSEP committee opted to leave the race multiplicity parameter at 8.

Population Group Level	MOE Target	Unbounded Privacy Loss		Bounded Privacy Loss	
		Step 2	Total	Step 2	Total
(Nation, Detailed): ρ_1	3	1.921	2.134	3.842	4.268
(State, Detailed): ρ_2	3	1.921	2.134	3.842	4.268
(County, Detailed): ρ_3	11	0.143	0.159	0.286	0.318
(Tract, Detailed): ρ_4	11	0.143	0.159	0.286	0.318
(Place, Detailed): ρ_5	11	0.143	0.159	0.286	0.318
(AIANNH, Detailed): ρ_6	11	0.143	0.159	0.286	0.318
(Nation, Regional): ρ_7	50	0.007	0.008	0.014	0.016
(State, Regional): ρ_8	50	0.007	0.008	0.014	0.016
(County, Regional): ρ_9	50	0.007	0.008	0.014	0.016
(Tract, Regional): ρ_{10}	50	0.007	0.008	0.014	0.016
(Place, Regional): ρ_{11}	50	0.007	0.008	0.014	0.016

Table 7: MOE targets for the statistics released (in Step 2 of the adaptive algorithm) at different population group levels, along with the corresponding privacy loss (unbounded and bounded ρ-zCDP for discrete Gaussian). The privacy loss is reported for the Step 2 (to match the MOE) as well as the total loss for that level. Step 2 loss is 90% of Total loss at each population group level. Note that the privacy losses reported here have already been aggregated over all the population groups at the given level, so the *Total* column represents the privacy loss input parameters of the SafeTab-P algorithm.

MOE, ρ, and γ: Recall that γ is the fraction of ρ reserved for the Step 1 of the adaptive procedure in SafeTab-P. The SafeTEx tool provided an interface for adjusting ρ's and γ to observe the impact on expected MOE as derived in Section 6.1. The Census Bureau selected $\gamma = 0.1$ and set ρ_i's as displayed in Table 7 for the production run of SafeTab-P on the 2020 Census data.

7   Conclusion

In this article, we presented SafeTab-P, a differentially private algorithm designed for releasing the Detailed DHC-A data product. We explained the adaptive nature of SafeTab-P and proved the algorithm satisfies zCDP. We also looked at selected implementation details, including how the SafeTab-P program was built on the Tumult Analytics platform. We described our contributions to the tuning of parameters for SafeTab-P. In future papers, we will discuss algorithms designed for the release of other 2020 Census data products, such as the Detailed DHC-B and S-DHC.

References

[1] 2020 Census Detailed Demographic and Housing Characteristics File A (Detailed DHC-A) Technical Documentation. `https://www2.census.gov/programs-surveys/decennial/2020/technical-documentation/complete-tech-docs/detailed-demographic-and-housing-characteristics-file-a/2020census-detailed-dhc-a-techdoc.pdf`.

[2] 2020 Census Operational Plan. `https://www2.census.gov/programs-surveys/decennial/2020/program-management/planning-docs/2020-oper-plan5-and-memo.pdf`.

[3] U.S. Code Title 13—Census. `https://www.law.cornell.edu/uscode/text/13`.

[4] Memorandum 2019.25: 2010 Demonstration Data Products – Design Parameters and Global Privacy-Loss Budget. `https://www.census.gov/programs-surveys/decennial-census/2020-census/planning-management/memo-series/2020-memo-2019_25.html`, October 2019.

[5] Skye Berghel, Philip Bohannon, Damien Desfontaines, Charles Estes, Sam Haney, Luke Hartman, Michael Hay, Ashwin Machanavajjhala, Tom Magerlein, Gerome Miklau, Amritha Pai, William Sexton, and Ruchit Shrestha. Tumult Analytics: A Robust, Easy-to-Use, Scalable, and Expressive Framework for Differential Privacy. `https://arxiv.org/abs/2212.04133`, 2022.

[6] Seth Borenstein. Potential privacy lapse found in Americans' 2010 census data. `https://apnews.com/article/aba8e57c145047b5bab11b62baaa7f7a`, February 2019.

[7] Mark Bun and Thomas Steinke. Concentrated Differential Privacy: Simplifications, Extensions, and Lower Bounds. *CoRR*, abs/1605.02065, 2016.

[8] Clément L. Canonne, Gautam Kamath, and Thomas Steinke. The Discrete Gaussian for Differential Privacy. *CoRR*, abs/2004.00010, 2020.

[9] Samuel Haney, William Sexton, Ashwin Machanavajjhala, Michael Hay, and Gerome Miklau. Differentially Private Algorithms for 2020 Census Detailed DHC Race & Ethnicity. *CoRR*, abs/2107.10659, 2021.

[10] Rachel Marks and Merarys Rios-Vargas. Improvements to the 2020 Census Race and Hispanic Origin Question Designs, Data Processing, and Coding Procedures. `https://www.census.gov/newsroom/blogs/random-samplings/2021/08/improvements-to-2020-census-race-hispanic-origin-question-designs.html`.

[11] Frank McSherry. Privacy Integrated Queries: An Extensible Platform for Privacy-Preserving Data Analysis. In Ugur Çetintemel, Stanley B. Zdonik, Donald Kossmann, and Nesime Tatbul, editors, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 - July 2, 2009*, pages 19–30. ACM, 2009.