Generate-then-Verify: Reconstructing Data from Limited Published Statistics

Terrance Liu*, Eileen Xiao*, Adam Smith†, Pratiksha Thaker*, and Zhiwei Steven Wu*

*Carnegie Mellon University
†Boston University

***Abstract*—We study the problem of reconstructing tabular data from aggregate statistics, in which the attacker aims to identify interesting claims about the sensitive data that can be verified with 100% certainty given the aggregates. Successful attempts in prior work have conducted studies in settings where the set of published statistics is rich enough that entire datasets can be reconstructed with certainty. In our work, we instead focus on the regime where many possible datasets match the published statistics, making it impossible to reconstruct the entire private dataset perfectly (i.e., when approaches in prior work fail). We propose the problem of partial data reconstruction, in which the goal of the adversary is to instead output a *subset* of rows and/or columns that are *guaranteed to be correct*. We introduce a novel integer programming approach that first generates a set of claims and then verifies whether each claim holds for all possible datasets consistent with the published aggregates. We evaluate our approach on the housing-level microdata from the U.S. Decennial Census release, demonstrating that privacy violations can still persist even when information published about such data is relatively sparse.**

1. Introduction

The problem of data privacy lies at the heart of data stewardship. While many organizations aim to provide data products that maximize utility for downstream users, this goal is at direct odds with protecting the privacy of those who contribute data. In this paper, we study this problem from the perspective of tabular data reconstruction, in which an adversary is given access only to a set of aggregate statistics about the private dataset. Specifically, we are interested in the setting in which the adversary aims to reconstruct (some portion of) the private dataset with absolute certainty. In other words, we answer the question, *"What must exist in the private dataset according to the published statistics?"*

The aforementioned problem of data stewardship is at the forefront of issues faced by the U.S. Census Bureau, which provides billions of statistics to the public while needing to fulfill a legal mandate to protect the privacy of its respondents [1]. As a result, the bureau itself has conducted various studies investigating the vulnerability of the US Decennial Census release to potential reconstruction attacks. Most recently, for example, Abowd et al. [2] tackle this problem from the lens of guaranteeing correctness (as part of a larger set objectives in their work) and find that by using 34 person-level tables from the 2010 Summary File 1, one can reconstruct the entire data for 70% of the blocks in the United States with 100% certainty simply by solving an integer program.

Such alarming results suggest that reconstruction of person-level data using the Decennial Census release is far too easy—the amount of information (statistics) available to the adversary is so rich that reconstruction becomes trivial for the majority of blocks. In light of this observation, one might ask whether releasing less descriptive statistics that do not admit a unique IP solution would be sufficient to protect individuals' information. In this work, we therefore study to what extent data reconstruction with 100% certainty can still occur even in more difficult regimes in which the released statistics are not informative enough (relative to the size of the data domain) for prior approaches (e.g., Abowd et al. [2]) to reconstruct *entire* tabular datasets with absolute certainty.

Contributions. We summarize our contributions as the following:

1) We introduce the problem of *partial* tabular data reconstruction to help better understand the vulnerability of data releases like the Decennial Census: rather than reconstructing the entire dataset with guaranteed correctness, the adversary aims to output *verified* claims about individuals in the data (see, e.g., Figure 1).
2) We consider claims about the number of rows with specific values in a subset of columns, such as "in this dataset, there exists exactly one household whose head of the household is a 32-year old, Black woman." In particular, inspired by Cohen and Nissim [3], we focus on reconstructing *"singleton claims"*, which are reconstructed attributes that single out exactly one individual in the dataset.
3) We introduce an integer programming formulation that departs from the approaches of previous work [2], [4], [5] and allows us to tackle this problem. Specifically, given some set of aggregate statistics about the dataset, our method **(1)** generates a set of candidate claims and

• *First two authors contributed equally. Remaining authors are ordered alphabetically.*

TABLE 1: Examples of verified, singleton claims (multiplicity $m = 1$)

Block, Tract, County, State	Reconstructed Information
1008, 010200, Baldwin, AL	A household with just a single female householder. She owns the home without a mortgage. The householder is white, of Hispanic or Latino origin, and is between 65 and 75 years old.
3027, 271801, Baltimore City, MD	A renting household of size 2. It is a non-family household, and no one in the household is under 18 or over 65 years old. The householder is black, not of Hispanic or Latino origin, and between 25 and 34 years old.
1006, 564502, Wayne, MI	A household of size 4 with a married couple that owns the home with a mortgage. No one in the household is over 65 years old, but there is at least one child under 18 years old. The householder is Black, not of Hispanic or Latino origin, and is between 45 and 54 years old.
1049, 005828, Clark, NV	A married couple household (of unknown size) that does not own the home but also does not pay rent. No one in the household is over 65 years old, but there is at least one child under 18 years old. The householder of Hispanic or Latino origin and between 25 and 34 years old. Their race does not belong to one of the 5 major census race categories.
1087, 940100, McKenzie, ND	A renting household of size 4. There is a cohabiting couple living with at least one child under 18 years old. No one in the household is over 65 years old. The householder is male, American Indian/Alaskan Native, not of Hispanic or Latino origin, and between 15 and 24 years old.

then **(2)** verifies whether these claims must be true according to the published statistics.

4) We evaluate our approach and that of previous work on the household unit-level data and tables from the Decennial Census release (2010 Summary File 1 (SF1)). We find that the method proposed in Abowd et al. [2] for reconstructing entire blocks is ineffective—in our experiments, not a single block could be reconstructed uniquely. In contrast, our approach reconstructs many individual households with 100% certainty, demonstrating that partial reconstruction is still feasible, even when full reconstruction is not (Table 1 provides examples of verified claims).
5) We find that a nontrivial number of households can be reconstructed using some subset of columns that uniquely identifies them (i.e., singles them out). Among the blocks evaluated in our experiments, approximately 40% contain at least one household that can be singled out by 8 (out of 10 total) columns (Figure 3), averaging out to one household per block (Table 4). For 6 columns, the percentage of blocks containing singled out households increases to over 80% (Figure 3).

1.1. Additional Related Work

Real-world examples of privacy risks resulting from aggregate statistical releases have long been well-documented [1], [5], [6], [7], [8]. As a result, a long line of research, beginning with the seminal work of Dinur and Nissim [9], have both studied reconstruction attacks using public statistical information [10], [11], [12], [13] and developed notions for privacy guarantees—namely, Differential Privacy [14].

Mitigating such privacy risks [1], [2], [5], [8], [15], [16] remains at the center of issues facing the U.S. Census Bureau, which has addressed such privacy concerns by incorporating Differential Privacy into the 2020 Decennial Census release [17]. Our work, in part, extends such findings, further demonstrating the risks that individuals face when aggregate statistics derived from them are released freely. While Abowd et al. [2] show that at the person-level, the majority of blocks can be completely reconstructed, their method relies on the Decennial Census release being rich enough so that there can only exist one possible set of individuals that correspond to the released statistics. At the household-level, in which there are far more columns but relatively the same number of statistics, this condition is no longer met—we find that for any given block, there exists many possible solutions (groups of households) that would produce the same set of released statistics. Nevertheless, we devise a method that can partially reconstruct a block with absolute certainty.

Lastly, the notion of singleton claims, which later works like Cohen and Nissim [3] expand upon, can be traced back to as early as Sweeney [6], which exposed the susceptibility of uniquely identified individuals to linkage attacks. As a by-product of reconstructing entire datasets (e.g., Abowd et al. [2]), one can single out individuals by identifying the rows that are unique in the reconstruction. We, however, make the observation that complete reconstruction is not strictly necessary for the purpose of singling out. Focusing on *partial* reconstruction, our work demonstrates that individual records can still be reconstructed with certainty and that some individuals can be singled out even by just a subset of the columns in the data domain.

2. Preliminaries

In this setting, we have some *dataset* D that is comprised of a multiset of N records from a discrete domain $\mathcal{X}$. Let Q be some set of n queries corresponding to the data domain $\mathcal{X}$, and let $Q(D) \in \mathbb{R}^n$ be a vector of aggregate statistics on dataset D where each element is a statistic corresponding to a query in Q. Then, in its most general form, tabular data reconstruction can be set up as a simple constraint satisfaction problem (i.e., find any dataset D' that matches the statistics Q),

$$\text{Find } D' \quad \text{s.t. } Q(D) = Q(D') \tag{1}$$

In our work, we consider statistics in the form of counting queries

$$q_\phi(D) = \sum_{x \in D} \phi(x), \tag{2}$$

where $\phi(x)$ denotes the condition that indicates whether a row x satisfies some property. Thus, $q_\phi(D)$ counts the number of rows $x \in D$ that satisfy that property. Our work focuses on k-way marginal queries[1], where the ϕ indicates whether a set k columns matches some set of values (e.g., SEX $= Male$ and AGE $\in \{10, 20\}$). Table 2 provides an example of a set of queries tabulated in the U.S. Decennial Census Release.

2.1. Record-level reconstruction.

In our work, we focus on *record*-level reconstruction, where the goal is to output claims about sets of rows $x \in \mathcal{X}$. Suppose there exists k columns in $\mathcal{X}$ such that we rewrite $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_k$. Let $\mathcal{X}'_i = \mathcal{X}_i \cup \{\perp\}$, where $\perp$ indicates that column i can take on any value in $\mathcal{X}$. A vector a in $\mathcal{X}' = \mathcal{X}'_1 \times \ldots \times \mathcal{X}'_k$ specifies a *partial assignment* to the attributes, and we say x matches a if they agree in all coordinates where $a_i \neq \perp$.

Using this notation, we define $R(a, m)$ to be the (reconstruction) **claim** that there exist **exactly** $m \in \{0, 1, \ldots, N\}$ rows (e.g., $m = 2$ in Figure1; claim 1) that match $a \in \mathcal{X}'$ (e.g., a describes a 35-year old, Black householder in Figure1; claim 1). We can then define a *singleton* claim as some claim $R(a, m)$ where $m = 1$.

Let $\text{COUNT}(a, D) : \mathcal{X}' \times \mathcal{X}^N \to \mathbb{N} \cup \{0\}$ be the number of rows in D that match a. Then we say a claim $R(a, m)$ is correct for some dataset D if $\text{COUNT}(a, D) = m$.

Finally, we define *verified* claims as the following:

Definition 1 (Verified Claim). Given some set of summary statistics $Q(D)$, we say that a claim $R(a, m)$ is *verified* with respect to $Q(D)$ if and only if

$$\text{COUNT}(a, D') = m$$

for all datasets D' such that $Q(D) = Q(D')$.

1. In the typical formulation of k-way marginals, ϕ checks whether a column is equal to one specific value (e.g., AGE $= 10$). Our work considers a more general definition, where the column can take on a set of values (e.g., AGE $\in \{10, 20\}$).

In other words, a claim is verified (i.e., guaranteed to be correct) if it must be correct for any dataset D' where $Q(D) = Q(D')$.

2.2. Guaranteeing the correctness of claims.

At a high level, to verify the correctness of any claim $R(a, m)$, one can ask the question: is it possible to construct a synthetic dataset D' that matches the published statistics, even when the multiplicity of number of rows with attributes a does not equal m? If such a dataset D' does not exist, then $R(a, m)$ must be correct. Concretely then, we check claims by again solving Problem 1 but with the added constraint that $\text{COUNT}(a, D) \neq m$:

$$\text{Find} \quad D' \quad \text{s.t.} \quad Q(D) = Q(D') \text{ and } \text{COUNT}(a, D) \neq m. \tag{3}$$

Note that in this formulation, we can make use of all statistics (queries Q defined over all columns in $\mathcal{X}$) available to us, even when verifying claims that are defined over only some subset of columns in $\mathcal{X}$ (attributes a where $a_i = \perp$ for some columns i).

2.3. Generate-then-Verify

At a high level, our approach can be broken down into two integer programming steps:

1) **Generate:** We generate a list of claims $R(a, m)$ that we then verify in step 2. Specifically, we solve Problem 1 $K = 100$ times.[2] For each generated synthetic dataset D', we identify all claims $R(a, m)$ (i.e., all possible combinations of attributes a and the corresponding multiplicities m in D'). We then take the intersection of the K sets of claims to use as our final list.[3]
2) **Verify:** For each claim $R(a, m)$, we check if Problem 3 is feasible via integer programming. If no solutions can be found, then we conclude that $R(a, m)$ must be correct.

We defer to Section 5.1 the details of how we encode the input for the integer programming solver.

3. Empirical Evaluation

3.1. Setup

3.1.1. Dataset. In our experiments, we use the 2010 Privacy-Protected Microdata File, a synthetic dataset, statistically similar to the private 2010 Decennial Census microdata, that is generated and released by the U.S. Census Bureau. As the private 2010 Census microdata are not public, we treat

2. In Gurobi, we can simply set the solver to output up to K solutions that satisfy the constraints.

3. The intersection contains *all* claims that are plausible based on the aggregate statistics $Q(D)$. If a claim does not belong in the intersection, then there exists some feasible reconstruction D' consistent with $Q(D)$ that refutes the claim, meaning that we cannot be certain that the claim is correct for D. We explain this filtering logic further in 5.4.2.

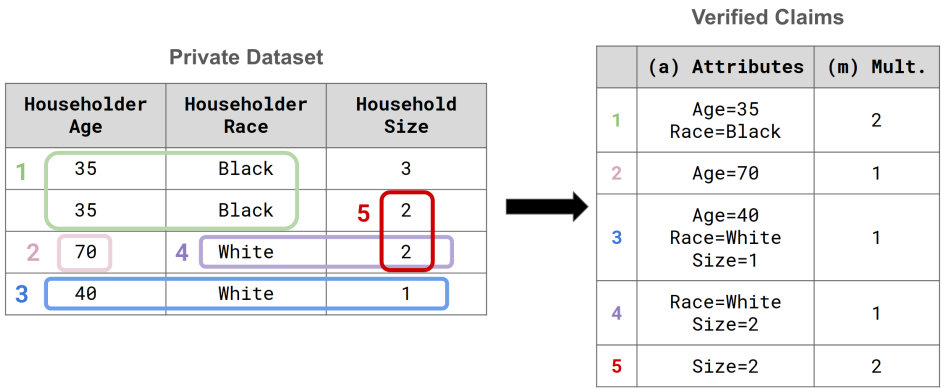


Figure 1: We provide a visual diagram of example claims studied in our work. On the left-hand side is the private dataset, where the colored boxes denote various claims $R(a, m)$ that are then enumerated in the table on the right-hand side.

TABLE 2: We provide an example of a table (Summary File 1: P20) released in the Decennial Census, including the text descriptions of each query contained in the table and the count of households matching that description for some block. In detail, **Condition** ϕ denotes what each query checks for (e.g., query 1 checks whether column HHT2 = 1), and indented rows mean that the corresponding query must satisfy the condition *and* all parent conditions above them. For example, query number 10 corresponds to HHT2 = 9, while query number 11 corresponds to HTT2 = 8 *AND* THHLDRAGE = 7, 8, or 9. **Text Description** describes what each value means. For example, HHT2 = 1 means that the household is a *married couple household with their own children under the age of 18*.

Query No.	Condition ϕ (Column = Value)	Text Description	Count
		Married couple household:	
1	HHT2 = 1	With own children under 18	6
2	HHT2 = 2	No own children under 18	1
		Cohabiting couple household:	
3	HHT2 = 3	With own children under 18	0
4	HHT2 = 4	No own children under 18	0
		Female householder, no spouse or partner present:	
5	HHT2 = 5	Living alone	0
6	THHLDRAGE = 7, 8, or 9	65 years and over	0
7	HHT2 = 6	With own children under 18	1
8	HHT2 = 7	With relatives, no own children under 18	0
9	HHT2 = 8	No relatives present	0
		Male householder, no spouse or partner present:	
10	HHT2 = 9	Living alone	1
11	THHLDRAGE = 7, 8, or 9	65 years and over	0
12	HHT2 = 10	With own children under 18	0
13	HHT2 = 11	With relatives, no own children under 18	1
14	HHT2 = 12	No relatives present	0

the Privacy-Protected Microdata as the ground truth during evaluation. The PPMF (and Summary File 1, from which the PPMF is derived from) contains data for every housing unit in the United States. Each row of the PPMF represents one synthetic household response from the 2010 Decennial Census. There are 10 columns in total described by block-level tables (listed in Appendix A), in contrast to the simpler, person-level data studied in prior work [2], [15] that only contains 4 columns.

From each U.S. state (50 in total), we select blocks in the following ways:

- For each state, we calculate the median block size (Figure 2) and randomly select 5 blocks of that size.
- We calculate the median block size ($N = 10$) of the country and select 5 blocks of that size from each state.

Crucially, unlike for the census release of the person-level data studied in Abowd et al. [2], **no blocks we evaluate on can be fully reconstructed with 100% certainty**—when solving Problem 1, we found at least 2 different solutions D' for every block, meaning each block D is not uniquely identifiable by the released statistics Q.

3.1.2. Statistics. In addition to the Privacy-Protected Microdata File, the U.S. Census Bureau releases aggregate statistics of features listed above, calculated from their private microdata, in the form of data tables called Summary File 1 (SF1). Each of these tables are released for every block and includes counts for the number of people corresponding

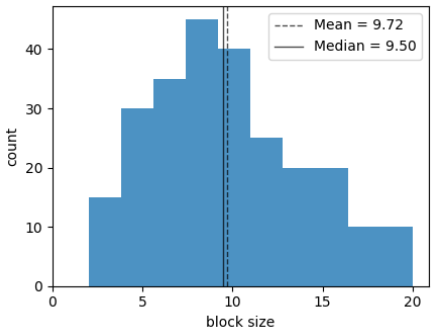


Figure 2: We plot the distribution of block sizes for blocks whose size equals the median block size in the state. The minimum block size is 2 and the maximum is 20.

to certain feature values defined by the table. As noted previously, these statistics correspond to k-way marginal queries, where $k \leq 4$ columns[4] for SF1. In total, we have 24 partial sets of k-way marginals (see Table 3).

k	1	2	3	4	total
# k-way marginal sets	4	10	8	2	24

TABLE 3: Number of sets of k-way marginals per value k.

We provide an example of a table from this release in Table 2. Here, query number 11 counts the number of households owned by a female householder who is over the age of 65 and lives alone. As suggested by Table 2, the statistics released by the Census Bureau only partial cover the k-way histogram for any set of k attributes. For example, Table 2 bins together the values 7, 8, and 9 for column THHLDRAGE and does not tabulate over instances where THHLDRAGE takes on values 1-6.

Utilizing all tables (listed in Appendix A) tabulated at the block-level, we have $|Q| = 621$ queries as inputs to our integer programming approach.

3.2. Baseline reconstruction rates

While we contend that finding any records that can be reconstructed with 100% confidence is already interesting, we would like to further provide context for our results by providing some baseline measure for how likely a block corroborates some claim. To do so, we calculate the probability of each verified claim being correct in a block of size N that is randomly sampled from the tract or state that the block is located in.

Let us assume that records in a block are drawn from some prior distribution P. Then the multiplicity m of some candidate record x appearing in a block D of size N follows the binomial distribution,

$$P(\textsc{Count}(x, D) = m) = \binom{N}{m} p^m (1-p)^{N-m}, \quad (4)$$

where $p = P(x)$ is the probability of a single record x being drawn from the prior P.

In typical settings in which an adversary has no prior information about the block of interest, p is simply the uniform distribution (i.e., $P(x) = \prod_{j=1}^{k} \frac{1}{|\mathcal{X}|} = \prod_{j=1}^{k} \frac{1}{|\mathbf{x}^{(j)}|}$, where $\mathbf{x}$ is the one-hot encoded representation of x with columns $\{c_j\}_{j=1}^{k}$). However, this comparison is uninteresting since $P(\textsc{Count}(x, D) = m)$ is close to 0 in such cases.

In our evaluation, we instead construct a setting in which we assume that the prior distribution of the *tract* and *state* that some block D belongs to is known (similar to baselines considered in Dick et al. [15]). Let D_{tract} and D_{state} be the set of records in the tract and state. Then, we can express p as

$$P(x) = \frac{\textsc{Count}(x, \tilde{D})}{|\tilde{D}|}$$

for $\tilde{D} = D_{\text{tract}}$ and $\tilde{D} = D_{\text{state}}$, respectively, and use Equation 4 to calculate the baseline probability for any given candidate claim x.

4. Results

We now present our empirical results for verified *singleton* claims.[5] For conciseness, we report results only for claims that cover $k \geq 6$ columns since the claims containing more attributes are relatively more interesting.

We also note that there exist claims that can be "read" directly off the tables themselves. For example, a table reporting 2 households with White householders already tells us that the claim $R(\text{Race=White}, 2)$ must be correct. As mentioned in Section 3, however, the marginal statistics capture at most, $k = 4$ columns. Thus, none of the claims reported in this section (i.e., with $k \geq 6$) are among this set of "trivial" claims.

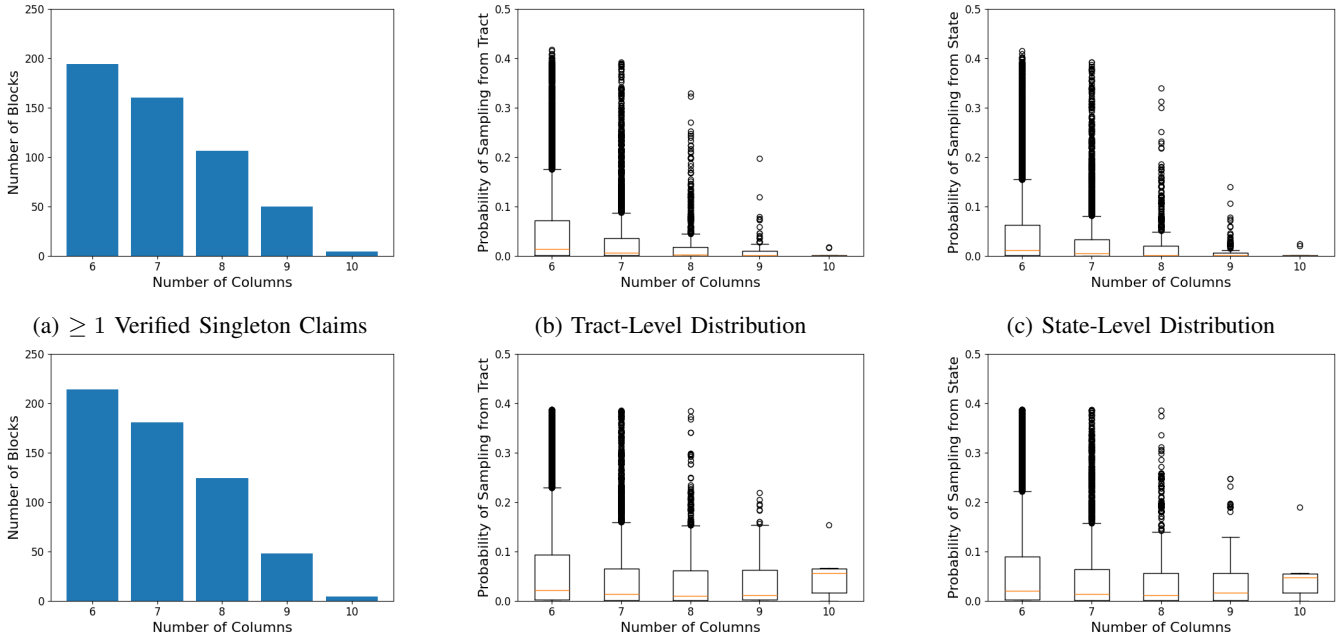
4.1. Main findings

We present our main results in Figure 3 and Table 4. In both the figure and table, we split the 500 blocks into two sets: one for blocks whose size equals the national median and one for those whose size equals the respective state median. Interestingly, the results do not differ much across the two sets, suggesting that reconstruction rates do not depend heavily on the size of the blocks we evaluated on.

In Figure 3a and 3d, we report the number of blocks (y-axis) for which we can reconstruct some set of k columns (x-axis) for *at least* one household. We find that while reconstruction (with 100% certainty) of all 10 columns is

4. Using the notation presented in Section 5.1, $k \leq 4$ is equivalent to saying that $r_{\max} = 4$.

5. Figures and tables for all claims, regardless of multiplicity, can be found in Appendix B.



(a) ≥ 1 Verified Singleton Claims (b) Tract-Level Distribution (c) State-Level Distribution

(d) ≥ 1 Verified Singleton Claims (e) Tract-Level Distribution (f) State-Level Distribution

Figure 3: We present results for verified **singleton** claims collected from experiments on 5 blocks selected from each state (250 total on each row). **Top row:** 5 blocks whose size are equal the median block size of the respective state are selected. **Bottom row:** 5 blocks whose size are equal the median block size of the country (i.e., 10 households). **a & d:** The number of blocks (out of 50) for which we can reconstruct at least one singleton claim about k columns (x-axis). **b & e and c & f:**: Box and whisker plots of the probabilities that each verified claim would also be true in a set of N households randomly sampled from the (b, e) tract and (c, f) state-level distributions. The orange line within each box indicates the median probability. The ends of the box indicate the first and third quartiles, and the whiskers end at the furthest point within 1.5 times the interquartile range. All points beyond the whiskers are outliers. Lower probabilities denote more "surprising" claims.

TABLE 4: For each number of columns, we report total and average number of households that are represented among the verified **singleton** claims. We tabulate the verified claims over 500 total blocks: (top two rows) 5 blocks from each state whose size is equal to the median block size in the state and (bottom two rows) 5 blocks from each state whose size is equal the country median block size (i.e., 10 households). n is the total and average number of households over all 250 blocks.

			# of households identified by verified claims w/ k columns				
		# households	k=6	7	8	9	10
State Median	Total	2500	659	471	254	97	7
	Avg. per block	10.00	2.64	1.88	1.02	0.39	0.03
Country Median	Total	2430	669	437	230	71	6
	Avg. per block	9.72	2.88	1.88	0.99	0.31	0.03

often not possible, we can still partially reconstruct singletons from most blocks. For example, we verify at least one singleton claim with $k = 8$ columns in approximately 40% of the blocks and for $k = 6$, we can verify at least one claim in 80% of them.

In Figures 3b, 3c, 3e, and 3f, we evaluate the baseline probabilities (Section 3.2; Equation 4) for all claims verified by our approach to understand how "surprising" they are, given some prior information about the state and tract demographics. Interestingly, the distribution of probabilities is similar for both tract and state-level priors, suggesting using the tract-level prior is no more informative than the state-level one. We find that in general, these baseline probabilities are quite low. In almost cases (except verified claims with $k = 10$; Figures 3e and 3f), the median probability is under 2% (and often is very close to 0%). The 75th and 90th percentiles are under 10% and 25% respectively, and even among outliers, the maximum baseline probability never exceeds 50%. As stated previously, we argue that verifying any claim with *100% certainty* is already interesting and significant. However, these results help demonstrate that if someone were to make guesses about households based on

prior information about the tract or state, it is highly unlikely that these guesses would include the claims that our approach outputs.

Finally, in Table 4 we report the number of unique households that we reconstruct, given some number of columns k. Here, instead of counting the total number of verified claims, we total up the number of unique households covered by the claims.[6] Again, despite the difficulty of reconstructing all $k = 10$ columns of households, we find that a non-trivial fraction (over 10%) of households are uniquely identifiable by some claim that describes $k = 8$ columns. This proportion increases to over a quarter when considering claims that describe $k = 6$ columns.

4.2. Ablation: removing single count queries

TABLE 5: We report how the total number of households that are represented among the verified **singleton** claims changes when different sets of queries are removed as input to our approach. For example, the second row corresponds to removing queries that evaluate to 1 ($q(D) = 1$). We evaluate on blocks (250 in total) whose size is equal to the country median (i.e., $N = 10$ for all blocks).

input queries	% queries removed	# of households identified by verified claims w/ k columns k=6	7	8	9	10
Q	0%	669	437	230	71	6
$Q \setminus \{q(D) = 1\}$	3.40%	249	132	59	18	0
$Q \setminus \{q(D) = 0\}$	90.88%	460	246	85	7	0
$Q \setminus \{q(D) = 0, 1\}$	94.28%	47	20	9	0	0

In Section 4.1, we present results for verified singleton claims that cannot be read directly off the input tables (i.e., number of columns $k > 4$). We note however that in some cases, it might be possible for humans to manually find additional claims without too much difficulty by combining single count queries (queries $q(D) = 1$). To give a toy example, suppose we have the following statistics:

1) $\sum \mathbb{I}\{A = 0, B = 0\} = 1$
2) $\sum \mathbb{I}\{B = 0, C = 0\} = 1$
3) $\sum \mathbb{I}\{B = 0\} = 1$

Queries 1 and 2 tell us that there is exactly one row with columns $A = 0$ and $B = 0$ and one row with $B = 0$ and $C = 0$. Because query 3 tells us that there exactly one row with $B = 0$, we know that query 1 and 2 are describing the same row. Thus, one can look at this set of queries and deduce an additional claim that there is a singleton with the attributes $A = 0$, $B = 0$, and $C = 0$.

Therefore, to further eliminate the possibility of including "easy" claims in our results, we simulate a setting where queries that evaluate to 1 are removed from our integer programming approach. Evaluating only on the country median-sized blocks so the block size in our ablation study is fixed, we report in Table 5 how the number of households that can be uniquely identified changes when the single count queries are removed. We find that although the number of singled out households decreases significantly,[7] a nontrivial number are still uniquely identifiable. For example, approximately 10% (249 out of 2500) households can still be singled out by $k = 6$ columns.

To stress test our approach, we also report in Table 5 the number of uniquely identifiable households when we remove queries that (a) evaluate to 0 and (b) evaluate to 0 or 1. Unsurprisingly, many of the households are no longer identifiable, especially in the case where queries that evaluate to 0 or 1 are removed. Still, we show that some privacy risks persist, given that a nonzero number of households are singled out by $k = 6$ to 8 columns.

4.3. Analysis of reconstructed columns

Finally, in Figure 4 and Table 6, we take a closer look at what columns make up the verified claims outputted by our approach. As shown in Figure 4, there generally is an even distribution of columns represented in the verified claims. However, the column HHT2 (detailed household type) is most often omitted, followed by TP65 (presence of someone over 65 years). Examining the most common combinations of k columns reconstructed by approach, we observe similar patterns in Table 6. For example, for each number of columns k, the most common set of columns does not include HHT2. In fact, HHT2 does not appear at all among the top five most common combinations for $k \leq 8$. Similarly, TP65 does not appear in the top for $k \leq 7$.

5. Integer Programming Details

In this section, we describe the exact details of our integer programming approach, including how we set up and solve Problems 1 and 3.

5.1. Setup

5.1.1. One-hot encoded records. Unlike prior work [2], [4], [5] which represents datasets as histograms over $\mathcal{X}$, our proposed integer programming optimization problem relies on one-hot encoded representations of $\mathcal{X}$. Specifically, let k be the number of columns, which we denote as columns $\{c_j\}_{j=1}^{k}$, in the domain. Given that all columns in $\mathcal{X}$ are discrete, we represent records in $\mathcal{X}$ as one-hot encoded vectors $\mathbf{x} = (\mathbf{x}^{(1)} \dots \mathbf{x}^{(k)})$, where each $\mathbf{x}^{(j)}$ encodes the column c_j. Thus, we have rows $\mathbf{x} \in \{0,1\}^d$ where $d = \sum_{j=1}^{k} |\mathbf{x}^{(j)}|$ and $k = \sum_{i=1}^{d} \mathbf{x}_i$. Finally, we let the matrix $\mathbf{X} \in \{0,1\}^{N \times d}$ denote a one-hot encoded dataset with N rows.

6. For example, in Figure 1, the total multiplicity of claims 2 and 4 is two. However, only one household is represented among these claims (row 3 on the left-hand table). Thus, Table 4 groups the verified claims by the number of columns (in a) and reports the number of households (in the left-hand table) that are represented in the claims.

7. On average, 3.4% of queries are removed for each block. However, because 90.88% of queries evaluate to 0, the single count queries account for almost 37.28% of nonzero queries counts. As a result, it is unsurprising that the number of verified singleton claims decreases by such a large amount.

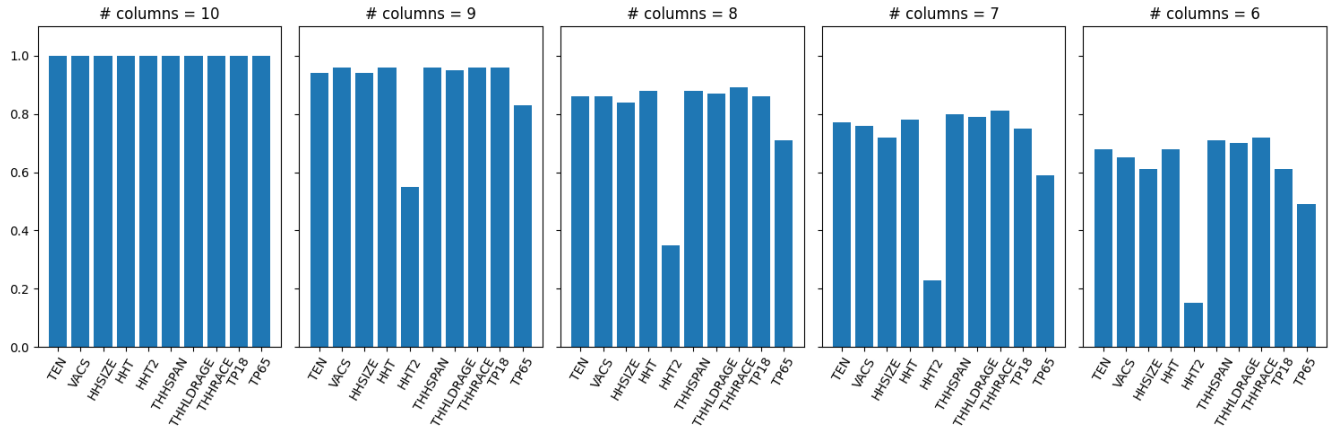


Figure 4: For each number of columns k, we plot the proportion of verified singleton claims that contain each column.

TABLE 6: For each number of columns k, we list the 5 most common combinations of columns among the verified claims. In addition, we report what percentage of claims with k columns each combination makes up. A checkmark (✓) indicates that the column is included. For example, 44.54% of claims comprised of $k = 9$ columns omit the column HTT2, while 16.59% omit the column TP65.

# columns	TEN	VACS	HHSIZE	HHT	HHT2	THHSPAN	THHLDRAGE	THHRACE	TP18	TP65	%
10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
9	✓	✓	✓	✓		✓	✓	✓	✓	✓	44.54%
	✓	✓	✓	✓	✓	✓	✓	✓	✓		16.59%
	✓	✓		✓	✓	✓	✓	✓	✓	✓	6.11%
		✓	✓	✓	✓	✓	✓	✓	✓	✓	6.11%
	✓	✓	✓	✓	✓	✓		✓	✓	✓	4.8%
8	✓	✓	✓	✓		✓	✓	✓	✓		12.61%
	✓	✓		✓		✓	✓	✓	✓	✓	8.01%
	✓		✓	✓		✓	✓	✓	✓	✓	7.58%
	✓	✓	✓	✓		✓	✓	✓		✓	7.03%
	✓	✓	✓	✓		✓		✓	✓	✓	6.82%
7	✓	✓	✓	✓		✓	✓	✓			4.68%
	✓	✓		✓		✓	✓	✓	✓		4.37%
	✓		✓	✓		✓	✓	✓	✓		3.81%
	✓	✓	✓			✓	✓	✓	✓		3.13%
		✓	✓	✓		✓	✓	✓	✓		3.07%
6	✓	✓		✓		✓	✓	✓			3.21%
	✓		✓	✓		✓	✓	✓			2.31%
	✓			✓		✓	✓	✓	✓		2.03%
	✓	✓	✓			✓	✓	✓			1.9%
		✓	✓	✓		✓	✓	✓			1.83%

5.1.2. Query functions. In this setting, we consider statistical queries (Equation 2) in the form of marginal queries where the predicate function ϕ is an indicator function for whether some set of columns takes on some set of values (note that ϕ is equivalent to what we call attributes a in Section 2). For example, one can ask the marginal query about the columns SEX and RACE: "How many people are (1) FEMALE and (2) WHITE or BLACK?" We note that one can break down any predicate ϕ into a set of sub-predicates, where each sub-predicate corresponds to one unique column pertaining to ϕ. Concretely, given some column c and target values V, we denote the *sub-predicate* function as

$$\phi_{c,V}(x) = \mathbb{I}\{x_c \in V\},$$

where x_c is the value that x takes on for column c. Then, any predicate can be rewritten as the product of its sub-predicates (e.g. in the above example, ϕ can be written as the product of $\phi_{\text{SEX},\{\text{FEMALE}\}}$ and $\phi_{\text{RACE},\{\text{WHITE, BLACK}\}}$).

Given the one-hot encoded representation $\mathbf{x}$, $\phi_{c,V}$ can also be rewritten as a vector $\mathbf{q} \in \{0,1\}^d$ that takes on the value 1 for indices in $\mathbf{x}$ corresponding column c and values $v \in V$ (and 0 otherwise). In this case, we can then rewrite the sub-predicate function as $\mathbf{x}\mathbf{q}^T$. Likewise, any predicate with r sub-predicates $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_r$ can be rewritten as a matrix $\mathbf{Q} = \begin{pmatrix}\mathbf{q}_1 & \mathbf{q}_2 & \ldots & \mathbf{q}_r\end{pmatrix}^T \in \{0,1\}^{r \times d}$. Then ϕ can be written as $\mathbb{I}\{\mathbf{x}\mathbf{Q}^T = \mathbf{1}_r\}$ where $\mathbf{1}_r$ is a row vector of ones with length r. Finally, a statistical query q_ϕ can be

written as

$$q_\phi(x) = \sum_{j=1}^{N} \mathbb{I}\{(\mathbf{X}\mathbf{Q}^T)_j = \mathbf{1}_r\}. \tag{5}$$

In other words, we check whether each row in $\mathbf{X}$ evaluates to $\mathbf{1}_r$.[8]

5.1.3. Evaluating multiple queries. In our setting, the set of queries Q contain queries that can differ in the number of sub-predicates (i.e., columns that are being asked about). For instance, using the above example data domain, one query may ask about the column SEX while another may ask about both SEX and RACE. To handle such cases, given some set of queries Q, we let r_{max} be the maximum number of sub-predicates for queries in Q. Then, in cases where some query predicate is comprised of $r < r_{\text{max}}$ sub-predicates, we can pad its matrix representation $\mathbf{Q}$ with rows corresponding to dummy sub-predicates $\mathbf{q}_{\text{pad}} = \mathbf{1}_d$. In this way, Equation 5 still holds (with $\mathbf{1}_r$ being replaced by $\mathbf{1}_{r_{\text{max}}}$).

Given that now the matrix representation for all queries in Q have the same shape, we can represent Q as a single 3-dimensional tensor $\mathbf{Q}^{(n)} \in \{0,1\}^{n \times r_{\text{max}} \times d}$. Then, we can calculate the statistics for all queries in Q by evaluating the product $\mathbf{Z} = \mathbf{Q}^{(n)}\mathbf{X}^T \in \{0,1\}^{n \times r_{\text{max}} \times N}$, where the i-th query answer is

$$Q(X)_i = \sum_{k=1}^{N} \mathbb{I}\{\mathbf{Z}[i,:,k] = \mathbf{1}_{r_{\text{max}}}\}. \tag{6}$$

Note that we can interpret the vector $\mathbf{Z}[i,:,k]$ as a boolean vector that checks whether record k satisfies each of the r_{max} sub-predicates for query i. For ease of notation, we will assume going forward that r refers to r_{max}.

5.2. Generating Synthetic Data Using Aggregate Statistics (Problem 1)

We first describe how we set up the integer programming optimization problem for Problem 1—namely, how we represent the constraint $Q(D) = Q(D')$. Using the notation in Section 5.1, we wish to find some synthetic dataset X (whose one-hot representation we denote as $\mathbf{X}$) such that $Q(D) = Q(X)$.

As suggested above, we first evaluate whether each record $\mathbf{X}_k$ satisfies all r sub-predicates for each query q_i (i.e., $\mathbf{Z}[i,:,k] = \mathbf{1}_r$). To do so, we want to add a helper binary variable $\mathbf{W} \in \{0,1\}^{n \times N}$ such that

$$\mathbf{W}[i,k] = \mathbb{I}\{\mathbf{Z}[i,:,k] = \mathbf{1}_r\}.$$

To enforce this relationship, we add the following constraints,

$$\mathbf{W}[i,k] \leq \mathbf{Z}[i,j,k], \quad \forall j \in \{1,2,\ldots,r\} \tag{7}$$

$$\mathbf{W}[i,k] \geq \sum_{j=1}^{r} \mathbf{Z}[i,j,k] - (r-1), \tag{8}$$

so that $\mathbf{W}[i,k]$ evaluates query q_i for record X_k. Then, we add the constraints

$$Q(D)_i = \sum_{k=1}^{N} \mathbf{W}[i,k], \quad \forall i \in \{1,2,\ldots,m\} \tag{9}$$

to ensure that the aggregate count corresponding to query i on the private dataset D match that on $\mathbf{X}$.

Explanation. Suppose $\mathbf{W}[i,k] = 1$. Given that $\mathbf{Z}$ is binary, Equation 7 is true if and only if $\mathbf{Z}[i,j,k] = 1$ for all j, thereby giving us $\mathbb{I}\{\mathbf{Z}[i,:,k] = \mathbf{1}_r\} = 1$. Moreover, Equation 8 is not violated since we have that

$$\begin{aligned} 1 &\geq \sum_{j=1}^{r} \mathbf{Z}[i,j,k] - (r-1) \\ &\geq r - (r-1) \quad (\mathbf{Z} \text{ is binary}) \\ &= 1 \end{aligned}$$

Similarly, if $\mathbf{W}[i,k] = 0$, then by Equation 8,

$$\begin{aligned} 0 &\geq \sum_{j=1}^{r} \mathbf{Z}[i,j,k] - (r-1) \\ \implies r - 1 &\geq \sum_{j=1}^{r} \mathbf{Z}[i,j,k], \end{aligned}$$

which, because $\mathbf{Z}$ is binary, can hold if and only if there exists some j such that $\mathbf{Z}[i,j,k] = 0$, meaning that $\mathbb{I}\{\mathbf{Z}[i,:,k] = \mathbf{1}_r\} = 0$. In this case, Equation 7 is not violated since $0 \leq \mathbf{Z}[i,j,k]$ (again, because $\mathbf{Z}$ can only take on the values 0 and 1).

5.3. Verifying Claims (Problem 3)

We now discuss the constraints for Problem 3: verifying whether some claim must be correct in the private dataset D according to the released statistics $Q(D)$.

In this setting, we have a claim $R(a,m)$ that is composed of attributes and claimed multiplicity of that record, m. Suppose attributes a are defined over some subset of columns indexed by the set A (i.e, the columns $\{c_j\}_{j \in A}$). Then we can define attributes a as a vector $\mathbf{a} = (\mathbf{a}^{(1)} \ldots \mathbf{a}^{(k)})$, where $\mathbf{a}^{(j)}$ is a one hot encoded representation of the attribute for column j if all zeros otherwise.

In order to check whether $R(\mathbf{a},m)$ is verifiable according to $Q(D)$, we stipulate in the optimization problem that the number of times $\mathbf{a}$ appears in $\mathbf{X}$ cannot equal m. If a feasible solution does not exist, then we can conclude that $R(\mathbf{a}',m)$ must be correct.

5.3.1. Constraints (part 1). We first define a constant $M \gg N$ (used for ensuring other constraints are held) and let A^{oh} correspond to the list of indices that columns $\{c_j\}_{j \in A}$ correspond to in $\mathbf{a}$.

Next, let us introduce the binary variable $\mathbf{T} \in \{0,1\}^{N \times d}$, where

$$\mathbf{T}_{ij} = \begin{cases} \mathbf{1}\{\mathbf{X_{ij}} = \mathbf{a_j}\}, & \text{if } j \in A_{oh} \\ 1, & \text{otherwise.} \end{cases}$$

8. We note that a simpler alternative to checking $(\mathbf{Q}\mathbf{X}^T)_j = \mathbf{1}_r$ is to check whether the row product is equal to 1 (i.e. $\prod_k (\mathbf{Q}\mathbf{X}^T)_{jk} = 1$). However, our integer programming solver (Gurobi) does not support this operation.

In other words, it indicates whether each column in $\mathcal{X}$ matches the corresponding column value in $\mathbf{a}$. In addition, we introduce $\mathbf{U} \in \{0,1\}^{N \times d}$, which is a helper variable used to set $\mathbf{T}$ properly in the constraints.

Now, we add constraints with $\mathbf{T}$ and $\mathbf{U}$. Let $v = |A^{(oh)}|$ be the number of (one-hot) indices we need to check. For each index $j \in A^{(oh)}$, we add the constraints,

$$\mathbf{X}_{i,j} - v \leq M(1 - \mathbf{T}_{i,j}) \tag{10}$$
$$v - \mathbf{X}_{i,j} \leq M(1 - \mathbf{T}_{i,j}) \tag{11}$$
$$\mathbf{X}_{i,j} - v \geq 1 - M\mathbf{T}_{i,j} - M(1 - \mathbf{T}_{i,j})\mathbf{U}_{i,j} \tag{12}$$
$$v - \mathbf{X}_{i,j} \geq 1 - M\mathbf{T}_{i,j} - M(1 - \mathbf{T}_{i,j})(1 - \mathbf{U}_{i,j}). \tag{13}$$

Explanation. Consider the case when $\mathbf{T}_{i,j} = 1$. From constraints 10 and 11, we have:

$$\mathbf{X}_{i,j} \leq v \quad \text{and} \quad \mathbf{X}_{i,j} \geq v \quad \Rightarrow \quad \mathbf{X}_{i,j} = v.$$

Thus, the indicator for the feature value in the one-hot encoding of the candidate is equal to its corresponding value in row i of $\mathbf{X}$.

From constraints 12 and 13, we also have:

$$\mathbf{X}_{i,j} - v \geq 1 - M \quad \text{and} \quad \mathbf{X}_{i,j} - v \leq M - 1.$$

Given that M is a large constant and that v and $\mathbf{X}_{i,j}$ are both in the domain $\{0,1\}$, these constraints are also met.

Now consider the case when $\mathbf{T}_{i,j} = 0$. From constraints 10 and 11, we have:

$$\mathbf{X}_{i,j} - v \leq M \quad \text{and} \quad \mathbf{X}_{i,j} - v \geq -M$$

Given that M is large, these constraints are always satisfied.

From constraints 12 and 13, we obtain:

$$\mathbf{X}_{i,j} - v \geq 1 - M\mathbf{U}_{i,j},$$
$$v - \mathbf{X}_{i,j} \geq 1 - M(1 - \mathbf{U}_{i,j}).$$

Case 1: $\mathbf{U}_{i,j} = 0$

$$\mathbf{X}_{i,j} - v \geq 1,$$
$$v - \mathbf{X}_{i,j} \geq 1 - M \quad \Rightarrow \quad 1 \leq \mathbf{X}_{i,j} - v \leq M - 1,$$

which enforces $\mathbf{X}_{i,j} \neq v$ (since $\mathbf{X}_{i,j} - v \neq 0$).

Case 2: $\mathbf{U}_{i,j} = 1$

$$\mathbf{X}_{i,j} - v \geq 1 - M,$$
$$v - \mathbf{X}_{i,j} \geq 1 \quad \Rightarrow \quad 1 \leq v - \mathbf{X}_{i,j} \leq M - 1,$$

which also enforces $\mathbf{X}_{i,j} \neq v$.

5.3.2. Constraints (part 2). Next, we add the binary variable $\mathbf{S} \in \{0,1\}^N$, which is an indicator that checks whether each row $\mathbf{X}_i$ matches on the attributes $\mathbf{a}$. If an entire row $\mathbf{X}_i$ matches the attributes, then the entire corresponding row of $\mathbf{T}_i$ should be equal to 1. This can be enforced with the following constraints:

$$\mathbf{S}_i \leq \mathbf{T}_{i,j} \quad \forall j \in A^{(oh)} \tag{14}$$
$$\mathbf{S}_i \geq \sum_{j \in A^{(oh)}} \mathbf{T}_{i,j} - (v - 1) \tag{15}$$

Explanation. When $\mathbf{S}_i = 1$, all values of $\mathbf{T}_i$ must be equal to 1 from constraint 14. When $\mathbf{S}_r = 0$, we have that at least one of value in row r of $\mathbf{T}$ must not be equal to 1 from constraint 15.

5.3.3. Constraints (part 3). With $\mathbf{S}$ indicating which rows $\mathbf{X}$ match attributes $\mathbf{a}$, we now check whether the claimed multiplicity m is correct by summing $\mathbf{S}$ and checking if a there exists some dataset $\mathbf{X}$ s.t. $\sum_{i=0}^{N-1} \mathbf{S}_i \neq m$. If the solver is unable to find a solution $\mathbf{X}$ under these constraints, then we conclude that dataset matching the statistics $Q(D)$ cannot exist without having exactly m rows that match attributes $\mathbf{a}$.[9]

Let Y be a scalar binary helper variable. Then we add the constraints,

$$\sum_{i=0}^{N-1} \mathbf{S}_i - m \leq MY - 1 \tag{16}$$
$$\sum_{i=0}^{N-1} \mathbf{S}_i - m \geq 1 - M(Y - 1) \tag{17}$$

Explanation. Suppose $Y = 0$. Then we have that

$$\sum_{i=0}^{N-1} \mathbf{S}_i - m \leq -1 \quad \text{and} \quad \sum_{i=0}^{N-1} \mathbf{S}_i - m \geq 1 - M,$$

which implies that

$$1 \leq m - \sum_{i=0}^{N-1} \mathbf{S}_i \leq M - 1.$$

Thus, we enforce that $m \neq \sum_{i=0}^{N-1} \mathbf{S}_i$.

Similarly, suppose $Y = 1$. Then

$$\sum_{i=0}^{N-1} \mathbf{S}_i - m \leq M - 1 \quad \text{and} \quad \sum_{i=0}^{N-1} \mathbf{S}_i - m \geq 1,$$

which implies that

$$1 \leq \sum_{i=0}^{N-1} \mathbf{S}_i - m \leq M - 1$$

Thus, we again enforce that $m \neq \sum_{i=0}^{N-1} \mathbf{S}_i$.

5.4. Additional implementation details

5.4.1. Generating Unique Datasets. As stated previously, we set the integer programming solver to output up to K solutions that we then use to generate claims. However, in the one-hot encoded representation of datasets, two datasets with the same set of records that are ordered differently will be considered two unique solutions. To encourage unique

9. While it is not the focus of our work, we would like to point out that a similar integer programming problem can be set up to confirm that a candidate at multiplicity m **cannot** exist by replacing constraints 16 and 17 so that they instead ensure $\sum_{i=0}^{N-1} S_i = m$. If the solver cannot find a solution where exactly m rows match $\mathbf{a}$, then we conclude that candidate *cannot* exist at that multiplicity in the dataset.

solutions, we use a (fixed) vector $\mathbf{h} \in \mathbb{N}^d$ of randomly generated integers as a hash function, where the hash value for any one-hot encoded record $\mathbf{x}$ is $\mathbf{h}^T\mathbf{x}$. Then, for every $i \in \{1, 2, \ldots, N-1\}$, we add the constraint,

$$\mathbf{h}^T\mathbf{X}_i \geq \mathbf{h}^T\mathbf{X}_{i-1} \tag{18}$$

so that any solution $\mathcal{X}$ outputted must have its records ordered by their hash values. While this approach is imperfect because, theoretically, different records in $\mathcal{X}$ may map to the same hash value, we found it to be, in practice, a simple and computationally-efficient approach to filtering out duplicate solutions.

5.4.2. Paring down candidate claims. Suppose our goal is to find all reconstruction claims R that *must exist* in D according to $Q(D)$. Let us denote $U(D)$ as the set of all claims that are correct with respect to D. Then any claim R is correct if $R \in U(D)$.

Using this notation, our goal of verifying claims to find all claims R such that $\forall D' \in \mathcal{X}^N$ s.t. $Q(D') = Q(D)$, $R \in U(D')$. In other words, the only way we can be absolutely certain that some claim R is correct according to $Q(D)$ is for it be correct for all datasets D' where $Q(D') = Q(D)$.

To generate candidate claims, one can simply generate a single synthetic dataset D' by solving Problem 1 and taking all unique claims $R(a, m)$ consistent with D' (i.e., $\textsc{Count}(a, D') = m$). Furthermore, to narrow down the set of candidates to check, one can generate many synthetic datasets X_i and take their intersection $\bigcap_i^K U(X_i)$. If there exists some $R \in U(X_i)$ such that $R \notin \bigcap_{i-1}^K U(X_i)$, then R violates the above condition that $R \in U(D')$ for any D' that matches the released statistics $Q(D)$.

Finally, we note that adjusting K (i.e., the number of synthetic datasets to output in the **generate** step) allows one to trade-off computational resources between the two steps. Generating more synthetic datasets (i.e., decreasing the size of the intersection) will decrease the number of candidates that need to be verified.

6. Conclusion

In conclusion, our work introduces the problem of partial tabular data reconstruction and proposes an integer programming approach that reconstructs individual records with guaranteed correctness. Evaluating on the household-level microdata and tables from the U.S. Decennial Census, we demonstrate that one can still (partially) reconstruct individual households with certainty, even when many possible blocks may satisfy the published statistics. We note that one limitation of using integer programming in our approach is that evaluating on larger datasets (more rows or columns) or sets of statistics may induce computational costs far more demanding than those required for our experiments. Nevertheless, our experiments show that for releases like the decennial census, in which the average dataset (i.e., block) is relatively small, reconstruction is very much possible while being computationally inexpensive. Overall, we contend that our initial work on partial reconstruction represents just the tip of the iceberg in terms of communicating the privacy risks that come with releasing aggregate information. We hope that our work inspires future research to build upon such notions of partial reconstruction (e.g., extending our approach to other data domains or using our approach of singling out households as part of larger, more systematic study on the dangers of linkage attacks).

References

[1] 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*, vol. 10, no. 1, pp. 119–144, 2023.

[2] J. M. Abowd, T. Adams, R. Ashmead, D. Darais, S. Dey, S. L. Garfinkel, N. Goldschlag, D. Kifer, P. Leclerc, E. Lew *et al.*, "The 2010 census confidentiality protections failed, here's how and why," National Bureau of Economic Research, Tech. Rep., 2023.

[3] A. Cohen and K. Nissim, "Towards formalizing the gdpr's notion of singling out," *Proceedings of the National Academy of Sciences*, vol. 117, no. 15, pp. 8344–8352, 2020.

[4] C. Dwork, K. Greenewald, and M. Raghavan, "Synthetic census data generation via multidimensional multiset sum," *arXiv preprint arXiv:2404.10095*, 2024.

[5] R. Steed, D. Qing, and Z. S. Wu, "Quantifying privacy risks of public statistics to residents of subsidized housing," *arXiv preprint arXiv:2407.04776*, 2024.

[6] L. Sweeney, "Weaving technology and policy together to maintain confidentiality," *The Journal of Law, Medicine & Ethics*, vol. 25, no. 2-3, pp. 98–110, 1997.

[7] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *2008 IEEE Symposium on Security and Privacy (sp 2008)*. IEEE, 2008, pp. 111–125.

[8] A. Flaxman and O. Keyes, "The risk of linked census data to transgender youth: A simulation study," *Journal of Privacy and Confidentiality*, vol. 15, no. 1, 2025.

[9] I. Dinur and K. Nissim, "Revealing information while preserving privacy," in *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2003, pp. 202–210.

[10] C. Dwork, F. McSherry, and K. Talwar, "The price of privacy and the limits of lp decoding," in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, 2007, pp. 85–94.

[11] S. P. Kasiviswanathan, M. Rudelson, A. Smith, and J. Ullman, "The price of privately releasing contingency tables and the spectra of random matrices with correlated rows," in *Proceedings of the forty-second ACM symposium on Theory of computing*, 2010, pp. 775–784.

[12] S. P. Kasiviswanathan, M. Rudelson, and A. Smith, "The power of linear reconstruction attacks," in *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2013, pp. 1415–1433.

[13] C. Dwork, A. Smith, T. Steinke, and J. Ullman, "Exposed! a survey of attacks on private data," *Annual Review of Statistics and Its Application*, vol. 4, no. 1, pp. 61–84, 2017.

[14] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*. Springer, 2006, pp. 265–284.

[15] T. Dick, C. Dwork, M. Kearns, T. Liu, A. Roth, G. Vietri, and Z. S. Wu, "Confidence-ranked reconstruction of census microdata from published statistics," *Proceedings of the National Academy of Sciences*, vol. 120, no. 8, p. e2218605120, 2023.

[16] S. Garfinkel, J. M. Abowd, and C. Martindale, "Understanding database reconstruction attacks on public data," *Communications of the ACM*, vol. 62, no. 3, pp. 46–53, 2019.

[17] J. M. Abowd, "The U.S. census bureau adopts differential privacy," in *ACM International Conference on Knowledge Discovery & Data Mining*, 2018, p. 2867.

Appendix

1. Additional experimental details

Dataset. We list and describe the 10 columns described by the block-level tables below.

Tenure (TEN): One of 4 tenancy statuses: owned with mortgage, owned free and clear, rented, or occupied without payment of rent
Vacancy status (VACS): Not vacant, or one of 7 vacancy statuses: for rent, rented but not occupied, for sale, sold and not occupied, for seasonal or occasional use, for migrant workers, or other.
Household size (HHSIZE): Size of household: 1, 2, 3, 4, 5, 6, or 7 or more
Household type (HHT): One of 7 types: married couple household, other family household (with a male/female householder), nonfamily household (with a male/female householder, living alone/not living alone).
Household type; detailed (HHT2): One of 12 types: married couple (with/without children < 18), cohabiting couple (with/without children < 18), no spouse/partner present (male/female householder, with own children < 18/with relatives and without own children < 18/only nonrelatives present/living alone)
Hispanic householder status (THHSPAN): Whether or not the householder is of Hispanic origin,
Householder age (THHLDRAGE): Age of the householder in one of 7 age buckets: 15-24, 24-35, ..., 75-84, or 85 years and older.
Householder race (THHRACE): Race of the householder in one of 7 categories: White alone, Black alone, American Indian or Alaskan Native alone, Asian alone, Native Hawaiian or Pacific Islander alone, some other race alone, or two or more races.
Presence of people under 18 years in household (TP18): Whether or not one or more people younger than 18 are in the household
Presence of people over 65 years in household (TP65): Whether or not one or more people 65 years and over are in the household

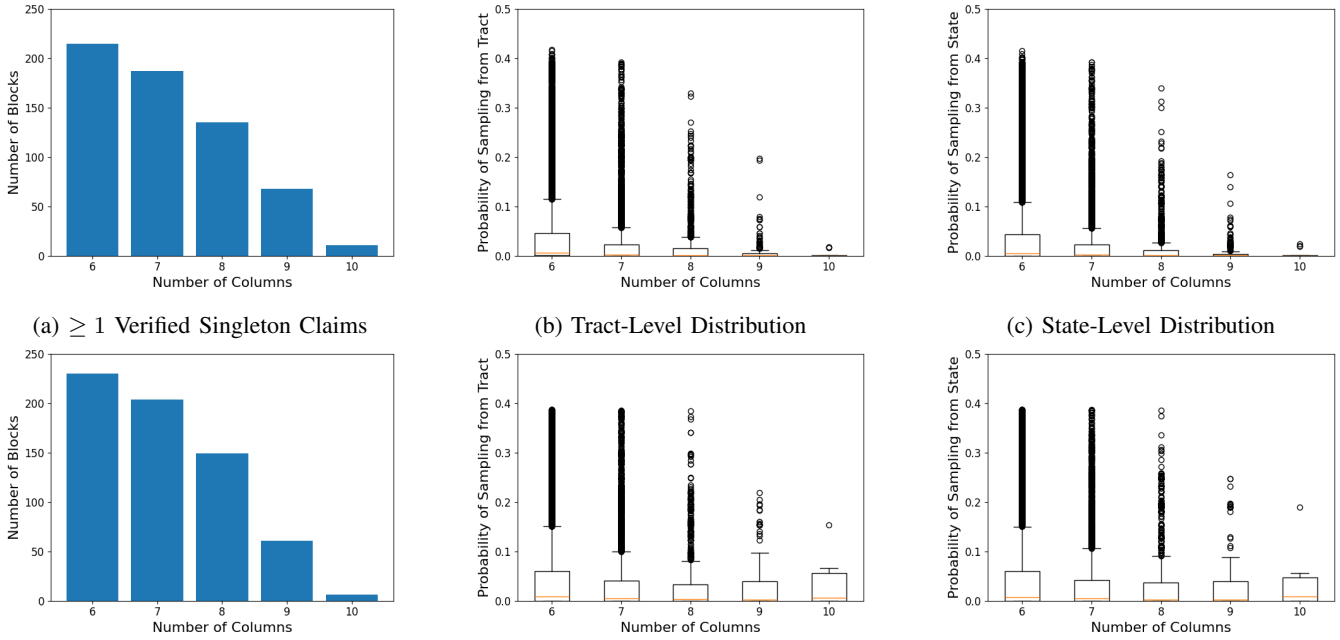
Statistics. We list the household-level Summary File 1 tables names below, along with the descriptors given by the Census Bureau.

P16: Household type
P16 A-G: Household type (iterated by race)
P16 H: Household type for households with a householder who is Hispanic or Latino
P16 I-O: Household type for households with a householder who is not Hispanic or Latino (iterated by race)
P16 P-V: Household type for households with a householder who is Hispanic or Latino (iterated by race)
P19: Households by presence of people 65 years and over, household size, and household type
P20: Households by type and presence of own children under 18 years
P21: Households by presence of people under 18 years
H1: Housing units (total count)
H3: Occupancy status
H4: Tenure
H4 A-G: Tenure (iterated by race)
H4 H: Tenure of housing units with a householder who is Hispanic or Latino
H4 I-O: Tenure of housing units with a householder who is not Hispanic or Latino (iterated by race)
H4 P-V: Tenure of housing units with a householder who is Hispanic or Latino (iterated by race)
H5: Vacancy status of vacant housing units
H6: Race of householder
H7: Hispanic or Latino origin of householder by race of householder
H9: Household size
H10: Tenure by race of householder
H11: Tenure by Hispanic or Latino origin of Householder
H12: Tenure by household size
H12 A-G: Tenure by household size (iterated by race)
H12 H: Tenure by household size of households with a householder who is Hispanic or Latino
H12 I: Tenure by household size of households with a householder who is White only and not Hispanic or Latino
H13: Tenure by age of householder
H13 A-G: Tenure by age of householder (iterated by race)
H13 H: Tenure by age of householder for housing units with a householder who is Hispanic or Latino
H13 I: Tenure by age of householder for housing units with a householder who is White alone and not Hispanic or Latino
H14: Tenure by household type by age of householder
H15: Tenure by presence of people under 18 years, excluding householders, spouses, and unmarried partners

IP Solver. We use the Gurobi Optimizer to solve our integer programming optimization problems. We specify parameters "feasibility tolerance" and "integer feasibility tolerance" to their smallest value of 10^{-9} to enforce constraints as tightly as possible. We set "pool search mode" to 2 in order to find as many solutions as possible, up to the some maximum number defined by "pool solutions", which is set to $K = 100$ when generating claims and to 1 when validating claims. We also set the "timeout" parameter to 3 minutes to control the total runtime of our experiments for the validation step. In cases where Gurobi times out, we mark that claim as unverified.

2. Additional results for verifying *all* claims

In the main body of our work, we focus on "singling out" (singleton claims). However, we note that data reconstruction for multiplicity $m > 1$ can be equally interesting (or privacy



(a) ≥ 1 Verified Singleton Claims (b) Tract-Level Distribution (c) State-Level Distribution

(d) ≥ 1 Verified Singleton Claims (e) Tract-Level Distribution (f) State-Level Distribution

Figure 5: We present presents results for **all** verified claims collected from experiments on 5 blocks selected from each state (250 total on each row). **Top row:** 5 blocks whose size are equal the median block size of the respective state are selected. **Bottom row:** 5 blocks whose size are equal the median block size of the country (i.e., 10 households). **a & d:** The number of blocks (out of 50) for which we can reconstruct at least one singleton claim about k columns (x-axis). **b & e and c & f:**: Box and whisker plots of the probabilities that each verified claim would also be true in a set of N households randomly sampled from the (b, e) tract and (c, f) state-level distributions. The orange line within each box indicates the median probability. The ends of the box indicate the first and third quartiles, and the whiskers end at the furthest point within 1.5 times the interquartile range. All points beyond the whiskers are outliers. Lower probabilities denote more "surprising" claims.

TABLE 7: For each number of columns, we report total and average number of households that are represented among the **all** verified claims. We tabulate the verified claims over 500 total blocks: (top two rows) 5 blocks from each state whose size is equal to the median block size in the state and (bottom two rows) 5 blocks from each state whose size is equal the country median block size (i.e., 10 households). n is the total and average number of households over all 250 blocks.

			# of households identified by verified claims w/ k columns				
		# households	k=6	7	8	9	10
State Median	Total	2500	1696	1220	642	257	53
	Avg. per block	10.00	6.78	4.88	2.57	1.03	0.21
Country Median	Total	2430	1809	1262	691	216	35
	Avg. per block	9.72	7.80	5.44	2.98	0.93	0.15

violating). Thus, we present in Figure 5 and Table 7 results for all claims (not just singletons). In general, we make conclusions similar to those in Section 3. The baseline probabilities of most claims are still extremely small, and as expected, more claims (about more households) can be made. For example, Table 7 shows that now, approximately a quarter of households are identified by claims of $k = 8$ columns and 70% by claims of $k = 6$ columns.