Differentially Private Hierarchical Count-of-Counts Histograms

Yu-Hsuan Kuo

Computer Science & Engineering
Penn State University

August 30 2018

Collaborators

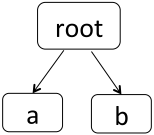
- Cho-Chun Chiu, Daniel Kifer, Michael Hay, Ashwin Machanavajjhala

Outline

1. Introduction: hierarchical count-of-counts histograms
2. Non-hierarchical count-of-counts histograms publishing
3. Hierarchical count-of-counts histograms publishing
4. Experimental results

Scenario

- Table **Persons**(person_name, group_id, location)
- A hierarchy Γ on location associated with each group



name	g_id	loc.
Alice	1	a
Bob	1	a
Carol	1	a
Dave	1	a
Eve	2	b
Frank	2	b
Judy	3	a
Nick	4	b

Queries: In the United States,

- How many groups have size 1 ?
- How many groups have size 2 ?

In New York,

- How many groups have size 1 ?
- How many groups have size 2 ?

Application:

1. group = a taxi, data item = a pick up, size = # of pickup
2. group = a census block, data item = a person of a specific race, size = # people of a specific race

Convenient Views of the Dataset

- `A=SELECT groupid, COUNT(*) AS size FROM Persons GROUPBY groupid`
- `H=SELECT size, COUNT(*) FROM A GROUPBY size`

SQL query resulting table `A`:

g_id	size	loc.
1	4	a
2	2	b
3	1	a
4	1	b

- **count-of-counts histogram (coco)** H is
  $H^{root} = [2, 1, 0, 1]$
  $H^{a} = [1, 0, 0, 1]$
- **unattributed histogram** [HRMS10] H_g is
  $H_g^{root} = [1, 1, 2, 4]$
  $H_g^{a} = [1, 4]$
- **cumulative count-of-counts histogram** H_c is
  $H_c^{root} = [2, 3, 3, 4]$
  $H_c^{a} = [1, 1, 1, 2]$

Protect Privacy

Definition (Differential Privacy [DMNS06])

A mechanism M satisfies ϵ-differential privacy if, for any pair of databases D_1, D_2 that differ by the presence or absence of one record in the Persons table, and for any possible set S of outputs of M, the following is true:

$$P(M(D_1) \in S) \leq e^{\epsilon} P(M(D_2) \in S)$$

Geometric Mechanism

Definition (Sensitivity)

Given a query q (which outputs a vector), the global sensitivity of q, denoted by $\Delta(q)$ is defined as:

$$\Delta(q) = \max_{D_1, D_2} ||q(D_1) - q(D_2)||_1,$$

where databases D_1, D_2 contain the public Hierarchy and Groups tables, and differ by the presence or absence of one record in the Persons table.

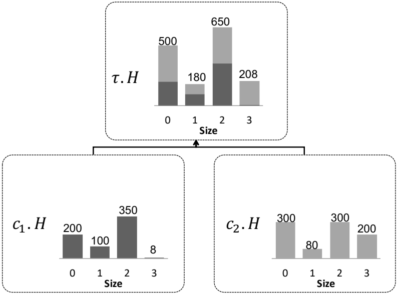
Definition (Geometric Mechanism [GRS09])

Given a database D, a query q that outputs a vector, a privacy loss budget ϵ, the global sensitivity $\Delta(q)$, the geometric mechanism adds independent noise to each component of $q(D)$ using distribution:
$P(X = k) = \frac{1-e^{-\epsilon}}{1+e^{-\epsilon}} e^{-\epsilon|k|/\Delta(q)}$ (for $k = 0, \pm 1, \pm 2$, etc.). This distribution is known as the double-geometric with scale $\Delta(q)/\epsilon$.

Problem Definition

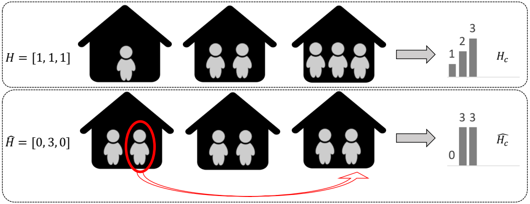
For each node τ in hierarchy Γ, create differentially private estimate $\tau.\widehat{H}$ of count-of-counts histogram H such that

- $\tau.\widehat{H}$ is a count-of-counts histogram (its entries are nonnegative integers)
- The counts are accurate ($\tau.\widehat{H}$ and $\tau.H$ are close)
- $\tau.\widehat{H}$ matches publicly known total number of groups G in τ
- satisfy consistency: children histograms sum up to the parent.



Error Measure

- The **Earthmover's distance (emd)**: the minimum number of people that must be added or removed from groups in $\tau.H$ to get $\tau.\widehat{H}$.



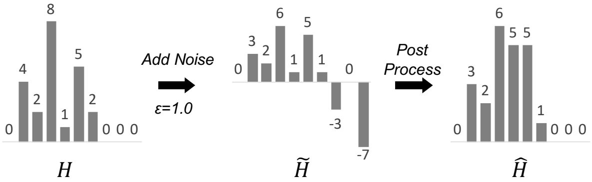
emd = $| H_c - \widehat{H}_c |_1 = | H_g - \widehat{H}_g |_1 = 2$

Lemma ([NLV07])

The earthmover's distance between H and $\widehat{H}$ can be computed as $||H_c - \widehat{H}_c||_1$, where H_c (resp., $\widehat{H}_c$) is the cumulative histogram of H (resp., $\widehat{H}$). It is the same as the L_1 norm in the H_g representation when the number of groups is fixed.

Outline

Naive Strategy



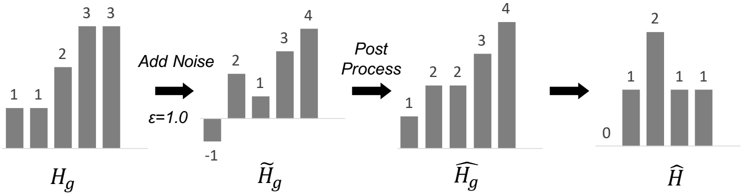
1. $\widetilde{H}$: Add independent double-geometric noise with scale $2/\epsilon$ to each element of coco histogram H
2. Post-process $\widetilde{H}$ with optimization problem:

$$\widehat{H} = \arg\min_{\widehat{H}} ||\widetilde{H} - \widehat{H}||_2^2$$

$$\text{s.t. } \widehat{H}[i] \geq 0 \text{ for all i} \quad \text{and } \sum_i \widehat{H}[i] = G$$

3. To get integers, we set $r = G - \sum_i \lfloor \widehat{H}[i] \rfloor$, round the cells with the r largest fractional parts up, and round the rest down.
4. Solver: quadratic program (e.g., Gurobi [GO16])

Unattributed Histogram [HRMS10] $\mathbf{H_g}$



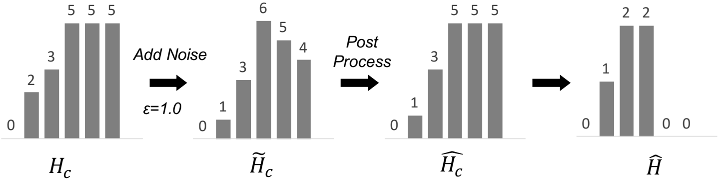
1. Convert coco histogram $H \Rightarrow$ unattributed histogram H_g
2. $\widetilde{H}_g$: Add independent double-geometric noise with scale $1/\epsilon$ to each element of H_g
3. Post-process with optimization problem with either $p = 1$ or $p = 2$:

$$\widehat{H}_g = \arg\min_{\widehat{H}_g} ||\widetilde{H}_g - \widehat{H}_g||_p^p$$

$$\text{s.t. } 0 \leq \widehat{H}_g[i] \leq \widehat{H}_g[i+1] \text{ for } i = 0, \ldots, G-1$$

4. Round each entry of $\widehat{H}_g$ to the nearest integer and convert it back to $\widehat{H}$
5. Solver: min-max algorithm [BB72], pool-adjacent violators (PAV) [BBBB, RW+68], Gurobi [GO16]

Cumulative Sum Histograms $\mathbf{H_c}$



1. Convert coco histogram $H \Rightarrow$ cumulative sum histogram H_c
2. $\widetilde{H}_c$: Add independent double-geometric noise with scale $1/\epsilon$ to each element of H_c
3. Post-process with optimization problem with either $p=1$ or $p=2$:

$$\widehat{H}_c = \arg\min_{\widehat{H}_c} ||\widehat{H}_c - \widetilde{H}_c||_p^p$$

$$\text{s.t. } 0 \leq \widehat{H}_c[i] \leq \widehat{H}_c[i+1] \text{ for } i = 0, \ldots, K$$

$$\text{and } \widehat{H}[K] = G$$

4. Round each entry of $\widehat{H_c}$ to the nearest integer and convert it back to $\widehat{H}$
5. Solver: min-max algorithm [BB72], pool-adjacent violators (PAV) [BBBB, RW$^+$68], Gurobi [GO16]

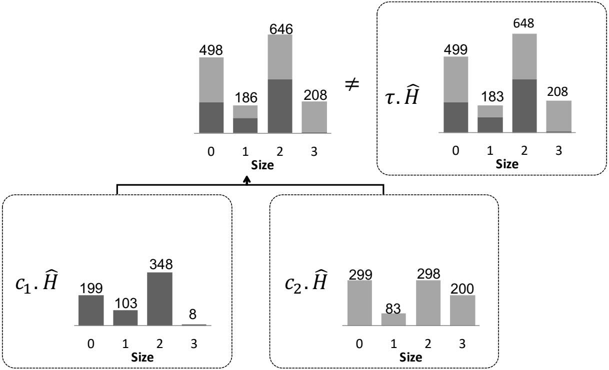
Methods Summary

- Naive approach had several orders of magnitude worse error than the unattributed histogram $\mathbf{H_g}$ and cumulative sum histogram $\mathbf{H_c}$ method
- For most datasets, $\mathbf{H_c}$ method generally performs better
- For sparse datasets, $\mathbf{H_g}$ method is better

Outline

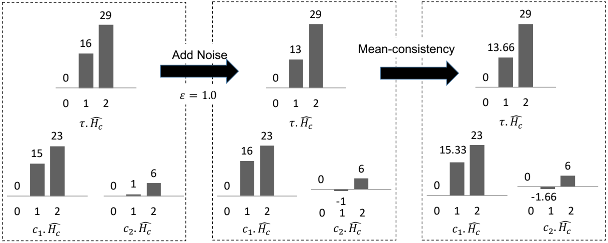
Non-hierarchical Methods Issue

- Estimate coco histograms at each node τ, c_1, c_2
- Drawback: parent $\tau.\widehat{H}$ does not equal to the sum of children $(c_1.\widehat{H} + c_2.\widehat{H})$



Mean-Consistency Algorithm [HRMS10]

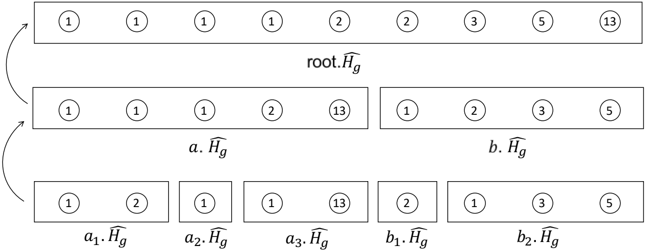
1. Take cumulative coco histograms H_c at every node
2. Add independent double-geometric noise with scale $1/\epsilon$ to each element of H_c
3. Post-process with mean-consistency algorithm



- Drawback: counts can be negative and fractional

Bottom-up Aggregation

1. Estimate coco histogram H only at the leaves
2. Aggregate them up the hierarchy



- Drawback: it introduces high error at non-leaf nodes (like in other hierarchical problems [HRMS10, QYL13])

Consistency Solution

- Our proposed solution:
  1. Converts estimated coco $\tau.\widehat{H} \Rightarrow$ the unattributed histogram $\tau.\widehat{H_g}$
  2. Find a 1-to-1 optimal matching between groups at the child nodes and groups at the parent node
  3. Merge those two estimates

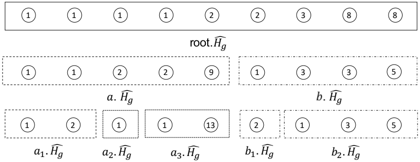


Figure: Before matching

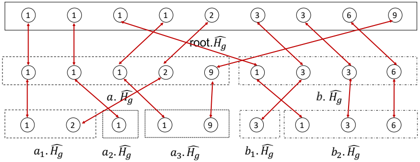
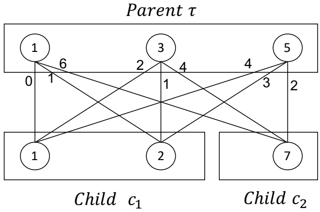


Figure: Consistency result

Optimal Matching Algorithm

- For each node τ and its children, we set up a bipartite weighted graph
- There are $\tau.G$ vertices on the top: $(\tau, 1), (\tau, 2), \ldots, (\tau, \tau.G)$. Each vertices on the bottom has the form (c, j), where c is a child of τ and j is an index into $c.\widehat{H}_g$.
- Edge between every vertex (τ, i) and (c, j) has weight $|\tau.\widehat{H}_g[i] - c.\widehat{H}_g[j]|$: measure the difference in estimated size



- Our desired matching is least cost weighted matching on this bipartite graph.
- Optimal algorithm: matching the smallest unmatched group in τ to the smallest unmatched group among any of its children.

Top-down Consistency

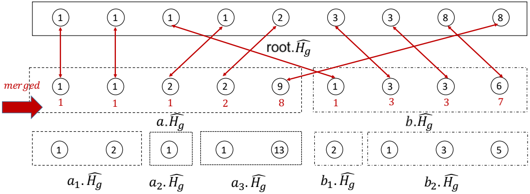


Figure: Level 0 and Level 1 consistency matching

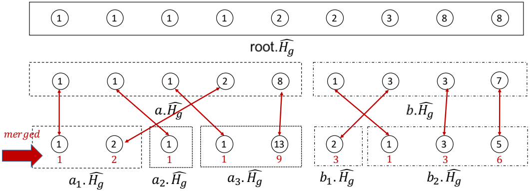


Figure: Level 1 and Level 2 consistency matching

1. Consistency matching at top level
2. Use new estimates for next level consistency
3. Use the new merged estimates at the leaves for back substitution to get unattributed histogram:

$$\widehat{H}_g^a = [1, 1, 1, 2, 9]$$
$$\widehat{H}_g^b = [1, 3, 3, 6]$$
$$\widehat{H}_g^{root} = [1, 1, 1, 1, 2, 3, 3, 6, 9]$$

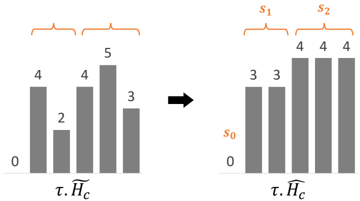
4. Convert consist unattributed histogram into count-of-counts histogram

Initial Variance Estimation

Recall: we convert $\tau.\widehat{H}$ into the unattributed histogram $\tau.\widehat{H}_g$.

For each i, we need an estimate of the variance of the i^{th} largest group $\tau.\widehat{H}_g[i]$, so that it can be used to merge two estimates during matching.

- Let S_i be the number of groups that were in the same partition as i in the solution



- Let ϵ be the privacy budget used in node τ in level ℓ of Γ

For the $\mathbf{H_g}$ method:

- Variance estimate for the i^{th} largest group: $\tau.V_g[i] = \frac{2}{|S_i|\epsilon^2}$

For the $\mathbf{H_c}$ method:

- Variance estimate of the i^{th} largest group: $\tau.V_g[i] = 4/(\epsilon^2 \times \text{number of estimated groups of size } \tau.\widehat{H}_g[i])$

Merge Estimates

Given a node τ, the matching algorithm assigns one group i in τ to one group j in some child of τ
$\Rightarrow$ for every group, two estimates of its size: $\tau.\widehat{H}_g[i]$ and $c.\widehat{H}_g[j]$ & estimates of variance $\tau.V_g[i]$ and $c.V_g[j]$.

- Optimal linear combination of the estimates [HRMS10]: weighted average

$$\left(\frac{\tau.\widehat{H}_g[i]}{\tau.V_g[i]} + \frac{c.\widehat{H}_g[j]}{c.V_g[j]}\right) \Big/ \left(\frac{1}{\tau.V_g[i]} + \frac{1}{c.V_g[j]}\right) \tag{1}$$

and the variance of this estimator is

$$\left(\frac{1}{\tau.V_g[i]} + \frac{1}{c.V_g[j]}\right)^{-1} \tag{2}$$

Outline

Experiments

Use 4 datasets:

- **Race distribution - White** (2010 Census data [Bur12]): For West Coast/State/County and a given race, for each j, how many Census blocks contain j people of that race?
- **Race distribution - Hawaiian** [Bur12]
- **Partially synthetic housing**: The number of individuals in each facility is important but this information was truncated past households of size 7 in the 2010 Decennial Census Summary File 1 [Bur12]. We add a heavy tail as would be expected from group quarters (e.g., dormitories, barracks, correctional facilities).
- **NYC taxi**: In 2013, how many taxis had j pickups in Manhattan/Town/Neighborhood?

Ruling out Naive Strategy

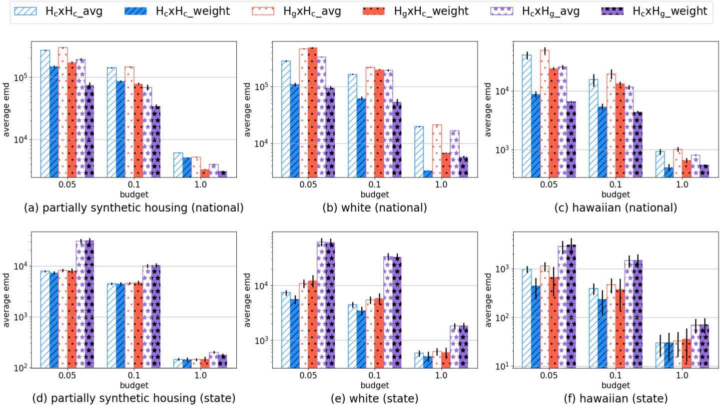
- Naive strategy's average error is in the billions

Table: Average error with $\epsilon = 1.0$ at top level

Method	Synthetic	White	Hawaiian	Taxi
Naive	4,462,728,374	4,809,679,734	4,027,891,692	208,977,518
$\mathbf{H_c}$	3,742.0	1,838.9	254.0	2,819.8
$\mathbf{H_g}$	2,219.6	6,115.3	516.2	11,227.6

Weighted average estimation comparison

- Two choices at each level: $\mathbf{H_c}$, $\mathbf{H_g}$
- Weighted average method consistently produces large reductions in error at the top level



(a) partially synthetic housing (national)
(b) white (national)
(c) hawaiian (national)
(d) partially synthetic housing (state)
(e) white (state)
(f) hawaiian (state)

Figure: Merging estimates using weighted average vs. normal average. x-axis: privacy budget per level.

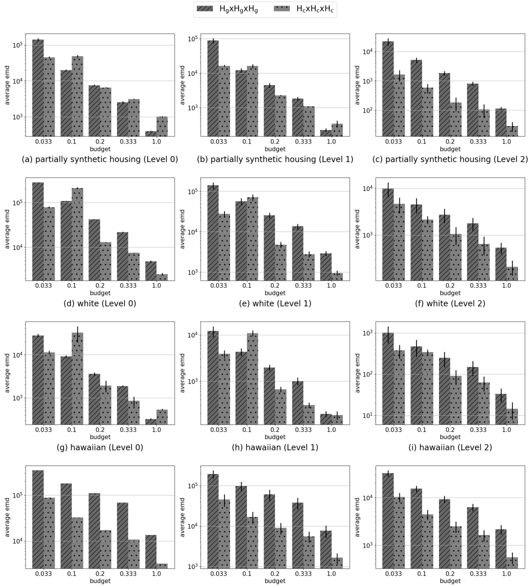
Comparison to Bottom-up Aggregation

- Allocate all privacy budget (total privacy budget of $\epsilon = 1.0$ in the table) to the leaves and set the coco histogram of a parent to be the sum of the histograms at the leaves.
- Very low error at the leaves but higher error everywhere else

	Part. Synth.	White	Hawaiian	Taxi
		Level 0		
BU	78,459.0	448,909.0	13,968.0	20,731.0
H_c	**32,480.0**	**17,000.0**	**1,381.0**	**10,547.0**
		Level 1		
BU	1,512.2	8,722.0	270.1	10,405.5
H_c	**1,000.3**	**1,511.8**	**117.7**	**5,431.5**
		Level 2		
BU	24.9	152.3	4.3	772.8
H_c	80.1	363.8	21.6	1,601.8

3-Level Hierarchy Results

- Two alternatives $H_g \times H_g \times H_g$ and $H_c \times H_c \times H_c$
- Data dependent performance: $\mathbf{H_c}$ performs better in dense region while $\mathbf{H_g}$ performs better in sparse region
- Figure: 3-level consistency at each level. x-axis: privacy budget per level



(a) partially synthetic housing (Level 0) (b) partially synthetic housing (Level 1) (c) partially synthetic housing (Level 2)

(d) white (Level 0) (e) white (Level 1) (f) white (Level 2)

(g) hawaiian (Level 0) (h) hawaiian (Level 1) (i) hawaiian (Level 2)

(j) taxi (Level 0) (k) taxi (Level 1) (l) taxi (Level 2)

Summary

- Introduced hierarchical count-of-counts problem, along with appropriate error metrics
- Proposed a differentially private solution that generates non-hierarchical and hierarchical version of count-of-counts histograms.
- $\mathbf{H_c}$ method generally performs better on dense dataset while datasets with more sparsity favor $\mathbf{H_g}$ method

Questions?

References I

- RE Barlow and HD Brunk.
  The isotonic regression problem and its dual.
  Journal of the American Statistical Association, pages 140–147, 1972.

- RE Barlow, DJ Bartholomew, JM Bremner, and HD Brunk.
  Statistical inference under order restrictions. 1972.

- U.S. Census Bureau.
  2010 census summary file 1, 2010 census of population and housing, technical documentation.
  https://www.census.gov/prod/cen2010/doc/sf1.pdf, 2012.

- Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith.
  Calibrating noise to sensitivity in private data analysis.
  In Proceedings of the Third Conference on Theory of Cryptography, pages 265–284, 2006.

References II

Inc. Gurobi Optimization.
Gurobi optimizer reference manual, 2016.

Arpita Ghosh, Tim Roughgarden, and Mukund Sundararajan.
Universally utility-maximizing privacy mechanisms.
In STOC, pages 1673–1693, 2009.

Michael Hay, Vibhor Rastogi, Gerome Miklau, and Dan Suciu.
Boosting the accuracy of differentially private histograms through consistency.
PVLDB, 3(1-2):1021–1032, 2010.

T. Li N. Li and S. Venkatasubramanian.
t-closeness: Privacy beyond k-anonymity and l-diversity.
In ICDE, pages 106–115, 2007.

References III

Wahbeh Qardaji, Weining Yang, and Ninghui Li.
Understanding hierarchical methods for differentially private histograms.
PVLDB, 6(14):1954–1965, 2013.

Tim Robertson, Paul Waltman, et al.
On estimating monotone parameters.
The Annals of Mathematical Statistics, pages 1030–1039, 1968.