

# Singling-out vs. Blending-in: Outlier Detection and Differential Privacy in Data

Yu-Hsuan Kuo

Computer Science & Engineering  
Penn State University

January 15 2019

- Outlier detection
- Outlier explanation
- Differentially private count-of-counts histograms
- Differentially private unattributed histograms
- Social science



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Motivation

- In large-scale sensor datasets, there could be a significant amount of outliers due to sensor malfunction or human operation faults.

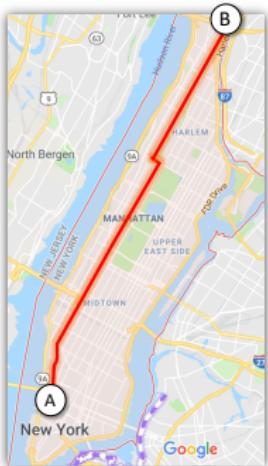


Figure: long moving distance but unreasonably low trip fare

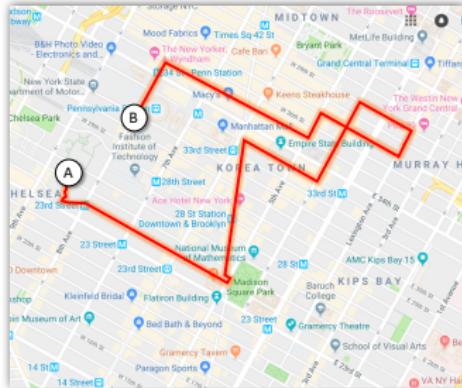


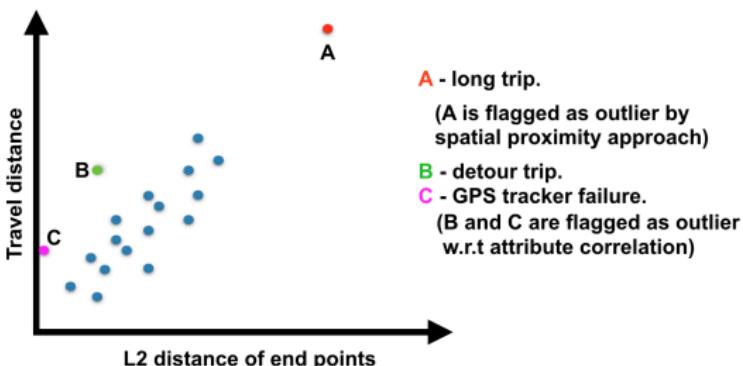
Figure: short L2 distance between pickup (A) and dropoff (B) but long trip distance

- Such outliers in the original datasets can break effective travel time estimation methods [WKKL16].



# Contextual Outlier [SWJR07]

- Typical outlier detection defines a sample as an outlier if it significantly deviates from other data samples.  $\Rightarrow$  not apply in our case.

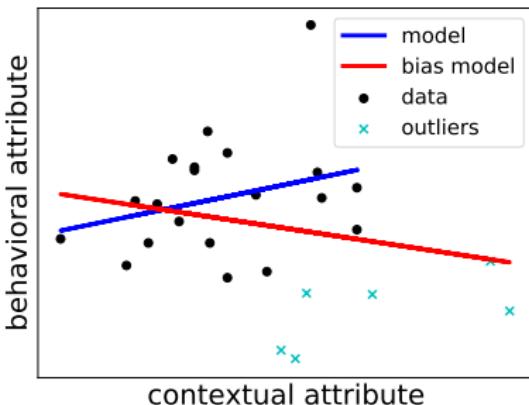


- We detect outliers based on empirical correlations of attributes. (e.g., trip time and trip distance)
- Contextual outlier detection: use the correlation between contextual attributes and behavioral attributes [SWJR07, HH15, LP16].
- Anomaly: attributes of a data sample significantly deviate from expected correlations.



## Related Work

- One problem with contextual outlier detection [SWJR07, HH15, LP16] is that outliers can bias a model learned from noisy data.



- Clean data is almost not available  $\Rightarrow$  contextual outlier detector trained on noisy data.
- Our solution: a robust regression model that explicitly considers outliers.



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- **Outlier Modeling**
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

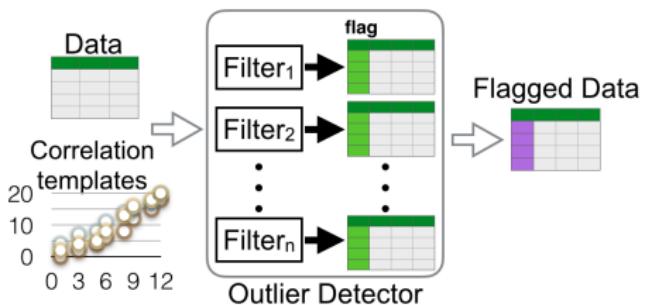
## 3 Contributions

## 4 Backup Slides



# Outlier Detector Overview

- Input: Data & Correlation templates  $(j, S)$  where  $j$  is behavior attribute and  $S$  is a set of contextual attributes
- Output: flagged suspicious records.
- A **filter**:
  - ➊ take correlation template  $(j, S)$  and learn, for each record  $\vec{z}_i$ , how to predict behavior attribute  $\vec{z}_i[j]$  from contextual attributes  $\vec{z}_i[s]$  for  $s \in S$ .
  - ➋ assign an outlier score  $t_i$  to every record.
  - ➌ provide an estimate for the total number of outliers.
- A record is marked as outlier if at least one filter marks it as an outlier.



# Mixture Model

- For a correlation  $(j, S)$ , let  $y_i$  be behavioral attribute value and  $\vec{x}_i$  be the vector of contextual attribute values in  $S$ .
- Learn a model that can predict  $y_i$  from the attributes  $\vec{x}_i$ .

$$y_i = \vec{w} \cdot \vec{x}_i + \epsilon_i$$

- Model the prediction error: a mixture of light-tailed distributions (for non-outliers) and heavy-tailed distributions (for outliers).
- Assume there is a probability  $p$  that a data point is an outlier  $\Rightarrow$  Noise distribution  $\epsilon_i$  for record  $i$ : with prob.  $1 - p$  it is a Gaussian, and with prob.  $p$  it is a Cauchy random variable.



# Parameters Learning

- EM algorithm [DLR77] to solve the likelihood function  $L$ . (see backup slides 68, 69)
- E step:
  - parameter  $\tau_i$  of Cauchy density
  - estimated probability that it is an outlier  $t_i$  (i.e. expected value of  $\chi_i$ )
  - scale parameter  $b$
- M step:
  - estimated fraction of outliers  $p$
  - the variance of non-outliers  $\sigma^2$
  - model coefficients  $\vec{w}$
- Outlier labeling: every filter model assigns to every record  $i$  a score  $t_i$ . It then labels a record an outlier if it has one of the top  $K$  values of  $t_i$ ; where  $K = \lfloor \sum_{i=1}^n t_i \rfloor \approx p \times$  total number of records  $n$ .



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- **Experimental Results**
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Experiment Setups

Datasets: NYC Taxi, Intel Lab Sensor, ElNino, Houses

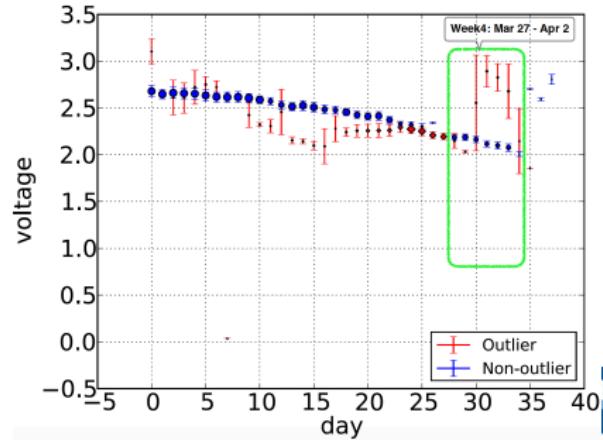
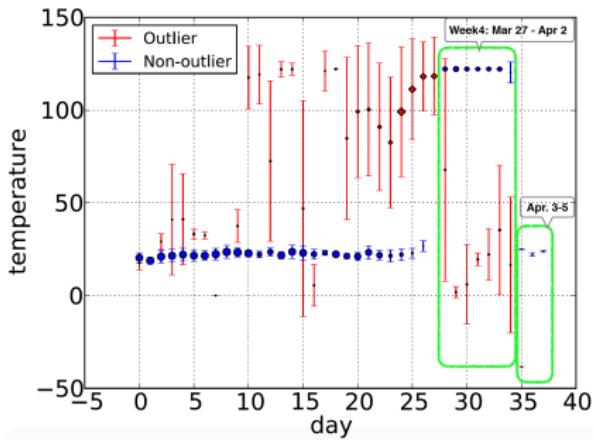
Baselines:

- **Density-based method.** A widely referenced density-based algorithm LOF [BKNS00] outlier mining.
- **Distance-based method.** A recent distance-based outlier detection algorithm with sampling [SB13].
- **OLS.** The linear regression with ordinary least square estimation.
- **GBT.** The gradient boosting tree regression model [Fri01].
- **CAD.** Conditional Anomaly Detection [SWJR07].
- **ROCOD.** Robust Contextual Outlier Detection [LP16].



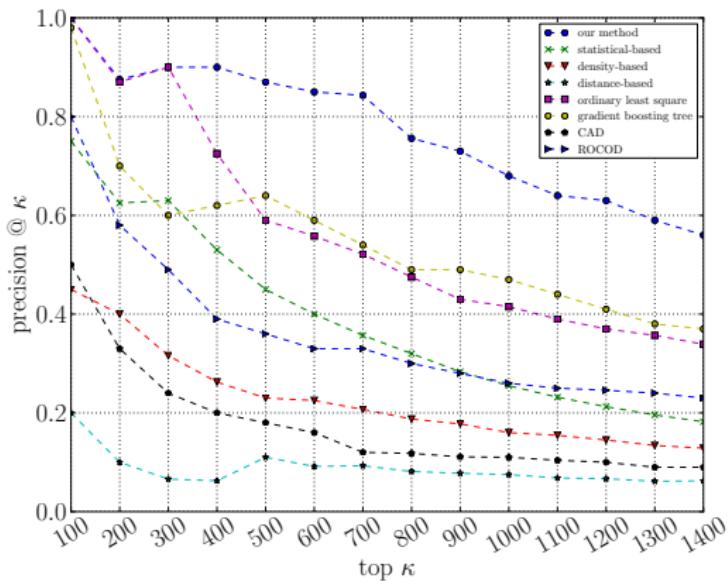
# Intel Sensor Data Results

- No ground truth  $\Rightarrow$  validate with findings in the Scorpion system [WM13], & case study.
- Observed a general sensor's malfunction pattern as it is unlikely to be real temperature in the lab.
- A decreasing trend in voltage for this batch of sensors.



# NYC Taxi Data Results

- We designed a human labeling system for experienced taxi riders to determine outlier trips.
- Evaluation metric: Precision  $@\kappa = \frac{\# \text{ trips whose rank} \leq \kappa \text{ and label} = \text{Outlier}}{\kappa}$



# Experiments on Synthetic Outlier Data

- We inject synthetic outliers into Elnino and Houses datasets.
- Perturbation scheme: inject  $q$  % of outliers into  $N$  data samples.
  - randomly select  $q \times N$  records  $\vec{z}_i = (\vec{x}_i, y_i)$  to be perturbed.
  - a random number from  $(0, \alpha)$  is added up to target attribute  $y_i$  as  $y_i'$ .
  - add new sample  $\vec{z}' = (\vec{x}_i, y_i')$  as outlier.
- Evaluation metric: the Area Under the Curve (AUC) of the Precision-Recall curve.



# Synthetic Outlier - Perturb Behavioral Attributes (Houses)

- Our outlier detector consistently performs the best when more outliers are involved.

Table: PR AUC w.r.t different fractions of synthetic outliers in behavioral attribute

method	Houses				
	q=0.01	q=0.03	q=0.05	q=0.1	q=0.15
Doc	<b>0.93</b>	<b>0.92</b>	<b>0.93</b>	<b>0.95</b>	<b>0.96</b>
ROCOD (non-linear)	0.50	0.49	0.50	0.49	0.50
CAD	0.58	0.67	0.68	0.72	0.75
OLS	0.92	0.91	0.92	0.91	0.91
GBT	<b>0.93</b>	0.91	0.92	0.91	0.91
distance-based	0.76	0.19	0.57	0.4	0.39
density-based	0.84	0.58	0.46	0.53	0.58



# Synthetic Outlier - Perturb Contextual Attributes (Houses)

- A small fraction of outliers in contextual attribute hurts the performance considerably for the other methods.
- Our method is robust and resistant to the fraction of outliers.

**Table:** PR AUC w.r.t different fractions of synthetic outliers in contextual attribute

method	Houses				
	q=0.005	q=0.01	q=0.03	q=0.05	q=0.07
Doc	<b>0.86</b>	<b>0.80</b>	<b>0.88</b>	<b>0.88</b>	<b>0.91</b>
ROCOD (non-linear)	0.03	0.01	0.02	0.04	0.05
CAD	0.51	0.54	0.56	0.61	0.63
OLS	0.84	0.75	0.71	0.59	0.50
GBT	0.04	0.04	0.08	0.11	0.15
distance-based	0.54	0.73	0.22	0.20	0.42
density-based	0.01	0.01	0.03	0.04	0.06



# Synthetic Outlier- Degree of Outlierness (Houses)

- As  $\alpha$  increases, larger magnitude of noise will have more chance to be added to the original value.
- Our performance increased as more extreme outliers are added.

Table: PR AUC w.r.t degree of outlierness  $\alpha$  in contextual attribute

method	Houses				
	$\alpha = 30$	$\alpha = 50$	$\alpha = 100$	$\alpha = 300$	$\alpha = 500$
Doc	<b>0.75</b>	<b>0.8</b>	<b>0.94</b>	<b>0.97</b>	<b>0.99</b>
ROCOD (non-linear)	0.01	0.01	0.01	0.01	0.01
CAD	0.37	0.54	0.58	0.74	0.85
OLS	0.72	0.75	0.87	0.86	0.83
GBT	0.04	0.04	0.03	0.02	0.01
distance-based	0.14	0.73	0.79	0.85	0.80
density-based	0.01	0.01	0.01	0.02	0.05



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Contributions

- We develop a system to detect outliers by correlations between measurements.
- It is a robust model as compared to the existing algorithms built on all the data records where their model parameters are skewed by outliers.
- We compare our approach against traditional outlier detectors, contextual outlier detectors and regression models. Our method outperformed competing methods and continues to perform well even in situations where other methods break down.



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

### • Background

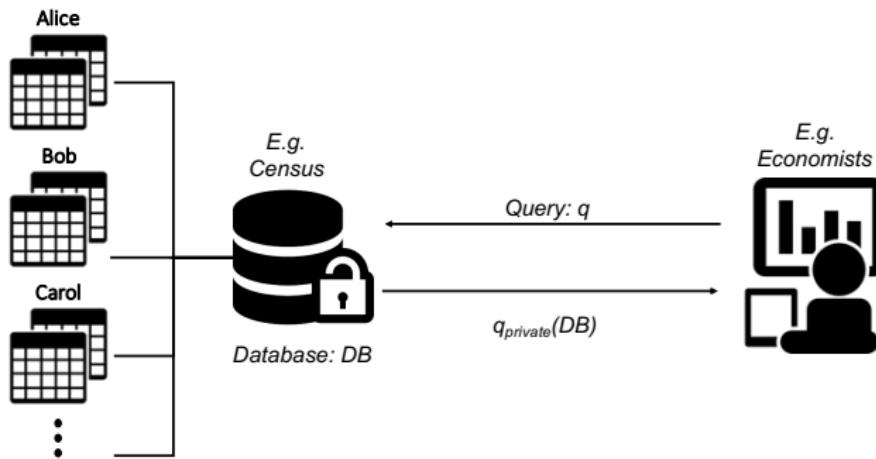
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# The Privacy



- The goal of differential privacy : analyzing aggregated personal data with guarantees of not disclosing individual records
- Utility for user:  $q_{private}(DB)$  should close to  $q(DB)$

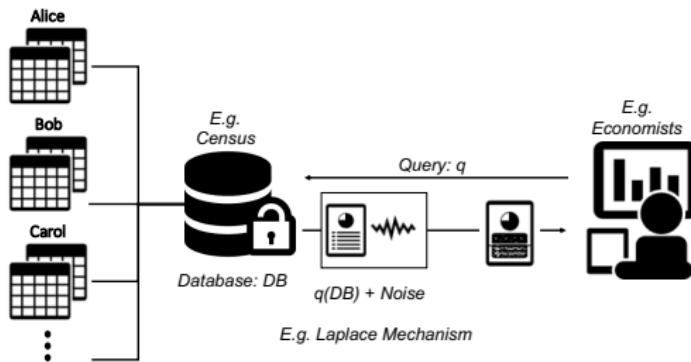


# Differential Privacy

## Definition (Differential Privacy [DMNS06])

A mechanism  $M$  satisfies  $\epsilon$ -differential privacy if, for any pair of databases  $D_1, D_2$  that differ in one tuple, and for any possible set  $S$  of outputs of  $M$ , the following holds:  $P(M(D_1) \in S) \leq e^\epsilon P(M(D_2) \in S)$

- Attacker should not be able to use output  $S$  to distinguish between any  $D_1$  and  $D_2$
- Smaller  $\epsilon$  implies more privacy so worse utility



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- **Introduction: hierarchical count-of-counts histograms**
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

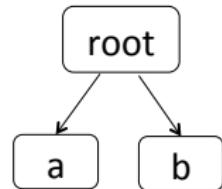
## 4 Backup Slides



# Scenario

- Table **Persons**(person\_name, group\_id, location)
- A hierarchy  $\Gamma$  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:

- group = a taxi, data item = a pick up, size = # of pickup
- 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:

<u>g_id</u>	size	loc.
1	4	a
2	2	b
3	1	a
4	1	b

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



# Protect Privacy

## Definition (Differential Privacy [DMNS06])

A mechanism  $M$  satisfies  $\epsilon$ -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])

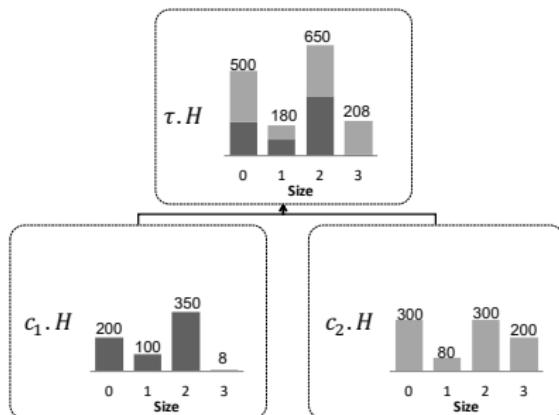
Given a database  $D$ , a query  $q$  that outputs a vector, a privacy loss budget  $\epsilon$ , the global sensitivity  $\Delta(q)$ , the geometric mechanism adds independent noise to each component of  $q(D)$  using distribution:

$P(X = k) = \frac{1-e^{-\epsilon}}{1+e^{-\epsilon}} e^{-\epsilon|k|/\Delta(q)}$  (for  $k = 0, \pm 1, \pm 2$ , etc.). This distribution is known as the double-geometric with scale  $\Delta(q)/\epsilon$ .

## Problem Definition

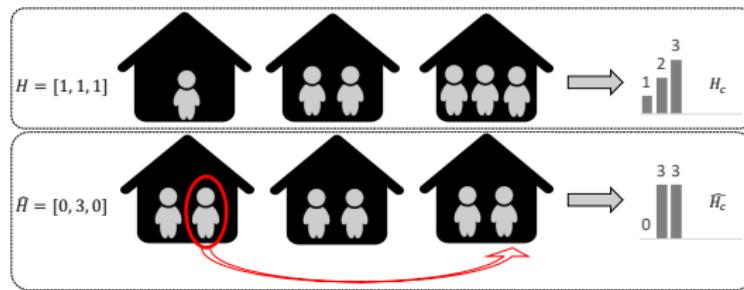
For each node  $\tau$  in hierarchy  $\Gamma$ , create differentially private estimate  $\tau.\hat{H}$  of count-of-counts histogram  $H$  such that

- $\tau.\hat{H}$  is a count-of-counts histogram (its entries are nonnegative integers)
- The counts are accurate ( $\tau.\hat{H}$  and  $\tau.H$  are close)
- $\tau.\hat{H}$  matches publicly known total number of groups  $G$  in  $\tau$
- 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  $\tau.\hat{H}$



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

## Lemma ([NLV07])

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

# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

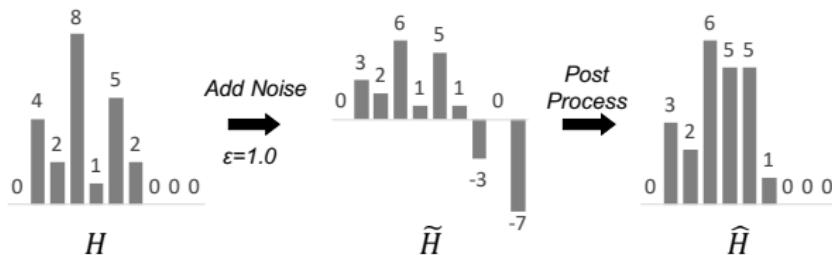
- Background
- Introduction: hierarchical count-of-counts histograms
- **Non-hierarchical count-of-counts histograms publishing**
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Naive Strategy



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

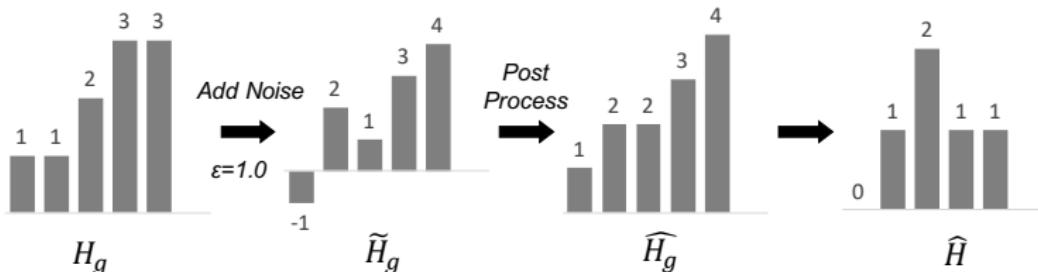
$$\hat{H} = \arg \min_{\hat{H}} \|\tilde{H} - \hat{H}\|_2^2$$

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

- ③ To get integers, we set  $r = G - \sum_i \lfloor \hat{H}[i] \rfloor$ , 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] $H_g$



- ① Convert coco histogram  $H \Rightarrow$  unattributed histogram  $H_g$
- ②  $\tilde{H}_g$ : 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  $p = 2$ :

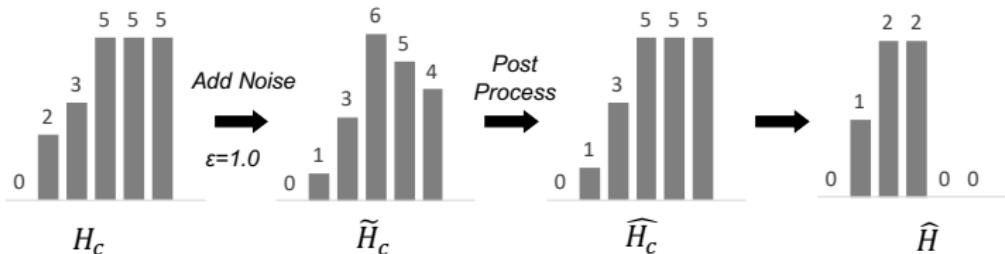
$$\hat{H}_g = \arg \min_{\hat{H}_g} \|\tilde{H}_g - \hat{H}_g\|_p^p$$

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

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



# Cumulative Sum Histograms $H_c$



- ① Convert coco histogram  $H \Rightarrow$  cumulative sum histogram  $H_c$
- ②  $\tilde{H}_c$ : 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  $p = 2$ :

$$\hat{H}_c = \arg \min_{\hat{H}_c} \|\hat{H}_c - \tilde{H}_c\|_p^p$$

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

and  $\hat{H}_c[K] = G$

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



## Methods Summary

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



# 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
$H_c$	3,742	1,838	254	2,819
$H_g$	2,219	6,115	516	11,227



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing**
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

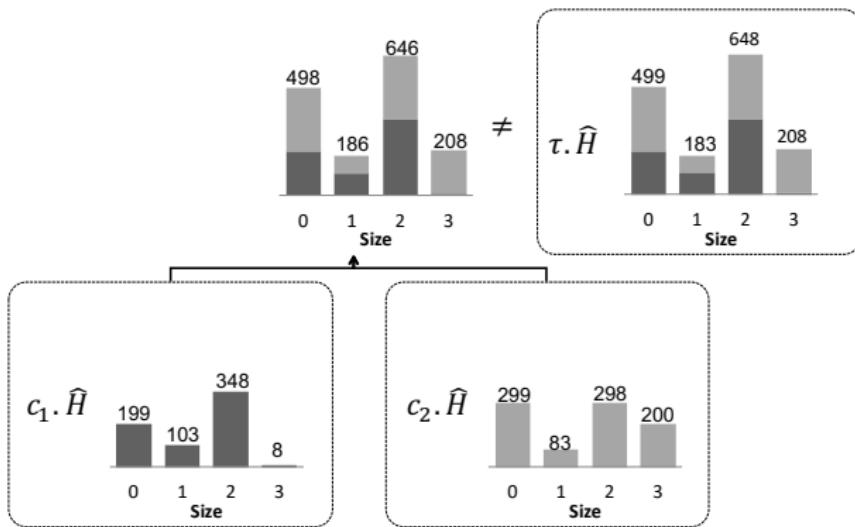
## 3 Contributions

## 4 Backup Slides



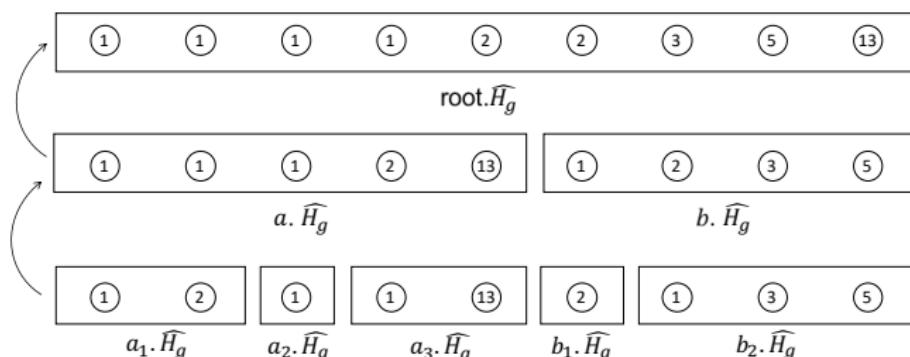
# Non-hierarchical Methods Issue

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



# Bottom-up Aggregation

- ➊ Estimate coco histogram  $H$  only at the leaves
- ➋ Aggregate them up the hierarchy



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



# Consistency Solution

- Our proposed solution:

- Converts estimated coco  $\tau.\hat{H}$   $\Rightarrow$  the unattributed histogram  $\tau.\hat{H}_g$
- Find a 1-to-1 optimal matching between groups at the child nodes and groups at the parent node
- Merge those two estimates

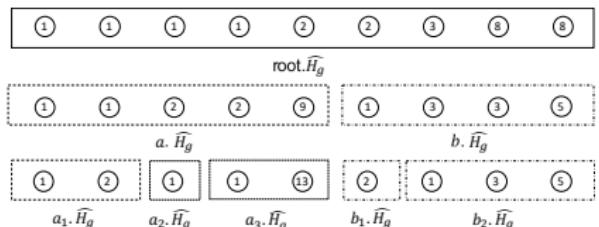


Figure: Before matching

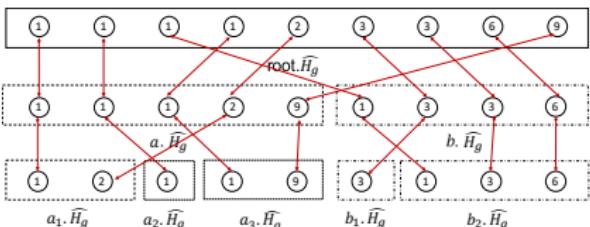
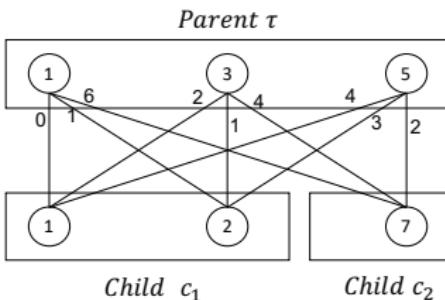


Figure: Consistency result



# Optimal Matching Algorithm

- For each node  $\tau$  and its children, we set up a bipartite weighted graph
- There are  $\tau.G$  vertices on the top:  $(\tau, 1), (\tau, 2), \dots, (\tau, \tau.G)$ . Each vertex on the bottom has the form  $(c, j)$ , where  $c$  is a child of  $\tau$  and  $j$  is an index into  $c.\hat{H}_g$
- Edge between every vertex  $(\tau, i)$  and  $(c, j)$  has weight  $|\tau.\hat{H}_g[i] - c.\hat{H}_g[j]|$ : 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  $\tau$  to the smallest unmatched group among any of its children



# Top-down Consistency

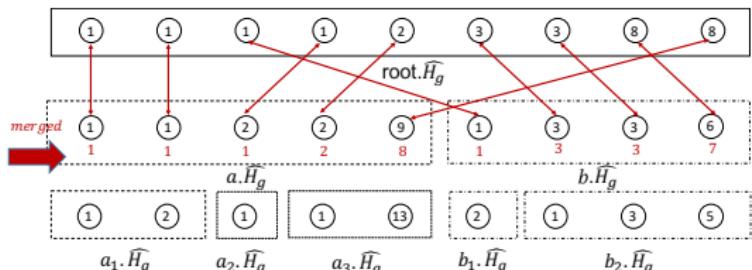


Figure: Level 0 and Level 1 consistency matching

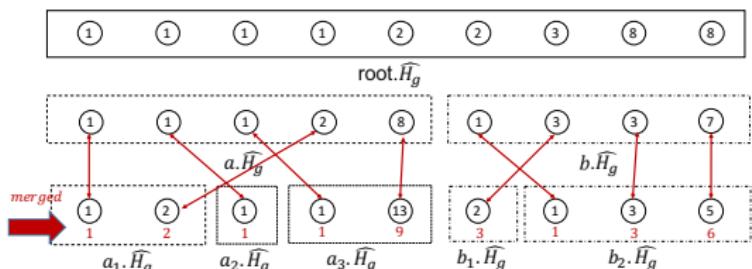


Figure: Level 1 and Level 2 consistency matching

- 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:

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

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

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

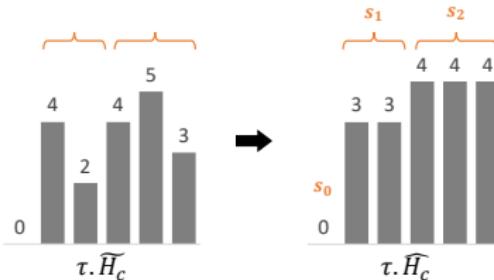
- ④ Convert consist unattributed histogram into count-of-counts histogram

# Initial Variance Estimation

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

For each  $i$ , we need an estimate of the variance of the  $i^{\text{th}}$  largest group  $\tau.\hat{H}_g[i]$ , 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  $\epsilon$  be the privacy budget used in node  $\tau$  in level  $\ell$  of  $\Gamma$

For the **H<sub>g</sub>** method:

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

For the **H<sub>c</sub>** method:

- Variance estimate of the  $i^{\text{th}}$  largest group:  

$$\tau.V_g[i] = 4 / (\epsilon^2 \times \text{number of estimated groups of size } \tau.\hat{H}_g[i])$$



# Merge Estimates

Given a node  $\tau$ , the matching algorithm assigns one group  $i$  in  $\tau$  to one group  $j$  in some child of  $\tau$

$\Rightarrow$  for every group, two estimates of its size:  $\tau.\hat{H}_g[i]$  and  $c.\hat{H}_g[j]$  & estimates of variance  $\tau.V_g[i]$  and  $c.V_g[j]$

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

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

and the variance of this estimator is

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



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results**
  - Application to unattributed histograms publishing
  - Empirical Evaluation
  - Summary

## 3 Contributions

## 4 Backup Slides



# Experiments

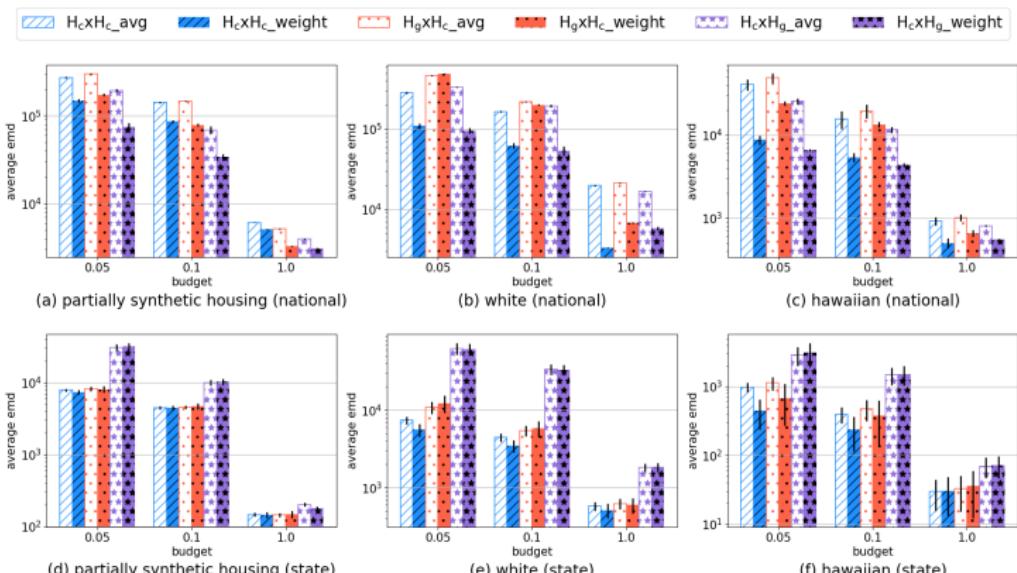
Use 4 datasets:

- **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?



# Weighted Average Estimation Comparison

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



**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
$H_c$	<b>32,480.0</b>	<b>17,000.0</b>	<b>1,381.0</b>	<b>10,547.0</b>

Level 1

BU	1,512.2	8,722.0	270.1	10,405.5
$H_c$	<b>1,000.3</b>	<b>1,511.8</b>	<b>117.7</b>	<b>5,431.5</b>

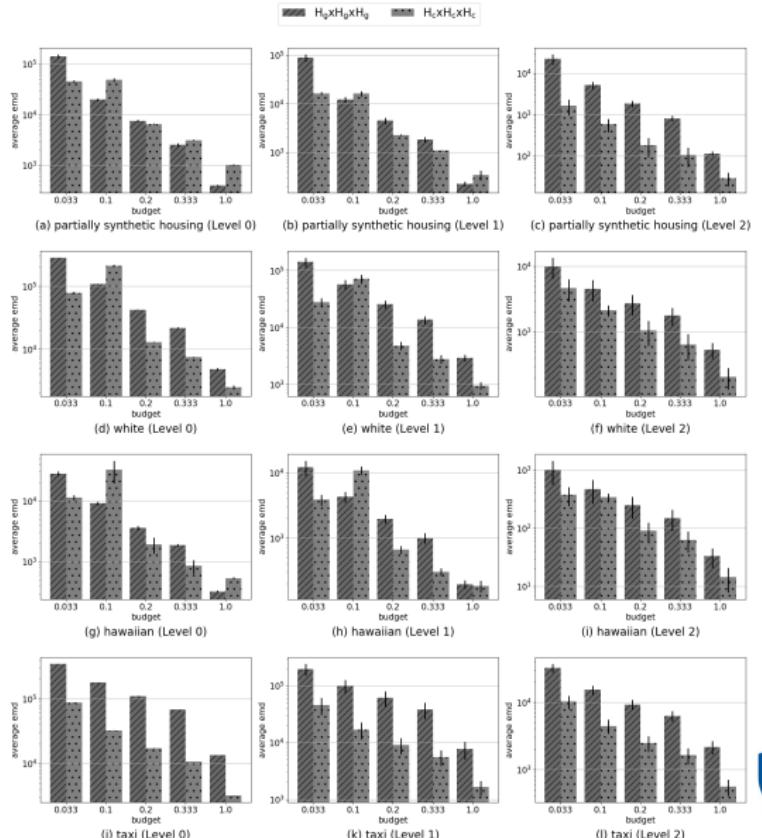
Level 2

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



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing**
- Empirical Evaluation
- Summary

## 3 Contributions

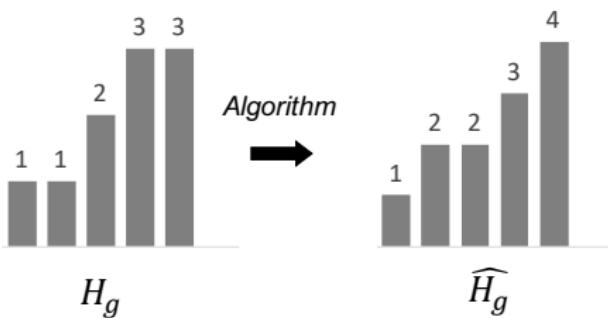
## 4 Backup Slides



# Problem Formulation

- ① Given a unattributed histogram  $H_g$ , create differentially private estimate  $\widehat{H}_g$  for  $H_g$  such that

- $\widehat{H}_g$  is a unattributed histogram (entries are nonnegative sorted integers)
- The values (i.e. group size) are accurate ( $\widehat{H}_g$  and  $H_g$  are close)



- ② Error measure:

- $L_1$  error:  $\|H_g - \widehat{H}_g\|_1$
- $L_2$  error:  $\|H_g - \widehat{H}_g\|_2$



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- **Empirical Evaluation**
- Summary

## 3 Contributions

## 4 Backup Slides



# Experiments

Use 30 datasets from different scales:

- small-scale: datasets from [LK13], 5 race categories and taxi data at level 1 (e.g. state level)
- medium-scale: datasets from Yahoo! passwords [Bon12] and taxi data at top level. The HMM.forward-backward and HMM.viterbi are excluded due to long run time
- large-scale: datasets from Yahoo! passwords and 5 race categories at top level (e.g. U.S.). Only consider efficient algorithms



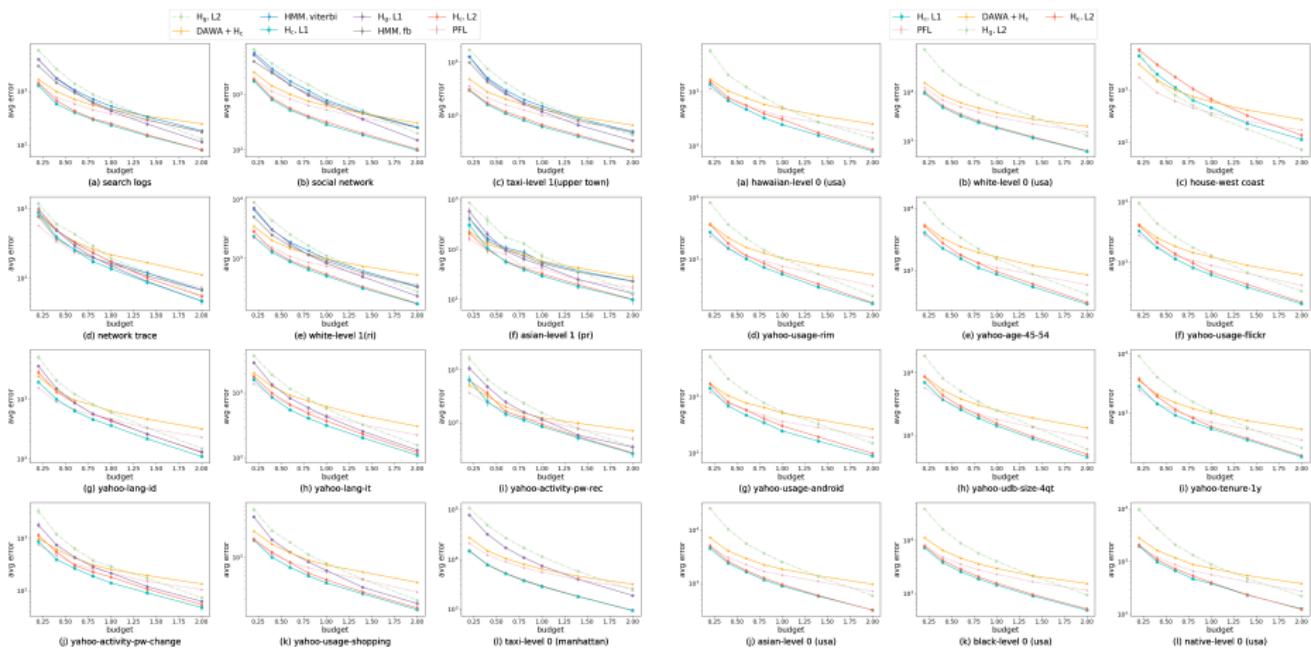
# Identify Competitive Algorithms

	Social		Network trace	
	$L_1$ error	$L_2$ error	$L_1$ error	$L_2$ error
Hc.L1	<b>288.0</b>	<b>22.7</b>	<b>136.4</b>	21.6
Hc.L2	316.2	28.9	173.2	29.8
Hg.L1	684.4	27.1	150.6	14.5
Hg.L2	1001.2	32.4	192.2	15.8
PFL	525.8	25.0	154.2	17.3
HMM.fb	721.8	27.5	163.0	<b>13.8</b>
viterbi	790.4	29.1	171.8	14.4
DAWA+Hc	623.4	34.4	216.2	26.9
DAWA+Hg	958.8	34.7	242.0	22.0
DAWA+H	18766.4	7437.2	10035.6	2036.4
AHP+Hc	4176.8	1379.9	1009.6	339.8
AHP+Hg	3455.2	74.6	4168.8	110.4
AHP+H	23517311.0	1238953.1	10627347.8	835915.6
HbTree+Hc	1724.6	196.9	496.6	95.3
HbTree+Hg	4580.2	75.8	992.2	47.1
HbTree+H	1119693450.0	10517399.4	221270147.2	4698290.4



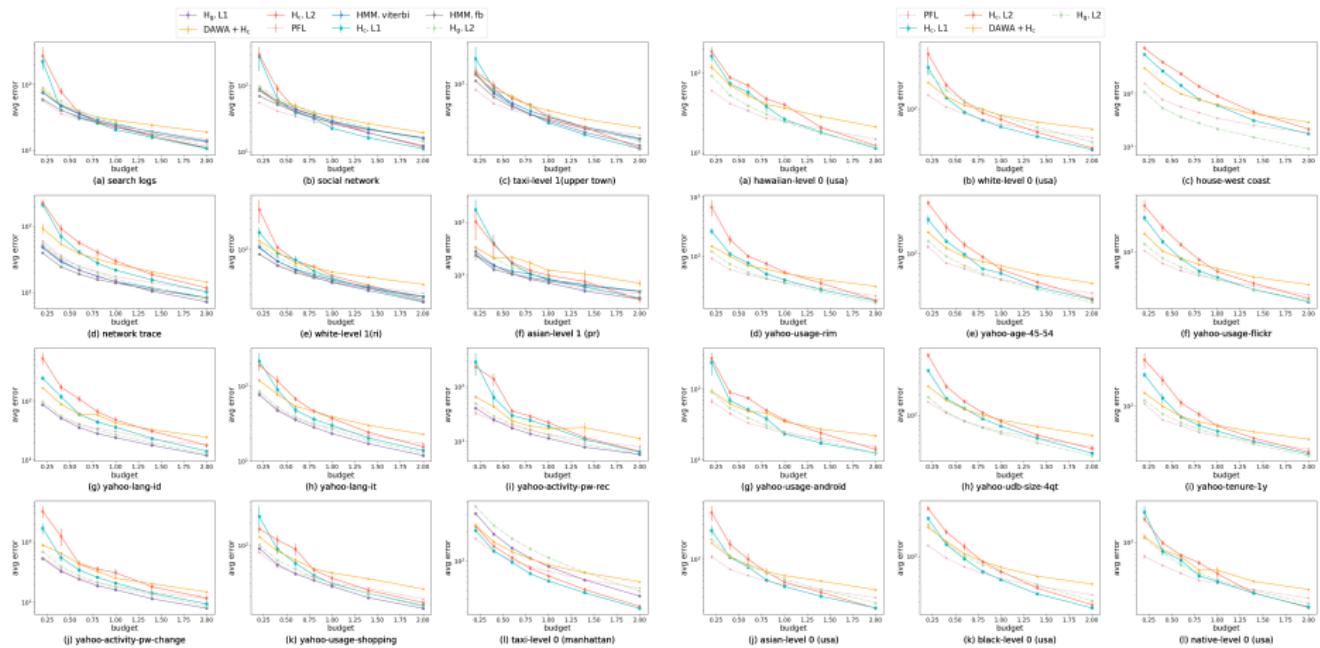
# $L_1$ Error Results

- $H_{\epsilon}.L1$  is usually good at  $L_1$  error except for partially synthetic housing data (at west coast).



# $L_2$ Error Results

- For  $\epsilon \leq 1.0$ : PFL consistently produces good  $L_2$  errors
- For  $\epsilon > 1.0$ : When the fraction of tail in coco histogram is large, use Hg.L1 or Hg.L2. when the fraction of dense (i.e. large counts) region in coco histogram is large, use Hc.L1



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Conclusion

- 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
- In publishing count-of-counts histograms,  $H_c$  method generally performs better on dense dataset while datasets with more sparsity favor  $H_g$  method
- Identify methods that could be used for unattributed histogram task
- Empirically evaluate methods on a variety of datasets and provide a better understanding for when the competitive algorithms do well



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Thesis Contributions

- Develop a system to detect outliers by correlations between measurements which facilitates the public or research use of large-scale datasets collected from a network of sensors
- Introduce hierarchical count-of-counts problem along with appropriate error metrics
- Propose a differentially private solution for non-hierarchical and hierarchical count-of-counts histograms which could be used for tables that are published in truncated form in Summary File 1 of the 2010 U.S. Census
- Empirically evaluate methods for publishing unattributed histogram on a variety of datasets which provides data owner a guidance of when the algorithms do well



# Future Directions

For outlier detection,

- In some scenarios, sensors collect time-series data. Such temporal factor could be considered with the correlations between attributes in modeling the outlier
- The current outlier model could be adapted to an active learning based approach to expand its application

For differentially private histograms,

- In data-dependent situation, models that take datasets as input and predict the least error algorithm could be helpful for data owner to select algorithms



# Publications

- ① Yu-Hsuan Kuo, Zhenhui Li, Daniel Kifer. Detecting Outliers in Data with Correlated Measures. CIKM 2018.
- ② Yu-Hsuan Kuo, Cho-Chun Chiu, Daniel Kifer, Michael Hay, Ashwin Machanavajjhala. Differentially Private Hierarchical Count-of-Counts Histograms. PVLDB 2018.
- ③ Hongjian Wang, Yu-Hsuan Kuo, Daniel Kifer, Zhenhui Li. A Simple Baseline for Travel Time Estimation using Large-Scale Trip Data. SIGSPATIAL 2016.
- ④ Corina Graif, Brittany Freelin, Yu-Hsuan Kuo, Hongjian Wang, Zhenhui Li, Daniel Kifer. Network spillovers and neighborhood crime: A computational statistics analysis of employment-based networks of neighborhoods. (Under Review)



# Questions?



# Outline

## 1 Detecting Outliers with Correlated Measures

- Introduction: Contextual Outliers in Big Data
- Outlier Modeling
- Experimental Results
- Summary

## 2 Differentially Private Count-of-counts Histograms

- Background
- Introduction: hierarchical count-of-counts histograms
- Non-hierarchical count-of-counts histograms publishing
- Hierarchical count-of-counts histograms publishing
- Experimental results
- Application to unattributed histograms publishing
- Empirical Evaluation
- Summary

## 3 Contributions

## 4 Backup Slides



# Likelihood Function

- ① A zero mean Gaussian with unknown variance  $\sigma^2$  has probability density

$$f_G(\epsilon_i; \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\epsilon_i^2}{2\sigma^2}\right)$$

- ② Cauchy distribution with scale parameter  $b$  is a heavy-tailed distribution with undefined mean and variance  $\Rightarrow$  ideal for modeling outliers.
- A sample  $\epsilon_i$  from this distribution: first sampling a value  $\tau_i$  from the  $\text{Gamma}(0.5, b)$  distribution then sampling  $\epsilon_i$  from the  $\text{Gaussian}(0, 1/\tau_i)$  distribution [BL09]:

$$f_C(\epsilon_i, \tau_i; b) = \frac{b^{0.5}}{\Gamma(0.5)} \tau_i^{0.5-1} e^{-b\tau_i} \frac{\sqrt{\tau_i}}{\sqrt{2\pi}} \exp\left(-\frac{\tau_i \epsilon_i^2}{2}\right)$$



# Likelihood Function (Cont.)

- ③ Latent indicator  $\chi_i$ : where the error of contextual attribute  $\vec{x}_i$  comes (i.e. from Cauchy or Gaussian)
- ④ With the model parameters  $\vec{w}$ , unknown noise parameters  $\sigma^2$  (variance of non-outliers),  $p$  (outlier probability),  $b$  (scale parameter of outlier distribution), the likelihood function is

$$\begin{aligned}
 L(\vec{w}, \sigma^2, p, b, \vec{\chi}, \vec{\tau}) &= \prod_{i=1}^n \left[ (1-p) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \vec{w} \cdot \vec{x}_i)^2}{2\sigma^2}\right) \right]^{1-\chi_i} \times \\
 &\quad \left[ p \frac{b^{0.5}}{\Gamma(0.5)} \tau_i^{0.5-1} e^{-b\tau_i} \frac{\sqrt{\tau_i}}{\sqrt{2\pi}} \exp\left(-\frac{\tau_i(y_i - \vec{w} \cdot \vec{x}_i)^2}{2}\right) \right]^{\chi_i}
 \end{aligned}$$



# Synthetic Outlier - Perturb Behavioral Attributes (ElNino)

- Our outlier detector consistently performs the best when more outliers are involved.

Table: PR AUC w.r.t different fractions of synthetic outliers in behavioral attribute

method	ElNino				
	q=0.01	q=0.03	q=0.05	q=0.1	q=0.15
Doc	<b>0.96</b>	<b>0.97</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>
ROCOD (non-linear)	0.73	0.73	0.74	0.73	0.72
CAD	0.80	0.84	0.86	0.85	0.88
OLS	<b>0.96</b>	0.95	0.95	0.92	0.90
GBT	<b>0.96</b>	0.95	0.95	0.92	0.90
distance-based	0.81	0.74	0.77	0.83	0.60
density-based	0.21	0.38	0.45	0.38	0.34



# Synthetic Outlier - Perturb Contextual Attributes (Elnino)

- A small fraction of outliers in contextual attribute hurts the performance considerably for the other methods.
- Our method is robust and resistant to the fraction of outliers.

**Table:** PR AUC w.r.t different fractions of synthetic outliers in contextual attribute

	Elnino				
method	q=0.005	q=0.01	q=0.03	q=0.05	q=0.07
Doc	<b>0.97</b>	<b>0.95</b>	<b>0.97</b>	<b>0.98</b>	<b>0.98</b>
ROCOD (non-linear)	0.01	0.01	0.02	0.02	0.03
CAD	0.80	0.83	0.86	0.88	0.87
OLS	0.92	0.86	0.68	0.45	0.32
GBT	0.11	0.15	0.28	0.37	0.40
distance-based	0.88	0.74	0.81	0.50	0.83
density-based	0.08	0.07	0.08	0.09	0.10



# Synthetic Outlier - Degree of Outlierness (Elnino)

- As  $\alpha$  increases, larger magnitude of noise will have more chance to be added to the original value.
- Our performance increased as more extreme outliers are added.

Table: PR AUC w.r.t degree of outlierness  $\alpha$  in contextual attribute

	Elnino				
method	$\alpha = 20$	$\alpha = 30$	$\alpha = 50$	$\alpha = 100$	$\alpha = 300$
Doc	<b>0.91</b>	<b>0.94</b>	<b>0.95</b>	<b>0.98</b>	<b>0.99</b>
ROCOD (non-linear)	0.01	0.01	0.01	0.01	0.01
CAD	0.78	0.8	0.83	0.87	0.93
OLS	0.88	0.89	0.86	0.85	0.73
GBT	0.17	0.17	0.15	0.17	0.17
distance-based	0.21	0.79	0.74	0.88	0.91
density-based	0.13	0.10	0.07	0.05	0.04



# Outlier Rules Discovery

- The outlier explainer will search for rules that combine features in ways that respect units

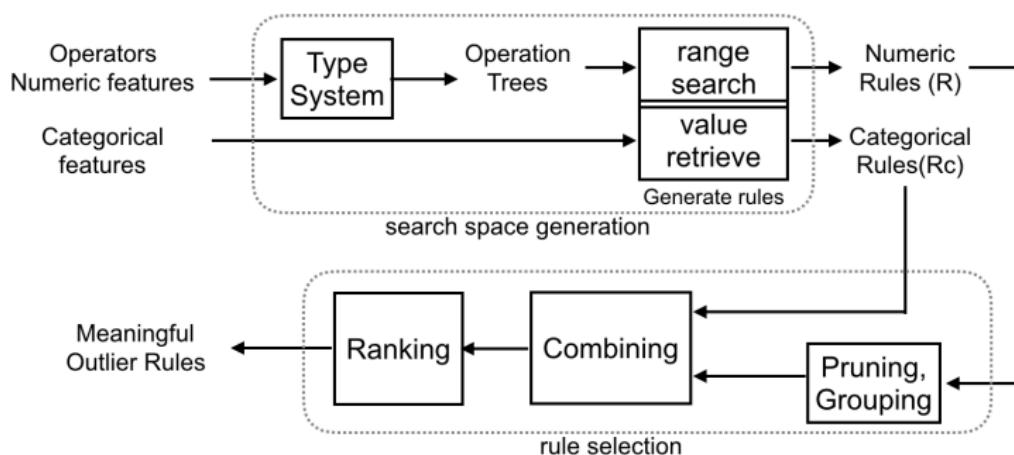


Figure: Framework overview - extract outlier rules



# Outlier Rules Discovery - Search Space Generation

- Operation trees are created by combining numeric features based on pre-defined type system
- Convert every operation tree into numeric rules by searching for ranges
- Construct initial categorical rules by retrieving the values of categorical features that result in reasonable recall and precision

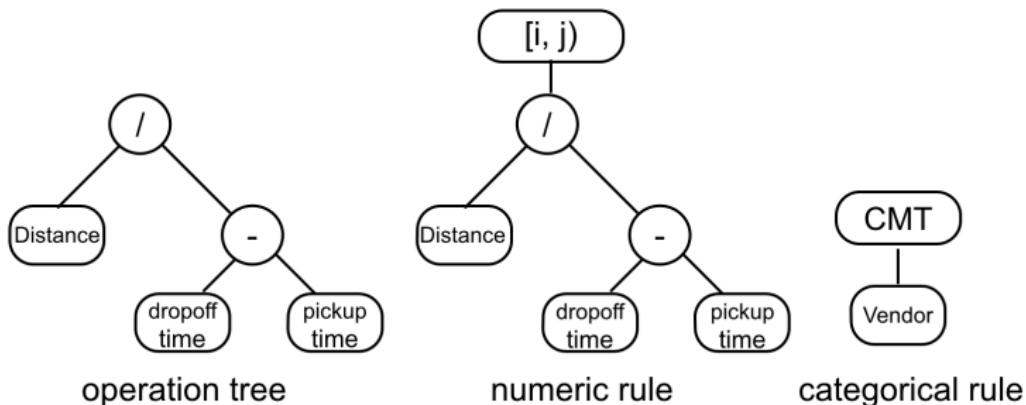


Figure: Examples: operation tree, numeric rule, categorical rule



# Outlier Rules Discovery - Rule Selection Phrase

- Eliminate the numeric rule candidates by pruning and grouping similar rules together
- Combine numeric rules and categorical rules to improve the precision and/or recall
- Rank the rules by F1 score
- Return top ranked rules



# Top Ranked Outlier Rules from Intel Sensor Data

- The majority of outliers are covered by

SRule1: Week 3  $\wedge$   $122.15 \leq \text{Temp} < 175.68$

Precision: 1.0 Recall: 0.572

- Records generating temperature readings  $\in [122.15, 175.68)$  earlier on Week 1 or Week 2 are located in the upper right corner of the Intel lab where the coordinates are  $X < 6$  &  $Y < 17 \Rightarrow$  sensors in this room are having shorter lifetime (average 20 days) than the majority ones (average 35 days)

SRule1a:  $X < 6 \wedge Y < 17 \wedge 122.15 \leq \text{Temp} < 175.68$

$\wedge$  (Week 1, Week 2)

Precision: 1.0 Recall: 0.12

- A decreasing trend in voltage for this batch of sensors

SRule5a:  $V \geq 2.81 \wedge \text{Week 4}$

Precision: 1.0 Recall 0.027



# Top Ranked Outlier Rules from NYC Taxi Data

- The trip time tracking systems provided by Creative Mobile Technologies (CMT) are programmed differently from Verifone (VTS).

TRule1:  $|Time - (dtime - ptime)| \geq 3 \text{ sec} \wedge \text{CMT}$

Precision: 0.99 Recall: 0.112

- Travel time in Manhattan longer than 70 minutes  $\Rightarrow$  not common

TRule4:  $dtime - ptime \geq 70.7 \text{ min}$

Precision: 0.9 Recall 0.002

- Average speed is  $\in (3.56\text{mph}, 33.27\text{mph})$   $\Rightarrow$  supported by the fact that the local speed limit in NYC of year 2013 is 30 mph.

TRule5:  $D / (dtime - ptime) < 3.46 \text{ mph}$

Precision: 0.92 Recall: 0.24

TRule6:  $D / (dtime - ptime) > 33.27 \text{ mph}$

Precision: 0.99 Recall: 0.0014

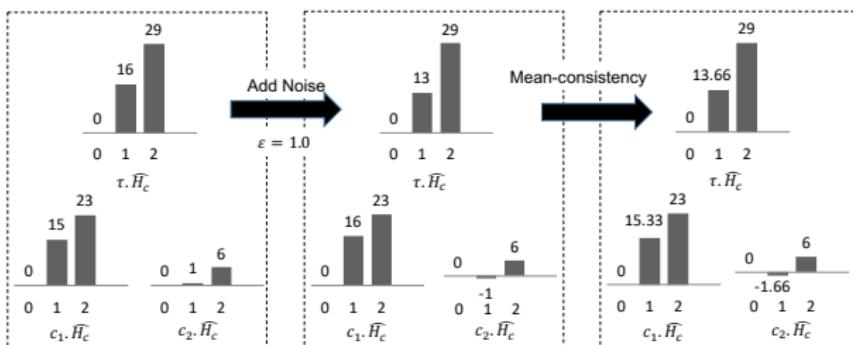


## Differentially Private Histograms



# Mean-Consistency Algorithm [HRMS10]

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

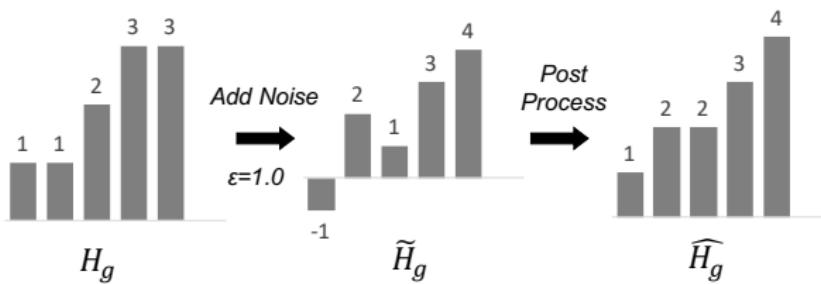


- ➍ Drawback: counts can be negative and fractional



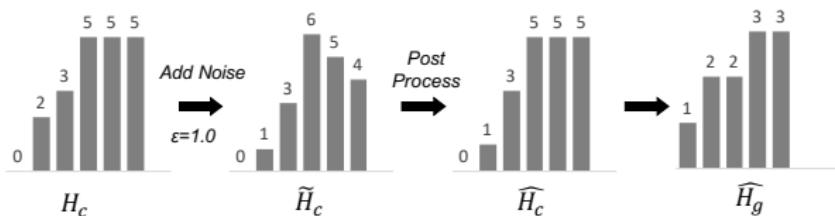
# Algorithms Designed for Unattributed Histogram

- ① PFL [BDB16]: efficient version of exponential mechanism [MT07]
- ② HMM.forward-backward [LK13]: model unattributed histogram  $H_g$  as a Hidden Markov Model (HMM) with forward-backward algorithm as inference procedure. In HMM, the time  $\Rightarrow$  group id and the state at time  $t \Rightarrow$  size of group id  $t$
- ③ HMM.viterbi: HMM with Viterbi as inference algorithm
- ④  $H_g$ : isotonic regression on unattributed histogram



# Algorithms Designed for Other Problems

- ⑤  $H_c$ : isotonic regression on cumulative coco histogram



- ⑥ HbTree [QYL13]: hierarchical method for range queries
- ⑦ AHP [ZCX<sup>+</sup>14]: clustering technique for range queries
- ⑧ DAWA [LHM14]: 2-stage mechanism for range queries. We adopt its partitioning algorithm for estimation.



# How We Use Algorithms for Other Problems

- HbTree, AHP, DAWA:  
can be applied to  
different  
representations
- Combined with  
corresponding  
post-processing  
algorithms

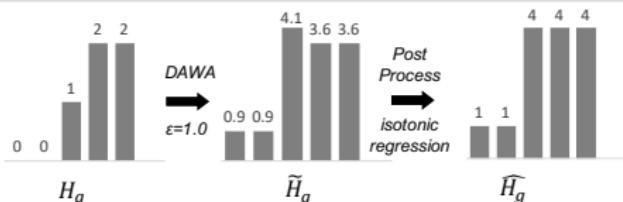


Figure: Take  $H_g$  as input. Use DAWA+Hg post processing



Figure: Take  $H$  as input. Use DAWA+H post processing

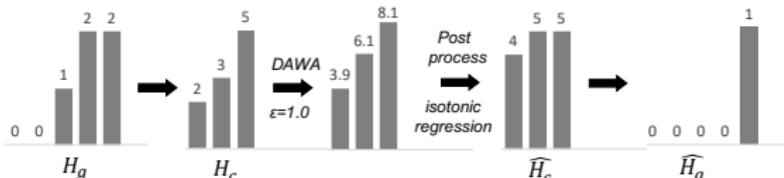


Figure: Take  $H_c$  as input. Use DAWA+Hc post processing

# 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.
-  Jeremiah Blocki, Anupam Datta, and Joseph Bonneau.  
Differentially Private Password Frequency Lists.  
In NDSS '16: The 2016 Network and Distributed System Security Symposium, page 153, February 2016.
-  Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander.  
Lof: identifying density-based local outliers.  
In SIGMOD, 2000.



## References II

-  Narayanaswamy Balakrishnan and Chin-Diew Lai.  
Continuous bivariate distributions.  
Springer Science & Business Media, 2009.
-  Joseph Bonneau.  
The science of guessing: analyzing an anonymized corpus of 70 million passwords.  
In Security and Privacy (SP), 2012 IEEE Symposium on, pages 538–552. IEEE, 2012.
-  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.



## References III

-  A. P. Dempster, N. M. Laird, and D. B. Rubin.  
Maximum likelihood from incomplete data via the EM algorithm.  
Journal of the Royal Statistical Society: Series B, 39:1–38, 1977.
-  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.
-  Jerome H Friedman.  
Greedy function approximation: a gradient boosting machine.  
Annals of statistics, pages 1189–1232, 2001.
-  Inc. Gurobi Optimization.  
Gurobi optimizer reference manual, 2016.



## References IV

-  Arpita Ghosh, Tim Roughgarden, and Mukund Sundararajan.  
Universally utility-maximizing privacy mechanisms.  
In [STOC](#), pages 1673–1693, 2009.
-  Charmgil Hong and Milos Hauskrecht.  
Mcode: Multivariate conditional outlier detection.  
[arXiv preprint arXiv:1505.04097](#), 2015.
-  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.



## References V

-  C. Li, M. Hay, and G. Miklau.  
A data- and workload-aware algorithm for range queries under differential privacy.  
PVLDB, 7(5):341–352, 2014.
-  Bing-Rong Lin and Daniel Kifer.  
Information preservation in statistical privacy and bayesian estimation of unattributed histograms.  
In ACM SIGMOD, pages 677–688, 2013.
-  Jiongqian Liang and Srinivasan Parthasarathy.  
Robust contextual outlier detection: Where context meets sparsity.  
In CIKM. ACM, 2016.



## References VI

-  Frank McSherry and Kunal Talwar.  
Mechanism design via differential privacy.  
In Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on, pages 94–103. IEEE, 2007.
-  T. Li N. Li and S. Venkatasubramanian.  
t-closeness: Privacy beyond k-anonymity and l-diversity.  
In ICDE, pages 106–115, 2007.
-  Wahbeh Qardaji, Weining Yang, and Ninghui Li.  
Understanding hierarchical methods for differentially private histograms.  
PVLDB, 6(14):1954–1965, 2013.



## References VII

-  Tim Robertson, Paul Waltman, et al.  
On estimating monotone parameters.  
The Annals of Mathematical Statistics, pages 1030–1039, 1968.
-  Mahito Sugiyama and Karsten Borgwardt.  
Rapid distance-based outlier detection via sampling.  
In NIPS, 2013.
-  Xiuyao Song, Mingxi Wu, Christopher Jermaine, and Sanjay Ranka.  
Conditional anomaly detection.  
ICDE, 2007.
-  Hongjian Wang, Yu-Hsuan Kuo, Daniel Kifer, and Zhenhui Li.  
A simple baseline for travel time estimation using large-scale trip data.  
In SIGSPATIAL. ACM, 2016.



## References VIII

-  Eugene Wu and Samuel Madden.  
Scorpion: Explaining away outliers in aggregate queries.  
In VLDB Journal, 2013.
-  X. Zhang, R. Chen, J. Xu, X. Meng, and Y. Xie.  
Towards accurate histogram publication under differential privacy.  
In ICDM, pages 587–595, 2014.

