Differentially Private CDF Evaluation and Development

Daniel Muise[1,2] Mentors: Victor Balcer[2], Mark Bun[2], Kobbi Nissim[2]

[1]NSF REU-2015 student from UMass Lowell, [2]Harvard University SEAS

Privacy Tools for Sharing Research Data

A National Science Foundation Secure and Trustworthy Cyberspace Project

with additional support from the Sloan Foundation and Google, Inc.

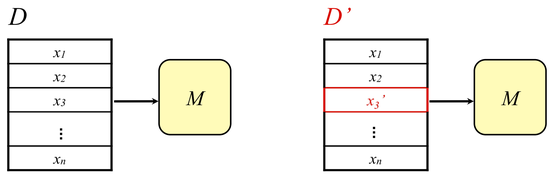
MOTIVATING QUESTION

How can we measure the utility of various differentially private statistics, and use that to bring differential privacy theory into social science practice?

OBJECTIVES

1. Develop a system of comparatively measuring the utility of all possible differentially private CDF-computation methods.
2. Use this system to evaluate existing methods and thereby guide the design of newer methods, which we also undertake.
3. Convey the usefulness and nuances of differentially private computation to end users.

DIFFERENTIAL PRIVACY

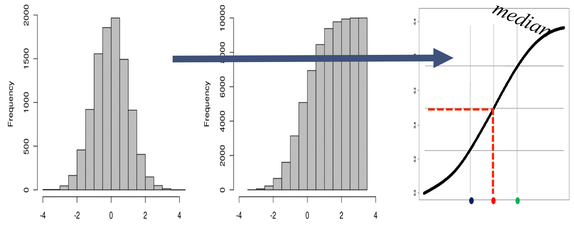


D and D' are neighbors if they differ only on one user's data

An algorithm M is (ε)-differentially private if for all neighbors D, D' and every $S \subseteq \text{Range}(M)$,

$$\Pr[M(D) \in S] \leq e^{\varepsilon} \Pr[M(D') \in S]$$

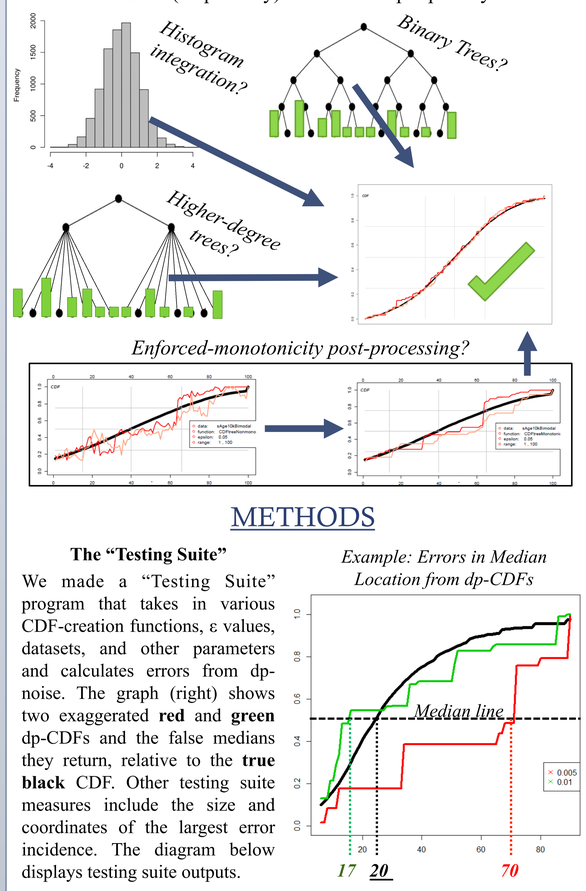
CUMULATIVE DENSITY FUNCTIONS (CDFs)



We focused on CDFs, integrations of random variables' probability mass functions. They're popular for answering threshold queries and quantile queries. Social science uses include voting counts, tax brackets, income distributions, GINI coefficients, and more.

CDF-CREATION ALGORITHMS

Which method is (empirically) least-errored per privacy allotment?



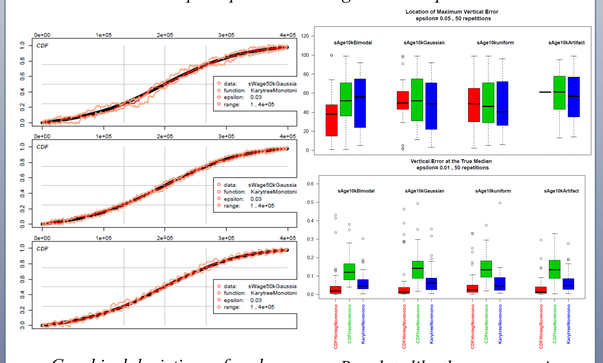
METHODS

The "Testing Suite"

We made a "Testing Suite" program that takes in various CDF-creation functions, ε values, datasets, and other parameters and calculates errors from dp-noise. The graph (right) shows two exaggerated **red** and **green** dp-CDFs and the false medians they return, relative to the **true black** CDF. Other testing suite measures include the size and coordinates of the largest error incidence. The diagram below displays testing suite outputs.

Example: Errors in Median Location from dp-CDFs

Example: dp-CDF "Testing Suite" Output



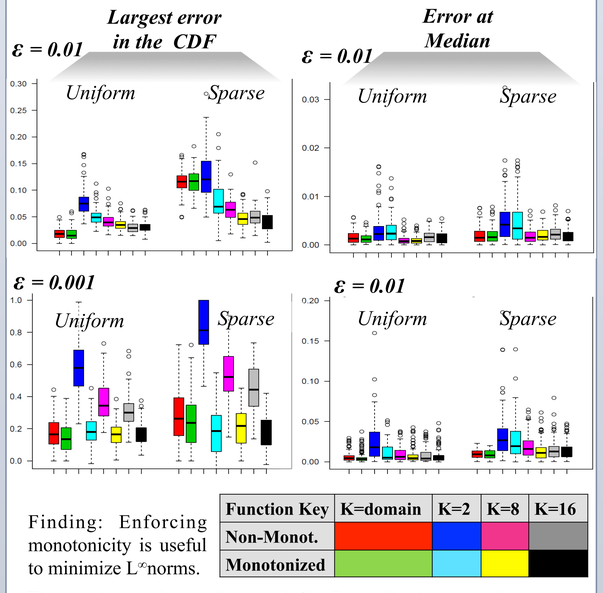
Graphical depictions of each function show one possible dp-CDF output and visualizations of its outer variance boundaries, overlaid on the data's true CDF (in black).

Boxplots like these summarize absolute errors (of various types) averaged over many repetitions of random noise application, compared across functions and parameters.

RESULTS: ALGORITHM COMPARISONS

Relative extent of errors per algorithm

N= 100k, R = [1:85], gran =1, 100 iterations each



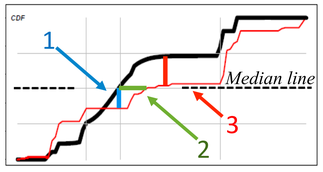
Finding: Enforcing monotonicity is useful to minimize L^{∞}norms.

Function Key	K=domain	K=2	K=8	K=16
Non-Monot.				
Monotonized				

This is clear at low ε (bottom left). Generally, however, histogram integration appears more useful than tree-construction for this domain size (85). Trees generally show improvement with higher degree.

EVALUATION: DP vs. SAMPLING ERROR

Users often mistrust random noise introduction, not seeing that they already face random sampling error in all statistics. We used the testing suite to compare the two types of error.



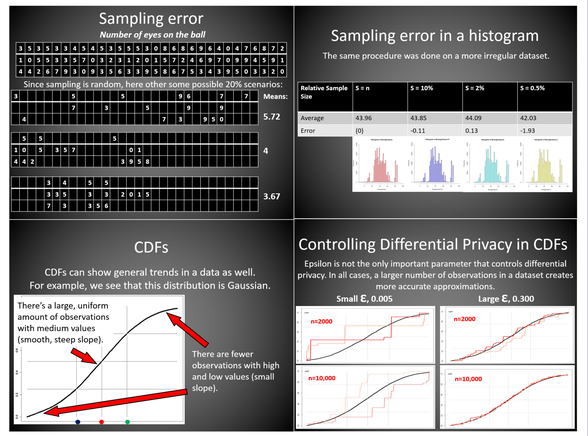
We made four datasets, (gran =1, range [1:80], ***n***=100k) and ran 100 iterations each of four ε values and four sample sizes, and averaged absolute errors on each. We used a 16-degree tree-based function. Monotonicity-enforced versions are included. Importantly, we find that errors are similar.

Parameters mimic realistic settings. Direct vertical comparisons between green and blue boxes are arbitrary.

ε	0.1	0.05	0.01	0.005
sample	**10%**	**1%**	**0.5%**	**0.1%**
Mean abs. error at median value				1
D.P.	0.003	0.010	0.004	0.016
Monot.	0.003	0.010	0.004	0.015
Sampling	0.004	0.003	0.005	0.009
Mean $\lvert$(dp-median$_i$)- (true median)$\rvert$				2
D.P.	0.128	0.170	0.874	1.613
Monot.	0.133	0.188	0.738	1.643
Sampling	0.260	0.416	0.359	2.918
Mean maximum error				3
D.P.	0.003	0.007	0.033	0.066
Monot.	0.003	0.006	0.029	0.056
Sampling	0.006	0.009	0.008	0.059

DP-CDF "REFERENCE MANUAL"

We're using the testing suite results to create a document for end users. Our strategy is to introduce differential privacy as analogous to sampling error, something already tolerated and understood.



CONCLUSIONS

DP-CDF implementations are nearly ready for public use, and their utility can be expressed empirically. Histogram integration with monotonicity is sufficient for most datasets. We seek future research with high computing power on tree-based CDFs of wide datasets.

REFERENCES

[BNSV14] Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil Vadhan. Differentially private release and learning of thresholds. *Manuscript*, 2014.

[HRMS10] Michael Hay, Vibhor Rastogi, Gerome Miklau, and Dan Suciu. Boosting the accuracy of differentially private histograms through consistency. *Proc. VLDB Endow.*, vol. 3, no. 1-2, pp. 1021–1032, Sep. 2010

[H15] James Honaker. Efficient Use of Differentially Private Binary Trees. In *TPDP15,* 2015.

CONTACT

Daniel_Muise@student.uml.edu

57 Monument Street

Haverhill MA 01832

University of Massachusetts Lowell Honors College: Economics, Political Science, & Asian Studies depts.

Harvard John A. Paulson School of Engineering and Applied Sciences

CRCS Center for Research on Computation and Society at Harvard School of Engineering and Applied Sciences

BERKMAN CENTER FOR INTERNET & SOCIETY AT HARVARD UNIVERSITY

The Institute for Quantitative Social Science at Harvard University

DATA PRIVACY LAB