

# Synthetic Data Demo Day

Gary Howarth, PhD PSCR, Prize Manager

Christine Task, PhD Knexus Research, Computer Scientist



# DISCLAIMER

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology (NIST), nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

Christine Task of Knexus Research Corporation, produced and presented content, under contact 1333ND21FNB670041 from U.S. Department of Commerce, NIST. Knexus Research Corporation produced the video presented. Posted with Permission. Presentations from guests speakers are representatives of teams winning Development Execution Contest prizes in the 2020 NIST PSCR Temporal Map Challenge. The contents of their presentation do not necessarily reflect the views or policies of the NIST or the U.S. Government.

Please note, unless mentioned in reference to a NIST Publication, all information and data presented is preliminary/in-progress and subject to change.

# Agenda

## **Approximate times (Eastern)**

- 11:00 AM -- Welcome -- NIST PSCR
- 11:10 AM -- Introduction to the Challenge -- Knexus Research
- 11:25 AM -- Benchmark problems for synthetic data -- Sarus Technologies
- 11:30 AM -- Demo 1: Minutemen
- 12:00 PM -- Demo 2: DPSyn
- 12:30 PM -- Demo 3: Jim King
- 1:00 PM -- Open Problems Discussion
- 1:30 PM -- Conclusion

# PSCR Overview

PSCR is the primary federal laboratory conducting research, development, testing, and evaluation for public safety communications technologies.



**NIST**  
**National Institute of  
Standards and Technology**  
U.S. Department of Commerce



**PUBLIC SAFETY  
COMMUNICATIONS  
RESEARCH**  **PSCR**

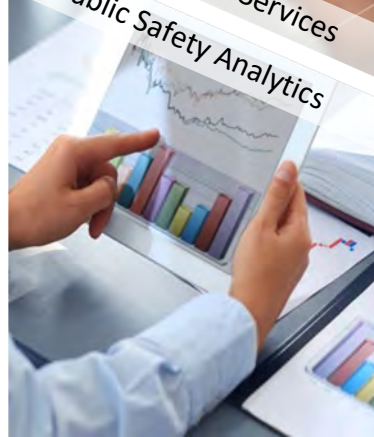
# 5 Key Research Areas



User Interface User Experience  
Mission Critical Voice



Location-Based Services  
Public Safety Analytics



Security  
Resilient Systems



Cross Cutting  
Research Areas



LMR to LTE

# WHAT'S THE PROBLEM?

## Public Safety As Data Generators

- As Public Safety entities make enormous gains in cyber and data infrastructure leading to the routine collection of many large datasets.
- Governments and the public are demanding greater protections on individual privacy and the privacy of individual records.
- Open data initiatives are pushing for the release of more information.

## Public Safety Generates Sensitive Information

- Included in the data is personally identifiable information (PII) for police officers, victims, persons of interest, witnesses, suspects, etc.
- Studies have found that a combination of just 3 “quasi-identifiers” (date of birth, 5 digit postal code, and gender) uniquely identifies 87% of the population.

# DATA COLLECTED BY PUBLIC SAFETY



## Calls for Service

- Calls to “911” for emergency assistance
- May include calls non-emergency calls
- Typically maintained in law enforcement computer-aided dispatch systems



## Incidents

- Collected by an agency for management
- Stored in Records Management Systems (RMS)
- Officer reports on crimes, situations, concerns, suspects, citizen public safety issues, etc.



## Stops, Citations, Arrests

- Proactive and reactive stop of pedestrians or motor vehicles
- May be resolved through warnings, citations, summons, or physical arrests
- Data may be overlapping such as a stop followed by a citation or arrest



## Complaints

- Potential mistreatment by authorities
- Policy, procedure, and legal violations
- May include internal affairs investigations
- Collection process required by national law and accreditation standards

# WHY RELEASE DATA ?



## Analytics

Many cities are developing algorithms to analyze crime, fire, and health data. Developers would like to access other localities' data for training, analysis, and validation.



## Open Access to Data

Many public safety agencies are required to report certain data. Others wish to share data with the public and researchers.

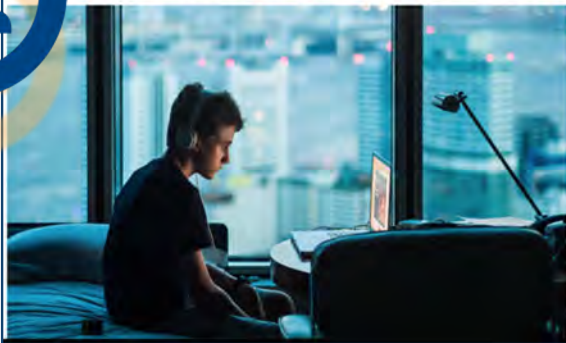


# ATTACKS ON PRIVACY: DE-ANONYMIZATION



## 'Data is a fingerprint': why you aren't as anonymous as you think online

So-called 'anonymous' data can be easily used to identify even you, from our medical records to purchase histories



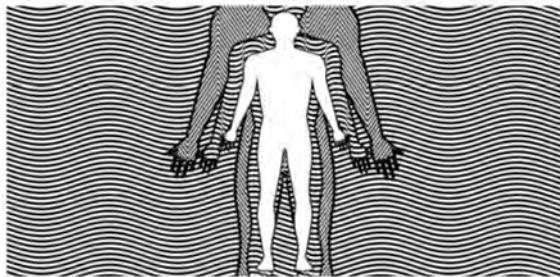
## Keeping Secrets: Anonymous Data Isn't Always Anonymous

March 12, 2014 by [datascience@berkeley Staff](mailto:datascience@berkeley Staff)

12.10.18

## Sorry, your data can still be identified even if it's anonymized

Urban planners and researchers at MIT found that it's shockingly easy to "reidentify" the anonymous data that people generate all day, every day in cities.



ars TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE STG

POLICY —

## "Anonymized" data really isn't—and here's why not

Companies continue to store and sometimes release vast databases of " ...

DATE ANDERSON - 10/10/2018, 7:25 AM



# DE-ANONYMIZATION NEW YORK TAXI DATA

“Using a simulation of the medallion data, we show that our attack can re-identify over 91% of the taxis that ply in NYC even when using a perfect pseudonymization of medallion numbers.”

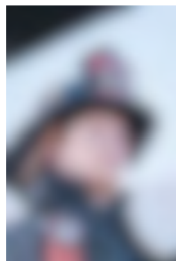
Douriez, Marie, et al. "Anonymizing nyc taxi data: Does it matter?." 2016 *IEEE international conference on data science and advanced analytics (DSAA)*. IEEE, 2016.



## WHAT DO WE MEAN BY PRIVACY?

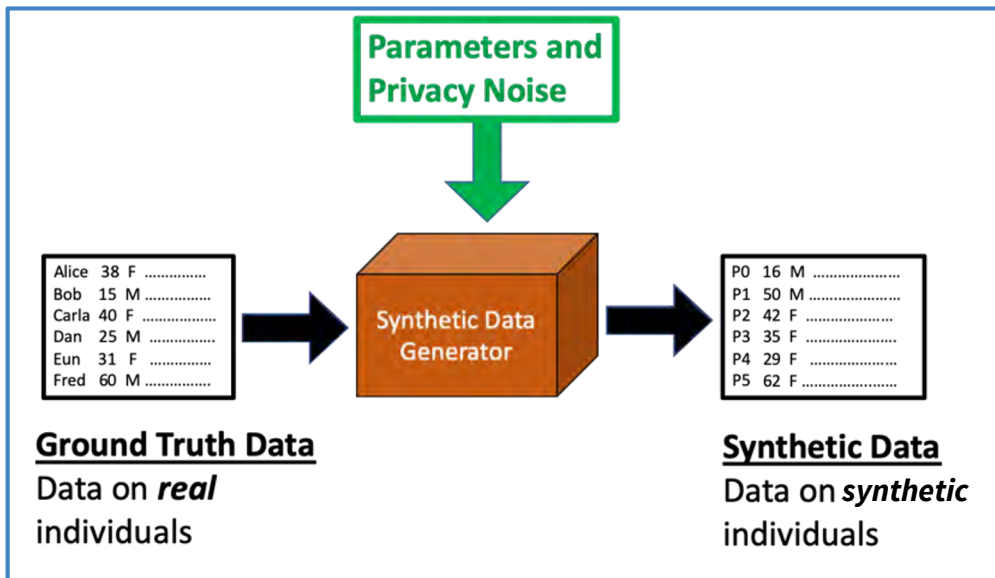
Privacy-preserving data-mining algorithms allow trusted data-owners to release useful, aggregate information about their data-sets (such as common user behavior patterns) while at the same time protecting individual-level information.

Intuitively, the concept of making large patterns visible while protecting small details makes sense. You just 'blur' things a bit:



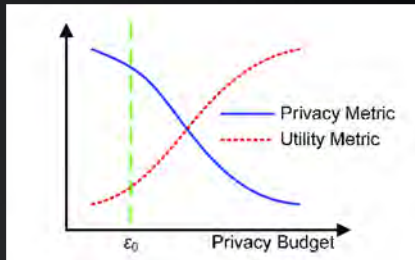
If we refine this idea into a mathematically formal definition, we can create a standard for individual privacy.

# TARGET APPLICATION: DIFFERENTIALLY PRIVATE SYNTHETIC DATA



# How Private is Private?

- Differentially private synthetic data works by learning a model of the private data and then using that model to generate a new data set in the same distribution as the private data.
- To guarantee privacy, randomness is injected into the modeling process. For the demos today, all the winning contestants used noisy marginal counts as input for their models. The more noise added, the more remote the model is from the original data, providing more privacy.
- The amount of noise added to the dataset is characterized by epsilon ( $\epsilon$ ). The lower the  $\epsilon$ , the more noise is added, producing less precise data. With  $\epsilon = \infty$ , no noise is added.
- The value of  $\epsilon$  must be selected heuristically, balancing the risk of privacy against the utility of the data. Well engineered solutions provide good utility even at lower values of  $\epsilon$ .



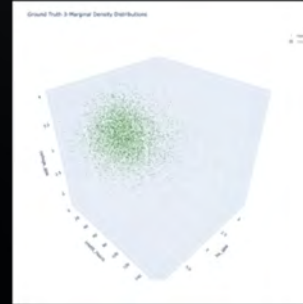
*Illustration of the privacy-utility tradeoff.  
From Liu et al. "Privacy-Preserving Monotonicity of  
Differential Privacy Mechanisms." 2018.*

$\epsilon$ in The Wild	
2020 Census demo data (people)	10.3
2020 Census demo data (housing)	1.9
Apple emoji prediction	4

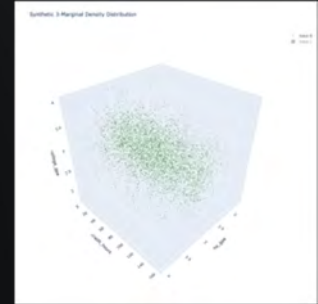
# What Does it Mean to Provide Good Utility?

- Synthetic data which closely mimics the distribution of the private ground truth data can be used in place of the original data for population level analyses. This data has “high utility”.
- The above is vague--we'll make it more concrete:
  - Measuring distributional similarity in high dimensional data (ie, larger schemas) is challenging; across sufficient dimensions, the space of *possible* records dwarfs the set of *actual* records, and direct comparison isn't meaningful.
  - To address this, we use 3-marginal evaluation, along with other metrics. This uses many overlapping 3-dimensional snapshots of the data to get an overall measurement of how the distributions differ, across all combinations of features.

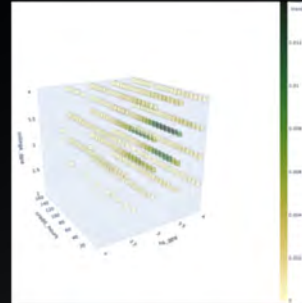
**Ground Truth**



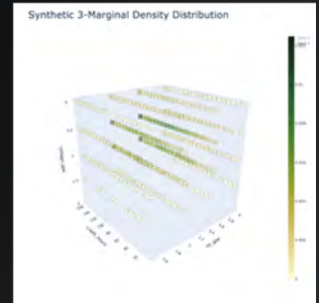
**Synthetic**



**Binned Ground Truth**



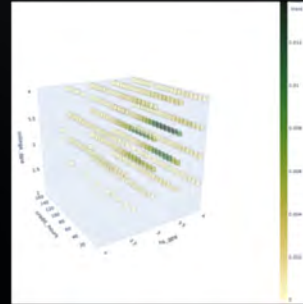
**Binned Synthetic**



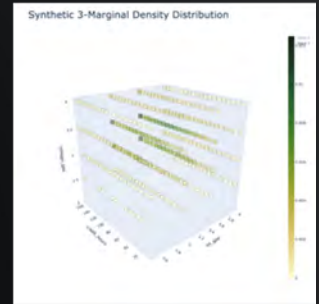
# What Does it Mean to Provide Good Utility?

- Synthetic data which closely mimics the distribution of the private ground truth data can be used in place of the original data for population level analyses. This data has “high utility”.
- The above is vague--we’ll make it more concrete:
  - Measuring distributional similarity in high dimensional data (ie, larger schemas) is challenging; across sufficient dimensions, the space of *possible* records dwarfs the set of *actual* records, and direct comparison isn’t meaningful.
  - To address this, we use 3-marginal evaluation, along with other metrics. This uses many overlapping 3-dimensional snapshots of the data to get an overall measurement of how the distributions differ, across all combinations of features.

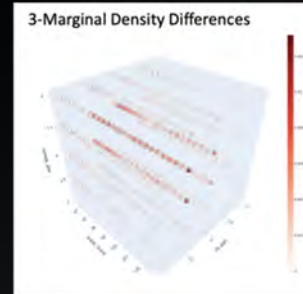
Binned Ground Truth



Binned Synthetic



Absolute Edit Distance



Bin Number	Real Density	Synthetic Density	Difference	Absolute Value
1	0.10	0.08	0.02	0.02
2	0.11	0.10	0.01	0.01
3	0.16	0.17	-0.01	0.01
4	0.15	0.18	-0.03	0.03
5	0.09	0.08	0.01	0.01
6	0.08	0.10	-0.02	0.02
7	0.14	0.13	0.01	0.01
8	0.17	0.16	0.01	0.01
SUM				0.12

Compute the absolute difference between the two density distributions



# What Does it Mean to Provide Good Utility?

- Synthetic data which closely mimics the distribution of the private ground truth data can be used in place of the original data for population level analyses. This data has “high utility”.
- The above is vague--we'll make it more concrete:
  - Measuring distributional similarity in high dimensional data (ie, larger schemas) is challenging; across sufficient dimensions, the space of *possible* records dwarfs the set of *actual* records, and direct comparison isn't meaningful.
  - To address this, we use 3-marginal evaluation, along with other metrics. This uses many overlapping 3-dimensional snapshots of the data to get an overall measurement of how the distributions differ, across all combinations of features.

**Repeat over very many, overlapping, randomly selected marginal tests**





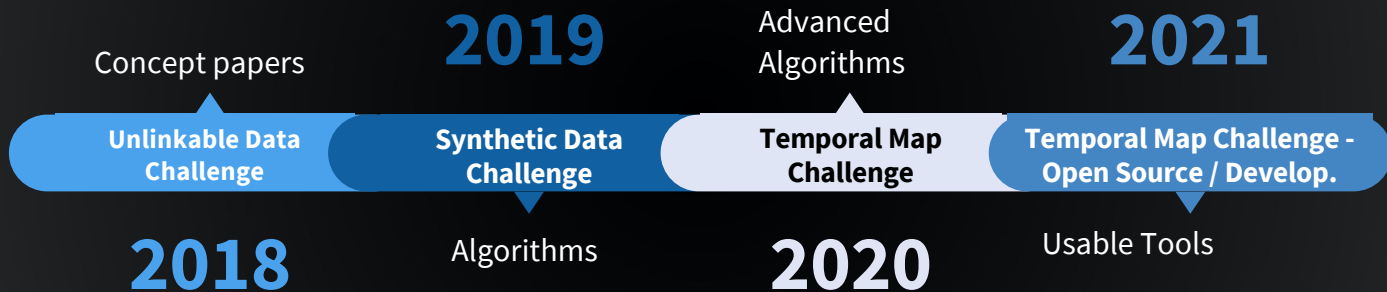
# What Does it Mean to Provide Good Utility?

- Synthetic data which closely mimics the distribution of the private ground truth data can be used in place of the original data for population level analyses. This data has high utility.
- The above statement is vague--we'll make it more concrete:
  - 3-marginal evaluation was developed by Dr. Sergey Pogodin for the 2018 NIST challenge but has since been used in other synthetic data research projects.
  - As a benchmark for what comprises “high” utility, we use our evaluation metrics to compare a 40% subsample of the data to the original data distribution. Here the difference between the two distributions is induced by sampling error, rather than modeling or added noise.
  - When synthetic data scores similarly to the subsample benchmark on the same data, **the error induced by the process of generating synthetic individuals is comparable to the error induced by subsampling real ones.**

# What Does it Mean to Provide Good Utility?

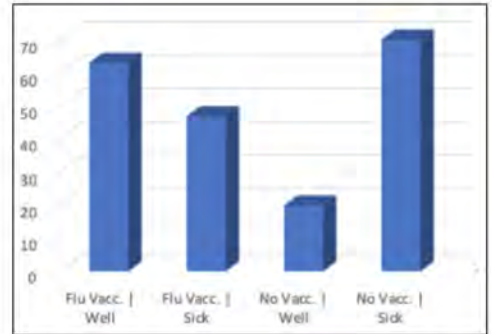
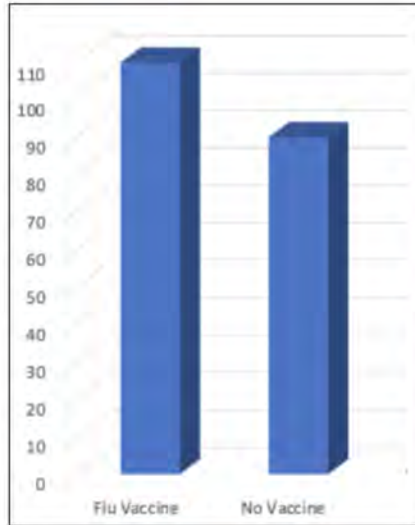
- Synthetic data which closely mimics the distribution of the private ground truth data can be used in place of the original data for population level analyses. This data has high utility.
- If epsilon is large enough (ex.  $\epsilon > 20$ ), and/or the data schema is very small ( $< 5$  features), any reasonable algorithm should produce high utility data.
- However, with larger schemas the data is spread more sparsely through the data space. And at smaller epsilons, more noise is added across the data space, potentially effectively burying the data. In these conditions, many differentially private synthetic data generation algorithms perform very poorly.
- Well engineered differentially private systems can improve these trade-offs, producing results with better utility at lower values of epsilon. The objective of the NIST Synthetic Data Challenges was to discover if it was possible to produce differentially private synthetic data with good utility over realistically large, complex real world data sets, at values of  $\epsilon \leq 10$ .
- Essentially--*Could we get this to work in practice?*

# Differential Prize Challenge Series



# WHY IS THE TEMPORAL MAP PROBLEM DIFFICULT TO SOLVE?

*Problem size and complexity increase with amount of information shared and number of map locations*

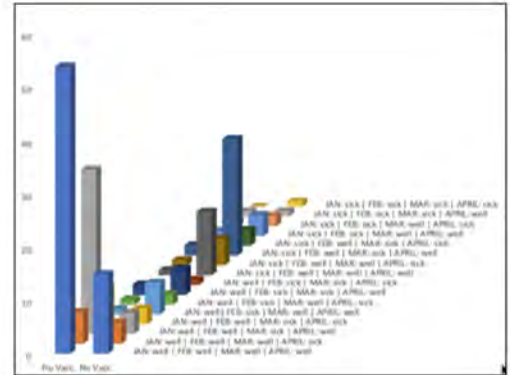
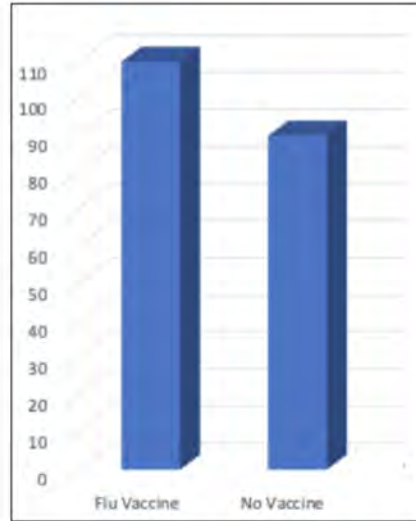


# WHY IS THE TEMPORAL MAP PROBLEM DIFFICULT TO SOLVE?

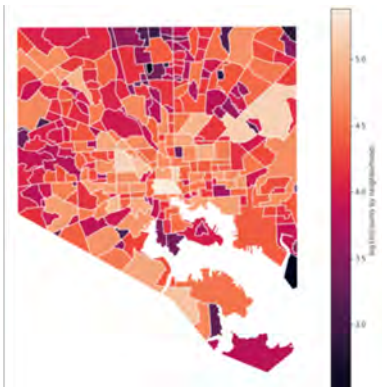
## PSCR Better Meter Stick Contest

*Problem size and complexity increase with amount of information shared and number of map locations*

*Problem size and complexity increase exponentially with number of time steps (per individual).*



# Algorithm Contest: 3 Sprints

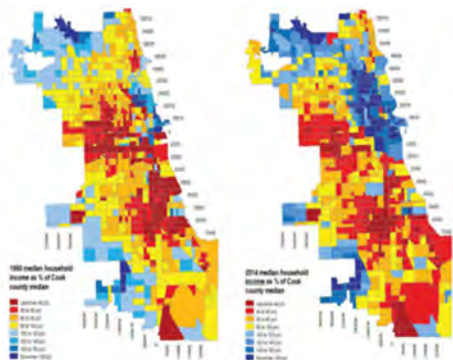


## Sprint 1

Baltimore 911 Incidents

Oct 1 - Dec 11 2020

[Problem Description](#) | [Results](#)



## Sprint 2

American Community Survey

Jan 6 - Mar 23 2021

[Problem Description](#) | [Results](#)



## Sprint 3

Chicago Taxi Rides

Mar 29 - June 16 2021

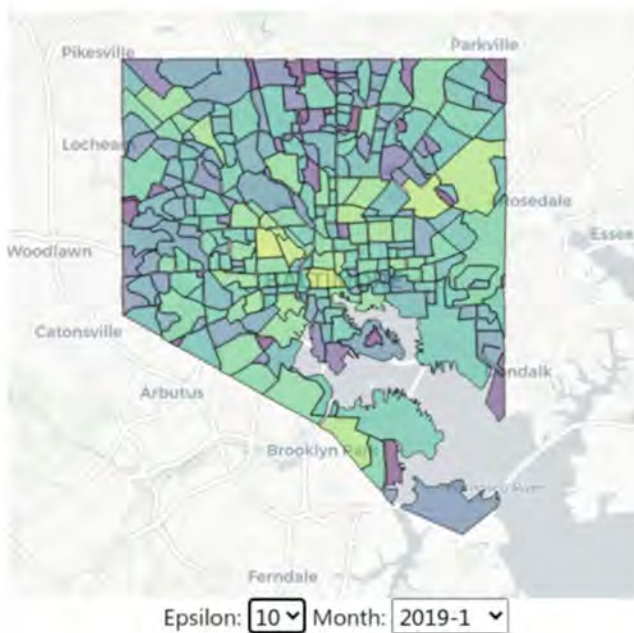
[Problem Description](#) | [Results](#)

# Algorithm Contest: Sprint overview

	Sprint 1	Sprint 2	Sprint 3
<b>Data</b>	Baltimore 911 Incidents	American Community Survey (ACS)	Chicago Taxi Rides
<b>Map segments</b>	Neighborhoods (278)	PUMAs (varies)	Community Areas (78)
<b>Time segments</b>	Months (12)	Years (7)	8-hour shifts (21)
<b>Max records per individual</b>	20	7	200
<b>Number of variables</b>	4	35	13
<b>Max values per variable</b>	278 (neighborhoods)	181 (PUMAs)	78 (community areas)
<b>Scoring data</b>	Public: 2019 Final: 2016, 2018	Public: IL, OH Final: NY-PA, NC-SC-GA	Public: 2019 Prescreened: 2018 Final: 2016, 2020
<b>Values of epsilon</b>	10, 2, 1	10, 1, 0.1	10, 1
<b>Metric</b>	<a href="#">Pie chart</a>	<a href="#">K-marginal w/ bias</a>	<a href="#">K-marginal, graph-edge &amp; HOC</a>

# APPROACHES AND RESULTS

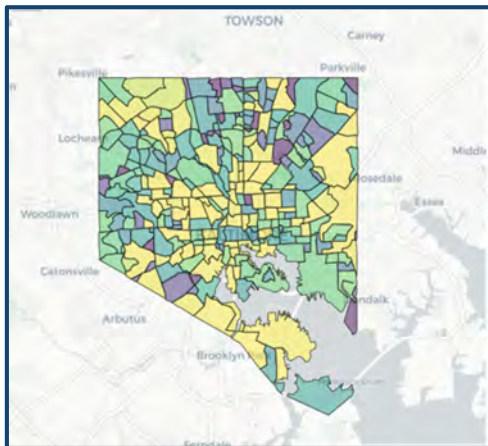
The **Interactive Map** allows you to see your scores geographically (across all map segments). Here we see that dense urban neighborhoods closer to the city center, which generally contain more records, have better scores than rural and suburban neighborhoods where records may be more sparse. These are challenges that will need to be creatively overcome to achieve good performance on the Sprint 1 task.





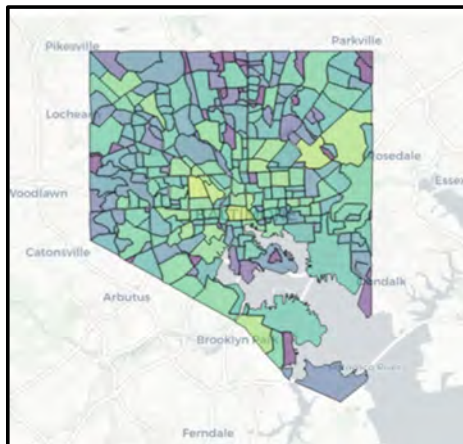
# APPROACHES AND RESULTS

Baltimore 911:  
1st Place Winner



**Data with  
poor utility**

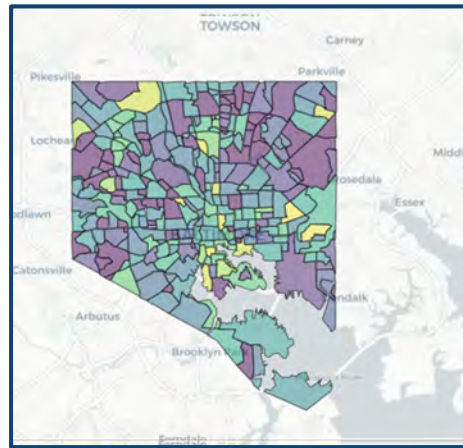
Baltimore 911:  
50% Subsample  
Benchmark



0.0



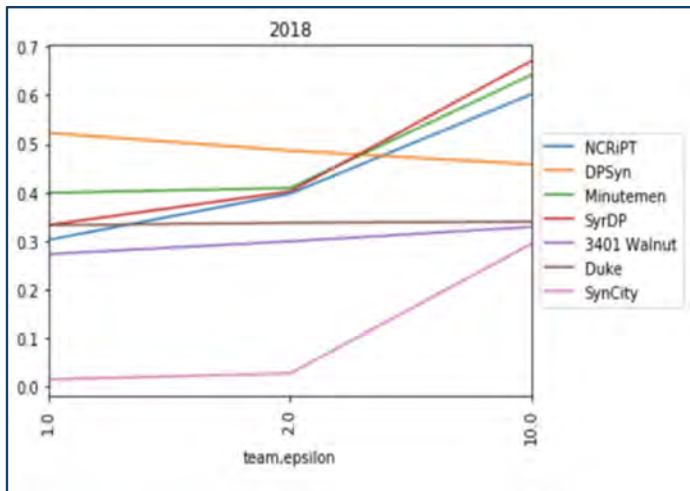
Baltimore 911:  
5th Place Winner



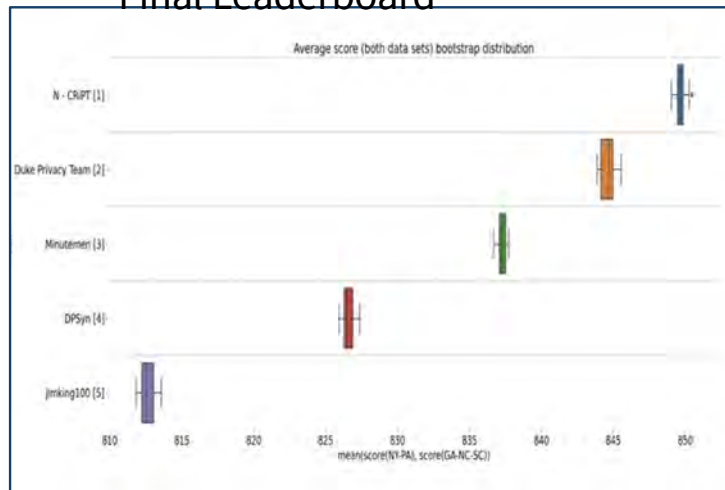
**Data with  
more utility**

# APPROACHES AND RESULTS

## Baltimore 911 Final Leaderboard

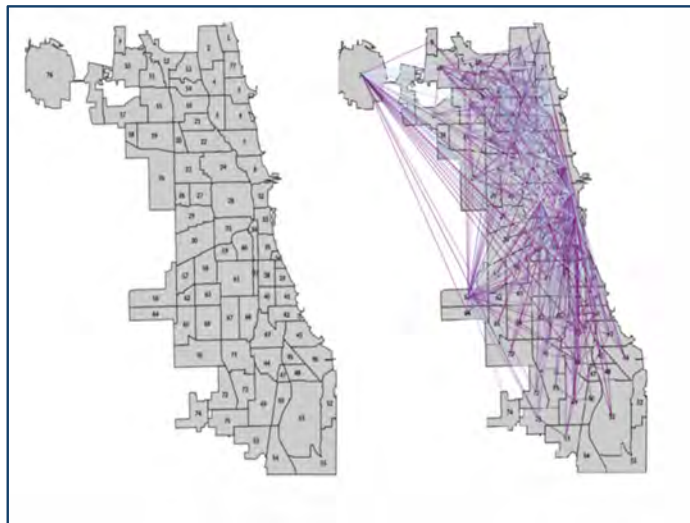


## American Community Survey Final Leaderboard



# APPROACHES AND RESULTS: Final Sprint

- **taxi\_id** (int) — Simulated individual ID for each taxi, derived from taxi-weeks in the original data.
- **shift** (int) — Computed variable combining the above two variables into 21 eight-hour segments. Each day has three shifts: "night" (20:00-4:00), "morning" (4:00-12:00), and "afternoon" (12:00-20:00). A "night" shift is contiguous; that is, it includes the last four hours of one day and the first four hours of the following day (e.g. Friday night includes late Friday night and early Saturday morning).
- **company\_id** (int) — Company that the taxi works for.
- **pickup\_community\_area** (int) — **Community Area** where trip started.
- **dropoff\_community\_area** (int) — **Community Area** where trip ended.
- **payment\_type** (int) — Type of payment used.
- **trip\_day\_of\_week** (int) — Day of the week that the trip occurred (0 is Monday).
- **trip\_hour\_of\_day** (int) — Hour of the day that the trip occurred on a 24-hour clock.
- **fare** (int) — Fare paid to nearest dollar.
- **tips** (int) — Tips paid to nearest dollar.
- **trip\_total** (int) — Total amount paid for the trip.
- **trip\_seconds** (int) — Total duration of the trip to the nearest second.
- **trip\_miles** (int) — Total length of the trip to the nearest mile.

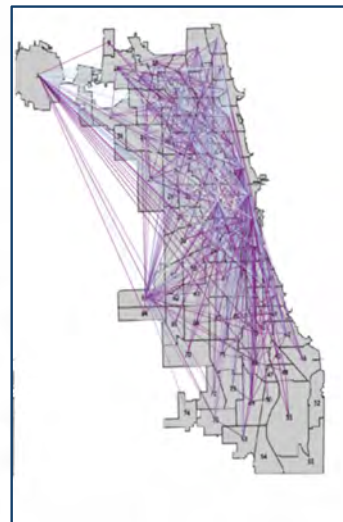


## APPROACHES AND RESULTS: Final Sprint

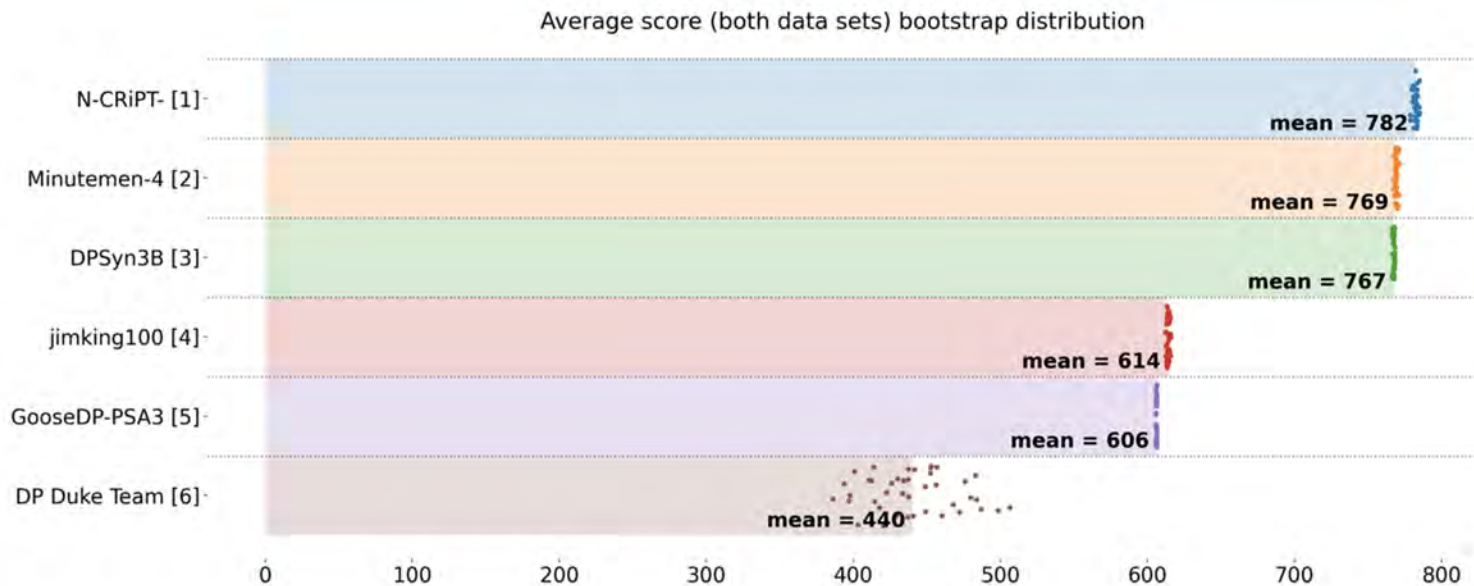
To assess how well synthetic records preserved meaningful patterns like this from the original data, solutions were evaluated for data utility at different levels of guaranteed privacy (indicated by epsilon). Aspects of the scoring approach were informed by the [metrics contest](#) run as part of the broader Differential Privacy Temporal Map Challenge. In this sprint, solutions were [evaluated](#) on three different measures of utility, each contributing equally to the total score:

- K-marginal feature similarity reflected how well the **distribution of trip features** was preserved for each and every community area (even ones that used taxis relatively rarely).
- Graph-edge map similarity reflected how well **location connections** between pickup and dropoff locations were conserved.
- Individual higher-order conjunction reflected how well **types of individual taxis** were conserved. Did the synthetic taxi drivers have the same patterns of shift work and driving areas as the original taxis?

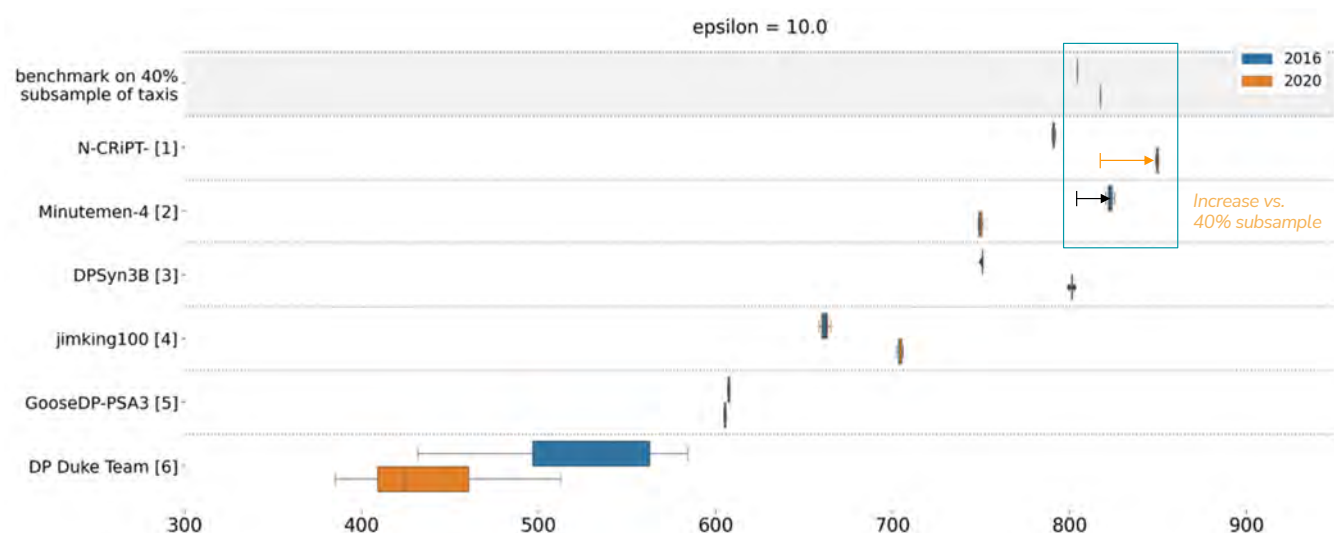
To produce final scores, algorithms were run on two new data sets representing taxi rides in 2016 and 2020. 2016 preceded the dramatic rise in popularity of ride sharing apps like Uber and Lyft. 2020 saw major shifts both in work and social life with COVID-19. Compared with the public data from 2019 that was provided for competitors to use while they were developing their algorithms, these final scoring data sets **demand robust solutions that could handle significant differences in data distribution.**



# APPROACHES AND RESULTS: Final Sprint



# Final scores vs. subsampling benchmark (eps 10): Sprint 3



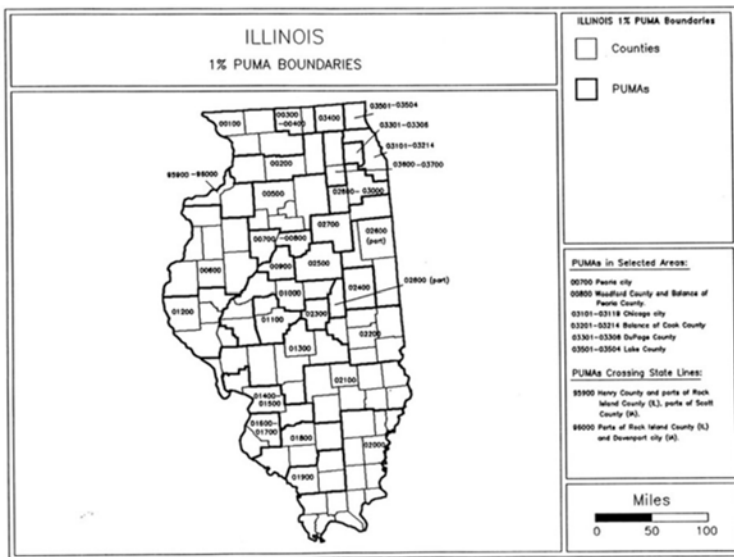
Graph reflects final score distribution for average runs per final scoring data sets (2016 and 2020) for epsilon = 10, for all validated teams from final scoring. Benchmark reflects 40% random subsample of taxis from each data set (note: this is for benchmarking purposes, these do NOT reflect differentially private solutions at epsilon = 10).<sup>32</sup>

# APPROACHES AND RESULTS

We beat the sampling error benchmark on the small problem in Sprint 1, and the much larger problem in Sprint 3. Did we manage it in Sprint 2, on American Community Survey data?

Not during the sprint itself, although we got promisingly close, and the final solutions with further tuning might very well make it across the bar.

So let's put that one up to Future Work. And if you'd like to join us in that future work, testing out your own approach and ideas, we're working to make it much easier for you to do so.





# sarus & NIST collaboration

**Sarus** is a French company building tools for privacy-preserving ML and data analysis. Synthetic data is at the heart of Sarus's offer as it improves user experience and helps save privacy.

Collaboration objective:

Transform the 2020 Temporal map synthetic data challenge into a standard benchmark for synthetic data generation with differential privacy constraints

- Ease access to challenge datasets and standardize workflows
- Ease scoring of new methods and comparison with baselines
- Improve reproducibility of challenge submissions
- Investigate algorithms' performances



**sarus**  
technologies

**NIST**



# sarus & NIST collaboration

1. Access the challenge data as a `pd.DataFrame`:

```
% pip install sdnist
```

```
>>> public_data, schema = sdnist.census(public=True)
```

2. Write your own synthetic data generator

```
>>> synthesizer = MySynthesizer()
```

3. Compute the score using the metric of the challenge:

```
>>> score = sdnist.run(synthesizer, challenge="census", eps=1)
```

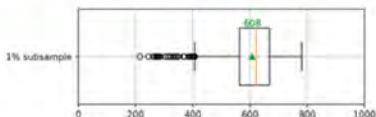
```
>>> score
```

```
CensusKMarginalScore(608/1000)
```

# sarus & NIST collaboration

4. Investigate:

```
>>> score.map()  
>>> score.boxplot()
```



- + Compare your solution to the challenge winners
- + Additional metrics that better capture individual consistency
- + Empirical evaluation of privacy guarantees
- + Interoperability with other benchmarking frameworks (e.g sdgym)

Feel free to reach out to us if you are interested! [gl@sarus.tech](mailto:gl@sarus.tech)

## CAVEATS and TUNING

Our three teams that participated in the Open Development Contest will now give you tutorials on their approaches.

We've done our best to make these as easy to use in real world data contexts as possible, so we have a few caveats:

- Assume Public Schema, can be learned directly from data without loss of privacy
- Public training data optional
- Map/Sequence data optional (sensitivity must be set to max records per individual for sequence data)

When you go to configure these solutions yourself, some tips to keep in mind:

1. Begin with an initial configuration, and public test data\*
2. Then use evaluation tools in the repos to see how well you perform and where problem areas are.
3. Revisit your configuration to improve your performance.

# Adaptive Granularity Mechanism from Minutemen (UMass Amherst)

Point of contact:

Ryan McKenna <rmckenna@umass.edu>

Cite:

McKenna, R., Wu, J., Tajima, A., Mullins, B., Ferrando, C., &  
Pradhan, S. (2021). Adaptive Granularity Mechanism

[Computer software]. <https://github.com/ryan112358/nist-synthetic-data-2021>

---

# DPSyn: An algorithm for synthesizing microdata for data analysis while satisfying differential

Point of contact:

Tianhao Wang <[tianhao@virginia.edu](mailto:tianhao@virginia.edu)>

Cite:

Chen A., Li N., Li Z., Wang T. (2021). DPSyn: An algorithm for synthesizing microdata for data analysis while satisfying differential privacy (version 1.0). URL: [https://github.com/agl-c/deid2\\_dpsyn](https://github.com/agl-c/deid2_dpsyn)

---

# Private Histograms

Point of contact:

Jim King <[jim.king.mv@gmail.com](mailto:jim.king.mv@gmail.com)>

Cite:

King, J. (2021). Privitized Histograms (Version 1.0.0)  
[Computer software].

<https://github.com/JimKing100/PrivacyHistos>

---

# Pursuing a Formal Understanding of How These Work

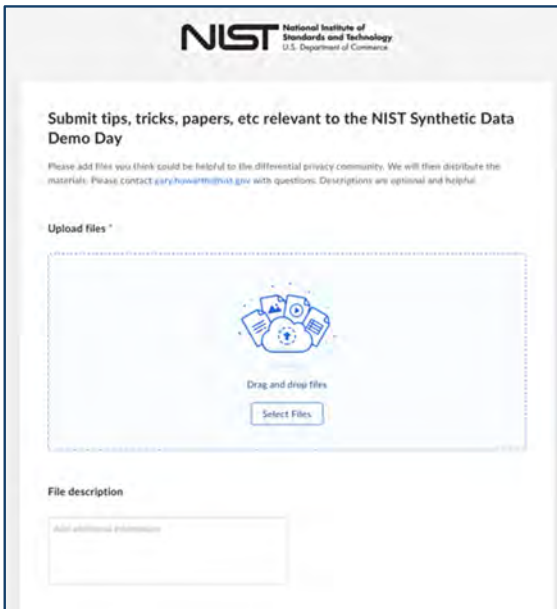
We knew it was easy to do differentially private synthetic data well on very small schemas, with just a few possible record values.

What was most surprising from these challenges is that we were able to well on large data schemas with many variables and very many possible record values (and map segments). Potentially these large problems are smaller than they initially seem, if you approach them correctly.

By and large this work was done through empirical hypothesis checking at the frenetic pace of the challenge. What we'd like to help support going forwards is more formal mathematical analysis into the characteristics of data sets that impact algorithm performance, how we can leverage them to produce new and better algorithms with not just provable privacy, but also *provable tighter bounds on utility for real world data sets*. We want algorithms that both perform well in practice and are very well understood.

We'll now briefly introduce three types of data-dependent utility improving tricks that our competitors used. The first has to do with handling temporal data, the second has to do with constraints in the data, and the last one is possibly the most interesting, it has to do with patterns of correlations between variables in the data. *We know they work*. Everyone used them, on very diverse sets of human data (event data, demographic surveys, location data). *But there's a lot we currently don't know about \*how\* they work*.

Those of you who have data yourselves, think about where these properties might occur in your data. [If you have public anonymized data, please feel free to drop us a line here at our submission box with your thoughts](#) on the properties of your data, and a link to access it. We'll incorporate that in our work and make it available to researchers.



The image shows a web form for submitting tips, tricks, papers, etc. relevant to the NIST Synthetic Data Demo Day. The form is titled "Submit tips, tricks, papers, etc relevant to the NIST Synthetic Data Demo Day" and includes a "File description" section. The NIST logo is at the top right. The form has a "Submit" button and a "Select Files" button. The "File description" section has a text area for "Add additional information".

NIST National Institute of Standards and Technology  
U.S. Department of Commerce

Submit tips, tricks, papers, etc relevant to the NIST Synthetic Data Demo Day

Please add files you think could be helpful to the differential privacy community. We will then distribute the materials. Please contact [gary.howarth@nist.gov](mailto:gary.howarth@nist.gov) with questions. Descriptions are optional and helpful.

Upload files \*

Drag and drop files

Select Files

File description

Add additional information

# Pursuing a Formal Understanding of How These Work

## **Subsampling/clipping and Reweighting**

This is a type of data pre-processing that makes privatizing sequence data feasible.

### **How does it help?**

It reduces the sensitivity of the query (reducing the amount of added noise required) by bounding the maximum amount each individual can contribute to the data.

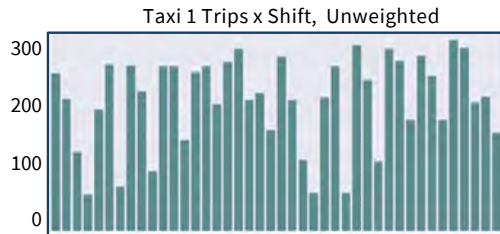
### **Why is it interesting?**

Taken trivially, these techniques shouldn't work. They reduce the amount of added noise required for each query, at the expense of also reducing the amount of data going to the query. If I cut both the added noise and the data in half, the relative error (noise vs signal) stays the same.

But there's two factors to keep in mind. First, different individuals contribute different amounts of information (different length sequences), so in practice clipping and reweighting don't reduce the data evenly-- the signal doesn't shrink uniformly at the same rate as the noise. Second, lowering sensitivity decreases the additive noise amounts (even if the relative error stays the same), which may be helpful for some algorithms.

### **So here's a question--**

Do different individuals in your data contribute different amounts of information? Even if they all contribute the same number of records, does their information have different levels of complexity (ie, an individual who does the same thing every day, vs. someone whose pattern is more complex, higher entropy)? Do your individuals fall into different categories/types, with different behavior? What types of assertions can we make about the distribution of data on an individual level? These will enable us to formally analyze the impact of subsampling/clipping and reweighting preprocessing.





# Pursuing a Formal Understanding of How These Work

## **Subsampling/clipping and Reweighting**

This is a type of data pre-processing that makes privatizing sequence data feasible.

### **How does it help?**

It reduces the sensitivity of the query (reducing the amount of added noise required) by bounding the maximum amount each individual can contribute to the data.

### **Why is it interesting?**

Taken trivially, these techniques shouldn't work. They reduce the amount of added noise required for each query, at the expense of also reducing the amount of data going to the query. If I cut both the added noise and the data in half, the relative error (noise vs signal) stays the same.

But there's two factors to keep in mind. First, different individuals contribute different amounts of information (different length sequences), so in practice clipping and reweighting don't reduce the data evenly-- the signal doesn't shrink uniformly at the same rate as the noise. Second, lowering sensitivity decreases the additive noise amounts (even if the relative error stays the same), which may be helpful for some algorithms.

### **So here's a question--**

Do different individuals in your data contribute different amounts of information? Even if they all contribute the same number of records, does their information have different levels of complexity (ie, an individual who does the same thing every day, vs. someone whose pattern is more complex, higher entropy)? Do your individuals fall into different categories/types, with different behavior? What types of assertions can we make about the distribution of data on an individual level? These will enable us to formally analyze the impact of subsampling/clipping and reweighting preprocessing.



# Pursuing a Formal Understanding of How These Work

## **Subsampling/clipping and Reweighting**

This is a type of data pre-processing that makes privatizing sequence data feasible.

### **How does it help?**

It reduces the sensitivity of the query (reducing the amount of added noise required) by bounding the maximum amount each individual can contribute to the data.

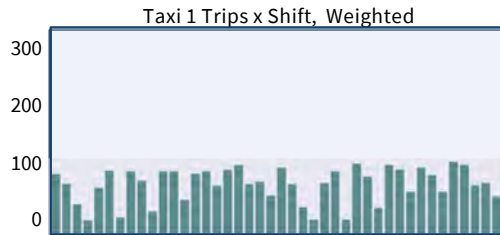
### **Why is it interesting?**

Taken trivially, these techniques shouldn't work. They reduce the amount of added noise required for each query, at the expense of also reducing the amount of data going to the query. If I cut both the added noise and the data in half, the relative error (noise vs signal) stays the same.

But there's two factors to keep in mind. First, different individuals contribute different amounts of information (different length sequences), so in practice clipping and reweighting don't reduce the data evenly-- the signal doesn't shrink uniformly at the same rate as the noise. Second, lowering sensitivity decreases the additive noise amounts (even if the relative error stays the same), which may be helpful for some algorithms.

### **So here's a question--**

Do different individuals in your data contribute different amounts of information? Even if they all contribute the same number of records, does their information have different levels of complexity (ie, an individual who does the same thing every day, vs. someone whose pattern is more complex, higher entropy)? Do your individuals fall into different categories/types, with different behavior? What types of assertions can we make about the distribution of data on an individual level? These will enable us to formally analyze the impact of subsampling/clipping and reweighting preprocessing.



### **Relative Error for Counting Queries**

$$\frac{\text{var}(\text{Lap}(\text{scale}/2))}{(n/2)} = \frac{\text{var}(\text{Lap}(\text{scale}))}{n}$$

# Pursuing a Formal Understanding of How These Work

## Subsampling/clipping and Reweighting

This is a type of data pre-processing that makes privatizing sequence data feasible.

### How does it help?

It reduces the sensitivity of the query (reducing the amount of added noise required) by bounding the maximum amount each individual can contribute to the data.

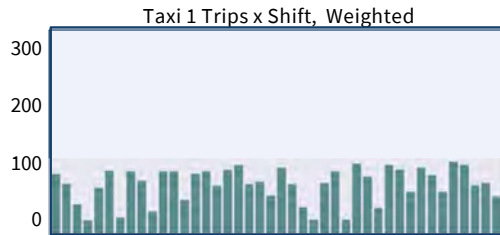
### Why is it interesting?

Taken trivially, these techniques shouldn't work. They reduce the amount of added noise required for each query, at the expense of also reducing the amount of data going to the query. If I cut both the added noise and the data in half, the relative error (noise vs signal) stays the same.

But there's two factors to keep in mind. First, different individuals contribute different amounts of information (different length sequences), so in practice clipping and reweighting don't reduce the data evenly-- the signal doesn't shrink uniformly at the same rate as the noise. Second, lowering sensitivity decreases the additive noise amounts (even if the relative error stays the same), which may be helpful for some algorithms.

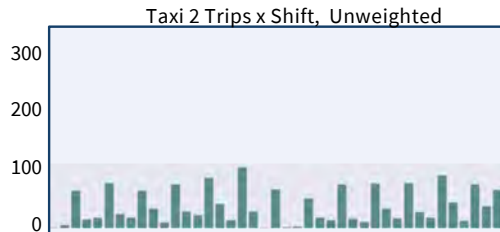
### So here's a question--

Do different individuals in your data contribute different amounts of information? Even if they all contribute the same number of records, does their information have different levels of complexity (ie, an individual who does the same thing every day, vs. someone whose pattern is more complex, higher entropy)? Do your individuals fall into different categories/types, with different behavior? What types of assertions can we make about the distribution of data on an individual level? These will enable us to formally analyze the impact of subsampling/clipping and reweighting preprocessing.



### Relative Error for Counting Queries

$$\frac{\text{var}(\text{Lap}(\text{scale}/2))}{(n/2)} = \frac{\text{var}(\text{Lap}(\text{scale}))}{n}$$



# Pursuing a Formal Understanding of How These Work

## Public Hard Constraints and Public Soft Constraints

This is a way of reducing the data space by taking into account information beyond the basic data schema.

### How does it help?

It reduces the impact of added noise by using public knowledge to effectively cross out parts of the data space where data either can't live, or is very unlikely to live. It turns a big problem into a much smaller one.

### Why is it interesting?

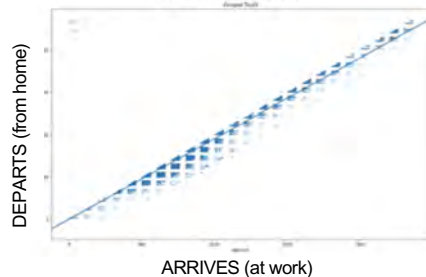
First, because it's powerful, often resulting in an exponential decrease of the data space. We can incorporate it in preprocessing, post-processing (removing unrealistic points), or by redefining the schema to disallow generation of impossible points. However, this ends up requiring a fair amount of bespoke data engineering, the most time consuming and least fun part of working with data. So the second point is: "How can we incorporate this information more gracefully?" Both in terms of algorithm configuration, and for formal analysis on utility bounds (computing expected utility as a function of the space the data actually occupies, rather than the full schema space).

### So here's a question--

Where do constraints like this exist in your data? What different forms do they take? (ie, where are there records that would be possible in your schema, but unambiguously absurd in reality?) A diverse collection of real world examples will help us consider how we might create a simple, efficient representation format that can provide more information for algorithm developers and researchers. This will support the development of algorithms and utility analysis that take these constraints up front, rather than as an afterthought.

## Arrives vs. Departs

### Ground Truth



# Pursuing a Formal Understanding of How These Work

## Public Hard Constraints and Public Soft Constraints

This is a way of reducing the data space by taking into account information beyond the basic data schema.

### How does it help?

It reduces the impact of added noise by using public knowledge to effectively cross out parts of the data space where data either can't live, or is very unlikely to live. It turns a big problem into a much smaller one.

### Why is it interesting?

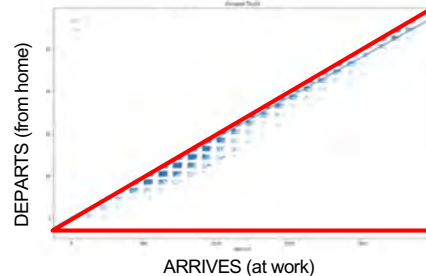
First, because it's powerful, often resulting in an exponential decrease of the data space. We can incorporate it in preprocessing, post-processing (removing unrealistic points), or by redefining the schema to disallow generation of impossible points. However, this ends up requiring a fair amount of bespoke data engineering, the most time consuming and least fun part of working with data. So the second point is: "How can we incorporate this information more gracefully?" Both in terms of algorithm configuration, and for formal analysis on utility bounds (computing expected utility as a function of the space the data actually occupies, rather than the full schema space).

### So here's a question--

Where do constraints like this exist in your data? What different forms do they take? (ie, where are there records that would be possible in your schema, but unambiguously absurd in reality?) A diverse collection of real world examples will help us consider how we might create a simple, efficient representation format that can provide more information for algorithm developers and researchers. This will support the development of algorithms and utility analysis that take these constraints up front, rather than as an afterthought.

## Arrives vs. Departs

### Ground Truth



### Hard Public Constraint:

In the schema ARRIVE and DEPART are simple time variables on a 24hr clock (with roughly  $150^2$  possible combined values). However, if we insist that synthetic records can't arrive at work before they depart from home, it reduces the data space by a factor of 2, with no loss in privacy or accuracy.

# Pursuing a Formal Understanding of How These Work

## Public Hard Constraints and Public Soft Constraints

This is a way of reducing the data space by taking into account information beyond the basic data schema.

### How does it help?

It reduces the impact of added noise by using public knowledge to effectively cross out parts of the data space where data either can't live, or is very unlikely to live. It turns a big problem into a much smaller one.

### Why is it interesting?

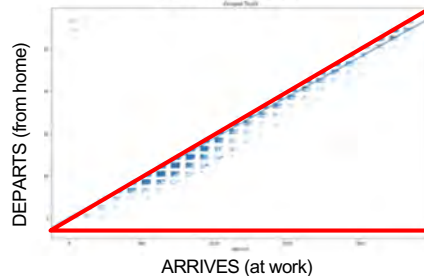
First, because it's powerful, often resulting in an exponential decrease of the data space. We can incorporate it in preprocessing, post-processing (removing unrealistic points), or by redefining the schema to disallow generation of impossible points. However, this ends up requiring a fair amount of bespoke data engineering, the most time consuming and least fun part of working with data. So the second point is: "How can we incorporate this information more gracefully?" Both in terms of algorithm configuration, and for formal analysis on utility bounds (computing expected utility as a function of the space the data actually occupies, rather than the full schema space).

### So here's a question--

Where do constraints like this exist in your data? What different forms do they take? (ie, where are there records that would be possible in your schema, but unambiguously absurd in reality?) A diverse collection of real world examples will help us consider how we might create a simple, efficient representation format that can provide more information for algorithm developers and researchers. This will support the development of algorithms and utility analysis that take these constraints up front, rather than as an afterthought.

## Arrives vs. Departs

### Ground Truth



### Soft Public Constraint:

Note that most of our new reduced space is still empty. That's because most folks don't have 12 hour commutes, but our data space still includes those points.

# Pursuing a Formal Understanding of How These Work

## Public Hard Constraints and Public Soft Constraints

This is a way of reducing the data space by taking into account information beyond the basic data schema.

### How does it help?

It reduces the impact of added noise by using public knowledge to effectively cross out parts of the data space where data either can't live, or is very unlikely to live. It turns a big problem into a much smaller one.

### Why is it interesting?

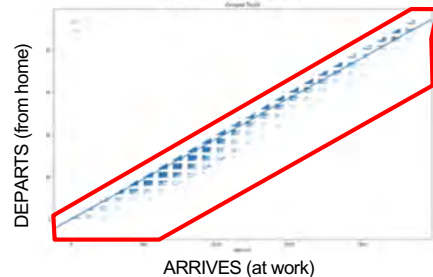
First, because it's powerful, often resulting in an exponential decrease of the data space. We can incorporate it in preprocessing, post-processing (removing unrealistic points), or by redefining the schema to disallow generation of impossible points. However, this ends up requiring a fair amount of bespoke data engineering, the most time consuming and least fun part of working with data. So the second point is: "How can we incorporate this information more gracefully?" Both in terms of algorithm configuration, and for formal analysis on utility bounds (computing expected utility as a function of the space the data actually occupies, rather than the full schema space).

### So here's a question--

Where do constraints like this exist in your data? What different forms do they take? (ie, where are there records that would be possible in your schema, but unambiguously absurd in reality?) A diverse collection of real world examples will help us consider how we might create a simple, efficient representation format that can provide more information for algorithm developers and researchers. This will support the development of algorithms and utility analysis that take these constraints up front, rather than as an afterthought.

## Arrives vs. Departs

### Ground Truth



### Soft Public Constraint:

Note that most of our new reduced space is still empty. That's because most folks don't have 12 hour commutes, but our data space still includes those points. We can use public information (such as morning traffic reports) to further reduce this space. This should be done cautiously, removing only unambiguous absurd/outlier points.

# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

## How does it help?

Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

## Why is it interesting?

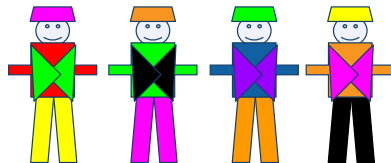
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

## So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data

**Outfit  
Colors**

white pink, red, orange, yellow,  
green, blue, purple, black



Possible Data Space



# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

### How does it help?

Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

### Why is it interesting?

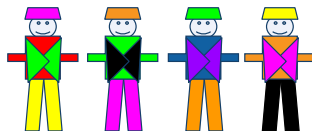
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

### So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data

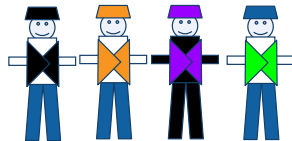
### Outfit Colors

white pink, red, orange, yellow, green, blue, purple, black



Possible Data Space

Hat	Shirt	Vest	Pants
orange	white	orange	black
green	blue	green	blue
black	white	black	blue
blue	black	purple	black



Actual Data Distribution

# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

### How does it help?

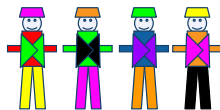
Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

### Why is it interesting?

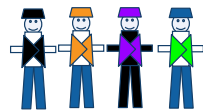
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

### So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data



Possible Data Space



Actual Data Distribution

Hat	Shirt	Vest	Pants
orange	white	orange	black
green	blue	green	blue
black	white	black	blue
blue	black	purple	black

How can we best capture this distribution?

If we take accounts across all four features at once, the data may be too sparsely spread across the four dimensional space (individuals complete outfits are too unique).

# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

### How does it help?

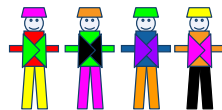
Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

### Why is it interesting?

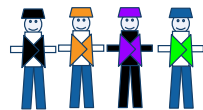
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

### So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data



Possible Data Space



Actual Data Distribution

We can use 2-way marginals to capture variable correlations. There's six possible here.

Vest	Pants
orange	black
green	blue
...	...

Vest	Shirt
orange	black
green	blue
...	...

Shirt	Vest
white	orange
blue	green
...	...

Hat	Pants
orange	black
green	blue
...	...

Hat	Vest
orange	black
green	blue
...	...

Hat	Shirt
orange	white
green	blue
...	...

# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

### How does it help?

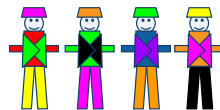
Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

### Why is it interesting?

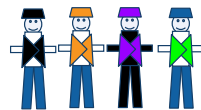
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

### So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data



Possible Data Space



Actual Data Distribution

We can use 2-way marginals to capture variable correlations. There's six possible here.

You might think that to capture  $k$  features in 2-way marginals, we need  $\binom{k}{2}$  queries. However, each new query increases sensitivity and the amount of added noise needed, which isn't ideal.

Vest	Pants
orange	black
green	blue
...	...

Vest	Shirt
orange	black
green	blue
...	...

Shirt	Vest
white	orange
blue	green
...	...

Hat	Pants
orange	black
green	blue
...	...

Hat	Vest
orange	black
green	blue
...	...

Hat	Shirt
orange	white
green	blue
...	...

# Pursuing a Formal Understanding of How These Work

## Pruning Marginal Queries

Reducing the amount of added noise needed by reducing the number of queries asked.

### How does it help?

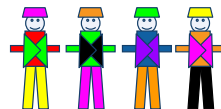
Rigorously pruning the query set, eliminating as many queries as possible while still capturing the population level distribution, reduces the required added noise without decreasing the amount of data. Every single algorithm to place in the top 3 in two years of challenges has leveraged this.

### Why is it interesting?

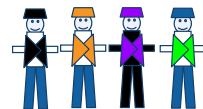
The fact that this works gets at the fundamental structure of human data. It's remained the primary successful strategy on long form Census data, as well as fire, 911 call, and taxi data. Intuitively, this has something in common with soft public constraints-- If there are some things you've done already, there are some other things you're likely to do, and to capture these trends well enough to reproduce a population level distribution we need to ask relatively few questions. This isn't a new idea. But a more formal understanding of how it impacts algorithms might be very helpful, not just for us but for Machine Learning in general.

### So here's a question-- What public data do you have that you can point us to?

Building Maximum Spanning Trees for diverse different data sets will allow us to look at how the tree structures vary between domains, how these variations impact the performance of modeling algorithms (both for our use case, but possibly also for others), and consider carefully at what information might get lost (especially as the complexity of the domain, \*or\* the population subgroups increases). Formally understanding both the power and limitations of these techniques is important. Lend us a hand, and we'll share with you what we find about your data



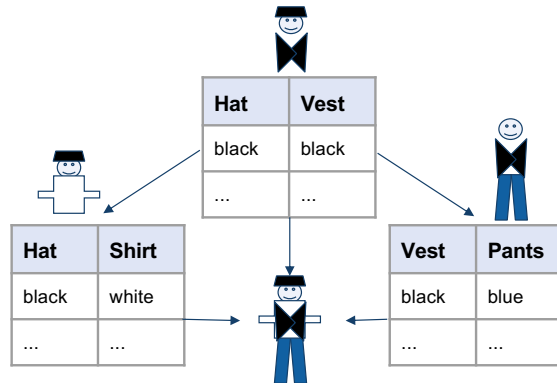
Possible Data Space



Actual Data Distribution

But our data has some pretty strong patterns of dependence/independence between features. Can we capture the distribution with fewer queries?

The minuteman solution uses Maximum Spanning Trees on the correlation graph to capture distribution across  $k$  features with  $(k-1)$  queries.



# Pursuing a Formal Understanding of How These Work

Just a caveat--- These three tricks aren't exhaustive. We're planning to continue to reach out to the research community for their own observations and neat tricks (feel free to share our submission box with your colleagues!).

We'll be working on a white paper and other resources that support further work. It's great to have reliably high utility high privacy synthetic data generators. It's even better to get them to perform extremely well on very hard problems. But it's best to have a formal understanding and a framework that allows for proof-work on precisely how/why/where they perform well.

In the meantime, please feel free to try out these tools for yourself, see how they perform on your own data, let us know what you find!

Your Data  
(or papers,  
reports,  
references)



Your  
Thoughts

A screenshot of the NIST Synthetic Data Demo Day submission form. The form is titled 'Submit tips, tricks, papers, etc relevant to the NIST Synthetic Data Demo Day'. It includes a section for 'Upload files' with a 'Select Files' button and a 'File description' section with a text input field. The NIST logo is at the top right.

**NIST** National Institute of Standards and Technology  
U.S. Department of Commerce

**Submit tips, tricks, papers, etc relevant to the NIST Synthetic Data Demo Day**

Please add files you think could be helpful to the differential privacy community. We will then distribute the materials. Please contact [gary.howarth@nist.gov](mailto:gary.howarth@nist.gov) with questions. Descriptions are optional and helpful.

**Upload files \***

Drag and drop files

Select Files

**File description**

Optional description

# WHERE DO I FIND RESOURCES?

Visit the NIST Temporal Map Challenge Website:

<https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/current-and-upcoming-prize-challenges/2020-differential>

....Just Google “NIST Temporal Map Challenge”

(or any vaguely plausible approximation...including simply “NIST Challenge”)

Winners				
Algorithm Contest				
<a href="#">Visit the Challenge data and scoring code repository</a>				
Team	Total Awards	Open Sourced	Development Contest	APA Citation
<a href="#">N-CRIP</a>	\$44,000.00	--	--	
<a href="#">Minutemen</a>	\$58,000.00	Yes	<a href="#">Repository link</a>	McKenna R. (2021). Adaptive Granularity Mechanism (version 1.0). URL: <a href="https://github.com/ryan112358/nist-synthetic-data-2021">https://github.com/ryan112358/nist-synthetic-data-2021</a>
<a href="#">DPSyn</a>	\$48,000.00	Yes	<a href="#">Repository link</a>	Chen A., Li N., Li Z., Wang T. (2021). DPSyn: An algorithm for synthesizing microdata for data analysis while satisfying differential privacy (version 1.0). URL: <a href="https://github.com/agl-c/deid2_dpsyn">https://github.com/agl-c/deid2_dpsyn</a>
<a href="#">jimking100</a>	\$24,000.00	Yes	<a href="#">Repository link</a>	King, J. (2021). Privitized Histograms (Version 1.0.0) [Computer software]. <a href="https://github.com/JimKing100/PrivacyHistos">https://github.com/JimKing100/PrivacyHistos</a>

# THANK YOU!



**Gary Howarth**  
Prize Challenge Manager  
NIST, PSCR  
[Gary.Howarth@nist.gov](mailto:Gary.Howarth@nist.gov)

**Christine Task**  
Computer Scientist  
Knexus Research  
[christine.task@knexusresearch.com](mailto:christine.task@knexusresearch.com)

