Utilizing Large-Scale Randomized Response at Google: RAPPOR and its lessons

Úlfar Erlingsson, Vasyl Pihur, Aleksandra Korolova, Steven Holte, **Ananth Raghunathan**, Giulia Fanti, Ilya Mironov, Andy Chu

DIMACS Security and Privacy Workshop (April 2017)

RAPPOR Motivation: Hijacking of Chrome Settings

Find the Chrome homepages/search-engines used by clients
... with privacy for each user

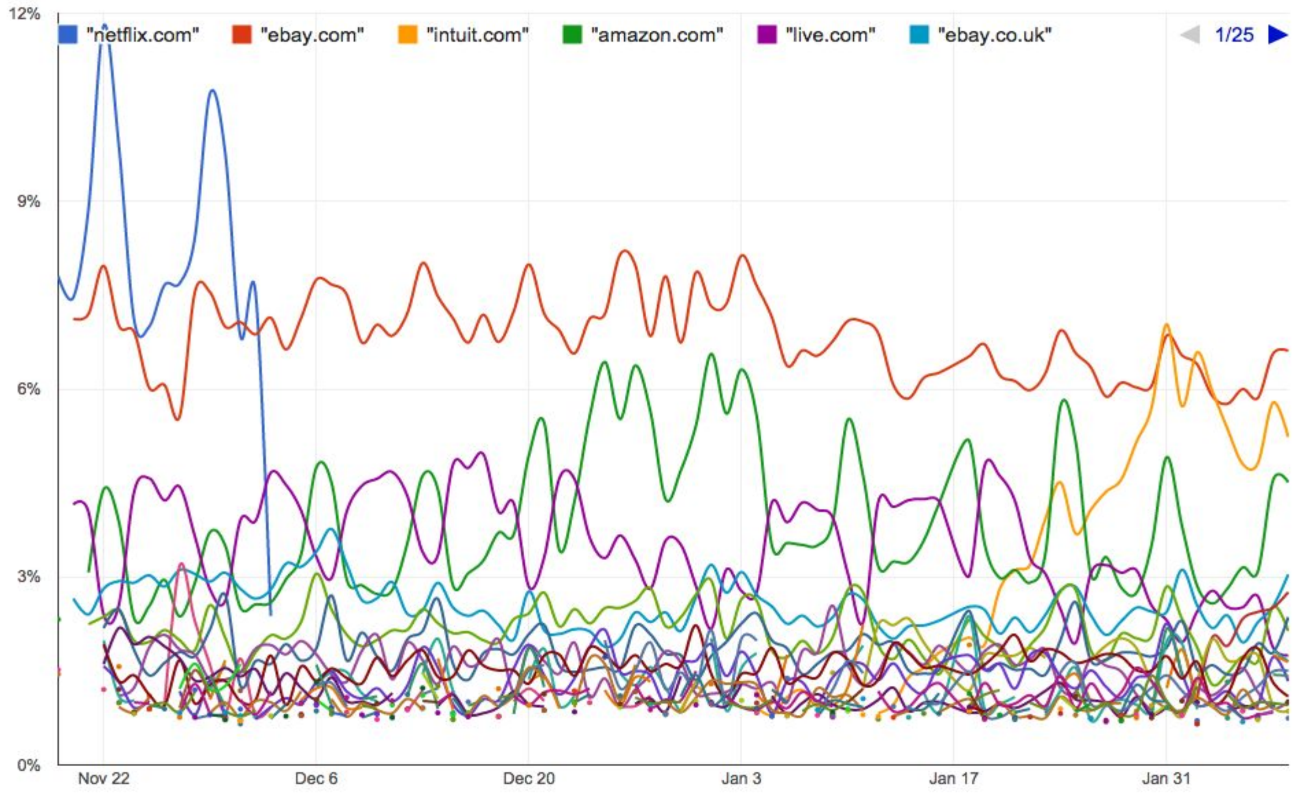
I.e., find popularity %'s of
Yahoo! Search, Bing, ...
Also: detect unusually high %'s for
sites installing unwanted software

RAPPOR can find them, without
seeing any user's homepage!

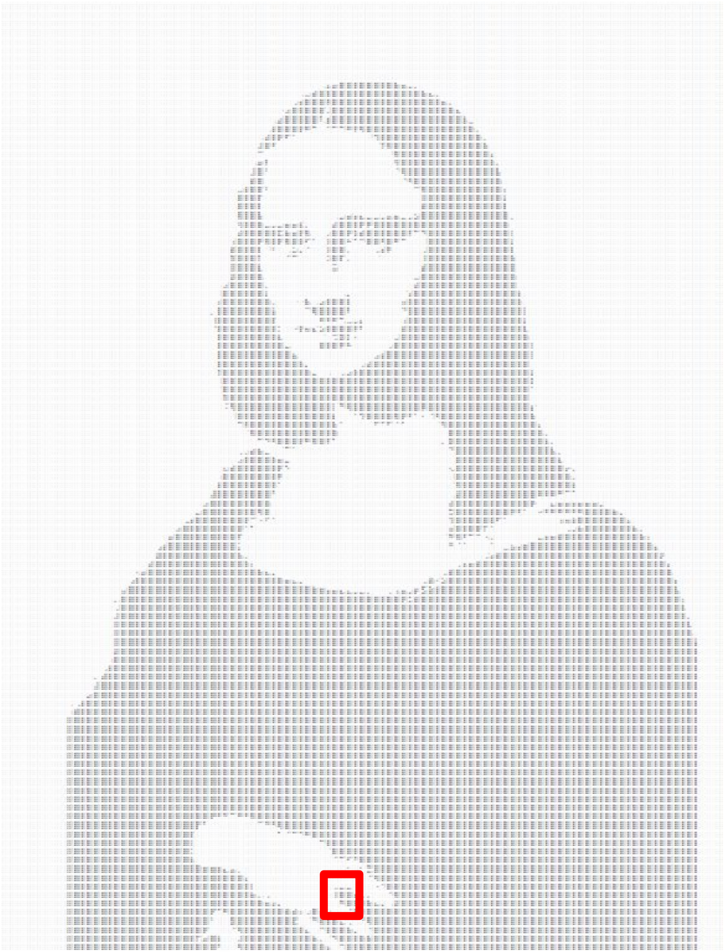
Who on the Web is still using Silverlight?

Estimated by RAPPOR

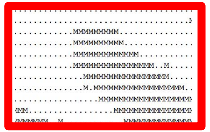
netflix
ebay
intuit
amazon
live

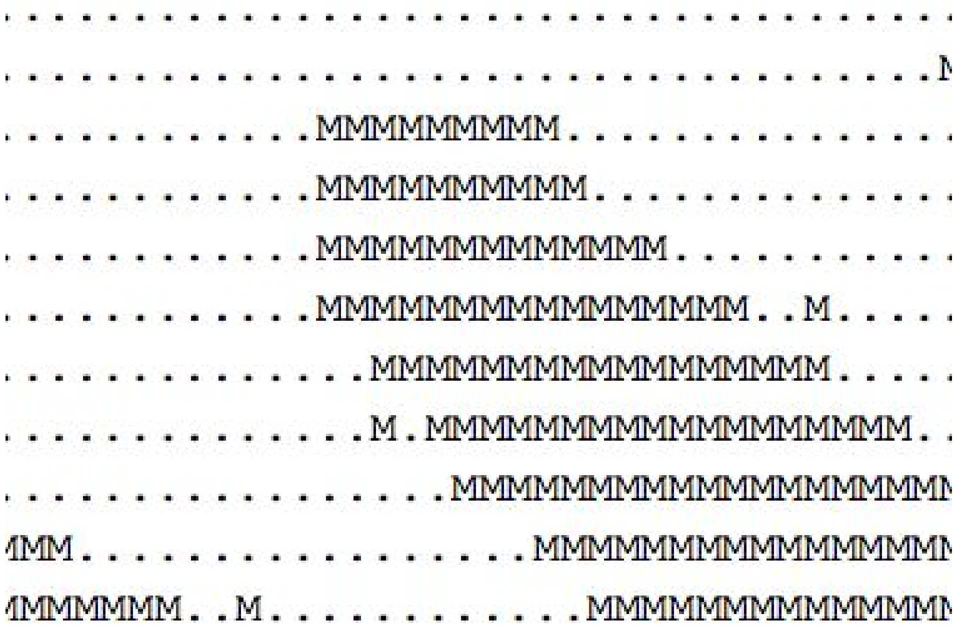


Metaphor for RAPPOR



Microdata: An individual’s report





Microdata: An individual's report

Each bit is flipped with probability 25%

```
.......M........MM.M.......MMM.M..
......................MM...MMMM...
....M..MM.MM..MMM.M.MM.M...M..MM..
.MM......MMM....MMMMMMMMM...M...MM
..M....M........MM..MMMMMMM...M...
M.......M..MM.MMMMMMMMMMMMMMM....M
.....M.....M.M.M.MMMMMM...MMMMM...
...M.....M.MM.M.MM..M..M..MM.MMMMM
M...M.M.....M.M..M..MMM.MMMMM.MMMM
.MMM.M....M.M.M........MMMMMMMMM.M
```

Big picture remains!

Best practice for learning statistics about users/clients

- **Collect** user data (perhaps with unique id for each user)
- **Scrub** IP addresses, timestamps, etc., from user data

- **Keep central database** of scrubbed data (e.g., for 2 weeks)
    - Keep only aggregates for older data
- **Report aggregates of data over a threshold** (e.g., 10 users)

Can be the best approach (e.g., for opt-in, low-sensitivity data)

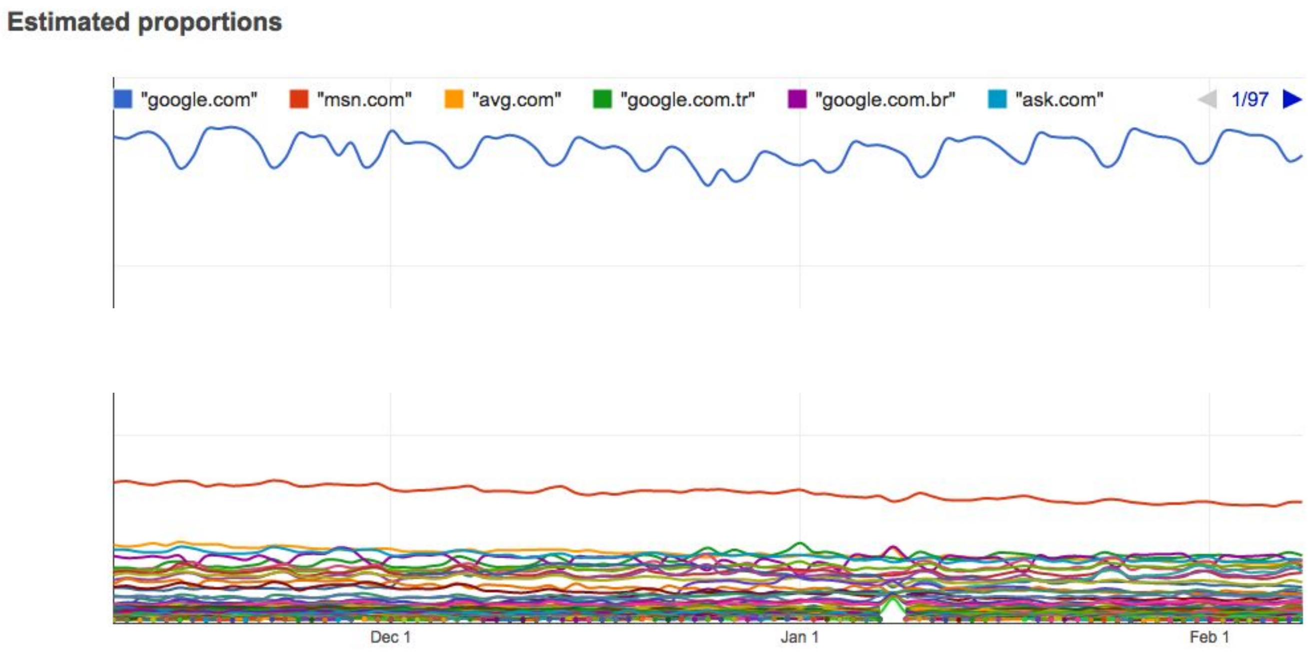
RAPPOR: Learn user statistics with much stronger privacy

- **Rigorous and meaningful privacy guarantees** for each user
- **No central database** (hackable, subpoenable) of user data
- User's privacy **doesn't depend on a trusted third party**
- **No privacy externalities** (e.g., from trackable user IDs)

Well-suited to sensitive user data, such as URLs from users

Dashboard at [redacted]

Chrome homepages (over 90 days)



google
msn
avg
google tr
google br

Gold Standard of Security

Same key aspects in software construction & computer security

In programming		**In security**
Specification	=	Security policy
Implementation	=	Enforcement mechanism
Correctness	=	Assurance
Methodology*	=	Security model

* e.g., **functional** vs. **declarative** vs. **imperative programming**

Gold Standard of Privacy

Same key aspects in software construction & computer security

In programming		**In privacy**
Specification	=	Privacy policy
Implementation	=	Enforcement mechanism
Correctness	=	Assurance
Methodology	=	Privacy model*

* e.g., **HIPAA** vs. **usage control** vs. **local- or database-differential privacy**

Takeaways from this talk

1. **Randomized response**
   Learning categorical data and aggregating Bloom filters
2. **RAPPOR's 2-level randomized response**
   Longitudinal differential privacy and anonymity
3. Lessons learnt from the large-scale deployment of a randomized-response privacy mechanism
4. Follow-up works

1. Randomized Response: Collecting a sensitive Boolean

Developed in 1960’s for sensitive surveys

“Are you now, or have you ever been, a member of the communist party?”

a. Flip a coin, **in private**
b. If coin comes up heads, respond “Yes”

c. If coin comes up tails, tell the truth

Estimate true “Yes” ratio with: “Yes”% - 50%

1. Randomized Response: Collecting a sensitive Boolean

Developed in 1960's for sensitive surveys

"Are you now, or have you ever been, a member of the communist party?"

a. Flip a coin, **in private**
b. If coin comes up heads,
   --- flip another coin to select randomly "Yes" or "No"
c. If coin comes up tails, tell the truth

Satisfies differential privacy property (with two coins)

Still easy to estimate true "Yes" ratio

Randomized response on categorical Boolean values

- If number of categories is small, can do an independent randomized response for each category
  - Bit-by-bit array of randomized responses

0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	0	1	0

- Example: The categories may refer to salary ranges
  - Users do a “yes/no” randomized response for each range

0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	0	1	0

Randomized response on categorical Boolean values

- If number of categories is small, can do an independent randomized response for each category
  - Bit-by-bit array of randomized responses

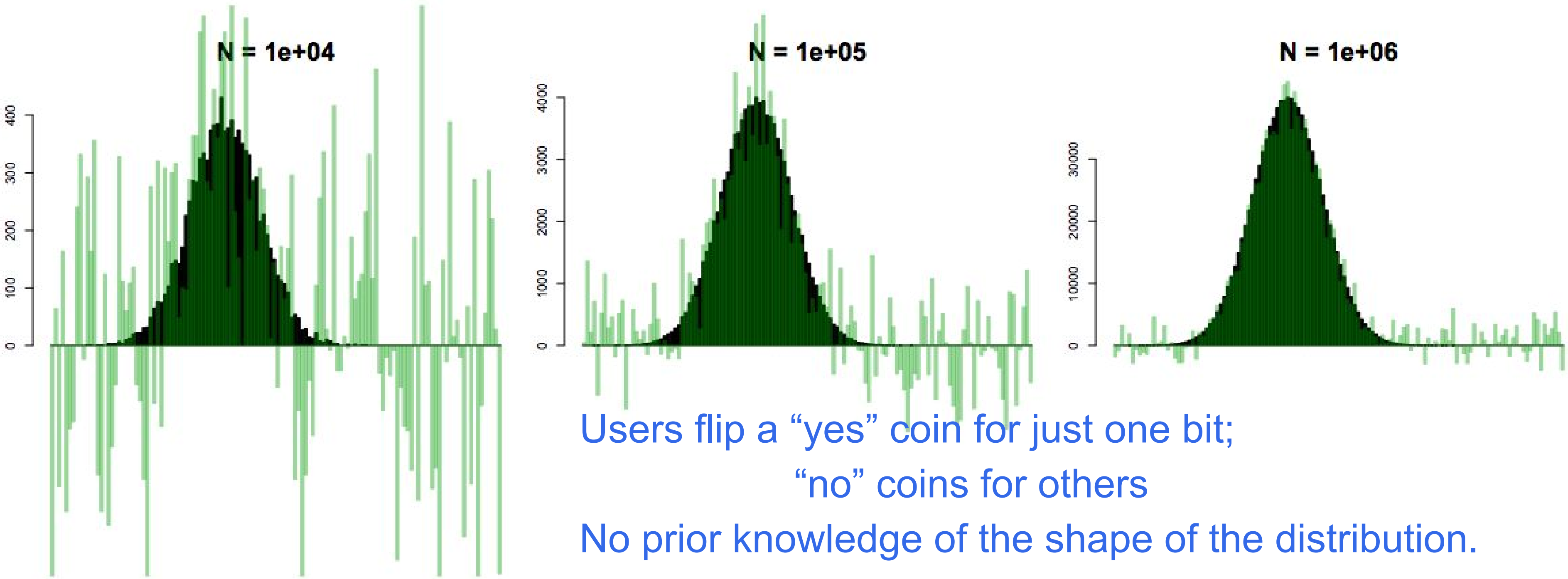
0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	0	1	0

- Example: The categories may refer to salary ranges
  - Users do a “yes/no” randomized response for each range

0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	0	1	0

This user’s salary lies in this range.
The “Yes” coin came up heads, so bit is “1”.

Learning the shape of the Salaries distribution

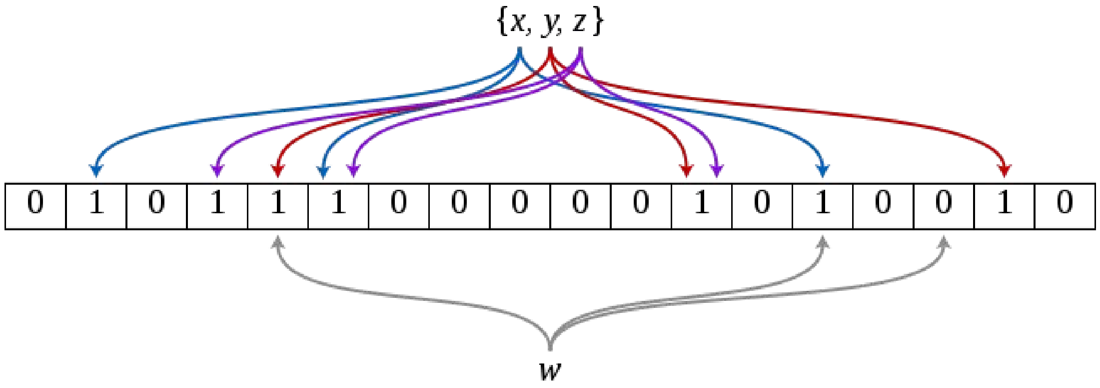


Users flip a “yes” coin for just one bit;
“no” coins for others

No prior knowledge of the shape of the distribution.

Bloom filters to handle large sets of categories

- Compressed representation of a large set



- To minimize collisions/false positives, use multiple cohorts
  - Randomly assign clients to one of *m* cohorts
  - Each cohort uses different Bloom-filter hash functions

2. RAPPOR two-level randomization and differential privacy

- Problem to ask the communist question repeatedly
  - Average of coin flips eventually reveals the true answer
- **Memoization** is the trick: Reuse the same answer

- But memoized random bits can hurt anonymity
  - Repeated bit sequence forms a unique tracking ID

- **Randomization of memoized response** is the answer!
  - Flip coins on a value, and memoize
  - Then report coin flips on the memoized data

RAPPOR algorithm

1. Hash a value *v* into Bloom filter *B* using *h* hash functions
2. Memoize a **Permanent Randomized Response** *B'*

$$B'_i = \begin{cases} 1, & \text{with probability } \frac{1}{2}f \\ 0, & \text{with probability } \frac{1}{2}f \\ B_i, & \text{with probability } 1-f \end{cases}$$

3. Report an **Instantaneous Randomized Response** *S*

$$P(S_i = 1) = \begin{cases} q, & \text{if } B'_i = 1. \\ p, & \text{if } B'_i = 0. \end{cases}$$

RAPPOR algorithm

1. Hash a value *v* into Bloom filter *B* using *h* hash functions
2. Memoize a **Permanent Randomized Response** *B'*

$$B'_i = \begin{cases} 1, & \text{with probability } \frac{1}{2}f \\ 0, & \text{with probability } \frac{1}{2}f \\ B_i, & \text{with probability } 1 - f \end{cases}$$

$f = ½$
for example

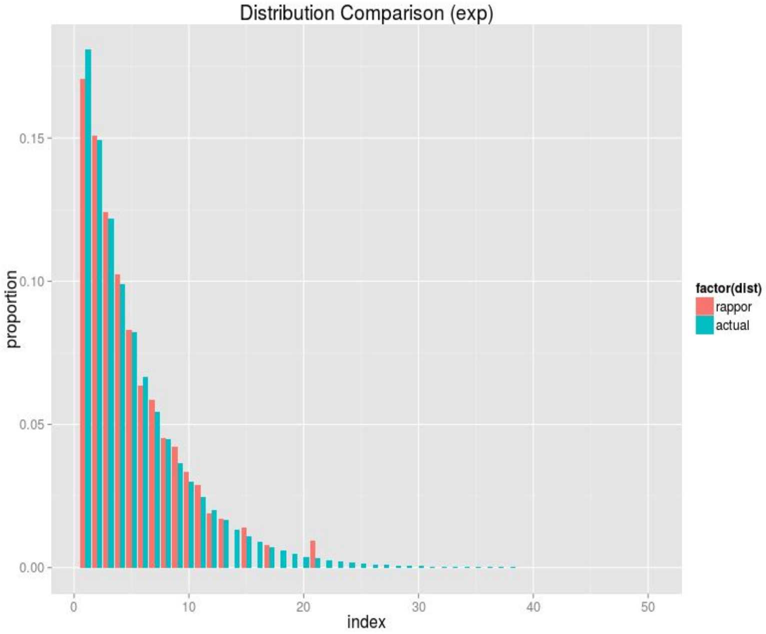
3. Report an **Instantaneous Randomized Response** *S*

$$P(S_i = 1) = \begin{cases} q, & \text{if } B'_i = 1. \\ p, & \text{if } B'_i = 0. \end{cases}$$

$q = ¾$ and $p = ½$
for example

OSS project

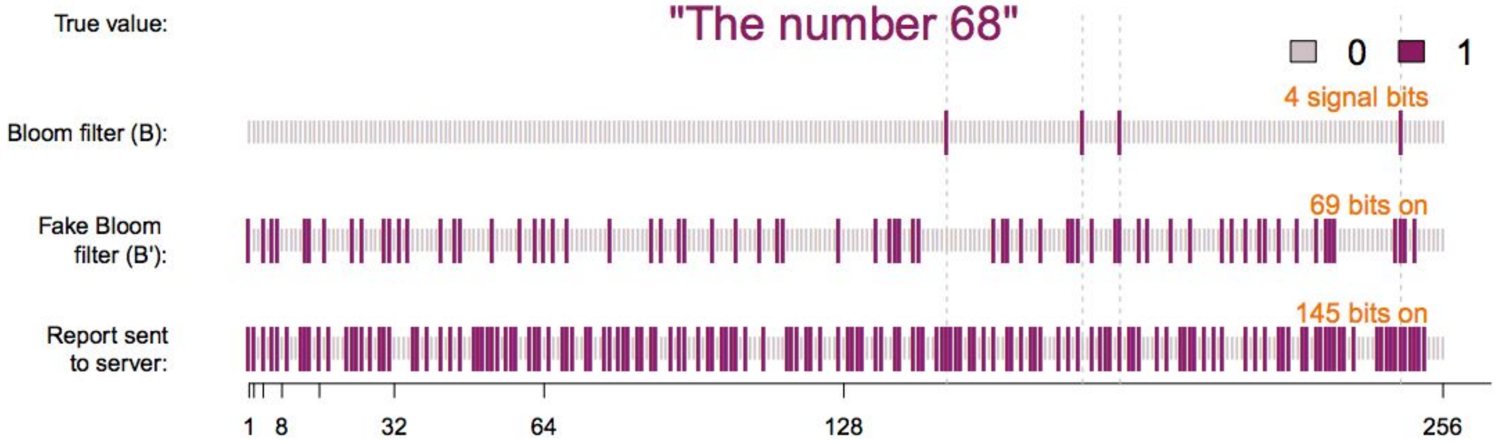
- Contents of https://github.com/google/rappor
  - Demo that you can run with a couple shell commands
  - Client library
  - Analysis tools and simulation
  - Documentation
  - Analysis service
  - Clients code in a few languages



Lessons Learnt

Design for simple explainability

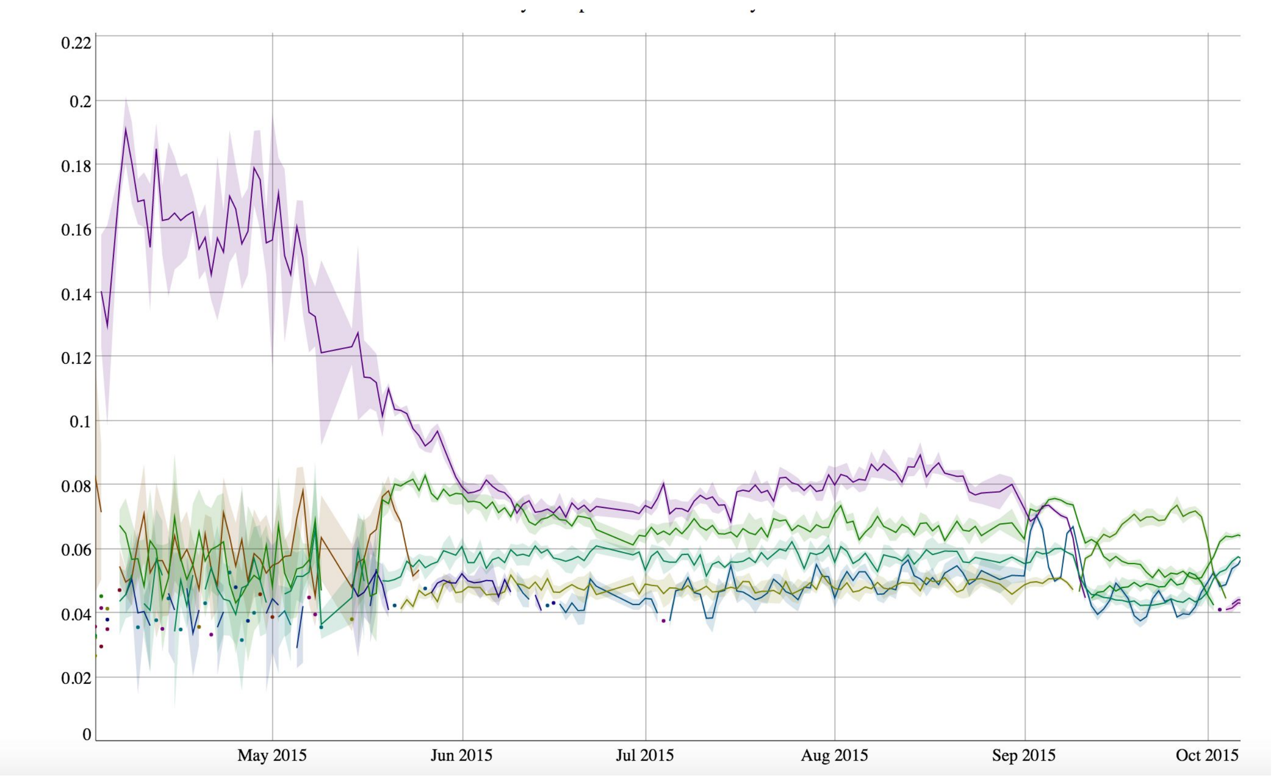
Critical to get comfort / acceptance from ***everybody***
… (also need reasonable ε, and may want user opt-in)



There will be growing pains

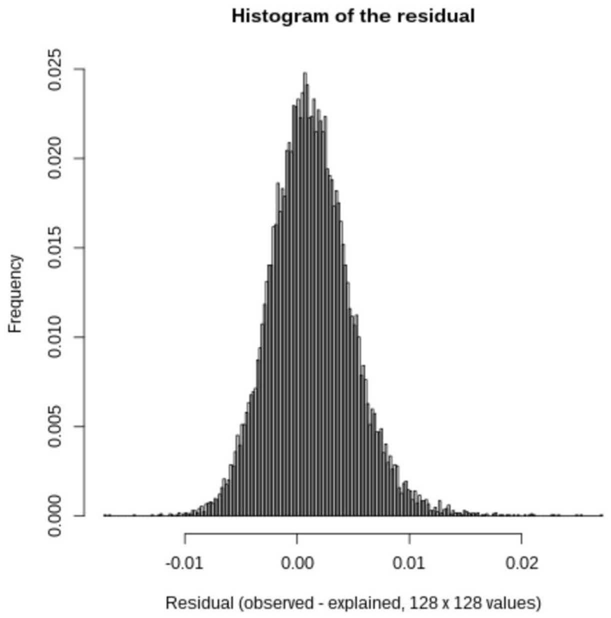
- Transitioning from a research prototype to a real product
- Scalability
- Versioning

Communicate Uncertainty

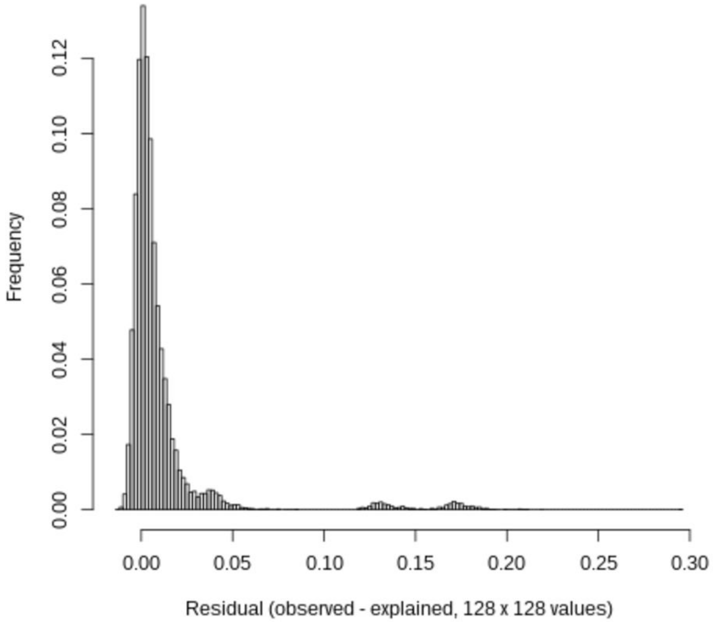


Candidates? – Enable diagnostics on collected data

No missing candidates



Three missing candidates



Know thy Enemies and Friends

If **raw data** is being collected:

- privacy people & technology are a hindrance to utility
- hard to avoid the slippery slope

… bodes ill for (pure) database-differential privacy

If **statistical/privacy-protected data** is collected:

- privacy people become essential to utility
- big step onto the slippery slope

… good reason to add noise early

Keep your friends close ...

- Partner closely with the users, and monitor their use
  - `tools/metrics/rappor/rappor.xml - chromium/src`
- Avoid users treating your technology as a black box
  - they'll be disappointed & affect user privacy w/o utility
- Set and manage expectations
  - e.g., local differential privacy can only see peaky tops

The world depends on trust; we can’t do without it

- Google provides data for Chrome and RAPPOR!
- The ε for RAPPOR’s are just worst-case fallbacks
  … do much better, unless Google explicitly chooses evil
- But, without trust, those ε only allow seeing peaky tops

- Need to work on better basis for combining trust with privacy
  - E.g., via technical and contractual separation of concerns
  - Backed by verifiable enforcement teeth

Follow-up Works

- Giulia Fanti, Vasyl Pihur, Úlfar Erlingsson, "Building a RAPPOR with the Unknown: Privacy-Preserving Learning of Associations and Data Dictionaries", PoPETS 2016
  - Two-way contingency tables and recovering missing candidates
- Bassily, Smith, "Local, Private, Efficient Protocols for Succinct Histograms," STOC 2015
- Kairouz, Bonawitz, Ramage, "Discrete Distribution Estimation under Local Privacy", https://arxiv.org/abs/1602.07387
- Qin et al., "Heavy Hitter Estimation over Set-Valued Data with Local Differential Privacy", CCS 2016

Follow-up Works

- Abadi, Chu, Goodfellow, McMahan, Mironov, Talwar, Zhang. “Deep learning with differential privacy.” ACM CCS 2016.

- Papernot, Abadi, Erlingsson, Goodfellow, Talwar. “Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data.” ICLR 2017.

Conclusions

RAPPOR – locally differentially-private mechanism for reporting of categorical and string data

- First Internet-scale deployment of differential privacy
- Explainable
- Conservative
- Open-sourced
- Challenging
- … just the beginning

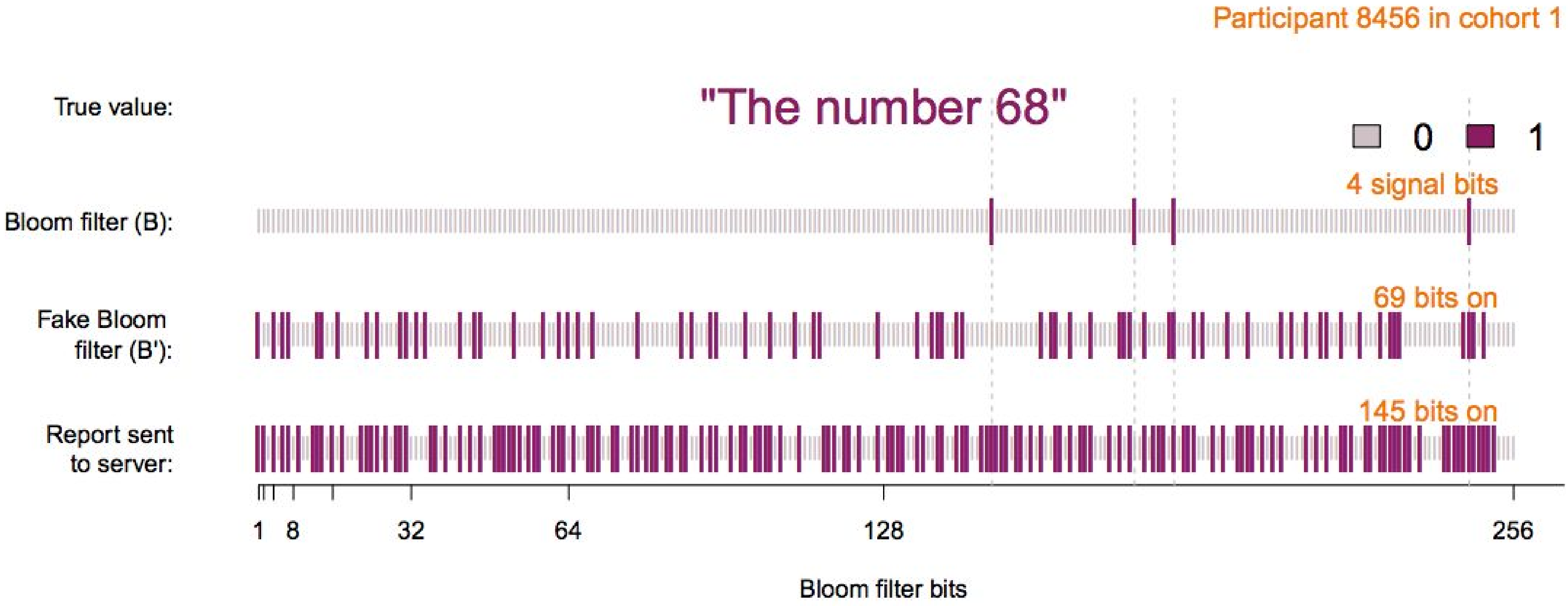
Thank you!

Any questions?

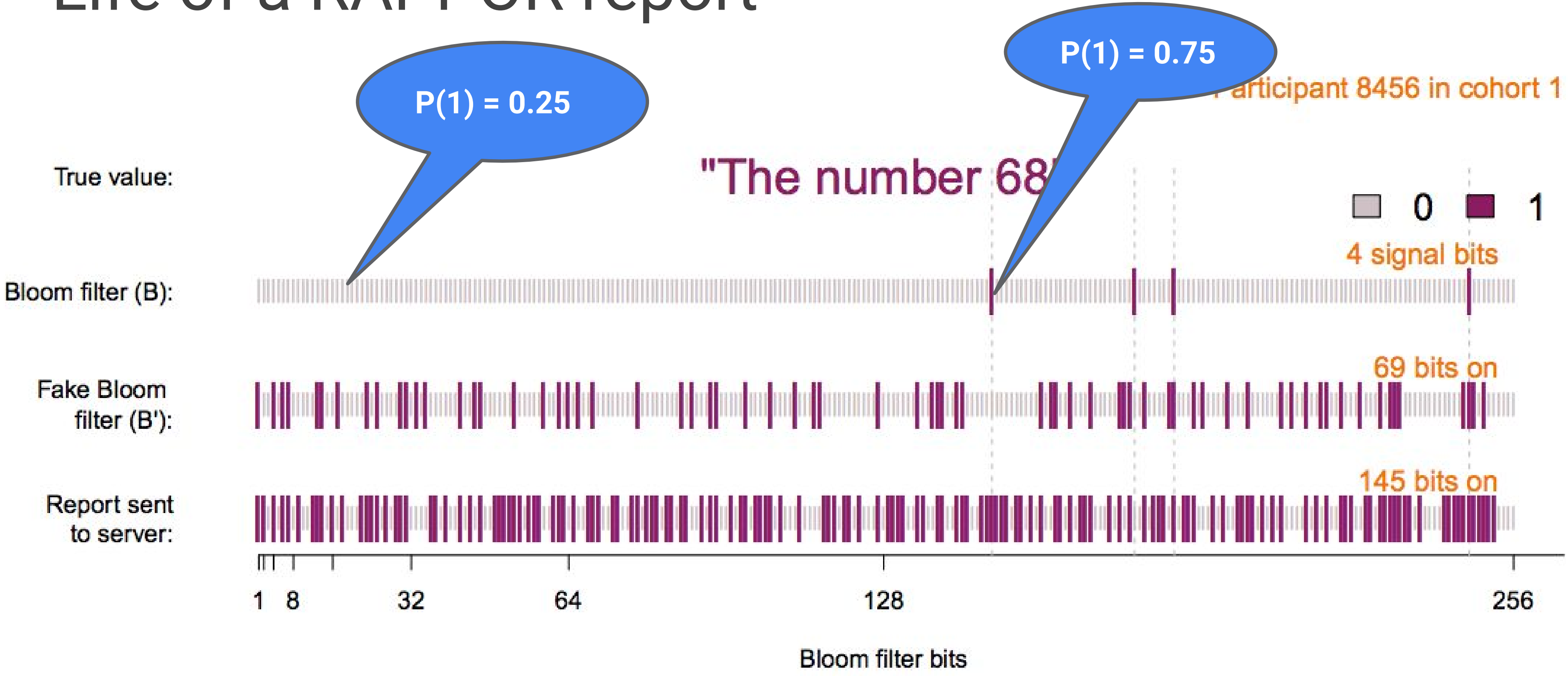
—pseudorandom@google.com—

Backup

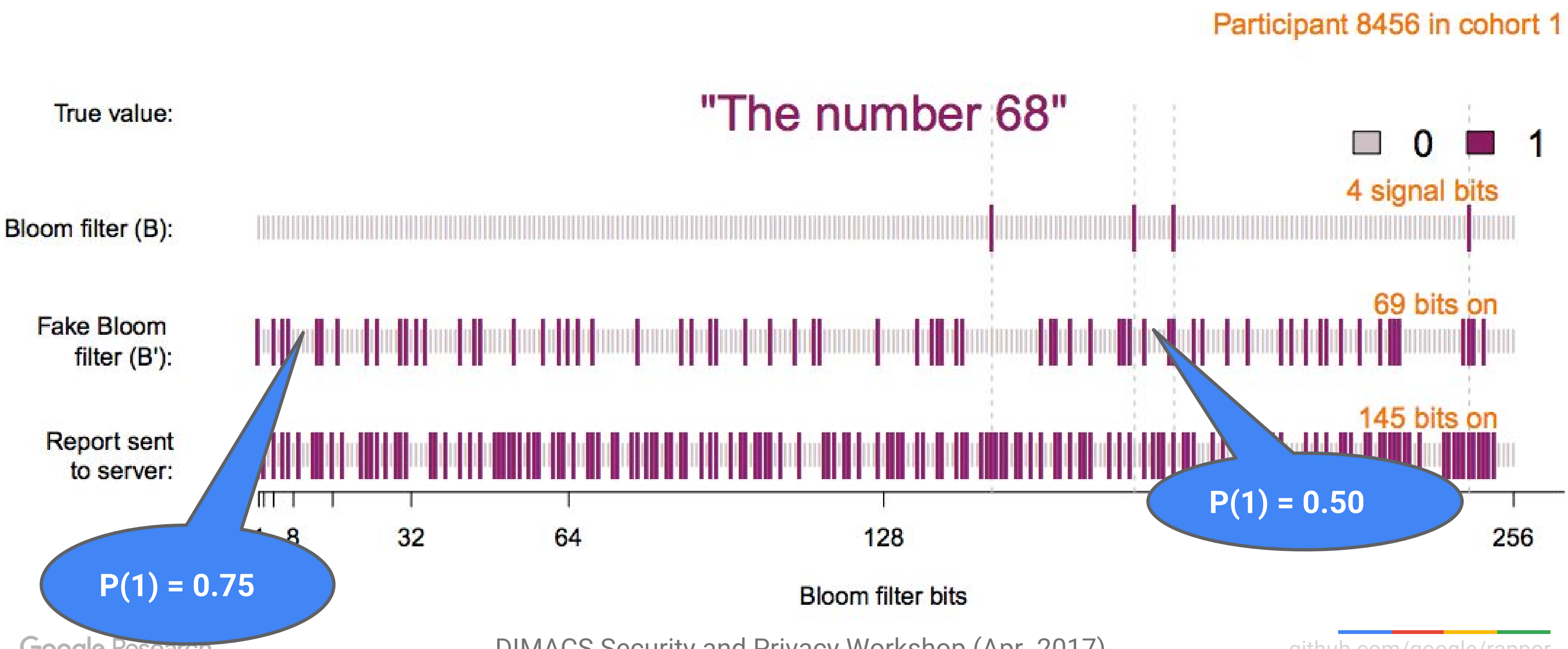
Life of a RAPPOR report



Life of a RAPPOR report



Life of a RAPPOR report



Differential Privacy of RAPPOR

- **Permanent Randomized Response** satisfies differential privacy at

$$\epsilon_\infty = 2h \ln\left(\frac{1 - \frac{1}{2}f}{\frac{1}{2}f}\right).$$

- **Instantaneous Randomized Response** has differential privacy at

$$\epsilon_1 = h \log\left(\frac{q^*(1 - p^*)}{p^*(1 - q^*)}\right)$$

Differential Privacy of RAPPOR

- **Permanent Randomized Response** satisfies differential privacy at

$$\epsilon_\infty = 2h \ln\left(\frac{1 - \frac{1}{2}f}{\frac{1}{2}f}\right)$$ = **4 ln(3)**, *for example*

- **Instantaneous Randomized Response** has differential privacy at

$$\epsilon_1 = h \log\left(\frac{q^*(1-p^*)}{p^*(1-q^*)}\right)$$ ≈ **ln(3)**, *for example*

Decoding RAPPOR

True bit counts, with no noise

Decoding RAPPOR

True bit counts, with no noise

De-noised RAPPOR reports

google.com:

yahoo.com:

bing.com:

From denoised counts to distribution

Linear Regression:

$$\min_X \|B - A\,X\|_2$$

LASSO:

$$\min_X (\|B - A\,X\|_2)^2 + \lambda\|X\|_1$$

Hybrid:

1. Find support of X via LASSO
2. Solve linear regression to find weights