Algorithmic Stability for Interactive Data Analysis: An Overview

Jonathan Ullman, Northeastern University

Based on several (dis)joint works: [H**U**'14], [DFHPRR'15abc], [S**U**'15], [BNSSS**U**'16]

Optimization, Statistics, and Uncertainty Workshop, Berkeley, Nov 30, 2017.

Statistical Theory: One-Way Streets

Hypothesis

Data

Conclusions

Statistical analysis guarantees that your conclusions generalize to the population

And Yet…

PLOS | MEDICINE

OPEN ACCESS

ESSAY

1,140,912 VIEWS

1,413 CITATIONS

Why Most Published Research Findings Are False

John P. A. Ioannidis

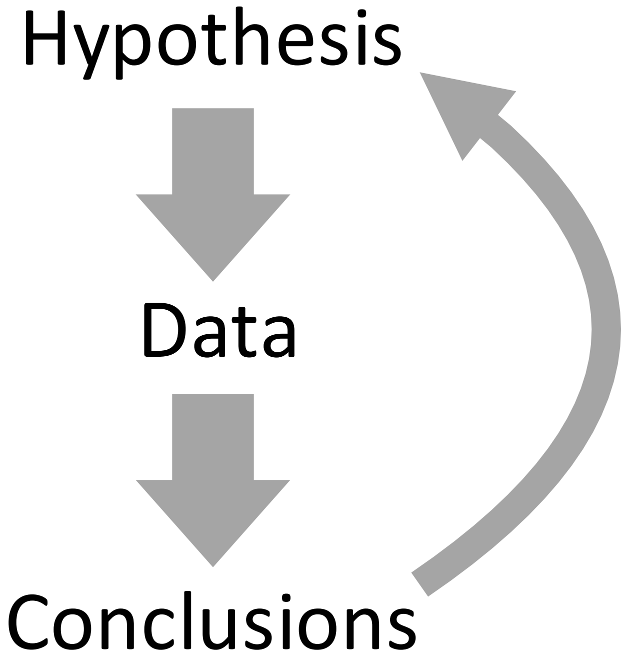
Published: August 30, 2005 • DOI: 10.1371/journal.pmed.0020124

The Statistical Crisis in Science

Data-dependent analysis—a "garden of forking paths"— explains why many statistically significant comparisons don't hold up.

Andrew Gelman and Eric Loken

Statistical Practice: Traffic Circles



Statistical guarantees no longer apply when the dataset is re-used interactively

Examples of Interaction

- Well specified multi-stage algorithms
  - Example: fit a model after selecting features
  - Could try to analyze explicitly

- Data exploration / “researcher degrees of freedom”
  - Example: data science competitions

- Multi-researcher re-use of datasets
  - Example: publications involving public or standard datasets
  - Cannot hope to analyze explicitly

Possible Approaches

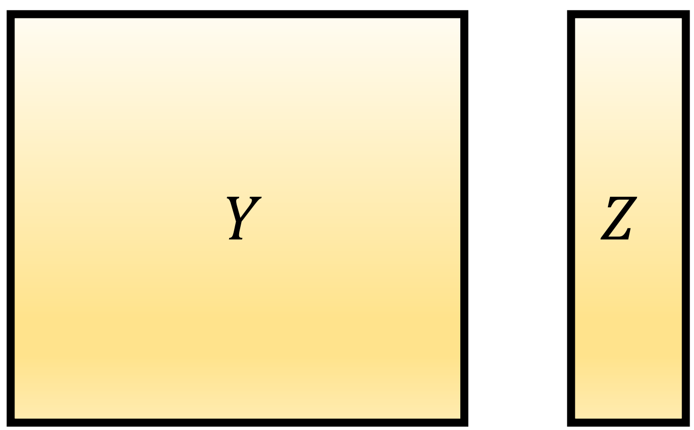
- Hypothesis testing
  - Assumes hypotheses are independent of the data
  - Multiple-hypothesis testing addresses a different problem
- Explicit post-selection inference
  - Tractable for well specified algorithms
  - More amenable to analysis than algorithm design
- Holdout sets / data splitting
  - Once the holdout is used, we are back where we started
  - Need data linear in the number of interactive rounds

This Talk

- A general approach to interactive data analysis
  - Introduced in [DFHPRR'15, H**U**'14]
  - New general tools and methodology
  - Leads to new algorithms for preventing overfitting
- Key ingredient: algorithmic stability
  - Strong notions of stability inspired by differential privacy
  - Uses randomization to improve generalization
- New inherent bottlenecks [H**U**'14,S**U**'15]
  - Both statistical and computational

Overfitting in Interactive Data Analysis

- Population P of uniformly random labeled examples
- Sample $X = (Y_1, Y_1), \dots, (Y_n, Z_n) \in \{\pm 1\}^d \times \{\pm 1\}$
- Goal: find $h\colon \{\pm 1\}^d \to \{\pm 1\}$ maximizing $s_P(h) = \mathbb{E}_P[h(y)z]$
- If we use $s_X(h)$ as a proxy for $s_P(h)$, we can quickly overfit

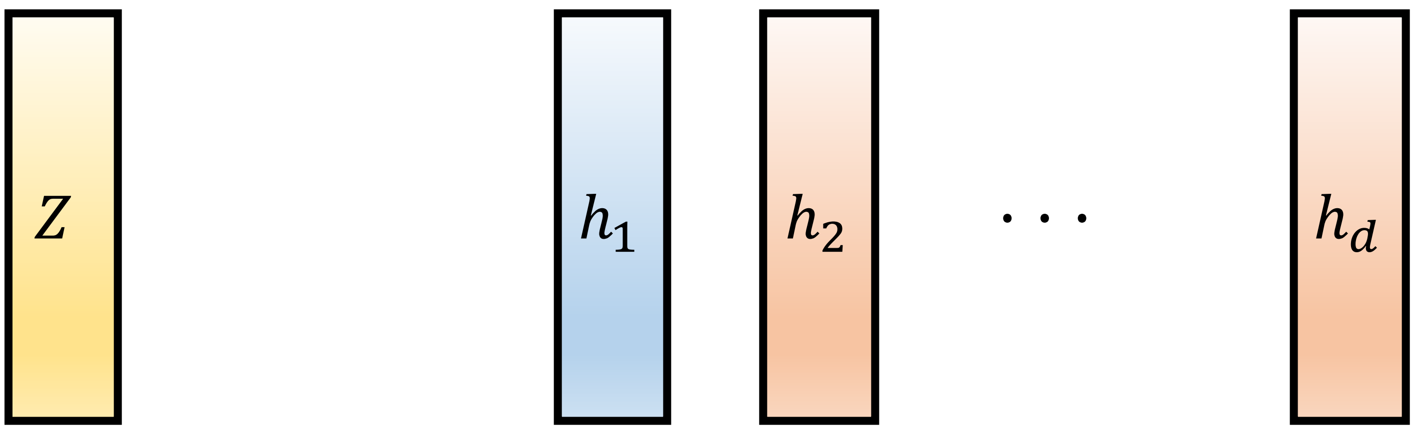


Overfitting in Interactive Data Analysis

- Freedman's Paradox:
  - For $j = 1, \dots, d$ consider the hypothesis $h_j(y) = y_j$

Random labels
$Z \in \{\pm 1\}^n$

Labels of initial hypotheses
(random and independent)



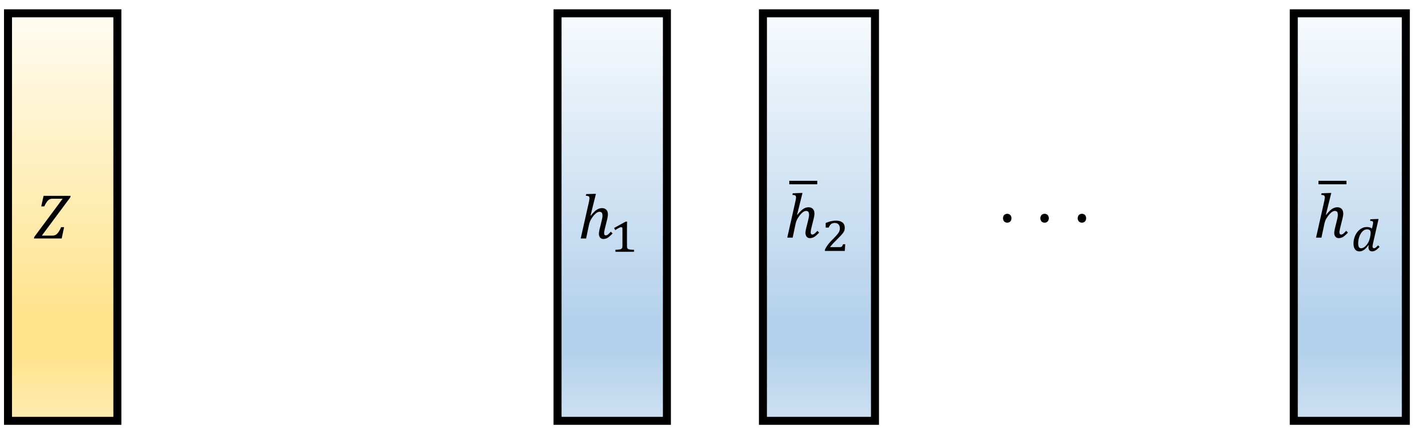
$s_X(h_1) \approx \frac{+1}{\sqrt{n}}$ $s_X(h_2) \approx \frac{-1}{\sqrt{n}}$ $s_X(h_d) \approx \frac{-1}{\sqrt{n}}$

Overfitting in Interactive Data Analysis

- Freedman's Paradox:
  - For $j = 1, \ldots, d$ consider the hypothesis $h_j(y) = y_j$
  - Flip signs as needed so $s_X(h_j) \geq 0$ for all $j = 1, \ldots, d$

Random labels
$Z \in \{\pm 1\}^n$

Labels of initial hypotheses
(random and independent)

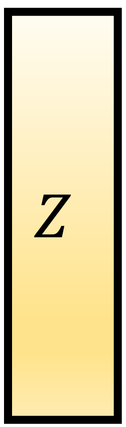


$s_X(h_1) \approx \frac{+1}{\sqrt{n}}$ $s_X(\bar{h}_2) \approx \frac{+1}{\sqrt{n}}$ $s_X(\bar{h}_d) \approx \frac{+1}{\sqrt{n}}$

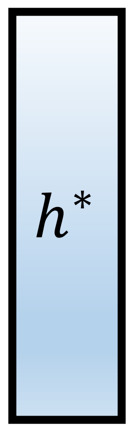
Overfitting in Interactive Data Analysis

- Freedman's Paradox:
  - For $j = 1, \ldots, d$ consider the hypothesis $h_j(y) = y_j$
  - Flip signs as needed so $s_X(h_j) \geq 0$ for all $j = 1, \ldots, d$
  - Let $h^*(y) = \mathrm{majority}\left(h_1(y), \bar{h}_2(y), \ldots, \bar{h}_k(y)\right)$

Random labels
$Z \in \{\pm 1\}^n$



Labels of majority vote h^*

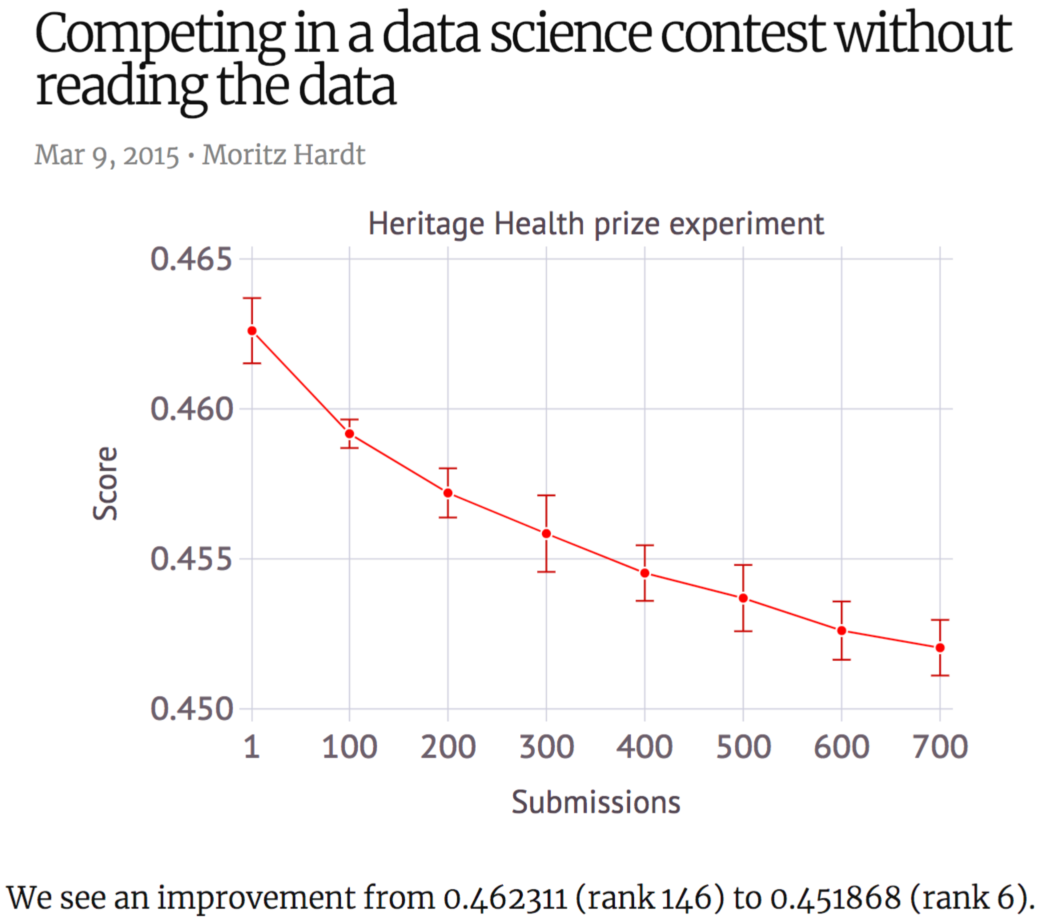


Thm: $s_X(h^*) = \Theta\left(\sqrt{\frac{d}{n}}\right)$

A factor of $\approx \sqrt{d}$ more overfitting because of dataset re-use!

Overfitting in Interactive Data Analysis

- A Real-World Example: Data Science Competitions [BH'15]



How to Avoid This Trap?

- What went wrong?
  - The scores $s_X(h_1), \dots, s_X(h_d)$ revealed a lot of information about the unknown labels
- What do we do about it?
  - Minimize the amount of information that is leaked about the dataset
- How would we do that?
  - Use ideas from differential privacy [DMNS'06]
  - Private algorithms have strong stability properties

Output Stability

- Stability has been a central concept since the seventies, e.g. [DW'78, KR'99, BE'02, SSSS'10]
- Typically, some kind of output stability: for all neighboring samples X, X',

$$d\big(A(X), A(X')\big) \le \epsilon$$

close inputs map to close outputs

- An output-stable $A(X)$ can reveal X entirely, does not prevent overfitting in interactive settings
  - See Freedman's Paradox

Distributional Stability (aka Privacy)

- Differential Privacy [DMNS'06]: for all neighboring samples X, X' and all $O \subseteq \mathrm{Range}(A)$

$$\Pr[A(X) \in O] \le e^{\varepsilon} \Pr[A(X') \in O] + \delta$$

close inputs map to close distributions

- A private A reveals little about X, prevents overfitting even after seeing $A(X)$

Distributional Stability

- Distributional Stability (DS, for short): for all neighboring samples X, X'

$$A(X) \approx_{\varepsilon,\delta} A(X')$$

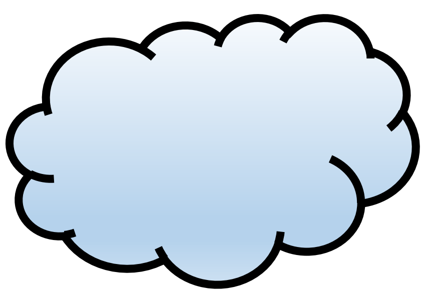
close inputs map to close distributions

- A DS A reveals little about X, prevents overfitting even after seeing $A(X)$
- Growing family of distributional stability notions
  - [DFHPRR’15, RZ’15, WLF’15, BNSSS**U**’16, BF’16, DR’16, BS’16, BDRS’17,…]

A General Framework

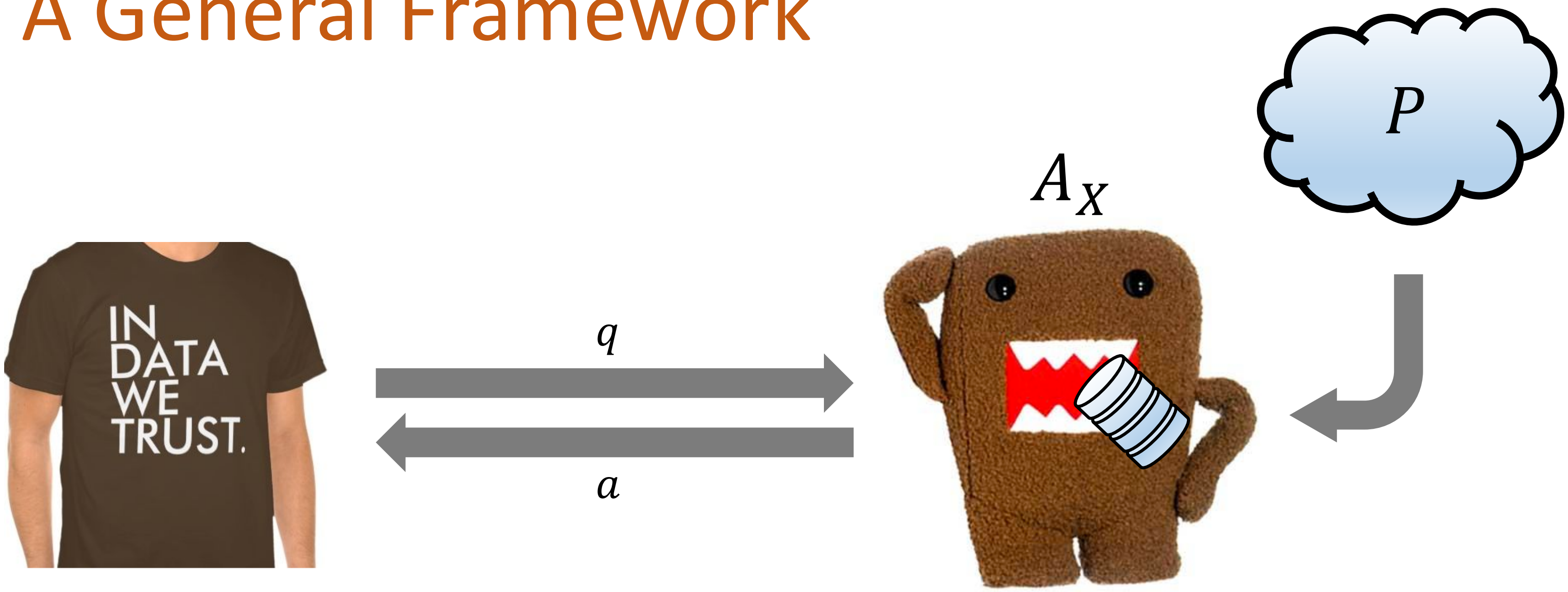
- A population P over some universe U
- A sample $X = (X_1, \ldots, X_n)$ from P
- A class of statistics Q
  - For example "What fraction of P has the property q?"

P



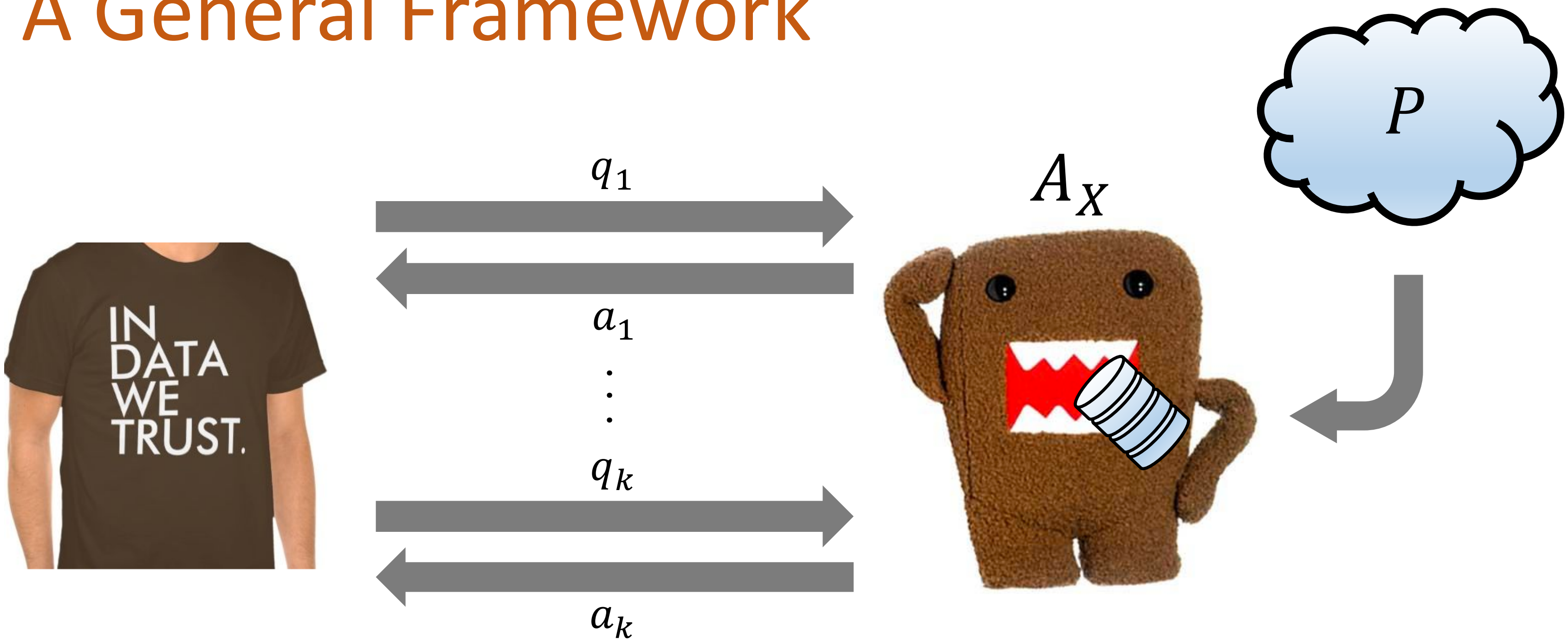
X

A General Framework



- Goal: design an A that accurately estimates $q(P)$
  - Accurate depends on Q, typically $|a - q(P)| \leq \alpha$
  - Challenge: A does not observe P

A General Framework



- Modeling interactive data analysis:
  - Allow an analyst to request a sequence $q_1, \ldots, q_k$
  - Each q_j depends arbitrarily on $q_1, a_1, \ldots, q_{j-1}, a_{j-1}$
- Goal: one estimator for every analyst
  - Want to avoid assumptions about the analyst strategy

Example: Statistical Queries (SQs)

- Given a bounded function

$$\phi: U \to [\pm 1]$$

- The statistical query q_ϕ is defined as

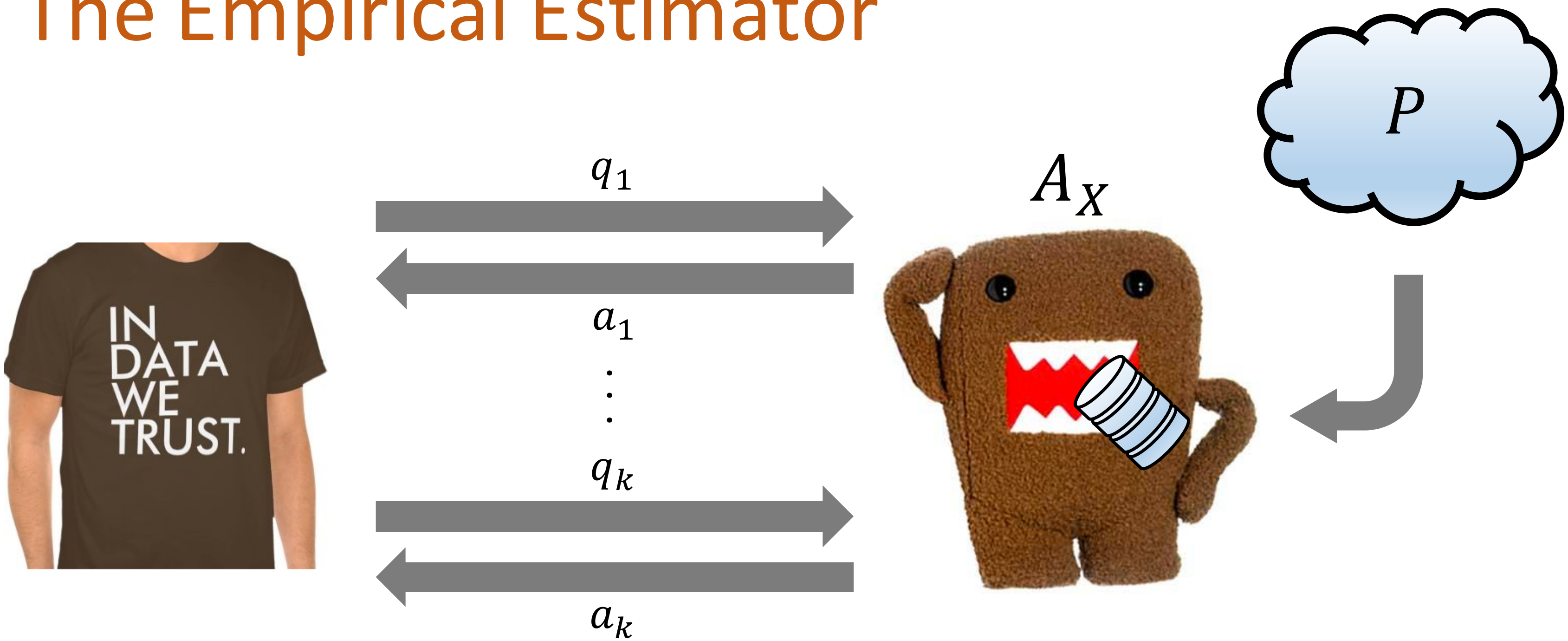
$$q_\phi(P) = \mathbb{E}\left[\phi(P)\right]$$

- An answer a is α-accurate if $\left|a - q_\phi(P)\right| \leq \alpha$

- Highly useful and general family of queries
  - Mean, variance, covariance
  - Score of a classifier
  - Gradient of the score of a classifier
  - Almost all PAC learning algorithms
  - …

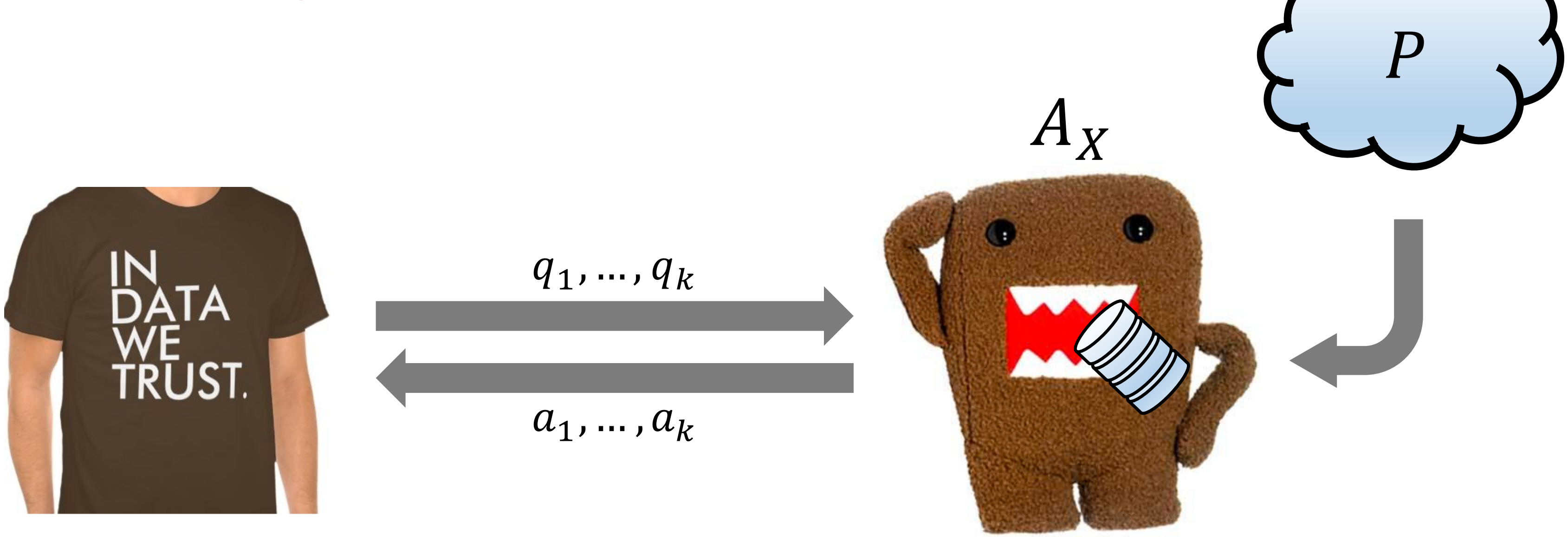
Captures Freedman's Paradox

The Empirical Estimator



- Empirical estimator: $A_X(q) = q(X)$

The Empirical Estimator

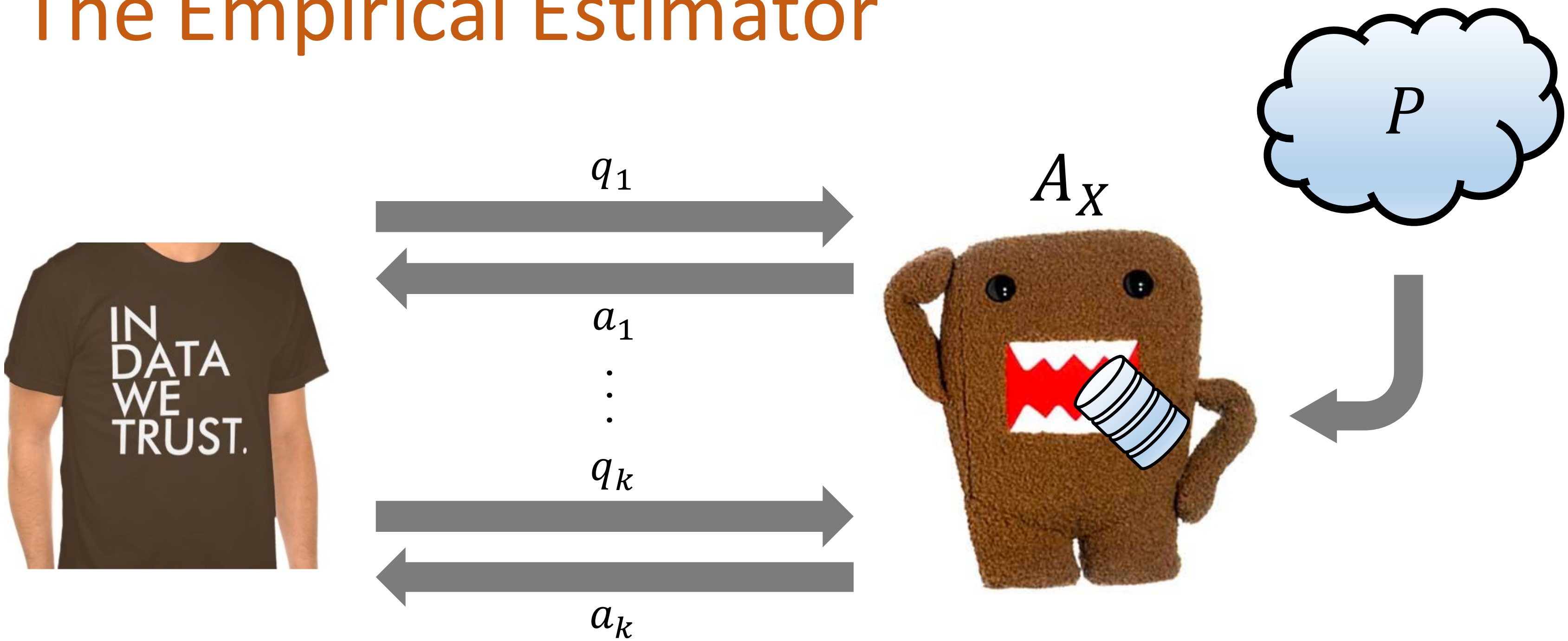


- Empirical estimator: $A_X(q) = q(X)$

Thm: For arbitrary **non-interactive** SQs,

$$\max_{j=1,\dots,k} \left| A_X(q_j) - q_j(P) \right| \lesssim \frac{\sqrt{\log k}}{\sqrt{n}}$$

The Empirical Estimator



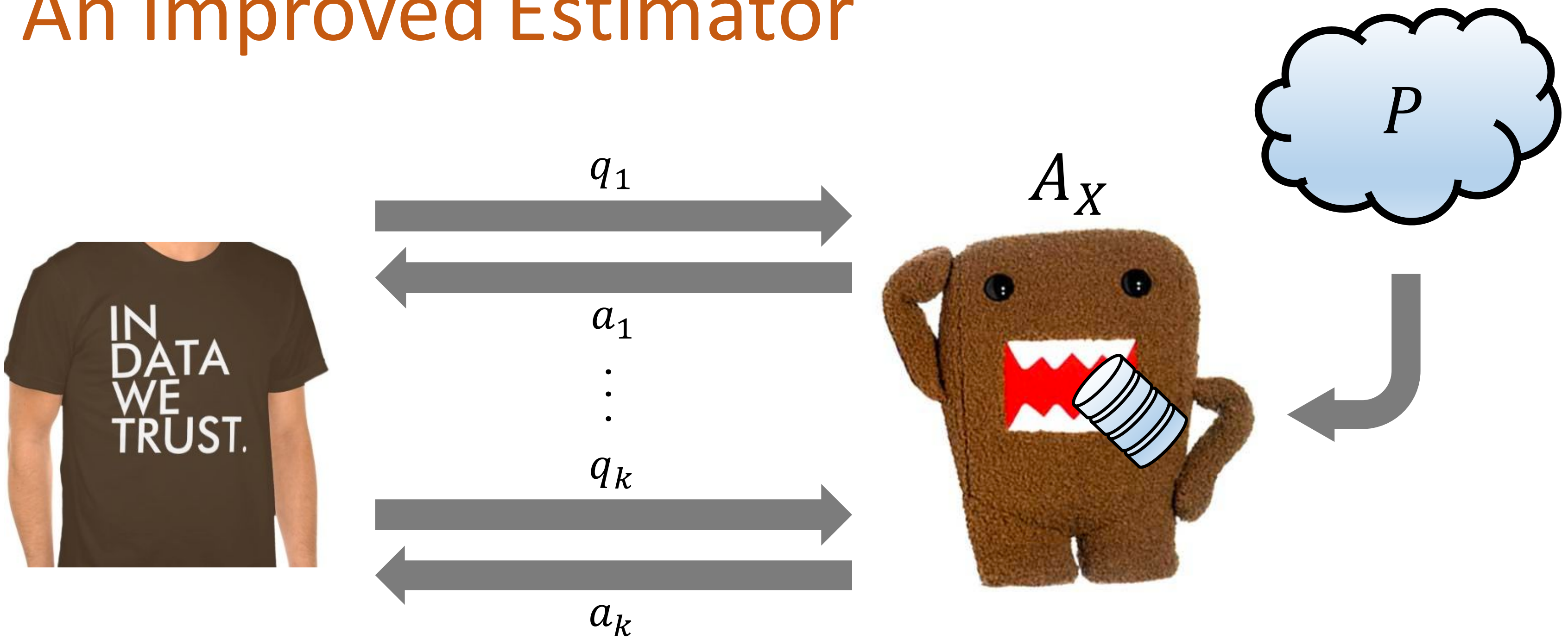
- Empirical estimator: $A_X(q) = q(X)$

Thm: For arbitrary **interactive** SQs,

$$\max_{j=1,\dots,k} \left| A_X(q_j) - q_j(P) \right| \lesssim \frac{\sqrt{k}}{\sqrt{n}}$$

See Freedman's Paradox!

An Improved Estimator



- Noisy empirical estimator: $A_X(q) = q(X) + N(0, \sigma^2)$

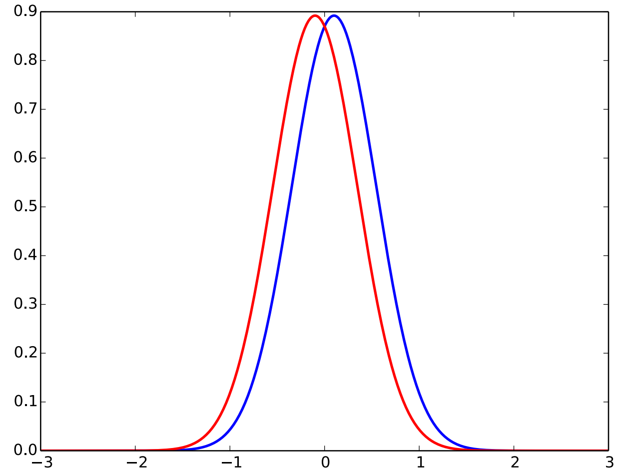
Thm [DFHPRR'15, BNSSS**U**'16]: For arbitrary **interactive** SQs,

$$\max_{j=1,\dots,k} \left| A_X(q_j) - q_j(P) \right| \lesssim \frac{\sqrt[4]{k}}{\sqrt{n}}$$

Adding noise reduces the error!

Proof Overview

- Claim 1: If $q_1, a_1, \ldots, q_k, a_k$ is a sequence of SQs and noisy empirical means, then $(\vec{q}, \vec{a})$ is DS
  - Stability parameters ε, δ will depend on n, k, σ
    - In this example, $\sigma \approx \sqrt[4]{k}/\sqrt{n}$
- Intuitively, the noise masks the influence of any one sample X_i on the mean $q(X) = \frac{1}{n}\sum_i \phi(X_i)$



$q(X) + N(0, \sigma^2)$

$q(X') + N(0, \sigma^2)$

Proof Overview

- Claim 2 [DFHPRR'15,BNSSS**U**'16]: If M is a DS algorithm mapping samples to SQs, then whp
$$q_{M(X)}(X) \approx q_{M(X)}(P)$$
  - Intuitively: no DS algorithm can output a query such that X and P are different (even though they exist).

- Why is Claim 2 useful?
  - Each query q_j is the output of some DS algorithm $M_j(X)$, so the queries satisfy $q_j(X) \approx q_j(P)$
  - The noisy answers a_j satisfy $a_j \approx q_j(X)$
  - Therefore $a_j \approx q_j(P)$

Proof Overview

- Claim 2' [DFHPRR'15,BNSSS**U**'16]: If M is a DS algorithm mapping samples to SQs, then

$$\mathbb{E}_{X,M}\left[q_{M(X)}(X)\right] \approx \mathbb{E}_{X,M}\left[q_{M(X)}(P)\right]$$

- Proof Sketch:
  - Consider $\left(i, X_i, q_{M(X)}\right)$ and $\left(i, Z, q_{M(X)}\right)$ where $i \sim [n]$, $X \sim P^n, Z \sim P$ independently, and M is randomized

$$\left(i, X_i, q_{M(X)}\right)$$

$$\approx_{\varepsilon,\delta} \left(i, X_i, q_{M(Z,X_{-i})}\right) \quad \text{Distributional Stability}$$

$$\approx \left(i, Z, q_{M(X_i,X_{-i})}\right) \quad \text{Symmetry}$$

$$\approx \left(i, Z, q_{M(X)}\right)$$

Summary of Results

Theorem [DFHPRR'15, BNSSSU'16]: There is an estimator A_X that answers any k interactive SQs with error

$$\alpha = \tilde{O}\left(\frac{\sqrt[4]{k}}{\sqrt{n}}\right)$$

- Adding independent Gaussian noise to the answers improves stability and reduces total error!
- Can extend to other types of queries
  - Lipschitz queries: $|q(X) - q(X')| \leq \frac{1}{n}$ [BNSSS**U**'16]
  - ERM queries: $q(X) = \operatorname{argmin}_{\theta \in \Theta} \ell(\theta; X)$ [BNSSS**U**'16]
  - Jointly Gaussian queries: $q(X) \sim N(\mu, \Sigma)$ [RZ'15, WLF'15, BF'16]

Summary of Results

Theorem [DFHPRR'15, BNSSSU'16]: There is an estimator A_X that answers any k interactive SQs with error

$$\alpha = \tilde{O}\left(\min\left\{\frac{\sqrt[4]{k}}{\sqrt{n}}, \frac{\sqrt[6]{d}\sqrt[3]{\log k}}{\sqrt[3]{n}}\right\}\right)$$

- When the data dimensionality is bounded (i.e. $U = \{\pm 1\}^d$), we can use more powerful DS algorithms from privacy
  - Can answer
- Two issues with this approach:
  - Statistical: Only improves when d is sufficiently small
  - Computational: Running time is exponential in d

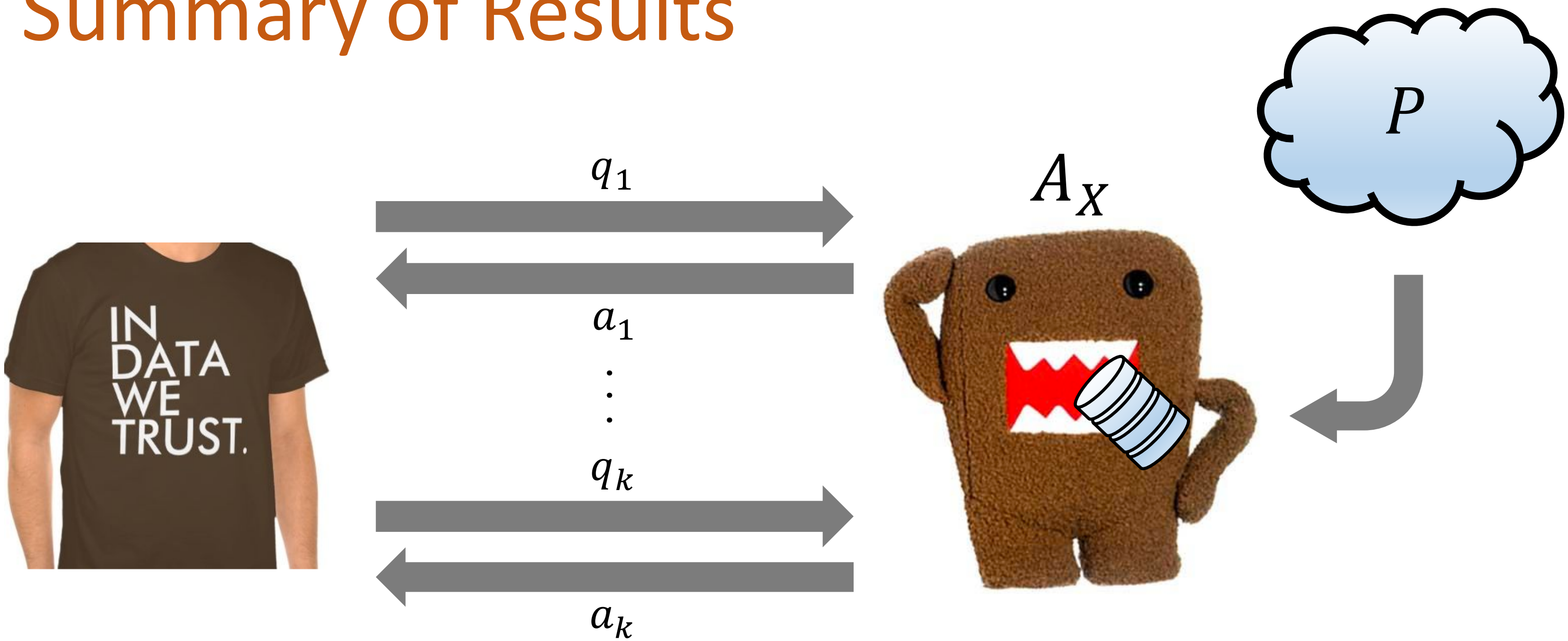
Summary of Results

Theorem [HU'14,SU'15]: If $k \gtrsim n^2$, and $d \geq k$, then there is a malicious analyst that forces **every estimator** to have error at least $1/3$.

Theorem [HU'14,SU'15]: If $k \gtrsim n^2$, and $d \gtrsim \log(n)$, then there is a malicious analyst that forces **every polynomial-time estimator** to have error at least $1/3$.

- Borrows techniques from differential privacy lower bounds [B**U**V'14,DSS**U**V'15], namely fingerprinting codes [BS'95,T'03]

Summary of Results



- There is a malicious analyst such that for any accurate estimator A_X, the analyst can learn the dataset X after $k = O(n^2)$ queries
  - Requires that A_X works for all P
  - Analyst must know P

Summary of Results

Theorem [DFHPRR'15]: If the k queries are issued in $r \ll k$ **rounds** then there is an estimator A_X with error

$$\alpha = \tilde{O}\left(\sqrt{\frac{r \log k}{n}}\right)$$

- Does not require knowing the timing of the rounds
- Application: re-usable holdout sets [DFHPRR'15]
  - Keep a holdout set, only use it to verify your conclusions
  - Each of the r rounds corresponds to one of your conclusions failing
  - "Only pay proportional to the number of times you truly overfit."

This Talk

- A general approach to interactive data analysis
  - Introduced in [DFHPRR'15, H**U**'14]
  - New general tools and methodology
  - Leads to new algorithms for preventing overfitting
- Key ingredient: algorithmic stability
  - Strong notions of stability inspired by differential privacy
  - Uses randomization to improve generalization
- New inherent bottlenecks [H**U**'14,S**U**'15]
  - Both statistical and computational

Thank you!