Algorithmic Stability for Interactive Data Analysis: An Overview

Jonathan Ullman, Northeastern University

Based on several (dis)joint works: [HU’14], [DFHPRR’15abc], [SU’15], [BNSSSU’16]

Statistical analysis guarantees that your conclusions generalize to the population
And Yet...

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, HU’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 [HU’14, SU’15]
  • Both statistical and computational
Overfitting in Interactive Data Analysis

- Population $P$ of uniformly random labeled examples
- Sample $X = (Y_1, Y_1), \ldots, (Y_n, Z_n) \in \{\pm 1\}^d \times \{\pm 1\}$
- Goal: find $h: \{\pm 1\}^d \to \{\pm 1\}$ 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 $j = 1, \ldots, d$ 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 $Z \in \{\pm 1\}^n$

\[
\begin{align*}
Z & \quad h_1 \quad \bar{h}_2 \quad \cdots \quad \bar{h}_d \\
\approx & \quad \frac{1}{\sqrt{n}} \quad \frac{1}{\sqrt{n}} \quad \frac{1}{\sqrt{n}} \quad \frac{1}{\sqrt{n}}
\end{align*}
\]
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 \( h^*(y) = \text{majority}(h_1(y), h_2(y), \ldots, h_k(y)) \)

Random labels \( Z \in \{\pm 1\}^n \)  
Labels of majority vote \( h^* \)

\[ s_X(h^*) = \Theta \left( \frac{\sqrt{d}}{\sqrt{n}} \right) \]

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]

Competing in a data science contest without reading the data

Mar 9, 2015 • Moritz Hardt

We see an improvement from 0.462311 (rank 146) to 0.451868 (rank 6).
How to Avoid This Trap?

• What went wrong?
  • The scores $s_X(h_1), \ldots, s_X(h_d)$ 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 $X, X'$,

$$d(A(X), A(X')) \leq \epsilon$$

• 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 $O \subseteq \text{Range}(A)$

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

- 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 $X, X'$

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

- 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, BNSSSU’16, BF’16, DR’16, BS’16, BDRS’17,...]
A General Framework

- A population $P$ over some universe $U$
- A sample $X = (X_1, ..., X_n)$ from $P$
- A class of statistics $Q$
  - For example “What fraction of $P$ has the property $q$?”
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
  \[ \phi: U \rightarrow [\pm 1] \]
- The statistical query \( q_\phi \) is defined as
  \[ q_\phi (P) = \mathbb{E} [\phi(P)] \]
- An answer \( a \) is \( \alpha \)-accurate if \( |a - q_\phi (P)| \leq \alpha \)

- 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,

$$\max_{j=1,\ldots,k} |A_X(q_j) - q_j(P)| \leq \frac{\sqrt{\log k}}{\sqrt{n}}$$
The Empirical Estimator

Empirical estimator: \( A_X(q) = q(X) \)

Thm: For arbitrary interactive SQs,
\[
\max_{j=1,...,k} \left| A_X(q_j) - q_j(P) \right| \lesssim \frac{\sqrt{k}}{\sqrt{n}}
\]
See Freedman’s Paradox!
An Improved Estimator

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

Thm [DFHPRR’15, BNSSSU’16]: For arbitrary interactive SQs,

\[
\max_{j=1,\ldots,k} |A_X(q_j) - q_j(P)| \leq \frac{4\sqrt{k}}{\sqrt{n}}
\]

Adding noise reduces the error!
Proof Overview

- **Claim 1**: If $q_1, a_1, ..., q_k, a_k$ is a sequence of SQs and noisy empirical means, then $(\tilde{q}, \tilde{a})$ is DS
  - Stability parameters $\varepsilon, \delta$ will depend on $n, k, \sigma$
  - In this example, $\sigma \approx \frac{4k}{\sqrt{n}}$
- 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, BNSSSU’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, BNSSSU’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 $(i, X_i, q_{M(X)})$ and $(i, Z, q_{M(X)})$ where $i \sim [n]$, $X \sim P^n$, $Z \sim P$ independently, and $M$ is randomized

  \[
  (i, X_i, q_{M(X)}) \\
  \approx_{\varepsilon, \delta} (i, X_i, q_{M(Z,X_{-i})}) \quad \text{Distributional Stability} \\
  \approx (i, Z, q_{M(X_{i}, X_{-i})}) \quad \text{Symmetry} \\
  \approx (i, Z, q_{M(X)})
  \]
Summary of Results

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

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

- Adding independent Gaussian noise to the answers improves stability and reduces total error!
- Can extend to other types of queries
  - Lipschitz queries: $|q(X) - q(X')| \leq \frac{1}{n}$ [BNSSSU’16]
  - ERM queries: $q(X) = \arg\min_{\theta \in \Theta} \ell(\theta; X)$ [BNSSSU’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

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

• 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 \( d \gtrsim k \), 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 [BUV’14,DSSUV’15], namely fingerprinting codes [BS’95,T’03]
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, HU’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 [HU’14, SU’15]
  - Both statistical and computational
Thank you!