Learning with Untrusted Data
New Models for Robust Learning and Optimization

Gregory Valiant
Based on three joint works with various subsets of:

Jacob Steinhardt  Michela Meister  Moses Charikar
To what extent can we develop algorithms for estimation, learning, and optimization that are robust to a significant fraction of untrusted data?

Setting:

• $n$ data points
• $\alpha n$ are drawn from the distribution of interest
• NO assumptions on remaining $(1-\alpha)n$ (i.e. could be very biased, noisy, arbitrary, or malicious even chosen as a function of the $\alpha n$ points).

Goal: answer as accurately as if you only have the $\alpha n$ “good” points.
Motivation: reliable/robust/secure ML

- **Security vulnerabilities** of the future (e.g. IoT, self-driving cars,...)
- Online community-enforced behavior standards (e.g. “like”/ “flag as inappropriate”/ “report”)
- **Crowdsourced data** curations (unreliable/fraudulent workers, malicious reviews, etc.), **peer-grading** (MOOCs), etc.

Often tradeoff between amount of data that can be collected, and quality of the data. Helpful to understand value of untrusted data.
Context
see last two talks : )
A Case for Optimism

$\alpha = \frac{2}{3}$  Goal: learn *mean* of spherical dist of radius 1

Intuition: if adversarial/arbitrary points are too weird, can identify and remove.
A Majority of Untrusted Data?

\( \alpha = \frac{1}{3} \)  

Goal: learn mean of spherical dist of radius 1

Impossible!
...but let's try anyway
“List-Covering Learning”

Setting:
• $n$ data points
• $\alpha n$ are drawn from dist. of interest
• NO assumptions on remaining $(1-\alpha)n$
• Goal: return a list of $< 1/\alpha$ answers [or poly(1/\alpha)] s.t. at least one is accurate.

Is the only hurdle in the $\alpha < 1/2$ case that we cannot decide among several options? If so, then no big deal....

Note: Positive results => ability to robustly learn mixture models with $1/\alpha$ components....
“Semi-Verified Learning”

$\alpha = 1/3$

Setting:

- $n$ data points
- $\alpha n$ are drawn from dist. of interest
- NO assumptions on remaining $(1-\alpha)n$
- $k$ “verified”/”trusted” datapoints drawn from dist. of interest.

For what problems is accurate estimation/learning/optimization possible in the semi-verified model using $k << n$ verified points?

How can the availability of a tiny amount of clean/trusted data facilitate the extraction of information from large, untrusted dataset?
Another View for $\alpha << 1/2$: A Database Prism

- Consider a large dataset of customer’s product evaluation: $m$ products, each customer evaluates $r$ products. 
  (e.g. $m =$ millions, $r = 5$)
- Long after the data is collected, suppose we want to understand preferences of one specific demographic of customers 
  (e.g. the student demographic)

**Our results imply:**
- Can hire CONSTANT number of students, ask them to review a CONSTANT number of products, then use this information, together with large dataset to accurately estimate preferences of student demographic on all $m$ products. *(Without having demographics info in dataset.)*
- Successful as long as fraction of student in the large dataset is not too small: $\alpha > \frac{1}{2^r}$
  
  e.g. If each customer evaluates $r =$10 products, can recover any demographic that is at least $1/1024$-fraction of population.
The Database Prism (or Aggregating Untrusted Ratings in Semi-Verified Model)

- $m$ items to evaluate
- $n$ reviewers, each will evaluate $r$ items
- $\alpha n$ give noisy but unbiased ratings
- no assumptions about behavior of remaining $(1-\alpha)n$ reviewers
- can obtain reviews of $k$ random items for the demographic of interest (and get noisy but unbiased evaluations)

*How large must $r$ and $k$ be to get accurate evals of $>(1-\varepsilon)$ fraction of items?*

- If $r=1$ then need $k = \Theta(n)$ [get no info from large dataset, must rely entirely on $k$ reviews]
Aggregating Untrusted Ratings in Semi-Verified Model

*How large must* $r$ and $k$ be to get accurate evals of $>(1-\varepsilon)$ fraction of items?*

- If $r = 1$ then need $k = \theta(n)$ : (Independent of $m,n$ !!!!)

**Thm I:** *Suffices to have* $r = \text{poly}(1/\alpha, 1/\varepsilon)$ and $k = \text{poly}(1/\alpha, 1/\varepsilon)$ and $n = O(m)$.

**Thm II:** *Suffices to have* $r = \log(1/\alpha)$ and $k = \text{poly}(1/\alpha, 1/\varepsilon)$ with sufficiently many reviewers ($n >> m$)
Thm I: Suffices to have \( r = \text{poly}(1/\alpha, 1/\varepsilon) \) and \( k = \text{poly}(1/\alpha, 1/\varepsilon) \) and \( n = O(m) \).

Alg intuition: Submatrix of \( \alpha n \) “good” reviewers noisy low-rank

\[
\begin{align*}
\max_{\mathbf{Y}} & \quad \langle \mathbf{X}, \mathbf{Y} \rangle \\
\text{s.t.} & \quad ||\mathbf{Y}||_* < (rnm/\alpha \varepsilon^2)^{1/2} \quad \text{bdd nuclear norm} \\
& \quad \text{for all } i : \Sigma_{j} Y_{ij} < rm \quad \text{no row “too powerful”}
\end{align*}
\]

- Submatrix of “good” rows of \( \mathbf{Y}_{\text{opt}} \) are all very good
- Use \( k \) “verified” points to identify “good” component of \( \mathbf{Y} \)
Beyond Rating Aggregation: General Learning/Optimization with Untrusted Data

- Observe $n$ convex functions $f_1, \ldots, f_n$
- $\alpha n$ drawn from $p^*$
- no assumptions about remaining $(1-\alpha)n$

**Goal:** find $w$ to minimize $E_{f \sim p^*}[f(w)]$
(in semi-verified or list-decodable sense)

Learning view: $f_i$ are loss-functions of sampled data points $x_1, \ldots, x_n$

e.g. Goal, output mean of distribution $D$. Given $x_1, \ldots, x_n$ interpret as sampling functions $f_i(w) = (w-x_i)^2$, goal: recover $\min_w E[f(w)]$

e.g. Linear/logistic regression, etc.
Thm III [informally]: Strong positive results, as long as matrix of gradients of “good” $f_i$’s has nice concentration.

Alg idea: try to find $w_i$ for EACH $f_i$ and encourage $w_i$’s to agree with each other.

$$\min_{w_1, \ldots, w_n, Y} \sum c_i f_i (w_i) + \text{tr}(Y)$$

s.t. $w_i w_i^T \leq Y$ (semidefinite constraint)

- Finds $w_i$ with small objective function value, subject to being contained in a small trace ellipse ($Y$)
- Will iteratively solve, and adjust weights $c_i$ to decrease effect of outliers.
Worst-case analysis (too pessimistic)

Avg case/distributional assumptions
(strong results possible, but assumptions suspect)

Robust/semi-verified/list-covering
(strong results + very weak assumptions??)

• Room for deep/interesting algorithms that might be extremely useful in practice
• Lots of open questions.