Computational and Statistical Tradeoffs via Convex Relaxation

Venkat Chandrasekaran
Caltech

Joint work with Michael Jordan
Time-constrained Inference

- Require decision after a fixed (usually small) amount of time
Classical Analysis in Statistics

- Previously, key bottleneck: amount of data

- Consider "best" estimator without much regard for computational considerations
  - Minimax analysis

- More recently, time is key bottleneck
  - Data is plentiful in several domains

- Need to incorporate time constraints
A Thought Experiment

- Consider a typical inference scenario
  - 1 hour for inference task with $n = 5000$, risk = 0.03
  - 20 days for same task with $n = 500000$, risk = 0.0003

- Happy with risk = 0.03, but given $n = 500000$
  - Don’t care about small improvements in risk
  - Statistical models are only approximations of reality
A Thought Experiment

- Consider a typical inference scenario
  - 1 hour for inference task with $n = 5000$, risk = 0.03
  - 20 days for same task with $n = 500000$, risk = 0.0003

- Happy with risk = 0.03, but given $n = 500000$
  - Don’t care about small improvements in risk
  - Statistical models are only approximations of reality

- More data useful for less computation?
Computer Science vs. Statistics

- Runtime
- Error
- Amount of data
- Time/data tradeoffs
- Numerical computation
- Statistics
Consider an inference problem with \textit{fixed} risk. Inference procedures viewed as points in plot.
Consider an inference problem with \textit{fixed} risk.

- Need \textit{“weaker”} algorithms for larger datasets.
- At some stage, throw away data.
- Tradeoff runtime \textit{upper bounds}.
  - More data means smaller runtime upper bound.
An Estimation Problem

- Signal $x^* \in S \subset \mathbb{R}^p$ from known (bounded) set
- Noise $z \sim \mathcal{N}(0, I_{p \times p})$

- Observation model

$$y = x^* + \sigma z$$

- Observe $n$ i.i.d. samples $\{y_i\}_{i=1}^n$
Convex Programming Estimator

- Sample mean $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$ is sufficient statistic

- Natural M-estimator

  $\hat{x}_n(S) = \arg \min_{x \in \mathbb{R}^p} \frac{1}{2} \| \bar{y} - x \|_{\ell_2}^2 \quad \text{s.t. } x \in S$

- Convex programming M-estimator

  $\hat{x}_n(C) = \arg \min_{x \in \mathbb{R}^p} \frac{1}{2} \| \bar{y} - x \|_{\ell_2}^2 \quad \text{s.t. } x \in C$

  - $C$ is a convex set such that $S \subset C$
Convex Programming Estimator

- Long history of shrinkage estimation in statistics
  - James, Stein (1961)
  - Donoho, Johnstone (early 1990s)
  - Shrinkage onto convex sets for tractability

- Many surprises in high dimensions, i.e., large $p$

- More recently
  - L1 norm, trace norm, max norm, ...
Defn 1: The **cone of feasible directions** into a convex set $C$ is defined as

$$T(x^*, C) = \text{cone}\{w - x^* | w \in C\}$$
Statistical Performance of Estimator

- **Defn 1**: The *cone of feasible directions* into a convex set $C$ is defined as

\[ T(x^*, C) = \text{cone}\{w - x^*|w \in C}\]

- **Defn 2**: The *Gaussian (squared) complexity* of a cone $T$ is defined as

\[ g(T) = \mathbb{E} \left[ \sup_{\delta \in T, \|\delta\|_2 \leq 1} \langle z, \delta \rangle^2 \right] \]
Statistical Performance of Estimator

- **Prop**: The risk of the estimator $\hat{x}_n(C)$ is

$$\mathbb{E} \left[ \| \hat{x}_n(C) - x^* \|_2^2 \right] \leq \frac{\sigma^2}{n} g \left( T(x^*, C) \right)$$

- **Proof**: Apply optimality conditions

- **Intuition**: Only consider error in feasible cone
Weakening via Convex Relaxation

- Prop: The risk of the estimator $\hat{x}_n(C)$ is

$$\mathbb{E} \left[ \|\hat{x}_n(C) - x^*\|_2^2 \right] \leq \frac{\sigma^2}{n} g\left(T(x^*, C)\right)$$

- Corr: To obtain risk of at most 1,

$$n \geq \sigma^2 g\left(T(x^*, C)\right)$$
Weakening via Convex Relaxation

- **Corr:** To obtain risk of at most 1,

\[
    n \geq \sigma^2 \ g\left(T(x^*, C')\right)
\]

Monotonic in \( C' \)

![Graphical representation of sets](image)
Weakening via Convex Relaxation

If we have access to larger $n$, can use larger $C$

$\rightarrow$ Obtain “weaker” estimation algorithm
Hierarchy of Convex Relaxations

- If $S$ "algebraic", then one can obtain family of outer convex approximations

\[
\text{conv}(S) \subseteq \cdots \subseteq C_3 \subseteq C_2 \subseteq C_1
\]

- Polyhedral, semidefinite, hyperbolic relaxations (Sherali-Adams, Boyd, Parrilo, Lasserre, Renegar)

- Sets $\{C_i\}$ ordered by computational complexity
  - Central role played by \textit{lift-and-project}
Contrast to Previous Work

- **Binary classifier learning**
  - Decatur et al. [1998], Servedio [2000], Shalev-Shwartz et al. [2008, 2012], Perkins & Hallett [2010]
  - Lots of extra data required for simpler algorithms
  - Our examples: modest extra data for simpler algorithms

- **Sparse PCA, clustering, network inference**
  - Amini & Wainwright [2009], Kolar et al. [2011]

- **Model selection**
  - Agarwal et al. [2011]
Contrast to Previous Work

- **Our work**: Emphasis on *algorithm weakening*

- **Convex relaxation** is a principled, general way to do this

\[
\text{conv}(S) \subseteq \cdots \subseteq C_3 \subseteq C_2 \subseteq C_1
\]
Example 1

○ $S$ consists of cut matrices

$$ S = \{ a a' \mid a \text{ consists of } \pm 1's \} $$

○ E.g., collaborative filtering, clustering

<table>
<thead>
<tr>
<th></th>
<th>Runtime</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv($S$) (cut polytope)</td>
<td>super-poly($p$)</td>
<td>$c_1 \sqrt{p}$</td>
</tr>
<tr>
<td>elliptope</td>
<td>$p^{2.25}$</td>
<td>$c_2 \sqrt{p}$</td>
</tr>
<tr>
<td>nuclear norm ball</td>
<td>$p^{1.5}$</td>
<td>$c_3 \sqrt{p}$</td>
</tr>
</tbody>
</table>

$(c_1 < c_2 < c_3)$
Example 2

- Signal set $\mathcal{S}$ consists of all perfect matchings in complete graph
- E.g., network inference

<table>
<thead>
<tr>
<th>$C$</th>
<th>Runtime</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{conv}(\mathcal{S})$</td>
<td>$p^5$</td>
<td>$c_1 \sqrt{p} \log(p)$</td>
</tr>
<tr>
<td>hypersimplex</td>
<td>$p^{1.5} \log(p)$</td>
<td>$c_2 \sqrt{p} \log(p)$</td>
</tr>
</tbody>
</table>

$(c_1 < c_2)$
Example 3

- $S$ consists of all adjacency matrices of graphs with only a clique on square-root of the nodes.
- E.g., sparse PCA, gene expression patterns.
- Kolar et al. (2010)

<table>
<thead>
<tr>
<th>$C$</th>
<th>Runtime</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{conv}(S)$</td>
<td>super-poly($p$)</td>
<td>$\sim p^{0.25} \log(p)$</td>
</tr>
<tr>
<td>nuclear norm ball</td>
<td>$p^{1.5}$</td>
<td>$\sim \sqrt{p}$</td>
</tr>
</tbody>
</table>
Some Questions

- In several examples, not too many extra samples required for really simple algorithms

- Quality of approximation of convex sets
  - *Approximation ratio* is focus in theoretical CS
  - *Gaussian complexities* in statistical inference

![Diagram](attachment:diagram.png)

- Approximation ratio in CS
- Gaussian complexity in statistics
Summary

- Challenges with massive datasets
- Considered simple denoising problem
- Time-data tradeoffs via convex relaxation

Future work:
- Gaussian complexities of LP/SDP hierarchies
- Other methods to weaken algorithms
- Other inference problems