Training Well-Generalizing Classifiers for Fairness Metrics and Other Data-Dependent Constraints

Andrew Cotter¹, Maya Gupta¹, Heinrich Jiang¹, Nathan Srebro², Karthik Sridharan³, Serena Wang¹, Blake Woodworth², Seungil You⁴

¹Google Research, ²Toyota Technological Institute at Chicago, ³Cornell University, ⁴Kakao Mobility
(Partly performed while N.S. was visiting Google, and S.Y. was employed by Google)
Constrained Optimization

\[
\begin{align*}
\text{minimize} & : \mathbb{E}_{x \sim D} [\ell_0 (x; \theta)] \\
\text{subject to} & : \mathbb{E}_{x \sim D} [\ell_i (x; \theta)] \leq 0, \quad \forall i \in \{1, 2, \ldots, m\}
\end{align*}
\]

- Applications include ML fairness, churn reduction, constraining true/false positive/negative rates, and more
- We want the constraints to hold *in expectation*, but will typically train using a finite training set. In other words, we’re interested in *constraint generalization*
- We give a “trick” for improving constraint generalization (at a cost to the objective function)
Intuition: Hyperparameter Optimization

\[ \mathcal{L}(\theta, \lambda) = \mathbb{E}_{x \sim \mathcal{D}} \left[ \ell_0(x; \theta) + \sum_{i=1}^{m} \lambda_i \ell_i(x; \theta) \right] \]

Thought Experiment

- Have two \textit{i.i.d.} samples, “training” and “validation”
  - a. For several fixed \( \lambda \)s, train a model \( \theta^*(\lambda) \) that minimizes the Lagrangian on the \textit{training set}
  - b. Choose a \( \lambda^* \) such that \( \theta^*(\lambda^*) \) satisfies the constraints on the \textit{validation set}
- If it works, validation constraint generalization will depend on the complexity of the space of Lagrange multipliers \( \lambda \), \textit{not} of the model parameters \( \theta \)
Two-Player-Game

\[ \mathcal{L}(\theta, \lambda) = \mathbb{E}_{x \sim \mathcal{D}} \left[ \ell_0(x; \theta) + \sum_{i=1}^{m} \lambda_i \ell_i(x; \theta) \right] \]

Our “trick” for improving constraint generalization:
- Think of constrained optimization as a two-player game
- Assign different independent samples to the two players

The resulting game is non-zero-sum:
- The two players have different datasets, so they optimize different functions
- In recent work [ALT’19], we considered a Lagrangian-like non-zero-sum game
  - Here, we extend this work to prove better constraint generalization bounds
Results - Upper Bounds

We provide several algorithms for playing this two-player game:

- Under certain assumptions, the in-expectation bounds satisfy the above
  - Instead of depending on the model complexity, the two-dataset infeasibility bound depends on the number of constraints

- We also perform experiments
  - In practice, using two independent datasets generally improves constraint generalization

### Suboptimality Bound vs. Infeasibility Bound

<table>
<thead>
<tr>
<th>One dataset:</th>
<th>Two datasets:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depends on model complexity (e.g. Rademacher)</td>
<td>Depends on model complexity</td>
</tr>
</tbody>
</table>
Thank You!
Poster: Pacific Ballroom #203

{acotter,mayagupta,heinrichj,serenawang}@google.com
{sridharan@cs.cornell.edu
{nati,blake}@ttic.edu
seungil.you@gmail.com