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

Andrew Cotter[1], Maya Gupta[1], Heinrich Jiang[1], Nathan Srebro[2], Karthik Sridharan[3], Serena Wang[1], Blake Woodworth[2], Seungil You[4]

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

Constrained Optimization

$$\underset{\theta \in \Theta}{\text{minimize}} : \mathbb{E}_{x \sim \mathcal{D}} \left[\ell_0 \left(x; \theta \right) \right]$$

$$\underset{\forall i \in \{1,2,\dots,m\}}{\text{subject to}} : \mathbb{E}_{x \sim \mathcal{D}} \left[\ell_i \left(x; \theta \right) \right] \leq 0$$

- 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

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

Thought Experiment

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

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

	Suboptimality Bound	Infeasibility Bound
One dataset:	Depends on model complexity (e.g. Rademacher)	
Two datasets:	Depends on model complexity	Independent of model complexity

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

{acotter,mayagupta,heinrichj,serenawang}@google.com

sridharan@cs.cornell.edu

Thank You!

Poster: Pacific Ballroom #203

{nati,blake}@ttic.edu

seungil.you@gmail.com