CS 165: Foundations of Machine Learning **Winter 2019**

Lecture 4: January 17th

Lecturer: Anima Anandkumar *Scribes: Ethan Pronovost, Chen Liang*

4.1 Competitive Optimization

In Competitive Optimization (a.k.a. multi-agent optimization), several different agents aim to influence a target function in different ways. We will consider the case of a two-agent min-max game, such as is used in Generative Adversarial Networks (GANs).

4.1.1 Min-Max Games

Two player min-max objectives take the form

$$\min_{\theta \in \mathbb{R}^p} \max_{\omega \in \mathbb{R}^q} U(\theta, \omega).$$

Such scenarios are common in adversarial zero-sum settings. Note that we can treat U as a function on $\mathbb{R}^{p+q}$.

The **first order conditions** for optimality are

$$\nabla_\theta U(\theta^*, \omega^*) = 0, \qquad \nabla_\omega U(\theta^*, \omega^*) = 0. \tag{4.1}$$

That is, (θ^*, ω^*) must be a stationary point of U.

The **second order conditions** for optimality are[1]

$$\nabla_{\theta\theta} U(\theta^*, \omega^*) \succ 0, \qquad \nabla_{\omega\omega} U(\theta^*, \omega^*) \prec 0. \tag{4.2}$$

That is, $U(\cdot, \omega^*)$ must have a local minimum at θ^*, and $U(\theta^*, \cdot)$ must have a local maximum at ω^*.

For a simple example of such a function, consider $U(\theta, \omega) = (\omega - \theta)^2 - 2\omega^2$. This has a unique equilibrium point at $(0, 0)$. For this case, the first order derivatives are

$$\nabla_\theta U(\theta, \omega) = 2(\theta - \omega)$$
$$\nabla_\omega U(\theta, \omega) = -2(\theta + \omega)$$

and the relevant second order derivatives are

$$\nabla_{\theta\theta} U(\theta, \omega) = 2$$
$$\nabla_{\omega\omega} U(\theta, \omega) = -4$$

As Figure 4.1 demonstrates, the equilibrium points are exactly saddle points of a specific form when the objective function is viewed as a function $U : \mathbb{R}^{p+q} \to \mathbb{R}$.

[1] N.B. Equations misstated in the lecture slides.

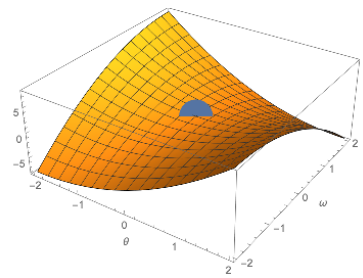

Figure 4.1: The target function $U(\theta, \omega) = (\omega - \theta)^2 - 2\omega^2$ has a unique equilibrium at $(0, 0)$.

4.1.2 Simultaneous Gradient Ascent

We are thus faced with the optimization problem of how to find equilibrium points. Perhaps the simplest idea we could try is to apply gradient descent to each of the two agents separately. That is, at each timestep t, we simultaneously update

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta U(\theta_t, \omega_t)$$
$$\omega_{t+1} = \omega_t + \eta \nabla_\omega U(\theta_t, \omega_t)$$

Note the sign change in the update for ω, as we are trying to maximize with respect to ω. This strategy is called **Simultaneous Gradient Ascent**[2] (SGA). Note that this algorithm ignores interaction between θ and ω.

To analyze the convergence of SGA, we will assume U obeys *Local Strong Convexity-Concavity.*

Definition 4.1 (Local Strong Convexity-Concavity) *Let $U(\theta, \omega) : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ be smooth and twice differentiable, and (θ^*, ω^*) an equilibrium point (i.e. satisfying eq. (4.1) and eq. (4.2)). Then there is some radius $r > 0$ such that for all $(\theta, \omega) \in B_2((\theta^*, \omega^*), r) = \{(\theta, \omega) \in \mathbb{R}^{p+q} : ||(\theta - \theta^*, \omega - \omega^*)||_2^2 < r\}$,*

$$\nabla_{\theta\theta} U(\theta, \omega) \succ 0 \qquad \nabla_{\omega\omega} U(\theta, \omega) \prec 0.$$

That is, in a local neighborhood of (θ^*, ω^*), U is strongly convex in θ and concave in ω.

We define the following block-wise abbreviation for the matrix of second derivatives:

$$\begin{bmatrix} \nabla_{\theta\theta} & \nabla_{\theta\omega} \\ -\nabla_{\omega\theta} & -\nabla_{\omega\omega} \end{bmatrix} = \begin{bmatrix} A & C \\ -C^T & B \end{bmatrix}$$

and define α, β as:

$$\alpha = \min_{(\theta,\omega) \in B_2((\theta^*,\omega^*),r)} \lambda_{\min}\left(\begin{bmatrix} A^2 & 0 \\ 0 & B^2 \end{bmatrix} \right)$$
$$\beta = \max_{(\theta,\omega) \in B_2((\theta^*,\omega^*),r)} \lambda_{\max}\left(\begin{bmatrix} A^2 + CC^T & -AC + CB \\ -C^T A + BC^T & B^2 + C^T C \end{bmatrix} \right)$$

where $\lambda_{\min}(M)$ and $\lambda_{\max}(M)$ are the minimum and maximum eigenvalues of M, respectively.

Then we have the following theorem (Liang and Stokes, 2018):

[2] Even though only ω is "ascending", we use this word to avoid having two things with the initials SGD.

Theorem 4.2 *Consider $U(\theta, \omega) : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ that near a local Nash equilibrium (θ^*, ω^*) has local strong convexity-concavity, i.e. $A \succeq 0, B \succeq 0$, then with appropriate initial point, the SGA with fixed learning rate $\eta = \sqrt{\alpha}/\beta$ can obtain an ϵ-minimizer as long as $T \geq \lceil 2\frac{\beta}{\alpha} \log \frac{r}{\epsilon} \rceil$.*

Let's consider another extreme case, where $A = B = 0$. Then we have $U(\theta, \omega) = \theta^T C \omega$ which is called bi-linear problem. Generally, SGA will lead to oscillation and instability (see the question in Problem Set 2 with $U(\theta, \omega) = \theta\omega$).

If U is not strongly convex-concave about (θ^*, ω^*), then A^2 or B^2 will have small eigenvalues. In this case α will be small, so it will take longer to converge.

If there is no interaction and $C = 0$, then the matrix in the expression for β is the same as that of α. In this case, $\frac{\beta}{\alpha}$ is the square of the condition number of

$$\begin{bmatrix} A^2 & 0 \\ 0 & B^2 \end{bmatrix} = \nabla^2 U,$$

which intuitively measures how sensitive U is to small changes in the input near (θ^*, ω^*).

To summarize, the SGA fails when 1) $A, B \approx 0$ or 2) $\|C\|$ is large. The main take away is that interactions between variables will **hurt** the SGA. To deal with the interactions, Predictive Method (PM) such as Optimistic Mirror Descent (OMD) or Consensus Optimization (CO) have been proposed. Interested readers are referred to (Liang and Stokes, 2018) for more discussions.

References

Liang, T. and J. Stokes
2018. Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. *arXiv preprint arXiv:1802.06132.*