1 Minimax Optimization

1.1 Convex-Concave Minimax

Let $\mathcal{X}, \mathcal{Y}$ be nonempty set, $\phi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R} \cup \{\pm\infty\}$. To simplify notation, let $f(\cdot) = \max_{y\in\mathcal{Y}} \phi(\cdot, y)$ and $g(\cdot) = \min_{x\in\mathcal{X}} \phi(x, \cdot)$. In general, we are intersted in the minimax problem of the form.

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \phi(x, y)$$

In particular, we want to find a saddle point, a pair $(x^\star, y^\star)$ where

$$x^\star \in \arg\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \phi(x, y) = \arg\min_{x\in\mathcal{X}} f(x)$$
$$y^\star \in \arg\max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} \phi(x, y) = \arg\max_{y\in\mathcal{Y}} g(y)$$

Definition. ***(Weak Minimax)*** *For any $\phi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R} \cup \{\pm\infty\}$, we have*

$$\max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} \phi(x, y) \le \min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \phi(x, y)$$

A good way to remember this is to note the follower (inner optimization) has advantage since it knows what the leader's (outer optimization) strategy. So when the follower wants to maximize the objective, it often achieves this.

Definition. ***(Strong Minimax)*** *If $\phi(\cdot, y)$ convex for all $y \in \mathcal{Y}$ and $\phi(x, \cdot)$ concave for all $x \in \mathcal{X}$, we have equality*

$$\max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} \phi(x, y) = \min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \phi(x, y)$$

1.2 Mirror Descent

1.3 Non-Convex-Non-Concave Minimax

Let $\mathbf{z}^k = (\mathbf{x}^k, \mathbf{y}^k)$ be the iterates, $f^k = f(\mathbf{z}^k)$ be function evaluated at current iterates, and

$$\mathbf{H}^k = \mathbf{J}(\nabla f^k) = \begin{bmatrix} \mathbf{H_{xx}} & \mathbf{H_{xy}} \\ \mathbf{H_{yx}} & \mathbf{H_{yy}} \end{bmatrix}$$

be block-wise hessian.

Competitive Gradient Descent (CGD) Competitive Gradient Descent [9] is an generalized gradient descent algorithm for the general-sum two player game, which we will specialize to zero-sum game. Each iteration involves solving a a quadratic regularized bilinear game that approximates the general game at the current iterate.

$$\min_{\boldsymbol{\delta}_\mathbf{x}} \max_{\boldsymbol{\delta}_\mathbf{y}} \left[F^k(\boldsymbol{\delta}_\mathbf{x}, \boldsymbol{\delta}_\mathbf{y}) := \boldsymbol{\delta}_\mathbf{x}^T \nabla_\mathbf{x} f^k + \boldsymbol{\delta}_\mathbf{y}^T \nabla_\mathbf{y} f^k + \frac{1}{2} \boldsymbol{\delta}_\mathbf{x}^T \mathbf{H}_{\mathbf{x},\mathbf{y}} \boldsymbol{\delta}_\mathbf{y} + \frac{1}{2\eta} \|\boldsymbol{\delta}_\mathbf{x}\|_2^2 - \frac{1}{2\eta} \|\boldsymbol{\delta}_\mathbf{y}\|_2^2 \right]$$

Finding 1st order local nash equilibrium involves solving a system of equations given by

$$0 = \nabla_{\boldsymbol{\delta}_\mathbf{x}} F^k = \nabla_\mathbf{x} f^k + \mathbf{H}_{\mathbf{xy}}^k \boldsymbol{\delta}_\mathbf{y} + \frac{1}{\eta} \boldsymbol{\delta}_\mathbf{x}$$
$$0 = \nabla_{\boldsymbol{\delta}_\mathbf{y}} F^k = -\nabla_\mathbf{y} f^k - \mathbf{H}_{\mathbf{yx}}^k \boldsymbol{\delta}_\mathbf{x} + \frac{1}{\eta} \boldsymbol{\delta}_\mathbf{y}$$

which emits closed form equations for $\boldsymbol{\delta}_{\mathbf{x}}, \boldsymbol{\delta}_{\mathbf{y}}$, giving rise to update of the form,

$$\begin{bmatrix} \mathbf{x}^{k+1} \\ \mathbf{y}^{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{x}^{k} \\ \mathbf{y}^{k} \end{bmatrix} - \eta \begin{bmatrix} \mathbf{I} & \eta \mathbf{H}_{\mathbf{xy}}^{k} \\ -\mathbf{H}_{\mathbf{yx}}^{k} & \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \nabla_{\mathbf{x}} f^{k} \\ -\nabla_{\mathbf{y}} f^{k} \end{bmatrix}$$

Note solving for local nash of a full quadratic Taylor approximation of the game at current iterates (include terms involving $\mathbf{H}_{\mathbf{xx}}, \mathbf{H}_{\mathbf{yy}}$) recovers damped and regularized Newton's method. In the paper, the author uses computes approximate matrix inverse to compute the updates.

CGD with cubic regularization We can apply per-player cubic regularization,

$$\min_{\boldsymbol{\delta}_{\mathbf{x}}} \max_{\boldsymbol{\delta}_{\mathbf{y}}} \left[F^{k}(\boldsymbol{\delta}_{\mathbf{x}}, \boldsymbol{\delta}_{\mathbf{y}}) := \boldsymbol{\delta}_{\mathbf{x}}^{T} \nabla_{\mathbf{x}} f^{k} + \boldsymbol{\delta}_{\mathbf{y}}^{T} \nabla_{\mathbf{y}} f^{k} + \frac{1}{2} \boldsymbol{\delta}_{\mathbf{x}}^{T} \mathbf{H}_{\mathbf{x},\mathbf{y}} \boldsymbol{\delta}_{\mathbf{y}} + \frac{L}{6} \left\| \boldsymbol{\delta}_{\mathbf{x}} \right\|_{2}^{3} - \frac{L}{6} \left\| \boldsymbol{\delta}_{\mathbf{y}} \right\|_{2}^{3} \right]$$

It is not possible to write analytic equation for optimal solution. Instead, the subproblem can be computed using first order gradient methods with guaranteed on convergence to local saddle points, e.g. extra-gradient [10], concensus optimization [11].

Follow the Ridge (FR) Gradient Descent/Ascent (GDA) fails to converge with any constant learning rate. *Follow-the-Ridge* modifies gradient descent-ascent by applying an asymmetric correction term on the leader's gradient step, which encourage players to stay on the ridge of the loss surface. The approach is proved to converge and only converge to *local minimax* under mild assumptions (f twice differentiable, thrice differentiable at critical points, $\mathbf{H}_{\mathbf{yy}}$ is invertible)

$$\begin{bmatrix} \mathbf{x}^{k+1} \\ \mathbf{y}^{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{x}^{k} \\ \mathbf{y}^{k} \end{bmatrix} - \begin{bmatrix} \eta_{\mathbf{x}} \mathbf{I} & \mathbf{0} \\ -\eta_{\mathbf{x}} \mathbf{H}_{\mathbf{yy}}^{-1} \mathbf{H}_{\mathbf{yx}} & \eta_{\mathbf{y}} \mathbf{I} \end{bmatrix} \begin{bmatrix} \nabla_{\mathbf{x}} f^{k} \\ \nabla_{\mathbf{y}} f^{k} \end{bmatrix}$$