Algorithmic Foundations of Learning

Lecture 9
Oracle Model. Gradient Descent Methods

Patrick Rebeschini

Department of Statistics
University of Oxford

Recap

- Training data: $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \{-1, 1\}$, with $\mathcal{X} \subseteq \mathbb{R}^d$
- Loss function: $\varphi : \mathbb{R} \to \mathbb{R}_+$ (**convex**: reasonable by Zhang's lemma)
- Predictors $\mathcal{A} = \{x \in \mathbb{R}^d \to a_w(x) : w \in \mathcal{W}\}$ ($\mathcal{W}$ **convex** in many cases)
  NB. There are many settings where $\mathcal{A}$ is **not** convex (e.g., neural networks)

Risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(a_w(X)Y) \\ \text{subject to} \quad & w \in \mathcal{W} \end{aligned} \qquad \Longrightarrow \qquad \text{Let } w^\star \text{ be a minimizer}$$

Empirical risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(a_w(X_i)Y_i) \\ \text{subject to} \quad & w \in \mathcal{W} \end{aligned} \qquad \Longrightarrow \qquad \text{Let } W^\star \text{ be a minimizer}$$

$$\boxed{r(W) - r(w^\star) \le \underbrace{R(W) - R(W^\star)}_{\texttt{Optimization}} + \underbrace{\sup_{w \in \mathcal{W}} \{r(w) - R(w)\} + \sup_{w \in \mathcal{W}} \{R(w) - r(w)\}}_{\texttt{Statistics}}}$$

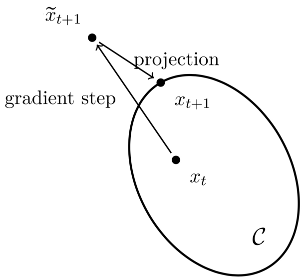
Projected Subgradient Method

Goal: $\boxed{\min_{x \in \mathcal{C}} f(x)}$ with f convex, $\mathcal{C}$ convex and compact

Projected Subgradient Method

$$\boxed{\begin{aligned} &\tilde{x}_{t+1} = x_t - \eta_t g_t, \text{where } g_t \in \partial f(x_t) \\ &x_{t+1} = \Pi_{\mathcal{C}}(\tilde{x}_{t+1}) \end{aligned}}$$

with the projection operator $\Pi_{\mathcal{C}}(y) = \operatorname{argmin}_{x \in \mathcal{C}} \|x - y\|_2$.



Non-Expansivity of Projections

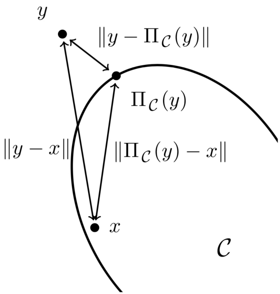
Non-expansivity (Proposition 9.2)

Let $x \in \mathcal{C}$ and $y \in \mathbb{R}^d$. Then,

$$(\Pi_{\mathcal{C}}(y) - x)^\top (\Pi_{\mathcal{C}}(y) - y) \leq 0$$

which implies $\|\Pi_{\mathcal{C}}(y) - x\|_2^2 + \|y - \Pi_{\mathcal{C}}(y)\|_2^2 \leq \|y - x\|_2^2$ and, in particular,

$$\boxed{\|\Pi_{\mathcal{C}}(y) - x\|_2 \leq \|y - x\|_2}$$



First Order Optimality Condition

First Order Optimality Condition (Proposition 8.10)

Let f be convex, and $\mathcal{C}$ be a closed set on which f is differentiable. Then,

$$x^\star \in \operatorname*{argmin}_{x \in \mathcal{C}} f(x) \quad \Longleftrightarrow \quad \nabla f(x^\star)^\top (x^\star - x) \leq 0 \quad \text{for any } x \in \mathcal{C}$$

Proof of Proposition 9.2. This is a direct consequence of Proposition 8.10 since $\Pi_{\mathcal{C}}(y)$ is a minimizer of the function $z \to f_y(z) = \|y - z\|_2$, and $\nabla f_y(z) = (z - y)/\|z - y\|_2$.

Results for Lipschitz Functions

A function f is γ-**Lipschitz on** $\mathcal{C}$ if there exists $\gamma > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(x) - \gamma\|x - y\|_2 \leq f(y) \leq f(x) + \gamma\|x - y\|_2$
- For every $x, y \in \mathcal{C}$, $|f(y) - f(x)| \leq \gamma\|x - y\|_2$
- For every $x \in \mathcal{C}$, any subgradient $g \in \partial f(x)$ satisfies $\|g\|_2 \leq \gamma$

Projected Subgradient Method—Lipschitz (Theorem 9.3)

- Function f is γ-Lipschitz
- Assume $\|x_1 - x^\star\|_2 \leq b$

Then, the projected subgradient method with $\eta_s \equiv \eta = \frac{b}{\gamma\sqrt{t}}$ satisfies

$$\boxed{f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \leq \frac{\gamma b}{\sqrt{t}}}$$

- It is **not** a descent method: the value function can increase in one time step
- The reference point $x^\star$ can be anything, not just a minimizer of f

Proof of Theorem 9.3)

- Convexity yields:

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \leq \frac{1}{t}\sum_{s=1}^{t} f(x_s) - f(x^\star) \leq \frac{1}{t}\sum_{s=1}^{t} g_s^\top (x_s - x^\star)$$

- Using $2a^\top b = \|a\|_2^2 + \|b\|_2^2 - \|a-b\|_2^2$ and $g_s = \frac{1}{\eta}(x_s - \tilde{x}_{s+1})$:

$$\begin{aligned} g_s^\top (x_s - x^\star) &= \frac{1}{\eta}(x_s - \tilde{x}_{s+1})^\top (x_s - x^\star) \\ &= \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 + \|x_s - \tilde{x}_{s+1}\|_2^2 - \|\tilde{x}_{s+1} - x^\star\|_2^2\right) \\ &= \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 - \|\tilde{x}_{s+1} - x^\star\|_2^2\right) + \frac{\eta}{2}\|g_s\|_2^2 \\ &\leq \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 - \|x_{s+1} - x^\star\|_2^2\right) + \frac{\eta}{2}\|g_s\|_2^2 \end{aligned}$$

  where we used that $\|\tilde{x}_{s+1} - x^\star\|_2 \geq \|x_{s+1} - x^\star\|_2$ by Proposition 9.2.
- Summing from $s = 1$ to t:

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \leq \frac{1}{2\eta t}\left(\|x_1 - x^\star\|_2^2 - \|x_{t+1} - x^\star\|_2^2\right) + \frac{\eta\gamma^2}{2} \leq \frac{b^2}{2\eta t} + \frac{\eta\gamma^2}{2}$$

  Minimizing the right-hand side we have $\eta = \frac{b}{\gamma\sqrt{t}}$ which yields the result.

Results for Smooth Functions

A function f is β-**smooth on** $\mathcal{C}$ if there exists $\beta > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(y) \le f(x) + \nabla f(x)^\top (y - x) + \frac{\beta}{2}\|y - x\|_2^2$
- For every $x, y \in \mathcal{C}$, $|\nabla f(y) - \nabla f(x)| \le \beta \|x - y\|_2$ (gradient is β-Lipschitz)
- For every $x \in \mathcal{C}$, $\nabla^2 f(x) \preccurlyeq \beta I$ (if f is twice-differentiable)

Projected Gradient Descent—Smooth (Theorem 9.4)

- Function f is β-smooth
- Assume $\|x_1 - x^\star\|_2 \le b$

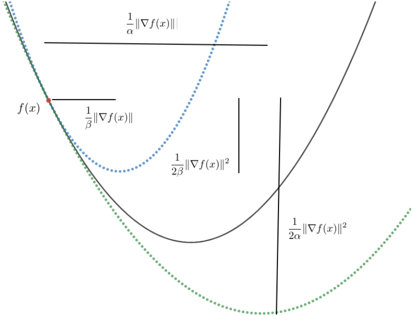
Then, projected gradient descent with $\eta_s \equiv \eta = 1/\beta$ satisfies

$$\boxed{f(x_t) - f(x^\star) \le \frac{3\beta b^2 + f(x_1) - f(x^\star)}{t}}$$

In the case of smooth functions, gradient descent is a natural algorithm...

Interpretation for Smooth Functions

... it is the algorithm that at each time step moves to the point in $\mathcal{C}$ that maximizes the guaranteed local decrease given by the quadratic function that uniformly upper-bounds the function f at the current location



$$\operatorname*{argmin}_{y\in\mathcal{C}}\left\{f(x)+\nabla f(x)^\top(y-x)+\frac{\beta}{2}\|y-x\|_2^2\right\}=\operatorname*{argmin}_{y\in\mathcal{C}}\left\{\left\|\left(x-\frac{1}{\beta}\nabla f(x)\right)-y\right\|_2^2\right\}$$

$$\equiv \Pi_{\mathcal{C}}\left(x-\frac{1}{\beta}\nabla f(x)\right)$$

Results for Smooth and Strongly Convex Functions

A function f is α-**strongly convex on** $\mathcal{C}$ if there is $\alpha > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\alpha}{2}\|y - x\|_2^2$
- For every $x \in \mathcal{C}$, $\nabla^2 f(x) \succcurlyeq \alpha I$ (if f is twice-differentiable)

Gradient Descent—Smooth and Strongly Convex (Theorem 9.5)

- Assume $\mathcal{C} = \mathbb{R}^d$ (same type of result holds for projected gradient descent)
- Function f is α-strongly convex and β-smooth

Then, gradient descent with $\eta_s \equiv \eta = 1/\beta$ satisfies

$$\boxed{f(x_t) - f(x^\star) \leq \left(1 - \frac{\alpha}{\beta}\right)^{t-1} (f(x_1) - f(x^\star))}$$

Proof: (see illustration on the previous slide)

- Guaranteed progress in one step: $f(x_{s+1}) \leq f(x_s) - \frac{1}{2\beta}\|\nabla f(x_s)\|_2^2$
- Lower bound on objective function: $f(x^\star) \geq f(x_s) - \frac{1}{2\alpha}\|\nabla f(x_s)\|_2^2$

Oracle Complexity, Lower Bounds, Accelerated Methods

- **Convergence rates:**

	L-Lipschitz	β-smooth
Convex	$O(\gamma b/\sqrt{t})$	$O((\beta b^2 + c)/t)$
α-strongly convex	$O(\gamma^2/(\alpha t))$	$O(e^{-t\alpha/\beta} c)$

where $\|x_1 - x^\star\|_2 \leq b$ and $f(x_1) - f(x^\star) \leq c$

- **Oracle complexities:**

	L-Lipschitz	β-smooth
Convex	$O(\gamma^2 b^2/\varepsilon^2)$	$O((\beta b^2 + c)/\varepsilon)$
α-strongly convex	$O(\gamma^2/(\alpha\varepsilon))$	$O((\beta/\alpha)\log(c/\varepsilon))$

- **Optimal rates (lower bounds)**

	L-Lipschitz	β-smooth
Convex	$\Omega(\gamma a/(1+\sqrt{t}))$	$\Omega(\tilde{b}^2\beta/(t+1)^2)$
α-strongly convex	$\Omega(\gamma^2/(\alpha t))$	$\Omega(\alpha\tilde{b}^2 e^{-t\sqrt{\alpha/\beta}})$

where $a := \max_{x\in\mathcal{C}} \|x\|_2$ and $\tilde{b} := \max_{x,y\in\mathcal{C}} \|x-y\|_2$

Apart from Lipschitz, optimal rates are achieved only by **accelerated** algorithms

NB. Quantities α, β, γ and $a, b, c, \tilde{b}$ depend implicitly on dimension d

Back to Learning: Linear Predictors with ℓ_2 Ball

Risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(w^\top XY) \\ \text{subject to} \quad & \|w\|_2 \leq c_2^{\mathcal{W}} \end{aligned} \quad \Longrightarrow \quad \text{Let } w^\star \text{ be a minimizer}$$

Empirical risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(w^\top X_i Y_i) \\ \text{subject to} \quad & \|w\|_2 \leq c_2^{\mathcal{W}} \end{aligned} \quad \Longrightarrow \quad \text{Let } W^\star \text{ be a minimizer}$$

$$\boxed{r(\overline{W}_t) - r(w^\star) \leq \underbrace{R(\overline{W}_t) - R(W^\star)}_{\texttt{Optimization}} + \underbrace{\sup_{w\in\mathcal{W}}\{r(w) - R(w)\} + \sup_{w\in\mathcal{W}}\{R(w) - r(w)\}}_{\texttt{Statistics}}}$$

$$\boxed{\mathbf{E}\,\texttt{Statistics} \leq \frac{4c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{n}}} \qquad \boxed{\texttt{Optimization} \leq \frac{2c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{t}}}$$

Principled approach: Enough to run algorithm for $t \sim n$ time steps
(ONLY BASED ON UPPER BOUNDS!)