Conservative Projection Algorithm

Siddhant Chaudhary

CMI, December 2021

Quick Recap

- We are in the standard OCO setting.
- $\Theta \subseteq \mathbf{R}^d$: a convex domain.
- $f_t : \Theta \to [\epsilon_l, \epsilon_u]$: convex bounded differentiable functions.
- Objective of the learner $\mathcal{U}$ is to minimise the regret.

$$R_T(\mathcal{U}) := L_T - \overline{L}_T$$

- But with a *conservativeness constraint*.

Conservativeness

- We pick a default parameter $\tilde{\theta} \in \Theta$ before the learning process.
- For all time steps t, we want the following.

$$L_t \leq (1+\alpha)\tilde{L}_t$$

  Above, L_t is the loss our algorithm incurs till time t, and $\tilde{L}_t$ is the loss that the fixed strategy $\tilde{\theta}$ incurs till time t. $\alpha > 0$ is the *conservativeness level*.
- Define the budget $Z_t(\mathcal{U})$ as follows.

$$Z_t(\mathcal{U}) := (1+\alpha)\tilde{L}_t - L_t$$

- We assume that $\tilde{L}_t \geq \mu t$ for some $\mu > \epsilon_l$, i.e the fixed strategy $\tilde{\theta}$ is sub-optimal.

Conservative Ball

- At all time steps t, we defined a ball $B(\tilde{\theta}, r_t) \subseteq \mathbf{R}^d$ with the following radius.

$$r_t := \left[1 - \left(\frac{L_{t-1} - (1+\alpha)\tilde{L}_{t-1} - \alpha\epsilon_l}{DG} + 1 \right)^+ \right] D$$

- D is a bound on the diameter of the domain Θ.
- G is an upper bound on the gradients norms $||f_t(x)||_2$.
- $a^+ =$ max$(0, a)$.
- Defining the radius this way ensures the conservativeness constraint, as given on the next slide.

Points in the conservative ball

Theorem

Suppose the conservativeness constraint is satisfied at time $t-1$, i.e

$$(1+\alpha)\tilde{L}_{t-1} - L_{t-1} \geq 0$$

Then, each point $\theta \in B(\tilde{\theta}, r_t) \cap \Theta$ satisfies the conservativeness constraint at time t, i.e

$$(1+\alpha)\tilde{L}_t - L_t \geq 0$$

- Computing projections onto a ball is easy, and can be done in time $O(d)$ (here d is the dimension of the space).

Formula for the projection

- Let z_t be any point, and consider the conservative ball $B(\tilde{\theta}, r_t)$.
- Then the following holds.

$$\theta_t = \Pi_{B(\tilde{\theta}, r_t)}(z_t) = \beta_t \tilde{\theta} + (1 - \beta_t) z_t$$

where

$$\beta_t = \begin{cases} 1 - \frac{r_t}{||z_t - \tilde{\theta}||_2} & , \quad z_t \notin B(\tilde{\theta}, r_t) \\ 0 & , \quad z_t \in B(\tilde{\theta}, r_t) \end{cases}$$

The CP algorithm

- Suppose we have access to an OCO algorithm $\mathcal{A}$, a conservativeness level $\alpha > 0$ and a default parameter $\tilde{\theta} \in \Theta$.
- Suppose we have chosen $\theta_1, ..., \theta_{t-1}$.
- We set $\tilde{L}_0 = 0$, $L_0 = 0$ and $\beta_0 = 1$. So, $\theta_0 = \tilde{\theta}$.
- At each time step t, we feed the algorithm $\mathcal{A}$ a loss function g_{t-1}, and get a prediction z_t. g_{t-1} is defined as follows.

$$g_{t-1}(x) = (1 - \beta_{t-1})f_{t-1}(x)$$

(contd.)

- We then compute r_t, and project z_t onto the conservative ball.

$$\theta_t = \Pi_{B(\tilde{\theta}, r_t)}(z_t)$$

- Then suffer loss $f_t(\theta_t)$.
- Also observe $f_t(z_t)$ (loss of the algorithm $\mathcal{A}$) and $f_t(\tilde{\theta})$ (loss of the fixed strategy).
- Repeat for T steps.

Pseudocode

Input: Online algorithm $\mathcal{A}$, conservativeness level $\alpha > 0$, default parameter $\tilde{\theta} \in \Theta$.
$\tilde{L}_0 \leftarrow 0$, $L_0 \leftarrow 0$, $\beta_0 \leftarrow 1$.
for $t \in [T]$ **do**
    Feed $\mathcal{A}$ the loss function g_{t-1} and get point z_t.
    Compute r_t.
    Set $\theta_t = \Pi_{B(\tilde{\theta}, r_t)}(z_t)$.
    Get loss $f_t(\theta_t)$.
    Also get $f_t(z_t)$ and $f_t(\tilde{\theta})$.
    Set $g_t(x) \leftarrow (1 - \beta_t) f_t(x)$
end for

Some Simple Observations

- The CP algorithm has small overhead w.r.t the subroutine $\mathcal{A}$, i.e an overhead proportional to d (this comes from computing the projection and the losses $f_t(\theta_t)$, $f_t(\tilde{\theta})$).

Corollary of earlier theorem

The CP algorithm applied to a generic online learning algorithm $\mathcal{A}$ is conservative.

- So, we only need to prove regret bounds.
- For sublinear regret algorithms $\mathcal{A}$, the CP algorithm also has sublinear regret!

Main Result 1

Theorem 1

Let $\mathcal{A}$ be any OCO algorithm that guarantees a regret of $R_T(\mathcal{A}) \leq \xi\sqrt{T}$. Then, the CP algorithm ensures the following inequality for all T.

$$L_T - \tilde{L}_T \leq \xi\sqrt{T}$$

In other words, the CP algorithm has sub-linear regret w.r.t an algorithm that always chooses the default parameter $\tilde{\theta}$.

Proof of Theorem 1

- Note that $\theta_t = \beta_t \tilde{\theta} + (1 - \beta_t) z_t$. Using the convexity of the f_ts, we get the following.

$$L_T = \sum_{t=1}^{T} f_t(\theta_t) \leq \sum_{t=1}^{T} [\beta_t f_t(\tilde{\theta}) + (1 - \beta_t) f_t(z_t)]$$

- So, we get

$$\begin{aligned} L_T - \tilde{L}_T &\leq \sum_{t=1}^{T} [\beta_t f_t(\tilde{\theta}) + (1 - \beta_t) f_t(z_t) - f_t(\tilde{\theta})] \\ &= \sum_{t=1}^{T} (1 - \beta_t)[f_t(z_t) - f_t(\tilde{\theta})] \\ &= \sum_{t=1}^{T} g_t(z_t) - g_t(\tilde{\theta}) \end{aligned}$$

(contd.)

- But clearly,

$$\sum_{t=1}^{T} g_t(z_t) - g_t(\tilde{\theta}) \leq \sup_{\theta \in \Theta} \left(\sum_{t=1}^{T} g_T(z_t) - g_T(\theta) \right)$$
$$\leq \xi \sqrt{T}$$

- The last inequality is true by the regret bound of $\mathcal{A}$. This completes the proof.

Main Result 2

Theorem 2

Let $\mathcal{A}$ be any OCO algorithm with regret bound $R_T(\mathcal{A}) \leq \xi\sqrt{T}$. Then, the CP algorithm using $\mathcal{A}$ as a subroutine has the following regret bound for all $T > \tau$.

$$R_T(\mathsf{CP}) \leq \xi\sqrt{T} + \tau DG$$

Here, τ is the solution of the equation

$$1 + \frac{\xi\sqrt{\tau} - (\tau - 1)\mu\alpha}{DG} = 0$$

Proof of Theorem 2

1. First we show that eventually at all times t, the point z_t produced by the algorithm $\mathcal{A}$ lies in the conservative ball $B(\tilde{\theta}, r_t)$; in other words, the CP algorithm and the algorithm $\mathcal{A}$ eventually predict the same thing.
2. This is equivalent to showing that there is some τ such that for all $t > \tau$, $\beta_t = 0$. As you might have guessed, this τ will be the same as in the theorem statement.
3. So suppose $0 < \beta_t < 1$ for some t. Recall that

$$\beta_t = 1 - \frac{r_t}{\left\|z_t - \tilde{\theta}\right\|_2}$$

(contd.)

- Note that it must be true that $r_t < D$; otherwise a contradiction, because $\left|\left|z_t - \tilde{\theta}\right|\right|_2 \leq D$).
- So because $r_t < D$, we must have (recall the definition of r_t)

$$r_t = \frac{(1+\alpha)\tilde{L}_{t-1} - L_{t-1} + \alpha\epsilon_l}{G}$$

- So in this case, we have

$$\begin{aligned}\beta_t &= 1 - \frac{r_t}{\left|\left|z_t - \tilde{\theta}\right|\right|_2} \\ &\leq 1 - \frac{r_t}{D}\end{aligned}$$

(contd.)

- Substituting the value of r_t, we will get

$$\beta_t \leq 1 + \frac{L_{t-1} - (1+\alpha)\tilde{L}_{t-1} - \alpha\epsilon_l}{DG}$$
$$= 1 + \frac{L_{t-1} - \tilde{L}_{t-1} - \alpha(\tilde{L}_{t-1} + \epsilon_l)}{DG}$$

- Now we use **Theorem 1** and also the fact that there exists $\mu > \epsilon_l > 0$ such that $\tilde{L}_{t-1} > \mu(t-1)$.
- Doing so, we get

$$\beta_t \leq 1 + \frac{\xi\sqrt{t} - (t-1)\mu\alpha}{DG}$$

- The RHS goes to $-\infty$ as $t \to \infty$.

(contd.)

- So we compute the zero of the quadratic in $\sqrt{t}$, which is the RHS of the previous inequality. Doing so, we get

$$\tau = \frac{2\alpha\mu(DG + \alpha\mu) + \xi(\sqrt{\xi^2 + 4\alpha\mu(DG + \alpha\mu)} + \xi)}{2\alpha^2\mu^2}$$

- So for all $t > \tau$, $\beta_t = 0$. This also means that for such t, $g_t = f_t$, i.e the losses fed into $\mathcal{A}$ and the actual losses f_t coincide.
- So, by writing the regret of CP till time T as the regret till time τ and the regret in the interval $[\tau + 1, T]$, we can easily obtain

$$R_T(\text{CP}) \leq \tau DG + \xi\sqrt{T}$$

Logarithmic Bounds

- Some OCO algorithms $\mathcal{A}$ can achieve $\rho O(\log T)$ regret bounds.
- Using the exact same strategy as before (with minor modifications), we can obtain

$$R_T(\mathsf{CP}) \leq \rho \log(T) + \tau DG$$