Labs
Optimization for Machine Learning
Spring 2022

EPFL
School of Computer and Communication Sciences
Martin Jaggi & Nicolas Flammarion
github.com/epfml/OptML_course

Problem Set 9 — Solutions (Frank-Wolfe)

Convergence of Frank-Wolfe

Exercise 63. *Given some constant $C > 0$ and a sequence of real values $h_0, h_1, \dots$ satisfying* (10.12)*, i.e.*

$$h_{t+1} \le (1-\gamma_t)h_t + \gamma_t^2 C \qquad \text{for } t = 0,1,\dots$$

for $\gamma = \frac{2}{t+2}$, prove that

$$h_t \le \frac{4C}{t+1} \qquad \text{for } t \ge 1.$$

Solution: Proof by induction. The base case $t = 1$ follows directly from applying (10.12) for $\gamma_0 = \frac{2}{0+2} = 1$ in which case $h_1 \le C$ is obtained. For the induction step, considering $t \ge 1$, we have

$$\begin{aligned} h_{t+1} \le\ & (1-\gamma_t)h_t + {\gamma_t}^2 C \\ =\ & \left(1 - \tfrac{2}{t+2}\right)h_t + \left(\tfrac{2}{t+2}\right)^2 C \\ \le\ & \left(1 - \tfrac{2}{t+2}\right)\tfrac{4C}{t+1} + \left(\tfrac{2}{t+2}\right)^2 C \ , \end{aligned}$$

where in the last inequality we have used the induction hypothesis for h_t. Simply rearranging the terms gives

$$\begin{aligned} h_{t+1} \le\ & \tfrac{4C}{t+2}\left(\tfrac{t}{t+1} + \tfrac{1}{t+2}\right) \\ \le\ & \tfrac{4C}{t+2}, \end{aligned}$$

which is our claimed bound for $t+1$.

Exercise 64. *Prove Lemma 10.6:*

Solution: By the definition of smoothness (Definition 2.2), we have that for any $\mathbf{x}, \mathbf{y} \in X$,

$$f(\mathbf{y}) - f(\mathbf{x}) - (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) \le \tfrac{L}{2}\left\|\mathbf{y}-\mathbf{x}\right\|^2 \ .$$

We want to use this upper bound in the definition (10.15) of the curvature constant. Observing that for any $\mathbf{x}, \mathbf{s} \in X$, we have that also $\mathbf{y} := \mathbf{x} + \gamma(\mathbf{s}-\mathbf{x}) \in X$ and $1/\gamma^2 \left\|\mathbf{y}-\mathbf{x}\right\|^2 = \left\|\mathbf{s}-\mathbf{x}\right\|^2$, we can therefore upper bound the curvature as

$$C_{(f,X)} \le \sup_{\substack{\mathbf{x},\mathbf{s}\in X,\\ \gamma\in(0,1],\\ \mathbf{y}=\mathbf{x}+\gamma(\mathbf{s}-\mathbf{x})}} \frac{1}{\gamma^2}\frac{L}{2}\left\|\mathbf{y}-\mathbf{x}\right\|^2 = \sup_{\mathbf{x},\mathbf{s}\in X} \frac{L}{2}\left\|\mathbf{s}-\mathbf{x}\right\|^2 \le \frac{L}{2}\operatorname{diam}(X)^2 \ ,$$

which is the claimed bound.

(Note that this result can be extended to arbitrary norms, in which case smoothness L is measured w.r.t. that norm, and so is the diameter of X. For smoothness w.r.t. other norms, see e.g. [Nes04, Lemma 1.2.3]).

Applications of Frank-Wolfe

Exercise 66. *Consider the matrix completion problem, that is to find a matrix Y solving*

$$\min_{Y \in X \subseteq \mathbb{R}^{n\times m}} \sum_{(i,j)\in\Omega} (Z_{ij} - Y_{ij})^2$$

where the optimization domain X is the set of matrices in the unit ball of the trace norm (or nuclear norm), which is defined the convex hull of the rank-1 matrices

$$X := conv(\mathcal{A}) \;\; \textit{with} \;\; \mathcal{A} := \left\{ \mathbf{u}\mathbf{v}^\top \;\middle|\; \substack{\mathbf{u}\in\mathbb{R}^n,\ \|\mathbf{u}\|_2=1 \\ \mathbf{v}\in\mathbb{R}^m,\ \|\mathbf{v}\|_2=1} \right\}.$$

Here $\Omega \subseteq [n]\times[m]$ is the set of observed entries from a given data matrix Z (collecting the ratings given by users to items for example).

1. *Derive the* LMO_X *for this set X for a gradient at iterate $Y \in \mathbb{R}^{n\times m}$.*
2. *Derive the* projection *step onto X. How do the* LMO_X *and the projection step compare, in terms of computational cost?*

Solution:

1. Because the set X is a convex combination of rank-1 matrices, LMO_X would give one of the corners of the set and Frank-Wolfe will result in an update of the form $\mathbf{s} = \mathbf{u}\mathbf{v}^\top$, $\|\mathbf{u}\|_2 = 1$, $\|\mathbf{v}\|_2 = 1$ that is a 1-rank update.

   The gradient of the objective function is

$$\frac{\partial F}{\partial Y_{ij}} = \begin{cases} 2(Y_{ij} - Z_{ij}), & (i,j)\in\Omega \\ 0, & \text{otherwise.} \end{cases}$$

   LMO_X is equivalent to maximizing over $\mathbf{u}$, $\mathbf{v}$ the following objective:

$$2\sum_{(i,j)\in\Omega} u_i v_j (Z_{ij} - Y_{ij}) = 2\mathbf{u}^\top B\mathbf{v},$$

   where the matrix B is

$$B_{ij} = \begin{cases} Z_{ij} - Y_{ij}, & (i,j)\in\Omega, \\ 0, & \text{otherwise.} \end{cases}$$

   Taking the SVD-decomposition of B, we get that

$$\mathbf{u}^\top B\mathbf{v} = \mathbf{u}^\top UDV^\top\mathbf{v},$$

   which is a convex combination of diagonal elements of D (singular values σ_i). Hence the largest possible value is achieved by taking singular vectors corresponding to the largest singular value: $\mathbf{u} = \mathbf{u}_1$, $\mathbf{v} = \mathbf{v}_1$, then $\mathbf{u}^\top UDV^\top\mathbf{v} = \sigma_1$.

   LMO_X gives a rank-1 matrix $\mathbf{u}\mathbf{v}^\top$ with $\mathbf{u} = \mathbf{u}_1$, $\mathbf{v} = \mathbf{v}_1$ are singular vectors of B corresponding to its largest singular value.

2. By definition of the projection,

$$\begin{aligned}\Pi_X(S) &= \underset{C\in X}{\mathrm{argmin}}\, \|C - S\|_F^2 = \underset{Tr(C)=1}{\mathrm{argmin}}\, \|C - S\|_F^2 = \underset{\sum_i d'_{ii}=1}{\mathrm{argmin}}\, \|U'D'V'^\top - UDV^\top\|_F^2 = \\ &= \underset{\sum_i d'_{ii}=1}{\mathrm{argmin}}\, \|U^\top U'D'V'^\top V - D\|_F^2,\end{aligned}$$

   because U, V are orthogonal matrices.

   If $U' \neq U$ or $V' \neq V$, then the solution for $\mathrm{argmin}_{\sum_i d'_{ii}=1} \|U^\top U'D'V'^\top V - D\|_F^2$ is worse to the solution in case then $U' = U$ and $V' = V$.

   This is because if $U' = U$ and $V' = V$ then $\Pi_X(S) = \mathrm{argmin}_{\sum_i d'_{ii}=1} \|D' - D\|_F^2$.

   But if $U' \neq U$ or $V' \neq V$ then if we denote by F the matrix $U^\top U'D'V'^\top V$ which minimizes expression, then

$$\Pi_X(S) = \|F - D\|_F^2 = \sum_i (F_{ii} - D_{ii})^2 + \sum_{j\neq i} (F_{ij} - D_{ij})^2 \geq \underset{\sum_i d'_{ii}=1}{\mathrm{argmin}}\, \|D' - D\|_F^2,$$

because the second term is always greater than zero.

Then,

$$\Pi_X(S) = \underset{\sum_i d'_{ii}=1}{\operatorname{argmin}} \|D' - D\|_F^2.$$

This is a projection of diagonal elements of D to the unit l_1 ball. We already know from Section 3.5 of lecture notes that this is equal to

$$d'_{ii} = \begin{cases} d_{ii} - \theta_p, & i < p \\ 0 & \text{otherwise} \end{cases},$$

where $\theta_p = \frac{1}{p}\left(\sum_{i=1}^{p} d_{ii} - 1\right)$ $p = \max\{p' \in \{1, \ldots, d\} : d_{pp} - \theta_p > 0\}$ (assuming that all d_{ii} are sorted in decedent order).

3. For a projection step we need to compute the full SVD-decomposition, which takes $\mathcal{O}(mn^2)$, for LMO_X we need only top 1 singular vectors, which is much faster.

References

[Nes04] Yurii Nesterov. *Introductory Lectures on Convex Optimization*. A Basic Course. Kluwer Academic Publishers, 2004.