Labs
Optimization for Machine Learning
Spring 2020

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

Problem Set 8 — Solutions (Frank-Wolfe)

Convergence of Frank-Wolfe

Exercise 1:

Assuming $h_0 \leq 2C$, and the sequence $h_0, h_1, \dots$ satisfies

$$h_{t+1} \leq (1-\gamma)h_t + \gamma^2 C \qquad t = 0, 1, \dots$$

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

$$h_t \leq \frac{4C}{t+2} \qquad t = 0, 1, \dots$$

Solution: By induction. Considering $t \geq 1$, we have

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

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

$$\begin{aligned} h_{t+1} \leq\ & \tfrac{4C}{t+2}\left(1 - \tfrac{1}{t+2}\right) \ = \ \tfrac{4C}{t+2}\tfrac{t+2-1}{t+2} \\ \leq\ & \tfrac{4C}{t+2}\tfrac{t+2}{t+3} \ = \ \tfrac{4C}{t+3} \ , \end{aligned}$$

which is our claimed bound for $t \geq 1$.

Applications of Frank-Wolfe

Exercise 2:

Derive the LMO formulation for matrix completion, that is

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

when $\Omega \subseteq [n] \times [m]$ is the set of observed entries from a given matrix Z.

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

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

1. Derive the LMO for this set X for a gradient at iterate $Y \in \mathbb{R}^{n \times m}$.
2. Dervie the *projection* step onto X. How does the computational operations (or costs) needed to compute the LMO and the projection step compare?

Solution:

1. Because the set X is a convex combination of rank-1 matrices, LMO 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 is equivalent to maximizing over $\mathbf{u}$, $\mathbf{v}$ the following

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

   where the matrix B is

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

   Taking SVD-decomposition of B, we get that

$$\mathbf{u}^\top B \mathbf{v} = \mathbf{u}^\top U D V^\top \mathbf{v},$$

   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 $\mathbf{u}^\top U D V^\top \mathbf{v} = \sigma_1$.

   LMO 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 projection,

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

   because U, V are orthogonal matrices.

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

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

   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

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

   because the second term is always greater than zero.

   Then,

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

   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 $O(mn^2)$, for LMO we need only top 1 singular vectors, which is much faster.