Note: Block Frank Wolfe On Optimal Transport

Pengyu Kan

Oct 26th, 2019

1 Idea of Semi Relaxed Primal:

From the paper ”Smooth and Sparse Optimal Transport” (Blondel et al., 2018), we know the fact that a **Relaxed Primal** actually corresponds to a **Strong Dual**. Therefore, let’s relax the condition of **Optimal Transport** problem.
We have the relaxed primal expression of the problem:

$$ROT_{\Phi}(\mathbf{a}, \mathbf{b}) = \min_{T \geq 0} < T, C > + \frac{1}{2}\Phi(T\mathbf{1}_n, \mathbf{a}) + \frac{1}{2}\Phi(T\mathbf{1}_m, \mathbf{b})$$

And its corresponding strong dual is:

$$ROT_{\Phi}(\mathbf{a}, \mathbf{b}) = \max_{\alpha, \beta \in \mathcal{P}(C)} -\frac{1}{2}\Phi^*(-2\alpha, \mathbf{a}) - \frac{1}{2}\Phi^*(-2\beta, \mathbf{b})$$

where, Φ^* is the conjugate convex function of Φ.
Such strong dual is a function with hard penalty from constraint and also with a strong convex function. Actually it will be useful enough if to only relax one of the two constraints for color transfer. And such equation is called **semi-relaxed primal**. The reason for relaxing only one constraint to be useful enough is that ”we would like all the probability mass of the source image to be accounted for but not necessary for the reference image” (Blondel et al., 2018).
Therefore, as we only relax the condition of the source iamge, we can get:

$$\widetilde{ROT_{\Phi}(\mathbf{a}}, \mathbf{b}) := \min_{\substack{T \geq 0 \\ T^{\top}\mathbf{1}_m = \mathbf{b}}} < T, C > + \Phi(T\mathbf{1}_n, \mathbf{a})$$

The dual of the semi relaxed primal is:

$$\widetilde{ROT_{\Phi}(\mathbf{a}}, \mathbf{b}) = \max_{\alpha, \beta \in \mathcal{P}(C)} -\Phi^*(-\alpha, \mathbf{a}) + \beta^{\top}\mathbf{b}$$

2 Idea of Applying Block Frank-Wolfe Algorithm

For the constraint $T^{\top}\mathbf{1}_m = \mathbf{b}$ in the semi-relaxed primal, we can rewrite it in the following way:

$$\sum_{i=1}^{m} T_{ij} = b_j \quad \text{for } j = 1, 2, \ldots, n$$

$$\iff \sum_{i=1}^{m} |T_{ij}| = b_j \quad \text{for } j = 1, 2, \ldots, n$$

The reason for this step is $T_{ij} \geq 0$.
Then, let's denote t_j as the $j-th$ column of the matrix T, so each $t_j \in \mathbb{R}^m$.
Thus, such constraint is equivalent with:

$$||t_j||_1 = b_j$$

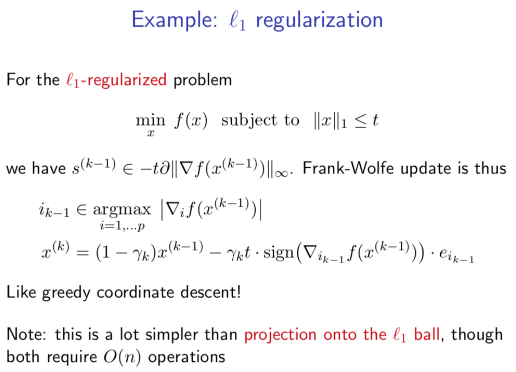
Now, the primal becomes a optimization problem with L1 norm constraints.

$$\widetilde{ROT_\Phi}(\mathbf{a}, \mathbf{b}) := \min_{\substack{T \geq 0 \\ ||t_1||_1 = b_1 \\ \vdots \\ ||t_n||_1 = b_n}} < T, C > + \Phi(T\mathbf{1}_n, \mathbf{a})$$

In order to solve such $L1$ norm constraint optimization problem, instead of applying projection with SGD method, we can directly apply Frank Wolfe method.
Even further, the constraint domain has Cartesian product structure, which can be expressed as $b_1\triangle^m \times b_2\triangle^m \cdots \times b_n\triangle^m$. Therefore, based on the idea of block Frank-Wolfe Algorithm (Lacoste-Julien et al., 2012), we can randomly select a dimension $j \in \{1, \ldots, n\}$ to update the variable, with applying Frank-Wolfe Algorithm in the dimension j.

3 Derivation for Formula of Block Frank-Wolfe Algorithm with $L1$-norm Constraint:

Here is one resource for the derivation.



Example: ℓ_1 regularization

For the ℓ_1-regularized problem

$$\min_x f(x) \text{ subject to } \|x\|_1 \leq t$$

we have $s^{(k-1)} \in -t\partial\|\nabla f(x^{(k-1)})\|_\infty$. Frank-Wolfe update is thus

$$i_{k-1} \in \underset{i=1,\ldots p}{\operatorname{argmax}} \left|\nabla_i f(x^{(k-1)})\right|$$
$$x^{(k)} = (1-\gamma_k)x^{(k-1)} - \gamma_k t \cdot \operatorname{sign}\left(\nabla_{i_{k-1}} f(x^{(k-1)})\right) \cdot e_{i_{k-1}}$$

Like greedy coordinate descent!

Note: this is a lot simpler than projection onto the ℓ_1 ball, though both require $O(n)$ operations

Figure 1: $L1$ norm Frank Wolfe

Based on the idea of **Block Frank Wolfe Algorithm**, it is only needed to consider the constraint of $||t_j||_1 = b_j$, for $\forall j = 1, 2, \ldots, n$ and the constraint of $T \geq 0$ together.
Also, the extreme points of the feasible set must be one of the optimal solutions. Thus, it will be sufficient to check through all the extreme points of the feasible set to find an optimal solution.
For $\forall j = 1, 2 \ldots, n$, the extreme points of the feasible set $||t_j||_1 = b_j$ and $T \geq 0$ are only the points $\{w_k \in \mathbb{R}^m\}_{1 \leq k \leq m}$, s.t.:

$$\begin{cases} w_{k(i)} = b_j, & \text{For } i = k \\ w_{k(i)} = 0, & \text{otherwise} \end{cases}$$

Then, it is needed to find the w_i, for $i \in \{1, \dots, m\}$, s.t.

$$\begin{aligned} w_i &= \arg\min_{w_s} < w_s, \nabla_{(:,j)} f(T) > \quad \text{for } s \in \{1, \dots, m\} \\ &= \arg\min_{w_s} b_j \cdot \nabla_{(s,j)}(f(T)) \quad \text{for } s = 1, 2, \dots, m \\ &= \arg\min_{w_s} \nabla_{(s,j)}(f(T)) \quad \text{for } s = 1, 2, \dots, m \end{aligned}$$

(The reason for the last step is due to the fact that b_j is fixed and it is independent from the choice of w_s.)

The final selection of vector w_i will be used to linearly combined with the $j - th$ column t_j of current transfer matrix T, which is the only update for transfer matrix T based on the **Block Frank Wolfe Algorithm**.
In summary, the algorithm will be following:

BlockFrankWolfeForOT Algorithm:
Input:

- T is the current value of the transfer matrix, and $T \in \mathbb{R}^{m \times n}$
- $grad$ is the gradient value based on the current vlaue of transform matrix, and $grad \in \mathbb{R}^{m \times n}$.
- a is the mass vector of source image, and $a \in \mathbb{R}^m$. And a is the relaxed constraint.
- b is the mass vector of the new target image, and $b \in \mathbb{R}^n$. Also, b is the hard penalty constraint.
- γ is the linear combination coefficient or step size

Output: Current iteration update for the transfer matrix T will be returned.

```
Procedure BlockFrankWolfeForOT ($T, grad, a, b, \gamma$)
    Pick $j$ in random in {1,..., n}
    Find $i = \arg\min_{i \in \{1,\dots,m\}} grad_{(i,j)}$
    Declare $w = \mathbf{0}^m$
    $w_{(j)} = b_j$
    Update $T_{(j)} = (1-\gamma) \cdot T_{(j)} + \gamma w$
    Return $T$
```

4 References:

Mathieu Blondel, Vivien Seguy, Antoine Rolet. Smooth and Sparse Optimal Transport. *AISTATS*, 2018.

Simon Lacoste-Julien, Martin Jaggi, Mark Schmidt, Patrick Pletscher. Block-Coordinate Frank-Wolfe Optimization for Structural SVMs. *30th International Conference on Ma- chine Learning*, 2013.

Ryan Tibshirani. Conditional Gradient (Frank-Wolfe) Method.