10-725/36-725: Convex Optimization **Fall 2019**

Lecture 22: November 13

Lecturer: Ryan Tibshirani *Scribes: Yuan Dong, Yihui He, Rui Yan*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

22.1 Recap: ADMM

For problem

$$\min_{x,z} f(x) + g(z) \quad \text{subject to } Ax + Bz = c$$

we form the **augmented Lagrangian** (scaled form):

$$L_\rho(x, z, w) = f(x) + g(z) + \frac{\rho}{2}\|Ax + Bz - c + w\|_2^2 - \frac{\rho}{2}\|w\|_2^2$$

The alternating direction method of multipliers or **ADMM** has the following update steps:

$$\begin{aligned} x^{(k)} &= \operatorname*{argmin}_x L_\rho(x, z^{(k-1)}, w^{(k-1)}) \\ z^{(k)} &= \operatorname*{argmin}_z L_\rho(x, z^k, w^{(k-1)}) \\ w^{(k)} &= w^{(k-1)} + Ax^{(k)} + Bz^{(k)} - c \end{aligned}$$

ADMM converges like a first-order method and is a very flexible framework. It can be used in simple problems (e.g. LASSO) or more difficult problems (e.g. SDPs).

22.2 Projected Gradient Descent

Consider a constrained problem which constrain the solution in the convex set C.

$$\min_x f(x) \quad \text{subject to } \ x \in C$$

where f is convex and smooth. Recall that **projected gradient descent** chooses an initial $x^{(0)}$ and update by

$$x^{(k)} = P_C(x^{(k-1)} - t_k \nabla f(x^{(k-1)}))$$

for $k = 1, 2, 3, \ldots$. Here P_C is the projection operator onto the set C.

One special case of proximal gradient, motivated by local quadratic expansion of f is that,

$$x^{(k)} = P_C\Big(\underset{y}{\text{argmin}} \nabla f(x^{(k-1)})^T(y - x^{(k-1)}) + \frac{1}{2t}\|y - x^{(k-1)}\|_2^2\Big)$$

One motivation for exploring Frank-Wolfe is that in projections are not always easy. For example, if the constraint set is a polyhedron, $C = \{x : Ax \leq b\}$, the projection is generally very hard.

22.3 Frank-Wolfe Method

The **Frank-Wolfe method** is also called conditional gradient method, that uses a local linear expansion of f, instead of using a quadratic expansion as in projected GD methods.

$$\begin{aligned} s^{(k-1)} &\in \underset{s \in C}{\text{argmin}} \nabla f(x^{(k-1)})^T s \\ x^{(k)} &= (1 - \gamma_k)x^{(k-1)} + \gamma_k s^{(k-1)} \end{aligned}$$

We take a convex combination of the new point $s^{(k-1)}$ and $x^{(k-1)}$, so we can remain in the convex set C without going beyond the set boundary. Therefore there is **no projections** involved and the update is solved directly on C, as shown in Figure 22.1.

The default step size is

$$\gamma_k = \frac{2}{k+1}$$

for $k = 1, 2, 3, \ldots$. Note that for any $0 \leq \gamma_k \leq 1$, we have $x^{(k)} \in C$ by convexity. The update can be written as:

$$x^{(k)} = x^{(k-1)} + \gamma_k(s^{k-1} - x^{(k-1)})$$

where $(s^{k-1} - x^{(k-1)})$ is the direction we are going to, and γ_k is the step size. Since γ_k is decreasing, we are moving less and less in the direction of the linearization minimizer as the algorithm proceeds.

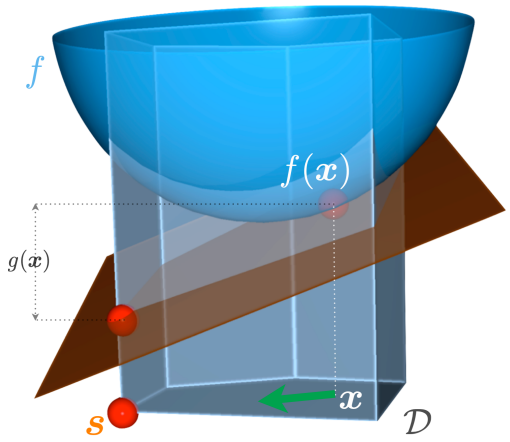
22.4 Norm Constraints

Let's see an example, when the $C = \{x : \|x\| \leq t\}$ for a norm $\|\cdot\|$. By the definition of the problem we have

$$s \in \underset{\|s\| \leq t}{\text{argmin}} \nabla f(x^{(k-1)})^T s$$

Since

$$\begin{aligned} \min_{\|s\| \leq t} \nabla f(x)^T s &= -\max_{\|s\| \leq t} -\nabla f(x)^T s \\ &= -t \cdot \max_{\|z\| \leq 1} -\nabla f(x)^T z \\ &= -t \cdot \max_{\|z\| \leq 1} \nabla f(x)^T z \end{aligned}$$



(From Jaggi 2011)

Figure 22.1: Illustration from Jaggi (2011). $f(x)$ is a differentiable convex function, and D is the constraint set. The brown plane is the linear approximation of the function at x, and s is the point that minimizes the approximation constrained by D. The upate is a convex combination of point x and s.

Therefore

$$\underset{\|s\|\leq t}{\text{argmin}} \nabla f(x)^T s = -t \underset{\|s\|\leq 1}{\text{argmax}} \nabla f(x)^T s$$

So the dual norm can be written as

$$\|z\|_* = \max_{\|z\|\leq 1} z^T x$$

Therefore

$$\begin{aligned} s &\in \underset{\|s\|\leq t}{\text{argmin}} \nabla f(x^{(k-1)})^T s \\ &= -t \cdot \left(\underset{\|s\|\leq 1}{\text{argmax}} \nabla f(x^{(k-1)})^T s \right) \\ &= -t \cdot \partial \|\nabla f(x^{(k-1)})\|_* \end{aligned}$$

where $\|\cdot\|_*$ denotes the corresponding dual norm. That is, if we know how to compute the **subgradients of the dual norm**, then we can easily perform Frank-Wolfe steps. With the closed form update, Frank-Wolfe is simpler or cheaper then taking projection onto $C = \{x : \|x\| \leq t\}$.

The following sections are related to some examples of norm based constraints to see how to perform Frank-Wolfe in these special cases.

22.5 Example: Trace Norm Regularization

Consider the trace-regularized problem

$$\min_X f(X) \quad \text{subject to} \quad \|X\|_{\text{tr}} \leq t$$

Applying Frank-Wolfes algorithm, noticing that the dual of trace norm is operator norm, so we will get

$$S^{(k-1)} \in -t\partial \left\| \nabla f\left(X^{(k-1)}\right) \right\|_{\text{op}}$$

Notice that the operator norm is the maximum singular value, so when we denote u and v being the leading left and right singular vectors of $\nabla f\left(X^{(k-1)}\right)$, we can get

$$S^{(k-1)} = -t \cdot uv^T$$

This make Frank-Wolfe updates much cheaper than the projection onto the trace norm ball, which need at least a truncated SVD: keep finding singular values until finding a value smaller than t, the radius of the norm ball, thus much more steps than only getting leading singular vectors.

22.6 Note: Constrained and Lagrange forms

Notice that the constrained form

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

is equal to the Lagrange form

$$\min_x f(x) + \lambda \|x\|$$

as long as we let the tuning parameters t and λ vary over $[0, \infty)$. And there is also no strong preference over either constrained form or Lagrangian form, and we will just solve whichever form is easier to solve *, and choose best via something like a cross validation (CV).

In the previous chapters, we just show the superiority of Frank-Wolfe over constrained form on some specific problems. So that's not enough, and we should also show that Frank-Wolfe is also superior over corresponding Laplacian form of the problems.

- ℓ_1 norm: Frank-Wolfe update scans for maximum of gradient; proximal operator soft-thresholds the gradient step; both use O(n) ops.
- ℓ_p norm: Frank-Wolfe update computes raises each entry of gradient to power and sums, in O(n) ops; proximal operator not generally directly computable.
- Trace norm: Frank-Wolfe update computes top left and right singular vectors of gradient; proximal operator soft-thresholds the gradient step, requiring a singular value decomposition.

Q.E.D.

* However, the two forms are not exactly equivalent. Solving for the best t or λ actually results in different estimators and don't have the same operator characteristics. Equivalency between t and λ is instead data dependent.

Various other constraints yield efficient Frank-Wolfe updates, e.g., special polyhedra or cone constraints, sum-of-norms (group-based) regularization, atomic norms, etc.

22.7 An example where Frank-Wolfe isn't superior

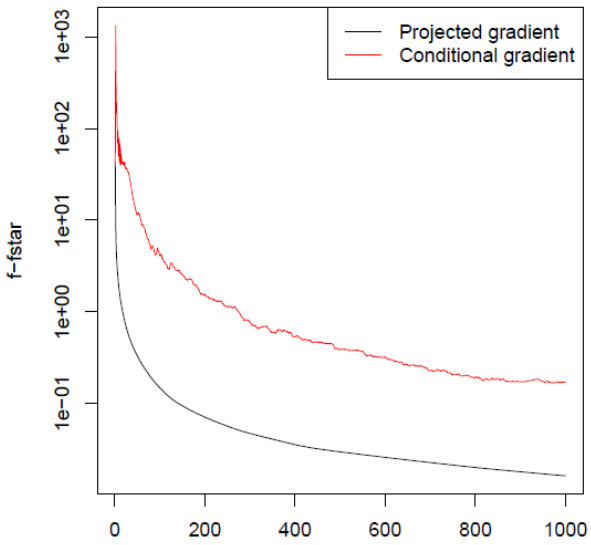


Figure 22.2: Comparison of conditional v.s. projected gradient over constrained lasso problem

Comparing projected and conditional gradient for the constrained lasso problem, with n=100, p=500: Frank-Wolfe converges slower than projected gradient *. Note that both projected gradient and Frank-Wolfe are both O(n), so there is no reason to choose FW in this case.

Also, this graph told us that Frank-Wolfe is not a descent estimator as the objective is not monotonically decreasing over each k.

* Frank-Wolfe in this problem uses standard step sizes, and a different step size method such as line search would probably help in terms of convergence.

22.8 Duality Gap

Frank-Wolfe iterations admit a very natural duality gap:

$$g\left(x^{(k)}\right) = \nabla f\left(x^{(k)}\right)^T \left(x^{(k)} - s^{(k)}\right)$$

Claim: it holds that

$$f\left(x^{(k)}\right) - f^\star \leq g\left(x^{(k)}\right)$$

Proof: by the first-order condition for convexity

$$f(s) \geq f\left(x^{(k)}\right) + \nabla f\left(x^{(k)}\right)^T \left(s - x^{(k)}\right)$$

Minimizing both sides over all $s \in C$ yields

$$\begin{aligned} f^\star &\geq f\left(x^{(k)}\right) + \min_{s \in C} \nabla f\left(x^{(k)}\right)^T \left(s - x^{(k)}\right) \\ &= f\left(x^{(k)}\right) + \nabla f\left(x^{(k)}\right)^T \left(s^{(k)} - x^{(k)}\right) \end{aligned}$$

Which can then be re-written as

$$\begin{aligned} f^* &\geq f\left(x^{(k)}\right) + \nabla f\left(x^{(k)}\right)^T \left(s - x^{(k)}\right) \\ -\nabla f\left(x^{(k)}\right)^T \left(s^{(k)} - x^{(k)}\right) &\geq f\left(x^{(k)}\right) - f^* \\ \nabla f\left(x^{(k)}\right)^T \left(x^{(k)} - s^{(k)}\right) &\geq f\left(x^{(k)}\right) - f^* \end{aligned}$$

Q.E.D.

Why do we call it a "duality gap"?

If we rewrite original problem as

$$\min_x f(x) + I_C(x)$$

and the dual problem will be

$$\max_u -f^*(u) - I_C^*(-u)$$

where I_C^* is the support function of C. Duality gap at $x; u$ is

$$f(x) + f^*(u) + I_C^*(-u) \geq x^T u + I_C^*(-u)$$

Evaluating this at

$$x = x^{(k)}, u = \nabla f\left(x^{(k)}\right)$$

and we get

$$\nabla f\left(x^{(k)}\right)^T x^{(k)} + \max_{s \in C} -\nabla f\left(x^{(k)}\right)^T s = \nabla f\left(x^{(k)}\right)^T \left(x^{(k)} - s^{(k)}\right)$$

which is exactly our gap.

22.9 Convergence Analysis

Following Jaggi [2], define the curvature constant of f over C:

$$M = \max_{\substack{\gamma \in [0,1], x, s, y \in C \\ y = (1-\gamma)x + \gamma s}} \frac{2}{\gamma^2}\left(f(y) - f(x) - \nabla f(x)^T (y - x)\right)$$

Note that $M = 0$ for linear f, and $f(y) - f(x) - \nabla f(x)^T(y - x)$ is called the Bregman divergence, defined by f

Theorem 22.1 *The Frank-Wolfe method using standard step sizes $\gamma_k = \frac{2}{k+1}, k = 1, 2, 3 \ldots$, satisfies*

$$f\left(x^{(k)}\right) - f^* \leq \frac{2M}{k+2}$$

Thus number of iterations needed for $f\left(x^{(k)}\right) - f^\star \leq \epsilon$ is $O(1/\epsilon)$. This matches the sublinear rate for projected gradient descent fo Lipschitz ∇f with constant L, recall,

$$f(y) - f(x) - \nabla f(x)^T (y - x) \leq \frac{L}{2}\|y - x\|_2^2$$

Maximizing over all $y = (1 - \gamma)x + \gamma s$, and multiplying by $2/\gamma^2$,

$$\begin{aligned} M &\leq \max_{\substack{\gamma \in [0,1], x, s, y \in C \\ y = (1-\gamma) \in C}} \frac{2}{\gamma^2} \cdot \frac{L}{2}\|y - x\|_2^2 \\ &\quad y = (1 - \gamma)x + \gamma s \\ &= \max_{x, s \in C} L\|x - s\|_2^2 = L \cdot \text{diam}^2(C) \end{aligned}$$

Where $diam^2(C)$ is the squared diameter of the set C. So, if f has a gradient that is Lipschitz, and C is compact, then it immediately has a curvature that is finite and that is at most $L \cdot diam^2(C)$ Hence assuming a bounded curvature is basically no stronger than what we assumed for projected gradient.

22.10 Basic Inqeuality

The key inequality used to prove the Frank-Wolfe convergence rate:

$$f\left(x^{(k)}\right) \leq f\left(x^{(k-1)}\right) - \gamma_k g\left(x^{(k-1)}\right) + \frac{\gamma_k^2}{2} M$$

Here $g(x) = \max_{s \in C} \nabla f(x)^T (x - s)$ is duality gap defined earlier

Proof: write $x^+ = x^{(k)}, x = x^{(k-1)}, s = s^{(k-1)}, \gamma = \gamma_k$. Then

$$\begin{aligned} f\left(x^+\right) &= f(x + \gamma(s - x)) \\ &\leq f(x) + \gamma \nabla f(x)^T (s - x) + \frac{\gamma^2}{2} M \\ &= f(x) - \gamma g(x) + \frac{\gamma^2}{2} M \end{aligned}$$

Second line used definition of M, and third line the definition of g. The proof of the convergence result is now straightforward. Denote by $h(x) = f(x) - f^\star$ the suboptimality gap at x. Basic inequality:

$$\begin{aligned} h\left(x^{(k)}\right) &\leq h\left(x^{(k-1)}\right) - \gamma_k g\left(x^{(k-1)}\right) + \frac{\gamma_k^2}{2} M \\ &\leq h\left(x^{(k-1)}\right) - \gamma_k h\left(x^{(k-1)}\right) + \frac{\gamma_k^2}{2} M \\ &= (1 - \gamma_k)\, h\left(x^{(k-1)}\right) + \frac{\gamma_k^2}{2} M \end{aligned}$$

where in the second line we used $g\left(x^{(k-1)}\right) \geq h\left(x^{(k-1)}\right)$. To get the desired result we use induction:

$$h\left(x^{(k)}\right) \leq \left(1 - \frac{2}{k+1}\right) \frac{2M}{k+1} + \left(\frac{2}{k+1}\right)^2 \frac{M}{2} \leq \frac{2M}{k+2}$$

22.11 Affine Invariance

Frank-Wolfe updates are affine invariant: for nonsingular matrix A, define $x = Ax', F(x') = f(Ax')$, consider Frank-WOlfe on F:

$$\begin{aligned} s' &= \operatorname*{argmin}_{z \in A^{-1} C} \nabla F\left(x'\right)^T z \\ \left(x'\right)^+ &= (1 - \gamma) x' + \gamma s' \end{aligned}$$

Multiplying by A produces same Frank-Wolfe update as that from f. Convergence analysis is also affine invariant: curvature constant

$$M = \max_{\substack{\gamma \in [0,1], x, s, y \in C \\ y' = (1-\gamma)x' + \gamma s'}} \frac{2}{\gamma^2} \left(F\left(y'\right) - F\left(x'\right) - \nabla F\left(x'\right)^T \left(y' - x'\right)\right)$$

matches that of f, because $\nabla F\left(x'\right)^T \left(y' - x'\right) = \nabla f(x)^T (y - x)$

22.12 Inexact Updates

Jaggi [2] also analzes inexact Frank-Wolfe updates suppose we choose $s^{(k-1)}$ so that

$$\nabla f\left(x^{(k-1)}\right)^T s^{(k-1)} \leq \min_{s \in C} \nabla f\left(x^{(k-1)}\right)^T s + \frac{M\gamma_k}{2} \cdot \delta$$

where $\delta \geq 0$ is an inaccuracy parameter. Then we attain the same rate.

Theorem 22.2 *Theorem: Frank-Wolfe using step sizes $\gamma_k = 2/(k+1), k = 1, 2, 3, \ldots$ and inaccuracy parameter $\delta \geq 0$, satisfies*

$$f\left(x^{(k)}\right) - f^\star \leq \frac{2M}{k+1}(1+\delta)$$

Note: the optimization error at step k is $M_{\gamma_k}/2 \cdot \delta$. Since $\gamma k \to 0$, we require the errors to vanish.

22.13 Two variants

There are two important variants of Frank-Wolfe method.

1. **Line search**. instead of using standard step sizes, use

   $$\gamma_k = \underset{\gamma \in [0,1]}{\operatorname{argmin}} f\left(x^{(k-1)} + \gamma\left(s^{(k-1)} - x^{(k-1)}\right)\right)$$

   at each $k = 1, 2, 3, \ldots$ Or, we could use backtracking

2. **Fully corrective**: directly update according to

   $$x^{(k)} = \underset{y}{\operatorname{argmin}} f(y) \quad \text{subject to} \quad y \in \operatorname{conv}\left\{x^{(0)}, s^{(0)}, \ldots, s^{(k-1)}\right\}$$

   Both variants lead to the same $O(1/\epsilon)$ iteration complexity Another popular variant: away steps, which get linear convergence under strong convexity

22.14 Path Following

Given the norm constrained problem

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

Frank-Wolfe can be used for path following, i.e., we can produce an approximate solution path $\hat{x}(t)$ that is ϵ-suboptimal for every $t \geq 0$. Let $t_0 = 0$ and $x^\star(0) = 0$, fix $m > 0$, repeat for $k = 1, 2, 3, \ldots$:

1. Calculate

   $$t_k = t_{k-1} + \frac{(1 - 1/m)\epsilon}{\|\nabla f\left(\hat{x}\left(t_{k-1}\right)\right)\|_*}$$

   and set $\hat{x}(t) = \hat{x}\left(t_{k-1}\right)$ for all $t \in (t_{k-1}, t_k)$

2. Compute $\hat{x}(t_k)$ by running Frank-Wolfe at $t = t_k$, terminating when the duality gap is $\leq \epsilon/m$

(This is a simplification of the strategy from Giesen et al. [1])

Claim: this produces (piecewise-constant) path with

$$f(\hat{x}(t)) - f(x^\star(t)) \leq \epsilon \quad \text{for all } t \geq 0$$

Proof: rewrite the Frank-Wolfe duality gap as

$$g_t(x) = \max_{\|s\| \leq t} \nabla f(x)^T (x - s) = \nabla f(x)^T x + t\|\nabla f(x)\|_*$$

This is a linear function of t. Hence if $g_t(x) \leq \epsilon/m$, , then we can increase t until $t^+ = t + (1 - 1/m)\epsilon/\|\nabla f(x)\|_*$, because

$$g_{t^+}(x) = \nabla f(x)^T x + t\|\nabla f(x)\|_* + \epsilon - \epsilon/m \leq \epsilon$$

i.e., the duality gap remains $\leq \epsilon$ for the same x, between t and t^+

References

[1] Joachim Giesen, Martin Jaggi, and Sören Laue. Approximating parameterized convex optimization problems. *ACM Transactions on Algorithms (TALG)*, 9(1):10, 2012.

[2] Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 427–435, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL `http://proceedings.mlr.press/v28/jaggi13.html`.