1 Frank-Wolfe algorithm

1.1 Introduction

In this lecture, we consider the minimization problem

$$\min_{w \in B} g(w)$$

under the following assumptions:

- $g$ is convex and differentiable.
- $B \subset \mathbb{R}^d$ is a convex set.
- The linear optimization problem $\max_{s \in B} \langle c, w \rangle$ is "easy".

A few applications in which this condition is reasonable:

1. The minimum norm problem $\min_{w \in B} \frac{1}{2}||w||^2$

2. Minimize $g(w)$ under the set $B = \text{conv}(A)$, or the bounded atomic norm $||w||_A \leq R$ (defined in Lecture 13).

We present here the Frank-Wolfe algorithm that solves the given optimization, which is also called the conditional gradient method.

1.2 The algorithm

Frank-Wolfe algorithm
Start with $w^0 \in B$. For $t = 1, 2, ..T$

- Compute $s^t \in \arg \min_{s \in B} \langle s, \nabla g(w^t) \rangle$
- Set $w^{t+1} = (1 - \eta_t)w^t + \eta_t s^t$
Intuitively, at each step, one chooses a vertex \( s^t \) that minimizes the current gradient, then walk toward \( s^t \) with step size \( \eta_t \). In case of convex polytope, the minimizers are the vertices of the polytope.

How to choose \( \eta_t \)? There are a few possibilities, all of which achieves a convergence rate of \( O(1/t) \):

1. Set \( \eta_t = \frac{2}{t+2} \).
2. Another possibility is to pick the minimizing point on the segment \((w_t, s_t)\). Specifically:
   \[
   \eta_t = \arg \min_{\eta \in [0,1]} g((1 - \eta)w_t + \eta s_t)
   \]
3. We could also consider the complete history. Note that \( w_{t+1} \) is a convex combination of \( w^0, s^1, \ldots, s^t \), so we could try to solve the optimization problem:
   \[
   w_{t+1} = \arg \min_{\tilde{\eta}_0, \ldots, \tilde{\eta}_t} g(\tilde{\eta}_1 s^1 + \ldots + \tilde{\eta}_t s^t + \tilde{\eta}_0 w^0)
   \]

The latter step choices will make each optimization more aggressive; however, it also increases the time to find the minimizer.

So what is the advantages and disadvantages of the Frank-Wolfe algorithm? On one hand, this algorithm has very slow convergence rate (at the order of \( O(1/t) \)). On the other
hand, this algorithm is very simple, using only linear optimization and itertes through at most $t$ vertices after $t$ steps. This helps maintain the sparsity of the candidate solution. In certain applications (such as low-rank approximation), we typically start with a low rank candidate and increase the rank by at most 1 in each step. This contrasts with the projection algorithm, which one starts with a high rank solution and projects onto a low rank space.

**Example 1:** consider the sparse approximation problem
\[
\min_w \frac{1}{2} \| Dw - y \|^2 \text{ such that } \|w\|_1 \leq 1.
\]
This is equivalent to optimizing over the
\[
A = \text{conv}\{\pm e_i\}.
\]
We start with $w^0 = e_1$. In each step, we compute
\[
s^t \in \arg \min_{\pm e_i} \langle \nabla g(w^t), \pm e_i \rangle
\]
where $\nabla g(w) = D^T(Dw - y)$. Therefore
\[
w^{t+1} = w^t(1 - \eta_t) + \eta_t e_j
\]
for some vertex $e_j$.

**Example 2:** We want to minimize $g(W)$ such that $tr(W) \leq 1$. That is, the atomic norm $\|W\|_A \leq 1$ where $A = \{uv^T, \|u\| \leq 1, \|v\| \leq 1\}$. To apply the Frank-Wolfe algorithm, one needs to solve the linear optimizer $\max_S \langle S, M \rangle$. The standard method requires computing the full SVD of $M$ which takes $O(mn^2)$. However, given that the trace norm is the convex hull of rank-1 matrices, we can write
\[
\min_S \langle S, W \rangle = \min_{\|u\| \leq 1, \|v\| \leq 1} u^T W u = \sigma_1
\]
implying it suffices to approximate the maximum eigenvalue of $M$. Using Lanczos’ algorithm, for example, one can approximate with in error $\epsilon$ in $O(\frac{N_g}{\sqrt{\epsilon}})$ where $N_g$ is the number of non-zero entries in $\nabla g(W)$.

Before proceeding, we shall argue that the $O(\frac{1}{t})$ is tight.

**Lemma 1.** ([2], Appendix C, Lemma 3) For $f(x) = \|x\|^2$ and $1 \leq k \leq n$
\[
\min_{x \in \Delta_n, \|x\|_0 \leq k} f(x) = \frac{1}{k}
\]
This is achieved when exactly $k$ of $n$ components of $x$ is $\frac{1}{k}$. If the algorithms starts at a vertex of the $\ell_1$-ball (one nonzero entry in $x^0$, in each iteration it can only add one more nonzero entry, and hence after $k$ steps the solution has at most $k$ nonzero entries. The optimal solution has value $\frac{1}{n}$ (in $\mathbb{R}^n$), which can be arbitrarily small as $n$ grows large.

However, for certain problems, the convergence rate can be improved.
1.3 Primal Convergence

To prove the convergence of the Frank-Wolfe algorithm, we introduce the curvature constant. The curvature constant $C$ of a differentiable function $g$ satisfies, for all $w, w'$ and $\eta \in [0, 1]$

$$g(w + \eta(w' - w)) \leq g(w) + \eta \langle \nabla g(w), w' - w \rangle + \frac{C}{2} \eta^2$$

We can see that, the curvature constant $C$ limits the deviation of the function from the linear approximation by $\nabla g(w)$. The curvature bound is also related to $\nabla g$ via this following lemma:

**Lemma 2.** ([2], Appendix D, Lemma 7) If $\Delta g$ is $L$-Lipschitz continuous with respect to some norm $\| \cdot \|$ over the domain $D$, then

$$C \leq L \cdot \text{diam}_{\| \cdot \|}(D)^2$$

Now we prove the convergence. Consider the $t$-th iteration step

$$g(w^{t+1}) = g((1 - \eta_t)w^t + \eta^t s^t)$$

$$\leq g(w^t) + \eta_t \langle s^t - w^t, \nabla g(w^t) \rangle + \frac{C}{2} \eta_t^2 \text{ (assume curvature constant)}$$

$$\leq g(w^t) + \eta_t \langle w^* - w^t, \nabla g(w^t) \rangle + \frac{C}{2} \eta_t^2 \text{ (since $s^t$ minimizes the linear approximation)}$$

$$= (1 - \eta_t)g(w^t) + \eta_t(g(w^t) + \langle w^* - w^t, \nabla g(w^t) \rangle) + \frac{C}{2} \eta_t^2$$

$$\leq (1 - \eta_t)g(w^t) + \eta_t g(w^*) + \frac{C}{2} \eta_t^2 \text{ (by convexity)}$$

Therefore

$$g(w^{t+1}) - g(w^*) \leq (1 - \eta_t)(g(w^t) - g(w^*)) + \frac{C}{2} \eta_t^2$$

By choosing $\eta_t = \frac{2}{t+2}$ it follows that

$$g(w^t) - g(w^*) \leq \frac{C}{t+2}$$

Note that the Frank-Wolfe algorithm can be applied even if the linear optimization problem $\min_{w \in B} \langle s, \nabla g(w^t) \rangle$ can only be solved approximately. In this case, the bound becomes

$$g(w^t) - g(w^*) \leq \frac{2C(1 + \delta)}{t+2}$$

where $\delta$ is the approximation error.
2 Implication for submodular minimization

Consider the primal optimization problem

\[
\min_x f(x) + \frac{1}{2}||x||^2
\]

and its dual

\[
\max_w -\frac{1}{2}||w||^2 = -\min_w \frac{1}{2}||w||^2
\]

Given a pair of candidate \((x, w)\), define the duality gap to be

\[
gap(w) = f(x) + \frac{1}{2}||x||^2 - (-\frac{1}{2}||w||^2)
\]

**Theorem 1.** (Bach) If \(\gap(w_t) \leq \epsilon\) then there exists a level set \(S = \{x^t \geq \theta\}\) with

\[
F(S) - F(S^*) \leq \sqrt{2n\epsilon}
\]

If we apply the Frank-Wolfe algorithm to the dual problem, the linear optimization step becomes

\[
s^t \in \arg\min_{s \in B_F} \langle \nabla g(w^t), s \rangle = \arg\min_{s \in B_F} \langle w^t, s \rangle
\]

The Frank-Wolfe algorithm gives a bound on dual gap \(g(w^t) - g(w^*)\). How does this bound relate to the primal gap, and the duality gap?

From the duality theorem, \(f(x) + \frac{1}{2}||x||^2 = -\frac{1}{2}||w||^2\) if and only if \(x^* = -w^*\). That suggests setting \(x^t = -w^t\). Consider the duality gap (using \(x = -w\))

\[
f(x) + \frac{1}{2}||x||^2 - (-\frac{1}{2}||w||^2) = f(-w) + \frac{1}{2}||w||^2 + \frac{1}{2}||w||^2
\]

\[
= \max_{s \in B_F} \langle s, -w \rangle + ||w||^2
\]

\[
= \max_{s \in B_F} \langle w - s, \nabla g(w) \rangle
\]

From the assumption of curvature bound

\[
g(w^{t+1}) \leq g(w^t) + \eta_t \langle s^t - w^t, \nabla g(w^t) \rangle + \frac{C}{2}\eta_t^2
\]

\[
g(w^{t+1}) - g(w^*) \leq g(w^t) - g(w^*) + \eta_t \gap(w^t) + \frac{C}{2}\eta_t^2
\]

With \(\eta_t = \frac{2}{t+2}\), it follows \(g(w^t) - g(w^*) \leq \frac{C}{t+2}\) and \(\frac{C}{2}\eta_t^2 = O(\frac{1}{t})\). Therefore \(\gap(w^t)\) must also converge to 0 with rate of at least \(O(\frac{1}{t})\). In fact, we have the following theorem (stated without proof):
Theorem 2. (Jaggi)
After $T$ iterations, there is a $0 \leq t \leq T$ such that the duality gap

$$\text{gap}(w^t) \leq \frac{7C}{T+2}$$

3 Relation between Subgradient method and Conditional gradient descent

Again, we consider the primal problem

$$\min_x f(x) + \frac{1}{2}||x||^2$$

and its dual

$$\max_w -\frac{1}{2}||w||^2$$

given $f(x) = \max_{s \in B}s^Tx$.

Consider the subgradient method on the primal problem. First, rewrite the problem as

$$\min_x \max_{s \in B}s^Tx + \frac{1}{2}||x||^2$$

The $t$-th step of the subgradient method becomes

$$x^{t+1} = x^t - \alpha_t g_t$$

$$g_t = x_t + \arg \max_{s \in B}\langle s, x^t \rangle$$

Set $s^t_p = \arg \max_{s \in B}\langle s, x^t \rangle$, then $x^{t+1} = x^t - \alpha_t(x^t + s^t_p)$.

Now consider the conditional gradient method on the dual problem. We have $\nabla g(w) = w$, and $s^t = \arg \min_{s \in B}\langle s, w \rangle$ and $w^{t+1} = (1-\eta_t)w^t + \eta_t s^t_d$.

Now comes the crucial observation. Set $w^t = -x^t$, then

$$s^t_p = \arg \max_{s \in B}\langle s, x^t \rangle$$

$$= \arg \min_{s \in B}\langle s, -x^t \rangle$$

$$= \arg \min_{s \in B}\langle s, w^t \rangle = s^t_d$$

With that

$$x^{t+1} = -w^{t+1} = -w^t - \alpha_t(-w^t + s^t)$$
Negate the sign of two sides gives

\[ w^{t+1} = w^t + \alpha_t (-w^t + s^t) \]

However, from the dual problem, \( w^{t+1} = w^t + \eta_t (-w^t + s^t) \). Therefore, if we set \( \eta_t = \alpha_t \), then the two algorithms are direct mirror of each other!

## 4 Applications

The Frank-Wolfe algorithm appears in many different contexts. Here are some examples.

### 4.1 Structured SVM

Given \( n \) samples \( x = (x_1, \ldots, x_n) \) and their corresponding labels \( y = (y_1, \ldots, y_n) \). Given a weight vector \( w \), we would like to minimize

\[
\min_{y \in \{-1,+1\}^n} \langle w, \Phi_x(y) \rangle + \frac{\lambda}{2} \| w \|^2
\]

This objective function is a support function (of the convex hull \( \text{conv}\{\Phi_x(y) \mid y \in \{-1,1\}^m\} \)) plus a squared norm. The dual of it can be derived analogously to that of the Lovász extension plus squared norm, and looks similar to the min-norm problem for submodular optimization. Applying the Frank-Wolfe algorithm to the dual is, according to our above reasoning, equivalent to applying a subgradient method to the primal (non-smooth) SVM problem.

The paper [3] shows a Frank-Wolfe method for the structured SVM, and derive a stochastic block coordinate descent method. This can be related to a stochastic gradient method in the primal.

### 4.2 Herding Problem

In the herding problem, we are given a set of samples \( x_1, \ldots, x_n \) and are trying to approximate a given mean (expectation of a feature function or sufficient statistic)

\[
\mu = \mathbb{E}_{p(x)} \Phi(x)
\]

by the average of a few sample points. The original Herding method picks those greedily. This method can be viewed as a Frank-Wolfe method applied to the objective

\[
\min_{w \in \text{conv}\{x_j\}_{j=1}^n} \| w - \mu \|^2.
\]
With an appropriately chosen step size, we get \( w = \frac{1}{t} \sum_{j=1}^{t} \Phi(x_j) \), and hence the difference between the empirical and the population mean

\[
\| \frac{1}{t} \sum_{j=1}^{t} \Phi(x_j) - \mu \|^2
\]

that is being minimized.

The equivalence between Herding and Frank-Wolfe is discussed in [1].

### 4.3 Boosting

Boosting too can be viewed as a Frank-Wolfe method. Details are discussed in [4].

Suppose \( B \) is the convex hull of the set of all hypotheses. We aim to choose a weight function \( w(x) \) that minimizes

\[
\min_{w(x) \in B} \mathbb{E}_{x,y} \text{Loss}(w(x), y).
\]

### References


