Optimization for Machine Learning
CS-439

Lecture 9: Frank-Wolfe & Accelerated Gradient Descent

Martin Jaggi

EPFL – `github.com/epfml/OptML_course`

May 3, 2019

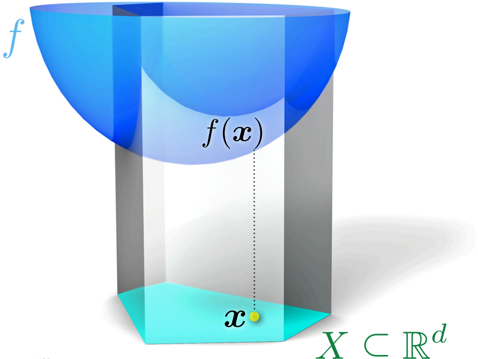
Chapter 9

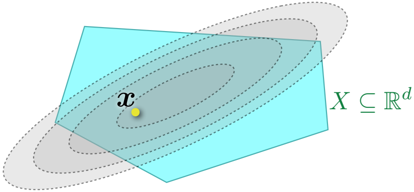
Frank-Wolfe

Constrained Optimization

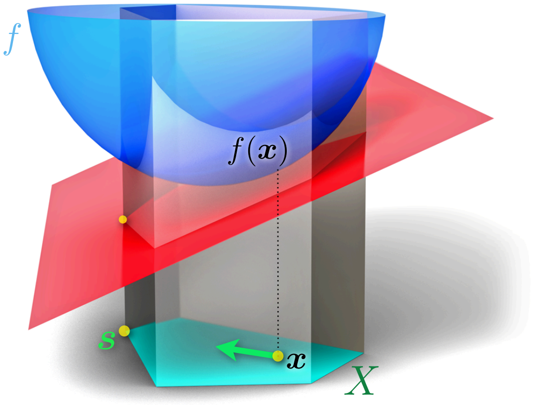
Constrained Optimization Problem

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in X \end{aligned}$$





Frank-Wolfe Algorithm



Frank-Wolfe Algorithm:

$$\begin{aligned}\mathbf{s} &:= \mathrm{LMO}(\nabla f(\mathbf{x}_t)),\\ \mathbf{x}_{t+1} &:= (1-\gamma)\mathbf{x}_t + \gamma\mathbf{s},\end{aligned}$$

for timesteps $t = 0, 1, \ldots,$ and
stepsize $\gamma := \frac{2}{t+2}$.

Linear Minimization Oracle:

$$\mathrm{LMO}(\mathbf{g}) := \operatorname*{argmin}_{\mathbf{s}\in X} \langle \mathbf{s}, \mathbf{g} \rangle$$

Properties

- Aways feasible: $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_t \in X$.
  $\mathbf{x}_{t+1}$ is on line segment $[\mathbf{s}, \mathbf{x}_t]$, for $\gamma \in [0, 1]$.
- Reduces non-linear to linear optimization
- Projection-free
- Sparse iterates (in terms of corners $\mathbf{s}$ used)

Invented and analyzed 1956 by Marguerite Frank and Philip Wolfe.

Example

Lasso Regression

$$\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|^2 \;\; s.t. \;\; \|\mathbf{x}\|_1 \leq 1$$

L1-ball is the convex hull of the unit basis vectors:
$X = \{\mathbf{x} \mid \|\mathbf{x}\|_1 \leq 1\} = \mathrm{conv}(\{\pm\mathbf{e}_1, \ldots, \pm\mathbf{e}_n\})$.

Example

Lasso Regression

$$\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|^2 \;\; s.t. \;\; \|\mathbf{x}\|_1 \leq 1$$

L1-ball is the convex hull of the unit basis vectors:
$X = \{\mathbf{x} \mid \|\mathbf{x}\|_1 \leq 1\} = \mathrm{conv}(\{\pm\mathbf{e}_1, \ldots, \pm\mathbf{e}_n\})$.

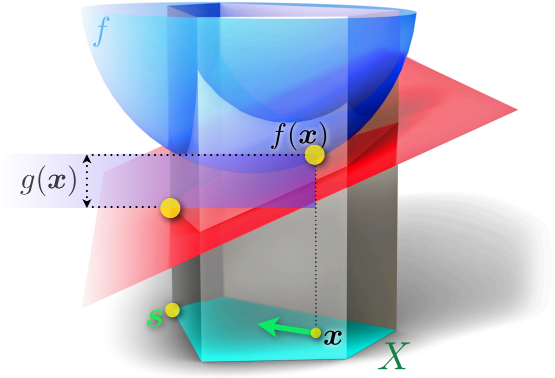
- $\nabla f(\mathbf{x}) = \mathbf{g} := A^\top(A\mathbf{x} - \mathbf{b})$
- $\mathrm{LMO}(\mathbf{g}) = -\mathrm{sign}(g_i)\mathbf{e}_i$ with $i := \operatorname{argmax}_{i \in [n]} |g_i|$

simpler than projection onto L1-ball !

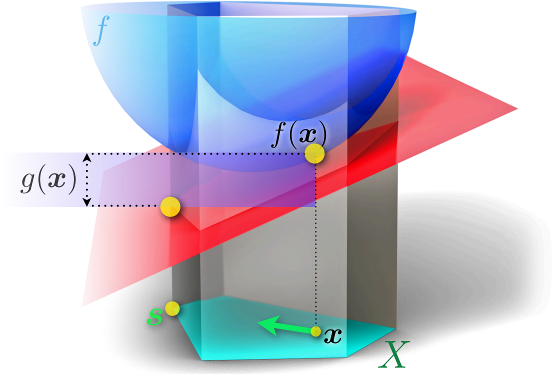
Duality Gap

Duality Gap

$$g(\mathbf{x}) := \langle \mathbf{x} - \mathbf{s}, \nabla f(\mathbf{x}) \rangle$$



Duality Gap



Duality Gap

$$g(\mathbf{x}) := \langle \mathbf{x} - \mathbf{s}, \nabla f(\mathbf{x}) \rangle$$

Certificate for optimization quality:

$$\begin{aligned} g(\mathbf{x}) &= \max_{\mathbf{s} \in X} \langle \mathbf{x} - \mathbf{s}, \nabla f(\mathbf{x}) \rangle \\ &\geq \langle \mathbf{x} - \mathbf{x}^\star, \nabla f(\mathbf{x}) \rangle \\ &\geq f(\mathbf{x}) - f(\mathbf{x}^\star) \end{aligned}$$

Stepsize variants

$$
\begin{aligned}
\gamma_t &:= \frac{2}{t+2}, \\
\gamma_t &:= \operatorname*{argmin}_{\gamma \in [0,1]} f\big((1-\gamma)\mathbf{x}_t + \gamma \mathbf{s}\big), && \text{(line-search)} \\
\gamma_t &:= \min\left\{ \frac{g(\mathbf{x}_t)}{L\,\|\mathbf{s}-\mathbf{x}_t\|^2}, 1 \right\}, && \text{(gap-based)}
\end{aligned}
$$

Convergence in $\mathcal{O}(1/\varepsilon)$ steps

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and smooth with parameter L, and $\mathbf{x}_0 \in X$. Then choosing any of the above stepsizes, the Frank-Wolfe algorithm yields

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \le \frac{2L\,\mathrm{diam}(X)^2}{T+1}$$

Where $\mathrm{diam}(X) := \max_{\mathbf{x},\mathbf{y}\in X} \|\mathbf{x}-\mathbf{y}\|$ is the diameter of X.

Proof of Convergence in $\mathcal{O}(1/\varepsilon)$ steps

Lemma

For a step $\mathbf{x}_{t+1} := \mathbf{x}_t + \gamma(\mathbf{s} - \mathbf{x}_t)$ *with arbitrary step-size* $\gamma \in [0, 1]$*, it holds that*

$$f(\mathbf{x}_{t+1}) \le f(\mathbf{x}_t) - \gamma g(\mathbf{x}_t) + \tfrac{\gamma^2}{2} L \operatorname{diam}(X)^2 \ ,$$

if $\mathbf{s} = \mathrm{LMO}(\nabla f(\mathbf{x}_t))$.

Proof of Convergence in $\mathcal{O}(1/\varepsilon)$ steps

Lemma

For a step $\mathbf{x}_{t+1} := \mathbf{x}_t + \gamma(\mathbf{s} - \mathbf{x}_t)$ *with arbitrary step-size* $\gamma \in [0,1]$*, it holds that*

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \gamma g(\mathbf{x}_t) + \tfrac{\gamma^2}{2} L \operatorname{diam}(X)^2 \ ,$$

if $\mathbf{s} = \mathrm{LMO}(\nabla f(\mathbf{x}_t))$*.*

Proof.

We write $\mathbf{x} := \mathbf{x}_t$, $\mathbf{y} := \mathbf{x}_{t+1} = \mathbf{x} + \gamma(\mathbf{s} - \mathbf{x})$. From the definition of smoothness of f, we have

$$\begin{aligned} f(\mathbf{y}) = &\quad f(\mathbf{x} + \gamma(\mathbf{s} - \mathbf{x})) \\ \leq &\quad f(\mathbf{x}) + \gamma\langle \mathbf{s} - \mathbf{x}, \nabla f(\mathbf{x})\rangle + \tfrac{\gamma^2}{2} L \operatorname{diam}(X)^2 \ . \end{aligned}$$

The lemma follows by definition of the duality gap. □

Proof of Convergence in $\mathcal{O}(1/\varepsilon)$ steps

From the Lemma we know that for every step of FW, it holds that

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \gamma g(\mathbf{x}_t) + \gamma^2 C,$$

if we chose $\gamma := \frac{2}{t+2}$ and write $C := \frac{1}{2} L \operatorname{diam}(X)^2$. This bound holds also for all mentioned line-search variants *(different LHS, same RHS)*.

Proof of Convergence in $\mathcal{O}(1/\varepsilon)$ steps

From the Lemma we know that for every step of FW, it holds that

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \gamma g(\mathbf{x}_t) + \gamma^2 C,$$

if we chose $\gamma := \frac{2}{t+2}$ and write $C := \frac{1}{2} L \operatorname{diam}(X)^2$. This bound holds also for all mentioned line-search variants *(different LHS, same RHS)*.

Writing $h(\mathbf{x}) := f(\mathbf{x}) - f(\mathbf{x}^\star)$ for the (unknown) objective error at any point $\mathbf{x}$, this implies that

$$\begin{aligned} h(\mathbf{x}_{t+1}) \leq& \quad h(\mathbf{x}_t) - \gamma g(\mathbf{x}_t) + \gamma^2 C \\ \leq& \quad h(\mathbf{x}_t) - \gamma h(\mathbf{x}_t) + \gamma^2 C \\ =& \quad (1-\gamma) h(\mathbf{x}_t) + \gamma^2 C \ , \end{aligned}$$

by the certificate property $h(\mathbf{x}) \leq g(\mathbf{x})$ of the duality gap.
The theorem then follows by induction (Exercise 1 of Lab 9). □

Affine Invariance

Curvature Constant

$$C_f := \sup_{\substack{\mathbf{x},\mathbf{s} \in X, \gamma \in [0,1] \\ \mathbf{y} = \mathbf{x} + \gamma(\mathbf{s} - \mathbf{x})}} \frac{2}{\gamma^2} \big(f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{y} - \mathbf{x}, \nabla f(\mathbf{x}) \rangle \big)$$

Algorithm is invariant to scaling (affine transformations) of the input problem.

So is C_f.

(same as Newton, but here for constrained problems)

Affine Invariance

Curvature Constant

$$C_f := \sup_{\substack{\mathbf{x},\mathbf{s}\in X,\gamma\in[0,1] \\ \mathbf{y}=\mathbf{x}+\gamma(\mathbf{s}-\mathbf{x})}} \frac{2}{\gamma^2}\big(f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{x})\rangle\big)$$

Algorithm is invariant to scaling (affine transformations) of the input problem.

So is C_f.

(same as Newton, but here for constrained problems)

$$C_f \leq L\,\mathrm{diam}(X)^2 \qquad \text{for any norm!}$$

All proofs hold for C_f, instead of picking a particular $L\,\mathrm{diam}(X)^2$.

Convergence in Duality Gap

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and smooth with parameter L, and $\mathbf{x}_0 \in X$, $T \geq 2$. Then choosing any of the above stepsizes, the Frank-Wolfe algorithm yields a $t, 1 \leq t \leq T$ s.t.

$$g(\mathbf{x}_t) \leq \frac{27/4\, C_f}{T+1}$$

Proof.

Idea: not all gaps can be small (use Lemma again). □

Extensions and Use Cases

Extensions:

- Approximate LMO (of additive of multiplicative accuracy)
- Randomized LMO
- unconstrained problems (Matching Pursuit variants)

Extensions and Use Cases

Extensions:

- Approximate LMO (of additive of multiplicative accuracy)
- Randomized LMO
- unconstrained problems (Matching Pursuit variants)

Use cases:

Whenever projection is more costly than solving a linear problem

- Lasso and other L1-constrained problems
- Matrix Completion: scalable algorithm
- Relaxation of combinatorial problems
  (e.g. matchings, network flows etc)

Applications

recall: $\mathrm{LMO}(\mathbf{g}) := \underset{\mathbf{s}\in X}{\operatorname{argmin}} \langle \mathbf{s}, \mathbf{g} \rangle$

$$X := conv(\mathcal{A})$$

Examples	$\mathcal{A}$	$\lvert\mathcal{A}\rvert$	d	$\mathrm{LMO}\ (\mathbf{g})$
L1-ball	$\{\pm\mathbf{e}_i\}$	$2d$	d	$\pm\mathbf{e}_i$ with $\operatorname{argmax}_i \lvert g_i\rvert$
Simplex	$\{\mathbf{e}_i\}$	d	d	$\mathbf{e}_i$ with $\operatorname{argmin}_i g_i$
Norms	$\{\mathbf{x}, \lVert\mathbf{x}\rVert \leq 1\}$	∞	d	$\underset{\mathbf{s},\lVert\mathbf{s}\rVert\leq 1}{\operatorname{argmin}} \langle \mathbf{s}, \mathbf{g} \rangle$
Nuclear norm	$\{Y, \lVert Y\rVert_* \leq 1\}$	∞	d^2	..
Wavelets	..	∞	∞	..

Chapter X

Accelerated Gradient Descent

Re-visiting gradient descent

Property of f	Learning Rate γ	Number of steps
$\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$, $\|\nabla f(\mathbf{x})\| \leq L$ for all $\mathbf{x}$	$\frac{R}{L\sqrt{T}}$	$\mathcal{O}(1/\varepsilon^2)$
f is L-smooth	$\frac{1}{L}$	$\mathcal{O}(1/\varepsilon)$
f is L-smooth and μ-strongly convex	$\frac{1}{L}$	$\mathcal{O}(\log(1/\varepsilon))$

Improving gradient descent

Problem: Can we do any better? In particular, can we accelerate gradient descent?

Solution: Nesterov's accelerated gradient methods come to the rescue.

Momentum

Idea:
Use **momentum** from “movement” so far

$$\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma \nabla f(\mathbf{x}_t) + \nu \big[\mathbf{x}_t - \mathbf{x}_{t-1} \big]$$

$\nu > 0$ is called the momentum parameter

Accelerated Gradient Method - AGD

Actual algorithm variant which can be analyzed:

$$
\begin{aligned}
\mathbf{x}_0 &:= \mathbf{y}_0 := \mathbf{z}_0 \\
\mathbf{y}_{t+1} &:= \mathbf{x}_t - \tfrac{1}{L}\nabla f(\mathbf{x}_t) && \text{the regular 'smooth' step} \\
\mathbf{z}_{t+1} &:= \mathbf{z}_t - \tfrac{t+1}{2L}\nabla f(\mathbf{x}_t) && \text{the fast 'aggressive' step} \\
\mathbf{x}_{t+1} &:= \tau \mathbf{y}_{t+1} + (1-\tau)\mathbf{z}_{t+1}
\end{aligned}
$$

for $\tau := \frac{t+1}{t+3}$.

Overview of Accelerated Gradient Method

Comparing Gradient Descent and Accelerated Gradient Descent for convex functions - number of updates to obtain an ε-optimal solution.

Properties of f	GD steps	AGD steps
f is L-smooth	$\mathcal{O}(1/\varepsilon)$	$\mathcal{O}(1/\sqrt{\varepsilon})$
f is L-smooth and μ-strongly convex	$\mathcal{O}(\frac{L}{\mu}\log(1/\varepsilon))$	$\mathcal{O}(\sqrt{\frac{L}{\mu}}\log(1/\varepsilon))$

Acceleration in practice

Application to a Lasso problem

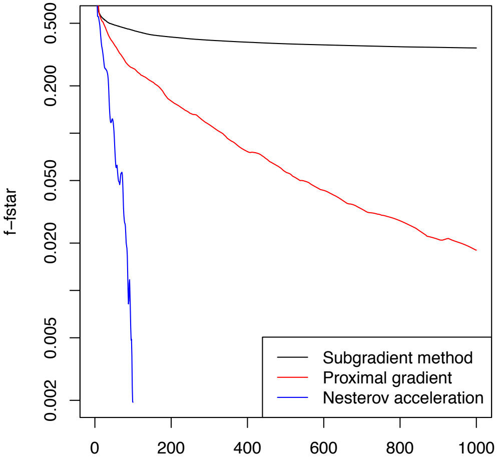


figure by Ryan Tibshirani, CMU

Acceleration in practice

Excellent illustration and simulation:

```
https://distill.pub/2017/momentum/
```

Potential issues

- requires tuning of a new hyperparameter (the momentum param)