CMPT 409/981: Optimization for Machine Learning

Lecture 2

Sharan Vaswani

September 12, 2022

Recap

Smooth functions: f is L-smooth if its gradient is Lipschitz continuous, and does not change arbitrarily fast i.e. $\forall x, y$, $|\nabla f(x) - \nabla f(y)| \leq L \, \|x - y\|$.

If f is L-smooth, then, for all $x, y \in \mathcal{D}$, $f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2$.

Objective: Find an ϵ-approximate stationary point $\hat{w}$ i.e. $\|\nabla f(\hat{w})\|^2 \leq \epsilon$ with access to a *first-order oracle* that returns $\{f(w), \nabla f(w)\}$ at any point $w \in \mathcal{D}$.

Minimizing the above upper-bound iteratively recovers gradient descent (GD) with $\eta = 1/L$.

Algorithmically, starting from an *initialization* equal to w_0, at iteration k, GD computes the gradient $\nabla f(w_k)$ at iterate w_k (call to the first-order oracle).

- If $\|\nabla f(w_k)\|^2 \leq \epsilon$, terminate and return $\hat{w} := w_k$.
- Else, update the iterate as: $w_{k+1} = w_k - \frac{1}{L} \nabla f(w_k)$.

Is GD guaranteed to terminate? If so, can we characterize the number of iterations?

Claim: For L-smooth functions, gradient descent with $\eta = \frac{1}{L}$ returns $\hat{w}$ such that $\|\nabla f(\hat{w})\|^2 \leq \epsilon$ and requires $T = \frac{2L\,[f(w_0) - \min_w f(w)]}{\epsilon}$ iterations (oracle calls).

Proof:
Using the L-smoothness of f with $x = w_k$ and $y = w_{k+1} = w_k - \frac{1}{L}\nabla f(w_k)$ in the quadratic bound (referred to as the *descent lemma*),

$$
\begin{aligned}
f(w_{k+1}) &\leq f(w_k) + \langle \nabla f(w_k), -\frac{1}{L}\nabla f(w_k)\rangle + \frac{L}{2}\left\|\frac{1}{L}\nabla f(w_k)\right\|^2 \\
\implies f(w_{k+1}) &\leq f(w_k) - \frac{1}{2L}\|\nabla f(w_k)\|^2
\end{aligned}
$$

By moving from w_k to w_{k+1}, we have decreased the value of f since $f(w_{k+1}) \leq f(w_k)$.

Gradient Descent

Rearranging the inequality from the previous slide, for every iteration k,

$$\frac{1}{2L}\left\|\nabla f(w_k)\right\|^2 \leq f(w_k) - f(w_{k+1})$$

By running GD for T iterations, adding up $k = 0$ to $T - 1$,

$$\frac{1}{2L}\sum_{k=0}^{T-1}\left\|\nabla f(w_k)\right\|^2 \leq \sum_{k=0}^{T-1}[f(w_k) - f(w_{k+1})] = f(w_0) - f(w_T) \leq [f(w_0) - \min_w f(w)]$$

$$\implies \frac{\sum_{k=0}^{T-1}\left\|\nabla f(w_k)\right\|^2}{T} \leq \frac{2L\,[f(w_0) - \min_w f(w)]}{T}$$

The LHS is the average of the gradient norms over the T iterates. Let $\hat{w} := \arg\min_{k\in\{0,1,\ldots,T-1\}}\left\|\nabla f(w_k)\right\|^2$. Since the minimum is smaller than the average,

$$\left\|\nabla f(\hat{w})\right\|^2 \leq \frac{2L\,[f(w_0) - \min_w f(w)]}{T}$$

Gradient Descent

Since $\|\nabla f(\hat{w})\|^2 \leq \frac{2L\,[f(w_0) - \min_w f(w)]}{T}$, the *rate of convergence* is $O(1/T)$.

If the RHS equal to $\frac{2L\,[f(w_0) - \min_w f(w)]}{T} \leq \epsilon$, this would guarantee that $\|\nabla f(\hat{w})\|^2 \leq \epsilon$ and we would achieve our objective.

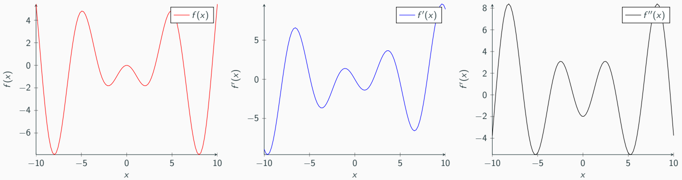
Hence, we need to run the algorithm for $T \geq \frac{2L\,[f(w_0) - \min_w f(w)]}{\epsilon}$ iterations. This is also referred to as an $O\left(\frac{1}{\epsilon}\right)$ convergence rate.

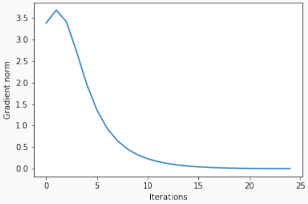
Lower-Bound: When minimizing a smooth function (without additional assumptions), any *first-order* algorithm requires $\Omega\left(\frac{1}{\epsilon}\right)$ oracle calls to return a point $\hat{w}$ such that $\|\nabla f(\hat{w})\|^2 \leq \epsilon$.

Hence, gradient descent is optimal for minimizing smooth functions!

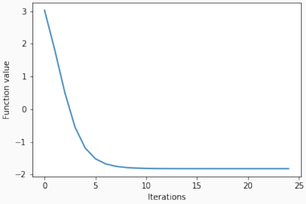
Gradient Descent – Example

$\min_{x\in[-10,10]} f(x) := -x\sin(x)$. Run GD with $\eta = 1/L \approx 0.1$ and $x_0 = 4$.





(a) Gradient norm



(b) Function value

Questions?

Gradient Descent

We have seen that we can reach a stationary point of a smooth function in $O\left(\frac{1}{\epsilon}\right)$ iterations of GD with step-size $\eta = \frac{1}{L}$.

Problems with this approach:

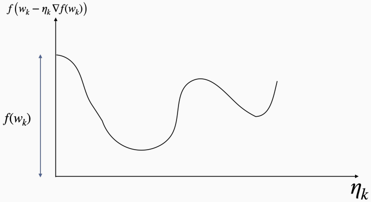
- Computing L in closed-form can be difficult as the functions get complicated.
- Theoretically computed L is global (the "local" L might be much smaller) and often loose in practice (typically we tend to overestimate L resulting in a smaller step-size).

Gradient Descent with Line-search

Instead of setting η according to L, we can "search" for a good step-size η_k in each iteration k.

Exact line-search: At iteration k, solve the following sub-problem:

$$\eta_k = \arg\min_{\eta} f(w_k - \eta \nabla f(w_k)).$$



After computing η_k, do the usual GD update: $w_{k+1} = w_k - \eta_k \nabla f(w_k)$.

- Can adapt to the "local" L, resulting in larger step-sizes and better performance.
- Can solve the sub-problem approximately by doing gradient descent w.r.t η (expensive).
- Can compute η_k analytically (only in special cases).

Gradient Descent with Line-search – Example

Recall linear regression: $\min_{w \in \mathbb{R}^d} f(w) := \frac{1}{2} \|Xw - y\|^2 = \frac{1}{2} [w^\intercal (X^\intercal X) w - 2 w^\intercal X^\intercal y + y^\intercal y]$.

For the exact line-search, we need to $\min_\eta h(\eta) := f(w_k - \eta \nabla f(w_k))$.

Since f is a quadratic, we can directly use the second-order Taylor series expansion.

$$
\begin{aligned}
h(\eta) &= f(w_k - \eta \nabla f(w_k)) \\
&= f(w_k) + \langle \nabla f(w_k), -\eta \nabla f(w_k) \rangle + \frac{1}{2} [-\eta \nabla f(w_k)]^\intercal \nabla^2 f(w_k) [-\eta \nabla f(w_k)]
\end{aligned}
$$

$$
\nabla h(\eta_k) = -\|\nabla f(w_k)\|^2 + \eta [\nabla f(w_k)]^\intercal \nabla^2 f(w_k) [\nabla f(w_k)] = 0 \implies \eta_k = \frac{\|\nabla f(w_k)\|^2}{\|\nabla f(w_k)\|^2_{\nabla^2 f(w_k)}}
$$

For linear regression, $\nabla^2 f(w_k) = X^\intercal X$ and $\nabla f(w_k) = X^\intercal (X w_k - y)$. With exact line-search, the GD update for linear regression is:

$$
w_{k+1} = w_k - \frac{\|X^\intercal (X w_k - y)\|^2}{\|X^\intercal (X w_k - y)\|^2_{X^\intercal X}} [X^\intercal (X w_k - y)]
$$

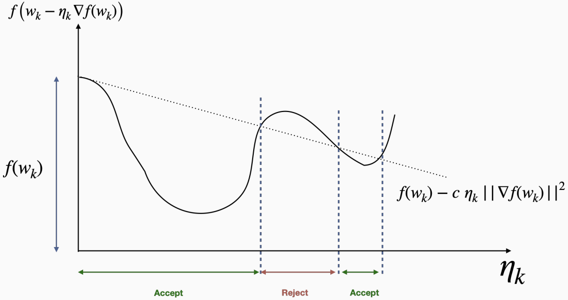
Gradient Descent with Line-search

Usually, the cost of doing an exact line-search is not worth the computational effort.

Armijo condition for a prospective step-size $\tilde{\eta}_k$:

$$f(w_k - \tilde{\eta}_k \nabla f(w_k)) \leq f(w_k) - c\, \tilde{\eta}_k \, \|\nabla f(w_k)\|^2$$

where $c \in (0, 1)$ is a hyper-parameter.



Gradient Descent with Line-search

Backtracking line-search: At iteration k, starting with an initial "guess" of the step-size η_{max}, check the Armijo condition for a prospective step-size $\tilde{\eta}_k$.

- If $\tilde{\eta}_k$ satisfies the Armijo condition, set $\eta_k = \tilde{\eta}_k$ and do the usual GD update.
- Else, decrease $\tilde{\eta}_k$ by a multiplicative factor $\beta \in (0, 1)$ and check the Armijo condition for the new prospective step-size equal to $\tilde{\eta}_k \beta$.
- Keep "backtracking" on $\tilde{\eta}_k$ until the Armijo condition is satisfied.
- Do the usual GD step: $w_{k+1} = w_k - \eta_k \nabla f(w_k)$ using the η_k for which the Armijo condition is satisfied.

Gradient Descent with Line-search

Claim: The (exact) backtracking procedure terminates and returns $\eta_k \geq \min\left\{\frac{2\,(1-c)}{L}, \eta_{\max}\right\}$.

Proof:

$$f(w_k - \tilde{\eta}_k \nabla f(w_k)) \leq \underbrace{f(w_k) - \|\nabla f(w_k)\|^2 \left(\eta_k - \frac{L\eta_k^2}{2}\right)}_{h_1(\tilde{\eta}_k)} \quad \text{(Quadratic bound using smoothness)}$$

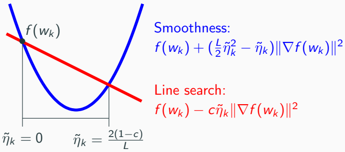
$$f(w_k - \tilde{\eta}_k \nabla f(w_k)) \leq \underbrace{f(w_k) - \|\nabla f(w_k)\|^2 (c\tilde{\eta}_k)}_{h_2(\tilde{\eta}_k)} \quad \text{(Armijo condition)}$$

If the Armijo condition is satisfied, the back-tracking line-search procedure terminates.

Case (i): For $\eta_{\max} \leq \frac{2(1-c)}{L}$,

$f(w_k - \eta_{\max} \nabla f(w_k)) \leq h_1(\eta_{\max}) \leq h_2(\eta_{\max})$

$\implies$ if $\eta_{\max} \leq \frac{2(1-c)}{L}$, then the line-search terminates immediately and $\eta_k = \eta_{\max}$.



Gradient Descent with Line-search

Case (ii): If $\eta_{\max} > \frac{2\,(1-c)}{L}$ and the Armijo condition is satisfied for step-size η_k, then
$f(w_k - \eta_k \nabla f(w_k)) \leq h_2(\eta_k) \leq h_1(\eta_k) \implies c\eta_k \geq \eta_k - \frac{L\eta_k^2}{2} \implies \eta_k \geq \frac{2(1-c)}{L}$.

Putting the two cases together, the step-size η_k returned by the Armijo line-search satisfies
$\eta_k \geq \min\left\{\frac{2\,(1-c)}{L}, \eta_{\max}\right\}$.

Gradient Descent with Line-search

Claim: Gradient Descent with (exact) backtracking Armijo line-search (with $c = 1/2$) returns point $\hat{w}$ such that $\|\nabla f(\hat{w})\|^2 \leq \epsilon$ and requires $T = \frac{2L\,[f(w_0) - \min_w f(w)]}{\epsilon}$ oracle calls or iterations.
Proof: Since η_k satisfies the Armijo condition and $w_{k+1} = w_k - \eta_k \nabla f(w_k)$,

$$
\begin{aligned}
f(w_{k+1}) &\leq f(w_k) - c\,\eta_k \, \|\nabla f(w_k)\|^2 \\
&\leq f(w_k) - \left(\min\left\{\frac{1}{2L}, \eta_{\max}\right\}\right) \|\nabla f(w_k)\|^2
\end{aligned}
$$

(Result from previous slide with $c = 1/2$)

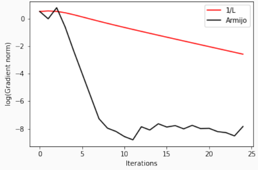
Continuing the proof as before,

$$
\implies \|\nabla f(\hat{w})\|^2 \leq \frac{\max\{2L, 1/\eta_{\max}\}\,[f(w_0) - \min_w f(w)]}{T}
$$

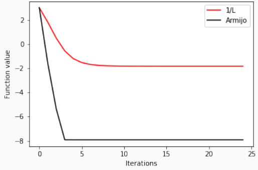
The claim is proved by reasoning as before.

Gradient Descent with Line-search – Examples

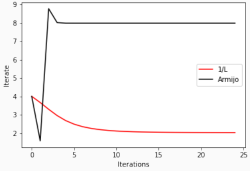
$\min_{x\in[-10,10]} f(x) := -x\sin(x)$. Compare GD (with $x_0 = 4$) with (i) $\eta = 1/L \approx 0.1$ and (ii) Armijo line-search with $\eta_{\max} = 10, c = 1/2, \beta = 0.9$.



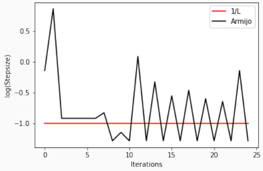
(a) Gradient norm



(b) Function value



(c) Iterate



(d) Stepsize

Questions?