Optimization for Machine Learning
CS-439

Lecture 6: SGD, Non-convex optimization

Martin Jaggi

EPFL – github.com/epfml/OptML_course

March 29, 2019

Tame strong convexity: $\mathcal{O}(1/\varepsilon)$ steps

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable and strongly convex with parameter $\mu > 0$; let $\mathbf{x}^\star$ be the unique global minimum of f. With decreasing step size

$$\gamma_t := \frac{2}{\mu(t+1)}$$

stochastic gradient descent yields

$$\mathbb{E}\Big[f\Big(\frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \mathbf{x}_t\Big) - f(\mathbf{x}^\star)\Big] \leq \frac{2B^2}{\mu(T+1)},$$

where $B^2 := \max_{t=1}^{T} \mathbb{E}\big[\,\|\mathbf{g}_t\|^2\,\big]$.

Almost same result as for subgradient descent, but in expectation.

Tame strong convexity: $\mathcal{O}(1/\varepsilon)$ steps II

Proof.

Take expectations over vanilla analysis, before summing up (with varying stepsize γ_t):

$$\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] = \frac{\gamma_t}{2}\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] + \frac{1}{2\gamma_t}\left(\mathbb{E}\big[\|\mathbf{x}_t - \mathbf{x}^\star\|^2\big] - \mathbb{E}\big[\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\big]\right).$$

"Strong convexity in expectation":

$$\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] = \mathbb{E}\big[\nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] \geq \mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big] + \frac{\mu}{2}\mathbb{E}\big[\|\mathbf{x}_t - \mathbf{x}^\star\|^2\big]$$

Putting it together (with $\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] \leq B^2$):

$$\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^\star)] \leq \frac{B^2\gamma_t}{2} + \frac{(\gamma_t^{-1} - \mu)}{2}\mathbb{E}\big[\,\|\mathbf{x}_t - \mathbf{x}^\star\|^2\,\big] - \frac{\gamma_t^{-1}}{2}\mathbb{E}\big[\,\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\,\big].$$

Proof continues as for subgradient descent, this time with expectations. $\square$

Mini-batch SGD

Instead of using a single element f_i, use an average of several of them:

$$\tilde{\mathbf{g}}_t := \frac{1}{m} \sum_{j=1}^{m} \mathbf{g}_t^j.$$

Extreme cases:

$m = 1 \Leftrightarrow$ SGD as originally defined

$m = n \Leftrightarrow$ full gradient descent

Benefit: Gradient computation can be naively parallelized

Mini-batch SGD

Variance Intuition: Taking an average of many independent random variables reduces the variance. So for larger size of the mini-batch m, $\tilde{\mathbf{g}}_t$ will be closer to the true gradient, in expectation:

$$\begin{aligned}
\mathbb{E}\left[\left\|\tilde{\mathbf{g}}_t - \nabla f(\mathbf{x}_t)\right\|^2\right] =& \mathbb{E}\left[\left\|\frac{1}{m}\sum_{j=1}^{m} \mathbf{g}_t^j - \nabla f(\mathbf{x}_t)\right\|^2\right] \\
=& \frac{1}{m}\mathbb{E}\left[\|\mathbf{g}_t^1 - \nabla f(\mathbf{x}_t)\|^2\right] \\
=& \frac{1}{m}\mathbb{E}\left[\|\mathbf{g}_t^1\|^2\right] - \frac{1}{m}\|\nabla f(\mathbf{x}_t)\|^2 \le \frac{B^2}{m} \, .
\end{aligned}$$

Using a modification of the SGD analysis, can use this quantity to relate convergence rate to the rate of full gradient descent.

Stochastic Subgradient Descent

For problems which are not necessarily differentiable, we modify SGD to use a subgradient of f_i in each iteration. The update of **stochastic subgradient descent** is given by

> sample $i \in [n]$ uniformly at random
>
> let $\mathbf{g}_t \in \partial f_i(\mathbf{x}_t)$
>
> $\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma_t \mathbf{g}_t.$

In other words, we are using an unbiased estimate of a subgradient at each step, $\mathbb{E}\big[\mathbf{g}_t \big| \mathbf{x}_t\big] \in \partial f(\mathbf{x}_t)$.

Convergence in $\mathcal{O}(1/\varepsilon^2)$, by using the subgradient property at the beginning of the proof, where convexity was applied.

Constrained optimization

For constrained optimization, our theorem for the SGD convergence in $\mathcal{O}(1/\varepsilon^2)$ steps directly extends to constrained problems as well.

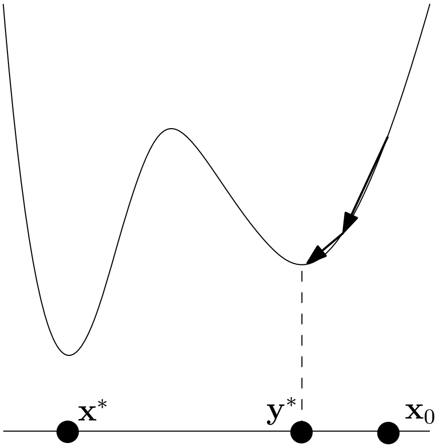
After every step of SGD, projection back to X is applied as usual. The resulting algorithm is called projected SGD.

Chapter 6

Non-convex Optimization

Gradient Descent in the nonconvex world

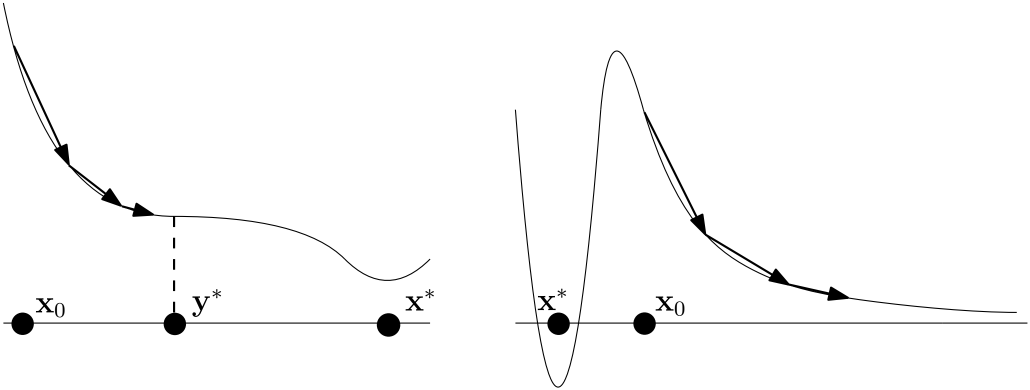
- may get stuck in a local minimum and miss the global minimum;



Gradient Descent in the nonconvex world II

Even if there is a unique local minimum (equal to the global minimum), we

- may get stuck in a saddle point;
- run off to infinity;
- possibly encounter other bad behaviors.



Gradient Descent in the nonconvex world III

Often, we observe good behavior in practice.

Theoretical explanations mostly missing.

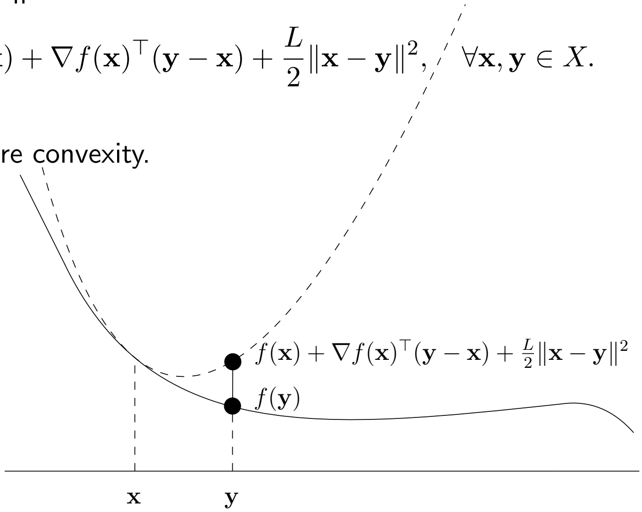
This lecture: under favorable conditions, we sometimes can say something useful about the behavior of gradient descent, even on nonconvex functions.

Smooth (but not necessarily convex) functions

Recall: A differentiable $f : \mathbf{dom}(f) \to \mathbb{R}$ is smooth with parameter $L \in \mathbb{R}_+$ over a convex set $X \subseteq \mathbf{dom}(f)$ if

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in X. \tag{1}$$

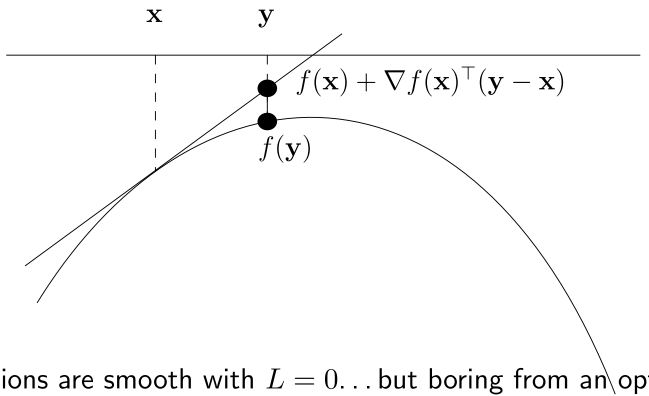
Definition does not require convexity.



Concave functions

f is called **concave** if $-f$ is convex.

For all $\mathbf{x}$, the graph of a differentiable concave function is below the tangent hyperplane at $\mathbf{x}$.



$\Rightarrow$ concave functions are smooth with $L = 0$... but boring from an optimization point of view (no global minimum), gradient descent runs off to infinity

Bounded Hessians $\Rightarrow$ smooth

Lemma

Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be twice differentiable, with $X \subseteq \mathbf{dom}(f)$ a convex set, and $\left\|\nabla^2 f(\mathbf{x})\right\| \leq L$ for all $\mathbf{x} \in X$, where $\|\cdot\|$ is spectral norm. Then f is smooth with parameter L over X.

Examples:

- all quadratic functions $f(\mathbf{x}) = \mathbf{x}^\top A\mathbf{x} + \mathbf{b}^\top\mathbf{x} + c$
- $f(x) = \sin(x)$ (many global minima)

Bounded Hessians $\Rightarrow$ smooth II

Proof.

By Theorem 1.10 (applied to the gradient function ∇f), bounded Hessians imply Lipschitz continuity of the gradient,

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|, \quad \mathbf{x}, \mathbf{y} \in X.$$

To show that this implies smoothness, we use $h(1) - h(0) = \int_0^1 h'(t)dt$ with

$$h(t) := f\big(\mathbf{x} + t(\mathbf{y} - \mathbf{x})\big), \quad t \in [0, 1],$$

Chain rule:

$$h'(t) = \nabla f\big(\mathbf{x} + t(\mathbf{y} - \mathbf{x})\big)^\top (\mathbf{y} - \mathbf{x}).$$

Bounded Hessians $\Rightarrow$ smooth III

Proof.

For $\mathbf{x}, \mathbf{y} \in X$:

$$
\begin{aligned}
& f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \\
= \; & h(1) - h(0) - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \quad \text{(definition of } h\text{)} \\
= \; & \int_0^1 h'(t)dt - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \\
= \; & \int_0^1 \nabla f(\mathbf{x}+t(\mathbf{y}-\mathbf{x}))^\top(\mathbf{y}-\mathbf{x})dt - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \\
= \; & \int_0^1 \left(\nabla f(\mathbf{x}+t(\mathbf{y}-\mathbf{x}))^\top(\mathbf{y}-\mathbf{x}) - \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x})\right)dt \\
= \; & \int_0^1 \left(\nabla f(\mathbf{x}+t(\mathbf{y}-\mathbf{x})) - \nabla f(\mathbf{x})\right)^\top(\mathbf{y}-\mathbf{x})dt
\end{aligned}
$$

Bounded Hessians $\Rightarrow$ smooth IV

Proof.

For $\mathbf{x}, \mathbf{y} \in X$:

$$
\begin{aligned}
& f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \\
= & \int_0^1 \big(\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\big)^\top (\mathbf{y} - \mathbf{x}) dt \\
\leq & \int_0^1 \left| \big(\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\big)^\top (\mathbf{y} - \mathbf{x}) \right| dt \\
\leq & \int_0^1 \left\| \big(\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\big) \right\| \|(\mathbf{y} - \mathbf{x})\| \, dt \quad \text{(Cauchy-Schwarz)} \\
\leq & \int_0^1 L \|t(\mathbf{y} - \mathbf{x})\| \|(\mathbf{y} - \mathbf{x})\| \, dt \quad \text{(Lipschitz continuous gradients (6.1))} \\
= & \int_0^1 Lt \|\mathbf{x} - \mathbf{y}\|^2 = \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2 .
\end{aligned}
$$

Smooth $\Rightarrow$ bounded Hessians?

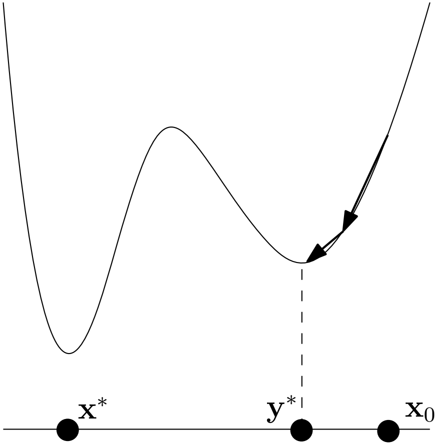
Yes, over any open convex set X (Exercise 33).

Gradient descent on smooth functions

Will prove: $\|\nabla f(\mathbf{x}_t)\|^2 \to 0$ for $t \to \infty$...

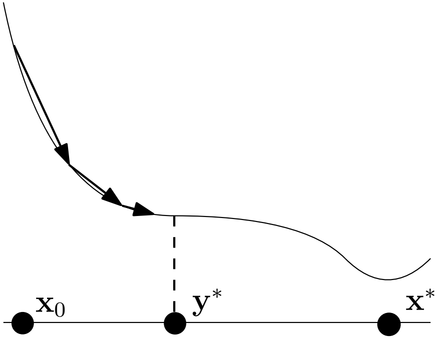
...at the same rate as $f(\mathbf{x}_t) - f(\mathbf{x}^\star) \to 0$ in the convex case.

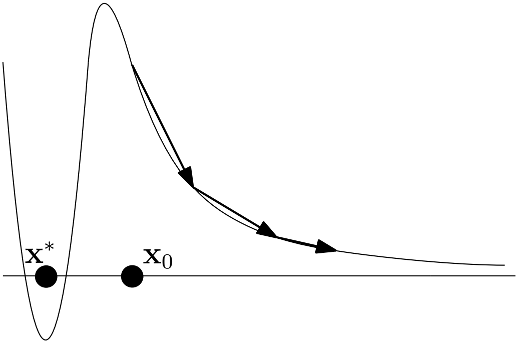
$f(\mathbf{x}_t) - f(\mathbf{x}^\star)$ itself may not converge to 0 in the nonconvex case:



What does $\|\nabla f(\mathbf{x}_t)\|^2 \to 0$ mean?

It may or may not mean that we converge to a **critical point** $(\nabla f(\mathbf{y}^\star) = \mathbf{0})$





Gradient descent on smooth (not necessarily convex) functions

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable with a global minimum $\mathbf{x}^\star$; furthermore, suppose that f is smooth with parameter L according to Definition 2.2. Choosing stepsize

$$\gamma := \frac{1}{L},$$

gradient descent yields

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(\mathbf{x}_t)\|^2 \leq \frac{2L}{T}\big(f(\mathbf{x}_0) - f(\mathbf{x}^\star)\big), \quad T > 0.$$

In particular, $\|\nabla f(\mathbf{x}_t)\|^2 \leq \frac{2L}{T}\big(f(\mathbf{x}_0) - f(\mathbf{x}^\star)\big)$ *for some* $t \in \{0, \ldots, T-1\}$.

And also, $\lim_{t\to\infty} \|\nabla f(\mathbf{x}_t)\|^2 = 0$ *(Exercise 34).*

Gradient descent on smooth (not necessarily convex) functions II

Proof.

Sufficient decrease (Lemma 2.6), does not require convexity:

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2, \quad t \geq 0.$$

Rewriting:

$$\|\nabla f(\mathbf{x}_t)\|^2 \leq 2L\big(f(\mathbf{x}_t) - f(\mathbf{x}_{t+1})\big).$$

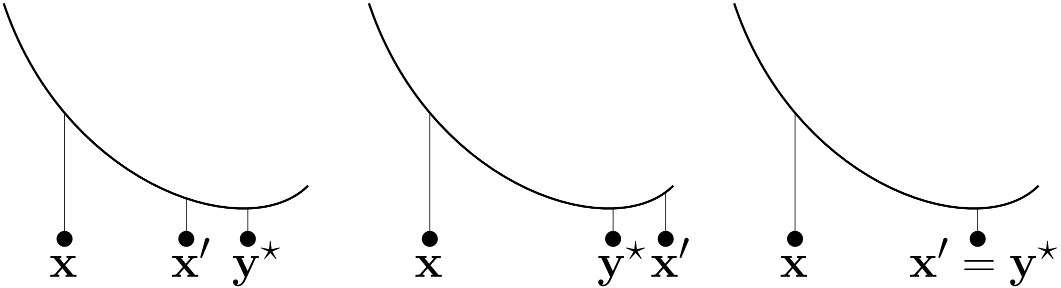
Telescoping sum:

$$\sum_{t=0}^{T-1} \|\nabla f(\mathbf{x}_t)\|^2 \leq 2L\big(f(\mathbf{x}_0) - f(\mathbf{x}_T)\big) \leq 2L\big(f(\mathbf{x}_0) - f(\mathbf{x}^\star)\big).$$

The statement follows (divide by T). □

No overshooting

In the smooth setting, and with stepsize $1/L$, gradient descent cannot overshoot, i.e. pass a critical point (Exercise 35).



$\mathbf{x}' = \mathbf{x} - \gamma \nabla f(\mathbf{x}), \gamma < 1/L$ overshooting may happen with $\gamma = 1/L$

Trajectory Analysis

Even if the “landscape” (graph) of a nonconvex function has local minima, saddle points, and flat parts, gradient descent may avoid them and still converge to a global minimum.

For this, one needs a good starting point and some theoretical understanding of what happens when we start there—this is **trajectory analysis**.

2018: trajectory analysis for training deep linear linear neural networks, under suitable conditions [ACGH18].

Here: vastly simplified setting that allows us to show the main ideas (and limitations).

Linear models with several outputs

Recall: Learning linear models

- n inputs $\mathbf{x}_1, \ldots, \mathbf{x}_n$, where each input $\mathbf{x}_i \in \mathbb{R}^d$
- n outputs $y_1, \ldots, y_n \in \mathbb{R}$
- Hypothesis (after centering):

$$y_i \approx \mathbf{w}^\top x_i,$$

for a weight vector $\mathbf{w} = (w_1, \ldots, w_d) \in \mathbb{R}^d$ to be learned.

Now more than one output value:

- n outputs $\mathbf{y}_1, \ldots, \mathbf{y}_n$, where each output $\mathbf{y}_i \in \mathbb{R}^m$
- Hypothesis:

$$\mathbf{y}_i \approx W\mathbf{x}_i,$$

for a weight matrix $W \in \mathbb{R}^{m \times d}$ to be learned.

Minimizing the least squares error

Compute

$$W^\star = \underset{W \in \mathbb{R}^{m \times d}}{\operatorname{argmin}} \sum_{i=1}^{n} \|W\mathbf{x}_i - \mathbf{y}_i\|^2 .$$

- $X \in \mathbb{R}^{d \times n}$: matrix whose columns are the $\mathbf{x}_i$
- $Y \in \mathbb{R}^{m \times n}$: matrix whose columns are the $\mathbf{y}_i$

Then

$$W^\star = \underset{W \in \mathbb{R}^{m \times d}}{\operatorname{argmin}} \|WX - Y\|_F^2,$$

where $\|A\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$ is the Frobenius norm of a matrix A.

Frobenius norm of A = Euclidean norm of $\operatorname{vec}(A)$ ("flattening" of A)

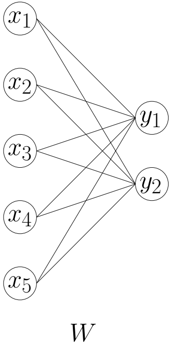
Minimizing the least squares error II

$$W^\star = \underset{W \in \mathbb{R}^{m \times d}}{\operatorname{argmin}} \|WX - Y\|_F^2$$

is the global minimum of a convex quadratic function $f(W)$.

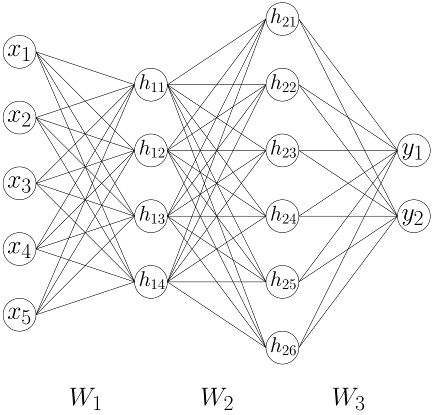
To find $W^\star$, solve $\nabla f(W) = \mathbf{0}$ (system of linear equations).

$\Leftrightarrow$ training a **linear neural network with one layer** under least squares error.



$$\mathbf{x} \mapsto \mathbf{y} = W\mathbf{x}$$

Deep linear neural networks



$$\mathbf{x} \mapsto \mathbf{y} = W_3 W_2 W_1 \mathbf{x}$$

Not more expressive:

$$\mathbf{x} \mapsto \mathbf{y} = W_3 W_2 W_1 \mathbf{x} \quad \Leftrightarrow \quad \mathbf{x} \mapsto \mathbf{y} = W\mathbf{x},\ W := W_3 W_2 W_1.$$

Training deep linear neural networks

With ℓ layers:

$$W^\star = \underset{W_1, W_2, \ldots, W_\ell}{\operatorname{argmin}} \|W_\ell W_{\ell-1} \cdots W_1 X - Y\|_F^2,$$

Nonconvex function for $\ell > 1$.

Simple playground in which we can try to understand why training deep neural networks with gradient descent works.

Here: all matrices are 1×1, $W_i = x_i, X = 1, Y = 1, \ell = d \quad \Rightarrow f : \mathbb{R}^d \to \mathbb{R}$,

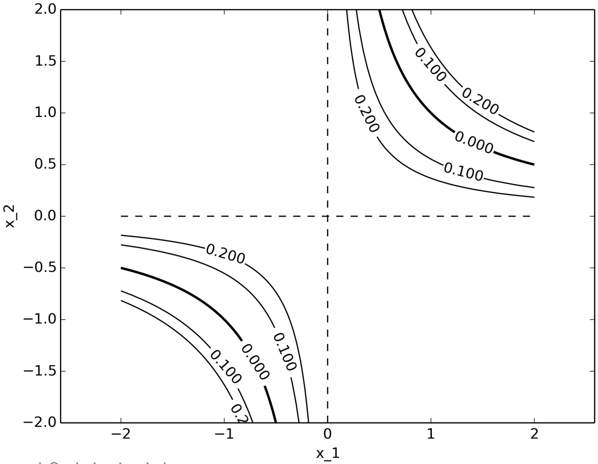
$$f(\mathbf{x}) := \frac{1}{2}\left(\prod_{k=1}^{d} x_k - 1\right)^2.$$

Toy example in our simple playground.

But analysis of gradient descent on f has similar ingredients as the one on general deep linear neural networks [ACGH18].

A simple nonconvex function

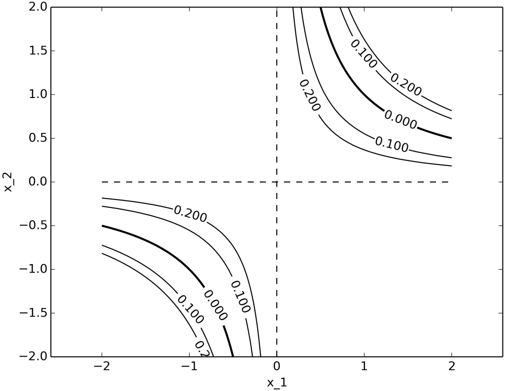
As d is fixed, abbreviate $\prod_{k=1}^{d} x_k$ by $\prod_k x_k$: $f(\mathbf{x}) = \frac{1}{2}\left(\prod_k x_k - 1\right)^2$



Level set plot

The gradient

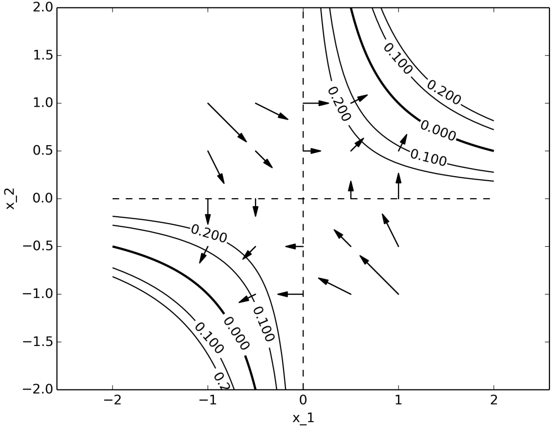
$$\nabla f(\mathbf{x}) = \left(\prod_k x_k - 1\right)\left(\prod_{k\neq 1} x_k, \ldots, \prod_{k\neq d} x_k\right).$$



Critical points ($\nabla f(\mathbf{x}) = \mathbf{0}$):

- $\prod_k x_k = 1$ (global minima)
    - $d = 2$: the hyperbola $\{(x_1, x_2) : x_1 x_2 = 1\}$
- at least two of the x_k are zero (saddle points)
    - $d = 2$: the origin $(x_1, x_2) = (0, 0)$

Negative gradient directions (followed by gradient descent)



Difficult to avoid convergence to a global minimum, but it is possible (Exercise 37).

Convergence analysis: Overview

Want to show that for any $d > 1$, and from anywhere in $X = \{\mathbf{x} : \mathbf{x} > \mathbf{0}, \prod_k \mathbf{x}_k \leq 1\}$, gradient descent will converge to a global minimum.

f is not smooth over X. We show that f is smooth along the trajectory of gradient descent for suitable L, so that we get sufficient decrease

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2, \quad t \geq 0.$$

Then, we cannot converge to a saddle point: all these have (at least two) zero entries and therefore function value $1/2$. But for starting point $\mathbf{x}_0 \in X$, we have $f(\mathbf{x}_0) < 1/2$, so we can never reach a saddle while decreasing f.

Doesn't this imply converge to a global mimimum? No!

- Sublevel sets are unbounded, so we could in principle run off to infinity.
- Other bad things might happen (we haven't characterized what can go wrong).

Convergence analysis: Overview II

For $\mathbf{x} > \mathbf{0}, \prod_k \mathbf{x}_k \geq 1$, we also get convergence (Exercise 36).

$\Rightarrow$ convergence from anywhere in the interior of the positive orthant $\{\mathbf{x} : \mathbf{x} > \mathbf{0}\}$.

But there are also starting points from which gradient descent will not converge to a global minimum (Exercise 37).

Main tool: Balanced iterates

Definition

Let $\mathbf{x} > \mathbf{0}$ (componentwise), and let $c \geq 1$ be a real number. $\mathbf{x}$ is called c-balanced if $x_i \leq c x_j$ for all $1 \leq i, j \leq d$.

Any initial iterate $\mathbf{x}_0 > \mathbf{0}$ is c-balanced for some (possibly large) c.

Lemma

Let $\mathbf{x} > \mathbf{0}$ be c-balanced with $\prod_k x_k \leq 1$. Then for any stepsize $\gamma > 0$, $\mathbf{x}' := \mathbf{x} - \gamma \nabla f(\mathbf{x})$ satisfies $\mathbf{x}' \geq \mathbf{x}$ (componentwise) and is also c-balanced.

Proof.

$$\Delta := -\gamma\left(\prod_k x_k - 1\right)\left(\prod_k x_k\right) \geq 0. \qquad \nabla f(\mathbf{x}) = \left(\prod_k x_k - 1\right)\left(\prod_{k\neq 1} x_k, \ldots, \prod_{k \neq d} x_k\right).$$

Gradient descent step:

$$x'_k = x_k + \frac{\Delta}{x_k} \geq x_k, \quad k = 1, \ldots, d.$$

For i, j, we have $x_i \leq c x_j$ and $x_j \leq c x_i$ ($\Leftrightarrow 1/x_i \leq c/x_j$). We therefore get

$$x'_i = x_i + \frac{\Delta}{x_i} \leq c x_j + \frac{\Delta c}{x_j} = c x'_j.$$

$\square$

Bounded Hessians along the trajectory

Compute $\nabla^2 f(\mathbf{x})$:

$\nabla^2 f(\mathbf{x})_{ij}$ is the j-th partial derivative of the i-th entry of $\nabla f(\mathbf{x})$.

$$(\nabla f)_i = \left(\prod_k x_k - 1\right) \prod_{k \neq i} x_k$$

$$\nabla^2 f(\mathbf{x})_{ij} = \begin{cases} \left(\prod_{k \neq i} x_k\right)^2, & j = i \\ 2 \prod_{k \neq i} x_k \prod_{k \neq j} x_k - \prod_{k \neq i,j} x_k, & j \neq i \end{cases}$$

Need to bound $\prod_{k \neq i} x_k, \quad \prod_{k \neq j} x_k, \quad \prod_{k \neq i,j} x_k$!

Bounded Hessians along the trajectory II

Lemma

Suppose that $\mathbf{x} > \mathbf{0}$ is c-balanced. Then for any $I \subseteq \{1, \dots, d\}$, we have

$$\left(\frac{1}{c}\right)^{|I|} \left(\prod_k x_k\right)^{1-|I|/d} \leq \prod_{k \notin I} x_k \leq c^{|I|} \left(\prod_k x_k\right)^{1-|I|/d} .$$

Proof.

For any i, we have $x_i^d \geq (1/c)^d \prod_k x_k$ by balancedness, hence $x_i \geq (1/c)(\prod_k x_k)^{1/d}$.It follows that

$$\prod_{k \notin I} x_k = \frac{\prod_k x_k}{\prod_{i \in I} x_i} \leq \frac{\prod_k x_k}{(1/c)^{|I|}(\prod_k x_k)^{|I|/d}} = c^{|I|} \left(\prod_k x_k\right)^{1-|I|/d} .$$

The lower bound follows in the same way from $x_i^d \leq c^d \prod_k x_k$. $\square$

Bounded Hessians along the trajectory III

Lemma

Let $\mathbf{x} > \mathbf{0}$ be c-balanced with $\prod_k x_k \leq 1$. Then

$$\left\|\nabla^2 f(\mathbf{x})\right\| \leq \left\|\nabla^2 f(\mathbf{x})\right\|_F \leq 3dc^2.$$

where $\|A\|_F$ is the Frobenius norm and $\|A\|$ the spectral norm.

Proof.

$\|A\| \leq \|A\|_F$: Exercise 38. Now use previous lemma and $\prod_k x_k \leq 1$:

$$\left|\nabla^2 f(\mathbf{x})_{ii}\right| = |(\prod_{k \neq i} x_k)^2| \leq c^2$$

$$\left|\nabla^2 f(\mathbf{x})_{ij}\right| \leq |2 \prod_{k \neq i} x_k \prod_{k \neq j} x_k| + |\prod_{k \neq i,j} x_k| \leq 3c^2.$$

Hence, $\left\|\nabla^2 f(\mathbf{x})\right\|_F^2 \leq 9d^2c^4$. Taking square roots, the statement follows. $\square$

Smoothness along the trajectory

Lemma

Let $\mathbf{x} > \mathbf{0}$ be c-balanced with $\prod_k x_k < 1$, $L = 3dc^2$. Let $\gamma := 1/L$. Then for all $0 \leq \nu \leq \gamma$,

$$\mathbf{x}' := \mathbf{x} - \nu \nabla f(\mathbf{x}) \geq \mathbf{x}$$

is c-balanced with $\prod_k x'_k \leq 1$, and f is smooth with parameter L over the line segment connecting $\mathbf{x}$ and $\mathbf{x} - \gamma \nabla f(\mathbf{x})$.

Proof.

- $\mathbf{x}' \geq \mathbf{x} > \mathbf{0}$ is c-balanced by Lemma 6.5.
- $\nabla f(\mathbf{x}) \neq \mathbf{0}$ (due to $\mathbf{x}' > \mathbf{0}, \prod_k x_k < 1$, we can't be at a critical point).
- No overshooting: we can't reach $\prod_k x'_k = 1$ (global minimum) for $\nu < \gamma$, as f is smooth with parameter L between $\mathbf{x}$ and $\mathbf{x}'$ (using previous bound on Hessians in Lemma 6.1).
- By continutity, $\prod_k x'_k \leq 1$ for all $\nu \leq \gamma$.
- f is smooth with parameter L between $\mathbf{x}$ and $\mathbf{x}'$ for $\nu = \gamma$.

Convergence

Theorem

Let $c \geq 1$ and $\delta > 0$ such that $\mathbf{x}_0 > \mathbf{0}$ is c-balanced with $\delta \leq \prod_k (\mathbf{x}_0)_k < 1$. Choosing stepsize

$$\gamma = \frac{1}{3dc^2},$$

gradient descent satisfies

$$f(\mathbf{x}_T) \leq \left(1 - \frac{\delta^2}{3c^4}\right)^T f(\mathbf{x}_0), \quad T \geq 0.$$

- Error converges to 0 exponentially fast.
- Exercise 39: iterates themselves converge (to an optimal solution).

Convergence: Proof

Proof.

- For $t \geq 0$, f is smooth between $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$ with parameter $L = 3dc^2$.
- Sufficient decrease:

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{6dc^2} \left\|\nabla f(\mathbf{x}_t)\right\|^2 .$$

For every c-balanced $\mathbf{x}$ with $\delta \leq \prod_k x_k \leq 1$, $\left\|\nabla f(\mathbf{x})\right\|^2$ equals

$$2f(\mathbf{x}) \sum_{i=1}^{d} \left(\prod_{k \neq i} x_k \right)^2 \geq 2f(\mathbf{x}) \frac{d}{c^2} \left(\prod_k x_k \right)^{2-2/d} \geq 2f(\mathbf{x}) \frac{d}{c^2} \left(\prod_k x_k \right)^2 \geq 2f(\mathbf{x}) \frac{d}{c^2} \delta^2.$$

- Hence, $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \dfrac{1}{6dc^2} 2f(\mathbf{x}_t) \dfrac{d}{c^2} \delta^2 = f(\mathbf{x}_t) \left(1 - \dfrac{\delta^2}{3c^4}\right).$

Discussion

Fast convergence as for strongly convex functions!

But there is a catch...

Consider starting solution $\mathbf{x}_0 = (1/2, \ldots, 1/2)$.

$\delta \leq \prod_k (\mathbf{x}_0)_k = 2^{-d}$.

Decrease in function value by a factor of

$$\left(1 - \frac{1}{3 \cdot 4^d}\right),$$

per step.

Need $T \approx 4^d$ to reduce the initial error by a constant factor not depending on d.

Problem: gradients are exponentially small in the beginning, extremely slow progress.

For polynomial runtime, must start at distance $O(1/\sqrt{d})$ from optimality.

Bibliography

Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu.
A convergence analysis of gradient descent for deep linear neural networks.
CoRR, abs/1810.02281, 2018.