

IV.4 | Gradient Descent Toward the Minimum

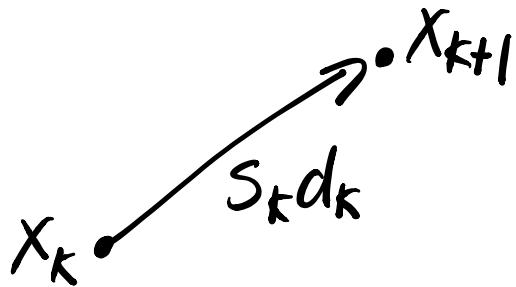
Algorithms for solving unconstrained optimization problems

$$\min_{x \in \mathbb{R}^n} f(x)$$

iterate

$$x_{k+1} = x_k + s_k d_k, \quad k=0, 1, 2, \dots,$$

where $s_k \in \mathbb{R}$ are the nonnegative step lengths and $d_k \in \mathbb{R}^n$ are the directions we move each iteration.



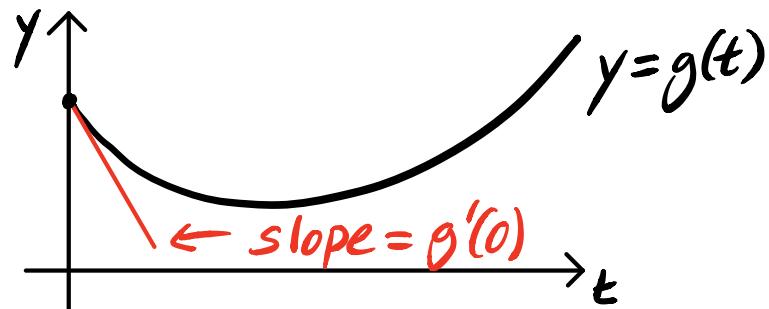
Line search methods choose d_k first and s_k second. Trust region methods choose s_k first and d_k second.

The direction of steepest descent

For which direction $d \in \mathbb{R}^n$ does the function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ have the steepest negative slope at the point $\bar{x} \in \mathbb{R}^n$?

The value of the function f starting at \bar{x} and moving in the direction d is given by:

$$g(t) = f(\bar{x} + td), \quad t \geq 0.$$



We want the $d \in \mathbb{R}^n$ such that

$g'(0) = \nabla f(\bar{x})^T d$ is the most negative.

We need to restrict ourselves to

$\|d\|_2 = 1$ for this to make sense. Also,

we need to assume that $\nabla f(\bar{x}) \neq 0$.

Thus, we want to solve

$$\min \nabla f(\bar{x})^T d \text{ subject to } \|d\|_2^2 = 1.$$

The Lagrangian

$$L(d, \lambda) = \nabla f(\bar{x})^T d + \lambda(1 - \|d\|_2^2)$$

has

$$\nabla_d L(d, \lambda) = \nabla f(\bar{x}) - 2\lambda d = 0$$

$$\text{when } d = \frac{1}{2\lambda} \nabla f(\bar{x}).$$

(Note that $\lambda \neq 0$ since $\nabla f(\bar{x}) \neq 0$.)

Now we use $\|d\|_2 = 1$ to find λ :

$$1 = \left\| \frac{1}{2\lambda} \nabla f(\bar{x}) \right\|_2 = \frac{1}{2|\lambda|} \left\| \nabla f(\bar{x}) \right\|_2$$

$$\Rightarrow \lambda = \pm \frac{1}{2} \left\| \nabla f(\bar{x}) \right\|_2.$$

The positive λ gives the maximum slope and the negative λ gives the minimum.

$$\therefore \lambda = -\frac{1}{2} \|\nabla f(\bar{x})\|_2 \Rightarrow d = -\frac{\nabla f(\bar{x})}{\|\nabla f(\bar{x})\|_2}$$

is the direction of steepest descent.

(Note: We also find that $d = \frac{\nabla f(\bar{x})}{\|\nabla f(\bar{x})\|_2}$ is the direction of steepest ascent.)

Gradient Descent: $d_k = -\nabla f(x_k)$

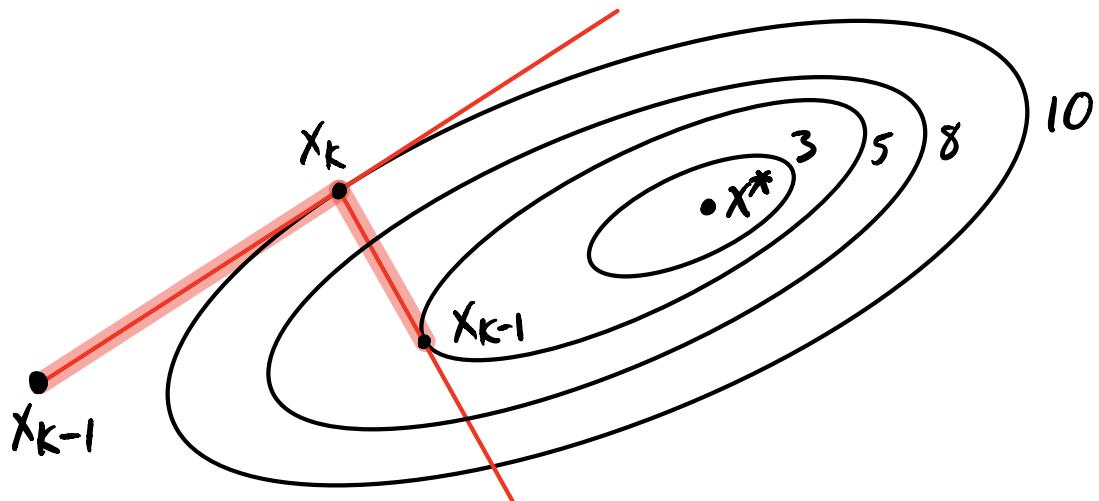
$$x_{k+1} = x_k - s_k \nabla f(x_k)$$

The step-size s_k can be found with an exact line search,

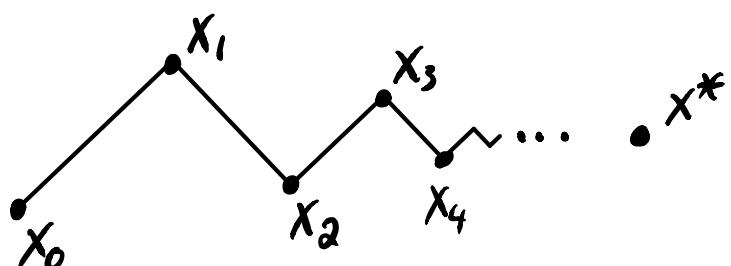
$$s_k = \underset{s \geq 0}{\operatorname{argmin}} f(x_k - s \nabla f(x_k))$$

or a backtracking line search that guarantees sufficient decrease.

Using the exact line search, our step directions will be orthogonal since $d_k = -\nabla f(x_k)$ is orthogonal to the current level curve $f(x) = f(x_k)$ and tangent to the next level curve $f(x) = f(x_{k+1})$.



We zig-zag toward the sol'n x^* .



The rate of convergence can be very slow.

Suppose $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is convex
and the eigenvalues of its
Hessian matrix are bounded below
by $m > 0$ and above by $M \geq m$:

$$\left[\begin{array}{l} m \leq \lambda \leq M \quad \text{for all eigenvalues} \\ \lambda \text{ of } D^2f(x), \text{ for all } x. \end{array} \right]$$

Then we will reduce the distance
between the objective value and the
optimal value by at least a factor
of $(1 - \frac{m}{M})$:

$$f(x_{k+1}) - f(x^*) \leq \left(1 - \frac{m}{M}\right) (f(x_k) - f(x^*)).$$

This is a linear rate of convergence:

$$f(x_k) - f(x^*) \leq \left(1 - \frac{m}{M}\right)^k (f(x_0) - f(x^*)).$$

If m/M is close to zero, it can take a huge number of iterations to achieve a 99% reduction in the distance.

m/M	k
0.9	2
0.1	44
0.01	459

Backtracking line search also gives a linear convergence rate, but with a larger reduction factor.

Momentum and the Path of a Heavy Ball

$$x_{k+1} = x_k - s z_k$$

$$z_k = \nabla f(x_k) + \beta z_{k-1}$$

Add momentum
to the gradient.

How to choose s and β ?

Consider $f(x) = \frac{1}{2}x^T S x$ with S positive definite and let

λ_{\min} = smallest eigenvalue of S ,

λ_{\max} = largest eigenvalue of S .

The optimal s and β are

$$s = \left(\frac{2}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}} \right)^2, \quad \beta = \left(\frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}} \right)^2.$$

Example:

$$f(x, y) = \frac{1}{2}(x^2 + b y^2), \quad 0 < b \leq 1.$$

$$(x_0, y_0) = (b, 1), \quad (x^*, y^*) = (0, 0)$$

Gradient descent w/ exact line search:

$$f(x_k, y_k) = \left(\frac{1-b}{1+b} \right)^{2k} f(x_0, y_0).$$

Heavy ball:

$$f(x_k, y_k) = \left(\frac{1-\sqrt{b}}{1+\sqrt{b}} \right)^{2k} f(x_0, y_0).$$

If $b = 0.01$, then

$$\left(\frac{1-b}{1+b}\right)^2 = 0.96 \text{ and } \left(\frac{1-\sqrt{b}}{1+\sqrt{b}}\right)^2 = 0.67.$$

Thus, one step of heavy ball is the same as ten steps of gradient descent.

Nesterov Acceleration

$$x_{k+1} = y_k - s \nabla f(y_k)$$

$$y_{k+1} = x_{k+1} + \beta(x_{k+1} - x_k)$$

Evaluate ∇f at a shifted point.

For $f(x) = \frac{1}{2}x^T S x$, with S pos. def.,

we use

$$s = \frac{1}{\lambda_{\max}}, \quad \beta = \frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}}.$$

Adaptive methods (e.g ADAM) use all earlier points x_0, \dots, x_k to compute x_{k+1} .

Summary:

All three methods can be written as:

$$x_{k+1} = x_k + \beta(x_k - x_{k-1}) - s \nabla f(x_k + \gamma(x_k - x_{k-1}))$$

Gradient descent: $s \neq 0, \beta = \gamma = 0$

Heavy ball: $s, \beta \neq 0, \gamma = 0$

Nesterov: $s, \beta, \gamma \neq 0$

Constraints and Proximal Methods

$$\min f(x) \text{ subject to } x \in K$$

Projection onto K :

$$\text{proj}_K(x) = \underset{z \in K}{\operatorname{arg\,min}} \frac{1}{2} \|x - z\|_2^2$$

Projected gradient descent:

$$x_{k+1} = \text{proj}_K(x_k - s_k \nabla f(x_k))$$

A generalization:

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

$f: \mathbb{R}^n \rightarrow \mathbb{R}$ convex and smooth

$g: \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ convex and nonsmooth

If $g(x) = \begin{cases} 0 & \text{if } x \in K \\ +\infty & \text{if } x \notin K \end{cases}$

Then $\min_x f(x) + g(x) = \min_{x \in K} f(x).$

Another example is LASSO regression:

$$\min_x \frac{1}{2} \|b - Ax\|_2^2 + \lambda \|x\|_1.$$

Here $f(x) = \frac{1}{2} \|b - Ax\|_2^2$ and $g(x) = \lambda \|x\|_1.$

Proximal mapping:

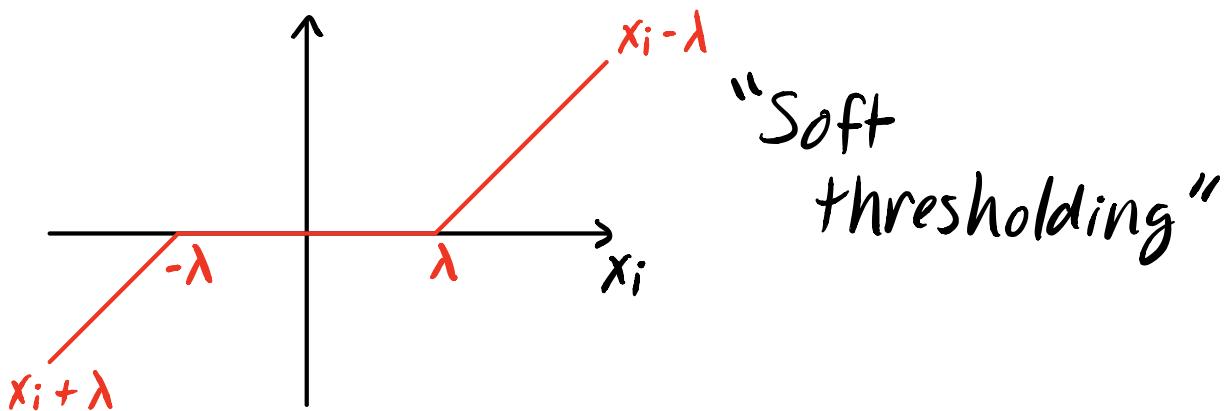
$$\text{prox}_g(x) = \underset{z \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} \|x - z\|_2^2 + g(z)$$

$$\text{If } g(x) = \begin{cases} 0, & x \in K \\ +\infty, & x \notin K \end{cases}$$

Then $\text{prox}_g(x) = \text{proj}_K(x)$.

If $g(x) = \lambda \|x\|_1$, then

$$\text{prox}_g(x)_i = \text{sign}(x_i) \cdot \max(|x_i| - \lambda, 0).$$



Proximal gradient method:

$$x_{k+1} = \text{prox}_{s_k g}(x_k - s_k \nabla f(x_k))$$