Lecture 3: Local Descent

Optimization T2023

Màster de Fonaments de Ciència de Dades

UNIVERSITAT DE BARCELONA

$f(\mathbf{x}) \to min, \quad \mathbf{x} \in D \subseteq \mathbb{R}^n, \quad n \geq 1, \quad f$ is smooth

Goal: Iteratively find a sequence $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots \to \mathbf{x}^*$, where $\mathbf{x}^*$ is a solution of the optimization problem (local or global minimum), realizing the descent

$$f\left(\mathbf{x}^{(1)}\right) > f\left(\mathbf{x}^{(2)}\right) > \cdots$$

(for all or most* of the iterates)

Recall that $\nabla f(\mathbf{x}^*) = 0$

General descent method.

given a starting point $\mathbf{x}^{(1)} \in D$
repeat

1. Determine *descent direction* $\boldsymbol{p}^{(k)}$ (often, $\left\|\boldsymbol{p}^{(k)}\right\| = 1$)
2. Determine *step size/learning rate* $\alpha^{(k)}$
3. Update $\mathbf{x}^{(k+1)} := \mathbf{x}^{(k)} + \alpha^{(k)}\boldsymbol{p}^{(k)}$

until *stopping criterion* is satisfied

III. Descent direction?

II. Step size?

I. Stopping criterion?

Digression: Why gradient?

Recall that from the Taylor formula

$$f(\mathbf{x}+\mathbf{v}) =_{\color{magenta}{up\ to\ H.O.T.}} f(\mathbf{x}) + \mathbf{v}^T \cdot \nabla f(\mathbf{x})$$
$$= f(\mathbf{x}) + \nabla_{\mathbf{v}} f(\mathbf{x})$$

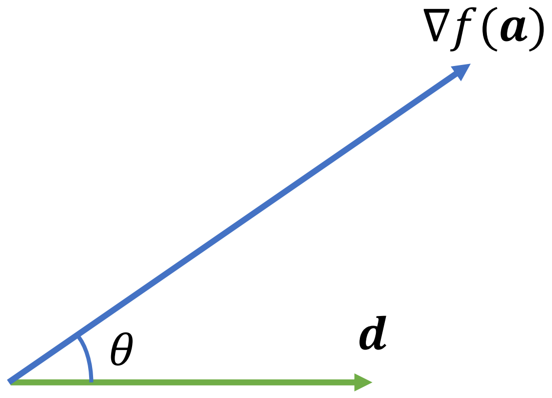
Directional derivative → We want it to be negative

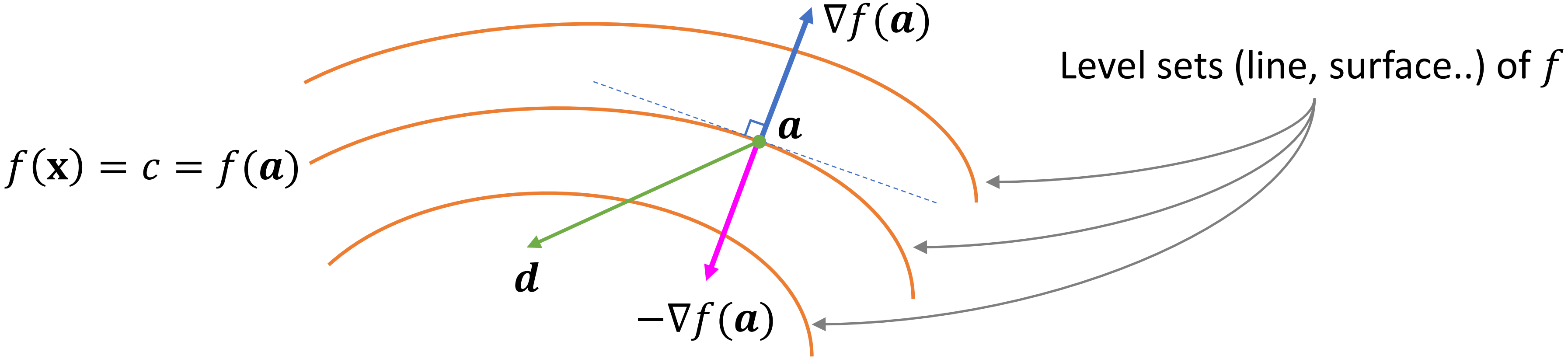
Theorem:

Let $f: D \subseteq \mathbb{R}^n \to \mathbb{R}$ be a differentiable function, $\boldsymbol{a} \in D$, $\boldsymbol{d} \in \mathbb{R}^n$ with $\|\boldsymbol{d}\| = 1$. If θ is the angle between $\boldsymbol{d}$ and $\nabla f(\boldsymbol{a})$. Then

$$\nabla_{\boldsymbol{d}} f(\boldsymbol{a}) = \boldsymbol{d}^T \cdot \nabla f(\boldsymbol{a}) = \|\nabla f(\boldsymbol{a})\| \cos\theta$$

In particular, the vector $-\nabla f(\boldsymbol{a})$ gives the maximum descent direction of f at the point $\boldsymbol{a}$.





I. Stopping criteria/termination conditions

- **Maximum iterations**: repeat until $k \leq k_{max}$
- **Absolute improvement**: repeat until

$$f\left(\mathbf{x}^{(k)}\right) - f\left(\mathbf{x}^{(k+1)}\right) < \epsilon_a$$

- **Relative improvement**: repeat until

$$f\left(\mathbf{x}^{(k)}\right) - f\left(\mathbf{x}^{(k+1)}\right) < \epsilon_r \left| f\left(\mathbf{x}^{(k)}\right) \right|$$

- **Gradient magnitude**: repeat until

$$\left\| \nabla f\left(\mathbf{x}^{(k+1)}\right) \right\| < \epsilon_g$$

✓ One or more termination conditions can be used

✓ If there are several local minima, one can add ***random restart*** with $\mathbf{x}^{(1),new}$ sampled randomly from D

II. Step size/learning rate

Suppose $x = x^{(k)}$ and $p = p^{(k)}$ is given. How to find $\alpha = \alpha^{(k)}$?

Methods:

1. Exact line search

2. Approximate line search

3. Trust region methods

Exact line search

$\text{minimize}_{\alpha}\ f(\mathbf{x}+\alpha \boldsymbol{p})$

Definition: A *bracket* is an interval $[\alpha', \alpha"]$ containing α^*

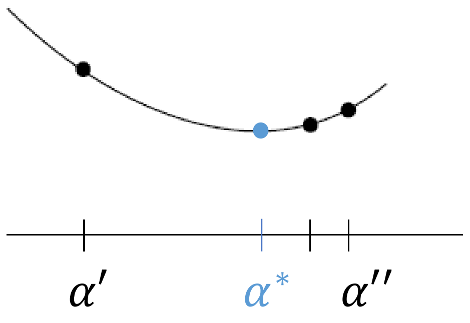
- This is univariant optimization problem for $\phi(\alpha) := f(\mathbf{x}+\alpha \boldsymbol{p})$ $\rightarrow$

  → Find a **bracket** for the optimal solution α^*

  (α^* is characterized by $\phi(\alpha^*) < \phi(\alpha)$ for all α near α^*)

  → Use univariant optimization methods to find an approximate of α^* by successively shrinking the bracket. Methods include:

  - Dyadic/binary search
  - Fibonacci search
  - Quadratic fit search
  - Shubert–Piyavskii method
  - Bisection method

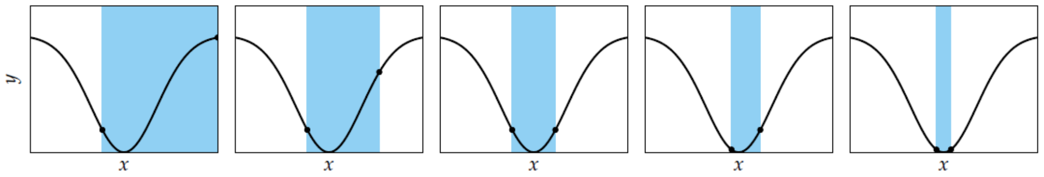


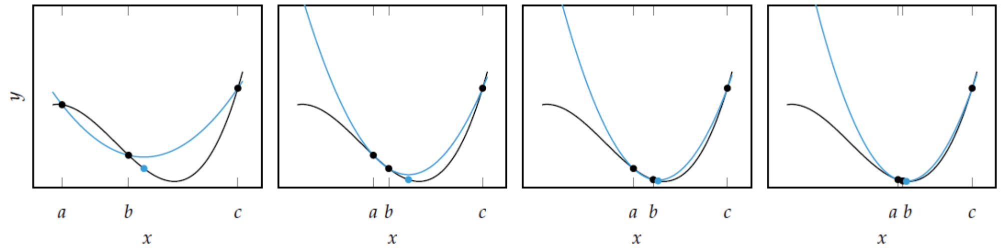
Digression: some univalent optimization methods
[KW, Ch.3]

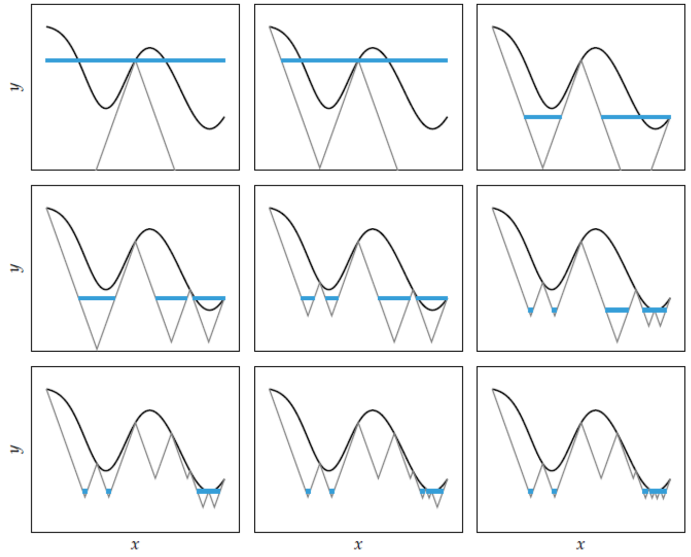
- **Dyadic/binary search**: subdivide interval 'in half' at each step
- **Fibonacci search :** max reduction of interval size for given number of function evaluations
- **Quadratic fit search**
- **Shubert-Piyavskii method** : assuming ϕ is Lipshitz, e.g.

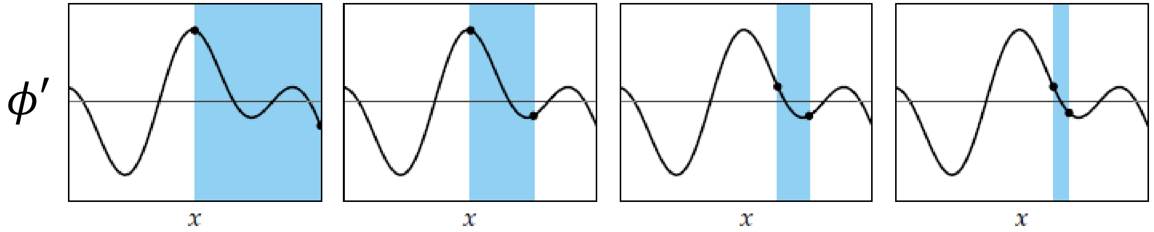
$$|\phi(x) - \phi(y)| \leq \ell \cdot |x - y|, \quad \forall x, y \in [\alpha', \alpha'']$$

- **Bisection method**: solve $\phi'(\alpha) = 0$ instead

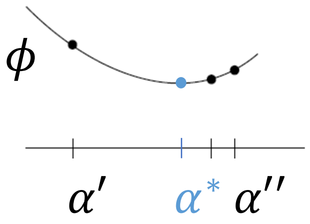








Dyadic/binary and Fibonacci search

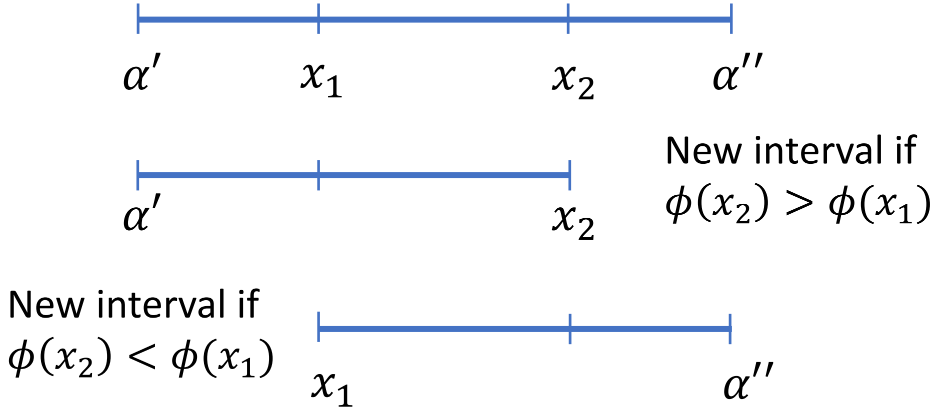


Assumption ★:
ϕ is unimodal, that is, ϕ has a unique minimum on (α', α'')

$\Longrightarrow$ ϕ is decreasing on $[\alpha', \alpha^*]$ and ϕ is increasing on $[\alpha^*, \alpha'']$

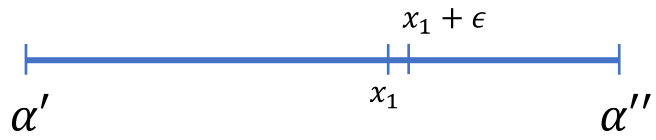
$\Longrightarrow$ ϕ is convex on $[\alpha', \alpha'']$ $(\Leftrightarrow \phi'' > 0)$

Basic Splitting Step: for a pair $x_1 < x_2$ of points in the starting bracket



Exercise: Check that, under Assumption ★, for all $x_1 < x_2$ after the Basic Splitting Step the new interval contains α^* (hence, is a bracket).

Basic Splitting Step in "almost" two parts: do the basic splitting step for x_1 and $x_1 + \epsilon$, where $\epsilon > 0$ is small



- Each Basic Splitting Step requires 2 evaluations of the function at x_1 and x_2.
- In general, i.e., if Assumption ★ is violated, the Basic Splitting Step doesn't work!

Exercise: Give an example

Dyadic/binary search:
under Assumption ★

given the desired size $\epsilon > 0$ of the bracket

choose $\delta < \epsilon$ (usually much smaller)

repeat

1. Pick the midpoint $x_1 = \frac{\alpha' + \alpha''}{2}$
2. Do the **Basic Splitting Step in 'almost' two parts** using x_1 and $x_1 + \delta$
3. Update $[\alpha', \alpha'']$ with the new bracket from step 2 above

until $|\alpha'' - \alpha'| < \epsilon$

Exercise: How many evaluations of the function ϕ is required in the dyadic search in order to shrink the bracket by a factor of 100?

Fibonacci search (under Assumption ★)

- Fibonacci numbers are given by the recursive relation $F_{n+2} = F_{n+1} + F_n$, with starting condition $F_1 = F_2 = 1$.
- This generates the sequence $1, 1, 2, 3, 5, 8, 11, \dots$
- This sequence grows as $F_n \sim \frac{1}{\sqrt{5}} \varphi^n$, for n large, where $\varphi = \frac{1+\sqrt{5}}{2} \approx 1.6$ is the Golden ratio.

---

given the number of steps N

for $i = N, N-1, \dots, 1$ **do**

if $i \neq 1$,

1. Compute $x_1, x_2 \in [\alpha', \alpha'']$ such that
$$\frac{\alpha'' - x_1}{\alpha'' - \alpha'} = \frac{F_i}{F_{i+1}} \text{ and } \frac{x_2 - \alpha'}{\alpha'' - \alpha'} = \frac{F_i}{F_{i+1}}$$
2. Do the **Basic Splitting Step** using x_1 and x_2
3. Update $[\alpha', \alpha'']$ with the new bracket from step 2 above

Observe that after this step, the length of the new bracket is proportional to the length of the previous bracket as F_i to F_{i+1}

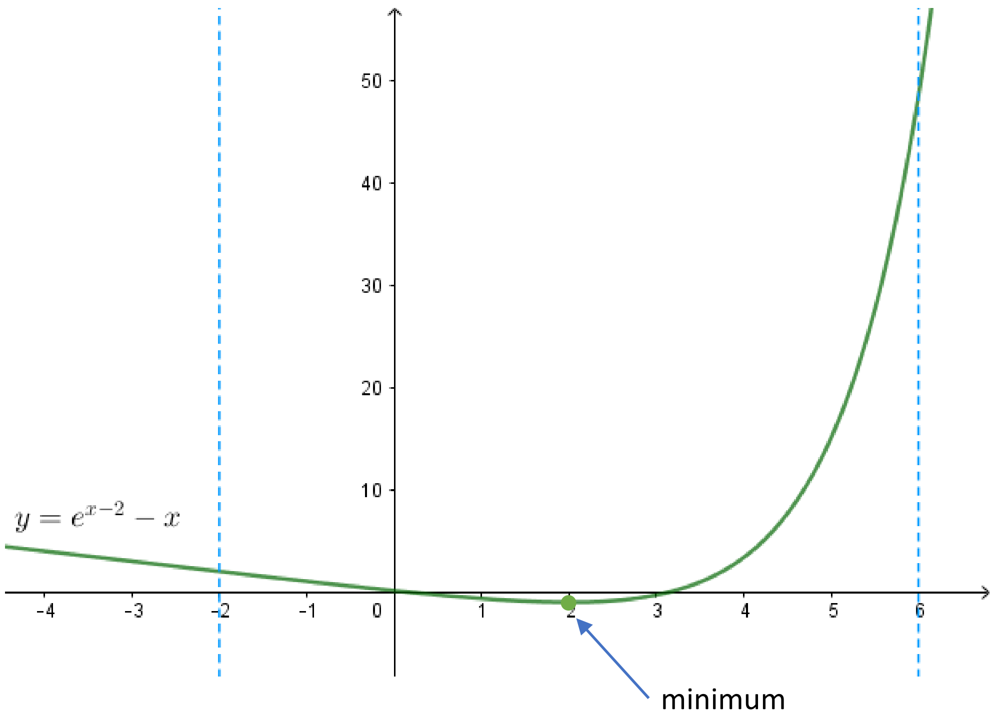
otherwise

Do the **Basic Splitting Step in 'almost' two parts** using $\frac{\alpha' + \alpha''}{2}$ and $\frac{\alpha' + \alpha''}{2} + \epsilon$

Key Advantage: Fibonacci search uses significantly smaller evaluations of the function than the dyadic search because it re-uses some evaluation points! (see example on the next slide)

Exercise: How many evaluations of the function ϕ is required in the Fibonacci search in order to shrink the bracket by a factor of 100? Compare it to the corresponding result of the dyadic search.

Fibonacci search (under Assumption ★): an example



Consider using Fibonacci search with five function evaluations to minimize $f(x) = \exp(x-2) - x$ over the interval $[a, b] = [-2, 6]$. The first two function evaluations are made at $\frac{F_5}{F_6}$ and $1 - \frac{F_5}{F_6}$, along the length of the initial bracketing interval:

$$f(x^{(1)}) = f\left(a + (b-a)\left(1 - \frac{F_5}{F_6}\right)\right) \qquad = f(1) = -0.632$$
$$f(x^{(2)}) = f\left(a + (b-a)\frac{F_5}{F_6}\right) \qquad = f(3) = -0.282$$

The evaluation at $x^{(1)}$ is lower, yielding the new interval $[a, b] = [-2, 3]$. Two evaluations are needed for the next interval split:

$$x_{\text{left}} = a + (b-a)\left(1 - \frac{F_4}{F_5}\right) = 0$$
$$x_{\text{right}} = a + (b-a)\frac{F_4}{F_5} = 1$$

A third function evaluation is thus made at x_{left}, as x_{right} has already been evaluated:

$$f(x^{(3)}) = f(0) = 0.135$$

The evaluation at $x^{(1)}$ is lower, yielding the new interval $[a, b] = [0, 3]$. Two evaluations are needed for the next interval split:

$$x_{\text{left}} = a + (b-a)\left(1 - \frac{F_3}{F_4}\right) = 1$$
$$x_{\text{right}} = a + (b-a)\frac{F_3}{F_4} = 2$$

A fourth functional evaluation is thus made at x_{right}, as x_{left} has already been evaluated:

$$f(x^{(4)}) = f(2) = -1$$

The new interval is $[a, b] = [1, 3]$. A final evaluation is made just next to the center of the interval at $2 + \epsilon$, and it is found to have a slightly higher value than $f(2)$. The final interval is $[1, 2 + \epsilon]$.

Quadratic fit search

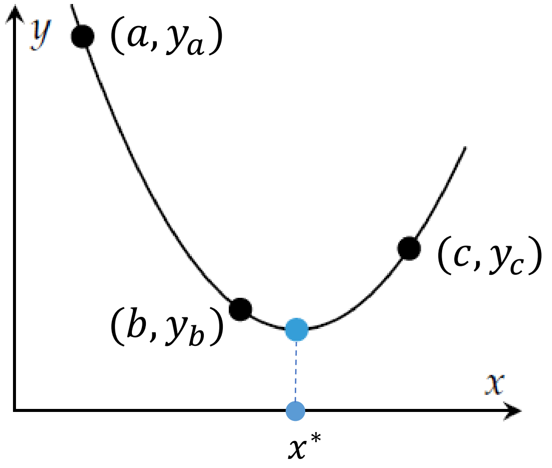
The method is based on the following observations:

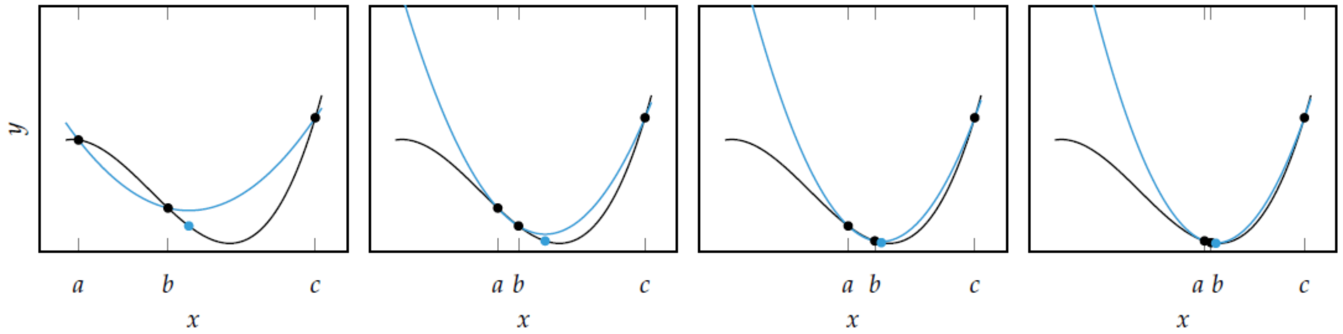
- 'close' to the minima functions look like quadratic functions
- we can explicitly find minima of quadratic functions:

Lemma:

There exists a unique parabola that passes through any triple of distinct points $(a, y_a), (b, y_b), (c, y_c)$. This parabola has its extremum at

$$x^* = \frac{1}{2} \frac{y_a(b^2 - c^2) + y_b(c^2 - a^2) + y_c(a^2 - b^2)}{y_a(b - c) + y_b(c - a) + y_c(a - b)}$$





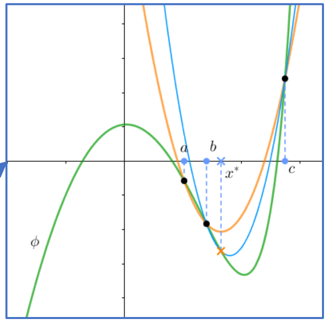
Exercise*: Show that the algorithm described on the next slide converges to a local minimum (assuming the function is smooth)

Quadratic fit search

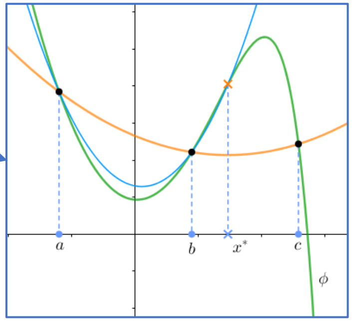
given a triple $a < b < c$ where $[a, c]$ is a bracket of ϕ and $\overbrace{\phi(b) < \phi(a),\ \phi(b) < \phi(c)}^{\#}$
repeat

1. Compute the critical point x^* of the parabola passing through a, b, c
2. If $x^* \in [b, c]$, then
   - Check which value is larger, $\phi(x^*)$ or $\phi(b)$:
     - i. If $\phi(b) > \phi(x^*)$, update the triple (a, b, c) with (b, x^*, c)
     - ii. If $\phi(b) < \phi(x^*)$, update the triple (a, b, c) with (a, b, x^*)
3. Otherwise
   - Again, check which value is larger, $\phi(x^*)$ or $\phi(b)$:
     - i. If $\phi(b) > \phi(x^*)$, update the triple (a, b, c) with (a, x^*, b)
     - ii. If $\phi(b) < \phi(x^*)$, update the triple (a, b, c) with (x^*, b, c)

until the $|a - c| < \epsilon$

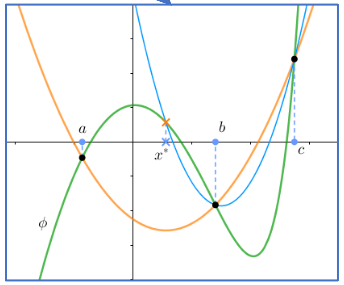


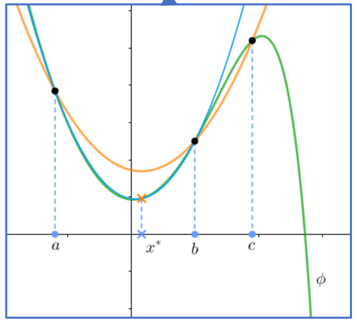
In these examples, parabola at the current step is in orange; parabola at the next step is in blue



That is, when $x^* \in [a, b)$ because of condition #

Or any other stopping criterion based on variation of the function





Bisection method

The method is based on the following observations:

- Instead of looking for a local minimum of ϕ, we can look for a solution of $\phi' = 0$
- We assume that $[\alpha', \alpha'']$ is a bracket for ϕ, and hence there exists a solution of $\phi' = 0$ on this interval

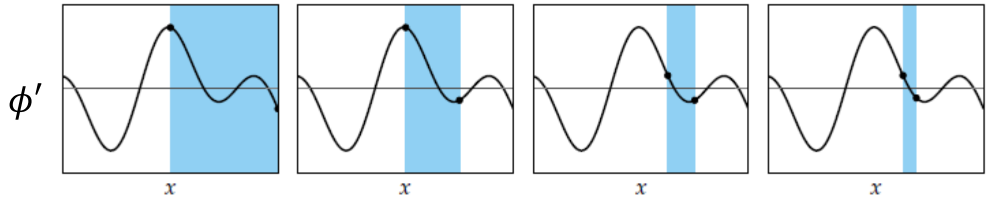
---

given an interval $[a, b]$ such that $\phi'(a) \cdot \phi'(b) < 0$

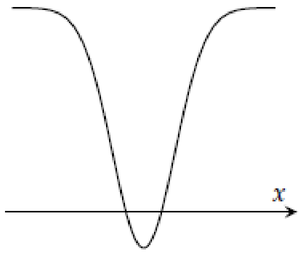
repeat

1. Compute the midpoint $c = \frac{a+b}{2}$
2. If $\phi'(a) \cdot \phi'(c) < 0$, update interval $[a, b]$ with $[a, c]$
3. If $\phi'(b) \cdot \phi'(c) < 0$, update interval $[a, b]$ with $[c, b]$

until $|a - b| < \epsilon$



- If $[\alpha', \alpha'']$ doesn't satisfy the condition $\phi'(\alpha') \cdot \phi'(\alpha'') < 0$, then one can try iteratively shrink this interval by a constant factor (say 2), until the condition is fulfilled. However, it might not always work (see an example of the function on the left where the bisection method can fail; this is the situation of a local minimum in a 'deep valley'). More sophisticated methods should be used instead.



Exercise: Let $\phi(x) = \frac{x^2}{2} - x$. Apply the bisection method to find an interval containing the minimizer of ϕ starting with the interval $[0,1000]$. Execute 3 steps of the algorithm.

End of digression

Exact line search

$$\text{minimize}_{\alpha} \; f(\mathbf{x} + \alpha \boldsymbol{p})$$

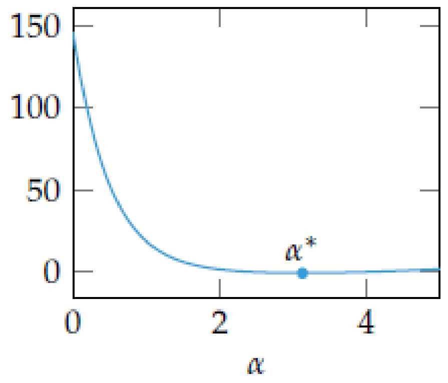
Consider conducting a line search on $f(x_1, x_2, x_3) = \sin(x_1 x_2) + \exp(x_2 + x_3) - x_3$ from $\mathbf{x} = [1, 2, 3]$ in the direction $\mathbf{d} = [0, -1, -1]$. The corresponding optimization problem is:

$$\underset{\alpha}{\text{minimize}} \sin((1 + 0\alpha)(2 - \alpha)) + \exp((2 - \alpha) + (3 - \alpha)) - (3 - \alpha)$$

which simplifies to:

$$\underset{\alpha}{\text{minimize}} \sin(2 - \alpha) + \exp(5 - 2\alpha) + \alpha - 3$$

The minimum is at $\alpha \approx 3.127$ with $\mathbf{x} \approx [1, -1.126, -0.126]$.



Approximate line search

Find $\alpha^{(k)}$ approximately and move on with the descent method

For simplicity, $x_k = \mathbf{x}^{(k)}, p_k = \boldsymbol{p}^{(k)}, \alpha_k = \alpha^{(k)}$

We impose the following condition for α_k

$$\phi(\alpha_k) := f(x_k + \alpha_k p_k) < f(x_k) + c_1 \alpha_k (\nabla f(x_k))^T p_k, \quad c_1 \in (0,1).$$

The condition is called (sufficient decrease condition).

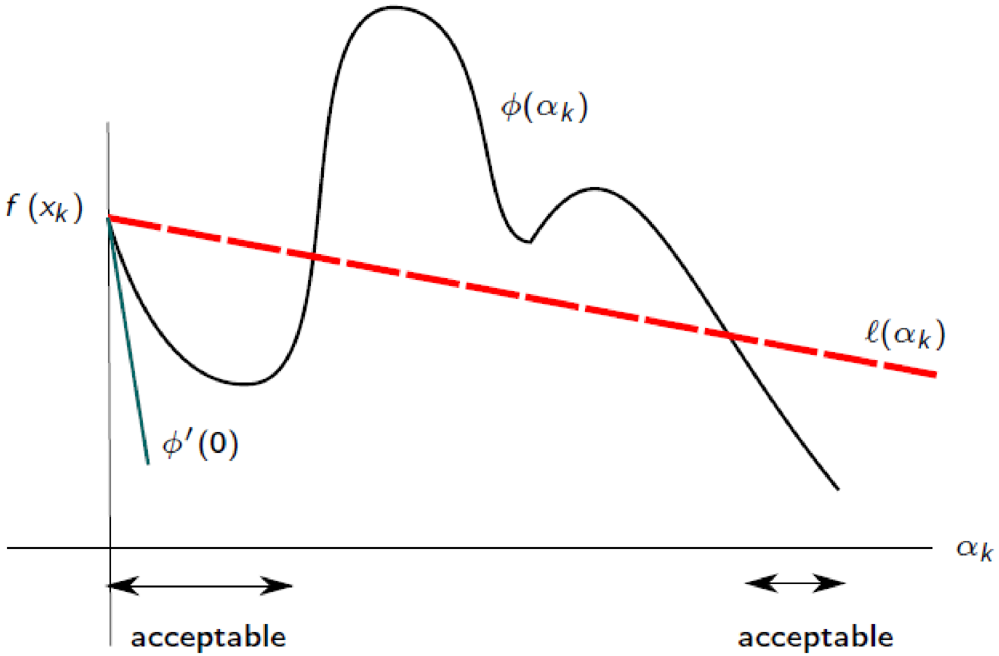
Remarks.

- $\ell(\alpha_k) := f(x_k) + c_1 \alpha_k \nabla f^T(x_k) p_k$ is a linear function.
- For small values of $\alpha_k > 0$ we have $\phi(\alpha_k) < \ell(\alpha_k)$. This is so because $c_1 \in (0,1)$ and then

$$\phi'(0) = (\nabla f(x_k))^T p_k < c_1 (\nabla f(x_k))^T p_k = \ell'(0) < 0.$$

Approximate line search

Sufficient decrease. We ask for a decrease proportional to α and $\phi'(0) = (\nabla f(x_k))^T p_k$. Usually $c_1 \approx 0.1$.

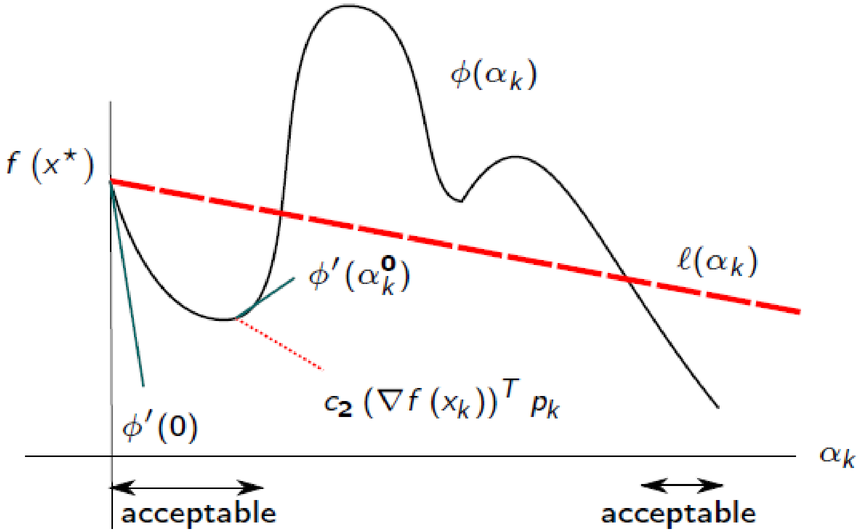


Approximate line search

Curvature condition. Since the previous condition is always satisfied for small values of α_k we need to add further conditions for termination. We use the so called curvature condition

$$(\nabla f(x_k + \alpha_k p_k))^T p_k \geq c_2 (\nabla f(x_k))^T p_k, \ \ c_2 \in (c_1, 1)$$

In other words if $\phi'(\alpha_k)$ is not negative enough we terminate the k-step.



Approximate line search

Wolfe conditions

Definition. The conditions (together) to terminate the k-step given by

$$f(x_k + \alpha_k p_k) < f(x_k) + c_1 \alpha_k (\nabla f(x_k))^T p_k,$$
$$(\nabla f(x_k + \alpha_k p_k))^T p_k \geq c_2 (\nabla f(x_k))^T p_k,$$

with $0 < c_1 < c_2 < 1$ are usually called Wolfe conditions.

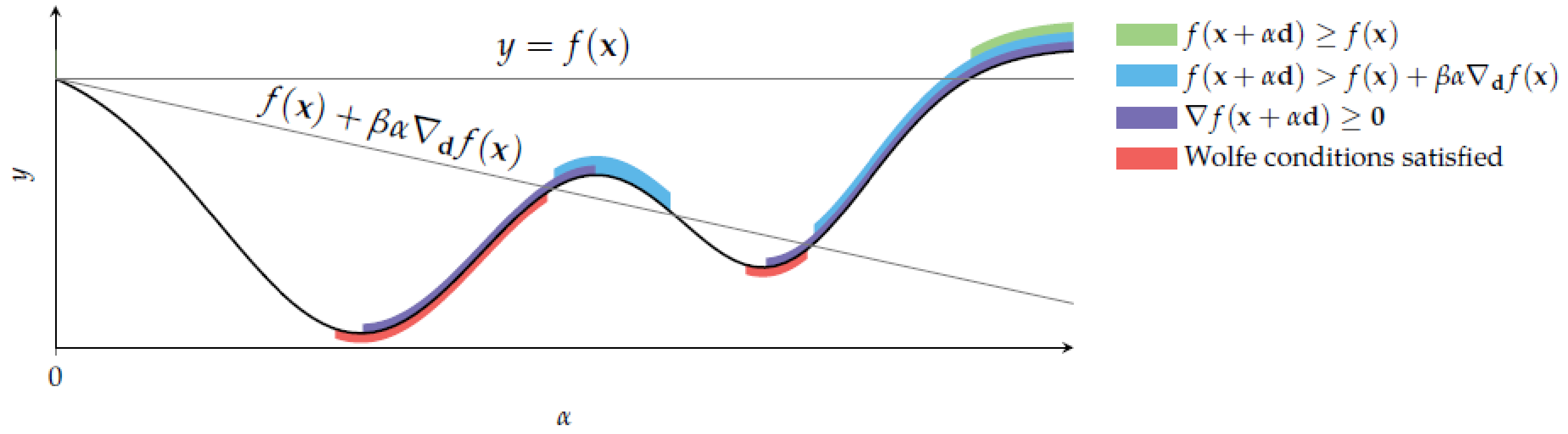
Definition. The conditions (together) to terminate the k-step given by (we do not allow $\phi'(\alpha_k)$ to be too positive).

$$f(x_k + \alpha_k p_k) < f(x_k) + c_1 \alpha_k (\nabla f(x_k))^T p_k,$$
$$|(\nabla f(x_k + \alpha_k p_k))^T p_k| \leq |c_2 (\nabla f(x_k))^T p_k|,$$

with $0 < c_1 < c_2 < 1$ are usually called strong Wolfe conditions.

Approximate line search

Wolfe conditions



Wolfe conditions: existence

Lemma. Suppose $f : D \subset \mathbb{R}^n \to \mathbb{R}$ be a $\mathcal{C}^1$ function. Let p_k a descent direction at the point $x_k \in D$ and assume $f|L_{p_k}$ is bounded below where $L_{p_k} = \{x \in \mathbb{R}^n \mid x = x_k + \alpha p_k,\ \alpha > 0\}$. Then if $0 < c_1 < c_2 < 1$ there exist intervals of step lengths satisfying the (strong) Wolfe conditions

Proof. Since $\ell'(\alpha_k) < 0$ (and constant) there exists a first intersection, $\hat{\alpha}_k > 0$, between $\ell(\alpha_k)$ and $\phi(\alpha_k)$:

$$f(x_k + \hat{\alpha}_k p_k) = f(x_k) + c_1 \hat{\alpha}_k (\nabla f(x_k))^T p_k. \tag{1}$$

The sufficient decrease condition it is satisfied for all $\alpha_k \in [0, \hat{\alpha}_k]$. By the Mean Value Theorem we have that there exists $\tilde{\alpha}_k \in [0, \hat{\alpha}_k]$ such that

$$f(x_k + \hat{\alpha}_k p_k) - f(x_k) = \hat{\alpha}_k (\nabla f(x_k + \tilde{\alpha}_k p_k))^T p_k$$

All together imply

$$(\nabla f(x_k + \tilde{\alpha}_k p_k))^T p_k = c_1 \hat{\alpha}_k (\nabla f(x_k))^T p_k > c_2 \hat{\alpha}_k (\nabla f(x_k))^T p_k.$$

Therefore $\tilde{\alpha}_k$ satisfies the Wolfe conditions and smoothness gives the desired interval.

Approximate line search

Convergence

Remark. Until this moment we just consider the definition of the process, that is the election of p_k and α_k. But we need to study if the process converge to somewhere.

Let p_k be a descent direction, and let θ_k the angle of p_k and $-\nabla f(x^\star)$

$$\cos(\theta_k) = -\frac{1}{||\nabla f(x_k)||\ ||p_k||} (\nabla f(x_k))^T p_k$$

Theorem. Assume notation above with p_k a descent direction and α_k satisfying Wolfe's conditions. Suppose f is $\mathcal{C}^2$ and bounded below in $\mathbb{R}^n$. Then

$$\sum_{k=0}^{\infty} \cos^2(\theta_k)||\nabla f(x_k)|| < \infty. \tag{2}$$

Corollary. Under the above notation and assumptions we have

$$\cos^2(\theta_k)||\nabla f(x_k)|| \to 0$$

Moreover if there exists $\delta > 0$ such that $\cos(\theta) > \delta$ then

$$\lim_{k\to\infty} ||\nabla f(x_k)|| = 0 \quad \text{(globally convergent algorithms)}$$

Remark. The final δ-condition basically means that p_k do not get arbitrarily orthogonal to the gradient vector. This is, for instance, the case of the steepest descent method.