# Modern Optimization Techniques

## 2. Unconstrained Optimization / 2.1. Gradient Descent

Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL)
Institute for Computer Science
University of Hildesheim, Germany

# Syllabus

| | | |
|---|---|---|
| Mon. 07.11. | (1) | 0. Overview |
| | | **1. Theory** |
| Mon. 14.11. | (2) | 1. Convex Sets and Functions |
| | | **2. Unconstrained Optimization** |
| Mon. 21.11. | (3) | 2.1 Gradient Descent |
| Mon. 28.11. | (4) | 2.2 Stochastic Gradient Descent |
| Mon. 05.12. | (5) | 2.3 Newton's Method |
| Mon. 12.12. | (6) | 2.4 Quasi-Newton Methods |
| Mon. 19.12. | (7) | 2.5 Subgradient Methods |
| | — | — *Christmas Break* — |
| Mon. 09.01. | (8) | 2.6 Coordinate Descent |
| | | **3. Equality Constrained Optimization** |
| Mon. 16.01. | (9) | 3.1 Duality |
| Mon. 23.01. | (10) | 3.2 Methods |
| | | **4. Inequality Constrained Optimization** |
| Mon. 30.01. | (11) | 4.1 Primal Methods |
| Mon. 06.01. | (12) | 4.2 Barrier and Penalty Methods |
| Mon. 13.02. | (13) | 4.3 Cutting Plane Methods |

# Outline

# Outline

# Unconstrained Convex Optimization Problem

$$\arg\min_{x \in X} \; f(\mathbf{x})$$

where

- $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ is
  - convex
  - twice continuously differentiable
  - esp. $\operatorname{dom} f = X = \mathbb{R}^N$ or convex and open.
- An optimal $\mathbf{x}^*$ exists and $p^* := f(\mathbf{x}^*)$ is finite

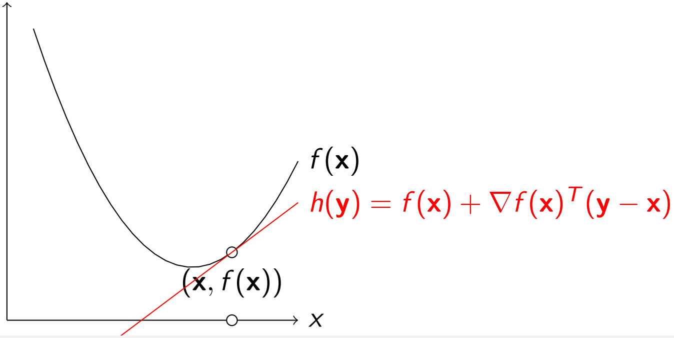
# Reminder: 1st-order condition

**1st-order condition:** a differentiable function $f$ is convex iff

- dom $f$ is a convex set
- for all $\mathbf{x}, \mathbf{y} \in \operatorname{dom} f$

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})$$

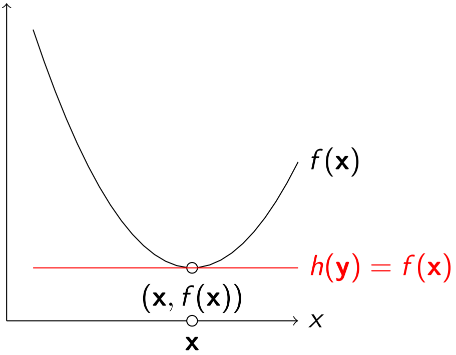
(the function is above any of its tangents.)



# Minimality Condition

$\mathbf{x}$ is minimal iff

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




Note: Often also called **optimality condition**.

# Outline

# Iteative Methods

- Start with an initial (random) point: $\mathbf{x}^{(0)}$
- Generate a sequence of points: $\mathbf{x}^{(k)}$ with

$$f(\mathbf{x}^{(k)}) \to f(\mathbf{x}^*)$$

```
min-unconstrained(f, x^(0)):
  k := 0
  repeat
    x^(k+1) := next-point(f, x^(k))
    k := k + 1
  until converged(x^(k), x^(k-1), f)
  return x^(k), f(x^(k))
```

# Iteative Methods

- Start with an initial (random) point: $\mathbf{x}^{(0)}$
- Generate a sequence of points: $\mathbf{x}^{(k)}$ with

$$f(\mathbf{x}^{(k)}) \rightarrow f(\mathbf{x}^*)$$

```
min-unconstrained(f, x^(0), K):
  for k := 0 : K − 1:
    x^(k+1) := next-point(f, x^(k))
    if converged(x^(k+1), x^(k), f):
      return x^(k+1), f(x^(k+1))
  raise exception "not converged in K iterations"
```

# Convergence Criterion

$$\textbf{converged}(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}, f)$$

- Different criteria in use
  - different optimization methods may use different criteria.
- One would like to use the **optimality gap**:

$$||\mathbf{x}^{(k+1)} - \mathbf{x}^\star||_2^2 < \epsilon$$

  - not possible as $\mathbf{x}^\star$ is unknown
- **Minimum progress/change $\epsilon$ in $x$ in last iteration**:

$$\textbf{converged}(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}, f) := ||\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}||_2^2 < \epsilon$$

  - cheap to compute.
  - can be used with any method.
  - requires parameter $\epsilon \in \mathbb{R}^+$.
  - may stop too early when the loss surface is too flat.

# Descent Methods

- a class/template of methods
- the next point is generated as:

$$\mathbf{x}^{(k+1)} := \mathbf{x}^{(k)} + \mu \Delta \mathbf{x}^{(k)}$$

  with
  - a **search direction** $\Delta \mathbf{x}^{(k)}$ and
  - a **step size** $\mu > 0$ such that

$$f(\mathbf{x}^{(k)} + \mu \Delta \mathbf{x}^{(k)}) < f(\mathbf{x}^{(k)})$$

    - always exists if the step size $\mu$ is sufficient small
      if the search direction $\Delta \mathbf{x}^{(k)}$ is a **descent direction**:

$$\nabla f(\mathbf{x}^{(k)})^T \Delta \mathbf{x}^{(k)} < 0$$

- search directions $\Delta \mathbf{x}^{(k)}$ can be computed different ways
  - Gradient Descent
  - Steepest Descent
  - Newton's Method

# Descent Methods

**min-descent**$(f, \mathbf{x}^{(0)}, K)$:
  for $k := 0 : K-1$:
    $\Delta\mathbf{x}^{(k)} :=$ **search-direction**$(f, \mathbf{x}^{(k)})$
    $\mu^{(k)} :=$ **step-size**$(f, \mathbf{x}^{(k)}, \Delta\mathbf{x}^{(k)})$
    $\mathbf{x}^{(k+1)} := \mathbf{x}^{(k)} + \mu^{(k)}\Delta\mathbf{x}^{(k)}$
    if **converged**$(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}, f)$:
      return $\mathbf{x}^{(k+1)}$, $f(\mathbf{x}^{(k+1)})$
  raise exception "not converged in $K$ iterations"

# Outline

# Gradient Descent

- The gradient of a function $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ at $\mathbf{x}$ yields the direction in which the function is maximally growing locally.

- Gradient Descent is a descent method that searches in the opposite direction of the gradient:

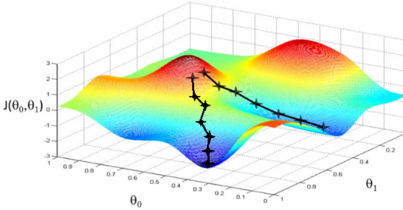
$$\Delta \mathbf{x} := -\nabla f(\mathbf{x})$$

- Gradient:

$$\nabla f(\mathbf{x}) := \nabla_x f(\mathbf{x}) := (\frac{\partial f}{\partial x_n}(\mathbf{x}))_{n=1:N}$$

# Gradient Descent

$$
\begin{array}{l}
\textbf{min-GD}(f, \mathbf{x}^{(0)}, K): \\
\quad \text{for } k := 0 : K-1: \\
\qquad \Delta\mathbf{x}^{(k)} := -\nabla f(\mathbf{x}^{(k)}) \\
\qquad \mu^{(k)} := \textbf{step-size}(f, \mathbf{x}^{(k)}, \Delta\mathbf{x}^{(k)}) \\
\qquad \mathbf{x}^{(k+1)} := \mathbf{x}^{(k)} + \mu^{(k)} \Delta\mathbf{x}^{(k)} \\
\qquad \text{if } \textbf{converged}(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}, f): \\
\qquad\quad \text{return } \mathbf{x}^{(k+1)}, f(\mathbf{x}^{(k+1)}) \\
\quad \text{raise exception "not converged in } K \text{ iterations}
\end{array}
$$



# Gradient Descent / Implementations

- for analysis usually all updated variables are indexed

$$\mathbf{x}^{(k)}, \Delta\mathbf{x}^{(k)}, \mu^{(k)}$$

- in implementations, one usually does only need one copy
    - or two, to compare against the last one

```
min-GD(f, x, K):
  for k := 0 : K − 1:
    Δx := −∇f(x)
    μ := step-size(f, x, Δx)
    x^old := x
    x := x^old + μΔx
    if converged(x, x^old, f):
      return x, f(x)
  raise exception "not converged in K iterations"
```

# Gradient Descent / Considerations

- Stopping criterion: $||\nabla f(\mathbf{x})||_2 \leq \epsilon$

$$\textbf{converged}(\mathbf{x}, \mathbf{x}^{\text{old}}, f) := \textbf{converged}(\nabla f(\mathbf{x})) := ||\nabla f(\mathbf{x})||_2 \leq \epsilon$$

  - cheap to use as GD has to compute the gradient anyway.

- GD is simple and straightforward.

- GD has slow convergence.
  - esp. compared to Newton's method (see next chapter)

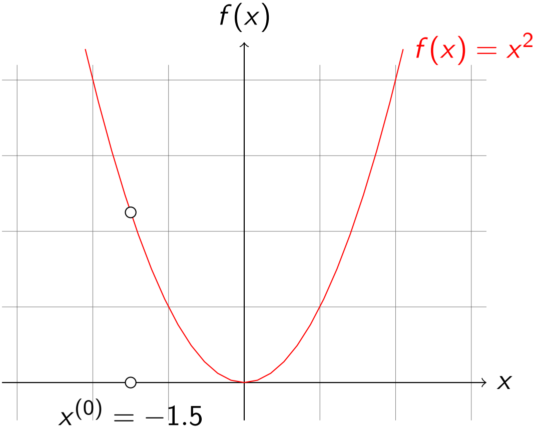
- Out-of-the-box, GD works only well for convex problems, otherwise will get stuck in local minima.

# Gradient Descent Example

**Task:** minimize $f(x) := x^2$

- $\mu = 0.3$
- $-\nabla f(x) = -2x$

Initial point: $x^{(0)} = -1.5$

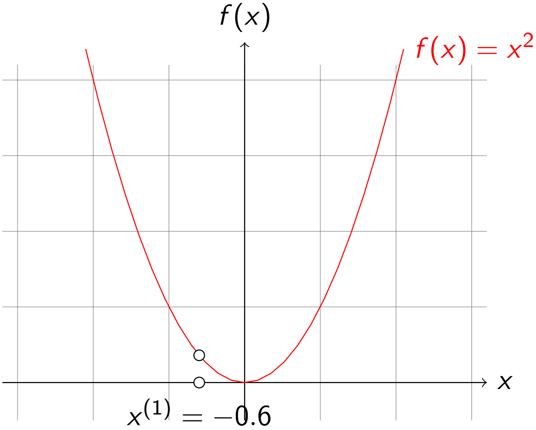


# Gradient Descent Example

**Task:** minimize $f(x) := x^2$

- $\mu = 0.3$
- $-\nabla f(x) = -2x$

$$\begin{aligned} x^{(0)} &= -1.5 \\ x^{(1)} &= -1.5 - 0.3 \cdot (2 \cdot (-1.5)) \\ &= -0.6 \end{aligned}$$

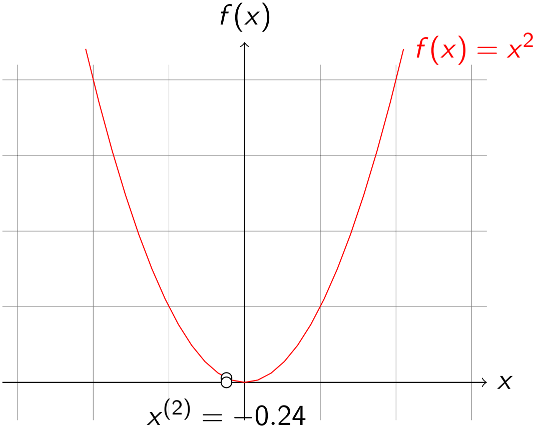


# Gradient Descent Example

**Task:** minimize $f(x) := x^2$

- $\mu = 0.3$
- $-\nabla f(x) = -2x$

$$x^{(1)} = -0.6$$
$$x^{(2)} = -0.6 - 0.3 \cdot (2 \cdot (-0.6)) = -0.24$$

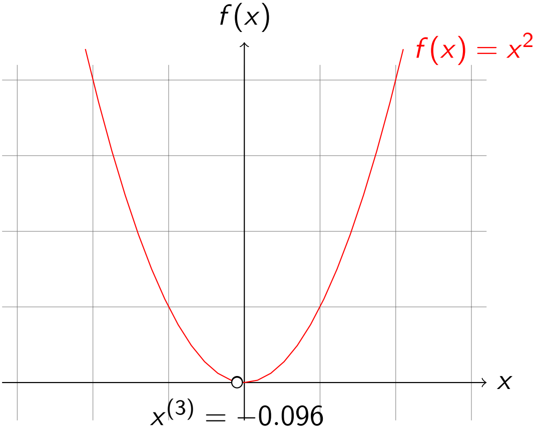


# Gradient Descent Example

**Task:** minimize $f(x) := x^2$

- $\mu = 0.3$
- $-\nabla f(x) = -2x$

$$x^{(2)} = -0.24$$
$$\begin{aligned} x^{(3)} &= -0.24 - 0.3 \cdot (2 \cdot (-0.24)) \\ &= -0.096 \end{aligned}$$

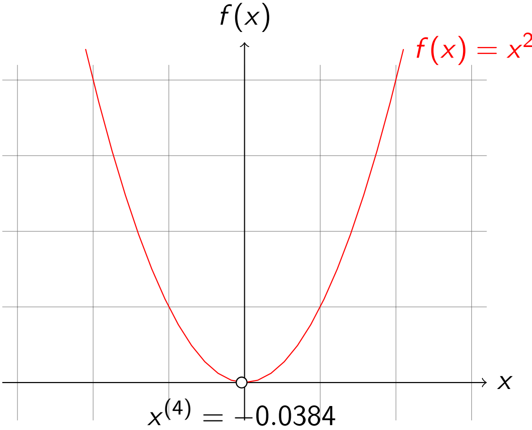


# Gradient Descent Example

**Task:** minimize $f(x) := x^2$

- $\mu = 0.3$
- $-\nabla f(x) = -2x$

$$
\begin{aligned}
x^{(3)} &= -0.096 \\
x^{(4)} &= -0.096 - 0.3 \cdot (2 \cdot (-0.096)) \\
&= -0.0384
\end{aligned}
$$

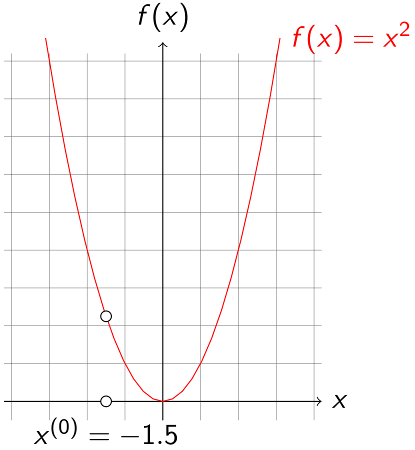


# How About a Larger Step Size?

**Task:** minimize $f(x) := x^2$

- $\mu = 1.5$
- $-\nabla f(x) = -2x$

Initial point: $x^{(0)} = -1.5$



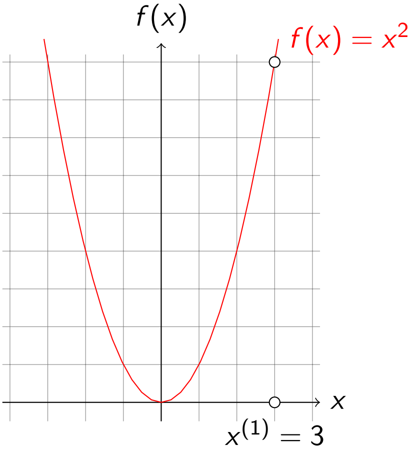
# How About a Larger Step Size?

**Task:** minimize $f(x) := x^2$

- $\mu = 1.5$
- $-\nabla f(x) = -2x$

$$x^{(0)} = -1.5$$

$$x^{(1)} = -1.5 - 1.5 \cdot (2 \cdot (-1.5))$$
$$= 3$$

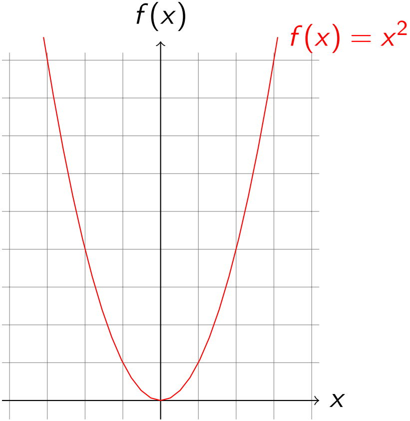


# How About a Larger Step Size?

**Task:** minimize $f(x) := x^2$

- $\mu = 1.5$
- $-\nabla f(x) = -2x$

$$
\begin{aligned}
x^{(1)} &= 3 \\
x^{(2)} &= 3 - 1.5 \cdot (2 \cdot 3) \\
&= -6
\end{aligned}
$$



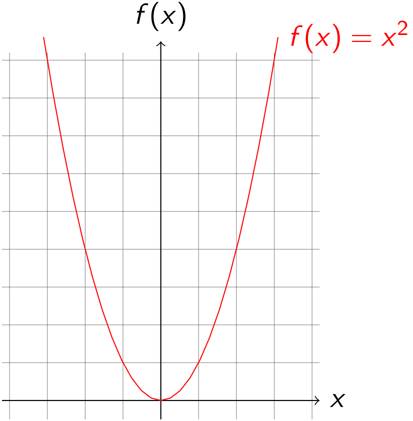
# How About a Larger Step Size?

**Task:** minimize $f(x) := x^2$

- $\mu = 1.5$
- $-\nabla f(x) = -2x$

$$x^{(1)} = 3$$
$$x^{(2)} = 3 - 1.5 \cdot (2 \cdot 3)$$
$$= -6$$

$\rightsquigarrow$ the algorithm diverges!

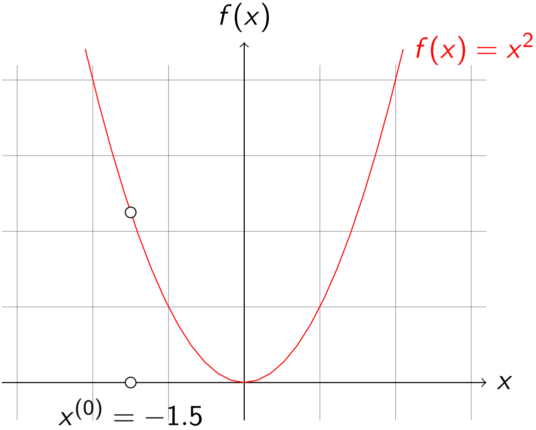


# Gradient Descent Example — Optimal Step Size

**Task:** minimize $f(x) := x^2$

- $\mu = 0.5$
- $-\nabla f(x) = -2x$

Initial point: $x^0 = -1.5$



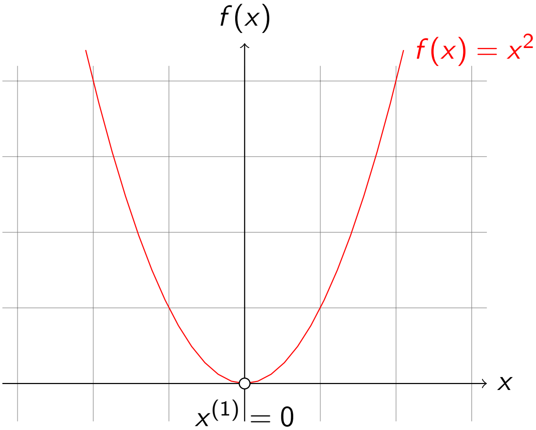
# Gradient Descent Example — Optimal Step Size

**Task:** minimize $f(x) := x^2$

- $\mu = 0.5$
- $-\nabla f(x) = -2x$

$$x^{(0)} = -1.5$$

$$x^{(1)} = -1.5 - 0.5 \cdot (2 \cdot (-1.5)) = 0$$



# Gradient Descent Example — Optimal Step Size

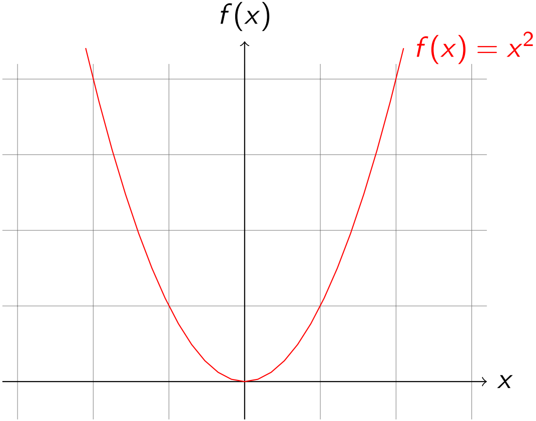
**Task:** minimize $f(x) := x^2$

- $\mu = 0.5$
- $-\nabla f(x) = -2x$

$$x^{(0)} = -1.5$$

$$x^{(1)} = -1.5 - 0.5 \cdot (2 \cdot (-1.5)) = 0$$

$\rightsquigarrow$ the algorithm converges in 1 step!



# How to Choose the Step Size $\mu$?

- Step size $\mu$ is crucial for the convergence of the algorithm.
  - Step size too small. $\rightsquigarrow$ slow convergence.
  - Step size too large. $\rightsquigarrow$ divergence!
- How to choose a good step size?
  $\rightsquigarrow$ **line search** (aka **step size control**).

# Outline

# Computing the Step Size

The step size can be computed in various ways:

- constant value
  - e.g., 1
- decreasing sequence, e.g., $\gamma^k$ for $\gamma \in (0, 1)$
  - e.g., for $\gamma = \frac{1}{2}$: $1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots$
- line search
- various heuristics depending on the specific algorithm

# Line Search

- **line search** is the task to compute the step size in a descent algorithm.

- itself a one-dimensional optimization problem in $\mu$:

$$\arg\min_{\mu \in \mathbb{R}^+} f(\mathbf{x} + \mu \Delta \mathbf{x})$$

# Line Search Methods

- **exact line search**:
  - Used if the problem can be solved analytically or with low cost.
  - e.g., for **unconstrained quadratic optimization**:

$$\arg\min_{x \in \mathbb{R}^N} f(x) := \frac{1}{2} x^T A x + b^T x, \quad A \in \mathbb{R}^{N \times N} \text{ pos. def.}, b \in \mathbb{R}^N$$

# Line Search Methods

- **exact line search**:
  - Used if the problem can be solved analytically or with low cost.
  - e.g., for **unconstrained quadratic optimization**:

$$\arg\min_{x\in\mathbb{R}^N} f(x) := \frac{1}{2}x^T A x + b^T x, \quad A \in \mathbb{R}^{N\times N} \text{ pos. def.}, b \in \mathbb{R}^N$$

- **backtracking line search**:
  - only approximative
  - guarantees that the new function value is lower than a specific bound.

# Backtracking Line Search

**stepsize-backtracking**$(f, \mathbf{x}, \Delta\mathbf{x}, \alpha \in (0, 0.5), \beta \in (0, 1))$:

  $\mu := 1$

  while $f(\mathbf{x} + \mu\Delta\mathbf{x}) > f(\mathbf{x}) + \alpha\mu\nabla f(\mathbf{x})^T\Delta\mathbf{x}$:

    $\mu := \beta\mu$

  return $\mu$

Q: Why does the backtracking condition guarantee $f(\mathbf{x}^{\text{next}}) < f(\mathbf{x})$ ?

# Backtracking Line Search

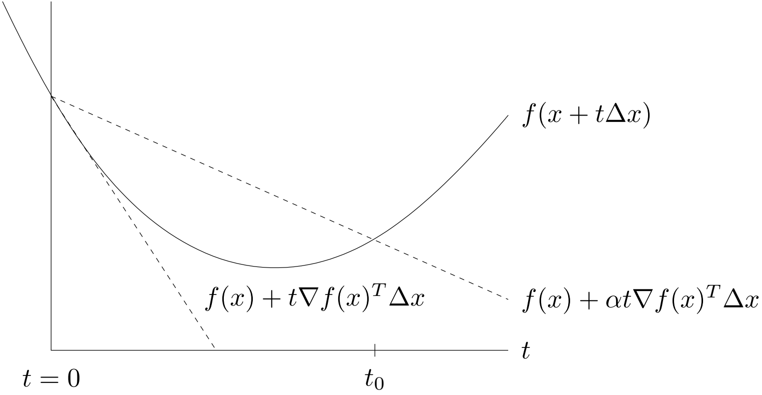
**stepsize-backtracking**$(f, \mathbf{x}, \Delta\mathbf{x}, \alpha \in (0, 0.5), \beta \in (0, 1))$:
  $\mu := 1$
  while $f(\mathbf{x} + \mu\Delta\mathbf{x}) > f(\mathbf{x}) + \alpha\mu\nabla f(\mathbf{x})^T \Delta\mathbf{x}$:
    $\mu := \beta\mu$
  return $\mu$

Loop eventually terminates: for sufficient small $\mu$:

$$f(x + \mu\Delta x) \approx f(x) + \mu\nabla f(x)^T \Delta x < f(x) + \alpha\mu\nabla f(x)^T \Delta x$$

as for a descent direction: $\nabla f(x)^T \Delta x < 0$

# Backtracking Line Search




source: Boyd and Vandenberghe, 2004, p. 465

# Outline

# Sublevel Sets

**sublevel set** of $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ at level $\alpha \in \mathbb{R}$:

$$S_\alpha(f) := \{x \in \operatorname{dom} f \mid f(x) \leq \alpha\}$$

# Sublevel Sets

**sublevel set** of $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ at level $\alpha \in \mathbb{R}$:

$$S_\alpha(f) := \{x \in \operatorname{dom} f \mid f(x) \leq \alpha\}$$

basic facts:

- if $f$ is convex, then all its sublevel sets $S_\alpha$ are convex sets.
  - useful to show that a set is convex:
    - show that it can be represented as a sublevel set of a convex function.

# Sublevel Sets / Examples

$$S_\alpha(x^2) =$$

$$S_\alpha(-\log x; \mathbb{R}^+) =$$

$$S_\alpha(\frac{1}{x}; \mathbb{R}^+) =$$

$$S_\alpha(x; \mathbb{R}^+) =$$

$S_\alpha(f) := \{x \in \operatorname{dom} f \mid f(x) \leq \alpha\}$

# Sublevel Sets / Examples

$$S_\alpha(x^2) = \begin{cases} [-\sqrt{\alpha}, \sqrt{\alpha}], & \alpha \geq 0 \\ \emptyset, & \text{else} \end{cases}$$

$$S_\alpha(-\log x; \mathbb{R}^+) = [e^{-\alpha}, \infty)$$

$$S_\alpha(\frac{1}{x}; \mathbb{R}^+) = \begin{cases} [\frac{1}{\alpha}, \infty), & \alpha \geq 0 \\ \emptyset, & \text{else} \end{cases}$$

$$S_\alpha(x; \mathbb{R}^+) = \begin{cases} (0, \alpha], & \alpha > 0 \\ \emptyset, & \text{else} \end{cases}$$

$S_\alpha(f) := \{x \in \text{dom}\, f \mid f(x) \leq \alpha\}$

# Closed Functions

$f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ **closed** $:\Longleftrightarrow$ all its sublevel sets are closed.

# Closed Functions

$f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ **closed** $:\Longleftrightarrow$ all its sublevel sets are closed.

examples:

- $f(x) = x^2$ is closed.
- $f(x) = 1/x$ on $\mathbb{R}^+$ is closed.
- $f(x) = x$ on $\mathbb{R}^+$ is not closed.
  - but $f$ on $\mathbb{R}_0^+$ is closed.
- $f(x) = x \log x$ on $\mathbb{R}^+$ is not closed.
  - but $f$ on $\mathbb{R}_0^+$ is closed, defined by

$$f(x) := \begin{cases} x \log x, & \text{if } x > 0 \\ 0, & \text{else} \end{cases}$$

# Closed Functions

$f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ **closed** $:\Longleftrightarrow$ all its sublevel sets are closed.

examples:

- $f(x) = x^2$ is closed.
- $f(x) = 1/x$ on $\mathbb{R}^+$ is closed.
- $f(x) = x$ on $\mathbb{R}^+$ is not closed.
  - but $f$ on $\mathbb{R}_0^+$ is closed.
- $f(x) = x \log x$ on $\mathbb{R}^+$ is not closed.
  - but $f$ on $\mathbb{R}_0^+$ is closed, defined by

$$f(x) := \begin{cases} x \log x, & \text{if } x > 0 \\ 0, & \text{else} \end{cases}$$

Classes of closed functions:

- continuous functions on all of $\mathbb{R}^N$
- continuous functions on an open set
  that go to infinity everywhere towards the border

# Semidefinite Matrices II

Let $A, B \in \mathbb{R}^{N \times N}$ symmetric matrices:

$$A \succeq B :\Longleftrightarrow A - B \succeq 0$$

- $A \succeq mI, m \in \mathbb{R}^+$:
  - all eigenvalues of A are $\geq m$
- $A \preceq MI, M \in \mathbb{R}^+$:
  - all eigenvalues of A are $\leq M$

# Strongly Convex Functions

Let $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ be twice continuously differentiable.

$f$ is **strongly convex** $:\Longleftrightarrow$

- $\operatorname{dom} f = X$ is convex and
- the eigenvalues of the Hessian are uniformly bounded from below:

$$\nabla^2 f(x) \succeq mI, \quad \exists m \in \mathbb{R}^+ \ \forall x \in \operatorname{dom} f$$

# Strongly Convex Functions

Let $f : X \to \mathbb{R}, X \subseteq \mathbb{R}^N$ be twice continuously differentiable.

$f$ is **strongly convex** $:\Longleftrightarrow$

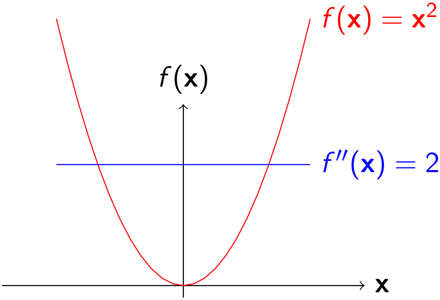
- $\operatorname{dom} f = X$ is convex and
- the eigenvalues of the Hessian are uniformly bounded from below:

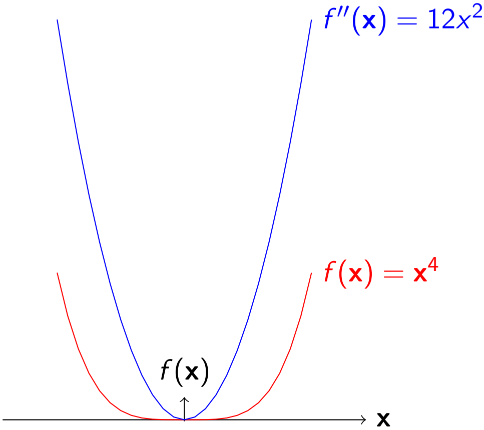
$$\nabla^2 f(x) \succeq mI, \quad \exists m \in \mathbb{R}^+ \; \forall x \in \operatorname{dom} f$$

Every strongly convex function $f$ is also strictly convex.

- but not the other way around
  - $f(x) = x^4$ (on $\mathbb{R}$) is strictly, but not strongly convex
- do not confuse strongly and strictly convex!

# Strongly Convex Functions / Examples




Q: Is $f$ convex, strictly or strongly convex?

(convex: $\forall x : \nabla^2 f(x) \succeq 0$, strictly convex: $\forall x : \nabla^2 f(x) \succ 0$, strongly convex: $\exists m > 0 \; \forall x : \nabla^2 f(x) \succeq mI$)

# Strongly Convex Functions / Basic Facts

(i) $f$ is above a parabola:

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2} ||y - x||_2^2$$
$$p^* \geq f(x) - \frac{1}{2m} ||\nabla f(x)||_2^2$$

(ii) if $f$ is closed and $S$ one of its sublevel sets, then

a) the eigenvalues of the Hessian are also uniformly bounded from above on $S$:

$$\nabla^2 f(x) \preceq MI, \quad \exists M \in \mathbb{R}^+ \ \forall x \in S$$

b) $f$ is below a parabola ("sandwiched between two parabolas"):

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{M}{2} ||y - x||_2^2, \quad x, y \in S$$
$$p^* \leq f(x) - \frac{1}{2M} ||\nabla f(x)||_2^2$$

# Strongly Convex Functions / Basic Facts / Proofs

(i) for $x, y \in \operatorname{dom} f\ \exists z \in [x, y]$
(Taylor expansion with Lagrange mean value remainder):

$$f(y) = f(x) + \nabla f(x)^T (y-x) + \frac{1}{2} \underbrace{(y-x)^T \nabla^2 f(z)(y-x)}_{\geq m||y-x||_2^2}$$

$$\begin{aligned}
f(y) &\geq f(x) + \nabla f(x)^T (y-x) + \frac{m}{2}||y-x||_2^2 \\
&\geq \min_y f(x) + \nabla f(x)^T (y-x) + \frac{m}{2}||y-x||_2^2 \\
&\qquad\qquad \text{considered as function in } y \text{ has} \\
&\qquad\qquad \text{minimum at } \tilde{y} := x - \frac{1}{m}\nabla f(x) \\
&= f(x) + \nabla f(x)^T (\tilde{y}-x) + \frac{m}{2}||\tilde{y}-x||_2^2 \\
&= f(x) - \frac{1}{2m}||\nabla f(x)||_2^2
\end{aligned}$$

$$\rightsquigarrow\ p^* = f(y = x^*) \geq f(x) - \frac{1}{2m}||\nabla f(x)||_2^2$$

# Strongly Convex Functions / Basic Facts / Proofs (2/2)

(ii.a)
- due to (i) all sublevel sets are bounded
- the maximal eigenvalue of $\nabla^2 f(x)$ is a continuous function on a closed bounded set and thus itself bounded,
  - i.e., it exists $M \in \mathbb{R}^+$: $\nabla^2 f(x) \preceq MI$

(ii.b) as for (i), using (ii.a)

# Theorem (Convergence of Gradient Descent — exact line search)

If (i) $f$ is strongly convex,

(ii) the initial sublevel set $S := \{x \in \operatorname{dom} f \mid f(x) \leq f(x^{(0)})\}$ is closed,

(iii) an exact line search is used,

then

$$f(x^{(k)}) - p^* \leq (1 - \frac{m}{M})^k \; (f(x^{(0)}) - p^*)$$

Equivalently, to guarantee $f(x^{(k)}) - p^* \leq \epsilon$, GD requires

$$k := \frac{\log \frac{f(x^0) - p^*}{\epsilon}}{\log \frac{1}{1 - \frac{m}{M}}} \quad \text{iterations.}$$

Especially,

- GD converges, i.e., $f(x^{(k)})$ approaches $p^*$
- the convergence is exponential in $k$ (with basis $c := 1 - \frac{m}{M}$)
    - called **linear convergence** in the optimization literature

# Convergence of Gradient Descent / Proof

$$\tilde{f}(t) := f(x - t\nabla f(x)), \quad t \in \{t \in \mathbb{R}_0^+ \mid x - t\nabla f(x) \in S\}$$

$$\begin{aligned}
f(x^{\text{next}}) &= \tilde{f}(t_{\text{exact}}) = \tilde{p}^*, && \tilde{p}^* := \min_t \tilde{f}(t) \\
&\leq \tilde{f}(0) - \frac{1}{2M}(\tilde{f}'(0))^2, && \tilde{f} \text{ strongly convex (ii.b)} \\
&= f(x) - \frac{1}{2M} \underbrace{||\nabla f(x)||_2^2}_{\geq 2m(f(x) - p^*)}, && f \text{ strongly convex (i)} \\
&\leq f(x) - \frac{m}{M}(f(x) - p^*)
\end{aligned}$$

$$\begin{aligned}
f(x^{\text{next}}) - p^* &\leq f(x) - p^* - \frac{m}{M}(f(x) - p^*) = (1 - \frac{m}{M})(f(x) - p^*) \\
f(x^{(k)}) - p^* &\leq (1 - \frac{m}{M})^k (f(x^{(0)}) - p^*)
\end{aligned}$$

# Convergence of Gradient Descent / in $x$

GD's convergence can also be described in $x$ (instead of in $f$):

$$\begin{aligned}
||x^{(k)} - x^*||^2 &\underset{\text{s.c.(i)}}{\leq} \frac{2}{m}(f(x^{(k)}) - p^*) \\
&\underset{\text{conv}}{\leq} \frac{2}{m}(1 - \frac{m}{M})^k (f(x^{(0)}) - p^*) \\
&\underset{\text{s.c.(i)}}{\leq} (1 - \frac{m}{M})^k \frac{2}{m} \frac{1}{2m} ||(\nabla f(x))||^2 \\
&= (1 - \frac{m}{M})^k \frac{||(\nabla f(x^{(0)}))||^2}{m^2}
\end{aligned}$$

# Theorem (Convergence of Gradient Descent — Backtracking)

If (i) $f$ is strongly convex,
(ii) the initial sublevel set $S := \{x \in \operatorname{dom} f \mid f(x) \leq f(x^{(0)})\}$ is closed,
(iii) a backtracking line search is used,

then

$$f(x^{(k)}) - p^* \leq c^k \left(f(x^{(0)}) - p^*\right), \quad c := 1 - \min\{2\alpha m, 2\beta\alpha m/M\}$$

Equivalently, to guarantee $f(x^{(k)}) - p^* \leq \epsilon$, GD requires

$$k := \frac{\log \frac{f(x^0) - p^*}{\epsilon}}{\log \frac{1}{c}} \quad \text{iterations.}$$

Especially,

- GD converges, i.e., $f(x^{(k)})$ approaches $p^*$
- the convergence is exponential in $k$ (with basis $c$; linear convergence)

# Summary (1/2)

- **Unconstrained optimization** is the minimization of a function over all of $\mathbb{R}^N$ or an open subset $X \subseteq \mathbb{R}^N$.
  - In **Unconstrained convex optimization** $X$ also has to be convex (and $f$, too).
- **Descent methods** iteratively find a next iterate $x^{(k+1)}$ with lower function value than the last iterate and require:
  - **search direction**: in which direction to search.
    - **Gradient Descent** (GD): negative gradient of the target function
  - **step size**: how far to go.
  - **convergence criterion**: when to stop.
    - small last step
    - small gradient

# Summary (2/2)

- step size (aka **line search**) in rare cases can be computed exactly.
  - one-dimensional optimization problem (**exact line search**)
- **backtracking line search**:
  - Choose the largest stepsize that guarantees a decrease in function value.
  - guaranteed to terminate
- GD has **linear convergence**
  - exponential in the number of steps
    - with basis $1 - m/M$
      for smallest/largest eigenvalues $m$,$M$ of the Hessian
  - if $f$ is strongly convex, its initial sublevel set closed and exact line search is used.

# Further Readings

- Unconstrained minimization problems:
  - Boyd and Vandenberghe, 2004, chapter 9.1
- Descent methods:
  - Boyd and Vandenberghe, 2004, chapter 9.2
- Gradient descent:
  - Boyd and Vandenberghe, 2004, chapter 9.3
- also accessible from here:
  - steepest descent — Boyd and Vandenberghe, 2004, chapter 9.4

# References

Boyd, Stephen and Lieven Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.