MS-C2105 - Introduction to Optimization
Lecture 9

Fabricio Oliveira (with modifications by Harri Hakula)

Systems Analysis Laboratory
Department of Mathematics and Systems Analysis

Aalto University
School of Science

March 22, 2022

Outline of this lecture

Multidimensional functions

Optimality conditions

Optimisation methods

- Steepest descent/ gradient method
- Newton's method

Reading: Taha: Chapter 20; Winston: Chapter 11

n-dimensional functions

Definition 1

Let $f : \mathbb{R}^n \to \mathbb{R}$. The partial derivative of f with respect to x_i is

$$\frac{\partial f}{\partial x_i} = \lim_{\Delta x_i \to 0} \frac{f(x_1, \dots, \Delta x_i \dots, x_n) - f(x_1, \dots, x_i \dots, x_n)}{\Delta x_i}$$

Partial differentiability isolates variation in an individual dimension.

- Suppose that $x + \epsilon = (x_1 + \epsilon_1, \dots, x_n + \epsilon_n)$, with $\epsilon_i > 0$ for some $i = 1, \dots, n$. Then the approximate variation in $f(x)$ is

$$\sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x + \epsilon).$$

- Similarly, second-order partials can be obtained by successively taking partials, denoted by $\frac{\partial^2 f}{\partial x_i \partial x_j}$. If they exist and are continuous in the domain of f, $\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}$.

Gradients and Hessians

Definition 2 (Gradient)

The gradient of $f : \mathbb{R}^n \to \mathbb{R}$ is given by

$$\nabla f(x) = \left[\frac{\partial f(x)}{\partial x_1}, \dots, \frac{\partial f(x)}{\partial x_n}\right]^\top .$$

Definition 3 (Hessian)

The Hessian (matrix) of $f : \mathbb{R}^n \to \mathbb{R}$ is a $n \times n$ matrix given by

$$\nabla^2 f(x) = H(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \dots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \dots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}.$$

Gradients and Hessians

For a function $f : \mathbb{R}^n \to \mathbb{R}$, gradients and Hessians describe the local (growth) behaviour and shape of the function.

Considering the definitions of the gradient and the Hessian, the Taylor's series becomes

$$f(x) = f(x_0) + \nabla f(x_0)^\top (x - x_0) + \frac{1}{2}(x - x_0)^\top H(x_0)(x - x_0) + o(||x - x_0||^2)$$

Remarks:

- The residual is represented using the little-o notation for convenience (shows that it can be dropped for sufficiently small $\Delta x = (x - x_0)$.
- $o(f(x))$ represents a function that goes to zero "faster" than $f(x)$, i.e., $\lim_{x\to 0} \frac{o(f(x))}{f(x)} = 0$.

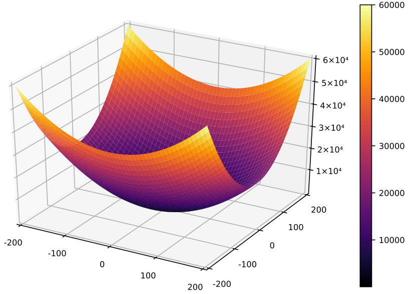
Gradients and Hessians

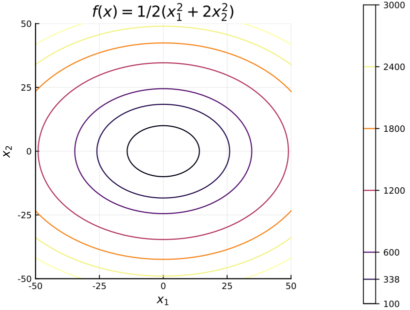
Considering the first-order approximation at x_0, we have that:

$$f(x) = f(x_0) + \nabla f(x_0)(x - x_0)$$

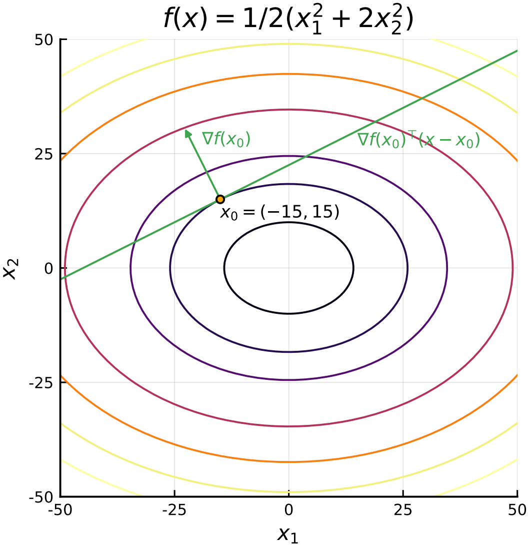
That is: for any level $z = f(x_0)$, $\nabla f(x_0)$ is the normal of the tangent hyperplane $\nabla f(x_0)(x - x_0) = 0$.

Example: $f(x) = 1/2(x_1^2 + 2x_2^2)$. $\nabla f(x) = (x_1, 2x_2)$.





Gradients and Hessians



Optimality conditions

Analogously to the one-dimensional case, we can define first- and second- order optimality conditions using gradients and Hessians.

First, let us define the concept of descent/ ascent directions.

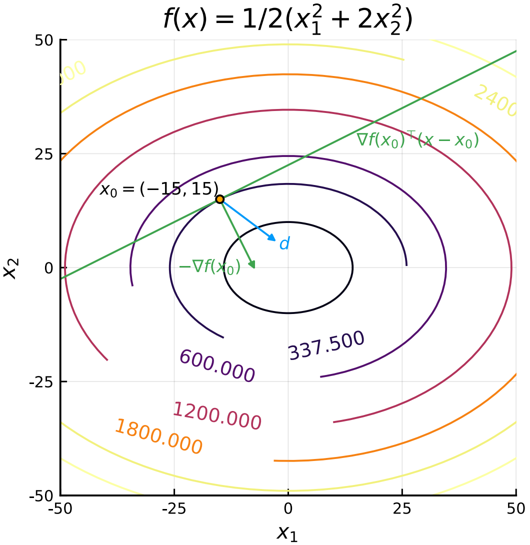
Definition 4 (Descent direction)

Let $f : \mathbb{R}^n \to \mathbb{R}$ be differentiable. The vector $d = (x - x_0)$ is a descent direction for f at x_0 if $\nabla f(x_0)^\top d < 0$.

Remarks:

- Analogously, d is an ascent direction if $\nabla f(x_0)^\top d > 0$;
- for descent, any d with a component projected on $-\nabla f(x_0)$ ($\nabla f(x_0)$ for ascent) is such a direction;
- or $d \in H = \left\{p : \nabla f(x_0)^\top p \leq 0\right\}$ (descent; ≥ 0 for ascent).

Optimality conditions



Optimality conditions

The necessary optimality conditions can be stated as follows.

Theorem 5 (First-order optimality conditions)

Let $f : \mathbb{R}^n \to \mathbb{R}$ be differentiable. If $\overline{x}$ is a local optimum, then $\nabla f(\overline{x}) = 0$.

Alternative ways of proving:

1. Apply Theorem 5, Lecture 8 for each component of $\overline{x}$.
2. If no descent direction exists from $\overline{x}$, then we must have $\nabla f(\overline{x}) = 0$.

Second-order conditions require analysing $H(\overline{x})$. Consider the second-order expansion of f at x_0

$$f(x) = f(x_0) + \nabla f(x_0)^\top (x - x_0) + \frac{1}{2}(x - x_0)^\top H(x_0)(x - x_0) + o(||x - x_0||^2)$$

Optimality conditions

For $\nabla f(x_0) = 0$, we notice that

$$f(x) = f(x_0) + \frac{1}{2}(x - x_0)^\top H(x_0)(x - x_0) + o(||x - x_0||^2)$$

$$f(x) - f(x_0) = \frac{1}{2}(x - x_0)^\top H(x_0)(x - x_0) + o(||x - x_0||^2).$$

The optimality of x_0 is tied to the sign of $(x - x_0)^\top H(x_0)(x - x_0)$:

- if, for all x, $(x - x_0)^\top H(x_0)(x - x_0) > 0$, then x_0 is a local minimum
- if, for all x, $(x - x_0)^\top H(x_0)(x - x_0) < 0$, then x_0 is a local maximum

These condition can be tested using positive definiteness.

Definition 6

A $n \times n$ matrix H is positive definite if $x^\top Hx > 0$ for all $x \in \mathbb{R}^n$.

Optimality conditions

For symmetric matrices, the following are equivalent:

1. H is positive definite
2. the eigenvalues of H are positive
3. the determinant of H is positive

From 2., all others can be proved. Recall that $\mathbf{det}(H) = \prod_{i=1}^{n} \lambda_i$, where λ_i for $i = 1, \dots, n$ are the eigenvalues of H. To calculate eigenvalues, we solve,

$$Hx = \lambda x \Rightarrow (H - \lambda I)x = 0,$$

which only has nonzero solutions for $x \neq 0$ if $\mathbf{det}(H - \lambda I) = 0$.

The eigenvectors are the nonzero x that solve

$$(H - \lambda_j I)x = 0, \text{ for } j = 1, \dots, n.$$

Optimality conditions

Example: f(x) $= 1/2(x_1^2 + 2x_2^2)$.

$\nabla f(x) = [x_1, 2x_2]^\top$, $H(x) = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$;

First-order conditions: $\nabla f(x) = 0 \Rightarrow \overline{x} = (0, 0)$.

Second-order conditions: $H(\overline{x}) = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$;

$\mathbf{det}\left(\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right) = 0 \Rightarrow \mathbf{det}\left(\begin{bmatrix} 1-\lambda & 0 \\ 0 & 2-\lambda \end{bmatrix}\right) = 0 \Rightarrow$

$(1-\lambda)(2-\lambda) = 0 \Rightarrow \lambda_1 = 1, \lambda_2 = 2$ or vice versa.

Thus, $H(\overline{x})$ is positive definite and $\overline{x}$ is a local minimum.

Remark: $H(x)$ being positive definite for all x implies that f is strictly convex. Hence $\overline{x}$ is a global optimum.

Multidimensional optimisation methods

Most optimisation methods can be represented by this pseudocode:

Algorithm Conceptual optimisation algorithm

initialise. iteration count $k = 0$, starting point x_0
while stopping criteria are not met **do**
    compute direction d_k
    compute step size λ_k
    $x_{k+1} = x_k + \lambda_k d_k$
    $k = k + 1$
end while
return x_k.

where

- k is an iteration counter;
- λ_k is a suitable step size;
- d_k is a direction vector;

Gradient (descent) method

Recall that if d is a descent direction, there exists $\delta > 0$ such that $f(x + \lambda d) < f(x)$ for all $\lambda \in (0, \delta)$. The following result provides directions of steepest descent.

Lemma 7

Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable at $x \in \mathbb{R}^n$ and $\nabla f(x) \neq 0$. Then $\overline{d} = -\frac{\nabla f(x)}{||\nabla f(x)||}$ is the direction of steepest descent of f at x.

Proof.

d is a descent direction if $\nabla f(x)^\top d < 0$. Thus,

$$\overline{d} = \underset{||d|| \leq 1}{\operatorname{argmin}} \left\{ \nabla f(x)^\top d \right\} = -\frac{\nabla f(x)}{||\nabla f(x)||} \quad \square$$

Gradient (descent) method

Algorithm Gradient descent method

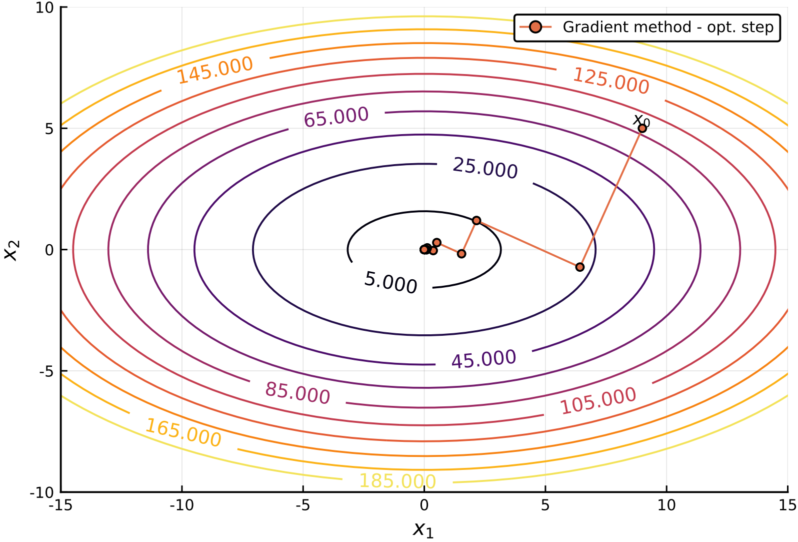
initialise. tolerance $\epsilon > 0$, initial point x_0, iteration count $k = 0$.
while $||\nabla f(x_k)|| > \epsilon$ **do**
    $d_k = -\nabla f(x_k)$.
    $\overline{\lambda} = \text{argmin}_{\lambda \in \mathbb{R}} \{f(x_k + \lambda d_k)\}$.
    $x_{k+1} = x_k + \overline{\lambda} d_k$.
    $k \leftarrow k + 1$.
end while
return x_k.

Remarks:

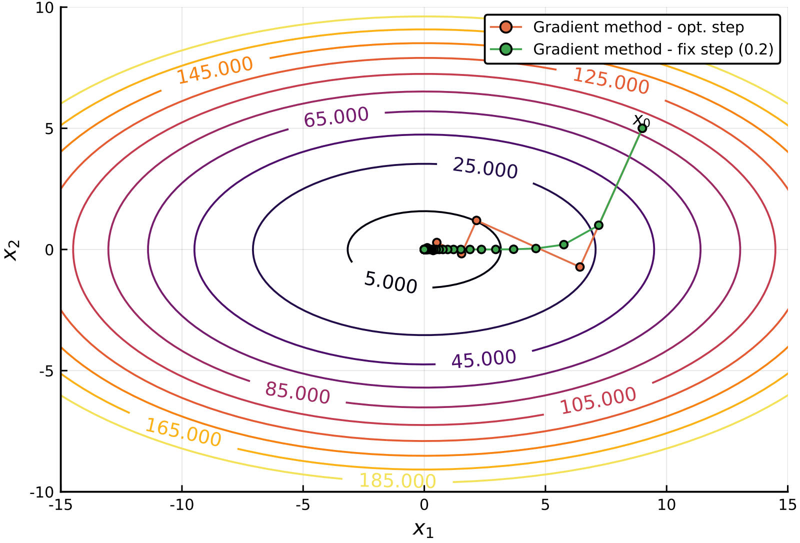
1. Steepest descent and gradient methods are different. When $||d|| \leq 1$ uses Euclidean norm, they are equivalent;
2. Poor convergence and zigzagging in later iterations due to the linear approximations;
3. Stochastic gradient methods use randomly selected blocks of coordinates x_i to calculate gradients.

Gradient (descent) method

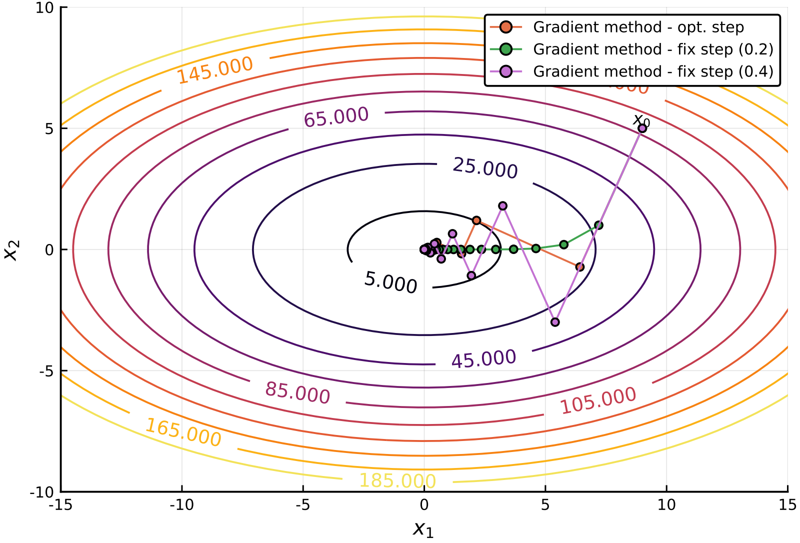
Example: $f(x) = 1/2(x_1^2 + 2x_2^2)$. Starting at (9,5).



Gradient (descent) method



Gradient (descent) method



Newton's method

Same idea as in the univariate case. Can also be seen as deflected steepest descent.

Deflection is achieved using the Hessian, which is equivalent to relying on quadratic approximations (rather than linear).

Consider the second-order approximation of f at x_k:

$$q(x) = f(x_k) + \nabla f(x_k)^\top (x - x_k) + \frac{1}{2}(x - x_k)H(x_k)(x - x_k),$$

where $H(x_k)$ is the Hessian at x_k. Once again, we require that $\nabla q(x_{k+1}) = 0$, which leads to

$$\nabla f(x_k) + H(x_k)(x - x_k) = 0.$$

Assuming that $H^{-1}(x_k)$ exists, we obtain the update rule:

$$x_{k+1} = x_k - H^{-1}(x_k)\nabla f(x_k).$$

Newton's method

Algorithm Newton's method

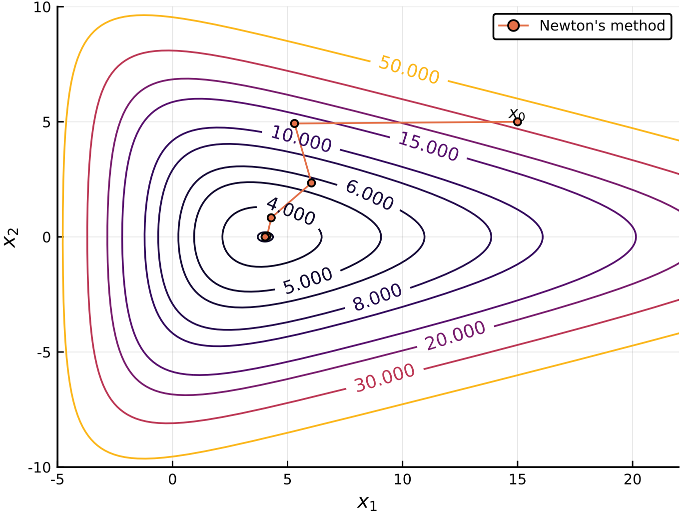
1: **initialise.** tolerance $\epsilon > 0$, initial point x_0, iteration count $k = 0$.
2: **while** $||\nabla f(x_k)|| > \epsilon$ **do**
3: $\quad d = -H^{-1}(x_k)\nabla f(x_k)$.
4: $\quad \overline{\lambda} = \text{argmin}_{\lambda \in \mathbb{R}} \{f(x_k + \lambda d)\}$.
5: $\quad x_{k+1} = x_k + \overline{\lambda} d$.
6: $\quad k \leftarrow k + 1$.
7: **end while**
8: **return** x_k.

Remarks:

1. Setting $\overline{\lambda} = 1$ recovers the "pure" Newton's method;
2. If x_0 is too far from optimal, Newton might not converge;

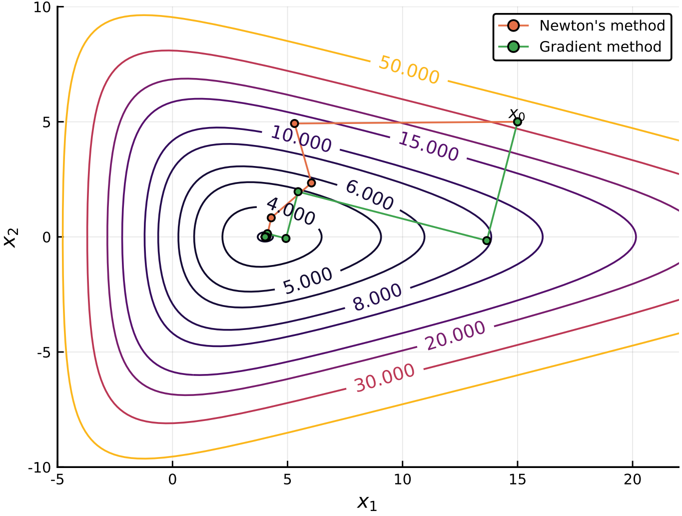
Newton's method

Example: $e^{-(x_1-3)/2} + e^{(4x_2+x_1)/10} + e^{(-4x_2+x_1)/10}$; $x_0 = (15{,}5)$.



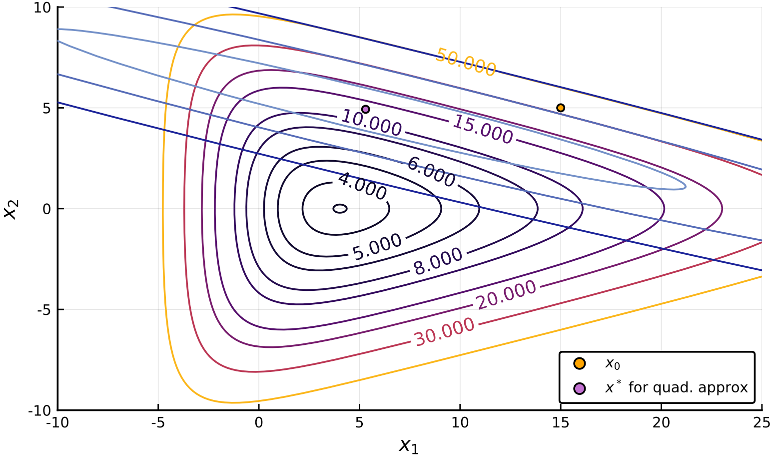
Newton's method

Example: $e^{-(x_1-3)/2} + e^{(4x_2+x_1)/10} + e^{(-4x_2+x_1)/10}$; $x_0 = (15,5)$.



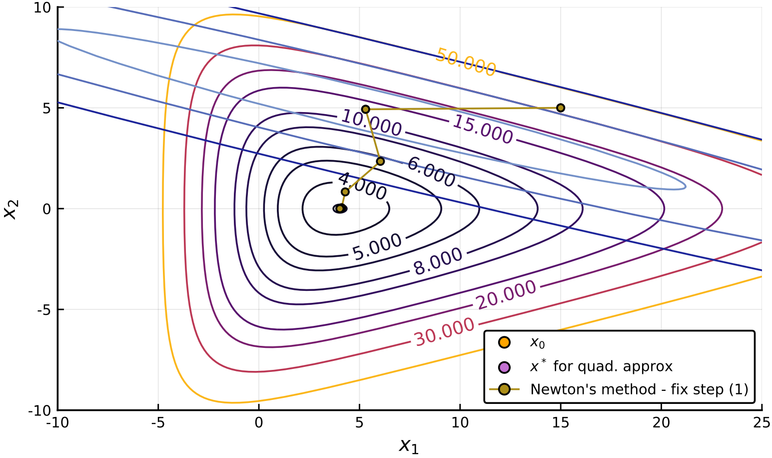
Newton's method

Example: $e^{-(x_1-3)/2} + e^{(4x_2+x_1)/10} + e^{(-4x_2+x_1)/10}$; $x_0 = (15,5)$.



Newton's method

Example: $e^{-(x_1-3)/2} + e^{(4x_2+x_1)/10} + e^{(-4x_2+x_1)/10}$; $x_0 = (15{,}5)$.



Newton's method

Example: $e^{-(x_1-3)/2} + e^{(4x_2+x_1)/10} + e^{(-4x_2+x_1)/10}$; $x_0 = (15,5)$.