

Optimisation Methods: Assignment 2

Déborah Sulem
deborah.sulem@usi.ch

Spring 2025

The purpose of this assignment is to familiarise yourself with Taylor's expansion, univariate and multivariate calculus, and the Gradient Descent algorithm.

It is due for **Wednesday 12th March at 2 pm**. Only Part 2 will be graded but you can also submit your answers for Part 1 and the latter will also be corrected (though not graded). For Part 2, you need to submit a Python notebook for each problem. For Part 2 you can submit a PDF with a scan of your handwritten notes or generated from a \LaTeX file. Submit all your files in iCorsi in a compressed folder with a `.zip` extension. Recall the basic instructions:

- Put the answers for each part of the question into separate cells.
- Before each cell, put a markdown header that says which part of the question comes in the following cell.
- Good coding style is part of the grade: add clear comments throughout the code when it is necessary.
- Before you submit your notebook, make sure it runs.

Part 0: reading

Read Chapter 2 of the book “Numerical Optimisation” (Nocedal & Wright).

Part 1: written exercises

Exercise 1 (Partial derivatives, gradient and Hessian)

Consider the function in 3 dimensions:

$$f(x) = x_1 \cos(x_3) + x_1 x_2^3 + x_2 \log x_2.$$

- Write down all first and second-order partial derivatives, i.e., $\frac{\partial f}{\partial x_i}$ for all $i = 1, 2, 3$, and the gradient $\nabla f(x)$.
- Write down all the second-order partial derivatives, i.e., $\frac{\partial f}{\partial x_i}$ and $\frac{\partial^2 f}{\partial x_i \partial x_j}$ for all $i, j = 1, 2, 3$ and the Hessian $H(x)$.

Exercise 2 (Convex and univariate functions)

Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be a univariate function. Prove the following statements.

1. Assume f is continuously differentiable. Then f is convex if and only if

$$f(y) \geq f(x) + f'(x)(y - x), \quad \forall x, y \in \mathbb{R}.$$

Give a graphical interpretation of this property.

Hint: if f is continuously differentiable then for any $x, t \in \mathbb{R}$,

$$f(x + t) - f(x) = \int_0^1 f'(x + tp) p dt.$$

2. Assume f is twice continuously differentiable. Then f is convex if and only if $f''(x) \geq 0, \forall x \in \mathbb{R}$.
3. if $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex and x^* is a local minimiser of f , then x^* is a global minimiser.
4. if $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex and x^* is a stationary point of f , then x^* is a global minimiser.

Exercise 3 (Strictly convex and univariate functions)

Let $f : \mathbb{R} \rightarrow \mathbb{R}$. Prove the following statements.

1. Assume $f : \mathbb{R} \rightarrow \mathbb{R}$ is twice continuously differentiable. If $f''(x) > 0, \forall x \in \mathbb{R}$, then f is strictly convex. Show that the converse is not true by giving a counter-example, i.e., find a function which is strictly convex but for which there exists x^* such that $f''(x^*) = 0$.
2. if $f : \mathbb{R} \rightarrow \mathbb{R}$ is strictly convex, then there exists at most one global minimiser.

Exercise 4 (Strongly convex and univariate functions)

Let $f : \mathbb{R} \rightarrow \mathbb{R}$. Prove the following statements.

1. If f is μ -strongly convex with $\mu > 0$, then

$$f(y) \geq f(x) + f'(x)(y - x) + \frac{\mu}{2}(y - x)^2, \quad \forall x, y \in \mathbb{R}.$$

Give a graphical interpretation of this property.

2. If f is μ -strongly convex with $\mu > 0$ then it has a unique global minimiser.

Exercise 5 (Smooth and univariate functions)

Let $f : \mathbb{R} \rightarrow \mathbb{R}$.

1. If f is L -smooth (differentiable with L -Lipschitz continuous derivative) then

$$f(y) \leq f(x) + f'(x)(y - x) + \frac{L}{2}(y - x)^2, \quad \forall x, y \in \mathbb{R}.$$

Hint: you may use the fact that $f(x + p) = \int_0^1 f'(x + tp) p dt$.

2. If f is μ -strongly convex and L -smooth, then $\mu \leq L$.

Part 2: programming problems

Problems 1 (Taylor's expansion of sin function)

In this problem we consider the univariate function $f(x) = \sin(x)$.

1. Plot the graph of $f(x)$ for $x \in [0, 10]$.
2. What is the k -th order derivative of f for any $k \geq 0$? Find its analytical form then write a function that calculates the k -th order derivative of f at any point $x \in \mathbb{R}$ and for any $k \geq 0$. Note that the zero-th order derivative corresponds to f .
3. Consider the point $\bar{x} = 0$.

(a) Find the form of the tangent line $t^1(x; \bar{x})$ of f at \bar{x} and plot it on top of the graph of f for $x \in [0, 10]$. Recall that the tangent line is the first-order Taylor approximation.

(b) We now want to compute the k -th order Taylor approximation of f for $k = 3, 5, 7, \dots, 19$:

$$t^k(x; \bar{x}) = f(\bar{x}) + \frac{1}{1!}f'(\bar{x})(x - \bar{x}) + \frac{1}{2!}f''(\bar{x})(x - \bar{x})^2 + \dots + \frac{1}{k!}f^{(k)}(\bar{x})(x - \bar{x})^k.$$

Compute and plot each curve $t^k(x; \bar{x})$, $k = 1, 3, 5, \dots, 19$ and the graph of f for $x \in [0, 10]$ (all curves need to be on the same figure).

(c) Compute the approximation error of f at $x = 2$, using the k -th Taylor approximation of f at $\bar{x} = 0$, for $k = 1, 3, 5, \dots, 19$, i.e., calculate

$$|f(x) - t^k(x; \bar{x})|,$$

for each k . Put the results in a table and comment.

Problems 2 (Gradient Descent)

In this problem we consider the bivariate function $f(x) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$.

1. Define a function that computes $f(x)$ for any $x \in \mathbb{R}^2$.
2. Plot the graph of f in 2D using contour plots on $[-2, 2] \times [-2, 2]$, then “zoom in” on the minimum and determine its minimum value and optimal variables visually. Hint: use the `contour` function in `matplotlib.pyplot`.
3. Plot the graph of f in 3D using contour surface plots. Hint: use the `plot_surface` function in `matplotlib.pyplot`.
4. Find the analytical form of the gradient of f , $\nabla f(x)$, and define a function `grad_f` that computes $\nabla f(x)$ for any $x \in \mathbb{R}^2$.
5. Write a function that implements the gradient descent algorithm. This function should take as input
 - a function f and its gradient ∇f
 - a step size parameter α

- a starting point x_{start} .
- a maximum number of iterations max_{iter} (default value: 1000).
- a tolerance level ϵ for the norm of the gradient (default value: 10^{-6}).

Moreover this function should return a list containing all iterates of the gradient descent algorithm.

6. Run the gradient descent algorithm for the function f , starting from $x^{(0)} = (0.5, 2)^T$ and using a step size $\alpha = 0.001$. Print the last iterate, plot the trajectory of iterates and comment the results.