Assignment 2

Numerical Optimization / Optimization for CS WS2023

Lea Bogensperger, lea.bogensperger@icg.tugraz.at
Alexander Falk, alexander.falk@icg.tugraz.at
Clemens Krenn, clemens.krenn@student.tugraz.at
Fabian Ofner, f.ofner@student.tugraz.at

November 14, 2023

Deadline: Dec 5$^{\text{th}}$, 2023 at 12:00

Submission: Upload your report and your implementation to the TeachCenter. Please use the provided framework-file for your implementation. Make sure that the total size of your submission does not exceed 50MB. Include **all** of the following files in your submission:

- `report.pdf`: This file includes your notes for the individual tasks. Keep in mind that we must be able to follow your calculation process. Thus, it is not sufficient to only present the final results. You are allowed to submit hand written notes, however a compiled LaTeX document is preferred. In the first case, please ensure that your notes are well readable.
- `main.py`: This file includes your python code to solve the different tasks. Please only change the marked code sections. Also please follow the instructions defined in `main.py`.
- `figures.pdf`: This file is generated by running `main.py`. It includes a plot of all mandatory figures on separate pdf pages. Hence, you do not have to embed the plots in your report.

1 Lagrange Multiplier Problem (15 P.)

Given the following constrained optimization problems over $\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2$, compute the optimal solution using Lagrangian formulations.

a)
$$\min_{\mathbf{x}} -x_1 + 2x_2, \quad \text{s.t.} \quad x_1 = -\frac{1}{2}x_2^2 + 3, \quad \frac{1}{2}x_2 \geq -3x_1 + 5,$$

b)
$$\min_{\mathbf{x}} \frac{1}{3}x_1^2 + 3x_2, \quad \text{s.t.} \quad \frac{1}{x_2}(x_1^2 - 3) = 1, \quad \frac{1}{2}x_1 - 1 \leq x_2,$$

c)
$$\min_{\mathbf{x}} \frac{1}{2}(2x_1^2 + x_1x_2 + 4x_2^2), \quad \text{s.t.} \quad x_2 = \frac{1}{2}x_1 - 1, \quad x_1^2 \leq -x_2 + 3.$$

In Python:

1. Plot the level sets of the above functions using a contour plot[1]
2. Draw the equality-/inequality-constraints on top as shown in the example provided in the code framework.
3. Add markers to all candidate points that you computed with Pen & Paper. Specifically, mark the points that violate the KKT conditions and highlight the overall optimal solution with different colors.

With Pen & Paper:

[1] `https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.contour.html`

4. Formulate the Lagrangian equation. As shown in the exercise.
5. Formulate the KKT conditions.
6. Discuss every case with respect to the given constraints. Compute all possible solutions and identify valid ones.
7. Show which solution is the optimal one.

2 Optimization of Glider's Trajectory (10 P.)

The lift-to-drag ratio is an important measure in aerodynamics for the design of an aircraft. It quantifies its efficiency by providing the ratio of the lift force generated by an aircraft to the drag force that opposes its motion through air resistance. To model this in an optimization problem we express the drag coefficient and lift coefficient by $x_1 \in \mathbb{R}^+\backslash\{0\}$ and $x_2 \in \mathbb{R}^+$, respectively, yielding the following objective function:

$$\max_{x_1\in\mathbb{R}^+\backslash\{0\},x_2\in\mathbb{R}^+} f(x_1,x_2) := \frac{x_2}{x_1}. \tag{1}$$

The behaviour of the lift-to-drag ratio is governed by the so-called "drag polar" which is represented by a function $h(x_1, x_2)$. Neglecting some physical technicalities, we can use a very simplistic model for the drag polar using constants $a, b, c \in \mathbb{R}$

$$h(x_1, x_2) = a + b(x_2 - c)^2 - x_1. \tag{2}$$

In Python:

1. Plot the level sets of the objective function in (1) using a filled contour plot. Select 10 meaningful level sets $\{\alpha_i\}_{i=1}^{10}$ that you want to plot and include a colorbar[2].
2. Additionally, include the constraint denoting the "drag polar" in the same plot by using a contour plot to plot the level set $\alpha = 0$.
3. Graphically determine the (approximate) location of the optimum (x_1^*, x_2^*) and mark it in your plot. Report the (graphically) determined optimum (x_1^*, x_2^*), evaluate its function value and check the deviation from the constraint (this deviation is expected, it does not have to be entirely accurate).

With Pen & Paper:

4. Formulate the corresponding Lagrangian $\mathcal{L}(x_1, x_2, \lambda)$.
5. Analytically compute the gradient of the Lagrangian $\nabla\mathcal{L}(x_1, x_2, \lambda)$.
6. Using $a = 1, b = \frac{1}{2}, c = 0$, solve the system for x_1^*, x_2^*, λ^*. You can use a solver[3] to compute the roots of an equation when it is not easily feasible by hand anymore. Make sure to state all possible solutions and decide which one is feasible.
7. Graphically add the analytic (exact) optimum (x_1^*, x_2^*) in your plot. State the analytically determined optimum (x_1^*, x_2^*) in your report, evaluate its function value and check whether the constraint is fulfilled.
8. Analytically (do not plug in the solution here) show that the gradient of the objective function in (1) is orthogonal to the tangent line of $h(x_1^*, x_2^*) = 0$.

[2] https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.colorbar.html
[3] https://www.wolframalpha.com/