Dr. Peter Arndt

Master for AI and Data Science
Universität Düsseldorf

Math for AI
Winter term 2023/24

Exercise sheet 6

For some of these exercises visualization may be helpful. Here is some code that shows how to plot level sets. For practice maybe try to draw the set of feasible points and some level sets of the objective function by hand first.

Exercise 20 (11 points)

Consider the optimization problem

$$\begin{array}{rc} \text{minimize} & (x-3)^2+(y-1)^2 \\ \text{subject to} & y \geq x^2+1 \\ & y \leq -3x+11 \end{array}$$

Write down the Lagrange function and the KKT conditions. Find all pairs $((x,y),\lambda)$ that satisfy the KKT conditions and find out which one(s) attain the minimum.

[Remark: You can go trough all the conditions and find out the points. Alternatively you can save yourself some work by checking whether the problem is convex, visually guessing the right subcase of the complementary slackness conditions and using what you learn on Tuesday: For a convex optimization problem the KKT conditions imply optimality.

At some point you might encounter the polynomial $4x^3+2x-6=(x-1)(4x^2+4x-6)$. You can use without further calculations that $x=1$ is its only root.]

Exercise 21 (12 points)

Consider the optimization problem

$$\begin{array}{rl} \text{minimize} & f(x,y) := -4xy+3x^2+2x+4y \\ \text{subject to} & (x-1)^2 \leq 4 \\ & y \geq 0 \end{array}$$

(a) (2 points) Show that the set of feasible points is convex.

(b) (1 point) Write down the Lagrangian and the KKT conditions.

(c) (4 points) Find all pairs $((x,y),\lambda)$ that satisfy the KKT conditions.

(d) (1 points) Is one of the points (x,y) belonging to a KKT pair $((x,y),\lambda)$ an optimal point? Justify your answer.

(e) (1 points) Is the objective function convex? Justify your answer.

[Hint: On Tuesday we will see that for a convex optimization problem the KKT conditions imply optimality.]

(f) (3 points) State the dual problem and show that it has no feasible points.

[Remark: The dual problem will be introduced on Tuesday. Dual feasibility also requires that the dual function $g(\lambda,\nu)$ is defined, i.e. that the infimum in question exists.]

Exercise 22 (9 points)

Let $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^n$. For the Euclidean norm $\| - \|_2$ on $\mathbb{R}^n$ consider the unconstrained optimization problem

$$\text{minimize} \quad \|Ax - b\|_2$$

Since this problem is unconstrained, the dual function is constant with value p^* (think about why this is true!), so the dual problem won't help us here.

Now consider the related problem

$$\begin{array}{rc} \text{minimize} & \|z\|_2^2 \\ \text{subject to} & Ax - b = z \end{array}$$

[Remark: For this to make sense, you have to consider the objective function as a function $\mathbb{R}^{n+m} \to \mathbb{R}$, $(x_1, \dots, x_m, z_1, \dots, z_n) \mapsto \|z\|_2$ If $(x_1, \dots, x_m, z_1, \dots, z_n)$ is an optimal point for this new problem, the vector $(x_1, \dots, x_m)$ will be an optimal point for the previous problem.]

(a) (6 points) Show that the Lagrange dual function for this problem is given by

$$g\colon \operatorname{dom} g \to \mathbb{R}, \qquad \nu \mapsto -\frac{1}{4}\|\nu\|_2^2 - b^T\nu$$

with $\operatorname{dom} g = \{\nu \mid \nu^T A = 0\} \subseteq \mathbb{R}^n$.

[Remark: The domain of the dual function consists of the points where the infimum in question exists. It is helpful here to write the summand of the Lagrangian coming from the constraints as $\nu^T(Ax - b - z)$ for a vector ν of the right size.]

(b) (3 points – slightly tricky, do at your own risk) Now consider the problem

$$\begin{array}{rc} \text{minimize} & \|z\|_2 \\ \text{subject to} & Ax - b = z \end{array}$$

Show that the Lagrange dual function for this problem is given by

$$g\colon \operatorname{dom} g \to \mathbb{R}, \qquad \nu \mapsto -b^T\nu$$

with $\operatorname{dom} g = \{\nu \mid \nu^T A = 0 \text{ and } \|\nu\| = 1\} \subseteq \mathbb{R}^n$

Exercise 23 (8 points)

In this exercise you will get to know an optimization package for Python: Pyomo. First, you should follow these installation instructions. Use the conda installation option if you can, because with pip you need to install the `ipopt` solver package by hand. If you don't have conda, the Pyomo Workshop slides offered on this page start with very detailed installation instructions – beware: those slides come in a 110 MB pdf file.

And here is the problem (finally): Tasmanian Bully Bugs (which don't really exist) are insects which really smell badly. Even they themselves think so, and therefore they try to keep distant from each other. If you throw 50 of them into a box, they will distribute such that no two of them are very close.

More precisely, if the bugs are called $1, \ldots, 50$ and $d(i, j)$ is the distance between bugs number i and j, they will solve the optimization problem

$$\begin{array}{rc} \text{maximize} & \min\{d(i,j) \mid 1 \leq i,j, \leq 50,\ i \neq j\} \\ \text{subject to} & \text{“all bugs are in the box”} \end{array}$$

Look at the notebook in the exercise folder to see how one can solve this problem with Pyomo.

(a) (3 points) Now a diagonal wall is put into the box, from the upper left to the lower right corner, and the bugs are all thrown into the lower left half of the now divided box. Modify the given program to show how the bugs distribute now.

(b) (5 points) Now a cookie is thrown into the box. Tasmania is a remote place, and the bugs have never seen such a tasty looking thing! They all want to be close to the cookie, but still keep their distance from each other. In fact their personal space is 17 times more important to them than being close to the cookie, i.e. they want to solve the problem

$$\begin{array}{rc} \text{maximize} & 17 \cdot \min\{d(i,j) \mid 1 \leq i,j, \leq 50,\ i \neq j\} - \sum_{i=1}^{50} d(i, \text{cookie}) \\ \text{subject to} & \text{“all bugs are in the lower left triangle of the box”} \end{array}$$

Modify the given program to show how the bugs distribute now.

[Remarks: *1.* If you do both, parts (a) and (b), you don't have to document your solution of part (a) separately – just do it all and earn 8 points.
2. In part (b) the cookie can still fall anywhere in the box, also into the upper right triangle.]

Deadline: Friday 24th of November, 10:00.
Upload your solution to this link.