Homework #5

Due August 1 @ 11:59pm

Submission requirements

Upload a **single PDF or HTML file** of your IJulia notebook for this entire assigment. Clearly denote which question each section of your PDF corresponds to.

Problem 1 -- Nonconvex or convex?

Determine whether the following functions are convex, concave, or neither, over the given ranges of values. Hint: for parts 6 and 7, you should use the results from some of the previous parts and the fact that a sum of convex functions is convex, and a sum of concave functions is concave.

- $f(x) = x^{1/2}$ for $x \geq 0$
- $f(x) = xe^x$ for $x \geq 0$
- $f(x) = x^2$ for $x \in \mathbb{R}$ (all numbers)
- $f(x) = x^3$ for $x \in \mathbb{R}$ (all numbers)
- $f(x) = x^3$ for $x \geq 0$
- $f(x_1, x_2, x_3) = x_1^{1/2} - x_2 e^{x_2} + 4x_3$ for $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$.
- $f(x_1, x_2) = x_1^3 + x_2^2$ for $x_1 \geq 0, x_2 \geq 0$

For the following problem, determine whether or not it can be guaranteed that a local optimal solution will be a global optimal solution, and justify your conclusion. (You do not need to attempt to solve the model.)

$$\begin{aligned} \max \ & a x_1^{1/2} - b x_2 e^{x_2} + c x_3 \\ \text{subject to } & x_1 + 3x_2 + 2x_3 \leq 10 \\ & x_1^3 + x_2^2 \leq 20 \\ & x_1 \geq 0, \ x_2 \geq 0, x_3 \geq 0 \end{aligned}$$

Solution

- $f(x) = x^{1/2}$ for $x \geq 0$. $f'(x) = (1/2)x^{-1/2}$ and $f''(x) = -1/4x^{-3/2}$. Thus $f''(x) \leq 0$ for any $x \geq 0$, and so this function is concave for $x \geq 0$.
- $f(x) = xe^x$ for $x \geq 0$. $f'(x) = e^x + xe^x$ and $f''(x) = 2e^x + xe^x$. Thus $f''(x) \geq 0$ for $x \geq 0$ and so the function is convex for $x \geq 0$.
- $f(x) = x^2$ for $x \in \mathbb{R}$ (all numbers). $f'(x) = 2x$ and $f''(x) = 2$. Thus $f''(x) \geq 0$ for all x, and so the function is convex over all x.
- $f(x) = x^3$ for $x \in \mathbb{R}$ (all numbers). $f'(x) = 3x^2$ and $f''(x) = 6x$. Thus $f''(x)$ may be positive or negative depending on if x is positive or negative, and so this function is neither concave nor convex over all $x \in \mathbb{R}$.
- $f(x) = x^3$ for $x \geq 0$. Based on the last part, we know $f''(x) = 6x \geq 0$ for $x \geq 0$, and thus this function is convex over $x \geq 0$.
- $f(x_1, x_2, x_3) = x_1^{1/2} - x_2 e^{x_2} + 4x_3$ for $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$. $x_1^{1/2}$ is a concave function by part 1, and $x_2 e^{x_2}$ is convex by part 2 and so $-x_2 e^{x_2}$ is concave. As $4x_3$ is also concave, we conclude that the function is a sum of concave functions and hence is concave.
- $f(x_1, x_2) = x_1^3 + x_2^2$ for $x_1 \geq 0, x_2 \geq 0$. By part 4 x_1^3 is convex for $x_1 \geq 0$ and x_2^2 is convex for all x from part 3, thus the function is a sum of convex functions and hence is convex.

For the following problem, determine whether or not it can be guaranteed that a local optimal solution will be a global optimal solution, and justify your conclusion. (You do not need to attempt to solve the model.)

$$\begin{aligned} \max \ & a x_1^{1/2} - b x_2 e^{x_2} + c x_3 \\ \text{subject to } & x_1 + 3x_2 + 2x_3 \leq 10 \\ & x_1^3 + x_2^2 \leq 20 \\ & x_1 \geq 0, \ x_2 \geq 0, x_3 \geq 0 \end{aligned}$$

Solution

If a and b are nonnegative, the objective function is concave and we are maximizing, so a local solution will be a global solution if the feasible region is convex. If a and/or b is negative, the function is not concave and we cannot guarantee a global solution. c can take any value.

The feasible region consists of linear constraints, which always define a convex feasible region, plus the nonlinear inequality $x_1^2 + x_2^3 - 20 \leq 0$. We saw from the previous part that this function is convex over $x_2 \geq 0$, and since $x_2 \geq 0$ is part of the constraints, the function defining this constraint is convex. Thus, the feasible region is convex and we can conclude that a local optimal solution will be a global optimum.

Problem 2 -- NLP modeling

You have been contracted by Fizzy Water Inc (FWI) to determine the optimal locations of two distribution centers (A and B). FWI has 5 "customers" (stores where their drinks are sold). Every day, a number of delivery trucks must drive from the distribution centers to the customers. The locations of each customer and the total number of delivery trucks that should travel from each distribution center to each customer each day is given in the table below. Customer locations are given as coordinates on a 2-dimensional grid.

For purposes of this problem, measure the distance between each pair of locations as the *square* of the Euclidean distance between them (so we don't have to deal with the square root function!).

Customer	Location	Trucks from A	Trucks from B
1	(0,0)	0	6
2	(1,3)	4	0
3	(2,4)	1	1
4	(2,1)	1	2
5	(3,0)	2	4

For purposes of this problem, the distribution centers can be located anywhere in the $(x, y)-$plane.

(a) Formulate a nonlinear programming problem that will minimize the distribution costs (i.e. The weighted squared distances between distribution centers and customers).

Hint: You should not need any constraints on your model.

(b) Solve the instance given in part (a) to determine the optimal locations.

Solution 2a

(a) Formulate a nonlinear programming problem that will minimize the distribution costs (i.e. The weighted squared distances between distribution centers and customers).

Define the following variables:

- x_A & x-coordinate of location of distribution center A
- y_A & y-coordinate of location of distribution center A
- x_B & x-coordinate of location of distribution center B
- y_B & y-coordinate of location of distribution center B

Also (just for ease of notation), we will let the set of customers be C, and make the following parameter definitions $\forall c \in C$:

- χ_c & x-coordinate of customer c
- υ_c & x-coordinate of customer c
- α_c & Number of trucks that must travel from A to c
- β_c & Number of trucks that must travel from B to c

Then the following is a nonlinear programming problem that will minimize the distribution distances

$$minimize \sum_{c \in C} (\alpha_c (x_A - \chi_c)^2 + \alpha_c (y_A - \upsilon_c^2) + \beta_c (x_B - \chi_c)^2 + \beta_c (y_B - \upsilon_c)^2)$$

Solution 2b

(b) Solve the instance given in part (a) to determine the optimal locations.

Coordinates for Distribution Center A = (1.75,2.125)

Coordinates for Distribution Center B = (1.3846153846153846,0.4615384615384615)

In [1]:
```
using JuMP, Ipopt

cust = 1:5
x_coord_cust = Dict(zip(cust,[0 1 2 2 3]))
y_coord_cust = Dict(zip(cust,[0 3 4 1 0]))

demand_a = Dict(zip(cust,[0 4 1 1 2]))
demand_b = Dict(zip(cust,[6 0 1 2 4]))

m = Model(Ipopt.Optimizer)

@variable(m, xa)
@variable(m, ya)
@variable(m, xb)
@variable(m, yb)

@objective(m, Min, sum(demand_a[i]*((xa-x_coord_cust[i])^2 +(ya-y_coord_cust[i])^2) for i in cust)
                  + sum(demand_b[i]*((xb-x_coord_cust[i])^2 +(yb-y_coord_cust[i])^2) for i in cust))

set_silent(m)
optimize!(m)

println("optimal location of dc A: (", value(xa), ", ", value(ya), ")")
println("optimal location of dc B: (", value(xb), ", ", value(yb), ")")
```

```
******************************************************************************
This program contains Ipopt, a library for large-scale nonlinear optimization.
 Ipopt is released as open source code under the Eclipse Public License (EPL).
         For more information visit https://github.com/coin-or/Ipopt
******************************************************************************

optimal location of dc A: (1.75, 2.125)
optimal location of dc B: (1.3846153846153846, 0.4615384615384615)
```

Problem 3 -- Geometric Programming

Fizzy Water, Inc. is designing a pipe that disinfects their water before it is packaged. The pipe is designed to heat the water to T (degrees above ambient temperature). The pipe has a fixed length and circular cross section with radius r. The pipe is wrapped in a layer of material with thickness w (much smaller than r) for insulation to minimize heat loss. The design variables for the pipe are T, r, and w.

Heat loss can be converted to energy cost and is roughly equal to $(\alpha_1 T r)/w$ for some constant α_1. The cost of the pipe is approximately proportional to the total material used: $\alpha_2 r$ for a constant α_2. The insulation cost, similarly,is porportional to the total insulation material used, or roughly $\alpha_3 r w$ for a constant α_3. The total cost of constructing and using the pipe is the sum of these three costs.

Fizzy Water wants to maximize the total heat flow through the pipe, which is proportional to the velocity of the water through the pipe. This value can be approximated by $\alpha_4 T r^2$ for a constant α_4. All α constants and all variables T, r, w are strictly positive.

The problem Fizzy Water wants you to help them solve is to maximize the total heat flow through the pipe, subject to their budget $C_{\max}$ which is an upper bound on total cost. They also have the following constraints:

- $T_{\min} \leq T \leq T_{\max}$
- $r_{\min} \leq r \leq r_{\max}$
- $w_{\min} \leq w \leq w_{\max}$
- $w \leq 0.1r$

(a) Express this problem as a geometric program, and convert it into a convex optimization problem.

(b) Consider a simple instance of this problem where $C_{\max} = 500$ and all α_i values equal 1. Assume for simplicity that every variable (T, r, w) has a lower bound of 1 and no upper bound. Solve this problem using JuMP. Use the Ipopt solver (or any other solver that can solve convex optimization problems) and the "@NLconstraint(...)" command to specify any nonlinear constraints. What are the optimal T, r, and w?

Solution 3a

(a) Express this problem as a geometric program, and convert it into a convex optimization problem.

(Nonconvex) geometric program is:

$$\begin{aligned} \min \ & [\alpha_4 T r^2]^{-1} \\ \text{s.t. } & \frac{\alpha_1 T r}{C_{\max} w} + \frac{\alpha_2}{C_{\max}} r + \frac{\alpha_3}{C_{\max}} r w \leq 1 \\ & 0 \leq \frac{T}{T_{\max}} - \frac{T_{\min}}{T_{\max}} \leq 1 \\ & 0 \leq \frac{r}{r_{\max}} - \frac{r_{\min}}{r_{\max}} \leq 1 \\ & 0 \leq \frac{w}{w_{\max}} - \frac{w_{\min}}{w_{\max}} \leq 1 \\ & 10w/r \leq 1 \\ & T, r, w > 0 \end{aligned}$$

Define $x = \log T$, $y = \log r$, and $z = \log w$. These are all free variables.

Rewrite the model using these identities, and take the log of every constraint and objective:

$$\begin{aligned} \min \ & \log(1/\alpha_4) - x - 2y \\ \text{s.t. } & \log(\sum_{i=1}^{\alpha_1} e^{x+y-z} + \sum_{i=1}^{\alpha_2} e^y + \sum_{i=1}^{\alpha_3} e^{y+z}) \leq \log(C_{\max}) \\ & \log(T_{\min}) \leq x \leq \log(T_{\max}) \\ & \log(r_{\min}) \leq y \leq \log(r_{\max}) \\ & \log(w_{\min}) \leq z \leq \log(w_{\max}) \\ & \log(10) + z - y \leq 0 \end{aligned}$$

(b) Consider a simple instance of this problem where $C_{\max} = 500$ and all α_i values equal 1. Assume for simplicity that every variable (T, r, w) has a lower bound of 1 and no upper bound. Solve this problem using JuMP. Use the Ipopt solver (or any other solver that can solve convex optimization problems) and the "@NLconstraint(...)" command to specify any nonlinear constraints. What are the optimal T, r, and w?

Solution

Rewrite the model with the given data:

$$\begin{aligned} \min \ & -x - 2y \\ \text{s.t. } & \log(e^{x+y-z} + e^y + e^{y+z}) \leq 6.2146 \\ & 0 \leq x \\ & 0 \leq y \\ & 0 \leq z \\ & z - y \leq -\log(10) \end{aligned}$$

Solution 3b

(b) Consider a simple instance of this problem where $C_{\max} = 500$ and all α_i values equal 1. Assume for simplicity that every variable (T, r, w) has a lower bound of 1 and no upper bound. Solve this problem using JuMP. Use the Ipopt solver (or any other solver that can solve convex optimization problems) and the "@NLconstraint(...)" command to specify any nonlinear constraints. What are the optimal T, r, and w?

Temp: 23.840239437612354 Radius: 46.390428092691536 Width: 4.639042794473825

In [3]:
```
using JuMP, Ipopt

m = Model(Ipopt.Optimizer)

@variable(m, x >= 0)
@variable(m, y >= 0)
@variable(m, z >= 0)

@objective(m, Min, -x - 2y)

@NLconstraint(m, log(exp(x+y-z) + exp(y) + exp(y+z)) <= log(500))
@constraint(m, z-y <= -log(10))

set_silent(m)
optimize!(m)

println("Temp: ", exp(value(x)))
println("Radius: ", exp(value(y)))
println("Width: ", exp(value(z)))
```

```
Temp: 23.840239437612354
Radius: 46.390428092691536
Width: 4.639042794473825
```