MTH 451/EGR 551 - Homework 7
Due Monday April 12, 2021 at 5 p.m.
Please submit all computer codes you developed to solve these problems.

Consider the equality constrained optimization problem

$$\mathbf{EC}: \min_{x \in \mathcal{R}^n} f(x) \text{ subject to } h_j(x) = 0,\ j = 1, 2, \cdots, m.$$

A penalty function method that can be used to solve this problem is described as follows.

- Define the penalty function $\phi(x; \alpha_k) = f(x) + \frac{1}{2\alpha_k} \sum_{j=1}^{m} h_j(x)^2$, where $\alpha_k > 0$ is a scalar.
- Let x^k be the minimum of $\phi(x; \alpha_k)$. Therefore, x^k satisfies the necessary conditions

$$0 = \nabla \phi(x^k; \alpha_k) = \nabla f(x^k) + \frac{1}{\alpha_k} \sum_{j=1}^{m} \nabla h_j(x^k) h_j(x^k).$$

- Introduce the variables $\lambda_j^k = h_j(x^k)/\alpha_k$, so that the necessary conditions for a minimum can be written as the system of equations

$$F(q^k; \alpha_k) = \begin{bmatrix} \nabla f(x^k) + \nabla h(x^k) \lambda^k \\ h(x^k) - \alpha_k \lambda^k \end{bmatrix} = 0, \qquad q^k = \begin{bmatrix} x^k \\ \lambda^k \end{bmatrix} \in \mathcal{R}^{n+m},$$

  with $\nabla h(x^k) \in \mathcal{R}^{n \times m}$, $\lambda^k = [\lambda_1^k,\ \lambda_2^k, \ldots, \lambda_1^m]^T \in \mathcal{R}^m$.

- A minimum of **EC** is then determined by solving the $(n+m)$ equations $F(q^k; \alpha_k) = 0$ for the $(n+m)$ unknowns q^k, with a monotone decreasing sequence $\{\alpha_k\}_{k=1}^{\infty}$ such that $\alpha_k > 0$ and $\lim_{k \to \infty} \alpha_k = 0$.

1. Consider the equality constrained optimization problem $\min_{x_1, x_2} f(x)$, where

$$f(x) = \frac{1}{2} \left((x_1 - a)^2 + (x_2 - b)^2 \right)$$

   subject to

$$h(x) = x_1 + x_2 - c = 0.$$

   Here a, b and c are constants.

   (a) Using $f(x)$ and $h(x)$ derive the system of equations

$$F(x, \lambda; \alpha) = \begin{bmatrix} \nabla f(x) + \nabla h(x) \lambda \\ h(x) - \alpha \lambda \end{bmatrix} = 0.$$

   [*Hint: You should get three equations involving x_1, x_2, λ, α, a, b and c.*]

   (b) Solve the equations in (a) to determine x_1, x_2 and λ in terms of α, a, b and c.

   (c) Determine the values x_1, x_2 and λ in the limit as $\alpha \to 0$.

Consider the equality constrained optimization problem

$$\mathbf{EC}: \ \min_{x \in \mathcal{R}^n} f(x) \text{ subject to } h_j(x) = 0, \ j = 1, 2, \cdots, m.$$

An **augmented Lagrangian** penalty function method that can be used to solve this problem is described as follows.

Algorithm 0.1 Augmented Lagrangian Algorithm

Input: $f(x)$, $h(x)$, x^0, λ^0, $\alpha_0 > \alpha_1 > \ldots > \alpha_N > 0$, $\epsilon_t > 0$
Output: x, λ such that $\|\nabla f(x) + \nabla h(x)\lambda\| \leq \epsilon_t$ and $\|h(x)\| \leq \epsilon_t$
1: Define $\theta(x; \lambda, \alpha) = f(x) + \lambda^T h(x) + \frac{1}{2\alpha} h(x)^T h(x)$
2: **for** $k = 0, 1, \ldots, N$ **do**
3: Set $\alpha = \alpha_k$, $\lambda = \lambda^k$
4: Using initial estimate x^k solve $\min_x \theta(x; \lambda, \alpha)$ to get x^{k+1}
5: $\lambda^{k+1} = \lambda^k + \frac{1}{\alpha_k} h(x^{k+1})$
6: **if** $\|\nabla f(x^{k+1}) + \nabla h(x^{k+1})\lambda^{k+1}\| \leq \epsilon_t$ **and** $\|h(x^{k+1})\| \leq \epsilon_t$ **return** $x = x^{k+1}$
7: **end for**

Recall that the Lagrangian function is defined as $\mathcal{L}(x, \lambda) = f(x) + \lambda^T h(x)$, where $\lambda \in \mathcal{R}^m$ are the Lagrange multipliers associated with the equality constraints.

The Augmented Lagrangian penalty function is thus the Lagrangian *augmented* with the quadratic penalty term, i.e., $\theta(x; \lambda, \alpha) = \mathcal{L}(x, \lambda) + \frac{1}{2\alpha} h(x)^T h(x) = f(x) + \lambda^T h(x) + \frac{1}{2\alpha} h(x)^T h(x)$.

The Augmented Lagrangian Algorithm obtains a solution to the problem **EC** by minimizing a sequence of unconstrained functions defined by $\theta(x; \lambda, \alpha)$. In this unconstrained minimization problem λ and α are fixed. The penalty parameter α is selected from a monotone decreacing sequence of values. The Augmented Lagrangian Algorithm has a significant advantage over the classical penalty function algorithm, in that, the Augmented Lagrangian Algorithm can converge to a solution of **EC** without requiring that $\alpha \to 0$.

The main iteration loop is defined on lines 2-7. On line 3 assigns the Lagrange multiplier estimate λ^k and the penalty parameter α_k. On line 4 the unconstrained minimization problem is solved using x^k as an initial estimate. The result of the minimization is x^{k+1}. (Note that the implementation `fmin_BFGS` can be used to solve the unconstrained minimization problem.)

An update of the Lagrange multiplier is set on line 5. This update is easily justified by noting that

$$\begin{aligned} 0 &= \nabla \theta(x^{k+1}; \lambda^k, \alpha_k), \\ &= \nabla f(x^{k+1}) + \nabla h(x^{k+1}) \left(\lambda^k + \frac{1}{\alpha_k} h(x^{k+1}) \right), \\ &= \nabla f(x^{k+1}) + \nabla h(x^{k+1}) \lambda^{k+1}, \\ &= \nabla \mathcal{L}(x^{k+1}, \lambda^{k+1}). \end{aligned}$$

The algorithm terminate on line 6 if the gradient of the Lagrangian and the constraint violation are sufficiently small.

2. Implement the Augmented Lagrangian Algorithm using MATLAB/Octave. Use your implementation to solve the following problems.

   (a) $f(x) = \frac{1}{2}\left((x_1 - a)^2 + (x_2 - b)^2\right)$, $h(x) = x_1 + x_2 - c$, $a = 3$, $b = -3$, $c = 3$ and $x^0 = [10; 10]$.

   (b) $f(x) = \exp(x_1 x_2 x_3 x_4 x_5)$, $h_1(x) = 2x_1 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10$, $h_2(x) = x_2 x_3 - 5x_4 x_5$, $h_3(x) = x_1^3 + x_2^3 + 1$ and $x^0 = [-1; -1; -1; -1; -1]$.

   For each problem use $\epsilon_t = 10^{-4}$, print out the number of iterations required for each α, the optimal value for x and the Lagrange multipliers λ.