MAA209 Numerical Optimization

École Polytechnique

Practical Session #6

Instructions: The codes should be written in Python. The use of notebooks is encouraged and you may start from the given examples. **Remove irrelevant cells from the Notebooks** if you start from a code given on Moodle. **Write comments** regarding your work and the experiments you did with the code. **Remove plotting** if the dimension is larger than two.

Upload your work on Moodle (one or multiple notebooks). **Solve one of the exercises** for a full activity point. Students who, in addition, solve correctly parts from more than one exercises will get bonus points. Students solving one of the Exercises 3 or 4 will automatically get bonus points.

Before doing the exercises:

Recall the discussion in Course 3 regarding the **projected gradient algorithm**!

Observe the implementation of the Projected Gradient algorithm given on Moodle in order to find the minimizers for the function $f(x) = \frac{1}{2}x^T Ax - b^T x$, where $A = \begin{pmatrix} 1 & 0.4 \\ 0.4 & 2 \end{pmatrix}$ and $b = (1, 1)^T$ under the following constraints constraints:

1. the solution lies in a ball: $x \in B((0, 1), 1.5) = \{x : |x - (0, 1)| \leq 1.5\}$
2. the solution lies in a box: $x \in [-0.3, 0.5] \times [0.3, 1.5]$.
3. the solution lies on a circle: $x \in C((0, 1), 1.5) = \{x : |x - (0, 1)| = 1.5\}$

You may modify the constraints and observe the behavior of the algorithm. Note that the algorithm also converges in the case where the set representing the constraints is not convex.

Exercise 1 Optimization under constraints

Consider again $f(x) = \frac{1}{2}x^T Ax - b^T x$, where $A = \begin{pmatrix} 1 & 0.4 \\ 0.4 & 2 \end{pmatrix}$ and $b = (1, 1)^T$ for the constraint $x \in C((0, 1), 1.5) = \{x : |x - (0, 1)| = 1.5\}$, which can be modeled by the equation

$$0 = g(x) = x^2 + (y - 1)^2 - 1.5^2$$

1. Find the solution of the above problem using Lagrange multipliers.
2. Apply the penalization method in order to find the minimizers of the function defined above under the constraint $g(x) = 0$. In the course it was shown that in order to solve the problem

   $$\min_{g(x)=0} f(x) \qquad (P_0)$$

   one may minimize instead the function

   $$f(x) + \frac{1}{\varepsilon} g(x)^2 \qquad (P_\varepsilon)$$

   for some $\varepsilon > 0$. Notice that the problem (P_ε) is unconstrained. Use an optimization algorithm of your choice to approximate its minimizer for $\varepsilon \in \{10^{-1}, 10^{-2}, 10^{-3}\}$.
3. **(Challenge: The Miklmaid Problem)** Recall the milkmaid problem shown in the course: given two points A, B on the same side of the river R given by the equation $g(x) = 0$ find the point $C \in R$ such that $AC + CB$ is minimal.
   Use the penalization method in order to find the solution when $g : \mathbb{R}^2 \to \mathbb{R}$ is given by

   $$g(x) = x_2 - 0.4 \sin(x_1),$$

   and the points are given by $A(0, -5)$ and $B(10, -5)$. Verify that the optimality condition is verified at the optimum (the gradient of the objective function and the gradient of the constraint are colinear).

Exercise 2 Optimization under Constraints (2)

1. Solve the problem
$$\min_{x_1+...+x_n=1}(x_1^2+...+x_n^2) \tag{1}$$
using Lagrange multipliers.
2. What is the projection operator on the constraint $x_1+...+x_n=1$? (See Course 3).
3. Solve the same problem numerically using the projected gradient algorithm. An implementation of the projected gradient algorithm is already available on Moodle.
4. Suppose $f:\mathbb{R}^n\to\mathbb{R}$. Suppose, moreover, that the constraint set K is given by
$$K=\{x:Ax=b\},$$
where A is a $m\times n$ matrix ($m<n$) of full rank and $b\in\mathbb{R}^m$. Note that the projected gradient algorithm is given by
$$x_{i+1}=P_K(x_i-\gamma\nabla f(x_i)). \tag{2}$$
Denote by $K_0=\{x:Ax=b\}$, the kernel of A. Prove that the previous algorithm is equivalent to
$$x_0\in K, x_{i+1}=x_i-\gamma P_{K_0}(\nabla f(x_i)), \tag{3}$$
i.e. it is enough to start from a point in K and project the gradient on K_0 instead.
The advantage of the latter version is the fact that (3) **does not need access to the optimization code** since the projection is done in the computation of the gradient, **provided by the user**. Therefore (3) may be implemented using an already existing optimization algorithm from `scipy.optimize`, while (2) cannot.
5. Implement (3) and test it using the algorithms from Scipy viewed in Session 5.

Exercise 3 Eigenvalue computation (Challenge)

1. The objective of this part is to find the eigenvalues of a symmetric positive-definite matrix using iterative optimization algorithms. In order to do this, we recall below some useful characterizations. In the following, denote by $0<\lambda_1\leq...\lambda_n$ the n eigenvalues of a symmetric positive definite matrix A. Then the first eigenvalue is given by the optimization problem with constraint
$$\lambda_1=\min_{|u|=1}u^TAu.$$
Suppose that the first $k<n$ eigenvalues $\lambda_1\leq...\leq\lambda_k$ are known and have corresponding eigenvectors $v_1,...,v_k$. Then the $k+1$-th eigenvalue can be obtained by minimizing u^TAu under the normalization constraint $|u|=1$ and the orthogonality constraints $u\cdot v_k=0$:
$$\lambda_{k+1}=\min_{\substack{|u|=1\\ u\cdot v_i=0,i=1...k}}u^TAu.$$
   (a) Recall how to compute the eigenvalues and eigenvectors of a matrix using `numpy.linalg`.
   (b) What is the projection on the set $K_1=\{x\in\mathbb{R}^n:|x|=1\}$? Implement the projected gradient algorithm in order to find the first eigenvalue λ_1. Compare your result with the one given by `numpy.linalg.eigvals`.
   (c) Recall from the course what is the projection on the set $K_2=\{x\in\mathbb{R}^n:x\cdot v_i=0, i=1,...,k\}$. Describe how to compute the projection on $K_1\cap K_2$. Implement the projected gradient algorithm in order to find the second eigenvalue of the matrix, once you've found the first eigenvalue and its corresponding eigenvector.
   Hints: The set K_2 can be characterized by an equality of the type $Mx=0$ where the matrix M contains the eigenvectors $v_1,...,v_k$ as lines. Use the formula
$$P_{K_2}(y)=y-M^T(MM^T)^{-1}My.$$
   Use the fact that the eigenvectors $v_1,...,v_k$ are orthogonal to simplify the above formula (compute MM^T).
   (d) Can the algorithm handle the case of multiple eigenvalues?

(e) Do you need to run the optimization algorithm in order to find the last eigenvalue λ_n?

(f) **(Challenge)** Construct a Python function which returns the first n_0 smallest eigenvalues of an $n \times n$ symmetric positive definite matrix A. The inputs should be the matrix A and the number n_0 of requested eigenvalues. Test the algorithm for some simple examples in order to validate it. Then try your code for some larger matrices: you may choose diagonal matrices or some symmetric matrices which are diagonally dominant. Make sure to have a reasonable condition number for the gradient algorithm to converge reasonably fast. Compare your results with the ones given by `numpy.linalg.eigvals`.

Note: Recall that the choice of the initialization may influence the results in the search for the minimizers, especially in this case where the problem changes for each eigenvalue. You may choose a random initial condition before each optimization by using the command `numpy.random.rand(N)` where `N` is the size of your vector.

Exercise 4 Linear Programming (Challenge)

This exercise is related to the last session, but you might find it interesting. Consider the following problems:

P1. *A store sells two types of toys, A and B. The store owner pays $8 and $14 for each one unit of toy A and B respectively. One unit of toys A yields a profit of $2 while a unit of toys B yields a profit of $3. The store owner estimates that no more than 2000 toys will be sold every month and he does not plan to invest more than $20,000 in inventory of these toys. How many units of each type of toys should be stocked in order to maximize his monthly total profit profit?*

P2. *A farmer plans to mix two types of food to make a mix of low cost feed for the animals in his farm. A bag of food A costs $10 and contains 40 units of proteins, 20 units of minerals and 10 units of vitamins. A bag of food B costs $12 and contains 30 units of proteins, 20 units of minerals and 30 units of vitamins. How many bags of food A and B should the consumed by the animals each day in order to meet the minimum daily requirements of 150 units of proteins, 90 units of minerals and 60 units of vitamins at a minimum cost?*

P3. *John has $20,000 to invest in three funds F1, F2 and F3. Fund F1 is offers a return of 2% and has a low risk. Fund F2 offers a return of 4% and has a medium risk. Fund F3 offers a return of 5% but has a high risk. To be on the safe side, John invests no more than $3000 in F3 and at least twice as much as in F1 than in F2. Assuming that the rates hold till the end of the year, what amounts should he invest in each fund in order to maximize the year end return?*

1. Identify the variables and write a linear programming problem which solves the question **P1**.
2. Plot the objective function on the constraint set and identify the solution visually.
3. Solve the problem using `scipy.optimize.linprog`. (See the examples available for Session 7).
4. Do the same thing for the **P2**.