Instructions: (Please read carefully and follow them!)

Try to solve all problems on your own. If you have difficulties, ask the instructor or TAs.

In this session, we will start with implementation of algorithms for solving nonlinear optimization problems. Today, we will discuss gradient descent to solve problems of the form $\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$.

Gradient descent is one of the oldest algorithms (dating back to Cauchy), yet simple and popular in many scientific communities even today. Gradient descent works iteratively where in each iteration a suitable descent direction and an appropriate step length are chosen to update the current iterate.

The implementation of the optimization algorithms in this lab will involve extensive use of the `numpy` Python package. It would be useful for you to get to know some of the functionalities of `numpy` package. For details on `numpy` Python package, please consult https://numpy.org/doc/stable/index.html

For plotting purposes, please use `matplotlib.pyplot` package. You can find examples in the site https://matplotlib.org/examples/.

Please follow the instructions given below to prepare your solution notebooks:

- Please use different notebooks for solving different Exercise problems.
- The notebook name for Exercise 1 should be `YOURROLLNUMBER_IE684_Lab1_Ex1.ipynb`.
- Similarly, the notebook name for Exercise 2 should be `YOURROLLNUMBER_IE684_Lab1_Ex2.ipynb`, etc.
- Please post your doubts in MS Teams Discussion Forum channel so that TAs can clarify.

There are only 2 exercises in this lab. Try to solve all problems on your own. If you have difficulties, ask the Instructors or TAs.

Only the questions marked [**R**] need to be answered in the notebook. You can either print the answers using `print` command in your code or you can write the text in a separate text tab. To add text in your notebook, click `+Text`. Some questions require you to provide proper explanations; for such questions, write proper explanations in a text tab. Some questions require the answers to be written in LaTeX notation. Please see the demo video to know how to write LaTeX in Google notebooks. Some questions require plotting certain graphs. Please make sure that the plots are present in the submitted notebooks.

After completing this lab's exercises, click File → Download .ipynb and save your files to your local laptop/desktop. Create a folder with name `YOURROLLNUMBER_IE684_Lab1` and copy your `.ipynb` files to the folder. Then zip the folder to create `YOURROLLNUMBER_IE684_Lab1.zip`. Then upload only the `.zip` file to Moodle. There will be extra marks for students who follow the proper naming conventions in their submissions.

Please check the **submission deadline announced in moodle.**

The first Laboratory exercise aims to help you learn the plotting of the level sets of the function and implementation of **optimization algorithms** such as zero order, first order (**Gradient Descent**) and second order (**Newton's method**).

Exercise 1 (15 marks) *Plot the good-looking visuals of level sets for each of the following functions in (1) to (4)-:*

1. $f(\mathbf{x}) = f(x_1, x_2) = 100x_2^2 + 100x_1^4 - 200x_2x_1^2 + x_1^2 - 2x_1 + 1$
2. $f(\mathbf{x}) = f(x_1, x_2) = x_1^4 + x_2^4 - 20x_1^3 - 20x_2^3 + 100x_1^2 + 100x_2^2$
3. $f(\mathbf{x}) = f(x_1, x_2) = -\cos(x_1)\cos(x_2)\exp(-((x_1 - \pi)^2 + (x_2 - \pi)^2))$
4. $f(\mathbf{x}) = f(x_1, x_2) = 2x_1^2 - 1.05x_1^4 + \frac{x_1^6}{6} + x_1x_2 + x_2^2$
5. *What do you observe from the level sets so obtained in (1) to (4) ?, Are they useful in characterizing the optimal solution visually, give reasoning via finding minima of each of the functions in (1) to (4) using usual first and second order optimality conditions.*

Note-: *Please use the theory references wisely related to the level sets, you need to explore the equivalent Python commands over the internet. Please use the axes range at least -4 to 4 and also choose levels in log or linear space wisely depending on particular function.*

Exercise 2 (20 marks) *In zero-order oracle optimization, the focus is solely on utilizing function values without access to derivatives. The goal is to find the optimal solution and optimal function value within a specified rectangle or box by iteratively evaluating the function within this region. The Algorithm 1 is corresponding to the zero-order oracle optimization:*

```
Algorithm 1 Zero Order Oracle Optimization
Require: $\varepsilon, L > 0$ and search space $S$
Choose grid resolution = $\frac{2\varepsilon}{L}$ and let $X :=$ set of all grid points on chosen search space $S$
Evaluate $f(x)$ for all $x \in X$
Output: $\widetilde{x} = \arg\min_{x \in X} f(x)$
```

The example for search space S being, $S = \{(x, y) : -4 \leq x \leq 4, -4 \leq y \leq 4\}$ for any two variable function, one can choose search space of their choice. The "grid resolution" refers to the distance between adjacent grid points on the search space. It is essentially a measure of how finely the optimization domain is discretized. The grid resolution is determined by the choice of ϵ, L.

1. *Implement the **Algorithm 1** for $f(x) = \sqrt{x^2 + 5}$, in $S = \{x : -\frac{A}{10} - 2 \leq x \leq \frac{A}{10} + 2\}$, where A is the last digit of your roll number. Take $L = 1$ and $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-8}, 10^{-10}\}$. Report the values of $\widetilde{x}$ corresponding to each of the ϵ. Comment on the observations. Comment about the objective function values obtained for different choices of the ϵ at $\widetilde{x}$.*
2. *Implement the **Algorithm 1** for $f(\mathbf{x}) = f(x_1, x_2) = \sqrt{x_1^2 + x_2^2}$, in $S = \{(x_1, x_2) : -\frac{A}{10} - 2 \leq x_1 \leq \frac{A}{10} + 2, -\frac{A}{5} - 2 \leq x_2 \leq \frac{A}{5} + 2\}$, where A is the last digit of your roll number. Take $L = 1$ and $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-8}, 10^{-10}\}$. Report the values of $\widetilde{\mathbf{x}}$ corresponding to each of the ϵ. Comment on the observations. Comment about the objective function values obtained for different choices of the ϵ at $\widetilde{\mathbf{x}}$. Plot the level set of this function showing the obtained minima.*
3. *What do you observe from (1) and (2), write a general observation related to (1) and (2) and any other general function. Also find the cardinality of the set of all grid points on chosen search space S in terms of ϵ, L assuming any general d variable function, what does this represent ?*

4. *What may be the shortcomings of this algorithm and suggests a possible solution to deal with it ?*

Exercise 3 (20 marks) ***Gradient descent*** *is one of the oldest algorithms (dating back to Cauchy), yet simple and popular in many scientific communities even today. Gradient descent works iteratively where in each iteration a suitable descent direction and an appropriate step length are chosen to update the current iterate.*

We will start with a procedure which helps to find a minimizer of the function $f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n$.

We will use the following gradient descent type algorithm:

```
Algorithm 2 Gradient Descent Procedure with Constant Step Length
Require: Starting point $\mathbf{x_0}$, Tolerance level $\tau$, Step length $\eta$
1: Initialize $k = 0$
2: while $\|\nabla f(\mathbf{x_k})\|_2 > \tau$ do
3:     $\mathbf{x_{k+1}} \leftarrow \mathbf{x_k} - \eta \nabla f(\mathbf{x_k})$
4:     $k \leftarrow k + 1$
5: Output: $\mathbf{x_k}$
```

1. *What is the minimizer and minimum function value of $f(\mathbf{x}) = f(x_1, x_2) = (a + 1 - x_1)^2 + b \cdot (x_2 - x_1^2)^2$?, Where a is the last digit of your Roll Number and b is 100 if a is an even number, else b is 10. Use these values a, b for other problems that follows.*
2. *With the starting point $x_0 = (-1.5, 1.5)$ and $\eta = 0.001$, we aim to analyze the behavior of the algorithm 2 for different tolerance values. We set $\tau = 10^{-p}$ where $p = 1, 2, \ldots, 13$. For each τ, record the final minimizer, objective function value at termination, and the number of iterations required for convergence in a tabular form. Generate a plot, illustrating the relationship between the number of iterations and τ values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the tolerance values.*
3. *Plot the level sets of the function in (1) and also plot the trajectory of the optimization on the same plot and report your observations. "In optimization, a trajectory refers to the path or sequence of points that a numerical optimization algorithm traverses while iteratively updating the solution in search of an optimal point".*
4. *What may be the shortcomings of this algorithm and suggests a possible solution to deal with it ?*

Exercise 4 (25 marks) ***Newton's method****, inspired by Isaac Newton's work, is a classic optimization approach. It iteratively updates the current solution by calculating a direction to descend and an optimal step size. This process relies on second-order information, specifically the Hessian matrix. Despite its complexity,Newton's method is widely used and valued in various scientific communities.*

We will start with a procedure which helps to find a minimizer of the function $f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n$.

We will use the following Newton's Method type of algorithm:

```
Algorithm 3 Newton's Method
Require: Starting point $\mathbf{x_0}$, Stopping tolerance $\tau$
1: Initialize $k = 0$
2: while $\|\nabla f(\mathbf{x_k})\|_2 > \tau$ do
3:     $\mathbf{x_{k+1}} \leftarrow \mathbf{x_k} - (\nabla^2 f(\mathbf{x_k}))^{-1} \nabla f(\mathbf{x_k})$
4:     $k \leftarrow k + 1$
5: Output: $\mathbf{x_k}$
```

1. *What is the minimizer and minimum function value of* $f(\mathbf{x}) = f(x_1, x_2) = (a + 1 - x_1)^2 + b \cdot (x_2 - x_1^2)^2$ *?, Where* $\frac{b-10}{10}$ *is the last digit of your Roll Number and* a *is* 1 *if* b *is an even number, else* a *is* 2*. Use these values* a, b *for other problems that follows.*

2. *Is the minimizer in (1) unique ?, Is it local or global minima ?, Is the function* $f(\mathbf{x})$ *convex ?, explain each of them.*

3. *With the starting point* $x_0 = (-1.5, 1.5)$*, we aim to analyze the behavior of the algorithm 3 for different tolerance values. We set* $\tau = 10^{-p}$ *where* $p = 1, 2, \ldots, 20$*. For each* τ*, record the final minimizer, objective function value at termination, and the number of iterations required for convergence in a tabular form. Generate a plot, illustrating the relationship between the number of iterations and* τ *values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the tolerance values. Plot the level sets of the function in (1) and also plot the trajectory of the optimization on the same plot and report your observations. "In optimization, a trajectory refers to the path or sequence of points that a numerical optimization algorithm traverses while iteratively updating the solution in search of an optimal point".*

4. *Redo (3) by implementing the algorithm 2 (With* $\eta = 0.001$*) instead of algorithm 3 on the same function* $f(\mathbf{x})$ *of (1). What do you observe ?, Compare algorithm 2 and algorithm 3 based on the results you got on this function* $f(\mathbf{x})$ *of (1).*

5. *What may be the shortcomings of this algorithm and suggests a possible solution to deal with it ?*

Bibiography

(1) Y. Nesterov, Introductory lectures on convex optimization: A basic course. Springer Science Business Media, 2003, vol. 87.
(2) https://people.eecs.berkeley.edu/ jiantao/227c2022spring/material.html