**Topic:** Optimization

**Concepts:** gradient vs. non-gradient-based methods for unconstrained optimization problems

***Note:*** *Feel free to use resources from the internet to help you write the python script, find commands, or understand concepts. You may use the following:*
*https://docs.scipy.org/doc/scipy/tutorial/optimize.html*
*https://pygad.readthedocs.io/en/latest/*
*https://matplotlib.org/stable/gallery/mplot3d/surface3d.html*
*https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.contour.html*

## Problem-1:

Minimize the following Rosenbrock function with respect to x and y

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2$$

**(1)** Use scipy.optimize with an algorithm of your choice from the available options to solve the above unconstrained optimization problem. Change the initial guess and see if the solution changes. Did you find a local minimum or global miminimum? (Hint: plot the function in both 2D & 3D to understand the nature of the function)

**(2)** Solve the above problem using genetic algorithms (pygad.ga). Change the initial guess and see if the solution changes. Did you find a local minimum or global miminimum?

**(3)** Compare the performance of the two methods (number of iterations, time taken, solution obtained for the same accuracy (decimal points) level).

## Problem-2:

Minimize the following function with respect to x and y.

$$f(x, y) = (x^2 + y^2 - 1)^2 + (x - 1)^2 + y^2$$

(1) Use scipy.optimize with an algorithm of your choice from the available options to solve the above unconstrained optimization problem. Change the initial guess and see if the solution changes. Did you find a local minimum or global miminimum? (Hint: plot the function in both 2D & 3D to understand the nature of the function)

(2) Solve the above problem using genetic algorithms (pygad.ga). Change the initial guess and see if the solution changes. Did you find a local minimum or global miminimum?

(3) Compare the performance of the two methods (number of iterations, time taken, solution obtained for the same accuracy (decimal points) level).