Home Assignment – 3

Aarav Aryaman, 220012

Problem:

Write a computer program (preferably in python) to minimize the following function using steepest descent method

$$f(x_1, x_2) = ({x_1}^2 + x_2 - 11)^2 + ({x_2}^2 + x_1 - 7)^2$$

Use ${x_1}^{(0)} = {x_2}^{(0)} = 0$

Solution converges when norm of descent direction is less than 0.001

Do not use a constant step size, rather use an appropriate line search technique to find the optimum step size. Plot the function as a surface and as contours. Show the path of convergence connecting the points $({x_1}^{(k)},\ {x_2}^{(k)})$ for $k = 0, 1, 2, 3, \ldots.$

Solution Algorithm:

1. **Define Objective Function and Gradient**:
    - Objective Function is defined as given in the problem statement.

      $$f(x_1, x_2) = ({x_1}^2 + x_2 - 11)^2 + ({x_2}^2 + x_1 - 7)^2$$

    - Gradient Function is defined to determine the direction of steepest descent.

$$\nabla f(x_1, x_2) = [4x_1({x_1}^2 + x_2 - 11) + 2({x_2}^2 + x_1 - 7), 2({x_1}^2 + x_2 - 11) + 4x_2({x_2}^2 + x_1 - 7)]$$

2. **Line Search Function**:
    - This function determines an appropriate step size (or learning rate) for the steepest descent method.
    - It performs a backtracking line search to ensure that the step size is suitable for decreasing the objective function.
    - Start with an initial *stepsize* (1 in this case) and decrease it by a factor of *tau* = 0.5 until the condition for sufficient decrease is satisfied.
    - The condition checked is:

      $$f(x - stepsize * gradient) \leq f(x) - \beta * stepsize * ||gradient||^2$$

      where β is a small constant (0.1 in this case), ensuring the step size provides a sufficient decrease in the function value.

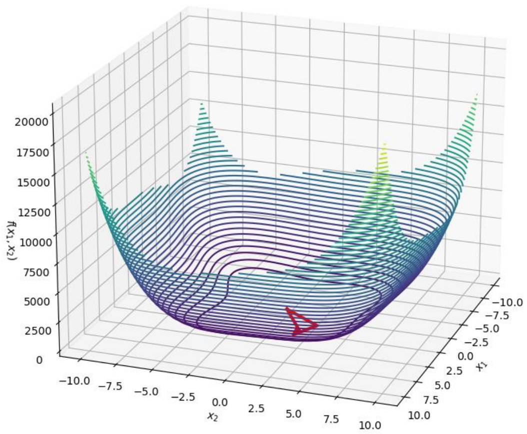
3. **Steepest Descent Method**:
    - Set up the parameters: maximum iterations (*maxit* = 10000), convergence tolerance (*epsilon* = 0.001) and initial value (x = [0, 0]).
    - For each iteration:

- Store the current values of x_1 and x_2 for later plotting.
- Compute the gradient at the current point.
- Check the norm of the gradient (b). If it's less than the tolerance (*epsilon*), stop the iteration as the solution is sufficiently close to the minimum.
- Use the line search function to determine the optimal step size.
- Update the current point x by moving in the direction of the negative gradient scaled by the step size.

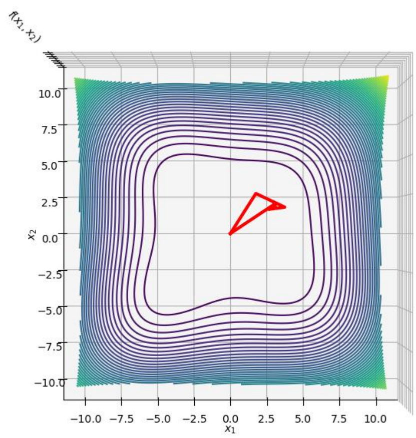
- If the gradient's norm (b) is below *epsilon*, the algorithm stops early, assuming convergence. Otherwise, it continues until reaching the maximum number of iterations.

4. **Output Results**:
   - Optimum stepsize: 0.015625
   - Minimum value: 7.368947568261478e-09
   - Minimum location: [2.99999944 1.99997953]
   - Iteration: 16

5. **Plotting**:
   - Create a mesh grid for x_1 and x_2 over a specified range.
   - Calculate the objective function values over this grid.
   - Plot the 3D surface of the objective function and overlay the path taken by the steepest descent method in red.
   - Contour is similar to the 3D plot but viewed from above to show the trajectory of the optimization in the x_1- x_2 plane.



Surface Plot



Contour Plot