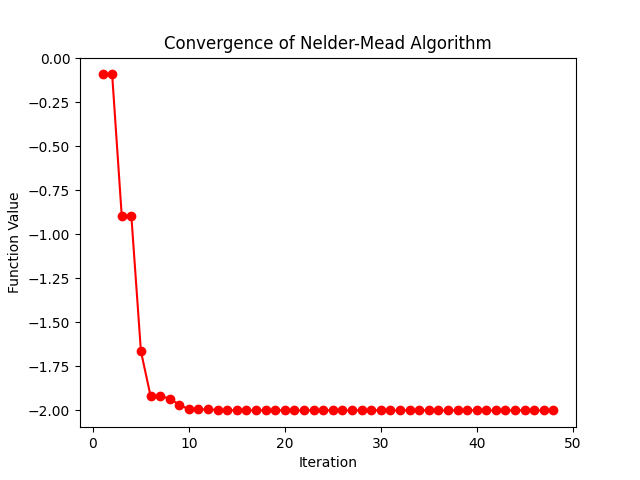
**ME 575- Assignment #6 Gradient-Free Optimization**

**6.1: Below are the contour and convergence plots for my Nelder-Mead function starting at the point [0.5 0.5].**

**A diagram of an egg carton function

AI-generated content may be incorrect.**

**6.2: Below is a table that explicitly shows the comparison of my Nelder-Mead Function to the Scipy library implementation.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column 1** | **Start Point** | **End Point** | **Function Value** | **Function Calls** |
| **My NM** | **[0.5, 0.5]** | **[-2.8e-09 1.1e-09]** | **-2** | **125** |
| **Scipy NM** | **[0.5, 0.5]** | **[ 3.35e-09 -1e-09]** | **-2** | **125** |
| **My NM** | **[-1, 1]** | **[1.34e-09 2e+00]** | **-1.57** | **125** |
| **Scipy NM** | **[-1, 1]** | **[-2e+00 -1.4e-09]** | **-1.57** | **137** |
| **My NM** | **[4, -3]** | **[ 4.097 -2.04]** | **0.145** | **94** |
| **Scipy NM** | **[4, -3]** | **[ 4.097 -2.05]** | **0.145** | **93** |
| **My NM** | **[10, -10]** | **[ 2.048 -8.19]** | **5.29** | **138** |
| **Scipy NM** | **[10, -10]** | **[ 10.2 -10.2]** | **19.4** | **85** |
| **My NM** | **[-4, 5]** | **[-4.097 4.097]** | **1.43** | **110** |
| **Scipy NM** | **[-4, 5]** | **[-4.097 4.097]** | **1.43** | **93** |

**6.3: Below is a table that describes the three cases in which different metrics in the genetic algorithm were changed.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Case** | **Population Size** | **Generations** | **Crossover** | **Mutation Level** | **Best Function Value** |
| **1 - Baseline** | **30** | **30** | **Tournament** | **0.1** | **-1.999936371** |
| **2- Less Generations** | **30** | **5** | **Tournament** | **0.1** | **-1.841102829** |
| **3- Higher Mutation** | **30** | **30** | **Tournament** | **2** | **-1.998140035** |
| **4- Smaller Population** | **2** | **30** | **Tournament** | **0.1** | **-1.570767728** |

**Case 1: Baseline**

**A colorful pattern with red dots

AI-generated content may be incorrect.**

**Case 2: Less Generations**

**A colorful pattern with circles

AI-generated content may be incorrect.**

**Case 3: Higher Mutation**

**A colorful pattern with a red dot

AI-generated content may be incorrect.**

**Case 4: Smaller population**

**A colorful pattern with red and blue circles

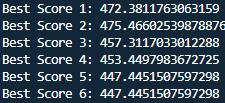
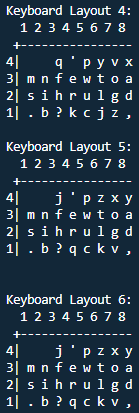
AI-generated content may be incorrect.**

**6.4:**

The goal of our project is to create a one-handed keyboard that is optimized to minimize the time it takes to type, maximize efficiency, and maximize ergonomics. For this homework section, we decided to work on minimizing the time it takes to type. The objective function computes the total distance traveled by all fingers when typing a given input string. We used a modified genetic algorithm (GA) to converge on the most efficient (minimum total distance traveled) keyboard layout. The key modification to the GA was defined in the crossover section. Instead of crossing over between two parents, we took the best attributes of the best individual – which was defined as the top 5 most frequent characters in the English language – and modeled all subsequent children with these best attributes fixed and everything else randomized. Then we repeated this process with each generation until all 30 keys were fixed. The first generation was completed with a population of 1,000,000 and all subsequent others with a population of 100,000.

Here are all 6 keyboard layouts and the best total distance traveled per generation:

A screenshot of a keyboard

AI-generated content may be incorrect.

Here is a plot of the convergence:

