Optimization of a Cost Function Using Differential Evolution in Python

1 Project Overview

Differential Evolution (DE) is a stochastic, population-based optimization algorithm used for solving complex, multi-dimensional optimization problems. This project aims to implement and analyze the DE algorithm for minimizing cost functions in Python.

2 Learning Objectives

The main objectives of this project are:

- Understand the working principles of the Differential Evolution algorithm.
- Implement DE in Python and test it on benchmark cost functions.
- Analyze the performance and convergence behavior of DE.
- Compare manual implementation with existing optimization libraries.

3 Methodology

3.1 Understanding Differential Evolution

The DE algorithm consists of three main steps:

- **Mutation**: Generating new candidate solutions.
- **Crossover**: Mixing solutions to explore the search space.
- **Selection**: Choosing the best candidates for the next generation.

3.2 Benchmark Cost Functions

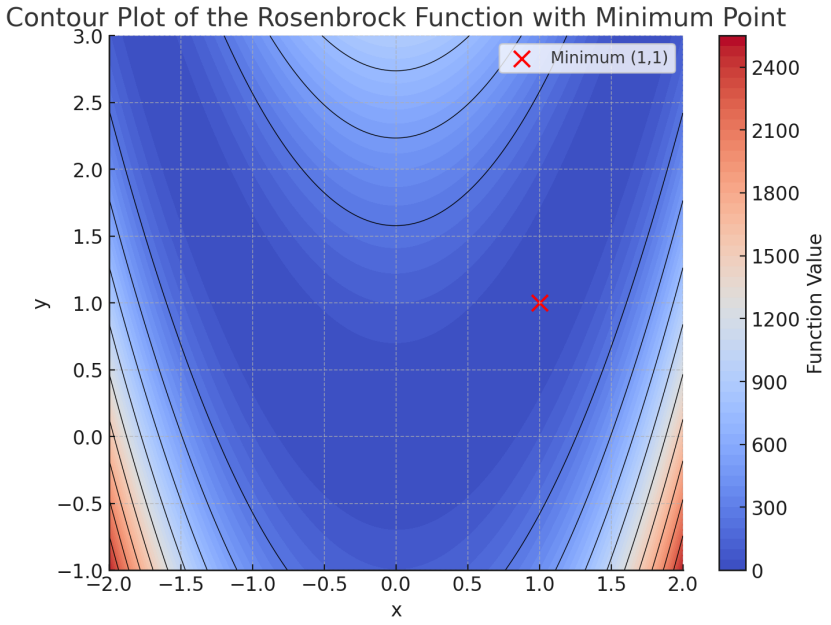
We test DE on different benchmark functions, such as:

1. **Sphere Function**:
$$f(x) = \sum_{i=1}^{n} x_i^2, \quad x \in \mathbb{R}^n \tag{1}$$
2. **Rosenbrock Function**:
$$f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 \tag{2}$$

3.3 Implementation in Python

The implementation includes:

- Writing a Python script for DE from scratch.
- Using the SciPy function `scipy.optimize.differential_evolution` for comparison.
- Running the algorithm on different cost functions.



3.4 Evaluation and Analysis

- Visualizing convergence using cost function plots.
- Studying the impact of parameters like mutation factor (F) and crossover probability (CR).
- Comparing performance across different functions.

4 Expected Outcomes

- A working Python implementation of Differential Evolution.
- Insights into parameter tuning and optimization efficiency.
- A comparison between manual and SciPy implementations.

5 Tools and Libraries

- Python (NumPy, Matplotlib, SciPy)
- Jupyter Notebook for visualization

6 Note:

Take as much time as you need, but don't use ChatGPT!

7 References

For a detailed description of the algorithm and its implementation, please refer to the paper by Storn, R., and Price, K.:

Storn, R., Price, K. *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces.* Journal of Global Optimization, 11, 341–359 (1997). https://doi.org/10.1023/A:1008202821328.