Assignment 5: Evolutionary Algorithms
Machine Learning

(Preliminary) Deadline: Sunday, 22 of December 2019, Time 23:59

Introduction

In this assignment, you will further deepen your understanding of Evolutionary Algorithms. Please provide a latex based report in the PDF format. We provide you a sample latex project you might use for writing and generating your report. If latex is new to you, we recommend using overleaf.

Your report, code and all generated files must be archived in a file named firstname.lastname and uploaded to the iCorsi website before the deadline expires. Late submissions will result in 0 points.

Where to get help

We encourage you to use the tutorials to ask questions or to discuss exercises with other students. However, do not look at any report written by others or share your report with others. Violation of that rule will result in 0 points for all students involved. For further questions you can send email to *xingdong@idsia.ch*.

Problems

In this assignment, we are going to implement a few evolutionary algorithms against test functions and analyze the characteristics of different algorithms.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a test function with n-dimensional domain.

- Sphere function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} x_i^2$$

- Rastrigin function:

$$f(\mathbf{x}) = An + \sum_{i=1}^{n} \left[x_i^2 - A\cos(2\pi x_i)\right]$$

  where $A = 10$

For simplicity, let us constraint the search domain as $x_i \in [-5, 5]$.

1. **Visualize test functions** (10 points)

   (a) Generate an 2D contour plot of 2-dimensional Sphere function (i.e. $n = 2$)

   (b) Generate an 2D contour plot of 2-dimensional Rastrigin function (i.e. $n = 2$)

   (c) For each test function, uniformly sample 100 points in the domain, evaluate them with the test function and guess what might be the region of the global optimum.

2. **Cross-Entropy Method (CEM)** (20 points)

   (a) Run CEM 3 times for both of test functions with 100-dimensional domain. (i.e. $n = 100$) Note that you can uniformly sample the initial population parameters as long as they are reasonably far from the global optimum.

   (b) Try different population size and elite set ratio and see what best performance you can obtain.

   (c) Try different number of generations. What is the minimum number of generations that you can obtain a solution close enough to the global optimum?

   (d) For each test function, plot the best and the worse fitness for each generation (averaged over 3 runs). Let x-axis be the generations and y-axis be the fitness values. Note that this should be one single figure with two curves (one for best fitness and another for worst fitness).

3. **Natural Evolution Strategy (NES)** (30 points)

   (a) Run NES 3 times for both of test functions with 100-dimensional domain. (i.e. $n = 100$) Note that you can uniformly sample the initial population parameters as long as they are reasonably far from the global optimum.

   (b) Try different population size and learning rate and see what best performance you can obtain.

   (c) Try different number of generations. What is the minimum number of generations that you can obtain a solution close enough to the global optimum?

   (d) For each test function, plot the best and the worse fitness for each generation (averaged over 3 runs). Let x-axis be the generations and y-axis be the fitness values. Note that this should be one single figure with two curves (one for best fitness and another for worst fitness).

4. **Covariance Matrix Adaptation Evolution Strategy (CMA-ES)** (30 points)

   (a) Run CMA-ES 3 times for both of test functions with 100-dimensional domain. (i.e. $n = 100$) Note that you can uniformly sample the initial population parameters as long as they are reasonably far from the global optimum.

(b) Try different population size and learning rate and see what best performance you can obtain.

(c) Try different number of generations. What is the minimum number of generations that you can obtain a solution close enough to the global optimum?

(d) For each test function, plot the best and the worse fitness for each generation (averaged over 3 runs). Let x-axis be the generations and y-axis be the fitness values. Note that this should be one single figure with two curves (one for best fitness and another for worst fitness).

5. **Benchmarking** (10 points)

(a) Plot the comparison of CEM, NES and CMA-ES for the best fitness in each generation. Note that this should be one single figure with three curves, each for one algorithm.

(b) Plot the comparison of CEM, NES and CMA-ES for the worst fitness in each generation. Note that this should be one single figure with three curves, each for one algorithm.

(c) For each test function, which algorithm is best? which is the worst?

Good luck !