

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
Кафедра програмної інженерії

ЗВІТ
Лабораторної роботи № 4
з дисципліни «Інтелектуальний аналіз даних»
на тему «Генетичні алгоритми»

Виконав

студент групи ІПЗм-24-2

Голодніков Дмитро

Перевірив

ст. викл. Онищенко К.Г.

Харків 2024

1 МЕТА РОБОТИ

Ознайомитися з генетичними алгоритмами як методом евристичної оптимізації. Вивчити основні оператори: селекцію, кросовер, мутацію. Застосувати генетичний алгоритм до оптимізації тестових функцій.

2 ХІД ВИКОНАННЯ РОБОТИ

2.1 Теоретичні основи

Генетичний алгоритм - метаевристичний метод оптимізації, що імітує процес природного відбору. Популяція рішень еволюціонує через оператори селекції, кросоверу та мутації. Реалізація базових операторів генетичного алгоритму:

```
# Ініціалізація популяції
def init_population(pop_size, dim, bounds):
    return np.random.uniform(bounds[0], bounds[1], (pop_size, dim))

# Турнірна селекція
def tournament_selection(population, fitness, tournament_size=3):
    indices = np.random.choice(len(population), tournament_size)
    best_idx = indices[np.argmin(fitness[indices])]
    return population[best_idx].copy()

# Одноточковий кросовер
def crossover(parent1, parent2, crossover_rate=0.8):
    if np.random.random() < crossover_rate:
        point = np.random.randint(1, len(parent1))
        child1 = np.concatenate([parent1[:point], parent2[point:]])
        child2 = np.concatenate([parent2[:point], parent1[point:]])
        return child1, child2
    return parent1.copy(), parent2.copy()

# Гаусова мутація
def mutate(individual, mutation_rate=0.1, mutation_scale=0.1):
    mask = np.random.random(len(individual)) < mutation_rate
    individual[mask] += np.random.normal(0, mutation_scale, mask.sum())
    return individual
```

2.2 Оптимізація функції Sphere

Функція Sphere $f(x) = \sum x_i^2$ - проста унімодальна функція з глобальним мінімумом у точці $(0, 0, \dots, 0)$.

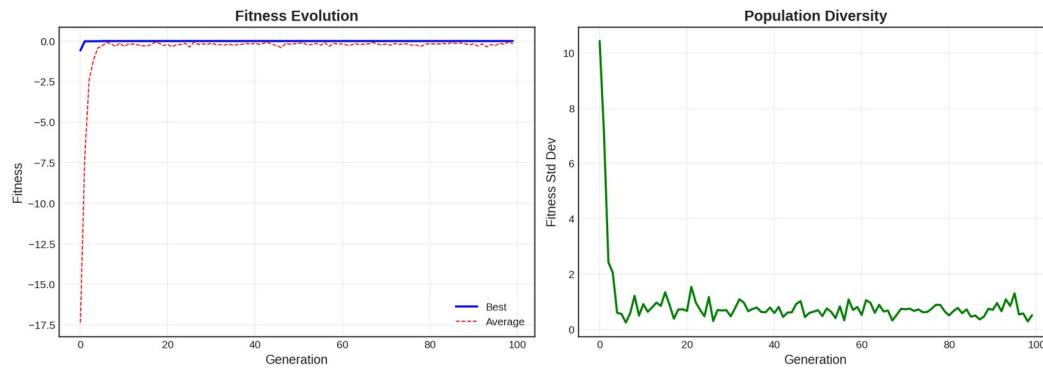


Рис. 2.1 - Еволюція оптимізації функції *Sphere*

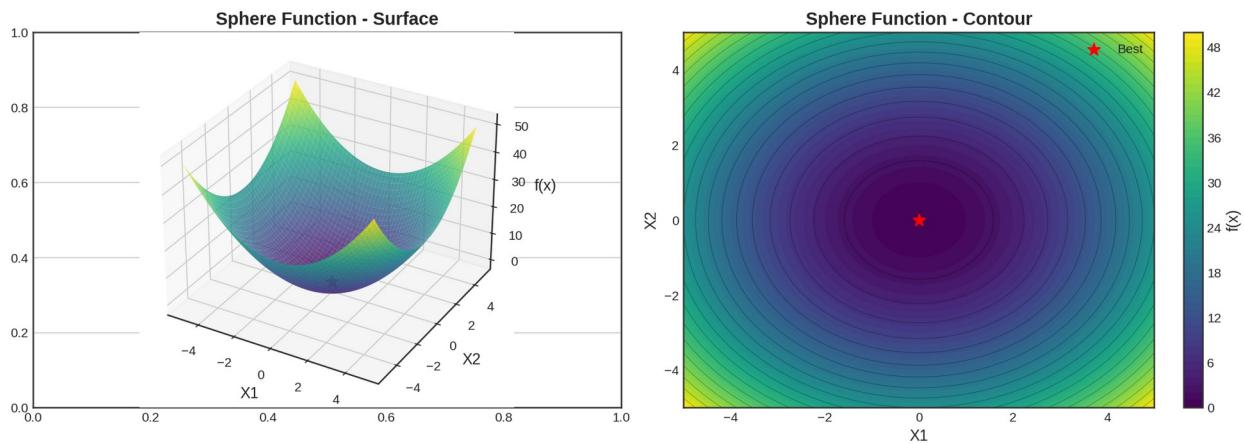


Рис. 2.2 - Поверхня функції *Sphere* та знайдений оптимум

2.3 Оптимізація функції Rastrigin

Функція Rastrigin - складна мультимодальна функція з багатьма локальними мінімумами. Це серйозний тест для оптимізаційних алгоритмів.

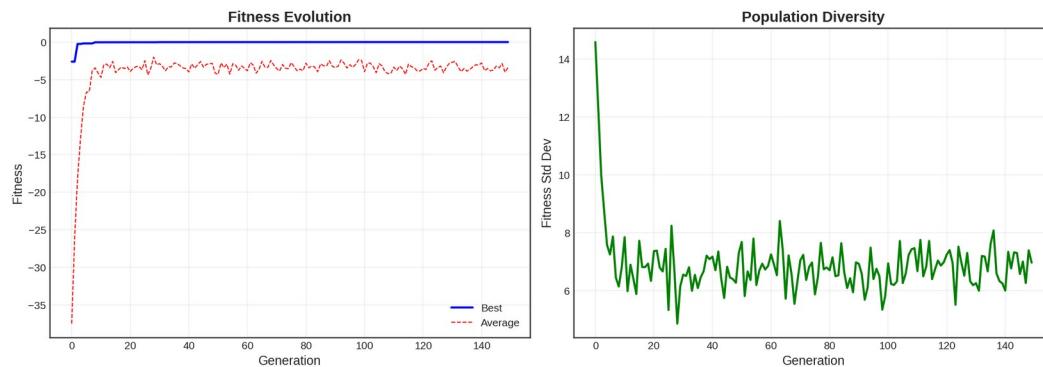


Рис. 2.3 - Еволюція оптимізації функції *Rastrigin*

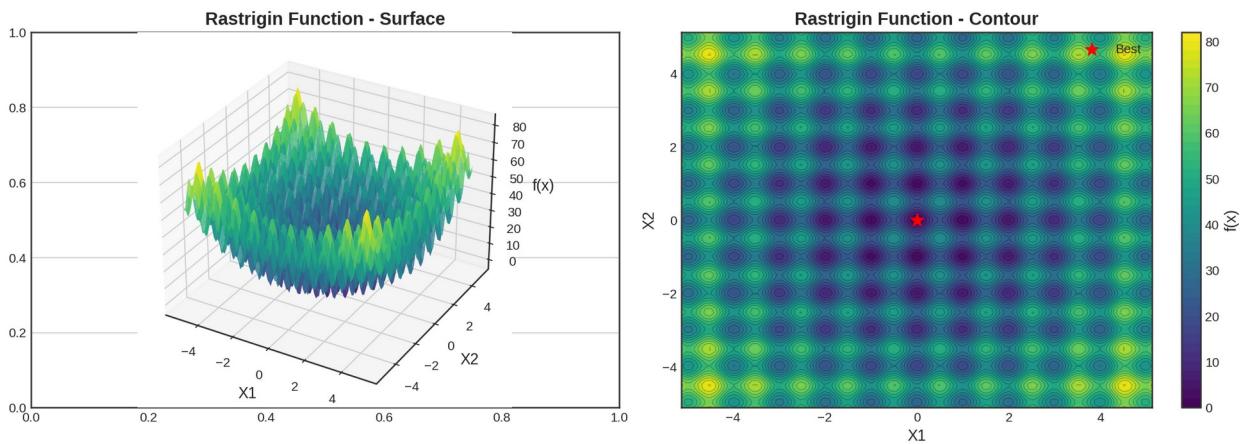


Рис. 2.4 - Поверхня функції Rastrigin

2.4 Оптимізація функції Rosenbrock

Функція Rosenbrock (banana function) має вузьку параболічну долину, що ускладнює пошук глобального мінімуму.

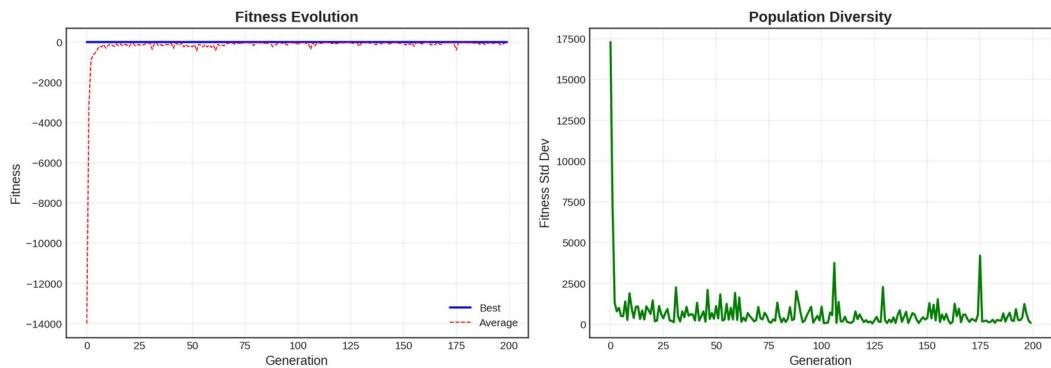


Рис. 2.5 - Еволюція оптимізації функції Rosenbrock

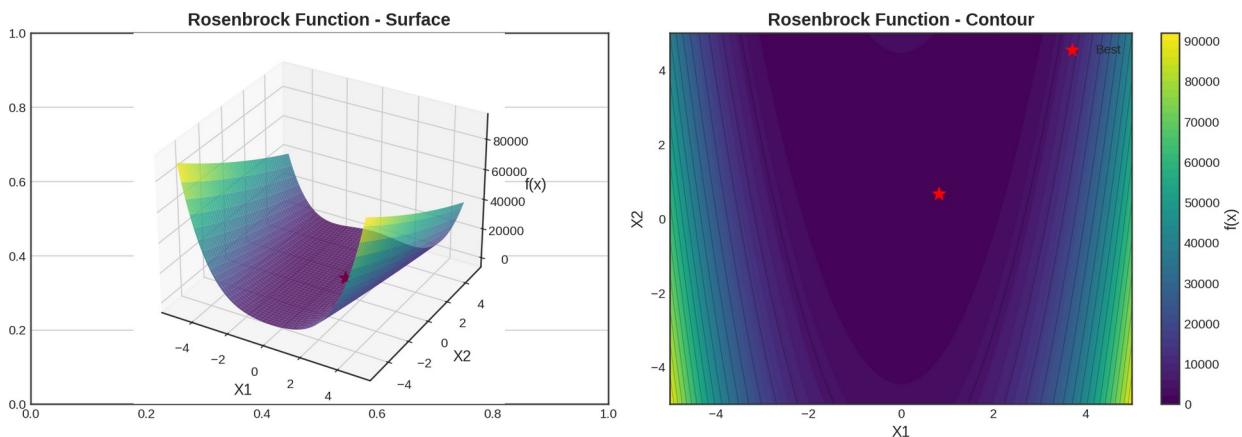


Рис. 2.6 - Поверхня функції Rosenbrock

2.5 Порівняння методів селекції

Порівняно три методи селекції: турнірна, рулеткова та рангова. Експеримент проведено на функції Sphere.

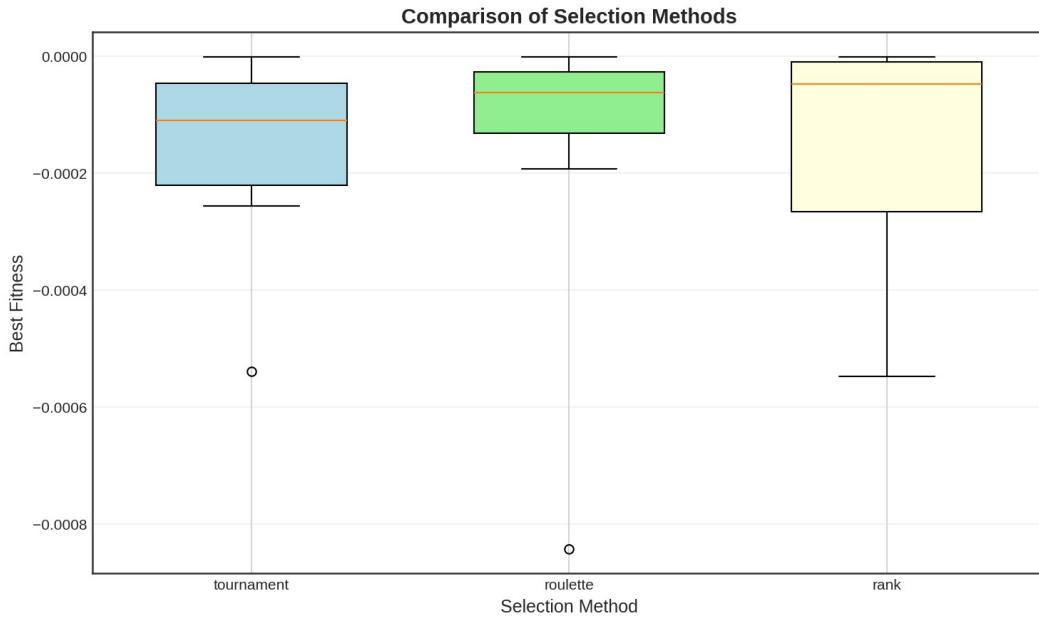


Рис. 2.7 - Порівняння методів селекції

2.6 Багатовимірна оптимізація

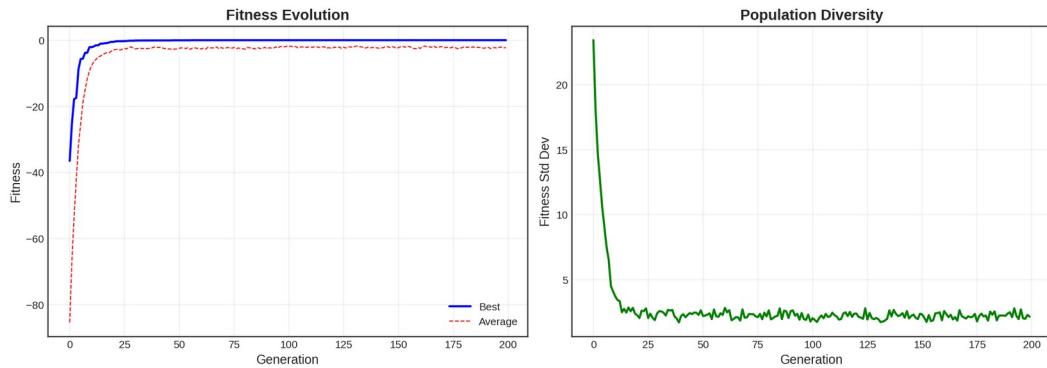


Рис. 2.8 - Оптимізація 10-вимірної функції Sphere

3 ВИСНОВКИ

У ході виконання лабораторної роботи було реалізовано генетичний алгоритм для задач неперервної оптимізації. Досліджено оптимізацію тестових функцій Sphere, Rastrigin та Rosenbrock. Порівняно ефективність різних методів селекції: турнірної, рулеткової та рангової. Проведено експерименти з

багатовимірною оптимізацією. Генетичні алгоритми показали здатність знаходити глобальні оптимуми складних функцій.

ПОСИЛАННЯ

Код проекту доступний у репозиторії GitHub: <https://github.com/na-naina/data-analysis-khnure>