

Проектування систем штучного інтелекту. Практична робота 5

Виконав студент групи ШІДМ-51 Тertiшний Владислав Юрійович

Тема:

Дослідження генетичних алгоритмів на задачі пошуку екстремумів функції за допомогою MATLAB.

Мета:

Вивчити основні принципи генетичних алгоритмів та придбати навички оптимізації функцій за допомогою MATLAB.

Функція:

$$z(x, y) = x \cdot e^{-(x^2 + y^2)}$$

Інтервал: $[-2; 2]$

Ціль: знайти мінімум.

Рішення

1. Створення m-файлу для цільової функції

```
function z = fitnessFunction(x)
    % Цільова функція для оптимізації
    z = x(1) * exp(-(x(1)^2 + x(2)^2));
end
```

2. Побудова графіка функції

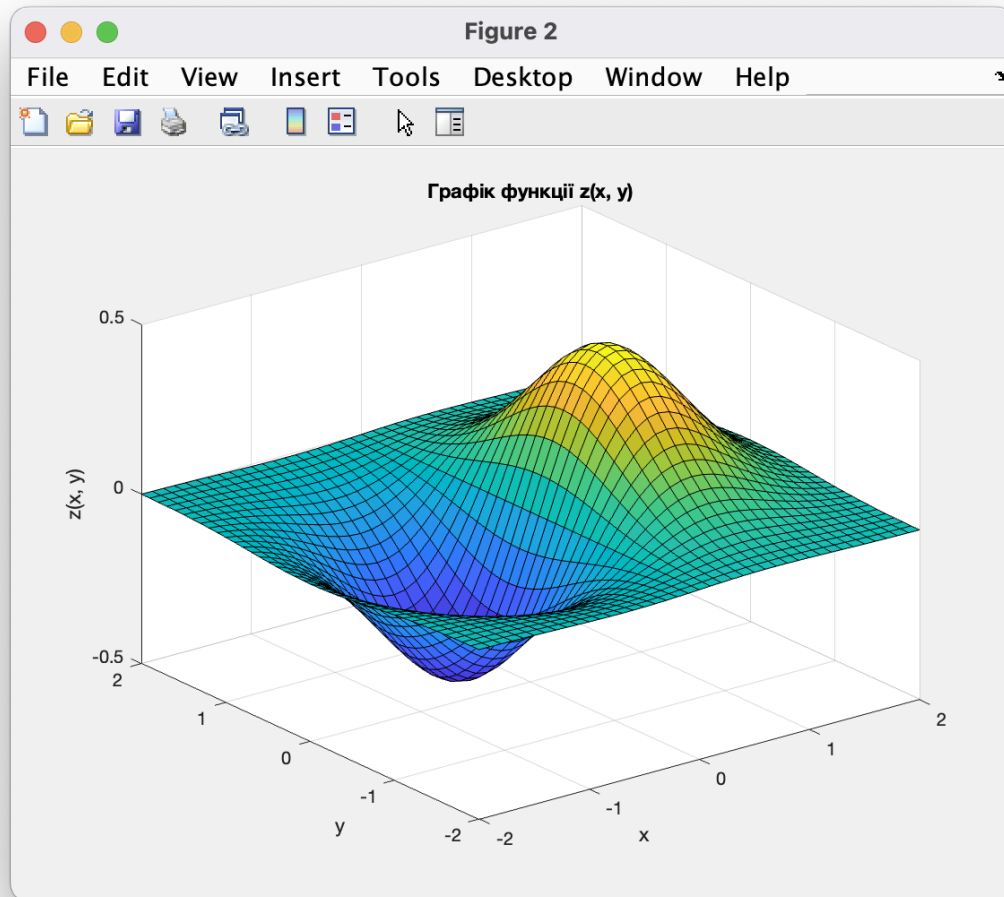
```
% Побудова графіка функції
[X, Y] = meshgrid(-2:0.1:2, -2:0.1:2);
Z = X .* exp(-(X.^2 + Y.^2));

% Відображення поверхні
figure;
surf(X, Y, Z);
title('Графік функції z(x, y)');
```

```

xlabel('x');
ylabel('y');
zlabel('z(x, y)');

```



3. Оптимізація за допомогою генетичного алгоритму

```
% Кількість змінних
```

```
nvars = 2;
```

```
% Налаштування параметрів ГА
```

```
options = optimoptions('ga', ...
```

```
    'PopulationSize', 50, ...           % Розмір популяції
```

```
    'MaxGenerations', 100, ...          % Кількість поколінь
```

```
    'CrossoverFraction', 0.8, ...       % Частка схрещувань
```

```
    'MutationFcn', @mutationgaussian, ... % Мутація Гаусса
```

```
    'SelectionFcn', @selectionroulette, ... % Рулетковий відбір
```

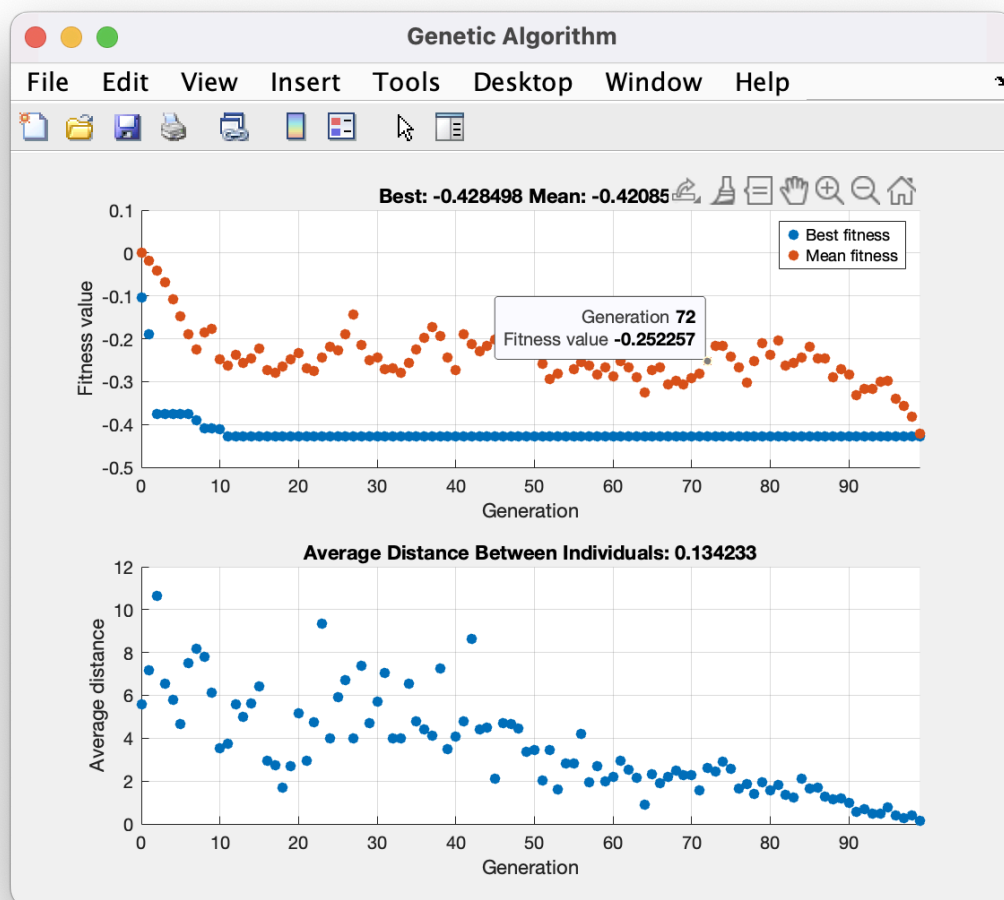
```

'PlotFcn', {@gaplotbestf, @gaplotdistance}); % Графіки

% Виклик генетичного алгоритму
[x, fval] = ga(@fitnessFunction, nvars, [], [], [], [], ...
    [-2, -2], [2, 2], [], options);

% Результати
disp('Оптимальні значення:');
disp(['x = ', num2str(x(1)), ', y = ', num2str(x(2))]);
disp(['Мінімум функції: ', num2str(fval)]);

```



```

%% ga stopped because the average change in the fitness value is less than options.FunctionTolerance.
Оптимальні значення:
x = -0.68646, y = -0.0058004
Мінімум функції: -0.4285
>> optimtool
>> pr5
>>

```

4. Оптимізація через GUI інтерфейс

MATLAB більше не підтримує старий інструмент `optimtool`, тому для вирішення задач оптимізації використовується новий, інтуїтивно зрозумілий підхід через **Optimize Live Editor Task**. Цей підхід дозволяє:

1. **Графічний інтерфейс:** легко налаштувати функцію та параметри.
2. **Зрозумілі результати:** відразу побачити оптимальні значення змінних та результати.
3. **Підтримка сучасних алгоритмів:** використання Genetic Algorithm з можливістю детального налаштування.

Рішення

1. Побудова функції

В **Optimize Live Editor Task** функція задається через поле **Objective**.

Синтаксис для введення функції:

```
x*exp(-(x.^2 + y.^2))
```

Примітка: Використовується `.` для елементних операцій, щоб забезпечити коректність обчислень для векторів.

2. Встановлення меж змінних

У вкладці **Variables** задаються обмеження для кожної змінної:

- xxx: Lower bound = -2, Upper bound = 2.
- yyy: Lower bound = -2, Upper bound = 2.

3. Вибір алгоритму

1. У секції **Solver** вибираємо **Genetic Algorithm (ga)**.
2. Налаштовуємо параметри алгоритму:
 - Population size: 50
 - Maximum generations: 100
 - Mutation: Gaussian
 - Selection: Roulette wheel

4. Виконання оптимізації

1. Натискаємо **Run** у Live Editor Task.
2. MATLAB знаходить мінімальне значення функції $z(x,y)$ у заданих межах -0.42.

▼ Create optimization variables

Name	Dimensions	Type	Lower bound	Upper bound	Initial point	
x	1x1 ▼	Continuous ▼	-2 ▼	2 ▼	0 ▼	- +
y	1x1 ▼	Continuous ▼	-2 ▼	2 ▼	0 ▼	- +

▼ Define problem

Goal Minimize Maximize Feasibility Solve equations

Objective Define on one line ▼

Constraints

▼ Specify problem-dependent solver options

Solver

Options

OptimizationProblem :

Solve for:

x, y

minimize :

$(x \cdot \exp(-(x^2 + y^2)))$

variable bounds:

$-2 \leq x \leq 2$

$-2 \leq y \leq 2$

Solving problem using ga.

ga stopped because the average change in the fitness value is less

solution = struct with fields:

x: -0.7071

y: 1.0230e-05

reasonSolverStopped =

SolverConvergedSuccessfully

objectiveValue = -0.4289

Контрольні питання

1. **Основні особливості генетичних алгоритмів:**
 - Виконують пошук на основі популяції можливих рішень.
 - Використовують тільки цільову функцію, без необхідності в похідних.
 - Відбір відбувається за ймовірнісними правилами.
 2. **Генетичні оператори:**
 - Селекція.
 - Схрещування.
 - Мутація.
 3. **Критерії зупинки:**
 - Досягнення максимальної кількості поколінь.
 - Досягнення мінімальної зміни цільової функції.
 - Вичерпання часу.
 4. **Схема класичного ГА:**
 - Генерація початкової популяції.
 - Оцінка пристосованості хромосом.
 - Селекція батьків.
 - Схрещування батьків для створення нащадків.
 - Мутація нащадків.
 - Створення нової популяції.
 - Перевірка критеріїв зупинки.
 5. **Особливості використання генетичних операторів:**
 - Селекція забезпечує вибір найкращих хромосом.
 - Схрещування створює нові комбінації генів.
 - Мутація запобігає застряганням в локальних екстремумах.
-

Висновки

- Генетичний алгоритм ефективно знаходить мінімум функції $z(x,y)=x \cdot e^{-(x^2+y^2)}$ навіть за наявності нелінійностей.
- Налаштування параметрів алгоритму, таких як розмір популяції, кількість поколінь, імовірності схрещування та мутації, впливають на швидкість і точність знаходження оптимального розв'язку.
- MATLAB надає інструменти як для програмного, так і для графічного використання ГА.

!