

2024 Probabilistic Model Class

Modern Optimization with R

- Chapter 5. Population Based Search



순천향대학교 미래융합기술학과

Senseable AI Lab

석사과정 김병훈

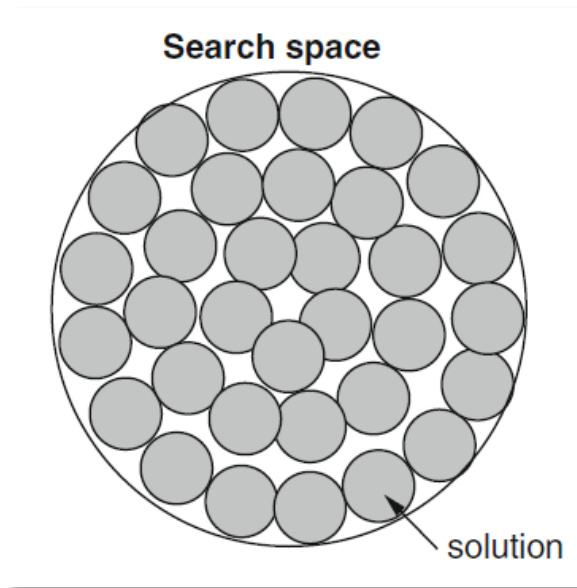
- 1 Introduction**
- 2 Genetic and Evolutionary Algorithms**
- 3 Differential Evolution**
- 4 Python Result**

1 Introduction

1. Introduction

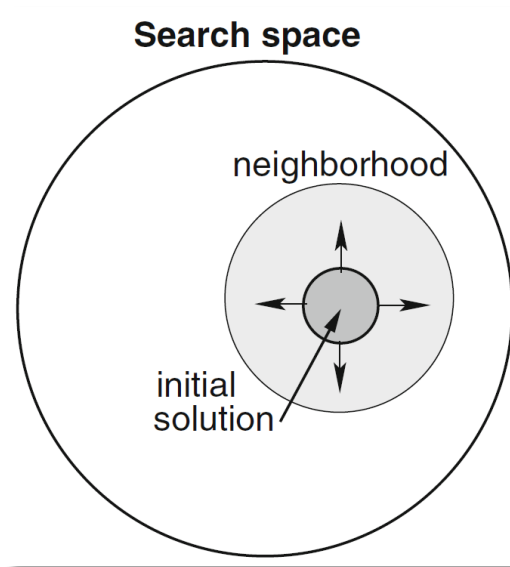
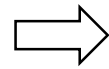
Population Based Search

Methodology comparison



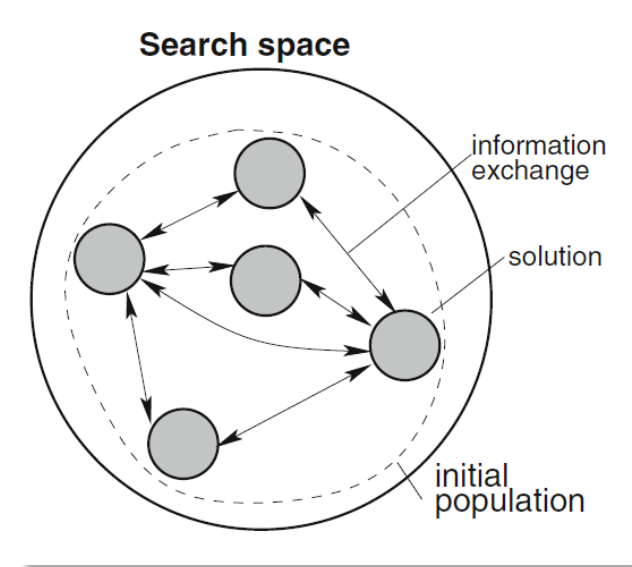
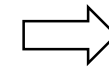
[Full Blind Search]

- 가능한 탐색 지점을 무차별적으로 탐색
- 시간 복잡도가 높음



[Local Search]

- 주변 지역(이웃)만을 탐색하여 점진적으로 해를 개선
- 지역 극솟값(local minima) 문제



[Population Based Search]

- 단일 탐색 지점 대신 여러 후보 솔루션을 동시에 고려
- 탐색 공간을 보다 넓게 탐색
- 지역 극솟값(local minima)에 빠지는 것을 피할 수 있음

1. Introduction

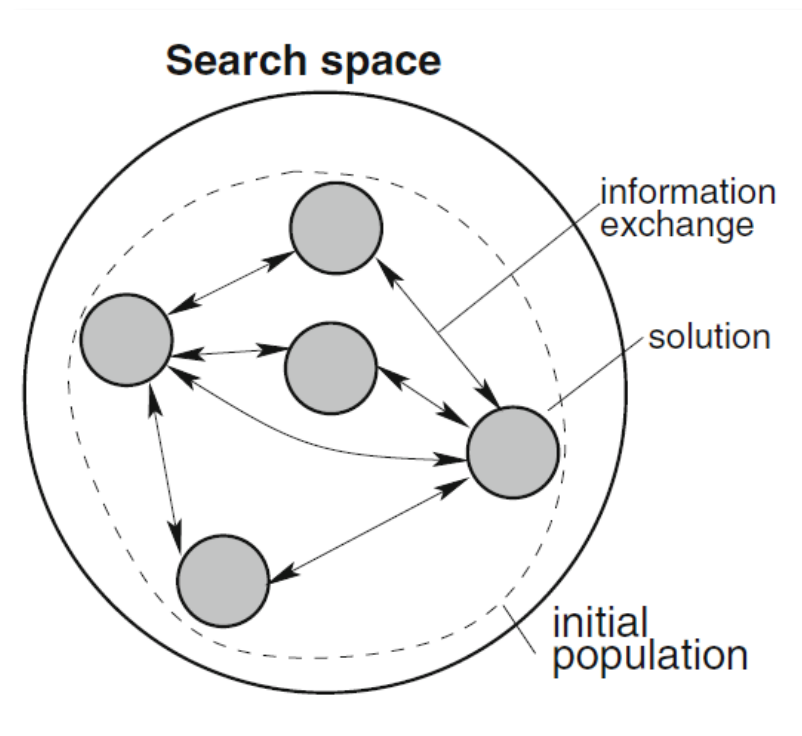
Population Based Search

Population Based Search 정의

- 개별 해 대신 해의 집단을 사용하여 탐색을 수행하는 최적화 전략
- 전체 해 공간에 대한 더 넓은 탐색을 가능하게 함
- 지역 최적해(local optima)에서 벗어나 글로벌 최적해(global optima)에 더 가까운 해를 찾는 데 도움을 줌

Population Based Search 특징

1. 탐색 범위: 탐색 공간의 더 다양한 영역을 탐색
2. 글로벌 최적화: 글로벌 최적화를 수행하는 데 있어 더 효과적으로 작동
3. 계산 요구도: 간단한 Local Search와 비교할 때, 더 많은 계산을 요구
4. 자연 현상: 주로 자연 현상에서 기인한 방법(유전 알고리즘, 개미 군집화 등)

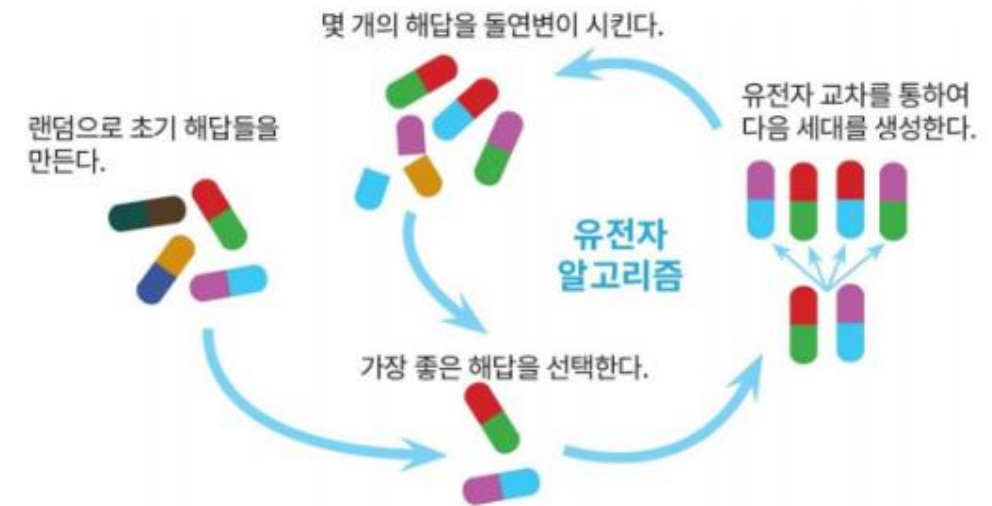


1. Introduction

Local Search

Population Based Search 종류

- Genetic and Evolutionary Algorithms
- Differential Evolution
- Particle Swarm Optimization
- Ant Colony Optimization
- Estimation of Distribution Algorithms



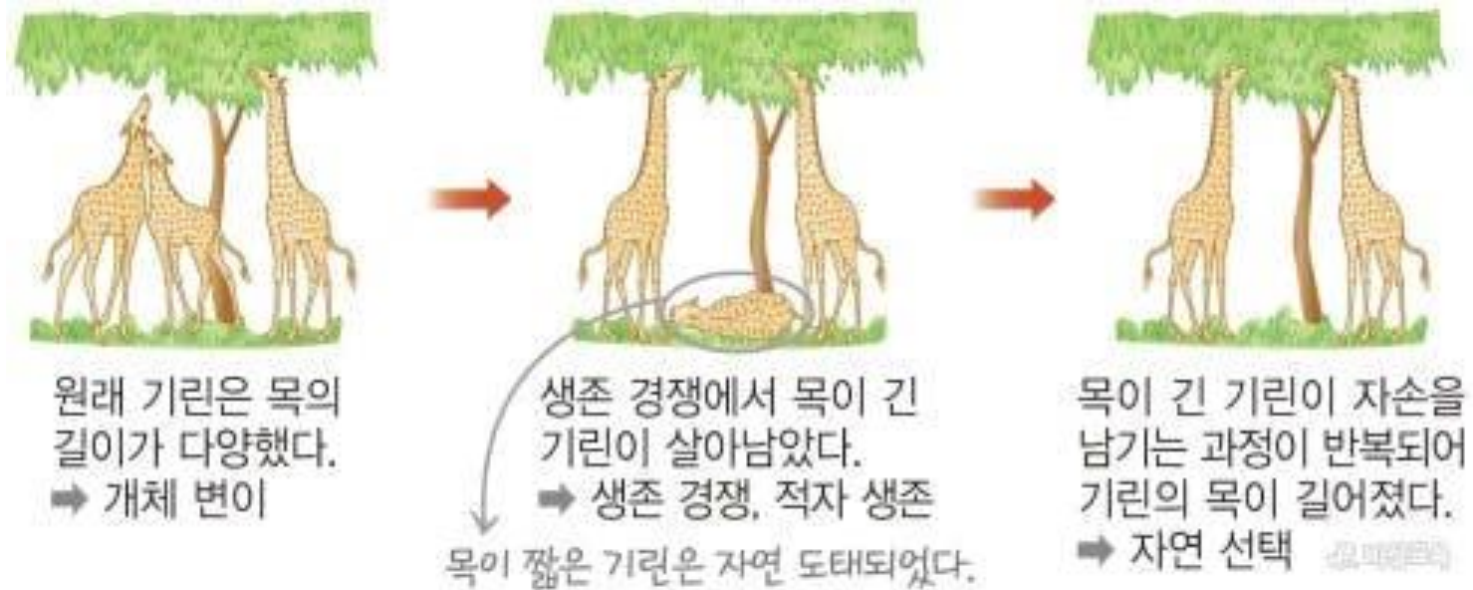
2 Genetic and Evolutionary Algorithms

2. Genetic and Evolutionary Algorithms

Define

Genetic and Evolutionary Algorithms

- 생물학적 진화 과정을 모방한 최적화 알고리즘 (다윈의 진화론: 자연 선택 이론)
- 솔루션들의 집단을 진화(개체 변이)시켜 가며 최적의 해를 찾음

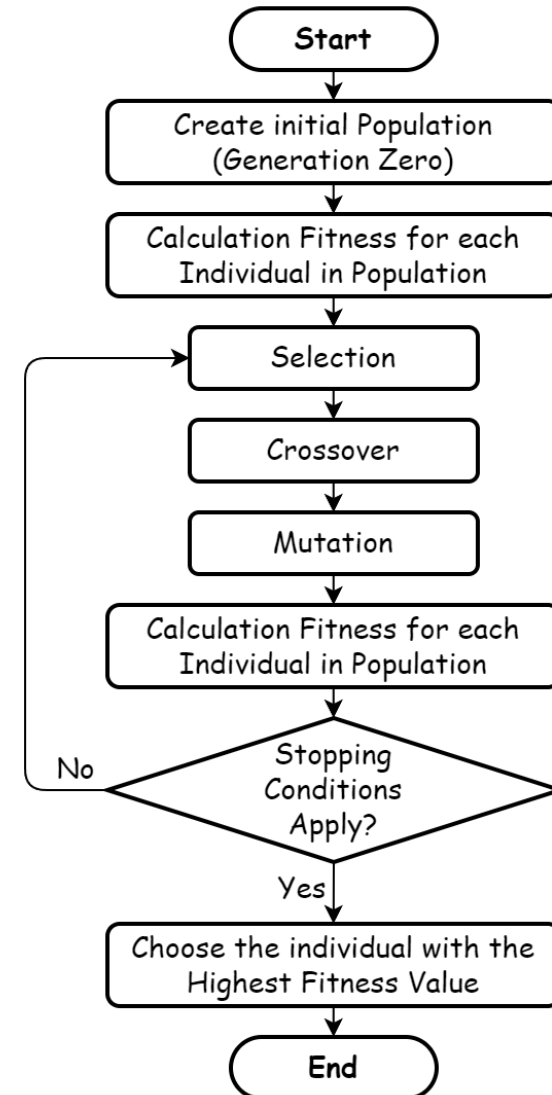


2. Genetic and Evolutionary Algorithms

Algorithms operations

Operations

1. Initial
2. Evaluation
3. Selection
4. Crossover
5. Mutation
6. Iteration Steps 2-5
7. Provide Optimal Solutions



2. Genetic and Evolutionary Algorithms

Advantage and Disadvantage

Advantage

- 다양한 탐색 공간에 대한 광범위한 탐색 -> 글로벌 최적화에 효과적
- 다양하고 복잡한 문제 유형에 적용 가능 -> 메타휴리스틱에 기반
- 진화 과정에서 Crossover, Mutation을 통해 다양한 솔루션 도출 -> Local optima 에 빠질 가능성 낮음

Disadvantage

- 계산 비용이 높고 시간이 오래 걸림
- 적절한 파라미터 설정이 필요
- Global optima를 찾는 것이 보장되지 않음

2. Genetic and Evolutionary Algorithms

R packages

genalg packages (rbga.bin(이산 값) / rbga (연속 값))

- popSize: population 크기
- iters: 최대 반복 수
- mutationChance: 돌연변이 확률
- crossoverProb: 교차 발생 확률
- elitism: 가장 적합한 개체가 다음 세대로 직접 전달되는 개체 수

ecr packages

- mu: 부모 세대 크기
- lambda: 자식 세대 크기
- representation: 솔루션 표현 방식(이산, 연속 등)

그 외 GA, mcga, NMOF 패키지 등이 있음

3 Differential Evolution

3. Differential Evolution

Define

Differential Evolution Algorithms

- 연속적인 수치 최적화 문제를 해결하기 위해 개발된 전역 최적화 알고리즘
- 유전 및 진화 알고리즘과 유사하게 솔루션의 집단을 진화시키며 학습
- 하지만 교차 및 변이 연산자 대신 산술 연산자를 사용하여 새로운 솔루션 생성
- 각 솔루션은 실수 값의 문자열로 구성

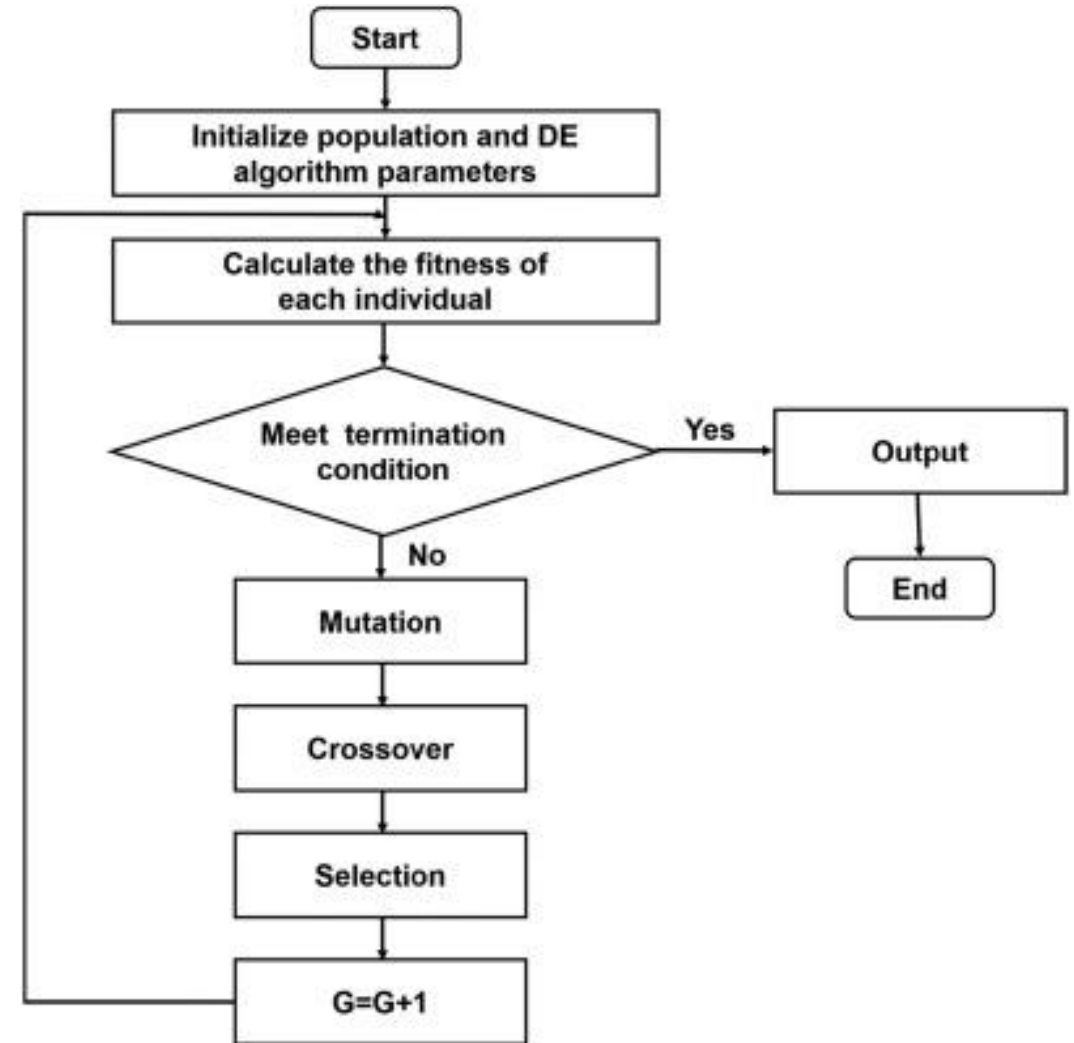
3. Differential Evolution

Algorithms operations

Operations

1. Initial
2. Evaluation
3. Selection
4. Mutation
5. Crossover
6. Iteration Steps 2-5
7. Provide Optimal Solutions

Genetic and Evolutionary Algorithms과 동일

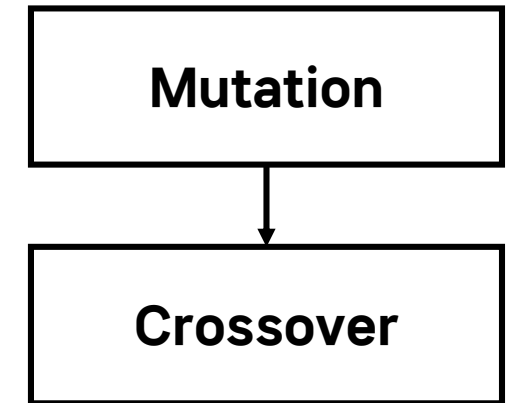


3. Differential Evolution

Algorithms operations

Mutation

- 주요 차이점: 벡터 간의 차이를 활용하여 변이(유전 알고리즘 : 무작위 변이)
 1. 임의로 세 개의 서로 다른 개체 s_1, s_2, s_3 를 선택
 2. 선택된 세 개체 중 두 개체 간의 차이 계산 ($s_2 - s_3$)
 3. 스케일링 인자 F 를 곱함
 4. 계산된 결과를 나머지 개체(s_1)에 더해 donor vector 생성($s_{mut} = s_1 + F(s_2 - s_3)$)



Crossover

- 주요 차이점: donor vector와 현재 개체 사이에서 일정 비율을 섞어 새로운 벡터 생성(유전 알고리즘: 교환 방법)
 - 교차 확률(parameter)에 따른 Crossover 실행 -> doner vector + current vector = new vector
 - 적어도 한 개의 개체는 new vector에서 가져오도록 함(조기 수렴 방지 및 새로운 솔루션으로의 강제)

※ donor vector: 유전공학 용어, 세포 외부에서 얻은 유전자, 이를 통해 새로운 DNA 형질 획득

3. Differential Evolution

Advantage and Disadvantage

Advantage

- 산술 연산을 기반으로 하여 구현이 간단함
- 연속 최적화 문제에서 효과적인 수렴 성능
- 적은 파라미터
- 효율적 탐색

Disadvantage

- 복잡한 문제에서 수렴 속도가 상대적으로 느림(이전 알고리즘에 비해)
- Global optima를 찾는 것이 보장되지 않음
- 이산 최적화 문제에 적합하지 않음(하지만 추가 조정을 통해 구현은 가능)

3. Differential Evolution

R packages

DEoptim packages

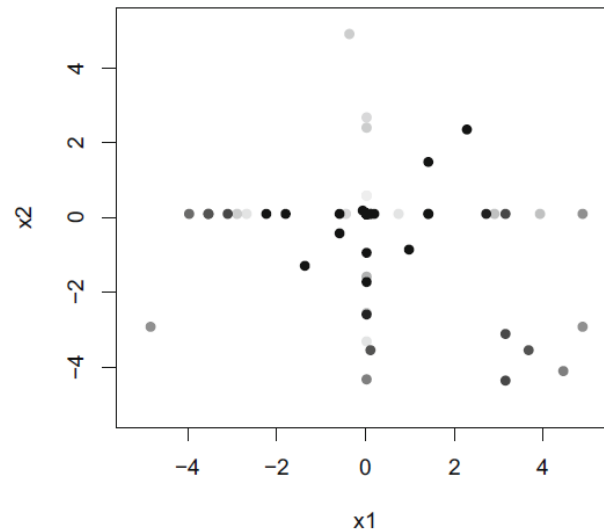
- fn: 최소화 함수
- VTR: 목표 값(Threshold)
- CR: 교차 확률
- F: 차분 스케일링 인자

그 외 DEoptimR, NMOF, GA 패키지 등이 있음

3. Differential Evolution

Compare results(sphere)

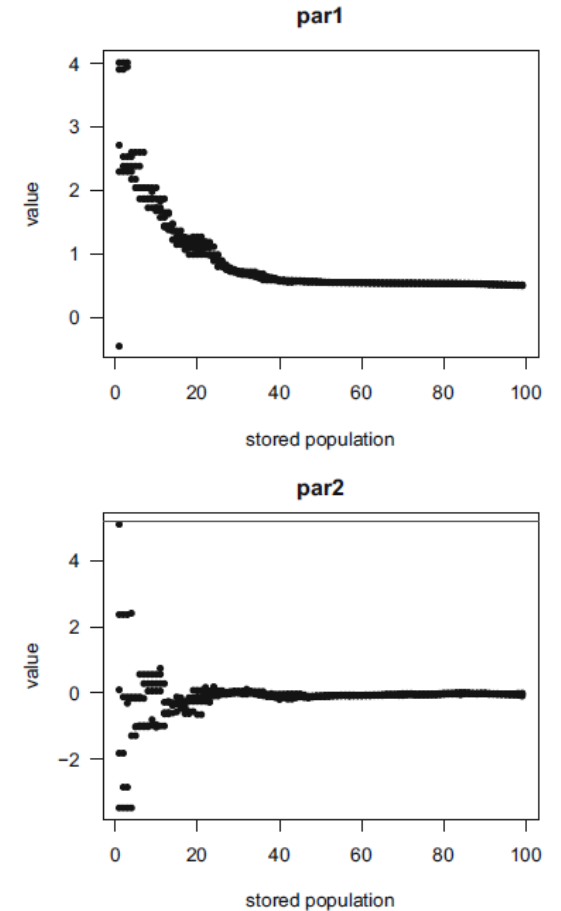
Fig. 5.4 Example of an evolutionary algorithm search for **sphere** ($D = 2$)



Genetic and Evolutionary Algorithms
($f = 0.0098$)



Fig. 5.6 Population evolution in terms of x_1 (top) and x_2 (bottom) values under the differential evolution algorithm for **sphere** ($D = 2$)



Differential Evolution
($f = 0.25$)

4 Python Result

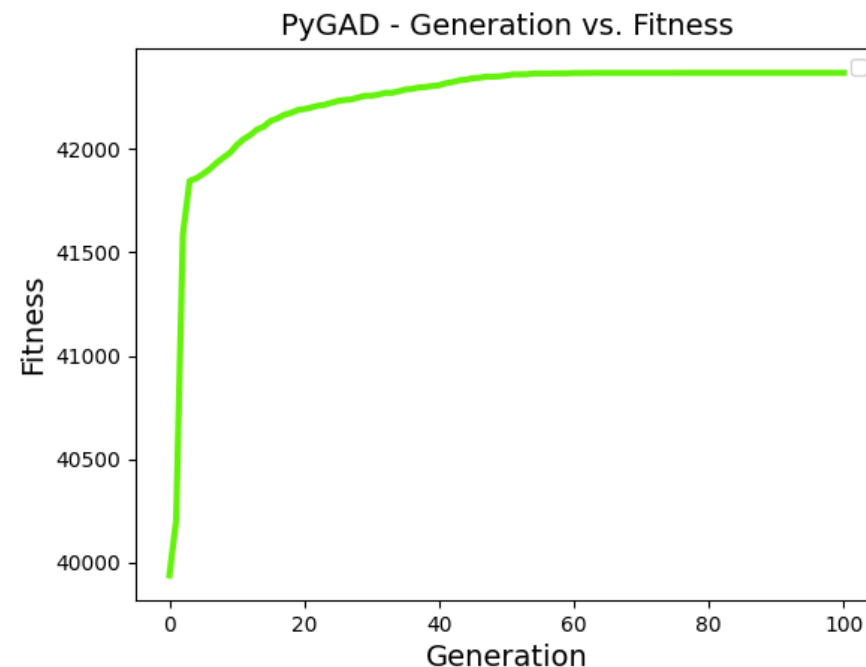
4. Python Result

Genetic and Evolutionary Algorithms

pygad 라이브러리

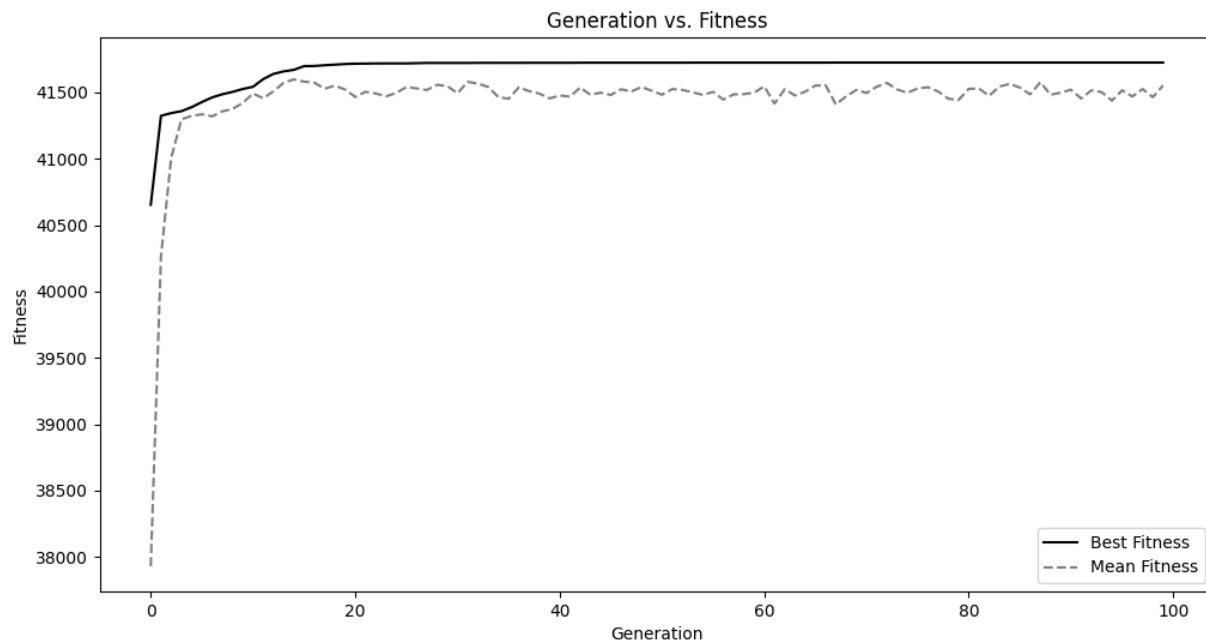
- pygad.GA 함수 제공
- best_solution(), plot_fitness() 함수 등 제공

```
ga_instance = pygad.GA(num_generations=100,  
                        num_parents_mating=5,  
                        fitness_func=fitness_function,  
                        sol_per_pop=50,  
                        num_genes=5,  
                        gene_space=[{'low': 1, 'high': 1000}] * 5,  
                        mutation_percent_genes=10,  
                        mutation_num_genes=1,  
                        parent_selection_type="sss",  
                        crossover_type="single_point",  
                        mutation_type="random",  
                        keep_parents=1)
```



4. Python Result

Genetic and Evolutionary Algorithms(Bag Price) – Generations 변경



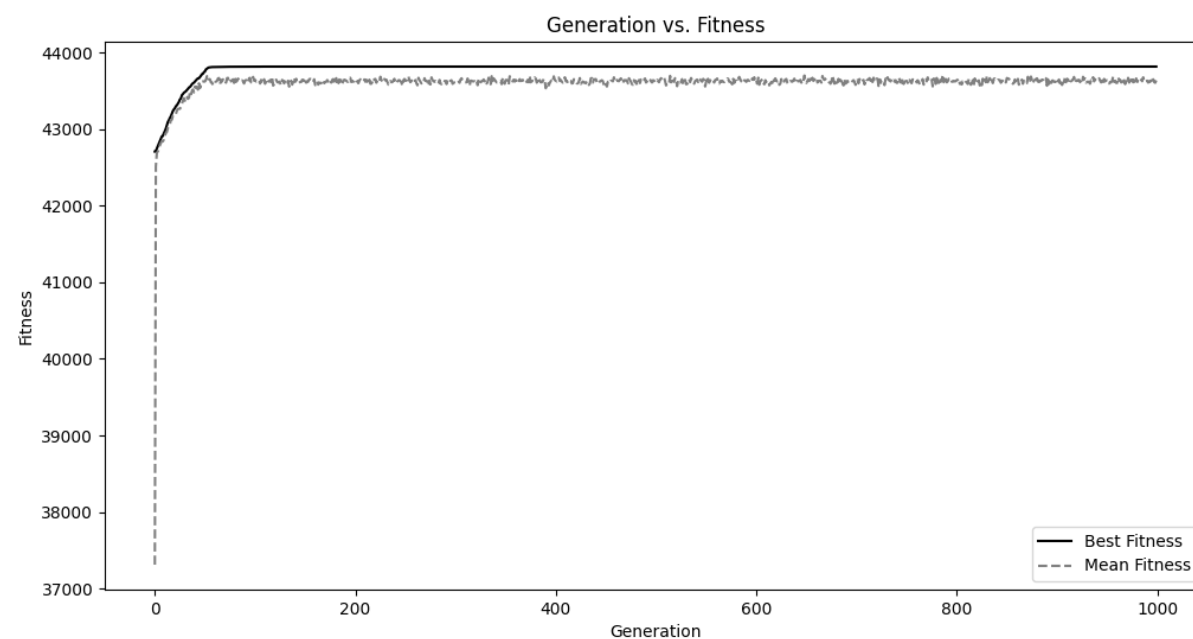
Generations: 100
Time : 0.53(s)

[실제 최적 가격]

Bag 1: 414
Bag 2: 404
Bag 3: 408
Bag 4: 413
Bag 5: 395
Total : \$43899

[예측 최적 가격]

Bag 1: 401.83
Bag 2: 418.87
Bag 3: 634.19
Bag 4: 526.04
Bag 5: 504.42
Total : \$41725



Generations: 1000
Time : 5.20(s)

[실제 최적 가격]

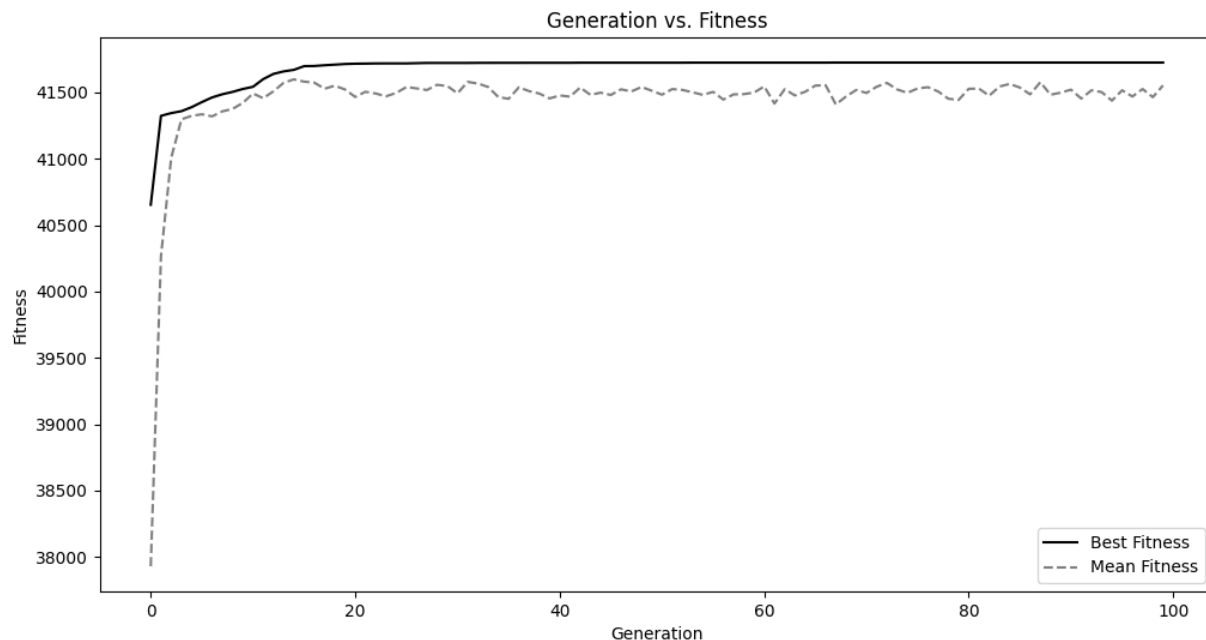
Bag 1: 414
Bag 2: 404
Bag 3: 408
Bag 4: 413
Bag 5: 395
Total : \$43899

[예측 최적 가격]

Bag 1: 377.84
Bag 2: 404.49
Bag 3: 425.01
Bag 4: 413.07
Bag 5: 447.09
Total : \$43817

4. Python Result

Genetic and Evolutionary Algorithms(Bag Price) – Mutation Percent 변경



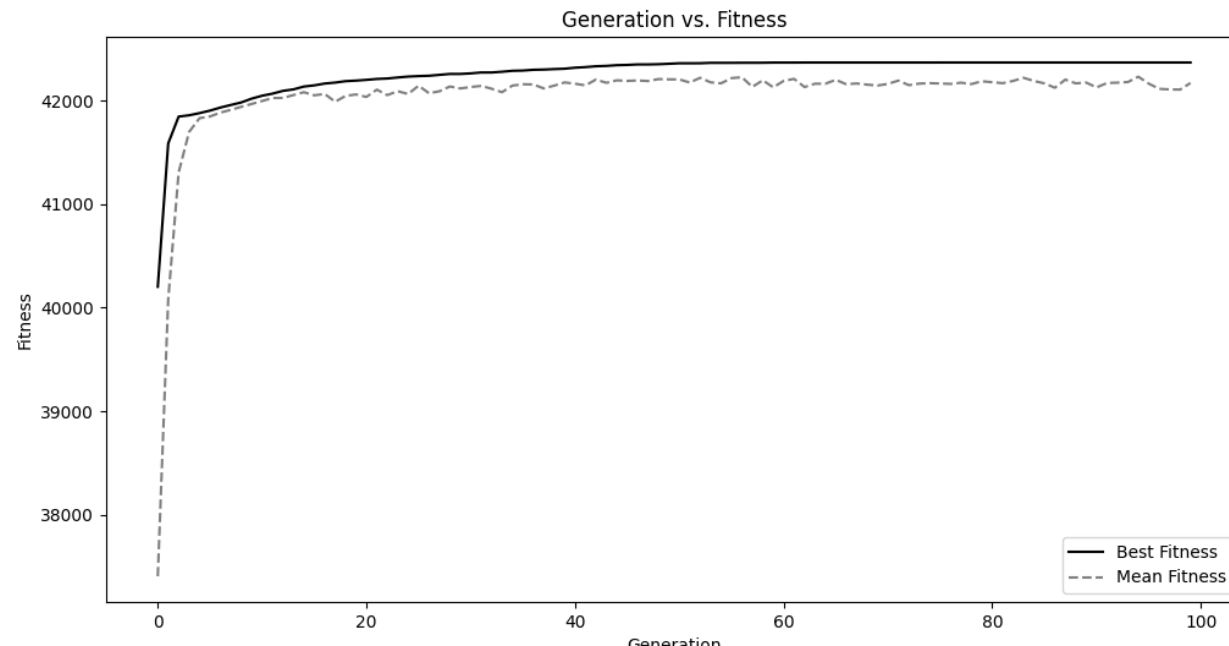
Mutation: 10%
Time : 0.53(s)

[실제 최적 가격]

Bag 1: 414
Bag 2: 404
Bag 3: 408
Bag 4: 413
Bag 5: 395
Total : \$43899

[예측 최적 가격]

Bag 1: 401.83
Bag 2: 418.87
Bag 3: 634.19
Bag 4: 526.04
Bag 5: 504.42
Total : \$41725



Mutation: 50%
Time : 0.53(s)

[실제 최적 가격]

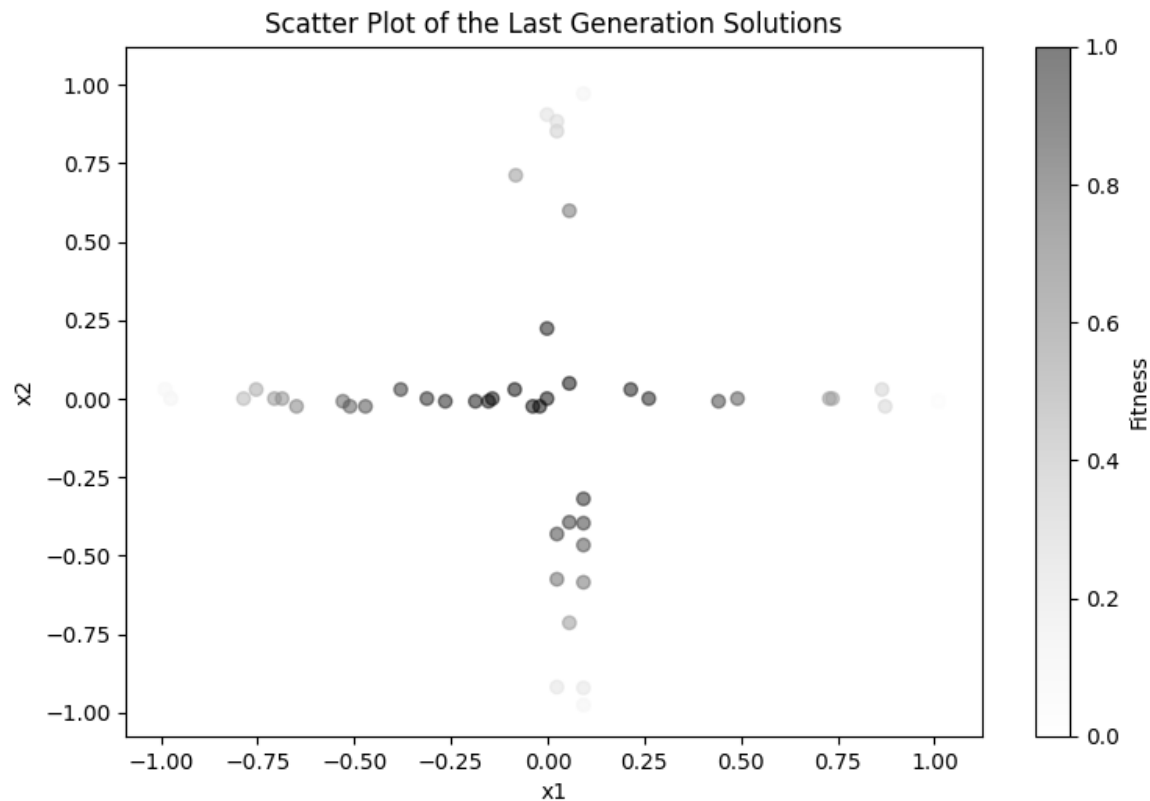
Bag 1: 414
Bag 2: 404
Bag 3: 408
Bag 4: 413
Bag 5: 395
Total : \$43899

[예측 최적 가격]

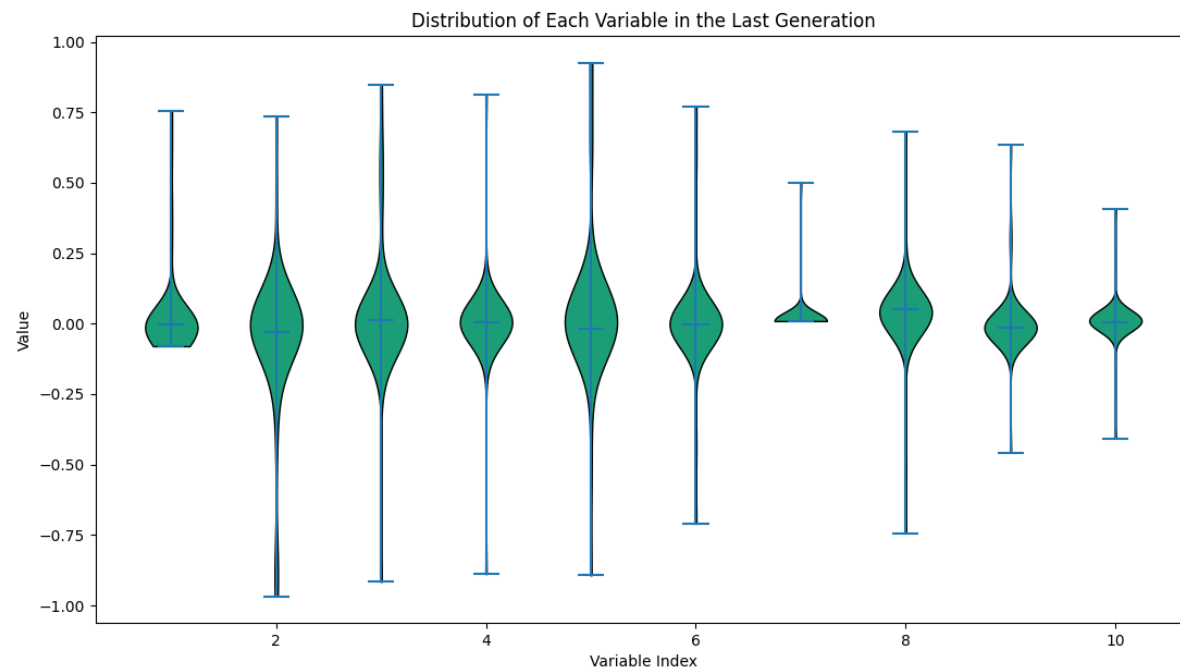
Bag 1: 527.61
Bag 2: 404.49
Bag 3: 391.69
Bag 4: 290.67
Bag 5: 568.55
Total : \$42367

4. Python Result

Genetic and Evolutionary Algorithms(Sphere)



D = 2 설정
최적 해: -6.34

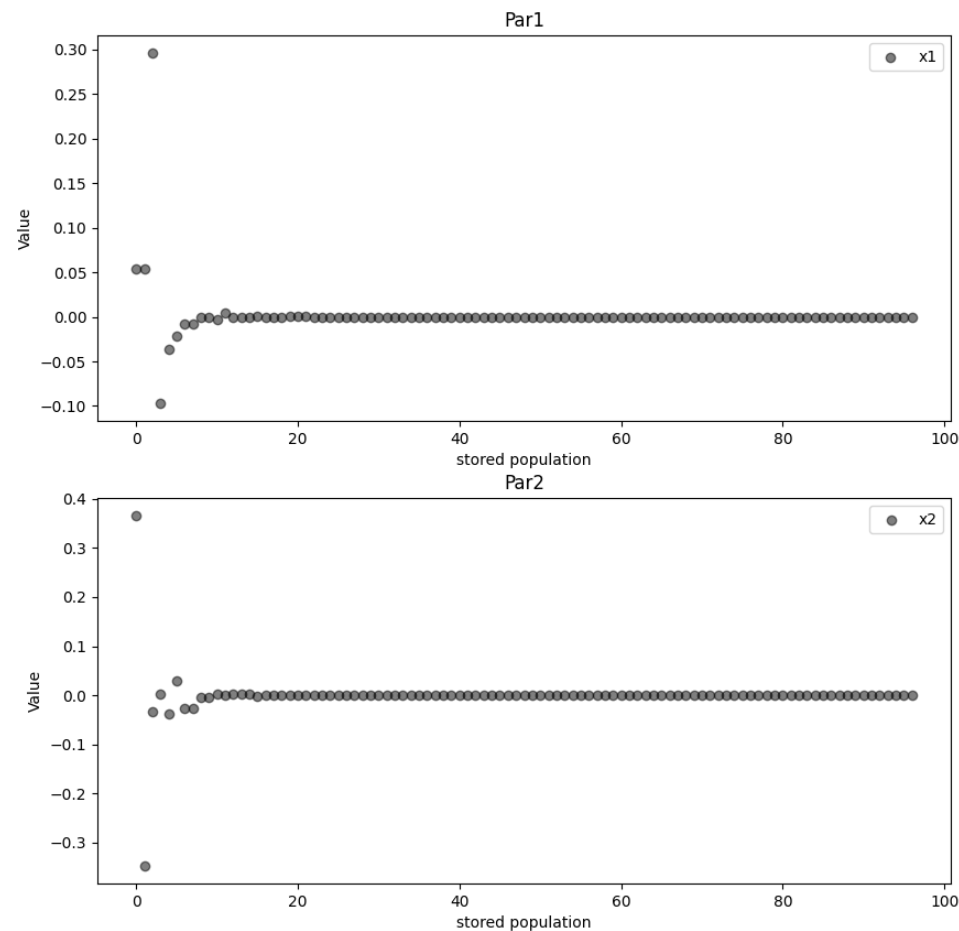
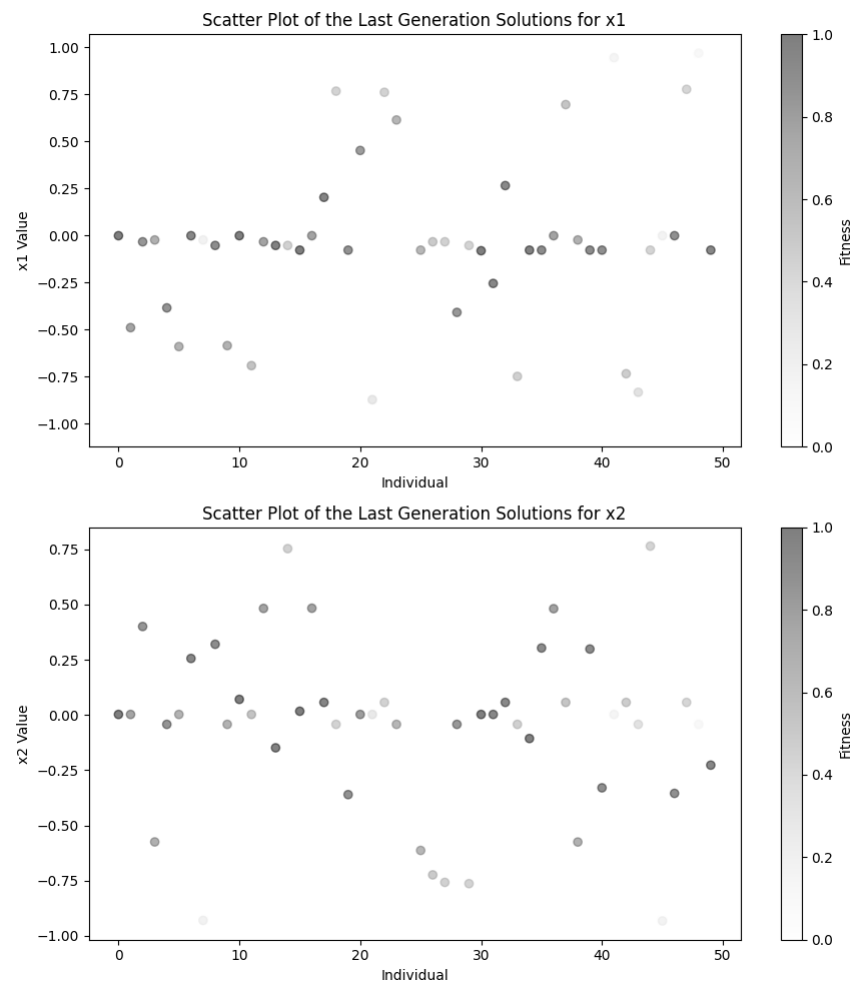


D = 10 설정
최적 해: -0.0015

https://github.com/Kim-Byeong-Hun/Probability_Modeling

4. Python Result

Differential Evolution Algorithms(Sphere)



Thanks