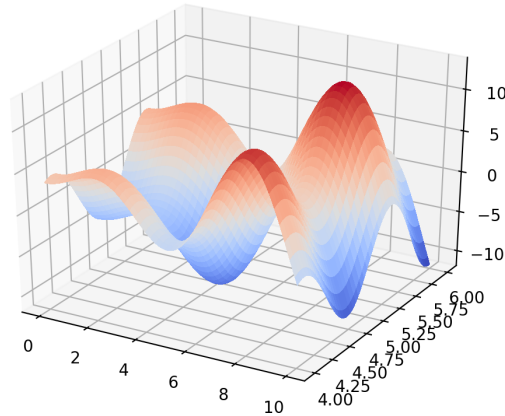


Optimization Theory and Application

- 1.1 find maximum of $f(x) = x_1 \sin(x_1) + x_2 \sin(5x_2)$, constraint set = $\{x: 0 \leq x_1 \leq 10, 4 \leq x_2 \leq 6\}$



(1) Real-number genetic algorithm

Parameters: population size(N), mutation rate(p_m), crossover rate(p_c), generations(g_iter), fitness function(func)

Fitness function: $f(x)$ function we want to find out maximum

Step 1: Initial Population

任意從 constraint set 選出 N 組 $[x_1, x_2]$ 作為 P_0 , 用 P_0 計算 $f(x)$, 找出最大值與其 x_1, x_2

Step 2: Termination when population have converged.

P 計算出 $f(x)$ 的結果 y, 減去其中的最小值 $y = y - \min(y)$, 讓結果全為正數, 若結果總和為 0, 則 population have converged, 並停止演算法

Step 3: Selection

P 計算出 $f(x)$ 的結果 y, 減去其中的最小值(全正數), 並除以結果總和 $y = y / \sum(y)$: probability of being selected

Roulette-Wheel selection:

將 y 累加(結果:0~1), 重複 N 次選出 y - random number(0~1) 中大於 0 的最小 y, 作為 mating pool(M_k)

Tournament selection

重複 N 次從 y 中任意取 2 值, 選其中較大的值, 作為 mating pool(M_k)

Step 4: Crossover

如果 1 random number(0~1) < crossover rate, 則從 M_k 任選出 2 parents, $\alpha = \text{random number}(0 \sim 1)$
 $\text{offspring1} = \alpha * \text{parent1} + (1 - \alpha) * \text{parent2} + (\text{random number}(0 \sim 1) - 0.5) * (\text{upper_bound} - \text{lower_bound}) / 10$

$$\text{offspring2} = \alpha * \text{parent2} + (1 - \alpha) * \text{parent1} + (\text{random number}(0 \sim 1) - 0.5) * (\text{upper_bound} - \text{lower_bound}) / 10$$

Projection:

如果 offspring 超過 bound, offspring = bound

將原 M_k 中的 parent 改為 offspring

Step 5: Mutation

重複 N 次, 當 $\text{random number}(0 \sim 1) < \text{mutation rate}$, M_k 中的第 n 個($M_k[n]$),

$$M_k[n] = \alpha * M_k[n] + (1 - \alpha) * (\text{lower_bound} + \text{random number}(0 \sim 1) * (\text{upper_bound} - \text{lower_bound}))$$

$$P_{k+1} = M_k$$

Step 6: Compute fitness function by P_{k+1}

$y = f(P_{k+1})$, 找出其中的最大值和其 x_1, x_2

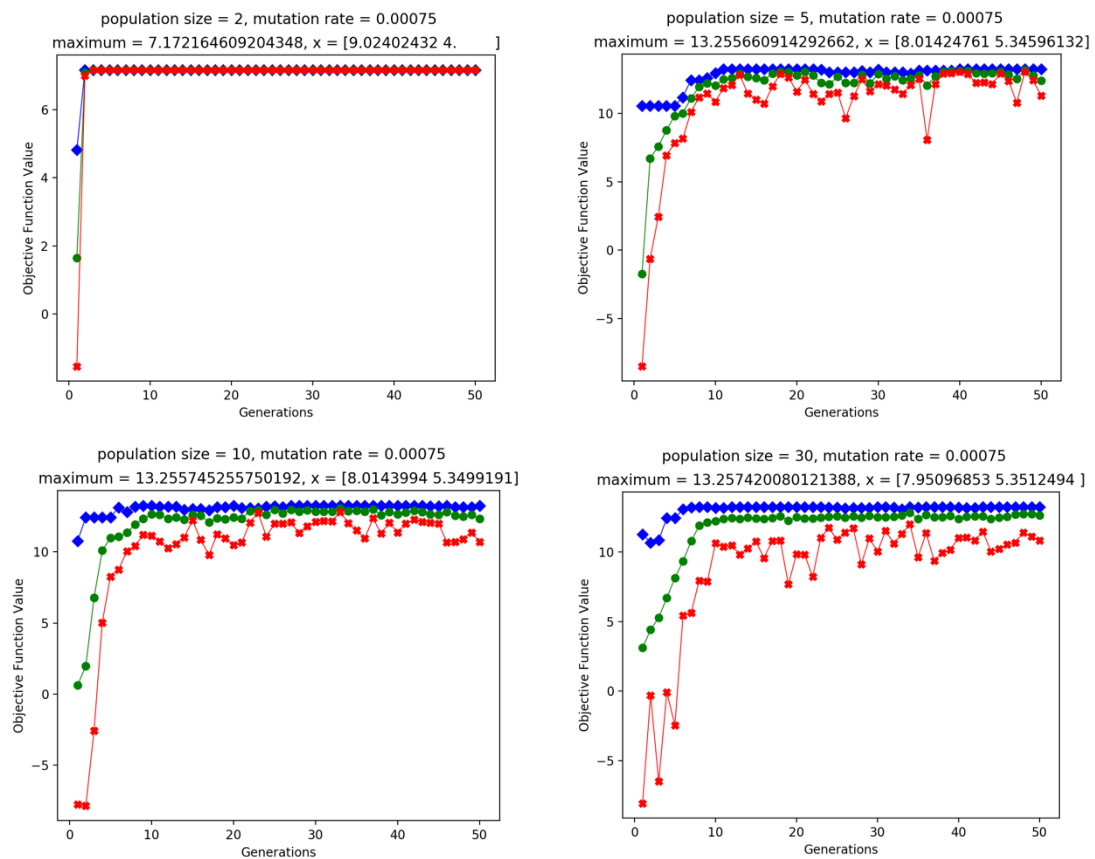
重複 Step 2~Step 6

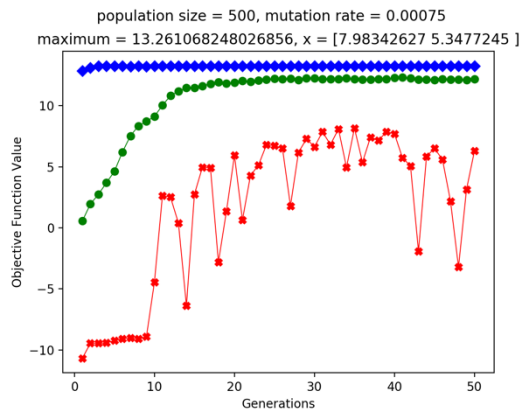
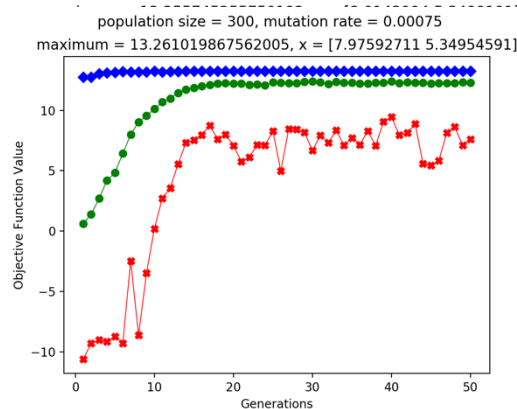
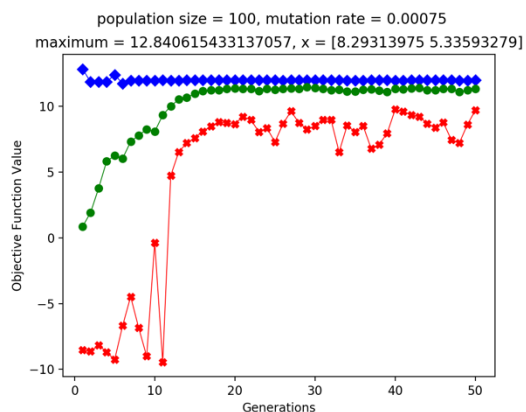
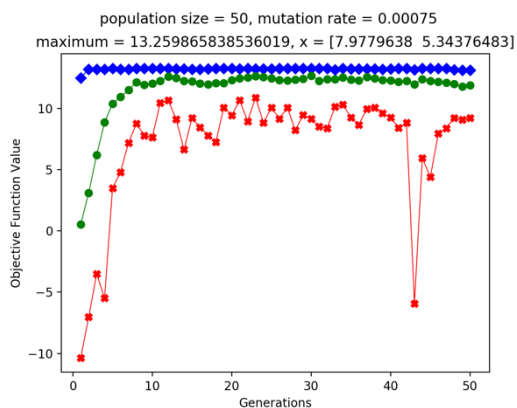
(2) 實作結果

1. Population size 參數設定

固定 crossover rate = 0.75, mutation rate = 0.0075, generations = 50

Population size = [2, 5, 10, 30, 50, 100, 300, 500]

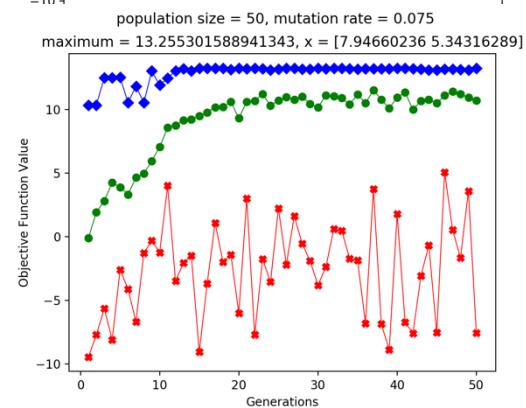
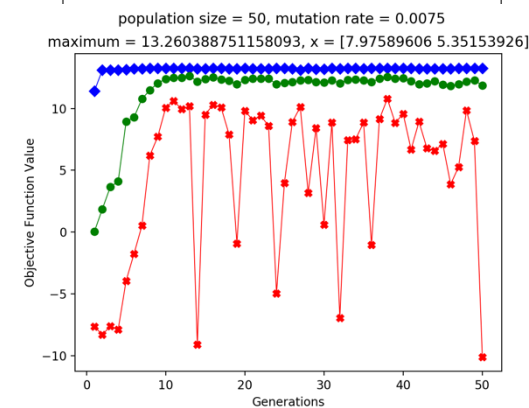
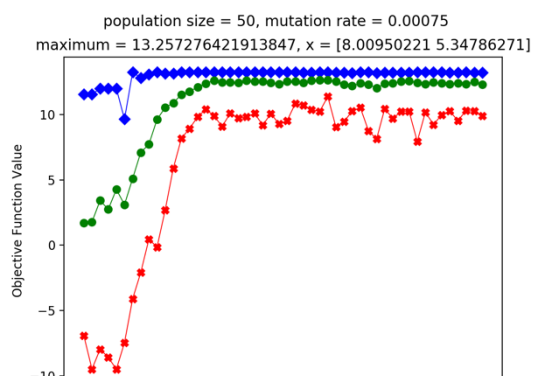
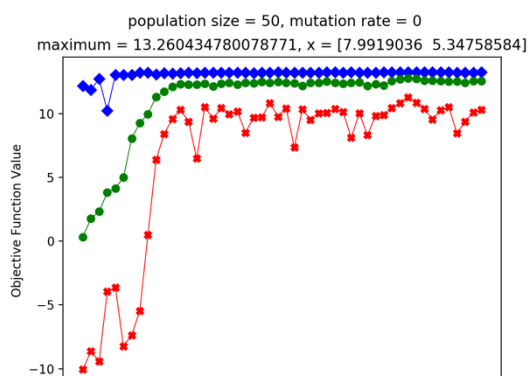


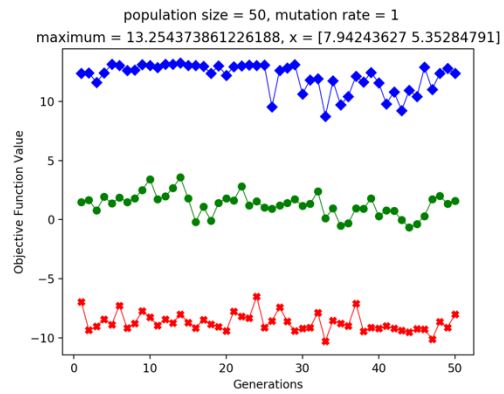
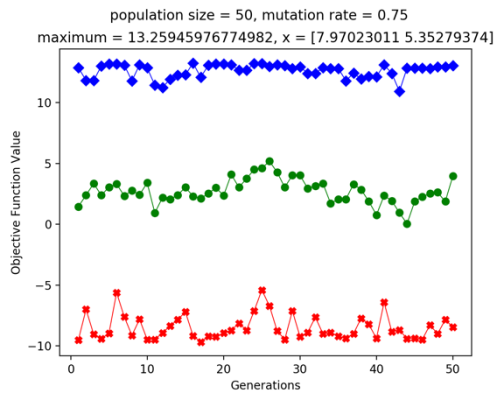


2. Mutation rate 参数设定

固定 crossover rate = 0.75, population size = 50, generations = 50

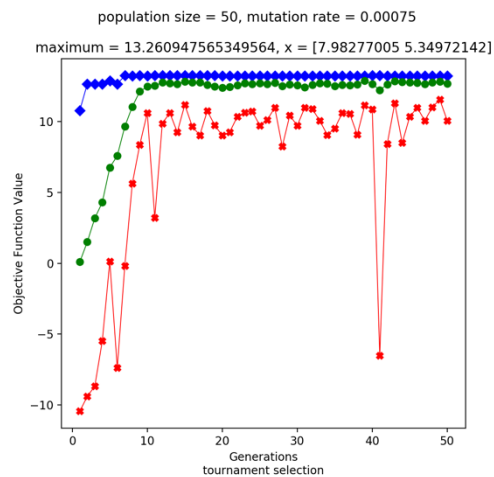
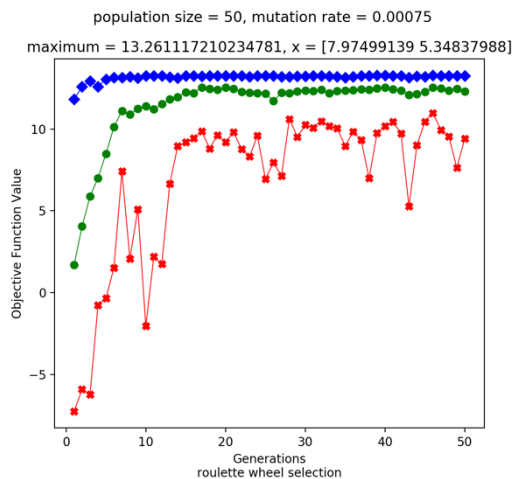
Mutation rate = [0, 0.00075, 0.0075, 0.075, 0.75, 1]





3. 比較 roulette wheel 和 tournament selection

固定 crossover rate = 0.75, population size = 50, generations = 50,
mutation rate = 0.0075



(3) 分析

1. Population size 參數設定

由上述結果可以看出，當 population size 越小時，越容易在還沒找到最大值以前就已經收斂了，但當 population size 越大時，容易找到最大值，不過 population 也容易包含到其他較小的值，在運算中，可能會浪費許多空間。

2. Mutation rate 參數設定

Mutation rate 越大時，表示對原本資料產生的變異越大。由上述結果可以得知，當 Mutation rate 越大時，與前一 population 差異過大，會導致每一 generation 之間毫無意義，但如果 Mutation rate 為零時，可能會難以跳脫出前一 population，而找不到 global maximum。

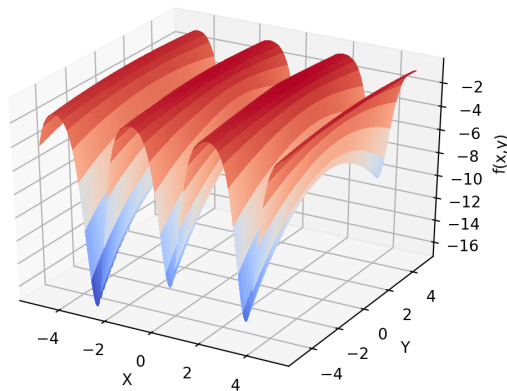
3. 比較 roulette wheel 和 tournament selection

roulette wheel 是按 chromosome 對應的函數值給予被選取的機率之後做選擇，tournament 則是任選二 chromosome 後選出較大的，就以定義上可以發現 tournament 是不會選到最小值的，但 roulette wheel 有機會選到最小值，tournament 整體選出的

chromosome 會比 roulette wheel 來得大，這個結果也可以在上述的實驗結果中發現，平均值 tournament 會比 roulette wheel 更接近最大值。

1.2

a. find minimum of $-e^{0.2\sqrt{(x-1)^2+(y-1)^2}} + \cos(2x) + \sin(2y)$, constraint set = $\{-5 \leq x, y \leq 5\}$



(1) 將 Real-time genetic algorithm 的 maximize selection 轉為 minimize selection

1. Roulette wheel selection

$y = -(y - \max(y)) / \sum(y)$ 讓 y 中越小的值，被選取的機率越大

2. Tournament selection

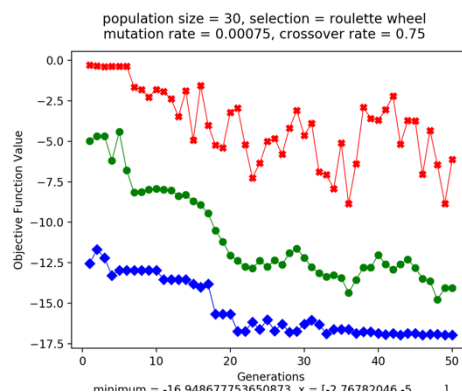
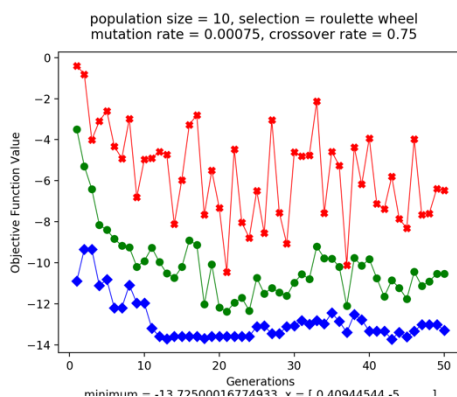
重複 N 次從 y 中任意選取 2 值，以其中較小的值，作為 mating pool

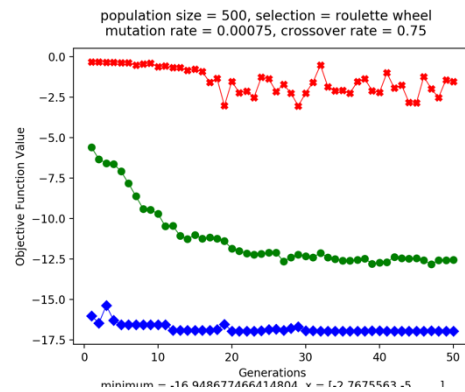
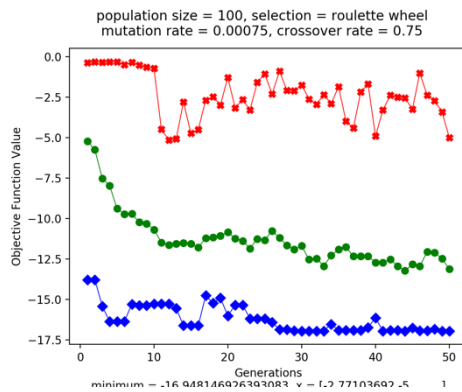
(2) 實作結果

1. Population size 參數設定

固定 crossover rate = 0.75, mutation rate = 0.0075, generations = 50

Population size = [10, 30, 100, 500]

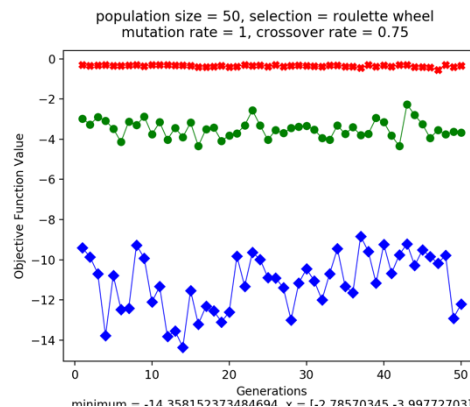
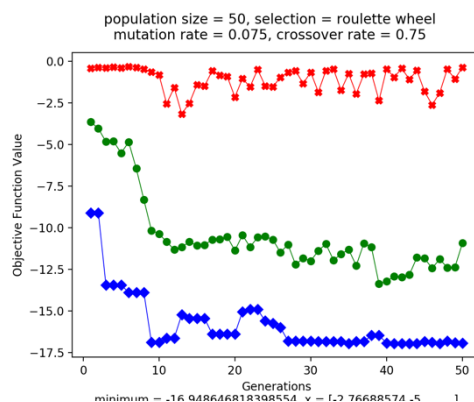
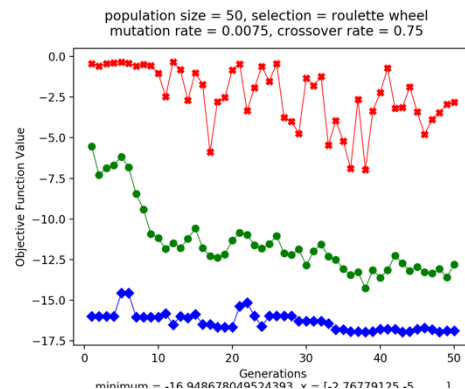




2. Mutation rate 參數設定

固定 crossover rate = 0.75, population size = 50, generations = 50

Mutation rate = [0, 0.0075, 0.075, 1]



(3) 分析

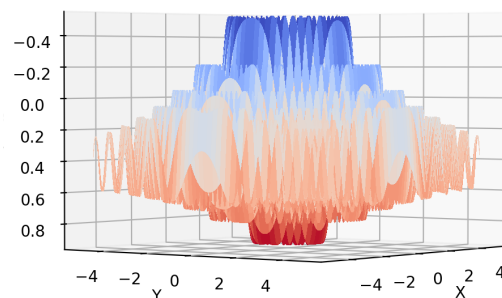
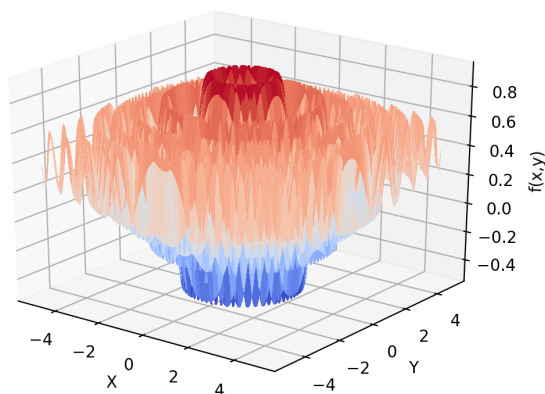
1. Population size 參數

這個函數有較多的極大值，由上述結果可以發現，當 population size 越大時，generation 的走勢趨於平緩，無較大的震盪。

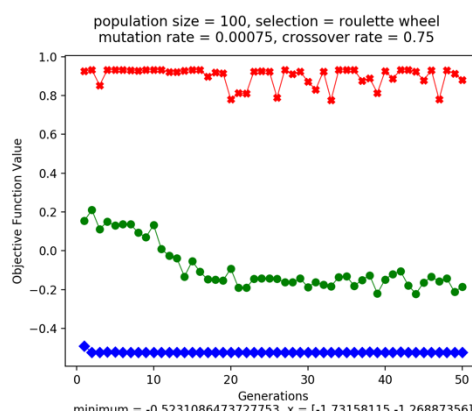
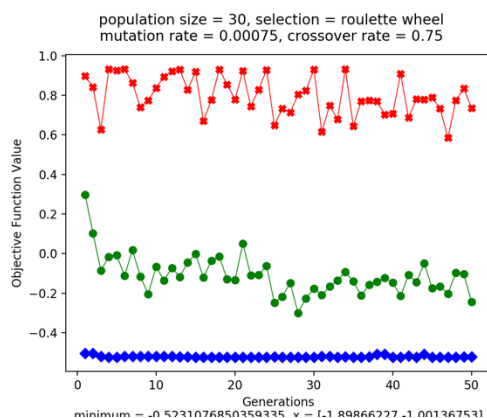
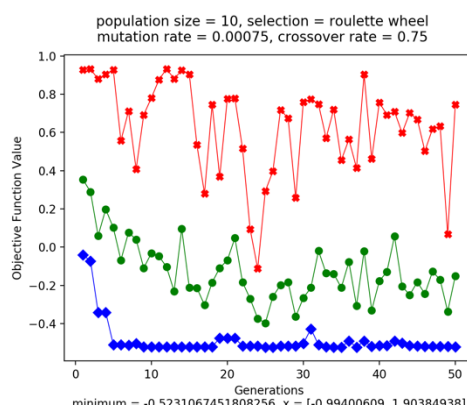
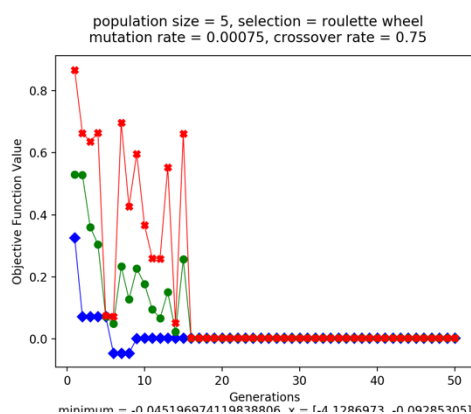
2. Mutation rate 參數

由上述結果可以得知，當 Mutation rate 越大時，與前一 population 差異過大，會導致每一 generation 之間毫無意義，但如果 Mutation rate 為零時，可能會難以跳脫出前一 population，而找不到 global maximum。

b. find minimum of $0.5 + \frac{\sin(x^2+y^2)-0.5}{1+0.1(x^2+y^2)}$, constraint set = $\{-\infty \leq x, y \leq \infty\}$



(1) 實驗結果



(2) 分析

Genetic algorithm 是用來找出 global maximum/minimum，可以和之前學習找 local maximum/minimum 比較發現，找 local maximum/minimum 的方法主要是沿著可行方向找，容易停在 local maximum/minimum 跳不出來，而 genetic algorithm 是一直不斷有機率地隨機抽樣，並且嘗試做一些變異，在計算是容易跳脫 local maximum/minimum，更能找到我們真正需要的 global maximum/minimum。