

EE345-AI: Assignment-6

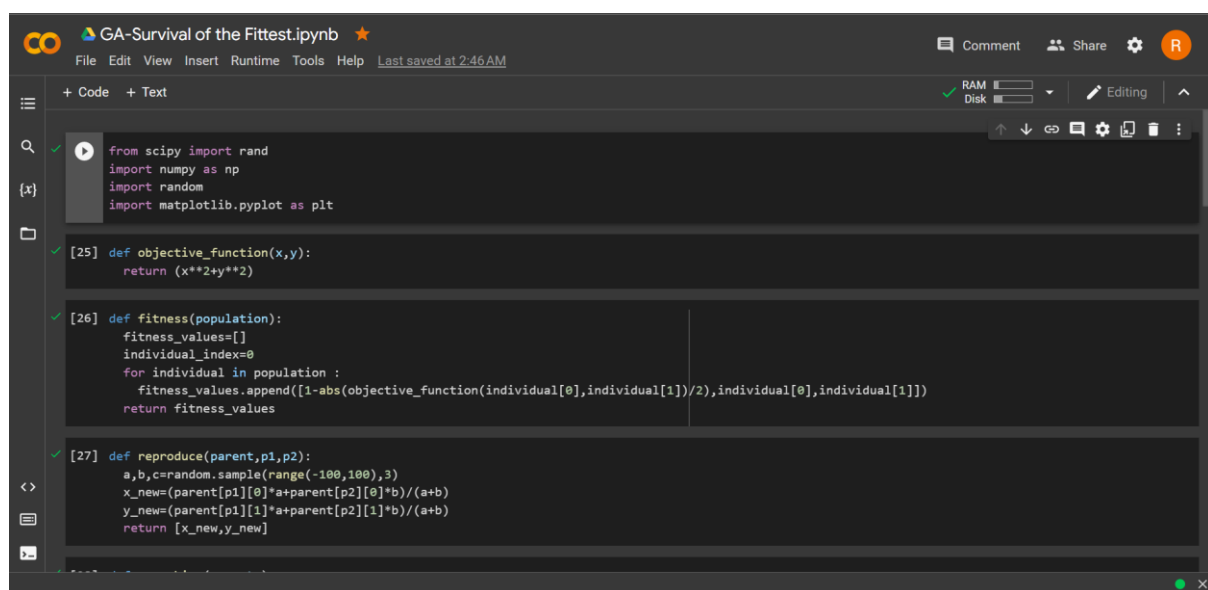
Name-Rohan Saha | RollNo.-20085086 | Electrical

Q. Write the program for maximising an objective function with at least two variables as discussed in today's class. Run it for 10 iterations and give the following:

- screenshot of the program code
- screenshot of the output giving the last 5 best values along with their fitness.
- the best value and its fitness for each iteration

Program Link :- https://colab.research.google.com/drive/1ByfHOa3c_1-jH54PC_UbX1u77psCI98?usp=sharing

Screenshots of the Source Code :-



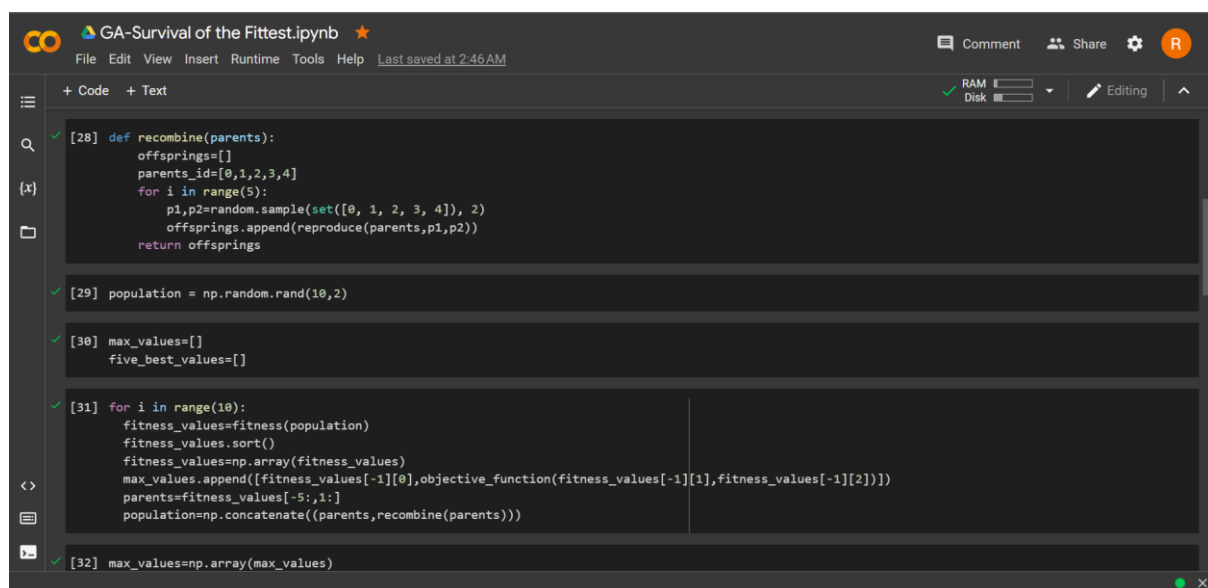
```
GA-Survival of the Fittest.ipynb
File Edit View Insert Runtime Tools Help Last saved at 2:46 AM

+ Code + Text
from scipy import rand
import numpy as np
import random
import matplotlib.pyplot as plt

[25] def objective_function(x,y):
    return (x**2+y**2)

[26] def fitness(population):
    fitness_values=[]
    individual_index=0
    for individual in population :
        fitness_values.append([1-abs(objective_function(individual[0],individual[1])/2),individual[0],individual[1]])
    return fitness_values

[27] def reproduce(parent,p1,p2):
    a,b,c=random.sample(range(-100,100),3)
    x_new=(parent[p1][0]*a+parent[p2][0]*b)/(a+b)
    y_new=(parent[p1][1]*a+parent[p2][1]*b)/(a+b)
    return [x_new,y_new]
```



```
GA-Survival of the Fittest.ipynb
File Edit View Insert Runtime Tools Help Last saved at 2:46 AM

+ Code + Text

[28] def recombine(parents):
    offsprings=[]
    parents_id=[0,1,2,3,4]
    for i in range(5):
        p1,p2=random.sample(set([0, 1, 2, 3, 4]), 2)
        offsprings.append(reproduce(parents,p1,p2))
    return offsprings

[29] population = np.random.rand(10,2)

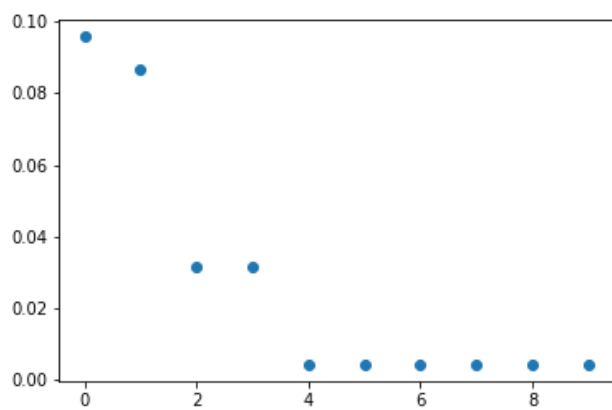
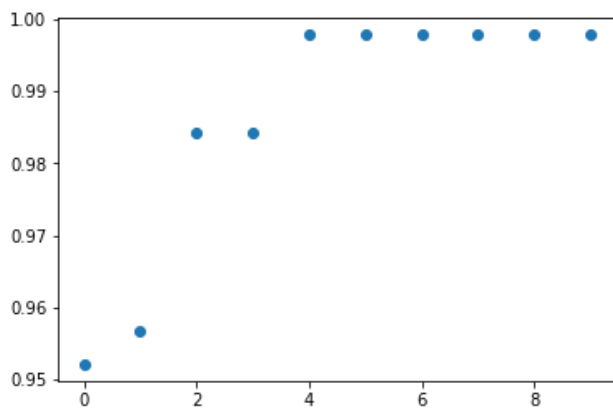
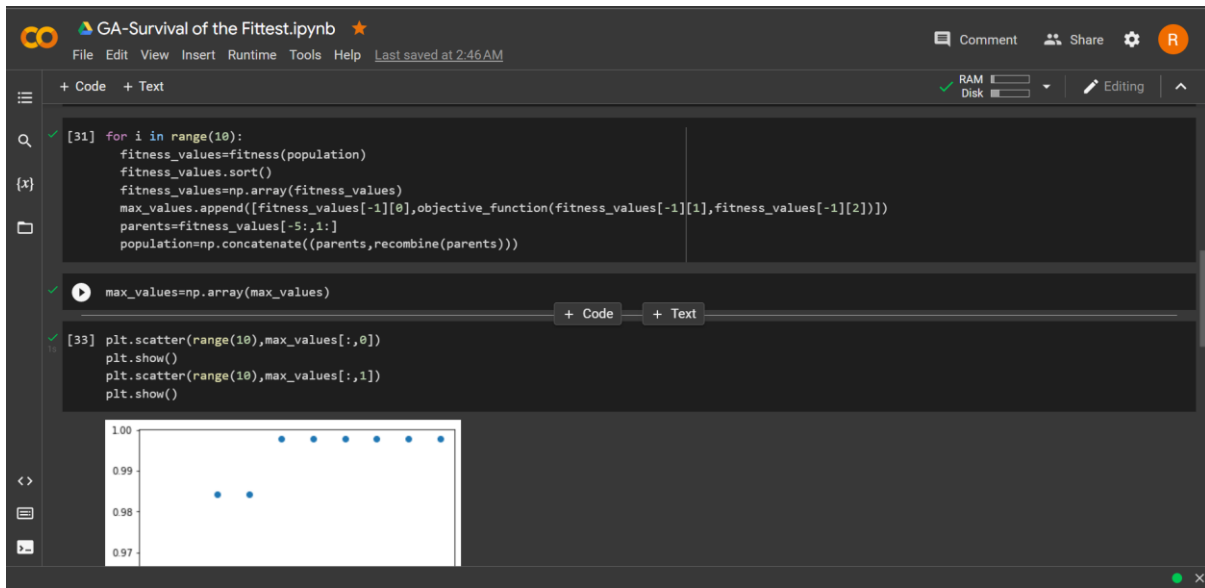
[30] max_values=[]
    five_best_values=[]

[31] for i in range(10):
    fitness_values=fitness(population)
    fitness_values.sort()
    fitness_values=np.array(fitness_values)
    max_values.append([fitness_values[-1][0],objective_function(fitness_values[-1][1],fitness_values[-1][2])])
    parents=fitness_values[-5:,1:]
    population=np.concatenate((parents,recombine(parents)))

[32] max_values=np.array(max_values)
```

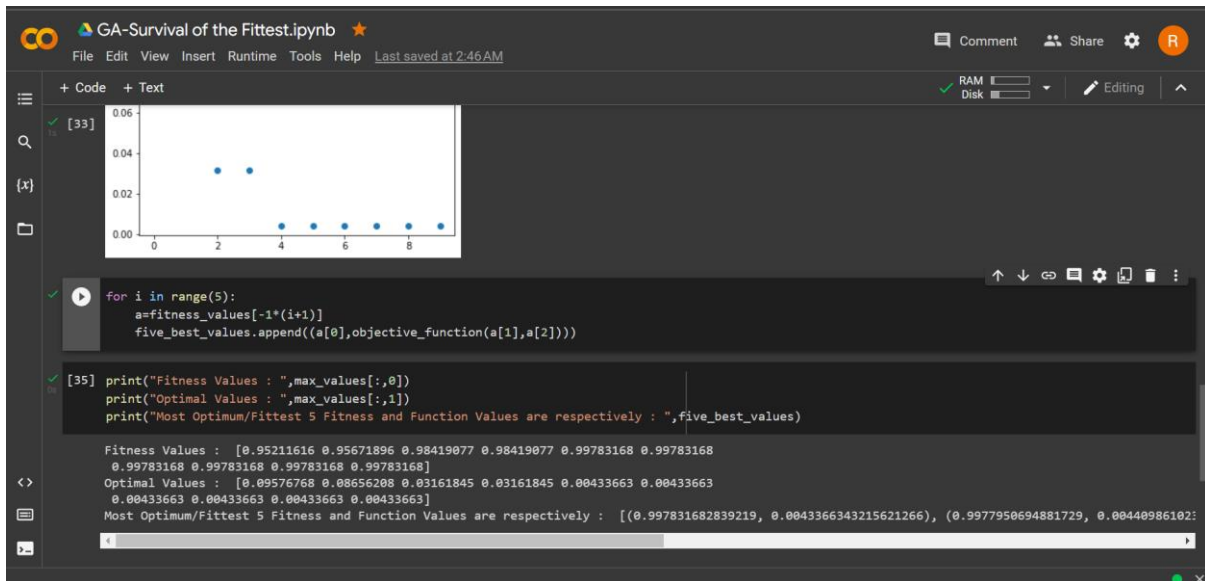
EE345-AI: Assignment-6

Name-Rohan Saha | RollNo.-20085086 | Electrical



EE345-AI: Assignment-6

Name-Rohan Saha | RollNo.-20085086 | Electrical



Screenshot of the Output giving the last 5 Best Values along with their Fitness :-



Fitness Values : [0.95211616 0.95671896 0.98419077 0.98419077 0.99783168
0.99783168
0.99783168 0.99783168 0.99783168 0.99783168]

Optimal Values : [0.09576768 0.08656208 0.03161845 0.03161845 0.00433663
0.00433663
0.00433663 0.00433663 0.00433663 0.00433663]

Most Optimum/Fittest 5 Fitness and Function Values are respectively :
[(0.997831682839219, 0.0043366343215621266), (0.9977950694881729,
0.004409861023654269), (0.9977762542565995, 0.0044474914868009585),
(0.9977463593469373, 0.004507281306125386), (0.9976637936789635,
0.0046724126420729165)]