Md. Mahbubur Rahman

19101069, Section – 05

CSE422 Lab 02

GitHub:

import numpy as np

def fitness(population, n):

n= []

for i in population:

cumulative = 0

for j in range(len(i)):

if i[j] == 1:

amount = int(data[j].split()[1])

category = data[j].split()[0]

if category == "l":

cumulative -= amount

elif category == "d":

cumulative += amount

n.append((cumulative))

return n

def select(population, fit):

# a = [0,1,2,3,4]

# size = 1

# p = [.31, .29, 0.26, 0.14]

pp = [k/np.sum(fit) for k in fit]

a, b = np.random.choice(len(population), 2, pp)

return population[a], population[b]

def crossover(x, y):

idx = np.random.randint(0, len(x))

child\_chromosome = np.concatenate((x[:idx],y[idx:]))

return child\_chromosome

def mutate(child):

mutate\_idx = np.random.randint(0,len(child))

# if str\_list[index] == '1':

# str\_list[index] = '0'

if child[mutate\_idx] == 1:

child[mutate\_idx] = 0

else:

child[mutate\_idx] = 1

return child

def GA(population, n, mutation\_threshold = 0.3):

iter = 2000

for i in range(iter):

n = fitness(population, n)

list = []

# while iter < 1000:

# fit = fitness(population,data)

# if fit.count(0) != 0:

for j in range(len(population)):

x, y = select(population, n)

child = crossover(x,y)

random = np.random.random()

if random < mutation\_threshold:

child = mutate(child)

if all (k == 0 for k in child):

list.append(child)

continue

if fitness([child], n)[0] == 0:

output=""

for i in child:

output += str(i)

return output

list.append(child)

population = list

return -1

def genetic\_algorithm(filename):

file=open(filename, "r")

n = int(file.read(2))

global data

data = []

data.append(file.readline(n))

for x in range(1, n):

lines = file.readline()[:-1]

data.append(lines)

start\_population = 10

mutation\_threshold = 0.3

population = np.random.randint(0, 2, (start\_population, n))

print(GA(population, n, mutation\_threshold))

genetic\_algorithm("input1.txt")

# genetic\_algorithm("input2.txt")