



Clase6: 24 Agosto Actividad 2

🕒 Created	@August 24, 2023 6:09 PM
📅 Class	Algoritmos evolutivos 🧬

Actividad 2

```
from random import randrange, uniform
import numpy as np
#prueba_pesos = [8.61, 8.2, 1.51]
poblacion = [[1,0,1],[1,1,1],[1,0,0],[0,0,0],[1,1,0]]
prueba_poblacion = poblacion[0]
#print(prueba_poblacion)

p_rel = [[11.3],[5.5],[19.0],[7.7]]
#print(poblacion[1])
#print(5.8 + 5.5)

#print(frand_rounded)
def pesos_aleatorios(cantidad):

    lista = []
    for x in range(cantidad):
        aleatorio = uniform(1, 10)
        aleatorio_rounded = round(aleatorio, 1)
        lista.append(aleatorio_rounded)

    return lista

prueba_pesos2 = pesos_aleatorios(3)

def knapsack_capacity(pesos):
    suma = np.sum(pesos)/2

    return suma

def rel_sum(poblacion):
    pesos = [5.8, 5.5, 7.7]
```

```

resultados = []

for fila in poblacion:
    suma = 0.0
    for i in range(len(pesos)):
        suma += fila[i] * pesos[i]
    resultados.append([suma])

return resultados

def relacion_suma_individual(pesos, poblacion):
    array_1 = np.array(pesos)
    array_2 = np.array(poblacion)

    resultado = np.dot(array_1, array_2)

    return resultado

def RP(poblacion,w,V):
    resultados = []
    index = 1
    for fila in poblacion:
        # fila = poblacion
        xj = fila
        print(f"Cromosoma Numero {index} -----")
        print(f"Cromosoma Inicial-> {xj}")
        # pesos = [5.8, 5.5, 7.7]
        # cromosomas -> [1,1,0]
        print(f"Pesos -> {w}")
        print(f"Capacidad De Mochila -> {V}")

        suma_rela = relacion_suma_individual(xj,w)
        #print(f"Sumatoria -> {suma_rela}")
        knapsak_full = False
        if(suma_rela > V):
            knapsak_full = True

        while knapsak_full == True:
            for i in range(len(xj)):
                if xj[i] == 1:
                    xj[i] = 0
                    suma_rela_2 = relacion_suma_individual(xj,w)
                    #print(suma_rela_2)
                    if suma_rela_2 < V:
                        knapsak_full = False
                        break

            resultados.append(xj)
            print(f"Cromosoma Final -> {xj}")
            index += 1

    return resultados

print(RP(poblacion,prueba_pesos2,knapsack_capacity(prueba_pesos2)))

```

