

Cap 3. Ejercicios

1. Maximizar la función $f(x) = x \sin(10 \pi x) + 1$, con $x \in [0,1]$.

2. Verdadera democracia. Suponga que usted es el jefe de gobierno y está interesado en que pasen los proyectos de su programa político. Sin embargo, en el congreso conformado por 5 partidos, no es fácil su tránsito, por lo que debe repartir el poder, conformado por ministerios y otras agencias del gobierno, con base en la representación de cada partido. Cada entidad estatal tiene un peso de poder, que es el que se debe distribuir. Suponga que hay 50 curules, distribuya aleatoriamente, con una distribución no uniforme entre los 5 partidos esas curules. Defina una lista de 50 entidades y asígneles aleatoriamente un peso político de 1 a 100 puntos. Cree una matriz de poder para repartir ese poder, usando AGs.

Para este ejercicio, vamos a tener 5 partidos políticos, A, B, C, D y E, entre los cuales se van a repartir 50 curules de forma aleatoria y no uniforme:

Partido	Número de curules
A	10
B	5
C	16
D	12
E	7

A partir de esto, se crearán aleatoriamente los pesos de las 50 entidades, identificadas con índices de 0 al 49, las cuales se van a repartir con respecto a las curules de los partidos, y se determinará que tan bien está repartido el poder político haciendo uso del error cuadrático medio como nuestra función de aptitud:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

El algoritmo genético lo desarrolle en python, usando una población de 8 individuos, cada uno conteniendo 5 listas con el ordenamiento de los individuos entre los distintos partidos políticos. Además, se indicará una probabilidad de mutación del 0.2, 100 generaciones antes de acabar, y una selección por torneo de 3 sobrevivientes. Todo el código en el archivo "Ejercicio_2.py".

Al ejecutar una de las pruebas, obtenemos inicialmente los siguientes pesos para las 50 entidades que estaran siendo distribuidas:

Pesos de las entidades				
36	45	59	65	62
57	90	44	18	99
73	22	22	35	100
36	83	65	80	42
95	11	93	29	94
2	92	57	38	19
33	76	65	57	99
82	14	60	98	1
85	57	88	2	15
4	8	16	85	92

Luego se empieza a evaluar y a pasar las generaciones de individuos, cada vez mejorando nuestra funcion objetivo:

Generación 1, mejor aptitud: 0.003462
 Generación 2, mejor aptitud: 0.002735
 Generación 3, mejor aptitud: 0.002722
 Generación 4, mejor aptitud: 0.002264
 Generación 5, mejor aptitud: 0.002264
 Generación 6, mejor aptitud: 0.002264
 Generación 7, mejor aptitud: 0.002264
 Generación 8, mejor aptitud: 0.002264
 Generación 9, mejor aptitud: 0.002264
 Generación 10, mejor aptitud: 0.002264
 Generación 11, mejor aptitud: 0.002264
 Generación 12, mejor aptitud: 0.002264
 Generación 13, mejor aptitud: 0.002170
 Generación 14, mejor aptitud: 0.001589
 Generación 15, mejor aptitud: 0.001472
 Generación 16, mejor aptitud: 0.001285
 Generación 17, mejor aptitud: 0.001108
 Generación 18, mejor aptitud: 0.001108
 Generación 19, mejor aptitud: 0.001101
 Generación 20, mejor aptitud: 0.001098
 Generación 21, mejor aptitud: 0.000986
 Generación 22, mejor aptitud: 0.000463
 Generación 23, mejor aptitud: 0.000463
 Generación 24, mejor aptitud: 0.000463
 Generación 25, mejor aptitud: 0.000460
 Generación 26, mejor aptitud: 0.000239
 Generación 27, mejor aptitud: 0.000213
 ...
 Generación 95, mejor aptitud: 0.000213
 Generación 96, mejor aptitud: 0.000176
 Generación 97, mejor aptitud: 0.000157
 Generación 98, mejor aptitud: 0.000157
 Generación 99, mejor aptitud: 0.000120
 Generación 100, mejor aptitud: 0.000118

Al final se logra encontrar un individuo que logra reordenar los puestos hasta llegar a una solución bastante cercana a cero, o sea bastante buena al dejar bastante equitativamente distribuido el poder entre los partidos políticos:

Mejor individuo:

A		527		0.195185		[13, 32, 33, 33, 19, 24, 24, 16]	
B		482		0.178519		[6, 14, 14, 14, 26]	
C		530		0.196296		[21, 29, 41, 42, 43, 44, 45, 46, 47, 48, 49, 28, 28, 5]	
D		516		0.191111		[7, 10, 15, 23, 39, 39, 31, 38, 9, 2]	
E		530		0.196296		[20, 18, 20, 20, 18, 40]	

3. Una empresa proveedora de energía eléctrica dispone de cuatro plantas de generación para satisfacer la demanda diaria de energía eléctrica en Cali, Bogotá, Medellín y Barranquilla. Cada una puede generar 3, 6, 5 y 4 GW al día respectivamente. Las necesidades de Cali, Bogotá, Medellín y Barranquilla son de 4, 3, 5 y 3 GW al día respectivamente. Los costos por el transporte de energía por cada GW entre plantas y ciudades se dan en la siguiente tabla:

	Cali	Bogotá	Medellín	Barranq.
Planta C	1	4	3	6
Planta B	4	1	4	5
Planta M	3	4	1	4
Planta B	6	5	4	1

Los costos del KW-H por generador se dan en la siguiente tabla:

Generador	\$KW-H
Planta C	680
Planta B	720
Planta M	660
Planta B	750

Encontrar usando AGs el mejor despacho de energía minimizando los costos de transporte y generación.

4. Genere aleatoriamente una población de 50 matrices de 120 por 180, con números de 0 a 255, preséntelas como una gráfica RGB. La función de aptitud es una imagen cualquiera. Evolucione la población inicial hasta llegar a la imagen.

5. Genere aleatoriamente una población de 50 conversaciones de 5 segundos, que se escuche por el parlante del computador. La función de aptitud es una conversación suya de 5 segundos. Evolucione la población inicial hasta llegar a su conversación.

6. Genere aleatoriamente una población de 50 palabras, que se escuche por el parlante del computador. Tomando como función de aptitud una palabra suya, usando AGs, con base en las palabras generadas aleatoriamente llegue a la palabra que usó como función de aptitud.

Desarrolle tres ejercicios de los 6 propuestos, utilice todas las librerías y herramientas disponibles.