

Monstrinhos de Algoritmos Genéticos (GA)

1 Comparando as performances

Objetivo: Compare a performance de três algoritmos diferentes de otimização (busca aleatória, busca em grade e algoritmos genéticos) para resolver o problema das caixas binárias.

Lembrete: nós estudamos performance de algoritmos na disciplina de lógica computacional.

Dica: O enunciado do objetivo não definiu o número de caixas. É esperado de um cientista que ele entenda que não adianta resolver esse problema para apenas *um* valor de n caixas, mas sim buscar uma tendência resolvendo esse problema para alguns valores de n diferentes em busca de encontrar um padrão. Considere pelo menos 10 valores de n diferentes na sua resolução.

Dica 2: Lembre-se de que o único algoritmo determinístico dos três sendo estudados é o de busca em grade. Sendo assim, é esperado que um cientista entenda que situações não-determinísticas demandam o uso de estatística para quantificar um valor médio de performance e seu desvio.

2 A senha de tamanho variável

Objetivo: Resolver o problema da senha sem fornecer a informação do tamanho da senha para a função que gera a população. Considere que a senha pode ser qualquer string de 1 até 30 caracteres.

Dica: A função objetivo terá que quantificar em sua métrica tanto se o candidato acertou as letras quanto se acertou o tamanho da senha.

Dica 2: Você pode criar diferentes estratégias de mutação, não precisa ser apenas uma! Quem sabe uma função de mutação pode alterar letras e a outra pode alterar o tamanho da senha?

Dica 3: Observe que você terá que pensar um pouco sobre como fará o cruzamento no caso de senhas de tamanhos diferentes. Quem sabe tenha que fazer alguma consideração adicional sobre quais são os valores possíveis para o ponto de corte...

3 O caixeiro que prefere cidades ímpares

Objetivo: Encontre o caminho de menor distância no problema do caixeiro viajante que prefere cidades ímpares e mostre ele de forma gráfica.

Considerações do experimento: Considere um número $n \geq 7$ de coordenadas (x, y) de cidades (cada cidade ocupa uma posição (x, y) diferente). Você pode gerar as coordenadas de forma aleatória ou simplesmente usar as coordenadas que desejar. O caixeiro só anda em linha reta e apenas entre duas cidades. O caixeiro começa e termina seu trajeto na mesma cidade e, fora a cidade inicial, ele não visita nenhuma outra cidade mais de uma vez. Além disso, atribua um número inteiro para cada uma das n cidades que o caixeiro irá visitar, iniciando a contagem pelo número zero e aumentando esse número de 1 em 1. O caixeiro deverá necessariamente visitar primeiro as cidades com números ímpares antes das cidades com números pares. A cidade de número zero deve ser a cidade inicial.

4 Novos palíndromos

Objetivo: Encontre pelo menos 10 palíndromos de 5 letras. Estes palíndromos devem ter pelo menos uma vogal. Não é necessário que eles formem palavras válidas em português ou qualquer outro idioma.

Dica: Nada te impede de inventar um novo operador de mutação. Existe uma forma de melhorar bastante a convergência deste problema com um operador de mutação especializado para esta tarefa.

5 A liga ternária mais cara do mundo

Objetivo: Encontre uma liga de três elementos que tenha o maior custo possível. A liga ternária deve ser da forma $x\text{A}.y\text{B}.z\text{C}$ sendo que $x + y + z = 100$ g, $x \geq 5$ g, $y \geq 5$ g, $z \geq 5$ g e "A", "B" e "C" são elementos químicos diferentes. Utilize o preço dado abaixo [1]. Considere que qualquer composto com 3 elementos químicos é chamado de liga.

Dica: Pode ser interessante criar uma função que gere a população inicial de forma a garantir que $x + y + z = 100$ g.

Dica 2: Pode ser interessante criar uma ou mais funções de cruzamento e mutação neste problema de forma que todas elas garantam que, ao final do processo do operador, os indivíduos mantenham a característica de ter $x + y + z = 100$ g.

```
# preço em dólares por kilograma
```

```
preco = {  
    "H": 1.39,  
    "He": 24,  
    "Li": 85.6,  
    "Be": 857,  
    "B": 3.68,  
    "C": 0.122,  
    "N": 0.14,  
    "O": 0.154,  
    "F": 2.16,  
    "Ne": 240,  
    "Na": 3.43,  
    "Mg": 2.32,  
    "Al": 1.79,  
    "Si": 1.7,  
    "P": 2.69,  
    "S": 0.0926,  
    "Cl": 0.082,  
    "Ar": 0.931,  
    "K": 13.6,  
    "Ca": 2.35,  
    "Sc": 3460,  
    "Ti": 11.7,  
    "V": 385,  
    "Cr": 9.4,  
    "Mn": 1.82,  
    "Fe": 0.424,  
    "Co": 32.8,  
    "Ni": 13.9,  
    "Cu": 6,  
    "Zn": 2.55,  
    "Ga": 148,  
    "Ge": 1010,  
    "As": 1.31,  
    "Se": 21.4,  
    "Br": 4.39,  
    "Kr": 290,  
    "Rb": 15500,
```

"Sr": 6.68,
"Y": 31,
"Nb": 85.6,
"Mo": 40.1,
"Tc": 100000,
"Ru": 10600,
"Rh": 147000,
"Pd": 49500,
"Ag": 521,
"Cd": 2.73,
"In": 167,
"Sn": 18.7,
"Sb": 5.79,
"Te": 63.5,
"I": 35,
"Xe": 1800,
"Cs": 61800,
"Ba": 0.275,
"La": 4.92,
"Ce": 4.71,
"Pr": 103,
"Nd": 57.5,
"Pm": 460000,
"Sm": 13.9,
"Eu": 31.4,
"Gd": 28.6,
"Tb": 658,
"Dy": 307,
"Ho": 57.1,
"Er": 26.4,
"Tm": 3000,
"Yb": 17.1,
"Lu": 643,
"Hf": 900,
"Ta": 312,
"W": 35.3,
"Re": 4150,
"Os": 12000,
"Ir": 56200,
"Pt": 27800,
"Hg": 30.2,
"Tl": 4200,
"Pb": 2,
"Bi": 6.36,

```

"Po": 492000000000000,
"Ac": 290000000000000,
"Th": 287,
"Pa": 280000,
"U": 101,
"Np": 660000,
"Pu": 6490000,
"Am": 750000,
"Cm": 160000000000,
"Bk": 185000000000,
"Cf": 185000000000,
}

```

6 A liga ternária leve mais cara do mundo

Objetivo: Encontre uma liga de três elementos que tenha o maior custo e o menor peso atômico. A liga ternária deve ser da forma $x\text{A}.y\text{B}.z\text{C}$ sendo que $x + y + z = 100$ g, $x \geq 5$ g, $y \geq 5$ g, $z \geq 5$ g e "A", "B" e "C" são elementos químicos diferentes. Utilize o preço dado no exercício anterior e peso atômico dados abaixo [3]. Considere que qualquer composto com 3 elementos químicos é chamado de liga.

```

# peso atômico em gramas por mol
peso_atomico = {
    "H": 1.008,
    "He": 4.002602,
    "Li": 6.94,
    "Be": 9.0121831,
    "B": 10.81,
    "C": 12.011,
    "N": 14.007,
    "O": 15.999,
    "F": 18.998403163,
    "Ne": 20.1797,
    "Na": 22.98976928,
    "Mg": 24.305,
    "Al": 26.9815385,
    "Si": 28.085,
    "P": 30.973761998,
    "S": 32.06,
    "Cl": 35.45,
    "Ar": 39.948,
    "K": 39.0983,
    "Ca": 40.078,

```

"Sc": 44.955908,
"Ti": 47.867,
"V": 50.9415,
"Cr": 51.9961,
"Mn": 54.938044,
"Fe": 55.845,
"Co": 58.933194,
"Ni": 58.6934,
"Cu": 63.546,
"Zn": 65.38,
"Ga": 69.723,
"Ge": 72.63,
"As": 74.921595,
"Se": 78.971,
"Br": 79.904,
"Kr": 83.798,
"Rb": 85.4678,
"Sr": 87.62,
"Y": 88.90584,
"Nb": 92.90637,
"Mo": 95.95,
"Tc": 97.90721,
"Ru": 101.07,
"Rh": 102.9055,
"Pd": 106.42,
"Ag": 107.8682,
"Cd": 112.414,
"In": 114.818,
"Sn": 118.71,
"Sb": 121.76,
"Te": 127.6,
"I": 126.90447,
"Xe": 131.293,
"Cs": 132.90545196,
"Ba": 137.327,
"La": 138.90547,
"Ce": 140.116,
"Pr": 140.90766,
"Nd": 144.242,
"Pm": 144.91276,
"Sm": 150.36,
"Eu": 151.964,
"Gd": 157.25,
"Tb": 158.92535,

```

"Dy": 162.5,
"Ho": 164.93033,
"Er": 167.259,
"Tm": 168.93422,
"Yb": 173.045,
"Lu": 174.9668,
"Hf": 178.49,
-Ta": 180.94788,
"W": 183.84,
"Re": 186.207,
"Os": 190.23,
"Ir": 192.217,
"Pt": 195.084,
"Hg": 200.592,
-Tl": 204.38,
"Pb": 207.2,
-Bi": 208.9804,
-Po": 209.0,
-Ac": 227.0,
-Th": 232.0377,
-Pa": 231.03588,
-U": 238.02891,
-Np": 237.0,
-Pu": 244.0,
-Am": 243.0,
-Cm": 247.0,
-Bk": 247.0,
-Cf": 251.0,
}

```

7 Referências

1. Material suplementar de FORREST, Robert M.; GREER, A. Lindsay. Evolutionary design of machine-learning-predicted bulk metallic glasses. Digital Discovery, 2023.
2. MENTEL, Łukasz. mendeleev – A Python resource for properties of chemical elements, ions and isotopes. Disponível em: <https://github.com/lmmentel/mendeleev>.
3. EYAL WIRSANSKY. Hands-On Genetic Algorithms with Python: Applying genetic algorithms to solve real-world deep learning and artificial intelligence problems, 2020.

