

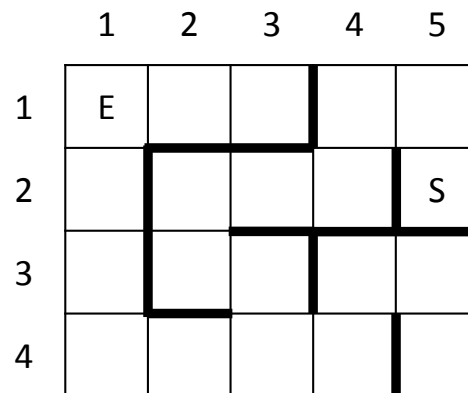
REDES NEURAIS / INTELIGÊNCIA ARTIFICIAL

LISTA DE EXERCÍCIOS 6

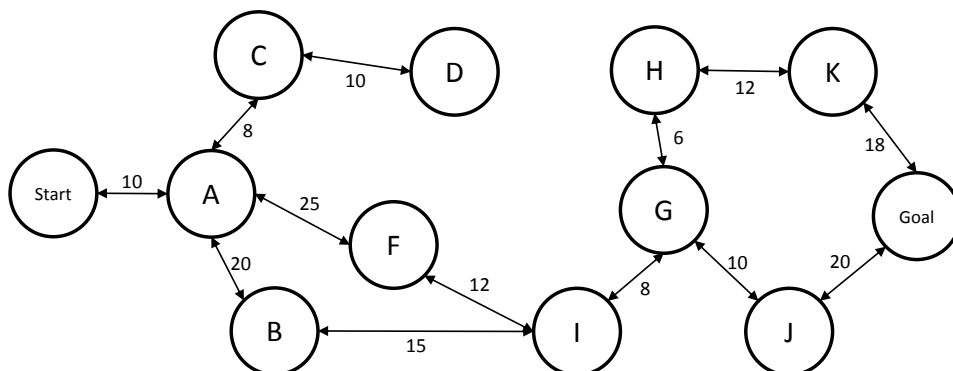
Aluno:

1. Defina o problema de busca (espaço de estados, estado inicial, estado final, ações possíveis, custo) para o seguinte caso: uma pessoa, um lobo, um carneiro e um cesto de alface estão à beira de um rio. Dispondo de um barco no qual pode carregar apenas um dos outros três, a pessoa deve transportar tudo para a outra margem. Em nenhum momento devem ser deixados juntos e sozinhos o lobo e o carneiro ou o carneiro e o cesto de alface.
2. Em um labirinto, mostrado na figura a seguir, um robô é colocado na célula inicial indicada por "E" e deve encontrar um caminho até a saída, denotada pela letra "S". O robô não pode se mover na diagonal, somente acima, abaixo, direita e esquerda. Ele também não pode atravessar paredes (as linhas mais grossas da grade) ou as bordas do labirinto, de modo que ele é forçado a contornar obstáculos. Felizmente, o robô possui um mapa do ambiente. A solução é o caminho mais curto até a saída e todos os movimentos do robô possuem os mesmos custos.

- (a) Descreva o problema em termos de um problema de busca definindo o espaço de estados, o estado inicial, o estado final, os operadores de transição entre os estados (ações) e o custo.
- (b) Construa um grafo do espaço de estados rotulando os arcos com os operadores de transição adequados.

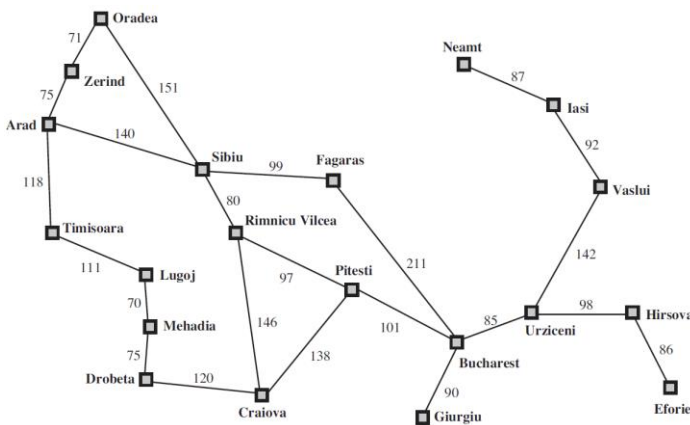


3. Considerando o seguinte mapa:



Responda as questões abaixo considerando “Start” como o estado inicial e “Goal” o estado final buscado.

- (a) Monte as árvores de busca que seriam geradas pelos algoritmos de busca cega vistos em aula (busca em largura, busca de custo uniforme, busca em profundidade, busca com aprofundamento iterativo, busca bidirecional).
- (b) Qual dos algoritmos apresentou melhor resultado? Considerando o custo do caminho e o número de nós avaliados até que a solução fosse encontrada.
4. Realize uma busca utilizando o algoritmo A* para encontrar o melhor caminho para chegar a Bucharest partindo de Lugoj. Construa a árvore de busca criada pela execução do algoritmo apresentando os valores de $f(n)$, $g(n)$ e $h(n)$ para cada nó. Utilize a heurística de distância em linha reta.



Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374
Hirsova	151	Urziceni	80

5. Você precisa construir um sistema para auxiliar os usuários do metro de Paris a encontrar a melhor rota entre as diversas estações do metro.

Considere que:

- O metro tem somente 4 linhas (Figura 1);
- A distância real entre duas estações é dada pela tabela 1 e a distância em linha reta é dada pela tabela 2;
- A velocidade média de um trem é de 40 km/h;
- O tempo gasto para trocar de linha dentro da mesma estação é de 5 minutos;

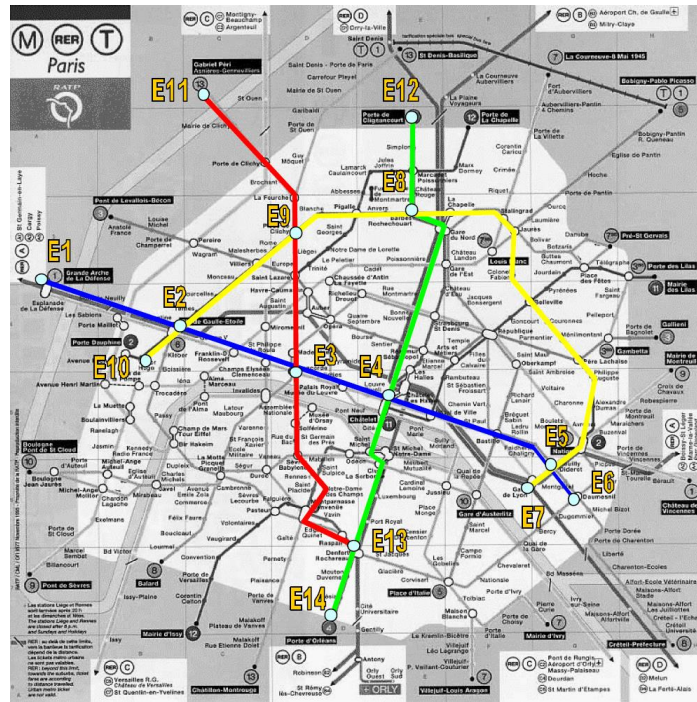


Figura 1: Linhas do metro de Paris.

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14
E1	-	10												
E2		-	8,5						10	3,5				
E3			-	6,3					9,4				18,7	
E4				-	13			15,3					12,8	
E5					-	3	2,4	30						
E6						-								
E7							-							
E8								-	9,6			6,4		
E9									-		12,2			
E10										-				
E11											-			
E12												-		
E13													-	5,1
E14														-

Tabela 1: Distancias reais entre as estações do metro de Paris.

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14
E1	-	10	18,5	24,8	36,4	38,8	35,8	25,4	17,6	9,1	16,7	27,3	27,6	29,8
E2		-	8,5	14,8	26,6	29,1	26,1	17,3	10	3,5	15,5	20,9	19,1	21,8
E3			-	6,3	18,2	20,6	17,6	13,6	9,4	10,3	19,5	19,1	12,1	16,6
E4				-	12	14,4	11,5	12,4	12,6	16,7	23,6	18,6	10,6	15,4
E5					-	3	2,4	19,4	23,3	28,2	34,2	24,8	14,5	17,9
E6						-	3,3	22,3	25,7	30,3	36,7	27,6	15,2	18,2
E7							-	20	23	27,3	34,2	25,7	12,4	15,6
E8								-	8,2	20,3	16,1	6,4	22,7	27,6
E9									-	13,5	11,2	10,9	21,2	26,6
E10										-	17,6	24,2	18,7	21,2
E11											-	14,2	31,5	35,5
E12												-	28,8	33,6
E13													-	5,1
E14														-

Tabela 2: Distancias em linha reta entre as estações do metro de Paris.

- (a) Descreva o problema em termos de um problema de busca definindo o espaço de estados, o estado inicial, o estado final, os operadores de transição entre os estados (ações) e a função de avaliação utilizada pelo algoritmo de busca heurística A*.

(b) Realize uma busca utilizando o algoritmo A* para encontrar o melhor caminho para chegar a estação E7 partindo da estação E13. Construa a árvore de busca criada pela execução do algoritmo apresentando os valores de $f(n)$, $g(n)$ e $h(n)$ para cada nó.

6. Problema SAT: seja $x = (x_1, x_2, \dots, x_n)$ um vetor de n variáveis booleanas (i.e. cada variável x_i assume um dos valores em $\{0,1\}$). Seja $f(x) = [c_1(x) \wedge c_2(x) \wedge \dots \wedge c_m(x)]$ uma fórmula normal conjuntiva com m cláusulas, onde cada cláusula $c_j(x)$ é uma disjunção de literais, e um literal é uma das variáveis booleanas ou sua negação.

Por exemplo, considere um vetor com três variáveis $x = (x_1, x_2, x_3)$. Um exemplo de fórmula normal conjuntiva seria:

$$f(x) = [(x_1) \wedge (\neg x_2) \wedge (x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_3)]$$

Composta pelas seguintes cláusulas:

$$c_1(x) = (x_1)$$

$$c_2(x) = (\neg x_2)$$

$$c_3(x) = (x_2 \vee \neg x_3)$$

$$c_4(x) = (x_1 \vee \neg x_3)$$

Uma fórmula é dita *satisfatível* quando existe uma atribuição de valores para (x_1, x_2, \dots, x_n) tal que todas as cláusulas da fórmula sejam satisfeitas, isto é, $c_j(x) = 1$ para $j=1, \dots, m$. No exemplo acima, $f(x)$ é satisfatível e $x = (1, 0, 0)$ é uma possível atribuição de valores para as variáveis x_1, x_2 e x_3 que tornam verdadeiras todas as quatro cláusulas da fórmula.

O problema SAT consiste em: dada uma fórmula, responder se a fórmula é satisfatível ou não. Encontrar uma atribuição de valores que satisfaçam uma dada fórmula é uma tarefa que pode ser formulada como um problema de busca. Assim, para um conjunto qualquer de variáveis $x = (x_1, x_2, \dots, x_n)$ e uma dada fórmula $f(x) = [c_1(x) \wedge c_2(x) \wedge \dots \wedge c_m(x)]$, proponha uma solução para o problema SAT como Algoritmos Genéticos.

Para isso, use o seguinte caso base:

$$(\neg x \vee \neg z \vee y) \wedge (\neg y \vee z) \wedge (x \vee \neg z) \wedge (y) \wedge (\neg y \vee \neg x \vee \neg w \vee z)$$

- Proponha uma maneira de codificar os cromossomos.
- Defina uma função de aptidão para avaliar a qualidade dos cromossomos.
- Defina como o método de seleção dos pais será utilizado.
- Defina os operadores genéticos de recombinação e mutação.
- Gere uma população inicial de 4 cromossomos e avalie a aptidão deles.
- Aplique os operadores de recombinação e mutação sobre essa população para gerar uma nova geração, em seguida avalie a aptidão da nova geração. Repita esse processo por 8 gerações ou até que a solução do problema seja encontrada.

7. Deseja-se construir um sistema para determinar os dias bons para jogar golfe. Para isso, foi coletado um conjunto de exemplos que incluem informações sobre o Clima, Temperatura, Humidade, Vento, e também a decisão que indica se aquele caso é um bom dia para jogar golfe (Saída/Classe). Com base nesse conjunto de exemplos, construa uma Árvore de Decisão baseada no cálculo da entropia dos atributos. Apresente a Árvore de Decisão criada pelo algoritmo e todos os cálculos da entropia dos atributos.

	Clima	Temperatura	Humidade	Vento	Saída/Classe
X₁	Ensolarado	Quente	Alta	Não	Não
X₂	Ensolarado	Quente	Alta	Sim	Não
X₃	Nublado	Quente	Alta	Não	Sim
X₄	Chuvoso	Suave	Alta	Não	Sim
X₅	Chuvoso	Frio	Normal	Sim	Não
X₆	Nublado	Frio	Normal	Sim	Sim
X₇	Ensolarado	Suave	Alta	Não	Não
X₈	Ensolarado	Frio	Normal	Não	Sim
X₉	Chuvoso	Suave	Normal	Não	Sim
X₁₀	Ensolarado	Suave	Normal	Sim	Sim
X₁₁	Nublado	Suave	Alta	Sim	Sim
X₁₂	Chuvoso	Suave	Alta	Sim	Não

8. Com base nas imagens dos Simpsons utilizadas na Lista de Exercícios 5, selecione um conjunto de pelo menos 4 características numéricas para serem utilizadas no algoritmo KNN.
- (a) Utilizando as características escolhidas, crie um conjunto de treinamento (com pelo menos 10 exemplos) utilizando algumas das imagens que estão no conjunto de treinamento.
- (b) Selecione 5 imagens do conjunto de teste, gere o vetor de características destas novas imagens e realize a classificação dos exemplos utilizando o algoritmo KNN. Em seguida, calcule a precisão da classificação.
9. Considere o seguinte conjunto de treinamento:

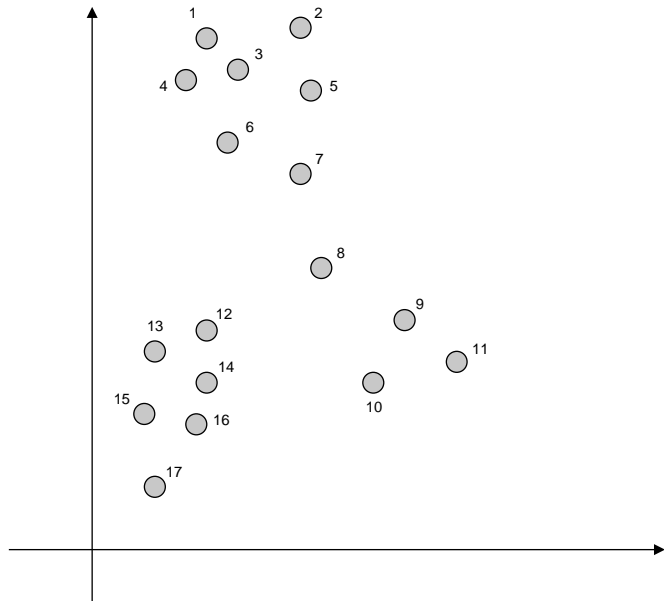
X1	X2	X3	X4	Classe
0	1	1	0	0
1	0	1	0	0
1	1	1	1	1
0	1	1	1	0

- a) Defina (desenhe) a estrutura de um Perceptron para realizar a classificação deste conjunto de dados.
- b) Inicialize os pesos aleatoriamente e, em seguida, mostre como os pesos serão modificados durante o processo de treinamento do Perceptron. Use uma taxa de

aprendizagem de 0.1 e um Threshold Linear de 0.2. Mostre todos os cálculos realizados.

10. Considere o seguinte conjunto de exemplos não rotulados:

ID	X1	X2
1	1.9	7.3
2	3.4	7.5
3	2.5	6.8
4	1.5	6.5
5	3.5	6.4
6	2.2	5.8
7	3.4	5.2
8	3,6	4
9	5	3.2
10	4.5	2.4
11	6	2.6
12	1.9	3
13	1	2.7
14	1.9	2.4
15	0.8	2
16	1.6	1.8
17	1	1



Utilize o algoritmo K-Means para encontrar as 3 classes existentes. Calcule e mostre a posição dos centróides durante todas as iterações do algoritmo.