

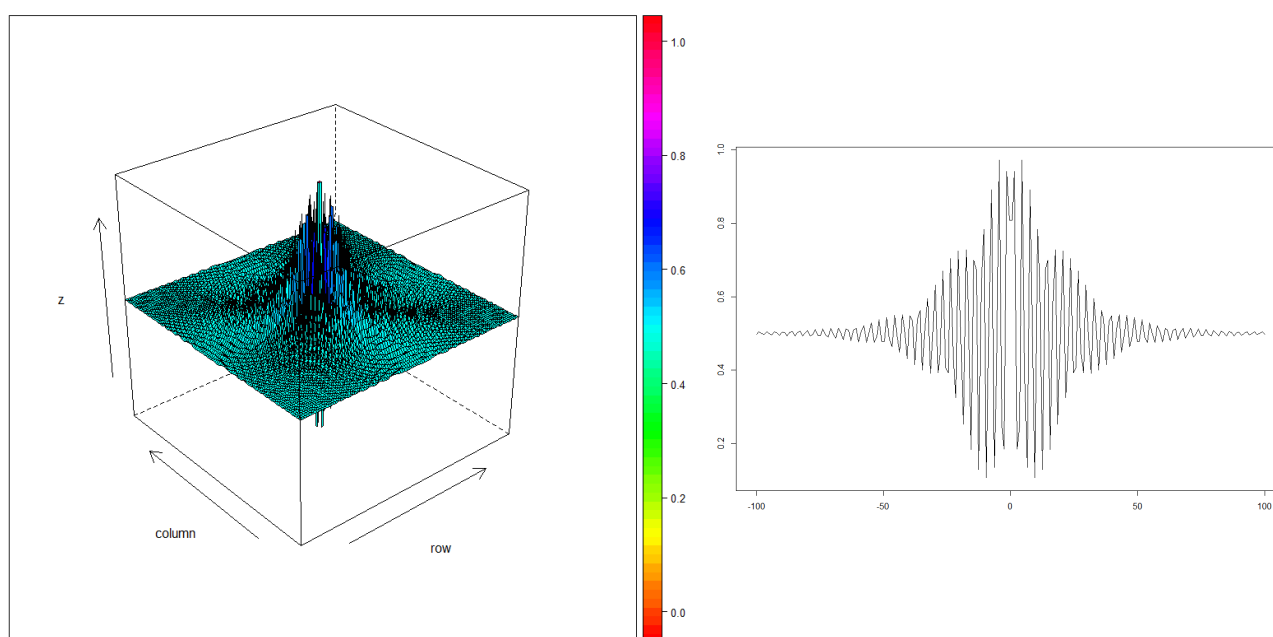
Tema: Computação Evolucionária

Objetivos: Familiarização com os principais conceitos teóricos e práticos da Computação Evolucionária e, em particular, dos Algoritmos Genéticos.

Exercício 1

Considere a seguinte função, bem como as vistas 2D e 3D do gráfico que ela gera, para valores de x e y entre -100 e 100.

$$f(x, y) = 0.5 + \frac{\sin(\sqrt{x^2 + y^2})^2 - 0.5}{1 + 0.001 * (x^2 + y^2)^2}$$



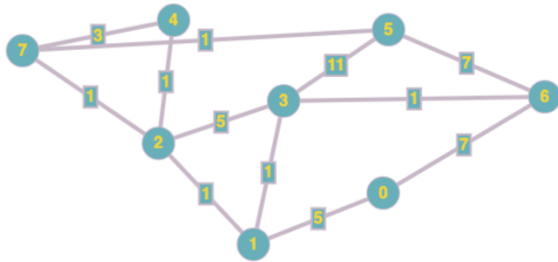
Neste domínio, pretende-se encontrar o valor máximo da função, entre -100 e 100.

Para este problema:

- Implemente uma função de inicialização que gera soluções aleatórias para este problema.
- Implemente uma função de fitness que, dada uma solução, devolve um valor numérico que indica quão boa a solução é.
- Inicialize uma população de soluções e ordene-as por fitness descendente. Comprove, através da observação do gráfico, que as soluções obtidas e o seu fitness aparentam estar corretas.

Exercício 2

Considere o seguinte grafo, cuja matriz de adjacências é dada ao lado (em Java).



```
int matriz[][]={{0, 5, 0, 0, 0, 0, 7, 0},
{5, 0, 1, 1, 0, 0, 0, 0},
{0, 1, 0, 5, 1, 0, 0, 1},
{0, 1, 5, 0, 0, 11, 1, 0},
{0, 0, 1, 0, 0, 0, 0, 3},
{0, 0, 0, 11, 0, 0, 7, 1},
{7, 0, 0, 1, 0, 7, 0, 0},
{0, 0, 1, 0, 3, 1, 0, 0}};
```

Neste domínio, pretende-se encontrar o caminho mais curto entre 2 pontos.

Para este problema:

- Implemente uma função de inicialização que gera soluções aleatórias para este problema. A função deve receber como parâmetros o ponto de partida e o ponto de chegada. Relativamente ao tamanho da solução, pode defini-lo como fixo ou aleatoriamente variável (entre determinados limites). Analise ainda a implicação de cada uma destas abordagens.
- Implemente uma função que indica se uma solução é válida ou não.
- Implemente uma função de fitness que, dada uma solução, devolve um valor numérico que indica quão boa a solução é. A função deve funcionar quer para soluções válidas quer inválidas.
- Inicialize uma população de soluções e ordene-as por fitness descendente. Selecione apenas as soluções válidas. Comprove, através da observação do grafo, que as soluções obtidas e o seu fitness estão efetivamente corretas.

Exercício 3

Considere o seguinte conjunto de pontos (em Java).

```
List<Double> targetPoints = Arrays.asList(new Double[]{2.000,2.155,1.640,0.485,-
1.280,-3.625,-6.520,-9.935,-13.840,-18.205,-23.000,-28.195,-33.760,-39.665,-
45.880,-52.375,-59.120,-66.085,-73.240,-80.555,-88.000,-95.545,-103.160,-
110.815,-118.480,-126.125,-133.720,-141.235,-148.640,-155.905,-163.000,-
169.895,-176.560,-182.965,-189.080,-194.875,-200.320,-205.385,-210.040,-
214.255,-218.000,-221.245,-223.960,-226.115,-227.680,-228.625,-228.920,-
228.535,-227.440,-225.605,-223.000,-219.595,-215.360,-210.265,-204.280,-
197.375,-189.520,-180.685,-170.840,-159.955,-148.000,-134.945,-120.760,-
105.415,-88.880,-71.125,-52.120,-31.835,-
10.240,12.695,37.000,62.705,89.840,118.435,148.520,180.125,213.280,248.015,284.3
60,322.345,362.000,403.355,446.440,491.285,537.920,586.375,636.680,688.865,742.9
60,798.995,857.000,917.005,979.040,1043.135,1109.320,1177.625,1248.080,1320.715,
1395.560,1472.645,1552.000});
```

Estes pontos foram gerados por uma função quadrática do 4º grau para valores de x entre 0 e 100. Neste domínio, pretende-se encontrar a função que gerou esta sequência de pontos. Para simplificar o espaço da solução, considere que cada um dos 4 coeficientes da função (a, b, c e d) apenas pode tomar valores entre -9 e 9.

Para este problema:

- Implemente uma função de inicialização que gera soluções aleatórias para este problema.
- Implemente uma função de fitness que, dada uma solução, devolve um valor numérico que indica quão boa a solução é.
- Inicialize uma população de soluções e ordene-as por fitness decendente. Escolha algumas destas funções, trace o seu gráfico e determine se as que estão mais próximas são efetivamente aquelas que têm um melhor valor de fitness.

A Figura 1 mostra os pontos originais (a preto) bem como duas das soluções geradas por um algoritmo genético.

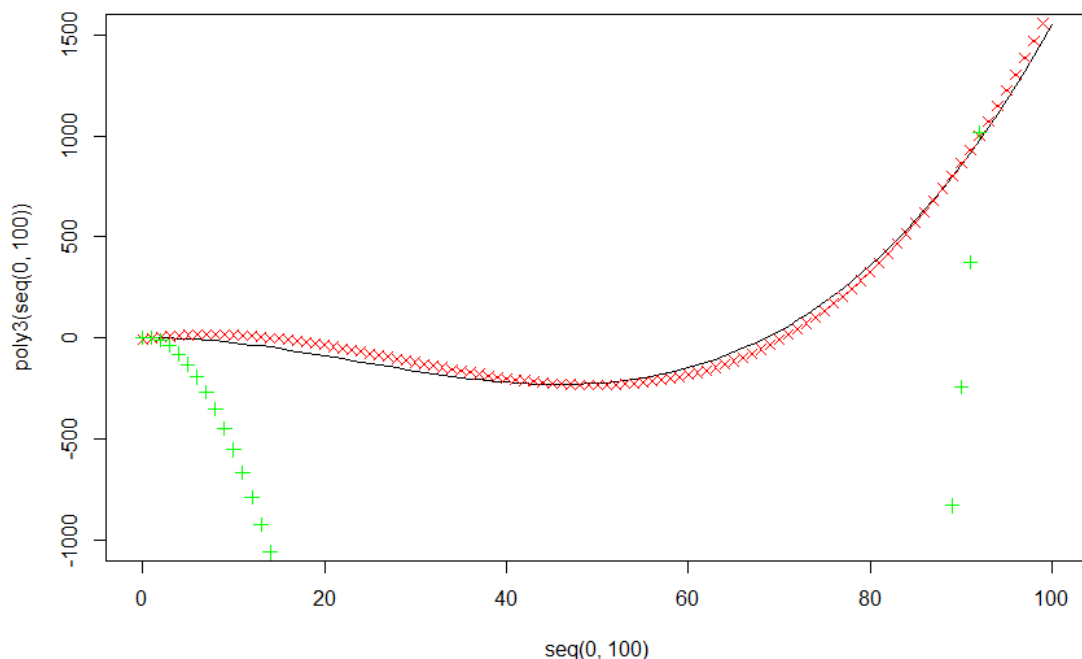


Figura 1: Plot dos pontos originais (preto), da melhor solução da primeira geração de uma execução do algoritmo (verde, $fitness = 5122.38$) e da melhor solução encontrada no final da execução (vermelho, $fitness = 37.78$)

Exercício 4

Considere o seguinte mapa topográfico que mostra a altitude num determinado par de coordenadas, e cuja matriz é dada no fim do exercício (em Java).

[illegible]


Considere que se pretende encontrar o *melhor* caminho entre dois pontos A e B, em que B fica sempre à direita de A. Admita que apenas se viaja da esquerda para a direita e que, em cada “passo”, se pode avançar para qualquer casa da coluna imediatamente a seguir.

Para este problema:

- Implemente uma função de inicialização que gera soluções aleatórias para este problema. A função deve receber como parâmetros o ponto de partida e o ponto de chegada.
- Implemente uma função de fitness que, dada uma solução, devolve um valor numérico que indica quão boa a solução é. A função deve ponderar tanto a distância como a altitude, de acordo com os critérios que pretender. Por exemplo, até um certo ponto, será preferível caminhar um pouco mais indo à volta de uma montanha, do que subindo e descendo a mesma.
- Inicialize uma população de soluções e ordene-as por fitness descendente. Escolha algumas destas soluções, faça o seu plot, e determine, observando o mapa topográfico, se o seu valor de fitness se coaduna com a qualidade da solução.

int

[illegible]

 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>LEI – Licenciatura em Engenharia Informática</p> <p>IA – Inteligência Artificial</p> <p>2º Semestre – Docente: DCarneiro Ficha Prática 4</p>
---	--

- Um gráfico com a evolução do fitness da melhor solução encontrada ao longo do tempo (por geração)
- Estatísticas sobre o tempo que cada geração demora a ser processada
- ...

Por último, torne o seu código o mais configurável possível, de forma a que diferentes configurações possam ser facilmente testadas, de forma a perceber qual a melhor para cada problema.

Exercício 6

Considere o mesmo conjunto de pontos do Exercício 3. Admita agora que o grau do polinómio é desconhecido, sabendo-se apenas que os dados foram gerados por um polinómio cujo grau poderá estar entre 2 e 6. Utilize um algoritmo genético para tentar adivinhar qual a função que deu origem a estes pontos.