

Inteligência Artificial

BCC35G

Diego Bertolini

diegobertolini@utfpr.edu.br

<http://www.inf.ufpr.br/diegob/>

Aula 006

- **Aula Anterior:**
 - Busca Local;
- **Aula de Hoje:**
 - Algoritmo Genético ;

Objetivo

O que vocês devem saber ao final da aula:

Conceitos básicos de Algoritmos Genéticos;

Algoritmos Genéticos

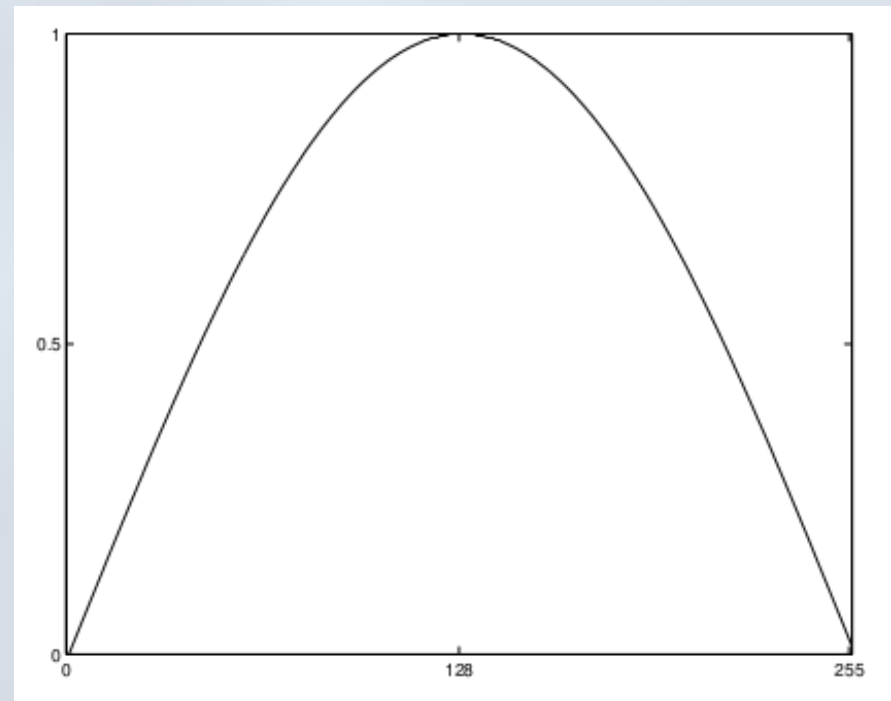
Algorithm 1 Algoritmo Genético Clássico

```
1:  $t \leftarrow 0$ 
2: Inicializar População( $t$ )
3: while condição de término não for satisfeita do
4:    $t \leftarrow t + 1$ 
5:   Seleciona População( $t$ ) da População( $t - 1$ )
6:   Cruzamento População( $t$ )
7:   Mutação População( $t$ )
8:   Avaliação da População( $t$ )
9: end while
```

Algoritmos Genéticos

Encontrar o valor de x que maximize a função:

$$f(x) = \text{sen} \left(\frac{\pi * x}{256} \right)$$



Algoritmos Genéticos

- **Esse problema contém uma única variável (x), a qual pode assumir valores entre 0 e 255.**
- **Utilizando uma codificação binária, a variável x pode ser codificada em uma string de 8 bits:**
- **00000000 -> 0**
- **11111111 -> 255**
- **Nesse exemplo usaremos uma população de 8**
- **indivíduos, inicializados aleatoriamente.**

Algoritmos Genéticos

- **Nesse exemplo usaremos uma população de 8 indivíduos, inicializados aleatoriamente.**

Indivíduos
10111101
11011000
01100011
11101100
10101110
01001010
00100011
00110101

Fitness da População Inicial

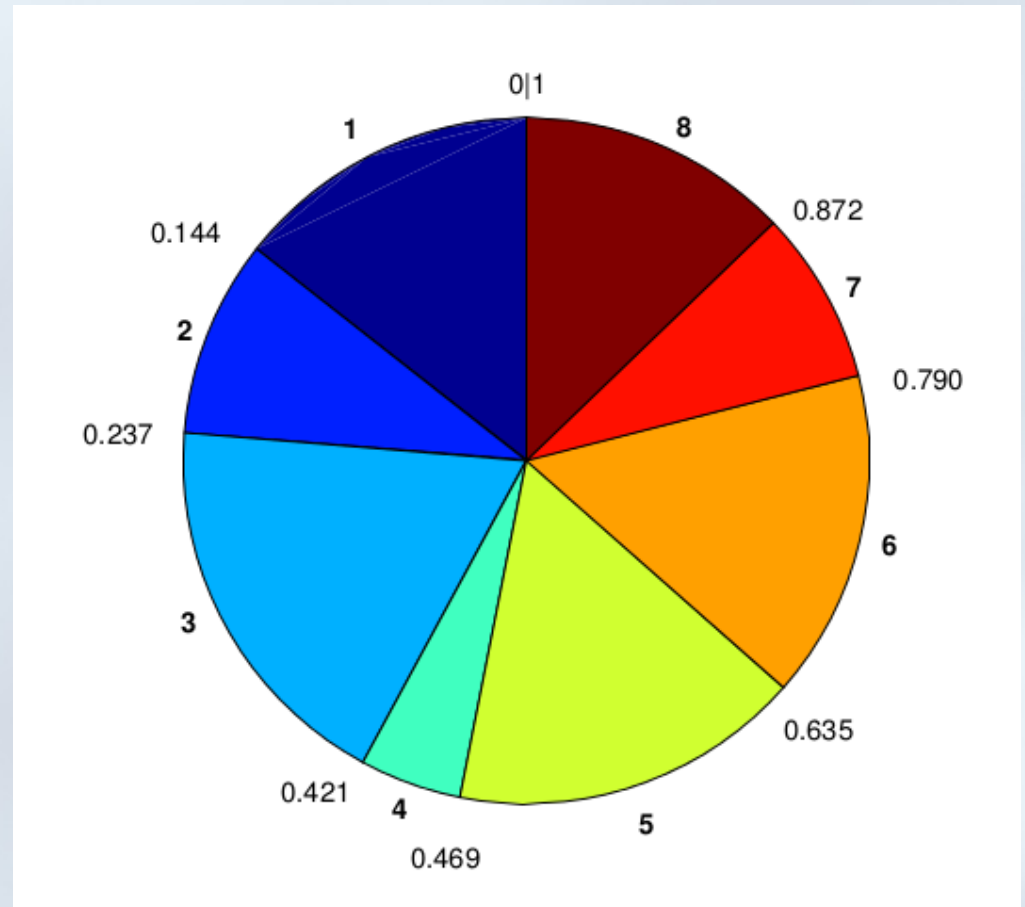
Indivíduos	x	$f(x)$	f_{norm}	f_{acm}
10111101	189	0.733	0.144	0.144
11011000	216	0.471	0.093	0.237
01100011	99	0.937	0.184	0.421
11101100	236	0.243	0.048	0.469
10101110	174	0.845	0.166	0.635
01001010	75	0.788	0.155	0.790
00100011	35	0.416	0.082	0.872
00110101	53	0.650	0.128	1.000

Reprodução

- **Após o cálculo da fitness acontece a reprodução.**
- **Gerar uma nova população com o mesmo número de indivíduos.**
- **Processo estocástico que leva em consideração a fitness normalizada.**
- **Indivíduos ruins também tem chance de reproduzir (probabilidade baixa).**

Roleta

- As porções maiores tem mais chances de serem selecionadas, porem as menores também tem chances (reduzidas, é claro).

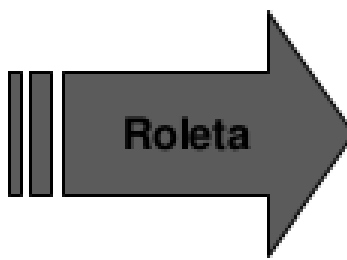


Reprodução

- **Rodamos a roleta oito vezes.**
 - Se o número cair entre 0 e 0.144, o indivíduo selecionado é o 1, e assim por diante.
- **Número gerados aleatoriamente:**
**0.293, 3, 0.971, 0.160, 0.169, 0.664, 0.568, 0.371,
0.109**
- **Indivíduos selecionados:**
3, 8, 2, 5, 6, 5, 3, 1

População Reproduzida

- Neste caso os piores individuos (4 e 7) não fazem parte da nova populacao;

População Inicial		Indivíduos Reproduzidos
Indivíduos		Índice Indivíduos
10111101		3 01100011
11011000		8 00110101
01100011		2 11011000
11101100		5 10101110
10101110		6 01001010
01001010		5 10101110
00100011		3 01100011
00110101		1 10111101

Cruzamento

- **Sendo assim, 3 pares de indivíduos serão selecionados aleatoriamente e o restante será simplesmente copiado para a nova população.**
- **Por uma questão de simplicidade, selecionamos os 6 primeiros indivíduos para o cruzamento.**

Cruzamento

	Indivíduos	X
<div> <div>1</div> <div>2</div> <div> 0 1 1 0 0 0 0 0 1 1 0 1 </div> </div>	<div> <div>1 0 1</div> <div>0 0 0</div> <div>1 1</div> <div>1 1</div> </div>	<div>119</div> <div>33</div>
<div> <div>1</div> <div>2</div> <div> 1 1 0 1 1 0 1 0 </div> </div>	<div> <div>0 1 0</div> <div>1 0 1</div> <div>1 0 0 0</div> <div>1 1 1 0</div> </div>	<div>168</div> <div>222</div>
<div> <div>1</div> <div>2</div> <div> 0 1 0 0 1 0 1 1 0 1 0 1 1 1 </div> </div>	<div> <div>1 0 1 1 1</div> <div>1 0 1 0 1</div> <div>0</div> <div>0</div> </div>	<div>138</div> <div>110</div>
Cópia-->	<div> <div>0 1 1 0 0 0 1 1</div> <div>1 0 1 1 1 1 0 1</div> </div>	<div> <div>0 1 1 0 0 0 1 1</div> <div>1 0 1 1 1 1 0 1</div> <div>99</div> <div>189</div> </div>

Mutação

- **Evitar a convergência prematura do algoritmo.**
- **Taxas de mutação entre 0.1 e 1% são geralmente utilizadas.**
- **Altas taxas de mutação faz com que o AG explore diferentes áreas do espaço.**
- **Geralmente inicia-se com taxas de mutação mais elevadas e diminui-se conforme o algoritmo converge..**

Mutação

		Indivíduos após o cruzamento	Fitness
Bit Selecionado Aleatoriamente $P_m = 1/64$	→	0 1 1 1 0 1 1 1	119
		0 0 1 0 0 0 1 1	33
		1 0 1 0 1 0 0 0	168
		1 1 0 1 1 1 1 0	222
		0 1 1 0 1 1 1 0	138
		1 0 1 0 1 0 1 0	110
		0 1 1 0 0 0 1 1	227
		1 1 1 0 0 0 1 1	227
		1 0 1 1 1 1 0 1	189

Nova População

- **A nova população (que dá início a segunda geração do algoritmo) deve ser do tamanho da população inicial, ou seja, 8 indivíduos.**

População Inicial	Fitness	População Intermediária	Fitness
0 1 1 0 0 0 1 1	189	0 1 1 1 0 1 1 1	119
0 0 1 1 0 1 1 1	216	0 0 1 0 0 0 1 1	33
1 1 0 1 1 0 0 0	99	1 0 1 0 1 0 0 0	168
1 0 1 0 1 1 1 0	236	1 1 0 1 1 1 1 0	222
0 1 0 0 1 0 1 0	174	0 1 1 0 1 1 1 0	138
1 0 1 0 1 1 1 0	75	1 0 1 0 1 0 1 0	110
0 1 1 0 0 0 1 1	35	1 1 1 0 0 0 1 1	227
1 0 1 1 1 1 0 1	53	1 0 1 1 1 1 0 1	189

Mutação

- **Estratégias mais comuns para selecionar a nova população:**
 - Roleta: Processo estocástico, onde o melhor indivíduo pode ser perdido
 - Ranking : Garante o melhor indivíduo na próxima população ; Estratégia Elitista

Considerações

- Representação de Variáveis
- População: Tamanho e Inicialização
- Operador de Cruzamento
- Operador de Mutação
- Seleção