

# Algoritmo Genético

SIAI - 2019/2

---

Bruna Zamith (628093)

Matheus Vrech (727349)

# Agenda

---

1. Algoritmo Genético
2. Motivação
3. Aplicação
4. Integração
5. Desenvolvido
6. Pseudo-código
7. Próximos passos

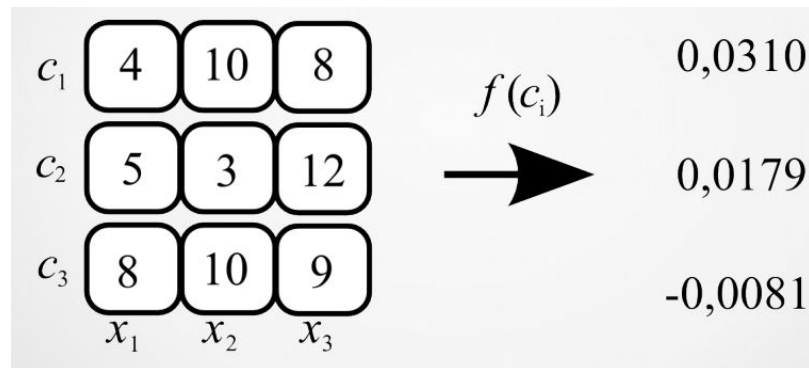
# Algoritmo Genético

---

- Heurística de busca
- Algoritmo de otimização
- Teoria da Seleção Natural
- Conceitos
  - Cromossomo
  - Função de Aptidão
  - População
  - Seleção
  - Cruzamento
  - Mutação

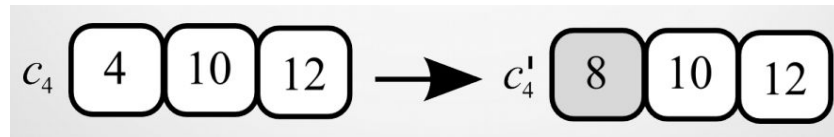
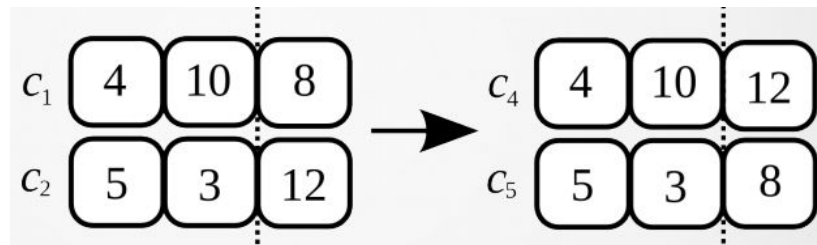
# Algoritmo Genético

- Cromossomo
  - Possível solução do problema. Genótipo.
- Função de aptidão
  - Avalia o resultado que um cromossomo consegue obter. Fenótipo.
- População
  - Conjunto de individuos, cada um representa uma solução



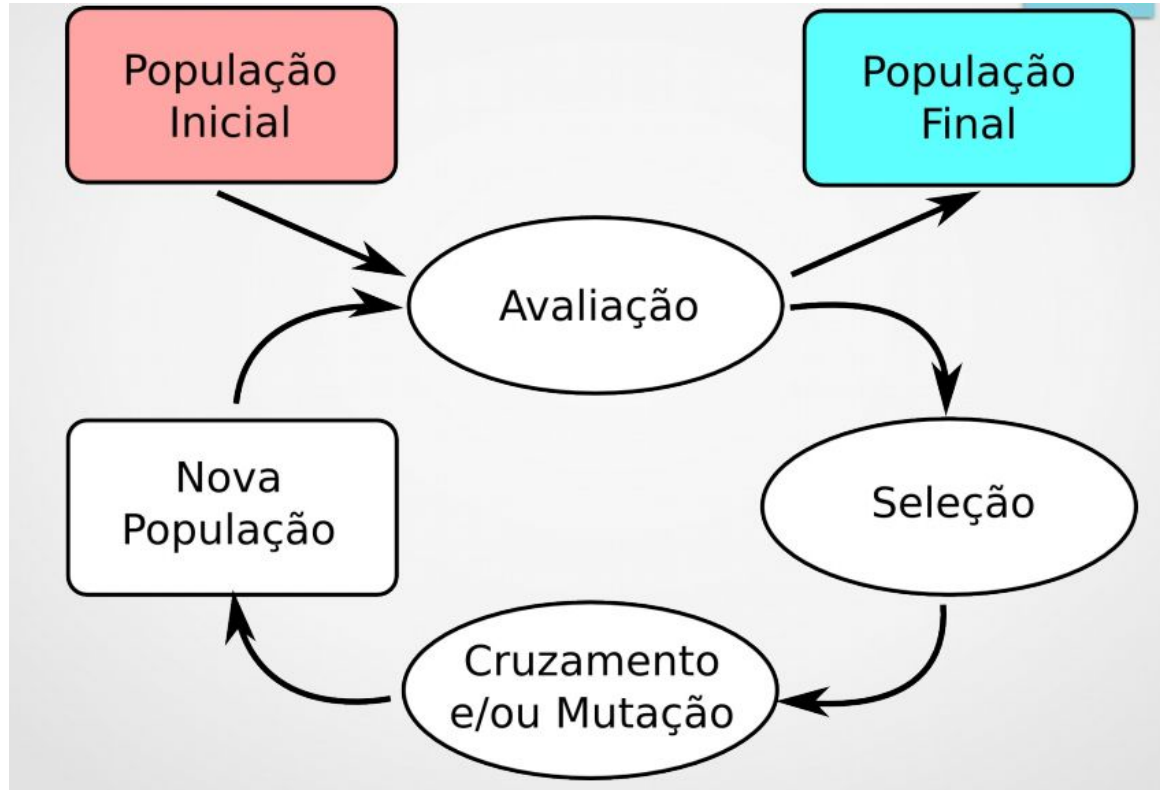
# Algoritmo Genético

- Seleção
  - Baseados na aptidão. Os indivíduos mais aptos devem possuir maior probabilidade de seleção.
- Cruzamento
  - Os indivíduos selecionados são cruzados dois a dois
- Mutação
  - Modifica parte do cromossomo (aleatoriamente ou por heurística)
  - Gera soluções inéditas



# Algoritmo Genético

---



# Motivação

---

- Metaheurística de busca
- Sintonização do  $K_p$ ,  $K_i$  e  $K_d$ .
- Problema combinatório: Otimização da seleção.
- AG x Ziegler-Nichols
- Variáveis são correlacionadas.
- Possibilidade: Usar na definição dos parâmetros de Fuzzy

# Aplicação

---

- Cromossomo: Valores de  $K_p$ ,  $K_i$  e  $K_d$
- Função de aptidão: Taxa de erro ou tempo de resposta
- População inicial: Randômica, dependente de um intervalo
- Seleção: Roleta viciada
- Operadores Genéticos: Cruzamento e Mutação
- Elitismo



# Integração

---

- Grupo do PID Adaptativo
  - Fornecer intervalo
- Grupo de Simulação
  - Taxa de erro para cada cromossomo



# Desenvolvido

---

- Simulações (valores fictícios)
- Python
- Funções
  - Gerar a população inicial
  - Seleção
  - Cruzamento
- Github: <https://bit.ly/2kTEvVZ>

# Pseudo-código

---

```
1  ALGORITMO popGenetica
2
3  INÍCIO
4  VARIAVEIS
5  Geracao, POM_TAM, POP_SEL, MAX_GERACAO: Real
6  MaxCkp, MinCkp, MaxCki, MinCki, MaxCkd, MinCkd: Real
7  Cromossomo, nova_pop: Vetor[50, 6]
8  pop_gerada: Vetor[50, 6]
9
10 POM_TAM <- 12
11 POP_SEL <- POM_TAM / 2
12 MAX_GERACAO <- 5
13
14 Cromossomo <- PopulacaoInicial()
15 Geracao <- 1
16
```

```
17 ENQUANTO Geracao <= MAX_GERACAO FAÇA
18     pop_gerada <- Selecao(Cromossomo)
19     nova_pop <- pop_gerada + Cruzamento(pop_gerada)
20     Geracao <- Geracao + 1
21     Cromossomo <- nova_pop
22 FIMENQUANTO
23 FIM
24
25 FUNÇÃO PopulacaoInicial(): Vetor[50, 6]
26 VAR
27     Cromossomo: Vetor[50, 6]
28     I: Inteiro
29 INÍCIO
30     PARA I DE 1 ATÉ POP_TAM FAÇA
31         Cromossomo[I][0] <- rand(MinCkp, MaxCkp)
32         Cromossomo[I][1] <- rand(MinCki, MaxCki)
33         Cromossomo[I][2] <- rand(MinCkd, MaxCkd)
34     FIM PARA
35     RETORNE Cromossomo
36 FIMFUNCAO
37
```

# Pseudo-código

---

```
38 FUNCAO Selecao(Cromossomo: Vetor[50, 6]): Vetor[25, 6]
39 VAR
40     I: Inteiro
41     Selecionado: Vetor[25, 6]
42 INICIO
43     PARA I DE 1 ATÉ TAM_POP FAÇA
44         Cromossomo[I][3] <- TaxaDeErro
45     FIM PARA
46     Selecionado <- ordena(Cromossomo, 3)
47     RETORNA Selecionado
48 FIMPROCEDIMENTO
49
```

```
50 FUNCAO Cruzamento(Cromossomo: Vetor[25, 6])
51 VAR
52     I: Inteiro
53     Filhos: Vetor[50, 6]
54 INICIO
55     Filhos <- Cromossomos
56     PARA I DE 1 ATÉ 49
57         Filhos[I][0] <- Cromossomo[I + 1][0]
58     FIMPARA
59 FIMFUNCAO
```

# Próximos passos

---

- Passar código para C++
- Integração
- Funções
  - Mutação
  - Determinar momento de parada
  - Retornar melhor indivíduo
- Testes

# Dúvidas?

Material de Apoio: Slides do professor Murilo Naldi, disciplina Inteligência Artificial.