

1. Reconhecimento de Padrões

- **Objetivo:** Implemente um algoritmo genético para o exemplo de reconhecimento de padrões apresentado em aula.
- **Descrição:** Em vez de reconhecer o número 1, o algoritmo deve reconhecer o número 0, representado pela bitstring: [1 1 1 1 0 1 1 0 1 1 1 1].
- **Tarefas:**
 - Verifique, em média, quantas gerações são necessárias para atingir o alvo.
 - Teste diferentes taxas de crossover e mutação, comparando os resultados.
 - Realize experimentos utilizando apenas crossover e apenas mutação, e compare os resultados obtidos.

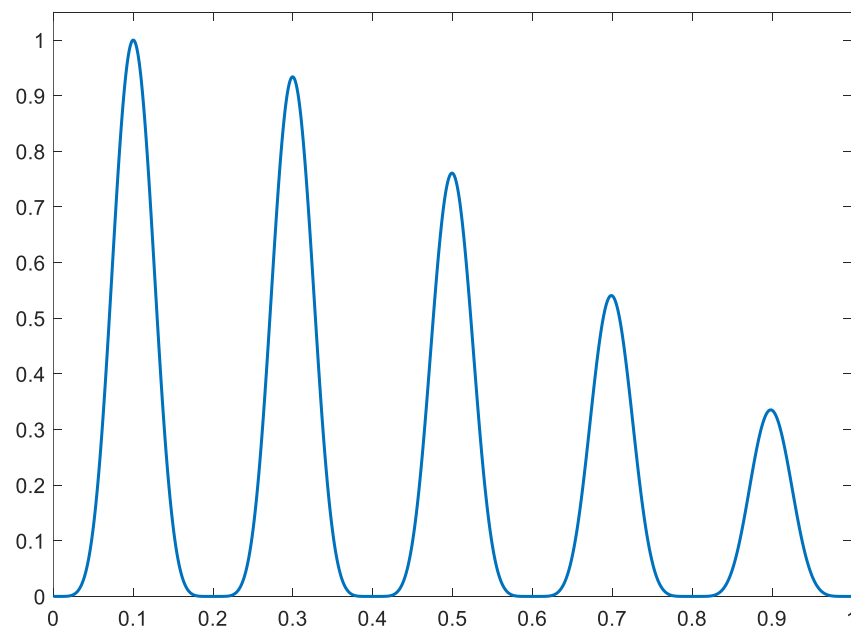
2. Maximização de Função

- **Objetivo:** Implemente um algoritmo genético para maximizar a função utilizada em aula, definida por:

$$g(x) = 2^{-2((x-0,1)/0,9)^2} (\sin(5\pi x))^6$$

Intervalo: $x \in [0, 1]$

- **Requisitos:**
 - Utilize uma representação por bitstring com precisão suficiente para pelo menos três casas decimais.
 - Considere o alvo como desconhecido e registre os maiores valores obtidos para $g(x)$.
 - Compare os resultados com os obtidos anteriormente utilizando os algoritmos de Subida da Colina e Recozimento Simulado, levando em conta tanto os valores de $g(x)$ quanto os tempos de execução.
 - Explore diferentes esquemas de seleção (por exemplo, roleta, torneio e amostragem universal estocástica).



- **Dica:** Você também pode aplicar Subida da Colina e Recozimento Simulado utilizando uma representação em bitstring, adotando uma perturbação semelhante ao operador de mutação dos algoritmos genéticos. Dessa forma, a própria representação binária cuidará do mapeamento do domínio de x .

3. Minimização de Função (Rosenbrock)

- **Objetivo:** Implemente um algoritmo genético para minimizar a seguinte função (geralmente conhecida como função de Rosenbrock):

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2$$

Intervalo: $x, y \in [-10, +10]$

- **Requisitos:**
 - Considere o alvo como desconhecido e registre os menores valores obtidos para $f(x, y)$.
 - Teste diferentes tamanhos de população, experimentando versões sem elitismo e com elitismo (com diferentes quantidades de indivíduos).

4. Relatório e Documentação

- **Conteúdo do Relatório:** Para TODOS os exercícios, elabore um relatório contendo:
 - Dados e configurações de todos os experimentos realizados.
 - Resultados obtidos, incluindo médias e desvios padrão (devido à natureza estocástica dos algoritmos).
 - Análise das diferentes versões do algoritmo (representação, seleção, etc.) e indicação de qual se adequa melhor a cada problema.
 - Número de iterações necessárias e tempo de execução para a convergência dos algoritmos.
 - Um ou mais gráficos por experimento, demonstrando a evolução do valor mínimo e médio da função de aptidão ao longo das iterações.
- **Observações Gerais:**
 - Otimize os hiperparâmetros para obter os melhores resultados e repita os experimentos sob condições iguais para garantir a confiabilidade dos dados.
 - Além de encontrar um indivíduo com boa aptidão, procure alcançar uma convergência rápida.

5. Diretrizes para Entrega

- **Código-fonte:**
 - Pode ser implementado em qualquer linguagem de programação e ambiente de desenvolvimento.
 - O código deve ser devidamente comentado.
 - Os códigos-fonte devem ser postados em arquivo(s) separado(s) no Google Classroom; eles poderão ser compactados em .zip ou formato similar.
- **Relatório:**
 - O relatório deve ser entregue em formato PDF e submetido separadamente dos códigos, via upload no Google Classroom.
 - Não compacte nem anexe códigos no arquivo PDF.