

UNIVERSIDADE FEDERAL DE MINAS GERAIS

**ESCOLA DE ENGENHARIA**

**ENGENHARIA ELÉTRICA**

**COMPUTAÇÃO EVOLUCIONÁRIA**

**Relatório 3 – Evolução Diferencial**

*Alunos*:

Gabriel Saraiva Espeschit - 2015065541

04 de novembro de 2019

1. **Introdução**

Algoritmos de evolução diferencial são métodos de otimização que utilizam de representações e operações vetoriais para encontrar mínimos e máximos das funções objetivos. Esse tipo de algoritmo segue o pseudocódigo a seguir:

**:**

**:**

Sendo assim, foram selecionadas duas funções, *rastrigin* e *peaks*, para serem submetidas a um algoritmo de evolução diferencial.

1. **Metodologia**

A metodologia de implementação consistiu na tradução das funções repassadas pelo professor para linguagem Python, e da implementação, em Python, do pseudocódigo acima apresentado.

1. **Resultados**

Para as cada uma das funções testadas, o resultado ótimo esperado era:

* *Peaks*: mínimo global em x\* = [0.228, -1.625] com f(x\*) = -6.5511;
* *Rastrigin*: mínimo global em x\* = [0, 0] com f(x\*) = -20.

Abaixo algumas imagens que mostram as gerações 0, 50 e 100 do algoritmo convergindo na função *peaks*:

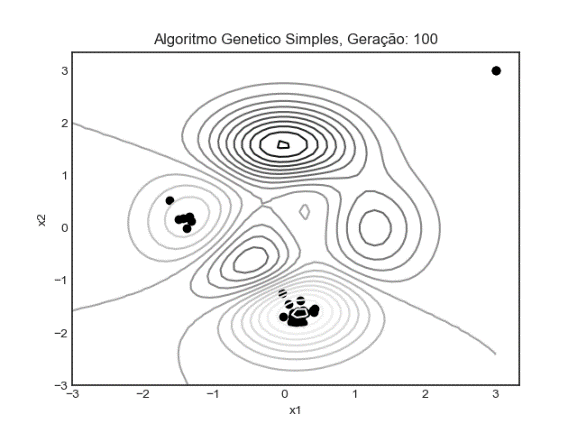
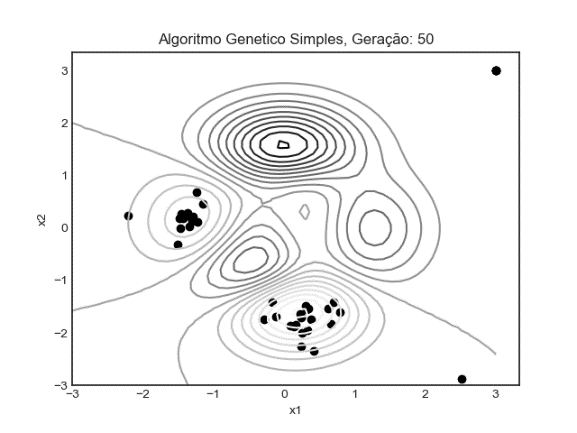
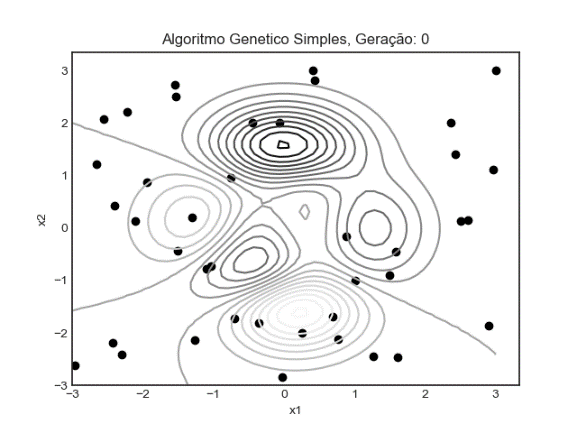


Figura 1 - Gráfico mostrando a convergência do algoritmo na função peaks.

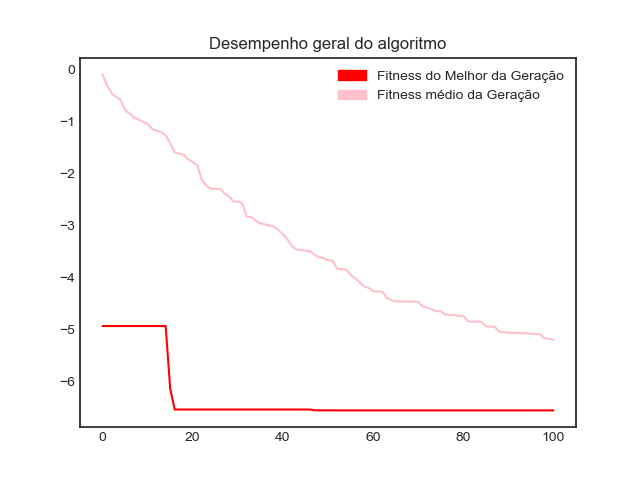


Figura 2 - Desempenho do algoritmo. A convergência no caso média mostra que o comportamento do algoritmo foi conforme esperado.

Abaixo algumas imagens que mostram as gerações 0, 50 e 100 do algoritmo convergindo na função *rastrigin*:

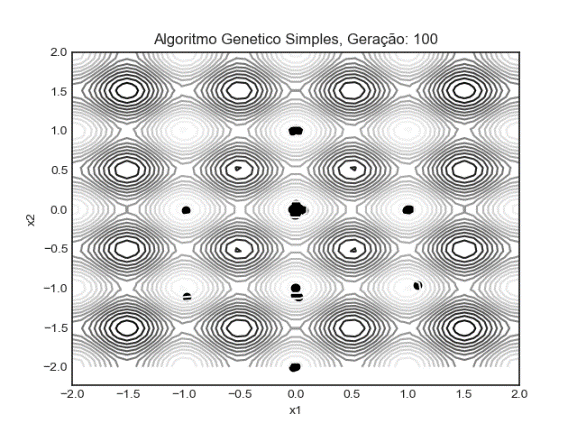
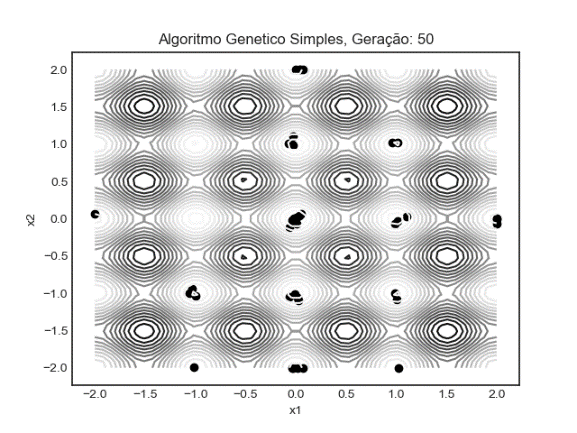
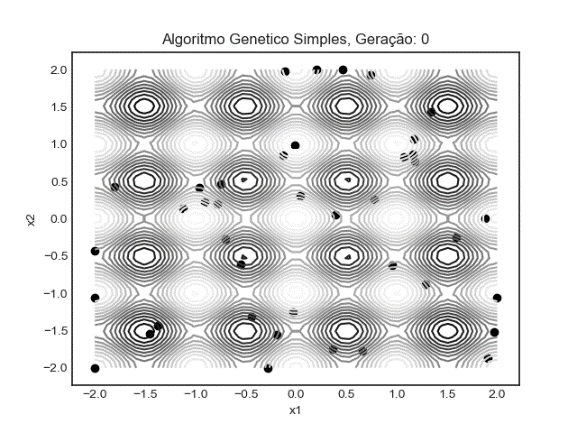


Figura 3 - Gráfico mostrando a convergência do algoritmo na função rastrigin.

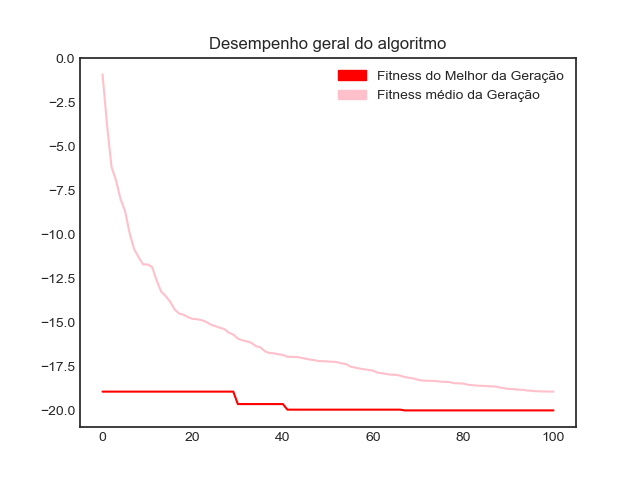


Figura 4 - Desempenho do algoritmo. A convergência no caso média mostra que o comportamento do algoritmo foi conforme esperado.

1. **Código**

O código desenvolvido em Python foi anexado na pasta zip enviada.