

Relatório Técnico sobre a implementação do Algoritmo Genético ao framework ML.js

Edgar Vieira Lima Neto

Departamento de Computação
Universidade Federal de Sergipe (UFS) – São Cristóvão, SE – Brasil
`edgarvln@dcomp.ufs.br`

1. Introdução

Propõe-se neste trabalho a implementação do Algoritmo Genético para enriquecimento do framework ML.js. O framework foi desenvolvido em JavaScript e tem capacidade de auxiliar no desenvolvimento de aplicações simples que utilizem aprendizado de máquina onde sejam aplicadas as atividades de classificação, agrupamento ou regressão. Com a implementação deste trabalho com algoritmo genético será acrescentada a atividade de busca e otimização, disponibilizando uma funcionalidade ainda não atendida pelo framework.

Os algoritmos genéticos são métodos que utilizam conceitos do princípio de seleção natural e genética para otimização de uma variedade de problemas. Para entender melhor o processo de um algoritmo genético, Holland [HOL 75] decompôs de maneira similar ao processo de evolução das espécies (ver figura 1). De forma resumida pode-se entender o funcionamento do algoritmo genético com a geração de uma população inicial com n indivíduos que contêm informações para gerar uma possível solução. Enquanto não seja encontrado um indivíduo que contenha uma solução satisfatória ou que os limites definidos não sejam alcançados, os indivíduos com maiores fitness (proximidade com a solução) passam por cruzamentos até alcançar uma população de filhos com o mesmo tamanho da população de pais. Após criar essa nova população, os pais são descartados e os indivíduos dessa nova população podem passar por um processo de mutação, de acordo com a probabilidade pré-definida. Os indivíduos que sofreram mutação passam por uma avaliação onde é verificado o fitness e, caso ainda não seja uma solução satisfatória, todo o processo é reiniciado.

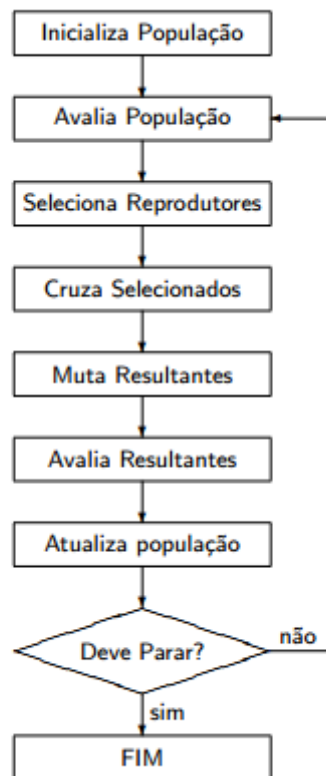


Figura 1 - Estrutura de funcionamento de um AG tradicional.

Para o desenvolvimento do algoritmo foram construídas funções que o deixassem mais genérico, e não apenas visando na resolução de um problema específico. A Seção 2 mostra detalhes da implementação do algoritmo ao ML.js e a Seção 3 apresenta as conclusões e a aplicação de um problema.

2. Implementação e aplicação do Algoritmo Genético no framework ML.js

O desenvolvimento do algoritmo se baseou no padrão adotado pelo framework ML.js, ou seja, a nova funcionalidade foi construída em semelhante as funcionalidades já existentes.

Em relação a parte de código, foi adicionado um novo modelo, no arquivo *gui.index.js*, chamado de “genetico” que representa o Algoritmo Genético no framework, semelhante aos modelos “kmeans” ou “id3”. Após isso criou-se os arquivos *ml.genetico.js*, que contém as funções relacionadas ao framework, e *ga.js*, que contém toda a estrutura de funções referentes ao Algoritmo Genético.

O funcionamento do algoritmo ocorre com o fornecimento da população inicial juntamente com os seguintes parâmetros:

- a) **tipo:** representa o tipo de otimização a ser realizada (definindo o número máximo de gerações, o número mínimo do fitness ou o tempo máximo de otimização);
- b) **valorTipo:** representa o valor a ser atribuído ao parâmetro anterior;

- c) **probMutacao:** representa a probabilidade de ocorrer mutação nos indivíduos;
- d) **probCross:** representa a probabilidade de ocorrer cruzamento entre os indivíduos; e,
- e) **geneBusca:** representa o gene a ser otimizado.

Após essas informações, o algoritmo começa definindo alguns atributos a serem utilizados como o tamanho da população inicial, o tamanho do indivíduo e os genes utilizados. Com esses valores definidos, é chamada a função de simulação referente ao tipo de otimização escolhida. Dentro dessa função são chamadas todas as funções necessárias para seguir o processo descrito na Seção 1. Por fim é retornado o melhor indivíduo encontrado e o seu fitness.

Para testar o algoritmo foi simulado o problema de otimização de genes em uma população. Foi inserido o arquivo *populacaoInicial.arff* que contém os 5 indivíduos iniciais do problema. Em seguida foram definidos os seguintes valores para os parâmetros: tipo = Por máximo de gerações, valorTipo = 100, probMutacao = 45, probCross = 45 e geneBusca = 1.

Busca e Otimização

Selecione o modelo

Algoritmo Genético

Tipo de Otimização

Por máximo de gera

Valor atribuído ao tipo

100

Porcentagem de mutação

45

Porcentagem de
crossover

45

Gene para otimizar

1

Executar

Figura 2 - Aplicando o algoritmo.

Ao executar o teste foi recebido o resultado mostrado na figura 3. Nessa situação o melhor indivíduo, em 100 gerações e com porcentagens de mutação e cruzamento de 45%, foi o 1111101111 que contém o fitness 9.

Indivíduo Máximo: 1 1 1 1 1 0 1 1 1 1

Fitness: 9

Figura 3 - Resultado obtido.

3. Conclusão

A implementação do Algoritmo Genético ao framework ML.js atribuiu um novo escopo de opções para o desenvolvedor. Até então não havia nenhuma funcionalidade de busca no sistema e, com essa nova funcionalidade, será possível a aplicação de diversos problemas.

A intenção dessa primeira implementação foi deixar o algoritmo genético mais genérico, o que foi dificultado devido à complexidade do problema. De forma geral, é possível testar diversos tipos de aplicações com o algoritmo genético implementado, tomando cuidado apenas com a função do fitness que essa aplicação utiliza.

Desse modo, concluo que a implementação desse algoritmo complementou o framework e criou uma eficiente ferramenta de otimização para diversos problemas, desde que seja devidamente observada a sua função fitness.

Referências

- E. P. Bento; N. Kagan. “Algoritmos genéticos e variantes na solução de problemas de configuração de redes de distribuição”. Disponível: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-17592008000300006, Abril/2017.
- Poole, D. L.; Mackworth, A. Artificial Intelligence: Foundations of Computational Agents, 2ª edição, 2010.
- Alcaraz, R. C. “geneticAlgorithm”. Disponível: <https://github.com/rcalcaraz/Generic-Genetic-Algorithm>, Abril/2017.
- Lucas, D. C. “Algoritmos Genéticos: uma Introdução”. Disponível: <http://www.inf.ufrgs.br/~alvares/INF01048IA/ApostilaAlgoritmosGeneticos.pdf>, Abril/2017.