Universidade Federal de Minas Gerais
Disciplina de Computação Natural
Prof Gisele L. Pappa

Trabalho Prático 3

Pode ser feito em dupla.
Data de Entrega: 06 de Dezembro
Observação: Trabalhos entregues após essa data não serão aceitos!

Regressão Simbólica

O objetivo deste trabalho é desenvolver conceitos e metodologias chave para a construção de soluções para problemas usando Programação Genética (PG), envolvendo o entendimento e a implementação dos componentes básicos de um arcabouço de PG, bem como a análise de sensibilidade dos seus parâmetros (como eles afetam o resultado final, a natureza da convergência, etc) e procedimentos para avaliação das soluções alcançadas.

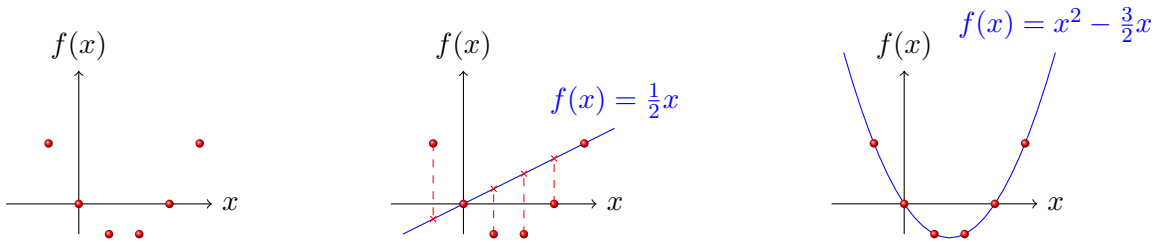
Esse trabalho propõe o uso de um arcabouço baseado em PG para resolver um problema de Regressão Simbólica [3]. Dado um conjunto de m amostras provenientes de uma função *desconhecida* $f : \mathbb{R}^n \mapsto \mathbb{R}$, representadas por uma dupla $\langle X, Y \rangle$ onde $X \in \mathbb{R}^{m \times n}$ e $Y \in \mathbb{R}^m$, o objetivo é encontrar a expressão simbólica de f que melhor se ajusta às amostras fornecidas.

Suponha que sejam fornecidas as seguintes $m = 6$ amostras de uma função hipotética $f : \mathbb{R} \mapsto \mathbb{R}$:

$$X = \begin{pmatrix} -0.5 \\ 0.0 \\ 0.5 \\ 1.0 \\ 1.5 \\ 2.0 \end{pmatrix} \qquad Y = \begin{pmatrix} 1.0 \\ 0.0 \\ -0.5 \\ -0.5 \\ 0.0 \\ 1.0 \end{pmatrix}$$

conforme ilustrados na Figura 1(a). O algoritmo de Programação Genética irá, a partir de um conjunto inicial de soluções (funções, representadas por árvores), buscar por uma solução que melhor se ajusta a esses pontos. Trata-se do problema de Regressão Simbólica: determinar a expressão simbólica da função que melhor se ajusta a um conjunto de pontos fornecido.

O que define a qualidade de uma solução é a qualidade do ajuste da mesma aos dados: quanto menor a discrepância entre o valor obtido pela função ajustada e os pontos fornecidos (erro do ajuste), melhor a solução. Por exemplo, uma das soluções iniciais poderia ser aquela ilustrada na Figura 1(b), e o algoritmo poderia convergir para a função ilustrada na Figura 1(c). O ajuste da função linear é pior do que o ajuste quadrático ilustrado, como pode ser observado pelos segmentos de reta verticais ilustrados (erros do ajuste).



(a) Entrada (b) Ajuste Ruim (c) Ajuste Ótimo

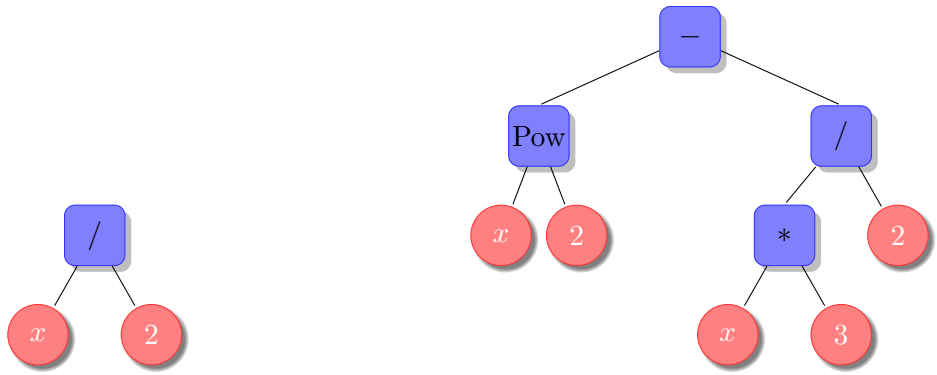
Figura 1: Exemplo: Regressão Simbólica

Como mencionado, no arcabouço de programação genética a ser desenvolvido, os indivíduos são representados por árvores, compostas por nós terminais e não-terminais, conforme exemplificado na Figura 2. Será de sua responsabilidade determinar ambos os conjuntos para solucionar o problema de regressão simbólica fornecido, lembrando que é importante considerar a presença de constantes (para a representação de coeficientes), bem como das variáveis do problema (e.g., variável x nos exemplos anteriores). Ainda, o arcabouço de programação genética deverá permitir o uso de elitismo.

Para esse trabalho, a função a ser obtida é do tipo $f : \mathbb{R}^2 \mapsto \mathbb{R}$, ou seja, trata-se de uma função de duas variáveis $f(x, y) = z$, em que $x, y, z \in \mathbb{R}$. Um critério de avaliação possível para avaliar a qualidade de um indivíduo é a soma do erro absoluto, dada por:

$$f(Ind) = \sum_{\{\langle x,y,z \rangle\}} \|\textsc{Eval}(Ind, x, y) - z\|,$$

onde *Ind* é o indivíduo sendo avaliado, $\textsc{Eval}(Ind, \text{x}, \text{y})$ avalia o indivíduo *Ind* com as variáveis x e y, $\{\langle x, y, z \rangle\}$ é o conjunto de entrada fornecido e, finalmente, z é a saída correta da função para ambas as entradas x e y.



(a) Indivíduo $\frac{1}{2}x$ (b) Indivíduo $x^2 - \frac{3x}{2}$

Figura 2: Representação dos indivíduos em árvore

O trabalho se divide em três etapas principais:

1. Implementação de um arcabouço de programação genética que seja facilmente instanciável para solucionar outros problemas;

- Deve permitir fácil configuração de parâmetros, definição de terminais/não-terminais, função de *fitness*, etc.
- Cuidado: o problema tratado nesse trabalho pode facilmente ser acometido pelo "*bloating*" [4]

2. Modelagem do problema de Regressão Simbólica, para ser solucionado pelo arcabouço implementado;

3. Avaliação experimental.

Para o primeiro item (implementação do arcabouço de programação genética), alguns aspectos chave devem ser cuidadosamente considerados:

1. Como representar um indivíduo (genótipo);

2. Como gerar a população inicial (e.g., *ramped half and half*)

3. Quais operadores genéticos serão utilizados;

4. Facilidades para variação de parâmetros—parâmetros *hardcoded* no arcabouço certamente dificultarão a avaliação dos parâmetros;

5. Como prover uma avaliação de fitness que seja facilmente instanciada para outros problemas (fenótipo)—por exemplo, deve ser fácil incluir funções para cálculo da fitness dos indivíduos de forma específica para um determinado problema;

Além desses aspectos, também devem ser consideradas as seguintes questões relativas à representação dos indivíduos:

1. Deve-se prever a utilização de constantes (para representar os coeficientes, se existirem)

2. Deve-se prever a utilização de variáveis (que serão usadas para avaliar o ajuste aos pontos fornecidos)—no caso desse trabalho, as amostras disponibilizadas referem-se a uma função de *duas* variáveis, mas o arcabouço deve ser facilmente extensível para problemas de regressão simbólica com funções mais complexas (e.g., mais de duas variáveis). Ou seja, deve ser fácil para o usuário do arcabouço definir novos terminais/não-terminais.

Por ser um método estocástico, a avaliação experimental do algoritmo baseado em PG deve ser realizada com *repetições*, de forma que os resultados possam ser reportados segundo o valor médio obtido e o respectivo desvio-padrão. A realização de 30 repetições pode ser um bom ponto de partida (lembrando que desvio-padrão alto sugere um maior número de repetições).

Mínimos de Funções

A segunda parte do trabalho prático consiste em encontrar o mínimo da função descoberta na primeira parte (ou seja, a melhor solução obtida na etapa anterior, referente ao uso do arcabouço de programação genética). Para tanto, será utilizado um arcabouço baseado em Otimização por Enxames de Partículas (Particle Swarm Optimization—PSO). PSO nada mais é do que uma estratégia de otimização estocástica baseada em população (como a estratégia de otimização baseada em colônia de formigas). Em PSO, um conjunto de partículas é utilizado para varrer o espaço de busca, objetivando encontrar a solução para o problema sendo resolvido. No nosso caso, o problema será determinar o mínimo da função $f : \mathbb{R}^2 \mapsto \mathbb{R}$ encontrada na primeira parte do trabalho [1].

Mais especificamente, as partículas são inicialmente dispostas no espaço de busca de forma aleatória. O objetivo é que as partículas se movam em direção ao único ponto do espaço de busca referente ao mínimo da função (de fato, a função associada aos pontos disponibilizados na primeira parte do trabalho prático possui um mínimo global—embora existam diversos mínimos locais). Entretanto, nenhuma partícula conhece tal posição, e o comportamento coletivo do enxame permitirá descobrí-la. Esse comportamento coletivo é caracterizado pela estratégia de seguir as melhores partículas em direção a melhores regiões do espaço de busca.

Algoritmo 1 PSO—Pseudo-código

```
function PSO(expr)                                   ▷ expr é a expressão simbólica da função sendo avaliada.
    Inicialização:
    for each partícula p_i do
        Inicializa a posição x_i de p_i aleatoriamente no espaço de busca
        Inicializa a melhor posição conhecida pela partícula b_i ← x_i
        if Eval(expr, p_i) < Eval(expr, g) then      ▷ g é a melhor posição global (conhecida pelo enxame)
            g ← p_i
        end if
    end for
    Iterações do PSO:
    for each i = 1 ··· maxIter do
        for each partícula p_i do
            Atualize a velocidade v_i de p_i         ▷ Considerando a melhor posição conhecida pela partícula e pelo enxame
            Atualize a posição x_i de p_i            ▷ Idem acima
            if Eval(expr, x_i) < Eval(expr, p_i) then ▷ Atualize a melhor posição conhecida pela partícula
                p_i ← x_i
                if Eval(expr, p_i) < Eval(expr, g) then ▷ Atualiza a melhor posição global (conhecida pelo enxame)
                    g ← p_i
                end if
            end if
        end for
    end for
end function
```

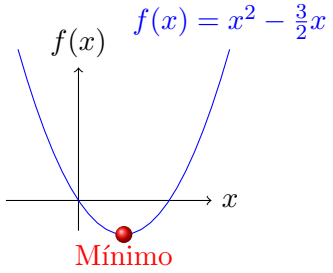


Figura 3: Mínimo global da função $x^2 - \frac{3x}{2}$

Faz parte da modelagem do algoritmo determinar como atualizar tanto a posição das

partículas quanto suas velocidades. Vale observar que a função explorada nesse trabalho prático possui diversos mínimos *locais*, sendo imprescindível evitar a convergência prematura.

Avaliação Experimental

Programação Genética

Será disponibilizado um conjunto de dados com amostras provenientes de uma função desconhecida do tipo $f : \mathbb{R}^2 \mapsto \mathbb{R}$, no seguinte formato:

```
      x          y          z

  19.814661  25.047982 20.535656
 -16.680174 -25.972818 21.140511
  31.655971 -15.236533 21.789619
  29.506773 -17.428474 22.174377
 -25.742460 -31.792588 21.542199
 -23.303729 -8.092452  20.820386
     ...        ...        ...
```

Figura 4: Exemplo de entrada para o problema de Regressão Simbólica.

Esses pontos formam a superfície apresentada na figura abaixo:

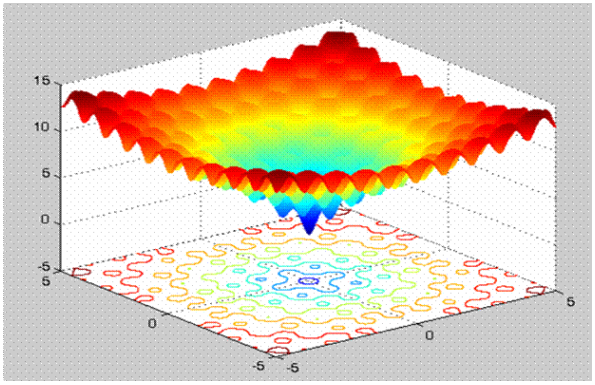


Figura 5: Superfície da função amostrada.

Quanto à avaliação experimental, mais especificamente, a análise de sensibilidade dos parâmetros, pede-se que, ao menos, os seguintes parâmetros sejam avaliados:

- Tamanho da população
- Número de gerações
- Pressão seletiva (número de indivíduos considerados durante a seleção)
- Análise de diversidade da população e ponto de convergência

- Probabilidades de aplicação dos operadores genéticos

Por se tratar de um processo estocástico, a avaliação experimental deverá ser conduzida com replicações, de forma a levar em conta a variabilidade dos resultados. A avaliação deverá, portanto, se basear em valores médios em conjunto com medidas de variabilidade (como o desvio-padrão).

PSO

A entrada para o problema de otimização via PSO será a melhor função encontrada na primeira parte desse trabalho. Uma vez determinada a expressão da função previamente desconhecida, basta utilizar o arcabouço de PSO para encontrar seu mínimo global. É de sua responsabilidade a condução de uma análise aprofundada da sensibilidade do arcabouço em relação aos parâmetros do mesmo [2] (e.g., fator de inércia ao atualizar a velocidade das partículas, o fator que controla a influência da melhor posição local conhecida pela partícula e o fator que controla a influência da melhor posição gloval conhecida pelo enxame), considerando que a convergência prematura é altamente prejudicial para esse problema (dado que as amostras são provenientes de uma função com diversos mínimos locais).

Guia para Execução dos Experimentos

Abaixo, serão discriminadas algumas sugestões para análise de sensibilidade dos parâmetros. Ela não é exaustiva e você deverá estudar as configurações de parâmetros mais apropriadas para cada problema.

- Escolha do número de partículas e de iterações do algoritmo. Essa escolha deverá considerar propriedades como convergência, melhoria das soluções encontradas, etc., para ambos os problemas;
- Configurações de parâmetros referentes às probabilidades de cada operador genético, para a primeira parte do trabalho;
- Configurações de parâmetros referentes às estratégias de atualização da velocidade/posição das partículas, para a segunda parte do trabalho;

Lembre-se que, ao variar um determinado parâmetro, todos os outros devem ser fixados, e que a análise de sensibilidade dos parâmetros é uma tarefa interativa: você deve estudar bem os efeitos até encontrar configurações boas. Pequenos testes preliminares podem ser de grande valia para determinar direções mais interessantes para determinar melhores configurações. Obviamente, um estudo cuidadoso desses aspectos é imprescindível para a obtenção de soluções de qualidade.

Estatísticas Importantes

- Melhores e piores soluções médias obtidas (bem como o desvio padrão);
- Solução média obtida (bem como o desvio padrão);
- Qualidade média da solução encontrada em cada iteração do algoritmo, etc.

O que deve ser entregues

- Código fonte do programa
- Documentação do trabalho, composta por:
  - Introdução;
  - Implementação: descrição sobre a implementação dos arcabouços de Programação Genética e PSO, incluindo detalhes referentes à representação dos dados, parâmetros do algoritmo, dentre outros detalhes de implementação. Incluir aspectos relevantes de modelagem específicos para cada problema;
  - Experimentos: Análise de sensibilidade dos parâmetros dos arcabouços desenvolvidos em relação à qualidade das soluções obtidas, levantamento de estatísticas acerca do processo evolucionário, etc.
  - Conclusões;
  - Bibliografia.

A entrega **DEVE** ser feita pelo *Moodle* na forma de um único arquivo zipado (ou tarball), contendo o código e a documentação do trabalho.

Considerações Finais

Os parâmetros listados para execução dos experimentos são sugestões iniciais, e podem ser modificados a sua conveniência. Além disso, estratégias mais sofisticadas podem também ser incorporadas ao algoritmo.

Referências

[1] K. E. Parsopoulos, M. N. Vrahatis. Recent approaches to global optimization problems through Particle Swarm Optimization. Natural Computing. 1(2-3):235-306, 2002.
URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.19.4058&rep=rep1&type=pdf

[2] Yuhui Shi and Russell C. Eberhart. Parameter Selection in Particle Swarm Optimization. Proceedings of the 7th International Conference on Evolutionary Programming, pp.591-600, 1998.
URL: http://www.engr.iupui.edu/ shi/PSO/Paper/EP98/psof6/ep98_pso.html

[3] Tobias Blickle . Evolving Compact Solutions in Genetic Programming: A Case Study Proceedings of the 4th International Conference on Parallel Problem Solving, pp.564-573, 1996.
URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.29.8703&rep=rep1&type=pdf

[4] William B. Langdon. Quadratic Bloat in Genetic Programming. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000), pp.451-458, 2000.
URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.7993&rep=rep1&type=pdf