Model Breeder – Um Algoritmo Genético para Criação de Modelos

Adair Santa Catarina
Colegiado de Informática
UNIOESTE - PR
asc@dpi.inpe.br

João Ricardo de F. Oliveira
DPI - INPE
joao@dpi.inpe.br

Antônio M. V. Monteiro
DPI - INPE
miguel@dpi.inpe.br

Resumo

Este artigo apresenta um Algoritmo Genético (AG) capaz de gerar modelos matemáticos para um conjunto de dados de entrada. O AG é capaz de encontrar um modelo matemático que relaciona a variável dependente a um conjunto de variáveis independentes. Dois conjuntos de dados foram testados: um conjunto simples com apenas uma variável independente e um conjunto mais complexo com três variáveis independentes. Os resultados obtidos foram então comparados com os resultados fornecidos pelo método dos mínimos quadrados. Para o primeiro conjunto de dados o AG ajustou modelos menos precisos que o método dos mínimos quadrados. Para o segundo conjunto de dados o AG ajustou modelos até mais precisos que os modelos obtidos com o método dos mínimos quadrados. Estes resultados apontaram a ferramenta implementada como adequada para se realizar uma análise exploratória de dados de forma semi-automática.

***Palavras-chave**: algoritmos genéticos, modelos matemáticos, análise de dados, análise exploratória*

1 Introdução

Dados e informações são matérias-primas para a indústria da Tecnologia da Informação (TI). A sobrevivência e a prosperidade de muitos setores governamentais, públicos e comerciais dependerão do bom uso das fontes de informação que estão disponíveis.

Openshaw, 1997a, descreve um problema progressivamente sério causado pelo desenvolvimento da TI. Este problema afeta a todas as pessoas, ameaça governos, impacta nos lucros das companhias e, rapidamente, torna-se cada vez mais sério: é o grande volume de dados disponíveis.

A informatização de processos, a digitalização dos dados, o desenvolvimento de mecanismos automáticos de captura de dados, a redução do custo de armazenamento, etc., fazem com que mais e mais dados estejam disponíveis.

A problemática que se apresenta é como extrair informações em enormes repositórios de dados. Mecanismos manuais são inviáveis para analisar tamanho volume de dados. A alternativa que se apresenta é a criação e o desenvolvimento de mecanismos semi-automáticos, ou automáticos, para a análise de dados.

Uma proposta de uma ferramenta para modelagem automática é apresentada a seguir.

2 *Model breeder*

Openshaw, 1997b, propôs uma ferramenta, chamada de *model breeder* (construtor de modelos) para modelagem automática. Esta ferramenta seria capaz de encontrar um modelo matemático que relaciona a variável dependente (variável resposta) a um conjunto de variáveis independentes (variáveis preditoras), seguindo a expressão geral:

$$y = f(x_1, x_2, \cdots, x_n)$$

Estatisticamente o processo de criar um modelo que relacione variáveis em estudo é conhecido como análise de regressão. Uma técnica estatística automática utilizada neste ajuste de modelos é a regressão *stepwise*.

Esta técnica consiste em adicionar, ou remover, variáveis ao modelo com o objetivo de identificar o conjunto de variáveis preditoras que expliquem o comportamento da variável resposta (Neter *et al.*, 1996).

O *model breeder* proposto é uma alternativa à regressão *stepwise*, sendo que o mecanismo utilizado para automatizar o ajuste do modelo foi um Algoritmo Genético (AG).

Holland, 1975, formalizou os algoritmos genéticos (AGs). Eles são um tipo de algoritmo de busca que se utiliza do paradigma genético/evolucionário. Goldberg, 1989, afirma que os AGs foram criados com o intuito de imitar alguns dos processos observados na evolução natural das espécies.

A partir de um conjunto de soluções iniciais, obtidas aleatoriamente, este algoritmo é capaz de evoluir, por sucessivas gerações, até obter a solução ótima, ou aproximadamente ótima. Neste caso, encontrar um modelo matemático que explique o comportamento de uma variável dependente em função de um conjunto de variáveis independentes.

3 O *model breeder* implementado

Neste trabalho implementou-se um Model Breeder diferente daquele proposto por Openshaw, 1997b. Uma das principais diferenças diz respeito à forma de codificação utilizada nos cromossomos genéticos. Em sua proposta Openshaw utilizou-se da codificação clássica.

A codificação clássica e até hoje a mais usada, consiste em usar *strings* de bits, mas com o passar do tempo outros pesquisadores apresentaram outras formas de codificação.

A codificação clássica, quando utilizada em problemas que possuem variáveis contínuas e cujas soluções requeridas necessitam boa precisão numérica, torna os cromossomos longos. Para cada casa decimal acrescentada na precisão, é necessário adicionar 3,3 bits na string (Galvão & Valença, 1999).

A conseqüência imediata do aumento da string, que representa o cromossomo, é o aumento no tempo necessário para calcular o equivalente decimal deste cromossomo.

Por este motivo, formas não clássicas de codificação dos cromossomos foram desenvolvidas, gerando codificações adequadas para problemas específicos. (Herrera, Lozano & Verdegay, 1996)

Uma das formas não clássicas de codificação mais utilizada é a codificação real. Esta forma de codificação consiste em representar, num gene ou cromossomo, uma variável numérica contínua através de seu próprio valor real. Um cromossomo pode ser composto por múltiplos genes quando o problema a ser resolvido envolve duas ou mais variáveis.

As primeiras aplicações da codificação real foram propostas por Lucasius & Kateman (1989) e Davis (1989). A partir de então a codificação real tornou-se padrão em problemas de otimização numérica com variáveis contínuas.

Nesta implementação utilizou-se uma codificação híbrida envolvendo cadeias binárias e números reais em base decimal. Maiores detalhes da codificação empregada são apresentados na seção seguinte.

3.1. A codificação empregada

O modelo matemático adotado neste trabalho segue o polinômio geral:

$$y = c_1 \cdot x_i^{Exp_1}\ op_1\ c_2 \cdot x_j^{Exp_2} op_2 \cdots c_k \cdot x_k^{Exp_k}$$

onde:
- y: variável independente;
- c_i: coeficiente de cada termo do polinômio;
- x_{ind}: variáveis independentes;
- Exp_i: expoentes das variáveis independentes;
- op_i: operadores que relacionam os termos do polinômio (+, –, x, /).

Uma mesma variável independente pode figurar em vários termos do polinômio. Esta forma de representação polinomial foi empregada pois permite aproximar qualquer função matemática.

Num AG a evolução ocorre sobre os cromossomos. Estes codificam uma possível solução para o problema em estudo. A figura 1 mostra, através de um exemplo, a codificação utilizada. O cromossomo pode ser formado por um ou mais termos.

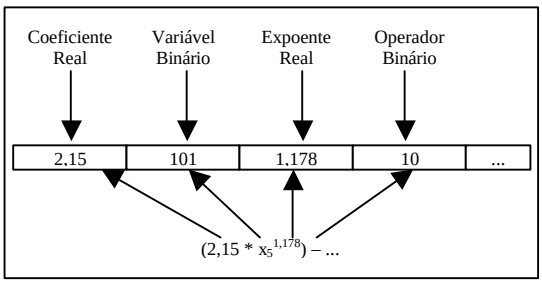

Figura 1. Exemplo da codificação empregada no cromossomo

Cada termo é composto por 4 genes. O primeiro gene corresponde ao coeficiente do termo; o segundo gene identifica a variável independente utilizada no termo; o terceiro gene corresponde ao expoente da variável independente; o quarto gene indica que operação aritmética será utilizada para operar o termo atual

com seu subseqüente. As operações válidas são as quatro operações aritméticas fundamentais.

3.2. A função de avaliação

As possíveis soluções (cromossomos) são avaliadas para se verificar quão boas são. Desta avaliação resulta um valor, chamado de grau de aptidão, ou simplesmente *fitness*. Quanto maior o *fitness* melhor a qualidade da solução encontrada. Esta informação é utilizada no mecanismo evolutivo dos AGs.

A função de avaliação utilizada baseou-se na soma dos quadrados dos desvios e pode ser calculada através da expressão:

$$Fitness_k = \frac{Min\left(SQT_1, SQT_2, \cdots, SQT_{Tp}\right)}{SQT_k}$$

onde:
- $Fitness_k$: grau de aptidão da k-ésima solução, com $k = 1..Tp$;
- Tp: tamanho da população avaliada;
- SQT: somatório dos quadrados dos desvios total:

$$SQT = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2$$

- Y_i: valor assumido pela variável dependente na amostra i;
- $\hat{Y}_i$: valor estimado para a variável dependente na amostra i;
- n: número total de amostras coletadas.

3.3. O mecanismo evolutivo

O processo de seleção utilizou-se do método de seleção universal, também conhecido como roleta. Este método confere àqueles indivíduos com maior *fitness*, maior probabilidade de serem escolhidos.

A ferramenta desenvolvida possibilita o uso do elitismo. Nesta técnica um número de indivíduos mais aptos, a elite, são automaticamente copiados para a nova geração, sendo assim preservados.

O método de cruzamento empregado foi o aritmético, para os genes com valores reais, e o cruzamento em um ponto, para os genes binários.

O cruzamento aritmético, proposto por Michalewicz, 1994, gera dois cromossomos filhos (c_1 e c_2) a partir de dois cromossomos pais (p_1 e p_2), usando a expressão:

$$c_1 = bp_1 + (1-b)p_2$$
$$c_2 = (1-b)p_1 + bp_2$$

onde b é um valor aleatório entre 0 e 1 gerado segundo uma distribuição aleatória uniforme.

O cruzamento em um ponto, efetuado sobre os genes com codificação binária, é exemplificado na figura 2.

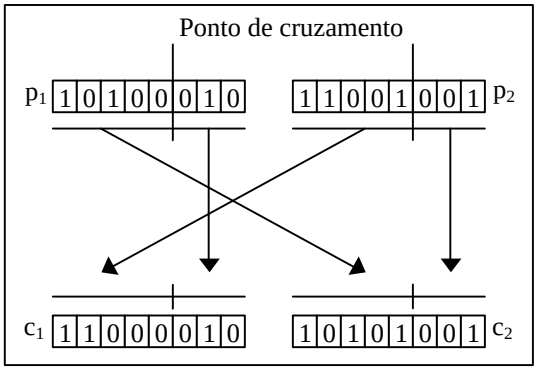

Figura 2. Exemplo de cruzamento em um ponto efetuado sobre cromossomo binário

A mutação empregada consistiu apenas em sortear um novo valor para o gene a ser mutado, seja ele real ou binário.

A cada passo do processo evolutivo, ou geração, uma nova população é gerada e avaliada. O processo pára quando, por um número pré-definido de gerações, não houver evolução, ou seja, nenhum novo indivíduo mais apto foi encontrado.

3.4. Os dados de entrada

Realiza-se a entrada de dados através da leitura de um arquivo no formato texto, com os números separados por tabulação. A primeira linha deste arquivo texto é composto por dois números: o número de amostras e o número de variáveis analisadas. Seguem-se tantas linhas quanto o número de amostras, constituídas por tantos valores quanto o número de variáveis.

Os demais parâmetros de entrada são o tamanho da população inicial, o número de gerações sem evolução (critério de parada), o tamanho da elite, a taxa de cruzamento e a taxa de mutação.

4 Testes Realizados

Foram realizados testes com dois conjuntos de dados, apresentados nas tabelas 1 e 2. O primeiro conjunto de dados é composto por duas variáveis sendo Y a variável dependente e X_1 a variável independente. O segundo conjunto de dados é composto por 4 variáveis, sendo 3 delas independentes.

Para o primeiro conjunto de dados utilizou-se um polinômio com 2 termos executando-se o programa desenvolvido por 3 vezes. Para o segundo conjunto de dados utilizou-se polinômios com 3 e 6 termos; em cada uma destas condições foram feitas 3 execuções do programa desenvolvido, totalizando 6 execuções para este conjunto de teste.

Tabela 1. Primeiro conjunto de dados

Y	X_1
15	2
10	2
12	3
9	3
7	4
8	4
5	5
4	5
3	6
4	6

Tabela 2.Segundo conjunto de dados

Y	X_1	X_2	X_3
577,4697	33,34174	57,4292	30,67267
115,2303	11,38446	93,2278	9,81286
1757,449	70,01826	80,7605	87,22733
98,43049	10,25272	42,64221	13,67697
796,2734	41,31088	71,49172	85,7296
260,3444	19,60505	97,9023	55,67425
1227,517	55,1413	41,05835	80,10589
1908,845	74,5187	10,43207	89,37881
2835,676	96,32211	18,05063	13,03298
995,497	47,93226	88,56805	38,19035
693,3578	37,66102	71,37934	4,159479
517,3588	30,98926	51,73708	38,36988
2137,435	79,77563	83,50052	74,88671
1407,239	60,38237	61,74924	84,28272
1792,998	70,95884	72,97117	72,2688
10,30469	2,277112	82,62714	6,023979
238,2047	18,47414	94,88823	13,39719
1092,964	52,28775	6,988204	78,01506
1791,684	70,92228	14,36206	5,049713
382,8107	25,43657	31,44326	95,77541

Os parâmetros genéticos foram assim definidos:

- Tamanho da população inicial: 500;
- Número de gerações sem evolução: 300;
- Tamanho da Elite: 5;
- Taxa de cruzamento: 80%
- Taxa de mutação: 5%.

Estes valores foram assim configurados com o intuito de assegurar um comportamento mais robusto ao AG desenvolvido.

5 Resultados obtidos

Em 3 execuções do programa para o primeiro conjunto de dados foram obtidos os resultados apresentados na figura 3.

$$y = 41{,}329x_1^{-1{,}234} - 36{,}734x_1^{-2{,}965}$$
$$\text{SQT}^* = 25{,}7$$

$$y = 37{,}566x_1^{-1{,}184} - 27{,}771x_1^{-2{,}968}$$
$$\text{SQT} = 26{,}4$$

$$y = 61{,}978x_1^{-1{,}455} - 76{,}781x_1^{-2{,}974}$$
$$\text{SQT} = 23{,}1$$

*SQT = Soma do quadrado dos desvios total

Figura 3. Resultados de 3 execuções do programa considerando o primeiro conjunto de dados e um polinômio com 2 termos

Utilizando-se o método dos mínimos quadrados ajustou-se um modelo linear e um modelo cúbico para este primeiro conjunto de teste. Os resultados são apresentados na figura 4.

$$y = -2{,}4x_1 + 17{,}3$$
$$\text{SQT} = 20{,}9$$

$$y = 0{,}25x_1^3 - 2{,}857x_1^2 + 7{,}607x_1 + 6{,}7$$
$$\text{SQT} = 18{,}5$$

Figura 4. Modelos ajustados pelo método dos mínimos quadrados

Em 3 execuções do programa para o segundo conjunto de dados com 3 termos no polinômio foram obtidos os resultados apresentados na figura 5.

$$y = 3{,}593x_1^{1{,}458} - 44{,}831x_2^{-0{,}857} - 7{,}005x_3^{0{,}158}$$
$$\text{SQT} = 3915{,}1$$

$$y = 4{,}953x_1^{1{,}402} - 21{,}276x_1^{0{,}489} + 4{,}689x_2^{0{,}346}$$
$$\text{SQT} = 3760{,}6$$

$$y = 3{,}813x_1^{1{,}446} + 54{,}676x_1^{-1{,}37796} - 67{,}644x_2^{-0{,}269}$$
$$\text{SQT} = 4261{,}5$$

Figura 5. Resultados de 3 execuções do programa considerando o segundo conjunto de dados e um polinômio com 3 termos

Nas 3 execuções do programa para o segundo conjunto de dados com 6 termos no polinômio obteve-se os resultados apresentados na figura 6.

$$y = \frac{\left(34{,}261x_1^{1{,}672} \cdot 52{,}534x_2^{-1{,}658}\right) + 2{,}301x_1^{1{,}677} - 40{,}102x_1^{2{,}651}}{-13{,}645x_1^{1{,}146}} + 43{,}065x_3^{-1{,}599}$$

SQT = 153,6

$$y = -\left(\frac{7{,}049x_3^{-0{,}196} - 33{,}649x_1^{0{,}359} + 35{,}451x_3^{0{,}105}}{-10{,}552x_3^{-0{,}122}}\right) \cdot 69{,}128x_1^{-0{,}401} + 7{,}536x_1^{1{,}308}$$

SQT = 13851,6

$$y = \left(-0{,}243x_1^{1{,}982} + 88{,}652x_2^{0{,}388} + 63{,}281x_1^{1{,}162}\right) \cdot 48{,}385x_1^{0{,}552}$$

SQT = 5333,4

Figura 6. Resultados de 3 execuções do programa considerando o segundo conjunto de dados e um polinômio com 6 termos

Utilizando-se o método dos mínimos quadrados ajustou-se um modelo linear e um modelo cúbico para este primeiro conjunto de teste. Os resultados são apresentados na figura 7.

$$y = 30{,}779x_1 + 0{,}196x_2 - 1{,}812x_3 - 290{,}178$$

SQT = 207207,3

$$y = -0{,}001x_1^3 + 0{,}284x_1^2 + 9{,}37x_1 - 0{,}039x_2^2 + 2{,}505x_2 - 0{,}004x_3^2 - 0{,}138x_3 - 66{,}232$$

SQT = 376,6

Figura 7. Modelos ajustados pelo método dos mínimos quadrados

6 Considerações Finais

A comparação dos resultados obtidos pelo *Model Breeder* implementado com os resultados obtidos pelo método dos mínimos quadrados possibilitou observar que os modelos ajustados pelo primeiro podem ser considerados bons.

Para dados simples, como os do conjunto de teste 1, o *Model Breeder* apresentou modelos menos precisos que os modelos ajustados pelo método dos mínimos quadrados. O primeiro ajustou modelos com somatório do quadrado dos desvios total (SQT) variando entre 23,1 e 26,4; o segundo ajustou um modelo linear com SQT = 20,9 e um modelo cúbico com SQT = 18,5.

Para dados mais complexos, como os do conjunto de teste 2, o *Model Breeder* conseguiu ajustar modelos até mais precisos que os modelos obtidos através do método dos mínimos quadrados. O primeiro ajustou modelos com SQT variando entre 153,6 e 13851,6; o segundo ajustou um modelo linear com SQT = 207207,3 e um modelo cúbico com SQT = 376,6.

Um número maior de testes deve ser realizado para assegurar que o *Model Breeder* baseado em AGs é melhor que o método dos mínimos quadrados. Os testes preliminares permitem considerá-lo um bom método para análise exploratória de dados.

Uma vantagem apresentada pelo *Model Breeder* é a natureza semi-automática do processo. Não é necessário preparar os dados de entrada. Programas que ajustam modelos através do método dos mínimos quadrados, como o *Microsoft Excel* e o *Minitab*, exigem a preparação dos dados de entrada, através da inserção das colunas correspondentes às variáveis que serão consideradas no modelo.

7 Referências

[1] Davis, L., Adapting operator probabilities in Genetic Algorithms. *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers Inc., San Mateo, 1989, 61-69.

[2] Galvão, C. O.; Valença, M. J. S., *Sistemas inteligentes: aplicações a recursos hídricos e sistemas ambientais*, Ed. Universidade/UFRGS/ABRH, Porto Alegre, 1999.

[3]Goldberg, D. E., *Genetic algorithms in search, optimization & machine learning*. Addison-Wesley, Reading, 1989.

[4] Herrera, F.; Lozano, M.; Verdegay, J. L., Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioural Analysis, *Artificial Intelligence Review*, Kluwer Academic Publishers, 1998, 12, 265-319.

[5] Holland, J. H., *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor, 1975.

[6] Lucasius, C. B.; Kateman, G. Applications of genetic algorithms in chemometrics. *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers Inc., San Mateo, 1989, 170-176.

[7] Michalewicz, Z., *Genetic algorithms + data structures = evolution programs*, Springer-Verlag, Berlim, 1994.

[8] Neter, J.; Kutner, M. H.; Nachtsheim, C. J.; Wasserman, W., *Applied Linear Statistical Models*, Irwin, Chicago, 1996.

[9] Openshaw, S. *Conventional, Neural, and Genetic Spatial Interation Models***,** 1997a. <http:// www.ccg.leeds.ac.uk/staff/s.openshaw/vienna97/index. html>. Visitado em 02/03/2005.

[10] Openshaw, S.; Openshaw, C., *Artificial intelligence in geography*, John Wiley & Sons Ltd., West Sussex, 1997b.