Construção de Conhecimento de Alto Nível a Partir de *Datasets* Biológicos com Alta Dimensionalidade Utilizando Algoritmos Genéticos

Reslley Gabriel Oliveira Silva[1], **Fabrício Alves Rodrigues**[1], **André Bevilaqua**[1], **Franciny Medeiros Barreto**[1], **Thamer Horbylon Nascimento**[1], **Laurence Rodrigues do Amaral**[1]

[1]Curso de Ciência da Computação - Universidade Federal de Goiás/Jataí (UFG)
BR364 Km 192, Setor Industrial, Jataí - GO - Brasil

{reslleygabriel,laurence.amaral}@gmail.com

Abstract. *An area where the application of computational techniques has shown more promising is the Molecular Biology, where the selection of relevant genes to a particular disease becomes an important task, and may in the near future, be applied in medical diagnosis. In pursuit of these small sets of predictors genes, Genetic Algorithms (GAs) are increasingly used because of their ability to learn automatically from large volumes of data and generate useful hypotheses. Seeking this subset of genes, we implemented the genetic algorithm proposed by Amaral and Oliveira, that originally was applied using only 55 genes from the database NCI60, applying it to all the genes present in NCI60 (1,000 genes) by checking if it can converge to the same fitness values.*

Resumo. *Uma das áreas em que a aplicação de técnicas computacionais tem se mostrado mais promissora é a Biologia Molecular, principalmente na seleção de genes que possuem fortes correlações a uma doença. Na busca destes conjuntos de genes preditores, os Algoritmos Genéticos são cada vez mais empregados, devido a sua capacidade de aprender automaticamente a partir de grandes volumes de dados e produzir hipóteses úteis. Buscando este subconjunto de genes, implementamos o Algoritmo Genético proposto por Amaral e Oliveira, que originalmente foi aplicado utilizando somente 55 genes da base de dados NCI60, aplicando-o a todos os genes presentes na NCI60 (1.000 genes) verificando se o mesmo consegue convergir, obtendo os mesmos valores de aptidão.*

Palavras-chave: Bioinformática, Algoritmos Genéticos, Mineração de Dados, Regras IF-THEN

1. Introdução

Do início até meados do século passado, os geneticistas e químicos se questionaram sobre a natureza química do material genético. Das pesquisas desenvolvidas, concluiu-se que o DNA (sigla em inglês para ácido desoxirribonucléico) era a molécula que armazenava a informação genética e em 1953 sua estrutura química foi descoberta no clássico trabalho de Watson e Crick [Watson and Crick 1953]. Com a posterior descoberta do código genético e do fluxo da informação biológica dos ácidos nucléicos, tais polímeros passaram a constituir os principais objetos de estudo de uma nova ciência, a Biologia Molecular.

Logo surgiram métodos de sequenciamento destes polímeros, principalmente do DNA, que permitiam a investigação de suas sequências monoméricas constituintes. Desde então, mais de 18 bilhões dessas sequências já foram produzidas e estão disponíveis nos bancos de dados públicos [Borém et al. 2003].

Na segunda metade de década de 90, com o surgimento dos sequenciadores automáticos de DNA, houve uma explosão na quantidade de sequências a serem armazenadas, exigindo recursos computacionais cada vez mais eficientes. Além do armazenamento ocorria, paralelamente, a necessidade da análise desses dados, o que tornava indispensável a utilização de plataformas computacionais eficientes para a interpretação dos resultados obtidos [Borém et al. 2003].

Atualmente, a Bioinformática é imprescindível para a manipulação dos dados biológicos. Ela pode ser definida como uma modalidade que abrange todos os aspectos de aquisição, processamento, armazenamento, distribuição, análise e interpretação da informação biológica. Através da combinação de procedimentos e técnicas advindas da matemática, estatística e ciência da computação, são elaborados vários métodos e ferramentas que auxiliam na compreensão do significado biológico representado nos dados genômicos [Borém et al. 2003].

Uma das áreas em que a aplicação de técnicas computacionais tem se mostrado mais promissora é a Biologia Molecular [Setúbal and Meidanis 1997]. O termo, expressão gênica, refere-se ao processo em que a informação codificada por um determinado gene é decodificada em uma proteína, manifestando assim, características particulares àquele gene. As células e tecidos têm suas funções normais quando os genes são expressos de forma regulada. A expressão alterada de um gene pode alterar o equilíbrio do organismo, podendo vir a gerar uma doença. Assim, a seleção de genes relevantes a uma determinada doença torna-se uma tarefa importantíssima, podendo num futuro próximo, ser aplicada no diagnóstico médico.

Na busca destes pequenos conjuntos de genes preditores, técnicas advindas da Inteligência Artificial (IA), tais como, os Algoritmos Genéticos (AGs) e as Redes Neurais Artificiais (RNAs), são cada vez mais empregadas, devido a sua capacidade de aprender automaticamente a partir de grandes volumes de dados e produzir hipóteses úteis [Baldi and Brunak 2001].

De posse destes conjuntos preditores se faz extremamente necessária a análise dos mesmos utilizando ferramentas bioinformáticas tradicionais, buscando assim estabelecer padrões e relações entre os genes analisados e todos os demais encontrados na literatura, na busca de um melhor entendimento desta complexa relação entre gene/gene e gene/câncer, contribuindo assim para o diagnóstico destes tipos de câncer.

2. Objetivos e Metas

O presente trabalho tem como objetivo implementar o Algoritmo Genético (AG) [Goldberg 1989] proposto por Amaral e Oliveira [Amaral 2007], adaptando-o para trabalhar com expressão gênica de 1.000 genes (o original trabalhava com no máximo 55 genes, utilizando métodos propostos por outros autores como método de pré-seleção gênica) e utilizando o método de validação de dados *leave-one-out* (o empregado no trabalho foi o método 2:1), mais adequado para o problema em questão.

Nossa meta é mostrar que o método proposto por Amaral e Oliveira [Amaral 2007] consegue convergir, isto é, obtém os mesmos valores de aptidão, quando utilizado na mineração de regras em bases de dados biológicas, com características de alta dimensionalidade. Iremos propor novas estratégias, tais como: utilizar abordagens não lineares ou operadores específicos, e mudanças nos operadores genéticos do Algoritmo Genético, tais como: tamanho de população, métodos de seleção, taxa de crossover, taxa de mutação, dentre outros.

3. Metodologia

Implementamos o método proposto por Amaral e Oliveira [Amaral 2007], alterando o mesmo para trabalhar com 1.000 genes, utilizando o método de validação *leave-one-out* e colocamos o ambiente evolutivo para minerar a base NCI60 [Ross et al. 2000]. Ela é composta pela expressão de 1.000 genes (colunas), medida para 61 amostras de células (linhas), sendo que cada amostra é classificada em uma das nove classes de câncer, tais como: mama (7), sistema nervoso central (6), cólon (7), leucemia (6), melanoma (8), pulmão (9), ovário (6), renal (8) e células reprodutivas (4), totalizando 61 amostras, onde o número de ocorrências de cada classe é dado pelo valor entre parênteses.

É importante salientar que utilizamos todos os parâmetros genéticos do Algoritmo Genético idênticos aos propostos por Amaral e Oliveira [Amaral 2007]. Além disso, a base NCI60 é considerada um desafio para os algoritmos de classificação por suas características peculiares: um número relativamente alto de classes (9) para um número relativamente baixo de amostras (61), resultando em número baixo de amostras por classe, variando de 4 a 9 amostras por classe, além da alta dimensionalidade (1.000 colunas). Segue abaixo as principais características deste ambiente evolutivo.

O modelo do AG implementado foi o mesmo desenvolvido em [Amaral 2007]. Em [Amaral 2007], o método desenvolvido foi adaptado a partir do modelo de AG proposto em [Fidelis et al. 2000]. O AG em [Fidelis et al. 2000] foi desenvolvido na ferramenta GALOPPS [Goodman 1996] e foi elaborado com o objetivo de obter regras de classificação do tipo IF-THEN em bases de dados clínicos de pacientes.

Dessa forma, as bases de dados onde o ambiente de Fidelis e colaboradores [Fidelis et al. 2000] foram aplicadas eram formadas por registros que se caracterizavam por dados do paciente, no caso, a idade e presença da doença em histórico familiar e por dados relacionados aos sintomas do paciente, tal como, presença abundante de manchas brancas na face. As características que se relacionavam aos sintomas, que eram a maioria, foram todas discretizadas em: 0-ausente, 1-ocorrência leve, 2- ocorrência moderada e 4-ocorrência severa.

O ambiente proposto em [Amaral 2007] foi implementado na linguagem Delphi®, e precisou ser adaptado para trabalhar com bases de dados de expressão gênica, onde os registros apresentam os níveis de expressão de dezenas (centenas ou milhares) de genes, que são valores contínuos e com precisão variável (números reais). Em nossa implementação utilizamos a linguagem de programação Java®.

4. Características do AG

O indivíduo ou cromossomo do nosso AG é composto por N genes, onde cada gene do indivíduo está relacionado a uma condição envolvendo um atributo (um gene do *dataset*

NCI60), onde *N* é o número de genes encontrados na base de expressão gênica. A primeira posição do indivíduo corresponde ao primeiro gene encontrado na base de dados e assim sucessivamente até que todos os genes do *dataset* estejam representados. O indíviduo é ilustrado na Figura 1.

$Gene_1$			...	...	$Gene_N$		
P_1	O_1	V_1	...	...	P_N	O_1	V_1

Figure 1. Cromossomo ou Indivíduo

Cada i-ésima posição do indivíduo é subdividida em três campos: Peso (P_i), Operador (O_i) e Valor (V_i), como ilustrado na Figura 1. Cada gene corresponde a uma condição na parte SE da regra e o individuo (cromossomo) a toda a parte SE da regra. O campo P_i é uma variável do tipo inteira e o seu valor está compreendido entre os valores 0 (zero) e 10 (dez). É importante dizer que este campo P_i é o responsável pela inserção ou exclusão do gene na regra. Caso este valor seja menor do que um valor limite este gene não fará parte da regra, caso contrário o mesmo fará. Neste trabalho foi utilizado como limite o valor 7 (sete). O campo O_i pode variar entre as operações $<$ (menor) e $\geq$ (maior ou igual). O campo V_i é uma variável do tipo ponto flutuante que pode variar entre o menor e o maior valor encontrados na base de expressão gênica avaliada.

Os operadores genéticos são necessários para que a população se diversifique e mantenha características de adaptação adquiridas pelas gerações anteriores, os mais utilizados são a mutação e o *crossover*. Na seleção dos pais para *crossover* aplicamos o método do Torneio Estocástico utilizando *tour* de tamanho 3 (três). Nestes pais selecionados, aplicamos *crossover* múltiplo com dois pontos de corte, gerando dois novos filhos com taxa de *crossover* de 100%. Nestes dois filhos gerados, aplicamos o operador de mutação. Os operadores de mutação utilizados neste trabalho variam com o tipo do gene avaliado e possui taxa de mutação por gene no valor de 30%. A mutação no campo P_i é dada pelo sorteio de um novo valor entre 1 e 10. Para o campo O_i ocorre o sorteio de um novo operador dentre os possíveis, excluindo o encontrado originalmente. Neste trabalho foram utilizados apenas dois operadores ($\geq$ e $<$), levando à troca de um pelo outro. O valor para o campo V_i é obtido pelo sorteio de um novo valor, que varia entre o menor e o maior valor presente na população. Na composição dos indivíduos que irão participar da próxima geração do AG, selecionamos os melhores pais e filhos.

A Aptidão (ou *fitness*) refere-se ao grau de contribuição de uma determinada solução candidata para a convergência do AG na busca da melhor solução dentro do espaço de busca.

Neste trabalho a Função de Avaliação ou Aptidão (FA) (Fitness Function) avalia a qualidade de cada regra (indivíduo). A FA utilizada pode ser encontrada em [Lopes et al. 1997]. Para o perfeito entendimento da FA aqui aplicada, alguns conceitos precisam ser reforçados. Quando utilizamos uma determinada regra na classificação de um exemplo, quatro diferentes tipos de resultados podem ser observados, dependendo da classe predita pela regra e a verdadeira regra do exemplo. São eles:

- *True Positive* (tp) - A regra prediz que o exemplo pertence a uma determinada

classe e o mesmo pertence;

- *False Positive* (fp) - A regra prediz que o exemplo pertence a uma determinada classe mas o mesmo não pertence;
- *True Negative* (tn) - A regra prediz que o exemplo não pertence a uma determinada classe e o mesmo não pertence;
- *False Negative* (fn) - A regra prediz que o exemplo não pertence a uma determinada classe mas o mesmo pertence;

A FA usa dois indicadores comumente utilizados em domínios médicos, chamados de sensibilidade (*Se*) e especificidade (*Sp*). *Se* e *Sp* são definidos abaixo:

$$Se = \frac{tp}{(tp + fn)} \tag{1}$$

$$Sp = \frac{tn}{(tn + fp)} \tag{2}$$

Finalmente, a FA utilizada é definida como a média aritmética simples destes dois indicadores, *Se* e *Sp*, como segue abaixo:

$$Aptidao = (Se + Sp)/2 \tag{3}$$

O objetivo do trabalho é maximizar ao mesmo tempo *Se* e *Sp* e consequentemente a *Aptidao*, utilizando para isso, as equações 1, 2 e 3. Em cada execução, o nosso AG trabalha com um problema de classificação de duas classes, isto é, quando o AG está procurando por regras de uma dada classe, todas as outras classes são agrupadas em uma única classe.

5. Resultados obtidos e Trabalhos Futuros

Após rodar o ambiente evolutivo, constatamos que o mesmo não consegue convergir utilizando os parâmetros utilizados por Amaral e Oliveira. Acreditamos que isso ocorreu devido à alta dimensionalidade presente na base de dados NCI60 (1.000 atributos).

Frente a este cenário, iremos alterar o modelo construído, inserindo novas abordagens e alterando os parâmetros genéticos do ambiente evolutivo. Segue abaixo as alterações que pretendemos fazer ao modelo:

- uso de populações maiores;
- aumentar o tamanho do *tour* utilizado no trabalho (*tour* = 3) e uso de outros métodos de seleção que possuam características mais seletivas que o torneio estocástico;
- usar taxas de *crossover* diferentes de 100%;
- usar percentuais de mutação maiores que os adotados no modelo proposto e propor métodos de mutação diferenciados (para fugirmos de ótimos locais);
- utilizar abordagens não lineares propostas por [Amaral and Hruschka 2011a];
- utilizar abordagens o operador *Transgenic* proposto por [Amaral and Hruschka 2011b];

- comparar os resultados obtidos com outros métodos tradicionais de classificação presentes no ambiente Weka [Homes et al. 1994];
- aplicar os conjuntos de genes encontrados a ferramentas tradicionais de análise bioinformática, tais como: BLAST, PHYLIP, ClustalW e ALIGN;
- dentre outros.

Desta forma, acreditamos que nosso ambiente evolutivo possa convergir quando aplicado à NCI60 (aplicado aos 1.000 genes) e consiga gerar conhecimento de alto nível na forma de regras IF-THEN e que este conhecimento possa ser auxiliar os especialistas na busca pela cura do câncer.

References

Amaral, L. R. (2007). Mineração de regras para classificação de oncogenes medidos por microarray utilizando algoritmos genéticos. Master's thesis, Universidade Federal de Uberlândia.

Amaral, L. R. and Hruschka, E. R. (2011a). Non-linear computational evolutionary environment (nlcee): Building high-level knowledge in complex biological databases. In *ECML/PKDD - European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases. No workshop: Data Mining in Functional Genomics and Proteomics: Current Trends and Future Directions, 2011, Atenas. Anais do ECML PKDD 2011 - The 5th International Workshop on Data Mining in Functional Genomics and Proteomics: Current Trends and Future Directions*.

Amaral, L. R. and Hruschka, E. R. (2011b). Transgenic, an operator for evolutionary algorithms. In Press, I., editor, *IEEE Congress On Evolutionary Computation, 2011, New Orleans. Proceedings of the IEEE CEC 2011*, Los Alamitos.

Baldi, P. and Brunak, S. (2001). *Bioinformatics: the Machine Learning approach*. MIT Press, 2 edition.

Borém, A., Giúcide, M., and Sedyiama, T. (2003). *Melhoramento Genômico*. Universidade Federal de Viçosa.

Fidelis, M. V., Lopes, H. S., and Freitas, A. A. (2000). Discovery comprehensible classification rules with a genetic algorithm. In *Congress on Evolutionary Computation - (CEC-2000)*, pages 805–810. La Jolla, CA, USA.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Adison-Wesley, USA.

Goodman, E. D. (1996). An introduction to gallops - the genetic algorithms optimized for portability and parallelism system. Technical report, Departament od Computer Science - Michigan State University.

Homes, G., Donkin, A., and Witten, I. H. (1994). Weka: A machine learning workbench. *Proceedings of the Second Australia and New Zealand Conference on Intelligent Information Systems*.

Lopes, H. S., Coutinho, M. S., and Lima, W. C. (1997). An evolutionary approach to simulate cognitive feedback learning in medical domain. In Sanchez, E., Shibata, T., and Zadeh, L. A., editors, *Genetic Algorithms and Fuzzy Logic Systems*, pages 193–207. World Scientific.

Ross, D. T., Scherf, U., Eisen, M. B., Perou, C. M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S. S., Van de Rijn, M., Waltham, M., Pergamenschikov, A., Lee, J. C. F., Lashkari, D., Shalon, D., Myers, T. G., Weinstein, J. N., Botstein, D., and Brown, P. O. (2000). Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics*.

Setúbal, J. C. and Meidanis, J. (1997). *Introduction to Computacional Molecular Biology*. PWS Publishing Company, Boston.

Watson, J. D. and Crick, F. H. (1953). Letters to nature: Molecular structure of nucleic acid. *Nature*, 171:737–738.