Programação Genética para Classificação de Dados de Pacientes Infectados com COVID-19

Gianni R. S. Da Conceição[1], **Camila S. De Magalhães**[1]

[1]Núcleo Multidisciplinar de Pesquisa em Computação
Universidade Federal do Rio de Janeiro (UFRJ) - Duque de Caxias - Brasil

ribeirogianni@ufrj.br, camila@caxias.ufrj.br

Abstract. *In this work, a Genetic Programming (GP) algorithm was developed to classify a database of COVID-19 infected patients. The algorithm presented about 85% of accuracy in predicting the disease prognosis based on symptoms, potentially serving as a valuable tool for prioritizing hospitalizations and identifying the main factors that may lead to mortality. Additionally, the algorithm was tested on reference datasets to validate its generalization capability, obtaining competitive results.*

Resumo. *Neste trabalho foi desenvolvido um algoritmo de Programação Genética (PG) para classificação de um banco de dados de pacientes infectados com COVID-19. O algoritmo apresentou cerca de 85% de acurácia na predição do prognóstico da doença a partir dos sintomas, podendo ser uma ferramenta útil para priorização de internações e na identificação dos principais fatores que podem levar ao óbito. O algoritmo também foi testado em conjuntos de dados de referência para validar sua capacidade de generalização, obtendo resultados competitivos.*

1. Introdução

Em problemas de classificação de dados o objetivo é desenvolver um classificador (modelo ou regra), que seja mais adequado a um conjunto de dados de treinamento do tipo: atributos (x1,...,xn): classe associada (y). Uma vez desenvolvido, o classificador pode ser utilizado para predizer a classe correta quando aplicado a novos dados de entrada para os quais a classe associada não é conhecida. Neste trabalho, um algoritmo de Programação Genética (PG) [Eiben and Smith 2015] foi implementado e avaliado para o problema de classificação de dados. A PG pertence à classe dos algoritmos evolucionistas, abrangendo algoritmos inspirados na teoria da evolução das espécies. Estes algoritmos são baseados na sobrevivência do mais apto e na evolução de uma população inicial aleatória de indivíduos (possíveis soluções para o problema alvo). Devido à seus indivíduos em formato de árvore de decisão, a PG tem sido aplicada com sucesso para a evolução de classificadores interpretáveis e explicáveis [Hu 2023], sendo de grande importância na área da saúde. O algoritmo desenvolvido foi aplicado para a classificação de dados de pacientes contaminados com COVID-19, identificando os indivíduos com maior probabilidade de desenvolver sintomas graves da doença. Para avaliar a capacidade de generalização do método e comparar seu desempenho com a literatura, o algoritmo foi testado em cinco bancos de dados, obtidos no UCI Machine Learning Repository, comumente utilizados para comparação de desempenho entre classificadores.

2. Metodologia

2.1. Algoritmo de Programação Genética

A população inicial da PG desenvolvida foi gerada através do método Ramped half-and-half [Eiben and Smith 2015] e teve seus parâmetros definidos empiricamente. A profundidade (número de níveis verticais da árvore de decisão) máxima selecionado foi 5 e cada indivíduo é uma árvore de decisão do tipo SE...ENTÃO...SENÃO. O tamanho de população utilizado foi 500 indivíduos. Para o conjunto de funções foram utilizados os operadores matemáticos de soma, subtração, multiplicação e divisão, além das funções potência (quadrado) e raiz quadrada. Os operadores relacionais maior, menor, igual e diferença e os operadores lógicos E, OU e NÃO também foram utilizados. Após a inicialização, a população inicial passa por um ciclo completo de evolução, chamado de geração. A cada geração os indivíduos tem sua aptidão (fitness) calculada como a acurácia na classificação. Ests indivíduos são então selecionados para gerar novos indivíduos por torneio binário [Eiben and Smith 2015]. A reprodução é realizada com a aplicação dos operadores genéticos de mutação e recombinação de subárvore, com probabilidade igual entre eles. No operador de mutação um individuo selecionado tem sua árvore de decisão recriada a partir de um ponto selecionado aleatoriamente. No operador de recombinação dois indivíduos são selecionados e tem partes de suas árvores de decisão trocadas em pontos (nós) selecionados de forma aleatória. Foi utilizado o modelo de substituição geracional com elitismo igual a 1 e o proceso de evolução seguiu por 200 gerações. A avaliação dos classificadores foi realizada com o método 10-Fold Cross Validation, no qual o banco de dados é dividido em 10 partições com mesmo número de amostras e são realizadas 10 iterações. Em cada iteração, uma das partições é utilizada como banco de teste e as demais como banco de treinamento. A acurácia final do classificador é dada pela média dos resultados das 10 iterações realizadas. O algoritmo foi aplicado para cinco bancos de dados obtidos do UCI Machine Learning Repository [Dua and Graff 2019]: Wisconsin Breast Cancer (WBC), Cleveland Heart Disease (CHD), Iris, Wine e Glass.

2.2. Banco de Dados de COVID-19

O banco de dados de pacientes com COVID-19 foi obtido através da Nature Scientific Data [Xu et al. 2020], contendo dados médicos de pacientes de Covid-19 de todo o mundo. Foi realizado um tratamento prévio destes dados, com seleção dos seguintes atributos: idade, sexo, doenças crônicas e sintomas do paciente. As diferentes doenças crônicas também foram agrupadas em subcategorias, por exemplo, sintomas como 'falta de ar' e 'problemas para respirar' foram agrupados em 'problemas respiratórios'. Como classe, foram utilizadas as informações de prognóstico bom (alta) ou ruim (óbito). Também foram selecionados apenas dados de pacientes que apresentaram ao menos um sintoma ou uma doença crônica. Após a preparação, o banco de dados utilizado contém 194 amostras, sendo 71 de pacientes de prognóstico bom e 123 de pacientes com prognóstico ruim.

3. Resultados

3.1. Banco de dados obtidos do *UCI Machine Learning Repository*

A Tabela 1 apresenta os resultados de treinamento e teste da PG implementada, além do número de registros por classe para cada banco. Como medida de comparação com a literatura, são apresentados os resultados de [Cheruku et al. 2018]. O algoritmo de PG obteve

um resultado competitivo com a literatura para a maioria dos bancos, com exceção dos bancos Glass e CHD. Estes são os dois bancos que possuem o maior número de classes, com uma distribuição altamente desbalanceda entre elas, evidenciando que o algoritmo desenvolvido possui melhor desempenho para classificação de bancos de dados binários e balanceados.

Tabela 1. Acurácia média para treinamento, teste e literatura

	Iris 50/50/50	**Glass** 9/13/17/29/70/76	**CHD** 13/35/36/55/164	**Wine** 48/59/71	**WBC** 241/458
Treinamento	97,60 %	65,80 %	63,20 %	93,00 %	94,60 %
Teste	94,70 %	61,20 %	54,40 %	89,80 %	90,20 %
Literatura	99,11 %	80,45 %	80,67 %	100,00 %	98,24 %

Ao observar as árvores de decisão geradas ao final do treinamento do algoritmo, é possível associar a perda de diversidade dos indivíduos da população aos resultados negativos. Para o futuro métodos de nichos como RTS [Harik 1995] podem ser utilizados para trazer uma maior diversidade populacional ao longo das gerações.

3.2. Banco de dados de pacientes com COVID-19

Observada a dificuldade em lidar com bancos de dados não balanceados, para a avaliação dos dados de pacientes contaminados com COVID-19 [Xu et al. 2020], foi implementada uma nova função fitness utilizada para bancos de dados binários. Nesta função, o cálculo de aptidão é feito conforme a equação abaixo. Esta função possui a vantagem de considerar os verdadeiros positivos (VP), verdadeiros negativos (VN), falsos positivos (FP) e falsos negativos (FN), com uma constante de parametrização W, que pode ser alterada para ajuste do algoritmo visando um melhor desempenho. Quanto maior o valor de W, mais a classe minoritária (classe positiva) contribuirá para a aptidão e quanto menor o valor de W, mais a classe majoritária (classe negativa) irá contribuir.

$$Fitness = W\left(\frac{VP}{VP + FN}\right) + (1 - W)\left(\frac{VN}{VN + FP}\right)$$

Figura 1. Função fitness para bancos binários

Na tabela abaixo são apresentados os resultados de aptidão utilizando diferentes valores de W, fixos e variáveis (aumento ou decréscimo linear durante a evolução). O melhor resultado foi encontrado ao se utilizar o parâmetro W fixo em 0,3 (maior contribuição da classe majoritária).

O melhor classificador obtido possui acurácia de 95,70% e revela que idade e problemas respiratórios são fatores-chave, assim como problemas cardíacos, corroborando informações já conhecidas sobre a COVID-19. Pacientes mais novos, sem doenças crônicas e que não apresentam problemas respiratórios, tendem a ter um bom prognóstico. Profissionais da saúde podem utilizar estes classificadores para a triagem de pacientes a partir de seus sintomas e histórico médico, facilitando o tratamento prioritário e otimizando a alocação de recursos médicos, pois mesmo pós-pandemia, a COVID-19 ainda é uma doença perigosa para pessoas nos grupos de riscos.

Tabela 2. Resultado para diferentes parâmetros da nova função fitness no banco de pacientes com COVID-19

	W = 0.3	W = 0.8	W variável Inicio: 0.3 Fim: 0.8	W variável Inicio: 0.8 Fim: 0.3
Treinamento	87,82 %	80,57 %	78,73 %	74,08 %
Teste	87,00 %	78,00 %	72,00 %	68,00 %

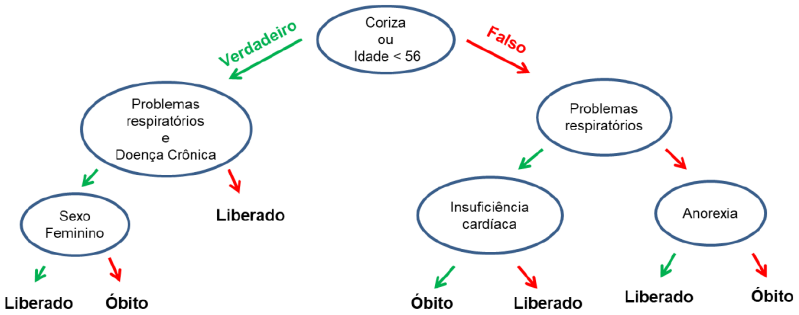

Figura 2. Melhor classificador obtido em forma de árvore de decisão

4. Conclusões

O algoritmo de programação genética (PG) implementado para classificação de dados de pacientes com COVID-19, obteve uma acurácia média de 87,00% e o melhor classificador obteve uma acurácia de 95,70%. A regra encontrada corrobora a correlação já conhecida entre a idade e problemas respiratórios ao agravamento da doença, mostrando a capacidade da PG para classificação e extração de informações diretamente das regras geradas. Esses resultados indicam que o algoritmo desenvolvido apresenta potencial e para ser aplicado a outros bancos de dados relacionados a outras doenças.

Referências

Cheruku, R., Edla, D. R., Kuppili, V., and Dharavath, R. (2018). Rst-batminer: A fuzzy rule miner integrating rough set feature selection and bat optimization for detection of diabetes disease. *Applied Soft Computing*, 67:764–780.

Dua, D. and Graff, C. (2019). Uci machine learning repository [http://archive. ics. uci. edu/ml]. *IEEE transactions on pattern analysis and machine intelligence*, 1(1):1–29.

Eiben, A. E. and Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer.

Harik, G. (1995). Finding multimodal solutions using restricted tournament selection.

Hu, T. (2023). *Genetic Programming for Interpretable and Explainable Machine Learning*. Springer Nature Singapore, Singapore.

Xu, B. et al. (2020). Epidemiological data from the covid-19 outbreak, real-time case information. *Scientific data*, 7(1):106.