UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA
Autorizada pelo Decreto Federal nº 77.496 de 27/04/76
Recredenciamento pelo Decreto nº17.228 de 25/11/2016
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
COORDENAÇÃO DE INICIAÇÃO CIENTÍFICA

XXIX SEMINÁRIO DE INICIAÇÃO CIENTÍFICA DA UEFS
SEMANA NACIONAL DE CIÊNCIA E TECNOLOGIA - 2025

Melhorias no Algoritmo Apriori para a Geração de Classificadores Baseados em Regras de Fuzzy Interpretáveis

João Gabriel Lima Almeida[1]; Fabiana Cristina Bertoni[2]
1.João Gabriel Lima Almeida, PROBIC/UEFS, ENGENHARIA DE COMPUTAÇÃO, e-mail: gabriel.lima.almeida@gmail.com
2.Fabiana Cristina Bertoni, DEPARTAMENTO DE CIÊNCIAS EXATAS, e-mail: fcbertoni@uefs.br

PALAVRAS-CHAVE: classificadores fuzzy; regras de associação; melhoria de desempenho.

INTRODUÇÃO

O uso de Sistemas Classificadores Baseados em Regras Fuzzy (SCBRF) é amplamente comum na resolução de problemas de classificação e tomada de decisões automatizadas. Diferente da lógica clássica, a lógica fuzzy permite que os SCBRFs utilizem regras de associação para lidar com dados contínuos, fracionados e não booleanos, que são o tipo de dado mais comum em aplicações reais.

Um dos principais fatores que determinam a eficiência desses classificadores são as bases de regras utilizadas. Um bom conjunto de regras se traduz em uma boa classificação e, portanto, a geração dessas regras é um ponto de extrema importância. Algoritmos como o FARC-HD — Fuzzy Association Rule-based Classification model for High-Dimensional problems (ALCALÁ-FDEZ *et al.*, 2011) —, são utilizados para gerar bases de regras de associação fuzzy para SCBRFs. A geração manual dessas regras, além de extremamente trabalhosa, não leva em conta todas as nuances e padrões escondidos em meio às bases de dados, ainda mais aquelas pouco legíveis como as de dados contínuos. Por isso, a utilização de algoritmos como o FARC-HD é essencial, principalmente quando se trata de problemas de alta dimensionalidade.

Entretanto, gerar essas regras é uma tarefa computacionalmente custosa. O FARC-HD, mesmo sendo o estado da arte no quesito geração de regras, possui problemas de desempenho bem conhecidos devido, principalmente, ao uso de uma variação do algoritmo Apriori em seu funcionamento interno: o Apriori Fuzzy. Outro defeito notável do FARC-HD está na resolução de problemas com base de dados desbalanceados. Devido a seleção de *itemsets* frequentes baseada em métricas dependentes de parâmetros, há um favorecimento de regras de classes majoritárias em detrimento das regras de classes minoritárias.

SANZ, J e colaboradores (2021) propõem o FARCI, uma versão modificada do FARC-HD com melhorias para problemas desbalanceados. Essas mudanças descritas no trabalho, ajudam a atenuar o problema das regras minoritárias, entretanto, a implementação disponível do Apriori Fuzzy do FARCI ainda carrega o desempenho computacional ruim do FARC-HD, além de outros problemas como acoplamento.

Nesse contexto, esse trabalho apresenta uma reimplementação do algoritmo Apriori Fuzzy que contém, além das melhorias propostas por SANZ e colaboradores (2021), melhorias focadas em diminuir o custo computacional do algoritmo.

MATERIAL E MÉTODOS OU METODOLOGIA (ou equivalente)

O Apriori Fuzzy é um algoritmo que realiza uma série de operações para, dado um *dataset* e os limites de suas variáveis, elencar o conjunto dos *itemsets* frequentes. Esses *itemsets* são conjuntos de antecedentes, representados por uma variável — coluna da matriz do *dataset* — e uma label — função de pertinência fuzzy da variável correspondente da coluna. Quando associados a uma classe consequente, formam uma regra de associação fuzzy: SE $\{A_1:X_1 \ldots A_n:X_n\}$, ENTÃO $\{C:X_n\}$, onde A são os antecedentes, X as labels e C o consequente (variável de saída) (ALCALÁ-FDEZ *et al*., 2011).

Para chegar nesses *itemsets* frequentes, o algoritmo realiza a expansão da árvore de combinações dos antecedentes, partindo dos *itemsets* de apenas 1 antecedente e combinando-os até não haver mais caminhos possíveis ou atingir a profundidade de limitação máxima. A cada combinação, o algoritmo verifica o suporte de classe do itemset — grau que mede o quão presente aquele conjunto de antecedentes está no *dataset* — e compara a um parâmetro de suporte mínimo que define se o itemset é ou não, frequente. O princípio de podagem do Apriori estabelece que: se um dado n-itemset não é frequente, todos os seus *supersets* (*itemsets* gerados a partir dele) também não serão frequentes. Portanto, todo o galho da árvore de busca originado a partir desse itemset pode ser podado, reduzindo exponencialmente o espaço de busca (ALCALÁ-FDEZ *et al.*, 2011).

Por fim, o Apriori Fuzzy utiliza a métrica de confiança — grau que mede a possibilidade do consequente acontecer, dado a ocorrência do antecedente — para selecionar quais *itemsets* frequentes se tornam regras de associação fuzzy. No caso do FARC-HD e FARCI outros passos são realizados após, entretanto, como o foco desse trabalho é especificamente o Apriori Fuzzy, as demais partes não serão detalhadas.

RESULTADOS E/OU DISCUSSÃO (ou Análise e discussão dos resultados)

O código do Apriori Fuzzy modificado foi implementado na linguagem Java 21.0.1, com orientação a objeto. Os *datasets* utilizados foram adquiridos através do repositório de *datasets* da KEEL (KEEL, 2018).

As melhorias inseridas na implementação do Apriori Fuzzy resultante desse trabalho podem ser categorizadas em dois tipos: melhoria de desempenho, aquelas que diminuem o custo computacional do algoritmo; melhorias lógicas, aquelas que melhoram o desempenho não-computacional do algoritmo.

Começando com as melhorias lógicas, como apontado por Alcala-Fdez *et al*. (2011), o uso de métricas dependentes de parâmetros de entrada como o suporte e a confiança favorecem as regras de classes majoritárias e prejudicam as regras de classes minoritárias. Isso acontece pois as regras de classes minoritárias naturalmente tem um suporte menor, já que possuem menos entradas no *dataset*. Sendo assim, definir um suporte mínimo que contemple essas regras se torna impossível, uma vez que um valor mínimo baixo faz com que os *itemsets* das classes minoritárias sejam contemplados,

mas deixa passar muitos *itemsets* das classes majoritárias, levando à explosão do espaço de busca e geração de regras fracas. O mesmo acontece na fase seguinte — a seleção das regras —, pois a métrica Confiança é calculada a partir do valor do suporte, o que gera o mesmo problema de definição de um valor mínimo da fase da poda.

Para resolver a questão da Confiança, foi implementado o uso do Lift, métrica que não depende de parâmetros de entrada e, por isso, não prejudica a geração de regras de classes minoritárias, como demonstrado por Sanz *et al*. (2021). Ainda seguindo as contribuições de Sanz *et al*. (2021), devido ao melhor desempenho lógico em *datasets* desbalanceados, também foi implementado o uso de média geométrica no cálculo do grau de pertinência dos *itemsets*.

Já o problema do Suporte na fase de poda, ocorre porque o cálculo do suporte de classe considera todas as entradas do *dataset*, ficando limitado pela quantidade de exemplos da classe em questão. Assim, mesmo que um dado *itemset* tenha alta ocorrência nos exemplos de uma classe minoritária, devido à escassez desses exemplos no conjunto de dados, o suporte de classe resultará em um valor baixo, levando à poda desse *itemset*. A solução implementada introduz uma ponderação do suporte mínimo baseada na frequência das classes, avaliando o suporte dos itemsets de forma adaptada a classe específica. Essa abordagem permite que o processo de poda elimine itemsets não frequentes de forma mais seletiva, preservando padrões relevantes para classes minoritárias.

Além dessas soluções, também foram implementadas mudanças de personalização das funções de pertinência, sendo elas: definir o formato das funções (triangular ou trapezoidal) e ajustar quantidade de *labels* por atributo, o que permite uma melhor alocação de recurso em atributos chave e melhora a especificidade das regras.

Sobre o desempenho, o Apriori Fuzzy possui um gargalo crônico. Durante a fase da geração e poda dos *itemsets* frequentes, para cada *itemset* único criado, é calculado o suporte (geral) e suporte de classe. Essa ação é extremamente custosa, pois é necessário varrer todo o *dataset* e calcular o grau de pertinência dos valores de cada entrada nas funções de pertinência dos antecedentes que compõem o *itemset*. Como esse cálculo é repetido exaustivamente para cada combinação única de antecedentes (ou seja, para cada *itemset* gerado), em cada classe de saída (consequente), a complexidade do algoritmo escala exponencialmente. O desempenho é agravado não apenas pelo tamanho do *dataset*, mas pelo número de antecedentes (atributos x *labels*), quantidade de classes de saída e quantidade de *supersets* gerados. Além disso, o algoritmo recalcula os mesmos antecedentes em *supersets* superiores, caracterizando um significativo e desnecessário retrabalho.

A solução implementada traz a criação de uma memória *cache*, responsável por guardar os valores de pertinência do antecedente na primeira vez que forem calculados. Assim, qualquer *itemset* que conter aquele antecedente só precisa recuperar os valores previamente calculados, sem necessidade de recalculo. Como a operação de acesso direto ao valor na memória do hashmap (estrutura de dados utilizada) é mais rápida que toda a série de cálculos, conversões e chamadas de função do cálculo do grau de pertinência, o algoritmo ganha desempenho computacional em troca de um uso maior de memória RAM.

O código da implementação do Apriori Fuzzy descrita neste trabalho está disponível em <https://github.com/JFooley/Apriori-Fuzzy> (ALMEIDA, 2025).

CONSIDERAÇÕES FINAIS (ou Conclusão)

Este trabalho apresentou uma reimplementação do algoritmo Apriori Fuzzy com o objetivo de melhorar o desempenho computacional e aprimorar sua eficácia na geração de regras para classes minoritárias em problemas de classificação desbalanceados.

As soluções de desempenho lógico — substituição da Confiança pela métrica *Lift*, o uso da média geométrica para o cálculo do suporte e a introdução de uma ponderação adaptativa do suporte mínimo baseada na frequência das classes — e de desempenho computacional — inserção de uma memória *cache* para salvar os cálculos de pertinência dos antecedentes e evitar retrabalho —, compilam as melhorias propostas por Sanz *et al.* (2021) e adicionam novas soluções que, em teoria, melhoram o desempenho computacional do algoritmo atacando problemas chave já conhecidos do método.

Outras mudanças não implementadas no código também poderiam melhorar o desempenho computacional do algoritmo, como o uso de *threads* para calcular as regras de classes distintas de forma paralela e o uso da teoria dos grandes números para diminuir a quantidade de iterações no cálculo do suporte de *itemsets* de classes majoritárias.

Em suma, a implementação realizada traz um código mais simples e desacoplado da arquitetura FARC-HD, que pode ser reintroduzido nela ou em outras arquiteturas, mas que é mais amigável para mudanças de cunho acadêmico.

REFERÊNCIAS

ALCALA-FDEZ, J.; ALCALA, R.; HERRERA, F. **A Fuzzy Association Rule-Based Classification Model for High-Dimensional Problems With Genetic Rule Selection and Lateral Tuning.** IEEE Transactions on Fuzzy Systems, v. 19, n. 5, p. 857–872, out. 2011.

SANZ, J.; SESMA-SARA, M.; BUSTINCE, H. **A fuzzy association rule-based classifier for imbalanced classification problems.** Information Sciences, v. 577, p. 265–279, 1 out. 2021.

KEEL. KEEL **Dataset Repository - Imbalanced Datasets.** 2018. Disponível em: https://sci2s.ugr.es/keel/imbalanced.php. Acesso em: 05 ago. 2025.

ALMEIDA, J. G. **Apriori-Fuzzy** (Versão 1.0.0). 2025. Disponível em: https://github.com/JFooley/Apriori-Fuzzy Acesso em: 08 set. 2025.