

Universidade Federal do Piauí
Centro de Ciências da Natureza
Programa de Pós-Graduação em Ciência da Computação

Rotulação Automática de Grupos com Árvores de Decisão: Uma Abordagem Comparativa

Manoel Messias Pereira Medeiros

Número de Ordem PPGCC: M001

Teresina-PI, Maio de 2024

Manoel Messias Pereira Medeiros

Rotulação Automática de Grupos com Árvores de Decisão: Uma Abordagem Comparativa

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Sistemas de Computação), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Universidade Federal do Piauí – UFPI

Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação

Orientador: Vinícius Ponte Machado

Teresina-PI

Maio de 2024

FICHA CATALOGRÁFICA
Universidade Federal do Piauí
Sistema de Bibliotecas UFPI - SIBi/UFPI
Biblioteca Setorial do CCN

M488r Medeiros, Manoel Messias Pereira.
Rotulação automática de grupos com árvores de decisão: uma abordagem comparativa / Manoel Messias Pereira Medeiros. -- 2024.
52 f.

Dissertação (Mestrado) - Universidade Federal do Piauí. Centro de Ciências da Natureza. Programa de Pós-Graduação em Ciência da Computação, Picos, 2024.
"Orientador: Prof. Dr. Vinicius Ponte Machado."

1. Rotulação de Grupos. 2. Àrvore de decisão. 3. Algoritmos. 4. Interpretação de agrupamentos. I. Machado, Vinicius Ponte. II. Titulo.

CDD 005.43

Bibliotecária: Caryne Maria da Silva Gomes - CRB3/1461

Manoel Messias Pereira Medeiros

Rotulação Automática de Grupos com Árvores de Decisão: Uma Abordagem Comparativa

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Sistemas de Computação), como parte dos requisitos necessários para a obtenção do Tí- tulo de Mestre em Ciência da Computação.

Trabalho aprovado. Teresina-PI, 06 de maio de 2024:

Documento assinado digitalmente
gov.br
VINICIUS PONTE MACHADO
Data: 10/06/2024 08:51:29-0300
Verifique em https://validar.iti.gov.br

Vinícius Ponte Machado
Orientador

Documento assinado digitalmente
gov.br
RODRIGO DE MELO SOUZA VERAS
Data: 10/06/2024 11:51:44-0300
Verifique em https://validar.iti.gov.br

Rodrigo de Melo Souza Veras
Convidado 1

Documento assinado digitalmente
gov.br
IVAN SARAIVA SILVA
Data: 10/06/2024 12:42:16-0300
Verifique em https://validar.iti.gov.br

Ivan Saraiva Silva
Convidado 2

Documento assinado digitalmente
gov.br
JOAO DALLYSON SOUSA DE ALMEIDA
Data: 11/06/2024 16:43:32-0300
Verifique em https://validar.iti.gov.br

João Dallyson Sousa de Almeida
Convidado Externo

Teresina-PI
Maio de 2024

Aos meus pais Maria do Carmo Pereira Medeiros (in memoriam) e Manoel Messias de Medeiros (in memoriam), aos quais me espelhei através de seus exemplos de vida.

Agradecimentos

Agradeço a primeiramente a Deus por ter me dados coragem a saúde para enfrentar esse desafio.

Agradeço aos meus pais, Maria do Carmo Pereira Medeiros *(in memoriam)* e Manoel Messias de Medeiros *(in memoriam)*

A minha esposa, Mônica Grasielli Pereira da Silva Medeiros, pela compreensão nos momentos de ausência.

Agradeço ao meu orientador, Vinicius Ponte Machado, por todos os conselhos, pela paciência e ajuda nesse período.

Aos meus companheiros de mestrado, em especial ao Jocines Dela Flora Silveira.

Aos professores Laurindo de Sousa Britto Neto e Rodrigo de Melo Souza Veras pelo conhecimentos essenciais repassados nas suas disciplinas.

Aos Meus colegas de Instituto Federal do Piauí - IFPI, Daniel de Sousa Luz e Gilberto Nunes Neto, pelo apoio que me possibilitou terminar este trabalho.

"Se o conhecimento pode criar problemas, não é através da ignorância que podemos solucioná-los"

(Isaac Asimov)

Resumo

O volume exponencial de dados impulsiona o crescimento da análise de dados. O agrupamento de dados é uma técnica importante, mas a interpretação dos grupos pode ser desafiadora. A rotulação automática de grupos com árvores de decisão pode ajudar a resolver este problema. Afim de se validar este modelo de rotulação automática de grupos com uso de árvores de decisão este trabalho apresenta uma comparação entre quatro algoritmos de árvore de decisão no contexto da rotulação automática de grupos. Inicialmente, destaca-se a importância dos modelos de rotulação de grupos como ferramentas para auxiliar especialistas na interpretação de dados, fornecendo uma definição concisa das principais características dos grupos. A escolha do algoritmo de árvore de decisão impacta significativamente na performance do modelo, conforme demonstrado em um estudo comparativo utilizando os algoritmos ID3, C4.5, CART e CHAID. Os resultados revelaram que o algoritmo CHAID obteve os melhores resultados nas bases Íris e Sementes, enquanto o algoritmo C4.5 foi mais eficaz nas bases Vinho e Vidros. As taxas de concordância do algoritmo de melhor desempenho em cada base se aproximaram ou superaram a média encontrada na literatura. A validade do modelo de rotulação automática de grupos foi confirmada, destacando a importância da escolha criteriosa do algoritmo de árvore de decisão.Os resultados deste estudo ressaltam o potencial dos modelos de rotulação automática de grupos e a relevância da seleção adequada do algoritmo de árvore de decisão. Pesquisas futuras são essenciais para aprimorar o modelo e expandir suas aplicações, contribuindo para diversos campos, como descoberta de padrões em bancos de dados, identificação de grupos em estudos de bioinformática e segmentação de mercado-alvo com necessidades semelhantes.

Palavras-chaves: Rotulação Automática de Grupos. Interpretação de Agrupamentos. Árvore de decisão.

Abstract

The exponential volume of data drives the growth of data analytics. Data clustering is an important technique, but interpreting the clusters can be challenging. Automatic group labeling with decision trees can help solve this problem. In order to validate this model of automatic group labeling using decision trees, this work presents a comparison between four decision tree algorithms in the context of automatic group labeling. Initially, the importance of group labeling models is highlighted as tools to assist experts in data interpretation, providing a concise definition of the main characteristics of groups. The choice of decision tree algorithm significantly impacts the model's performance, as demonstrated in a comparative study using the ID3, C4.5, CART and CHAID algorithms. The results revealed that the CHAID algorithm obtained the best results in the Iris and Seeds bases, while the C4.5 algorithm was more effective in the Wine and Glass bases. The agreement rates of the best performing algorithm in each database approached or exceeded the average found in the literature. The validity of the automatic group labeling model was confirmed, highlighting the importance of judicious choice of the decision tree algorithm. The results of this study highlight the potential of automatic group labeling models and the relevance of appropriate selection of the decision tree algorithm. decision. Future research is essential to improve the model and expand its applications, contributing to diverse fields, such as discovering patterns in databases, identifying groups in bioinformatics studies, and segmenting target markets with similar needs.

Keywords: Automatic labeling. Cluster interpretation. Decision trees.

Lista de ilustrações

Figura 1 – Passos de execução do algoritmo *k-means*. 9
Figura 2 – Um exemplo de árvore de decisão adaptada de (MITCHELL, 1997). . . 11
Figura 3 – Modelo proposto em (LOPES; MACHADO; RABÊLO, 2013). 21
Figura 4 – Fluxograma do modelo proposto. 28
Figura 5 – Árvore induzida com a base *Íris* a partir de seus *clusters* com o algoritmo ID3 30
Figura 6 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo C4.5 30
Figura 7 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo CART 31
Figura 8 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo CHAID 31
Figura 9 – Subárvore selecionada para o *cluster* zero. 32
Figura 10 – Subárvore selecionada para o *cluster* um. 32
Figura 11 – Subárvore selecionada para o *cluster* dois. 33

Lista de tabelas

Tabela 1 – Tabela comparativa entre os algoritmos de agrupamento interpretáveis. 20
Tabela 2 – Tabela com análise dos modelos de rotulação da literatura. 25
Tabela 3 – Distribuição de amostras por grupo da base *Íris*. 29
Tabela 4 – Tabela com as métricas utilizadas por cada algoritmo de árvore de decisão 33
Tabela 5 – Análise de rotulação para a base de dados *Íris* com o algoritmo ID3. . 36
Tabela 6 – Análise de rotulação para a base de dados *Íris* com o algoritmo C4.5. . 36
Tabela 7 – Análise de rotulação para a base de dados *Íris* com o algoritmo CART. 37
Tabela 8 – Análise de rotulação para a base de dados *Íris* com o algoritmo CHAID. 37
Tabela 9 – Análise de rotulação para base de dados *Íris* presente em (LOPES et al., 2016). 37
Tabela 10 – Análise de rotulação para base de dados Sementes utilizando o algoritmo ID3 39
Tabela 11 – Análise de rotulação para base de dados Sementes utilizando o algoritmo C4.5 39
Tabela 12 – Análise de rotulação para base de dados Sementes utilizando o algoritmo CART 39
Tabela 13 – Análise de rotulação para base de dados Sementes utilizando o algoritmo CHAID 39
Tabela 14 – Análise de rotulação para base de dados Sementes presente em (LOPES et al., 2016). 40
Tabela 15 – Análise de rotulação para base de dados Vinho com uso do algoritmo ID3 41
Tabela 16 – Análise de rotulação para base de dados Vinho com uso do algoritmo C4.5 41
Tabela 17 – Análise de rotulação para base de dados Vinho com uso do algoritmo CART 41
Tabela 18 – Análise de rotulação para base de dados Vinho com uso do algoritmo CHAID 41
Tabela 19 – Análise de rotulação para base de dados Vinhos presente em Silva et al. (2021) 42
Tabela 20 – Análise de rotulação para base de dados Vidros com uso do algoritmo ID3 43
Tabela 21 – Análise de rotulação para base de dados Vidros com uso do algoritmo C4.5 43
Tabela 22 – Análise de rotulação para base de dados Vidros com uso do algoritmo CART 44
Tabela 23 – Análise de rotulação para base de dados Vidros com uso do algoritmo CHAID 44

Tabela 24 – Análise de rotulação para base de dados Vidros encontrado em Lopes, Machado e Rabêlo (2013) 45
Tabela 25 – Comparativo entre taxa de concordância média entre as Abordagens de Rotulação 45

Lista de abreviaturas e siglas

CA	Coeficiente de Assimetria
CP	Comprimento da Pétala
CS	Comprimento da Sépala
CSS	Comprimento do Sulco da Semente
CSM	Comprimento da Semente
EWD	Discretização por Larguras Iguais
EFD	Discretização por Frequências Iguais
FHS	*Framingham Heart Study*
FCPS	*Fundamental Clustering Problems Suite*
IA	Inteligência Artificial
ICOT	*Clustering via Optimal Trees*
LS	Largura da Sépala
LP	Largura da Pétala
MIO	*Mixed Integer Optimization*
ML	*Machine Learning*
MLP	*Multilayer Perceptron*
MRA	Modelo de Rotulação Automática
MSA	Medida de Seleção de Atributos
OCT	*Optimal Classification Trees*
RNA	Rede Neural Artificial
UCI	*University of California Irvine*

Sumário

1 INTRODUÇÃO 1

Introdução 1
1.1 Motivação 2
1.2 Objetivos 3

2 REFERENCIAL TEÓRICO 4
2.1 Aprendizagem de Máquina 4
2.1.1 Aprendizagem Supervisionada 5
2.1.2 Aprendizagem não Supervisionada 5
2.2 Clustering 5
2.2.1 *k-means* 8
2.2.2 Rotulação de grupos 9
2.3 Árvores de decisão 10
2.3.1 Medidas de seleção de atributos 12
2.3.2 Algoritmo ID3 14
2.3.3 Algoritmo C 4.5 14
2.3.4 Algoritmo CART 15
2.3.5 Algoritmo CHAID 16

3 TRABALHOS RELACIONADOS 17

4 MODELO PROPOSTO 27
4.1 Modelos de conjuntos de árvores versus modelos simples de árvores de decisão 27
4.2 O Modelo 27
4.2.1 Passo I - Gerando *clusters* 29
4.2.2 Passo II - Geração do modelo de árvore de decisão 29
4.2.3 Passo III - Seleção de ramos com melhor classificação 29
4.2.4 Passo IV - Seleção de atributos e atribuição de faixas de valores 32
4.2.4.1 Processo de seleção de faixa de valores para os rótulos do *cluster* zero. 33
4.2.4.2 Processo de seleção faixa de valores para os rótulos do *cluster* um. 34
4.2.4.3 Processo de seleção faixa de valores para os rótulos do *cluster* dois. 34
4.2.5 Resultados de rótulos para Base *Íris* 35

5 RESULTADOS 38
5.1 Detalhes de Implementação 38

5.2 Base de dados Sementes 38
5.3 Base de dados Vinhos 40
5.4 Base de dados Vidros 42
5.5 Comparação entre os modelos 44

6 CONCLUSÕES E TRABALHOS FUTUROS 47
6.1 Conclusões 47

REFERÊNCIAS 49

1 Introdução

O volume exponencial de dados gerados por diversas fontes, como redes de sensores, transações comerciais e redes sociais, impulsiona o crescimento vertiginoso da ciência da análise de dados. Nas últimas décadas, esse crescimento se tornou ainda mais notável, acompanhando a rápida expansão da indústria de tecnologia e a crescente aplicabilidade de técnicas computacionais, aliadas a avanços em ferramentas analíticas, que são softwares ou plataformas que te auxiliam a coletar, processar, analisar e visualizar dados de diversas fontes (DIMOTIKALIS et al., 2021).

Assim, o uso de algoritmos para agrupamento e posterior análise por especialistas se torna cada vez mais relevante, atraindo a atenção de diversos estudos na área de Aprendizagem Não Supervisionada, uma subárea do Aprendizado de Máquina (AM). A formação não supervisionada de agrupamentos se configura como um problema de classificação de objetos em diferentes categorias, sem a pré-definição das saídas das categorias (RUSSELL; NORVIG, 2016). O *clustering* divide um conjunto de dados em subconjuntos menores, denominados *clusters*, que reúnem objetos com características em comum (LOPES et al., 2016).

Segundo Han, Kamber e Pei (2011b) as aplicações do *Clustering* são diversas e abrangem áreas como:

- *business intelligence*: auxiliar na segmentação de clientes, análise de mercado e otimização de campanhas.
- reconhecimento de padrões em imagens: identificar objetos em fotos, classificar imagens médicas e segmentar regiões em imagens de satélite.
- busca na web: personalizar os resultados de pesquisa e recomendar conteúdo relevante para os usuários.
- biologia: auxiliar na classificação de genes e proteínas, na identificação de novas espécies e na análise de dados genômicos.
- segurança da informação: detectar anomalias e fraudes, identificar comportamentos suspeitos e proteger sistemas contra ataques cibernéticos.

A similaridade e dissimilaridade são os componentes centrais dos algoritmos de agrupamento usados para agrupar pontos de dados nos mesmos agrupamentos, enquanto ponto de dados diferentes em agrupamentos distintos (SHIRKHORSHIDI; AGHABOZORGI;

WAH, 2015). A métrica de similaridade mede o quanto duas instâncias são parecidas e a dissimilaridade mede o quanto dois objetos são diferentes.

De acordo com Han, Kamber e Pei (2011b), outros fatores devem ser levados em conta para uma boa interpretação dos grupos, tais como:

- escalabilidade;
- habilidade de lidar com diferentes tipos de atributos;
- descoberta de *clusters* com forma arbitrária;
- requisitos de conhecimento de domínio para determinar os parâmetros de entrada;
- capacidade de lidar com dados ruidosos;
- novos objetos e sensibilidade a ordem;
- capacidade de lidar com alta dimensionalidade;
- *clustering* baseado em restrições;
- interpretação e usabilidade.

1.1 Motivação

A interpretabilidade dos grupos é um aspecto abordado por meio do processo de rotulação. Uma compreensão precisa dos grupos formados conduzirá a uma melhor apreensão dos dados agrupados. O extenso emprego dos métodos de agrupamento também tem suscitado interesse na interpretabilidade para uma compreensão acurada dos grupos gerados, sendo a rotulação dos grupos uma via para interpretá-los.

Segundo Lopes et al. (2016) o problema da rotulação é definido como uma identificação resumida para os grupos, nomeando-os de acordo com suas características. O processo de rotulação de grupos pretende identificar de forma única um grupo através da tupla atributo e faixa de valores. Diversas técnicas foram propostas em Lopes, Machado e Rabelo (2014), Machado, Ribeiro e RABêLO (2015), Filho et al. (2020), Moura, Veras e Machado (2022), Silva et al. (2021) e Alencar (2022) e este trabalho apresenta uma nova abordagem com o uso de árvore de decisão.

As árvores de decisão representam as decisões em termos de regras. A escolha do caminho em cada nó é determinada pelos valores dos atributos, seguindo um critério de divisão consistente com o objetivo da classificação. Os nós folha representam classes e são interpretados como regras, por exemplo: "Se A_1 e A_2 e A_3, então classe C_1". Dessa forma, cada classe pode ser descrita por um conjunto de regras(BASAK; KRISHNAPURAM,

2005a). Considerando que as árvores de decisão geram regras como saída, as quais podem ser facilmente analisadas e utilizadas para atribuir rótulos, optou-se por este algoritmo de aprendizado supervisionado no modelo proposto.

1.2 Objetivos

O objetivo principal desse trabalho é estabelecer um método automático de rotulação de *clusters* com o uso de árvores de decisão auxiliando assim na interpretação destes. Para alcançar o objetivo principal, tentou-se alcançar os seguintes objetivos específicos:

- validar o modelo de rotulagem automática de grupos, comprovando sua efetividade em diferentes conjuntos de dados.;
- comparar algoritmos de arvores de decisão para identificar o que gere melhores rótulos;
- analisar os melhores resultados alcançados por determinado algoritmo de árvore de decisão.

Estrutura do Trabalho

Este trabalho está organizado em uma Introdução e 5 capítulos subsequentes. Nesta Introdução os principais aspectos do trabalho foram apresentados, com destaque ao contexto e motivação, definição do problema e objetivos. O Capítulo 2 discorre sobre conceitos, técnicas e algoritmos que são fundamentais ao desenvolvimento desta pesquisa, como Aprendizagem Supervisionada e não supervisionada, *Clustering*, *K-means*, Árvores de Decisão, Ganho de Informação e Entropia. No capitulo 3 são demonstrados os trabalhos relacionados. No Capítulo 4 é apresentada a abordagem do problema e métodos aplicados para alcançar os objetivos definidos. O Capítulo 5 é uma apresentação dos resultados alcançados e finalmente no capítulo 6 são apresentadas as conclusões e trabalhos futuros.

2 Referencial Teórico

Este capítulo apresenta os fundamentos teóricos que sustentam a pesquisa proposta, com o intuito de facilitar sua compreensão. Serão abordados os seguintes tópicos:

- Aprendizado de Máquina (AM): uma introdução ao campo da AM, com foco em aprendizado supervisionado e não supervisionado.
- *Clustering*: conceito de clustering e sua importância na análise de dados.
- Método *K-means*: um algoritmo de clustering popular e sua aplicação prática.
- Árvores de Decisão: Descrição de diferentes algoritmos de árvores de decisão e suas características.
- Ganho de Informação e Entropia: conceitos de teoria da informação relevantes para a seleção de atributos em árvores de decisão.

A compreensão dos conceitos mencionados acima é fundamental para o desenvolvimento do método de rotulação automática de grupos apresentado nesta pesquisa.

Este trabalho propõe um modelo de rotulação automática de grupos com o uso de algoritmos de árvores de decisão a partir de dados agrupados por um algoritmo de aprendizado não supervisionado (*K-means*).

2.1 Aprendizagem de Máquina

O Aprendizado de Máquina (AM) é um ramo da Inteligência Artificial (IA) que desenvolve algoritmos que aprendem com a experiência. Dessa forma, computadores podem auxiliar humanos na análise de conjuntos de dados cada vez mais extensos e complexos (LIBBRECHT; NOBLE, 2015).

Segundo Mitchell (1997), o AM se concentra no desenvolvimento de técnicas computacionais que permitem que sistemas aprendam autonomamente a realizar uma tarefa específica por meio da experiência. Ao contrário das metodologias tradicionais, o AM busca que a própria máquina encontre, após o processo de aprendizado, uma hipótese que melhor explique o problema em questão.

Russell e Norvig (2016) identificam três tipos principais de aprendizagem: supervisionada, não supervisionada e por reforço. Haykin (1999) destaca que a escolha da abordagem de aprendizado determina como o algoritmo de AM se comporta em relação

ao seu ambiente, ou seja, define como se dará o seu aprendizado a partir de um conjunto de dados.

2.1.1 Aprendizagem Supervisionada

Aprendizagem supervisionada refere-se a um tipo de aprendizado de máquina em que os dados usados para treinar o algoritmo incluem as soluções desejadas. O termo supervisionada deriva da analogia com a presença de um supervisor externo que fornece a resposta correta para cada exemplo de treinamento (FACELI et al., 2021).

De acordo com Barber (2012) o conceito formal de Aprendizagem Supervisionada é a seguinte: dado um conjunto de dados $D = \{(x^n, y^n),\ n=1,\ \ldots\ ,\ N\}$, a tarefa é criar um modelo capaz de aprender a relação existente entre os valores de entrada x e os valores de saída y, de modo tal que quando uma nova entrada x^* é fornecida, a predição de sua saída y^* é precisa.

2.1.2 Aprendizagem não Supervisionada

A Aprendizagem não supervisionada envolve um tipo de aprendizado de máquina em que não há saída conhecida disponível e nenhum ”supervisor” é capaz de instruir o algoritmo de aprendizado. Nesse tipo de aprendizado, o algoritmo recebe apenas os dados de treinamento como entrada (GÉRON, 2021). A tarefa mais comum de aprendizagem não supervisionada é o agrupamento, cujo problema consiste em agrupar diversas amostras em classes distintas de modo que o grau de similaridade entre elementos de um mesmo grupo seja o máximo possível e ainda que o grau de dissimilaridade entre elementos de grupos distintos também seja o máximo possível (ATILGAN; NASIBOV, 2016)

2.2 Clustering

Clustering é o processo de dividir conjunto de objetos de dados em subconjuntos. Onde cada subconjunto é um *cluster*, de tal forma que os componentes em um *cluster* são semelhantes entre si, porém diferentes em outros *clusters* (HAN; KAMBER; PEI, 2011b). Segundo Aggarwal e Reddy (2014) como o *Clustering* lida com ausência de informações cujas as saídas são conhecidas é considerado como sendo uma técnica de aprendizagem não supervisionada, o agrupamento pode ser considerado um modelo conciso dos dados que podem ser interpretados no sentido de um resumo ou um modelo generativo. Ainda segundo Aggarwal e Reddy (2014) alguns do domínios de aplicações comuns nos quais o problema de *clustering* surge são:

- etapa intermediária para outros problemas fundamentais de mineração de dados - como um agrupamento pode ser considerada uma forma de resumo de dados, muitas

vezes serve como uma etapa intermediária chave para muitos problemas fundamentais de mineração de dados, como classificação ou análise de *outliers*;

- filtragem colaborativa - nos métodos de filtragem colaborativa, o agrupamento fornece um resumo de usuários com interesses semelhantes. As avaliações fornecidas pelos diferentes usuários uns para os outros são usado para realizar a filtragem colaborativa. Isso pode ser usado para fornecer recomendações em uma variedade de aplicações;

- segmentação de clientes - esta aplicação é bastante semelhante à filtragem colaborativa, pois cria grupos de clientes semelhantes nos dados. A principal diferença da colaboração filtragem é que, em vez de usar informações de classificação, atributos arbitrários sobre os objetos podem ser usado para fins de agrupamento;

- sumarização de dados - muitos métodos de agrupamento estão intimamente relacionados aos métodos de redução de dimensionalidade. Tais métodos podem ser considerados uma forma de sumarização de dados. A sumarização de dados pode ser útil na criação de representações de dados compactos, que são mais fáceis de processar e interpretar em uma ampla variedade de aplicações;

- detecção dinâmica de tendências - muitas formas de algoritmos dinâmicos e de *streaming* podem ser usadas para realizar a detecção de tendências em uma ampla variedade de aplicativos de redes sociais. Nesses aplicativos, os dados são agrupados dinamicamente em um fluxo contínuo e podem ser usados para determinar padrões importantes de alterações. Exemplos de tais dados de *streaming* podem ser dados multidimensionais, fluxos de texto, dados de séries temporais de *streaming* e dados de trajetória. As principais tendências e eventos nos dados podem ser descobertos com o uso de métodos de agrupamento;

- análise de dados multimídia - uma variedade de diferentes tipos de documentos, como imagens, áudio ou vídeo, enquadram-se na categoria geral de dados multimídia. A determinação de segmentos semelhantes tem inúmeras aplicações, como a determinação de trechos de música semelhantes ou fotografias semelhantes. Em muitos casos, os dados podem ser multimodais e podem conter diferentes tipos. Nesses casos, o problema se torna ainda mais desafiador;

- análise de dados biológicos - os dados biológicos tornaram-se difundidos nos últimos anos, devido ao sucesso do esforço do genoma humano e à crescente capacidade de coletar diferentes tipos de dados de expressão genética. Os dados biológicos são geralmente estruturados como sequências ou como redes. Os algoritmos de agrupamento fornecem boas ideias das principais tendências nos dados, bem como das sequências incomuns;

- análise de rede social - nessas aplicações, a estrutura de uma rede social é usada para determinar as comunidades importantes na rede subjacente. A detecção de comunidade tem aplicações importantes na análise de redes sociais, pois fornece uma compreensão importante da estrutura da comunidade na rede. O *clustering* possui também aplicações para sumarização de redes sociais, o que é útil em várias aplicações.

De acordo com Han, Kamber e Pei (2011b) os principais métodos de agrupamentos podem ser classificados nas seguintes categorias:

- métodos de particionamento - dado um conjunto de n objetos, um método de particionamento constrói k partições dos dados, onde cada partição representa um *cluster* e $k \leq n$. Ou seja, divide os dados em k grupos de forma que cada grupo contenha pelo menos um objeto. A maioria dos métodos de particionamento é baseada em distância;
- métodos hierárquicos - um método hierárquico cria uma decomposição hierárquica de um determinado conjunto de objetos de dados. Um método hierárquico pode ser classificado como aglomerativo ou divisivo. Os métodos de agrupamento hierárquicos podem ser baseados em distância ou baseados em densidade e continuidade. Os métodos hierárquicos sofrem com o fato de que uma vez que uma etapa (fusão ou divisão) é concluída, isso nunca pode ser desfeito;
- métodos baseados em densidade - a maioria dos métodos de particionamento agrupam objetos com base na distância entre eles. Tais métodos podem encontrar apenas *clusters* de forma esférica e encontram dificuldade em descobrir *clusters* de formas arbitrárias. Outros métodos de agrupamento foram desenvolvidos com base na noção de densidade. A ideia geral deles é continuar crescendo um determinado *cluster* enquanto a densidade (número de objetos ou pontos de dados) na "vizinhança" exceder algum limite.;
- métodos baseados em grade - os métodos baseados em grade quantificam o espaço do objeto em um número finito de células que formam uma estrutura de grade. Todas as operações de agrupamento são executadas na estrutura da grade (isto é, no espaço quantizado).

Os algoritmos de agrupamento podem incorporar conceitos de diversos métodos, dificultando sua classificação em apenas uma categoria específica. Algumas aplicações podem ainda necessitar da junção de várias técnicas de agrupamento.(HAN; KAMBER; PEI, 2011b).

No presente trabalho primeiramente agrupa-se os dados com o algoritmo *K-means* então é escolhe-se um número de *cluster* K, em seguida atribui-se um determinado *cluster* a cada amostra e então como uma tarefa de aprendizagem supervisionada treina-se um modelo de Árvore de decisão. Por fim, analisa-se a saída da Árvore de Decisão para gerar os rótulos de cada *cluster*.

2.2.1 *k-means*

O *K-means* é um método de particionamento que foi apresentado inicialmente por MacQueen (1967), e é usado para particionar dados em k grupos mutualmente exclusivos. O *k-means* trabalha com base em valores que calculam o grau de similaridade ao contrário de métodos hierárquicos que operam com medidas de dissimilaridade (LOPES et al., 2016). O nome vem da representação de cada um dos k *clusters* C_{j} pela média (ou média ponderada) c_{i} de seus pontos, o chamado centroide. Embora isso obviamente não funcione bem com atributos categóricos, tem bom senso geométrico e estatístico para atributos numéricos (BERKHIN, 2006).

De acordo com Arthur, Manthey e Röglin (2011) a entrada para o método *k-means* é um conjunto $X \subseteq R^d$ de n de dados. O algoritmo produz k centroides $c_1, ..., c_k \in R^d$ e uma partição de X em k *clusters* $C_1, ..., C_k$. O método *k-means* opera da seguinte forma:

(1) Seleciona os centroides do cluster $c_1, ..., c_k \in R^d$ arbitrariamente.
(2) Atribui todo $x \in X$ ao *cluster* C_i cujo centroide c_i está mais próximo dele, ou seja, $\| x - c_i \| \leq \| x - c_j \|$ para todo $j \neq i$.
(3) Definir $c_i = \frac{1}{|c_i|} \Sigma_{x \in c_i} x$.
(4) Se os *clusters* ou centroides tiverem mudado, vá para 2. Caso contrário, finalize.

Para k = 3 (divisão em três grupos), três objetos são selecionados aleatoriamente como centroides iniciais

Para $k = 3$(divisão em três grupos), três objetos são selecionados aleatoriamente como centroides iniciais. Cada objeto é então designado a um *cluster* com base na proximidade ao centroide do *cluster* correspondente. Essa atribuição inicial gera os *clusters* iniciais, como ilustrado na Figura 1(c). Posteriormente, os centroides dos *clusters* são atualizados, ou seja, o valor médio de cada cluster é recalculado considerando os objetos atuais contidos no *cluster*. Utilizando os novos centroides dos *clusters*, os objetos são redistribuídos para os *clusters* com base na proximidade ao centroide correspondente. Essa redistribuição gera novos grupos, conforme demonstrado na Figura 1(d). Esse processo é iterativo, levando à progressão ilustrada na Figura 1(e). A iteração continua através do processo de realocação iterativa, no qual os objetos são reatribuídos iterativamente aos *clusters* correspondentes, a fim de aprimorar o particionamento. Eventualmente, quando não ocorre mais nenhuma reatribuição dos objetos em qualquer *cluster*, o processo é

concluído. Os *clusters* resultantes são então retornados pelo processo de agrupamento.

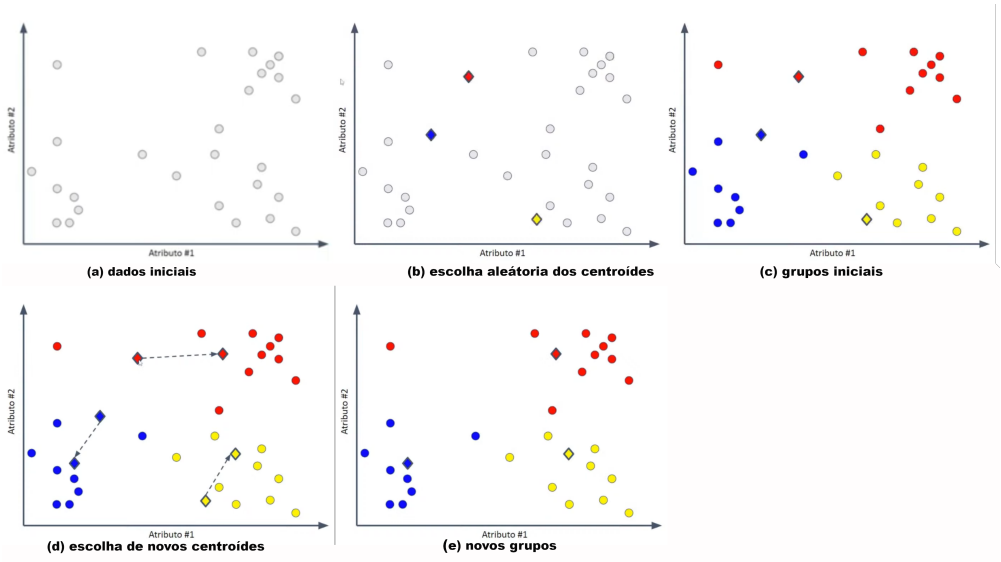


Figura 1 – Passos de execução do algoritmo *k-means*.

A complexidade de tempo do algoritmo *k-means* é $O(nkt)$, onde n é o número total de objetos, k é o número de *cluster* e t é número de iterações. Normalmente, $k \ll n$ e $t \ll n$. O método relativamente escalável e eficiente no processamento de grandes conjuntos de dados (HAN; KAMBER; PEI, 2011b). Dada essas características e sua eficiência levaram a escolha deste algoritmo como método de agrupamento deste trabalho.

2.2.2 Rotulação de grupos

O processo de agrupamento pode ser utilizado em diversas aplicações. A correta compreensão dos grupos criados levará a um melhor entendimento sobre os dados agrupados. Métodos de aprendizagem por agrupamento têm se tornado cada vez mais relevantes para a descoberta de conhecimento em grandes bases de dados. Tem havido um crescente interesse em métodos que auxiliem na compreensão desses grupos. A presente pesquisa apresenta um método comparativo de rotulação automática de grupos com a utilização de árvores de decisão. Segundo Lopes et al. (2016), o problema da rotulação é definido como uma identificação resumida para os grupos, nomeando-os de acordo com suas características. Abaixo temos definição formal do problema de Rotulação.

Problema de Rotulação: Dado um conjunto de grupos $C = \{c_1, ..., c_k | k \geq 1\}$ de modo que cada grupo contém um conjunto de elementos $\{c_i = e_1, .., e_{n(ci)}\ n^{(ci)} \geq 1\}$ que podem ser representados por um vetor de atributos definidos em $\mathbb{R}^{\mathrm{m}}$ e expresso por $e^{(ci)} = (a_1, ..., a_m)$, e que $c_i \cap c_{\mathrm{i'}} = \{\emptyset\}$ com $1 \leq i, i \leq k$ e $i \neq i'$; o objetivo consiste em apresentar um conjunto de rótulos $R = \{r_{\mathrm{c1}}, ..., r_{\mathrm{ck}}\}$ no qual cada rótulo específico é dado por um conjunto de pares de valores, o atributo e seu respectivo intervalo,

$r_{\mathrm{ci}} = \{(a_1, [p_1, q_1]), ..., (a_{\mathrm{m(ci)}},]p_{\mathrm{m(ci)}}, q_{\mathrm{m(ci)}}])\}$, o que é capaz de melhor expressar o grupo $\mathrm{c_i}$ associado.

Onde:

- K é o número de grupos;
- $\mathrm{c_i}$ é um grupo qualquer;
- $\mathrm{n^{(ci)}}$ é o número de elementos do grupo $\mathrm{c_i}$;
- $e_{\mathrm{j}}^{\mathrm{(ci)}}$ se refere ao j-ésimo elemento pertencente ao grupo $\mathrm{c_i}$;
- m é a dimensão do problema;
- $\mathrm{r_{ci}}$ é o rótulo referenciado ao cluster $\mathrm{c_i}$
- $]p_{\mathrm{m(ci)}}, q_{\mathrm{m(ci)}}]$[1] representa o intervalo de valores do atributo $a_{\mathrm{m(ci)}}$ onde $p_{\mathrm{m(ci)}}$ é o limite inferior e $q_{\mathrm{m(ci)}}$ é o limite superior;
- m^{ci} é a quantidade de atributos presente em um rótulo referente ao cluster c_{i}.

2.3 Árvores de decisão

Segundo Mitchell (1997) uma árvore de decisão é uma estrutura semelhante a um fluxograma em que cada nó interno representa um recurso ou atributo, cada ramificação representa uma regra de decisão e cada nó folha representa o resultado ou rótulo de classe. A árvore é construída recursivamente, selecionando o melhor atributo para dividir os dados em cada nó com base em determinados critérios. Esse processo continua até que uma condição de parada seja atendida, como atingir uma profundidade máxima ou ter um número mínimo de amostras em um nó folha.

Uma árvore de decisão implementa a classificação de uma determinada amostra de dados através de vários níveis de decisão para nos ajudar a chegar a uma decisão final. Essa sequência de as decisões são demonstradas em uma estrutura de árvore. A árvore de decisão é uma técnica supervisionada, que pode ser usada tanto em problemas de classificação como em problemas de regressão, no entanto sendo mais comum em problemas de classificação (GÉRON, 2022).

Árvores de decisão são ferramentas amplamente utilizadas para classificação em diversos domínios, como mineração de texto, processamento de fala, bioinformática e inteligência na *Web* e muitos outros campos que lidam com grandes conjuntos de dados (BASAK; KRISHNAPURAM, 2005a). Talvez, a característica mais importante delas seja sua capacidade de quebrar um complexo processo de tomada de decisão em uma coleção

de decisões mais simples, fornecendo assim uma solução que muitas vezes é mais fácil de interpretar (SAFAVIAN; LANDGREBE, 1991).

A Figura 2 ilustra uma árvore de decisão típica. Esta árvore de decisão classifica os dias de acordo com a sua adequação ou não para a prática de tênis.

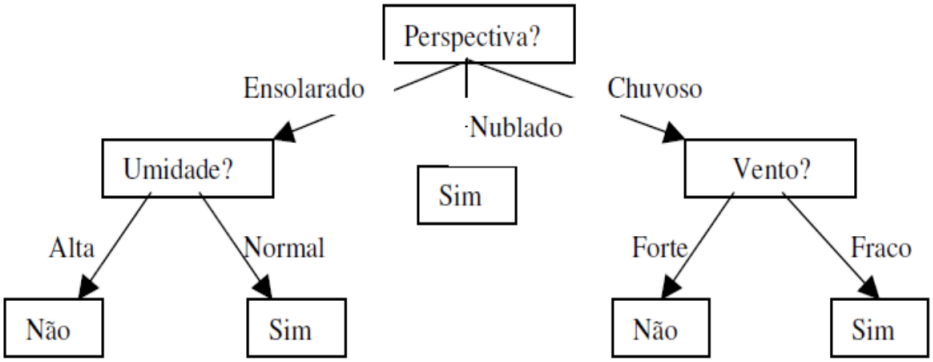


Figura 2 – Um exemplo de árvore de decisão adaptada de (MITCHELL, 1997).

Em uma Árvore de Decisão tem-se:

- **Nó-Raiz** - nó que inicia a árvore. A partir dele é divido em dois ou mais conjunto de dados.
- **Nó-Folha** - nó final não pode mais haver divisão após este nó.
- **Divisão** - é o a divisão do nó de decisão/nó-raiz em sub-nós de acordo com as condições geradas como regras para árvore.
- **Poda** - é quando precisamos remover ramos da árvore para ajustar seu desempenho. É considerada a parte mais importante do processo de construção da árvore, principalmente em dados ruidosos.
- **Nó pai/filho** - todos os demais nós são filhos do nó-raiz.

De acordo com Faceli et al. (2021), as árvores de decisão apresentam os seguintes pontos positivos e negativos:

Aspectos Positivos:

1. flexibilidade- as árvores de decisão não fazem suposições sobre a distribuição dos dados e são métodos não paramétricos. Elas podem se ajustar a diferentes modelos em subespaços do espaço de instâncias.

2. robustez - as árvores de decisão são invariantes a transformações monótonas de variáveis de entrada, o que as torna robustas em relação a distribuições com grande cauda e outliers.

3. seleção de atributos: O processo de construção de uma árvore de decisão seleciona os atributos mais relevantes, tornando o modelo robusto contra a adição de atributos irrelevantes ou redundantes.

4. interpretabilidade - as árvores de decisão permitem a aproximação de decisões complexas por meio de decisões mais simples e locais. Isso contribui para a sua popularidade, já que as decisões são baseadas nos valores dos atributos usados para descrever o problema.

5. eficiência - o algoritmo de aprendizado de árvore de decisão é eficiente, com uma complexidade de tempo linear em relação ao número de exemplos.

Aspectos Negativos:

1. replicação - a replicação ocorre quando uma sequência de testes é duplicada em diferentes ramos da árvore, resultando em representações menos concisas e baixa acurácia preditiva.

2. valores ausentes - a presença de valores ausentes em um atributo causa problemas na decisão de qual ramo seguir na árvore. Algoritmos precisam lidar com mecanismos especiais para tratar essa falta de valores.

3. atributos contínuos - a presença de atributos contínuos pode ser um gargalo para o algoritmo, exigindo operações de ordenação e afetando o tempo de indução da árvore.

4. instabilidade - pequenas variações no conjunto de treinamento podem levar a grandes variações na árvore final, tornando a árvore instável.

2.3.1 Medidas de seleção de atributos

Nas árvores de decisão existem diversas Medidas de Seleção de Atributos (MSA), que medem a relevância dos atributos para o processo de criação da árvore. Seguem algumas citadas por Han, Kamber e Pei (2011a):

1. **Ganho de informação e Entropia**: Um dos mais usados critérios de seleção de atributos é o ganho de informação, este critério é utilizado pelo algoritmo ID3 (QUINLAN, 1986) e C4.5 (QUINLAN, 1993). O ganho de informação se utiliza da entropia como métrica de impureza, o algoritmo ID3 foi pioneiro no uso da entropia

como métrica de impureza. Quanto maior o ganho de informação, mais relevante é o atributo. A entropia é originada da teoria da informação idealizada por Claude Shannon e Warren Weaver 1949. O ganho de informação é demonstrando na equação 2.1.

$$ganho_de_informação = entropia(pai) - \sum_{j=1}^{n} \left[\frac{N(v_j)}{N}\right] entropia(v_j) \quad (2.1)$$

Onde:
n o número de nó filhos;

N o número total de objetos do nó-pai;

$N(v-j)$ *é número associados ao nó filho* v_j.

O grau de entropia é demonstrado na equação 2.2.

$$entropia(\text{nó}) = -\sum_{i=1}^{c} p(\frac{i}{\text{nó}}).\log_2\left[p(\tfrac{i}{\text{nó}})\right] \quad (2.2)$$

Onde:
$p(i/\text{nó})$ é a fração dos registros pertencentes à classe i no *nó*;
c é o número de classes.

2. **Indice Gini (Gini Index):** O Índice Gini é a medida padrão utilizada no algoritmo CART (Classification and Regression Trees), usado para avaliar a impureza de partições ou tuplas de treinamento. O Índice Gini faz o calculo de impureza dos dados levando em conta as probabilidades de tuplas pertencerem a diferentes classes. Ele faz o calculo para cada atributo e divisão, e o atributo com um índice Gini mais baixo indica maior relevância do atributo.

3. **Razão de Ganho (Gain Ratio)**: A medida de ganho de informação é tendenciosa para testes com muitos resultados. Ou seja, prefere selecionar atributos com um grande número de valores. C4.5, um sucessor do ID3, usa uma extensão para ganho de informação conhecida como taxa de ganho, que tenta superar esse viés. Ele aplica um tipo de normalização ao ganho de informação usando um valor de "informação dividida" definido analogamente com *Info*(D) como é demonstrado na equação 2.3

$$SplitInfo_A(D) = \sum_{j=1}^{v} \frac{|D_J|}{|D|}.\log_2(\frac{|D_J|}{|D|}) \quad (2.3)$$

Este valor representa a informação potencial gerada pela divisão do conjunto de dados de treinamento, D, em v partições, correspondendo ao v resultados de um teste no atributo A. Observe que, para cada resultado, ele considera o número de

tuplas com esse resultado em relação ao número total de tuplas em D. Difere do ganho de informação, que mede as informações com relação à classificação que são adquiridas com base no mesmo particionamento. A taxa de ganho é definida na equação 2.4.

$$GanhoRatio(A) = \frac{Gain(A)}{SplitInfo(A)} \tag{2.4}$$

2.3.2 Algoritmo ID3

O algoritmo de árvore de decisão ID3 foi apresentado por Quinlan (1986). Segundo Singh e Gupta (2014) o algoritmo ID3 é considerado um algoritmo de árvore de decisão muito simples, pois usa uma abordagem gulosa para construir árvores. O ID3 funciona de forma iterativa, dividindo o conjunto de dados em subconjuntos menores até que cada subconjunto seja puro, ou seja, contenha apenas exemplos de uma única classe. Para realizar essa divisão, o algoritmo utiliza a seguinte estratégia:

1. seleção do Atributo: O ID3 seleciona o atributo do conjunto de dados que apresenta a maior ganho de informação. O ganho de informação mede a quantidade de informação que um atributo fornece sobre a classe alvo. O atributo com maior ganho de informação é escolhido como o nó raiz da árvore de decisão.

2. criação de Ramos: O algoritmo cria um ramo na árvore de decisão para cada valor possível do atributo selecionado. Cada ramo representa um subconjunto do conjunto de dados original.

3. repetição: O processo de seleção de atributo, criação de ramos e repetição é realizado recursivamente para cada subconjunto do conjunto de dados até que todos os subconjuntos sejam puros ou não seja possível realizar mais divisões.

2.3.3 Algoritmo C 4.5

O algoritmo C4.5 é uma melhoria do algoritmo ID3, desenvolvido por Quinlan (1993). É baseado no algoritmo de Hunt.

O C4.5 aceita atributos categóricos e contínuos, possui poda de árvore aprimorada para reduzir erros de classificação causados por ruído ou detalhes no conjunto de treinamento.

Como o algoritmo ID3, os dados são classificados em cada nó da árvore para determinar o melhor atributo de divisão. Ele usa o método de impureza da taxa de ganho para avaliar o atributo de divisão (PRIYAM et al., 2013).

Segundo Hajjej et al. (2022) o algoritmo C4.5 funciona dividindo os dados em partições menores, buscando o maior ganho de informação em cada nó. Esse processo é

feito de forma recursiva até que se chegue à árvore de decisão final. A etapas do algoritmo C4.5 são:

1. cálculo do ganho de informação: para cada atributo, o algoritmo calcula o ganho de informação.
2. seleção do atributo: o atributo com maior ganho de informação é selecionado para dividir os dados.
3. divisão dos dados: os dados são divididos em partições menores de acordo com o valor do atributo selecionado.
4. repetição das etapas 1 a 3: as etapas 1 a 3 são repetidas para cada partição de dados, até que se chegue às folhas da árvore de decisão.
5. atribuição de classes: as folhas da árvore de decisão são atribuídas a classes, determinando a classe a qual cada ponto de dados pertence.

2.3.4 Algoritmo CART

O algoritmo árvore de decisão CART (*Classification and Regression Trees*), pode lidar com problemas de classificação e regressão foi apresentado pela primeira vez por Breiman et al. (1984).

O CART é caracterizado pelo fato de construir árvores binárias, ou seja, cada nó interno tem exatamente duas arestas de saída. As divisões são selecionadas usando o critério de divisão como o índice Gini, que tem a função de selecionar os pontos de corte mais eficientes para dividir os dados em nós internos da árvore, e a árvore obtida é podada por poda de custo-complexidade (SINGH; GUPTA, 2014).

A divisão em subárvores no algoritmo CART se dá da seguinte maneira: O nó raiz é dividido em dois sub-nós, utilizando-se o atributo mais relvante e o valor limite para este atributo. O processo é então repetido recursivamente, até que cada sub-nó seja puro ou até que o número máximo de folhas seja alcançado.

Segundo Timofeev (2004) o algoritmo CART tem as seguintes vantagens e desvantagens:

Vantagens:

- lida facilmente com variáveis numéricas e categóricas;
- identifica as variáveis mais significativas e elimina as não significativas;
- lida facilmente com *outliers*.

Desvantagem:

- o CART pode gerar árvores de decisão instáveis.Modificações insignificantes na amostra de aprendizado, como a eliminação de várias observações, podem levar a mudanças radicais na árvore de decisão: aumento ou diminuição da complexidade da árvore, mudanças na divisão de variáveis e valores;
- CART divide apenas por uma variável. Em outras palavras, todas as divisões são perpendiculares ao eixo. Ou seja, pode lidar estuturas de dados simples, no entanto,se os dados tiverem uma estrutura mais complexa, então o CART pode não capturar a estrutura correta dos dados.

2.3.5 Algoritmo CHAID

Criado pelo estatístico Kass (1980), o algoritmo CHAID (*Chi-squared Automatic Interaction Detection*) é um dos métodos mais populares para construir árvores de decisão com base em estatísticas. Sua essência reside na busca por associações entre uma variável dependente categórica e múltiplas variáveis independentes, sejam elas categóricas ou numéricas (que, nesse caso, passam por um processo de codificação e transformação) (MILANOVIĆ; STAMENKOVIĆ, 2016).

Vantagens:

- simples de entender e interpretar;
- flexível e adaptável a diferentes tipos de dados;
- robusto e confiável, mesmo com conjuntos de dados pequenos;
- capaz de lidar com variáveis categóricas e numéricas.

Desvantagens:

- pode ser sensível a *outliers*;
- pode gerar árvores complexas com muitos nós;
- pode ter dificuldade em lidar com variáveis altamente correlacionadas.

3 Trabalhos Relacionados

Este capítulo aborda trabalhos relacionados levando em conta trabalhos sobre interpretabilidade de grupos sem uso de rotulação e trabalhos com rotulação automática de grupos da literatura, uma vez que o objetivo principal do processo de rotulação de grupos é o de auxiliar na interpretação dos grupos gerados.

Trabalhos Sobre Interpretabilidade de Grupos

A interpretabilidade ou explicabilidade de agrupamento visa explicar como uma instância de entrada $x \in \mathbb{R}^D$ (não necessariamente no conjunto de treinamento) é mapeada ou atribuída a um determinado *cluster* (GABIDOLLA; CARREIRA-PERPIÑÁN, 2022).

Modelos de interpretabilidade de agrupamento referem-se a modelos de agrupamento que explicam o que caracteriza um *cluster* e como esse *cluster* se distingue de outros. Os modelos de interpretabilidade de agrupamentos são importantes porque permitem aos usuários entender os resultados do agrupamento e tomar decisões informadas com base nos resultados. São úteis também para validar os resultados do agrupamento e garantir que os *clusters* sejam representativos. Alguns dos modelos de agrupamento são baseados em: regras; limites retangulares e árvores de decisão (CHEN, 2018). A seguir são apresentados alguns modelos de interpretabilidade que utilizam árvores de decisão.

Em Basak e Krishnapuram (2005b) os autores propõem um método de agrupamento hierárquico baseado em árvores de decisão não supervisionadas. O método explora a capacidade da árvore para segmentar diferentes conjuntos de dados e interpretá-la em termos de regras.

Cada nó folha da árvore representa um *cluster*, e o caminho do nó raiz para um nó folha representa uma regra. Essa interpretabilidade facilita a compreensão dos resultados do agrupamento.

Quatro medidas diferentes são utilizadas para selecionar o atributo mais apropriado para dividir os dados em cada nó de ramificação. Além disso, dois algoritmos diferentes são propostos para realizar essa divisão.

O trabalho fornece a base teórica para a abordagem e demonstra sua capacidade de segmentar vários conjuntos de dados. O desempenho da árvore de decisão não supervisionada é comparado com o da árvore supervisionada, mostrando resultados promissores. O método de agrupamento hierárquico com árvores de decisão não supervisionadas é uma ferramenta útil para a análise de dados, especialmente quando a interpretabilidade dos resultados é importante.

Em Bertsimas, Orfanoudaki e Wiberg (2020) os autores apresentam um novo método de aprendizado não supervisionado para agrupamento de dados, denominado *Interpretable Clustering via Optimal Trees* (ICOT). O ICOT se baseia em técnicas de otimização de múltiplos inteiros mistos (MIO) para gerar árvores de agrupamento interpretáveis que otimizam diversas métricas de validação interna. O método é capaz de lidar com dados numéricos e categóricos mistos, e oferece uma interpretabilidade significativamente maior que outros métodos de agrupamento.

As principais contribuições do trabalho são as seguintes:

1. formulação MIO para agrupamento globalmente ótimo: uma nova formulação MIO para o problema de aprendizado não supervisionado é proposta, a qual leva à criação de árvores de agrupamento globalmente ótimas. Essa formulação motiva o desenvolvimento do algoritmo ICOT.

2. extensão do algoritmo OCT: o ICOT se baseia no algoritmo *Optimal Classification Trees* (OCT) apresentado em Bertsimas e Dunn (2017), estendendo-o para a configuração não supervisionada.

3. interpretabilidade intrínseca: a interpretabilidade é incorporada ao processo de criação dos *clusters* no ICOT, tornando os resultados mais fáceis de entender e interpretar.

4. implementação eficiente e escalável: o ICOT é implementado com base em uma abordagem iterativa de descida de coordenadas, que é eficiente e escalável para problemas de grande porte.

5. validação abrangente: o ICOT utiliza dois critérios de validação amplamente aceitos: a *Silhouette Metric* e o *Dunn Index*, como a função objetivo do algoritmo.

6. ajuste automático da complexidade da árvore: o ICOT foi desenvolvido para que o ajuste da complexidade da árvore seja redundante, pois as funções de perda utilizadas levam em consideração tanto a densidade intracluster quanto a separação entre clusters.

7. flexibilidade para ajustes personalizados: o usuário pode ajustar o ICOT selecionando a profundidade máxima da árvore e o número mínimo de observações em cada cluster.

8. incorporação de dados mistos: o ICOT propõe uma solução para a incorporação de dados numéricos e categóricos mistos, utilizando uma medida de distância reponderada que evita que um único tipo de variável domine o cálculo da distância.

9. avaliação abrangente em cenários diversos: o desempenho do ICOT foi avaliado em relação a várias abordagens de agrupamento em diversos conjuntos de dados,

incluindo conjuntos sintéticos do FCPS e dados do mundo real como o Framingham Heart Study e um programa de compartilhamento de bicicletas.

10. escalabilidade para grandes conjuntos de dados: a capacidade do ICOT de escalar para grandes conjuntos de dados foi testada usando o FCPS e dados do programa de compartilhamento de bicicletas. Os resultados demonstram que o ICOT pode manipular com eficiência conjuntos de dados de até centenas de milhares de observações.

O método é capaz de gerar árvores de agrupamento globalmente ótimas, interpretar os resultados de forma clara e eficiente, lidar com dados numéricos e categóricos mistos, e escalar para grandes conjuntos de dados e suas contribuições podem ter um impacto positivo em diversas áreas de aplicação.

O trabalho de Gabidolla e Carreira-Perpiñán (2022) propõe um novo método para agrupamento interpretável de dados utilizando árvores oblíquas esparsas. Essa abordagem se concentra no caso, relativamente inexplorado, de agrupamento interpretável em dados de alta dimensão.

A principal vantagem do método reside no uso de árvores oblíquas esparsas, que possuem maior poder de modelagem do que as árvores tradicionais, especialmente em cenários de alta dimensionalidade. Ao mesmo tempo, as árvores oblíquas esparsas preservam a interpretabilidade do modelo, permitindo a compreensão das características que influenciam a atribuição de clusters.

Outras vantagens da abordagem incluem:

1. compreensão global: permite entender como cada instância é mapeada para um cluster, identificando quais características são relevantes para cada cluster.
2. estrutura hierárquica: impõe uma estrutura hierárquica nos clusters, enquanto otimiza a função objetivo original do agrupamento.
3. controle da complexidade: a complexidade da árvore é controlada por um hiperparâmetro, permitindo explorar um perde-e-ganha entre a precisão do agrupamento e a simplicidade da explicação.

O método foi demonstrado utilizando o método K-means, mas pode ser aplicado a outros métodos de agrupamento que se baseiam em uma função de custo.

Os três trabalhos apresentam algoritmos de agrupamento interpretáveis que são eficientes e robustos. Podem ser usados em uma variedade de aplicações, como descobrir grupos de clientes com características semelhantes em um banco, identificar grupos de genes com funções semelhantes em um estudo de bioinformática, ou segmentar um mercado-alvo em grupos de clientes com necessidades semelhantes. A Tabela 1, traça um paralelo entre os trabalhos.

Tabela 1 – Tabela comparativa entre os algoritmos de agrupamento interpretáveis.

Basak e Krishnapuram (2005b)	Constrói uma árvore hierárquica de um conjunto de dados não rotulado.	O algoritmo é interpretável: os usuários podem entender facilmente como os objetos são agrupados.	O algoritmo é robusto a ruídos e outliers: pode ser usado para construir árvores de conjuntos de dados com dados ausentes ou ncorretos.
Bertsimas, Orfanoudaki e Wiberg (2020)	Usa MIO para agrupar dados.	O algoritmo é interpretável: os usuários podem entender facilmente como os objetos são agrupados.	O algoritmo é robusto a ruídos e outliers: pode ser usado para agrupar dados com dados ausentes ou incorretos.
Gabidolla e Carreira-Perpiñán (2022)	Usa árvores de decisão oblíquas para agrupar dados.	O algoritmo é interpretável: os usuários podem entender facilmente como os objetos são agrupados.	O algoritmo é eficiente: executa rapidamente, mesmo para grandes conjuntos de dados.

Trabalhos Com Rotulação Automática de Grupos

O conceito de rotulação de grupos foi apresentado inicialmente em Lopes, Machado e Rabêlo (2013), neste trabalho uma base de dados é dada como entrada para um algoritmo de aprendizagem não-supervisionada, assim gerando os grupos a partir dos elementos iniciais. Então é aplicado um algoritmo de aprendizagem supervisionada para cada grupo gerado anteriormente para seleção de atributos mais relevantes. Caso haja valores contínuos na base de entrada é aplicado um método de discretização antes do uso do algoritmo de aprendizagem supervisionado. O modelo proposto é dividido em 4 etapas e é demonstrado na Figura 3.

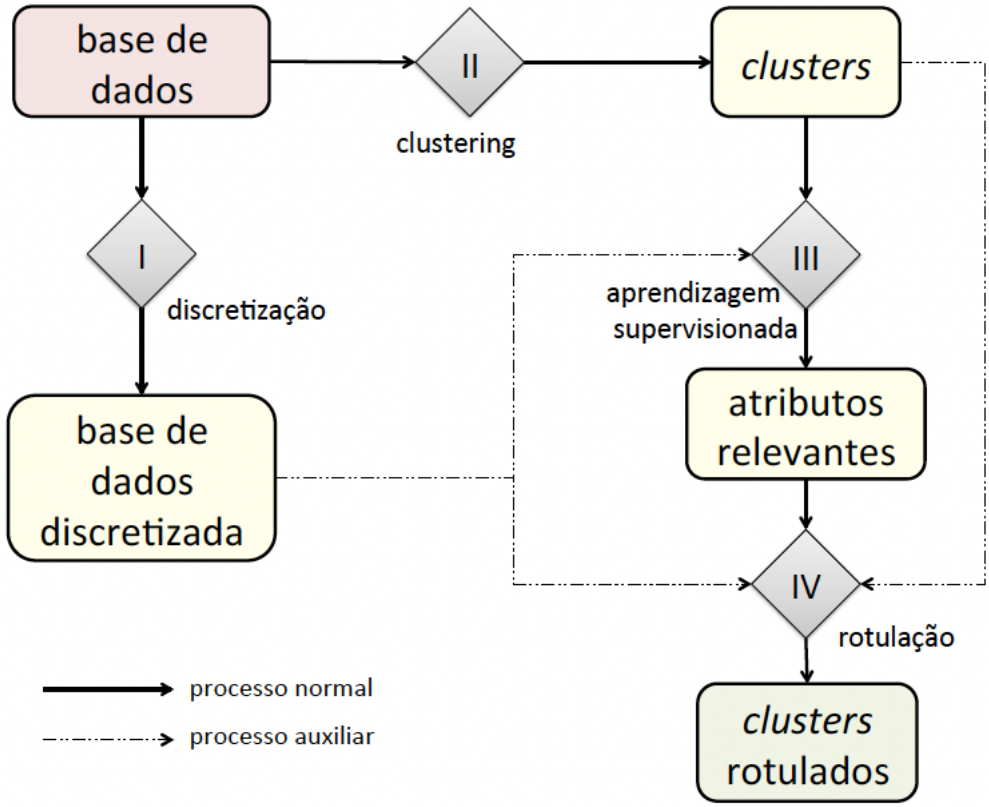


Figura 3 – Modelo proposto em (LOPES; MACHADO; RABÊLO, 2013).

As etapas do modelo são descritas a seguir:

1. na etapa I um método de discretização é utilizado com intuito de melhorar a escolha de uma faixa de valores de maneira mais assertiva para representar atributos que compõem o(s) rótulo(s) de determinado grupo. Os autores escolheram o método de discretização chamado técnica de Discretização por Larguras Iguais (EWD) que divide um intervalo em partes iguais e o método Discretização por Frequências Iguais (EFD) que divide o intervalo com base na frequência dos valores do atributo. O

número de faixas de valores para cada atributo é controlado pelo parâmetro R, que especifica a quantidade de faixas desejadas.

2. na etapa II a base de entrada é aplicada ao um algoritmo de aprendizagem não supervisionada, neste trabalho os autores utilizaram-se do algoritmo *K-means*.

3. na etapa III do modelo, o algoritmo supervisionado é uma Rede Neural Artificial (RNA) do tipo *Multilayer Perceptron* (MLP), como método de seleção dos atributos. A RNA prevê os atributos mais relevantes baseados no valores dos demais atributos. Um parâmetro "M"determina quantas vezes a RNA será executada. Nesta etapa ainda é aplicado uma parâmetro (V), que determina uma nota de corte para a escolha dos atributos previstos pela Rede Neural.

   na etapa IV depois de selecionados os valores mais comuns nos atributos são incluídos na classificação. Caso os dados já sejam discretos, é escolhido o valor mais frequente que compõe a classificação. O resultado final é uma lista de valores ou faixas de valores associados aos seus atributos correspondentes.

O Modelo de Rotulação Automática (MRA) proposto pelos autores foi capaz de gerar rótulos com taxas de acertos acima de de 90% . O método MRA foi ainda aplicado a outros problemas (LOPES; MACHADO; RABÊLO, 2013; LOPES; MACHADO; RABELO, 2014; LIMA; MACHADO; LOPES, 2015; LOPES et al., 2016) apresentando bons resultados.

O MRA segundo seus autores pode fazer o uso de outros algoritmos de aprendizagem não-superviosionada, aproveitando-se disso nos trabalho de Sousa et al. (2016) e Araujo et al. (2018) se utilizaram dos algoritmos DAMICORE e Cobweb respectivamente. Estes autores demonstraram que a melhoria da qualidade dos grupos gerados resultou em melhores resultados na rotulação dos grupos.

Apesar dos bons resultados apresentados pelo MRA e suas variações possuem alguns pontos fracos:

1. na etapa de discretização a escolha do método de pode alterar os resultados além da utilização de faixas pre fixadas;

2. custo computacional do uso de RNA;

3. muitos parâmetros a serem alterados (M, V e R) para cada conjunto de grupos analisados.

Foram propostos novos modelos de rotulação de grupos diferentes do MRA, no trabalho de Machado, Ribeiro e RABêLO (2015) é apresentado um modelo de rotulação utilizando-se do algoritmo *Fuzzy C-Means*, que tem como saída uma matriz U que atribui

um grau de pertinência a cada elemento de um conjunto de dados que se relaciona com cada grupo ou seja quanto mais próximo o elemento estiver de um grupo, maior é seu grau de pertinência em relação ao grupo.

É definido um parâmetro chamado Grau de Seleção (GS) que serve para selecionar os elementos de um grupo cujo grau de pertinência seja maior que o GS dado. O outro parâmetro definido é o Incremento do Grau de Seleção (IGS), que servirá para incrementar o GS com o intuito de diminuir os rótulos com intersecção de valores.

O modelo funciona dando entrada em uma base de dados ao algoritmo *Fuzzy C-Means*, defini-se o número de a quantidade de grupos e o GS igual 0,5 .Depois em uma primeira iteração são definidos os atributos com grau de pertinência maiores que o GS e seus valores inferiores e superiores para cada atributo, o seja os rótulos de cada grupo. Numa segunda iteração com o intuito de diminuir as intersecções entre os valores de rótulos, é feito um incremento ao GS de acordo com o valor atribuído ao parâmetro IGS. O processo se repete até que para cada grupo exista pelo menos uma faixa de valor que não possua intersecção.

Os resultados apresentados por este modelo foram satisfatórios se comparados ao de Lopes, Machado e Rabêlo (2013) alcançando em média de acerto de 96,61%. Apesar disto este modelo apresenta algumas limitações:

1. é necessário um número muito grande de iterações para se alcançar bons resultados;
2. o método é dependente do Algoritmo *Fuzzy C-Means*.

Afim de resolver o problema da dependência do Algoritmo *Fuzzy C-Means* Filho et al. (2020) propôs o uso de algoritmos baseados em distâncias como o *K-Means* , convertendo essas distâncias em grau de pertinência. Este modelo é divido em duas Etapas:

1. Na primeira etapa são gerados os grupos com um algoritmo não-supervisionado baseado em distâncias, neste trabalho foi utilizado o *K-Means*, depois as distâncias ao centroides são transformadas em Grau de Pertinência.
2. Na segunda é estabelecido os limites de valores para cada atributo em cada grupo criado, verificado se existem interseções entre os limites de valores e, por fim, atribuído e exibidos os rótulos de cada grupo com os limites de valores que não possuem intersecção. Isso é feito selecionando os elementos que possuem um GP maior que o parâmetro GS, que inicialmente é atribuído um valor de 0,5. Assim, em cada grupo selecionado são identificados os valores máximos e mínimos de cada atributo. Estes valores correspondem aos limites de valores de cada grupo.

O autores fazem uma comparação com os resultados alcançados por Machado, Ribeiro e RABêLO (2015), que aplicada a base *Íris* teve uma taxa percentual de acerto de 92,88% enquanto o destes teve uma taxa de acerto de 92,52%, sendo estes resultados próximos com uma pequena de diferença de 0,36%, para esta base.

No modelo de rotulação CAIBAL (Cluster-Attribute Interdependency Based Automatic Labeler), criado por Moura, Veras e Machado (2022), os autores propõem o uso do algoritmo de discretização CAIM (*Class-Attribute Interdependency Maximization*, com o intuito de diminuir a perdas de informação por dissociação de valores de alta frequência adjacentes no processo de discretização durante a escolha dos pontos de corte, que está presente no trabalho de Lopes, Machado e Rabêlo (2013).

O modelo CAIBAL propôs o uso de duas opções para a escolha dos intervalos de valores. No chamado padrão é selecionado o intervalo baseado na maior acurácia mais elevada após a discretização e o método alternativo que se utiliza do parâmetro V para avaliar as acurácias dos atributos. O modelo obteve 98,49% de taxa de acerto com o uso do método padrão e de 97,33% com o método alternativo em 3 das 5 bases de dados.

O modelo CAIBAL é dividido em seis etapas:

1. na primeira etapa é realizado o agrupamento a conjunto de dados utilizando-se do algoritmo *K-Means*;
2. na segunda etapa é inserido no conjunto um novo atributo, que identifica o grupo ao qual o objeto pertence;
3. durante a terceira etapa é feito um pré-processamento que tem o objetivo de mitigar a dissociação de informação, que consiste em selecionar como possíveis pontos de corte de um determinado atributo somente os valores com baixa frequência de ocorrência;
4. a discretização é realizada na quarta etapa utilizando o algoritmo CAIM;
5. a quinta etapa é responsável pela seleção dos atributos que comporão cada grupo;
6. e por fim na sexta etapa é aplicada uma técnica de agregação dos dados para produção dos rótulos que melhor descrevem cada *cluster*.

No trabalho de Silva et al. (2021) é definido um modelo de rotulação dividido em duas etapas: Na etapa I é realizado a definição dos pares atributo-intervalo e na etapa II é feita a seleção dos pares que melhor representam cada grupo. Durante a primeira etapa, o método utiliza técnicas de regressão para estimar a função que reflete o erro na previsão dos atributos dentro de seus respectivos domínios para cada grupo. Isso permite a delimitação de faixas nas quais o erro de previsão é o mínimo possível para cada atributo de cada grupo. Na segunda etapa, os pares atributo-intervalo identificados na primeira etapa

são escolhidos para representar a maioria dos elementos dos grupos e para diferenciá-los, desta maneira formando os rótulos.

O trabalho mais atual da literatura é o de Alencar (2022) que propôs um modelo de classificação que se utiliza da seleção de atributos. A autora enfatiza que a escolha do algoritmo não supervisionado tem relação direta com os resultados alcançados o algoritmo escolhido foi o *K-Means*. Após é aplicado o método de discretização EWD para transformar os valores contínuos em faixas de valores, dessa forma, um valor discreto é associado a cada intervalo distinto de valores contínuos. Por último, os filtros de seleção de atributos são utilizados em conjunto com a variação V para selecionar os atributos mais relevantes de cada grupo.

Na Tabela 2 temos uma análise comparativa entre os diversos modelos de rotulação propostos na literatura.

Tabela 2 – Tabela com análise dos modelos de rotulação da literatura.

Trabalhos	**Algoritmo**	**Discretização**	**Seleção de Atributos**
Lopes, Machado e Rabêlo (2013)	*K-Means*	EWD e EFD	Rede Neural Artificial
Machado, Ribeiro e RABêLO (2015)	*Fuzzy K-Means*	Não utiliza	Fuzzy C-Means com Grau de Pertinência
Filho et al. (2020)	*K-Means*	Não utiliza	Fuzzy com Grau de Pertinência
Moura, Veras e Machado (2022)	*K-Means*	CAIM	Rede Neural Artificial
Silva et al. (2021)	*K-Means*	Não utiliza	Regressões
Alencar (2022)	*K-Means*	EWD	Ganho de Informação

As árvores de decisão oferecem uma capacidade de interpretabilidade que as torna valiosas para a rotulagem de grupos. Este trabalho propõe um método de rotulação de grupos que se aproveita dessa qualidade, simplificando o processo em comparação com outros métodos da literatura.

O método proposto não se preocupa com discretização ou seleção de atributos sendo esta tarefas inerentes dos algoritmos de árvores de decisão, etapas presentes em outros métodos e que podem ser complexas e demoradas. Além disso, o método é flexível e pode ser usado com qualquer algoritmo de agrupamento.

Vantagens:

1. simplicidade: o método é simples e fácil de implementar, não exigindo conhecimento especializado em árvores de decisão.

2. flexibilidade: o método pode ser usado com qualquer algoritmo de agrupamento.

3. interpretabilidade: as árvores de decisão fornecem uma interpretação clara dos grupos formados.

4. parametrização mínima: o método possui poucos parâmetros, o que facilita sua aplicação.

4 Modelo Proposto

Este trabalho propões um modelo de rotulação automática de grupos com uso de árvore de decisão. Para escolha de qual melhor algoritmo de árvore de decisão é realizado um comparativo entre algoritmos ID3, C4.5, CART e CHAID para definir qual deles gera melhores rótulos. A motivação da escolha destes algoritmos é explicada a seguir:

4.1 Modelos de conjuntos de árvores versus modelos simples de árvores de decisão

Ao se criar modelos de aprendizagem de máquina deve se preocupar com a relação entre a precisão e interpretabilidade. Existe um crescente numero de áreas críticas onde a compreensão das sáidas é tão vital quanto a precisão da saída (TEODORO; MONACI; PALAGI, 2024), como é o caso da rotulação de grupos.

Modelos de conjunto de árvores de decisão, como *Random Forest* árvores [Boosted Trees, são populares em modelos de aprendizado de máquina, especialmente para tarefas de previsão. Devido ao seu alto desempenho de previsão, são um dos métodos mais recomendados ao lidar com problemas reais. No entanto segundo Hara e Hayashi (2016) o principal problema dos modelos de conjunto de árvores está na interpretabilidade. Pois dividem um espaço de entrada por um número de pequenas regiões e fazer previsões dependendo de uma região. Geralmente, o número de regiões que geram é superior a mil, o que significa aproximadamente que existem milhares de diferentes regras para previsão, fazendo com que não especialistas não consigam entender um número tão grande de regras.

Uma árvore de decisão simples, por outro lado, é bem conhecido como um dos mais interpretáveis modelos. Apesar da fraca capacidade de previsão, o número de regiões geradas por uma única árvore são drasticamente pequenas, o que torna o modelo transparente e compreensível. Esses modelos dividem um espaço de entrada por um número de pequenas regiões e fazem previsões dependendo de uma região. O modelo proposto faz uma análise das regras geradas pelas árvores de decisão, por isso modelos mais compreensíveis são melhores escolhas para esse modelo.

4.2 O Modelo

Dado o problema de rotulação definido em (LOPES; MACHADO; RABÊLO, 2013), o presente modelo apresenta uma abordagem capaz de rotular *clusters* se utilizando de árvores de decisão, uma vez que métodos de aprendizado supervisionados baseados em

árvore, são uma solução natural para problemas que priorizam a interpretabilidade, suas divisões de recursos e caminhos de decisão oferecem informações sobre os recursos diferenciadores entre os membros em cada folha (BERTSIMAS; ORFANOUDAKI; WIBERG, 2020).

Para a comparação do resultado alcançado por cada algoritmo foi desenvolvido um modelo que é dividido em quatro passos: (I) com o uso de um algoritmo não supervisionado de agrupamento (*k-means*) são gerados o grupos a partir de uma base de dados de entrada, (II) com a base agrupada é gerado o modelo de arvores de decisão, tendo como atributo classe o *cluster* gerado, nesta fase é utilizada um dos quatro algoritmos de árvore de decisão: ID3, C4.5, CART e CHAID, (III) são selecionados os ramos da árvore gerada que leva a maior quantidade acerto para cada *cluster* que foi usado como classe e por fim (IV) dentro de cada ramo selecionado são listados os atributos e calculado suas respectivas faixas de valores. As etapas são demonstradas na Figura 4.

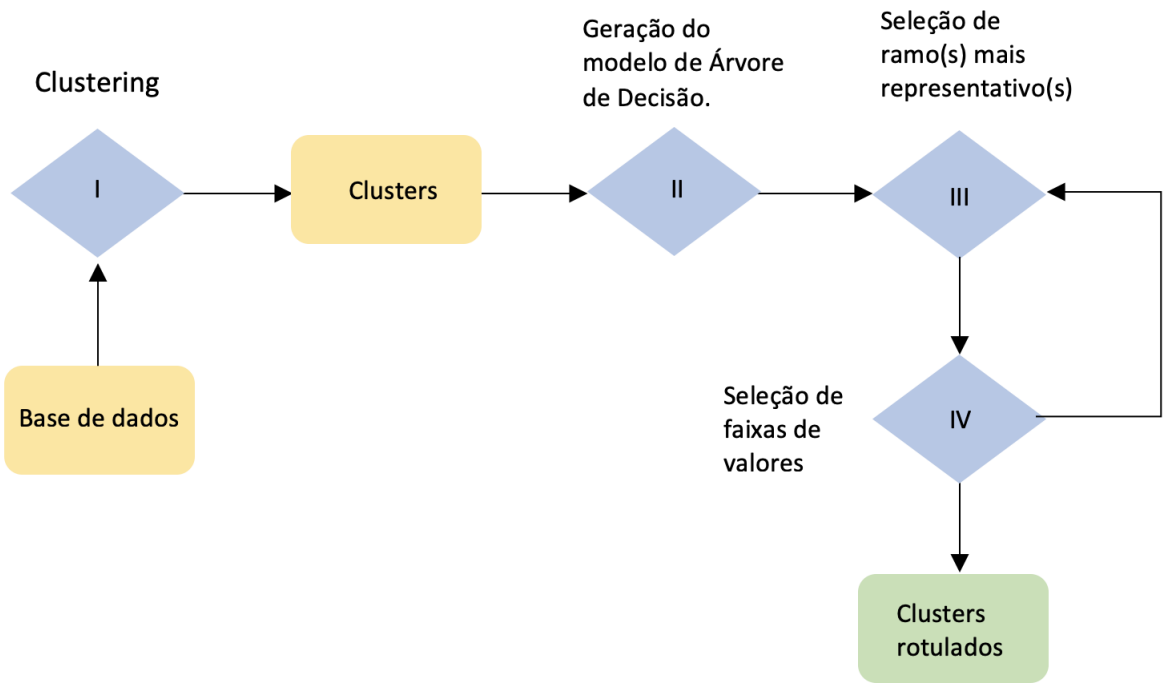


Figura 4 – Fluxograma do modelo proposto.

Para demonstrar o processo de rotulação, será utilizada a base *Iris Data Set*, apresentada inicialmente por Fisher (1936), que identifica plantas contendo 150 instâncias com 4 atributos que são: Comprimento da Sépala (CS), Largura da Sépala (LS), Comprimento da Pétala (CP) e Largura da Pétala (LP), expressos em centímetros (DUA; GRAFF, 2017). A base divide-se em três partes iguais para cada uma das especies: 50 registros do tipo Iris-setosa; 50 registros do tipo *Iris-versicolour* e 50 registros do tipo do tipo *Iris-virginica*.

A especie da planta é o atributo classe original da Base de Dados. Como existem três tipos de especies na base foi o utilizado esse mesmo número de grupos a serem gerados,

que foram utilizados como atributo alvo para a classificação da base agrupada pela árvore de decisão. A base é composta de 4 atributos contínuos. No entanto no caso de bases onde não se tem a informação sobre o número exato de para o K (número de *clusters*), deve-se utilizar algum método para determina-lo, como Método do Cotovelo (do inglês *Elbow Method*), Métrica da Silhueta, Distorção e Estatística de *Gap*.

4.2.1 Passo I - Gerando *clusters*

O processo de agrupamento da base de dados Íris foi efetuada com o uso do algoritmo *k-means* sendo definido um número para k(número total de *cluster*) igual a 3. A distribuição das instâncias por grupos ficou de acordo com a Tabela 3.

Tabela 3 – Distribuição de amostras por grupo da base *Íris*.

Cluster	**Quantidade de amostras**
0	62
1	50
2	38

4.2.2 Passo II - Geração do modelo de árvore de decisão

No segundo passo a base com seus respectivos *clusters* é gerado o modelo de árvore de decisão, utilizando como atributo classe o número do grupo. Para a geração das árvores demonstrada nas Figuras 5, 6, 7 e 8 foi utilizada os valores padrões para cada respectivo algoritmo (ID3, C4.5, CART e CHAID) da biblioteca *ChefBoost* [1] , que implementa cada uma destes quatro algoritmos (SERENGIL, 2021).

4.2.3 Passo III - Seleção de ramos com melhor classificação

Nesse passo a partir da arvore binária gerada no passo anterior é realizada a escolha das subárvores que levam aos melhores resultados de classificação para cada *cluster*, uma vez que os *clusters* gerados no passo I foram utilizados como classe para indução da árvore de decisão. Portanto ao fim do passo III temos o conjunto dos ramos que serão utilizados no passo IV.

A seguir é demonstrado cada ramo escolhido para cada *cluster*. Para efeito de simplificação será demonstrado somente como foram escolhidos os ramos da árvore gerado pelo algoritmo CHAID, que como pode ser visto nas Tabelas 5, 6, 7 e 8, resultou nas melhoras taxas de concordância dentre os demais algoritmo.

Para o *cluster* 0 (zero) foi selecionado o ramo composto pelos nós (1,3,4, 6 e 7) da árvore, conforme é demonstrado na subárvore da figura 9. Neste caso haviam dois ramos

[1] disponível em https://github.com/serengil/chefboost

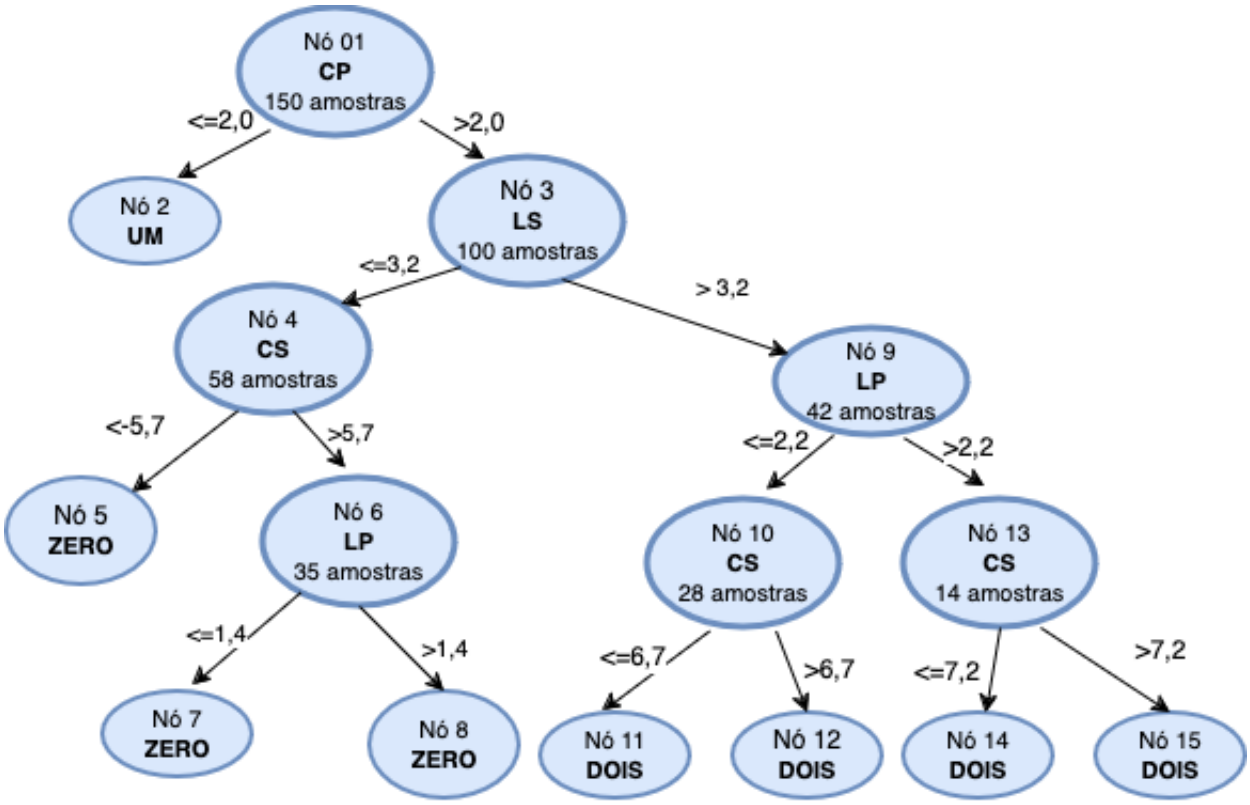


Figura 5 – Árvore induzida com a base *Íris* a partir de seus *clusters* com o algoritmo ID3

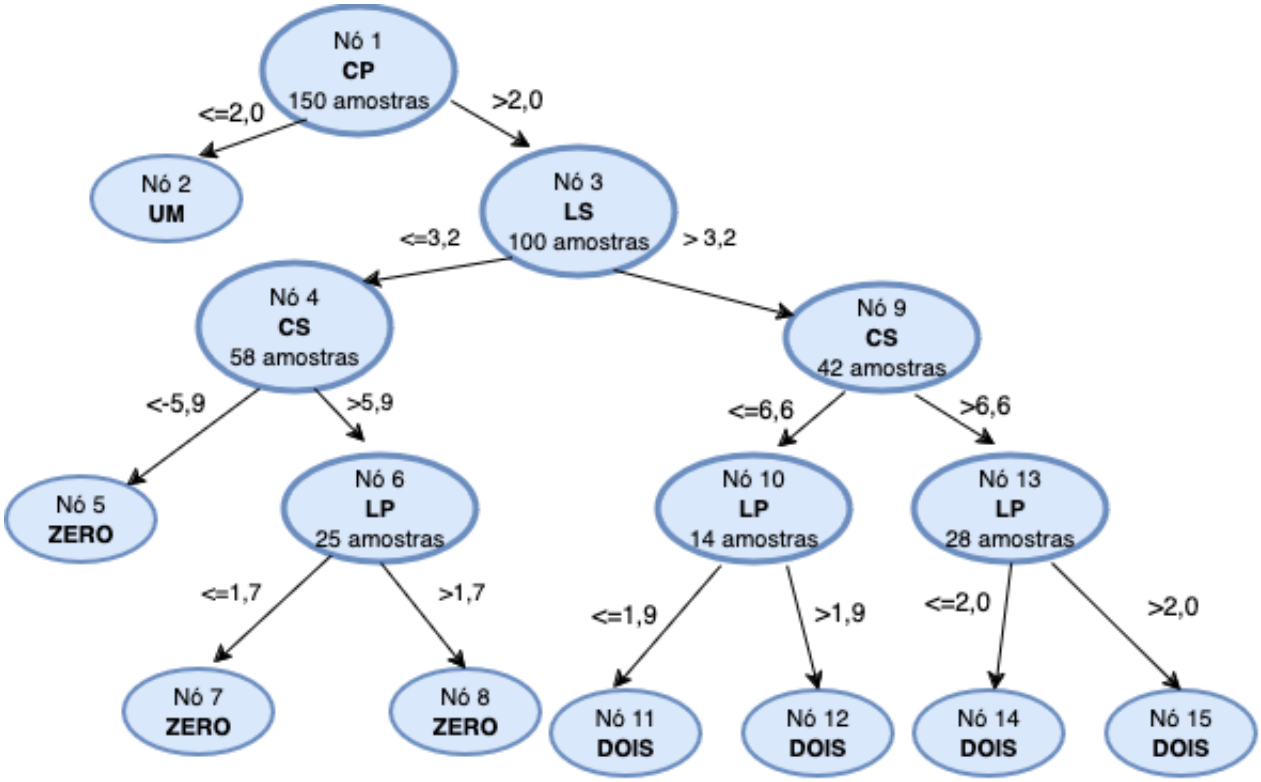


Figura 6 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo C4.5

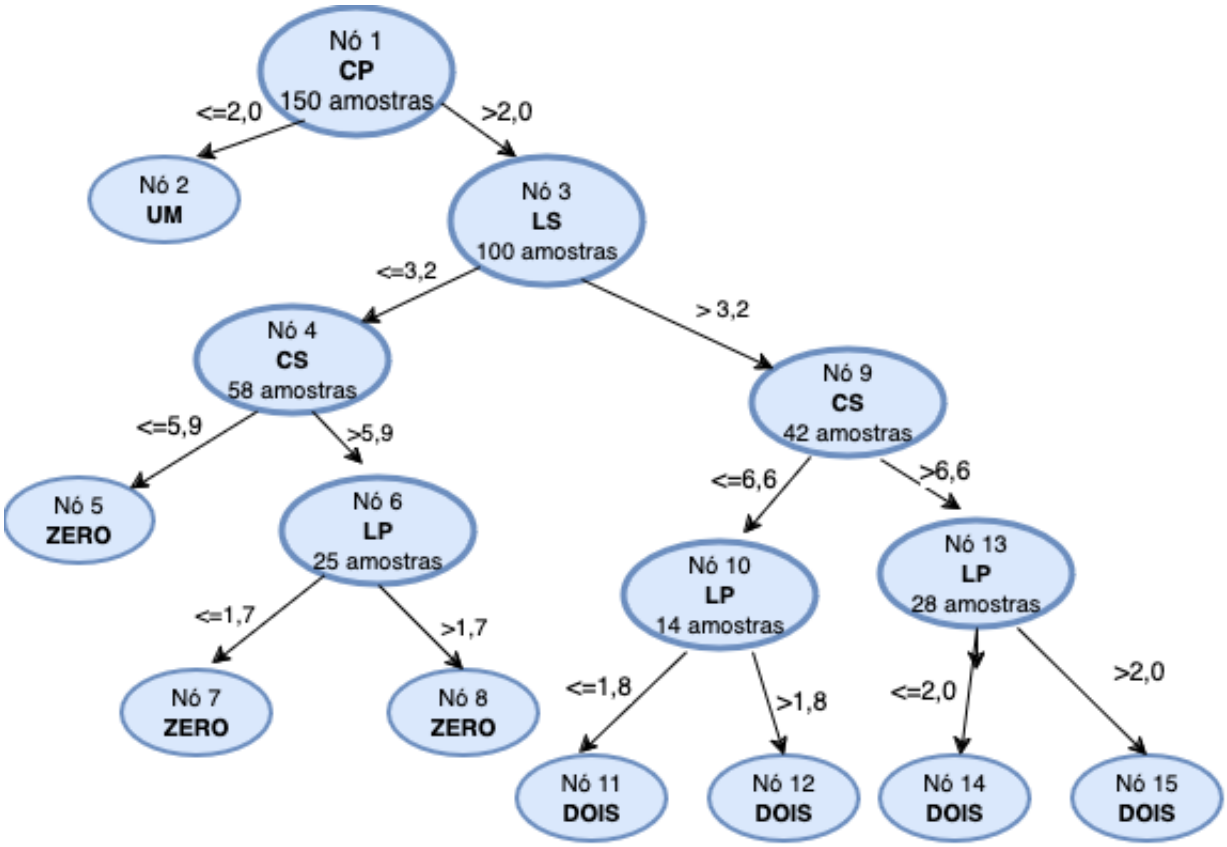


Figura 7 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo CART

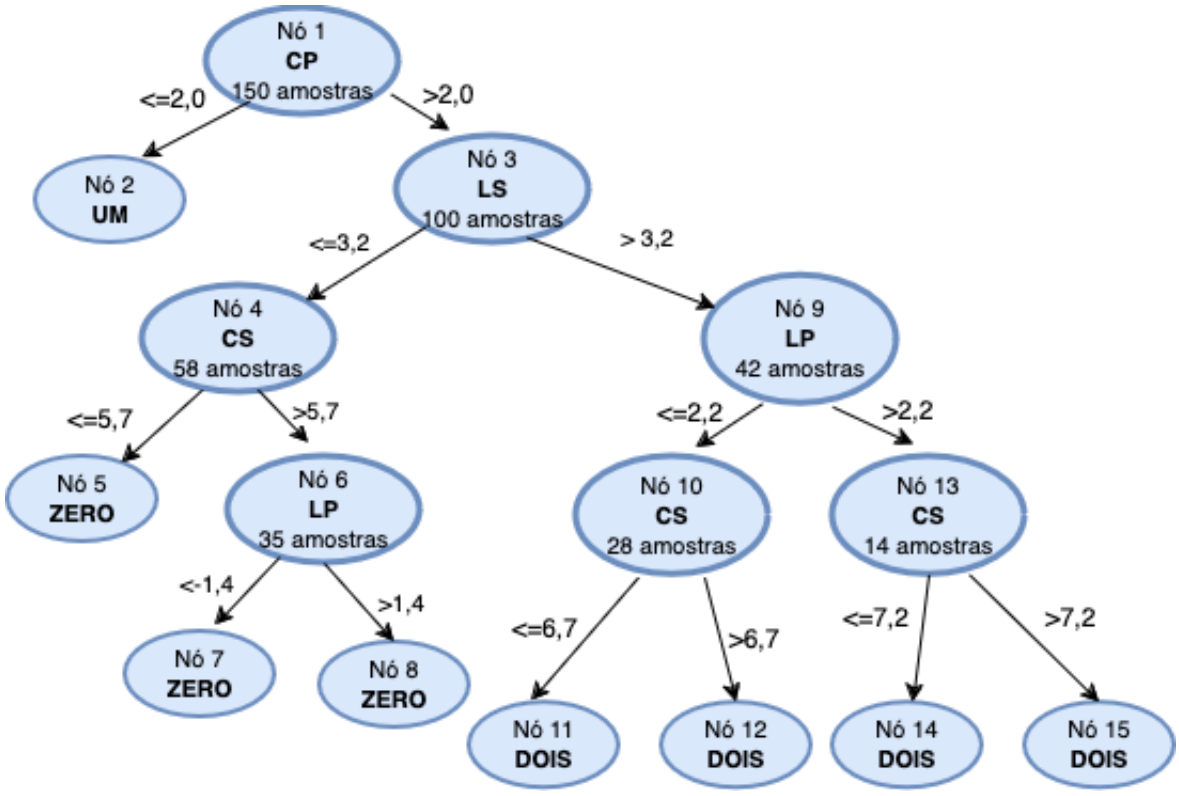


Figura 8 – Árvore induzida da base *Íris* a partir de seus *clusters* com o algoritmo CHAID

que levariam a classificação do do grupo 0, e foi selecionado aquele leva ao maior número de acerto na classificação, ou seja o com maior número de amostras no nó folha.

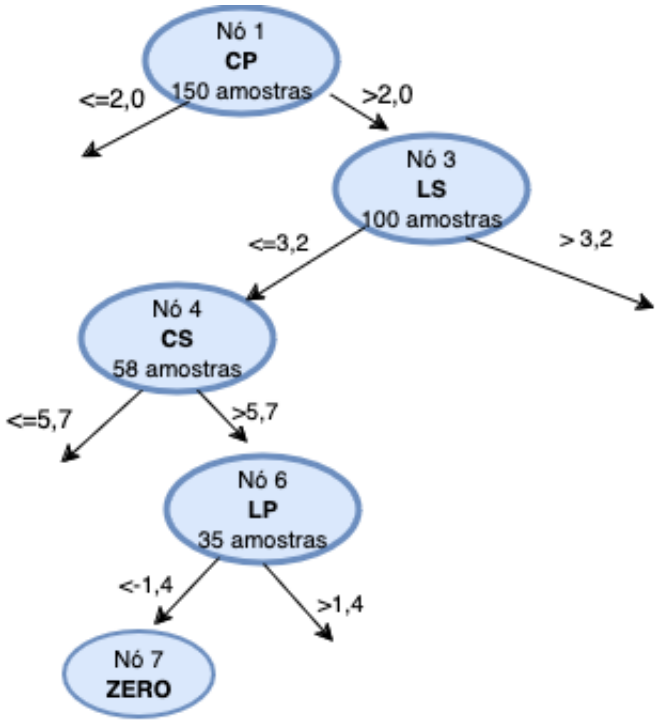


Figura 9 – Subárvore selecionada para o *cluster* zero.

Para o *cluster* 1 (um) foram selecionados os nós (1 e 2). Neste caso sendo o único caminho que leva a nó folha do *cluster* um. Este ramo é demonstrado na Figura 10.

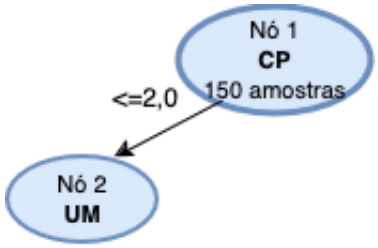


Figura 10 – Subárvore selecionada para o *cluster* um.

Para o *cluster* 2 (dois) o ramo selecionado foi o composto pelos nós (1, 3, 9 10 e 11) da árvore de decisão. Neste também foi necessário fazer a escolha do caminho que leva à folha maior número de acerto. Esta subárvore é demonstrada na Figura 11.

4.2.4 Passo IV - Seleção de atributos e atribuição de faixas de valores

Nesse passo a partir dos atributos que compõem as subárvores encontradas no passo anterior é determinado o conjunto dos atributos que farão parte do rótulo para cada *cluster*. Cada algoritmo utiliza um tipo de métrica para encontrar os pontos de decisão na árvore de decisão. Abaixo temos a Tabela 4 com as métricas utilizada por cada algoritmo da biblioteca *Chefboost*.

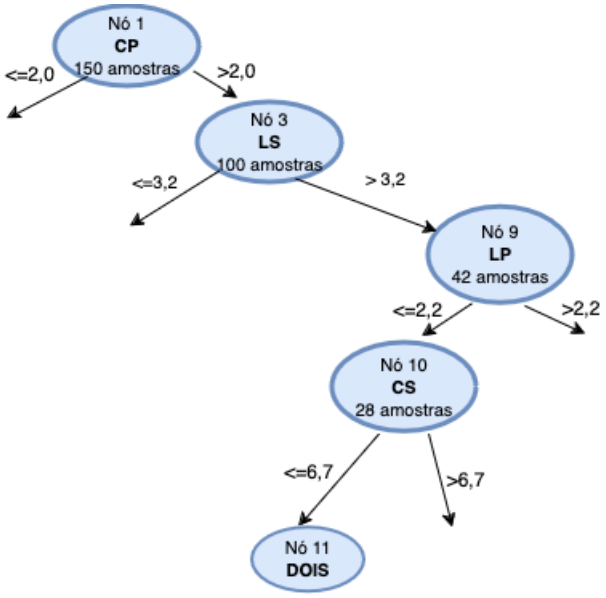


Figura 11 – Subárvore selecionada para o *cluster* dois.

Tabela 4 – Tabela com as métricas utilizadas por cada algoritmo de árvore de decisão

Algoritmo	**Métrica(s)**
ID3	Entropia, ganho de informação
C4.5	Entropia, taxa de ganho
CART	GINI
CHAID	Chi Square

Os atributos que compõem um rótulo são aqueles que pertencem aos respectivos ramos selecionados na etapa III. Dentro de cada nó de decisão gerado ao longo do ramo, além do atributo, encontramos os limites que serão utilizados para compor as faixas de valores dos rótulos. Neste caso, o número de faixas é de apenas duas, pois a partir das bases utilizadas sempre são geradas árvores binárias pelos algoritmos de árvore de decisão utilizados no estudo.

Segue a regra para a definição dos limites:

- se o nó da regra for o filho à esquerda do nó pai, a regra daquele nó determina o intervalo com o limite superior, e o limite inferior será selecionado na base de dados, de acordo com a regra.
- se o nó filho estiver à direita, a regra daquele nó determina o intervalo com o limite inferior, e o limite superior será escolhido na base de dados de acordo com a regra.

4.2.4.1 Processo de seleção de faixa de valores para os rótulos do *cluster* zero.

Para calcular a faixa de valores dos atributos do rótulo, é verificado em qual subárvore pertence o nó do atributo. Nesse caso, a faixa de valores para este *cluster* é

composta pelos atributos: Comprimento Pétala, Largura Sépala e Comprimento Sépala, da seguinte forma:

1. Atributo Comprimento Pétala:

- o primeiro valor maior que 2, neste caso o valor 3,0, é considerado o limite inferior. Como a partir deste nó o caminho a seguir é à direita (o nó número 3 da árvore).
- o limite superior e buscado na base de dados, que é o valor 6,3.

2. Atributo Largura Sépala:

- o valor 3,2 é considerado o limite superior, uma vez que este nó leva ao caminho à esquerda da subárvore.
- o limite inferior na base é 2,9.

3. Atributo Comprimento Sépalas:

- o limite maior que 5,7 é considerado o limite inferior, que na base é o valor 5,8. item o limite superior selecionado na base é 7,6.

4.2.4.2 Processo de seleção faixa de valores para os rótulos do *cluster* um.

Neste *cluster*, o único atributo é o Comprimento da Pétala. Para definir a faixa de valores, observa-se o seguinte:

- o valor 1,9 é definido como o limite superior, pois a regra do ramo da árvore definida para o atributo é "menor ou igual a 2,0".
- como o nó folha está na subárvore da esquerda, o valor 1,0 é selecionado da base como limite inferior.

4.2.4.3 Processo de seleção faixa de valores para os rótulos do *cluster* dois.

No *cluster* dois, os atributos Comprimento da Pétala, Largura da Sépala e Largura da Pétala compõem o rótulo. A formação das faixas de valores do rótulo se dá da seguinte forma:

1. Atributo Comprimento da Pétala:

- o valor maior que 2,0 é o limite inferior, pois o próximo nó faz parte da subárvore da direita.
- o valor 4,5 da base é o limite inferior e o valor 6,9 é o limite superior, ambos selecionados da base.

2. Atributo Largura da Sépala:

- o limite inferior é o valor constante do nó 3, que é maior que 3,2. Como o nó seguinte no ramo está na subárvore da direita, o valor 3,3 da base é o limite inferior.
- o valor 4,4 da base é o limite superior.

3. Atributo Largura da Pétala:

- o limite superior é "menor ou igual a 2,2".
- o limite inferior é o valor 1,4 da base.

Após estes quatro passos e suas possíveis iterações encontramos o conjunto de rótulos para a base modelo. O passos do modelo proposto são apresentados no Algoritmo 1.

Algoritmo 1: Algoritmo de rotulação com árvores de decisão

```
1  início
2      Carregar base de dados;
3      Executar o algoritmo não-supervisionado de agrupamento;
4      Carregar base com grupos;
5      Executar o algoritmo supervisionado (árvore de decisão);
6      for Para cada ramo da árvore do
7          Definir o(s) ramo(s) com maior numero de acertos;
8          for cada ramo encontrado do
9              for cada atributo no ramo escolhido do
10                 if nó estiver a esquerda then
11                     Atribuir imite superior e buscar o limite inferior na base;
12                 else
13                     Atribuir limite inferior e buscar o limite superior na base;
14                 end
15             end
16         end
17     end
18     Exibir rótulos;
19 fim
```

4.2.5 Resultados de rótulos para Base *Íris*

As Tabelas 5, 6, 7 e 8 apresentam as análise dos rótulos gerados, selecionando-se cada um dos quatro algoritmos de árvore de decisão, para base *Íris* sendo de acordo com o seguinte:

- O campo *Cluster* indica o número do grupo do qual o rótulo faz parte;

- O campo # Elem. indica a quantidade de elementos total do referido *cluster*;
- O campo Attri. indica o atributo que está sendo rotulado;
- O campo Faixa de Valores indica a faixa de valores que identifica um rótulo de um atributo para determinado *cluster*;
- O campo # Erros indica a quantidade de erros do rótulo;
- O Tx. Conc (Taxa de Concordância) % indica a quantidade percentual de acertos do rótulo, que é o percentual de elementos que se enquadram dentro da faixa de valores do rótulo em relação ao número de elementos de um respectivo *cluster*.

Tabela 5 – Análise de rotulação para a base de dados *Íris* com o algoritmo ID3.

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc%**
0	62	CP	3,0 ∼ 6,3	0	100
		LS	2,9 ∼ 3,2	8	87,09
		CS	5,8 ∼ 7,6	23	62,90
1	50	CP	1,0 ∼ 1,9	0	100
2	38	CP	4,5 ∼ 6,9	0	100
		LS	3,1 ∼ 4,4	4	89.47
		LP	1,4 ∼ 2,2	13	54,78

Tabela 6 – Análise de rotulação para a base de dados *Íris* com o algoritmo C4.5.

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc%**
0	62	CP	3,0 ∼ 6,3	0	100
		LS	2,9 ∼ 3,2	8	87,09
		CS	5,4 ∼ 5,9	29	53,22
1	50	CP	1,0 ∼ 1,9	0	100
2	38	CP	4,5 ∼ 6,9	0	100
		LS	3,3 ∼ 4,4	4	89.47
		CS	6,7 ∼ 7,9	12	68,42

Tabela 7 – Análise de rotulação para a base de dados *Íris* com o algoritmo CART.

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc%**
0	62	CP	3,0 ∼ 6,3	0	100
		LS	2,9 ∼ 3,2	8	87,09
		CS	5,4 ∼ 5,9	29	53,22
1	50	CP	1,0 ∼ 1,9	0	100
2	38	CP	4,5 ∼ 6,9	0	100
		LS	3,3 ∼ 4,4	4	89.47
		CS	6,7 ∼ 7,9	12	68,42

Tabela 8 – Análise de rotulação para a base de dados *Íris* com o algoritmo CHAID.

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc%**
0	62	CP	3,0 ∼ 6,3	0	100
		LS	2,9 ∼ 3,2	8	87,09
		CS	5,8 ∼ 7,6	23	62,90
1	50	CP	1,0 ∼ 1,9	0	100
2	38	CP	4,5 ∼ 6,9	0	100
		LS	3,3 ∼ 4,4	4	89.47
		LP	1,4 ∼ 2,2	13	65,78

De acordo com as taxas médias de concordância apresentadas nas tabelas 5, 6, 7 e 8, o algoritmo CHAID apresentou os melhores resultados com taxa de concordância média de 86,46%. Por isso, este algoritmo foi escolhido para a fase II do modelo proposto, resultado abaixo do da taxa de concordância média encontrada no modelo demonstrada em (LOPES et al., 2016) , que foi de 95,43%.

Tabela 9 – Análise de rotulação para base de dados *Íris* presente em (LOPES et al., 2016).

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc%**
1	50	LP	0,1 ∼ 1	0	100
		CP	1 ∼ 3,7	0	100
2	62	CP	3,7 ∼ 5,1	6	90.32
3	38	CP	5,1 ∼ 6,9	3	92.10
		LP	1,7 ∼ 2,5	2	94.73

5
Resultados

Neste capítulo, são apresentados os resultados da aplicação do modelo de agrupamento de dados proposto no Capítulo 4. Os resultados foram obtidos a partir de três conjuntos de dados: Sementes (KULCZYCKI; CHARYTANOWICZ, 2011), Vidros (EVETT; SPIEHLER, 1989) e Vinhos (AEBERHARD; COOMANS; VEL, 1994), disponíveis no repositório UCI ((DUA; GRAFF, 2017). A escolha desses conjuntos se deu pelo fato de serem utilizados em alguns dos trabalhos relacionados ao tema, o que permite a comparação dos resultados aqui obtidos com os da literatura. Os resultados para o conjunto Íris já foram apresentados no Capítulo 4.

Em todos os casos, os mesmos parâmetros foram utilizados para todos os testes, com exceção do número de *clusters* (k), que varia de acordo com o número de classes presentes em cada conjunto de dados.

5.1 Detalhes de Implementação

A implementação do modelo proposto foi desenvolvida na linguagem *Python* [1], utilizando a biblioteca de Aprendizagem de Máquina *ChefBoost*. O algoritmo *K-means*, com seus valores padrão, foi utilizado para gerar os *clusters*. Para a classificação da base agrupada, foram utilizados os quatro algoritmos de árvore de decisão suportados pela biblioteca: ID3, C4.5, CART e CHAID. A bliblioteca.

5.2 Base de dados Sementes

A base de dados é composta por grãos pertencentes a três diferentes variedades de trigo: Kama, Rosa e Canadense, com 70 elementos cada, selecionados aleatoriamente (DUA; GRAFF, 2017). A base possui 7 atributos que representam características geométricas: Área (A), Perímetro (P), Compacidade (C), Comprimento da Semente (CM), Largura da Semente (LS), Coeficiente de Assimetria (CA) e Comprimento do Sulco da Semente (CSS). O número de grupos foi definido como 3, que corresponde à mesma quantidade de variedades de trigo.

As Tabelas 10, 11, 12 e 13 apresentam os resultados do modelo utilizando os algoritmos ID3, C4.5, CART e CHAID respectivamente.

[1] disponível em https://www.python.org/downloads/

Tabela 10 – Análise de rotulação para base de dados Sementes utilizando o algoritmo ID3

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	72	A	12,0 ∼ 14,8	33	54,16
		CA	0,7 ∼ 3,9	10	86,11
1	61	A	16,5 ∼21,1	0	100
2	77	A	10,5 ∼14,2	0	100
		CA	4,04 ∼ 8,4	20	74,02

Tabela 11 – Análise de rotulação para base de dados Sementes utilizando o algoritmo C4.5

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	72	A	12,0 ∼ 16,6	0	100
		P	14,0 ∼ 15,4	12	83,33
1	61	A	17,9 ∼21,1	13	78,68
2	77	A	10,5 ∼14,2	0	100
		P	12,4 ∼13,9	1	98,70

Tabela 12 – Análise de rotulação para base de dados Sementes utilizando o algoritmo CART

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	72	A	12,0 ∼ 16,6	0	100
		P	14,0 ∼ 15,4	12	83,33
1	61	A	17,9 ∼21,1	13	78,68
2	77	A	10,5 ∼14,2	0	100
		P	12,4 ∼13,9	1	98,70

Tabela 13 – Análise de rotulação para base de dados Sementes utilizando o algoritmo CHAID

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	72	A	12,0 ∼ 16,6	0	100
		LS	3,1 ∼ 15,4	7	90,27
1	61	A	17,9 ∼21,1	13	78,68
2	77	A	10,5 ∼14,2	0	100
		LS	2,6 ∼3,0	1	92,20

A versão do modelo com o algoritmo CHAID foi a que se saiu melhor, com uma taxa média de concordância de 92,23%, um pouco abaixo do resultado encontrado em (LOPES; MACHADO; RABÊLO, 2013) que foi de 92,65% como pode ser visto na

tabela 14

Tabela 14 – Análise de rotulação para base de dados Sementes presente em (LOPES et al., 2016).

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Acerto %**
1	67	A	12.78 ∼16.14	8	88.05
		P	13.73 ∼15.18	9	86.56
2	82	A	10.59 ∼12.78	12	85.36
		P	12.41 ∼13.73	10	87.80
3	77	P	15.18 ∼17.25	0	100
		LS	3.465 ∼ 4.033	3	95.08
		CM	5.826 ∼ 6.675	1	98.36
		A	16.1 ∼ 21.18	0	100

Abaixo temos a interpretação dos rótulos gerados para cada grupo pelo modelo proposto, utilizando o algoritmo CHAID como algoritmo de árvore de decisão:

- para os elementos do grupo 0 a Área varia de 12,0 a 16.6cm^2 e largura da semente varia de 3,1 a 15,4cm;
- para os elementos do grupo 1 a Área varia de 17,9 a 21,1cm^2;
- para os elementos do grupo 2 a Área varia de 10,5 a 14,2cm^2 e o Largura da semente varia de 2,6 a 3,0cm.

5.3 Base de dados Vinhos

A Base é composta da análise química de vinhos produzidos na mesma região da Itália vinda de três cultivares diferentes (DUA; GRAFF, 2017). Possui 13 atributos com os componentes que constituem o cada um dos três tipos de vinhos: Álcool, Ácido málico, Cinza, Alcalinidade das cinzas, Magnésio, Fenóis totais, Flavanóides, Fenóis não flavanóides, Proantocianinas, Intensidade da cor, Matiz, OD280 / OD315 de vinhos diluídos e Prolina. Possui 173 registros com 3 tipos de classes distribuídos da seguinte forma:

- Classe 1: 59 amostras;
- Classe 2: 71 amostras;
- Classe 3: 48 amostras.

As Tabelas 15, 16, 17 e 18 apresentam as análise dos rótulos gerados para respectivamente os algoritmos ID3, C4.5, CART e CHAID, como algoritmos de árvore de decisão do modelo proposto.

Tabela 15 – Análise de rotulação para base de dados Vinho com uso do algoritmo ID3

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx . Conc. %**
0	62	*Proline*	750 ∼937	26	58,06
		Color Intensity	1,28 ∼6,0	0	100
1	47	*Proline*	970 ∼1680	0	100
2	69	*Proline*	278 ∼590	0	100
		Color Intensity	1,9 ∼ 7,3	4	94,20

Tabela 16 – Análise de rotulação para base de dados Vinho com uso do algoritmo C4.5

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx . Conc. %**
0	62	*Proline*	600 ∼937	0	100
		Alcalinity of ash	14,8 ∼30,0	1	98,38
1	47	*Alcalinity of ash*	17,5 ∼27,0	0	100
2	69	*Proline*	278 ∼590	0	100
		Alcalinity of ash	10,6 ∼25,0	0	100

Tabela 17 – Análise de rotulação para base de dados Vinho com uso do algoritmo CART

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx . Conc. %**
0	62	*Proline*	600 ∼740	24	61,29
		Color Intensity	1,2 ∼6,0	0	100
1	47	*Proline*	970 ∼1680	0	100
2	69	*Proline*	278 ∼590	0	100
		Color Intensity	1,9 ∼7,3	4	94,20

Tabela 18 – Análise de rotulação para base de dados Vinho com uso do algoritmo CHAID

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx . Conc. %**
0	62	*Proline*	600 ∼937	0	100
		Magnesium	70 ∼98	16	74,19
1	47	*Proline*	1065 ∼1680	14	70,21
2	69	*Proline*	278 ∼590	0	100
		Magnesium	100 ∼132	22	78,33

Tabela 19 – Análise de rotulação para base de dados Vinhos presente em Silva et al. (2021)

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Tx. Conc. %**
1	62	*Proline*	600.0 ∼937.0	0	100
2	47	*Proline*	953.5 ∼1680.0	0	100
3	69	*Proline*	278.0 ∼598.0	0	100

Para a base de dados Vinho o algoritmo de árvore de decisão que se saiu melhor pelo modelo proposto foi o C4.5, com taxa de concordância média de 99,67%, ficando somente um pouco abaixo (0,33%) do resultado alcançado na mesma base pelo modelo apresentado (SILVA et al., 2021) como é demonstrado na Tabela 19.

Abaixo demonstramos a interpretação dos rótulos gerados para cada grupo pelo modelo proposto com o algoritmo C4.5 como o algoritmo de árvore de decisão:

- para o *cluster* 0 o atributo *Proline* varia de 600 a 937, enquanto que o atributo *Alcalinity of ash* varia de 14,8 a 30,0.
- para o *cluster* 1 o atributo *Proline* varia de 970 a 1680.
- e por fim para o *cluster* 2 o atributo *Proline* varia de 278 a 590, e o atributo *Alcalinity of ash* varia de 10,6 a 25,0.

5.4 Base de dados Vidros

A base de dados Vidros possui um total de dados com 214 amostras contendo exemplos da análise química de 6 tipos diferentes de vidros. O problema é prever o tipo de classe com base na análise química. O estudo da classificação dos tipos de vidros foi motivado pela investigação criminológica. Na cena do crime, o vidro deixado pode ser usado como prova, se for corretamente identificado. (DUA; GRAFF, 2017). Possui 10 atributos com os seguintes componentes químicos: RI, Na, Mg, Al, Si, K, Ca, Ba e Fe. A base está desbalanceada da seguinte forma para cada classe: 76, 70, 29, 17, 13 ou 9 elementos.

Comparando o resultado alcançado pelo quatro algoritmos de árvore de decisão, de acordo com as tabelas 20, 21, 22 e 23, verifica-se que o melhor resultado foi alcançado com o uso do algoritmo C4.5 como algoritmo supervisionado, com uma taxa de concordância média de 99,28%. Este resultado ficou acima do resultado (LOPES et al., 2016) que foi de 95,54%.

Tabela 20 – Análise de rotulação para base de dados Vidros com uso do algoritmo ID3

Cluster	**# Elem**	**Rótulo**			
		Attrib.	**Faixa de Valores**	**# Erros**	**#Tx. Conc. % .**
0	35	Ca	8,6 ∼11,6	0	100
1	124	Mg	2,7 ~ 3,9	0	100
		K	0,3 ∼1,1	6	95,16
		Ca	7,0 ~ 9,4	0	100
2	5	K	1,4 ~ 6,2	0	100
		RI	1,516 ∼1,517	1	80
3	17	Mg	0,00 ∼1,88	0	100
		Ca	8,9 ∼12,5	0	100
4	26	Mg	0,0 ∼1,78	0	100
		K	0.0 ∼0,14	1	96,15
		Ca	7,5 ∼9,9	2	92,30
5	7	Mg	0,0 ∼0,0	0	100
		Ca	13,2 ∼16,1	0	100
		RI	1,518 ~ 1.517	1	85,71

Tabela 21 – Análise de rotulação para base de dados Vidros com uso do algoritmo C4.5

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	35	Ca	8,6 ∼11,6	0	100
		Mg	1,8 ~ 4,4	0	100
		K	0,0 ∼0,7	0	100
		Si	70,2 ∼72,7	0	100
1	124	Ca	7,0 ~ 9,4	0	100
		Mg	2,71 ∼3,98	0	100
		K	0,06 ∼1,10	0	100
		Na	10,7 ∼14,8	4	96,77
		Ba	0,0 ∼0,15	0	100
2	5	Ca	5,43 ~ 6,96	0	100
		K	1,46 ∼6,21	0	100
		Si	69,8 ∼72,8	0	100
		Al	1,8 ∼3,5	0	100
3	17	Ca	8,9 ∼12,5	0	100
		K	0,0 ∼0,9	1	94,11
4	26	Ca	6,4 ∼9,9	0	100
		K	0.0 ∼0,14	1	96,15
		RI	1,515 ∼1,526	0	100
5	7	Ca	13.2 ∼16,1	0	100
		K	0,0 ∼0,8	0	100
		Si	69,8 ∼73,2	0	100

Tabela 22 – Análise de rotulação para base de dados Vidros com uso do algoritmo CART

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx. Conc. %**
0	35	Mg	2,9 ∼ 4,4	8	77,14
		RI	1,511 ∼1,520	9	74,28
		Ba	0,0 ∼1,68	3	91,42
		Si	70,2 ∼72,3	2	94,28
1	124	Mg	2,71 ∼3,98	0	100
		K	0,35 ∼1,1	6	95,16
		Ca	7,08 ∼9,42	0	100
		Si	72,04 ∼73,81	4	96,77
2	5	Ca	5,43 ∼6,96	0	100
		K	1,76 ∼6,21	0	100
3	17	Mg	0,0 ∼1,88	0	100
		Ca	8,93 ∼12,5	0	100
		SI	72,18 ∼75,18	1	94,11
		Na	13,21 ∼15,79	2	88,23
4	26	Mg	0,0 ∼1,78	0	100
		K	0.0 ∼0,14	1	96,15
		Si	72,5 ∼75,41	0	100
5	7	Ca	13.2 ∼16,1	0	100

Tabela 23 – Análise de rotulação para base de dados Vidros com uso do algoritmo CHAID

Cluster	**# Elem**	**Rótulo**			
		Attrib	**Faixa de Valores**	**# Erros**	**#Tx Conc. %**
0	35	Mg	1,8 ∼ 4,4	0	100
		Ca	8,67 ∼11,6	1	97,14
		K	0,0 ∼0,59	1	97,14
1	124	Mg	2,71 ∼3,98	0	100
2	5	Ca	5,43 ∼6,96	0	100
3	17	Ca	8,93 ∼12,5	0	100
		K	0,0 ∼0,58	4	76,4
		Na	13,21 ∼15,79	2	88,23
4	26	Mg	0,0 ∼0,78	2	92,3
		K	0.0 ∼0,14	1	96,15
5	7	Mg	0,00 ∼0,00	0	100
		Ca	13,24 ∼16,19	0	100
		K	0,0 ∼0,58	1	85,71

5.5 Comparação entre os modelos

A tabela 25 demonstra a média de acertos dos demais trabalhos relacionados e do modelo proposto. Nota se que no caso da base *Íris* o resultado alcançado pelo modelo ficou abaixo de todos os outros da literatura.

Para a base Vinhos o modelo proposto alcançou os resultados um pouco abaixo

Tabela 24 – Análise de rotulação para base de dados Vidros encontrado em Lopes, Machado e Rabêlo (2013)

Cluster	**# Elem**	**Rótulo**		**Análise**	
		Attri.	**Faixa de Valores**	**# Erros**	**Acerto %**
1	74	Ba	0 ∼0.7875	0	100
		K	0 ∼1.5525	0	100
		Si	72.61 ∼74.01	2	97.29
		Na	12.3925 ∼14.055	3	95.94
2	5	Fe	0 ∼0.1275	0	100
		Ca	5.43 ∼8.12	0	100
3	19	K	0 ∼1.5525	0	100
		Ba	0 ∼0.7875	1	94.73
4	32	K	0 ∼1.5525	0	100
		Ba	0 ∼0.7875	1	96.87
		Ca	8.12 ∼10.81	1	96.87
5	56	Ba	0 ∼0.7875	0	100
		K	0 ∼1.5525	0	100
		Na	12.3925 ∼14.055	2	94.62
		Al	1.0925 ∼1.895	4	92.85
		Mg	3.3675 ∼4.49	6	89.28
6	28	Fe	0 ∼0.1275	0	100
		K	0 ∼1.5525	1	96.42

Tabela 25 – Comparativo entre taxa de concordância média entre as Abordagens de Rotulação

Modelo	**Técnicas Utilizadas**	**Base Íris**	**Base Vinhos**	**Base Sementes**	**Base Vidros**
Lopes (2014)	Discretização e Classificação com RNA	94,14%	X	89%	95,54%
Ribeiro (2016)	Lógica Fuzzy com Grau de Pertinência	98%	X	89%	98,43%
Filho et al. (2020)	Distância Euclidiana com Grau de Seleção	92,52%	X	92,86%	99,08%
Moura (2020)	Discretização com CAIM	89,47%	100%	97,19%	97,93%
Silva et al. (2021)	Regressão com RBF	94,33%	100%	89,33%	88,5%
Alencar (2022)	Filtro de Seleção de Atributos	98,33%	91,66%	87,59%	81,73%
Modelo proposto	Árvore de decisão	86,46%	99,67%	92,23%	99,28%

de Silva et al. (2021), no entanto deve-se enfatizar que o objetivo dos métodos de rotulação de grupos não visam tão somente apresentar os melhores resultados sob o ponto de vista das taxas de concordância, mas também apresentar novos e válidos rótulos, como nesse caso onde foram apresentados rótulos também para o atributo *Magnesium* e não somente para o atributo *Proline*, como apresentado no trabalho anterior. No caso da base Sementes o modelo proposto ficou abaixo dos resultados alcançado no trabalho Moura, Veras e Machado (2022) e Filho et al. (2020). Para base Vidros o modelo proposto obteve resultado de taxa de concordância média acima de todos os outros modelos da literatura, esse resultado ode ser explicado pela capacidade do algoritmo C4.5 de lhe dar bem com bases de dados com dimensionalidade alta.

Ao comparar o modelo de rotulação automática de grupos proposto com outros modelos da literatura, alguns fatores devem ser considerados:

1. limitação do número de faixas:

O uso de árvores binárias limita a quantidade de faixas a duas. Essa escolha visa evitar árvores muito complexas e de difícil interpretação. Foram realizados também testes com árvores *Multiway*, no entanto os rótulos gerados resultaram em taxas de concordância muito abaixo das geradas pelas das árvores binárias.

2. baseamento no atributo classe:

Em todos os trabalhos relacionados, o número de grupos foi escolhido com base no atributo classe de cada base de dados. Não foram realizados testes em bases sem atributos classe definido.

3. interpretabilidade dos rótulos:

Em muitos casos, como na base "vidros", a quantidade de atributos nos rótulos foi muito grande, o que gerou rótulos de difícil interpretação.

6
Conclusões e Trabalhos futuros

6.1 Conclusões

Os modelos de Rotulação de grupos fornecem auxilio ao especialista na interpretação dos dados dando a estes uma definição resumida das principais características dos grupos (SILVA et al., 2021).O estudo comparativo realizado neste trabalho, que se baseou em um modelo de rotulação automática de grupos utilizando árvores de decisão, testou os algoritmos ID3, C4.5, CART e CHAID. Os resultados demonstraram que a escolha do algoritmo impacta significativamente na performance do modelo, variando de acordo com a natureza do conjunto de dados utilizado.

Melhores resultados do algoritmo CHAID para bases Íris e Sementes, podem ser explicado pela capacidade do algoritmo CHAID ser mais eficiente em base de dados pequenas. O CHAID utiliza o teste *chi-squared* para selecionar as variáveis mais relevantes, o que o torna menos sensível a *outliers* e dados ruidosos. Isso é particularmente importante em bases como Iris e Sementes, que possuem um número relativamente pequeno de registros. O ID3, C4.5 e CART, por outro lado, utilizam a ganho de informação para selecionar as variáveis, o que pode torná-los mais sensíveis a outliers e dados ruidosos, especialmente em bases pequenas. O CHAID tende a gerar árvores de decisão menos complexas do que o ID3, C4.5 e CART. Isso pode ajudar a reduzir o risco de *overfitting* em bases como Iris e Sementes, onde o número de registros é menor e o risco de overfitting é maior.

O melhor desempenho do algoritmo C4.5 para as bases Vinhos e Vidros pode ser explicado por sua capacidade de poda na árvores de decisão, o que o possibilita a evitar o *overfitting* e auxilia na produção modelos mais generalizáveis. O algoritmo ainda trata melhor a questão do desbalanceamentos de classes e da alta dimensionalidade das bases como é o caso da base Vidros.

Os resultados demonstram a viabilidade do modelo de rotulação automática de grupos proposto, evidenciando a importância da escolha do algoritmo de árvore de decisão. A seleção do algoritmo impacta na acurácia do modelo, exigindo análise criteriosa da natureza dos dados e das características de cada algoritmo.

Propostas para trabalhos futuros:

1. ampliação da base de testes: realizar testes com um número maior de bases de dados para traçar um paralelo entre a natureza dos dados e a escolha do

algoritmo de árvore de decisão.

2. pesquisar modos de melhorar a interpretabilidade de modelos de conjuntos de árvores de decisão: uma vez que estes são mais precisos que modelo de árvores de decisão simples utilizados neste trabalho o que pode levar a geração de melhores rótulos.

Referências

AEBERHARD, S.; COOMANS, D.; VEL, O. D. Comparative analysis of statistical pattern recognition methods in high dimensional settings. *Pattern Recognition*, Elsevier, v. 27, n. 8, p. 1065–1077, 1994. Citado na página 38.

AGGARWAL, C. C.; REDDY, C. K. *Data clustering: algorithms and applications*. Boca Raton, Fla: CRC Press, 2014. (Chapman Hall/CRC data mining and knowledge discovery series). ISBN 9781466558212. Citado na página 5.

ALENCAR, M. dos R. B. *Rotulação Automática de Grupos Através do Uso de Filtros de Ganho de Informação de Atributos*. Dissertação (Mestrado) — Universidade Federal do Piauí - CCN, 2022. Citado 2 vezes nas páginas 2 e 25.

ARAUJO, F. N. de et al. Automatic cluster labeling based on phylogram analysis. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.]: IEEE, 2018. p. 1–8. ISBN 9781509060146. ISSN 2161-4407. Citado na página 22.

ARTHUR, D.; MANTHEY, B.; RöGLIN, H. Smoothed analysis of the k-means method. *Journal of the ACM*, ACM, NEW YORK, v. 58, n. 5, p. 1–31, 2011. ISSN 0004-5411. Citado na página 8.

ATILGAN, C.; NASIBOV, E. A memory efficient distributed fuzzy joint points clustering algorithm. In: *2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*. [S.l.: s.n.], 2016. p. 1–5. Citado na página 5.

BARBER, D. *Bayesian reasoning and machine learning*. [S.l.]: Cambridge University Press, 2012. Citado na página 5.

BASAK, J.; KRISHNAPURAM, R. Interpretable hierarchical clustering by constructing an unsupervised decision tree. *IEEE Transactions on Knowledge and Data Engineering*, v. 17, n. 1, p. 121–132, 2005. Citado 2 vezes nas páginas 3 e 10.

BASAK, J.; KRISHNAPURAM, R. Interpretable hierarchical clustering by constructing an unsupervised decision tree. *IEEE Transactions on Knowledge and Data Engineering*, v. 17, p. 121–132, 2005. Citado 2 vezes nas páginas 17 e 20.

BERKHIN, P. A survey of clustering data mining techniques. In: *Grouping multidimensional data*. [S.l.]: Springer, 2006. p. 25–71. Citado na página 8.

BERTSIMAS, D.; DUNN, J. Optimal classification trees. *Machine Learning*, Springer, v. 106, p. 1039–1082, 2017. Citado na página 18.

BERTSIMAS, D.; ORFANOUDAKI, A.; WIBERG, H. M. Interpretable clustering: an optimization approach. *Machine Learning*, v. 110, p. 89–138, 2020. Citado 3 vezes nas páginas 18, 20 e 28.

BREIMAN, L. et al. Classification and regression trees. belmont, ca: Wadsworth. *International Group*, v. 432, n. 151-166, p. 9, 1984. Citado na página 15.

CHEN, J. *Interpretable clustering methods*. Tese (Doutorado) — Northeastern University, 2018. Citado na página 17.

DIMOTIKALIS, Y. et al. *Applied Modeling Techniques and Data Analysis 1: Computational Data Analysis Methods and Tools*. Newark: John Wiley Sons, Incorporated, 2021. ISBN 9781786306739. Citado na página 1.

DUA, D.; GRAFF, C. *UCI Machine Learning Repository*. 2017. Disponível em: <http://archive.ics.uci.edu/ml>. Citado 4 vezes nas páginas 28, 38, 40 e 42.

EVETT, I. W.; SPIEHLER, E. J. Rule induction in forensic science. In: *Knowledge Based Systems*. [S.l.: s.n.], 1989. p. 152–160. Citado na página 38.

FACELI, K. et al. Inteligência artificial: uma abordagem de aprendizado de máquina. 2021. Citado 2 vezes nas páginas 5 e 11.

FILHO, F. I. et al. Group labeling methodology using distance-based data grouping algorithms. *Revista de Informática Teórica e Aplicada*, v. 27, n. 1, p. 48–61, 2020. ISSN 0103-4308. Citado 4 vezes nas páginas 2, 23, 25 e 46.

FISHER, R. A. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, Wiley Online Library, v. 7, n. 2, p. 179–188, 1936. Citado na página 28.

GABIDOLLA, M.; CARREIRA-PERPIÑÁN, M. Á. Optimal interpretable clustering using oblique decision trees. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022. Citado 3 vezes nas páginas 17, 19 e 20.

GÉRON, A. Mãos à obra: aprendizado de máquina com scikit-learn. *Keras & TensorFlow: Conceitos, ferramentas e técnicas para a construção de sistemas inteligentes.[S. l.: sn]*, 2021. Citado na página 5.

GÉRON, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. [S.l.]: "O'Reilly Media, Inc.", 2022. Citado na página 10.

HAJJEJ, F. et al. A comparison of decision tree algorithms in the assessment of biomedical data. *BioMed Research International*, Hindawi, v. 2022, 2022. Citado na página 14.

HAN, J.; KAMBER, M.; PEI, J. Data mining: concepts and. *Techniques (3rd ed), Morgan Kauffman*, 2011. Citado na página 12.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. 3rd. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011. ISBN 0123814790. Citado 5 vezes nas páginas 1, 2, 5, 7 e 9.

HARA, S.; HAYASHI, K. Making tree ensembles interpretable. *arXiv preprint arXiv:1606.05390*, 2016. Citado na página 27.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999. (International edition). ISBN 9780139083853. Disponível em: <https://books.google.com.br/books?id=M5abQgAACAAJ>. Citado na página 4.

KASS, G. V. An exploratory technique for investigating large quantities of categorical data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, Wiley Online Library, v. 29, n. 2, p. 119–127, 1980. Citado na página 16.

KULCZYCKI, P.; CHARYTANOWICZ, M. A complete gradient clustering algorithm. In: SPRINGER. *International Conference on Artificial Intelligence and Computational Intelligence*. [S.l.], 2011. p. 497–504. Citado na página 38.

LIBBRECHT, M. W.; NOBLE, W. S. Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, Nature Publishing Group, v. 16, n. 6, p. 321–332, 2015. Citado na página 4.

LIMA, B. V. A. de; MACHADO, V. P.; LOPES, L. A. Automatic labeling of social network users scientia. net through the machine learning supervised application. *Social Network Analysis and Mining*, Springer, v. 5, p. 1–10, 2015. Citado na página 22.

LOPES, L. A. et al. Automatic labelling of clusters of discrete and continuous data with supervised machine learning. *Knowledge-Based Systems*, Elsevier, v. 106, p. 231–241, 2016. Citado 9 vezes nas páginas 10, 1, 2, 8, 9, 22, 37, 40 e 42.

LOPES, L. A.; MACHADO, V. P.; RABÊLO, R. A. L. Automatic labeling of groupings through supervised machine learning. In: . [S.l.: s.n.], 2013. Citado 10 vezes nas páginas 9, 11, 21, 22, 23, 24, 25, 27, 39 e 45.

LOPES, L. A.; MACHADO, V. P.; RABELO, R. D. A. L. Automatic cluster labeling through artificial neural networks. In: *2014 International Joint Conference on Neural Networks (IJCNN)*. [S.l.]: IEEE, 2014. p. 762–769. ISBN 1479914843. ISSN 2161-4393. Citado 2 vezes nas páginas 2 e 22.

MACHADO, V. P.; RIBEIRO, V.; RABÊLO, R. Rotulacao de grupos utilizando conjuntos fuzzy. In: *XII Simposio Brasileiro de Automacao Inteligente-SBAI*. [S.l.: s.n.], 2015. p. 355–360. Citado 4 vezes nas páginas 2, 22, 24 e 25.

MACQUEEN, J. Classification and analysis of multivariate observations. In: *5th Berkeley Symp. Math. Statist. Probability*. [S.l.: s.n.], 1967. p. 281–297. Citado na página 8.

MILANOVIĆ, M.; STAMENKOVIĆ, M. Chaid decision tree: Methodological frame and application. *Economic Themes*, v. 54, n. 4, p. 563–586, 2016. Citado na página 16.

MITCHELL, T. M. *Machine learning*. [S.l.]: McGraw-hill New York, 1997. v. 1. Citado 4 vezes nas páginas 9, 4, 10 e 11.

MOURA, M.; VERAS, R.; MACHADO, V. Caibal: Cluster-attribute interdependency based automatic labeler. In: *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*. New York, NY, USA: Association for Computing Machinery, 2022. (SAC '22), p. 1109–1116. ISBN 9781450387132. Disponível em: <https://doi-org.ez117.periodicos.capes.gov.br/10.1145/3477314.3507140>. Citado 4 vezes nas páginas 2, 24, 25 e 46.

PRIYAM, A. et al. Comparative analysis of decision tree classification algorithms. *International Journal of current engineering and technology*, Citeseer, v. 3, n. 2, p. 334–337, 2013. Citado na página 14.

QUINLAN, J. R. Induction of decision trees. *Machine learning*, Springer, v. 1, n. 1, p. 81–106, 1986. Citado 2 vezes nas páginas 12 e 14.

QUINLAN, R. *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993. Citado 2 vezes nas páginas 12 e 14.

RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. [S.l.]: Pearson, 2016. ISBN 9781292153964. Citado 2 vezes nas páginas 1 e 4.

SAFAVIAN, S. R.; LANDGREBE, D. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, IEEE, v. 21, n. 3, p. 660–674, 1991. Citado na página 11.

SERENGIL, S. I. *ChefBoost: A Lightweight Boosted Decision Tree Framework*. [S.l.]: Zenodo, 2021. Https://doi.org/10.5281/zenodo.5576203. Citado na página 29.

SHIRKHORSHIDI, A. S.; AGHABOZORGI, S.; WAH, T. Y. A comparison study on similarity and dissimilarity measures in clustering continuous data. *PloS one*, Public Library of Science San Francisco, CA USA, v. 10, n. 12, p. e0144059, 2015. Citado na página 2.

SILVA, L. E. S. et al. Using regression error analysis and feature selection to automatic cluster labeling. In: SPRINGER. *Progress in Artificial Intelligence: 20th EPIA Conference on Artificial Intelligence, EPIA 2021, Virtual Event, September 7–9, 2021, Proceedings 20*. [S.l.], 2021. p. 376–388. Citado 7 vezes nas páginas 10, 2, 24, 25, 42, 46 e 47.

SINGH, S.; GUPTA, P. Comparative study id3, cart and c4. 5 decision tree algorithm: a survey. *International Journal of Advanced Information Science and Technology (IJAIST)*, v. 27, n. 27, p. 97–103, 2014. Citado 2 vezes nas páginas 14 e 15.

SOUSA, J. M. de et al. Automatic labelling of clusters with discrete and continuous data using supervised machine learning. In: *2016 35th International Conference of the Chilean Computer Science Society (SCCC)*. [S.l.: s.n.], 2016. p. 1–10. Citado na página 22.

TEODORO, G. D.; MONACI, M.; PALAGI, L. Unboxing tree ensembles for interpretability: a hierarchical visualization tool and a multivariate optimal re-built tree. *EURO Journal on Computational Optimization*, Elsevier, v. 12, p. 100084, 2024. Citado na página 27.

TIMOFEEV, R. Classification and regression trees (cart) theory and applications. *Humboldt University, Berlin*, v. 54, 2004. Citado na página 15.