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

Uso de Algoritmos de Aprendizado de Máquina Supervisionado para Rotulação de Dados

Tarcísio Franco Jaime

Número de Ordem PPGCC: M001

Teresina-PI, Junho de 2018

Tarcísio Franco Jaime

Uso de Algoritmos de Aprendizado de Máquina Supervisionado para Rotulação de Dados

Qualificação 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: Vinicius Ponte Machado

Teresina-PI

Junho de 2018

Tarcísio Franco Jaime
Uso de Algoritmos de Aprendizado de Máquina Supervisionado para Rotulação de Dados/ Tarcísio Franco Jaime. – Teresina-PI, Junho de 2018-
65 p. : il.

Orientador: Vinicius Ponte Machado

Qualificação (Mestrado) – Universidade Federal do Piauí – UFPI
Centro de Ciências da Natureza
Programa de Pós-Graduação em Ciência da Computação, Junho de 2018.

1. Rotulação. 2. Algoritmos Supervisionados. 3. CART. 4. Naive Bayes. I. Dr. Vinicius Ponte Machado. II. Universidade Federal do Piauí. III. Uso de Algoritmos de Aprendizado de Máquina Supervisionado para Rotulação de Dados.

CDU 02:141:005.7

Tarcísio Franco Jaime

Uso de Algoritmos de Aprendizado de Máquina Supervisionado para Rotulação de Dados

Qualificação 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.

Teresina-PI, 21 de junho de 2018:

Vinicius Ponte Machado
Orientador

Raimundo Santos Moura

Erico Meneses Leão

Teresina-PI
Junho de 2018

Resumo

Com o avanço da tecnologia, cada vez mais equipamentos estão se conectando nas redes, gerando fluxos e processamento de dados. Com isso, mais algoritmos de aprendizado de máquina estão sendo estudados para extraírem informações relevantes desses grandes volumes. Com o grande aumento desse fluxo de dados, a interpretação destes pode ser prejudicada, sendo o grau de dificuldade proporcional a esse crescimento. É nesse contexto que essa pesquisa atua, pois alguns algoritmos de aprendizado de máquina criam grupos de dados que possuem algumas características. Neste trabalho realizou-se uma pesquisa científica com o objetivo de identificar nesses grupos quais são os atributos mais significativos junto aos valores que mais se repetem a ponto de representar o grupo, denominando-se essa técnica de rotulação. Dessa forma, esta pesquisa utiliza a técnica algoritmos supervisionados, já implementados por um software de cálculo numérico (MATLAB), em que se pretende rotular grupos já criados em diferentes bases de dados, exibindo um resultado em porcentagem de acordo com o número de registros que são representados pelo rótulo criado.

Palavras-chaves: grupos. rotulação. aprendizado supervisionado.

Abstract

Keywords: cluster. rotulação.

Lista de ilustrações

Figura 1 – Hipóteses ajustadas – Função h próxima da função f real 6
Figura 2 – Exemplo de Fluxograma de árvore: R1 a R5 são as folhas relacionadas de acordo com as respostas sim ou não dos nós 6
Figura 3 – Exemplo do funcionamento do algoritmo KNN 11
Figura 4 – Exemplos de técnicas diferentes utilizada por algoritmos para dividir em grupos 13
Figura 5 – Ponto de Corte (R-1); 2(R) significa o valor de R=2; 2 - 1 = 1 14
Figura 6 – Discretização EWD 15
Figura 7 – Discretização EWD de acordo com a amostra da tabela 3 16
Figura 8 – Discretização EFD 17
Figura 9 – Discretização EFD de acordo com a amostra da tabela 3 18
Figura 10 – Modelo de (LOPES et al., 2016) 20
Figura 11 – Modelo de Resolução Proposto 25
Figura 12 – Exemplo da técnica de correlação aplicada ao atributo, atr1, sendo classe 26
Figura 13 – Exemplo da técnica de correlação aplicada aos atributos atr1, atr2 e atr3. 26
Figura 14 – Montagem da Matriz de Atributos Importantes 27
Figura 15 – Discretização de atributos utilizando EFD com R = 3 (Figura adaptada de (LOPES et al., 2016)) 29
Figura 16 – Resultado dos Algoritmos 31
Figura 17 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove** 39
Figura 18 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove** 39
Figura 19 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove** 40
Figura 20 – Gráfico da disposição de elementos da Base Iris entre os eixos **petal-length** e **petalwidth** 44
Figura 21 – Gráfico da disposição de elementos da Base Wine entre os eixos **FnF** e **Alcool** 56
Figura 22 – Disposição dos elementos da base wine na pespectiva dos atributos:Alcool e FnF 57

Lista de tabelas

Tabela 1 – Base de exemplo onde exibe através do atributo Decisão, se existe, ou não condição de jogo perante as outras características (Aspecto, temperatura, Umidade e Vento) 10
Tabela 2 – Tabela em ordem crescente das distâncias euclidianas encontradas do objeto a que se deseja classificar para os outros objetos da amostra, de acordo com as setas da figura 3 12
Tabela 3 – Amostra de dados para exemplificar a discretização EWD e EFD . . . 15
Tabela 4 – Base de Dados Modelo 28
Tabela 5 – Base de Dados Modelo Discretizada 30
Tabela 6 – Valores das faixas com R=3 da Base de Dados Modelo 30
Tabela 7 – Resultado da rotulação com o algoritmo Naive Bayes 36
Tabela 8 – Resultado da aplicação do algoritmo CART 37
Tabela 9 – Resultado da aplicação do algoritmo KNN 38
Tabela 10 – Resultado da aplicação do algoritmo Naive Bayes 42
Tabela 11 – Matriz de Atributos Importantes do algoritmo Naive Bayes na base Iris 42
Tabela 12 – Resultado da aplicação do algoritmo CART 43
Tabela 13 – Resultado da aplicação do algoritmo KNN 43
Tabela 14 – Resultado da aplicação do algoritmo Naive Bayes 46
Tabela 15 – Resultado da aplicação do algoritmo CART 48
Tabela 16 – Resultado da aplicação do algoritmo KNN 49
Tabela 17 – *Cluster* 5 - Louças de mesa - Base Glass 50
Tabela 18 – Acurácia média da Base Glass por *clusters* 51
Tabela 19 – Resultado da aplicação do algoritmo Naive Bayes 52
Tabela 20 – Resultado da aplicação do algoritmo CART 53
Tabela 21 – Resultado da aplicação do algoritmo KNN 53
Tabela 22 – Amostra da Base de Dados Wine com somente três atributos pertencentes ao *cluster* 1 do algoritmo Naive Bayes 55

Lista de abreviaturas e siglas

ANN	Artificial Neural Networks
EWD	Equal Width Discretization
EFD	Equal Frequence Discretization
CART	Classification and Regression Trees
RNA	Redes Neurais Artificiais
GP	Grau de Pertinência
GS	Grau de Seleção
IGS	Incremento do Grau de Seleção
SVM	Support Vector Machine
TEDA	Typicality and Eccentricity Data Analytics

Sumário

1 INTRODUÇÃO 1

2 REFERENCIAL TEÓRICO 4
2.1 Aprendizado de Máquina 4
2.1.1 Aprendizado Supervisionado 5
2.1.1.1 Algoritmo Classification And Regression Trees - CART 6
2.1.1.2 Algoritmo Naive Bayes 8
2.1.1.3 Algoritmo k-Nearest Neighbor - KNN 10
2.1.2 Aprendizado Não-Supervisionado 12
2.2 Discretização 13
2.2.1 Equal Weight Discretization - EWD 14
2.2.2 Discretização por Frequência Iguais - EFD 16
2.3 Trabalhos Correlatos 18

3 METODOLOGIA 23
3.1 Rotulação de Cluster 23
3.2 O Modelo de Resolução 24
3.3 Técnica de Correlação entre Atributos através de Algoritmos Supervisionados 26
3.4 Exemplo 28
3.4.1 Processo (I) - Discretização 29
3.4.2 Processo (II) - Algoritmos Supervisionados 30
3.4.3 Processo (III) - Rotulação 31

4 RESULTADOS E DISCUSSÕES 34
4.1 Implementação 35
4.2 Base de Dados 1 - Seeds - Identificação de Tipos de Semente . . . 35
4.2.1 Naive Bayes 36
4.2.2 Classification and Regression Trees - CART 37
4.2.3 K-Nearest Neighbor - KNN 37
4.2.4 Comparativo entre Algoritmos na Base de Dados Seeds 38
4.3 Base de Dados 2 - Iris - Identificação de Tipos de Plantas 41
4.3.1 Naive Bayes 41
4.3.2 Classification and Regression Trees - CART 42
4.3.3 K-Nearest Neighbor - KNN 43
4.3.4 Comparativo entre Algoritmos na Base de Dados Iris 43

4.4 Base de Dados 3 - Glass - Identificação de Tipos de Vidros 44
4.4.1 Naive Bayes 45
4.4.2 Classification and Regression Trees - CART 47
4.4.3 K-Nearest Neighbor - KNN 48
4.4.4 Comparativo entre Algoritmos na Base de Dados Glass 50
4.5 Base de Dados 4 - Wine - Dados de Reconhecimento de vinhos . . 51
4.5.1 Naive Bayes 52
4.5.2 Classification and Regression Trees - CART 52
4.5.3 K-Nearest Neighbor - KNN 53
4.5.4 Comparativo entre Algoritmos na Base de Dados Wine 53
4.6 Discussões 57

5 CONCLUSÕES, TRABALHOS FUTUROS 58
5.1 Conclusão 58
5.2 Trabalhos Futuros 59

REFERÊNCIAS 60

APÊNDICES 63

APÊNDICE A – PRIMEIRO APÊNCICE 64

APÊNDICE B – PERCEBA QUE O TEXTO DO TÍTULO DESSE SEGUNDO APÊNDICE É BEM GRANDE 65

1 Introdução

Agrupamento de dados, ou clustering, é o termo usado para identificar dois ou mais objetos pertencentes ao mesmo grupo que compartilham um conceito em comum (KUMAR et al., 2013). Cluster é um termo bastante pesquisado no aprendizado não supervisionado (subárea do aprendizado de máquina) e aplicado em vários contextos, como segmentação de imagens, recuperação de informação e reconhecimento de objetos. Os algoritmos de agrupamento, conforme (KUMAR et al., 2013), são aplicados em diferentes campos: Biologia (classificação de plantas e animais); Marketing (encontrar grupos de clientes com comportamentos semelhantes); Planejamento de cidades (identificação de casas de acordo com seu tipo, valor e localização geográfica), entre outros.

Com a popularização da internet e mídias sociais, cada vez mais dados são processados, transportados e produzidos. É nesse cenário, com grandes volumes de dados, que não só a formação de grupos ganha importância, mas também a sua compreensão, pois a interpretação dos grupos fornecerá informações úteis para análise desses clusters.

O grau de escalabilidade dos dados aumenta gradativamente no decorrer dos anos e, embora os estudos sobre o problema de agrupamento de dados estejam avançados, fica cada vez mais complexo entender como são formados esses clusters em razão do número crescente de grupos criados. Quanto maior é o número de grupos produzidos, mais difíceis são suas interpretações.

Diante desse contexto é que se extrai a temática desta proposta de mestrado - “Uso de algoritmos de aprendizado de máquina supervisionado para rotulação de dados”. O estudo em questão dedica-se à aplicabilidade de algoritmos supervisionados com paradigmas diferentes em bases de dados distintas, a fim de definir a tupla atributo/valor de maior importância nos clusters, determinando um significado para estes clusters (rotulação).

A rotulação dita neste trabalho segue a própria definição da palavra, que serve para informar sobre algo. Então, a partir de um grupo de dados, seria possível destacar neste grupo uma informação que o represente e uma forma seria encontrar por meio de técnicas uma tupla: atributo(s) e faixa(s), em que o atributo selecionado seria o que teria maior relevância no grupo, no sentido de representá-lo, e a faixa de valor escolhida seria a que mais tivesse ocorrência nos valores do atributo. Poderá também haver no grupo mais de um atributo com sua respectiva faixa representando o rótulo.

O termo rotulação, neste trabalho, segue a definição conforme (LOPES et al., 2016):

Definição 1 *Dado um conjunto de clusters* $C = \{c_1, ..., c_k | K \geqslant 1\}$*, de modo que cada cluster contém um conjunto de elementos* $c_i = \{\vec{e}_1, .., \vec{e}_{n^{(c_i)}} | n^{(c_i)} \geqslant 1\}$ *que podem ser representa-*

dos por um vetor de atributos definidos em $\mathbb{R}^m$ *e expresso por* $\vec{e}^{c_i} = (a_1, .., a_m)$ *e ainda que com* $c_i \cap c_{i'} = \emptyset$ *com* $1 \leqslant i, i \leqslant K$ *e* $i \neq i'$*; o objetivo consiste em apresentar um conjunto de rótulos* $R = \{r_{c1}, ..., r_{ck}\}$*, no qual cada rótulo específico é dado por um conjunto de pares de valores, atributo e seu respectivo intervalo,* $r_{ci} = \{(a_1, [p_1, q_1]), ..., (a_{m^{(c_i)}},]p_{m^{(c_i)}}, q_{m^{(c_i)}}])\}$ *capaz de melhor expressar o cluster* c_i *associado.*

- K *é o número de clusters;*
- c_i *é o i-ésimo cluster;*
- n^{c_i} *é o número de elementos do cluster* c_i*;*
- $\vec{e}_{j^{(c_i)}}$ *se refere ao j-ésimo elemento pertencente ao cluster* c_i*;*
- m *é a dimensão do problema;*
- r_{c_i} *é o rótulo referente ao cluster* c_i*;*
- $]p_{m^{(c_i)}}, q_{m^{(c_i)}}]$ *representa o intervalo de valores do atributo* $a_{m^{(c_i)}}$*, onde* $p_{m^{(c_i)}}$ *é o limite inferior e* $q_{m^{(c_i)}}$ *é o limite superior;*

A formação do problema desta pesquisa nasce a partir do trabalho realizado por (LOPES et al., 2016), que se dedicou a estudar a possibilidade de realização de rotulação automática de grupos, utilizando, para isso, dois algoritmos: i) Um para realizar a formação de grupos por meio de algoritmo não supervisionado (K-means); e ii) utiliza o algoritmo supervisionado (Redes Neurais Artificiais - RNA) para fazer a rotulação de grupos. Assim, partindo do estudo já realizado, este trabalho se dedica a realizar rotulação de grupos de dados a partir de outros algoritmos supervisionados não testados e realizando um comparativo entre eles.

Nesta pesquisa foi aferida a acurácia de cada resultado por meio do percentual de acertos dos atributos que são representados pelos rótulos gerados, sendo essa acurácia possível em virtude das bases de dados escolhidas já serem classificadas, possibilitando o comparativo dos rótulos encontrados com a classificação da base de dados. É importante destacar que um pré-requisito para utilização de uma base para teste é que esta base contenha registros já pertencentes a algum grupo. Isto posto, no desenvolvimento deste trabalho não há preocupação na criação de grupos e sim na rotulação dos mesmos, isto é, compreender os grupos de dados já formados.

Quando se analisam grupos que já estão formados, sabe-se que esses grupos existem, pois há uma correlação das características pelos quais seus dados se mantêm agrupados. Acontece que, com grandes números de grupos sendo criados, isso acaba por não deixar visível qual característica se apresenta mais significativa dentro desses grupos. Tem-se na rotulação a intenção de definir algum significado para eles, gerando um tipo de rótulo, $R = \{r_{c1}, ..., r_{ck}\}$, para melhor expressar o cluster c_i associado (Definição 1).

Tecnicamente, a informação do rótulo aplicada no cluster pode ajudar na tomada de decisão em algum contexto. A exemplo disso, supõe-se uma situação empregada na área urbana, onde pessoas circulam, e imagina-se que os dados de controle de seus celulares estão sendo capturados pelas células das torres e gravados em uma base de dados pelas operadoras. Uma vez em posse desses dados, são criados clusters, podendo ser aplicada rotulação nestes grupos e por meio disso pode-se personalizar alguns serviços para os grupos já formados.

Seguindo o exemplo dos dados capturados do celular, caso o rótulo (r_{c_i}) de um cluster (c_i) fosse o atributo localização e os valores desse atributo escolhido para compor o rótulo fossem as coordenadas geográficas, definir-se-ia o tipo de localização. Logo, percebe-se que os participantes desse grupo possuem a característica de frequentar alguma localização em comum. A interpretação deste rótulo poderá implicar uma tomada de decisão personalizada para o grupo, objetivando otimizar um problema.

O trabalho será disposto em cinco capítulos, já inclusas a Introdução e Conclusão nos capítulos 1 e 5 respectivamente. O Referencial Teórico, abordado no capítulo 2, esclarece as tecnologias utilizadas nesta pesquisa, que é dividida em três seções. Inicialmente na seção 2.1, em-se uma explanação sobre aprendizado de máquina e quais os aprendizados indutivos são mais relevantes para este trabalho; ademais, a explicação dos algoritmos supervisionados utilizados para fazer rotulação de dados. Já na seção 2.2 é realizada a divisão das faixas de valores de cada atributo, chamada de discretização. E logo na seção 2.3 são apresentadas pesquisas já consolidadas referentes a assuntos como aprendizado de máquina, classificação, agrupamentos e rotulação de dados.

No capítulo 3 é abordada a definição do problema da pesquisa. A partir dessa definição, um modelo de resolução é definido e apresentado um fluxograma exibindo os processos a serem seguidos. Logo na seção 3.3é demonstrado o funcionamento da técnica de correlação entre atributos. E na seção 3.4 uma base de dados fictícia é utilizada para exemplificar a execução dos processos do modelo de resolução nas seguintes etapas: discretização da base de dados, a aplicação do algoritmo supervisionado e resultado da rotulação. No capítulo 4,os resultados são separados por base de dados. Em cada seção referente a uma base de dados testada são criadas duas subseções referentes aos algoritmos utilizados. Cada algoritmo apresenta uma tabela com informações desde o número do cluster, rótulos, relevância do atributo até acurácia do rótulo no cluster. É também exibida uma tabela possuindo os valores de relevância dos atributos por cluster, posto que os valores desta tabela servirão de apoio para entender o quão os atributos estão bem correlacionados.

2 Referencial Teórico

Para se compreender a temática proposta, este capítulo abordará o conteúdo base deste trabalho dividido em 3 seções: Aprendizado de Máquina, Discretização e Trabalhos Correlatos.

Essa primeira seção discorrerá sobre aprendizado de máquina e os aprendizados indutivos, embora cada aprendizado tenha sua importância. Será dada maior ênfase ao aprendizado supervisionado, em virtude da utilização de algoritmos de aprendizado supervisionado na rotulação de grupos. Já na seção 2.2 se dissertará sobre as técnicas de discretizações adotadas nesta pesquisa, a qual oferece grande contribuição para os resultados gerados, e ganhando, assim, uma seção própria para explanação do funcionamento dessas técnicas. Na última seção serão abordados trabalhos com mesmas características desta pesquisa, adicionando conhecimento ao tema.

2.1 Aprendizado de Máquina

A aprendizagem de máquina, diferente das metodologias tradicionais de implementação, utiliza sua experiência anterior para melhorar suas respostas a partir de problemas em determinadas áreas.

"Um programa de computador aprende com a experiência E em relação a alguma classe de tarefas T e medida de desempenho P, se seu desempenho em tarefas em T, conforme medido por P, melhora com a experiência E" (MITCHELL, 1997, p. 2).

O aprendizado de máquina corresponde a algoritmos capazes de aprender automaticamente através de determinados exemplos ou comportamentos. Esse aprendizado automático preenche algumas lacunas no desenvolvimento de programas, posto que não é possível simplesmente exigir do projetista implementar melhorias em um sistema, de forma que ele esteja robusto bastante para lidar com todas as situações (RUSSEL; NORVIG, 2013), pois seria impossível um programador antecipar todas as situações possíveis de implementação.

Utilizando a ideia acima, uma vez inserida uma foto no banco de dados e determiná-la como masculina, nesse momento, estar-se-á fazendo uma classificação desse novo registro (nova foto). Uma vez com a base de dados classificada, pode-se utilizar algoritmos para predizer um novo registro e defini-lo como masculino ou feminino. Predizer uma determinada condição dependerá não só da base de dados como também do algoritmo utilizado para fazer essa classificação. Alguns exemplos de algoritmos são: RNA, K-Nearest Neighbor - KNN, Suport Vector Machine – SVM, etc. A escolha apropriada do algoritmo se dará por

meio de métricas que avaliarão seu desempenho e a melhor servirá de parâmetro para a escolha do algoritmo apropriado para aquele problema de classificação de dados.

Segundo (MOHRI et al., 2012) o aprendizado de máquina possui abordagens diferentes. São elas: aprendizado supervisionado, aprendizado não supervisionado, aprendizado semisupervisionado, aprendizado por reforço. Todavia, nesta pesquisa serão comentadas somente as abordagens de referências específicas utilizadas neste trabalho.

2.1.1 Aprendizado Supervisionado

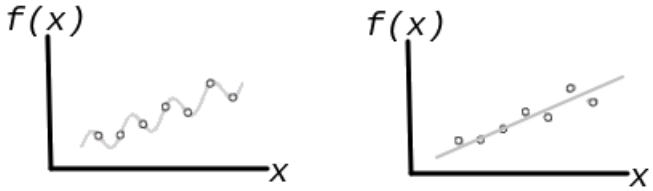
O aprendizado supervisionado é um método que, por meio de uma base de dados classificada, realiza uma predição de novos registros com base em vários desses exemplos já classificados, ou seja, é quando existem casos que possuem uma classificação disponível para determinados conjuntos de dados (conjunto de treinamento), mas precisa ser previsto para outras instâncias. Os responsáveis por essas predições de novos registros são algoritmos de aprendizado supervisionados projetados para determinados fins.

O termo "Supervisionado" indica uma correlação entre os dados de entrada com a saída desejada (classe); seguindo essa afirmação, considere uma base de dados de imagens de rostos, em que cada imagem possui uma saída representada por uma classe (masculino ou feminino). A tarefa seria criar um preditor capaz de acertar a cada novo registro se a imagem é masculina ou feminina. Seria difícil implementar de maneira tradicional, utilizando estruturas condicionais e laços, uma vez que são inúmeras as diferenças das faces masculinas e femininas. Embora haja uma dificuldade de distinção entre as faces, uma alternativa seria dar exemplos de rostos classificados, masculino ou feminino, e através desses exemplos aplicar o algoritmo que automaticamente faça a máquina aprender uma regra para predizer a qual sexo pertence cada rosto (BARBER, 2011).

Em (RUSSEL; NORVIG, 2013), é feita apresentação formal do funcionamento da aprendizagem supervisionada, pois dado um conjunto de treinamento, $(x_1, y_2), (x_2, y_2), ...(x_n, y_n)$, onde cada y_j foi gerado por $y = f(x)$ desconhecida, e encontrar uma função h (hipótese) dentre várias possíveis, que se aproxime ao máximo da função f (real). Quanto mais próxima de f, melhor o desempenho da função h, mas para medir esse desempenho é testado um conjunto diferente (dados de teste) do conjunto de treinamento e aferida a precisão da função hipótese.

O exemplo da figura 1a mostra a função h de grau 6,na qual acontece um sobreajuste (*overfitting*) no conjunto de dados de treinamento. Esse modelo exibe uma função mais complexa para atendera todo o conjunto de dados do gráfico, tornando-se um modelo específico para essa amostra de dados.

Já na figura 1b, o ajuste da função h se torna mais simples e mesmo o gráfico não passando por todos os pontos, acabou por generalizar melhor o conjunto de treinamento,



(a) Função **h** com ajuste polinomial de grau 6

(b) Função **h** com hipótese linear

Figura 1 – Hipóteses ajustadas – Função h próxima da função f real

tornando, um melhor resultado da predição de novos valores.

Em análise da figura 1 são apresentadas duas hipóteses que tentam se aproximar ao máximo da função verdadeira (f), que é desconhecida. Mesmo parecendo que na figura 1aobteve-se melhor resultado, pois todos os pontos são atingidos pelo gráfico da função, este modelo acabou se ajustando muito bem na amostra de dados, deixando a função h muito específica, não retratando os dados em um mundo real. Então, apesar de parecer que a figura 1a é a melhor opção por ela ser mais específica, não é, pois quanto mais generalizado for o modelo, melhor será para predizer os valores de y para novos conjuntos de dados.

2.1.1.1 Algoritmo Classification And Regression Trees - CART

O algoritmo Classification And Regression Trees - CART constrói modelos de previsão a partir de dados de treinamento e seus resultados podem ser representados em uma árvore de decisão. A árvore de decisão é uma ferramenta que dá suporte à decisão utilizando como modelo um fluxograma semelhante a uma árvore, em que, a cada nó interno, é feito um teste para tomada de decisão, tendo como resposta "sim" ou "não" (a exemplo da figura 2), permitindo uma abordagem do problema de forma estruturada e sistemática até chegar a uma conclusão lógica. "Uma árvore de decisão alcança sua decisão executando uma sequência de testes" (RUSSEL; NORVIG, 2013, p. 811)

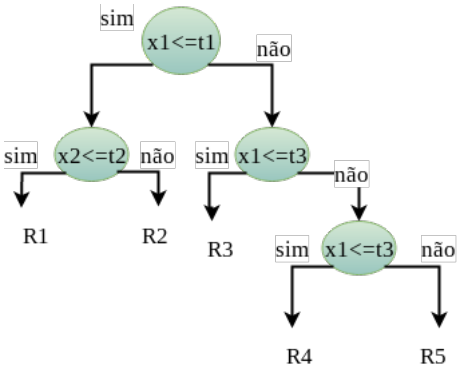


Figura 2 – Exemplo de Fluxograma de árvore: R1 a R5 são as folhas relacionadas de acordo com as respostas sim ou não dos nós

O CART pode se tornar uma árvore de classificação ou também uma árvore de regressão. O que definirá o tipo de árvore é o valor do atributo "classe", se categórico ou contínuo. Por exemplo, em um conjunto de dados de um paciente que tenta prever se ele possuirá câncer, ou não, a classe seria "Terá Câncer" ou "Não terá Câncer", podendo esse atributo assumir duas categorias (classes) e assumindo uma árvore de classificação. Na regressão, o atributo "classe" pode assumir um valor contínuo, tornando-se uma árvore de regressão, que poderá prever valores numéricos como: período de tempo de internação do paciente, preço de uma cirurgia, temperatura do paciente ou quantidade de água ingerida. No caso de não ser probabilístico, o grau de confiança em seu modelo de predição será embasado em respostas semelhantes em outras circunstâncias antes analisadas e utiliza uma técnica como partição recursiva binária, na qual cada nó pai é sempre decomposto em dois nós filhos, e cada nó filho irá ser tratado posteriormente no processo como nó pai.

De acordo com (YOHANNES; WEBB, 1999; RAIMUNDO et al., 2008), existem três componentes importantes na construção de uma árvore de decisão:

- Um conjunto de perguntas que servirá de base para fazer uma divisão;
- Regras de divisão para julgar o quanto é boa esta divisão;
- Regras para atribuir uma classe a cada nó;

Na divisão inicial será atribuída ao nó pai uma questão e, dependendo da resposta (sim ou não - a exemplo da figura 2), os registros irão para nó filho esquerdo ou direito, e depois será realizado o teste do ponto de divisão. O CART percorrerá todos os atributos e construirá uma árvore de decisão baseada no melhor ponto de divisão, uma vez que são testados todos os atributos como potencial divisor. Após a escolha do melhor ponto de divisão do nó, faz-se a atribuição de uma classe para o nó e, após essa etapa, o novo nó filho passa a ser nó pai, refazendo os mesmos passos anteriores para sua divisão.Logo, na equação 2.1 pode-se verificar como o CART faz a escolha da divisão dos nós em função da regra Gini de Impureza (BREIMAN et al., 1984). É definido o grau de pureza variando de 0 (zero) a 1 (um), portanto, quando o nó tende a um resultado de índice Gini aproximando-se de 1 (um), maior a impureza do nó, e o inverso, maior a pureza.

$$Gini(S) = 1 - \sum[p(j/t)]^2 \tag{2.1}$$

Onde: $p(j/t)$ é probabilidade a priori da classe j se formar no nó t, e S é um conjunto de dados que contém exemplos de n classes.

A equação 2.1 mede a impureza do conjunto S e caso todos os dados forem da mesma classe (dados puros), o resultado da equação seria $1 - 1 = 0$. tem como finalidade a escolha da divisão do nó (conjunto S), em que é medida a impureza da divisão do nó pai com os nós filhos, e para isso, contém a média ponderada de cada índice do subgrupo

formado por essa divisão de S. Então o menor valor encontrado em $Gini_{split}(S)$ será o escolhido para dividir o nó.

$$Gini_{split}(S) = \frac{S_l}{S} gini(S_l) + \frac{S_r}{S} gini(S_r) \tag{2.2}$$

Onde:

- S conjunto de dados que contém exemplos de n classes;
- S_l subconjunto esquerdo de S;
- S_r subconjunto direito de S;

Para a escolha da variável e ponto de divisão necessário do nó, terão que ser aplicados testes em todos os atributos através das equações 2.1 e 2.2 e após o envolvimento de todos os atributos, será escolhido o nó com menor valor $Gini_{split}(S)$.

No procedimento da divisão do nó em dois subconjuntos, o atributo poderá conter valores contínuos ou categóricos. Nas duas opções será aplicada a equação 2.2 em todos os valores e escolhido o melhor ponto de divisão. No caso de serem valores contínuos, simplesmente após o valor ser o escolhido para a divisão, ela será menor, igual (ramo da esquerda) ou maior (ramo da direita) que o valor escolhido. Em outra situação, sendo as variáveis categóricas - por exemplo X, Y e Z, terá que ser testada, dentre todas as possibilidades, qual a melhor divisão entre elas e como é uma divisão binária, pois o nó não poderá ser dividido em 3 (três) ramos X, Y e Z, e sim em grupos de dois, como:{X} e {Y,Z}, {Y} e {X,Z} ou {Z} e {X,Y}.

2.1.1.2 Algoritmo Naive Bayes

O algoritmo Naive Bayes é um modelo probabilístico de aprendizado que pode ser calculado diretamente entre seus dados de treinamento. Depois de calculado, o modelo pode ser utilizado para fazer previsões de novos dados através do teorema de Bayes. “O teorema de Bayes fornece uma maneira de calcular a probabilidade de uma hipótese com base em sua probabilidade anterior, as probabilidades de observar vários dados, dadas as hipóteses, e os dados observados em si” (MITCHELL, 1997, p. 156).

Esse teorema utiliza uma teoria estatística e probabilística para previsão de acontecimento de um evento, sendo este evento relacionado à condição da probabilidade de ocorrências anteriores a ele, portanto, é nesse segmento que o algoritmo Naive Bayes funciona, criando classificadores probabilísticos baseados no teorema de Bayes.

Pode-se citar, como exemplo desse evento, a descoberta do câncer em uma pessoa, pois se tal doença estiver relacionada ao sexo, então, utilizando o teorema de Bayes, o sexo de uma pessoa pode ser utilizado para dar maior precisão à probabilidade de câncer, em vez de fazer uma avaliação de probabilidade sem a utilização dele.

O Naive Bayes utiliza uma técnica de independência dos atributos, em que cada variável de entrada não depende de recursos de outras. Essa independência condicionada, entre os atributos que nem sempre ocorrem nos problemas reais, acabou ficando conhecida por Bayes ingênuo ou Naive Bayes.

Em (RUSSEL; NORVIG, 2013) a equação 2.3 mostra a relação $P(causa/efeito)$ onde o efeito é evidência de alguma causa desconhecida, e quer se determinar a causa.

$$P(causa|efeito) = \frac{P(efeito|causa)P(causa)}{P(efeito)} \tag{2.3}$$

Naive Bayes como classificador estatístico possui um modelo de simples construção, e ficou conhecido por ter bons resultados em relação a algoritmos mais sofisticados, mesmo trabalhando com grandes quantidades de dado, e possui uma característica de agrupar objetos de uma certa classe em razão da probabilidade do objeto pertencer a esta classe.

$$P(c/x) = \frac{P(x/c)P(c)}{P(x)} \tag{2.4}$$

$$P(c/x) = P(x_1|c) * P(x_2|c) * ... * P(x_n|c) * P(c) \tag{2.5}$$

- $P(c/x)$ probabilidade posterior da classe $c, alvo$ dada preditor $x, atributos$.
- $P(c)$ é a probabilidade original da classe.
- $P(x|c)$ é a probabilidade que representa a probabilidade de preditor dada a classe.
- $P(x)$ é a probabilidade original do preditor.

A utilização do algoritmo Naive Bayes já é bem difundida e está presente em vários trabalhos, como classificação de textos, filtro de SPAM, analisador de sentimentos, entre outros (MADUREIRA, 2017; LUCCA et al., 2013; WU et al., 2008; MCCALLUM; NIGAM, 1997), entretanto, mesmo atingido boa popularidade, o algoritmo se utiliza da suposição de ter preditores independentes e isso não acontece muito na vida real, pois acaba sendo difícil ter uma amostra de dados que sejam inteiramente independentes.

A tabela 1 é uma amostra de dados, na qual é possível aplicar o funcionamento do Naive Bayes através da equação 2.5, sendo esta tabela composta por 4 (quatro) características (Aspecto, Temperatura, Umidade e Vento) e pela coluna Decisão, representando a classe.

Então, cada registro assume uma condição de jogo Sim" ou "Não" - por exemplo, na linha 2 (dois) faz "Sol", é "Quente", possui umidade "Alta" e vento "Forte" e existe no atributo classe a não possibilidade de jogo (Decisão = Não). Pode-se com as informações

	Aspecto	Temperatura	Umidade	Vento	Decisão
1	Sol	Quente	Alta	Fraco	Não
2	Sol	Quente	Alta	Forte	Não
3	Nublado	Quente	Alta	Fraco	Sim
4	Chuva	Agradável	Alta	Fraco	Sim
5	Chuva	Fria	Normal	Fraco	Sim
6	Chuva	Fria	Normal	Forte	Não
7	Nublado	Fria	Normal	Forte	Sim
8	Sol	Agradável	Alta	Fraco	Não
9	Sol	Fria	Normal	Fraco	Sim
10	Chuva	Agradável	Normal	Fraco	Sim
11	Sol	Agradável	Normal	Forte	Sim
12	Nublado	Agradável	Alta	Forte	Sim
13	Nublado	Quente	Alta	Fraco	Sim
14	Chuva	Agradável	Alta	Forte	Não

Tabela 1 – Base de exemplo onde exibe através do atributo Decisão, se existe, ou não condição de jogo perante as outras características (Aspecto, temperatura, Umidade e Vento)

dessa amostra prever algumas possibilidades de jogo, dependendo de como estão dispostos os valores dessas características. Para exemplificar melhor, as seguintes condições são para saber se há possibilidade de jogo, "Sim": P(Jogar=sim|Aspecto=sol, Temperatura=fria, Umidade=alta e Vento=forte). Para obter a resposta, será necessária a aplicação na equação 2.5.

$$
\begin{aligned}
&= \frac{P(Sol|Sim) * P(Fria|Sim) * P(Alta|Sim) * P(Forte|Sim) * P(Sim)}{P(Sol) * P(Fria) * P(Alta) * P(Forte)} \\
&= \frac{\frac{2}{9} * \frac{3}{9} * \frac{3}{9} * \frac{3}{9} * \frac{9}{14}}{\frac{5}{14} * \frac{4}{14} * \frac{7}{14} * \frac{6}{14}} \\
&= \frac{0,0053}{0,02186} \\
&= 0,242
\end{aligned} \tag{2.6}
$$

O resultado $(0,242)$ mostra uma probabilidade aproximada de 20% de chance de acontecer o jogo de acordo com as características apresentadas, implicando aproximadamente 80% de não acontecer, portanto, seguindo esse resultado, pode-se afirmar que a previsão é de não ocorrer o jogo.

2.1.1.3 Algoritmo k-Nearest Neighbor - KNN

O K-Nearest Neighbor - KNN é um algoritmo de classificação simples, em que os objetos são classificados por meio de um conjunto de treinamento que estão próximos no espaço de características. Uma vez que seja necessário definir qual a classificação de um

objeto, serão averiguados quais são os exemplos mais próximos determinados por uma distância, e assim se definirá, por meio desses elementos próximos, qual sua classificação.

Na execução do KNN algumas considerações são importantes, como a definição da métrica entre os elementos e o número que a variável K assumirá; ademais, seguem alguns passos: i) o algoritmo funcionará calculando a distância entre todos os exemplos próximos do elemento a classificar; ii) serão identificados os K vizinhos mais próximos; iii) e através do número de K será determinada a mesma classe do vizinho para classificação do elemento.

Na figura 3 pode-se observar o comportamento do algoritmo: um objeto está próximo de outros objetos vizinhos e o número de vizinhos é determinado pelo K.

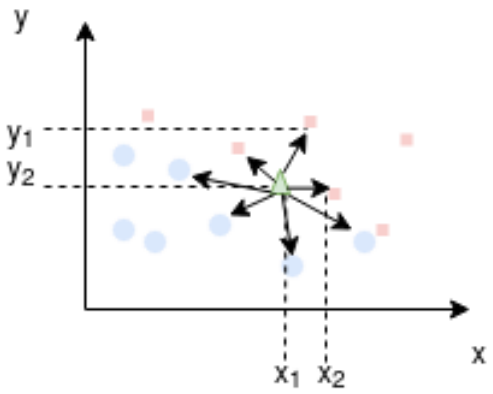


Figura 3 – Exemplo do funcionamento do algoritmo KNN

Neste exemplo da figura 3 um novo objeto precisa ser classificado e na amostra de dados distribuída no plano cartesiano o algoritmo KNN prevê sua classificação de acordo com objetos que estão próximos a ele. Na figura 3 existem objetos “círculos”, “quadrados”, e um novo objeto “triângulo” a ser classificado.

O algoritmo KNN calcula a distância do objeto “triângulo” para com os outros objetos utilizando a distância euclidiana (LACHI; ROCHA, 2005), conforme equação 2.7 que está na forma bidimensional de acordo com o exemplo apresentado na figura 3.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{2.7}$$

Onde:

- d resultado da distância euclidiana entre dois objeto no plano cartesiano;
- $x_2 - x_1$ distância no eixo x entre objetos;
- $y_2 - y_1$ distância no eixo y entre objetos;

Ao calcular todas as distâncias, o algoritmo irá ordenar estes resultados de forma crescente de acordo com a tabela 2 e depois, dependendo do valor de K, será determinada

a classe do objeto. A tabela de exemplo é formada por 3 (três) colunas, em que K significa o número de vizinhos, seguida da distância e também qual o tipo de objeto (quadrado ou círculo).

K	Distância	Classe
1	0,11	
2	0,29	
3	1,10	
4	1,40	
5	1,55	

Tabela 2 – Tabela em ordem crescente das distâncias euclidianas encontradas do objeto a que se deseja classificar para os outros objetos da amostra, de acordo com as setas da figura 3

Para definir qual a classe que o objeto "triangulo" assumirá, será necessário saber qual valor de K, pois o maior número de ocorrências de uma determinada classe será a classe eleita para o novo objeto - por exemplo, utilizando a tabela 2, com $K = 1$, a classe deste registro é "círculo", dado que o número de ocorrências é CÍRCULO=1 e QUADRADO=0, então, caso $K = 1$, o número de ocorrência do "círculo" é maior que "quadrado". No caso de $K = 2$, aparece uma ocorrência de "círculo" e uma ocorrência de "quadrado", totalizando CÍRCULO=1 e QUADRADO=1; nesse caso, não fica definida qual a classe, pois cada uma possui o mesmo número de ocorrências. No caso de$K = 3$, a contagem de ocorrências em cada classe totaliza CÍRCULO=1 e QUADRADO=2 e, desta vez, a classe que possui maior número de ocorrências é a escolhida para o novo objeto "quadrado" e assim acontece sucessivamente para os outros valores K assumirem. Cada vez que o valor de K aumenta, é feita a soma de ocorrências de cada classe, sendo a classe eleita a que apresentar mais ocorrências.

Ao escolher um valor par de K, poderá haver um empate no número de ocorrências das classes. Essa situação fica clara no exemplo acima, tabela 2, quando $K = 2$ (CÍRCULO=1 e QUADRADO=1) e $K = 4$ (CÍRCULO=2 e QUADRADO=2); para não acontecer um valor de empate, é necessário escolher sempre um valor K ímpar e este parâmetro é definido ao executar o algoritmo.

2.1.2 Aprendizado Não-Supervisionado

Outro cenário de aprendizado de máquina é o aprendizado não supervisionado, em que não existe uma tentativa de se encontrar uma função que se aproxime da real, logo porque os registros não são classificados, visto que o conjunto de treinamento não possui informação da saída sobre determinada entrada. Desta forma, os algoritmos procuram algum grau de similaridade entre os registros e tentam agrupá-los de forma a ter algum sentido para estarem juntos.

Quando o algoritmo encontra dados com mesma similaridade a ele, estes são agrupados, formando *clusters*. Os números de *clusters* encontrados dependerá do funcionamento, técnica, configuração dos algoritmos e também do grau de dissimilaridade entre elementos de grupos diferentes. Segundo (BARBER, 2011) não existe uma variável "classe" no aprendizado não-supervisionado, então, o maior interesse seria em uma perspectiva probabilística de distribuição p(x) de um determinado conjunto de dados $D = \{x_n, n = 1, ..., N\}$. Mesmo não possuindo rótulos (classes), novos dados inseridos são submetidos aos algoritmos não-supervisionados e esses algoritmos são capazes de encontrar padrões nos atributos em um conjunto de treinamento, conseguindo inferir sobre os dados de testes, classificando-os em algum grupo.

Cada algoritmo utilizará alguma característica para determinar grupos diferentes na base de dados; no exemplo da figura 4, fica visível a existência de agrupamentos diferentes (h1, h2, h3, h4).

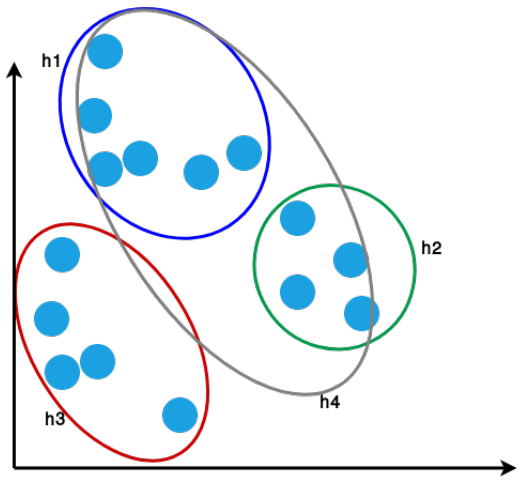


Figura 4 – Exemplos de técnicas diferentes utilizada por algoritmos para dividir em grupos

O que é apresentado na figura 4 são situações de agrupamentos em um conjunto distribuído bidimensional. Dependendo do algoritmo, ou mesmo da configuração, é possível ter 3 (três) grupos formados por h1, h2 e h3, ou 2 (dois) grupos formados por h3 e h4, ou mesmo, um grupo só pela união de h3 e h4.

2.2 Discretização

O método de discretização faz a conversão de valores contínuos em valores discretos. A partir de um atributo com valores contínuos, a discretização cria um ponto inicial e final definindo um intervalo e designando uma faixa para cada intervalo. Assim, ao invés de valores contínuos, novos conteúdos representando as faixas de valores.

Segundo alguns autores, a discretização melhora a precisão e deixa um modelo mais rápido em seu conjunto de treinamento (CATLETT, 1991; HWANG; LI, 2002). De acordo com (KOTSIANTIS; KANELLOPOULOS, 2006; DOUGHERTY et al., 1995), os métodos de discretização mais comumente utilizados, no âmbito dos métodos não supervisionados, são os de Discretização por Larguras Iguais (do inglês: Equal Width Discretization - EWD) e Discretização por Frequências Iguais (do inglês: Equal Frequency Discretization - EFD).

2.2.1 Equal Weight Discretization - EWD

O método de Discretização por Larguras Iguais (Equal Weight Discretization - EWD) faz a discretização de um intervalo, entre valores contínuos, dividindo através de um ponto de corte as faixas de tamanhos iguais. Logo, se existir um intervalo com valores contínuos [a,b] e deseja particionar em R faixas de tamanhos iguais, serão necessários $R-1$ pontos de corte (figura 5).

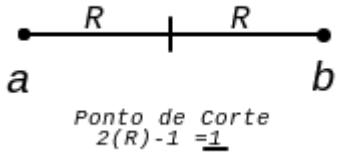


Figura 5 – Ponto de Corte (R-1); 2(R) significa o valor de R=2; 2 - 1 = 1

Para exemplificar, a figura 5 exibe uma faixa com início em "[a", e final "b]", e pretende obter a divisão do intervalo $[a, b]$ em 2 (duas) faixas iguais ($R = 2$), então, utilizando a regra (Número_De_Faixas - 1 = 2-1=1), obtendo sempre o número de faixas que deseja particionar, menos 1 (um).

Para haver o ponto de corte, terá que ser feita primeiro a ordenação dos dados e logo após definir a largura de cada faixa $r_1, ..., r_R$. O cálculo realizado na equação 2.8, para encontrar w, é a diferença entre os limites superior e inferior do intervalo, dividido pela quantidade de faixas (R).

$$w = \frac{b-a}{R} \tag{2.8}$$

De acordo com a figura 2.9, a variável w delimita o tamanho das faixas de valores e determina os pontos de corte $(c_1, ..., c_{R-1})$. O primeiro ponto de corte, c_1, é obtido por meio da soma do limite inferior a com a tamanho de w, e os pontos de corte seguintes são calculados pela soma do ponto de corte anterior com w.

$$c_i = \begin{cases} a + w, & se\, i = 1 \\ c_{i-1} + w, & caso\, contrário \end{cases} \tag{2.9}$$

Seguindo a figura 6, o valor da faixa do intervalo $[a, c_1]$ será o valor discreto igual ao índice de sua faixa, nesse caso r_1, então, o valor na faixa r_1 é reprensentado por $1(um)$, pois $i = 1$. No caso do intervalo $]c_1, c_2]$ definido na faixa r_2, é representado pelo valor discreto 2 (dois) e, e consequentemente, o valor que se encontra em uma faixa qualquer r_i será reprensentado por i.

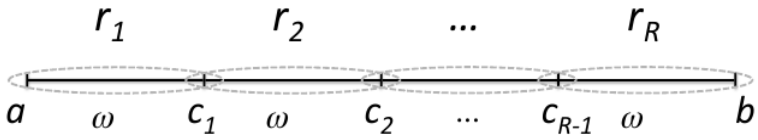


Figura 6 – Discretização EWD. Figura baseada em (LOPES et al., 2016)

A tabela 3 contém uma amostra de valores dividida em duas colunas: a primeira coluna significa o número da linha e a segunda coluna o valor propriamente dito. Essa tabela servirá para exemplificar o que foi dito neste método de discretização e pode ser vista como um vetor de 150 (cento e cinquenta) posições e, dependendo do número de faixas, poderá ser dividido em vários pedaços, sendo cada pedaço uma faixa.

no.	Valor	no.	Valor	no.	Valor	no.	Valor	no.	Valor	no.	Valor
1	5.10	26	5.00	51	7.00	76	6.60	101	6.30	126	7.20
2	4.90	27	5.00	52	6.40	77	6.80	102	5.80	127	6.20
3	4.70	28	5.20	53	6.90	78	6.70	103	7.10	128	6.10
4	4.60	29	5.20	54	5.50	79	6.00	104	6.30	129	6.40
5	5.00	30	4.70	55	6.50	80	5.70	105	6.50	130	7.20
6	5.40	31	4.80	56	5.70	81	5.50	106	7.60	131	7.40
7	4.60	32	5.40	57	6.30	82	5.50	107	4.90	132	7.90
8	5.00	33	5.20	58	4.90	83	5.80	108	7.30	133	6.40
9	4.40	34	5.50	59	6.60	84	6.00	109	6.70	134	6.30
10	4.90	35	4.90	60	5.20	85	5.40	110	7.20	135	6.10
11	5.40	36	5.00	61	5.00	86	6.00	111	6.50	136	7.70
12	4.80	37	5.50	62	5.90	87	6.70	112	6.40	137	6.30
13	4.80	38	4.90	63	6.00	88	6.30	113	6.80	138	6.40
14	4.30	39	4.40	64	6.10	89	5.60	114	5.70	139	6.00
15	5.80	40	5.10	65	5.60	90	5.50	115	5.80	140	6.90
16	5.70	41	5.00	66	6.70	91	5.50	116	6.40	141	6.70
17	5.40	42	4.50	67	5.60	92	6.10	117	6.50	142	6.90
18	5.10	43	4.40	68	5.80	93	5.80	118	7.70	143	5.80
19	5.70	44	5.00	69	6.20	94	5.00	119	7.70	144	6.80
20	5.10	45	5.10	70	5.60	95	5.60	120	6.00	145	6.70
21	5.40	46	4.80	71	5.90	96	5.70	121	6.90	146	6.70
22	5.10	47	5.10	72	6.10	97	5.70	122	5.60	147	6.30
23	4.60	48	4.60	73	6.30	98	6.20	123	7.70	148	6.50
24	5.10	49	5.30	74	6.10	99	5.10	124	6.30	149	6.20
25	4.80	50	5.00	75	6.40	100	5.70	125	6.70	150	5.90

Tabela 3 – Amostra de dados para exemplificar a discretização EWD e EFD

Para este exemplo descrito, é definido em 3 (três) o número de faixas ($R = 3$) e aplicada a equação 2.8 na tabela 3 para encontrar a largura (fixa) da faixa, pois o método de discretização EWD mantém faixas com mesmo tamanho. O intervalo $[a, b]$ seria o limite inferior, menor valor ($a = 4, 3$) e limite superior, maior valor ($b = 7, 9$), respectivamente, da tabela de amostra, portanto, logo que calculado o valor da largura w é utilizada a regra da equação 2.9 para encontrar os pontos de corte de cada faixa.

Utilizando a equação 2.8, encontra-se $w = 1.2$; portanto, uma vez em posse da largura e sendo o primeiro ponto de corte ($i = 1$), simplesmente utiliza-se equação 2.9 para encontrar o primeiro ponto de corte $a + w = 4.3 + 1.2 = 5.5$, que está como asterisco na posição $c_1 = 5.5$. Os asteriscos na figura 7 delimitam os pontos de cortes e os pontinhos são todos os valores dispostos nas faixas definidos na tabela 3.

Figura 7 – Discretização EWD de acordo com a amostra da tabela 3

A partir dos cálculos é definido as seguintes faixas de tamanhos iguais:

- Faixa 1 - [4.3,5.5]
- Faixa 2 -]5.5,6.7]
- Faixa 3 -]6.7,7.9]

2.2.2 Discretização por Frequência Iguais - EFD

Esse outro método de discretização já possui uma abordagem diferente do EWD, pois a ideia é manter a quantidade de elementos distintos entre os pontos de corte com o mesmo número. Dado um intervalo $[a, b]$, o número de faixas R e a quantidade de valores distintos ξ , em que $\xi \geqslant R$, o método EFD irá segmentar em R faixas de valores que possuem a mesma quantidade de elementos distintos λ. Então serão realizados $R - 1$ pontos de corte gerando R faixas de valores, $(r_1, ..., r_R)$, com a mesma quantidade de elementos distintos λ. Para encontrar λ , calcula-se o valor inteiro da divisão entre a quantidade de elementos distintos ξ pela quantidade de faixas de valores R, obtendo o número de elementos da faixa (equação 2.10).

$$\lambda = \frac{\xi}{R} \tag{2.10}$$

Uma observação nesse método é a ocorrência de uma má distribuição de valores entre as faixas, portanto, caso haja um número significativo de valores repetidos de um atributo, isso causa um desequilíbrio na distribuição dos elementos dentro da faixa. Essa situação reflete em faixas com muitos valores e outras sem nenhuma.

Uma vez no intervalo $[a, b]$ de elementos ordenados e calculado λ contendo R elementos $v_{[R]}$, pode-se determinar os pontos de corte $(c_1, ..., c_{R-1})$, que são os delimitadores das faixas. Cada ponto de corte c_i pode ser calculado por $v_{i\lambda} - \acute{e}simo$ elemento (equação 2.11).

$$c_i = v_{[i\lambda]} \tag{2.11}$$

Igual ao que aconteceu no método EWD, o valor que estiver no intervalo $[a, c_1]$ terá seu valor associado a um valor discreto igual ao índice i de sua faixa r_i, conforme figura 8. Então, caso o valor esteja na faixa r_2, ele passará a ter o valor de seu índice i igual a $2(dois)$. De maneira consecutiva, os valores que estiverem na faixa $r_3 =]c_2, c_3]$ terão valor 3 (três). Uma outra observação desse método é que, diferente do EWD, os intervalos podem assumir faixas com tamanhos diferentes.

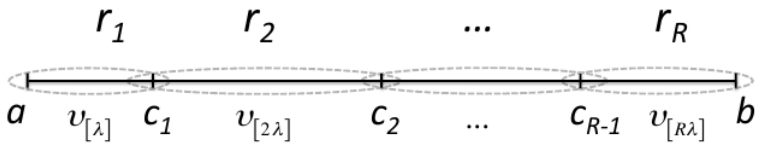


Figura 8 – Discretização EFD. Figura baseada em (LOPES et al., 2016)

Neste exemplo com o método EFD, exibido o resultado na figura 9, a tabela 3 também é utilizada na disposição dos valores em cada faixa, bem como o cálculo de λ (equação 2.10), que é a divisão do total de elementos distintos (ξ) pelo o número de faixas $R = 3$. Após tal cálculo, é realizado a ordenação dos valores, e uma vez os valores ordenados, soma-se o valor mínimo com λ, encontrando c_1 como primeiro ponto de corte, e assim sucessivamente, para os outros pontos de corte $(c_2 = c_1 + \lambda)$, até $R - 1$ pontos de corte. Na figura 9 pode-se perceber como a distribuição dos valores da amostra se comportam no método EFD. Os asteriscos delimitam o início e fim de cada faixa de valor, e os pontos são os valores propriamente ditos.

Neste exemplo com o método EFD, exibido o resultado na figura 9, a tabela 3 também é utilizada na disposição dos valores em cada faixa, bem como o cálculo de λ (equação 2.10), que é a divisão do total de elementos distintos (ξ) pelo número de faixas $R = 3$. Após tal cálculo, é realizada a ordenação dos valores e, uma vez os valores ordenados, soma-se o valor mínimo com λ, encontrando c_1 como primeiro ponto de corte e assim sucessivamente para os outros pontos de corte $(c_2 = c_1 + \lambda)$ até $R - 1$ pontos de corte. Na figura 9, pode-se perceber como a distribuição dos valores da amostra se comportam no método EFD. Os asteriscos delimitam o início e fim de cada faixa de valor e os pontos são os valores propriamente ditos.

O número de elementos distintos na amostra de dados da tabela 3 é de 35 (trinta e cinco) elementos e dividindo-se por $R = 3$, que é o número de faixas, encontrar-se-á $\lambda = 11$.

Figura 9 – Discretização EFD de acordo com a amostra da tabela 3

Na lista de números distintos o 11º (décimo primeiro) elemento a partir de a (menor valor da amostra) será o primeiro ponto de corte (c_1). Então, todos os valores de $4,3$ a $5,3$ (identificados por asterisco) fazem parte da primeira faixa.

A seguir são definidas as faixas, percebendo-se que são de tamanhos diferentes ao do método EWD, pois há valores repetidos:

- Faixa 1 - [4.3,5.3]
- Faixa 2 -]5.3,6.4]
- Faixa 3 -]6.4,7.9]

2.3 Trabalhos Correlatos

Esta seção propõe relacionar outros trabalhos, servindo de complemento teórico envolvendo assuntos como: agrupamentos de dados, aprendizado de máquina, classificação e rotulação de dados.

Em (JIRASIRILERD; TANGTISANON, 2018), os autores fazem uso de rotulação automática de textos em Tailandês retirados de notícias da internet. É utilizado para classificar esses textos com a devida categoria (TI, entretenimento, astronomia, etc). Essa pesquisa utiliza para rotulação vetor de representação de documentos e na separação das palavras Tailandesas é aplicado algoritmo de *Convolutional Neural Network* (CNN). Então o modelo faz: i) Conversão de parágrafos e palavras para os vetores utilizando a técnica de representação distribuída; ii) extrai vetores com parágrafos semelhantes; iii) cria um vetor de características; iv) extrai vetores com palavras semelhantes; v) rotula.

Em (YEGANOVA et al., 2010), é utilizada a ideia de dados naturalmente rotulados em uma abordagem que faz detecção e identificação de abreviações na literatura biomédica utilizando aprendizado de máquina supervisionado. Por meio dos textos é realizada uma extração de estruturas textuais (formas curtas, formas longas, formas curtas potenciais e formas longas potenciais), que quando são extraídas naturalmente em pares *i.g.* (forma curta - forma forma longa), (formas curtas potenciais - formas longas potenciais) são tratadas como exemplos positivos.

Nesse artigo, (CHEN et al., 2011) se propõe à criação de clusters a partir de textos e documentos através de uma abordagem eficaz de agrupamento de documentos, *Fuzzy Frequent Itemset-based Document Clustering* (F2IDC), que combina a mineração de regras de associação *fuzzy* com conhecimento da WordNet. A WordNet é um banco de dados léxico

em inglês que agrupa palavras (substantivos, verbos, adjetivos, advérbios) em conjunto de sinônimos e tem com isso o objetivo de melhorar a qualidade dos grupos através dos relacionamentos semânticos.

Nos trabalhos acima citados (JIRASIRILERD; TANGTISANON, 2018; YEGANOVA et al., 2010; CHEN et al., 2011) acontecem processos de agrupamentos e rotulação em textos, sendo um tema bastante estudado e diferente desta dissertação, que utiliza o conceito de rotulação de dados no contexto do significado ao grupo formado.

O artigo em (GAN et al., 2013) propõe aprendizado de máquina semisupervisionado, que combina agrupamento e classificação com os devidos algoritmos *Fuzzy C-Means* e *SVM* respectivamente. A pesquisa utiliza dados rotulados e não rotulados, apostando na análise do *cluster* como diferencial para compensar a limitação de dados não rotulados e através do conhecimento adquirido melhorar o treinamento do classificador. Essa pesquisa (GAN et al., 2013), diferente desta dissertação, não envolve a interpretação dos grupos após sua formação.

Outro trabalho de agrupamento de dados pode ser visto em (SUN et al., 2011), no qual ele aborda o problema de agrupamento de dados e propõe um algoritmo híbrido com o *support vector cluster* e *K-Means*, ambos de aprendizagem não- supervisionada. Em uma primeira etapa é utilizada uma abordagem do *support vector cluster* com o objetivo de identificar os *outliers* e os pontos sobrepostos e na segunda etapa obtêm-se os núcleos removendo os *outliers* e os pontos sobrepostos e aplicando o *K-Means* nos núcleos para obter o conjunto de dados em *clusters*. Foram utilizadas algumas variáveis de extrema importância para conclusão do trabalho e essas variáveis foram configuradas empiricamente em fases diferentes a fim de obter bons resultados nos agrupamentos de dados. O estudo desse trabalho é interessante no conceito da formação de bons grupos de dados, pois em rotulação de grupos há uma correlação de grupos bem definidos e bons rótulos.

Outro artigo, o autor (IWAMURA et al., 2013), faz rotulação automática de textos de cenas em uma base de dados. São textos que se encontram em uma imagem - por exemplo, uma imagem da fachada de uma casa que possui um número de identificação. O modelo utiliza imagens contendo caracteres, segmenta essas imagens e, após, classifica o caractere através de uma base de dados com várias amostras armazenadas, fazendo seu reconhecimento. Fica claro que esse artigo faz rotulação, mas não da mesma forma definida nesta dissertação, a qual faz uso de aprendizado de máquina a fim de rotular grupos e dando significado a eles.

O trabalho (COSTA et al., 2016) utiliza classificação não supervisionada, mas possui uma característica diferenciada por utilizar dados *online*. O método faz uso da aprendizagem a partir de uma base de regras vazias com o processamento das amostras desses dados online e não é necessário pré-definir parâmetros iniciais e nem número ou tamanho das classes. Efetivamente, o autor foca em conceito-evolução, portanto, o

algoritmo se adapta continuamente para lidar com mudanças de dados que podem surgir ao se lerem novas amostra de dados. Essa abordagem não supervisionada conta com o conceito de *Typicality and Eccentricity Data Analytics* (TEDA), utilizado em problemas no mundo real como detecção de falhas; no entanto, mesmo sendo um comportamento não-supervisionado, em vez de clusters tradicionais, o algoritmo trabalha com conceito de nuvens de dados. Nesta dissertação a utilização da rotulação são para grupos já formados, diferente dessa pesquisa, que possui o objetivo da formação dinâmica de grupos.

O trabalho proposto por (LOPES et al., 2016) faz um estudo abordando o tema de rotulação de dados e nele foi utilizado como entrada um conjunto de dados em que foi realizado o agrupamento automático com algoritmos não supervisionados para formação de *clusters* e logo após a formação destes, é utilizado um algoritmo supervisionado (RNA) nos grupos de dados já discretizados e apresentado como saída um rótulo específico que melhor define o grupo formado. Esses rótulos são formados por uma tupla, atributos mais relevantes e faixa de valores que mais se repetem. A figura 10 representa o modelo de atuação do autor citado, que por meio de 4 passos (I, II, III, IV), obtiveram-se os rótulos que representaram os clusters.

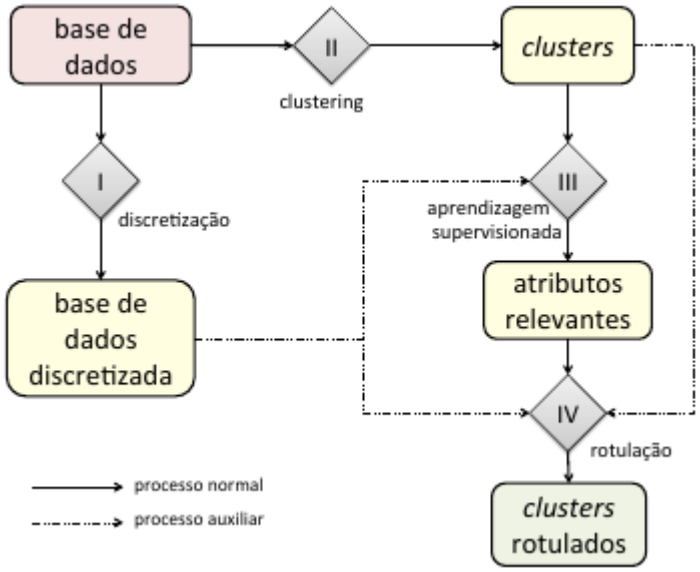


Figura 10 – Modelo de (LOPES et al., 2016)

Em analogia a este trabalho na figura 10, em que é aplicado o algoritmo de RNA, o processo III é o local exato que esta pesquisa utiliza para testar outros algoritmos supervisionados. Nesse modelo proposto, o autor não só tem a preocupação de processar a formação dos grupos como também aplicar um algoritmo supervisionado para rotular esses grupos formados. Já nesta dissertação, o foco é a aplicação dos algoritmos supervisionados na rotulação dos clusters já definidos em etapa anterior.

Em (LIMA, 2015; LIMA et al., 2015) são trabalhos que tem como objetivo principal a rotulação de grupos de dados, mas no primeiro trabalho o autor utiliza uma técnica

de aprendizado semisupervisionada, que visa fazer rotulação de dados através de uma pequena amostra rotulada. Logo em (LIMA et al., 2015) é aplicada rotulação a uma base de dados de uma rede social chamada de Scientia.Net, com o objetivo na criação de grupos e identificação dos atributos que podem ser importantes ao ponto de representar estes grupos, chamando-os de rótulos. O Scientia.Net é uma rede social para cientistas com o propósito do compartilhamento de suas pesquisas e criação de seus perfis, facilitando o contato e troca de informações no ambiente acadêmico entre pesquisadores. Nesse artigo, o autor utilizou o modelo de (LOPES et al., 2016) para rotulação de dados com os mesmos algoritmos, o algoritmo *k-means*, para agrupamento e logo na segunda parte a utilização do algoritmo supervisionado, *Artificial Neural Networks* - ANN. Diferente dos trabalhos (LIMA, 2015; LIMA et al., 2015), este texto tem o foco em testes somente nos algoritmos supervisionados nos grupos já formados de acordo com a origem da base de dados, a fim de gerar rótulos e, assim, aferir suas acurácias para cada um desses algoritmos testados.

Outra pesquisa (FILHO et al., 2015) aborda o mesmo problema de rotulação, mas com atuação diferente, pois o modelo utiliza o algoritmo não-supervisionado *Fuzzy C-Means* para composição dos grupos, em que o número de grupos é fornecido na inicialização do algoritmo e também na definição das faixas de valores de atributos de cada atributo rótulo do grupo formado. A diferença entre esta dissertação e esse artigo de Filho et al. (2015) é o modelo de resolução, que utiliza somente o algoritmo *Fuzzy C-Means* para criar os grupos e definir as faixas de valores de cada atributo rótulo e também a ausência da técnica de discretização dos valores dos dados para obter os rótulos.

Outro autor (IMPERES, 2018) em seu trabalho utiliza uma proposta de rotulação semelhante a outra pesquisa (FILHO et al., 2015), mas com outro algoritmo, *K-means* não-supervisionado, baseado em distância. Esse modelo é constituído de duas etapas, sendo a primeira a transformação da distância gerada pelo *K-means* em GP, e logo após, na segunda etapa, é realizada a rotulação de dados de acordo com a tabela gerada na primeira etapa. São feitas várias iterações até encontrar faixas únicas de valores para cada atributo rótulo. Ao se comparar esta dissertação com a pesquisa desse autor ((IMPERES, 2018)), percebe-se que também não há um processo de discretização de valores dos atributos e também não há aplicação de dois algoritmos de aprendizado.

Outro trabalho de rotulação (ARAÚJO, 2018) defende que uma etapa de fundamental importância para se ter bons resultados na rotulação de grupos de dados se dá na clusterização. Portanto, quanto mais eficiente for a técnica de agrupamento de dados utilizada, maior será acurácia dos grupos encontrados. A partir do que foi dito, o autor utiliza em conjunto o DAMICORE e DAMICORE-2 no método de rotulação automática para criar grupos. DAMICORE é um método de detecção de correlação de dados e tem como característica a não informação do número de *clusters* ao qual o algoritmo é aplicado. O modelo é dividido em cinco etapas até se obterem os rótulos. A etapa I e II preparam

os dados e ajudam a medir a similaridade dos elementos, oferecendo uma maior precisão e contribuindo para criação de grupos mais significativos. Na etapa III ocorre a clusterização, e como não é preciso informar o número de *cluster* na utilização do DAMICORE, os resultados nesta etapa acabam por superar um número razoável de *clusters* para uma melhor compreensão e, em razão disso, a etapa IV, de mesclagem, faz a junção de *clusters* para criação de super-clusters, que são *clusters* maiores representando um conceito mais geral e de mais fácil entendimento. Por fim, os super-clusters são submetidos ao método de rotulação automática na etapa V, permitindo identificar os atributos mais relevantes e suas respectivas faixas de valores. Essa pesquisa mantém o foco na etapa de formação dos clusters e se diferencia do texto desta dissertação exatamente nisso, pois nesta dissertação o foco é nos algoritmos supervisionados utilizados na etapa de rotulação dos grupos de dados.

3 Metodologia

O texto a seguir abordará o problema descrito neste trabalho e, logo em seguida, será apresentado um modelo de resolução utilizando algoritmos supervisionados e mais ao final deste capítulo é realizado um exemplo de rotulação, que será feita desde a discretização até o encontro dos rótulos para melhor explanar o conteúdo descrito.

3.1 Rotulação de Cluster

A abordagem do problema referente a essa proposta de mestrado segue uma linha já pesquisada, que seria o Problema de Rotulação. Muitas pesquisas realizadas na área de rotulação fazem referência à classificação dos dados e não da rotulação. Ao agrupar um conjunto de elementos por um determinado critério, está havendo uma classificação desses elementos de mesma similaridade, mas pouco se sabe qual é a compreensão desses grupos já classificados no sentido de quais os atributos são mais relevantes dentro desses grupos.

A importância do rótulo em um *cluster* é transparecer a compreensão do *cluster* formado, visto que, uma vez os clusters já agrupados, não fica claro o critério de criação desses grupos. Para o observador é interessante existir um rótulo de um grupo oferecendo elementos que possam ajudar em alguma tomada de decisão em razão de seu significado, ou seja, o rótulo. Dessa forma serão apresentadas 2 (duas) definições que se complementam.

Na definição 2 é expresso formalmente o comportamento dos *clusters*, e na definição 3 complementa a definição 2 definindo o comportamento do rótulo.

Definição 2 *Dado um conjunto de clusters* $C = \{c_1, ..., c_k | K \geqslant 1\}$*, de modo que cada cluster contém um conjunto de elementos* $c_i = \{\vec{e}_1, .., \vec{e}_{n^{(c_i)}} | n^{(c_i)} \geqslant 1\}$ *que podem ser representados por um vetor de atributos definidos em* $\mathbb{R}^m$ *e expresso por* $\vec{e}^{c_i} = (a_1, .., a_m)$ *e ainda que com* $c_i \cap c_{i'} = \emptyset$ *com* $1 \leqslant i, i \leqslant K$ *e* $i \neq i'$ *(Adaptada de (**??**)).*

- K *é o número de clusters;*
- a *é o atributo*
- c_i *é o i-ésimo cluster qualquer;*
- n^{c_i} *é o número de elementos do cluster* c_i*;*
- $\vec{e}_j^{(c_i)}$ *se refere ao j-ésimo elemento (registro na tabela) pertencente ao cluster* c_i*;*
- m *é o número de atributos da tabela de dados;*

A criação do rótulo é a escolha de uma tupla **atributo** e **faixa de valor**, em que o atributo possui o maior valor de correlacionamento entre os outros atributos, e a

faixa escolhida, uma vez com os dados já discretizados, é aquele valor que mais se repete, dentro do atributo rótulo selecionado - por exemplo, um vetor de valores já discretizados [1] , $\vec{v}_i = \{1, 1, 1, 2, 2, 2, 2, 3, 3\}$, sendo $i \leqslant m$ e $(\vec{v})$ representando todos os elementos da coluna representada pelo atributo rótulo (a). Neste vetor $(\vec{v})$ o valor que mais se repete é o número 2, então, a **faixa 2** do atributo rótulo é a escolhida para compor o rótulo. Isto posto, o rótulo é o atributo representado por a junto com a representação da faixa 2 (dois) e podendo, em outra situação, o rótulo de um *cluster* ser composto por mais de uma tupla: atributo e faixa (Definição 2).

3.2 O Modelo de Resolução

A partir da definição do problema - *Definição 2* - um estudo foi desenvolvido a fim de ser possível realizar rotulação de dados com algoritmos supervisionados com características distintas.

Este modelo de resolução consiste em apresentar como saída um conjunto de rótulos, em que cada rótulo específico é dado por um conjunto de pares de valores, atributo e seus respectivos intevalos, gerados a partir das frequências dos valores repetidos neste intervalo. Segue *Definição 3* formalizando a saída do modelo:

Definição 3 *Dado um conjunto de rótulos* $R = \{r_{c1}, ..., r_{ck}\}$*, no qual cada rótulo específico é dado por um conjunto de pares de valores, correspondendo a um vetor com atributo e seu respectivo intervalo,* $r_{ci} = \{(a_1, [p_1, q_1]), ..., (a_{m^{(c_i)}},]p_{m^{(c_i)}}, q_{m^{(c_i)}}])\}$ *capaz de melhor expressar o cluster* c_i *(Adaptada de (LOPES et al., 2016)).*

- k *número de rótulos;*
- R *representa o conjunto de rótulos na saída do modelo;*
- a *é o atributo*
- c_i *é o i-ésimo cluster;*
- r_{c_i} *é o rótulo referente ao cluster* c_i*;*
- $]p_{m^{(c_i)}}, q_{m^{(c_i)}}]$ *representa o intervalo de valores do atributo* $a_{m^{(c_i)}}$*, onde* $p_{m^{(c_i)}}$ *é o limite inferior e* $q_{m^{(c_i)}}$ *é o limite superior;*
- m *é o número de atributos da tabela de dados;*

Como apresentado na seção 2.3, o autor (LOPES et al., 2016) foca em rotulação automática de grupos utilizando a estratégia de aprendizagem de máquina supervisionada com paradigma conexionista para realizar seu trabalho. Porém, nesta pesquisa, foram aplicados no modelo de resolução três algoritmos supervisionados com atuações diferentes

[1] seção 2.2

do que já havia sido testado anteriormente, realizando a rotulação de dados. Logo na figura 11 é apresentado o modelo de resolução desta pesquisa.

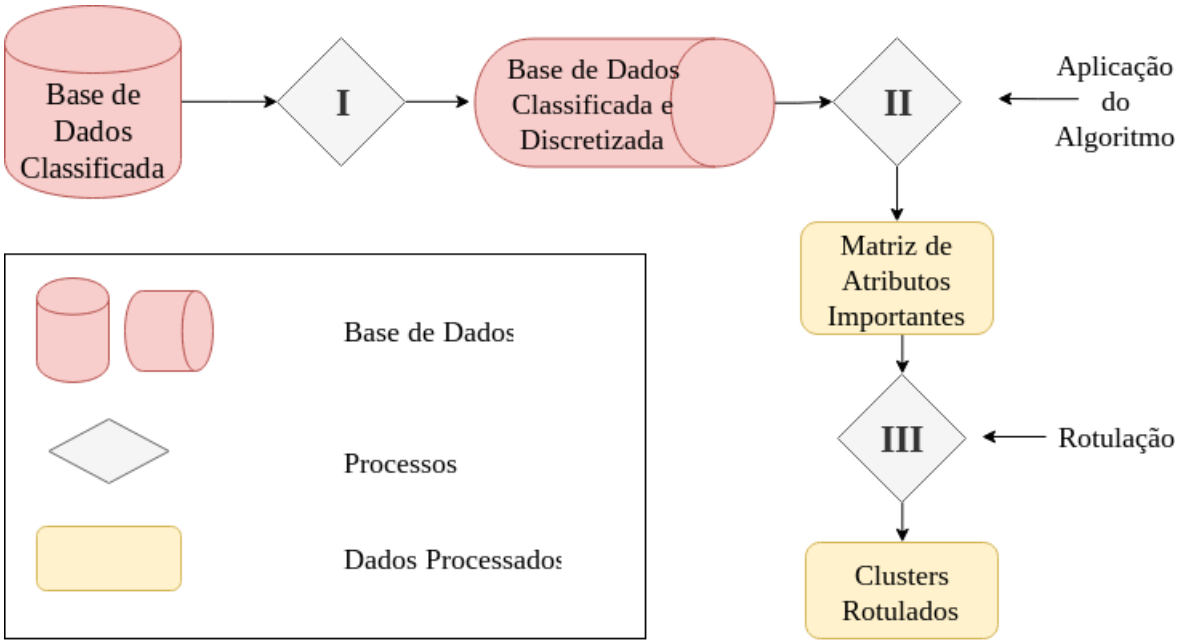


Figura 11 – Modelo de Resolução Proposto

A base de dados[2] do modelo (figura 11) já é classificada de acordo com o repositório UCI - Machine Learning Repository, possuindo valores contínuos, contudo, conforme modelo, aplicará o método de discretização (I). Uma vez com a base discretizada, ocorrerá a divisão em clusters, que nada mais é do que a separação dos grupos já classificados que servirá de entrada para o processo II. É importante ressaltar que os clusters já são definidos na própria base.

No passo II será executado o algoritmo de aprendizagem supervisionado, já visto nas subseções 2.1.1.1, 2.1.1.2 e 2.1.1.3. Essa etapa utiliza a técnica de correlação de atributos, que neste modelo de resolução é considerada um processo de grande importância e será vista na seção 3.3. Logo a saída do processo II gera uma matriz de atributos com seus respectivos valores e através desses valores armazenados é escolhido um, ou mais, atributo de maior relevância.

No processo (III) acontecerá a escolha do atributo mais relevante (maior valor), e podendo, em caso de mesmo valor, ter mais de um atributo relevante. Esta seleção será feita a partir da matriz (Atributos Importantes) criada pela implementação dos algoritmos supervisionados utilizando a técnica de correlação entre atributos seção (3.3), junto com o valor mais frequente desse(s) atributo(s). Após essa etapa é criado um conjunto de rótulos para cada *clusters*.

[2] Base de dados retirada da UCI - Machine Learning Repository. http://archive.ics.uci.edu/ml/

3.3 Técnica de Correlação entre Atributos através de Algoritmos Supervisionados

Essa técnica de correleção entre atributos empregada por (LOPES et al., 2016) utiliza por analogia a aprendizagem supervisionada, na qual os atributos de entrada são correlacionados com um atributo classe (atributo de saída), e conforme o número de atributos do *cluster*, o atributo classe seria alterado seguindo uma sequência do primeiro ao último atributo desse *cluster*. Através desse processo, cada atributo seria classe em relação aos outros atributos, gerando um valor que seria armazenado em uma matriz.

Para exemplificar a técnica é utilizado a figura 12 com os seguintes atributos: **atr1, atr2, atr3 e classe**, portando em um primeiro processamento de três, o primeiro atributo **atr1** se torna a classe da vez, e executado com os outros dois atributos **atr2, atr3** de entrada com um algoritmo supervisionado.

Figura 12 – Exemplo da técnica de correlação aplicada ao atributo, atr1, sendo classe

E na figura 13 ocorre um exemplo do funcionamento da técnica de correlação, em que uma base fictícia possuindo três atributos se correlacionam entre si, e que a quantidade de vezes que o algoritmo é executado é a mesma do número de atributos.

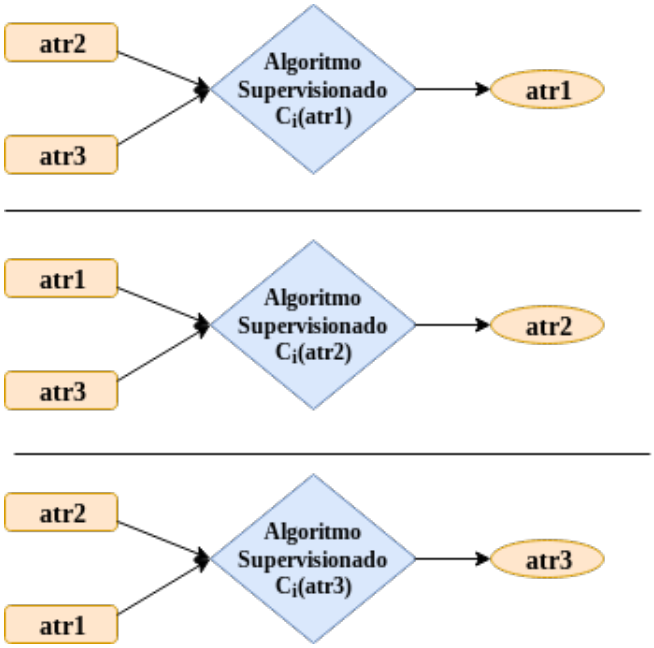


Figura 13 – Exemplo da técnica de correlação aplicada aos atributos atr1, atr2 e atr3.

Em resumo da técnica, o resultado da correlação entre os atributos **atr1, atr2, atr3** é armazanado em uma matriz denominada de **Atributos Importantes**, de acordo com figura 14. Por conseguinte, é realizado a aplicação do algoritmo com **atr1** sendo classe, e assim sucessivamente, até o último atributo (**atr3**). Essa etapa só é finalizada quando todos os atributos tiverem a chance de ser classe, e armazenados seus valores em porcentagem na tabela. Quanto maior sua porcentagem, mais bem correlacionado é o atributo em relação aos demais.

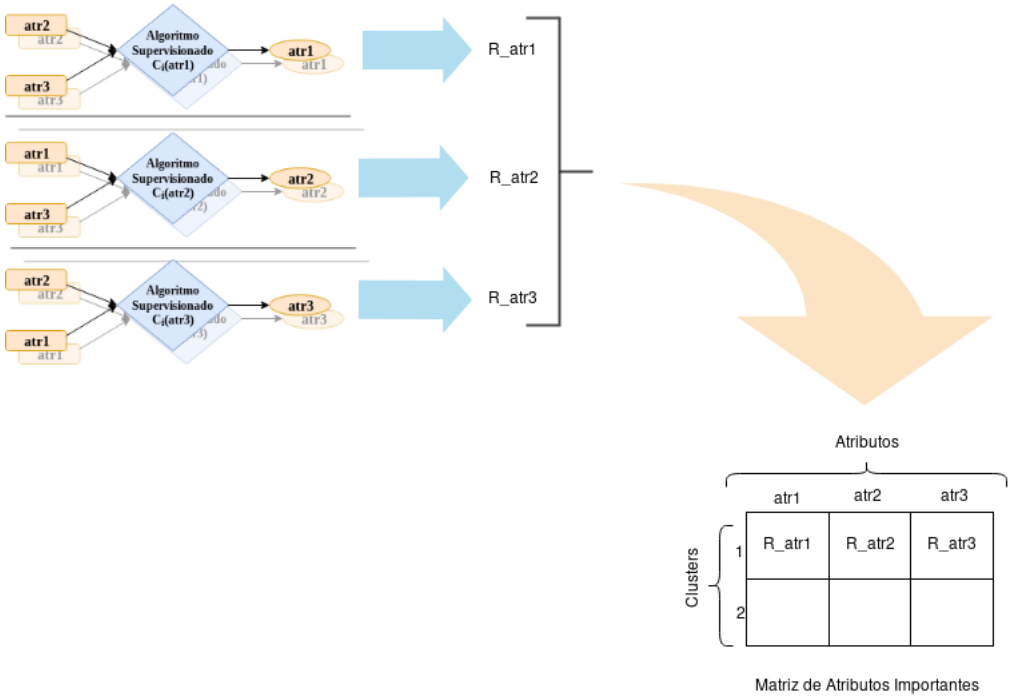


Figura 14 – Montagem da Matriz de Atributos Importantes

De acordo com essa técnica os atributos de um *cluster*, descartando o atributo classe, seriam percorridos um por um, até o último, e a cada iteração de um atributo enquanto classe, é armazenado o valor do resultado em uma matriz (atributos importantes). Essa matriz, após montada, mostraria os resultados de cada atributo enquanto classe e, quanto maior o valor, mais relevante será este atributo em relação aos demais. Essa matriz é definida em linhas, nas quais são representados os *clusters* e as colunas, são os resultados da aplicação do algoritmo de cada atributo classe. Então, caso uma base possua 2 (dois) *clusters* e 3 (três) atributos, a matriz seria de ordem 2x3 (2 linhas e 3 colunas) conforme exemplo da figura 14.

Tal técnica possui um grau de processamento diretamente proporcional à quantidade de características expressa na base de dados definido em R^m descrita na definição 3, em que R representa o conjunto de rótulos e m a dimensão do problema (número de atributos). Ela implica em utilizar todos os atributos, menos o definido como classe, para fazer uma correlação entre eles junto ao algoritmo.

3.4 Exemplo

Para melhor esclarecer as etapas do modelo de resolução exibido na figura 11, será utilizada a tabela 4 como um exemplo prático no modelo proposto nesta pesquisa. Essa tabela é composta por cinquenta registros (linhas) e quatro colunas (atributos), sendo o último um atributo de classe representando o *cluster*. Logo, a primeira coluna da tabela possui o índice da linha da tabela identificando cada registro e outros campos são atributos que definem características desse registro, e a quinta coluna, representando a classe de cada registro.

Tabela 4 – Base de Dados Modelo

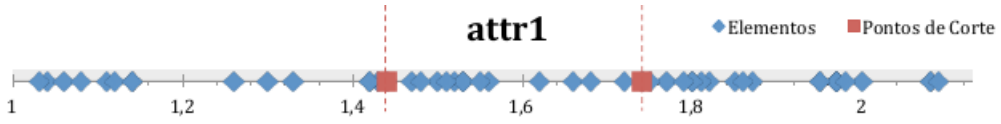
n.	atr1	atr2	atr3	classe	n.	atr1	atr2	atr3	classe
1	2.08	92.11	22.07	2	26	1.42	53.51	19.64	3
2	1.26	85.03	20.45	1	27	1.12	62.71	19.07	1
3	2.00	108.36	22.68	2	28	2.09	60.58	20.20	1
4	1.74	43.78	18.72	3	29	1.95	69.23	19.68	1
5	1.82	100.20	23.09	2	30	1.03	47.81	19.47	3
6	1.43	77.59	21.80	1	31	1.75	90.92	21.39	2
7	1.53	44.01	20.98	3	32	1.72	42.35	22.89	3
8	1.14	107.77	18.99	2	33	1.47	101.77	19.20	2
9	1.97	98.00	22.32	2	34	1.53	41.16	22.67	3
10	1.50	39.67	21.78	3	35	1.44	93.61	21.03	2
11	1.74	55.86	20.31	3	36	1.51	98.65	19.24	2
12	1.80	65.72	19.62	1	37	1.06	68.82	21.68	1
13	1.33	82.01	19.82	1	38	1.48	80.40	21.43	1
14	1.66	103.93	21.10	2	39	1.14	61.59	19.90	1
15	1.42	66.14	21.61	1	40	1.08	91.93	20.81	2
16	1.87	88.36	22.45	2	41	1.62	79.21	18.43	1
17	1.11	107.82	19.32	2	42	1.68	80.87	18.42	1
18	2.08	67.66	20.74	1	43	1.81	98.24	22.13	2
19	1.85	82.65	20.35	1	44	1.30	69.27	18.83	1
20	1.04	102.62	19.46	2	45	1.80	101.21	21.61	2
21	1.97	100.37	21.94	2	46	1.79	72.02	22.02	1
22	1.95	45.70	22.10	3	47	1.56	81.71	22.10	1
23	1.77	50.04	20.16	3	48	1.98	77.16	21.71	1
24	1.97	81.57	19.83	1	49	1.86	89.12	22.84	2
25	1.52	93.13	20.61	2	50	1.55	76.01	19.74	1

Seguindo a definição 2, um elemento é expresso por um vetor de dimensão m, com tamanho igual ao número de atributos. Um exemplo do elemento 2 da tabela 4 pode ser representado por $\vec{e_2} = (1.26, 85.03, 20.45)$.

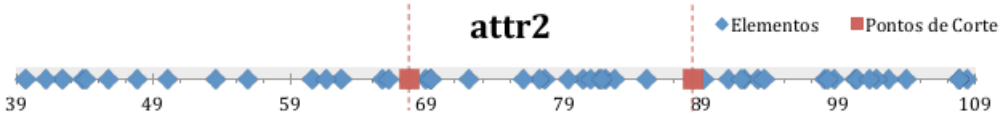
3.4.1 Processo (I) - Discretização

Segundo (CATLETT, 1991; HWANG; LI, 2002), por meio de resultados experimentais, na conversão em atributos discretos ordenados de vários domínios, constatou que a mudança de representação da informação, na maioria das vezes, pode aumentar a acurácia do sistema de aprendizado. Dessa maneira, a etapa de discretização ganha um papel importante no modelo e também no processo de Rotulação (III), pois é utilizada uma inferência na faixa discretizada para encontrar o intervalo na faixa.

Utilizando como exemplo a tabela 4, será utilizada a técnica de discretização por frequências iguais - EFD - e divisão de números de faixas igual a R=3. Na figura 15 pode-se visualizar como é feita a discretização.



(a) Discretização em atr1



(b) Discretização em atr2

(c) Discretização em atr3

Figura 15 – Discretização de atributos utilizando EFD com R = 3 (Figura adaptada de (LOPES et al., 2016))

Por meio da figura 15 fica claro o conteúdo da faixa 1, contendo o valor inicial, 1(um), até o primeiro ponto de corte. Na faixa 2, o valor inicial é o primeiro número após o primeiro ponto de corte (término da faixa 1) até o segundo ponto de corte, incluindo o próprio ponto de corte. E a faixa 3 contém todos valores a partir do segundo ponto de corte.

A tabela 5 é o resultado após a discretização de todos os atributos. Para cada base de dados será definido o número de faixas de acordo com a configuração inicial antes da execução. Nessa configuração do sistema, o número de faixas serve para toda a base de dados e não para cada atributo, então nesse exemplo o valor de $R = 3$, conforme figura 15. A escolha de R, neste exemplo, é por conta dos valores de cada atributo, pois com três faixas os dados ficam melhores distribuídos entre as faixas, ao contrário de quatro ou mais faixas. Também nesse exemplo, o R possui é o mesmo tanto no **atr1**, no **atr2** e **atr3**, conforme tabela 6.

Tabela 5 – Base de Dados Modelo Discretizada

	atr1	atr2	atr3	classe		atr1	atr2	atr3	classe
1	3	3	3	2	26	1	1	1	3
2	1	2	2	1	27	1	1	1	1
3	3	3	3	2	28	3	1	2	1
4	2	1	1	3	29	3	2	1	1
5	3	3	3	2	30	1	1	1	3
6	1	2	3	1	31	3	3	2	2
7	2	1	2	3	32	2	1	3	3
8	1	3	1	2	33	2	3	1	2
9	3	3	3	2	34	2	1	3	3
10	2	1	3	3	35	1	3	2	2
11	2	1	2	3	36	2	3	1	2
12	3	1	1	1	37	1	2	3	1
13	1	2	1	1	38	2	2	2	1
14	2	3	2	2	39	1	1	2	1
15	1	1	2	1	40	1	3	2	2
16	3	2	3	2	41	2	2	1	1
17	1	3	1	2	42	2	2	1	1
18	3	1	2	1	43	3	3	3	2
19	3	2	2	1	44	1	2	1	1
20	1	3	1	2	45	3	3	2	2
21	3	3	3	2	46	3	2	3	1
22	3	1	3	3	47	2	2	3	1
23	3	1	2	3	48	3	2	3	1
24	3	2	2	1	49	3	3	3	2
25	2	3	2	2	50	2	2	1	1

Tabela 6 – Valores das faixas com R=3 da Base de Dados Modelo

	Faixa 1	**Faixa 2**	**Faixa 3**
atr1	[1.03 ∼1.44]	] 1.44 ∼1.74]	] 1.74 ∼2.09]
atr2	[39.67 ∼67.66]	] 67.66 ∼88.36]	] 88.36 ∼108.36]
atr3	[18.42 ∼19.82]	] 19.82 ∼21.61]	] 21.61 ∼23.09]

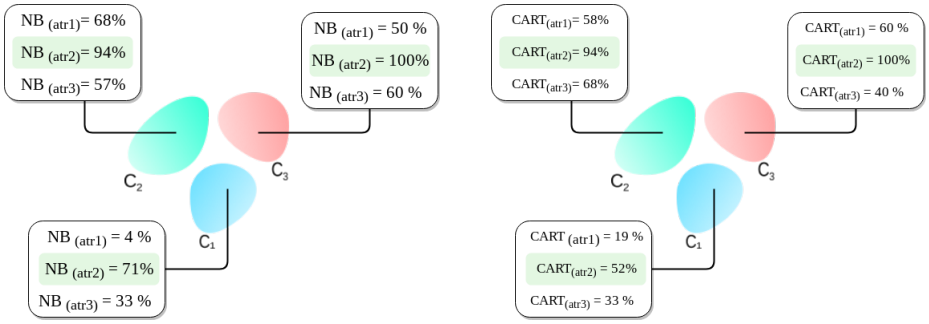
3.4.2 Processo (II) - Algoritmos Supervisionados

Ao chegar nessa etapa, Processo (II) do modelo já apresentado, já se tem uma base discretizada e clusters formados como visto na tabela 5. A partir desta etapa é feita a execução do algoritmo de aprendizado supervisionado obtendo como saída um valor, em porcentagem, informando o grau de correlacionamento entre os atributos compondo uma matriz, cuja função é armazenar o resultado da execução dos algoritmos utilizando a técnica de correlação de atributos (3.3).

O algoritmo irá selecionar *cluster* por *cluster*, percorrendo todos os atributos destes clusters, em que a cada iteração um atributo será a classe da vez. Nesse exemplo,

primeiramente, o atributo **atr1** será classe e os demais irão participar como entrada junto ao algoritmo, verificando seu grau de importância entre eles, através da do valor em porcentagem junto a tabela. Depois o atributo **atr2** irá ser classe, seguido do **atr3**, fechando o ciclo de todos os atributos do *cluster*.

Como amostra deste exemplo, serão executados dois algoritmos supervisionados para comparar os resultados (figura 16). Assim, apresentar-se-á valores em percentual dos atributos enquanto classe de cada *cluster*, após isso, será escolhido o maior valor definindo-se o atributo rótulo.



(a) Execução do Naive Bayes

(b) Execução do algorimo de Árvore de Decisão

Figura 16 – Resultado dos Algoritmos

A figura 16a mostra o resultado da execução do Naive Bayes trabalhando com a base modelo (tabela 5), exibindo os resultados em porcentagem de acerto de cada atributo em relação aos demais. O mesmo acontece com a figura 16b, em que é aplicado um algoritmo de Árvore de Decisão - CART - exibindo o resultado de todas as taxas de acerto, em porcentagem, dos atributos de seus respectivos clusters.

3.4.3 Processo (III) - Rotulação

No processo de rotulação, os rótulos de cada *cluster* (c_i) serão compostos conforme o equação 3.1.

$$r_{ci} = \{(a_1, [p_1, q_1]), ..., (a_{m^{(c_i)}},]p_{m^{(c_i)}}, q_{m^{(c_i)}}])\} \tag{3.1}$$

Cada rótulo é composto pela tupla: atributo de maior relevância (matriz de atributos importantes) e a faixa de valor desse atributo que mais se repete. Na figura 16 os rótulos em destaque são os que possuem maior valor, ademais, cada atributo que faz parte do rótulo possui um vetor de valores, do qual será escolhido a faixa de maior ocorrência. Uma vez calculada e definida a faixa, serão determinados os limites inferiores ($p_{m^{(c_i)}}$) e superiores ($q_{m^{(c_i)}}$), de acordo com a tabela discretizada (exemplo 6).

Por exemplo, utilizando a Base Modelo, mais especificamente o *cluster* 1 (c_1), cujo resultado é apresentado na figura 16a, o rótulo apresentado é o atributo **atr2** com a **faixa**

2, faixa esta encontrada após cálculo dos elementos de maior ocorrência, conforme descrito no parágrafo acima.

O rótulo apresentado ao final do processo terá a substituição do número da faixa pelos valores do intervalo conforme a tabela 6. Os rótulos dos *clusters* descrito neste exemplo - conforme figura 16a e figura 16b - aplicado na BD Modelo são:

- $r_{c_1} = (atr2,]67.66, 88.36])$;
- $r_{c_2} = (atr2,]88.36, 108.36])$;
- $r_{c_3} = (atr2, [39.67, 67.66])$;

A representação acima do rótulo informa que no rótulo do *cluster* 1 é uma tupla composta pelo atributo **atr2** e a faixa variando de valor maior que 67,66 até 88,36. No *cluster* 2 verifica-se o rótulo composto também pelo atributo **atr2**, mas com faixa diferente, variando de um valor maior que 88,67 até 108,36. E, por último, o rótulo do *cluster* 3, com a faixa variando de 39,67 até 67,66. Isso significa que qualquer registro com valor no **atr2** no intervalo]67.66,88.36] será agrupado no *cluster* 1, podendo ajudar um analista a interpretar ou tomar decisão em conformidade a esse rótulo. Então, caso esse **atr2** fosse localização e esse intervalo determinasse uma região, isso poderia ser utilizado para tomada de decisão, dependendo do problema que queira resolver.

O algoritmo 1 exibe a rotina em forma de pseudocódigo para melhor entendimento, mas as variáveis V, R e $TipoDiscretização$, não aparecem inicializadas por serem variáveis que dependem de testes para melhor otimização dos rótulos. A variância V foi a forma de eliminar uma possível ambiguidade entre os clusters, pois é utilizada para seleção de mais de um atributo rótulo na matriz de atributos importantes, caso aconteça dos rótulos se repetirem em clusters diferentes. Logo, todos os atributos que tiverem até uma diferença V em relação ao atributo de maior taxa de acerto, expresso em porcentagem, serão escolhidos como rótulo. Isto posto, se o atributo de maior taxa de acerto possuir 90%, e o $V = 10\%$ então todos outros atributos da matriz que tiverem valores a partir de 80% são selecionados como rótulo do cluster. A variável R é utilizada para definir em quantas faixas serão divididos os valores do atributo na discretização, e por fim a variável $TipoDiscretização$, já explicada anteriormente.

Algorithm 1: Rotina de Rotulação

```
1  Carrega_valores_auxiliares(V, R, TipoDiscretização);
2  Carrega_BD;
3  Discretiza_BD;
4  Separa_em_clusters_de_acordo_com_classificação_BD;
5  while existir clusters do
6      while existir atributos do
7          atributo_classe=seleciona_nova_classe(atributos) ;
8          Aplica_algoritmo_supervisionado(atributo_classe, atributos_naoClasse);
9          Calcula_matriz_de_porcentagem_de_acertos;
10     if V!=0 then
11         Carrega_atributos_importantes_considerando_V;
12     Associa_valores_aos_intervalos;
13 Exibe_rótulos_todos_clusters;
```

4 Resultados e Discussões

Este estudo conta com a realização de testes nas bases de dados da UCI Machine Learning[1] - um repositório de dados a serviço da comunidade de aprendizado de máquina criado por estudantes de pós-graduação na UC Irvine em 1987 e que é utilizado por educadores e pesquisadores como fonte primária de aplicações de aprendizado de máquina.

As bases de dados escolhidas para este trabalho foram: Seeds, Iris, Glass, Wine. Essas bases são bastante utilizadas em vários trabalhos[2] e por meio delas há possibilidade de análise de seus comportamentos mediante resultados, visto que todas as bases são classificadas e já possuem seus grupos definidos, servindo de referência para comparar os resultados desta pesquisa, sem esquecer que o trabalho é com os *clusters* já formados e não na formação deles. Outro motivo desta seleção das bases é mitigar ao máximo a utilização das técnicas de *Feature Engineering* (CASARI; ZHENG, 2018), que são técnicas de manipulação de dados empregadas na base para aplicação do algoritmo de aprendizado de máquina utilizado na fase de preparação dos dados.

As tabelas apresentadas como resposta dos algoritmos de rotulação para cada base de dados possuem o seguinte formato: a primeira coluna (**Cluster**) é um número que representa cada *cluster* de forma sequencial; a segunda coluna (**Rótulos**) é representada pelo par **Atributo** e **Faixa**, o qual define o rótulo do *cluster* respectivo, podendo haver ou não mais atributos para compor o rótulo; a coluna **Relevância** tem a importância de informar o quanto, em porcentagem, aquele atributo é relacionado aos demais na definição do rótulo daquele *cluster*. A quinta coluna, **Fora da Faixa**, exibe o erro em quantidade de elementos que não estão naquela faixa definida naquele rótulo e, por último, a **Acurácia Parcial**, que exibe, em porcentagem, o valor de acertos dos elementos que são representados pelo atributo e faixa da respectiva linha.

A obtenção da acurácia dos *clusters* só é possível porque as bases de dados já possuem seus grupos definidos, por conseguinte, há o conhecimento de quais elementos fazem parte de um determinado grupo. Dessa maneira, qualquer resultado da aplicação dos algoritmos nos clusters pode ser confrontado com as informações dos *clusters* inicialmente retirados da *UCI Machine Learning* e, assim, poderá ser calculada a porcentagem de acertos.

A divisão deste capítulo iniciará por uma explanação da implementação do trabalho, explicando as ferramentas utilizadas no desenvolvimento com configurações de algumas variáveis e cada seção referir-se-á a uma base de dados utilizada, sendo esta dividida em

[1] http://archive.ics.uci.edu/ml/

[2] https://archive.ics.uci.edu/ml/datasets/glass+identificpation, https://archive.ics.uci.edu/ml/datasets/iris, https://archive.ics.uci.edu/ml/datasets/wine, https://archive.ics.uci.edu/ml/datasets/seeds

subseções para os seguintes algoritmos: Naive Bayes, CART e KNN.

4.1 Implementação

Para gerar os resultados aqui descritos, foram realizadas implementações utilizando-se a ferramenta MATLAB[3], sendo possível utilizar suas funções de aprendizado de máquina já implementadas na *Statistics and Machine Learning Toolbox* . Por apresentar linguagem técnica e funções já prontas direcionadas para aprendizado de máquina,essa ferramenta foi selecionada para colocar em prática essa pesquisa.

Ao longo da pesquisa foram realizados vários testes, porém, houve alterações de algumas variáveis e métodos de discretização, sempre com o objetivo de obter os melhores resultados. As variáveis e métodos alterados são: variância (V), número de faixas (R) e tipo de discretização $TipoDiscretização$ (EWD,EFD).

Fazendo um breve resumo sobre as configurações das variáveis, a variação V existe para evitar a ambiguidade dos rótulos, ou seja, para quando os rótulos apresentarem os mesmos resultados: atributo e faixa de valor. O número de faixas (R) é definido de forma que os atributos tenham seus valores mapeados de acordo com a faixa de valor correspondente e, uma vez definido o número de faixas, será escolhido qual método de discretização será aplicado (EWD,EFD), decidindo-se com qual faixa cada valor irá ficar.

4.2 Base de Dados 1 - Seeds - Identificação de Tipos de Semente

Essa base é composta por sete atributos definindo suas características e mais um: a classe, que é responsável por identificar os tipos de sementes (CHARYTANOWICZ et al., 2010). Esses elementos foram construídos a partir de sete parâmetros geométricos medidos dos grãos de trigo: *area* - A, *perimeter* - P, *compactness* - C, *length of kernel* - Lkernel, *width of kernel* - Wkernel, *asymmetry coefficient* - asymetry, *length of kernel groove* - lkgroove.

Os valores dos atributos são todos contínuos e não existem valores em branco, possuindo um total de 210 registros classificados em três categorias bem distribuídas entre as classes, definindo-se estas como base de dados balanceada, por conta dessa distribuição dos registros:

- 70 elementos do tipo *Kama*;
- 70 elementos do tipo *Rosa*;
- 70 elementos do tipo *Canadian*.

[3] http://www.mathworks.com/products/matlab/ ; versão: R2016a(9.0.0.341360); 64-bit (glnxa64)

Para classificar as sementes como *Kama*, *Rosa* e *Canadian*, foi utilizada uma técnica de raio X que é relativamente mais barata que outras técnicas de imagem, como microscopia ou tecnologia a laser. O material foi colhido de campos experimentais, explorados no Instituto de Agrofísica da Academia Polonesa de Ciências em Lublin.

As tabelas com os resultados dos rótulos apresentadas a seguir (tabelas 7,8, 9) exibem os resultados de rotulação dos algoritmos Naive Bayes, CART e KNN, respectivamente, e no caso dos *clusters*, cada linha determina um grupo: (1) *Kama*, (2) *Rosa* e (3) *Canadian*. Como já mencionado neste capítulo, na seção 4.1, antes de executar o algoritmo, algumas configurações são necessárias para executar o algoritmo, como a do método de discretização do tipo EFD, e também a divisão dos valores dos atributos em faixas com $R = 3$ para todos os atributos e variação $V = 0\%$

4.2.1 Naive Bayes

Na tabela 7 pode-se verificar o resultado da aplicação do algoritmo e através dela percebe-se que o pior rótulo foi no *cluster* 1 com acurácia de 70%, e com isso deixa de representar nove registros no total de setenta amostras, mas nos outros dois *clusters* a acurácia sobe para 88% com um erro de nove registros, porém, os rótulos nestes *clusters* são diferentes, coincidindo somente os valores das colunas **Fora da Faixa** e **Acurácia Parcial**.

Tabela 7 – Resultado da rotulação com o algoritmo Naive Bayes

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
1	Lkernel	] 5.357 ∼ 5.826]	2.85%	21	70%
2	lkgroove	] 5.791 ∼ 6.55]	8.57%	9	87.15%
3	wkernel	[2.63 ∼ 3.049]	4.28%	9	87.15%

A última coluna, **Acurácia Parcial**, apresenta em porcentagem o grau de acerto, por *cluster*, dos registros que são representados pelo rótulo. Analisando-se a coluna **Relevância** da tabela 7 nota-se que os valores são baixos, contudo, uma vez escolhido o atributo rótulo o grau de confiabilidade passa a ser o número de erros do rótulo, pois quanto menores são os erros mais representabilidade possui o rótulo. Dessa forma, o menor valor de acurácia ficou em 70% (setenta por cento) no *cluster* 1, indicando que dentre as setenta amostras, 21 (vinte e um) não são representadas pelo rótulo, e nos outros *clusters* acima de 87% (oitenta e sete por cento).

Segue abaixo o resultado do algoritmo Naive Bayes na base de dados **Seeds** com seus rótulos:

- $r_{c_1} = \{(Lkernel,]5.357 \sim 5.826])\}$
- $r_{c_2} = \{(lkgroove,]5.791 \sim 6.55])\}$

- $r_{c_3} = \{(wkernel, [2.63 \sim 3.049])\}$

4.2.2 Classification and Regression Trees - CART

Na tabela 8 que segue o mesmo modelo da tabela visto anteriormente e apresenta o resultado da aplicação do algoritmo supervisionado na base Seeds. O CART é utilizado pela toolbox do MATLAB como algoritmo de classificação de árvore de decisão.

Tabela 8 – Resultado da aplicação do algoritmo CART

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
1	perimetro	] 13.73 ∼ 15.18]	5.71%	14	80%
2	lkgroove	]5.791 ∼ 6.55]	5.71%	9	87.15%
3	perimetro	[12.41 ∼ 13.73]	4.28%	5	92.8%

Analisando os rótulos encontrados percebe-se que as acurácias do CART são melhores aos do algoritmo visto anteriormente, e na coluna **Atributos** há uma repetição do **perimetro** em dois clusters: *Kama* e *Canadian*. Na situação apresentada na tabela 8 esses atributos repetidos não representam as mesmas amostras na base de dados, porque mesmo com o rótulo possuindo o mesmo atributo as faixas de cada atributo são diferentes tornando-os rótulos distintos.

O resultado da rotulação utilizando o algoritmo CART na base de dados **Seeds** tem como rótulos:

- $r_{c_1} = \{(perimetro,]13.73 \sim 15.18])\}$
- $r_{c_2} = \{(lkgroove,]5.791 \sim 6.55])\}$
- $r_{c_3} = \{(perimetro, [12.41 \sim 13.73])\}$

4.2.3 K-Nearest Neighbor - KNN

Na tabela 9 exibe o resultado da aplicação do KNN na base Seeds e os rótulos encontrados são os mesmos do CART com uma diferença somente nos dados de relevância do atributo, o que não muda o grau de representatividade do rótulo.

Tabela 9 – Resultado da aplicação do algoritmo KNN

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
	Atributos	Faixa			
1	perimetro	] 13.73 ∼ 15.18]	12.85%	14	80%
2	lkgroove	]5.791 ∼ 6.55]	7.14%	9	87.15%
3	perimetro	[12.41 ∼ 13.73]	5.71%	5	92.8%

O resultado da rotulação utilizando o algoritmo KNN na base de dados **Seeds** tem como rótulos:

- $r_{c_1} = \{(perimetro,]13.73 \sim 15.18])\}$
- $r_{c_2} = \{(lkgroove,]5.791 \sim 6.55])\}$
- $r_{c_3} = \{(perimetro, [12.41 \sim 13.73])\}$

4.2.4 Comparativo entre Algoritmos na Base de Dados Seeds

O CART e KNN foram dentre os três algoritmos, os que obtiveram os rótulos com melhores acurácias, que por sua vez tiveram os mesmos rótulos em todos os três grupos: *Kama*, *Rosa* e *Canadian*. Analisando-se os resultados: o Naive Bayes no *cluster* 1, $r_{c_1} = \{(Lkernel,]5.357 \sim 5.826])\}$, foi o que obteve maior número de elementos fora da faixa, isso quer dizer que no total de setenta amostras de dados no grupo, vinte e um ficaram de fora do rótulo, resultando na menor acurácia entre os três algoritmos que tiveram quatorze amostras cada um; no *cluster* 2 os três algoritmos coincidiram os resultados dos rótulos com nove erros cada; e no *cluster* 3 (*Canadian*) mais uma vez CART e KNN tiveram rótulos com acurácias mais altas com número de erros iguais a cinco registros nos dois algoritmos, comparado com nove registros do Naive Bayes.

Para compreender melhor a representatividade do rótulo nos *clusters* foi posto alguns gráficos apresentando o comportamento da base de dados em relação a alguns atributos. Com estes gráficos é possível visualizar se o rótulo está de fato representando um determinado grupo. Os gráficos são dispostos em duas dimensões e o atributo referente a cada eixo foi escolhido para melhor representar a amostra.

O gráfico da figura 17 contém uma relação dos atributos **Lkernel** com **lkgroove** e representa a amostra de dados dos três grupos: *Kama*, *Rosa* e *Canadian*, contudo o objetivo deste gráfico é apresentar o comportamento do rótulo no Naive Bayes, $r_{c_1} = \{(Lkernel,]5.357 \sim 5.826])\}$ no *cluster* 1, e poder visualizar a representatividade do rótulo no grupo.

Na situação da figura 17 , pode-se perceber que o grupo *kama* (cluster 1) contém alguns erros, pois logo no início da faixa $(Lkernel,]5.357 \sim 5.826])$ é fácil visualizar que alguns dados ficam fora da faixa indicando que não estão sendo representados pelo rótulo

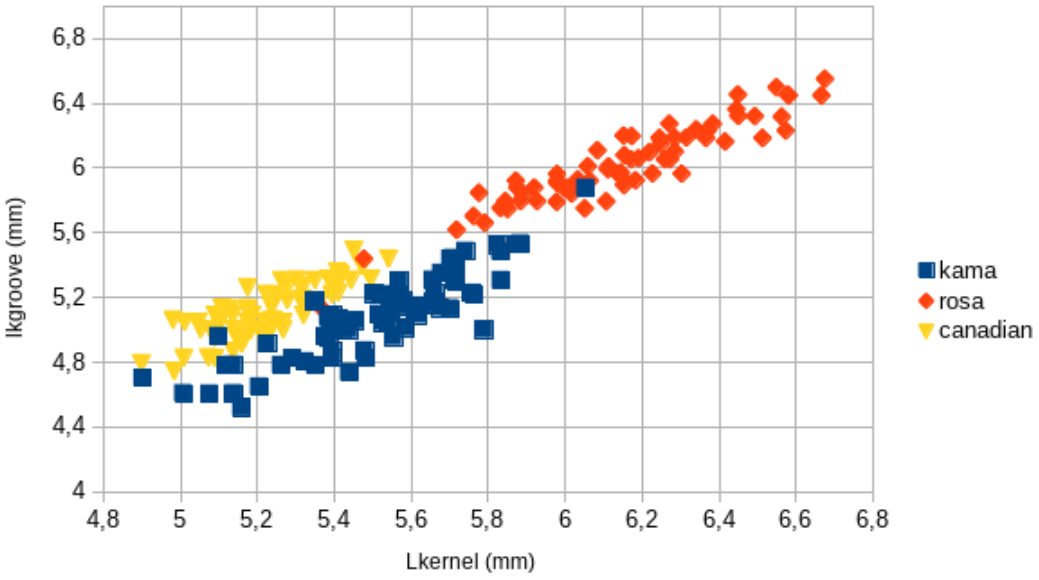


Figura 17 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove**

ocasionando os erros, contudo é possível ter uma impressão generalizada de todos os registros que o rótulo representa através dos valores de intervalo do rótulo.

Na figura 18 também apresenta a base Seeds distiguindo os grupos pelas formas geométricas: quadrados (*Kama*), losango (*Rosa*) e triângulos (*Canadian*), mas com uma relação entre **lkgroove** e **wkernel** que fazem parte dos resultados de rotulação do Naive Bayes no *cluster* 2 $r_{c_2} = \{(lkgroove,]5.791 \sim 6.55])\}$ e no *cluster* 3 $r_{c_3} = \{(wkernel, [2.63 \sim 3.049])\}$.

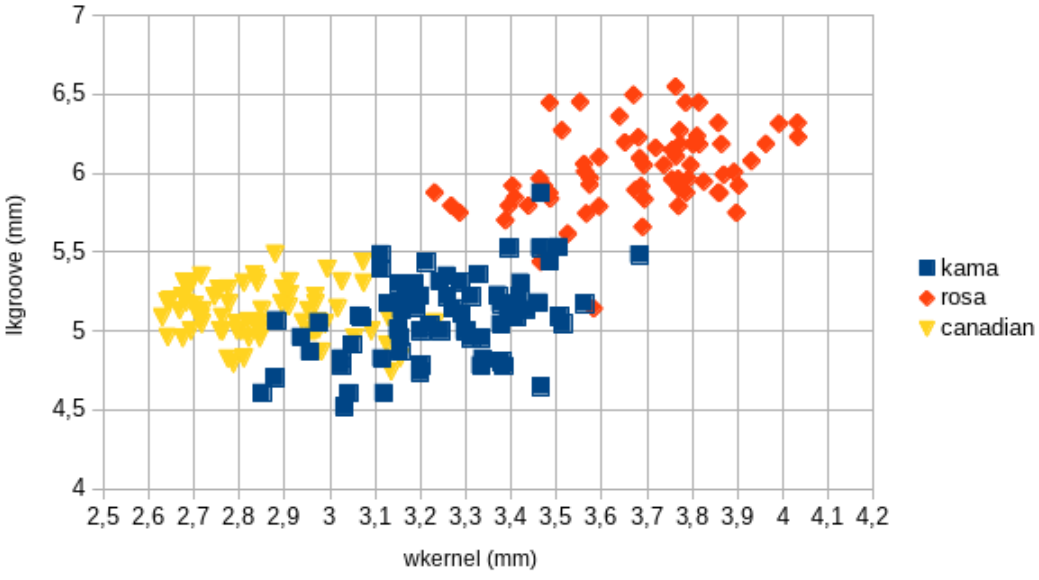


Figura 18 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove**

O atributo **lkgroove** faz parte do rótulo encontrado no grupo *Rosa* (*cluster* 2) pelo Naive Bayes, representado no gráfico pelo eixo Y da figura 18 e sua faixa de dados é de fácil visualização no gráfico. Os valores do eixo Y acima de 5.791 até 6.55 são representados por r_{c_2} do grupo Rosa figurado pelo losango no gráfico da figura 18. Já o r_{c_3} possui o atributo **wkernel** representado no gráfico no eixo X possuindo uma faixa de dados a partir de 2.63 até 3.049, e através desse intervalo é possível notar que há alguns dados não pertencentes ao r_{c_3} do grupo Canadian. Com o gráfico fica fácil perceber que na disposição das amostras o grupo que mais se mistura é o *Kama*, implicando também na menor acurácia entre os rótulos.

No outro gráfico, figura 19, são relacionados dois eixos X e Y com os atributos **perimetro** e **lkgroove** respectivamente, e através desse plano cartesiano é possível identificar todos os grupos da base Seeds. Os dois algoritmos CART e KNN possuem os mesmos resultados de rotulação para os três grupos mediante os dois atributos que estão representados no gráfico, dessa forma, com essa amostra de dados é possível representar o espaço de dados dos rótulos encontrados em cada algoritmo.

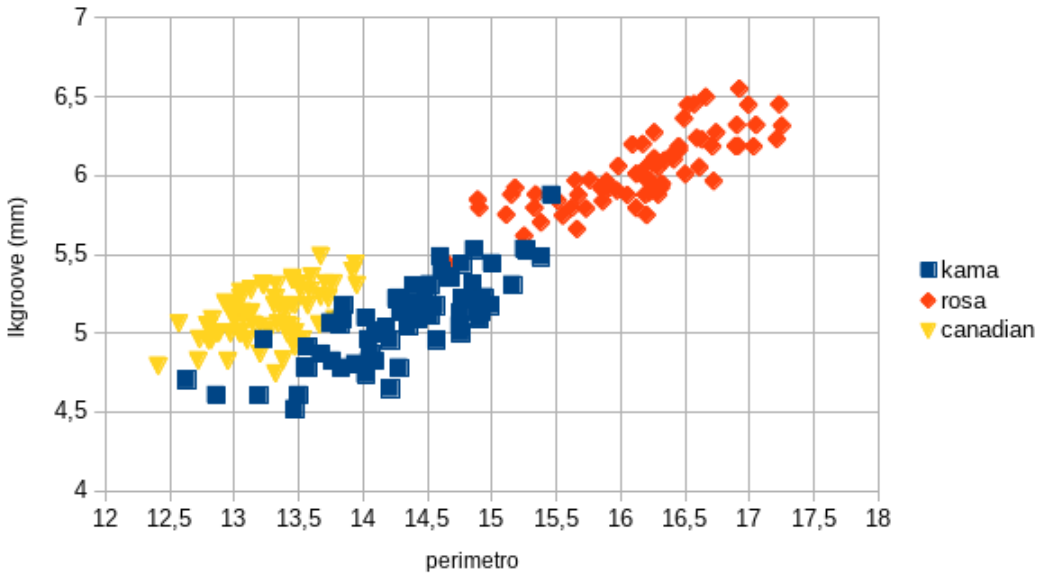


Figura 19 – Gráfico da disposição de elementos da Base Seeds entre os eixos **Lkernel** e **lkgroove**

Os elementos do rótulo $r_{c_1} = \{(perimetro,]13.73 \sim 15.18])\}$, representam o grupo *Kama*, que estão simbolizados de quadrados no gráfico, e através do atributo **perimetro** no intervalo de]13.73 até 15.78] consegue-se visualizar no gráfico a amostra que o rótulo representa do *cluster* 1 (*Kama*), como também, alguns dados que não estão neste intervalo configurando os erros.

No *cluster* 2 (*Rosa*) figurando no gráfico como um losango pode-se perceber, através de seu rótulo $r_{c_2} = \{(lkgroove,]5.791 \sim 6.55])\}$, que o atributo **lkgroove** e seu

intervalo estão bem definidos no gráfico. Qualquer valor no eixo Y no intervalo, a partir de 5.6791 até 6.55, é representado pelo rótulo r_{c_2}, mas mesmo com alguns erros visíveis na amostra, esse grupo foi o que ficou melhor delineado no gráfico e de fácil visualização para comprovar a representatividade dos rótulos encontrados pelos algoritmos CART e KNN. Já no *cluster* 3 *Canadian* não é diferente mediante ao rótulo encontrado pelos dois algoritmos $r_{c_3} = \{(perimetro, [12.41 \sim 13.73])\}$. A partir da faixa do r_{c_3} no atributo **perimetro** as amostras representadas iniciam em 12.41 até 13.73, mas mesmo no valor final do intervalo onde pode-se encontrar alguns erros, é possível mostrar que o r_{c_3} consegue representar o grupo *Canadian*.

4.3 Base de Dados 2 - Iris - Identificação de Tipos de Plantas

A base de dados Iris pertence a UCI Machine Learning, já usada em trabalhos como Lopes et al. (2016), Filho et al. (2015), e utilizada em reconhecimento de padrões de plantas por lidar com classes bem definidas, pois contém 3 classes de 50 instâncias cada, totalizando 150 registros de amostra de plantas. A base possui amostras com 3 tipos (FISHER, 1936):

- 50 elementos da classe *Iris-setosa* ;
- 50 elementos da classe *Iris-versicolor*;
- 50 elementos da classe *Iris-virginica*.

Os atributos correspondentes são comprimento da sepala - SL, largura da sepala - SW, comprimento da pétala - PL e largura da pétala - PW, e através dessas características há uma classificação para dizer se o tipo de planta é: *Iris-setosa*, *Iris-versicolor* e *Iris-virginica*.

Seguindo a análise, semelhante a da base de dados anterior, foram realizados testes utilizando os algoritmos Naive Bayes, CART e KNN e seus resultados depositados em tabelas divididas em colunas no mesmo formato das anteriores: **Cluster**, **Rótulo**, **Relevância**, **Fora da Faixa** e **Acurácia Parcial**. Os *clusters* são dispostos em números correspondentes a seguinte sequência: (1) *Iris-setosa*, (2) *Iris-versicolor* e (3) *Iris-virginica*.

Para gerar os resultados, da aplicação do algoritmo na base de dados, foi definido o método de discretização tipo EWD (seção 2.2.1), a divisão em três faixas de valores $R = 3$ para todos os atributos e inserido o valor de variação $V = 0\%$.

4.3.1 Naive Bayes

Através da tabela 10 são exibidos os resultados da rotulação de dados de todos os grupos: *cluster* 1 (*Iris-setosa*), *cluster* 2 (*Iris-versicolor*) e *cluster* 3 (*Iris-virginica*).

O Naive Bayes encontrou o mesmo atributo para cada rótulo, mas os intervalos foram diferentes assegurando que os rótulos se tornassem distintos.

Tabela 10 – Resultado da aplicação do algoritmo Naive Bayes

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
1	petalwidth	[0.1 ∼ 0.9]	42%	0	100%
2	petalwidth	] 0.9 ∼ 1.7]	36%	2	94%
3	petalwidth	] 1.7 ∼ 2.5]	20%	4	92%

O fato dos rótulos apresentarem o mesmo atributo pode ser analisado na tabela 11, que expõe os valores de correlacionamento entre atributos, esta é a matriz de atributos importante, e através desta que é escolhido o atributo de maior valor. Esta tabela é dividida em linhas representando os grupos e colunas representando os atributos.

Tabela 11 – Matriz de Atributos Importantes do algoritmo Naive Bayes na base Iris

		Atributos			
		sepallength	sepalwidth	petallength	petalwidth
Clusters	1	14	16	26	**42**
	2	16	12	14	**36**
	3	14	16	14	**20**

A tabela 11 mantém os resultados de correlacionamento entre os atributos, e quando ela está toda preenchida, é realizada uma varredura por linha para saber qual o maior valor, servindo para indicar qual o atributo será o rótulo, desta forma como todos os maiores números estão na coluna do atributo **petalwidth**, este atributo é o rótulo de cada *cluster*.

Os rótulos com o algoritmo Naive Bayes na base de dados **Iris** são dados abaixo:

- $r_{c_1} = \{(petalwidth, [0.1 \sim 0.9])\}$
- $r_{c_2} = \{(petalwidth,]0.9 \sim 1.7])\}$
- $r_{c_3} = \{(petalwidth,]1.7 \sim 2.5])\}$

4.3.2 Classification and Regression Trees - CART

A aplicação do algoritmo CART na base de dados **Iris** gerou a tabela 13 como resultado, e ao examinar pode-se observar uma semelhança com a rotulação do algoritmo anterior.

Ao se analisar a tabela percebe-se a importância do atributo **petalwidth** para essa base, pois este, tem participação nos três *clusters*: *Iris-setosa*, *Iris-versicolor*, *Iris-virginica*, e também diferenciados apenas pelo intervalo de dados. Este resultado para um especialista ajudaria em uma tomada de decisão, pois a partir da leitura destes rótulos, qualquer novo

Tabela 12 – Resultado da aplicação do algoritmo CART

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
1	petalwidth	[0.1 ∼ 0.9]	38%	0	100%
2	petalwidth	] 0.9 ∼ 1.7]	24%	2	94%
3	petalwidth	] 1.7 ∼ 2.5]	24%	4	92%

registro basta verificar o valor da largura da pétala (*petalwidth*) e assim poderá classificá-lo em algum grupo.

Segue abaixo os rótulos na base de dados **Iris** aplicado pelo algoritmo CART:

- $r_{c_1} = \{(petalwidth, [0.1 \sim 0.9])\}$
- $r_{c_2} = \{(petalwidth,]0.9 \sim 1.7])\}$
- $r_{c_3} = \{(petalwidth,]1.7 \sim 2.5])\}$

4.3.3 K-Nearest Neighbor - KNN

No resultado da rotulação utilizando KNN constata-se os mesmos rótulos dos dois algoritmos Naive Bayes e CART, mas com algumas diferenças nas relevâncias de atributos, porém na coluna **Fora da Faixa** não houveram alterações.

Tabela 13 – Resultado da aplicação do algoritmo KNN

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
1	petalwidth	[0.1 ∼ 0.9]	50%	0	100%
2	petalwidth	] 0.9 ∼ 1.7]	36%	2	94%
3	petalwidth	] 1.7 ∼ 2.5]	24%	4	92%

Segue abaixo os rótulos na base de dados **Iris** aplicado pelo algoritmo KNN:

- $r_{c_1} = \{(petalwidth, [0.1 \sim 0.9])\}$
- $r_{c_2} = \{(petalwidth,]0.9 \sim 1.7])\}$
- $r_{c_3} = \{(petalwidth,]1.7 \sim 2.5])\}$

4.3.4 Comparativo entre Algoritmos na Base de Dados Iris

Os resultados dos algoritmos na criação dos rótulos, nesta base de dados, foram bastantes concisos entre eles, cada rótulo possui somente um atributo e faixa de valor representando o grupo, e os três algoritmos: Naive Bayes, CART e KNN, tiveram exatamente os mesmos rótulos em todos os três *clusters*.

Uma forma de comprovar a qualidade do rótulo e mostrar que de fato ele cumpre seu papel de identificar o grupo, é colocar em prática um exemplo que pode ser visto na

figura 20. Para essa análise o gráfico exibe a relação de dois atributos representados pelos eixos X e Y, facilitando a visualização do comportamento dos três grupos: *Iris-setosa*, *Iris-versicolor* e *Iris-viginica*.

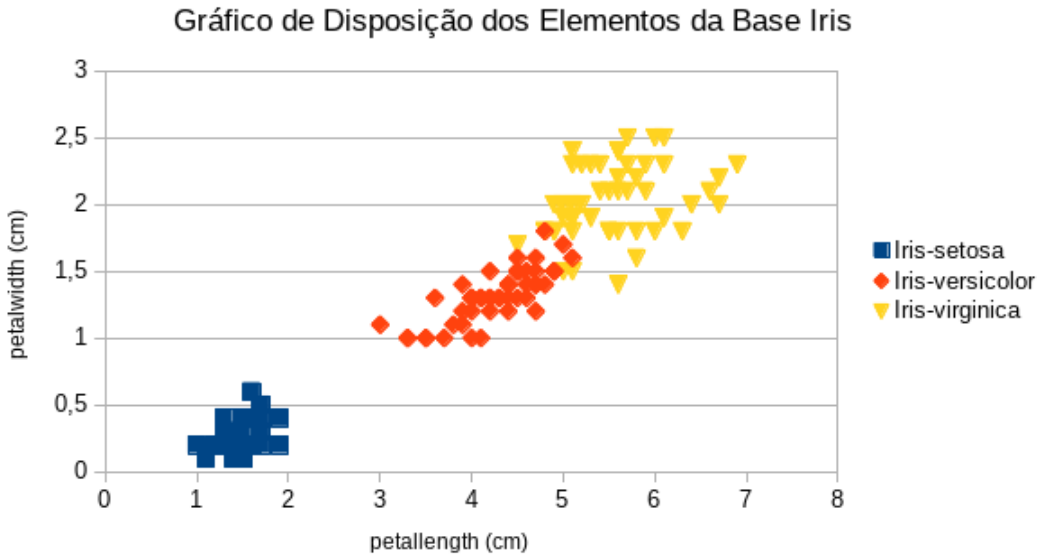


Figura 20 – Gráfico da disposição de elementos da Base Iris entre os eixos **petallength** e **petalwidth**

Todos os algoritmos: Naive Bayes, CART e KNN, encontraram os mesmos rótulos em todos os *clusters* da Iris, e ao analisar o gráfico, o *cluster* 1 (Iris-setosa), é possível perceber uma relação bem definida entre os demais grupos, pois todos os elementos que possuem largura da pétala (*petalwidth*) variando de 0.1 até 0.9 participam do grupo Iris-setosa, portanto, esse rótulo comprovadamente representa este grupo. Analisando os outros grupos através do gráfico da figura 20 percebe-se que a amostra de dados dos grupos Iris-versicolor e Iris-virginica contêm elementos que se misturam, não deixando um ponto de corte tão preciso quanto o da Iris-setosa, e isso acaba por também ocasionar ao rótulo alguns erros identificados nos *clusters* 1 e 2.

4.4 Base de Dados 3 - Glass - Identificação de Tipos de Vidros

Essa base ficou conhecida pelo Dr. Vina Spiehler, Ph.D. da DABFT Diagnostic Products Corporation, que conduziu pesquisas e testes de comparação em seu sistema baseado em regras, determinando se o tipo de vidro era temperado ou não. Institutos de investigação criminológica motivaram os estudos de classificação de tipos de vidro, porque em uma cena de crime, uma classificação de tipos de vidro corretamente identificada pode ser utilizada como prova, ajudando diretamente na investigação (EVETT; SPIEHLER, 1988).

Possui um total de 214 instancias, caracterizados por 9 atributos: índice de refração (RI), e os demais atributos correspondentes a porcentagem do óxido no Sódio (Na), Magnésio (Mg), Alumínio (Al), Silício (Si), Potássio (K), Cálcio (Ca), Bário (Ba) e Ferro (Fe).

Os tipos de vidro (atributo classe) foram divididos em 7 grupos distintos:

- janelas de construção - vidro temperado: 70 registros
- janelas de construção - vidro não-temperado: 76 registros
- janelas de veículos - vidro temperado: 17 registros
- janelas de veículos - vidro não-temperado: 0 registro
- recipientes: 13 registros
- louças de mesa: 9 registros
- lâmpadas: 29 registros

Essa é uma base de dados diferente das demais por possuir mais grupos e, especificamente, o grupo de janelas de veículos - vidro não-temperado, que não possuem amostras de dados para exemplificar. Nos resultados dos rótulos exibidos nas tabelas adiante a coluna **Cluster**, que exibe uma sequência numérica dos grupos, não contemplará o grupo janelas de veículos - vidro não-temperado, pois não há exemplos de dados para definir o grupo, ficando somente seis grupos determinados na seguinte sequência : (1) janelas de construção - vidro temperado, (2) janelas de construção - vidro não-temperado, (3) janelas de veículos - vidro temperado, (4) recipientes , (5) louças de mesa e (6) lâmpadas.

Nos teste desenvolvidos nesta pesquisa os valores de referência foram $R = 4$ para o número de faixas, o método de discretização e também a variância V sofrerão alterações nas execuções nos próprios algoritmos para melhorar seus resultados. O que mais diferenciou os resultados desta base, foi ter que utilizar valores de V bastantes altos para poder conseguir criar rótulos distintos.

4.4.1 Naive Bayes

Na aplicação desse algoritmo as melhores configurações para rotulação foram através do método EWD de discretização e a variação $V = 77\%$ que influencia diretamente nas escolhas dos atributos.

A tabela 14 apresenta os seis clusters nas ordens sequenciais:(1) janelas de construção - vidro temperado, (2) janelas de construção - vidro não-temperado, (3) janelas de veículos - vidro temperado, (4) recipientes , (5) louças de mesa e (6) lâmpadas, e junto dessa coluna são atributos e suas relevâncias com os erros (Fora da Faixa) e acurácia parcial indicado para cada atributo junto ao cluster.

Tabela 14 – Resultado da aplicação do algoritmo Naive Bayes

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
	Atributos	Faixa			
1	Ba	[0.0 ∼ 0.7875]	91%	0	100%
	Fe	[0.0 ∼ 0.1275]	58%	15	78.6%
2	Mg	] 3.3675 ∼ 4.49]	14%	25	68%
	K	[0.0 ∼ 1.5525]	11%	0	100%
	Ba	[0.0 ∼ 0.7875]	85%	1	98.7%
	Fe	[0.0 ∼ 0.1275]	51%	23	69.8%
3	Na	]12.3925 ∼ 14.055]	17%	3	82.4%
	Ba	[0.0 ∼ 0.7875]	94%	0	100%
	Fe	[0.0 ∼ 0.1275]	70%	3	82.4%
4	Mg	[0.0 ∼ 1.1225]	46%	5	61.6%
	Al	] 1.0925 ∼ 1.895]	15%	4	69.3%
	K	[0.0 ∼ 1.5525]	15%	3	77%
	Ba	[0.0 ∼ 0.7875]	84%	1	92.4%
	Fe	[0.0 ∼ 0.1275]	84%	2	84.7%
5	K	[0.00 ∼ 1.5525]	100%	0	100%
	Ba	[0.0 ∼ 0.7875]	100%	0	100%
	Fe	[0.0 ∼ 0.1275]	100%	0	100%
6	RI	[1.511150 ∼ 1.516845]	6%	11	62.1%
	Na	]14.055 ∼ 15.7175]	0%	6	79.4%
	Mg	[0.0 ∼ 1.1225]	68%	6	79.4%
	Al	] 1.895 ∼ 2.6975]	6%	11	62.1%
	Si	] 72.61 ∼ 74.01]	10%	6	79.4%
	K	[0.0 ∼ 1.5525]	15%	2	93.2%
	Ca	[8.12 ∼ 10.81]	0%	3	89.7%
	Ba	[0.0 ∼ 0.7875]	10%	15	48.3%
	Fe	[0.0 ∼ 0.1275]	58%	0	100%

Analisando a tabela 14 percebe através da coluna Relevância, que entre os clusters 1 ao 3 o atributo **Ba** é mais relevante com a mesma faixa de valor, dessa forma isso configura uma ambiguidade de rótulos entre três grupos, *i.e.*, o mesmo rótulo está representando três grupos ao mesmo tempo. A solução de evitar a ambiguidade entre rótulos é agregar mais atributos de menor relevância considerando valores para variância V até que os rótulos se tornem únicos por grupo.

Sabe-se que ao incluir atributos de menor relevância nos rótulos para deixá-los únicos, acaba por diminuir a acurácia média do grupo, porém essa diminuição de confiança do rótulo é necessára para manter a representatividade do rótulo. Essa situação ocorreu com o *cluster* 6, que possuiu umas das menores acurácias médias, devido ao alto valor da variância, $V = 77\%$, resultando também em um rótulo com todos os atributos.

Considerando a tabela 14 o *cluster* 5 foi o único a apresentar um rótulo com três atributos com relevância de 100%, e mesmo com $V = 77\%$ não foi adicionado nenhum outro atributo menos relevante no rótulo, e obteve o valor de 100% na acurácia média do grupo. As menores acurácias médias, por grupo, foi do *cluster* 4 (recipientes) e *cluster* 6 (lâmpadas), com 77% e 77.06% respectivamente.

Em análise a um fato que ocorreu em específico no *cluster* 6, foi que o maior valor de relevância do *cluster* é do atributo **Mg** de 68%, visto que a variância é igual a $V = 77\%$,

e tem o efeito de adicionar todos os atributos que possuem valores abaixo dele no rótulo, dessa forma o atributo **Na** com relevância de 0% também é adicionado ao rótulo. Uma vez que o atributo faz parte do rótulo é calculado qual faixa de valor, do atributo, possui mais elementos. Então o motivo de um atributo que tem 0% de relevância, entre atributos, e ter 86.3% de acurácia é justificado pelo fato de contabilizar os registros que possuem valores de **Na** no intervalo $]14.055 \sim 15.7175]$.

De acordo com a aplicação do Naive Bayes na base de dados **Glass** os rótulos são os seguintes:

- $r_{c_1} = \{(Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_2} = \{(Mg,]3.3675 \sim 4.49]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]),$ $(Fe, [0.0 \sim 0.1275])\}$
- $r_{c_3} = \{((Na,]12.3925 \sim 14.055]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_4} = \{(Mg, [0.0 \sim 1.1225]), (Al,]1.0925 \sim 1.895]), (K, [0.0 \sim 1.5525]),$ $(Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_5} = \{(K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_6} = \{(RI, [1.511150 \sim 1.516845]), (Na,]14.055 \sim 15.7175]), (Mg, [0.0 \sim 1.1225]),$ $(Al,]1.895 \sim 2.6975]), (Si,]72.61 \sim 74.01]),$ $(K, [0.0 \sim 1.5525]), (Fe, [0.0 \sim 0.1275])\}$

4.4.2 Classification and Regression Trees - CART

Através de testes foi descartado a utilização do método de discretização EFD, e adotado o EWD, em virtude da diminuirção de erros por atributo em comparação ao outro método, e a divisão de faixas foi mantido $R = 4$. Quanto a variável V sofreu alteração em relação ao algoritmo anterior, mas mesmo assim manteve-se um valor alto com $V = 71\%$, em razão do número de clusters com rótulos iguais. Este aumento da variável V tem influência direta diminuindo a acurácia média do cluster, muito embora seja justificado esse aumento de V devido não poder haver repetição dos rótulos nos *clusters*.

A execução do CART obteve os resultados apresentados na tabela 15.

As acurácias parciais exibidas na tabela 15 foram bem diversificadas bem como o número de atributos por rótulo, embora mais uma vez os elementos **Ba** e **Fe** são os atributos que aparecem com mais frequência mantendo os mesmos intervalos de dados. Somente no *cluster* 6 é que o **Ba** não participa do rótulo e **Fe** possui o maior valor de relevância, mas em todos os outros, o valor de relevância do **Ba** é maior que os outros atributos. Fazendo um cálculo da acurácia média por grupo, os valores variam de 77% o menor valor no *cluster* 4 (*recipientes*), e 90.8% o maior valor no *cluster* 6 (*lâmpadas*).

De acordo com a aplicação do CART na base de dados **Glass** os rótulos são os seguintes:

Tabela 15 – Resultado da aplicação do algoritmo CART

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
	Atributos	Faixa			
1	Ba	[0.0 ∼ 0.7875]	95%	0	100%
	Fe	[0.0 ∼ 0.1275]	51%	15	78.6%
2	Mg	] 3.3675 ∼ 4.49]	17%	25	68%
	Ba	[0.0 ∼ 0.7875]	84%	1	98.7%
	Fe	[0.0 ∼ 0.1275]	43%	23	69.8%
3	K	[0.0 ∼ 1.5525]	23%	0	100%
	Ba	[0.0 ∼ 0.7875]	88%	0	100%
	Fe	[0.0 ∼ 0.1275]	70%	3	82.7%
4	Mg	[0.0 ∼ 1.1225]	30%	5	61.6%
	Al	] 1.43 ∼ 1.83]	15%	4	69.3%
	K	[0.0 ∼ 1.5525]	15%	3	77%
	Ba	[0.0 ∼ 0.7875]	76%	1	92.4%
	Fe	[0.0 ∼ 0.1275]	69%	2	84.7%
5	Mg	[0.0 ∼ 1.1225]	33%	5	44.5%
	K	[0.00 ∼ 1.5525]	100%	0	100%
	Ba	[0.0 ∼ 0.7875]	100%	0	100%
	Fe	[0.0 ∼ 0.1275]	100%	0	100%
6	Mg	[0.0 ∼ 1.1225]	65%	6	79.4%
	K	[0.0 ∼ 1.5525]	48%	2	93.2%
	Fe	[0.0 ∼ 0.1275]	79%	0	100%

- $r_{c_1} = \{(Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_2} = \{(Mg,]3.3675 \sim 4.49]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_3} = \{((K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_4} = \{(Mg, [0.0 \sim 1.1225]), (Al,]1.0925 \sim 1.895]), (K, [0.0 \sim 1.5525]),$ $(Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_5} = \{(Mg, [0.0 \sim 1.1225]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]),$ $(Fe, [0.0 \sim 0.1275])\}$
- $r_{c_6} = \{(Mg, [0.0 \sim 1.1225]), (K, [0.0 \sim 1.5525]), (Fe, [0.0 \sim 0.1275])\}$

4.4.3 K-Nearest Neighbor - KNN

Na tabela 16 é apresentado o resultado da execução do algoritmo KNN na base de dados Glass aplicando o método de discretização EWD, com valor da variância $V = 81\%$ e alterando para $R = 3$. Na mesma situação do Naive Bayes, o *cluster* 6 recebeu um rótulo com todos os atributos, pelo alto valor de V, embora necessário para retirar as ambiguidades dos rótulos a consequência é aumentar o números de atributos nos rótulos, implicando também na diminuição da acurácia média por grupo.

Tabela 16 – Resultado da aplicação do algoritmo KNN

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
	Atributos	Faixa			
1	Al	[0.29 ∼ 1.36]	15%	10	85.8%
	K	[0.0 ∼ 2.07]	24%	0	100%
	Ba	[0.0 ∼ 1.05]	95%	0	100%
	Fe	[0.0 ∼ 0.17]	64%	8	88.6%
2	Mg	] 2.993333 ∼ 4.49]	18%	19	75%
	K	[0.0 ∼ 2.07]	18%	0	100%
	Ba	[0.0 ∼ 1.05]	92%	1	98.7%
	Fe	[0.0 ∼ 0.17]	51%	16	79%
3	Ba	[0.0 ∼ 1.05]	94%	0	100%
	Fe	[0.0 ∼ 0.17]	70%	2	88.3%
4	Mg	[0.0 ∼ 1.496667]	53%	5	61.6%
	Al	] 1.36 ∼ 2.43]	15%	3	77%
	K	[0.0 ∼ 2.07]	15%	2	84.7%
	Ba	[0.0 ∼ 1.05]	84%	1	92.4%
	Fe	[0.0 ∼ 0.17]	84%	2	84.7%
5	K	[0.00 ∼ 2.07]	100%	0	100%
	Ba	[0.0 ∼ 1.05]	100%	0	100%
	Fe	[0.0 ∼ 0.17]	100%	0	100%
6	RI	[1.511150 ∼ 1.516845]	0%	4	86.3%
	Na	]12.946667 ∼ 15.163333]	3%	2	93.2%
	Mg	[0.0 ∼ 1.1225]	79%	6	79.4%
	Al	[0.0 ∼ 1.496667]	6%	10	65.6%
	Si	] 71.676667 ∼ 73.543333]	6%	6	79.4%
	K	[0.0 ∼ 2.07]	55%	1	96.6%
	Ca	[5.43 ∼ 9.016667]	3%	7	75.9%
	Ba	[0.0 ∼ 1.05]	20%	14	51.8%
	Fe	[0.0 ∼ 0.17]	75%	0	100%

Os resultados apresentados na tabela 16 tiveram valores de intervalos diferentes dos demais algoritmos por suas faixas serem direrentes, pois foi utilizado um R=3 ao invés de R=4. Essa mudança de valor do número de faixas não altera os atributos rótulos, uma vez que a escolha do atributo rótulo é pelo valor de relevância dele junto aos demais.

Os rótulos sugeridos na tabela 16 apresentam em todos os *clusters* os elementos **BA** e **Fe**, mantendo a mesma faixa de dados. No *cluster* 6 é onde o **Ba** possui menor valor de relevância, em relação aos demais rótulos, por ser atributo adicionado para evitar ambiguidade de rótulo, embora, nos outros clusters seu valor de relevância são os mais altos, empatando nos *cluster* 4 e 5, com 84% e 100% respectivamente.

O *cluster* 6 teve o mesmo efeito do algoritmo Naive Bayes, mediante uma variância com valor mais alto ao maior valor de relevância. Nesse caso todos os atributos são incluído no rótulo, até os que tem valores de relevância 0%. É o que aconteceu em relação do atributo **RI** com 0% de relevância e acurácia de 86.3%.

De acordo com a aplicação do KNN na base de dados **Glass** os rótulos são os seguintes:

- $r_{c_1} = \{(Al, [0.29 \sim 1.36]), (K, [0.0 \sim 2.0700]), (Ba, [0.0 \sim 1.05]),$

$(Fe, [0.0 \sim 0.17])\}$

- $r_{c_2} = \{(Mg, [2.993333 \sim 4.490]), (K, [0.0 \sim 2.0700]), (Ba, [0.0 \sim 1.05]),$ $(Fe, [0.0 \sim 0.17])\}$
- $r_{c_3} = \{(Ba, [0.0 \sim 1.0500]), (Fe, [0.0 \sim 0.17])\}$
- $r_{c_4} = \{(Mg, [2.993333 \sim 4.490]), (Al, [0.29 \sim 1.36]), (K, [0.0 \sim 2.0700]),$ $(Ba, [0.0 \sim 1.0500]), (Fe, [0.0 \sim 0.17])\}$
- $r_{c_5} = \{(K, [0.0 \sim 2.0700]), (Ba, [0.0 \sim 1.0500]), (Fe, [0.0 \sim 0.17000])\}$
- $r_{c_6} = \{(RI, [1.511150 \sim 1.516845]), (Na,]12.946667 \sim 15.163333]), (Mg, [0.0 \sim 1.1225]),$ $(Al, [0.0 \sim 1.496667]), (Si,]71.676667 \sim 73.543333]),$ $(K, [0.0 \sim 2.07]), (Fe, [0.0 \sim 0.17])\}$

4.4.4 Comparativo entre Algoritmos na Base de Dados Glass

Os valores que compões a base Glass, por se tratar de representações de elementos químicos através de atributos, possui em cada um deles, uma quantidade em porcentagem de óxido, e como é um valor bastante complexo acaba por se tornar uma base com valores distintos em cada atributo.

A tabela 17 apresenta todos os nove registros do *cluster* 5 - louças de mesa - possuindo os valores respectivos de cada elemento em suas colunas.

Tabela 17 – *Cluster* 5 - Louças de mesa - Base Glass

RI	**Na**	**Mg**	**Al**	**Si**	**K**	**Ca**	**Ba**	**Fe**
1,51299	14,4	1,74	1,54	74,55	0	7,59	0	0
1,51115	17,38	0	0,34	75,41	0	6,65	0	0
1,51829	14,46	2,24	1,62	72,38	0	9,26	0	0
1,51888	14,99	0,78	1,74	72,5	0	9,95	0	0
1,51937	13,79	2,41	1,19	72,76	0	9,77	0	0
1,51969	14,56	0	0,56	73,48	0	11,22	0	0
1,51905	14	2,39	1,56	72,37	0	9,57	0	0
1,51916	14,15	0	2,09	72,74	0	10,88	0	0
1,51852	14,09	2,19	1,66	72,67	0	9,32	0	0

No caso do cluster 5 apresentado na tabela 17, percebe-se nessas amostras uma grande repetição de valores 0 (zero) em determinados atributos: **K**, **Ba** e **Fe**. Posto isso, o cluster 5 serve como exemplo prático para aferir a acurácia do rótulo, visto que com uma simples análise deste cluster é possível perceber um rótulo que o representa.

Dos algoritmos que sugeriram os mesmos rótulos foram Naive Bayes e KNN, encontrando 100% de acurácia média no cluster 5, $r_{c_5} = \{(K, [0.0 \sim 2.0700]), (Ba, [0.0 \sim 1.0500]),$ $(Fe, [0.0 \sim 0.17000])\}$, e o CART teve um atributo a mais inserido em seu rótulo que lhe custou uma acurácia média de 86.12%, $r_{c_5} = \{(Mg, [0.0 \sim 1.1225]), (K, [0.0 \sim 1.5525]),$ $(Ba, [0.0 \sim 0.7875]),$.

As informações da tabela 18 mostram as acurácias médias calculadas a partir das acurácias parciais dos *clusters*, servindo de referência para mensurar o quanto o rótulo está representando os clusters.

Tabela 18 – Acurácia média da Base Glass por *clusters*

Clusters	**Naive Bayes**	**CART**	**KNN**
1	89.3%	89.3%	93.6%
2	84.12%	78.83%	88.17%
3	88.2%	94.23%	94.15%
4	77%	77%	80.08%
5	100%	86.12%	100%
6	77.06%	90.86%	80.91%

Analisando os resultados da tabela 18 percebe-se que a acurácia média por rótulo no *cluster* 2 pelo CART foi de 78.83%, no *cluster* 4 nos algoritmos Naive Bayes e CART foram de 77% e no *cluster* 6 pelo Naive Bayes foi de 77.06%, e os demais, acima de 80%. Estes são índices indicando que cada tipo de vidro é representado de forma satisfatória pelos rótulos dos grupos

4.5 Base de Dados 4 - Wine - Dados de Reconhecimento de vinhos

Essa base (AEBERHARD et al., 1992) possui dados que são resultados de uma análise química de vinhos em uma mesma região da Itália, mas com três tipos de cultivos diferenciados, resultando em três classes distintas (tipos de vinho) e totalizando 178 registros. Através desta análise foram determinadas treze características determinantes para classificação do tipo de vinho, que estão distribuídas em 178 amostras. Esses treze atributos são: (1) álcool , (2) ácido málico - AM, (3) cinzas, (4) alcalinidade das cinzas - AC, (5) magnésio, (6) total de fenois - TF, (7) flavonoides, (8) fenóis não-flavonoides - FnF, (9) proantocianidinas, (10) intensidade da cor - IC, (11) matiz, (12) OD280/OD315 de vinhos diluídos - OD e (13) prolina.

A divisão em número de instâncias por classe são:

- classe 1: 59 elementos;
- classe 2: 71 elementos;
- classe 3: 48 elementos;

Na descrição da base Wine[4] não é dito quais seriam os nomes das classes (tipos de vinho) designadas como classe 1, 2 ou 3, portanto, os rótulos encontrados pelos algoritmos são apresentados em tabelas e na coluna **Cluster** as informações são inseridas por sequência numérica: *cluster* 1, *cluster* 2, *cluster* 3 com 59, 71 e 48 registros respectivamente.

[4] https://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.names

O método de discretização variou dentre os algoritmos e prevaleceu os que obtiveram melhores resultados (menos erros) nos rótulos encontrados, mantendo-se o $R = 3$ para todos os atributos e também a variância $V = 0\%$.

4.5.1 Naive Bayes

Os rótulos encontrados após aplicação do Naive Bayes são apresentados na tabela 19 e separados por *cluster* 1, 2 e 3, representando os três tipos de vinho. Foi utilizado o método de discretização EWD para demarcar os limites das faixas de dados dos atributos.

Tabela 19 – Resultado da aplicação do algoritmo Naive Bayes

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
1	AM	] 1.71 ∼ 2.89]	8%	28	52.6%
	magnesium	] 95.00 ∼ 113.00]	8%	24	59.4%
	proline	] 880.00 ∼ 1680.00]	8%	10	83.1%
2	alcohol	[11.03 ∼ 12.64]	11%	15	95%
3	FnF	] 0.44 ∼ 0.66]	10%	22	54.2%

O *cluster* 1 foi o único a apresentar três atributos no rótulo, com relevância de 8% cada; dois desses atributos **AM** e **magnesium** possuem acurácias parciais baixas, 52.6% e 59.4% respectivamente e outro **prolina** com 83.1%, totalizando uma acurácia média aproximada no grupo de 65%. Em números de registros o rótulos do cluster 1 consegue representar 19 registros dos 59 do grupo. No *cluster* 2 foi o que apresentou melhor acurácia com 95% representando 60 registros dos 71 do grupo, e o *cluster* 3 com a pior acurácia, representanto 26 amostras, um pouco mais da metade dos 48 registros do grupo.

De acordo com a aplicação do Naive Bayes na base de dados **Wine** os rótulos são:

- $r_{c_1} = \{(AM,]1.71 \sim 2.89]), (magnesium,]95.00 \sim 113.00]),$
  $(proline,]880.00 \sim 1680.00])\}$
- $r_{c_2} = \{(alcohol, [11.03 \sim 12.64])\}$
- $r_{c_3} = \{(FnF,]0.44 \sim 0.66])\}$

4.5.2 Classification and Regression Trees - CART

A tabela 20 apresenta os rótulos encontrados pelo CART na base Wine utilizando o número de faixas $R = 3$, e dentre os clusters percebe-se que o *cluster* 2, foi o que obteve a melhor acurácia de 76.1%. Foram realizados vários testes alterando o número de faixas, mas nenhuma melhora significativa nos erros foram encontrados nos rótulos dos clusters.

Tabela 20 – Resultado da aplicação do algoritmo CART

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
1	AC	[10.60 ∼ 17.066667]	11%	27	54.3%
2	magnesium	[70.00 ∼ 100.666667]	9%	17	76.1%
3	FnF	] 0.483333 ∼ 0.660]	14%	27	43.8%

Nos rótulos sugeridos pelo CART, os valores de porcentagens de correlacionamento entre as variáveis, coluna **Relevância**, foram valores baixos não passando de 14% nos grupos, indicando um baixo grau de correlação entre esse atributo e os demais. Embora a relevância do atributo implique na escolha dele como rótulo, a acurácia depende da divisão de valores da faixa do atributo, pois será contabilizado o número de registros em cada faixa.

Logo os rótulos da aplicação do CART na base de dados **Wine** são:

- $r_{c_1} = \{(AC, [10.60 \sim 17.066667])\}$
- $r_{c_2} = \{(magnesium, [70.00 \sim 100.666667])\}$
- $r_{c_3} = \{(FnF, [0.483333 \sim 0.660])\}$

4.5.3 K-Nearest Neighbor - KNN

No resultado do KNN, na tabela 21, percebe-se o atributo FnF sendo rótulo do *cluster* 1 e 3, mediante faixas distintas. O mesmo atributo se repete no *cluster* 3 dos três algoritmos, ainda que no Naive Bayes fora utilizado outro método de discretização mantendo o atributo FnF mas alterando os intervalos da faixa.

Tabela 21 – Resultado da aplicação do algoritmo KNN

	Rótulos				
Cluster	Atributos	Faixa	Relevância(%)	Fora da Faixa	Acurácia Parcial(%)
1	FnF	[0.13 ∼ 0.306667]	13%	20	66.2%
2	alcool	[11.03 ∼ 12.296667]	11%	33	53.6%
	AC	] 17.066667 ∼ 23.533333]	11%	21	70.5%
3	FnF	] 0.483333 ∼ 0.66]	20%	27	43.8%

Os rótulos encontrados após a execução do KNN na base Wine são:

- $r_{c_1} = \{(FnF, [0.13 \sim 0.306667])\}$
- $r_{c_2} = \{(alcool, [11.03 \sim 12.296667]), (AC, [17.066667 \sim 23.533333])\}$
- $r_{c_3} = \{(FnF,]0.483333 \sim 0.66])\}$

4.5.4 Comparativo entre Algoritmos na Base de Dados Wine

No geral as acurácias foram bastantes baixas entre os algoritmos e a única semelhança entre os rótulos aconteceu no *cluster* 3, o qual foi idêntico entre CART, KNN

e Naive Bayes mas nesse último, houve diferença do intervalo de dados ocasionando altereção na acurácia. Na execução do Naive Bayes foi utilizado discretização EFD com variação na faixa 1 de $[0.13 \sim 0.29]$, na faixa 2 $]0.29 \sim 0.44]$ e na faixa 3 $]0.44 \sim 0.66]$, e os outros dois algoritmos utilizaram EWD variando na faixa 1 de $[0.13 \sim 0.3067]$, na faixa 2 $]0.3067 \sim 0.483333]$ e na faixa 3 $]0.483333 \sim 0.66]$. A faixa 3 do método de discretização EFD ficou maior e por conseguinte mais registros foram representados aumentando a acurácia.

Dos três algoritmos o Naive Bayes obteve melhores acurácias nos rótulos dos clusters 2 e 3, e foi o único que no cluster 1 apresentou em um mesmo rótulo, três atributos para representar o grupo. Uma forma de melhor explicar quais registros o rótulo esta representando foi adicionar a tabela 22 onde exibe todos os 59 registros do grupo 1, mas com somente três atributos. Esta amostra servirá de exemplo para visualizar quais os registros são representados pelo rótulo do cluster 1 do Naive Bayes.

A tabela possui quatro colunas, sendo a primeira um índice da linha e as outras três os atributos correspondentes ao rótulo. No cabeçalho da tabela 22 é exibido os atributos com seus respectivos símbolos, cada símbolo tem a função visual de marcar a linha que faz parte do rótulo r_{c_1}, ou seja, destaca a qual atributo e faixa é representada a amostra:

- $(AM,]1.71 \sim 2.89])$ - □
- $(magnesium,]95.00 \sim 113.00])$ - ⊠
- $(proline,]880.00 \sim 1680.00])$ - ○

Então quando a linha for representada por um atributo e seu intervalo, este será marcado de acordo com o símbolo que ele representa, e caso esteja com os três símbolos então a linha é representada pelo rótulo r_{c_1}, e dessa forma é possível visualizar na própria base quais amostras o rótulo representa e não representa.

Tabela 22 – Amostra da Base de Dados Wine com somente três atributos pertencentes ao *cluster* 1 do algoritmo Naive Bayes

n.	**AM** □	**magnesium** ⊠	**proline** ○	
1	3,98	103	680	⊠○
2	1,95	113	1480	□⊠○
3	4,04	111	1080	⊠○
4	1,77	107	885	□⊠○
5	1,89	101	1095	□⊠○
6	3,99	128	760	
7	1,43	108	1285	⊠○
8	3,84	90	1035	○
9	1,68	101	985	⊠○
10	1,67	118	1060	○
11	1,77	93	1195	□○
12	1,7	118	970	○
13	1,71	117	795	
14	3,1	116	845	
15	1,73	116	1120	□○
16	1,35	98	1045	⊠○
17	1,75	111	1190	□⊠○
18	1,72	94	1285	□○
19	1,87	96	1290	□⊠○
20	3,8	102	770	⊠
21	2,36	101	1185	□⊠○
22	1,68	96	1035	⊠○
23	2,59	118	735	□
24	1,81	100	920	□⊠○
25	1,71	127	1065	○
26	3,59	102	1065	⊠○
27	1,78	100	1050	□□⊠○
28	1,76	112	1450	□⊠○
29	2,15	121	1295	□○
30	2,02	103	1060	□⊠○

n.	**AM** □	**magnesium** ⊠	**proline** ○	
31	1,92	120	1280	□○
32	1,97	102	1270	□⊠○
33	1,64	97	1045	⊠○
34	1,6	95	1015	○
35	1,64	110	880	⊠
36	1,63	126	780	
37	1,66	106	1515	⊠○
38	1,8	110	1095	□⊠○
39	1,86	101	1035	□⊠○
40	1,65	98	1105	⊠○
41	1,81	96	845	⊠□
42	1,73	92	1150	□○
43	1,73	108	1260	□⊠○
44	2,05	124	830	□
45	1,9	115	1375	□○
46	1,57	115	1130	○
47	1,73	89	1320	□○
48	1,48	95	1280	○
49	1,5	98	1020	⊠○
50	1,5	101	1285	⊠○
51	1,87	102	1547	□⊠○
52	1,83	104	990	□⊠○
53	1,59	108	1680	⊠○
54	1,65	94	1265	○
55	1,53	132	1235	○
56	1,9	107	915	□⊠○
57	2,16	105	1510	□⊠○
58	1,73	91	1150	□○
59	1,81	112	1310	□⊠○

Os rótulos dos clusters 2 e 3 do Naive Bayes são compostos por um atributo cada, e através da figura 21 é possível conhecer o espaço amostral da base Wine em função de dois eixos:FnF e alcool.

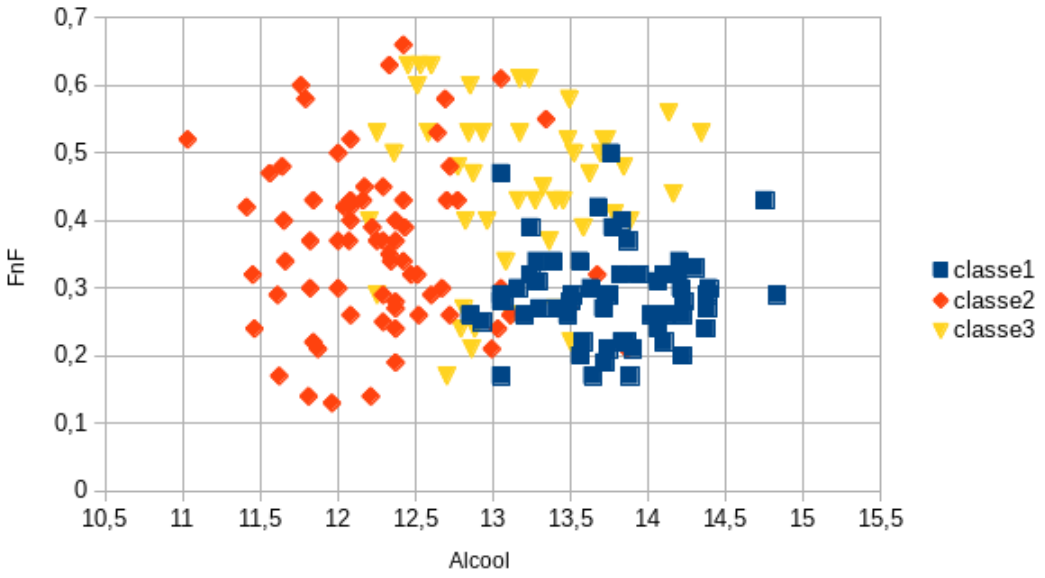


Figura 21 – Gráfico da disposição de elementos da Base Wine entre os eixos **FnF** e **Alcool**

Em uma análise do gráfico da figura 21, mesmo observando em uma pespectiva de duas variáveis, é possível visualizar o comportamento da base Wine quanto a disposição de seus dados, e através dos rótulos sugeridos de pelo Naive Bayes pode-se constatar suas representações nos grupos, bem como também visualizar seus erros. e.g., quando um especialista verificar um determinado dado que possui no atributo alcool o valor 12.1, e este conhece os rótulos sugeridos pelo Naive Bayes, logo o especialista atribui que esse dado pertence ao grupo de vinho tipo 2. Esse exemplo pode ser constatado com o gráfico da figura 21, pois ao confirmar o valor 12.1 no eixo X (alcool) percebe-se que este está junto a distribuição do grupo 2 conforme figura.

A mesma figura também pode ser utilizada para explicar os rótulos sugeridos pelo KNN, que teve em comparação com CART, melhor acurácia na rotulação do *cluster* 1, e igual ao *cluster* 3. Com rótulo do KNN $r_{c_1} = (FnF, [0.13 \sim 0.306667])\}$, e através da figura 21 percebe-se que a faixa coberta pelo rótulo aponta para os dados do grupo 1 (classe1 do gráfico). Da mesma forma acontece com o rótulo $r_{c_3} = \{(FnF,]0.483333 \sim 0.66])\}$ no qual valores do atributo **FnF** acima de 0.48 (vê gráfico figura 21) até 0.66 direcionam para o grupo 3 (classe3 do gráfico), de modo a constatar que os rótulos conseguem representar os grupos dos tipos de vinho.

Dos resultados encontrados pelo CART o rótulo do *cluster* 1, dentre os três algoritmos foi o melhor, com acurácia de 66.2%, já em relação a acurácia no *cluster* 2 foi pior comparando aos outros dois algoritmos, no *cluster* 3 já foi dito sobre a igualdade

deste resultado com CART e teve a acurácia mais baixa do que o Naive Bayes.

4.6 Discussões

A figura 22 mostra um gráfico geral da acurácia média de todos os rótulos encontrados na aplicação dos algoritmos nas bases de dados. No eixo **Y** corresponde a porcentagem de acurácia média alcançada de um algoritmo, e o eixo **X** está dividido nos *clusters* e qual base de dados ele pertence.

A figura 22 mostra um gráfico geral da acurácia média de todos os rótulos encontrados na aplicação dos algoritmos nas bases de dados. O eixo Y corresponde à porcentagem de acurácia média alcançada de um algoritmo, e o eixo X está dividido nos clusters e a qual base de dados ele pertence.

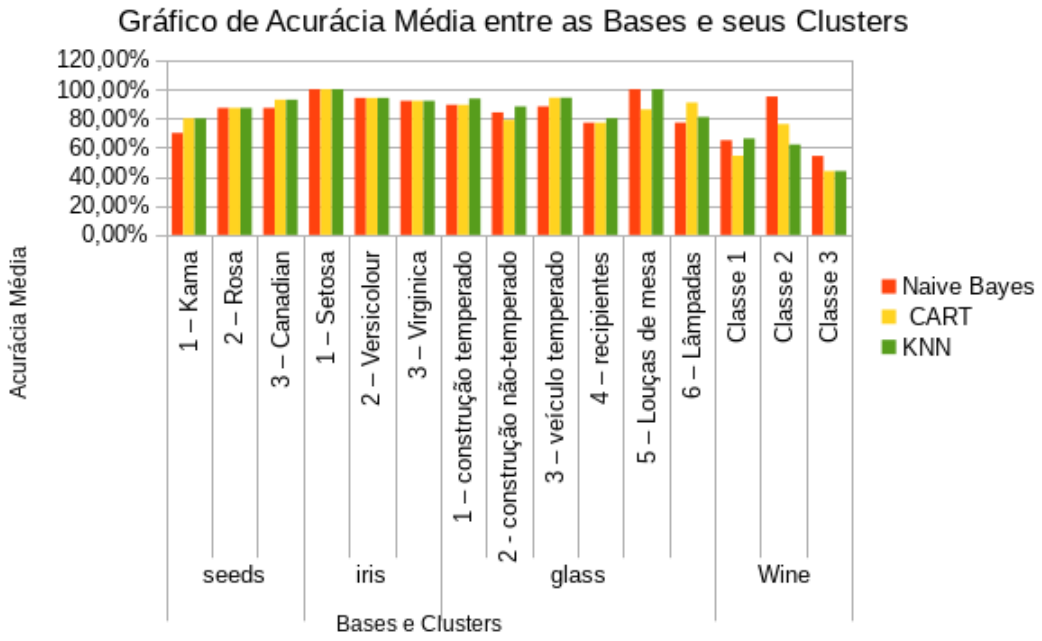


Figura 22 – Disposição dos elementos da base wine na pespectiva dos atributos:Alcool e FnF

Uma breve análise do gráfico da figura 22 referente as acurácias dos rótulos, fica claro que a qualidade deles foram satisfatórias, pois atingiram uma acurácia média com mínimo de 80%. A acurácia desses rótulos servem para mostrar o quanto esses rótulos representam o grupo através das amostras testadas, e os três algoritmos tiveram acurácias bem parecidas, e em específico o Naive Bayes e KNN. Os clusters com menores acurácias

5 Conclusões, Trabalhos Futuros

Neste capítulo serão apresentadas as considerações finais da dissertação, bem como a apresentação dos benefícios da pesquisa realizada, que teve como base a rotulação de algoritmos com vistas a analisar o seu comportamento considerando algumas bases de dados consolidadas. Também serão apresentadas sugestões para continuidade deste trabalho já efetuado.

5.1 Conclusão

O presente trabalho dedicou-se a estudar a aplicabilidade de algoritmos supervisionados com bases de dados distintas, as bases utilizadas foram: Seeds, Iris, Glass e Wine, com a finalidade de definir a tupla atributo/valor de maior importância nos *clusters*, determinando a sua rotulação. A formação do referido problema nasceu de outra pesquisa já concluída (LOPES et al., 2016), entretanto, teve como diferencial a realização de rotulação de grupos de dados a partir de algoritmos supervisionados não testados, Naive Bayes, Classification and Regression Trees - CART e K-Nearest Neighbor - KNN, assim, ao final foi possível demonstrar o comparativo de resultados.

As análises foram feitas a partir das bases de dados mencionadas acima, e o comportamento ao aplicar cada algoritmo nas bases e encontrar os rótulos foram satisfatórios. A avalição da qualidade destes rótulos foi feita da seguinte forma: inicialmente, escolheu-se as bases de dados já classificadas; feito isso foi possível saber quantas amostras pertencem a determinado grupo; em seguida, foi realizado um comparativo dos resultados dos rótulos através dos erros encontrados, e nos grupos com as amostras originais das bases de dados, desta forma foi possível mensurar a acurácia de cada rótulo.

O funcionamento dos algoritmos foram também marcados pelas bases utilizadas, a base de dados 1 - Seeds, é uma base de dados balanceada, que neste caso, possui um número de exemplos iguais para as três classes (70 elementos cada classe), destacando-se que a mesma obteve rótulos diferentes entre os três algoritmos, sua acurácia foi alta, considerando que a menor acurácia parcial de grupo chegou a 80%, e a maior, aproximadamente 94%. A base de dados 2 - Iris, tiveram rótulos idênticos em dois algoritmos, Naive Bayes e KNN, e no CART a diferença ficou somente no *cluster* 2 (*Iris-versicolor*), que deu maior importância para o largura da pétala (*petalwidth*) ao invés do comprimento da pétala (*petallength*) encontrada nos outros dois algoritmos, porém, a escolha do *petalwidth*, pelo CART, ocasionou um número de erros um pouco maior em comparação com os outros dois algoritmos (Naive Bayes e KNN).

Na base de dados 3 - Glass, os rótulos foram bastantes satisfatórios ao analisar as acurácias dos algoritmos chegando em vários *clusters* a acurácias de 100%, contudo, os rótulos do Naive Bayes e CART em específico no *cluster* 4 (grupo recipientes), mesmo tendo poucos erros a acurácia chegou em 77% em alguns atributos rótulos. Vale ressaltar que essa base não é classificada como uma base balanceada e no grupo, recipientes, conta com somente treze elementos. Já na base de dados 4 - Wine, possui um total de 178 (cento e setenta e oito) registros, de forma balanceada, pois possui um número mínimo de 48 (quarenta e oito) elementos no grupo. Nesta base os rótulos foram bastantes semelhantes, e diferenciados apenas no *cluster* 1 do KNN que obteve um atributo rótulo diferentes dos outros algoritmos, porém o número de erros de todos os rótulos foram iguais afirmando que os três possuem o mesmo poder de representar os clusters com seus rótulos.

Outro ponto que merece destaque é a tomada de decisão através dos rótulos encontrados neste trabalho, isto é, acurácias altas resultam em boa confiabilidade dos rótulos, portanto quando um especialista da área verificar um rótulo de um grupo, este será capaz de perceber o que é importante para o grupo, podendo fazer uma tomada de decisão mais eficiente a partir destes resultados.

Assim, diante do problema de rotulação, que visa encontrar informações relevantes em grupos de dados ao ponto de identificar esses grupos, e considerando que Lopes et al. (2016) utilizou algoritmo supervisionado com paradigma conexionista (Redes Neurais) neste problema, esta pesquisa apresentou aplicação de novos algoritmos com paradigmas diferentes, ainda não testados, mostrando a viabilidade do método de rotulação de dados com algoritmos de aprendizado supervisionados, e comparando os resultados através das acurácias dos rótulos encontradas em cada *cluster*.

5.2 Trabalhos Futuros

Espera-se realizar estudos mais profundos nos métodos de discretização, pois estes, tem influência comprovada na geração dos rótulos nos grupos de dados, e também no número de faixas, por está diretamente ligado aos métodos de discretização. A melhor discretização e o melhor número de faixas a ser utilizado está relacionado aos valores das bases de dados utilizada, e portanto será necessário um estudo aprofundado para comparar os resultados com os deste trabalho.

Referências

AEBERHARD, S.; COOMANS, D.; VEL, O. D. *Comparison of classifiers in high dimensional settings.* North Queensland, Austrália, 1992. 92–02 p. Citado na página 51.

ARAÚJO, F. N. C. *Rotulação Automática de Clusters Baseados em Análise de Filogenias.* Teresina - PI: [s.n.], 2018. 48 p. Citado na página 21.

BARBER, D. *Bayesian Reasoning and Machine Learning.* [s.n.], 2011. ISSN 9780521518147. ISBN 9780511804779. Disponível em: <http://ebooks.cambridge.org/ref/id/CBO9780511804779>. Citado 2 vezes nas páginas 5 e 13.

BREIMAN, L. et al. *Classification and Regression Trees.* Taylor & Francis, 1984. (The Wadsworth and Brooks-Cole statistics-probability series). ISBN 9780412048418. Disponível em: <https://books.google.com.br/books?id=JwQx-WOmSyQC>. Citado na página 7.

CASARI, A.; ZHENG, A. *Feature Engineering for Machine Learning. Principles and Techniques for Data Scientists.* First edition. Sebastopol, CA: O'Reilly Media, Inc., 2018. ISBN 9781491953235. Citado na página 34.

CATLETT, J. *On changing continuous attributes into ordered discrete attributes.* Springer, Berlin, Heidelberg: Springer Verlag, 1991. 164–178 p. Citado 2 vezes nas páginas 14 e 29.

CHARYTANOWICZ, M. et al. Complete gradient clustering algorithm for features analysis of X-ray images. *Advances in Intelligent and Soft Computing*, v. 69, p. 15–24, 2010. ISSN 18675662. Citado na página 35.

CHEN, C.-L.; TSENG, F. S. C.; LIANG, T. An integration of fuzzy association rules and WordNet for document clustering. *Knowledge and Information Systems*, v. 28, n. 3, p. 687–708, sep 2011. ISSN 0219-1377. Disponível em: <http://link.springer.com/10.1007/s10115-010-0364-2>. Citado 2 vezes nas páginas 18 e 19.

COSTA, B. S. J. et al. Unsupervised classification of data streams based on typicality and eccentricity data analytics. *2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016*, p. 58–63, 2016. Citado na página 19.

DOUGHERTY, J.; KOHAVI, R.; SAHAMI, M. Supervised and Unsupervised Discretization of Continuous Features. *Machine Learning Proceedings 1995*, Stanford, v. 0, p. 194–202, 1995. ISSN 0717-6163. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/B9781558603776500323>. Citado na página 14.

EVETT, I. W.; SPIEHLER, E. J. Knowledge based systems. In: DUFFIN, P. H. (Ed.). New York, NY, USA: Halsted Press, 1988. cap. Rule Induction in Forensic Science, p. 152–160. ISBN 0-470-21260-8. Disponível em: <http://dl.acm.org/citation.cfm?id=67040.67055>. Citado na página 44.

FILHO, V. P. R.; MACHADO, V. P.; LIRA, R. d. A. Rotulação de Grupos Utilizando Conjuntos Fuzzy. In: UNIVERSIDADE FEDERAL DO PIAUÍ. *XII Simpósio Brasileiro de Automação Inteligente (SBAI)*. Natal, RN, 2015. Disponível em: <http://pubs.acs.org/doi/abs/10.1021/ja103937v>. Citado 2 vezes nas páginas 21 e 41.

FISHER, R. A. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, v. 7, n. 2, p. 179–188, 1936. ISSN 20501420. Disponível em: <http://doi.wiley.com/10.1111/j.1469-1809.1936.tb02137.x>. Citado na página 41.

GAN, H. et al. Using clustering analysis to improve semi-supervised classification. *Neurocomputing*, v. 101, p. 290–298, feb 2013. ISSN 09252312. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/S0925231212006601>. Citado na página 19.

HWANG, G. J.; LI, F. A Dynamic Method for Discretization of Continuous Attributes. *Lecture Notes in Computer Science - Intelligent Data Engineering and Automated Learning - IDEAL 2002: Third International Conference*, v. 2412/2002, p. 506, 2002. ISSN 16113349. Disponível em: <http://www.springerlink.com/content/4n05b2n6x0cx4tlk>. Citado 2 vezes nas páginas 14 e 29.

IMPERES, F. d. C. *Rotulação de Grupos em Algoritmos de Agrupamento Baseados em Distância Utilizando Grau de Pertinência*. 2018. 60 p. Citado na página 21.

IWAMURA, M.; TSUKADA, M.; KISE, K. Automatic Labeling for Scene Text Database. In: *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 2013. p. 1365–1369. ISBN 978-0-7695-4999-6. Disponível em: <http://ieeexplore.ieee.org/document/6628837/>. Citado na página 19.

JIRASIRILERD, W.; TANGTISANON, P. Automatic Labeling for Thai News Articles Based on Vector Representation of Documents. In: *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)*. IEEE, 2018. p. 1–4. ISBN 978-1-5386-4956-5. Disponível em: <https://ieeexplore.ieee.org/document/8434457/>. Citado 2 vezes nas páginas 18 e 19.

KOTSIANTIS, S.; KANELLOPOULOS, D. Discretization Techniques : A recent survey. *GESTS International Transactions on Computer Science and Engineering*, v. 32, n. 1, p. 47–58, 2006. Citado na página 14.

KUMAR, A.; ANDU, T.; THANAMANI, A. S. Multidimensional Clustering Methods of Data Mining for Industrial Applications. *International Journal of Engineering Science Invention*, v. 2, n. 7, p. 1–8, 2013. Citado na página 1.

LACHI, R. L.; ROCHA, H. V. *Aspectos básicos de clustering: conceitos e técnicas*. Campinas, SP, 2005. 1–26 p. Disponível em: <http://www.ic.unicamp.br/{~}reltech/2005/05-03.p>. Citado na página 11.

LIMA, B. V. A. Dissertação (Programa de Pós-graduação em Ciência da Computação), *Método Semissupervisionado de Rotulação e Classificação Utilizando Agrupamento por Sementes e Classificadores*. Teresina - PI: [s.n.], 2015. 47 p. Citado 2 vezes nas páginas 20 e 21.

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*, v. 5, n. 1, p. 44, dec 2015. ISSN 1869-5450. Disponível em: <http://link.springer.com/10.1007/s13278-015-0285-x>. Citado 2 vezes nas páginas 20 e 21.

LOPES, L. A.; MACHADO, V. P.; RABELO, R. D. A. L. Automatic Labeling of Groupings through Supervised Machine Learning. *Knowledge-Based Systems*, v. 106, p. 231–241, 2016. Citado 13 vezes nas páginas 9, 1, 2, 15, 17, 20, 21, 24, 26, 29, 41, 58 e 59.

LUCCA, G. et al. Uma implementação do algoritmo Naïve Bayes para classificação de texto. In: CENTRO DE CIÊNCIAS COMPUTACIONAIS DA UNIVERSIDADE FEDERAL DO RIO GRANDE. *IX Escola Regional de Banco de Dados – ERBD 2013*. Rio Grande - RS, 2013. p. 1–4. Disponível em: <http://ifc-camboriu.edu.br/erbd2013>. Citado na página 9.

MADUREIRA, D. F. *Analise de sentimento para textos curtos*. Tese (Doutorado) — Fundacao Getulio Vargas, Rio de Janeiro, 2017. Citado na página 9.

MCCALLUM, A.; NIGAM, K. A Comparison of Event Models for Naive Bayes Text Classification. 1997. Citado na página 9.

MITCHELL, T. M. *Machine learning*. [S.l.]: McGraw-Hill Science/Engineering/Math, 1997. 432 p. ISSN 10450823. ISBN 9781577354260. Citado 2 vezes nas páginas 4 e 8.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations Machine Learning*. [S.l.: s.n.], 2012. ISBN 9780262018258. Citado na página 5.

RAIMUNDO, L. R. et al. O Algoritmo de Classificação CART em uma Ferramenta de Data Mining. In: *IV Congresso Sul Brasileiro de Ciências da Computação - SULCOMP*. Pelotas - RS: [s.n.], 2008. Citado na página 7.

RUSSEL, S.; NORVIG, P. *Inteligência Artificial*. 3ª. ed. Rio de Janeiro: [s.n.], 2013. ISBN 9780136042594. Citado 4 vezes nas páginas 4, 5, 6 e 9.

SUN, L.; YOSHIDA, S.; LIANG, Y. A support vector and k-means based hybrid intelligent data clustering algorithm. *IEICE Transactions on Information and Systems*, E94-D, n. 11, p. 2234–2243, 2011. ISSN 17451361. Citado na página 19.

WU, X. et al. *Top 10 algorithms in data mining*. [S.l.: s.n.], 2008. v. 14. 1–37 p. ISSN 02191377. ISBN 1011500701. Citado na página 9.

YEGANOVA, L.; COMEAU, D. C.; WILBUR, W. J. Identifying Abbreviation Definitions Machine Learning with Naturally Labeled Data. In: *2010 Ninth International Conference on Machine Learning and Applications*. IEEE, 2010. p. 499–505. ISBN 978-1-4244-9211-4. Disponível em: <http://ieeexplore.ieee.org/document/5708877/>. Citado 2 vezes nas páginas 18 e 19.

YOHANNES, Y.; WEBB, P. *Classification and Regression Trees, CART: A User Manual for Identifying Indicators of Vulnerability to Famine and Chronic Food Insecurity*. International Food Policy Research Institute, 1999. (Microcomputers in policy research). ISBN 9780896293373. Disponível em: <https://books.google.com.br/books?id=7iuq4ikyNdoC>. Citado na página 7.

Apêndices

APÊNDICE A – Primeiro Apêncice

Quisque facilisis auctor sapien. Pellentesque gravida hendrerit lectus. Mauris rutrum sodales sapien. Fusce hendrerit sem vel lorem. Integer pellentesque massa vel augue. Integer elit tortor, feugiat quis, sagittis et, ornare non, lacus. Vestibulum posuere pellentesque eros. Quisque venenatis ipsum dictum nulla. Aliquam quis quam non metus eleifend interdum. Nam eget sapien ac mauris malesuada adipiscing. Etiam eleifend neque sed quam. Nulla facilisi. Proin a ligula. Sed id dui eu nibh egestas tincidunt. Suspendisse arcu.

APÊNDICE B – Perceba que o texto do título desse segundo apêndice é bem grande

Maecenas dui. Aliquam volutpat auctor lorem. Cras placerat est vitae lectus. Curabitur massa lectus, rutrum euismod, dignissim ut, dapibus a, odio. Ut eros erat, vulputate ut, interdum non, porta eu, erat. Cras fermentum, felis in porta congue, velit leo facilisis odio, vitae consectetuer lorem quam vitae orci. Sed ultrices, pede eu placerat auctor, ante ligula rutrum tellus, vel posuere nibh lacus nec nibh. Maecenas laoreet dolor at enim. Donec molestie dolor nec metus. Vestibulum libero. Sed quis erat. Sed tristique. Duis pede leo, fermentum quis, consectetuer eget, vulputate sit amet, erat.

Donec vitae velit. Suspendisse porta fermentum mauris. Ut vel nunc non mauris pharetra varius. Duis consequat libero quis urna. Maecenas at ante. Vivamus varius, wisi sed egestas tristique, odio wisi luctus nulla, lobortis dictum dolor ligula in lacus. Vivamus aliquam, urna sed interdum porttitor, metus orci interdum odio, sit amet euismod lectus felis et leo. Praesent ac wisi. Nam suscipit vestibulum sem. Praesent eu ipsum vitae pede cursus venenatis. Duis sed odio. Vestibulum eleifend. Nulla ut massa. Proin rutrum mattis sapien. Curabitur dictum gravida ante.

Phasellus placerat vulputate quam. Maecenas at tellus. Pellentesque neque diam, dignissim ac, venenatis vitae, consequat ut, lacus. Nam nibh. Vestibulum fringilla arcu mollis arcu. Sed et turpis. Donec sem tellus, volutpat et, varius eu, commodo sed, lectus. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Quisque enim arcu, suscipit nec, tempus at, imperdiet vel, metus. Morbi volutpat purus at erat. Donec dignissim, sem id semper tempus, nibh massa eleifend turpis, sed pellentesque wisi purus sed libero. Nullam lobortis tortor vel risus. Pellentesque consequat nulla eu tellus. Donec velit. Aliquam fermentum, wisi ac rhoncus iaculis, tellus nunc malesuada orci, quis volutpat dui magna id mi. Nunc vel ante. Duis vitae lacus. Cras nec ipsum.