



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

# **Rotulação com Algoritmos Supervisionados**

**Tarcísio Franco Jaime**

**Número de Ordem PPGCC: M001**

**Teresina-PI, Janeiro de 2017**



Tarcísio Franco Jaime

## **Rotulação com Algoritmos Supervisionados**

**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

Janeiro de 2017

---

Tarcísio Franco Jaime

Rotulação com Algoritmos Supervisionados/ Tarcísio Franco Jaime. – Teresina-PI, Janeiro de 2017-

46 p. : il. (algumas color.) ; 30 cm.

Orientador: Vinicius Ponte Machado

Qualificação (Mestrado) – Universidade Federal do Piauí – UFPI

Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação, Janeiro de 2017.

1. Rotulação. 2. Algoritmos Supervisionados. 3. CART. 4. Naive Bayes. I. Vinicius Ponte Machado. II. Universidade Federal do Piauí. III. Rotulação com Algoritmos Supervisionados.

CDU 02:141:005.7

---

Tarcísio Franco Jaime

## Rotulação com Algoritmos Supervisionados

**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.

Trabalho aprovado. Teresina-PI, 01 de janeiro de 2018:

---

**Vinicius Ponte Machado**  
Orientador

---

Co-Orientador

---

**Professor**  
Convidado 1

---

**Professor**  
Convidado 2

---

**Professor**  
Convidado 3

Teresina-PI  
Janeiro de 2017



*Aos meus pais XXXXXXXX e YYYYYYY,  
por sempre estarem comigo em todos os momentos.*





# Agradecimentos

Agradeço a Deus.

Agradeço aos meus pais, XXXXX e YYYYY, por ...

Aos meus irmãos, por.....

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

Aos meus amigos ...

Aos professores ...

À XXXXXX pelo apoio financeiro para realização deste trabalho de pesquisa.



*“Não sei o que,  
não sei o que,  
não sei o que lá.”  
(Autor Desconhecido)*



# Resumo

Frente ao aumento do tráfego de dados em consequência de novas tecnologias, como também a necessidade de mais equipamentos conectados à rede passíveis de processamento de dados, cada vez mais algoritmos de aprendizado de máquina estão sendo estudados para extrair dados relevantes de grandes volumes de dados. A partir desse problema de interpretação, em grandes volumes de dados, tem-se um grau de dificuldade diretamente proporcional ao crescimento desse volume. É nesse tema onde este trabalho atua, no entendimento dos grupos que são formados e não na criação dos mesmos. Diante o entendimento desses grupos esta pesquisa realiza de forma empírica, ou seja, através de experimentos e testes, a identificação de atributos mais significativos no grupo, junto com faixa de valores que mais se repetem a ponto de representá-lo (rotulação). Dessa forma para a realização da rotulação de grupos de dados a proposta desta pesquisa é utilizar dois algoritmos supervisionados, cada um, com paradigmas diferentes: Naive Bayes (estatístico) e CART (simbólico). E a partir dos testes demonstrar que a rotulação é capaz de representar o grupo. Nos resultados obtemos uma acurácia acima de 70% de acerto dos valores representados pelo rótulo escolhido.

**Palavras-chaves:** cluster. rotulação. aprendizado supervisionado.



# Abstract

This is the english abstract.

**Keywords:** cluster. rotulação.





# Lista de ilustrações

Figura 1 – Hipóteses ajustadas . . . . .	7
Figura 2 – Ponto de Corte (R-1) . . . . .	11
Figura 3 – Discretização EWD . . . . .	12
Figura 4 – Discretização EFD . . . . .	13
Figura 5 – Modelo de Lopes (2014) . . . . .	14
Figura 6 – Comportamento da base de dados a cada iteração. Método (LIMA, 2015)	14
Figura 7 – Modelo de Resolução Proposto . . . . .	17
Figura 8 – Exemplo da técnica de correlação aplicada aos atributos atr1, atr2 e atr3 sendo classe . . . . .	18
Figura 9 – Exemplo da técnica de correlação aplicada ao atributo, atr1, sendo classe	19
Figura 10 – Discretização de atributos utilizando EFD com $R = 3$ (Figura adaptada de Lopes (2014)) . . . . .	21
Figura 11 – Resultado dos Algoritmos . . . . .	23
Figura 12 – Gráfico de Execuções dos algoritmos supervisionados na base de dados SEEDS. . . . .	40
Figura 13 – Gráfico de Execuções dos algoritmos supervisionados na base de dados IRIS. . . . .	41
Figura 14 – Gráfico de Execuções dos algoritmos supervisionados na base de dados GLASS. . . . .	42
Figura 15 – Acurácia por Clusters (Os clusters estão numerados em ordem crescente em cada Base de Dados . . . . .	43



# Lista de tabelas

Tabela 1	– Base de Dados Modelo . . . . .	20
Tabela 2	– Base de Dados Modelo Discretizada . . . . .	21
Tabela 3	– Valores das faixas com R=3 da Base de Dados Modelo . . . . .	22
Tabela 4	– Resultado da rotulação com o algoritmo Naive Bayes . . . . .	27
Tabela 5	– Resultado da Correlação dos atributos pelo Naive Bayes; Legenda dos Atributos: (A)area, (B)perimetro, (C)compacteness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove . . . . .	28
Tabela 6	– Resultado de 4( <i>quatro</i> ) execuções do algoritmo Naive Bayes; Legenda dos Atributos: (A)area, (B)perimetro, (C)compacteness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove . . . . .	28
Tabela 7	– Resultado da aplicação do algoritmo CART . . . . .	29
Tabela 8	– Resultado da Correlação dos atributos pelo CART; Legenda dos Atributos: (A)area, (B)perimetro, (C)compacteness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove . . . . .	29
Tabela 9	– Resultado de 4( <i>quatro</i> ) iterações do algoritmo CART; Legenda dos Atributos: (A)area, (B)perimetro, (C)compacteness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove . . . . .	29
Tabela 10	– Resultado da aplicação do algoritmo Naive Bayes . . . . .	31
Tabela 11	– Resultado (em %) de 4( <i>quatro</i> ) execuções do algoritmo Naive Bayes; Legenda dos Atributos: (SL)sepallength,(SW)sepalwidth,(PL)petallength,(PW)petalwidth . . . . .	31
Tabela 12	– Resultado da aplicação do algoritmo CART . . . . .	32
Tabela 13	– Resultado de 4( <i>quatro</i> ) iterações do algoritmo CART; Legenda dos Atributos: (SL)sepallength,(SW)sepalwidth,(PL)petallength,(PW)petalwidth . . . . .	32
Tabela 14	– Resultado da aplicação do algoritmo Naive Bayes . . . . .	34
Tabela 15	– Resultado de 4( <i>quatro</i> ) execuções do algoritmo Naive Bayes. . . . .	35
Tabela 16	– Resultado da aplicação do algoritmo CART . . . . .	36
Tabela 17	– Resultado de 4( <i>quatro</i> ) execuções do algoritmo CART. . . . .	36
Tabela 18	– Cronograma de atividades . . . . .	44



# Lista de abreviaturas e siglas

EWD	Discretização por Larguras Iguais
EFD	Discretização por Frequências Iguais
CART	Classification and Regression Trees



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>5</b>
<b>2.1</b>	<b>Aprendizado de Máquina</b>	<b>5</b>
2.1.1	Aprendizado Supervisionado	6
2.1.1.1	Algoritmo Classification and Regression Trees - CART	7
2.1.1.2	Algoritmo Naive Bayes	9
2.1.2	Aprendizado Não-Supervisionado	10
<b>2.2</b>	<b>Discretização</b>	<b>11</b>
2.2.1	Discretização por Larguras Iguais - EWD	11
2.2.2	Discretização por Frequência Iguais - EFD	12
<b>2.3</b>	<b>Trabalhos Correlatos</b>	<b>13</b>
<b>3</b>	<b>METODOLOGIA</b>	<b>15</b>
<b>3.1</b>	<b>Rotulação de Cluster</b>	<b>15</b>
<b>3.2</b>	<b>O Modelo de Resolução</b>	<b>16</b>
<b>3.3</b>	<b>Técnica de Correlação entre Atributos através de Algoritmos Super-</b>	
	<b>visionados</b>	<b>18</b>
<b>3.4</b>	<b>Exemplo</b>	<b>19</b>
3.4.1	Processo (I) - Discretização	20
3.4.2	Processo (II) - Algoritmos Supervisionados	22
3.4.3	Processo (III) - Rotulação	23
<b>4</b>	<b>RESULTADOS</b>	<b>25</b>
<b>4.1</b>	<b>Implementação</b>	<b>25</b>
<b>4.2</b>	<b>Seeds - Identificação de Tipos de Semente</b>	<b>26</b>
4.2.1	Naive Bayes	27
4.2.2	CART	28
<b>4.3</b>	<b>Iris - Identificação de Tipos de Plantas</b>	<b>30</b>
4.3.1	Naive Bayes	30
4.3.2	CART	32
<b>4.4</b>	<b>Glass - Identificação de Tipos de Vidros</b>	<b>33</b>
4.4.1	Naive Bayes	33
4.4.2	CART	35
<b>5</b>	<b>CONCLUSÕES, TRABALHOS FUTUROS E CRONOGRAMA</b>	<b>38</b>

5.1	Conclusão . . . . .	38
5.2	Trabalhos Futuros . . . . .	43
5.3	Cronograma . . . . .	44
	REFERÊNCIAS . . . . .	45



# 1 Introdução

Agrupamento de dados, ou clustering, é o termo que se usa para identificar dois ou mais objetos pertencentes ao mesmo grupo que compartilham um conceito em comum (KUMAR; ANDU; THANAMANI, 2013). Cluster é um termo bastante pesquisado no aprendizado não-supervisionado (subárea do aprendizado de máquina) e aplicada em vários contextos como segmentação de imagens, recuperação de informação e reconhecimento de objetos. Os algoritmos de agrupamento, conforme Kumar, Andu e Thanamani (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. E hoje, termo como Big Data, faz parte do cotidiano de empresas e pessoas. De acordo com o autor Montgomery (2013) Big Data são os dados que excedem a capacidade de sistemas de banco de dados. É nesse cenário, com grandes volumes de dados, que não só a formação de grupos ganha importância, mas também a compreensão dos mesmos, pois a interpretação dos grupos fornecerá informações úteis para análises desses clusters.

O grau de escalabilidade dos dados gradativamente aumenta no decorrer dos anos, e embora os estudos sobre o problema de agrupamento de dados estejam avançados, fica cada vez mais complexo o entendimento dos clusters formados pela razão do número crescentes de grupos criados. Quanto maiores são os números de grupos produzidos mais difícil são suas interpretações.

Diante desse contexto é que se extrai a temática desta proposta de mestrado - Rotulação automática de grupos através de algoritmos supervisionados baseados em árvores e estatísticos - o estudo em questão dedica-se na aplicabilidade de dois algoritmos supervisionados, com paradigmas diferentes e 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 de representá-lo seria encontrar, através de técnicas, uma tupla: atributo(s), faixa(s). Onde o atributo selecionado seria o que teria maior relevância no grupo. E a faixa de valor escolhida seria a que mais tivesse ocorrência nos valores do atributo, podendo também haver mais de um atributo com sua respectiva faixa, para representar o rótulo no grupo.

O termo rotulação, neste trabalho, segue a definição conforme [Lopes \(2014\)](#):

**Definição 1** *Dado um conjunto de clusters  $C = \{c_1, \dots, c_k | K \geq 1\}$ , de modo que cada cluster contém um conjunto de elementos  $c_i = \{\vec{e}_1, \dots, \vec{e}_{n(c_i)} | n(c_i) \geq 1\}$  que podem ser representados por um vetor de atributos definidos em  $\mathbb{R}^m$  e expresso por  $\vec{e}^i = (a_1, \dots, a_m)$  e ainda que com  $c_i \cap c_{i'} = \emptyset$  com  $1 \leq i, i' \leq K$  e  $i \neq i'$ ; o objetivo consiste em apresentar um conjunto de rótulos  $R = \{r_{c1}, \dots, r_{ck}\}$ , no qual cada rótulo específico é dados por um conjunto de pares de valores, atributo e seu respectivo intervalo,  $r_{c_i} = \{(a_1, [p_1, q_1]), \dots, (a_{m(c_i)}, [p_{m(c_i)}, q_{m(c_i)}])\}$  capaz de melhor expressar o cluster  $c_i$  associado.*

- $K$  é o número de clusters;
- $c_i$  é o  $i$ -ésimo cluster qualquer;
- $n^{c_i}$  é o número de elementos do cluster  $c_i$ ;
- $\vec{e}_{n(c_i)}$  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;
- $m$  é a dimensão do problema;

A formação do problema desta pesquisa nasce a partir do trabalho realizado por [Lopes \(2014\)](#), que se dedicou a estudar a possibilidade de realização de rotulação automática de grupos utilizando algoritmo não-supervisionado (K-means) para formação de grupos e algoritmo supervisionado (Redes Neurais) para a rotulação. Assim, partindo deste estudo já realizado, este trabalho dedica-se a realizar rotulação de grupos de dados a partir de outros algoritmos supervisionados não testados, em específico Naive Bayes e CART.

É necessário mensurar nesta pesquisa, a acurácia de cada resultado através do percentual de acertos dos atributos que são representados pelos rótulos gerados. É importante destacar, que este trabalho não se preocupa em criar grupos, mas dar maior relevância à rotulação dos mesmos, isto é, compreender os grupos de dados já formados.

Quando se analisa grupos que já estão formados sabe-se que esses grupos existem, pois existe uma correlação das características pelo qual seus dados se mantêm junto em grupos, e 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 estes grupos, gerando um tipo de rótulo,  $R = \{r_{c1}, \dots, r_{ck}\}$ , 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 na cidade 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 aplicado rotulação nestes grupos. E através dos rótulos pode-se personalizar alguns serviços para esses 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, o qual definiriam o tipo de localização. Logo percebe-se que os participantes desse grupo possuem característica de frequentar alguma localização em comum. A interpretação deste rótulo poderá implicar em uma tomada de decisão personalizada para este grupo, objetivando otimizar um problema.

O trabalho em questão tem como objetivo principal demonstrar a possibilidade de fazer rotulação de dados, em bases de dados com grupos já formados, utilizando dois algoritmos supervisionados distintos com paradigmas diferentes. Sendo este um algoritmo com paradigma estatístico - Naive Bayes - e outro com paradigma simbólico - Classification And Regression Tree (CART).

Para alcançar tal objetivo é necessário conhecer as técnicas e tecnologias utilizadas nessa pesquisa. Uma das técnicas utilizada é a discretização (seção 2.2) onde ocorre a representação do tipo de variáveis contínuas em discretas. Outra técnica é a correlação entre atributos (seção 3.3), visto que nesse processo é aplicado os algoritmos referentes da pesquisa. Todas essas técnicas foram estruturadas mediante a codificação por intermédio de uma linguagem de natureza técnica, onde fez uso de módulos de aprendizado de máquina, atuando em bases de dados e obtendo como saída deste programa, os rótulos dos grupos.

O trabalho será disposto em cinco capítulos já incluso a Introdução e Conclusão, capítulos 1 e 5 respectivamente.

O Referencial Teórico abordado no capítulo 2 é responsável em esclarecer as tecnologias utilizadas nesta pesquisa sendo dividida em três seções. Inicialmente na seção 2.1, tem-se uma explanação sobre aprendizado de máquina e quais os aprendizados indutivos são mais relevantes para este trabalho, ademais, a explicação dos dois algoritmos supervisionados utilizados para fazer rotulação de dados. Já na seção 2.2 é realizado a divisão das faixas de valores de cada atributo, chamada de discretização. E logo na seção 2.3 são apresentas pesquisas já consolidadas referentes ao assunto de rotulação de clusters.

Na 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 3.3 é 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: discretização da base

de dados no Processos (I), no Processo (II) é aplicado o algoritmo supervisionado e no Processo (III) o resultado da rotulação.

No capítulo 4 os resultados são separados por base de dados. Em cada seção referente a uma base de dados testada é criada 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 esses valores desta tabela servirão de apoio para entender como o quão os atributos estão bem correlacionados.

Diante do exposto fica claro que esta pesquisa além de dar continuidade a um tema específico aplicado na interpretação de agrupamento de dados, também serve como ponto de partida para outra pesquisa mais aprofundada, onde poderá esta tentar comprovar a possibilidade de fazer rotulação de dados utilizando qualquer algoritmo supervisionado.

## 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.

A aprendizagem de máquina utiliza métodos de inferências lógicas para aquisições de novos conhecimentos, e um dos tipos de inferência comentada nesta seção são os aprendizados indutivos, que dentre seus tipos terá o maior destaque ao aprendizado supervisionado, foco dessa proposta de mestrado. “Na aprendizagem indutiva os algoritmos podem, na melhor das hipóteses, garantir que a hipótese de saída se encaixe no conceito de destino sobre os dados de treinamento” (MITCHELL, 1997, p.23).

Já na seção 2.2 dissertará sobre a técnica de discretização adotada nesta pesquisa. Possuindo grande contribuição para os resultados gerados, e ganhando assim uma seção própria para explanação de como funciona essa técnica. E na seção 2.3, serão abordados trabalhos que possuam 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).

Para melhor explicar a citação acima, destaca-se o determinado exemplo: considerar o reconhecimento facial de uma pessoa utilizando aprendizado de máquina. Então caso fossem inseridas várias fotos tituladas de uma certa pessoa (T) no banco de dados, e após vários exemplos (E), fotos dessa pessoa, o programa de computador seria capaz de prever (P) se uma nova foto, ainda não inserida no banco de dados, seria dessa determinada pessoa através de aprendizado anterior (E), ou melhor, de fotos que foram anteriormente inseridas.

O aprendizado de máquina seriam 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 idéia do exemplo anterior, uma vez inserida uma foto no banco de dados e determiná-la como masculina, nesse momento, estará se fazendo uma classificação desse novo registro (nova foto). Uma vez com a base de dados classificada, pode-se utilizar algoritmos para prever um novo registro e defini-lo como masculino ou feminino. Prever uma determinada condição irá depender da base de dados como também do algoritmo utilizado para fazer essa classificação. Alguns exemplos de algoritmos são: redes neurais, árvores de decisão, Suport Vector Machine – SVM, etc. A escolha apropriada do algoritmo se dará através de métricas que avaliarão o desempenho de cada um e a melhor métrica servirá de parâmetro para a escolha do algoritmo apropriado para aquele problema de classificação de dados.

Em aprendizado de máquina são várias as abordagens encontradas, e segundo [Mohri, Rostamizadeh e Talwalkar \(2012\)](#) são eles: aprendizado supervisionado, aprendizado não-supervisionado, aprendizado semi-supervisionado, aprendizado por reforço e muitos outros. Todavia nesta pesquisa serão comentados somente algumas abordagens de referência específicas para esse trabalho.

### 2.1.1 Aprendizado Supervisionado

O aprendizado supervisionado é um método que através de uma base de dados classificada, será realizado 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 determinado conjunto de dados (conjunto de treinamento), mas precisa ser previstos 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 o padrão de exemplo anterior considere: uma base de dados de imagens de rostos, onde 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 prever qual sexo pertence cada rosto ([BARBER, 2011](#)).

Em [Russel e Norvig \(2013\)](#) é feita apresentação formal do funcionamento da

aprendizagem supervisionada. Dado um conjunto de treinamento

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n), \quad (2.1)$$

onde cada  $y_j$  foi gerado por  $y = f(x)$  desconhecida. Encontrar uma função  $h$  que se aproxime da função  $f$  real.

A função  $h$  é uma hipótese onde prevê um melhor desempenho entre as hipóteses possíveis através dos conjuntos de dados, que são diferentes do conjunto de treinamento equação 2.1.

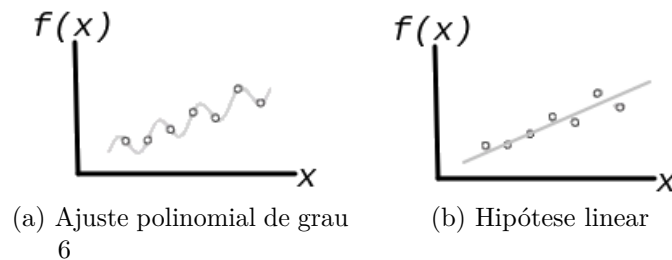


Figura 1 – Hipóteses ajustadas

O exemplo da figura 1a mostra uma função de grau 6 onde acontece um sobreajuste (overfitting) no conjunto de dados de treinamento. Esse modelo acabou exibindo uma função mais complexa para atender todo o conjunto de dados do gráfico, ficando específico para essa amostra.

Já na figura 1b o ajuste da função se torna mais simples e mesmo o gráfico não passando por todos os pontos, acabou por generalizar melhor o conjunto de treinamento, tornando talvez, um melhor resultado da predição de novos valores.

Em análise da figura 1 é apresentado duas hipóteses que tentam se aproximar ao máximo da função verdadeira ( $h$ ), que é desconhecida. Mesmo parecendo que na figura 1a obteve-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 1a por ser mais específica é a melhor função, não é a opção correta. Quanto mais generalizado for modelo, melhor será para predizer os valores de  $y$  para novos conjuntos de dados.

#### 2.1.1.1 Algoritmo Classification and Regression Trees - CART

Esse algoritmo constroi modelos de previsão a partir de dados de treinamento onde 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, onde a cada nó interno é feito um teste para tomada de decisão, 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)

O algoritmo CART pode se tornar uma árvore de classificação ou também um árvore de regressão, o que irá definir seu tipo seria o atributo classe. Por exemplo, em um conjunto de dados de um paciente onde tenta prever se o mesmo possuirá câncer. A classe seria “Terá Câncer” ou “Não terá Câncer”. Nesse exemplo o atributo assume duas classes.

Mas ao contrário de uma árvore de classificação que prediz uma classe, o CART também pode assumir uma árvore de regressão, onde poderá prever um valor numérico ou contínuo, como: período de tempo de internação do paciente, preço de uma cirurgia ou quantidade de água ingerida.

No caso de não ser probabilístico o grau de confiança em seu modelo de predição será embasada em respostas semelhantes em outras circunstâncias antes analisadas.

O CART utiliza uma técnica como partição recursiva binária. Binária porque a divisão do nó pai é sempre em dois nós filhos, e recursiva porque cada nó filho irá ser tratado posteriormente no processo como nó pai.

Inicialmente todas as amostras se concentram no nó raiz, e a partir daí é apresentado uma questão, onde a intenção é separar o nó raiz em dois grupos mais homogêneos. O objetivo é criar uma regra que inicialmente crie grupos ou nós binários que sejam internamente mais homogêneos que o nó raiz. Dependendo da questão as amostras irão para a folha esquerda ou direita do nó raiz. O CART faz a escolha dessa divisão em função da regra Gini de Impureza<sup>1</sup> (BREIMAN et al., 1984), e o índice Gini varia de 0 a 1, definindo o grau de pureza do nó.

$$Gini(S) = 1 - \sum p^2(j/t) \quad (2.2)$$

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

Para construção de uma árvore pode-se simplificar em três componente importantes (YOHANNES; WEBB, 1999; RAIMUNDO; MATTOS; WALESKA, 2008):

- 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 atribuído ao nó pai uma questão onde dependendo da resposta (sim ou não) os registros irão para nó filho esquerdo ou direito. Esse questionamento binário servirá dividindo o grupo em dois, para depois testar o ponto de divisão.

<sup>1</sup> O CART pode utilizar outros critérios de divisão de dados como: entropia e critério de Twoing



O CART percorrerá todos os atributos construindo uma árvore para descobrir qual melhor ponto de divisão. Uma vez testados todos os atributos é escolhido o que tiver menor grau de impureza do nó medido através do critério Gini, onde o grau de pureza do nó é mais puro, quando no grupo houver mais casos pertencente a uma única classe. Fazendo que o índice Gini seja mais alto.

Após a escolha do melhor ponto de divisão do nó faz-se a atribuição de uma classe para o nó, onde a escolha da classe será a que mais exemplos contiver no grupo.

### 2.1.1.2 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 a condição da probabilidade de ocorrência anteriores do mesmo. É nesse seguimento que o algoritmo Naive Bayes funciona. Criando classificadores probabilístico 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 utilizada para da maior precisão a probabilidade de câncer, ao invés de fazer uma avaliação de probabilidade sem a utilização do sexo da pessoa.

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

Em Russel e 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.

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

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 dados. Ele agrupa objetos de uma certa classe

em razão da probabilidade do objeto pertencer a esta classe.

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

$$P(c/x) = P(x_1|c) * P(x_2|c) * \dots * P(x_n|c) * P(c) \quad (2.5)$$

- $P(c/x)$  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 e Nigam \(1997\)](#). Mas mesmo atingido boa popularidade possui pontos negativos. A suposição de ter preditores independentes não acontece muito na vida real, pois acaba sendo difícil ter uma amostra de dados que sejam inteiramente independentes.

### 2.1.2 Aprendizado Não-Supervisionado

Outro cenário de aprendizado de máquina é o aprendizado não-supervisionado, onde nessa abordagem 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 deles estarem juntos.

Quando o algoritmo encontra dados com mesma similaridade ele os agrupa formando clusters. Os números de clusters encontrados dependerá do funcionamento dos algoritmos e também do grau de dissimilaridade entre elementos de grupos diferentes. Como não existe uma variável classe no aprendizado não-supervisionado, então segundo [Barber \(2011\)](#), o maior interesse seria em uma perspectiva probabilística de distribuição  $p(x)$  de um determinado conjunto de dados.

$$D = \{x_n, n = 1, \dots, N\} \quad (2.6)$$

Uma vez que no conjunto (2.6), não existe classe  $y$ , encontrado em um conjunto de treinamento, equação 2.1, o algoritmo precisa encontrar padrões nos atributos para fazer os agrupamentos.

## 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 os atributos possuíram novos conteúdos no formato de faixas de valores.

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

### 2.2.1 Discretização por Larguras Iguais - EWD

O método de Discretização por Larguras Iguais (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 2.

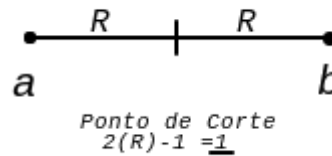


Figura 2 – Ponto de Corte (R-1)

Para haver o ponto de corte antes tem que ser realizado a ordenação dos dados. A largura de cada faixa  $r_1, \dots, r_R$  na equação 2.7 é representada por  $w$ , que é calculada pela diferença entre os limites superior e inferior do intervalo, dividido pela quantidade  $R$  de valores a serem gerados.

$$w = \frac{b - a}{R} \quad (2.7)$$

A variável  $w$  determina os pontos de corte  $(c_1, \dots, c_{R-1})$  que irão delimitar o tamanho das faixas de valores. O primeiro ponto de corte,  $c_1$ , é obtido através 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  $w$ .

O valor de cada faixa será representado por  $i$ , onde  $i$  é o índice indicando a faixa. De acordo com a figura 3 para dividir o intervalo  $[a, b]$  em  $R$  faixas será necessário de  $R - 1$  pontos de corte.

$$c_i = \begin{cases} a + w, & \text{se } i = 1 \\ c_{i-1} + w, & \text{caso contrário} \end{cases} \quad (2.8)$$

O valor da faixa do intervalo  $[a, c_1]$  será o valor discreto igual ao índice de sua faixa  $r_1$ . Então, um valor na faixa  $r_1$  terá o valor representado por  $1(um)$ , pois  $i = 1$  é o limite inferior mais largura da faixa, equação 2.8. E seguindo o mesmo raciocínio o valor da faixa  $r_2 = ]c_1, c_2]$  é representado por  $2(dois)$ , e conseqüentemente o valor que se encontra em uma faixa qualquer  $r_i$  será representado por  $i$ .

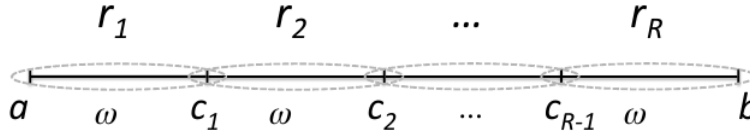


Figura 3 – Discretização EWD baseada em (LOPES, 2014)

### 2.2.2 Discretização por Frequência Iguais - EFD

Esse outro método de discretização já possui uma abordagem diferente a do EWD, pois a idéia 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  $\xi$ , onde  $\xi \geq R$  o método EFD irá segmentar em  $R$  faixas de valores que possuem a mesma quantidade de elementos distintos  $\lambda$ . Então serão realizados  $R - 1$  pontos de corte gerando  $R$  faixas de valores,  $(r_1, \dots, r_R)$ , com a mesma quantidade de elementos distintos  $\lambda$ . Para encontrar  $\lambda$  calcula-se o valor inteiro da divisão entre a quantidade de elementos distintos  $\xi$  pela quantidade de faixas de valores  $R$ , obtendo o número de elementos da faixa 2.9.

$$\lambda = \frac{\xi}{R} \quad (2.9)$$

Uma observação nesse método é a ocorrência em amostras que possuem uma má distribuição de valores de um dado atributo. Como um número significativo de repetições, causando um desequilíbrio nas distribuições dos elementos.

Uma vez no intervalo  $[a, b]$  de elementos ordenado e calculado  $\lambda$  contendo  $R$  elementos  $v_{[R]}$  pode-se determinar os pontos de corte  $(c_1, \dots, c_{R-1})$  que são os delimitadores das faixas. Cada ponto de corte  $c_i$  pode ser calculado por  $v_{i\lambda}$  - *ésimo* elemento, 2.10.

$$c_i = v_{[i\lambda]} \quad (2.10)$$

Igual o 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 4. 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.

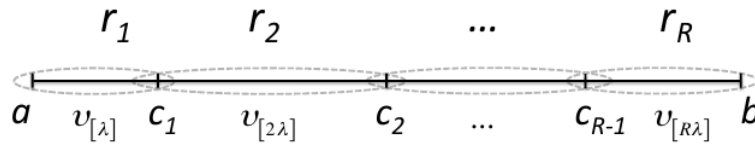


Figura 4 – Discretização EFD<sup>2</sup>

## 2.3 Trabalhos Correlatos

Esta seção propõe relacionar outros trabalhos servindo de complemento teórico para entender a variedade de aplicações referente ao assunto de rotulação de dados.

O trabalho escrito por [Lopes \(2014\)](#) fez um estudo abordando o tema de rotulação de dados, tema este, proposto também por esta pesquisa, mas com abrangência e execução diferentes do modelo da figura 5. No trabalho de [Lopes \(2014\)](#) foi utilizado como entrada um conjunto de dados onde foi feito o agrupamento automático, com algoritmos não-supervisionados formando clusters. Logo após é utilizado um algoritmo supervisionado (Redes Neurais) nos grupos de dados, e apresentado como saída um rótulo específico que melhor define o grupo formado. Esses rótulos são formados pela faixa de valor, que mais se repetem, em conjunto com os atributos mais relevantes.

Pode-se verificar na figura 5 que na parte onde é aplicado o algoritmo de redes neurais (processo III) é o local exato que esta pesquisa utiliza para testar outros algoritmos supervisionados, servindo para comprovar a hipótese desta proposta de mestrado e validação do modelo.

Em [Lima \(2015\)](#) o problema em questão é fazer classificação e rotulação em uma base que possuem poucos elementos classificados utilizando método semi-supervisionado. O método inicia com uma base dividida em elementos classificados(L) e não classificados(U). Após cada iteração o grupo L vai crescendo e automaticamente diminuindo o grupo U até que não tenha mais nenhum elemento em U, figura 6. Após isso é realizada uma etapa de agrupamento, sem levar em consideração os dados classificados anteriormente. Terminada essa etapa é feito uma validação para saber quais os rótulos foram considerados corretos.

<sup>2</sup> Figura extraída de ([LOPES, 2014](#))

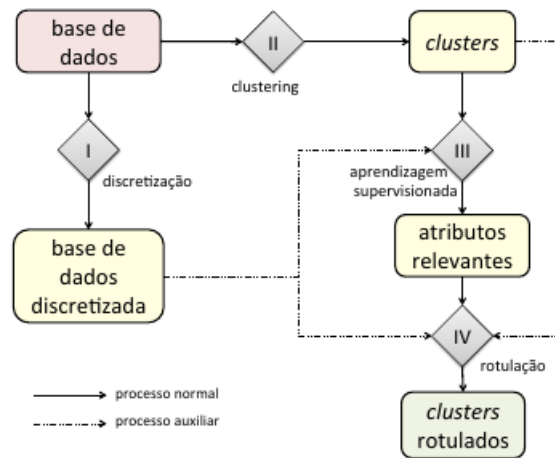


Figura 5 – Modelo de Lopes (2014)

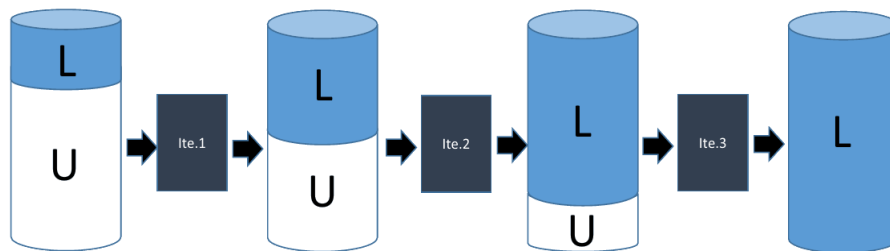


Figura 6 – Comportamento da base de dados a cada iteração. Método (LIMA, 2015)

O método proposto é uma combinação de um classificador com um método de agrupamento, onde a rotulação de um conjunto de dados é feita com conhecimento prévio de um outro conjunto menor rotulado. O classificador treina com a parte de dados rotulada e classifica os dados não-rotulados.

Outra pesquisa sobre rotulação está em (FILHO, 2015) onde aborda o mesmo Problema de Rotulação, mas a atuação é diferenciada, pois o modelo procura diferenças existentes em cada grupo através da seleção dos elementos que representam o grupo, e depois é construído a faixa de valores. Os grupos são formados pelo algoritmo Fuzzy C-Means e após isso que é selecionado os atributos.

## 3 Metodologia

O texto a seguir abordará o problema a ser atacado por esse trabalho, e logo em seguida, será apresentado um modelo de resolução para rotulação de dados utilizando dois algoritmos supervisionados baseados em paradigmas distintos. O objetivo ao final deste capítulo é poder provar que é possível realizar rotulação com utilização de dois algoritmos, e também revalidar o modelo de [Lopes \(2014\)](#) que utilizou somente redes neurais para realização da rotulação de grupos de dados.

### 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 a classificação dos dados e não da rotulação. Ao agrupar um conjunto de elementos por um determinado critério, esta havendo uma classificação desses elementos de mesma similaridade, mas pouco se sabe qual é a compreensão desses grupos já classificados.

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 (rótulo).

A criação do rótulo é a escolha de uma tupla **atributo** e **faixa de valor**, onde o atributo possui o maior valor de correlacionamento entre os outros atributos. E a faixa escolhida é aquela que mais se repete desse atributo rótulo. Podendo o cluster ter mais de um rótulo contendo a tupla atributo e faixa (Definição 2).

Essa faixa, é um intervalo de valor definido pela discretização seção 2.2, onde o intervalo escolhido, seria a faixa que representa os valores que se repetem com a maior frequência no atributo. A exemplo disso, tem-se um vetor de elementos já discretizados,  $\vec{a}_i = \{1, 1, 1, 2, 2, 2, 2, 3, 3\}$ , onde  $i \leq m$  e  $(\vec{a})$  representa todos os elementos da coluna representada pelo atributo  $(a)$ . Neste vetor o valor que mais se repete é o número 2, então, a **faixa 2** do atributo rótulo é a escolhida para compor o rótulo.

O Problema de Rotulação é formalmente definido como segue abaixo:

**Definição 2** Dado um conjunto de clusters  $C = \{c_1, \dots, c_k | K \geq 1\}$ , de modo que cada cluster contém um conjunto de elementos  $c_i = \{\vec{e}_1, \dots, \vec{e}_{n(c_i)} | n^{(c_i)} \geq 1\}$  que podem ser re-

presentados por um vetor de atributos definidos em  $\mathbb{R}^m$  e expresso por  $\vec{e}^i = (a_1, \dots, a_m)$  e ainda que com  $c_i \cap c_{i'} = 0$  com  $1 \leq i, i' \leq K$  e  $i \neq i'$  (Adaptada de [Lopes \(2014\)](#)).<sup>1</sup>

- $K$  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  $c_i$ ;
- $\vec{e}_{n(c_i)}$  se refere ao  $j$ -ésimo elemento pertencente ao cluster  $c_i$ ;
- $m$  é a dimensão do problema;

## 3.2 O Modelo de Resolução

A partir da definição do problema - *Definição 2* - um estudo científico foi desenvolvido nesta pesquisa, a fim de provar, que é possível a realização de rotulação de dados com dois algoritmos supervisionados de paradigmas diferentes: Naive Bayes e CART.

Este modelo de resolução consiste em apresentar como saída um conjunto de rótulos, onde cada rótulo específico é dado por um conjunto de pares de valores, atributo e seus respectivos intervalos, 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  $R = \{r_{c1}, \dots, r_{ck}\}$ , no qual cada rótulo específico é dados por um conjunto de pares de valores, tem como saída um vetor com atributo e seu respectivo intervalo,  $r_{c_i} = \{(a_1, [p_1, q_1]), \dots, (a_{m(c_i)}, [p_{m(c_i)}, q_{m(c_i)}])\}$  capaz de melhor expressar o cluster  $c_i$  (Adaptada de [Lopes \(2014\)](#)).<sup>2</sup>

- $k$  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 a dimensão do problema;

Como apresentado na seção 2.3, o autor foca em rotulação automática de grupos utilizando a estratégia de aprendizagem de máquina supervisionada, com paradigma connexionista, para provar seu trabalho. Porém, nesta pesquisa foi aplicado no modelo de resolução dois algoritmos com paradigmas de aprendizado diferente do que já havia



sido testado anteriormente, provando que é possível fazer rotulação de dados com algoritmos supervisionados com paradigmas simbólico e probabilístico, CART e Naive Bayes respectivamente.

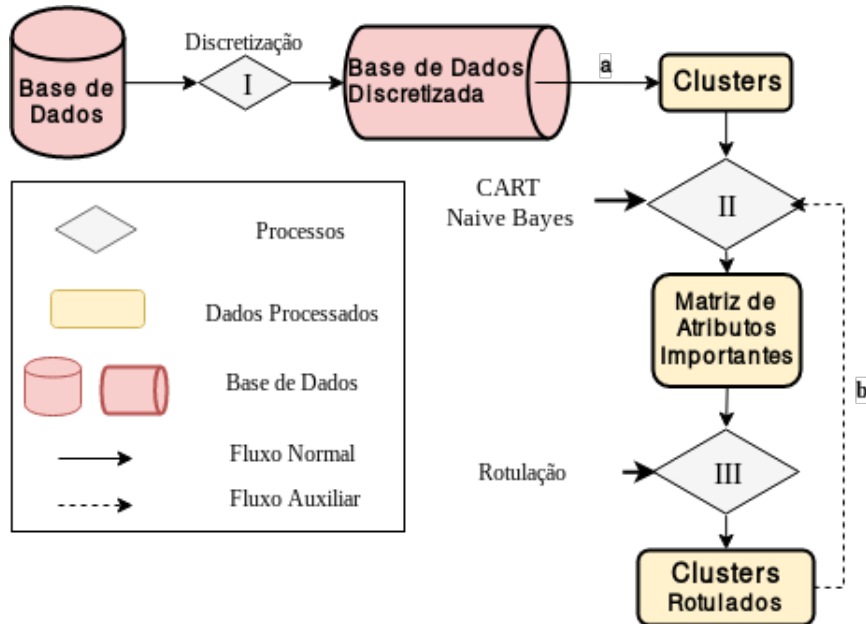


Figura 7 – Modelo de Resolução Proposto

A base<sup>3</sup> conterá valores contínuos, contudo, conforme modelo será necessário aplicar o método de discretização (I). Uma vez com a base discretizada ocorre a divisão em clusters que nada mais é do que a separação da base em grupos já classificados.

No passo II serão executados os algoritmos de aprendizagem supervisionados, já visto nas subseções 2.1.1.1 e 2.1.1.2. Essa etapa utiliza a técnica de correlação de atributos através de algoritmos supervisionados, que neste modelo de resolução é considerado o processo de maior importância, visto na seção 3.3.

A saída do processo II gera uma matriz de atributos com seus respectivos valores, e que através desses valores armazenados é escolhido o atributo de maior relevância. Se comparado ao trabalho de Lopes (2014) que utilizou somente redes neurais para fazer esse processo, nesse trabalho foi introduzido dois algoritmos supervisionados com paradigma probabilístico e de árvore de decisão, Naive Bayes e CART respectivamente.

Para gerar essa Matriz de Atributos Importantes, conforme modelo figura 7 o algoritmo supervisionado será aplicado em uma quantidade igual ao número de atributos. Após preenchido, esses valores da matriz serviram de referência para a escolha do atributo mais importante, sendo este o de maior valor. Utilizando a figura 8 como exemplo, o algoritmo supervisionado seria executado três vezes, sendo essa quantidade igual ao número de atributos: **atr1**, **atr2** e **atr3**.

<sup>3</sup> UCI - Machine Learning Repository. <http://archive.ics.uci.edu/ml/>

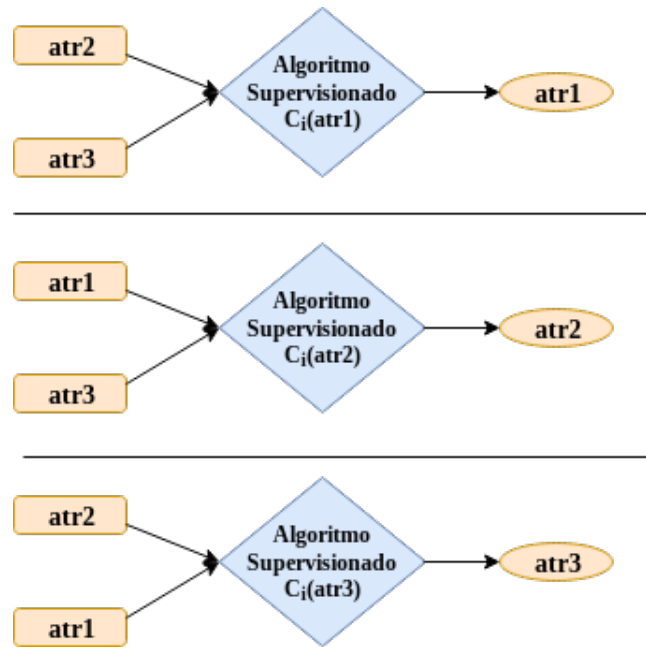


Figura 8 – Exemplo da técnica de correlação aplicada aos atributos atr1, atr2 e atr3 sendo classe

Seguindo para o 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 rotulos para cada clusters. O fluxo (b) será utilizado enquanto houver outros algoritmos para serem executados.

### 3.3 Técnica de Correlação entre Atributos através de Algoritmos Supervisionados

Essa técnica (LOPES, 2014) utiliza por analogia a aprendizagem supervisionada, no qual os atributos de entrada são correlacionados com um atributo classe (atributo de saída). 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.

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, seria armazenado o valor desse atributo em uma matriz de atributos importantes. Essa matriz após montada mostraria os valores de cada atributo enquanto classe, e quanto maior o valor, mais relevante será este atributo em relação aos demais.

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

Utilizando como exemplo uma base com os seguintes atributos: **atr1**, **atr2**, **atr3** e **classe**. Retirando o atributo classe, e atribuindo a cada iteração, um novo atributo classe, portanto, a base possui três atributos, então o algoritmo será aplicado três vezes, um para cada atributo gerando um valor que será armazenado em uma matriz. Em um primeiro processamento de três, o primeiro atributo **atr1** se torna classe e executado com os outros dois atributos, **atr2**, **atr3** de entrada com um algoritmo supervisionado, figura 9.



Figura 9 – Exemplo da técnica de correlação aplicada ao atributo, atr1, sendo classe

O resultado da correlação entre os atributos **atr2**, **atr3** em relação ao **atr1** (figura 9) é armazenado em uma matriz, denominada de **Atributos Importantes**, de acordo com figura 7. Por conseguinte é realizada a aplicação do algoritmo com **atr2** 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, conforme demonstrado na figura 8, e armazenado seus valores em porcentagem na tabela. No final uma tabela será formada pelos valores em porcentagem da correlação entre eles denominada Matriz de Atributos Importantes.

### 3.4 Exemplo

Para melhor esclarecer as etapas do modelo de resolução exibido na figura 7, será utilizado a tabela 1 como exemplo no modelo proposto nesta pesquisa. Essa tabela é composta por cinquenta linhas e quatro atributos, sendo o último um atributo classe, representando o cluster o qual o elemento pertence. Logo na primeira coluna da tabela, possui o índice da linha da tabela identificando cada registro, e outros campos são atributos que definem características do registro identificado pelo índice da primeira coluna até a quinta coluna representando a classe de cada registro.

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 1, pode ser representado por  $\vec{e}_2 = (1.26, 85.03, 20.45)$ .

Tabela 1 – Base de Dados Modelo

	atr1	atr2	atr3	classe		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

### 3.4.1 Processo (I) - Discretização

Segundo [Catlett \(1991\)](#), [Hwang e Li \(2002\)](#) através 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 1 será utilizada a técnica de discretização por frequências iguais - EFD - e divisão de números de faixas igual a  $R=3$ . Na figura 10 poderá ser visualizado como é feita a discretização.

Através da figura 10 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 na faixa 3 contém todos valores a partir do segundo ponto de

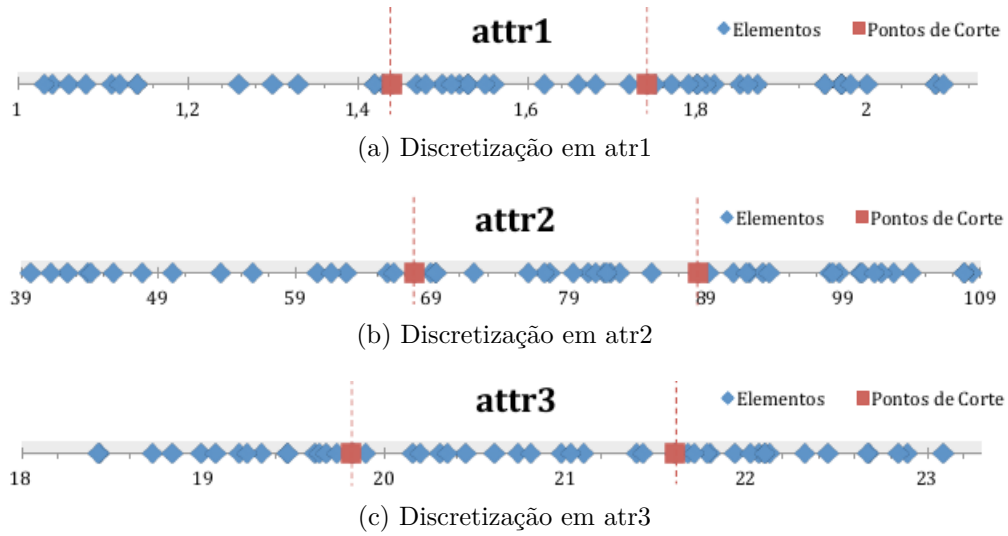


Figura 10 – Discretização de atributos utilizando EFD com  $R = 3$  (Figura adaptada de [Lopes \(2014\)](#))

corde.

Tabela 2 – 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 3 – Valores das faixas com  $R=3$  da Base de Dados Modelo

	Faixa 1	Faixa 2	Faixa 3
<b>atr1</b>	[ 1.03 ~1.44 ]	] 1.44 ~1.74 ]	] 1.74 ~2.09 ]
<b>atr2</b>	[ 39.67 ~67.66 ]	] 67.66 ~88.36 ]	] 88.36 ~108.36 ]
<b>atr3</b>	[ 18.42 ~19.82 ]	] 19.82 ~21.61 ]	] 21.61 ~23.09 ]

A tabela 2 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 10, onde  $R$  é o número de faixas a ser dividido tanto no **atr1** como também no **atr2** e **atr3** possuem os valores conforme tabela 3.

### 3.4.2 Processo (II) - Algoritmos Supervisionados

Ao chegar nessa etapa, Processo (II) da figura 7, já se tem uma base discretizada e clusters formados como visto na tabela 2. 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. Este valor irá compor uma matriz denominada de **Atributos Importantes**, cujo função é armazenar o resultado da execução dos algoritmos utilizando a técnica de correlação de atributos.

O algoritmo irá selecionar cluster por cluster, percorrendo todos os atributos destes clusters, onde 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, e verificar seu grau de importância entre eles. Depois o atributo **atr2** irá ser classe, e depois o **atr3**, fechando o ciclo de todos os atributos do cluster. Como visualizado na figura 8.

A cada aplicação do algoritmo supervisionado é armazenado para cada cluster  $c_i(atr)$  os valores de relevância dos atributos representada por uma porcentagem de acerto. O algoritmo será executado o mesmo número de vezes do número de atributos existente na base de dados, pois a cada iteração um atributo se torna um atributo classe, consequentemente é gerado seu valor de relevância em porcentagem. Quanto maior sua porcentagem, mais bem correlacionado é o atributo em relação aos demais (figura 11). Portanto esse atributo poderá resumir as características do problema, podendo ser considerado atributo mais relevante, e conseguinte escolhido como rótulo.

Na figura 11a mostra o resultado da execução do Naive Bayes trabalhando com a base modelo (tabela 2) exibindo os resultados em porcentagem de acerto de cada atributo em relação aos demais. O mesmo acontece com a figura 11b onde é aplicado um algoritmo de Árvore de Decisão - CART - exibindo o resultado de todas as taxas de acerto, em

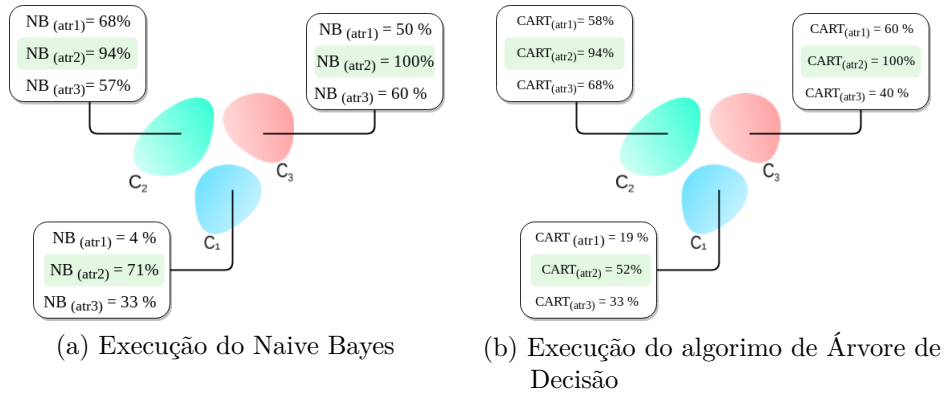


Figura 11 – Resultado dos Algoritmos

porcentagem, dos atributos de seus respectivos clusters.

Uma forma de eliminar uma possível ambiguidade entre os clusters foi adicionar na implementação uma variável  $V$ . Essa variável é utilizada para seleção dos atributos rótulos de um clusters, 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 que tiverem valores a partir de 80% são selecionados como rótulo do cluster.

O valor da variável  $V$  é subjetivo e irá ser arbitrado de acordo com os resultados em cada aplicação do algoritmo em um conjunto de dados. Nesse exemplo caso fosse utilizado a variância  $V = 12$  na matriz de atributos importantes representada pela figura 11a, teriam os atributos, por clusters,  $r_{c_i} : r_{c_1} = \{atr2\}$ ,  $r_{c_2} = \{atr2\}$ ,  $r_{c_3} = \{atr2\}$ .

O valor da variável  $V$  é subjetivo e irá ser arbitrado de acordo com os resultados em cada aplicação do algoritmo em cima de um conjunto de dados. Nesse exemplo caso fosse utilizado a variância  $V = 12$  na matriz de atributos importantes representada pela figura 11a, teriam os atributos, por clusters,  $r_{c_i} : r_{c_1} = atr2$ ,  $r_{c_2} = atr2$ ,  $r_{c_3} = atr2$ .

### 3.4.3 Processo (III) - Rotulação

No processo de rotulação os rótulos de cada cluster ( $c_i$ ) serão compostos conforme o modelo 3.1.

$$r_{c_i} = \{(a_1, [p_1, q_1]), \dots, (a_{m(c_i)}, [p_{m(c_i)}, q_{m(c_i)}])\} \quad (3.1)$$

Cada rótulo é composto pela tupla: atributo de maior relevância e a faixa de valor desse atributo que mais se repete. Na figura 11 os rótulos em destaque são os que possuem maior valor, ademais, cada atributo que faz parte do rótulo possui um vetor de valores, de onde será escolhido a faixa de maior ocorrência. Uma vez calculado e definido a faixa,

será determinado os limites inferiores ( $p_{m(c_i)}$ ) e superiores ( $q_{m(c_i)}$ ) de acordo com a tabela discretizada (exemplo 3).

Por exemplo, utilizando a Base Modelo, mais especificamente o cluster 1 ( $c_1$ ), cujo resultado é apresentado na figura 11a, 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 3. Os rótulos dos clusters descrito neste exemplo - conforme figura 11a e figura 11b - aplicado na BD Modelo são:

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

Ou seja, 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 possui 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.

Uma vez terminado o processo (III) de rotulação, o fluxo *b* da figura 7, só será executado caso seja necessário testar outro algoritmo.

O algoritmo 1 exibe a rotina em forma de pseudocódigo para melhor entendimento.

---

**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  $V \neq 0$  then
11      Carrega_atributos_importantes_considerando_V;
12      Associa_valores_aos_intervalos;
13 Exibe_rótulos_todos_clusters;
```

---



## 4 Resultados

Foram realizados testes com algumas bases de dados da UCI Machine Learning<sup>1</sup>, um repositório de dados a serviço da comunidade de aprendizado de máquina. Criado por estudantes de pós-graduação na UC Irvine em 1987 e até hoje é utilizado não só por estudantes mas também por educadores e pesquisadores como fonte primária de aplicações de aprendizado de máquina.

As bases de dados foram escolhidas não só por critério comparativo de outros trabalhos que também já as utilizaram servindo de referência para os resultados, como também, um cuidado de só escolher bases que estão classificadas, uma vez que esta pesquisa trabalhará com os clusters já formados e não na criação de grupos.

A divisão deste capítulo iniciará por uma explanação da implementação do trabalho, explicando as ferramentas utilizadas para desenvolver esse trabalho e quais configurações de algumas variáveis. Cada base de dados utilizada é destacada por uma seção, onde cada seção refere-se a uma base de dados, sendo esta, dividida em subseções para cada algoritmo utilizado: Naive Bayes e CART.

### 4.1 Implementação

Para gerar os resultados aqui escritos foram realizadas implementações utilizando a ferramenta MATLAB<sup>2,3</sup>, 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 direcionada para aprendizado de máquina essa ferramenta foi escolhida para colocar em prática essa pesquisa.

Ao longo da pesquisa foram realizados vários testes, porém, houveram alterações de algumas variáveis e métodos de discretização, sempre com o objetivo de obter os melhores resultados. Essas alterações, dependendo da base de dados utilizadas fazem as seguintes modificações na variável "V", na quantidade de faixas "R" e métodos de discretização "EWD,EFD".

Como dito na subseção 3.4.2 a variação  $V$  existe para evitar a ambiguidade dos rótulos, ou seja, quando rótulos apresentarem os mesmos resultados: atributo e faixa de valor. Além de evitar a ambiguidade dos rótulos a variável  $V$  pode ser utilizada também para selecionar mais de um atributo para ser o rótulo do cluster.

<sup>1</sup> <http://archive.ics.uci.edu/ml/>

<sup>2</sup> <http://www.mathworks.com/products/matlab/>

<sup>3</sup> versão: R2016a(9.0.0.341360); 64-bit (glnxa64)

É importante ressaltar a criação da tabela de correlação de atributos (Matriz de Atributos Importantes). Essa tabela é implementada conforme a técnica de correlação entre atributos com algoritmos supervisionados, seção 3.4.2, onde cada célula da tabela é preenchida através da execução de um algoritmo supervisionado. Estas execuções são realizadas em todos os atributos da cada cluster existente na base de dados.

Após a tabela preenchida, o atributo rótulo será selecionado a partir do maior valor em relação aos outros atributos do grupo, que é representado pela linha da tabela (matriz de atributos). Também poderá ser selecionado como rótulo os atributos que possuam o valor entre a diferença de  $V$  com o atributo de maior valor (mais relevante). Por exemplo, se o valor de  $V = 5\%$  e o atributo de maior valor é **95%**, então todos os atributos que possuírem o valor a partir de **90%** serão considerados rótulos também.

A cada base de dados descritas nas seções seguintes, são configuradas algumas variáveis, método de discretização e implementado dois algoritmos de aprendizado supervisionado com paradigmas diferentes para fazer rotulação. Cada algoritmo terá como resultado o rótulo por cluster de dados.

## 4.2 Seeds - Identificação de Tipos de Semente

Essa base pertence a UCI Machine Learning, composta por sete atributos definindo suas características e mais um atributo classe responsável por identificar os tipos de sementes. Em seus atributos seus valores são todos contínuos e não existem valores em branco, possuindo um total de 210 registros classificados em três categorias:

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

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 Polonês de Ciências em Lublin.

Como já mencionado neste capítulo, seção 4.1, antes de executar o algoritmo algumas configuração são necessárias. A primeira configuração é o método de discretização do tipo EFD, a segunda é a divisão dos valores dos atributos em faixas,  $R = 3$  para todos os atributos. Tanto a discretização como o valor do número de faixas foram alcançados através de testes para obter um melhor resultado.

Tabela 4 – Resultado da rotulação com o algoritmo Naive Bayes

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	area	] 12.78 ~ 16.14 ]	92%	14	80%
2	area	] 16.14 ~ 21.18 ]	95%	6	91,4%
3	perimetro	[ 12.41 ~ 13.73 ]	95%	5	92,8%

#### 4.2.1 Naive Bayes

Na tabela 4 é apresentado os resultados de rotulação do algoritmo Naive Bayes. Essa tabela é composta por colunas que informam os **Clusters**, **Rótulos** que integram **Atributo** e sua **Faixa** de valor, além da coluna **Relevância** exibida em porcentagem, bem como as colunas **Fora da Faixa** e **Acurácia Cluster** que mostra a quantidade de elementos que não estão dentro da faixa designada pelo do rótulo encontrado e a acurácia do cada cluster respectivamente.

A coluna **Relevância** demonstra o maior valor entre os atributos de cada cluster, e caso esses valores sejam ambíguos, serão exibidos na coluna todos estes atributos. Para ter maior clareza na escolha desses atributos foram inseridas as tabelas 5 que demonstram os valores de correlação entre eles.

Já na coluna, **Fora da Faixa**, tem a função de exibir, em números, a quantidade de valores que não estão participando da faixa definida pelo rótulo. Através de experimentos percebeu-se o mérito de apresentar em números a quantidade de elementos que não estão sendo representados pelo rótulo gerando mais realidade as informações, ao invés de exibir em porcentagem.

Na última coluna, **Acurácia Cluster**, apresenta em porcentagem o grau de acerto, por clusters, dos registros que são representados pelo rótulo. Foi possível expor estas informações visto que cada cluster já apresenta a quantidade e quais registros fazem parte de cada cluster.

Analizando a coluna Rótulos da tabela 4, nota-se que o atributo **area** aparece tanto no cluster 1 como também no cluster 2. A técnica envolve não só o atributo mais relevante, como também, a faixa que os valores mais se repetem dentro do atributo. Nesse caso pode-se observar que o atributo se repete entre os clusters, mas no cluster 1 a faixa de valores difere do cluster 2, sendo considerados rótulos distintos.

A tabela 5 é formada por clusters representado pelas linhas, e atributos representado por colunas. Essa tabela é fruto da aplicação do Naive Bayes na base de dados **Seeds**, e foi gerada para auxiliar a retirada dos(s) atributo(s) rótulo(s). Uma análise pode ser feita através desses dados e ajudar a definir um valor para a variável  $V$  caso necessário. Percebe-se que algumas características são mais bem correlacionadas que outras, através

Tabela 5 – Resultado da Correlação dos atributos pelo Naive Bayes; Legenda dos Atributos: (A)area, (B)perimetro, (C)compactness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove

		Atributos						
		A	B	C	D	E	F	G
Clusters	1	92.8	87.1	50.0	75.7	85.7	60.0	65.7
	2	95.7	91.4	47.1	92.8	90.0	28.5	85.7
	3	91.4	95.7	71.4	85.7	91.4	64.2	58.5

de seus valores mais altos. Isso indica o grau de relacionamento entre os atributos após a aplicação do algoritmo.

Tabela 6 – Resultado de 4(*quatro*) execuções do algoritmo Naive Bayes; Legenda dos Atributos: (A)area, (B)perimetro, (C)compactness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove

1a. Execução		Atributos						
		A	B	C	D	E	F	G
Clusters	1	92.8	87.1	48.5	77.1	82.8	57.1	65.7
	2	94.2	90.0	45.7	92.8	90.0	38.5	87.1
	3	91.4	95.7	72.8	85.7	91.4	64.2	60.0

2a. Execução		Atributos						
		A	B	C	D	E	F	G
Clusters	1	92.8	87.1	47.1	77.1	87.1	60.0	65.7
	2	94.2	90.0	47.1	92.8	91.4	32.8	87.1
	3	91.4	95.7	72.8	85.7	92.8	64.2	60.0

3a. Execução		Atributos						
		A	B	C	D	E	F	G
Clusters	1	94.2	85.7	48.5	77.1	82.8	61.4	65.7
	2	92.8	90.0	50.0	92.8	90.0	32.8	87.1
	3	91.4	95.7	72.8	85.7	92.8	64.2	60.0

4a. Execução		Atributos						
		A	B	C	D	E	F	G
Clusters	1	91.4	88.5	54.2	75.7	85.7	62.8	61.4
	2	95.7	90.0	50.0	92.8	90.0	38.5	85.7
	3	91.4	95.7	72.8	85.7	94.2	64.2	57.1

Para provar empiricamente os resultados, na tabela 6 é exposto o resultado de 4(*quatro*) execuções do Algoritmo Naive Bayes, e pode-se constatar que mesmo havendo algumas alterações em seus valores nos atributos em cada execução, a correlação entre os atributos não oferece muita alteração. Como exemplo, o atributo **area** nos clusters 1 e 2, possuem o melhor grau de correlacionamento em seus grupos, mesmo nas quatro execuções, como mostrado na tabela 6.

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

- $r_{c_1} = \{(area, [12.78 \sim 16.14])\}$
- $r_{c_2} = \{(area, [16.14 \sim 21.18])\}$
- $r_{c_3} = \{(perimetro, [12.41 \sim 13.73])\}$

## 4.2.2 CART

Já na tabela 7, do mesmo modelo da tabela 4, tem-se o resultado da aplicação do algoritmo supervisionado na rotulação. Ele é um algoritmo de classificação de árvore

de decisão utilizado pela toolbox do MATLAB. O intuito é testar a base de dados em diferentes paradigmas.

Tabela 7 – Resultado da aplicação do algoritmo CART

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	perimetro	[ 13.73 ~ 15.18 ]	94%	14	80%
2	area	] 16.14 ~ 21.18 ]	98%	6	90%
	perimetro	] 15.18 ~ 17.25 ]	98%	7	
3	wkernel	[ 2.63 ~ 3.049 ]	97%	9	87,1%

Foram realizadas vários teste, onde alguns desses testes estão na tabela 9. Essas operações foram execuções do algoritmo CART na base, para provar que a técnica de correlação de atributos, seção 3.3, é funcional para este algoritmo. O mesmo comportamento entre execuções pode ser visto no algoritmo de paradigma estatístico, subseção 4.2.1, realizado nessa pesquisa. O comportamento de ambos foram bem semelhantes, como também seus valores não se alteram muito a cada iteração.

Tabela 8 – Resultado da Correlação dos atributos pelo CART; Legenda dos Atributos: (A)area, (B)perimetro, (C)compactness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove

		Atributos						
		A	B	C	D	E	F	G
Clusters	1	91.4	94.2	58.5	80.0	81.4	61.4	61.4
	2	98.5	98.5	51.4	90.0	88.5	42.8	88.5
	3	92.7	95.7	80.0	88.5	97.1	58.5	78.5

Tabela 9 – Resultado de 4(*quatro*) iterações do algoritmo CART; Legenda dos Atributos: (A)area, (B)perimetro, (C)compactness, (D)Lkernel, (E)Wkernel, (F)asymetry, (G)lkgroove

1a. Execução	Atributos							
	A	B	C	D	E	F	G	
Clusters	1	91.4	94.2	58.5	80.0	74.2	55.7	60.0
	2	98.5	98.5	50.0	90.0	88.5	41.4	90.0
	3	92.8	95.7	80.0	88.5	97.1	55.7	77.1

2a. Execução	Atributos							
	A	B	C	D	E	F	G	
Clusters	1	91.4	94.2	62.8	78.5	81.4	61.4	57.1
	2	98.5	98.5	54.2	90.0	88.5	40.0	90.0
	3	92.8	95.7	80.0	88.5	97.1	60.0	77.1

3a. Execução	Atributos							
	A	B	C	D	E	F	G	
Clusters	1	93.8	93.6	61.8	83.2	89.2	53.2	71.0
	2	98.2	98.3	61.9	93.0	90.5	25.2	90.1
	3	95.5	96.3	82.4	90.9	97.7	59.3	77.0

4a. Execução	Atributos							
	A	B	C	D	E	F	G	
Clusters	1	92.8	94.2	60.0	80.0	84.2	64.2	60.0
	2	98.5	98.5	47.1	91.4	90.0	42.8	88.5
	3	91.4	95.7	80.0	88.5	97.1	55.7	77.1

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

- $r_{c_1} = \{(perimetro, ]13.73 \sim 15.18])\}$
- $r_{c_2} = \{(area, ]16.14 \sim 21.18]), (perimetro, ]15.18 \sim 17.25])\}$
- $r_{c_3} = \{(wkernel, [2.63 \sim 3.049])\}$

### 4.3 Iris - Identificação de Tipos de Plantas

A base de dados **Iris**, também pertencente a UCI Machine Learning, é muito conhecida em outras pesquisas [Lopes \(2014\)](#), [Filho \(2015\)](#) e outros, como também na literatura em reconhecimentos de padrões por utilizar classes de plantas bem definidas. Contêm 3 classes de 50 instâncias cada, totalizando 150 registros de amostra de plantas. O atributo classe classifica o tipo de planta em 3 tipos:

- 50 elementos da classe Iris-setosa ;
- 50 elementos da classe Iris-versicolour;
- 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. Através dessas características há uma classificação para dizer qual tipo de planta.

Foi aplicado na configuração de execução do algoritmo o método de discretização, tipo EFD<sup>4</sup>, a divisão de três faixas de valores  $R = 3$  para todos os atributos, e inserido o valor de variação  $V = 0\%$ . Mais uma vez, o valor  $V$  existe para evitar ambiguidade dos rótulos, podendo ser utilizado pelo pesquisador quando necessário após análise dos valores de correlação dos atributos nos grupos, tabela 11.

Seguindo a análise, semelhante da base de dados anterior, serão realizados testes utilizando dois algoritmos<sup>5</sup>, e cada resultado será exibido em tabelas. Portando as colunas são formadas por **Clusters**, **Rótulos**, **Relevância** e **Fora da Faixa** representando os elementos que não estão dentro da faixa escolhida como rótulo. Também foi posto nas tabelas 13 e 11 os resultados das correlações entre os atributos de cada grupo, servindo de informação para decisão do valor de  $V$ , caso fosse necessário. E também apresentado os resultados de outras iterações de cada algoritmo, para mostrar o comportamento dos atributos entre eles no grupo.

#### 4.3.1 Naive Bayes

Através da tabela 10 os resultados da rotulação são exibidos após a aplicação do algoritmo. Com essa base de dados nota-se que no cluster 1 houve um acerto de 100% da

<sup>4</sup> seção 2.2.2

<sup>5</sup> sessões 4.2.1, 4.2.2

rotulação. O cluster 2 e cluster 3 obtiveram rótulos distintos, cada um com grau de relevância acima de 80% em relação aos outros atributos de cada grupo.

Tabela 10 – Resultado da aplicação do algoritmo Naive Bayes

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	petallength	[ 1.0 ~ 3.7 ]	100%	0	100%
	petalwidth	[ 0.1 ~ 1.0 ]	100%	0	
2	petallength	] 3.7 ~ 5.1 ]	84%	7	86%
3	petalwidth	] 1.7 ~ 2.5 ]	90%	5	90%

A porcentagem representada na coluna de relevância não pode ser analisada isoladamente. Para isso a tabela 11 possui os valores de correlação de todos os atributos. Todos os números estão representados em porcentagem para melhor análise do grau de relacionamento entre os outros atributos.

Tabela 11 – Resultado (em %) de 4(*quatro*) execuções do algoritmo Naive Bayes; Legenda dos Atributos: (SL)sepalength,(SW)sepalwidth,(PL)petallength,(PW)petalwidth

1a. Execução		Atributos			
		SL	SW	PL	PW
Clusters	1	80	68	<b>100</b>	<b>100</b>
	2	72	76	<b>84</b>	<b>82</b>
	3	76	74	68	<b>90</b>

2a. Execução		Atributos			
		SL	SW	PL	PW
Clusters	1	80	68	100	100
	2	72	76	88	84
	3	70	74	70	90

3a. Execução		Atributos			
		SL	SW	PL	PW
Clusters	1	80	68	100	100
	2	72	74	84	84
	3	74	74	68	90

4a. Execução		Atributos			
		SL	SW	PL	PW
Clusters	1	80	68	100	100
	2	72	74	86	82
	3	70	74	70	92

Na tabela 11 foram inseridas quatro resultados de execuções do algoritmo. Foi escolhida na tabela 11 a 1a. execução para montar a tabela de rótulos, tabela 10. A partir dessas execuções o pesquisador poderá arbitrar sobre o valor de  $V$  para melhor adaptá-lo a base. Das várias execuções expostas na tabela 11, percebe-se que não há muita diferença entre os valores de cada execução. Isso mostra um padrão de valores de acordo com a base. No caso da 1a. execução os valores escolhidos como rótulo estão destacados em cada cluster.

Se na tabela 11 fosse escolhida a 3a. execução, os valores de rótulos seriam modificados, em virtude dos valores mais altos serem iguais, fazendo que o rótulo assumisse dois atributos: PL e PW. Em análise do cluster 2 percebe-se que os valores de PL e PW nas quatro execuções são bem próximos e até idênticos na terceira execução, como já dito anteriormente, então caso fosse necessário inserir um valor de variação  $V$ , um valor

aceitável seria  $V = 3$ . Desta maneira manteria os rótulos dos clusters 1 e 3 sem alteração, e um novo atributo seria incluído no cluster 2, assumindo o novo rótulo com dois atributos: PL e PW.

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

- $r_{c_1} = \{(petallength, [1.0 \sim 3.7]), (petalwidth, [0.1 \sim 1.0])\}$
- $r_{c_2} = \{(petallength, ]3.7 \sim 5.1])\}$
- $r_{c_3} = \{(petalwidth, ]1.7 \sim 2.5])\}$

### 4.3.2 CART

A aplicação do algoritmo CART na base de dados **Iris** gerou a tabela 12 como resultado, e ao examinar pode-se observar uma semelhança com a subseção anterior 4.3.1 onde foi aplicado o Naive Bayes.

Tabela 12 – Resultado da aplicação do algoritmo CART

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	petallength	[ 1.0 ~ 3.7 ]	100%	0	100%
	petalwidth	[ 0.1 ~ 1.0 ]	100%	0	
2	petalwidth	] 1.0 ~ 1.7 ]	90%	8	84%
3	petalwidth	] 1.7 ~ 2.5 ]	90%	5	90%

Ao observar a tabela 12 percebe-se que o resultado de rotulação no cluster 1 e 3 são idênticos ao do algoritmo apresentado anteriormente, mas no cluster 2 o rótulo é diferenciado pelo atributo petalwidth que atinge valores mais altos em todas as execuções, como mostra a tabela 13.

Tabela 13 – Resultado de 4(*quatro*) iterações do algoritmo CART; Legenda dos Atributos: (SL)sepalength,(SW)sepalwidth,(PL)petallength,(PW)petalwidth

1a. Execução		Atributos			
Clusters		SL	SW	PL	PW
	1	80	68	<b>100</b>	<b>100</b>
	2	74	76	<b>88</b>	<b>90</b>
	3	68	68	74	<b>90</b>

2a. Execução		Atributos			
Clusters		SL	SW	PL	PW
	1	80	68	100	100
	2	74	76	88	90
	3	70	70	74	90

3a. Execução		Atributos			
Clusters		SL	SW	PL	PW
	1	80	68	100	100
	2	74	76	86	90
	3	70	66	78	90

4a. Execução		Atributos			
Clusters		SL	SW	PL	PW
	1	80	68	100	100
	2	72	74	86	90
	3	68	66	78	90

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



- $r_{c_1} = \{(petallength, [1.0 \sim 3.7]), (petalwidth, [0.1 \sim 1.0])\}$
- $r_{c_2} = \{(petalwidth, ]1.0 \sim 1.7])\}$
- $r_{c_3} = \{(petalwidth, ]1.7 \sim 2.5])\}$

## 4.4 Glass - Identificação de Tipos de Vidros

Essa base ficou conhecida por Vina Spiehler, Ph.D. da DABFT Diagnostic Products Corporation, onde 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 vidros, porque em uma cena de crime, uma classificação de tipos de vidro corretamente identificada pode ser utilizada como prova, ajudando diretamente na investigação.

Possui um total de 214 instancias, caracterizados por 9 atributos (RI, Na, Mg, Al, Si, K, Ca, Ba e Fe), sendo que o atributo **RI** indica o índice de refração, e quanto aos demais atributos são valores correspondentes a porcentagem do óxido.

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

- 1 janelas de construção - vidro temperado: 70 registros
- 2 janelas de construção - vidro não-temperado: 76 registros
- 3 janelas de veículos - vidro temperado: 17 registros
- 4 janelas de veículos - vidro não-temperado: 0 registro
- 5 recipientes: 13 registros
- 6 louças de mesa: 9 registros
- 7 lâmpadas: 29 registros

Para execução dos algoritmos foram definidos a quantidade de faixas ( $R$ ) que serão divididos os valores dos atributos, qual o método de discretização e o valor de variação  $V$  caso haja ambiguidade. Nos teste desenvolvidos nesta pesquisa os valores de referência foram,  $R = 3$  para o número de faixas, o método de discretização EWD e o valor  $V = 0$ .

### 4.4.1 Naive Bayes

Ao observar a tabela 14 percebe-se que a coluna **Relevância** obteve porcentagens altas, ressaltando nos rótulos de cada grupo os atributos que mais bem se relacionaram. E em específico no **cluster 5** atributo **Na**, o valor da coluna de **Relevância = 100%**, mas na

coluna, **Fora da Faixa**, apresentam 2(dois) elementos que não estão sendo representados pelo rótulo.

Essa situação dita no parágrafo acima segue a Definição 3, mas é um exemplo prático que não aconteceu em outros testes das outras bases de dados. A definição é que cada rótulo específico é dado por um conjunto de pares de valores, tendo como saída um vetor com atributo e seu respectivo intervalo,  $r_{ci} = \{(a_1, [p_1, q_1]), \dots, (a_{m(c_i)}, [p_{m(c_i)}, q_{m(c_i)}])\}$  capaz de melhor expressar o cluster  $c_i$ . Então caso a coluna **Relevância** seja igual a 100%, isso não implica que todos os elementos tenham que estar dentro da faixa  $p_{m(c_i)}$  (limite inferior) e  $q_{m(c_i)}$  (limite superior), e sim, a maioria dos elementos, mostrando que o rótulo é capaz de melhor representar o cluster.

Além de apresentar dados desbalanceados o **Cluster 5** apresentado na tabela 14 conta com o total de nove elementos, e entre estes, nenhum participa da 1a. faixa, dois estão na 2a. faixa e os restantes (sete) estão na 3a. faixa. Dessa maneira justifica-se o porquê dos dois elementos estarem de fora do rótulo, pois a faixa rótulo escolhida é a 3a. faixa, onde contém a maioria dos elementos, por conseguinte, representada pelo rótulo.

Tabela 14 – Resultado da aplicação do algoritmo Naive Bayes

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	Mg	[ 2.245 ~ 4.490 ]	100%	0	100%
	K	[ 0.0 ~ 1.5525 ]	100%	0	
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
2	K	] 0.0 ~ 1.5525 ]	100%	0	100%
3	Mg	] 2.245 ~ 4.490 ]	100%	0	100%
	K	] 0.0 ~ 1.5525 ]	100%	0	
	Ca	] 8.12 ~ 10.81 ]	100%	0	
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
4	Al	[ 1.0925 ~ 1.895 ]	92%	4	69,2%
	K	[ 0.0 ~ 1.5525 ]	92%	3	
	Ba	[ 0.0 ~ 0.7875 ]	92%	1	
5	Na	[14.055 ~ 17.38 ]	100%	2	77,7%
	K	[ 0.0 ~ 1.5525 ]	100%	0	
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
	Fe	[ 0.0 ~ 0.1275 ]	100%	0	
6	Fe	[ 0.0 ~ 0.1275 ]	100%	0	100%

Os resultados da tabela 15, assim como nos resultados de bases anteriores, indica uma sequência de execuções onde é possível observar o comportamento das variáveis que são escolhidas como rótulo. Nestes exemplos fica claro que não foi necessária a utilização de uma variação  $V$  para a escolha dos rótulos, logo porque não houve ambiguidade entre eles. Por outro lado, quando testes utilizaram outro método de discretização, EFD, retornaram rótulos ambíguos obrigando o uso da variação  $V$ . Em consequência disto foi definindo o

método de discretização EWD como padrão para a rotulação de dados.

Tabela 15 – Resultado de 4(*quatro*) execuções do algoritmo Naive Bayes.

1a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	87.1	92.8	100	81.4	82.8	100	90.0	100	78.5
	2	65.7	86.8	85.5	82.8	56.5	100	73.6	98.6	61.8
	3	82.3	82.3	100	76.4	58.8	100	100	100	82.3
	4	84.6	69.2	30.76	92.3	76.9	92.3	76.9	92.3	69.2
	5	77.7	100	33.3	66.6	44.4	100	55.5	100	100
	6	58.6	79.3	79.3	72.4	79.3	93.1	93.1	13.7	100

2a. Exec	Atributos									
	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	
Clusters	1	87.1	92.8	100	81.4	81.4	100	90.0	100	78.5
	2	65.7	92.1	88.1	82.8	63.1	100	72.3	97.3	61.8
	3	72.4	82.3	100	76.4	47	100	100	100	82.3
	4	84.6	69.2	23	92.3	76.9	92.3	76.9	92.3	61.5
	5	77.7	100	33.3	66.6	44.4	100	55.5	100	100
	6	58.6	79.3	79.3	68.9	79.3	93.1	93.1	17.2	100

3a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	87.1	92.8	100	81.4	84.2	100	90.0	100	78.5
	2	68.4	89.4	86.8	84.2	60.5	100	72.3	98.6	64.4
	3	76.4	82.3	100	76.4	52.9	100	100	100	82.3
	4	84.6	69.2	23	92.3	76.9	92.3	76.9	92.3	76.9
	5	77.7	100	33.3	66.6	44.4	100	55.5	100	100
	6	58.6	79.3	79.3	68.9	79.3	93.1	89.6	13.7	100

4a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	87.1	92.8	100	81.4	84.2	100	90.0	100	78.5
	2	65.7	90.7	86.8	82.8	59.2	100	76.3	98.6	63.1
	3	76.4	82.3	100	76.4	52.4	100	100	100	82.3
	4	84.6	53.8	23	92.3	76.9	92.3	76.9	92.3	69.2
	5	77.7	100	33.3	66.6	44.4	100	55.5	100	100
	6	58.6	82.7	79.3	72.4	79.3	93.1	82.7	6.8	100

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

- $r_{c_1} = \{(Mg, [2.245 \sim 4.490]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875])\}$
- $r_{c_2} = \{(K, [0.0 \sim 1.5525])\}$
- $r_{c_3} = \{(Mg, [2.245 \sim 4.490]), (K, [0.0 \sim 1.5525]), (Ca, [8.12 \sim 10.81]), (Ba, [0.0 \sim 0.7875])\}$
- $r_{c_4} = \{(Al, [1.0925 \sim 1.895]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875])\}$
- $r_{c_5} = \{(Na, [14.055 \sim 17.380]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_6} = \{(Fe, [0.0 \sim 0.1275])\}$

#### 4.4.2 CART

Ao utilizar o algoritmo CART logo percebe-se a semelhança com os resultados apresentados na subseção 4.4.1. Apesar dessa semelhança os **Clusters 4 e 5** tiveram diferenças nos resultados em comparação ao algoritmo Naive Bayes.

Ao verificar a **linha 4** da tabela 15, correspondente ao **Cluster 4**, os atributos **Al**, **K**, **Ba** apresentaram sempre os mesmos valores, mas já na tabela 17, também na **linha 4** de cada execução, só o valor de **Ba** coincide já os outros atributos tiveram valores mais baixos, fazendo com que eles não participassem da composição do rótulo.

No **Cluster 5** o atributo **Na** não faz parte do rótulo, e diferente do Naive Bayes, vista suas execuções na tabela 15, verifica-se que os valores de **Na** são sempre 100% de correlação entre os outros atributos, no CART, os valores apresentados de **Na** nas execuções

da tabela 17, linha 5, são abaixo dos 78%. Na 1a. Exec da tabela 17 os atributos que compõem o rótulo do **Cluster 5** apresentam também 100%, portanto qualquer atributo com valor abaixo de 100% não será escolhido para compor o rótulo.

Tabela 16 – Resultado da aplicação do algoritmo CART

Cluster	Rótulos		Relevância(%)	Fora da Faixa	Acurácia Cluster(%)
	Atributos	Faixa			
1	Mg	[ 2.245 ~ 4.490 ]	100%	0	100%
	K	[ 0.0 ~ 1.5525 ]	100%	0	
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
2	K	] 0.0 ~ 1.5525 ]	100%	0	100%
3	Mg	] 2.245 ~ 4.490 ]	100%	0	100%
	K	] 0.0 ~ 1.5525 ]	100%	0	
	Ca	] 8.12 ~ 10.81 ]	100%	0	
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
4	Ba	[ 0.0 ~ 0.7875 ]	92%	1	92,3%
5	K	[ 0.0 ~ 1.5525 ]	100%	0	100%
	Ba	[ 0.0 ~ 0.7875 ]	100%	0	
	Fe	[ 0.0 ~ 0.1275 ]	100%	0	
6	Fe	[ 0.0 ~ 0.1275 ]	100%	0	100%

Tabela 17 – Resultado de 4(*quatro*) execuções do algoritmo CART.

1a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	88.5	90.0	100	92.8	84.2	100	92.8	100	75.7
	2	72.3	82.8	94.7	82.8	71.0	100	77.6	98.6	68.4
	3	76.4	70.5	100	47.0	76.4	100	100	100	76.4
	4	69.2	84.6	76.9	61.5	69.2	76.9	76.9	92.3	84.6
	5	77.7	77.7	44.4	66.6	66.6	100	55.5	100	100
	6	72.4	75.8	68.9	72.4	75.8	86.2	86.2	51.7	100

2a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	85.7	87.1	100	92.8	84.2	100	92.8	100	74.2
	2	76.3	86.8	96.0	82.8	64.4	100	76.3	98.6	68.4
	3	76.4	82.3	100	47.0	76.4	100	100	100	76.4
	4	76.9	84.6	76.9	69.2	69.2	76.9	76.9	92.3	84.6
	5	77.7	77.7	44.4	66.6	66.6	100	55.5	100	100
	6	72.4	75.8	65.5	72.4	75.8	93.1	93.1	51.7	100

3a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	88.5	85.7	100	92.8	84.2	100	92.8	100	75.7
	2	71.1	80.2	94.7	78.9	68.4	100	78.9	98.6	65.7
	3	76.4	82.3	100	58.8	76.4	100	100	100	82.3
	4	76.9	84.6	76.9	61.5	69.2	76.9	76.9	92.3	84.6
	5	77.7	77.7	44.4	66.6	66.6	100	55.5	100	100
	6	72.4	68.9	65.9	68.9	75.8	89.6	93.1	55.1	100

4a. Exec		Atributos								
		RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Clusters	1	88.7	87.1	100	92.8	84.2	100	92.8	100	75.7
	2	78.9	84.2	94.7	81.5	69.7	100	76.3	98.6	65.7
	3	76.4	82.3	100	64.7	76.4	100	100	100	76.4
	4	76.9	84.6	61.5	69.2	69.2	76.9	76.9	92.3	84.6
	5	77.7	77.7	44.4	66.6	66.6	100	55.5	100	100
	6	72.4	68.9	68.9	68.9	75.8	93.1	93.1	51.7	100

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

- $r_{c_1} = \{(Mg, [2.245 \sim 4.490]), (K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875])\}$
- $r_{c_2} = \{(K, [0.0 \sim 1.5525])\}$

- $r_{c_3} = \{(Mg, [2.245 \sim 4.490]), (K, [0.0 \sim 1.5525]), (Ca, [8.12 \sim 10.81]), (Ba, [0.0 \sim 0.7875])\}$
- $r_{c_4} = \{(Ba, [0.0 \sim 0.7875])\}$
- $r_{c_5} = \{((K, [0.0 \sim 1.5525]), (Ba, [0.0 \sim 0.7875]), (Fe, [0.0 \sim 0.1275])\}$
- $r_{c_6} = \{(Fe, [0.0 \sim 0.1275])\}$

## 5 Conclusões, Trabalhos Futuros e Cronograma

Este capítulo abordará as conclusões dessa proposta de mestrado referentes aos resultados do capítulo 4, bem como uma seção de Trabalhos Futuros e Cronograma. Na seção de Conclusão serão feitas considerações finais dos resultados de cada base de dados apresentadas, e logo após, em Trabalhos Futuros tem a pretensão de melhorar e expandir tudo que fora realizado nesta pesquisa, e expor que existe uma continuidade para todo esse estudo aqui elaborado. Já no Cronograma, será criada uma tabela temporal onde esta será dividida em meses e tarefas definindo os passos a serem seguidos até a conclusão da dissertação.

### 5.1 Conclusão

No capítulo 4 foi aplicado algoritmos supervisionados em algumas bases de dados a fim de provar se o problema proposto por este trabalho foi solucionado. Uma vez conhecido o problema, foi executado dois algoritmos supervisionados servindo de amostra para provar que era possível fazer rotulação de dados com estes algoritmos (Naive Bayes e CART), tema deste trabalho. E já identificando alguns trabalhos que já haviam feito rotulação, como [Lopes \(2014\)](#), utilizando algoritmos supervisionados, este estudo demonstra de forma empírica a execução de outros algoritmos com paradigmas diferentes, aos que já foram realizados em pesquisas anteriores.

Como o cerne da pesquisa é a rotulação de dados foi apresentados dois algoritmos com paradigmas diferentes, e em ambos, suas execuções nas bases de dados resultaram em respostas satisfatórias no âmbito da rotulação. Embora os rótulos encontrados em cada base de dados não tenham sido totalmente idênticos, tanto um algoritmo como outro mostraram semelhanças em vários rótulos gerados, como exemplo das bases IRIS e GLASS.

O processo de rotulação é composto por um, ou vários atributos, de maior relevância entre eles junto com sua(s) faixa(s) de valor(es) que mais se repetem, conteúdo já visto na subseção 3.4.3. Seguindo esse modelo foram adicionadas a cada resultado tabelas mostrando em porcentagem o grau de correlacionamento entre os atributos. A importância desta informação é passar o comportamento destes atributos adquirindo uma ideia geral da correlação entre eles na escolha do atributo rótulo.

No modelo de resolução proposto foi inicialmente utilizado na base de dados Seeds, seção 4.2.1, o algoritmo Naive Bayes. O atributo escolhido em cada cluster para compor o

rótulo foi o que obteve na tabela 6 o maior valor de porcentagem. Na base de dados IRIS os valores escolhidos como rótulo dos clusters não ocorreram ambiguidades, portanto não houve necessidade de utilizar a variável *V*. Todos os três clusters somente tiveram um atributo cada, compondo o rótulo.

Continuando com a base Seeds, após a escolha do atributo que fará parte do rótulo, o segundo passo é a escolha da faixa de valores do atributo. Essa segunda etapa é dependente totalmente da discretização<sup>1</sup> e independente da primeira etapa. O método é capaz de gerar a faixa de maior repetição de valores de qualquer atributo, mas aqui neste trabalho o que importa é a faixa do atributo rótulo. Para ter mais confiabilidade no rótulo o método escolhe a faixa de valores que mais se repetem. No caso desse algoritmo o resultado na tabela 4 consegue provar uma boa eficiência, pois em cada 70 elementos do cluster 1, somente 14, ficaram de fora dessa faixa. No cluster 2, somando os dois atributos rótulos tem-se 12 elementos que não estão dentro da representatividade do rótulo. Outro valor pequeno em relação aos 70 elementos. E no cluster 3, somente 5 elementos não estão dentro da faixa considerada rótulo.

No cenário da execução do algoritmo CART, os resultados foram diferentes dos apresentados pelo Naive Bayes, mas nem por isso foram insatisfatórios. Contudo uma breve análise sobre as execuções das tabelas 6 e 9 podem ser observadas nos gráficos da figura 12. Como já comentado anteriormente o comportamento dos valores do correlacionamento dos atributos ao longo das execuções mostra-se equilibrada, figura 12b. O gráfico do CART tem um movimento semelhante ao do aplicado do Naive Bayes (figura 12a), embora a variável **asymetry** saia um pouco do padrão, mas como seus valores são baixos, nada alterou nos rótulos, contudo o valor de **perimetro** ficou bastante encostado ao valor da **area**, fazendo o rótulo **perimetro** aparecer nos grupos 1 e 2. E também só não foi escolhido pelo grupo 3, pois a variável **Wkernel** estava com valor mais alto. E no gráfico percebe-se que **Wkernel** mantém valores altos em todas as execuções do grupo 3.

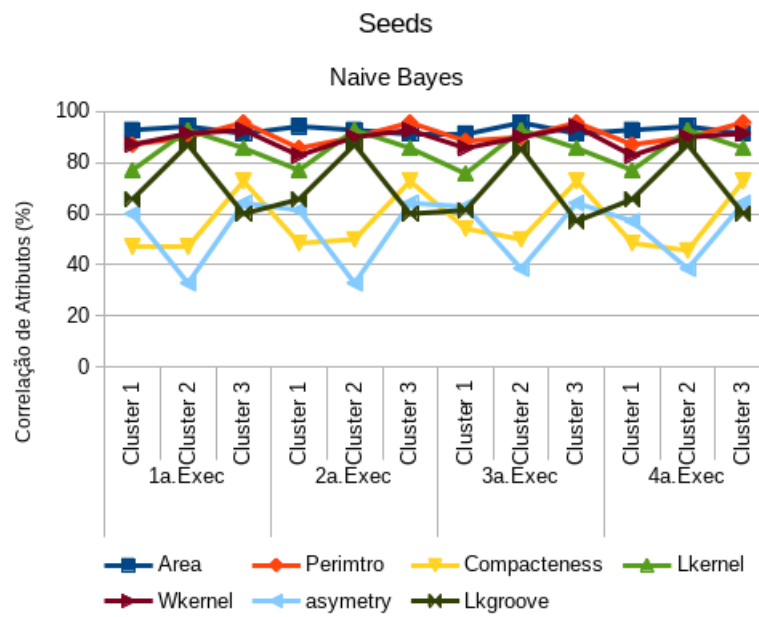
De acordo com o exposto no parágrafo anterior pode-se dizer nesta análise, que o Naive Bayes acabou tendo resultados um pouco melhores, pois no que diz respeito ao número de elementos fora da faixa definida pelo rótulo, o CART, acabou por ter mais elementos fora da faixa de rótulo comparado aos resultados do Naive Bayes. Isso implica dizer que o rótulo deixa de representar mais elementos usando o CART ao invés do Naive Bayes, melhor dizendo, o Naive Bayes representou mais elementos que o CART.

Já na base de dados IRIS, os dois algoritmos supervisionados testados apresentaram os mesmos rótulos nos clusters 1 e 3. Nos gráficos da figura 13 pode-se acompanhar como os valores dos atributos se comportam em seus clusters nas quatro execuções.

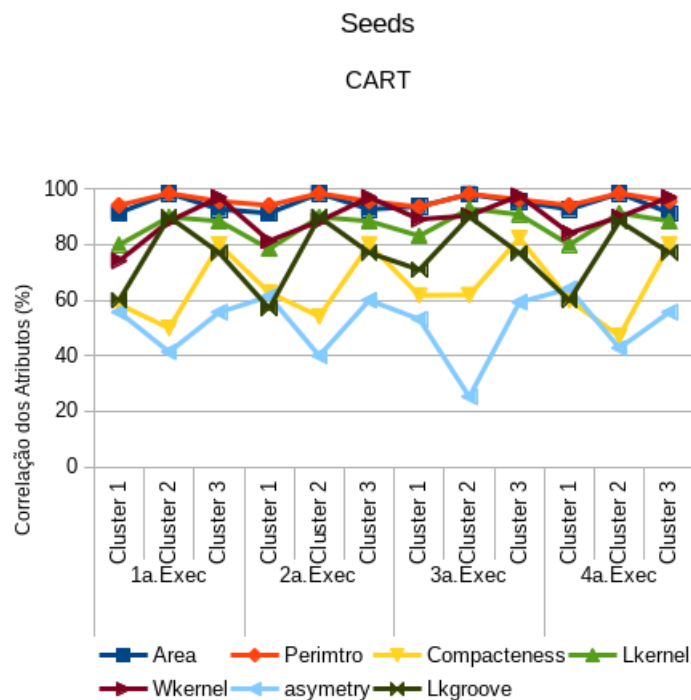
Os algoritmos aplicados na base IRIS tem resultados nos gráficos bastantes seme-

---

<sup>1</sup> seção 2.2



(a) Naive Bayes



(b) CART

Figura 12 – Gráfico de Execuções dos algoritmos supervisionados na base de dados SEEDS.

lhantes ao da base SEEDS, e logo percebe-se que a base IRIS contém características que possuem mais atributos bem correlacionados em relação ao da base SEEDS, pois nenhum atributo possui valor abaixo da linha 65(%) de relacionamento entre eles. Embora no gráfico as linhas referentes aos comportamentos dos atributos nos clusters 1 e 3 não sejam totalmente iguais em cada figura (13a e 13b), não modificou o resultado dos rótulos como



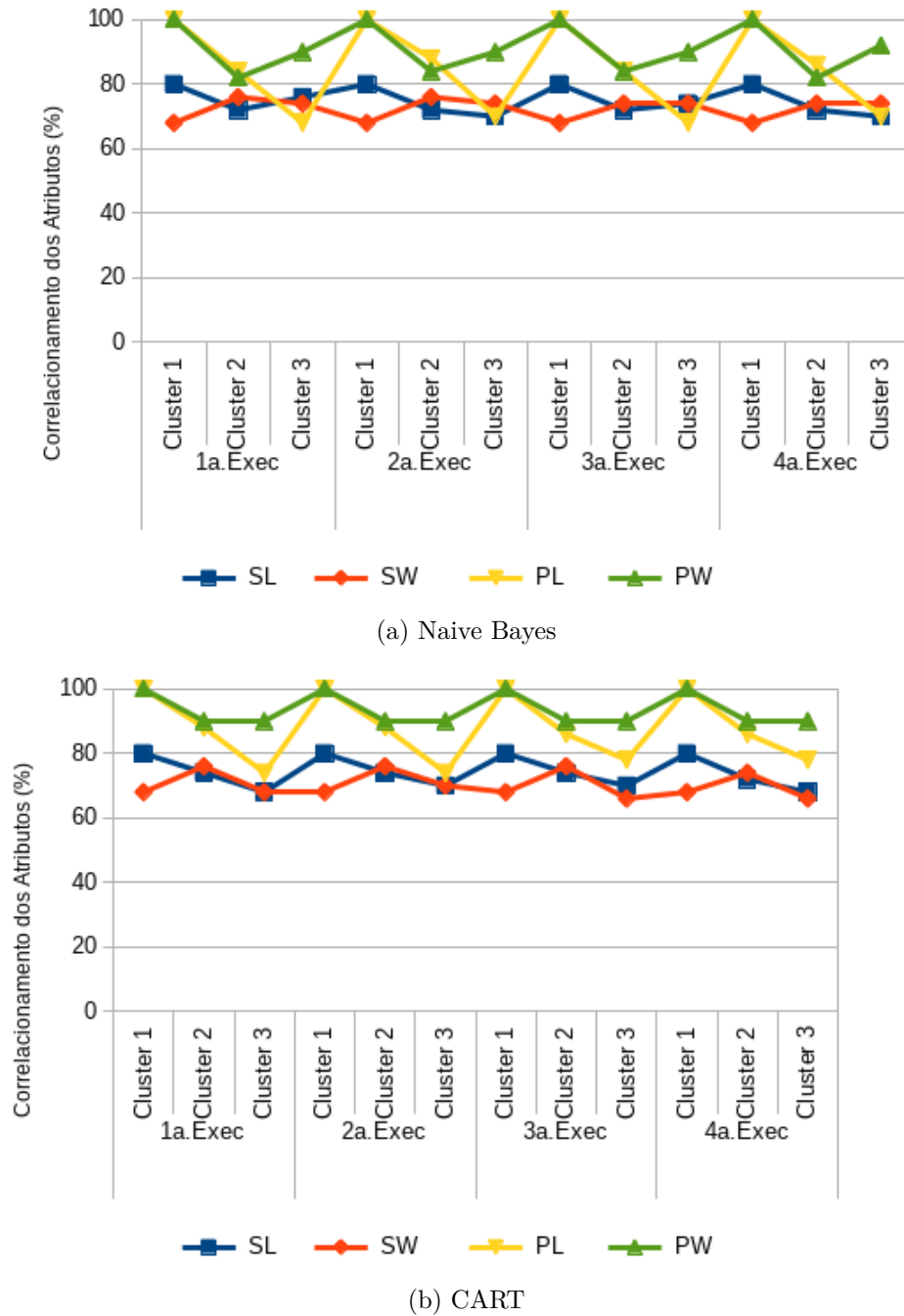
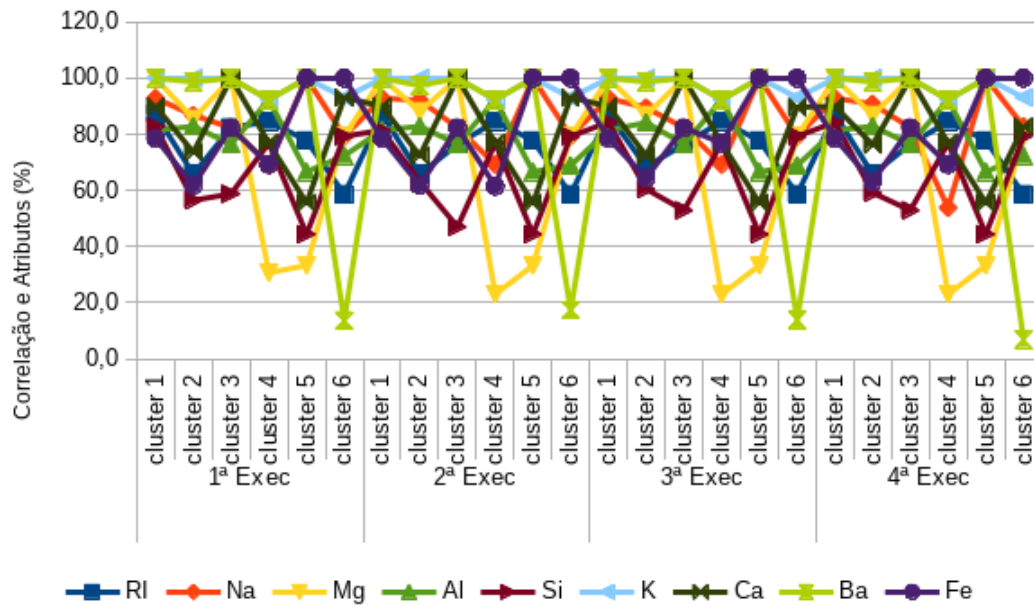


Figura 13 – Gráfico de Execuções dos algoritmos supervisionados na base de dados IRIS.

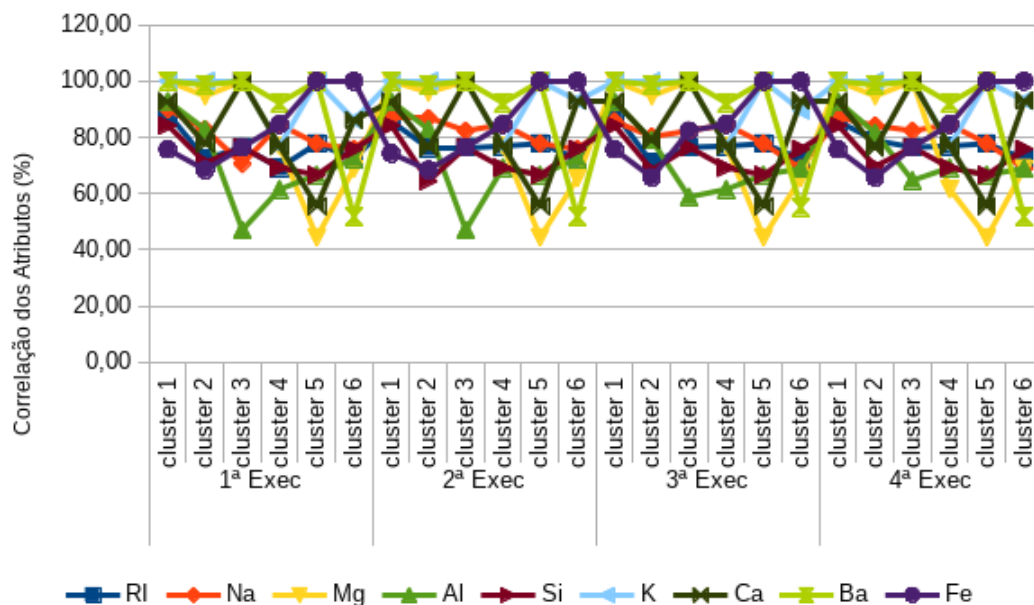
resposta.

Conforme resultados das tabelas 10 e 12 apresentadas pela execução dos dois algoritmos os rótulos escolhidos no cluster 1 foram dois atributos: **petalwidth** e **petal-length**. Onde cada um deles definiram faixas de valor que foi possível abranger 100% dos elementos. Já no cluster 2 cada algoritmo teve um atributo rótulo diferente, e embora não tivesse a mesma acurácia do cluster 1, obteve um total, de 86% de acurácia e deixando de representar 7 elementos do rótulo **petal-length** pelo Naive Bayes, e 84% de acurácia deixando de representar 8 elementos do rótulo **petalwidth** com CART. E no cluster 3

o atributo escolhido para compor o rótulo foi o **petalwidth** em ambos os algoritmos. Logo percebe-se a importância do atributo rótulo no cluster 3, pois o rótulo representa 45 elementos no total de 50 dentro do cluster, deixando somente 5 elementos fora dessa faixa representada pelo rótulo.



(a) Naive Bayes



(b) CART

Figura 14 – Gráfico de Execuções dos algoritmos supervisionados na base de dados GLASS.

A avaliação da base de dados GLASS referente a rotulação apresentada na tabela 14 do Naive Bayes, não foi tão bem sucedida quanto ao CART. Dos seis clusters definidos na rotulação somente dois deles não tiveram 100% de acurácia, e dentre esses dois clusters foi onde obtiveram os mais baixos valores de acurácia.

No cluster 4 em comparação aos dois algoritmos testados, houve uma diferença no Naive Bayes por apresentar três atributos compondo o rótulo, como pode ser visto na tabela 15, diferente do algoritmo CART que apresentou somente um atributo - **Ba** - também presente no rótulo do Naive Bayes. No CART por aprensetar somente um atributo com sua faixa, acabou mitigando o erro e obtendo uma maior porcentagem de acurácia.

Nos gráficos da figura 14 é apresentado o comportamento de correlacionamento dos atributos rótulos do Naive Bayes e CART respectivamente, e mesmo havendo semelhança nos gráficos os valores de correlação dos atributos no CART foram melhores, e por conseguinte teve melhor acurácia comprovado no gráfico 15.

Por fim, ao se analisar os resultados nos dois algoritmos supervisionados pode-se afirmar que foram bem satisfatórios nas bases utilizadas, conforme figura 15, que mostra uma acurácia de 80% na maioria dos resultados e desta maneira provando que é possível a rotulação de dados com Naive Bayes e CART, portanto os rótulos encontrados representam bem os clusters testados.

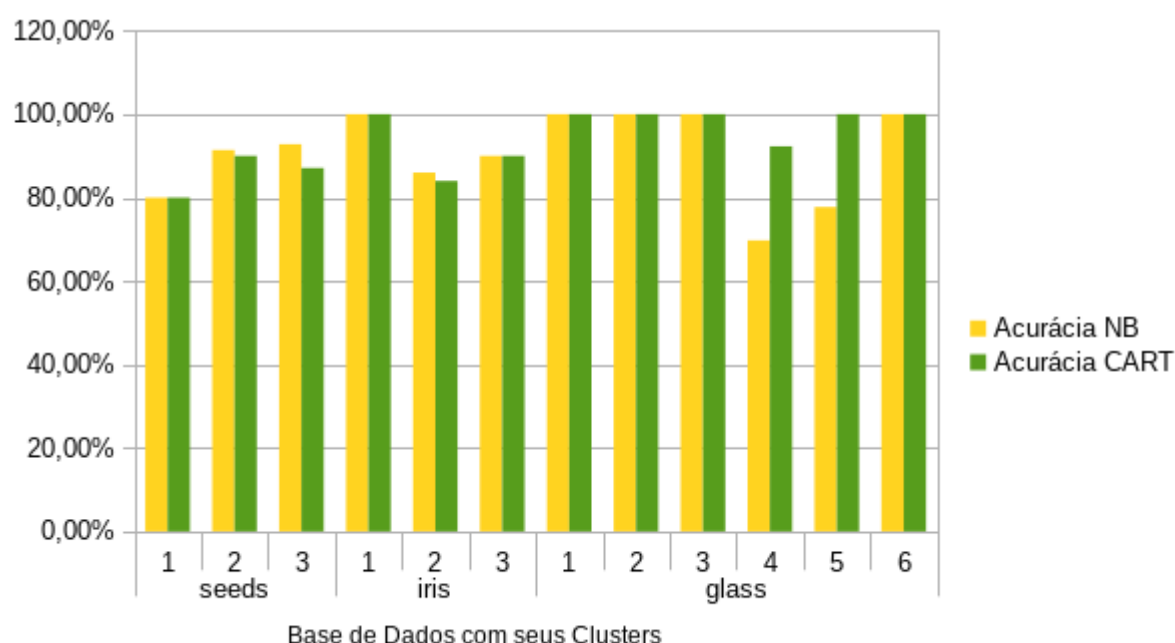


Figura 15 – Acurácia por Clusters (Os clusters estão numerados em ordem crescente em cada Base de Dados)

## 5.2 Trabalhos Futuros

A pesquisa ainda precisa de mais divulgação na esfera acadêmica, e para isso a publicação de um artigo sobre os resultados apresentados aqui é uma consolidação dessa proposta de mestrado já voltada para a dissertação propriamente dita.

Fazer testes com mais bases de dados provando que esse método pode ser utilizado em várias bases com características diferentes.

Outro ponto importante é inserir nos teste mais algoritmos, que pertençam a paradigmas diferentes dos que já foram utilizados.

### 5.3 Cronograma

Tabela 18 – Cronograma de atividades

Atividades	Meses				
	Maio	Junho	Julho	Agosto	Setembro
Testes com Novas Bases de Dados					
Modificar Números de Faixa (R)					
Testar com outros Métodos de Discretização					
Testar com outros Algoritmos com Paradigmas Diferentes					
Preparar Artigo					
Escrita da Dissertação					

# Referências

- 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 6 e 10.
- 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 8.
- CATLETT, J. *On changing continuous attributes into ordered discrete attributes*. Springer, Berlin, Heidelberg: Springer Verlag, 1991. 164–178 p. Citado 2 vezes nas páginas 11 e 20.
- 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 11.
- FILHO, V. P. R. Dissertação (Programa de Pós-graduação em Ciência da Computação), *Rotulacao de grupos utilizando conjuntos fuzzy*. Teresina: [s.n.], 2015. 25 p. Citado 2 vezes nas páginas 14 e 30.
- 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 11 e 20.
- 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 11.
- 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.
- 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 3 vezes nas páginas 15, 13 e 14.
- LOPES, L. A. Dissertação (Programa de Pós-graduação em Ciência da Computação), *Rotulação Automática de Grupos com Aprendizagem de Máquina Supervisionada*. Teresina: [s.n.], 2014. 73 p. Citado 11 vezes nas páginas 15, 2, 12, 13, 14, 16, 17, 18, 21, 30 e 38.
- 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 10.

MADUREIRA, D. F. *Análise de sentimento para textos curtos*. Tese (Doutorado) — Fundacao Getulio Vargas, Rio de Janeiro, 2017. Citado na página 10.

MCCALLUM, A.; NIGAM, K. A Comparison of Event Models for Naive Bayes Text Classification. 1997. Citado na página 10.

MITCHELL, T. M. *Machine learning*. [S.l.]: McGraw-Hill Science/Engineering/Math, 1997. 432 p. ISSN 10450823. ISBN 9781577354260. Citado 2 vezes nas páginas 5 e 9.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations Machine Learning*. [S.l.: s.n.], 2012. ISBN 9780262018258. Citado na página 6.

MONTGOMERY, K. *Big Data Now*. 1. ed. [S.l.]: O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2013. v. 53. 1689–1699 p. ISSN 1098-6596. ISBN 9788578110796. Citado na página 1.

RAIMUNDO, L. R.; MATTOS, M. C. D.; WALESKA, P. O Algoritmo de Classificação CART em uma Ferramenta de Data Mining. 2008. Citado na página 8.

RUSSEL, S.; NORVIG, P. *Inteligência Artificial*. 3ª. ed. Rio de Janeiro: [s.n.], 2013. ISBN 9780136042594. Citado 3 vezes nas páginas 6, 8 e 9.

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 10.

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 8.