]MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e dos Materiais
PPGE3M

EMPREGO DE DIFERENTES ALGORITMOS DE ÁRVORES DE DECISÃO NA CLASSIFICAÇÃO DA ATIVIDADE CELULAR IN VITRO PARA TRATAMENTOS DE SUPERFICIES EM TITÂNIO

Fabiano Rodrigues Fernandes

Dissertação para obtenção do título de Mestre em Engenharia

Porto Alegre
2017

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e dos Materiais
PPGE3M

EMPREGO DE DIFERENTES ALGORITMOS DE ÁRVORES DE DECISÃO NA CLASSIFICAÇÃO DA ATIVIDADE CELULAR IN VITRO PARA TRATAMENTOS DE SUPERFICIES EM TITÂNIO

Fabiano Rodrigues Fernandes

Dissertação para obtenção do título de Mestre em Engenharia

Trabalho realizado no Centro de Tecnologia da Escola de Engenharia da UFRGS, dentro do Programa de Pós – Graduação em Engenharia de Minas, Metalúrgica e de Materiais – PPGE3M, como parte dos requisitos para a obtenção do título de Mestre em Engenharia.

Área de Concentração: Processos de Fabricação

Porto Alegre
2017

Esta Dissertação foi julgada adequada para obtenção do título de Mestre em Engenharia, área de concentração Processos de Fabricação e aprovada em sua forma final, pelo Orientador e pela Banca Examinadora do Curso de Pós-Graduação.

Orientador: Prof. Dr. Alexandre da Silva Rocha
Coorientadora: Profa. Dra. Célia de Fraga Malfatti

Banca Examinadora:

Prof. Dr. Jovani Castelan (Faculdade SATC – SC)

Prof. Dr. Leandro Krug Wives (UFRGS – RS)

Dr. Leonardo Marasca Antonini (UFRGS – RS)

Prof. Dr. Tiago Lemos Menezes (UFRGS – RS)

Prof. Dr. Carlos Pérez Bergmann
Coordenador do PPGE3M

Dedico à minha família, esposa Monara e meu filho Kenzo, por todo amor, paciência e compreensão das seguidas ausências em detrimento desse trabalho.

Principalmente aos meus pais por todo amor e disciplina dispensados, pelas inúmeras cobranças já na infância mostrando a importância dos estudos.

AGRADECIMENTOS

Primeiramente a Deus por permitir a conclusão de mais esta etapa da minha vida.

Ao meu orientador, Prof. Dr. Alexandre da Silva Rocha e a minha coorientadora, Profª Dra. Célia de Fraga Malfatti pelas valiosas informações técnicas e metodológicas repassadas, meu sincero agradecimento.

Ao Prof. Dr. Leandro Krug Wives, pela colaboração e revisão, com observações importantes ao trabalho.

À todos os colegas da SATC que colaboraram de forma direta ou indiretamente na elaboração deste trabalho, o meu reconhecimento.

À todos os colegas da UFRGS que colaboraram de forma direta ou indiretamente na elaboração deste trabalho, o meu reconhecimento.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

Ti	Titânio
TiO_2	Dióxido de titânio
TN	Nanotubos de titânio
AuNP	Nanoparticulas de ouro
N	Nitrogênio
Fe	Ferro
O	Oxigênio
C	Carbono
H	Hidrogênio
MTT	3-(4,5-dimetiltiazol-2-il)-2,5-difenil tetrazólio bromide
Ticp	Titânio comercialmente puro
KDD	Knowledge Discovery in Databases
CART	Classification and Regression Trees
CHAID	CHi-square Automatic Interaction Detector
ICAAC	Índice de classificação do aumento da atividade celular

LISTA DE SÍMBOLOS

µm	Micrométrica
Ra	Rugosidade média
Θ	Ângulo teta
γ_s	Energia de superfície do solido
γ_{lv}	Tensão superficial do líquido em Equilíbrio com o Valor
γ_{sl}	Energia da interface sólido-líquido
°	Ângulo
nm	Nanométrica
N	Conjunto de exemplos
n	Número e valor nós dos filhos
v_j	Número de exemplos associados ao nodo-filho
c	Número de classes
p	Fração de registros pertencentes a classe
MPa	Megapascal

RESUMO

O interesse pela área de análise e caracterização de materiais biomédicos cresce, devido a necessidade de selecionar de forma adequada, o material a ser utilizado. Dependendo das condições em que o material será submetido, a caracterização poderá abranger a avaliação de propriedades mecânicas, elétricas, bioatividade, imunogenicidade, eletrônicas, magnéticas, ópticas, químicas e térmicas. A literatura relata o emprego da técnica de árvores de decisão, utilizando os algoritmos SimpleCart(CART) e J48, para classificação de base de dados (dataset), gerada a partir de resultados de artigos científicos. Esse estudo foi realizado afim de identificar características superficiais que otimizassem a atividade celular. Para isso, avaliou-se, a partir de artigos publicados, o efeito de tratamento de superfície do titânio na atividade celular in vitro (células MC3TE-E1). Ficou constatado que, o emprego do algoritmo SimpleCart proporcionou uma melhor resposta em relação ao algoritmo J48. Nesse contexto, o presente trabalho tem como objetivo aplicar, para esse mesmo estudo, os algoritmos CHAID (Chi-square iteration automatic detection) e CHAID Exaustivo, comparando com os resultados obtidos com o emprego do algoritmo SimpleCart. A validação dos resultados, mostraram que o algoritmo CHAID Exaustivo obteve o melhor resultado em comparação ao algoritmo CHAID, obtendo uma estimativa de acerto de 75,9% contra 58,6% respectivamente, e um erro padrão de 7,9% contra 9,1% respectivamente, enquanto que, o algoritmo já testado na literatura SimpleCart(CART) teve como resultado 34,5% de estimativa de acerto com um erro padrão de 8,8%. Com relação aos tempos de execução apurados sobre 22 mil registros, evidenciaram que o algoritmo CHAID Exaustivo apresentou os melhores tempos, com ganho de 0,02 segundos sobre o algoritmo CHAID e 14,45 segundos sobre o algoritmo SimpleCart(CART).

Palavras-chave: Algoritmos, Árvore de decisão, SimpleCart, CART, CHAID, CHAID Exaustivo, Tratamento de superfícies, TiO2, Titânio, MC3TE-E1.

ABSTRACT

The interest for the area of analysis and characterization of biomedical materials as the need for selecting the adequate material to be used increases. However, depending on the conditions to which materials are submitted, characterization may involve the evaluation of mechanical, electrical, optical, chemical and thermal properties besides bioactivity and immunogenicity. Literature review shows the application decision trees, using SimpleCart(CART) and J48 algorithms, to classify the dataset, which is generated from the results of scientific articles. Therefore the objective of this study was to identify surface characteristics that optimizes the cellular activity. Based on published articles, the effect of the surface treatment of titanium on the in vitro cells (MC3TE-E1 cells) was evaluated. It was found that applying SimpleCart algorithm gives better results than the J48. In this sense, the present study has the objective to apply the CHAID (Chi-square iteration automatic detection) algorithm and Exhaustive CHAID to the surveyed data, and compare the results obtained with the application of SimpleCart algorithm. The validation of the results showed that the Exhaustive CHAID obtained better results comparing to CHAID algorithm, obtaining 75.9 % of accurate estimation against 58.5%, respectively, while the standard error was 7.9% against 9.1%, respectively. Comparing the obtained results with SimpleCart(CART) results which had already been tested and presented in the literature, the results for accurate estimation was 34.5% and the standard error 8.8%. In relation to execution time found through the 22.000 registers, it showed that the algorithm Exhaustive CHAID presented the best times, with a gain of 0.02 seconds over the CHAID algorithm and 14.45 seconds over the SimpleCart(CART) algorithm.

Keywords: Algorithms, Decision tree, SimpleCart, CART, CHAID, Exhaustive CHAID, Surface treatment, TiO2, Titanium, MC3TE-E1.

1 INTRODUÇÃO

Os biomateriais têm desempenhado um papel cada vez mais importante para o sucesso de dispositivos biomédicos capazes de reconstituir ou substituir tecidos e órgãos, esse desenvolvimento representa um grande avanço na melhoria na qualidade de vida das pessoas que passaram por traumas ou patologias, aumentando sua expectativa de vida.

O titânio comercialmente puro (Ticp) e as ligas de titânio são as matérias primas mais utilizadas como biomateriais. O grande sucesso desses materiais resulta da combinação favorável de sua biocompatibilidade, propriedades mecânicas (aceitável limite de resistência à tração, ductilidade e reduzido módulo de elasticidade) e excelente resistência à corrosão.

As ligas metálicas passam por tratamento de superfície melhorando sua biocompatibilidade para uso em aplicação biomédica. Pesquisas em desenvolvimento procuram novas metodologias de tratamento para essas ligas. O objetivo do tratamento de superfície é modificar a morfologia superficial das ligas metálicas, principalmente do titânio. Através do tratamento de superfície, é possível otimizar o emprego das ligas pela melhoria das propriedades como rugosidade, molhabilidade, resistência à corrosão e biocompatibilidade. Com essas modificações, é possível melhorar o desempenho do material empregado nos implantes biomédicos.

Diversas técnicas de tratamento de superfície são descritas na literatura para melhorar a biocompatibilidade do titânio e suas ligas com o intuito de otimizar suas propriedades de superfície, visando aumentar a aderência das células ao implante, consequentemente a osseointegração. O desenvolvimento e o uso dessas técnicas baseiam-se na teoria de que o aumento do contato osso/implante pode ser atingido pela mudança da topografia ou pelo aumento da rugosidade superficial do implante [1]. Esses tratamentos incluem processos de modificação da topografia e da rugosidade superficial, como jateamento de partículas [2], ataque ácido [3], anodização eletroquímica, eletropolimento, recobrimento com camada de materiais biocompatíveis, técnicas de implantação iônica [4] e técnicas de aspersão a plasma.

A biocompatibilidade do titânio e suas ligas vem desencadeando um número crescente de pesquisas de diversos seguimentos, consequência disso, existe também inúmeras formas de representar essas informações, cada qual, registrando os dados da sua forma. Por essa falta de procedimento ou padrão de preparo dos dados, acaba dificultando a realização da correlação entre essas informações.

A mineração de dados é uma técnica que engloba algoritmos utilizados para a construção efetiva de um modelo de conhecimento, com a finalidade de descobrir e analisar quantidades de dados e informações para encontrar modelos e padrões significativos, a qual pode ser representada sob a forma de uma árvore de decisão.

Embora os algoritmos de árvores de decisão não possuam muitas pesquisas na área de biomateriais, são uma das técnicas de mineração de dados que se tem mostrado referência no desenvolvimento e análise de novas propostas para classificação em bases de dados. Este fato decorre de características importantes, como boa acurácia na classificação, rapidez no treinamento e na execução, não fazem suposições estatísticas sobre os dados e possuem habilidades para manipular dados de diferentes escalas de medidas, assim permitindo a utilização de um conjunto de dados amplo e variado.

Diante dessas considerações, o presente trabalho teve por objetivo avaliar a acurácia da classificação das informações coletadas em artigos científicos utilizando a técnica de árvores de decisão, empregando os algoritmos *CART, CHAID e CHAID Exaustivo*.

2 OBJETIVO

O objetivo do trabalho é prever a atividade celular a partir do uso de celular *in vitro* para superfície TiO_2/Ti, aplicando os algoritmos de árvores de decisão *CHAID* e *CHAID Exaustivo*. Pretende-se com isso avaliar os resultados de desempenho e de classificação dos algoritmos na influência das propriedades da superfície TiO_2/Ti.

2.1 Objetivos Específicos

- Pesquisar biomateriais, titânio e suas ligas metálicas, algoritmos de árvores de decisão;
- Comparar os resultados obtidos os algoritmos de árvores de decisão *CHAID* e *CHAID Exaustivo* com o empregado por Gamba (2015);
- Determinar qual algoritmo apresenta o melhor desempenho de acurácia em função das propriedades superficiais;
- Determinar o melhor desempenho de tempo de execução dos algoritmos estudados.

3 BIOMATERIAIS

O uso de biomateriais no corpo humano é utilizado em várias aplicações na medicina, como substituição funcional ou morfológica de sistemas biológicos de tecidos e órgãos, com caráter temporário ou permanentemente, vital ou estético; e em aplicações terapêuticas e diagnósticas [5].

A definição de biomateriais é qualquer substância ou combinação de substâncias, exceto fármacos, de origem natural ou sintética, que podem ser usadas durante qualquer período de tempo, como parte ou como sistemas que tratam, aumentam ou substituem quaisquer tecidos, órgãos ou funções do corpo [6].

A seleção dos materiais torna-se criteriosa quanto a extensão dessas reações, avaliando o fato dos materiais causarem reações desejáveis e indesejáveis. Dessa forma, os biomateriais são classificados como bioativos, biotoleráveis, bioinertes e absorvíveis [7,8].

- Bioativos: são materiais que não formam uma camada fibrosa. O material implantado favorece a ligação química entre o material implantado e o tecido ósseo [8].
- Biotoleráveis: são materiais isolados dos tecidos adjacentes pela formação de uma camada fibrosa no seu entorno, sendo tolerados pelo organismo. Quanto maior for a espessura desta camada fibrosa, menor é a tolerabilidade dos tecidos ao material [8].
- Bioinertes: são materiais que liberam apenas quantidades mínimas de componentes, provocando a formação de uma camada fibrosa muito pequena quando implantados no organismo humano [8].
- Absorvíveis: quando o material implantado é degradado, solubilizado ou fagocitado (englobado por células do sistema imunológico) no organismo [7].

Quanto à sua composição química, podem ser classificados em: metálicos, cerâmicos, poliméricos, compósitos e naturais [9]. Os metálicos aplicados neste trabalho, são utilizados em duas importantes áreas: ortopédica e de estimulação neuromuscular. As aplicações ortopédicas envolvem o uso do material para o reparo ou substituição de alguma parte do sistema esquelético, já na estimulação neural ou neuromuscular, são usados em um sistema eletrônico a fim de prover uma estimulação elétrica para os tecidos [10].

3.1 Titânio

Com o aumento da longevidade, desgastes não regeneráveis de ossos devido a atividades esportivas e acidentes de trânsito, requisitos para a substituição das articulações continuam a crescer. O titânio é um dos poucos materiais que correspondem naturalmente aos requisitos para implantação no corpo humano [11,12].

O baixo módulo de elasticidade, resistência à tração e resistência à corrosão são encarados como uma vantagem para aplicação do Ti como biomaterial [12].

As propriedades físicas do titânio (Ticp) podem variar de acordo com o grau de impureza de elementos, tais como: nitrogênio (N), ferro (Fe), oxigênio (O), carbono (C) e hidrogênio (H), sendo classificado de 1 a 4, de acordo com os traços dos elementos considerados como impurezas, como apresentado na Tabela 1[7].

Tabela 1 - Classificação do titânio em função dos teores de impureza [7].

Tipo	*% de Limites Máximos de impurezas*					*Resistencia à tração (Mpa)*	*% Alongamento*
	N	*Fe*	*O*	*C*	*H*		
Grau 1	*0,03*	*0,20*	*0,18*	*0,10*	*0,01*	*240*	*24*
Grau 2	*0,03*	*0,30*	*0,25*	*0,10*	*0,01*	*340*	*20*
Grau 3	*0,05*	*0,30*	*0,35*	*0,10*	*0,01*	*450*	*18*
Grau 4	*0,05*	*0,30*	*0,40*	*0,10*	*0,01*	*550*	*15*

O Ti possui duas formas cristalográficas. Na estrutura cristalina da fase alfa (α) quando em temperatura ambiente, tem forma hexagonal. E na fase beta (ß), com temperatura de 883°C o Ti passa para a forma cúbica de corpo centrado. Estas estruturas estão esquematicamente representadas na Figura 1[19].

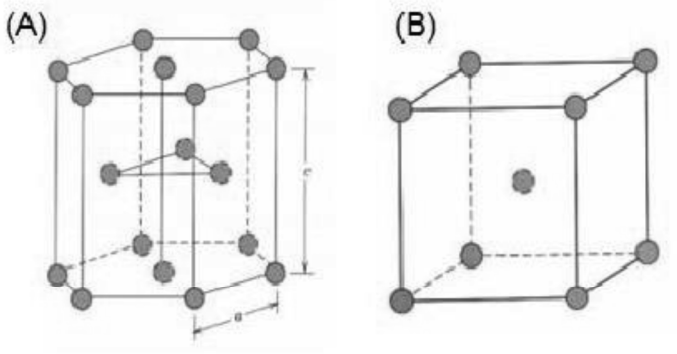

Figura 1 - Estrutura cristalina do titânio: a) Hexagonal compacta na temperatura ambiente; b) Cúbica de corpo centrado na temperatura de transformação alfa-beta [19]

A superfície TiO_2 formada sobre o Ti pode apresentar três estruturas cristalinas: anatase, rutilo com estrutura tetragonal e broquita com estrutura ortorrômbica. A presença de rutilo é predominante nos implantes que possuem tratamento de superfície. Os óxidos de titânio rutilo e anatase são formados pelos tratamentos de superfície, com o controle de rugosidade e composição química [13].

A rugosidade e a topografia da superfície são consideradas de extrema importância para a osseointegração. A rugosidade da superfície tem influência direta no ancoramento das células ósseas e consequentemente na sua proliferação no material, aderindo com maior facilidade nas superfícies rugosas e aparecendo de maneira diferenciada quando comparadas com a morfologia de uma matriz extracelular [14]. Outros fatores considerados determinantes para uma reação positiva e orientação das células no substrato, são: a largura, a profundidade e a quantidade de sulcos [15].

3.2 Tratamento de Superfícies

O objetivo do tratamento de superfície é obter uma topografia adequada, uma rugosidade específica, remover contaminações e/ou aumentar a aderência dos tratamentos que serão feitos posteriormente [16].

As seguintes técnicas produzem modificações morfológicas e físico-químicas, que podem ser obtidas pelos seguintes métodos:

- Adição: Adição de uma camada sobre a superfície, podendo produzir superfícies porosas e rugosas. Influenciando na camada de óxido a ser produzida;
- Subtração: Retirada da camada da superfície, alterando a sua morfologia;
- Combinadas: Modificações superficiais mais controladas e elaboradas com subtração e adição de novas camadas.

Os parâmetros adequados para uma superfície apropriada de implantes ainda não estão bem definidos na literatura. Assim novas técnicas de modificação de superfícies vêm sendo estudadas [17]. A Tabela 2 apresenta uma visão geral de alguns tratamentos de superfícies em Ti e suas ligas, de acordo com os propósitos clínicos necessários.

Tabela 2 - Métodos de modificação da superfície de Ti e suas ligas com os respectivos objetivos [Adaptada de 6]

Métodos de modificação da superfície	***Camada modificada***	***Objetivos***
Métodos Mecânicos		
Usinagem	*Superfície lisa ou rugosa formada por processos de subtração*	*Produzir topografias superficiais específicas; superfície limpa e rugosa; melhorar a adesão*
Polimento		
Jateamento de Partículas		
Abrasão		
Métodos Químicos		
Ataque Ácido	*Camada de óxido superficial menor que 10 nm*	*Remover camadas de óxidos e contaminações*
Tratamento alcalino	*Gel de titanato de sódio de ~1 μm*	*Melhorar a biocompatibilidade, bioatividade ou condutividade óssea*
Tratamento de peróxido de hidrogênio	*Camada de óxido inerte densa e camada exterior porosa de ~5 nm*	
Sol-gel	*Filme fino, tais como fosfato de cálcio, TiO_2 e sílica de ~10 μm*	
Oxidação Anódica	*Camada de TiO_2, com adsorção e inclusão de ânions de eletrólitos de ~10 nm a 40 μm*	*Produzir topografias superficiais específicas; melhora a resistência à corrosão; melhorar a biocompatibilidade, bioatividade ou condutividade óssea*
Métodos Físicos		
Flame Spray	*Revestimentos de titânio, HA, silicato de cálcio, Al_2O_3, ZrO_2, TiO_2 de ~30 μm a ~200 μm*	*Melhorar a resistência ao desgaste, resistência à corrosão e as propriedades biológicas*
Plasma Spray		

3.2.1 Efeito da rugosidade

No processo de fabricação de implantes não é possível produzir superfícies ideais. A superfície de um implante observada ao microscópio é dotada de regiões com maior ou menor planicidade, que é definida como sendo rugosidade, mesmo que estas peças sejam completamente lisas em aspecto microscópico[18].

A rugosidade da superfície oferece uma melhor adesão, por onde migram os osteoblastos para as proximidades da superfície do implante a fim de secretar matriz óssea, dando início a formação de uma interface osseointegrada [19].

Apesar da microrugosidade apresentar um importante efeito sobre a resposta aos biomateriais, também há estudos que indicam uma resposta biológica as irregularidades em

dimensões nanométricas [20]. A influência da configuração microgeométrica e textura da superfície final sobre a resposta óssea ainda não é completamente compreendida, uma vez que existem vários parâmetros para a medida da rugosidade superficial, no entanto a mais utilizada é a rugosidade média [21].

O parâmetro mais utilizado para medir a rugosidade da superfície é conhecido como a rugosidade média (Ra) que indica apenas o valor médio aritmético da altura dos picos e vales em relação a uma linha de referência [22].

A Figura 2 apresenta a adesão de fibroblastos em araldite com uma rugosidade micrométrica.

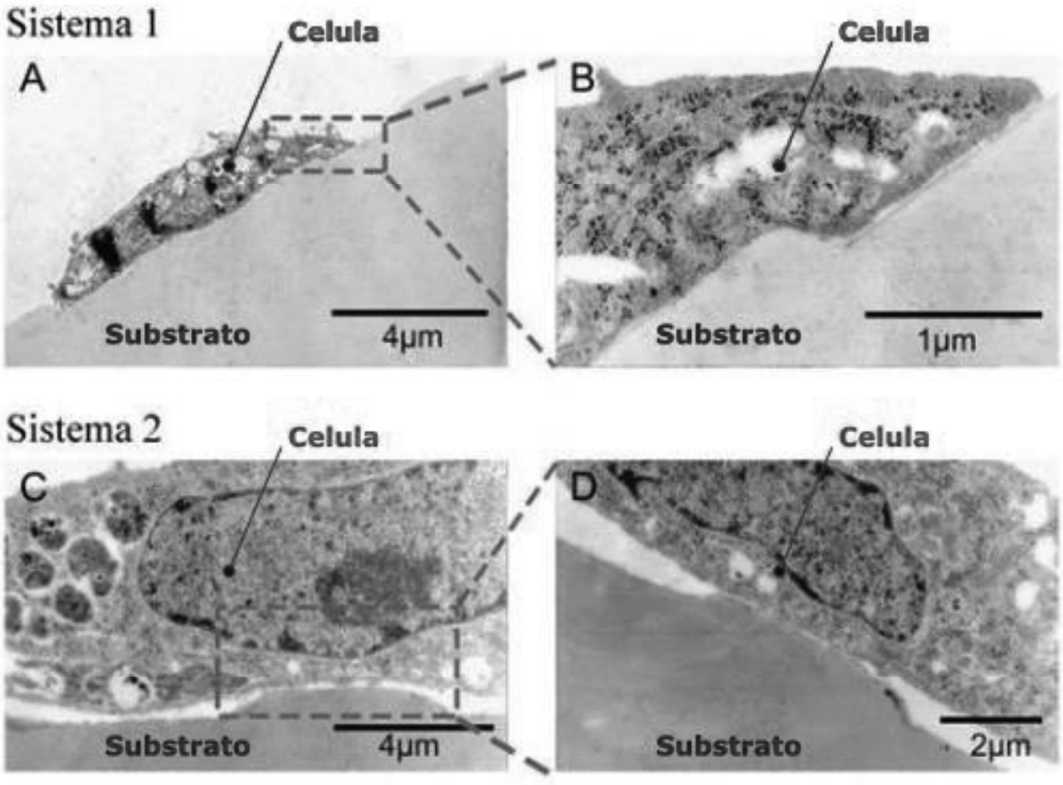

Figura 2 - Fibroblastos gengivais humanos crescendo em réplicas de Araldite incorporadas com borrifamento de Ti. (A) rugosidade que apresenta ranhuras 2 µm de largura e 0,4 µm de profundidade. (B) é ampliação da A. (C) rugosidade que apresenta ranhuras 5 µm de largura e 0,4 µm de profundidade. (D) é ampliação da C [Adaptada de 23*]*

A Figura 2 apresenta o sistema 1 onde as células aderiram perfeitamente à superfície, sem espaços entre as células e o substrato, devido as medidas da rugosidade estarem de acordo com o tamanho das células. O sistema 2 apresenta uma rugosidade com largura maior que a do sistema 1, sendo possível observar a presença de espaços entre o substrato e as células [23].

Assim, descoberto que as células não se estendem para dentro das ranhuras, a menos que a largura da ranhura seja de 10 μm.

As superfícies são classificadas como isotrópicas ou anisotrópicas. As superfícies isotrópicas possuem estruturas iguais em todas as direções com respeito a tamanhos e desvios espaciais. Já as superfícies anisotrópicas possuem estruturas irregulares com orientações diferentes e preferenciais. Superfícies isotrópicas por serem mais homogêneas permitem melhor adesão das células [24].

3.3 Efeito da molhabilidade

A molhabilidade é parâmetro de avaliação da biocompatibilidade, promovendo a adesão, amplia a fixação óssea, crescimento e proliferação de células sobre a superfície do implante, tornando-o mais estável e aderido aos tecidos orgânicos [25].

O grau de molhabilidade é expresso pela medida do ângulo de contato. A técnica de molhabilidade envolve a medida de um ângulo formado entre um plano tangente a uma gota do líquido, onde o líquido que se encontra depositado, forma o ângulo de contato, conforme a Figura 3 [25].

Figura 3 - Definição do ângulo de contato (Adaptado de [25]*)*

Definição de ângulo de contato θ entre uma gota líquida e uma superfície plana e horizontal. γS e γLV são a energia de superfície do sólido e a tensão superficial do líquido em equilíbrio com o vapor; γSL é a energia da interface sólido-líquido [25].

Quando este ângulo de contato estiver acima de 90°, é considerado hidrofóbica, ou seja, o líquido não molha a superfície; se o ângulo estiver abaixo de 90°, é considerado hidrofílica, onde o líquido molha a superfície [25]. Logo, quanto menor o ângulo formado entre a gota e o substrato, mais hidrofílica será a superfície, assim apresentando melhor adesão e por consequência melhor proliferação, crescimento celular e atividade celular [25].

A Figura 4 apresenta diferença em termos de uma superfície hidrofóbica e hidrofílica.

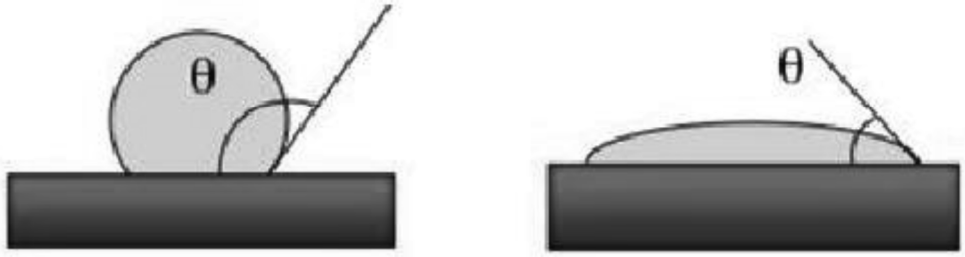

Figura 4 - Representação do ângulo formado entre a gota e a superfície: (a) Superfície hidrofóbica (b) Superfície hidrofílica (Adaptado de [25] *)*

3.4 Ensaios *in vitro* e *in vivo*

Os ensaios *in vitro* e *in vivo* são aplicados para avaliação da biocompatibilidade dos materiais com a interface celular [26].

Ensaios *in vivo* provem informações sobre as reações biológicas de um implante a partir de experimentos realizados em animais. Devido ao grande número e complexidade de eventos que ocorrem, os resultados são de difícil interpretação a nível celular [26,27].

Os ensaios *in vitro* surgiram para a avaliação da proliferação celular sem a interferência dos efeitos provocados pelo organismo [53].

Assim, o sistema de ensaio *in vitro*, em comparação à resposta de tecidos *in vivo,* representa um sistema válido de análise de resposta celular. Por exemplo, um ensaio *in vitro* permitindo identificar os materiais que não apresentam características adequadas à utilização em estudos clínicos, fornecendo com êxito informações úteis a respeito da interação do material com o ambiente fisiológico e dos possíveis riscos associados à sua aplicação [28].

O ensaio *in vitro* não é o modelo ideal para prever riscos, tornando-se importante e necessário o ensaio *in vivo* para prever quando um dispositivo médico apresenta riscos potenciais [26–28].

O ensaio *in vitro* se aproximando mais das propriedades biológicas do tecido original, é realizado por cultivo de células em poços de placas de cultura celular para avaliar a compatibilidade biológica dos biomateriais. As células do ensaio *in vitro* são obtidas de tecidos animais, ou linhagens celulares fornecidas comercialmente por banco de células [27].

O modelo *in vivo* continua sendo necessário para validar os resultados de um modelo *in vitro*. Os dados dos estudos *in vitro* apenas contribuem para a redução do uso de animais em experimentos [26].

3.5 Determinação de atividade celular e viabilidade

A avaliação das células em experimentos de viabilidade celular torna-se imprescindível. A quantificação celular é um método que avalia a viabilidade celular, com a finalidade de informar a quantidade de células que se encontram em determinada placa de cultivo [29].

A quantificação celular é realizada de forma direta e indireta. A forma direta é contado o número de células presente na placa de cultivo, a forma indireta é determinada pelas estruturas celulares, como pela quantificação do metabolismo celular, proteínas e DNA [29].

A contagem em câmara de Neubauer é o método mais utilizado para a quantificação direta. Esse método consiste em uma lâmina de vidro com divisões que auxiliam na contagem [29]. O corante Trypan é utilizado para analisar a viabilidade celular na contagem de células, onde as células mortas adquirem a coloração azul e as células vivas não permitem a passagem do corante [29].

Os métodos de contagem indireta mais utilizados para encontrar a viabilidade celular são os métodos colorimétricos. Dentre os diversos métodos, o corante 3-(4,5-dimetiltiazol-2-il)-2,5-difenil brometo de tetrazólio (MTT) é o mais comumente utilizado, por ser um modelo simples e eficaz para mensurar a viabilidade, proliferação e atividade das células [28,29].

O princípio do método é baseado na capacidade das enzimas desidrogenases localizadas nas mitocôndrias de células viáveis em converter o sal de tetrazólio em um sal de formazan de coloração azul escuro. A quantidade de cristais de formazan formados é diretamente proporcional ao número de células viáveis presente no experimento [27,28,30].

4 DESCOBERTA DE CONHECIMENTO EM BASE DE DADOS

A descoberta de conhecimento em bases de dados, também conhecida como *Knowledge Discovery in Databases* (KDD), é um processo que tem como objetivo principal extrair conhecimento a partir de grandes bases de dados. Para isso envolve diversas áreas de conhecimento, tais como: estatística, matemática, bancos de dados, inteligência artificial, visualização de dados e reconhecimento de padrões. São utilizadas técnicas, em seus diversos algoritmos, oriundos dessas áreas [31].

Existem inúmeras áreas de aplicação de KDD, como: bancária (aprovação de crédito), comerciais (segmentação, localização de consumidores, identificação de hábitos de consumo), engenharia (simulações e análises, reconhecimento de padrões, processamento de sinais e planejamento), ciências e medicina (descoberta de hipóteses, diagnóstico, classificação, predição) e entre outras como na biomedicina que vem sendo estudada [32].

Portanto, o KDD tem o objetivo de encontrar conhecimento a partir de um conjunto de dados para ser utilizado em processo decisório. Esta técnica, poderá auxiliar especialistas na tomada de decisão, uma vez que obtém um novo conhecimento, que não está explícito em suas bases de dados ou de informação [32].

Segundo Castanheira [31], o processo de KDD é composto pelas etapas de seleção dos dados, pré-processamento dos dados, transformação dos dados, mineração de dados e avaliação dos resultados, como apresentado na Figura 5.

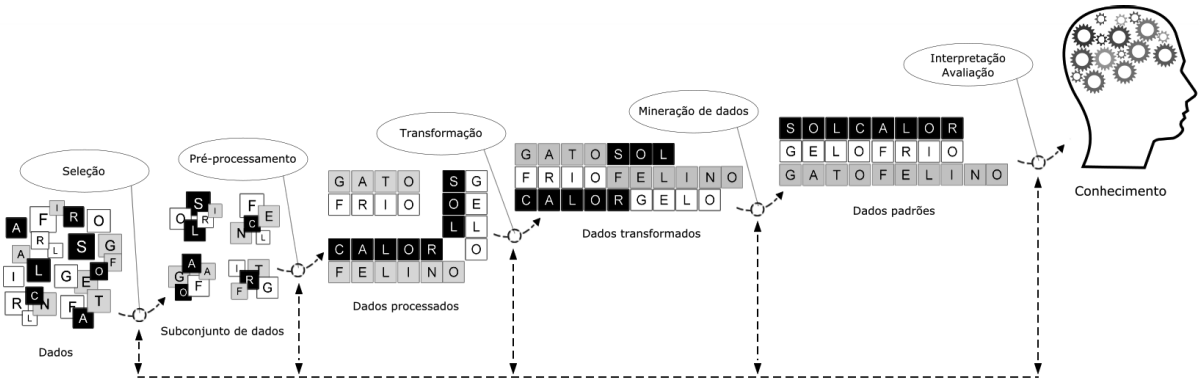

Figura 5 - Fases da descoberta de conhecimento em bases de dados (Adaptada de [31]*).*

Desde a seleção da base de dados até a descoberta do conhecimento, pode ser considerado um conjunto de atividades contíguas que compartilham conhecimento a partir de bases de dados [31].

4.1 Seleção dos dados

De acordo Castanheira [31], esta etapa envolve a compreensão do domínio e dos objetivos da tarefa, criação do conjunto de dados envolvendo as variáveis necessárias. Os dados são a espinha dorsal do processo de KDD, mas usualmente não estão disponíveis de uma forma pronta para seres aplicados, direto no algoritmo de mineração de dados, um dos principais problemas em coletar dados é descobrir onde encontrá-los.

Assim, para resolver problemas específicos, dados adequados devem ser extraídos de bancos de dados ou dados novos coletados que forneçam as exigências da tarefa a ser realizada.

4.2 Pré-processamento dos dados

Esta etapa envolve operações como tratar a falta de dados em alguns campos do banco de dados, limpeza de dados como a verificação de inconsistências, redução da quantidade de campos em cada registro, o preenchimento ou a eliminação de valores nulos e a remoção de dados duplicados [31].

Inconsistências são bastante comuns neste tipo de tarefa e ocorrem quando um atributo assume valores diferentes, mas que representam a mesma coisa [33].

4.3 Transformação dos dados

Esta fase antecede a seleção dos algoritmos de mineração de dados. Os algoritmos possuem padrões que devem ser respeitados, logo esta etapa do processo de KDD é realizada de acordo com o algoritmo de mineração que será utilizado na tarefa escolhida [31].

Estão envolvidas nesta etapa tarefas como identificação *outliers* (valores fora de uma faixa de valores aceitável para um atributo), generalização de atributos e discretização de variáveis.

Dados distorcidos ou que foram improvisados são comuns. Eles acontecem principalmente quando um sistema é desenvolvido para um propósito específico, e passa a ser utilizado para outro [31].

Generalizações podem ser utilizadas quando os dados são muito esparsos e não se consegue resultados satisfatórios com eles. Neste caso, dados primitivos são substituídos por conceitos.

A normalização é outro tipo de transformação de dados. O propósito da normalização é ajustar as escalas de valores dos atributos para um mesmo intervalo, e assim minimizar os problemas oriundos do uso de unidades de medida diferentes entre as variáveis.

Alguns algoritmos de classificação e agrupamento trabalham somente com dados no formato nominal, ou seja, não conseguem lidar com os atributos medidos na escala numérica.

4.4 Mineração de dados

Na fase de mineração de dados, é escolhida a tarefa e definido o algoritmo a ser utilizado, podendo ser executado mais de uma vez já que esta etapa é um processo iterativo, para que haja a extração de padrões [33]. Uma vez escolhido o algoritmo a ser utilizado, é necessário testá-lo e adaptá-lo à natureza da tarefa escolhida para a resolução do problema.

Aplicação de métodos de mineração de dados cega pode ser uma atividade perigosa, facilmente levando à descoberta de padrões sem sentido ou inválidos [34].

4.5 Avaliação dos resultados

Esta é a última etapa do processo de KDD, no qual os conhecimentos encontrados são interpretados e utilizados em processos de tomada de decisão. As medidas de desempenho (precisão, tempo, outros) também são exibidas nesta fase, podendo, caso necessário, ajustar parâmetros e voltar a alguma etapa anterior para ser executada novamente [35].

Os resultados da mineração de dados devem ser apresentados de forma clara, para que as informações possam ser interpretadas e visualizadas de diversas formas, utilizando-se de recursos visuais, como tabelas, gráficos, entre outros.

5 MINERAÇÃO DE DADOS

A Mineração de Dados (Data Mining) combina métodos tradicionais de análise de dados com algoritmos sofisticados para processar grandes volumes de dados. Essa análise busca identificar padrões e comportamentos dentro dos processos à ela submetida.

Esta área encontrar-se em expansão em diversas segmentos, como na área de estudos de biomateriais, e necessita de estudos complementares tanto na definição dos atributos a serem utilizados quanto nas técnicas de mineração de dados empregadas.

5.1 Tarefas e técnicas da mineração de dados

Os objetivos a serem alcançados são o fator responsável pela definição da escolha das tarefas a serem utilizadas na mineração de dados [33]. Não existe uma definição genérica de tarefa que seja mais ou menos eficiente em qualquer situação, cada caso é um caso.

Após a escolha da tarefa, define-se a técnica a ser utilizada nela. A tarefa se diferencia-se de técnica de mineração pelo fato de especificar qual a informação ou padrão deseja-se encontrar nos dados e a técnica específica dos métodos que serão aplicados para alcançar os objetivos desejados [36].

5.1.1 Tarefas preditivas

A predição é um dos objetivos fundamentais da mineração de dados, utiliza algumas variáveis que se encontram no banco de dados, com a finalidade de prever valores desconhecidos ou futuros de outras variáveis que sejam de interesse [36].

Nas tarefas preditivas (também conhecidas por modelos de descoberta) a pesquisa é feita de forma a encontrar padrões frequentes, tendências e generalizações, a fim de encontrar informações que estavão escondidas nos dados [33].

5.1.1.1 Classificação

A tarefa de classificação diz respeito ao processo de encontrar um modelo que descreve e distingue classes de dados ou conceitos, ou seja, é uma função de aprendizado que

mapeia dados ou conjuntos de entrada em um número finito de classes. Nela, cada exemplo pertence a uma classe, entre um conjunto pré-definido de classes [31].

O objetivo de um algoritmo de classificação é descobrir alguma correlação entre os atributos e uma classe, de modo que o processo de classificação possa usá-lo para predizer a classe de um exemplo novo e desconhecido [37].

A tarefa de classificação é dividida nas etapas de treinamento e de classificação. Na etapa de treinamento, também conhecida como aprendizado, utiliza-se um conjunto de dados denominados amostragem associados a suas classes (rótulos) para criar um modelo que será utilizado na construção do classificador. Esse é um tipo de aprendizado conhecido como supervisionado, uma vez que o conjunto de dados utilizados é pré-definido [38].

Na etapa da classificação, faz-se o uso do modelo criado para o classificador. Utilizam-se agora outros conjuntos de dados, também conhecidos como teste, para estimar a precisão do classificador.

A classificação é importante para evitar o *overfit*, ou seja, evitar que o classificador se ajuste de tal forma que acaba sendo um classificador muito eficaz para os dados de treinamento, porém não tão eficaz para as demais amostragens de teste [38].

A Figura 6 apresenta um modelo do processo de indução de um classificador e, em seguida, a sua utilização.

Figura 6 - Representação do processo de indução de um classificador (Adaptada de [39]*)*

De acordo com [31], um conjunto de treinamento, onde os rótulos das classes dos exemplos são conhecidos, é utilizado por um algoritmo de aprendizado para construir um modelo. Com a construção finalizada, esse classificador pode ser aplicado para predizer os

rótulos das classes dos exemplos do conjunto de teste, ou seja, exemplos cujas classes são desconhecidas.

5.1.1.2 Regressão Linear

Modelos de regressão linear assemelham-se com modelos de classificação. Na tarefa de classificação, os atributos alvos da predição são do tipo discreto enquanto na regressão são do tipo numérico e contínuo [36].

Esta tarefa também utiliza técnicas de classificação, porém, diferentemente da tarefa de classificação onde a técnica é utilizada para classificar instâncias, a regressão busca realizar uma estimativa de valor de uma determinada variável, ou seja, mapear um dado em um ou mais valores reais. Enquanto na tarefa anterior, os registros são classificados em uma classe, nesta tarefa os registros são classificados em um valor baseado em uma função matemática [33].

Modelos lineares também podem ser aplicados em problemas de classificação binária. Nesses casos, a linha produzida pelo modelo separa as duas classes. Ela define onde a decisão muda de uma classe de valores para a outra, tal linha é muitas vezes referida como fronteira de decisão [36].

5.1.2 Tarefas descritivas

A descrição também é um dos objetivos fundamentais da mineração de dados, busca por padrões que descrevem os dados, de forma que possam ser interpretáveis pelos usuários a fim de encontrar respostas que confirmem ou neguem as hipóteses [36].

Nas tarefas descritivas (também conhecidas por modelos supervisionados, modelos de verificação) a abordagem é do tipo top-down, ou seja, existem hipóteses que foram previamente formuladas e são testadas para a verificação da sua veracidade [33].

Dentre as tarefas descritivas na mineração de dados mais utilizadas, encontra-se a tarefa de análise de associações ou regras de associação. Esta tarefa consiste na descoberta de regras que mostram condições nos valores dos atributos que sugerem padrões de associação fazendo um levantamento de quanto um conjunto de atributos contribui para a presença de outro conjunto, realizando um estudo de como os itens estão relacionados [33].

Podem ser aplicadas em estudos de preferência, buscando por afinidade entre os dados. Seu principal objetivo é encontrar conjuntos de itens ou eventos que ocorram junto, baseado na

teoria de que a presença de um item em uma determinada transação implica na ocorrência de outro [38].

5.2 Classificação e árvores de decisão

Na tarefa de classificação, as técnicas mais utilizadas nos trabalhos estudados foram árvores de decisão.

A árvore de decisão pode ser utilizada com um custo computacional muito baixo. Além disso, a interpretação da árvore de decisão é uma das suas principais virtudes.

Uma árvore de decisão pode ser estruturada de diversas maneiras a partir de um conjunto de atributos. De forma exaustiva, à medida em que o número de atributos cresce, o número de árvores de decisão possíveis cresce exponencialmente, tornando impraticável definir a estrutura da árvore de decisão ótima para um determinado problema, devido ao elevado custo computacional envolvido nessa busca [39].

A árvore de decisão é formada pelas estruturas chamadas de raiz, nós internos, arestas e folhas (Figura 7). Os nós internos significam testes sobre um determinado atributo, cada aresta representando um possível valor para esse atributo e cada folha apresentando um valor do atributo classe (rótulo) com que se deseja classificar a tupla de entrada. A raiz é o primeiro atributo a ser testado [38].

O aprendizado em árvores de decisão é do tipo supervisionado, sua construção é baseada no modelo *Top-down*, partindo do nó raiz em direção às folhas terminais. Os algoritmos dessa categoria se utilizam da técnica de dividir para conquistar, dividindo os problemas em problemas de menores dimensões até encontrar a solução para cada um dos problemas divididos [36].

Classificadores com essa técnica procuram dividir sucessivamente o conjunto de dados, até que cada conjunto contemple apenas uma classe, dispensando as novas divisões [37].

A árvore de decisão é montada a partir de dados de treino onde a princípio tem-se apenas um nó que contém todas as classes. Recursivamente, escolhe-se um atributo que possa dividir esta classe, até que não haja mais divisões e cada nó folha represente uma única classe ou satisfação de um critério [33]. A escolha do atributo a ser testado em cada nodo é o que define o sucesso de um algoritmo de aprendizado, que gera a árvore de decisão.

A Figura 7 exibe um exemplo de árvore de decisão que classifica alunos da disciplina de programação entre “Confusos” e “Não Confusos” de acordo com os atributos “Número de Compilação Com Erros” e o atributo “Número de Pares de Compilações com o Mesmo Erro”.

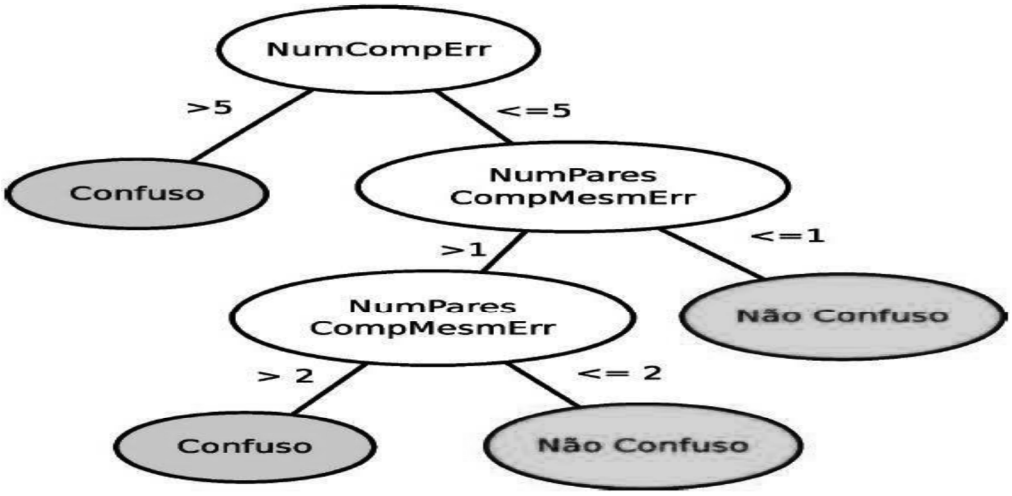

Figura 7 - Exemplo de árvore de decisão (Adaptada de [37]*)*

Seguindo o exemplo de [31], o primeiro teste feito na árvore é sobre a variável NumCompErr (número de compilação com erros) onde é verificado se ela é maior que 5, classificando o aluno como “confuso”, senão ela testa a segunda variável NumParesCompMesmErr (quando o aluno compilou o mesmo erro mais que uma vez, ou seja, um par), para valores maiores que 1 outro teste é feito sobre a mesma variável NumParesCompMesmErr que irá classificar os alunos como “confuso” quando o valor dessa variável for maior que 2.

De acordo com Castanheira [31], nesse sentido, algoritmos baseados em heurísticas têm sido desenvolvidos para a indução de árvores de decisão. Mesmo que eles não garantam uma solução ótima, apresentam resultados satisfatórios em tempo aceitável. Um desses algoritmos é o algoritmo de Hunt, que é a base de muitos algoritmos de indução de árvores de decisão existentes, como o CART [40], ID3 [41], C4.5 [42] e CHAID [43].

5.2.1 Modelo de indução Top-Down

Baseado no algoritmo Top-Down *Induction of Decision Tree* que serve como base para os principais algoritmos de indução para árvores de decisão, este modelo gera regras de decisão em uma árvore de decisão, a qual é construída por várias divisões do conjunto de dados de acordo com os valores de seus atributos preditivos [39].

Na prática, este modelo é baseado em um algoritmo recursivo, que busca sobre um conjunto de atributos, aqueles que "melhor" dividem o conjunto dos dados de exemplo em subconjuntos. Primeiramente, todos os dados são colocados em um único nodo, chamado de nodo raiz. Em seguida, um atributo preditivo é escolhido para representar o teste desse nodo e, consequentemente, dividir os dados em sub-conjuntos de dados. Esse processo se repete recursivamente até que todos os dados já estejam classificados ou então até que todos os atributos preditivos já tenham sido utilizados [36].

5.2.2 Seleção dos atributos preditivos para os nodos das árvores

A escolha por qual atributo preditivo será utilizado em cada nodo da árvore é baseada no critério de seleção. Existem diversos tipos de critérios de seleção, sendo esta uma das diferenças entre os variados algoritmos de indução de árvores de decisão. Esses critérios são baseados em termos da distribuição de classe dos dados antes e após a divisão [36].

A grande maioria dos algoritmos de indução busca dividir os dados de um nodo-pai de forma a minimizar o grau de impureza dos nodos-filhos. Os critérios para a seleção da melhor divisão são baseados em diferentes medidas, tais como dependência, impureza e distância. Quanto menor for o grau de impureza, mais desequilibrada é a distribuição das classes. Se todos os dados pertencem a uma mesma classe em um determinado nodo, a impureza dele é nula. Da mesma forma, se existir o mesmo número de exemplos para cada classe possível, o grau de impureza é máximo neste nodo [39].

5.2.3 Métricas para a melhor divisão da árvore

Existem muitas métricas que podem ser utilizadas para determinar a melhor forma de dividir os dados. Conforme mencionado anteriormente, essas métricas são definidas em termos da distribuição da classe dos dados antes e após a divisão.

Muitas vezes, o grau de impureza do nodo filho é a base utilizada por essas métricas para selecionar a melhor divisão. Quanto menor o grau de impureza, mais distorcida é a distribuição da classe [39].

O Ganho de Informação é uma das medidas baseadas em impureza, o qual utiliza a entropia como medida da impureza. O algoritmo ID3 [41] utiliza essa métrica. Para determinar quão boa é uma condição de teste realizada, é necessário comparar o grau de entropia do nodo-

pai (antes da divisão) com o grau de entropia dos nodos-filhos (após a divisão). O atributo que gerar uma maior diferença é escolhido como condição de teste. O ganho é definido pela Equação (1), na forma:

$$Ganho = Entropia(pai) - \sum_{j=1}^{n} \left[\frac{N(v_j)}{N} Entropia(v_j) \right] \qquad \textit{(Eq. 1)}$$

Onde *n* é o número de valores dos nodo-filhos, *N* é o número total de objetos do nodo-pai e (v_j) é o número de exemplos associados ao nodo-filho v_j. O grau de entropia é definido pela Equação (2) a seguir:

$$Entropia(\text{nó}) = \sum_{i=1}^{c} p\,(\frac{i}{\text{nó}})\ \log_2[p\left(\frac{i}{\text{nó}}\right)] \qquad \textit{(Eq. 2)}$$

Onde *p (i / nó)* é a fração dos registros pertencentes à classe *i* no *nó*, e *c* é o número de classes. O atributo-teste que maximiza o ganho de informação é selecionado pelo critério de ganho. O grande problema ao se utilizar o ganho de informação é que ele dá preferência a atributos com muitos valores possíveis [39].

Um caso clássico desse problema aconteceria ao utilizar um atributo insignificante. Nesse exemplo, conforme descrito por [39], seria criado um nodo para cada valor possível, e o total de nodos seria igual ao número de identificadores. Cada um desses nodos teria apenas um exemplo, o qual pertence a uma única classe, ou seja, os exemplos seriam totalmente discriminados. Assim, o valor da entropia seria mínimo porque, em cada nó, todos os exemplos pertencem à mesma classe. Essa divisão geraria um ganho máximo, embora seja totalmente inútil.

A razão de ganho, da sigla em Inglês (Gain Ratio), soluciona o problema do ganho de informação [41]. Ela nada mais é do que o ganho de informação relativo (ponderado) como critério de avaliação. A razão de ganho é definida pela Equação (3), na forma:

$$Razão\_de\_ganho(\text{nó}) = \frac{Ganho}{Entropia(\text{nó})} \qquad \textit{(Eq. 3)}$$

É possível perceber, pela Equação (3), que a razão de ganho não é definida quando o denominador é igual a zero. Além disso, favorece atributos cujo denominador, ou seja, a

entropia, possui valor pequeno. A razão de ganho é sugerido que seja realizada em duas etapas [41].

Primeiramente é calculado o ganho de informação para todos os atributos. Após isso, deve-se considerar apenas aqueles atributos que obtiveram um ganho de informação acima da média, e então escolher aquele que apresentar a melhor razão de ganho [44].

5.2.4 Atributos categóricos

O desempenho das árvores de decisão induzidas é influenciado de maneira decisiva pela forma de representação dos nodos. Existem diferentes tipos de representação dos nodos para a divisão dos dados, dependendo do tipo de atributo. A seguir, são apresentadas algumas das formas de representação considerando atributos categóricos não-ordinais e ordinais [39].

Um ramo por valor de atributo: Uma aresta é criada para cada valor de atributo usado como condição de teste. Embora esse tipo de partição permita extrair do atributo todo o seu conteúdo informativo, possui a desvantagem de tornar a árvore de decisão mais complexa. O algoritmo C4.5 [42] utiliza esse tipo de divisão para atributos categóricos não ordinais.

Atributos categóricos ordinais: Um atributo é ordinal quando há uma relação de ordem entre os seus possíveis valores. Por exemplo, tem-se um atributo renda que pode possuir os valores ⟨baixa⟩, ⟨média⟩ e ⟨alta⟩. Com atributos desse tipo, é possível realizar uma partição binária do tipo renda < ⟨média⟩, em que todos os exemplos cujo atributo renda tem valor = baixa seguem por uma aresta e os outros seguem por outra aresta. O algoritmo CART utiliza esse tipo de partição [40].

Valores agrupados em dois conjuntos: A divisão binária também pode ser realizada de uma forma mais complexa, onde cada um dos dois subconjuntos pode ser formado por registros com mais de um valor para o atributo utilizado como condição de teste [40]. O elevado custo computacional para encontrar a melhor divisão é o grande desafio desse tipo de divisão, pois o número de combinações possíveis é $(2^{n-1} - 1)$, onde n é o número de valores possíveis para o atributo em questão.

Valores agrupados em vários conjuntos: O algoritmo C4.5 gera uma solução de boa qualidade no intuito de permitir o agrupamento de valores em diversos conjuntos com uma complexidade de cálculo razoável [42]. Para isso, inicia criando uma aresta para cada valor do atributo em teste. Após, são verificadas todas as combinações possíveis de dois valores e, caso nenhuma dessas combinações produza um ganho maior que a divisão anterior, o processo é

interrompido e a divisão anterior é adotada. Caso contrário, o processo é repetido tendo como base a melhor das soluções anteriores. Percebe-se que não se pode garantir que a divisão encontrada seja a melhor possível, pois é verificado se houve melhoria apenas um passo à frente.

5.2.5 Atributos contínuos

De acordo com [31], os testes mais utilizados para partição de atributos contínuos são: testes simples ou pesquisa exaustiva e os testes múltiplos. Os testes múltiplos podem ser de segmentação global ou segmentação ao nível do nó.

Os atributos contínuos permitem uma maior variedade de testes e, consequentemente, implicam uma maior complexidade de cálculo.

O teste simples, também conhecido como pesquisa exaustiva, é o mais utilizado. Um dos algoritmos que o utiliza é o C4.5, e a divisão é sempre binária. Supondo um atributo contínuo Z a ser utilizado como nó teste, mesmo que seu domínio seja infinito, o número de exemplos num conjunto de treinamento Q é finito e, portanto, o número de valores diferentes para esse atributo também é finito.

5.2.6 Métodos de poda em árvores de decisão

Um cuidado que se deve ter com árvores de decisão é o crescimento exagerado da árvore. Caso isso ocorra, deve-se contornar a situação com a operação denominada poda da árvore de decisão. Esta operação consiste em substituir os nodos profundos por folhas, removendo as ligações que fornecem um baixo valor de ganho de informação.

Existem diversas formas de realizar poda em uma árvore de decisão, e todas elas são classificadas como pré-poda ou pós-poda [44].

O método pré-poda é realizado durante o processo de construção da árvore, em que o processo pode simplesmente parar de dividir o conjunto de elementos e transformar o nodo corrente em um nodo folha da árvore.

Um critério de poda que pode ser utilizado é o ganho de informação. Caso todas as divisões possíveis utilizando um atributo Z gerem ganhos menores que um valor pré-estabelecido, então esse nodo vira folha, representando a classe mais frequente no conjunto de dados.

O método pós-poda é realizado após a construção da árvore de decisão, removendo ramos completos, onde tudo que está abaixo de um nodo interno é excluído e esse nodo é transformado em folha, representando a classe mais frequente no ramo.

Para cada nodo interno da árvore, o algoritmo calcula a taxa de erro caso a sub-árvore abaixo desse nó seja podada. Em seguida, é calculada taxa de erro caso não haja a poda. Se a diferença entre essas duas taxas de erro for menor que um valor predeterminado, a árvore é podada. Caso contrário, não ocorre a poda [44].

5.2.7 Super ajuste ou Overfiting

No momento da construção das árvores de decisão, muitas das arestas ou sub-árvores podem refletir ruídos ou erros. Isso acarreta em um problema conhecido como sobre ajuste, que significa um aprendizado muito específico do conjunto de treinamento, não permitindo ao modelo generalizar.

Os erros mais cometidos por um modelo de classificação são geralmente divididos em dois tipos: erros de treinamento e erro de generalização [44]. Erros de treinamento são o número de erros de classificação equivocada contida nos dados de treinamento, enquanto erros de generalização são os erros esperados pelo modelo em dados não vistos anteriormente.

Um bom modelo de classificação deve não apenas se adaptar bem aos dados de treinamento, como também deve classificar com precisão os registros nunca vistos antes por ele. Em outras palavras, um bom modelo deve ter baixa quantidade de erros de treinamento assim como de erros de generalização.

Isso é importante porque um modelo que seja apropriado aos dados de treinamento pode muito bem ter um erro de generalização mais pobre do que um modelo com alto grau de erro de treinamento [44]. Tal situação é conhecida como overfiting do modelo.

5.3 Algoritmos de árvores de decisão

As Árvores de Decisão são modelos práticos e mais usados em inferência indutiva. As árvores são treinadas de acordo com um conjunto de treino (exemplos previamente classificados) e posteriormente, outros exemplos são classificados de acordo com essa mesma árvore. Nesta sessão, serão apresentados de forma sucinta os quatros principais algoritmos para

indução de árvores de decisão. Os algoritmos em estudo são: *ID3* [41], *CART* [40], *CHAID* [43] e *CHAID EXAUSTIVO* [45].

5.3.1 Algoritmo *ID3*

O *ID3* é o algoritmo pioneiro em indução de árvores de decisão. É um algoritmo recursivo, procurando sobre um conjunto de atributos, aqueles que "melhor" dividem os dados, gerando sub-árvores. A partir de um conjunto de dados, ele constrói árvores de decisão, sendo a árvore resultante usada para classificar amostras futuras [41].

O *ID3* separa um conjunto de treinamento em subconjuntos, de forma que esses contenham exemplos de uma única classe. A divisão é efetuada através de um único atributo, utilizando o ganho de informação para medir quanto informativo é um atributo.

O algoritmo *ID3* só lida com atributos categóricos não-ordinais, não sendo possível apresentar a ele conjuntos de dados com atributos contínuos, por exemplo. Nesse caso, os atributos contínuos devem ser previamente discretizados. Além disso, o algoritmo *ID3* também não apresenta nenhuma forma para tratar valores desconhecidos, ou seja, todos os exemplos do conjunto de treinamento devem ter valores conhecidos para todos os seus atributos, isso acaba tornando necessário gastar um bom tempo com pré-processamento dos dados para utilizar esse algoritmo [44].

O ganho de informação é utilizado pelo *ID3* para selecionar a melhor divisão. No entanto, esse critério não considera o número de divisões (número de arestas), e isso pode acarretar em árvores mais complexas. Além disso, o *ID3* também não apresenta nenhum método de pós-poda, o que poderia amenizar esse problema de árvores mais complexas.

A Tabela 3 apresenta um conjunto de dados chamado de *PlayTenis*, que simula um exemplo para decidir se as condições meteorológicas estão apropriadas para jogar uma partida de tênis [46].

Tabela 3 - Conjunto de dados PlayTenis [Adaptada de 65].

Clima	*Temperatura*	*Umidade*	*Vento*	*Jogar*
Sol	*Alta*	*Alta*	*Não*	*Não*
Sol	*Alta*	*Alta*	*Sim*	*Não*
Nublado	*Alta*	*Alta*	*Não*	*Sim*
Chuva	*Média*	*Alta*	*Não*	*Sim*
Chuva	*Baixa*	*Normal*	*Não*	*Sim*
Chuva	*Baixa*	*Normal*	*Sim*	*Não*
Nublado	*Baixa*	*Normal*	*Sim*	*Sim*
Sol	*Média*	*Ata*	*Não*	*Não*
Sol	*Baixa*	*Normal*	*Não*	*Sim*
Chuva	*Média*	*Normal*	*Não*	*Sim*
Sol	*Média*	*Normal*	*Sim*	*Sim*
Nublado	*Média*	*Alta*	*Sim*	*Sim*
Nublado	*Alta*	*Normal*	*Não*	*Sim*
Chuva	*Média*	*Alta*	*Sim*	*Não*

As colunas clima, temperatura, umidade e vento correspondem às condições climaticas, ou seja, são os atributos que representam o conjunto de dados de entrada. A coluna jogar é o rótulo de saída, que indica se é possível disputar a partida de tênis.

A árvore de decisão induzida com base nesses dados utiliza o algortimo *ID3* como forma de exemplificar.

A estratégia utilizada para definir a ordem em que os atributos são usados é a utilização da métrica de entropia.

O *dataset* possui um total de 14 registros e o rótulo de saída é composto pela coluna "jogar" possuindo 9 registros com valor "Sim" e 5 registros com valor "Não", resultando na probabilidade de 9/14 e 5/14.

Inicialmente deve ser calculada a entropia (Equação 1) do rótulo de saída "Jogar" com o total de registros resultando em:

$$Entropia(Jogar) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0{,}940$$

Para o dataset da Tabela 3, as possibilidades de acordo com o atributo de saída em cada atributo de entrada são apresentados na Figura 8.

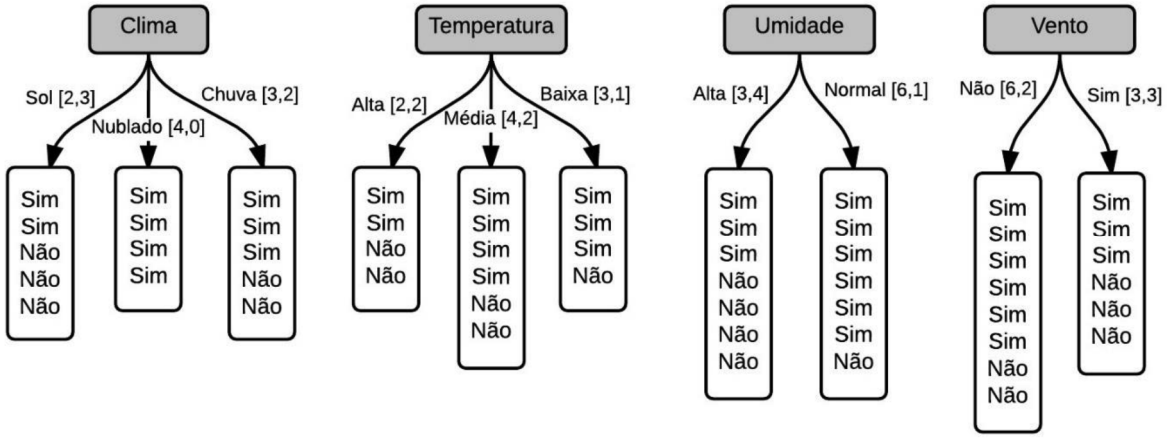

Figura 8 - Valores do atributo de saída em função dos atributos de entrada [Adaptado de 66]

As informações [x, y] de cada atributo de entrada devem ser calculadas de acordo com o total de atributos classificadas como "Sim" pelo atributo "jogar", utilizando a entropia (Equação 1). Para cada valor do atributo, o resultado corresponde a:

$$Entropia(Sol) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0{,}971$$

$$Entropia(Nublado) = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0{,}0$$

$$Entropia(Chuva) = -\frac{3}{3}\log_2\frac{3}{3} - \frac{2}{3}\log_2\frac{2}{3} = 0{,}971$$

Assim, o ganho de informação do atributo clima, calculado a partir da Equação (3), é dado por:

$$Ganho(clima) = 0{,}940 - \left(\frac{4}{14} \times 0{,}971 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0{,}971\right) = 0{,}247$$

Calculando as mesmas informações para os demais atributos de entrada, utilizando a Equação (3), obtem-se os seguintes valores:

$$Ganho(Temperatura) = 0{,}029$$

$$Ganho(Umidade) = 0{,}152$$

$$Ganho(Vento) = 0{,}048$$

O atributo clima é selecionado como a raiz da árvore, pois obteve o maior ganho. Em seguida, devem ser verificados os demais atributos, desconsiderando o atributo tempo que já foi selecionado. A Figura 9 apresenta os demais atributos, considerando o atributo clima com valor igual a sol da ramificação da árvore.

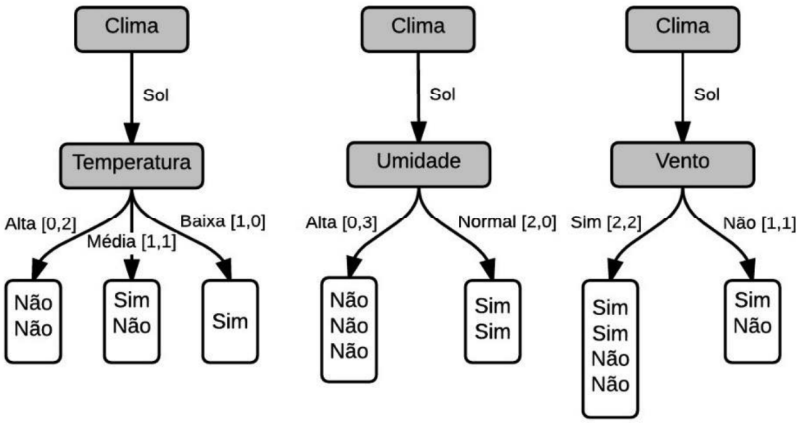

Figura 9 - Valores do atributo de saída em função dos atributos de entrada, considerando o atributo clima igual a sol [Adaptado de 65].

Neste caso, o ganho da informação para os demais atributos, utilizando a Equação (3), é:

$$Ganho(Temperatura) = 0{,}571$$

$$Ganho(Umidade) = 0{,}971$$

$$Ganho(Vento) = 0{,}02$$

O atributo umidade, por obter o maior ganho, é o próximo a ser incluído na árvore. Para o atributo nublado, não há necessidade de se expandir a árvore, pois todas as suas respostas correspondem ao rótulo de saída "jogar" com valor igual a "Sim", o que de fato é a condição de parada. O processo deve ser repetido até que todos os ramos tenham suas saídas

em apenas uma das duas possíveis saída (Jogar = Sim ou Jogar=não), conforme apresentando na Tabela 3.

A árvore de decisão *PlayTenis* gerada é apresentada na Figura 10.

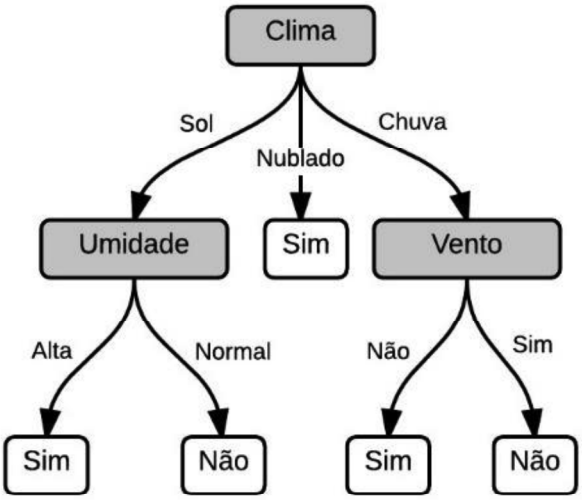

Figura 10 - Árvore de decisão PlayTenis [Adaptado de 65].

As árvores de decisão estão entre os métodos mais utilizados e práticos, sendo aplicadas com sucesso em uma ampla gama de tarefas, como: diagnosticar casos médicos e para avaliar o risco de crédito dos candidatos a empréstimos [47].

5.3.2 Algoritmo *CART*

O algoritmo *CART* (*Classification and Regression Trees*) foi proposto em Breiman [40] e consiste em uma técnica que induz tanto árvores de classificação quanto árvores de regressão, dependendo se o atributo é nominal (classificação) ou contínuo (regressão).

Uma das suas principais vantagens é a grande capacidade de pesquisa de relações entre os dados, mesmo quando elas não são evidentes, bem como a produção de resultados sob a forma de árvores de decisão de grande simplicidade e legibilidade.

O algoritmo *CART* gera árvores binárias, as quais podem ser percorridas da sua raiz até as folhas respondendo apenas a questões simples do tipo “sim” ou “não” [44].

Os nodos que correspondem a atributos contínuos são representados por agrupamento de valores em dois conjuntos. Utiliza a técnica de pesquisa exaustiva para definir os limiares a serem utilizados nos nodos para dividir os atributos contínuos. Também dispõe de um tratamento especial para atributos ordenados, além de permitir a utilização de combinações

lineares entre atributos, ou seja, agrupamento de valores em vários conjuntos. Diferente das abordagens adotadas por outros algoritmos, os quais utilizam pré-poda, o *CART* expande a árvore exaustivamente, realizando pós-poda por meio da redução do fator complexidade- custo [40].

5.3.3 Algoritmo *CHAID*

O algoritmo *CHAID (Chi-square Automatic Interaction Detector)* permite múltiplas divisões de um nó, especificamente é composto por de três etapas: fusão, divisão e paragem. Uma árvore é cultivada repetidas vezes usando as três etapas em cada nó a partir do nó raiz [43].

O algoritmo *CHAID* aceita preditores categóricos nominais ou ordinais. Quando os preditores são contínuos, eles são transformados em preditores ordinais [43].

Para melhor compreensão, as três etapas que compõem o algoritmo *CHAID* são descritos nos subcapítulos a seguir.

5.3.3.1 Fusão

Para cada variável preditora X, é necessário mesclar categorias não significativas. Cada categoria final de X resultará em um nó filho se X for usado para dividir o nó. O passo de fusão também calcula o valor de *p* ajustado, que deve ser usado no passo de divisão [43]:

1. Se X tiver apenas uma categoria, pare e defina o valor p ajustado como 1;

2. Se X tiver 2 categorias, vá para a passo 8;

3. Senão, encontre o par permissível de categorias de X (Um par permissível de categorias para o preditor ordinal é duas categorias adjacentes, e para o preditor nominal é duas categorias qualquer) que é menos significativamente diferente (isto é, mais semelhante). O par mais semelhante é o par cuja estatística de teste dá o maior valor de *p* em relação à variável dependente Y. Como calcular o valor de *p* sob várias situações será descrito em seções posteriores;

4. Para o par que possui o maior valor de *p*, verifique se o seu valor de *p* é maior que um nível alfa especificado pelo usuário. Se o fizer, este par é mesclado em uma única categoria composta. Em seguida, um novo conjunto de categorias de X é formado. Caso contrário, vá para o passo 7;

5. (Opcional). Se a categoria composta recém-formada consiste em três ou mais categorias originais, então encontre a melhor divisão binária dentro da categoria composta, cujo valor *p* é o menor. Execute esta divisão binária se seu valor *p* não for maior do que um nível alfa;

6. Vá para o passo 2;

7. (Opcional). Qualquer categoria que tenha muito poucas observações (em comparação com um tamanho de segmento mínimo especificado pelo usuário) é mesclada com a categoria mais similar, medida pelo maior dos valores de *p*;

8. O valor p ajustado é calculado para as categorias mescladas aplicando os ajustes de *Bonferroni* que serão discutidos mais adiante (5.3.5).

5.3.3.2 Divisão

A "melhor" divisão para cada preditor é encontrada na etapa de fusão. O passo de divisão seleciona qual o preditor a ser usado para dividir melhor o nó. A seleção é realizada comparando-se o valor de *p* ajustado associado a cada preditor, esse valor é obtido na etapa da fusão [43]:

1. Selecione o preditor que tenha o menor valor de *p* ajustado (isto é, o mais significativo);

2. Se este valor de *p* ajustado for menor ou igual a uma divisão do nível alfa especificado pelo usuário, divida o nó usando este preditor. Senão a divisão não ocorre e o nó é considerado como um nó terminal.

5.3.3.3 Parando

A etapa de parando verifica se o processo de crescimento da árvore deve ser parado de acordo com as seguintes regras de paragem [43]:

1. Se um nó se torna puro; ou seja, todos os casos em um nó têm valores idênticos da variável dependente, o nó não será dividido;

2. Se todos os casos em um nó tiverem valores idênticos para cada preditor, o nó não será dividido;

3. Se a profundidade da árvore atual atingir o limite máximo de profundidade da árvore especificado pelo usuário, o processo de crescimento da árvore será interrompido;

4. Se o tamanho de um nó for menor que o valor de tamanho de nó mínimo especificado pelo usuário, o nó não será dividido;

5. Se a divisão de um nó resultar em um nó filho cujo tamanho de nó seja menor que o valor de tamanho de nó filho mínimo especificado pelo usuário, os nós filhos que tiverem muito poucos casos (em comparação com esse mínimo) se fundirão com o nó filho mais semelhante, medido pelos maiores dos valores de *p.* No entanto, se o número resultante de nós filho for 1, o nó não será dividido.

5.3.4 Algoritmo *CHAID EXAUSTIVO*

Assim como no algoritmo *CHAID*, o algoritmo *CHAID Exaustivo* permite múltiplas divisões de um nó. Especificamente são constituídos de três etapas: fusão, divisão e paragem. Uma árvore é cultivada repetidas vezes usando as três etapas em cada nó a partir do nó raiz [45].

As etapas de divisão e paragem seguem os mesmos conceitos do algoritmo *CHAID*. Contudo, a etapa de fusão utiliza um procedimento de busca exaustivo para mesclar qualquer par semelhante até que apenas um único par permaneça.

Assim como no algoritmo *CHAID*, apenas os preditores categóricos nominais ou ordinais são permitidos. Os preditores contínuos são transformados primeiro em preditores ordinais.

Segue a etapa fusão do algoritmo *CHAID Exaustivo*.

5.3.4.1 Fusão

A fusão consiste em 9 passos, que são definidos a seguir [45]:

1. Se X tem apenas uma categoria, defina 1 para o valor de *p;*

2. Definir índice = 0. Calcule o valor de p com base no conjunto de categorias de X neste momento. Chamar o valor de *p(índice) = p(0);*

3. Caso contrário, encontre o par permissível de categorias de X que seja pelo menos significativamente diferente (isso é, mais semelhante). Isto pode ser determinado pelo par cuja estatística de teste dá o maior valor de *p* com respeito à variável dependente Y;

4. Mesclar o par que dá o maior valor de *p* em uma categoria composta;

5. (Opcional). Se a categoria for composta e conter três ou mais categorias originais, procure uma divisão binária dessa categoria composta que dê o menor valor de p. Se esse valor de p for maior do que aquele na formação da categoria de compostos por mesclagem na etapa anterior, execute a divisão binária nessa categoria composta;

6. Atualize o índice = índice + 1, calcule o valor p com base no conjunto de categorias de X neste momento. Denote *p(índice)* como o valor de *p;*

7. Repita os passos 3 a 6 até que apenas duas categorias permaneçam. Em seguida, entre todos os índices, encontrar o conjunto de categorias tais que *p(índice)* é o menor;

8. (Opcional). Qualquer categoria que tenha muito poucas observações (em comparação com um tamanho de segmento mínimo especificado pelo usuário) é mesclada com a categoria mais similar, medida pelo maior valor de p;

9. O valor de p ajustado é calculado aplicando os ajustes de *Bonferroni* que serão discutidos na seção seguinte.

5.3.5 Ajustes de *Bonferroni*

O valor de *p* ajustado é calculado multiplicando-o por um multiplicador de *Bonferroni*. O multiplicador de *Bonferroni* ajusta para testes múltiplos.

No algoritmo *CHAID*, suponhamos que uma variável preditora tenha originalmente categorias *I* e seja reduzida para *r* categorias após a etapa de fusão. O multiplicador *B* de *Bonferroni* é o número de maneiras possíveis que as categorias *I* podem ser fundidas em nas categorias *r*. Para $r = I$, $B = 1$. Para $2 \leq r < I$, use a seguinte equação apresentada na Figura 11.

$$B = \begin{cases} \binom{I-1}{r-1} & \text{Preditora ordinal} \\ \sum_{v=0}^{r-1} (-1)^v \frac{(r-v)^I}{v!(r-v)!} & \text{Preditora nominal} \\ \binom{I-2}{r-2} + r\binom{I-2}{r-1} & \text{Ordinal com uma categoria ausente} \end{cases}.$$

Figura 11 – Ajuste de Bonferroni do Algoritmo CHAID.

Já para algoritmo *CHAID Exaustivo* de forma iterativa ocorre a fusão das categorias semelhantes, até que restem apenas uma categoria. O multiplicador *B* de *Bonferroni* é a soma do número de formas possíveis de fundir duas categorias em cada iteração.

$$B = \begin{cases} \dfrac{I(I-1)}{2} & \text{Preditora ordinal} \\ \dfrac{I(I^2-1)}{2} & \text{Preditora nominal} \\ \dfrac{I(I-1)}{2} & \text{Ordinal com uma categoria ausente} \end{cases}.$$

Figura 12 – Ajuste de Bonferroni do Algoritmo CHAID Exaustivo.

Se a variável dependente de um caso estiver ausente, ela não será usada na análise. Se todas as variáveis de previsão de um caso estiverem ausentes, este caso será ignorado. Se o peso do caso estiver ausente, zero ou negativo, o caso será ignorado. Se o peso de frequência estiver ausente, zero ou negativo, o caso será ignorado.

Caso contrário, os valores em falta serão tratados como uma categoria de previsão. Para preditores ordinais, o algoritmo primeiro gera o "melhor" conjunto de categorias usando todas as informações não-faltantes dos dados. Em seguida, o algoritmo identifica a categoria que é mais semelhante à categoria ausente. Finalmente, o algoritmo decide se deve unir a categoria ausente com a sua categoria mais semelhante ou manter a categoria ausente como uma categoria separada. São calculados dois valores de *p,* um para o conjunto de categorias formado pela fusão da categoria em falta com a sua categoria mais semelhante e o outro pelo conjunto de categorias formado pela adição da categoria em falta como categoria separada. Tome a ação que dá o menor valor de *p.*

Para os preditores nominais, a categoria ausente é tratada da mesma forma que outras categorias na análise.

5.4 *IBM SPSS Statistics*

IBM® SPSS® Statistics é um software de análise estatística que fornece os principais recursos necessários para executar um processo de análise do início ao fim, com uma interface gráfica que o torna de fácil compreensão.

O precursor das versões atuais do *SPSS* foi desenvolvido no final da década de 1960. Em 1968, três estudantes de pós-graduação da *Universidade de Stanford*, *Norman H. Nie, C. Hadlai Hull* e *Dale Bent* desenvolveram sua primeira versão do *SPSS. SPSS* não era compatível para com computadores pessoais até 1984. Em 1984, o *SPSS*, que era uma entidade corporativa crescente sediada em Chicago, lançou o *SPSS* para computadores pessoais [48], popularizando seu uso entre os pesquisadores de diversas áreas.

Algumas visões da ferramenta, na Figura 13.

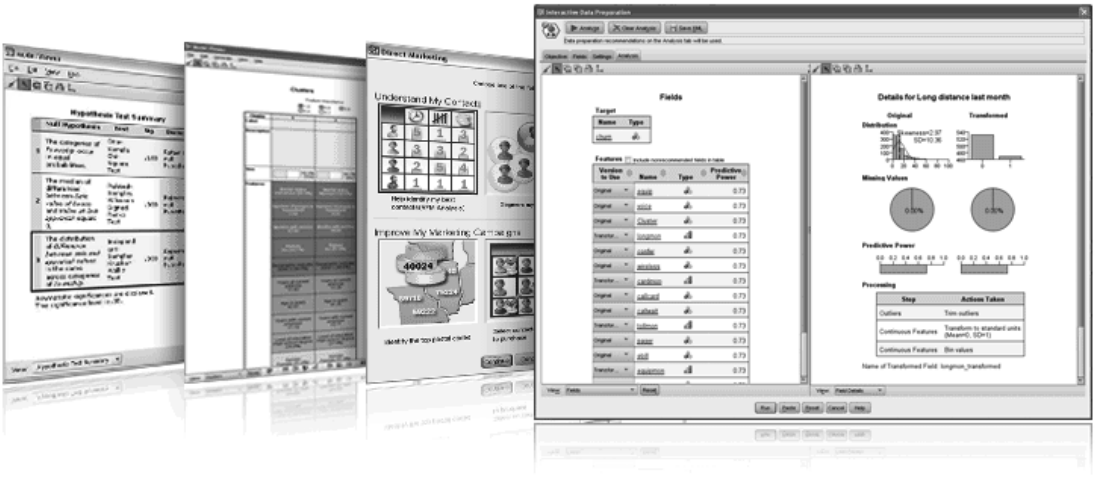

Figura 13 – Visões da ferramenta SPSS Statistics.

Como possui suporte aos algoritmos de arvore de decisão CART, CHAID e CHAID Exaustivos, o SPSS, se torna ideal para desenvolvimento desse trabalho.