UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS DA SAÚDE
PROGRAMA DE PÓS-GRADUAÇÃO EM ODONTOLOGIA

Fernanda Pretto Zatt

Utilização de *machine learning* na classificação de disfunções temporomandibulares: prova de conceito

Florianópolis
2023

Fernanda Pretto Zatt

Utilização de *machine learning* na classificação de disfunções temporomandibulares: prova de conceito

Dissertação submetida ao Programa de Pós-Graduação em Odontologia da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Mestra em Odontologia.

Orientador: Prof. Ricardo Armini Caldas, Dr.

Florianópolis
2023

Zatt, Fernanda Pretto
Utilização de machine learning na classificação de disfunções temporomandibulares: prova de conceito / Fernanda Pretto Zatt ; orientador, Ricardo Armini Caldas, 2023.
85 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro de Ciências da Saúde, Programa de Pós-Graduação em Odontologia, Florianópolis, 2023.

Inclui referências.

1. Odontologia. 2. Aprendizado de Máquina. 3. Disfunção Temporomandibular. 4. Diagnóstico. 5. Inteligência Artificial. I. Caldas, Ricardo Armini. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Odontologia. III. Título.

Fernanda Pretto Zatt

Utilização de machine learning na classificação de disfunções temporomandibulares: prova de conceito

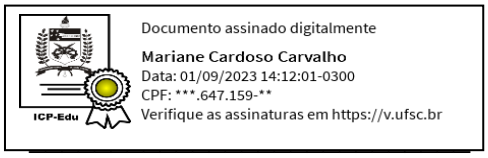
O presente trabalho em nível de Mestrado foi avaliado e aprovado, 12 de julho de 2023 pela banca examinadora composta pelos seguintes membros:

Prof. Ricardo Armini Caldas, Dr.
Universidade Federal de Santa Catarina

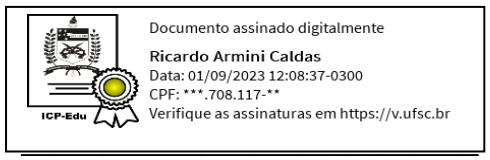
Prof. Gabriela Modesti Vedolin, Dr.ª
-

Prof. Marcio Holsbach Costa, Dr
Universidade Federal de Santa Catarina

Certificamos que esta é a versão original e final do trabalho de conclusão que foi julgado adequado para obtenção do título de Mestra em Odontologia



Prof. Mariane Cardoso, Drª.
Coordenadora do Programa de Pós-Graduação



Prof. Ricardo Armini Caldas, Dr.
Orientador

Florianópolis, 2023.

Este trabalho é dedicado aos meus pais, que têm sido os maiores incentivadores na busca pela realização dos meus sonhos

AGRADECIMENTOS

Gostaria de expressar minha gratidão não somente às pessoas envolvidas nesta dissertação, mas também a todos aqueles que me forneceram apoio, experiências e oportunidades valiosas ao longo do meu mestrado.

Agradeço primeiramente a **Deus** por todas as oportunidades concedidas a mim e pela força e nos momentos de fraqueza e dificuldades.

Agradeço aos meus queridos pais, **Ivânia e Neimar**, que me criaram com amor, respeito e dedicação. Esta conquista só foi possível graças a eles. Agradeço por todas as orações e palavras acolhedoras de minha mãe e as longas conversas sobre ética e educação com meu pai. Tenho certeza de que tudo isso me fortaleceu e me ajudou a superar cada desafio ao longo do caminho. Agradeço ao melhor presente que meus pais me deram, meu irmão **Augusto,** que é meu suporte e motivação para enfrentar e aceitar todos os desafios da vida. Agradeço ao **Felipe,** por todo amor, carinho, compreensão e apoio desde o início da minha vida acadêmica.

Agradeço ao meu orientador, **Professor Ricardo Armini Caldas,** pelos ensinamentos e por sua dedicação que o fez, por muitas vezes, abdicar de seus momentos de descanso para me orientar. Além de professor, foi alguém que me ensinou muito sobre dedicação, acolhimento e valores. Sem o seu apoio e amizade, não apenas este trabalho, mas todo o trajeto percorrido até o momento, não teriam sido possíveis. Agradeço ao **Professor Victor Emanoel Armini Caldas,** pela disposição em instruir e auxiliar na elaboração deste trabalho. Agradeço à **Professora Beatriz Mendes** e ao **CEMDOR** por fornecer os prontuários necessários pra a realização deste trabalho.

Agradeço aos meus amigos e colegas **Melissa, Aurélio e Lucas** que sempre estiveram ao meu lado nas conquistas e dificuldades do mestrado. Sem eles, não seria possível passar por essa jornada com a mesma leveza e alegria. Sou grata pelos incontáveis conselhos e exemplos dados por eles. Agradeço as minhas amigas **Thaís** e **Diandra** pelo apoio desde o início da graduação. Mesmo diante de toda a distância imposta pelas nossas escolhas pessoais, elas são incentivo para seguir os meus sonhos. Agradeço as pessoas que me acolheram desde o primeiro dia na Universidade Federal de Santa Catarina, **Helena** e **Ana Paula,** que sempre foram solícitas e empáticas ao meu auxiliar em toda e qualquer dificuldade, fosse ela pessoal ou profissional.

Agradeço imensamente à **banca examinadora**, por aceitarem contribuir com este trabalho. Agradeço ao **Prof. Dr. Giancarlo De La Torre** e ao **Prof. Dr. Marcio Holsbach Costa** por suas valorosas contribuições em minha banca examinadora de qualificação de projeto. Ainda, agradeço a banca examinadora da defesa de mestrado, **Prof.ª Dr.ª Gabriela Modesti Vedolin, Prof. Dr. Marcio Holsbach Costa** e **Prof. Dr. Rafael Pino Vitti** pela disponibilidade, atenção e contribuição.

Agradeço à **Universidade Federal de Santa Catarina** por viabilizar um mestrado de qualidade em um ambiente de ensino público. Agradeço a **Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES)** pelo apoio financeiro ao longo do mestrado.

“A tarefa não é tanto ver aquilo que ninguém viu, mas pensar o que ninguém ainda pensou sobre aquilo que todo mundo vê.”

(Arthur Schopenhauer)

RESUMO

O interesse mundial no uso da inteligência artificial está crescendo rapidamente devido aos avanços substanciais na capacidade de computação e novos algoritmos de aprendizagem. Estes recursos podem trazer maior precisão diagnóstica, respostas mais rápidas e melhores resultados na assistência ao indivíduo. Este projeto teve por objetivo criar um modelo de aprendizado de máquina *(machine learning)* que auxiliem na detecção e classificação de disfunções temporomandibulares (DTM) usando para dados sistematizados. A etapa de *machine learning* foi por meio do método de árvore de decisão, onde foi utilizada programação em linguagem Python/Scikit-Learn. Para a construção do modelo, foi utilizado o conjunto de dados de prontuários do acervo de prontuários clínicos do Centro Multidisciplinar de Dor Orofacial (CEMDOR) da Universidade Federal de Santa Catarina. Cada conjunto de dados possuía uma quantidade padronizada de informações básicas e dimensões para cada informação. Os diagnósticos de DTM foram classificados de acordo com a Classificação Internacional de Dor Orofacial (ICOP). Foram realizados dois experimentos independentes utilizando árvores de decisão para classificar dores miofasciais e dores articulares. Ambos os experimentos utilizaram as mesmas métricas para avaliar o desempenho dos modelos. Obteve-se acurácia de 78% para dores miofasciais e 83% para dores articulares. O desempenho dos modelos foi avaliado através de matriz de confusão e métricas adicionais para entendimento do funcionamento da árvore. Concluiu-se que essa a utilização de árvore de decisão por *machine learning* apresenta potencial relevante para auxílio no diagnóstico clínico das DTM.

Palavras-chave: Aprendizado de Máquina; Transtornos da Articulação Temporomandibular; Inteligência Artificial; Diagnóstico.

ABSTRACT

Worldwide interest in the use of artificial intelligence is growing rapidly due to substantial advances in computing power and new learning algorithms. These resources can bring greater diagnostic accuracy, faster responses, and improved outcomes in assisting the individual. This project aimed to create a machine learning model to aid in the detection and classification of temporomandibular dysfunctions (TMD) using for systematized data. The machine learning step was through the decision tree method, where Python/Scikit-Learn language programming was used. To build the model, the dataset of medical records from the Clinical Records collection of the Multidisciplinary Center for Orofacial Pain (CEMDOR) at the Federal University of Santa Catarina was used. Each dataset had a standardized amount of basic information and dimensions for each piece of information. TMD diagnoses were classified according to the International Classification of Orofacial Pain (ICOP). Two independent experiments were conducted using decision trees to classify myofascial pain and joint pain. Both experiments used the same metrics to evaluate model performance. Accuracy was 78% for myofascial pain and 83% for joint pain. The performance of the models was evaluated through a confusion matrix and additional metrics for understanding how the decision tree works. It was concluded that the use of machine learning decision trees has a relevant potential to aid in the clinical diagnosis of TMD.

Keywords: Machine Learning; Temporomandibular Joint Disorders; Artificial Intelligence; Diagnosis.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ATM	Articulação Temporomandibular
CEMDOR	Centro Multidisciplinar de Dor Orofacial
CEPSH	Comitê de Ética em Pesquisa com Seres Humanos
DC/TMD	Critério Diagnóstico em Disfunção Temporomandibular
DTM	Disfunção Temporomandibular
IA	Inteligência Artificial
ICHD	Classificação internacional de cefaleias
ICOP	Classificação Internacional de Dor Orofacial
RDC/TMD	Critério Diagnóstico para Pesquisa em Disfunção Temporomandibular
UFSC	Universidade Federal de Santa Catarina

SUMÁRIO

1 FUNDAMENTAÇÃO TEÓRICA

1.1 INTRODUÇÃO

O diagnóstico correto possui papel fundamental no campo da saúde e é um requisito para garantir um tratamento eficaz e de qualidade. O diagnóstico ideal refere-se ao processo de obter uma explicação precisa da condição do paciente, com valor clínico, a fim de oferecer opções de tratamento adequadas (YANG; FINEBERG; COSBY, 2021). No entanto, de acordo com o relatório "*Improving Diagnosis in Health Care*" do *Institute of Medicine* nos Estados Unidos, os erros de diagnóstico persistem em todos os ambientes de cuidados de saúde, no qual é relatado que pelo menos 30% da população já teve uma experiência com algum diagnóstico incorreto (BALOGH; MILLER; BALL, 2015). Assim, métodos auxiliares de diagnóstico são cada vez mais utilizados, com aumento considerável da inteligência artificial (IA).

O interesse global no uso da IA está crescendo rapidamente devido à disponibilidade de grandes conjuntos de dados, avanços significativos na capacidade de computação e novos algoritmos de *machine learning*, esses recursos estão proporcionando maior precisão diagnóstica, respostas mais rápidas e melhores resultados para os pacientes (RAJKOMAR; DEAN; KOHANE, 2019; THRALL et al., 2018). A aplicação da IA não se limita apenas à pesquisa acadêmica e se estende para o campo comercial, incluindo diversas áreas da odontologia. Esse avanço tem sido impulsionado pela digitalização da odontologia, atendendo à crescente demanda por soluções inovadoras (MACHOY et al., 2020). A aplicabilidade pode se estender ao atendimento em emergências odontológicas até o diagnóstico diferencial de lesões bucais, interpretação de radiografias, análise do crescimento facial durante tratamentos ortodônticos, planejamento e simulação de restaurações ideais para pacientes específicos e apresenta resultados promissores para a detecção de disfunções temporomandibulares (DTM)(CARRILLO-PEREZ et al., 2022; FAROOK et al., 2021; JORDAN; MITCHELL, 2015; KOSE et al., 2023; REDA et al., 2023; SCHWENDICKE; SAMEK; KROIS, 2020).

As DTM englobam um conjunto diversificado de condições que afetam os músculos mastigatórios, a articulação temporomandibular (ATM) e suas estruturas associadas (DE LEEUW; KLASSER, 2015). Essas condições podem apresentar uma ampla gama de sinais e sintomas que podem variar desde uma única condição isolada

até o envolvimento de múltiplos sistemas. Essa complexidade torna o diagnóstico das DTM um desafio que traz grande possibilidade de erro de diagnóstico, principalmente para profissionais não especialistas na área.(CALIL et al., 2020; CHOI et al., 2021; FAROOK et al., 2021; ORHAN et al., 2021; REDA et al., 2023). Embora a responsabilidade pelo diagnóstico e tratamento das DTM recaia principalmente sobre os dentistas, é importante notar que poucos profissionais têm recebido treinamento intensivo nessa área específica. Apesar do aumento exponencial de cursos de aperfeiçoamento e especialização em DTM e dor orofacial (MACHADO; LIMA; CONTI, 2014), de fato, apenas 1,1% do número total de cirurgiões-dentistas no Brasil possuem esse treinamento especializado (CONSELHO FEDERAL DE ODONTOLOGIA, 2023).

A classificação das DTM é geralmente baseada em um consenso internacional entre especialistas (ICOP, 2020; SCHIFFMAN et al., 2014) e envolve uma avaliação abrangente do paciente que considera diversos fatores. Muitas vezes essas condições não são identificadas ou recebem pouca atenção nos cuidados de saúde bucal. Isso pode ser observado pela diferença entre a quantidade de tratamento necessário estimada e o tratamento efetivamente realizado (LÖVGREN et al., 2016). A literatura apresenta evidências das deficiências no conhecimento dos cirurgiões-dentistas em relação ao manejo de pacientes com DTM (AGGARWAL et al., 2011; AL-HURAISHI et al., 2020; HADLAQ et al., 2019). Além disso, estudos têm demonstrado que uma proporção significativa desses profissionais carece de experiência e habilidades adequadas para realizar procedimentos diagnósticos e terapêuticos relacionados à DTM. De fato, a pesquisa revelou que 41,9% dos cirurgiões-dentistas consideram sua própria qualificação como um impedimento para o diagnóstico e tratamento de DTM, sendo essa percepção mais prevalente entre os profissionais com menos tempo de formação (REISSMANN et al., 2015). Uma possível explicação para isso é a insuficiente formação recebida durante os cursos de graduação, o que pode afetar a habilidade de dentistas jovens e inexperientes em identificar as DTM (LÖVGREN et al., 2016; REISSMANN et al., 2015). Sendo assim, a integração da IA no diagnóstico de DTM tem potencial para proporcionar apoio para profissionais não especialistas, resultando em uma prática odontológica mais eficaz (CHOI et al., 2021; FAROOK et al., 2021; ORHAN et al., 2021; REDA et al., 2023).

Dentre os diversos métodos de IA, o que apresenta funcionamento similar ao raciocínio humano na tomada de decisões é o algoritmo de árvore de decisão por

aprendizado de máquina. As árvores de decisão constituem um grupo de classificadores que realizam a classificação com critérios compreensíveis e permitem que tenha sua lógica avaliada por especialista (LOYOLA-GONZALEZ, 2019). Sob a perspectiva da descoberta de conhecimento, a habilidade de monitorar e avaliar cada etapa do processo de tomada de decisão emerge como um fator importante para a confiança de métodos auxiliares para diagnósticos na área da saúde (STIGLIC et al., 2012).

1.2 REVISÃO DE LITERATURA

1.2.1 Disfunções temporomandibulares

As DTM englobam um conjunto diversificado de condições que afetam os músculos mastigatórios, a ATM e suas estruturas associadas(LEEUW; KLASSER, 2015), sendo assim classificadas em disfunções de origem articular, relacionadas à ATM e disfunções de origem muscular, relacionadas à musculatura estomatognática (BENDER, 2012). Essas condições apresentam uma variedade de sinais e sintomas difusos podendo variar desde uma única condição isolada até o envolvimento de múltiplos sistemas, além de estarem potencialmente associadas a outros distúrbios sistêmicos (THOMAS et al., 2023). Essa complexidade torna o diagnóstico das DTM difícil e multifatorial, requerendo frequentemente uma abordagem interdisciplinar e multiprofissional no cuidado aos pacientes (BOND et al., 2020; DUBNER; OHRBACH; DWORKIN, 2016).

A prevalência de DTM é de aproximadamente 5% a 12% da população em geral (NATIONAL INSTITUTE OF DENTAL AND CRANIOFACIAL RESEARCH, 2018),porém, se todos os sintomas que envolvem a DTM forem considerados, a prevalência aumenta para até 50% da população adulta MANFREDINI et al., 2011). As ferramentas de classificação amplamente reconhecidas para o diagnóstico das DTM são o Critério Diagnóstico para Pesquisa em Disfunção Temporomandibular (RDC/DTM), o Critério Diagnóstico em Disfunção Temporomandibular (DC/DTM) e, mais recentemente, a Classificação Internacional de Dor Orofacial (ICOP). Essas abordagens têm como objetivo aumentar a consistência nos estudos, permitindo a padronização e a replicação dos resultados tanto no contexto clínico quanto na pesquisa ((DWORKIN; LERESCHE, 1992; ICOP, 2020; SCHIFFMAN et al., 2014).

O RDC/TMD (DWORKIN; LERESCHE, 1992) que foi revisado e aprimorado, tornando-se DC/TMD (SCHIFFMAN et al., 2014) é baseado no consenso de agrupamentos de sinais e sintomas bem caracterizados (SVENSSON; BENDIXEN, 2018). O DC/TMD, foi desenvolvido por examinadores altamente calibrados e com treinamento contínuo, assim, embora represente um avanço em relação à RDC/TMD, sua implementação imediata na pesquisa e na assistência clínica ainda não parece estar devidamente fundamentada (STEENKS; TÜRP; DE WIJER, 2018).

A Classificação Internacional de Dor Orofacial (ICOP), é a primeira classificação abrangente dedicada exclusivamente à dor orofacial. Essa classificação resultou de uma colaboração internacional envolvendo várias organizações, como o Grupo de Interesse Especial em Dor Orofacial e Cefálica da Associação Internacional para o Estudo da Dor, a Rede Internacional de Metodologia de Dor Orofacial e Distúrbios Relacionados da Associação Internacional para Pesquisa Odontológica, a Academia Americana de Dor Orofacial e a Sociedade Internacional de Cefaleia. A ICOP foi modelada na estrutura da Classificação Internacional de Cefaleias (ICHD), que é amplamente aceita e utilizada globalmente por médicos e pesquisadores (PIGG et al., 2021).

A ICOP fornece uma descrição abrangente das condições de dor que afetam a região orofacial, além de estabelecer critérios diagnósticos para cada uma delas. Os seis capítulos da ICOP abrangem dor em tecidos dentoalveolares e em estruturas anatomicamente relacionadas, dor muscular, dor na ATM, dor neuropática que afeta os nervos cranianos, dor semelhante a dores de cabeça primárias e dor idiopática. Essa classificação inclui tanto a dor primária, que não é atribuível a outro distúrbio específico, quanto a dor secundária, causada por outro distúrbio identificado, como inflamação, sensibilização dos tecidos, alterações estruturais, espasmo muscular ou lesão. Além disso, vários capítulos subdividem as condições de dor com base na duração, distinguindo entre tipos agudos e crônicos, sendo a dor crônica definida como aquela com duração de três meses ou mais (ICOP, 2020).

O capítulo “2. Dor miofascial orofacial” (ICOP, 2020) classifica a dor miofascial orofacial como dor nos músculos mastigatórios, com ou sem comprometimento funcional, atribuída ou não a outra disfunção. Existem duas categorias principais: dor miofascial orofacial primária e dor miofascial orofacial secundária. Dentro da categoria primária, há subdivisões que se referem à natureza e à duração da dor. A dor miofascial orofacial primária aguda é caracterizada pela ocorrência recente da dor, enquanto a dor miofascial orofacial primária crônica é uma dor persistente ao longo do tempo. A dor crônica pode ser classificada como infrequente ou frequente, dependendo da sua recorrência. Dentro da dor miofascial orofacial primária crônica frequente, existem duas subcategorias. A primeira é a dor miofascial orofacial primária crônica frequente sem dor referida, o que significa que a dor está localizada apenas na região afetada. A segunda é a dor miofascial orofacial primária crônica frequente com dor referida, o que indica que a dor pode se espalhar para outras áreas além da

região afetada. Além disso, há a dor miofascial orofacial primária crônica altamente frequente, que é caracterizada por uma dor persistente e intensa. Assim como na categoria anterior, essa dor pode ser sem dor referida ou com dor referida, dependendo se ela se espalha para outras áreas do corpo. Na categoria secundária, a dor miofascial orofacial é atribuída a condições específicas, como tendinite, miosite e espasmo muscular. Isso significa que a dor muscular na região da face e da boca é resultado dessas condições.

Da mesma forma, a ICOP classifica a dor na ATM no capítulo “3. Dor na articulação temporomandibular (ATM)” (ICOP, 2020) em diferentes categorias e subcategorias, também em dor primária e dor secundária. A dor primária pode ser aguda ou crônica. A dor primária aguda refere-se a episódios de dor recentes e de curta duração na ATM. Por outro lado, a dor primária crônica é uma condição de longa duração e pode ser subdividida em crônica infrequente, frequente e altamente frequente. Dentro da categoria de dor primária crônica, existem mais subdivisões. A dor primária crônica infrequente ocorre de forma esporádica e irregular na ATM. Já a dor primária crônica frequente ocorre com maior frequência e pode ser classificada em duas subcategorias: com ou sem dor referida. A dor primária crônica frequente sem dor referida significa que a dor está localizada apenas na ATM, sem se irradiar para outras regiões. Por outro lado, a dor primária crônica frequente com dor referida indica que a dor pode se espalhar para outras áreas além da ATM. A última subdivisão da dor primária crônica é a dor primária crônica altamente frequente. Essa categoria indica uma dor persistente e de alta frequência na ATM. Assim como na dor primária crônica frequente, essa condição pode ser classificada em duas subcategorias: com ou sem dor referida. Além da dor primária, há também a dor secundária na ATM. Essa categoria está relacionada a condições subjacentes que causam a dor na ATM, que incluem artrite, deslocamento de disco, doença articular degenerativa e subluxação.

Ainda, a dor secundária atribuída à artrite pode ser subdividida em artrite não sistêmica e artrite sistêmica e a dor secundária na ATM atribuída ao deslocamento de disco que pode ser classificada em duas subcategorias: deslocamento de disco com redução e deslocamento de disco sem redução. O deslocamento de disco com redução pode ser acompanhado de travamento intermitente, em que a mandíbula pode ficar temporariamente presa no lugar. A doença articular degenerativa também pode ser uma causa de dor secundária na ATM. Por fim, a subluxação é outra condição que pode levar à dor na ATM (ICOP, 2020).

1.2.2 Aprendizado de máquina

A IA é um campo que estuda técnicas para fazer com que os computadores ajam como humanos, tomando decisões independentes ou com pouca intervenção humana. Isso envolve lidar com problemas como representação do conhecimento, raciocínio, aprendizado, planejamento, percepção e comunicação, utilizando diferentes ferramentas e métodos, como raciocínio baseado em casos, sistemas baseados em regras, algoritmos genéticos, modelos difusos e sistemas multiagentes. Anteriormente, a pesquisa em IA focava em escrever declarações em linguagens formais, nas quais um computador poderia raciocinar automaticamente usando regras lógicas. No entanto, essa abordagem tem limitações, pois os humanos muitas vezes têm dificuldade em explicar todo o conhecimento implícito necessário para realizar tarefas complexas (JANIESCH; ZSCHECH; HEINRICH, 2021).

O aprendizado de máquina (*machine learning)* supera as limitações dos métodos tradicionais, permitindo que os sistemas adquiram conhecimento e resolvam problemas de forma automatizada. Utilizando algoritmos que analisam e processam iterativamente os dados de treinamento, o *machine learning* descobre padrões e *insights* ocultos sem a necessidade de programação explícita (JORDAN; MITCHELL, 2015). Isso permite a criação de sistemas inteligentes que aprendem por conta própria, economizando tempo e esforço no desenvolvimento de sistemas tradicionais. O algoritmo faz previsões com base nos dados conhecidos, ajustando seus parâmetros para se aproximar do valor correto, até que o erro entre os valores reais e previstos estabilize. No final, obtém-se um modelo capaz de prever valores para dados desconhecidos (JANIESCH; ZSCHECH; HEINRICH, 2021)

Um dos métodos mais populares e amplamente utilizados em *machine learning* em mineração de dados é a árvore de decisão (JORDAN; MITCHELL, 2015). A utilização difundida das árvores de decisão se deve à sua abordagem intuitiva para fornecer resultados e ao seu funcionamento similar ao raciocínio formal de um especialista. Dessa forma, o desenvolvimento de métodos baseados em árvores de decisão é uma área de pesquisa interessante e relevante no campo da análise de dados (HERNÁNDEZ et al., 2022; LOYOLA-GONZALEZ, 2019). Uma árvore de decisão é um modelo hierárquico utilizado para tomar decisões e diferentes algoritmos podem ser utilizados na construção das árvores de decisão, dentre eles o CART

(árvores de classificação e regressão ou, do inglês, *Classification and Regression Trees*). Nele, são construídas árvores binárias (cada nó possui dois nós-filhos) cujos nós são produzidos de acordo com o maior ganho de informação através de diferentes critérios utilizados como parâmetro (HERNÁNDEZ et al., 2022).

Um conjunto de dados de treinamento é usado para estabelecer uma estrutura de árvore, em que cada nó está associado a um recurso ou traço de dados e cada ramificação representa uma decisão viável. A construção da árvore prossegue através de um processo de particionamento recursivo, onde a cada passo as variáveis mais relevantes são selecionadas dependendo de medidas de pureza como ganho de informação ou entropia (LOYOLA-GONZALEZ, 2019). O processo é repetido até que classificações adequadas dos exemplos de treinamento sejam obtidas ou os critérios de finalização sejam atendidos. Depois que a árvore é refinada, as características dos dados são usadas para fazer previsões ou tomar decisões com base nas características de novas amostras que ainda não foram vistas antes (ROKACH; MAIMON, 2008).

Uma vantagem adicional da construção de árvores de decisão é a sua simplicidade e o fato de que o processo de classificação é geralmente mais rápido em comparação com outras técnicas de classificação. Além disso, as regras geradas pela árvore de decisão podem ser usadas diretamente como instruções em uma linguagem de acesso a bancos de dados (STIGLIC et al., 2012). Quando se utiliza árvores de decisão para descoberta de conhecimento, é comum envolver especialistas do domínio no processo de análise. Essas árvores de decisão finais são, então, apresentadas aos especialistas para avaliação do conhecimento extraído, ou seja, das regras que podem ser derivadas a partir da árvore.

Diversos estudos têm abordado a complexidade das árvores de decisão, visando reduzi-la e ao mesmo tempo manter ou aprimorar a precisão. O tamanho da árvore está fortemente relacionado ao tamanho do conjunto de treinamento. Portanto, várias abordagens que envolvem a remoção de instâncias de treinamento antes da construção das árvores podem resultar em árvores menores apenas devido à redução do conjunto de treinamento utilizado(GARCIA LEIVA et al., 2019; HERNÁNDEZ et al., 2022).

2 JUSTIFICATIVA

Embora as DTM possam ter um impacto negativo na vida cotidiana, parece que essas condições não são detectadas no seu total. A crescente demanda de pacientes com DTM e a dificuldade de realizar um diagnóstico preciso por parte de cirurgiões-dentistas não especializados tornam a integração da IA no processo de diagnóstico de DTM uma potencial solução para reduzir as disparidades diagnósticas associadas a essa condição. Essa abordagem tem o potencial de trazer benefícios significativos levando a tratamentos mais adequados aos pacientes e auxiliando os profissionais de saúde a estabelecer um diagnóstico preciso.

3 OBJETIVOS

3.1 GERAL

Criar um modelo de aprendizado de máquina *(machine learning)* com capacidade para classificar as DTM com base no ICOP.

3.2 ESPECÍFICOS

- Avaliar a acurácia do modelo;
- Avaliar sensibilidade e especificidade do modelo;
- Identificar quais critérios são mais importantes para as tomadas de decisões do modelo;
- Aprimorar os métodos de diagnóstico em DTM.

4 METODOLOGIA

4.1 ASPECTOS ÉTICOS

O presente estudo foi aprovado pelo Comitê de Ética em Pesquisa com Seres Humanos da Universidade Federal de Santa Catarina (CEPSH-UFSC) sob Certificado de Apresentação para Apreciação Ética nº. 60532822.8.0000.0121 (ANEXO A – , p. 80).

4.2 AQUISIÇÃO DA AMOSTRA

As análises foram realizadas utilizando o acervo de prontuários clínicos dos últimos 5 anos do Centro Multidisciplinar de Dor Orofacial (CEMDOR) da Universidade Federal de Santa Catarina. O CEMDOR é um centro de referência em dor orofacial, que recebe pacientes encaminhados de diversas instituições públicas e privadas do estado de Santa Catarina. As informações coletadas dos prontuários foram utilizadas de forma anônima, preservando a identificação dos pacientes. Assim, um operador realizou a triagem dos prontuários e registrou as informações em uma base de dados. Foram incluídos prontuários de pacientes com diagnóstico de DTM, que possuíam a documentação completa do atendimento do CEMDOR, com letra legível nas partes subjetivas que eram descritas com texto corrido. Pacientes menores de 18 anos e aqueles com prontuários clínicos incompletos foram excluídos da análise. Os dados coletados foram baseados nas informações presentes na ficha clínica do CEMDOR, conforme descrito no APÊNDICE A – Critérios Retirados Dos Prontuários Do C. Como resultado, obteve-se uma amostra de conveniência composta por 122 prontuários, dos quais foram extraídos os diagnósticos detalhados apresentados no Quadro 1.

Quadro 1. Diagnósticos dos prontuários do CEMDOR.

Diagnóstico	**Quantidade**
Dor miofascial	
Dor miofascial orofacial primária	31
Dor miofascial orofacial primária aguda	11
Dor miofascial orofacial primária crônica	80
Dor articular	
Dor articular primária	72
Dor articular secundária	34
Não-doença	16

Fonte: elaborado pelo autor

4.3 BIBLIOTECAS UTILIZADAS

A realização da etapa de *machine learning* foi realizada utilizando o método de árvore de decisão. Para tal, foi escolhida programação em linguagem Python (Python Software Foundation, Delaware, EUA) e as bibliotecas *Numpy, Pandas, Scikit-learn, Graphviz, Matplotlib* e *Seaborn*. Essas ferramentas são bibliotecas de código que fornecem uma série de funcionalidades para o desenvolvimento de software, como descrito no Quadro 2.

Quadro 2. Bibliotecas utilizadas para a linguagem de programação Python.

Biblioteca	Função
Numpy	Suporte para arrays multidimensionais e funções matemáticas de alto desempenho
Pandas	Estruturas de dados e ferramentas de análise de dados eficientes para manipulação e processamento de dados
Scikit-learn	Oferece algoritmos e ferramentas para tarefas de classificação, regressão, clustering e pré-processamento de dados para aprendizado de máquina
Graphviz	Criação e visualização de diagramas e gráficos em formato de árvore e grafos
Matplotlib	Criação de gráficos e visualizações de dados em Python
Seaborn	Biblioteca de visualização de dados baseada no Matplotlib

Fonte: elaborado pelo autor. Adaptado de https://awari.com.br/

4.4 CONSTRUÇÃO DO MODELO

Após a importação das bibliotecas necessárias para trabalhar com os dados, os dados dos prontuários em formato de arquivo CSV foram carregados em um objeto Data Frame do Pandas. Foram construídos dois modelos de árvore de decisão separados entre dor miofascial orofacial e dor na ATM, onde os passos e as análises descritas abaixo foram realizados individualmente para cada um dos modelos.

4.4.1 Separação da variável alvo das demais:

Em um modelo de *machine learning*, é necessário separar as variáveis independentes da variável alvo (ou variável dependente). As variáveis independentes foram utilizadas para prever a variável alvo. Elas foram responsáveis por fornecer informações que usadas pelo modelo para fazer suas previsões. Por outro lado, a

variável alvo é aquela que se pretende prever com base nas informações fornecidas pelas variáveis independentes.

4.4.2 Ajuste de parâmetros:

Anteriormente ao processo de treinamento, foram testadas várias combinações de valores para os parâmetros com o objetivo de aperfeiçoar a eficácia do modelo. Os parâmetros são configurações específicas que podem ser ajustadas de forma a maximizar o desempenho do modelo e para tal, analisou-se a acurácia das diferentes combinações dos parâmetros utilizados e foi selecionada aquela que apresentou o maior índice de acerto nos dados de treinamento para cada árvore. Nesse contexto, os parâmetros considerados foram o cálculo de impureza a ser utilizado, a profundidade da árvore e o número mínimo de amostras necessárias para que um nó seja considerado uma folha.

O cálculo de impureza para um nó pode ser Gini ou Entropia e é usado durante o treinamento do modelo para decidir como dividir o conjunto de dados em árvores de decisão. Através da entropia o algoritmo verifica como os dados estão distribuídos nas variáveis preditoras de acordo com a variação da variável target. É uma medida de incerteza ou desordem em um conjunto de dados que varia de 0 a 1, onde 0 representa pureza máxima (todos os elementos pertencem à mesma classe) e 1 representa incerteza máxima (os elementos estão igualmente distribuídos em todas as classes). A equação da entropia se dá por:

$$\boldsymbol{H(Q_m) = -\sum_{k} p_{mk} \log(p_{mk})}$$

Onde:

$\boldsymbol{H(Q_m)}$: impureza ou função de perda aplicada nos dados do nó $\boldsymbol{m}$.

$\boldsymbol{Q_m}$: Dados no nó $\boldsymbol{m}$ com $\boldsymbol{Nm}$ amostras.

$\boldsymbol{p_{mk}}$: proporção de observações de classes $\boldsymbol{k}$ no nó $\boldsymbol{m}$.

Quanto maior a entropia, maior a desordem dos dados; e quanto menor, maior será a ordem destes dados, quando analisados pela ótica da variável target. Partindo da entropia, o algoritmo confere o ganho de informação de cada variável. Aquela que apresentar maior ganho de informação será a variável do primeiro nó das árvores. O ganho de informação é uma medida que indica quão bem relacionadas estão as informações da variável preditora (ou variável independente) com os dados da variável

alvo (ou variável dependente), ou o quanto a variável target pode ser explicada a partir da variável preditora, sendo que a variável com melhor desempenho será a escolhida para iniciar a árvore.

Com o cálculo do índice Gini, assim como na Entropia, é verificada a distribuição dos dados nas variáveis preditoras de acordo com a variação da variável target, porém com um método diferente. O índice Gini mede a impureza de um conjunto de dados dividindo-o em classes. Ele varia de 0 a 1, onde 0 representa pureza máxima (todos os elementos pertencem à mesma classe) e 1 representa impureza máxima (os elementos são igualmente distribuídos em todas as classes). O cálculo do índice de Gini é baseado na probabilidade de um elemento escolhido aleatoriamente estar incorretamente classificado**.** A variável preditora com o menor índice Gini será a escolhida para o nó principal da árvore, pois um baixo valor do índice Sindica maior ordem na distribuição dos dados.

A equação do índice Gini se dá por:

$$\boldsymbol{H(Q_m) = \sum_{k} p_{mk}(1 - p_{mk})}$$

Onde:

$\boldsymbol{H(Q_m)}$: impureza ou função de perda aplicada nos dados do nó $\boldsymbol{m}$**.**

$\boldsymbol{Q_m}$**:** Dados no nó $\boldsymbol{m}$ com $\boldsymbol{Nm}$ amostras.

$\boldsymbol{p_{mk}}$: proporção de observações de classes $\boldsymbol{k}$ no nó $\boldsymbol{m}$.

A escolha entre utilizar o índice de Gini ou entropia geralmente depende da preferência ou do contexto do problema em questão. Em geral, a entropia tende a ser mais sensível a mudanças na distribuição de classes, enquanto o índice de Gini tende a ser mais sensível a erros de classificação. Ao escolher entre o índice de Gini e a entropia, pode ser levada em consideração a sensibilidade a *outliers*, o tempo de computação e a interpretabilidade, adaptando a escolha às necessidades e características específicas do problema de classificação em questão. O índice de Gini é menos sensível a *outliers* do que a entropia, tornando-o mais robusto em presença de valores extremos. Além disso, o índice de Gini é mais rápido de calcular do que a entropia, o que é vantajoso em termos de eficiência computacional. Por fim, o índice de Gini é mais fácil de interpretar intuitivamente, indicando a pureza ou homogeneidade dos dados

A profundidade da árvore refere-se ao número máximo de camadas que uma árvore de decisão pode ter. Essa técnica de controle limita o número de níveis ou

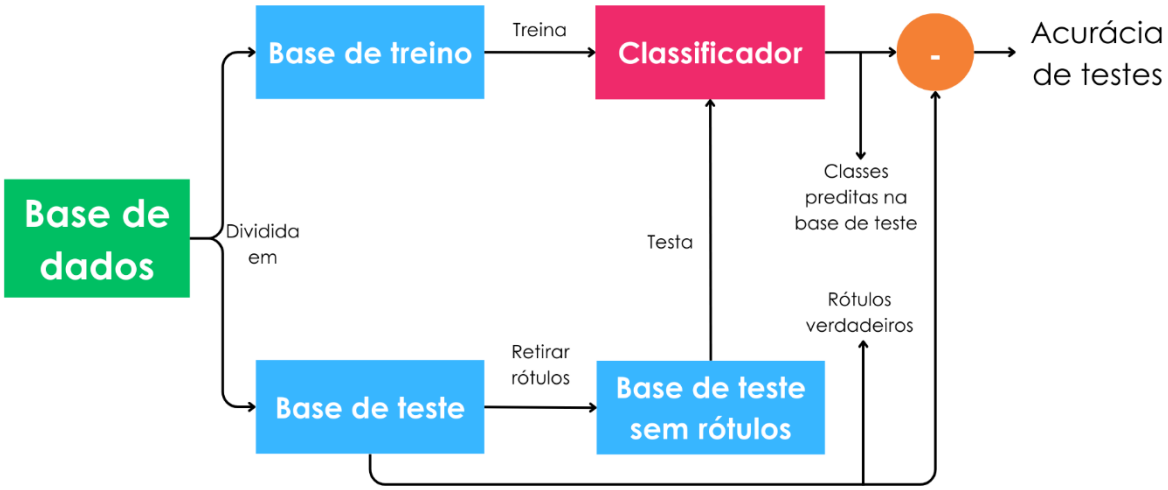
camadas da árvore. Cada nível representa uma pergunta ou um teste sobre os atributos do conjunto de dados para tomar uma decisão. A definição da profundidade máxima da árvore de decisão está relacionada a objetivos importantes como o controle do *overfitting*, a redução da complexidade da árvore e redução da quantidade de cálculos necessários para tomar uma decisão.

O número mínimo de amostras é um hiperparâmetro que determina o número mínimo de instâncias necessárias em um nó folha para que ele possa ser criado. Ao definir um número mínimo de amostras em um nó folha, se estabelece uma condição que impede a criação de nós folha que contenham um número muito baixo de amostras. Este hiperparâmetro ajuda a regular a complexidade do modelo e evitar o *overfitting*, permitindo que a árvore generalize melhor para dados não vistos.

4.4.3 Treinamento do modelo

Foi criada uma instância do DecisionTreeClassifier a partir da biblioteca scikit-learn. Essa instância representa o modelo de árvore de decisão que foi usado para realizar as previsões. Durante a criação do objeto, foram fornecidos os parâmetros previamente selecionados para o treinamento do modelo. Com o objeto DecisionTreeClassifier criado, o modelo foi treinado usando o conjunto de treinamento. Essa etapa envolve chamar o método "fit" do modelo e fornecer as variáveis independentes (características) e as variáveis alvo (rótulos ou diagnósticos) como argumentos. O método "fit" percorreu os dados de treinamento, construindo a árvore de decisão com base nas características e nos rótulos fornecidos. O modelo aprendeu a mapear as características para os rótulos correspondentes, ajustando os nós da árvore de decisão de acordo com as informações fornecidas pelos dados de treinamento. Ao final desse processo de treinamento, o modelo estava pronto para fazer previsões utilizando as características (variáveis independentes) de novos dados não vistos anteriormente. Esse fluxo pode ser compreendido pela Figura 1.

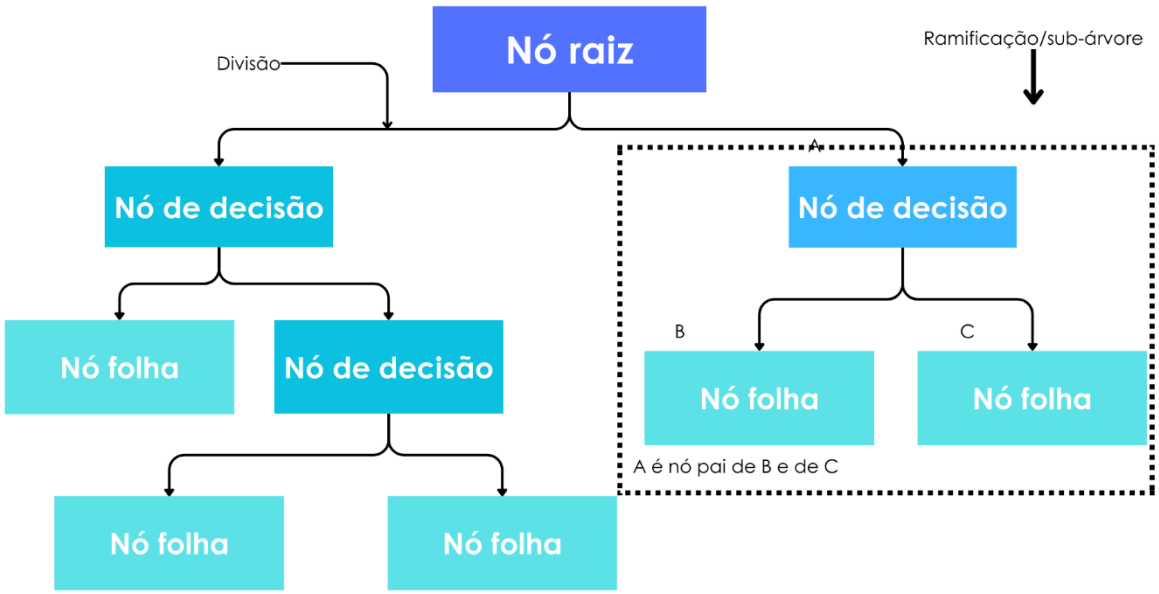
Figura 1. Fluxo de construção do modelo



Fonte: adaptado de ESCOVEDO; KOSHIYAMA (2020)

Após a construção do modelo obteve-se uma árvore de decisão que pode ser descrita pela Figura 2.

Figura 2. Modelo de árvore de decisão



Fonte: adaptado de ESCOVEDO; KOSHIYAMA (2020)

4.5 ANÁLISE DA ACURÁCIA DO MODELO

Após treinar a árvore de decisão, seu desempenho foi avaliado através da análise de matriz de confusão e utilizando as métricas acurácia, precisão, recall e f1-score, descritos no Quadro 3.

Quadro 3. Métricas de avaliação de desempenho do modelo

Métrica	Interpretação
Matriz de confusão	Tabela que mostra a quantidade de previsões corretas e incorretas feitas pelo modelo. São informações sobre verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos.
Acurácia	Proporção de previsões corretas em relação ao total de previsões. Fornece uma medida geral do quão bem o modelo está classificando as amostras.
Precisão	Indica a proporção de verdadeiros positivos em relação à soma de verdadeiros positivos e falsos positivos. Mede a precisão das previsões positivas feitas pelo modelo.
Recall	Também conhecido como taxa de sensibilidade ou taxa de verdadeiros positivos, é a proporção de verdadeiros positivos em relação à soma de verdadeiros positivos e falsos negativos. Mede a capacidade do modelo de identificar corretamente os casos positivos.
F1-score	Medida de média harmônica entre precisão e recall. Fornece uma pontuação única que equilibra a precisão e o recall do modelo.

Fonte: elaborado pelo autor (adaptado de https://scikit-learn.org/)

Além da avaliação do desempenho, foi realizada uma análise da coerência dos dados utilizados para tomar as decisões. Essa análise buscou verificar se os dados eram consistentes e confiáveis. A análise consistiu em verificar quais critérios estavam sendo usados pelo modelo e se eles condizem com o raciocínio logico humano para tomada de decisão em relação ao diagnóstico. Também foi examinada a importância atribuída pelo modelo a cada critério utilizado no processo de diagnóstico. Isso permitiu compreender como o classificador estava tomando suas decisões e qual foi a contribuição de cada critério na determinação dos diagnósticos. Essa análise permitiu interpretar e validar os resultados do modelo.

O código completo utilizado para construção do modelo e análises descritas encontra-se em APÊNDICE B - Modelo De Árvore De Decisão Para Dores Da ATM e APÊNDICE C – Modelo De Árvore De Decisão Para Dores Miofasciais Orofaciais

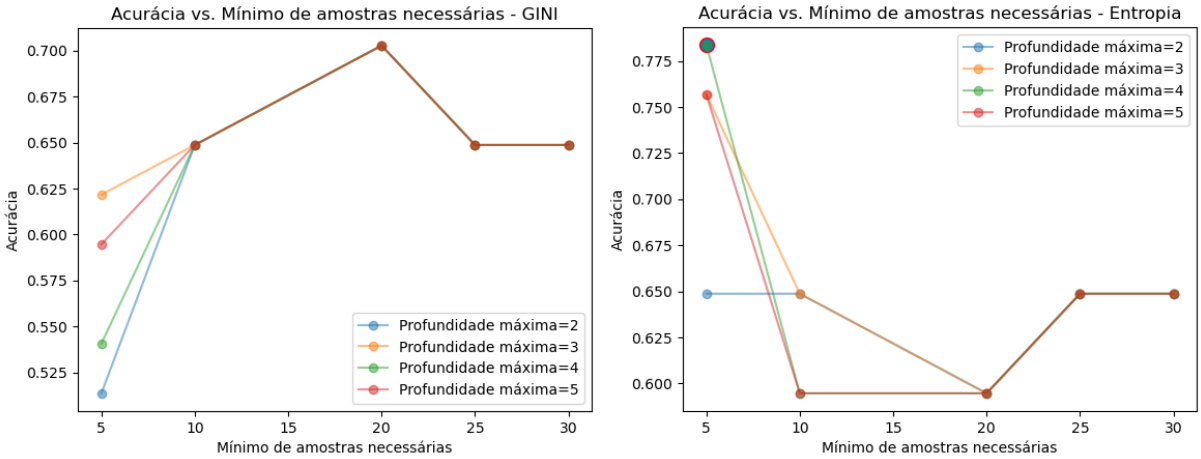
5 RESULTADOS

Foram realizados dois experimentos independentes utilizando árvores de decisão para classificar dores miofasciais e dores articulares. Ambos os experimentos utilizaram as mesmas métricas para avaliar o desempenho dos modelos.

5.1 DORES MIOFASCIAIS

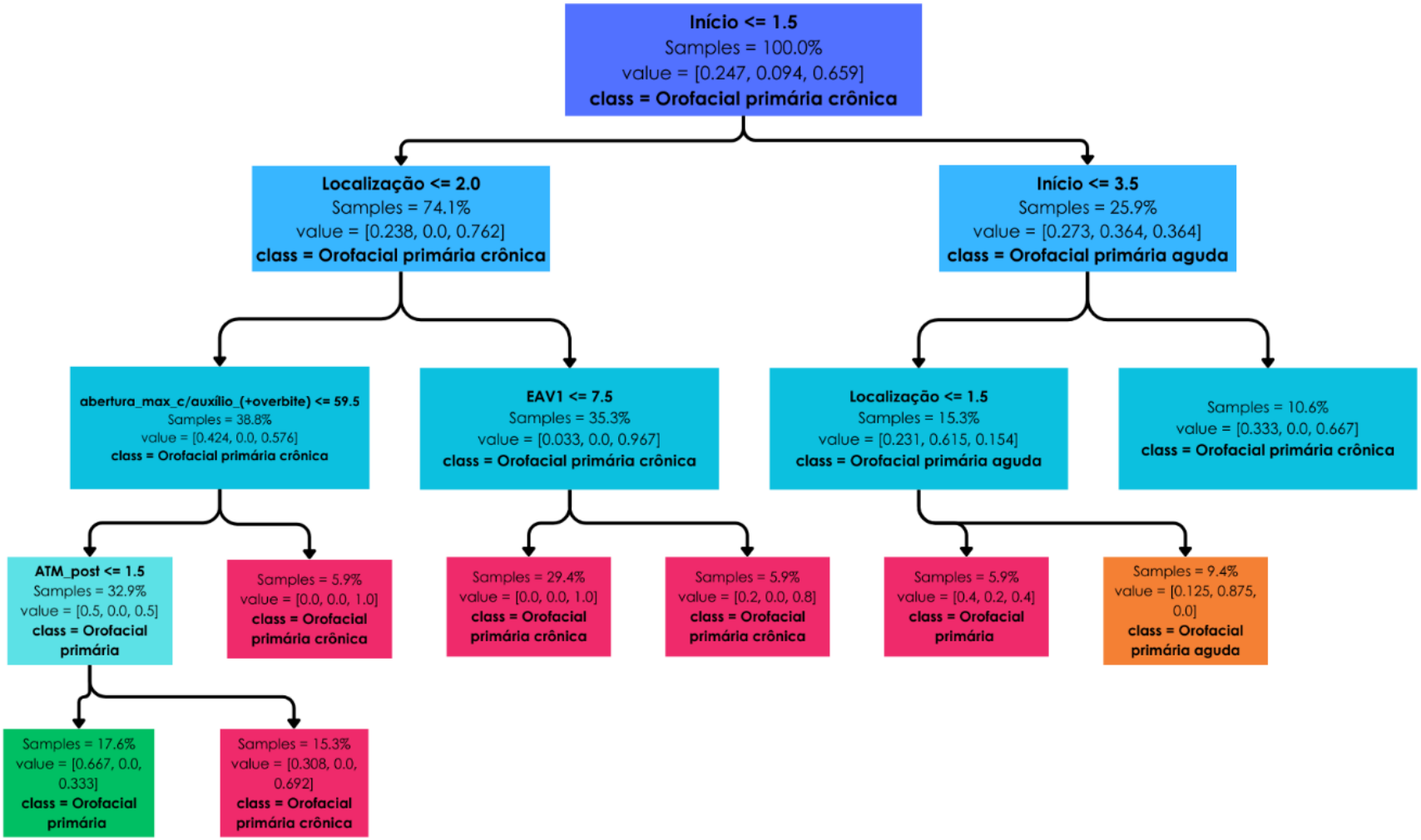
As Figura 3 ilustram os resultados de acurácia das diferentes árvores de decisão de acordo com o critério de cálculo de impureza dos dados (Gini e entropia), mínimo de amostras necessárias e profundidade máxima da árvore. O resultado para o cálculo de Gini apresentou acurácia com valores entre 0.513 e 0.702 (Figura 3A), enquanto o cálculo de entropia apresentou acurácia entre 0.594 e 0.783 (Figura 3B). O valor de maior acurácia (0.783) foi obtido pelos parâmetros entropia, mínimo de 5 amostras necessárias e profundidade máxima da árvore de 4 níveis de decisão na qual foi extraída a árvore de decisão completa (Figura 4)

Figura 3. Acurácia das diferentes árvores de decisão para dor miofascial orofacial. Círculo vermelho indica grupo de maior acurácia e que foi utilizado para detalhamento da árvore de decisão.



Fonte: elaborado pelo autor

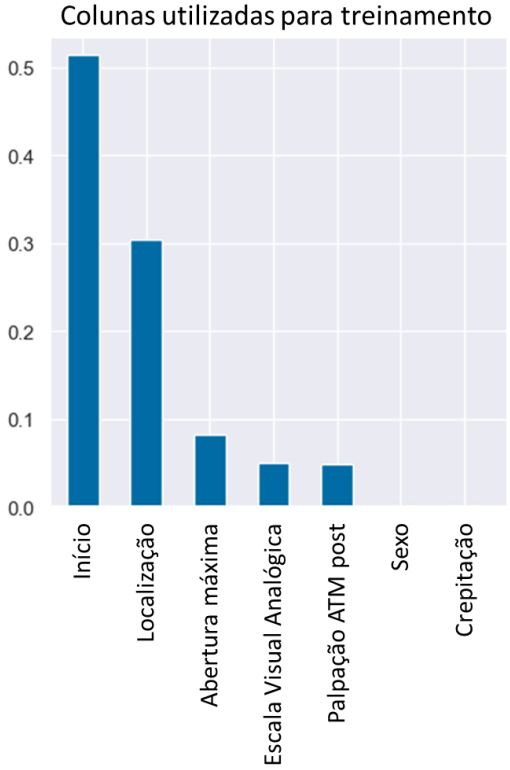
Figura 4. Árvore de decisão para dor miofascial orofacial extraída a partir da melhor acurácia encontrada.



Fonte: elaborado pelo autor

A árvore de decisão utilizou alguns dados de prontuários específicos para as tomadas de decisão, utilizando ao todo 5 características de um total de 54. A característica mais relevante foi o início da dor (0.514), seguido da localização (0.304), abertura máxima, escala analógica visual da dor e dor a palpação na ATM posterior, como mostra a Figura 5. As demais características fornecidas ao modelo não contribuíram de forma significativa para a previsão do resultado.

Figura 5. Colunas utilizadas para treinamento da árvore de decisão para dor miofascial orofacial.

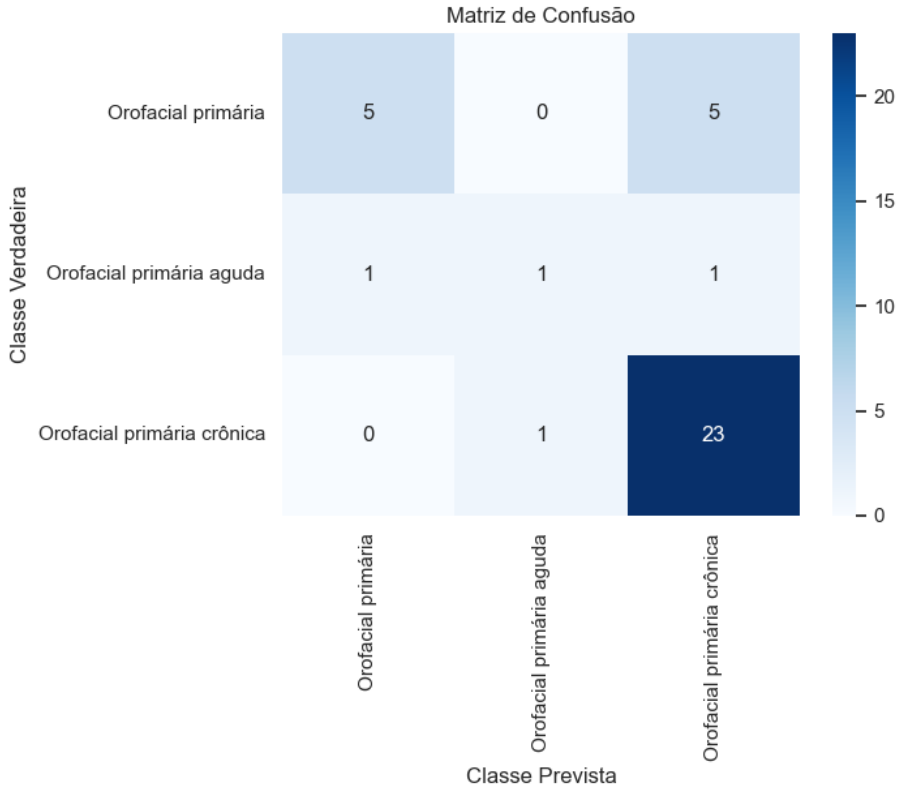


Fonte: elaborado pelo autor

5.1.1 Análise do desempenho do modelo

A matriz de confusão (Figura 6) mostra a performance o modelo de classificação em relação às classes previstas e classes verdadeiras do conjunto de dados.

Figura 6. Matriz de confusão para árvore de decisão para dor miofascial orofacial.



Fonte: elaborado pelo autor

Nesse caso, a matriz possui 3 classes. Podemos interpretar a matriz da seguinte maneira:

A classe "Dor miofascial orofacial primária" (linha 1) possui 5 instâncias que foram corretamente classificadas como "Dor miofascial orofacial primária" (coluna 1). Não há instâncias dessa classe que foram erroneamente classificadas como "Dor miofascial orofacial primária aguda" (coluna 2), mas 5 instâncias foram erroneamente classificadas como "Dor miofascial orofacial primária crônica" (coluna 3).

A classe "Dor miofascial orofacial primária aguda" (linha 2) possui 1 instância que foi corretamente classificada como "Dor miofascial orofacial primária aguda" (coluna 2). Há 1 instância dessa classe que foi erroneamente classificada como "Dor miofascial orofacial primária" (coluna 1), e 1 instância foi erroneamente classificada como "Dor miofascial orofacial primária crônica" (coluna 3).

A classe "Dor miofascial orofacial primária crônica" (linha 3) possui 23 instâncias que foram corretamente classificadas como "Dor miofascial orofacial primária crônica" (coluna 3). Há 1 instância dessa classe que foi erroneamente classificada como "Dor miofascial orofacial primária aguda" (coluna 2), e não há

instâncias erroneamente classificadas como "Dor miofascial orofacial primária" (coluna 1).

Essa matriz permite uma análise mais detalhada do desempenho do modelo, fornecendo informações sobre as classes em que ele tem mais dificuldade ou facilidade de classificar corretamente. Além disso, a partir dessa matriz, foi possível calcular as métricas de avaliação precisão, recall, F1-score, que forneceram uma visão geral do desempenho do modelo de classificação.

A Tabela 1 apresenta as métricas para avaliação do modelo de classificação. Essas métricas fornecem informações sobre o desempenho do modelo para cada classe individualmente, além de métricas agregadas que consideram o desempenho geral do modelo.

Tabela 1. Métricas de desempenho do modelo de árvore de decisão para dor miofascial orofacial.

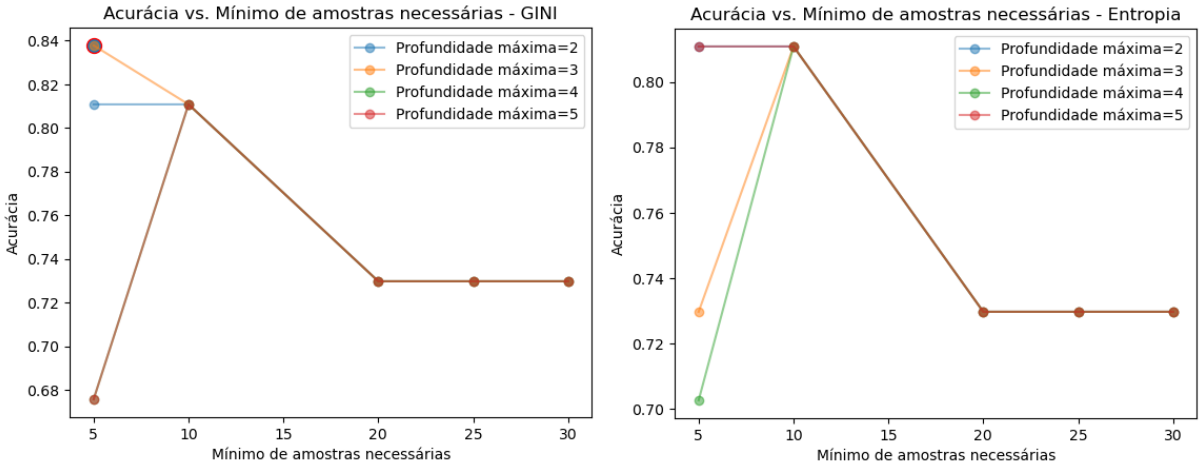
	Precision	*Recall*	*F1-score*	*Support*
Orofacial primária	0.83	0.50	0.62	10
Orofacial primária aguda	0.50	0.33	0.40	3
Orofacial primária crônica	0.79	0.96	0.87	24
accuracy		0.78		37
macro avg	0.71	0.60	0.63	37
weighted avg	0.78	0.78	0.76	37

Fonte: elaborado pelo autor

5.2 DORES ARTICULARES

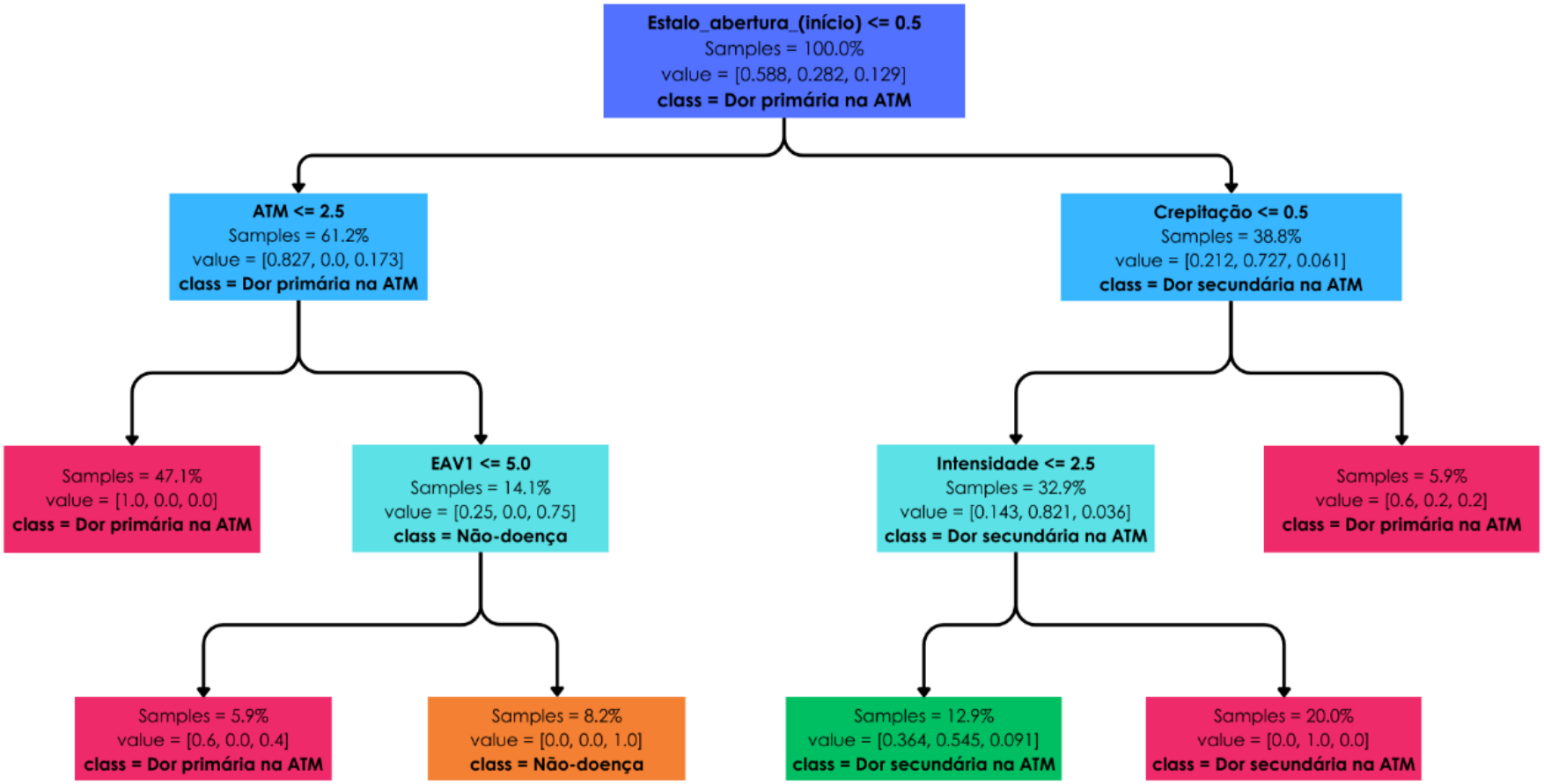
As Figura 7 ilustram os resultados de acurácia das diferentes árvores de decisão de acordo com o critério de cálculo de impureza dos dados (Gini e entropia), mínimo de amostras necessárias e profundidade máxima da árvore. O resultado para o cálculo de Gini apresentou acurácia com valores entre 0.675 e 0.837 (Figura 7A) enquanto que o cálculo de entropia apresentou acurácia entre 0.702 e 0.810 (Figura 7B). O valor de maior acurácia (0.837) foi obtido pelos parâmetros entropia, mínimo de 5 amostras necessárias e profundidade máxima da árvore de 3 níveis de decisão na qual foi extraída a árvore de decisão completa (Figura 8).

Figura 7. Acurácia das diferentes árvores de decisão para dor na articulação temporomandibular. Círculo vermelho indica grupo de maior acurácia e que foi utilizado para detalhamento da árvore de decisão



Fonte: elaborado pelo autor

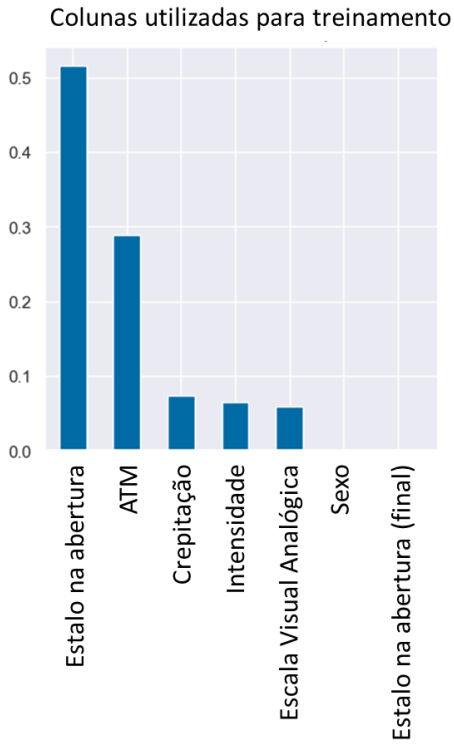
Figura 8. Árvore de decisão para dor na articulação temporomandibular extraída a partir da melhor acurácia encontrada



Fonte: elaborado pelo autor

A árvore de decisão utilizou alguns dados de prontuários específicos para as tomadas de decisão, utilizando ao todo 5 características de um total de 54. A característica mais relevante foi a presença de estalo no início da abertura bucal (0.515), seguido de dor à palpação na ATM (0.288), e ainda presença de crepitação, intensidade da dor e escala visual analógica de dor (Figura 9).

Figura 9. Colunas utilizadas para treinamento da árvore de decisão para dor na articulação temporomandibular.

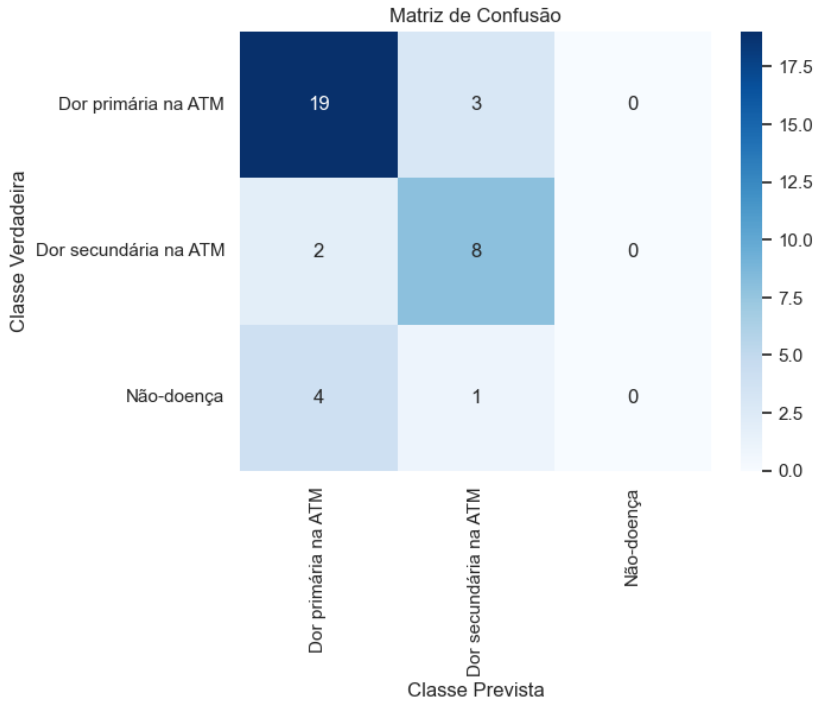


Fonte: elaborado pelo autor

5.2.1 Análise do desempenho do modelo

A matriz de confusão (Figura 10) mostra a performance o modelo de classificação em relação às classes previstas e classes verdadeiras do conjunto de dados.

Figura 10. Matriz de confusão para árvore de decisão para dor na articulação temporomandibular



Fonte: elaborado pelo autor

Nesse caso, a matriz possui 3 classes. Podemos interpretar a matriz da seguinte maneira:

A classe "Dor primária na ATM" (linha 1) possui 19 instâncias que foram corretamente classificadas como "Dor primária na ATM" (coluna 1). Há 3 instâncias dessa classe que foram erroneamente classificadas como "Dor secundária na ATM" (coluna 2), e nenhuma instância foi erroneamente classificada como "Não-doença" (coluna 3).

A classe "Dor secundária na ATM" (linha 2) possui 8 instâncias que foram corretamente classificadas como "Dor secundária na ATM" (coluna 2). Há 2 instâncias dessa classe que foram erroneamente classificadas como "Dor primária na ATM" (coluna 1), e nenhuma instância foi erroneamente classificada como "Não-doença" (coluna 3).

A classe "Não-doença" (linha 3) não possui instâncias corretamente classificadas pelo modelo, ou seja, todas as 5 instâncias dessa classe foram erroneamente classificadas como "Dor primária na ATM" (coluna 1).

A Tabela 2Tabela 1 apresenta as métricas para avaliação do modelo de classificação. Essas métricas fornecem informações sobre o desempenho do modelo

para cada classe individualmente, além de métricas agregadas que consideram o desempenho geral do modelo.

Tabela 2. Métricas de desempenho do modelo de árvore de decisão para dor na articulação temporomandibular

	Precision	*Recall*	*F1-score*	*Support*
Dor primária na ATM	0.76	0.86	0.81	22
Dor secundária na ATM	0.67	0.80	0.73	10
Não-doença	1.00	0.00	0.00	5
accuracy		0.73		37
macro avg	0.81	0.55	0.51	37
weighted avg	0.77	0.73	0.68	37

Fonte: elaborado pelo autor

6 DISCUSSÃO

Este estudo descreve uma abordagem de *machine learning* baseada em árvore de decisão para a detecção de (1) dores miofasciais orofaciais e (2) dores da ATM com os diagnósticos fundamentados pela ICOP. Desse modo, foram geradas duas árvores de decisão distintas, uma para cada condição avaliada. Os resultados alcançados evidenciam a efetividade do modelo na detecção de ambas as condições com precisão acima de 75%. Portanto, nossos resultados sugerem que a IA pode desempenhar um papel complementar valioso às ferramentas de diagnóstico existentes.

O modelo foi desenvolvido utilizando o acervo de prontuários de um centro de referência em dor orofacial, o CEMDOR. Desta forma, todos os pacientes possuíam algum diagnóstico de dor, sendo estas musculares e articulares ou apenas musculares e, consequentemente nenhum paciente com ausência de dor. Assim, após a triagem de todos os prontuários, obteve-se uma amostra por conveniência composta de 122 prontuários completos contendo diagnósticos revisados por um especialista em DTM e dor orofacial.

No caso das dores miofasciais orofaciais foi obtida acurácia de 78%, para classificação entre dor miofascial orofacial primaria, dor miofascial orofacial primaria aguda e dor miofascial orofacial primaria crônica. Esses dados são semelhantes a estudos realizados anteriormente sobre a aplicação de IA para detecção de condições dolorosas da face (CALIL et al., 2020; REDA et al., 2023).

Nosso modelo obteve um bom desempenho geral na classificação das condições propostas, semelhante a estudos anteriores utilizando a IA (CALIL et al., 2020; CHOI et al., 2021; ORHAN et al., 2021; REDA et al., 2023). Na análise das condições musculares, pode-se observar subdiagnóstico para a classe de dor primária aguda, onde um sujeito foi classificado como primária crônica e outro apenas como dor primária. A dificuldade na correta classificação descrita acima pode ser devido ao número reduzido de prontuários fornecidos ao modelo com esse tipo de diagnóstico quando em comparação às demais classes. A identificação da DTM é um longo processo baseado na história médica e exames complementares, com evidências sugerindo que dores agudas não são detectadas precocemente e o diagnóstico dessa condição só é concluído já em fase crônica (LÖVGREN et al., 2016). Dessa forma, ao optar por uma amostra de conveniência em um banco de dados, é possível ocorrer

uma distribuição desigual de diagnósticos, resultando em uma escassez de dados em algumas classes em comparação com outras (LI; SUN; ZHU, 2010).

Já para a detecção de dores na ATM, foi encontrada acurácia de 83% para classificação entre dor primária na ATM e dor secundária na ATM. Modelos criados para detecção de DTM articular a partir de exames de imagem atingiram acurácia similar ao nosso estudo, com 78% de acertos quando utilizado tomografia, valor semelhante ao de radiologistas especialistas (CHOI et al., 2021). Já em estudos utilizando ressonâncias magnéticas obteve-se acurácia de 89% (ORHAN et al., 2021). Tanto estudos utilizando dados de exames complementares por imagem quanto dados clínicos de prontuário atingiram valores relevantes de acerto diagnóstico, no entanto, nenhum estudo conseguiu classificar todos os diagnósticos corretamente. Visto que a associação de ambas as análises é essencial para o diagnóstico de 100% dos casos, essa associação é um fator a ser considerado para aprimoramento em modelos de diagnóstico assistido por IA.

Quando avaliado os diagnósticos de dores articulares o modelo não conseguiu identificar corretamente a classe “Não-doença”, classificando todos os sujeitos com alguma DTM articular. O desempenho do modelo é impactado devido a todos os pacientes do conjunto de dados apresentarem algum tipo de dor muscular, sendo a maioria deles também com dores nas articulações (106 pacientes). No entanto, existe um subconjunto de pacientes (16) que possuem apenas dor muscular e não apresentam dores nas articulações. O fato de a informação de presença de dor em face estar disponível para ambos os diagnósticos (muscular ou articular), cria um fator de confusão para o modelo em distinguir entre dor muscular e dor nas articulações, com tendência de classificar todos os pacientes como tendo dor, mesmo que essa dor seja exclusivamente muscular. Consequentemente, isso resulta em uma baixa precisão na classificação correta dos pacientes sem dor articular (classe “Não-doença”).

A complexidade e o volume excessivo de dados dos prontuários para diagnóstico de dor dificultam a interpretação clínica, especialmente para profissionais que não possuem expertise na área. Do ponto de vista clínico, todas as perguntas presentes no questionário são relevantes e contribuem para a elaboração de um tratamento adequado e personalizado para o paciente. Contudo, o modelo utilizou apenas algumas características dentre várias para prever uma condição, e, quando fornecido um número adequado de dados, obteve diagnósticos corretos.

Instrumentos longos apresentam dificuldades de administração tanto em ambientes clínicos quanto em pesquisas epidemiológicas devido ao tempo que consomem e ao cansaço que podem causar tanto nos pacientes quanto nos profissionais, afetando as respostas obtidas e a forma como são registradas. Apesar da complexidade do diagnóstico, é importante que o instrumento seja simples e contenha apenas perguntas relevantes (FIRMINO et al., 2022). No entanto, é importante ressaltar que este estudo não contemplou todos diagnósticos das DTM. Para definir quais variáveis têm influência relevante até a última etapa de cada possível diagnóstico, é necessário realizar uma análise mais abrangente. Isso permitiria determinar se outras características são ou não relevantes para o diagnóstico das condições em questão.

Para aperfeiçoar o modelo é necessário aumentar o número de pacientes e equilibrar a distribuição entre os diagnósticos. Uma das limitações da árvore de decisão é sua suscetibilidade ao *underfitting e overfitting*, especialmente quando a amostra de dados é pequena.(LI; SUN; ZHU, 2010; STIGLIC et al., 2012). O *underfitting* ocorre quando o modelo não se ajusta aos dados de treinamento por não aprender padrões importantes nos dados. Esse problema é identificado quando o modelo tem um desempenho baixo nos dados de treinamento e teste. Já o *overfitting* ocorre quando o modelo se ajusta excessivamente aos dados de treinamento, capturando até mesmo o ruído presente nos dados. Nesse caso, o modelo perde a capacidade de generalizar para novos dados. O *overfitting* pode ser identificado quando um modelo tem excelente desempenho para dados de treinamento, mas um desempenho ruim nos dados de teste.(HERNÁNDEZ et al., 2022). Isso compromete a capacidade de classificação e a robustez do modelo (LI; SUN; ZHU, 2010).

Embora os métodos de diagnóstico auxiliados por inteligência artificial (IA) apresentem menor relevância quando utilizados por especialistas, eles se mostram altamente valiosos para clínicos não especialistas (REDA et al., 2023). A utilização da IA na odontologia pode trazer diversos benefícios no processo de diagnóstico. Para isso, é fundamental que as ferramentas de IA sejam desenvolvidas com rigor científico, validadas por meio de estudos clínicos e utilizadas como uma ferramenta complementar ao julgamento clínico dos profissionais de saúde. A aplicação de IA na odontologia representa um grande potencial para impulsionar avanços e benefícios na área. É uma ferramenta poderosa que complementa e auxilia o trabalho dos cirurgiões-dentistas, proporcionando diagnósticos mais rápidos e precisos e uma

melhor experiência geral do paciente. Além disso, a IA pode ajudar a analisar grandes quantidades de dados odontológicos para uma melhor compreensão de padrões, tendências e predições de saúde bucal, auxiliando no desenvolvimento de estratégias de prevenção e tratamento mais eficazes.

7 CONCLUSÕES

Pode-se concluir que:

- A utilização de árvore de decisão por *machine learning* apresenta potencial relevante para auxílio no diagnóstico clínico das DTM;
- A implementação de métodos de diagnósticos clínicos das DTM auxiliados por IA devem ser considerados como rotina clínica;
- Profissionais não especialistas na área de DTM e dor orofacial podem se beneficiar do auxílio no diagnóstico clínico através da IA.

8 SUGESTÕES PARA TRABALHOS FUTUROS

Além de criar maior homogeneidade entre os diagnósticos fornecidos ao conjunto de dados, é recomendável aumentar o tamanho desse conjunto de dados. Também é sugerido explorar outros modelos de *machine learning*, tais como *Random forest* e kNN (*k-nearest neighbors*) ou até mesmo Redes neurais. Ao implementar essas sugestões em trabalhos futuros, espera-se melhorar a capacidade de generalização do modelo, melhorar a consistência do diagnóstico fornecido e explorar métodos de *machine learning*, resultando em sistemas mais robustos e melhores serviços odontológicos à população.

GLOSSÁRIO

A

ACURÁCIA (ACCURACY): métrica de avaliação comum usada para medir a precisão geral de um modelo de classificação. É a proporção de previsões corretas em relação ao total de previsões**.**

ALGORITMO CART (*CLASSIFICATION AND REGRESSION TREES*): algoritmo de *machine learning* usado para construir árvores de decisão que podem ser usadas para classificação ou regressão. CART é um acrônimo para Classification And Regression Trees, indicando sua capacidade de lidar com ambos os tipos de problemas.

ARRAY: Uma estrutura de dados multidimensional do NumPy que contém elementos de um único tipo de dados.

B

BIG DATA: Refere-se a conjuntos de dados extremamente grandes e complexos que requerem ferramentas e técnicas especializadas para armazenamento, processamento e análise eficientes.

C

CLASS WEIGHT (PESO DE CLASSE): parâmetro usado para tratar o desequilíbrio de classes em um conjunto de dados. Em certos problemas de classificação, as classes podem ter distribuições desiguais. O parâmetro class weight atribui pesos diferentes às classes durante o treinamento para lidar com esse desequilíbrio.

CLASSIFIER (CLASSIFICADOR): Um algoritmo de aprendizado de máquina usado para atribuir classes a dados.

CRITÉRIO (CRITERION): Um parâmetro que define a medida de qualidade usada para avaliar a qualidade da divisão em cada nó da árvore. Exemplos comuns são "gini" (índice de Gini) e "entropy" (entropia).

D

DADOS DE TESTE: dados independentes usados para avaliar o desempenho de um modelo de aprendizado de máquina treinado. Eles não são usados durante o treinamento, mas sim para testar o modelo e avaliar.

DADOS DE TREINAMENTO: dados usados para treinar um modelo de aprendizado de máquina. Os dados de treinamento consistem em exemplos, onde cada exemplo tem um conjunto de características e um rótulo associado, que é a classe correta para esse exemplo.

DATAFRAME: estrutura de dados tabular bidimensional encontrada em bibliotecas como pandas. Ele organiza os dados em linhas e colunas, semelhante a uma planilha ou tabela de banco de dados. Os data frames são frequentemente usados para manipulação, análise e limpeza de dados.

DECISIONTREECLASSIFIER: objeto que pertence à biblioteca scikit-learn. É uma implementação da árvore de decisão para classificação.

DIVISÃO (SPLIT): decisão tomada em um nó de uma árvore de decisão para separar as amostras com base em um valor específico de uma característica. A divisão divide o conjunto de dados em subconjuntos menores, criando ramos na árvore que representam diferentes caminhos de decisão.

E

ENSEMBLE LEARNING (APRENDIZADO EM CONJUNTO): técnica que combina vários modelos de aprendizado de máquina para melhorar o desempenho e a precisão das previsões. Em vez de confiar em um único modelo, o ensemble learning utiliza a diversidade dos modelos para obter um consenso ou uma média ponderada das previsões individuais.

ENTROPIA (ENTROPY): medida de impureza usada em algoritmos de árvore de decisão. Ela mede a incerteza ou a quantidade de informação aleatória presente em um conjunto de dados. A entropia é usada para calcular a qualidade de uma divisão em um nó, sendo desejável obter uma divisão com uma entropia baixa.

F

F1-SCORE: métrica que combina precisão e recall em uma única medida. É a média harmônica entre essas duas métricas e é útil quando há um desequilíbrio entre as classes.

FEATURE (CARACTERÍSTICA): Uma variável usada como entrada para um modelo de aprendizado de máquina.

FEATURE SELECTION (SELEÇÃO DE ATRIBUTOS): É o processo de escolha das melhores características (atributos) de um conjunto de dados para serem usadas no treinamento de um modelo. A seleção de atributos visa reduzir a dimensionalidade do conjunto de dados, eliminando características irrelevantes ou redundantes.

FIT: método comum usado em muitas bibliotecas de *machine learning*, como scikit-learn (sklearn). O método "fit" é usado para treinar um modelo de aprendizado de máquina com base nos dados fornecidos. Durante o processo de treinamento, o modelo aprende com os dados e ajusta seus parâmetros internos para fazer previsões precisas.

G

GROUPBY: Uma operação do Pandas que permite agrupar dados com base em uma ou mais variáveis e aplicar funções a cada grupo.

H

HIPERPARÂMETROS (HYPERPARAMETERS): Parâmetros definidos antes do treinamento do modelo e que afetam o comportamento do modelo.

I

IMPORTÂNCIA DA CARACTERÍSTICA (FEATURE IMPORTANCE): medida que indica o quanto uma característica contribui para as decisões tomadas por um modelo de árvore de decisão. É comumente calculada com base na redução do critério de impureza quando uma determinada característica é usada para fazer divisões na árvore.

INDEX: Uma estrutura de dados do Pandas que rotula as linhas ou colunas de um data frame ou série.

ÍNDICE GINI (GINI INDEX): medida de impureza usada em algoritmos de árvore de decisão, como CART. Ele mede a probabilidade de classificar aleatoriamente uma amostra incorretamente com base na distribuição das classes presentes. O índice de Gini é usado para avaliar a qualidade de uma divisão em um nó, sendo desejável obter uma divisão com um índice de Gini baixo.

M

MACHINE LEARNING (APRENDIZADO DE MÁQUINA): Um campo da inteligência artificial que se concentra no desenvolvimento de algoritmos e modelos que permitem aos computadores aprender e melhorar a partir de dados.

MACRO AVG (MÉDIA MACRO): métrica que calcula a média não ponderada das métricas de avaliação (como precisão, recall, F1-score etc.) para cada classe em um problema de classificação. Primeiro, são calculadas as métricas para cada classe individualmente e, em seguida, é feita uma média aritmética desses valores para obter a média macro. Essa abordagem trata cada classe igualmente, independentemente do desequilíbrio de amostras entre as classes.

MATRIZ DE CONFUSÃO (CONFUSION MATRIX): tabela que mostra as contagens de classificações corretas e incorretas feitas por um modelo de classificação. É frequentemente usado para calcular métricas como precisão, recall e F1-score.

MAX_DEPTH: Define a profundidade máxima da árvore de decisão. Limitar a profundidade ajuda a controlar o tamanho da árvore e evita overfitting.

MAX_FEATURES: O número máximo de características (atributos) a serem consideradas ao procurar a melhor divisão em cada nó.

MÉTODO DE K-VIZINHOS MAIS PRÓXIMOS (*K-NEAREST NEIGHBORS - KNN*): algoritmo de *machine learning* usado para classificação e regressão. Ele classifica

uma instância desconhecida atribuindo-lhe a classe mais comum entre seus k vizinhos mais próximos com base em uma medida de distância.

MIN_SAMPLES_LEAF: O número mínimo de amostras necessárias em uma folha da árvore. Se a quantidade de amostras em uma folha for menor do que esse valor, a folha não será dividida.

MIN_SAMPLES_LEAF: parâmetro que pode ser ajustado em algoritmos de árvore de decisão, como o algoritmo CART. Ele define o número mínimo de amostras necessárias em uma folha da árvore. Se o número de amostras em uma folha for menor do que o valor definido para "min_samples_leaf", a divisão da árvore será interrompida.

MIN_SAMPLES_SPLIT: O número mínimo de amostras necessárias em um nó para que a divisão seja considerada. Se o número de amostras em um nó for menor que esse valor, a divisão não será realizada.

N

NaN: Uma sigla para "Not a Number", que é uma marcação usada no pandas para representar valores ausentes ou inválidos.

NÓ (NODE): entidade em uma árvore de decisão que representa uma condição ou uma divisão. Os nós são usados para tomar decisões com base em valores de características específicas. Os nós internos representam divisões, enquanto os nós terminais (folhas) representam classes ou valores de destino.

NÓ FOLHA (LEAF NODE): também chamado de terminal, é um nó em uma árvore de decisão que não é dividido. Ele representa o resultado final ou a classe de uma instância após seguir um caminho de decisão específico. Os nós folha não têm filhos e representam as saídas finais do modelo.

NÓ RAIZ (ROOT NODE): primeiro nó em uma árvore de decisão. Ele não é resultado de uma divisão, mas é o ponto de partida para a construção da árvore. O nó raiz

contém todas as amostras e é dividido em nós filhos com base nas características dos dados.

O

OBJETO DATA FRAME: uma instância específica de um data frame em uma determinada linguagem de programação, como Python com a biblioteca pandas. É a representação em memória de um conjunto de dados organizado em formato de data frame.

OVERFITTING: Um problema em que um modelo de *machine learning* se ajusta demais aos dados de treinamento, resultando em baixa capacidade de generalização para dados não vistos.

P

PODA (PRUNING): processo em que partes de uma árvore de decisão são removidas para reduzir o tamanho e a complexidade do modelo. A poda é feita removendo-se os ramos menos importantes da árvore, com o objetivo de melhorar a capacidade de generalização do modelo e evitar overfitting.

PRECISÃO (PRECISION): métrica que mede a proporção de instâncias positivas corretamente classificadas em relação ao total de instâncias classificadas como positivas.

PROCESSAMENTO DE LINGUAGEM NATURAL (NATURAL LANGUAGE PROCESSING - NLP): Um campo da inteligência artificial que se concentra no processamento e análise de linguagem humana por computadores, incluindo tarefas como reconhecimento de fala, tradução automática e análise de sentimento.

R

RANDOM FOREST (FLORESTA ALEATÓRIA): algoritmo de *machine learning* baseado em *ensemble learning*. Ele combina várias árvores de decisão individuais para tomar uma decisão final. Cada árvore na floresta é treinada em uma amostra aleatória do conjunto de dados, e a decisão final é tomada por meio de uma votação ou média das predições individuais das árvores.

RANDOM STATE (ESTADO ALEATÓRIO): parâmetro que define a semente (seed) para a geração de números aleatórios em algoritmos de aprendizado de máquina. Ao fixar o valor do random state, garante-se que os resultados sejam reproduzíveis.

RECALL (REVOCAÇÃO OU SENSIBILIDADE - RECALL OU SENSITIVITY): métrica que mede a proporção de instâncias positivas corretamente classificadas em relação ao total de instâncias verdadeiramente positivas.

REDES NEURAIS ARTIFICIAIS (ARTIFICIAL NEURAL NETWORKS - ANN): Um modelo de aprendizado de máquina inspirado no funcionamento do cérebro humano, composto por camadas de neurônios artificiais que processam e transmitem informações.

REGULARIZAÇÃO (REGULARIZATION): técnica usada para evitar overfitting em modelos de *machine learning*. Isso envolve adicionar um termo de regularização à função de perda ou ajustar os pesos do modelo para evitar coeficientes excessivamente grandes.

RÓTULOS (LABELS): são as classes ou categorias às quais os exemplos de treinamento pertencem. Eles são usados para treinar o modelo de árvore de decisão, para que ele possa aprender a fazer previsões corretas para novos exemplos.

S

SAMPLES: Em *machine learning*, refere-se às instâncias ou observações individuais em um conjunto de dados. Cada amostra representa uma observação única que contém valores para várias características ou atributos.

SEED (SEMENTE): valor numérico utilizado para inicializar o gerador de números aleatórios. Ao fixar uma semente, é possível reproduzir os mesmos resultados em diferentes execuções do algoritmo, o que é importante para fins de consistência e comparabilidade.

SERIES: Uma estrutura de dados unidimensional do Pandas, semelhante a uma coluna em um Data Frame.

SHAPE: Uma propriedade de um array que retorna as dimensões do array, ou seja, o número de elementos ao longo de cada eixo.

SUPPORT (SUPORTE): número de ocorrências (amostras) de cada classe presente no conjunto de dados. O suporte indica quantas amostras pertencem a cada classe individualmente. É uma métrica útil para avaliar o desequilíbrio de classes, especialmente ao interpretar as métricas de avaliação em relação a cada classe.

U

UNDERFITTING: Um problema em que um modelo de *machine learning* não se ajusta suficientemente aos dados de treinamento, resultando em baixa capacidade de representação e desempenho inadequado.

V

VALOR (VALUE): um valor numérico associado a uma determinada característica ou variável em um conjunto de dados. Em machine learning, os valores são usados como entrada para algoritmos de aprendizado de máquina para treinar e fazer previsões.

VARIÁVEL TARGET: A variável target, também conhecida como variável dependente, é a variável que se deseja prever ou explicar em um problema de modelagem. Resultado estimado com base nas outras variáveis do conjunto de dados.

WEIGHTED AVG (MÉDIA PONDERADA): métrica que calcula a média das métricas de avaliação, levando em consideração o peso ou a proporção de cada classe em um problema de classificação. É calculada tomando-se uma média ponderada dos valores das métricas, onde os pesos são determinados pelo número de AGGARWAL, V. R. et al. Dentists' and Specialists' Knowledge of Chronic Orofacial Pain: Results from a Continuing Professional Development Survey. **Primary Dental Care**, v. os18, n. 1, p. 41–44, 1 jan. 2011.

REFERÊNCIAS

AGGARWAL, V. R. et al. Dentists' and Specialists' Knowledge of Chronic Orofacial Pain: Results from a Continuing Professional Development Survey. **Primary Dental Care**, v. os18, n. 1, p. 41–44, 1 jan. 2011.

AL-HURAISHI, H. A. et al. Newly graduated dentists' knowledge of temporomandibular disorders compared to specialists in Saudi Arabia. **BMC Oral Health**, v. 20, n. 1, p. 272, 7 dez. 2020.

BALOGH, E. P.; MILLER, B. T.; BALL, J. R. Improving diagnosis in health care. **Improving Diagnosis in Health Care**, p. 1–472, 29 dez. 2015.

BENDER, S. D. Temporomandibular Disorders, Facial Pain, and Headaches. **Headache: The Journal of Head and Face Pain**, v. 52, p. 22–25, maio 2012.

BOND, E. C. et al. **Temporomandibular Disorders: Priorities for Research andCare**. Washington, D.C.: National Academies Press, 2020.

CALIL, B. C. et al. Identification of arthropathy and myopathy of the temporomandibular syndrome by biomechanical facial features. **BioMedical Engineering OnLine**, v. 19, n. 1, 15 dez. 2020.

CARRILLO-PEREZ, F. et al. Applications of artificial intelligence in dentistry: A comprehensive review. **Journal of Esthetic and Restorative Dentistry**, v. 34, n. 1, p. 259–280, 29 jan. 2022.

CHOI, E. et al. Artificial intelligence in detecting temporomandibular joint osteoarthritis on orthopantomogram. **Scientific Reports**, v. 11, n. 1, 13 maio 2021.

CONSELHO FEDERAL DE ODONTOLOGIA. **Quantidade Geral de Cirurgiões-Dentistas Especialistas**. Disponível em: <https://website.cfo.org.br/estatisticas/quantidade-geral-de-cirurgioes-dentistas-especialistas/>. Acesso em: 14 mar. 2022.

DUBNER, R.; OHRBACH, R.; DWORKIN, S. F. The Evolution of TMD Diagnosis: Past, Present, Future. **J Dent Res**, v. 95, n. 10, p. 1093–1101, 1 set. 2016.

DWORKIN, S. F.; LERESCHE, L. Research diagnostic criteria for temporomandibular disorders: review, criteria, examinations and specifications, critique. **J Craniomandib Disord**, v. 6, n. 4, p. 301–355, 1992.

ESCOVEDO, T.; KOSHIYAMA, A. **Introdução a Data Science: Algoritmos de Machine Learning e métodos de análise**. Casa do Código ed. [s.l: s.n.].

FAROOK, T. H. et al. Machine Learning and Intelligent Diagnostics in Dental and Orofacial Pain Management: A Systematic Review. **Pain Research and Management**, v. 2021, 26 abr. 2021.

FIRMINO, R. T. et al. Development and validation of a short form of the BOHLAT-P. **Brazilian Oral Research**, v. 36, 2022.

GARCIA LEIVA, R. et al. A Novel Hyperparameter-Free Approach to Decision Tree Construction That Avoids Overfitting by Design. **IEEE Access**, v. 7, p. 99978–99987, 2019.

HADLAQ, E. et al. Dentists' knowledge of chronic orofacial pain. **Nigerian Journal of Clinical Practice**, v. 22, n. 10, p. 1365, 2019.

HERNÁNDEZ, V. A. S. et al. A Practical Tutorial for Decision Tree Induction. **ACM Computing Surveys**, v. 54, n. 1, p. 1–38, 31 jan. 2022.

ICOP. International Classification of Orofacial Pain, 1st edition (ICOP). **Cephalalgia : an international journal of headache**, v. 40, n. 2, p. 129–221, 1 fev. 2020.

JANIESCH, C.; ZSCHECH, P.; HEINRICH, K. Machine learning and deep learning. **Electronic Markets**, v. 31, n. 3, p. 685–695, 8 set. 2021.

JORDAN, M. I.; MITCHELL, T. M. Machine learning: Trends, perspectives, and prospects. **Science**, v. 349, n. 6245, p. 255–260, 17 jul. 2015.

KOSE, C. et al. Using artificial intelligence to predict the final color of leucite-reinforced ceramic restorations. **Journal of Esthetic and Restorative Dentistry**, v. 35, n. 1, p. 105–115, 2 jan. 2023.

LEEUW, R. DE; KLASSER, G. D. Orofacial Pain: Guidelines for Assessment, Diagnosis, and Management. **STOMATOLOGY EDU JOURNAL**, v. 2, n. 2, p. 173, 2015.

LI, Y.; SUN, G.; ZHU, Y. **Data Imbalance Problem in Text Classification**. 2010 Third International Symposium on Information Processing. **Anais**...IEEE, out. 2010.

LÖVGREN, A. et al. Validity of three screening questions (3Q/TMD) in relation to the DC/TMD. **Journal of Oral Rehabilitation**, v. 43, n. 10, out. 2016.

LOYOLA-GONZALEZ, O. Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View. **IEEE Access**, v. 7, 2019.

MACHADO, N. A. DE G.; LIMA, F. F.; CONTI, P. C. R. Current panorama of temporomandibular disorders' field in Brazil. **Journal of Applied Oral Science**, v. 22, n. 3, p. 146–151, jun. 2014.

MACHOY, M. E. et al. The ways of using machine learning in dentistry. **Advances in Clinical and Experimental Medicine**, v. 29, n. 3, p. 375–384, 1 mar. 2020.

MANFREDINI, D. et al. Research diagnostic criteria for temporomandibular disorders: a systematic review of axis I epidemiologic findings. **Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology, and Endodontology**, v. 112, n. 4, p. 453–462, out. 2011.

NATIONAL INSTITUTE OF DENTAL AND CRANIOFACIAL RESEARCH (2018) Prevalence of TMJD and its signs and symptoms. https://www.nidcr.nih.gov/research/data-statistics/facial-pain/prevalence.

ORHAN, K. et al. Development and Validation of a Magnetic Resonance Imaging-Based Machine Learning Model for TMJ Pathologies. **BioMed Research International**, v. 2021, 5 jul. 2021.

PIGG, M. et al. New International Classification of Orofacial Pain: What Is in It For Endodontists? **Journal of Endodontics**, v. 47, n. 3, p. 345–357, mar. 2021.

RAJKOMAR, A.; DEAN, J.; KOHANE, I. Machine Learning in Medicine. **New England Journal of Medicine**, v. 380, n. 14, p. 1347–1358, 4 abr. 2019.

REDA, B. et al. Artificial intelligence to support early diagnosis of temporomandibular disorders: A preliminary case study. **Journal of Oral Rehabilitation**, v. 50, n. 1, p. 31–38, 1 jan. 2023.

REISSMANN, D. R. et al. Impact of dentists' years since graduation on management of temporomandibular disorders. **Clinical Oral Investigations**, v. 19, n. 9, 7 dez. 2015.

ROKACH, L.; MAIMON, O. Data mining with decision trees. Theory and applications. Em: **Introduction to Decision Trees**. [s.l.] Mach Percept Artif Intell, 2008. v. 69.

SCHIFFMAN, E. et al. Diagnostic Criteria for Temporomandibular Disorders (DC/TMD) for Clinical and Research Applications: Recommendations of the International RDC/TMD Consortium Network * and Orofacial Pain Special Interest Group. **J Oral Facial Pain Headache**, v. 28, n. 1, p. 6–27, 2014.

SCHWENDICKE, F.; SAMEK, W.; KROIS, J. Artificial Intelligence in Dentistry: Chances and Challenges. **Journal of Dental Research**, v. 99, n. 7, p. 769–774, 21 jul. 2020.

STEENKS, M.; TÜRP, J.; DE WIJER, A. Reliability and Validity of the Diagnostic Criteria for Temporomandibular Disorders Axis I in Clinical and Research Settings: A Critical Appraisal. **J Oral Facial Pain Headache**, v. 32, n. 1, p. 7–18, jan. 2018.

STIGLIC, G. et al. Comprehensive Decision Tree Models in Bioinformatics. **PLoS ONE**, v. 7, n. 3, p. e33812, 30 mar. 2012.

SVENSSON, P.; BENDIXEN, K. Svensson P. Critical commentary: Validity of the Research Diagnostic Criteria for Temporomandibular Disorders Axis I in clinical and research settings. **J Oral Facial Pain Headache**, v. 32, n. 1, p. 24–25, 2018.

THOMAS, D. C. et al. Systemic Factors in Temporomandibular Disorder Pain. **Dental Clinics of North America**, v. 67, n. 2, abr. 2023.

THRALL, J. H. et al. Artificial Intelligence and Machine Learning in Radiology: Opportunities, Challenges, Pitfalls, and Criteria for Success. **J Am Coll Radiol**, v. 15, n. 3, p. 504–508, 1 mar. 2018.

YANG, D.; FINEBERG, H. V.; COSBY, K. Diagnostic Excellence. **JAMA**, v. 326, n. 19, p. 1905, 16 nov. 2021.

APÊNDICE A – Critérios Retirados Dos Prontuários Do CEMDOR

Critério	Possível resposta				
Sexo	feminino	masculino			
Queixa principal	relato do paciente				
Queixa Principal					
Início	até 1 semana	até 1 mês	até 6 meses	até 1 ano	Mais de 1 ano
Eventos Relacionados	traumas	trat. odontológico	trat. ortodôntico	estresse	
	perdas	trauma emocional	abertura excessiva	outros	
Localização	ATM	Face	Cabeça	Pescoço	Outro
Frequência	constante	1x semana	1x mês	esporádica	Após função	outro
Intensidade	incapacitante	severa	moderada	leve	incômodo
Intensidade de dor (EAV)	Valor entre 0 e 10 onde: 0= sem dor 10= pior dor imaginável				
Intensidade de dor (EAV) - média da semana	Valor entre 0 e 10 onde: 0= sem dor 10= pior dor imaginável				
Intensidade de dor (EAV) - média do mês	Valor entre 0 e 10 onde: 0= sem dor 10= pior dor imaginável				
Comorbidades					
Artrite Reumatoide	sim	não			
Osteoartrose	sim	não			
S. de Sjögren	sim	não			
Fibromialgia	sim	não			
Lúpus	sim	não			
Cervicalgia	sim	não			
S. Ardência bucal	sim	não			
Ouvido					
Zumbido					
Plenitude auricular					
Hábitos Ocupacionais					
Uso cont. telefone	sim	não			
Uso cont. computador	sim	não			
Tabagismo	sim	não			
Etilismo	sim	não			
Café	sim	não			
Hábitos Parafuncionais					
Relato Apertamento – noite	sim	não			

Dor de cabeça ao acordar	sim	não		
Travamento mand. ao acordar	sim	não		
Relato apertamento/ranger – dia	sim	não		
Dentes encostados ao acordar	sim	não		
Fatores psicossociais				
Ansiedade	sim	não		
Exame físico				
Assimetria facial	sim	não		
Abertura s/ dor	Valor em mm			
Abertura max s/ auxílio	Valor em mm	**Dor:**	sim	não
Abertura max c/ auxílio	Valor em mm	**Dor:**	sim	não
Abertura	simétrica	desvio	deflexão	Em “S”
Ruído de adesão	sim	não		
Estalo na abertura (início)	sim	não		
Estalo no fechamento	sim	não		
Estalo na abertura (final)	sim	não		
Crepitação	sim	não		
Hipermobilidade	sim	não		
Hipermobilidade no corpo	sim	não		
História de travamento	sim	não		
Dificuldade de abertura/fechamento	não	abertura	fechamento	ambos
Palpação da ATM e musculatura				
Dor a palpação – ATM	0 = sem dor	1 = leve	2 = moderada	3 = severa
Localização da dor – ATM	localizada	espalhamento	referência	
Dor a palpação – ATM (posterior)	0 = sem dor	1 = leve	2 = moderada	3 = severa
Localização da dor – ATM (posterior)	localizada	espalhamento	referência	
Dor a palpação – masseter	0 = sem dor	1 = leve	2 = moderada	3 = severa
Localização da dor – masseter	localizada	espalhamento	referência	
Dor a palpação – temporal	0 = sem dor	1 = leve	2 = moderada	3 = severa

APÊNDICE B - Modelo De Árvore De Decisão Para Dores Da ATM

Objetivo do Modelo

Classificar as dores articulares em 2 categorias diferentes baseado no ICOP e uma categoria de "não-doença"

Bibliotecas

Esta seção apresenta as bibliotecas importadas como recursos para:

* Análise inicial do banco de dados
* Pré-processamento de dados
* Construção e visualização de gráficos
* Construção e treinamento do modelo de árvore de decisão
* Avaliação do modelo

Ainda, são apresentados comandos para configurar o comportamento do ambiente e habilitar algumas funcionalidades.

```
#Análise de dados e visualização
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# Construção, treinamento e avaliação do modelo de árvore de decisão
import sklearn
#Criação de gráficos
import seaborn as sns
import graphviz
#Codificação de variáveis categóricas
from sklearn.preprocessing import LabelEncoder
```

Comportamento do ambiente

```
#Recarregar automaticamente módulos importados sempre que ocorrerem
alterações nesses módulos
%load_ext autoreload
#Definição da configuração da extensão autoreload. O valor 2 significa
que os módulos serão recarregados automaticamente sempre que forem
importados.
%autoreload 2
#Configuração da renderização de gráficos do Matplotlib diretamente na
célula do notebook, permitindo a exibição dos gráficos no próprio
notebook.
%matplotlib inline
```

Preparo Inicial dos Dados

Essa seção importa a tabela com dados brutos e realiza o preparo para posterior análise (i.e., remoção de colunas com muitos espacos vazios, substituição de valores e etc).

```
#variáveis editáveis, onde DATA é o arquivo para ser importado
DATA = "..."
#Importar excel e substituir campo em N/A por vazio
df = pd.read_excel(DATA)
df = df.replace('N/A', '')

#Secciona a tabela em "primeiras colunas" = colunas que indicam o
diagnóstico e "segundas colunas" = dados utilizados para o diagnóstico
#primeiras_colunas1 = df.iloc[:, 5:6]
primeiras_colunas2 = df.iloc[:, 10:11]
segundas_colunas = df.iloc[:, 14:]

# Criando um novo DataFrame com as colunas de interesse. Onde df_limpo1
indica o diagnóstico de dores miofasciais e df_limpo2 indica o
diagnóstico de dores articulares
#df_limpo1 = pd.concat([primeiras_colunas1, segundas_colunas], axis=1)
df_limpo2 = pd.concat([primeiras_colunas2, segundas_colunas], axis=1)

# Salvando o novo arquivo Excel
#df_limpo1.to_excel('ICOP_1_limpo.xlsx', index=False)
df_limpo2.to_excel('ICOP_2_limpo.xlsx', index=False)

#Visualizalção do topo do DataFrame a ser utilizado na criação deste
modelo (Dores Miofasciais)
df_limpo2.head()
```

Análise Inicial dos Dados

Esta seção contém análise dos dados brutos da tabela

```
# Contagem dos diagnósticos de Dores Miofasciais e Dores Articulares
num_repeticoes = df['Diagnóstico_ICOP'].value_counts()
print("O número indivíduos com Dor Miofascial, diagnóstico ",
num_repeticoes)
num_repeticoes2 = df['Diagnóstico_ICOP2'].value_counts()
print("O número indivíduos com Dor Articular, diagnóstico ",
num_repeticoes2)

# Contagem dos valores em branco por coluna
for col in df.columns:
    # conta o número de valores em branco na coluna
    num_null = df[col].isnull().sum()
```

```
    # calcula a porcentagem de valores em branco na coluna
    pct_null = num_null / len(df) * 100
    # exibe o resultado
    print(f'A coluna {col} tem {pct_null:.2f}% de valores em branco.')
```

Pré-processamento dos dados

O pré-processamento de dados para árvores de decisão envolve uma série de etapas que visam preparar os dados de entrada de forma adequada para o treinamento e aplicação de um modelo de árvore de decisão. Essas etapas são importantes para melhorar a qualidade dos resultados e o desempenho do modelo. As etapas do pré-processamento de dados incluem:

* Codificação de variáveis categóricas
* Padronização de variáveis
* Divisão do conjunto de dados em treinamento e teste

```
#Separação da variável alvo das demais
df_X = df_limpo2.drop('icop2_sub1', axis=1)
df_y = df_limpo2['icop2_sub1']

df_X = df_X.iloc[:, :]
df_X

#tipos de dados de cada coluna em df_x
df_X.dtypes
#Nomeando colunas categoricas e numéricas
categoric_cols = ["Sexo",
                  "Queixa_principal_1",
                  "Início",
                  "Eventos_Relacionados",
                  "Localização",
                  "Frequência_",
                  "Intensidade",
                  "Artrite_R.",
                  "Osteoartrose",
                  "Sjogren",
                  "Fibromialgia",
                  "Lupus",
                  "Cervicalgia",
                  "Ardência_bucal",
                  "Zumbido",
                  "Plenitude_auricular",
                  "Uso_cont._telefone",
                  "Uso_cont._computador",
                  "Tabagismo",
                  "Etilismo",
```

```
                    "Café",
                    "Relato_apertamento/noite",
                    "Dor_de_cabeça_ao_acordar",
                    "Travamento_mand._Ao_acordar",
                    "Relato_apertar_ranger/dia",
                    "Dentes_encostados_ac",
                    "Ansioso",
                    "Assimetria_facial_",
                    "dor",
                    "dor.1",
                    "Abertura",
                    "Ruído_de_ad.",
                    "Estalo_abertura_(início)",
                    "Estalo_fechamento",
                    "Estalo_abertura_(final)",
                    "Crepitação",
                    "Hipermobilidade",
                    "Hipermob._Corpo",
                    "História_de_travamento",
                    "Dificuldade_abertura/fechamento",
                    "ATM",
                    "LOCALIZAÇÃO_ATM",
                    "ATM_post",
                    "LOCALIZAÇÃO_ATMPOST",
                    "Masseter",
                    "LOCALIZAÇÃO_masseter",
                    "Temporal",
                    "LOCALIZACAO_temporal"]
categoric_as_number_cols = []
numeric_cols = ["EAV1",
                "EAV1_semana",
                "EAV1_mês",
                "abertura_s/_dor_(+overbite)",
                "abertura_max_s/_auxílio_(+overbite)",
                "abertura_max_c/auxilio_(+overbite)"]

#contar a quantidade de valores ausentes (NaN) em cada coluna do
DataFrame
nan_count = df_X.isna().sum()
print(nan_count )

#Codificação de variáveis categóricas
for coluna in categoric_cols:
    nova_coluna = LabelEncoder()
    df_X[coluna] = nova_coluna.fit_transform(df_X[coluna])
df_X
```

Construção e treinamento do modelo de Árvore de Decisão

Esta seção apresenta o ajuste de hiperparâmetros, divisão dos conjuntos de treinamento e teste e o treinamento do modelo

```
#Função que permite dividir um conjunto de dados em subconjuntos de
treinamento e teste.
from sklearn.model_selection import train_test_split
#Classe da Biblioteca Scikit-learn que permite construir, treinar e
fazer previsões com base em uma árvore de decisão.
from sklearn.tree import DecisionTreeClassifier
#Função da Biclioteca Scikit-learn que é usada para calcular a precisão
de um modelo de classificação.
from sklearn.metrics import accuracy_score

#Divisão do conjunto de dados em subconjuntos de treinamento e teste
definindo parâmetros:
X_train, X_test, y_train, y_test = train_test_split(df_X, df_y,
test_size=0.3, random_state=100, stratify=df_y)

#variável para acompanhar a melhor precisão (accuracy) alcançada durante
o treinamento e ajuste dos hiperparâmetros do modelo
best_acc = 0

#Loop para encontrar os melhores hiperparâmetros para um modelo
for criterion in "gini", "entropy":
    print(criterion)
    for max_depth in [2,3,4,5]:
        print("depth", max_depth)
        for min_samples_leaf in [5, 10, 20, 25, 30]:
            #print("Min", min_samples_leaf)
            dtree = DecisionTreeClassifier(max_depth=max_depth,
criterion=criterion, min_samples_leaf=min_samples_leaf)
            dtree.fit(X_train, y_train)
            predicitions = dtree.predict(X_test)
            acc = accuracy_score(y_test, predicitions)
            print(acc)
            if acc > best_acc:
                best_params = f"criterion: {criterion}, max_depth:
{max_depth}, min_samples_leaf: {min_samples_leaf}"
                best_acc = acc

#Exibir os melhores parâmetros encontrados e a melhor precisão
(accuracy) alcançada durante o processo de busca dos hiperparâmetros
print(best_params)
print(best_acc)
```

```
#Criação de um objeto DecisionTreeClassifier com os parâmetros
especificados
dtree = DecisionTreeClassifier(criterion='gini', max_depth=3,
min_samples_leaf=5)
#Treinamento usando os dados de treinamento X_train e os rótulos
correspondentes y_train
dtree.fit(X_train, y_train)

#Criação da figura da árvore de decisão
from sklearn.tree import plot_tree
from matplotlib import pyplot as plt

fig = plt.figure(figsize=((25,20)))
plot_tree(dtree,
            feature_names = df_X.columns,
            class_names=['Dor primária na ATM', 'Dor secundária na ATM',
'Não-doença'],
            impurity=False,
            proportion=True,
            filled=True,
            fontsize=12)

#Importância dos critérios utilizados
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_text
r = export_text(dtree, feature_names=df_X.columns.tolist())
print(r)

#Verificação os valores específicos dos parâmetros usados no modelo
treinado
dtree.get_params()
#Previsões para os dados de teste X_test e armazenamento das previsões
na variável 'predictions'
predictions = dtree.predict(X_test)
#obtenção das probabilidades estimadas para cada classe para as amostras
nos dados de teste X_test.
proba = dtree.predict_proba(X_test)
#Impressão das previsões feitas pelo modelo na saída e probabilidades
estimadas
print(predictions)
print(proba)

#Impressão da matriz que contém os dados do conjunto de treinamento ou
do conjunto de características.
feature_names = df_X.columns
feature_names
```

```
#Armazenamento um DataFrame ordenado pelas importâncias das
características, em que as características mais relevantes estão
listadas no início.
feature_importance = pd.DataFrame(dtree.feature_importances_, index =
feature_names).sort_values(0, ascending=False)
feature_importance

#Figura das colunas utilizadas para treinamento
feature_importance.head(10).plot(kind='bar')
plt.title("Colunas utilizadas para treinamento")
```

Avaliação do modelo criado

Nesta seção a avaliação do um modelo de Árvore de Decisão é feita usando métricas e técnicas:

* Precisão (Accuracy): Proporção de previsões corretas em relação ao total de previsões. Quanto maior a precisão, melhor o desempenho do modelo.
* Matriz de Confusão (Confusion Matrix): Análise da taxa de acertos e erros para cada classe.
* Precisão (Precision), Recall e F1-Score: A precisão mede a proporção de previsões corretas para uma classe específica, o recall mede a proporção de instâncias verdadeiras corretamente identificadas e o F1-score é uma métrica que combina a precisão e o recall.

```
#calculo da precisão (accuracy) do modelo de classificação em relação
aos rótulos verdadeiros (y_test) e as previsões feitas pelo modelo
(predictions).
accuracy_score(y_test, predictions)

#Função da Biblioteca Scikit-learn usada para calcular a matriz de
confusão
from sklearn.metrics import confusion_matrix
#Cálculo a partir dos rótulos verdadeiros (y_test) e das previsões
feitas pelo modelo (predictions).
confusion_matrix(y_test, predictions)
#Função da Biblioteca Scikit-learn usada para calcular a matriz de
confusão
from sklearn.metrics import confusion_matrix
#Cálculo a partir dos rótulos verdadeiros (y_test) e das previsões
feitas pelo modelo (predictions).
confusion_matrix= confusion_matrix (y_test, predictions)
class_names = ['Dor primária na ATM', 'Dor secundária na ATM', 'Não-
doença']
```

```
fig, ax = plt.subplots()
sns.heatmap(confusion_matrix, annot=True, cmap='Blues', fmt='d',
xticklabels=class_names, yticklabels=class_names)
ax.set_xlabel('Classe Prevista')
ax.set_ylabel('Classe Verdadeira')
ax.set_title('Matriz de Confusão')
plt.show()


#Função da Biblioteca Scikit-learn usada para calcular o recall ( taxa
de verdadeiros positivos (TPR))
from sklearn.metrics import recall_score

# Calcular o recall para cada classe individualmente
recall_per_class = recall_score(y_test, predictions, average=None)
print("Recall por classe:", recall_per_class)

# Calcular o recall médio ponderado
weighted_recall = recall_score(y_test, predictions, average='weighted')
print("Recall médio ponderado:", weighted_recall)

#Função da Biblioteca Scikit-learn usada para gerar um relatório de
classificação com métricas de avaliação para um problema de
classificação.
from sklearn.metrics import classification_report
#Relatório de classificação impresso no formato de tabela, contendo as
métricas de avaliação para cada classe, bem como a média ponderada de
todas as classes.
print(classification_report(y_test, predicitions, zero_division=1,
target_names=['Dor primária na ATM', 'Dor secundária na ATM', 'Não-
doença']))
```

APÊNDICE C – Modelo De Árvore De Decisão Para Dores Miofasciais Orofaciais

Objetivo do Modelo

Classificar as dores miofasciais em 3 categorias diferentes baseado na Classificação Internacional de Dor Orofacial, Primeira Edição (ICOP)

Bibliotecas

Esta seção apresenta as bibliotecas importadas como recursos para:

* Análise inicial do banco de dados
* Pré-processamento de dados
* Construção e visualização de gráficos
* Construção e treinamento do modelo de árvore de decisão
* Avaliação do modelo

Ainda, são apresentados comandos para configurar o comportamento do ambiente e habilitar algumas funcionalidades.

```
#Análise de dados e visualização
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# Construção, treinamento e avaliação do modelo de árvore de decisão
import sklearn
#Criação de gráficos
import seaborn as sns
import graphviz
#Codificação de variáveis categóricas
from sklearn.preprocessing import LabelEncoder
```

Comportamento do ambiente

```
#Recarregar automaticamente módulos importados sempre que ocorrerem
alterações nesses módulos
%load_ext autoreload
#Definição da configuração da extensão autoreload. O valor 2 significa
que os módulos serão recarregados automaticamente sempre que forem
importados.
%autoreload 2
#Configuração da renderização de gráficos do Matplotlib diretamente na
célula do notebook, permitindo a exibição dos gráficos no próprio
notebook.
%matplotlib inline
```

Preparo Inicial dos Dados

Essa seção importa a tabela com dados brutos e realiza o preparo para posterior análise (i.e., remoção de colunas com muitos espacos vazios, substituição de valores e etc)

```
#variáveis editáveis, onde DATA é o arquivo para ser importado
DATA = "..."
#Importar excel e substituir campo em N/A por vazio
df = pd.read_excel(DATA)
df = df.replace('N/A', '')

#Secciona a tabela em "primeiras colunas" = colunas que indicam o
diagnóstico e "segundas colunas" = dados utilizados para o diagnóstico
primeiras_colunas1 = df.iloc[:, 5:6]
primeiras_colunas2 = df.iloc[:, 11:12]
segundas_colunas = df.iloc[:, 14:]

# Criando um novo DataFrame com as colunas de interesse. Onde df_limpo1
indica o diagnóstico de dores miofasciais e df_limpo2 indica o
diagnóstico de dores articulares
df_limpo1 = pd.concat([primeiras_colunas1, segundas_colunas], axis=1)
df_limpo2 = pd.concat([primeiras_colunas2, segundas_colunas], axis=1)

# Salvando o novo arquivo Excel
df_limpo1.to_excel('ICOP_1_limpo.xlsx', index=False)
df_limpo2.to_excel('ICOP_2_limpo.xlsx', index=False)

#Visualizalção do topo do DataFrame a ser utilizado na criação deste
modelo (Dores Miofasciais)
df_limpo1.head()
```

Análise Inicial dos Dados

Esta seção contém análise dos dados brutos da tabela

```
# Contagem dos diagnósticos de Dores Miofasciais e Dores Articulares
num_repeticoes = df['Diagnóstico_ICOP'].value_counts()
print("O número indivíduos com Dor Miofascial, diagnóstico ",
num_repeticoes)
num_repeticoes2 = df['Diagnóstico_ICOP2'].value_counts()
print("O número indivíduos com Dor Articular, diagnóstico ",
num_repeticoes2)

# Contagem dos valores em branco por coluna
for col in df.columns:
    # conta o número de valores em branco na coluna
```

```
    num_null = df[col].isnull().sum()
    # calcula a porcentagem de valores em branco na coluna
    pct_null = num_null / len(df) * 100
    # exibe o resultado
    print(f'A coluna {col} tem {pct_null:.2f}% de valores em branco.')
```

Pré-processamento dos dados

O pré-processamento de dados para árvores de decisão envolve uma série de etapas que visam preparar os dados de entrada de forma adequada para o treinamento e aplicação de um modelo de árvore de decisão. Essas etapas são importantes para melhorar a qualidade dos resultados e o desempenho do modelo. As etapas do pré-processamento de dados incluem:

* Codificação de variáveis categóricas
* Padronização de variáveis
* Divisão do conjunto de dados em treinamento e teste

```
#Separação da variável alvo das demais
df_X = df_limpo1.drop('icop1_sub2', axis=1)
df_y = df_limpo1['icop1_sub2']

df_X = df_X.iloc[:, :]
df_X


#tipos de dados de cada coluna em df_x
df_X.dtypes


#Nomeando colunas categoricas e numéricas
categoric_cols = ["Sexo",
                  "Queixa_principal_1",
                  "Início",
                  "Eventos_Relacionados",
                  "Localização",
                  "Frequência_",
                  "Intensidade",
                  "Artrite_R.",
                  "Osteoartrose",
                  "Sjogren",
                  "Fibromialgia",
                  "Lupus",
                  "Cervicalgia",
                  "Ardência_bucal",
                  "Zumbido",
                  "Plenitude_auricular",
```

```
                    "Uso_cont._telefone",
                    "Uso_cont._computador",
                    "Tabagismo",
                    "Etilismo",
                    "Café",
                    "Relato_apertamento/noite",
                    "Dor_de_cabeça_ao_acordar",
                    "Travamento_mand._Ao_acordar",
                    "Relato_apertar_ranger/dia",
                    "Dentes_encostados_ac",
                    "Ansioso",
                    "Assimetria_facial_",
                    "dor",
                    "dor.1",
                    "Abertura",
                    "Ruído_de_ad.",
                    "Estalo_abertura_(início)",
                    "Estalo_fechamento",
                    "Estalo_abertura_(final)",
                    "Crepitação",
                    "Hipermobilidade",
                    "Hipermob._Corpo",
                    "História_de_travamento",
                    "Dificuldade_abertura/fechamento",
                    "ATM",
                    "LOCALIZAÇÃO_ATM",
                    "ATM_post",
                    "LOCALIZAÇÃO_ATMPOST",
                    "Masseter",
                    "LOCALIZAÇÃO_masseter",
                    "Temporal",
                    "LOCALIZACAO_temporal"]
categoric_as_number_cols = []
numeric_cols = ["EAV1",
                "EAV1_semana",
                "EAV1_mês",
                "abertura_s/_dor_(+overbite)",
                "abertura_max_s/_auxílio_(+overbite)",
                "abertura_max_c/auxilio_(+overbite)"]


#contar a quantidade de valores ausentes (NaN) em cada coluna do
DataFrame
nan_count = df_X.isna().sum()
print(nan_count )


#Conversão dos valores em números inteiros usando int(x) e adição do
prefixo "cat_" aos valores convertidos.
def append_cat(x):
```

```
    return f"cat_{int(x)}"
#A função to_int recebe um valor x como entrada e o converte em um
número inteiro usando int(x).
def to_int(x):
    return int(x)
#Loop para iteração sobre as colunas presentes em
categoric_as_number_cols. Feito para converter os valores categóricos
representados como números em números inteiros.
for coluna in categoric_as_number_cols:
    df_X[coluna] = df_X[coluna].apply(to_int)
df_X
#Verificando o tipo de dados de cada coluna em df_x após conversão
df_X.dtypes
#Codificação de variáveis categóricas
for coluna in categoric_cols:
    nova_coluna = LabelEncoder()
    df_X[coluna] = nova_coluna.fit_transform(df_X[coluna])
df_X
```

Construção e treinamento do modelo de Árvore de Decisão

Esta seção apresenta o ajuste de hiperparâmetros, divisão dos conjuntos de treinamento e teste e o treinamento do modelo

```
#Função que permite dividir um conjunto de dados em subconjuntos de
treinamento e teste.
from sklearn.model_selection import train_test_split
#Classe da Biblioteca Scikit-learn que permite construir, treinar e
fazer previsões com base em uma árvore de decisão.
from sklearn.tree import DecisionTreeClassifier
#Função da Biclioteca Scikit-learn que é usada para calcular a precisão
de um modelo de classificação.
from sklearn.metrics import accuracy_score

#Divisão do conjunto de dados em subconjuntos de treinamento e teste
definindo parâmetros:
X_train, X_test, y_train, y_test = train_test_split(df_X, df_y,
test_size=0.3, random_state=15, stratify=df_y)
#variável para acompanhar a melhor precisão (accuracy) alcançada durante
o treinamento e ajuste dos hiperparâmetros do modelo
best_acc = 0
#Loop para encontrar os melhores hiperparâmetros para o modelo
for criterion in "gini", "entropy":
    print(criterion)
    for max_depth in [2,3,4,5]:
        print("depth", max_depth)
        for min_samples_leaf in [5, 10, 20, 25, 30]:
            #print("Min", min_samples_leaf)
```

```
            dtree = DecisionTreeClassifier(max_depth=max_depth,
criterion=criterion, min_samples_leaf=min_samples_leaf)
            dtree.fit(X_train, y_train)
            predictions = dtree.predict(X_test)
            acc = accuracy_score(y_test, predictions)
            print(acc)
            if acc > best_acc:
                best_params = f"criterion: {criterion}, max_depth:
{max_depth}, min_samples_leaf: {min_samples_leaf}"
                best_acc = acc
#Exibir os melhores parâmetros encontrados e a melhor precisão
(accuracy) alcançada durante o processo de busca dos hiperparâmetros
print(best_params)
print(best_acc)


#Criação de um objeto DecisionTreeClassifier com os parâmetros
especificados
dtree = DecisionTreeClassifier(criterion='entropy', max_depth=4,
min_samples_leaf=5)
#Treinamento usando os dados de treinamento X_train e os rótulos
correspondentes y_train
dtree.fit(X_train, y_train)


#Figura da árvore de decisão
from sklearn.tree import plot_tree
from matplotlib import pyplot as plt
fig = plt.figure(figsize=(35, 15))
plot_tree(dtree,
            feature_names = df_X.columns,
            class_names=['Orofacial primária', 'Orofacial primária
aguda', 'Orofacial primária crônica'],
            impurity=False,
            proportion=True,
            filled=True,
            fontsize=14)
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_text
r = export_text(dtree, feature_names=df_X.columns.tolist())
print(r)


#Verificação os valores específicos dos parâmetros usados no modelo
treinado
dtree.get_params()
#Previsões para os dados de teste X_test e armazenamento das previsões
na variável 'predictions'
predictions = dtree.predict(X_test)
```

```
#obtenção das probabilidades estimadas para cada classe para as amostras
nos dados de teste X_test.
proba = dtree.predict_proba(X_test)
#Impressão das previsões feitas pelo modelo na saída e probabilidades
estimadas
print(predictions)
print(proba)
#Impressão da matriz que contém os dados do conjunto de treinamento ou
do conjunto de características.
feature_names = df_X.columns
feature_names
#Armazenamento um DataFrame ordenado pelas importâncias das
características, em que as características mais relevantes estão
listadas no início.
feature_importance = pd.DataFrame(dtree.feature_importances_, index =
feature_names).sort_values(0, ascending=False)
feature_importance
plt.style.use('tableau-colorblind10')
feature_importance.head(10).plot(kind='bar')
plt.title("Colunas utilizadas para treinamento")
```

Avaliação do modelo criado

Nesta seção a avaliação do um modelo de Árvore de Decisão é feita usando métricas e técnicas:

* Precisão (Accuracy): Proporção de previsões corretas em relação ao total de previsões. Quanto maior a precisão, melhor o desempenho do modelo.
* Matriz de Confusão (Confusion Matrix): Análise da taxa de acertos e erros para cada classe.
* Precisão (Precision), Recall e F1-Score: A precisão mede a proporção de previsões corretas para uma classe específica, o recall mede a proporção de instâncias verdadeiras corretamente identificadas e o F1-score é uma métrica que combina a precisão e o recall.

```
#cálculo da precisão (accuracy) do modelo de classificação em relação
aos rótulos verdadeiros (y_test) e as previsões feitas pelo modelo
(predictions).
accuracy_score(y_test, predictions)
#Função da Biblioteca Scikit-learn usada para calcular a matriz de
confusão
from sklearn.metrics import confusion_matrix
#Cálculo a partir dos rótulos verdadeiros (y_test) e das previsões
feitas pelo modelo (predictions).
```

```
confusion_matrix(y_test, predictions)
#Função da Biblioteca Scikit-learn usada para calcular a matriz de
confusão
from sklearn.metrics import confusion_matrix
#Cálculo a partir dos rótulos verdadeiros (y_test) e das previsões
feitas pelo modelo (predictions).
confusion_matrix= confusion_matrix (y_test, predictions)
class_names = ['Orofacial primária', 'Orofacial primária aguda',
'Orofacial primária crônica']
fig, ax = plt.subplots()
sns.heatmap(confusion_matrix, annot=True, cmap='Blues', fmt='d',
xticklabels=class_names, yticklabels=class_names)
ax.set_xlabel('Classe Prevista')
ax.set_ylabel('Classe Verdadeira')
ax.set_title('Matriz de Confusão')
plt.show()

#Função da Biblioteca Scikit-learn usada para calcular o recall ( taxa
de verdadeiros positivos (TPR))
from sklearn.metrics import recall_score
# Calcular o recall para cada classe individualmente
recall_per_class = recall_score(y_test, predictions, average=None)
print("Recall por classe:", recall_per_class)

# Calcular o recall médio ponderado
weighted_recall = recall_score(y_test, predictions, average='weighted')
print("Recall médio ponderado:", weighted_recall)


#Função da Biblioteca Scikit-learn usada para gerar um relatório de
classificação com métricas de avaliação para um problema de
classificação.
from sklearn.metrics import classification_report
#Relatório de classificação impresso no formato de tabela, contendo as
métricas de avaliação para cada classe, bem como a média ponderada de
todas as classes.
print(classification_report(y_test, predictions, zero_division=1,
target_names=['Orofacial primária', 'Orofacial primária aguda',
'Orofacial primária crônica']))
```

ANEXO A – Aprovação Em Comitê De Ética Em Pesquisa Com Seres Humanos

UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC

PARECER CONSUBSTANCIADO DO CEP

DADOS DO PROJETO DE PESQUISA

Título da Pesquisa: Utilização de deep learning na classificação das disfunções temporomandibulares: prova de conceito

Pesquisador: RICARDO ARMINI CALDAS

Área Temática:

Versão: 4

CAAE: 60532822.8.0000.0121

Instituição Proponente: Departamento de Odontologia

Patrocinador Principal: Financiamento Próprio

DADOS DO PARECER

Número do Parecer: 5.737.653

Apresentação do Projeto:

Segundo pesquisador: " O projeto tem por objetivo avaliar a aplicabilidade de aprendizagem profunda (deep learning) na detecção e classificação de pacientes com disfunções temporomandibulares (DTMs). A etapa de aprendizado de máquina será por meio do método de redes neurais convolucionais, onde será utilizada programação em linguagem Python e as ferramentas TensorFlow e Keras. Para a construção do modelo, será utilizado o conjunto de dados de prontuários do acervo de Prontuários Clínicos do Centro Multidisciplinar de Dor Orofacial (CEMDOR) da Universidade Federal de Santa Catarina. Cada conjunto de dados possuirá uma quantidade padronizada de informações básicas e dimensões para cada informação. Os diagnósticos de DTM serão classificados de acordo com o Diagnostic Criteria for Temporomandibular Disorders (DC/TMD) para desordens temporomandibulares e desordens dos músculos mastigatórios. A hipótese de pesquisa é que a aplicação de inteligência artificial permitirá identificar e classificar as disfunções temporomandibulares por meio do prontuário clínico do paciente de modo automatizado. Amostragem: dados secundários do acervo de Prontuários Clínicos (n=135) dos últimos 05 anos do Centro Multidisciplinar de Dor Orofacial (CEMDOR) da Universidade Federal de Santa Catarina, sob responsabilidade da Professora Coordenadora Beatriz Dulcinéia Mendes de Souza. Os acervos contêm um diretório com as fichas clínicas de pacientes ao longo dos anos, contendo as informações da doença e identificação do paciente. Critérios de inclusão: prontuários de pacientes

Endereço: Universidade Federal de Santa Catarina, Prédio Reitoria II, R: Desembargador Vitor Lima, nº 222, sala 701
Bairro: Trindade **CEP:** 88.040-400
UF: SC **Município:** FLORIANOPOLIS
Telefone: (48)3721-6094 **E-mail:** cep.propesq@contato.ufsc.br

com diagnóstico de DTM, que possuam a documentação completa do atendimento do Centro Multidisciplinar de Dor Orofacial da Universidade Federal de Santa Catarina, com letra legível nas partes subjetivas que são descritas com texto corrido. Critérios de exclusão: todos os pacientes menores de 18 anos e pacientes que possuam prontuário clínico incompleto."

Considerações éticas mencionadas pelos pesquisadores: "Nesse estudo não serão relatados casos clínicos, nem se trata de coletar informações que permitam identificação dos pacientes nos prontuários, apenas utilização das informações de anamnese utilizadas para diagnóstico. Previamente ao atendimento no CEMDOR, todos os pacientes concordam que as radiografias, fotografias, modelos, desenhos, histórico de saúde, resultados de exames e quaisquer outras informações referentes ao diagnóstico, planejamento e/ou tratamento possam ser usados como material didático e acadêmico, respeitando o Código de Ética Odontológica e o Sigilo Profissional, sempre preservando o direito de não identificação, ou seja, todos os pacientes atendidos pelo CEMDOR assinaram um termo de concordância com o uso de tais dados."

Procedimentos: "Assim, um operador realizará a triagem dos prontuários clínicos respeitando os critérios de inclusão e exclusão e transcreverá as informações para uma base de dados digital. Os dados serão armazenados no software REDcap. É um sistema gerenciador de banco de dados relacional de código aberto usado gratuitamente para gerir bases de dados. O serviço utiliza a linguagem SQL (Structure Query Language) e que permite a criptografia dos dados para maior segurança. A partir do número total de pacientes, serão removidos aleatoriamente 30% do total de pacientes que serão utilizados como inputs testes, para verificar o erro de diagnóstico do algoritmo. Sendo assim, o algoritmo será realizado com 70% das fichas clínicas, os 30% que restam serão utilizados para descobrir o erro.

Como aplicabilidade futura, depois de criado o algoritmo e com acurácia satisfatória, a proposta é expandir sua aplicação com a criação de um plugin ou software para disponibilização a demais profissionais que tenham interesse na área. Serão aventadas possibilidades de criação de patentes e registro dos algoritmos e modelos. Técnicas de reamostragem para otimização de hiperparâmetros serão aplicadas, utilizando parâmetros usuais e parâmetros de ajuste. Esta parte do projeto será realizada com a participação de um profissional cientista de dados, integrante da equipe deste projeto."

Objetivo da Pesquisa:

Segundo pesquisador: "Criar um algoritmo de aprendizagem profunda (deep learning) com capacidade para sistematizar dados que auxiliem na detecção e classificação das disfunções temporomandibulares (DTMs)."

Objetivos Secundários: "avaliar a acurácia do modelo proposto comparado a um padrão ouro; aprimorar os métodos de diagnóstico em DTM."

Avaliação dos Riscos e Benefícios:

Adequadamente contemplados.

Comentários e Considerações sobre a Pesquisa:

Vide campo "Conclusões ou Pendências e Lista de Inadequações".

Considerações sobre os Termos de apresentação obrigatória:

Vide campo "Conclusões ou Pendências e Lista de Inadequações".

Recomendações:

Vide campo "Conclusões ou Pendências e Lista de Inadequações".

Conclusões ou Pendências e Lista de Inadequações:

Não apresenta pendências e/ou inadequações.

Considerações Finais a critério do CEP:

Lembramos que a presente aprovação (versão projeto 04/10/2022 e TCUD 03/11/2022) refere-se apenas aos aspectos éticos do projeto. Qualquer alteração nestes documentos deve ser encaminhada para avaliação do CEPSH. Informamos que a dispensa de TCLE somente será utilizada para este projeto. Todo e qualquer outro uso que venha a ser planejado, será, obrigatoriamente, objeto de um novo projeto de pesquisa, o qual será submetido à apreciação do CEPSH-UFSC.

Lembramos aos senhores pesquisadores que o CEPSH/UFSC deverá receber, por meio de notificação, os relatórios parciais sobre o andamento da pesquisa e o relatório completo ao final do estudo.

Este parecer foi elaborado baseado nos documentos abaixo relacionados:

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações Básicas do Projeto	PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_1965847.pdf	03/11/2022 11:38:25		Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	Termo_de_Compromisso_para_Uso_de_Dados_assinado.pdf	03/11/2022 11:36:44	FERNANDA PRETTO ZATT	Aceito
Outros	Declaracao_arquivo.pdf	04/10/2022 18:04:40	FERNANDA PRETTO ZATT	Aceito
Outros	Declaracao_responsavel.pdf	04/10/2022 18:03:14	FERNANDA PRETTO ZATT	Aceito

Outros	emenda_comite.pdf	04/10/2022 18:01:26	FERNANDA PRETTO ZATT	Aceito
Outros	carta_resposta_as_pendencias.pdf	04/10/2022 18:00:48	FERNANDA PRETTO ZATT	Aceito
Projeto Detalhado / Brochura Investigador	Projeto_versao3.pdf	04/10/2022 17:58:57	FERNANDA PRETTO ZATT	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	Dispensa_TCLE_atualizado.pdf	25/08/2022 14:29:38	FERNANDA PRETTO ZATT	Aceito
Folha de Rosto	folha_De_Rosto.pdf	28/06/2022 22:32:23	FERNANDA PRETTO ZATT	Aceito
Declaração de Instituição e Infraestrutura	Declaracao_CCS.pdf	28/06/2022 17:12:03	FERNANDA PRETTO ZATT	Aceito

Situação do Parecer:
Aprovado

Necessita Apreciação da CONEP:
Não

FLORIANOPOLIS, 03 de Novembro de 2022

Assinado por:
Luciana C Antunes
(Coordenador(a))

ANEXO B – Dispensa Do Termo De Consentimento Livre E Esclarecido

Universidade Federal de Santa Catarina
Departamento de Odontologia

De: Prof. Dr. Ricardo Armini Caldas
Professor do Departamento de Odontologia da Universidade Federal de Santa Catarina

SOLICITAÇÃO DE DISPENSA DO TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO – TCLE

Eu, Ricardo Armini Caldas, solicito a dispensa da aplicação do Termo de Consentimento Livre e Esclarecido, do projeto de Pesquisa intitulado "Utilização de *deep learning* na classificação das disfunções temporomandibulares: prova de conceito". Declaro que o acesso aos prontuários clínicos dos pacientes do Centro Multidisciplinar de Dor Orofacial (CEMDOR) pertencentes ao acervo, será realizado somente para fins de pesquisa científica, e será feito somente após aprovação do Projeto de Pesquisa pelo Comitê de Ética.

A dispensa do TCLE se pauta nas seguintes justificativas:

1. Por ser um estudo observacional, que empregará apenas informações de prontuários odontológicos, e informações clínicas disponíveis na instituição sem previsão de utilização de material biológico;
2. Porque todos os dados serão manejados e analisados de forma anônima, sem identificação nominal dos participantes de pesquisa;
3. Porque os resultados decorrentes do estudo serão apresentados de forma agregada, não permitindo a identificação individual dos participantes;
4. Porque se trata de um estudo não intervencionista (sem intervenções clínicas) e sem alterações/influências na rotina/tratamento do participante de pesquisa, e consequentemente sem adição de riscos ou prejuízos ao bem-estar dos mesmos;
5. Após a busca pelo atendimento no para resolução da dor, os pacientes retornaram para os centros e/ou consultórios em que eram atendidos anteriormente, não frequentando mais o CEMDOR;
6. De acordo com o controlador dos dados, por conta da pandemia de Sars-coV-2, muitos cadastros estão desatualizados e/ou sem contato de e-mail, o que dificulta o acesso aos pacientes.

O investigador principal e demais colaboradores envolvidos no estudo acima se comprometem, individual e coletivamente, a utilizar os dados provenientes deste, apenas para os fins descritos e a cumprir todas as diretrizes e normas regulamentadoras descritas na Res. CNS Nº 466/12, e suas complementares, no que diz respeito ao sigilo e confidencialidade dos dados coletados.

Atenciosamente,

Prof. Dr. Ricardo Armini Caldas
Email: ricardo.caldas@ufsc.br

Florianópolis, 26 de agosto de 2022.