MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21d/2024/01.26.12.36-TDI

AVALIAÇÃO DO USO DO AUTOML PARA A CLASSIFICAÇÃO DE ÁREAS QUEIMADAS USANDO SÉRIES TEMPORAIS DO SATÉLITE LANDSAT-8

Marcelly Homem Coelho

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Rafael Duarte Coelho dos Santos, e Olga Regina Fradico de Oliveira Bittencourt, aprovada em 13 de dezembro de 2023.

URL do documento original:
<http://urlib.net/8JMKD3MGP3W34T/4AKLQD2>

INPE
São José dos Campos
2023

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE
Coordenação de Ensino, Pesquisa e Extensão (COEPE)
Divisão de Biblioteca (DIBIB)
CEP 12.227-010
São José dos Campos - SP - Brasil
Tel.:(012) 3208-6923/7348
E-mail: pubtc@inpe.br

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21d/2024/01.26.12.36-TDI

AVALIAÇÃO DO USO DO AUTOML PARA A CLASSIFICAÇÃO DE ÁREAS QUEIMADAS USANDO SÉRIES TEMPORAIS DO SATÉLITE LANDSAT-8

Marcelly Homem Coelho

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Rafael Duarte Coelho dos Santos, e Olga Regina Fradico de Oliveira Bittencourt, aprovada em 13 de dezembro de 2023.

URL do documento original:
<http://urlib.net/8JMKD3MGP3W34T/4AKLQD2>

INPE
São José dos Campos
2023

Dados Internacionais de Catalogação na Publicação (CIP)

Coelho, Marcelly Homem.

Co65a Avaliação do uso do AutoML para a classificação de áreas queimadas usando séries temporais do satélite Landsat-8 / Marcelly Homem Coelho. – São José dos Campos : INPE, 2023.

xxiv + 103 p. ; (sid.inpe.br/mtc-m21d/2024/01.26.12.36-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2023.

Orientadores : Drs. Rafael Duarte Coelho dos Santos, e Olga Regina Fradico de Oliveira Bittencourt.

1. Área queimada. 2. Aprendizado de máquina. 3. Classificação. 4. Série temporal. I.Título.

CDU 004.032.2:528.8

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

DEFESA FINAL DE DISSERTAÇÃO MARCELLY HOMEM COELHO
BANCA Nº 305/2023, REG. 674995/2021

No dia 13 de dezembro de 2023, as 09h, por teleconferência, o(a) aluno(a) mencionado(a) acima defendeu seu trabalho final (apresentação oral seguida de arguição) perante uma Banca Examinadora, cujos membros estão listados abaixo. O(A) aluno(a) foi APROVADO(A) pela Banca Examinadora, por unanimidade, em cumprimento ao requisito exigido para obtenção do Título de Mestre em Computação Aplicada, com a exigência de que o trabalho final a ser publicado deverá incorporar as correções sugeridas pela Banca Examinadora, com revisão pelo(s) orientador(es).

Novo Título: "AVALIAÇÃO DO USO DO AUTOML PARA A CLASSIFICAÇÃO DE ÁREAS QUEIMADAS USANDO SÉRIES TEMPORAIS DO SATÉLITE LANDSAT-8."

Membros da Banca:

Dr. Gilberto Ribeiro de Queiroz – Presidente – INPE
Dr. Rafael Duarte Coelho dos Santos – Orientador – INPE
Dra. Olga Regina Fradico de Oliveira Bittencourt – Orientadora – INPE
Dra. Ana Carolina Lorena – Membro Externo – ITA
Dr. Anderson Luiz Fernandes Perez – Membro Externo – UFSC

Documento assinado eletronicamente por **Ana Carolina Lorena (E)**, **Usuário Externo**, em 14/12/2023, às 11:28 (horário oficial de Brasília), com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Olga Regina Fradico de Oliveira Bittencourt (E)**, **Usuário Externo**, em 14/12/2023, às 11:33 (horário oficial de Brasília), com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Rafael Duarte Coelho dos Santos**, **Pesquisadora**, em 14/12/2023, às 15:01 (horário oficial de Brasília), com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Anderson Luiz Fernandes Perez (E)**, **Usuário Externo**, em 15/12/2023, às 10:07 (horário oficial de Brasília), com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Gilberto Ribeiro de Queiroz**, **Tecnologista**, em 18/12/2023, às 08:50 (horário oficial de Brasília), com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.mcti.gov.br/verifica.html, informando o código verificador **11594743** e o código CRC **28E669F2**.

Referência: Processo nº 01340.010423/2023-44 SEI nº 11594743

"O conhecimento é uma aventura em aberto. O que significa que aquilo que saberemos amanhã é algo que desconhecemos hoje e, esse algo pode mudar as verdades de ontem".

Karl Popper

Dedico este trabalho a todos aqueles que acreditaram em mim e, junto comigo, tornaram-no realidade.

AGRADECIMENTOS

Agradeço ao Professor Doutor Rafael Duarte Coelho dos Santos, meu orientador, pela orientação, paciência e apoio durante todo o período do mestrado. À Doutora Olga Oliveira Bittencourt, minha coorientadora, pela orientação e profunda compreensão do tópico de pesquisa. Suas contribuições foram fundamentais para a realização deste trabalho.

À banca avaliadora, formada pelos Doutores Ana Carolina Lorena, Anderson Luiz Fernandes Perez e Gilberto Ribeiro. Agradeço pela disponibilidade em aceitar o convite para participar da avaliação da minha dissertação.

Aos colegas de mestrado e amigos, agradeço pelos ensinamentos, companheirismo e ambiente agradável compartilhado.

Ao Instituto Nacional de Pesquisas Espaciais e ao Programa Queimadas, por proporcionarem os recursos necessários para a realização deste trabalho. Este estudo foi possível graças ao apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), por meio dos processos 88882.461660 e 88887.603912.

À minha família que sempre me incentivou a aprender. À minha irmã Nathália, por me apoiar na construção da dissertação. Ao meu namorado, Rian Koja, por me ajudar a refletir, instigando o meu raciocínio e me ajudando a adquirir novos conhecimentos, estando ao meu lado em todos os momentos.

Grata a todas as pessoas que me ajudaram ao longo do percurso.

RESUMO

As queimadas representam um desafio global e afetam grandes extensões de vegetação nativa, causando impactos negativos no âmbito social, econômico e ecológico. A classificação de áreas queimadas em imagens de satélite é de interesse para automatizar o mapeamento de regiões que sofreram queimas e, dessa forma, ajudar a otimizar a alocação de recursos destinados a essa problemática. Esta dissertação descreve o desenvolvimento de um método baseado em aprendizado de máquina para a classificação automática de áreas queimadas por meio de análises de séries temporais do satélite Landsat-8. A pergunta de pesquisa que pretende-se responder consiste em "*É possível determinar áreas queimadas por meio de séries temporais de índices espectrais referentes a pontos geográficos?*". Dentro do escopo desse trabalho, demonstra-se que um modelo de classificação supervisionada, treinado com amostras de queimadas de um ano específico, é capaz de ser generalizado para classificar ocorrências de queimadas em períodos anuais subsequentes. Para avaliação do método proposto, foram conduzidos seis experimentos distintos, com o experimento final usando os conjuntos de dados correspondentes aos anos de 2018 e 2019 para o treinamento do modelo, enquanto o conjunto de dados de 2020 foi empregado para fins de teste. Foram analisadas as métricas de desempenho: taxa de acerto média, precisão, revocação e *F1-score*. Os resultados obtidos por meio do modelo *Support Vector Machine* (SVM) treinado com o algoritmo de otimização *Stochastic Gradient Descent* (SGD) revelaram uma taxa de acerto média na classificação de áreas queimadas e não queimadas de 95,55% com desvio padrão de 1,78%. Esta dissertação contribui para o avanço das técnicas de identificação de queimadas, oferecendo uma abordagem eficaz e precisa que se mostra promissora para a gestão de recursos e a mitigação de impactos ambientais.

Palavras-chave: Área queimada. Aprendizado de máquina. Classificação. Série temporal.

ASSESSMENT OF AUTOML USAGE ON BURNED AREA CLASSIFICATION USING LANDSAT-8 TIME SERIES

ABSTRACT

Wildfires pose a worldwide challenge, impacting vast stretches of native vegetation and giving rise to adverse effects on social, economic, and ecological dimensions. The classification of burned areas in satellite imagery is of interest for automating the mapping of affected regions, ultimately aiding the allocation of resources addressed to this issue. This dissertation describes the development of a machine learning-based approach for the automated classification of burned areas employing some time series of Landsat-8 images. The proposed research question is: "*Is it possible to identify burned areas from spectral indexes time series referred to geographical locations?*". Within the scope of this work, it is shown that a supervised classification model, trained with samples from a specific year is capable of generalizing into subsequent years of data. Six separate experiments were conducted to evaluate the proposed method, with the final one using datasets corresponding to the years of 2018 and 2019 for training the model, while datasets from 2020 were used for testing. The following performance metrics were assessed: accuracy, precision, recall, and F1-score. The results obtained with a Support Vector Machine (SVM) model trained with the Stochastic Gradient Descent (SGD) optimization algorithm reveled an accuracy of 95,55% with a standard deviation of 1,78%. This dissertation contibutes to the advancement of burned area identification methods, presenting an effective and accurate approach with promissing applications in resource management and environmental impact mitigation.

Keywords: Burnt area. Machine learning. Classification. Time series.

LISTA DE FIGURAS

Pág.

LISTA DE TABELAS

Pág.

LISTA DE ABREVIATURAS E SIGLAS

ACM	–	*Association for Computing Machinery*
AM	–	Aprendizado de Máquina
AUC	–	*Area Under the Curve*
AutoML	–	*Automated Machine Learning*
BAI	–	*Burn Area Index*
BAIML	–	*Burned Area Index Modified with Longer*
BAIMS	–	*Burned Area Index Modified with Short*
BDC	–	*Brazil Data Cube*
BDG	–	Banco de Dados Geográficos
CAPES	–	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CE	–	Critério de Exclusão
CI	–	Critério de Inclusão
CSI	–	*Char Soil Index*
DOI	–	*Digital Object Identifier*
DNN	–	*Deep Neural Network*
EVI	–	*Enhanced Vegetation Index*
FN	–	Falso Negativo
FP	–	Falso Positivo
GEMI	–	*Global Environmental Monitoring Index*
IA	–	Inteligência Artificial
IBGE	–	Instituto Brasileiro de Geografia e Estatística
IEEE	–	*Institute of Electrical and Electronics Engineers*
INPE	–	Instituto Nacional de Pesquisas Espaciais
KNN	–	*K-Nearest Neighbors*
MAE	–	*Mean Absolute Error*
MIRBI	–	*Mid-Infrared Burn index*
MSE	–	*Mean Squared Error*
NBR	–	*Normalized Burn Ratio*
NBR2	–	*Variation of Normalized Burn Ratio*
NBRT1	–	*Normalized Burn Ratio Thermal*
NDMI	–	*Normalized Difference Moisture Index*
NDVI	–	*Normalized Difference Vegetation Index*
NDWI	–	*Normalized Difference Wetness Index*
NIR	–	*Near Infrared Reflectance*
OLI	–	*Operational Land Imager*
PCA	–	*Principal Component Analysis*
PIB	–	Produto Interno Bruto
QA	–	Questão de Análise
RFE	–	*Recursive Feature Elimination*
RMSE	–	*Root Mean Squared Error*
RBR	–	*The Relativized Burn Ratio*
RF	–	*Random Forest*

RSL – Revisão Sistemática da Literatura
RVI – *Radar Vegetation Index*
SAVI – *Soil Adjusted Vegetation Index*
SGD – *Stochastic Gradient Descent*
SVM – *Support Vector Machine*
SWIR – *Short Wave Infrared*
TPOT – *Tree-Based Pipeline Optimization Tool*
VARI – *Visual Atmospheric Resistance Index*
VI6T – *Vegetation Index 6 Thermal*
VN – Verdadeiro Negativo
VP – Verdadeiro Positivo
WTSS – *Web Time Series Service*

LISTA DE SÍMBOLOS

$\bar{x}$ – Média aritmética
s – Desvio padrão amostral
Q_1 – Primeiro quartil
Q_2 – Segundo quartil
Q_3 – Terceiro quartil
τ – Taxa de acerto média

SUMÁRIO

Pág.

1 INTRODUÇÃO

As queimadas nos biomas brasileiros representam um tema de crescente preocupação e relevância, demandando uma atenção especial por parte das autoridades e da sociedade em geral. De acordo com dados estatísticos divulgados pelo Programa Queimadas do Instituto Nacional de Pesquisas Espaciais (INPE), no período de janeiro a dezembro de 2022, verificou-se um total de 200.763 focos de fogo ativos detectados nos diferentes biomas brasileiros. A Amazônia registrou o maior percentual, representando 57,3% do total de focos ativos. Em seguida, o Cerrado apresentou uma proporção de 28,3%, a Caatinga 7,8%, a Mata Atlântica 5,4%, o Pantanal 0,8% e o Pampa 0,4%. No mesmo período, dentre os países que compõem a América do Sul, o Brasil registrou a maior quantidade de focos de fogo ativo, representando 55,3% do total detectado (INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE), 2023).

Os incêndios florestais, juntamente com as alterações climáticas e a seca, são considerados um dos principais distúrbios, que provocam, entre outros impactos, a destruição da vegetação no Cerrado (SILVA et al., 2022). Estudos apontam que os efeitos das queimadas nos ecossistemas são complexos, abrangendo desde a perda de biodiversidade da fauna e flora e a redução da fertilidade do solo a impactos ecológicos, pois influenciam na poluição atmosférica e nas mudanças climáticas. Além disso, representam uma fonte adicional de emissões de gases de efeito estufa, podendo resultar em diversas implicações, tais como: paralisação de atividades aeroportuárias, aumento de morbimortalidade por doenças respiratórias, danos ao patrimônio público e privado, entre outras (SANTIAGO; LOPES, 2021)(SANTOS et al., 2017).

Os avanços tecnológicos na área espacial viabilizaram o desenvolvimento de sensores remotos embarcados em satélites, dotados da capacidade de capturar imagens abrangentes da Terra, apresentando uma alta frequência temporal (JENSEN, 2009)(RODRIGUES et al., 2018). Conforme discutido por Zaglia (), as imagens de satélite representam digitalmente a superfície terrestre. Os sensores remotos desses satélites têm a habilidade de medir a energia refletida por diferentes alvos na superfície terrestre. Após a recepção do sinal pelo satélite, inicia-se o processo em que o sinal é transformado em imagens, organizadas espacialmente em cenas. Cada cena refere-se a uma divisão do espaço determinada previamente, geralmente identificada por par órbita-ponto.

Esta realidade favoreceu o aprimoramento de técnicas de sensoriamento remoto, visto que, as imagens constituem importante fonte de informação que pode ser usada para a prática de monitoramento. As técnicas de sensoriamento têm sido usadas de

várias maneiras para auxiliar no mapeamento de áreas queimadas Pereira et al. (2021), na previsão do tamanho final do fogo Coffield et al. (2019), nas estimativas de propagação do fogo Jamal et al. (2021) e na avaliação da gravidade dos danos causados por incêndios Farasin et al. (2020).

A detecção e o monitoramento de incêndios florestais são fundamentais para viabilizar o controle do fogo e a redução dos custos relacionados às operações de combate às queimadas. De acordo com Wang et al. (2021), enquanto a complexidade dos incêndios florestais desafia a modelagem, as técnicas de aprendizado de máquina surgiram como novas ferramentas para avançar na compreensão e análise dos incêndios florestais, explorando suas variáveis preditoras. Conforme Bittencourt et al. (2019) e Bittencourt et al. (2020), o grande desafio da aplicação dessas técnicas é a necessidade de um conjunto de dados de conhecimento para modelar adequadamente o problema e gerar classificações mais precisas. Os autores destacam que muitas regiões do Brasil possuem uma escassez de conjuntos de dados validados por especialistas.

Os sistemas automáticos, baseados em algoritmos de aprendizagem de máquina, são usados para auxiliar especialistas em domínios no acompanhamento da evolução dos incêndios florestais durante o evento, bem como na identificação automática de áreas queimadas após a extinção do fogo (FARASIN et al., 2020). No entanto, um desafio enfrentado é a definição de modelos e configurações de parâmetros. Bittencourt et al. (2019) e Coelho et al. (2022) ressaltam a ampla variedade de modelos de aprendizado de máquina que podem ser aplicados para a classificação de áreas queimadas, o que torna a seleção manual de um algoritmo uma tarefa complexa. Novas tecnologias estão surgindo para resolver esse problema. De acordo com Olson e Moore (2016), as ferramentas de geração de Aprendizado de Máquina Automatizado (do Inglês, *Automated Machine Learning*) (AutoML) podem ser empregadas para selecionar modelos e suas respectivas configurações de hiperparâmetros, visando obter o melhor desempenho nos conjuntos de dados de entrada fornecidos ao algoritmo de treinamento.

Esta dissertação visa desenvolver um método, baseado em aprendizado de máquina, para a classificação automática de áreas queimadas por meio de análises de séries temporais do satélite Landsat-8. A pergunta de pesquisa que norteia este trabalho: "*É possível determinar áreas queimadas por meio de séries temporais de índices espectrais referentes a pontos geográficos?*". Mais especificamente, busca-se determinar se um modelo de classificação supervisionada, treinado com amostras de queimadas

de um ano específico, é capaz de classificar ocorrências de queimadas em períodos anuais subsequentes. Para avaliar o método proposto, foram analisadas as métricas de desempenho: Taxa de acerto média, Precisão, Revocação e *F1-score*.

1.1 Objetivos

Esta seção apresenta o objetivo geral e os objetivos específicos do presente trabalho.

1.1.1 Objetivo geral

Este trabalho tem como principal objetivo desenvolver um método voltado à classificação de áreas queimadas por meio da análise de séries temporais do satélite Landsat-8.

1.1.2 Objetivos específicos

Os objetivos específicos deste trabalho encontram-se descritos abaixo:

- Realizar uma revisão sistemática na literatura sobre a classificação automática de áreas queimadas.
- Realizar a coleta, preparação e limpeza dos conjuntos de dados de queimadas.
- Realizar o estudo e identificação das variáveis pertinentes ao domínio do problema.
- Empregar uma ferramenta de aprendizado de máquina automatizado que permita a seleção e avaliação do modelo proposto.
- Avaliar o desempenho de diferentes experimentos, orientados a dados, para promover a classificação de áreas queimadas de períodos anuais distintos.

1.2 Hipótese de pesquisa

Acredita-se que algoritmos de aprendizado de máquina, usando séries temporais de bandas espectrais e índices espectrais, derivadas de históricos de dados de queimadas rotulados, são capazes de estimar a classificação automática de novas áreas.

1.3 Organização do trabalho

Além desta introdução, esta proposta de dissertação está estruturada em outros cinco capítulos e dois apêndices que abordam os seguintes conteúdos:

- Capítulo 2: Este capítulo discutirá estudos relacionados a classificação de áreas queimadas por meio do desenvolvimento de uma Revisão Sistemática da Literatura.
- Capítulo 3: Este capítulo apresentará um levantamento bibliográfico de aprendizado de máquina.
- Capítulo 4: Este capítulo descreverá o método proposto para a classificação automática de áreas queimadas.
- Capítulo 5: Este capítulo apresentará os resultados da aplicação do método proposto.
- Capítulo 6: Este capítulo apresentará as conclusões obtidas com base na discussão dos resultados, assim como contribuições geradas pela pesquisa.
- Apêndice A: Este apêndice lista as publicações científicas usadas na Revisão Sistemática da Literatura.
- Apêndice B: Este apêndice apresentará os algoritmos resultantes das execuções do aprendizado de máquina automatizado.
- Apêndice C: Este apêndice ilustra uma visão geral dos modelos de aprendizado de máquina e as métricas de desempenho para cada experimento.

2 ESTADO DA ARTE

Este capítulo apresenta uma Revisão Sistemática da Literatura com o propósito de levantar o estado da arte em classificação de áreas queimadas por meio da aplicação de aprendizado de máquina. As seções iniciais abrangem as etapas de planejamento, condução e resultados das análises da revisão. Ao final, são destacadas as considerações finais do capítulo.

2.1 Revisão Sistemática da Literatura

Uma Revisão Sistemática da Literatura (RSL) consiste em uma modalidade de pesquisa científica que é caracterizada por seguir um protocolo específico no seu desenvolvimento. Trata-se de um tipo de investigação que contém questões bem definidas, que objetiva identificar, selecionar, avaliar e sintetizar as evidências relevantes sobre o tema de pesquisa (KITCHENHAM; CHARTERS, 2007).

Os principais objetivos da realização de uma revisão são identificar avanços do conhecimento, reunir evidências relacionadas a um método ou tecnologia e identificar lacunas em um determinado tema de pesquisa científica. De acordo com Klompenburg et al. (2020), uma RSL permite aos pesquisadores vislumbrar novas perspectivas que favorecem o entendimento do estado da arte sobre uma determinada área.

A metodologia usada nesta RSL foi baseada na proposta apresentada por Kitchenham e Charters (2007). Os autores destacam que, inicialmente, é definido um protocolo de revisão. Este protocolo descreve um conjunto de diretrizes que devem ser usadas na elaboração da investigação. O protocolo seguido neste trabalho pode ser dividido em três partes: planejamento da revisão, condução da revisão e resultados das análises da revisão.

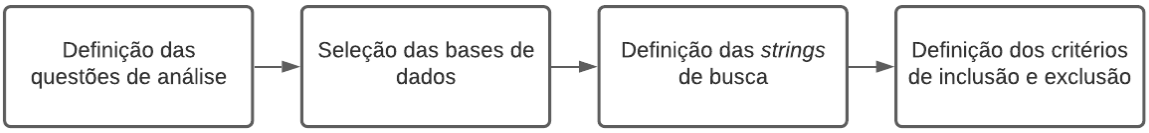
2.1.1 Planejamento da revisão

De acordo com Kitchenham e Charters (2007), na etapa de planejamento da revisão, é necessário definir um protocolo de pesquisa. Esta definição visa possibilitar que a RSL seja replicada por outros pesquisadores.

Para essa revisão, a primeira etapa do protocolo de pesquisa consiste na definição das Questões de Análise (QAs). Na sequência, é realizada a seleção das bases de dados digitais para a busca de artigos científicos. Na próxima etapa, são definidas as *strings* de busca e construída a expressão lógica. Por fim, são aplicados filtros para seleção de artigos relevantes. Estes filtros são chamados de critérios de inclusão

e exclusão. A Figura 2.1 apresenta um mapeamento esquemático com as etapas do processo de planejamento da revisão.

Figura 2.1 - Mapeamento das etapas do processo do planejamento da revisão.



Fonte: Elaborada pela autora.

Esta RSL tem como objetivo obter informações sobre estudos publicados no domínio de aprendizado de máquina para a classificação de áreas queimadas. Com base nesse objetivo, foram definidas três Questões de Análise (QAs).

- QA1: "Quais características têm sido usadas na literatura para a classificação de áreas queimadas por meio de aprendizado de máquina?"
- QA2: "Quais algoritmos de aprendizado de máquina têm sido usados na literatura para a classificação de áreas queimadas?"
- QA3: "Quais parâmetros de avaliação têm sido usados na literatura para avaliar um método de classificação de áreas queimadas?"

Para este estudo, as bases de dados selecionadas foram Scopus[1] e *Web of Science*[2]. A motivação para a escolha da Scopus foi por englobar outras bases em sua busca como, por exemplo, *Elsevier*, *Springer* e IEEE, sendo estas de grande relevância para a área de Ciência da Computação. Enquanto que a *Web of Science* foi por contemplar uma ampla quantidade de periódicos e conferências indexadas. As duas bases de dados selecionadas possuem recursos *web* para suporte a buscas a partir de expressões lógicas, a possibilidade de pesquisar por metadados de publicações científicas e a disponibilidade de um mecanismo de filtragem temporal.

[1]Scopus: é um banco dados abrangente de resumos e citações organizado por especialistas com dados enriquecidos e associados à literatura acadêmica em uma grande variedade de disciplinas.

[2] *Web of Science*: Consiste em uma base de dados bibliográfica multidisciplinar, especialmente reconhecida por sua ênfase na indexação de revistas científicas de alta qualidade e impacto.

Para realizar as buscas automáticas nas bases de dados digitais, o próximo passo foi formular as *strings* de buscas. Estas foram compostas pelas palavras-chave *"Burned"*, *"Classification"* e *"Machine learning"*, todas pesquisadas em inglês. As *strings* foram vinculadas por meio do operador lógico AND, conforme descrito abaixo.

"$Burned$" AND "$Classification$" AND "$Machine\ Learning$"

A expressão lógica mencionada acima conduziu à identificação de 77 artigos na base de dados Scopus e 83 na *Web of Science*. Algumas publicações científicas foram encontradas em ambas as plataformas. Os artigos foram mantidos ou descartados conforme Critérios de Inclusão (CIs) e Critérios de Exclusão (CEs). Os CIs representam as características que os estudos recuperados devem possuir para serem considerados neste trabalho, enquanto que os CEs são particularidades que causam a desconsideração desses artigos.

Para esta RSL, foram definidos sete CEs. Se um artigo recuperado das bases de dados digitais atender a pelo menos um desses critérios, este é desconsiderado do conjunto de publicações. Os CEs são listados a seguir.

- CE1: Excluir artigos que não sejam estudos primários.
- CE2: Excluir artigos que não sejam do tipo completo.
- CE3: Excluir artigos que não possuem resumo.
- CE4: Excluir artigos que não estejam escritos nos idiomas inglês ou português.
- CE5: Excluir artigos que não sejam acessíveis na *web* ou no Portal de Periódicos da CAPES.
- CE6: Excluir artigos que não foram publicados entre os anos de 2013 a 2022.
- CE7: Excluir artigos que não estejam relacionados a classificação de áreas queimadas.

Na sequência, após a seleção por meio dos CEs, os artigos recuperados foram analisados de acordo com os CIs. Ao total, foram definidos três CIs. Se pelo menos um desses critérios fosse atendido, então este artigo seria estudado de forma mais aprofundada. Os tópicos a seguir explicam cada um dos CIs.

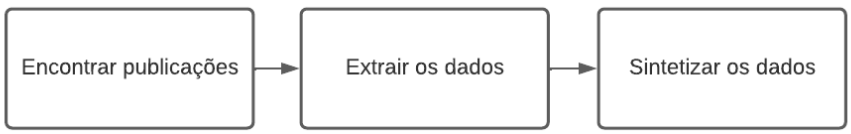
- CI1: O artigo apresenta uma técnica de aprendizado de máquina, consolidada na literatura, para a classificação de áreas queimadas.
- CI2: O artigo apresenta um estudo comparativo de técnicas para a classificação de áreas queimadas.
- CI3: O artigo apresenta uma combinação entre uma nova abordagem e técnicas de aprendizado de máquina, consolidadas na literatura, para a classificação de áreas queimadas.

Após a aplicação dos CEs e dos CIs, um total de 36 artigos científicos permaneceram para a análise das Questões de Análise. As informações de identificador (ID), ano de publicação, título, base de dados e DOI dos 36 artigos são apresentadas na Tabela A.1 do Apêndice A.

2.1.2 Condução da revisão

A segunda fase da RSL consiste na condução da revisão. O mapeamento das etapas do processo de condução da revisão é mostrado na Figura 2.2. Inicialmente, os artigos científicos são encontrados de acordo com as definições do protocolo de pesquisa. Em um segundo momento, os dados são extraídos e, por fim, são sintetizados para a geração de *insights*. Estas informações visam fornecer uma visão geral dos artigos relevantes publicados até o momento e responder às questões de análise definidas no início da revisão.

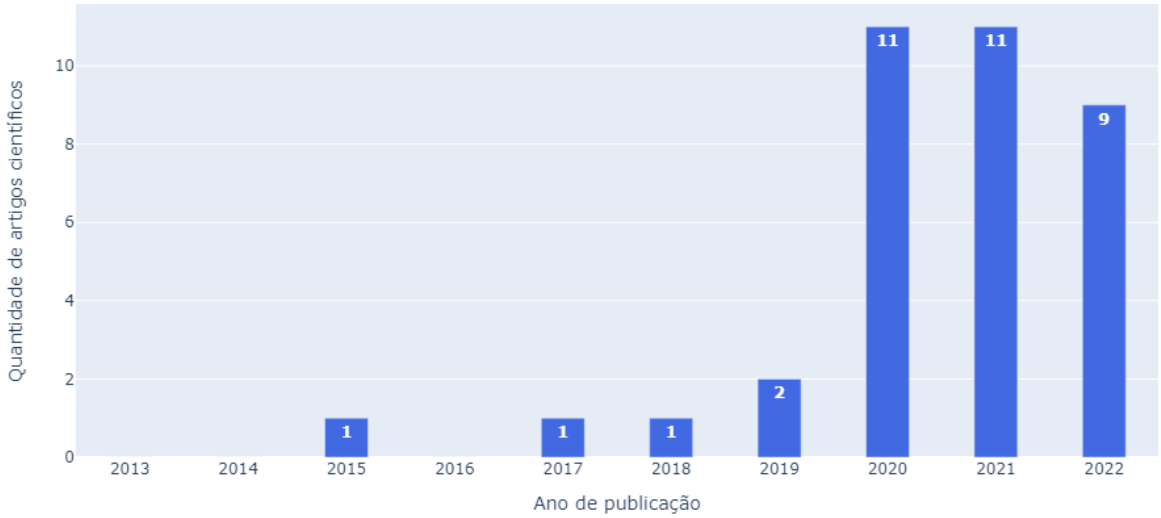
Figura 2.2 - Mapeamento das etapas do processo de condução da revisão.



Fonte: Elaborada pela autora.

A Figura 2.3 apresenta a quantidade de artigos científicos publicados anualmente, entre os anos de 2013 e 2022, que tratam do uso de aprendizado de máquina para classificação de áreas queimadas.

Figura 2.3 - Distribuição dos artigos ao longo dos anos de publicação.



Fonte: Elaborada pela autora.

Conforme a Figura 2.3, pode-se observar que, ao longo do período de dez anos, houve um crescimento significativo do interesse acadêmico nesse campo de pesquisa, especialmente a partir de 2020. Até esse ano, o número de publicações se manteve relativamente baixo, com apenas algumas ocorrências esporádicas. Entretanto, a partir de 2020, houve um aumento substancial no número de estudos, que continuou significativo nos anos subsequentes.

Esse padrão de crescimento pode indicar um provável aumento no reconhecimento da importância do uso de aprendizado de máquina para a classificação de áreas queimadas. Esse tema pode estar se tornando mais relevante no contexto de monitoramento ambiental e prevenção de incêndios florestais, já que o aprendizado de máquina pode desempenhar um papel crucial na análise e interpretação de dados geoespaciais, possibilitando uma detecção mais rápida e precisa das áreas afetadas por queimadas. O aumento no número de publicações também sugere que a comunidade científica está direcionando seus esforços para o desenvolvimento de novas abordagens, algoritmos e técnicas que melhorem a precisão e a eficácia desses sistemas de classificação.

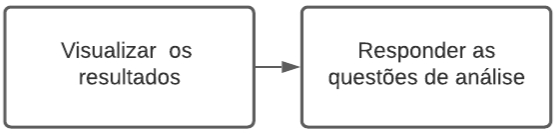
É importante notar que, embora o crescimento no número de publicações seja promissor, ainda há a necessidade de investigações adicionais e avanços na área. O

aprendizado de máquina para classificação de áreas queimadas é um campo complexo que envolve desafios técnicos e científicos, como a obtenção de dados precisos, a seleção adequada de algoritmos, o treinamento de modelos com grande volume de informações e a validação de resultados. Portanto, o aumento contínuo na quantidade de pesquisas reflete o interesse crescente, mas também indica que há uma demanda por novos estudos que possam impulsionar o avanço dessa área e contribuir para a mitigação de problemas ambientais relacionados às queimadas.

2.1.3 Resultados das análises da revisão

A terceira fase da RSL consiste em analisar os resultados obtidos nas etapas anteriores. A Figura 2.4 mostra o fluxo do processo de análise de artigos científicos. Inicialmente, a proposta é visualizar os resultados e, em um segundo momento, responder às questões de análise.

Figura 2.4 - Mapeamento das etapas do processo de análise de artigos.



Fonte: Elaborada pela autora.

2.1.3.1 QA1: "Quais características têm sido usadas na literatura para a classificação de áreas queimadas por meio de aprendizado de máquina?"

Para responder à primeira questão de análise (QA1), foram investigadas as características usadas para treinamento dos algoritmos de aprendizado de máquina. A Tabela 2.1 apresenta quais foram os tipos de variáveis identificadas nos artigos científicos, as características representadas dentro dos tipos de variáveis e os identificadores únicos dos artigos.

Tabela 2.1 - Análise das características usadas para treinamento dos algoritmos de aprendizado de máquina dos artigos científicos selecionados.

Tipo de variável	**Característica**	**IDs dos artigos**
Índices espectrais	NDVI	[3, 7, 9, 11, 13, 15, 16, 22, 23, 26, 31, 32, 35, 36]
	NBR	[3, 5, 8, 9, 11, 13, 15, 21, 26, 28, 30, 32]
	NBR2	[8, 9, 13, 21, 26, 27, 29]
	BAI	[3, 9, 13, 26, 29, 32]
	MIRBI	[3, 13, 26, 29, 32]
	EVI	[13, 15, 16, 26]
	GEMI	[3, 13, 26, 32]
	NDWI	[13, 16, 26, 32]
	NDMI	[13, 26, 32]
	CSI	[13, 26]
	SAVI	[13, 26]
	RBR	[11]
	NBRT1	[13]
	VI6T	[13]
	VARI	[16]
	RVI	[30]
	BAIMS	[32]
	BAIML	[32]
Bandas espectrais	NIR	[5, 9, 10, 11, 12, 21, 25, 27, 29, 30, 33]
	SWIR1	[5, 9, 11, 12, 21, 25, 27, 29, 30, 33]
	SWIR2	[9, 12, 21, 25, 27]
	Red	[9, 10, 12, 25, 30]
	Green	[9, 10, 12, 25]
	Blue	[9, 12, 25]
Antropogênicas	Distância da malha viária	[16, 24, 36]
	Assentamento humano	[16, 24]
	Distância da área residencial	[24, 36]
	Distância de rios	[16]

(Continua)

Tabela 2.1 - Análise das características usadas para treinamento dos algoritmos de aprendizado de máquina dos artigos científicos selecionados.

Tipo de variável	**Característica**	**IDs dos artigos**
Combustão	Área queimada	[1, 14, 20, 22, 33, 34]
	Intensidade do fogo	[20, 34]
	Tempo de queima	[1]
	Altura de combustão	[15]
	Radiação do fogo	[17]
	Propagação do fogo	[20]
	Velocidade da chama	[20]
	Focos de fogo ativo	[33]
	Duração da temporada de incêndio	[34]
	Frequência do fogo	[34]
Climáticas	Temperatura do ar	[1, 2, 4, 7, 14, 15, 16, 18, 20, 22, 24, 31, 35, 36]
	Precipitação	[1, 2, 4, 7, 15, 16, 18, 20, 23, 24, 26, 31, 35, 36]
	Velocidade do vento	[1, 4, 14, 15, 16, 20, 22, 23, 35, 36]
	Umidade relativa do ar	[1, 4, 7, 14, 18, 20, 22, 23, 36]
	Umidade do solo	[2, 5, 6, 14, 18, 31]
	Temperatura da superfície terrestre	[22, 23, 24]
	Direção do vento	[15, 22]
	Pressão do vapor	[4]
	Temperatura da superfície do mar	[6]
	Radiação solar	[16]
	Evapotranspiração	[35]
	Déficit hídrico	[35]
	Pressão barométrica	[36]
	Horas de sol	[36]
Paisagem	Inclinação	[4, 15, 16, 19, 22, 23, 35, 36]
	Tipo de vegetação	[1, 2, 4, 11, 14, 15, 18]
	Aspecto	[15, 16, 24, 31, 35, 36]
	Elevação	[4, 16, 23, 24, 31, 35]
	Cobertura da terra	[7, 19, 31, 33]
	Declive	[15, 16, 24]

(Continua)

Tabela 2.1 - Análise das características usadas para treinamento dos algoritmos de aprendizado de máquina dos artigos científicos selecionados.

Tipo de variável	Característica	IDs dos artigos
	Tipo de solo	[17, 18]
	Altitude	[19, 36]
	Sombra	[15]
	Inclinação	[31]
Socioeconômicas	Densidade populacional	[2, 14, 15, 23, 24, 31, 36]
	Produto Interno Bruto (PIB)	[23, 36]
	Interface urbana de Wildland	[15]
Geoespaciais	Latitude	[17, 23, 35, 36]
	Longitude	[17, 23, 35, 36]

A Tabela 2.1 mostra uma visão abrangente das características empregadas no processo de treinamento de algoritmos de aprendizado de máquina em estudos científicos voltados para a classificação de áreas queimadas. Ao observar a tabela, fica evidente a interdisciplinaridade e a complexidade subjacente à análise, que abrange uma ampla variedade de fatores para realizar análises de áreas queimadas.

O tipo de variável relacionado aos índices espectrais foi o mais citado, com o total de 21 menções, seguido pelos tipos de variáveis: climáticas (18), paisagem (18), bandas espectrais (11), combustão (8), socioeconômicas (7), geoespaciais (4) e antropogênicas (3). Dentre os índices espectrais apresentados na tabela, quatro se destacam como os mais frequentes, são eles: NDVI, NBR, NBR2 e BAI.

Segundo Escuin et al. (2008), o Índice de Vegetação de Diferença Normalizada (do Inglês, *Normalized Difference Vegetation Index*) (NDVI) é um indicador da biomassa fotossinteticamente ativa que refere-se a uma medida que quantifica a quantidade de matéria viva que está envolvida na fotossíntese em um ecossistema. Este índice desempenha um papel crucial na distinção entre a vegetação e outros tipos de cobertura de superfície, bem como na identificação e classificação de áreas cultivadas em mapas temáticos. Além disso, desempenha um papel importante na detecção de mudanças nos padrões de cobertura da terra. Notavelmente, devido à capacidade desse índice de refletir diminuições nos valores nas áreas afetadas por queimadas, tornou-se um dos indicadores mais usados para a identificação desses eventos. O NDVI é calculado por meio da diferença entre a reflectância das bandas infraverme-

lho próximo (NIR) e do vermelho (RED), dividida pela soma das duas reflectâncias, conforme a Equação 2.1.

$$NDVI = \frac{NIR - RED}{NIR + RED} \tag{2.1}$$

O Índice de Área Queimada (do Inglês, *Burn Area Index*) (BAI) destaca as áreas que sofreram queimas por meio do espectro do vermelho (RED) ao infravermelho próximo (NIR), enfatizando alvos carbonizados em imagens pós-incêndio (VERA-VERBEKE et al., 2011). O BAI é calculado por meio da Equação 2.2.

$$BAI = \frac{1}{(0,1 + RED)^2 + (0,06 + NIR)^2} \tag{2.2}$$

O Índice de Queimada Normalizada (do Inglês, *Normalized Burn Ratio*) (NBR) é usado na detecção de cicatrizes de queimadas e avaliação da severidade do fogo, uma vez que, nas composições do índice, são usadas as regiões do espectro eletromagnético que sofrem alterações após a queima. Em áreas com vegetação, o índice assume valores positivos, enquanto que, em áreas de solo descoberto, seus valores são negativos (ESCUIN et al., 2008). O NBR é obtido por meio da Equação 2.3.

$$NBR = \frac{NIR - SWIR}{NIR + SWIR} \tag{2.3}$$

Conforme Lestari et al. (2020), os autores destacam que o índice espectral NBR é um dos parâmetros mais significativos para a identificação de áreas queimadas. Em Pereira et al. (2021), os autores observaram que o uso das bandas espectrais SWIR1 e SWIR2 mostram uma maior separabilidade entre as classes queimada e não queimada, nas regiões do Cerrado. Diante dessa observações, tais índices espectrais foram usados para calcular as bandas espectrais do NBR1 e NBR2.

Nos trabalhos de Bittencourt et al. (2020), Escuin et al. (2008) e Veraverbeke et al. (2011), foi testada com sucesso a capacidade da combinação de bandas e índices espectrais para caracterizar áreas queimadas. É importante destacar que o conhecimento do domínio deve ser usado para orientar a geração de atributos, visto que, cada atributo adiciona complexidade computacional durante o treinamento de um modelo. A adição de recursos redundantes e altamente correlacionados deve ser evitada.

Em Pereira et al. (2015), os autores destacam que os índices espectrais são sensíveis a mudanças nas características espectrais nos diferentes tipos de biomas, visto que são provenientes de expressões matemáticas envolvendo valores de reflectância. No estudo, é realizada uma investigação sobre os índices espectrais que possuem uma maior capacidade de diferenciação de queimadas de demais alvos no bioma Cerrado. A partir dos resultados obtidos, observou-se uma maior separabilidade nos índices espectrais MIRBI e NBR2. Por esse motivo, tais índices foram considerados os mais indicados para o mapeamento de queimadas.

As variáveis climáticas usadas com mais frequência nos artigos científicos foram: temperatura do ar, precipitação, velocidade do vento, umidade relativa do ar e umidade do solo. De acordo com Vanderhoof et al. (2021), o uso de dados de precipitação pode potencialmente apoiar os esforços para identificar áreas queimadas. Entretanto, os autores destacam que os conjuntos de dados de precipitação tipicamente possuem resolução grosseira. Assim, permanece o desafio de separabilidade entre as classes queimada e não queimada, especialmente em áreas úmidas.

De acordo com Sánchez et al. (2021), as ecorregiões podem ser definidas como áreas relativamente grandes de terra ou água, formadas por conjuntos distintos de espécies que compartilham condições ambientais semelhantes. A biodiversidade da flora, fauna e ecossistemas difere de uma ecorregião para outra. Os autores destacam que esse fator é importante para se considerar em estudos que pretendem avaliar incêndios florestais. As informações sobre as características vegetais de cada ecorregião ajudaram a tratar a heterogeneidade espacial para a determinação de classes. Assim, para uma boa interpretação dos resultados, é importante levar em conta os diferentes tipos de ecorregiões e as classes de cobertura.

2.1.3.2 QA2: "Quais algoritmos de aprendizado de máquina têm sido usados na literatura para a classificação de áreas queimadas?"

Para responder à segunda questão de análise (QA2), foram investigados os algoritmos de aprendizado de máquina aplicados nos artigos científicos. A Tabela 2.2 apresenta todos os tipos de algoritmos encontrados em pelo menos um dos artigos. Destaca-se que algumas das publicações usaram mais de um modelo de aprendizado de máquina e realizaram comparação entre eles em suas análises.

Com base nos resultados apresentados na Tabela 2.2, pode-se observar que uma grande parte dos artigos científicos aplicou o modelo de aprendizado de máquina *Random Forest* (RF).

Tabela 2.2 - Análise dos algoritmos de aprendizado de máquina implementados nos artigos científicos selecionados.

Algoritmo	**IDs dos artigos**
Random Forest	[2, 4, 5, 6, 7, 8, 9, 11, 14, 15, 16, 18, 19, 20, 21, 22, 24, 26, 30, 31, 32, 36]
Support Vector Machine	[3, 6, 8, 17, 20, 22, 24, 27, 29, 31, 32, 33, 35, 36]
Multilayer Perceptron	[4, 6, 10, 20, 22, 29, 33]
Logistic Regression	[3, 14, 22, 28, 29]
Decision Trees	[3, 4, 9, 18, 35]
Light Gradient Boosting	[4, 6, 13, 21, 33]
K-Nearest Neighbors	[4, 21, 35]
Deep Learning	[8, 12, 21]
Extreme Gradient Boosting	[23, 31]
Convolution Neural Network	[20, , 33]
K-means	[25, 34]
Artificial Neural Network	[35, 36]
Ada Boost	[18, 33]
Genetic Algorithm	[1]
Quantile Regression Forests	[14]
Maximum Entropy	[24]
Extreme Learning Machine	[29]
SelfOrganizing Map	[33]
Bayesian Linear Regression	[35]
Polynomial Linear Regression	[35]

De acordo com Sánchez et al. (2021), o RF é um dos modelos mais eficazes para a avaliação de suscetibilidade de incêndios florestais. O RF destaca-se, entre os demais modelos, pelos seguintes fatores: facilidade de calibração dos parâmetros, capacidade de produzir um *ranking* de importância das variáveis usadas no treinamento e o fato de os dados não precisarem ser redimensionados ou transformados.

O segundo algoritmo de aprendizado de máquina mais usado foi o *Support Vector Machine* (SVM). Em Lestari et al. (2020), é realizado um estudo comparativo entre os modelos RF, SVM e *Deep Neural Network* (DNN). Os autores destacam que esses modelos são promissores para classificar áreas queimadas. Avaliando-se as métricas de desempenho precisão, revocação e acurácia, foi observado que o modelo RF obteve melhores resultados comparado aos modelos SVM e DNN, tanto para um conjunto de dados equilibrado quanto para um conjunto de dados desequilibrado.

Duas observações importantes destacadas no artigo é que conjuntos de dados desequilibrados afetam diretamente o desempenho na classificação e que a quantidade de árvores, no modelo RF, não influenciou na precisão.

Além de sua aplicação na classificação de áreas queimadas, os modelos de aprendizado de máquina também têm sido usados na análise preditiva de propagação do fogo (JAMAL et al., 2021), determinação do total de área queimada (WANG et al., 2021), estimativa da gravidade da queimada (HUANG et al., 2020) e previsibilidade do tamanho do fogo (COFFIELD et al., 2019). Em Coffield et al. (2019), os autores visam obter uma previsibilidade do tamanho final do fogo no momento da ignição. Para isso, foram testados os modelos *Decision Trees*, RF e MLP. Neste estudo, o modelo que se sobressaiu foi o *Decision Trees*. Os autores ressaltam que o uso inadequado de modelos RF e MLP pode causar excessos de adaptações, enquanto que o modelo *Decision Trees* é um método mais simples, facilmente interpretável e que pode alcançar uma precisão superior em relação aos outros.

2.1.3.3 QA3: "Quais parâmetros de avaliação têm sido usados na literatura para avaliar um método de classificação de áreas queimadas?"

A avaliação de modelos de aprendizado de máquina envolve a escolha de métricas apropriadas que dependem do tipo de problema que está sendo abordado. Existem duas categorias comuns de problemas em aprendizado de máquina, são elas: classificação e regressão. As métricas usadas para avaliar essas duas categorias são diferentes devido à natureza das saídas previstas.

Nos problemas de classificação, quer-se prever a classe qualitativa a que uma amostra de dados pertence com base nas variáveis de entrada. Particularmente, os problemas de classificação possuem saída discreta, como, por exemplo, classificar áreas queimadas e não queimadas. As métricas de desempenho mais comuns são: a precisão, que mede a proporção de exemplos classificados corretamente em relação ao total de exemplos; a revocação, que mede a proporção de exemplos positivos corretamente identificados em relação a todos os exemplos positivos reais; o *F1-score*, que combina precisão e revocação, útil quando o desequilíbrio entre as classes é significativo.

Já nos problemas de regressão, o objetivo é prever um valor numérico com base nas variáveis de entrada. Uma das principais características de algoritmos de regressão consiste na previsão de valores contínuos, podendo, por exemplo, ser usados para prever a extensão das áreas queimadas. As métricas de desempenho mais comuns

são: o MAE, que mede a média das diferenças absolutas entre as previsões e os valores reais; o MSE, que é a média das diferenças quadradas entre as previsões e os valores reais; a RMSE, que fornece uma medida do erro em uma escala semelhante aos dados originais.

Para responder à terceira questão de análise (QA3), foram identificadas as métricas de desempenho para avaliação dos modelos de aprendizado de máquina aplicados nos artigos científicos. A Tabela 2.3 fornece uma visão geral das métricas e dos identificadores dos artigos que as utilizaram como parte de sua avaliação.

Tabela 2.3 - Análise das métricas de desempenho usadas para avaliação dos algoritmos de aprendizado de máquina nos artigos científicos selecionados.

Métrica de desempenho	**IDs dos artigos**
Precisão	[3, 4, 8, 9, 10, 11, 12, 14, 15, 18, 19, 21, 22, 26, 28, 29, 30, 31, 32, 33, 35, 36]
Sensibilidade (Revocação)	[4, 8, 9, 12, 14, 17, 18, 22, 24, 30, 31, 35, 36]
F1-*score*	[9, 12, 14, 18, 22, 28, 30, 33, 36]
Erro de comissão	[3, 5, 13, 21, 25, 26, 27, 29]
Erro de omissão	[3, 5, 13, 21, 25, 26, 27, 29]
AUC	[14, 16, 19, 24, 26, 31, 35, 36]
RMSE	[8, 11, 14, 15, 20, 23]
Especificidade	[18, 22, 24, 31, 35]
Acurácia	[4, 8, 9, 22, 32, 36]
Coeficiente de Kappa	[10, 21, 28, 31, 33]
Coeficiente de dados	[3, 5, 25, 26]
MAE	[1, 14, 20]
Viés relativo	[13, 25, 26]
Índice de concordância	[2, 23]
Coeficiente de Pearson	[7, 35]
Variância fracionária	[2]
Coeficiente de determinação	[14]
Coeficiente de Sorensen	[17]
Taxa de falso alarme	[17]
MSE	[20]
G-*mean*	[22]
Coeficiente de correlação	[23]
Índice de sucesso crítico	[27]
Tempo de treinamento	[29]
Índice Sorensen-Dice	[29]
Índice de Dunn	[34]
ROC	[35]

Um conjunto considerável de artigos, um total de 21 estudos, empregou a métrica de precisão em suas avaliações. Isso sugere que a precisão é uma métrica amplamente adotada para avaliar o desempenho de algoritmos de aprendizado de máquina em diversas pesquisas relacionadas a classificação de áreas queimadas. Além disso, a sensibilidade, também conhecida como revocação, apareceu em 13 dos artigos selecionados. A métrica *F1-score*, que combina precisão e revocação, foi encontrada em 9 artigos.

De acordo com Buczak e Guven (2015), a escolha das métricas a serem empregadas em uma aplicação específica está intrinsecamente relacionada ao contexto do problema a ser avaliado. No cenário abordado neste estudo, que se enquadra na categoria de classificação, a análise das métricas apresentadas na Tabela 2.1 revela que as medidas mais adotadas para a avaliação de modelos de classificação incluem precisão, revocação e o *F1-score*. Assim, em concordância com essa constatação, optou-se por utilizar essas métricas para avaliar o desempenho dos experimentos realizados no presente trabalho. Adicionalmente, complementando a avaliação, também se considerou a taxa de acerto média, que é equivalente à acurácia.

Os artigos científicos selecionados nesta RSL, que serviram de base para responder às questões de análise QA1, QA2 e QA3, foram listados na Tabela A.1, disponível no Apêndice A. A tabela contém o identificador único do artigo científico, criado exclusivamente para identificação do trabalho em questão, ano de publicação, título do artigo, base de dados que foi recuperado e o *Digital Object Identifier* (DOI).

2.2 Considerações finais do capítulo

Neste capítulo foi apresentada uma RSL com o objetivo de obter o estado da arte relacionado a pesquisas que propõem a aplicação de aprendizado de máquina para a classificação automática de áreas queimadas. Pode-se destacar que, nas últimas décadas, vários algoritmos de aprendizado de máquina têm sido aplicados na identificação de áreas queimadas. Esta revisão mostrou que os artigos científicos selecionados usaram uma variedade de características, algoritmos e métricas de desempenho. Todos os artigos investigam a classificação de áreas queimadas por meio de algoritmos de aprendizado do máquina, mas diferem-se quanto as características e métricas de desempenho usadas na pesquisa.

De acordo com Mithal et al. (2018) e Sánchez et al. (2021), a determinação de quais características devem ser usadas para treinamento de um modelo depende principalmente do escopo da pesquisa e da disponibilidade dos conjuntos de dados. Diversos

trabalhos discutem a quantidade de características usadas para treinamento de um modelo de classificação de áreas queimadas. A observação geral é que nem sempre a adição de características está associada ao melhor desempenho na classificação.

Os índices espectrais desempenham um papel crucial na extração e análise de dados provenientes de sensoriamento remoto (NEGRI et al., 2022). Para representar uma característica de interesse, um índice espectral pode ser usado para auxiliar na sua identificação. De um modo geral, os índices espectrais são derivados de operações algébricas nos atributos que caracterizam o comportamento de cada pixel. Além disso, eles são indicados para o treinamento de modelos de aprendizado de máquina devido à sua sensibilidade a variações de cor, composição, umidade do solo e níveis de clorofila na vegetação.

O mapeamento de áreas queimadas por meio de índices espectrais concentra-se nas bandas mais sensíveis ao fogo, com o propósito de reduzir o ruído no sinal. Após um incêndio, é possível observar uma diminuição na clorofila, o que resulta no aumento da refletância no espectro visível, enquanto o dano ao tecido foliar pode estar relacionado à redução da refletância no infravermelho próximo (NIR). Os índices espectrais desenvolvidos para mapear áreas queimadas buscam aproveitar essas diferenças para obter uma alta capacidade de distinguir entre áreas queimadas e não queimadas (CHANDEL et al., 2022).

Em Santos et al. (2023), foi desenvolvida uma metodologia para detectar grandes incêndios florestais, usando um algoritmo de aprendizado de máquina em conjunto com séries temporais de imagens Landsat. Os autores enfatizaram dois pontos cruciais que contribuíram para o bom desempenho da metodologia proposta. Primeiramente, eles abordaram a análise de máscaras de nuvens, reconhecendo que a presença de nuvens pode causar confusão entre áreas queimadas e outras dinâmicas de cobertura do solo. Para superar esse desafio, foi usada a banda de avaliação de qualidade de cada imagem da série, a fim de mitigar os problemas relacionados à contaminação de pixel por nuvens e sombras de nuvens. Em segundo lugar, a utilização de índices e bandas espectrais específicas foi fundamental para ampliar a caracterização do espaço espectral. A etapa de análise e seleção de atributos permitiu a escolha das melhores bandas para o classificador, resultando em um espaço de atributos mais compacto e com melhor desempenho durante a classificação.

Vários algoritmos de aprendizado de máquina têm sido usados em pesquisas científicas relacionadas a classificação de áreas queimadas. Alguns dos artigos selecionados nesta RSL aplicam somente um modelo em suas análises (WANG et al., 2021) (BAR-

RETO; ARMENTERAS, 2020), enquanto que outros sugerem a comparação de diferentes modelos (YU et al., 2020) (COUGHLAN et al., 2021). De acordo com os resultados da síntese, para a QA2, o modelo mais usado foi o *Random Forest*, com o total de 22 menções. Embora esse modelo tenha sido o mais aplicado, em (BITTENCOURT et al., 2019) e (BITTENCOURT et al., 2020), os autores destacam a dificuldade em determinar quais são os modelos de aprendizado de máquina e configurações de parâmetros considerados mais adequados para promover a classificação de áreas queimadas.

Em Chandel et al. (2022), os autores destacam que modelos de aprendizado de máquina, como *Random Forest*, *Support Vector Machine* e Regressão Logística, têm sido amplamente aplicados na classificação de áreas queimadas devido a sua capacidade de aprender as características espectrais e reconhecer esses padrões. Além disso, eles possuem a vantagem de não assumir uma distribuição normal dos dados, sendo adequados para automatização de processos. Outro fator mencionado é que esses modelos são menos sensíveis as variações de parâmetros quando comparados as Redes Neurais Artificiais e que estas, por sua vez, geralmente demandam mais tempo de treinamento.

De acordo com os resultados da síntese, para a QA3, as métricas de desempenho mais usadas foram a precisão e a revocação. Como abordagem de avaliação, uma grande parte dos artigos científicos aplicou a validação cruzada. A validação cruzada de 10 vezes foi a preferida pelos pesquisadores (WANG et al., 2021) (BARRETO; ARMENTERAS, 2020) (WANG; WANG, 2020).

Esta RSL revelou que os artigos científicos selecionados usaram diversos tipos de variáveis, dependendo do escopo da pesquisa e da disponibilidade de conjuntos de dados. Todos os artigos aplicam aprendizado de máquina para a classificação de áreas queimadas, mas diferem-se nas variáveis usadas para treinamento de modelos. Estudos também indicaram que a adição de características nem sempre fornecem um melhor desempenho para a classificação. Outra observação é que não há uma conclusão sobre qual é o modelo de aprendizado de máquina ideal para a classificação de áreas queimadas, mas a grande maioria dos artigos científicos aplicou o modelo *Random Forest*.

O presente trabalho apresenta um diferencial em relação aos artigos científicos selecionados na RSL, no campo de classificação automática de áreas queimadas, por meio do uso de aprendizado de máquina e análise de séries temporais. Em Santos et al. (2023), os autores concentraram-se na detecção de grandes incêndios florestais, com ênfase na análise de máscaras de nuvens e na seleção de bandas espectrais específi-

cas. A dissertação em questão tem como foco um método semelhante, mas voltado para a automação da classificação de dados de queimadas de anos subsequentes.

O estudo proposto visa determinar se um modelo de classificação supervisionada, treinado com amostras de incêndios de um ano específico, por exemplo, 2018, é capaz de automatizar a classificação de ocorrências de queimadas em anos subsequentes, por exemplo 2019 e 2020. Essa abordagem representa um avanço importante, pois permite a aplicação contínua e eficaz do modelo para identificar queimadas em diferentes períodos temporais, economizando tempo e recursos que seriam gastos na validação manual a cada ano.

Adicionalmente, a escolha de aplicar AutoML para determinar qual modelo de aprendizado de máquina é mais adequado para os conjuntos de dados específicos é uma abordagem inovadora e prática. A utilização do AutoML elimina a necessidade de seleção manual entre uma variedade de algoritmos de aprendizado de máquina, o que resulta em economia de esforço e aprimoramento da eficiência do processo de modelagem.

Por fim, pode-se destacar que esta pesquisa representa um avanço significativo na aplicação de aprendizado de máquina para a classificação de áreas queimadas, combinando a automação da classificação em anos subsequentes com a escolha automatizada do modelo mais apropriado. Essa abordagem contribui não apenas para a eficácia da classificação de áreas queimadas, mas também demonstra a aplicação prática e escalável da tecnologia em um contexto de monitoramento ambiental de suma importância.

3 FUNDAMENTAÇÃO TEÓRICA

Este capítulo fornece uma visão geral de aprendizado de máquina. Em especial, são destacadas as características do aprendizado supervisionado. A justificativa dessa ênfase é o fato de que o método desenvolvido nesta dissertação faz uso de conjuntos de dados rotulados, beneficiando-se da rica base de dados proveniente de projetos desenvolvidos pelo INPE ao longo dos anos. As últimas seções do capítulo apresentam uma ferramenta de geração de aprendizado automático e métricas de avaliação de modelos de classificação.

3.1 Fundamentos de aprendizado de máquina

De acordo com Trappenberg (2020), o aprendizado de máquina consiste na construção de máquinas, geralmente em software, que têm a capacidade de aprender a executar tarefas específicas. Essa abordagem envolve a modelagem de dados e a descrição de incertezas por meio do uso de técnicas que contribuem para o desenvolvimento de tecnologias relacionadas à Inteligência Artificial (IA). Em vez de programar explicitamente uma máquina para realizar uma tarefa específica, o objetivo é desenvolver programas de aprendizado que possuem a capacidade de generalização. Essa abordagem é particularmente benéfica quando a tarefa em questão é difícil de ser codificada em um sistema baseado em regras explícitas.

Nos sistemas computacionais, a experiência é representada a partir dos dados disponíveis. A principal finalidade do aprendizado de máquina é desenvolver algoritmos que, a partir desses conjuntos de dados, sejam capazes de construir modelos. Ao alimentar esses algoritmos com dados de experiência, conseguimos obter modelos capazes de fazer previsões em novas observações. De forma geral, seja $D = \left\{x_1, x_2, ..., x_m\right\}$ um conjunto de dados com m instâncias, onde cada instância $x_i = (x_{i1}; x_{i2}; ...; x_{id}) \in \chi$ é um vetor d-dimensional de amostras no espaço χ, d é chamado de dimensionalidade da instância x_i, e x_{ij} é o valor do j-ésimo atributo da instância x_i.

Durante o processo de treinamento de um modelo de aprendizado de máquina, pode-se ter acesso às informações de resultados correspondentes. Essas informações são comumente chamadas de rótulos. Cada amostra de treinamento pode ser descrita como um par ordenado (x_i, y_i), onde y_i pertence ao conjunto de rótulos de saída $\mathcal{Y}$. Os rótulos de saída podem assumir natureza discreta ou contínua. Os problemas de classificação são caracterizados por rótulos de saída discretos, enquanto que, problemas de regressão envolvem rótulos de saída contínuos (ZHOU, 2021).

O campo do aprendizado de máquina tem se disseminado amplamente na pesquisa científica, sendo incorporado em diversas aplicações, tais como mineração de texto, análise de dados biomédicos de câncer de próstata (OLSON et al., 2016), reconhecimento de comportamentos anormais de veículos em rodovias (ANDRADE et al., 2022), previsão de infecção por COVID-19 (MUHAMAD et al., 2020), prognóstico de degradação de equipamentos industriais, entre outras. Conforme (ALZUBAIDI et al., 2021), o contínuo surgimento de novos estudos na área de aprendizado de máquina deve-se tanto ao crescimento na capacidade de obtenção de conjuntos de dados quanto ao progresso alcançado nas tecnologias de hardware.

No estudo realizado por Alzubaidi et al. (2021), são destacados alguns tópicos que discorrem sobre a recomendação de aplicação do aprendizado de máquina. Os autores enfatizam que a utilização da IA em diversos cenários demonstra-se promissora e que pode ser considerada uma solução viável para as seguintes problemáticas:

- Nos casos em que os especialistas humanos não estão disponíveis.
- Nos casos em que os humanos são incapazes de explicar as decisões tomadas usando seus conhecimentos.
- Nos casos em que a solução do problema é atualizada com o tempo.
- Nos casos em que a solução do problema requer adaptação com base em especificações.
- Nos casos em que o tamanho do problema é extremamente grande e excede as habilidades do raciocínio humano.

Segundo Muhamad et al. (2020), o processo de aprendizado de máquina inicia-se com a coleta de dados, abrangendo uma variedade de fontes de origem. Em seguida, realiza-se a etapa de pré-processamento dos dados, visando corrigir problemas relacionados aos mesmos, bem como reduzir o tamanho do espaço de dados. Na próxima etapa, os algoritmos de aprendizado de máquina são desenvolvidos com base em conceitos, como estatística, teoria de controle e probabilidade, a fim de analisar os dados, extrair conhecimentos úteis, descobrir padrões ocultos ou informações de experiências anteriores. O próximo passo é a avaliação de desempenho dos modelos e, finalmente, a otimização do modelo.

A indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos. Ela é caracterizada como o raciocínio

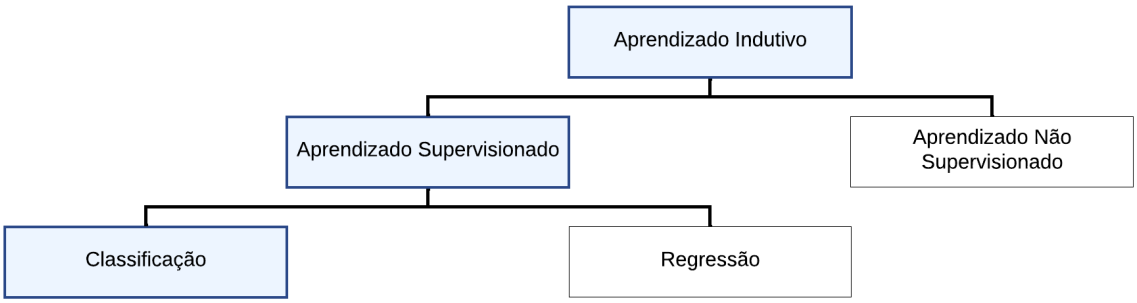
que se origina em um conceito específico e o generaliza, ou seja, da parte para o todo. Na indução um conceito é aprendido efetuando-se inferência indutiva sobre os exemplos apresentados. Portanto, as hipóteses geradas por meio da inferência indutiva podem ou não preservar a verdade. O aprendizado indutivo é efetuado a partir de raciocínio sobre exemplos fornecidos por um processo externo ao sistema de aprendizado.

Conforme Monard et al. (2003), o aprendizado indutivo pode ser dividido em duas categorias principais: aprendizado supervisionado e não supervisionado. No aprendizado supervisionado, o conjunto de dados é rotulado. Nesse tipo de aprendizado, são fornecidos ao indutor as variáveis de entrada e o rótulo de saída. O objetivo do algoritmo de indução é construir um preditor que possa determinar corretamente a classe de novos exemplos ainda não rotulados. Para rótulos de classe discretos, o problema é conhecido como classificação, enquanto que para valores contínuos, o problema é conhecido como regressão.

Em contrapartida, no aprendizado não supervisionado não há categorização ou rotulação do conjunto de dados. Nesse tipo de aprendizado, o indutor analisa os exemplos fornecidos e tenta determinar quais destes podem ser agrupados de alguma maneira, formando um *clusters*. Após a determinação dos agrupamentos, normalmente, é necessária uma análise para determinar o que cada agrupamento significa no contexto do problema que está sendo analisado.

Na Figura 3.1 é ilustrada uma representação da hierarquia de aprendizado de máquina. Os itens sombreados, na cor azul claro, correspondem ao tipo de aprendizado usado nesta dissertação.

Figura 3.1 - Hierarquia do aprendizado de máquina.



Fonte: Adaptada de Monard et al. (2003)

3.1.1 Aprendizado não supervisionado

No aprendizado não supervisionado, o conjunto de dados de treinamento não é rotulado, classificado ou categorizado previamente. Isso significa que os efeitos das variáveis de entrada são desconhecidos (MUHAMAD et al., 2020). O aprendizado busca descobrir padrões entre as variáveis de entrada para que os dados semelhantes possam ser agrupados. Esta abordagem é usada quando o problema não tem os resultados desejados. As técnicas de aprendizado não supervisionadas incluem: análise de agrupamentos, detecção de anomalias, geração de *insights* e visualização de dados.

3.1.2 Aprendizado supervisionado

O aprendizado supervisionado é uma estratégia que se destaca por sua capacidade de coletar dados ou gerar saídas com base no conhecimento prévio disponível. Essa abordagem é especialmente útil quando existem informações sobre as classes ou rótulos das amostras de treinamento.

De acordo com Alzubaidi et al. (2021), a desvantagem do aprendizado supervisionado surge quando o conjunto de treinamento não abrange todas as possibilidades ou variações presentes nos conjuntos de dados. Isso significa que, se houver uma classe específica que não esteja bem representada nas amostras de treinamento, o modelo resultante pode enfrentar dificuldades em tomar decisões corretas para essa classe. Esse problema é conhecido como sobrecarga do limite de decisão.

Em termos gerais, o limite de decisão é a fronteira que o modelo estabelece para separar diferentes classes ou categorias. Se um conjunto de treinamento não incluir amostras que deveriam pertencer a uma determinada classe, o modelo não é capaz de generalizar e reconhecer corretamente as instâncias dessa classe durante a fase de teste ou predição. Portanto, em aplicações de aprendizado supervisionado, é fundamental se ter um conjunto de treinamento representativo e abrangente, que inclua exemplos de todas as classes que se deseja reconhecer e classificar. Caso contrário, o desempenho do modelo pode ser comprometido.

3.2 Aprendizado de máquina automatizado

Com o rápido crescimento das aplicações de ciência de dados, o aprendizado de máquina passou por uma revolução, à medida que instituições de ensino e pesquisa, empresas privadas e governamentais descobriram novas formas de utilizar algoritmos automatizados capazes de aprender com os conjuntos de dados. Essa tendência tem gerado uma crescente demanda por ferramentas que tornem o aprendizado de

máquina mais acessível, escalável e flexível (OLSON et al., 2016).

O Aprendizado de Máquina Automatizado (do Inglês, *Automated Machine Learning*) (AutoML) tem como objetivo automatizar as tarefas essenciais para a construção de modelos de aprendizado de máquina. Os métodos baseados em AutoML automatizam uma ou mais etapas do processo de geração de um modelo como, por exemplo, preparação de dados, engenharia de *features*, otimização de hiperparâmetros, seleção de algoritmos de aprendizado de máquina (OLSON; MOORE, 2016) (HE et al., 2021).

Devido à ampla diversidade de modelos de aprendizado de máquina e configurações de hiperparâmetros, o desenvolvimento, avaliação e ajuste dos sistemas de aprendizado de máquina se tornam tarefas complexas. No entanto, as tecnologias baseadas em AutoML têm como objetivo simplificar e automatizar esses processos, possibilitando que os usuários desenvolvam modelos de forma ágil e personalizada para suas aplicações.

Existem vários sistemas para geração de aprendizado de máquina automatizado. Em He et al. (2021), os autores destacam que muitas empresas de IA têm criado e compartilhado publicamente esses sistemas para a sociedade. Neste trabalho foi usada a ferramenta AutoML *Tree-Based Pipeline Optimization Tool*[1] (TPOT). A justificativa desta escolha é que o TPOT é de código aberto, está bem documentado e em desenvolvimento ativo no período de escolha de ferramentas. De acordo com Olson e Moore (2016), o *framework* tem a capacidade de avaliar *pipelines* de forma automática e eficiente. Em seu estudo, Feurer et al. (2015) destaca que uma das características mais importante do TPOT é a capacidade de exportar um modelo preditivo e inseri-lo diretamente em um código-fonte.

O TPOT é baseado na linguagem de programação Python. O *framework* otimiza *pipelines* de aprendizado de máquina para produzir modelos de regressão ou de classificação. O TPOT explora vários canais possíveis para encontrar o melhor dentre eles para uma base de dados específica. Para isso, são considerados vários algoritmos de aprendizado de máquina, bem como diversas configurações de hiperparâmetros (OLSON; MOORE, 2016).

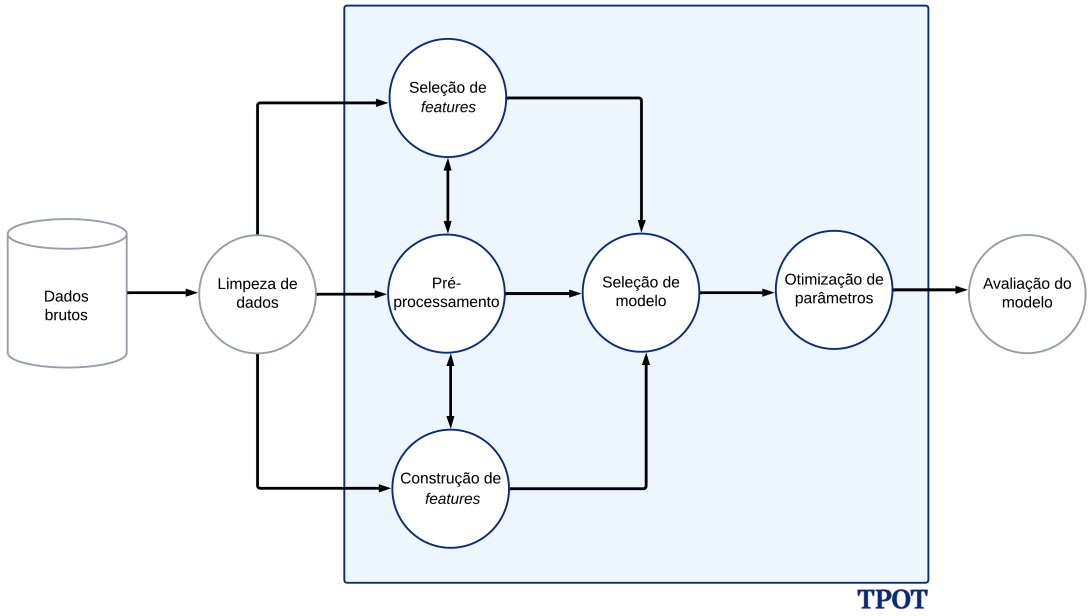
Para construir e otimizar *pipelines*, o TPOT usa programação genética. A ideia do algoritmo evolutivo é criar, inicialmente, uma população de *pipelines* de aprendizado de máquina de forma aleatória e evoluí-la com mutações e cruzamentos de geração em geração. Os *pipelines* são avaliados e recebem uma pontuação de condicionamento

[1] http://epistasislab.github.io/tpot/using/

físico, para que o procedimento de seleção possa determinar quais indivíduos estarão na próxima geração. O critério de parada pode ser quanto ao tempo máximo de execução ou ao tamanho da população.

O TPOT exerce a função de ser um assistente que indique ideias sobre como resolver um determinado problema de aprendizado de máquina, explorando configurações de *pipelines* e realizando ajustes de hiperparâmetros. A Figura 3.2 representa o processo de aprendizado de máquina realizado pelo TPOT.

Figura 3.2 - Automatização de etapas do processo de aprendizado de máquina por TPOT.



Fonte: Adaptada de Olson e Moore (2016).

Inicialmente, é necessário realizar a preparação e limpeza dos dados brutos. A construção, o processamento e a seleção de características podem facilitar o entendimento dos dados de entrada, por meio da adição de regras implícitas. Como resultado deste processo iterativo, tem-se a seleção de um modelo de AM. Na sequência, é realizada a otimização de parâmetros que consiste em ajustar diferentes restrições, pesos ou taxas de aprendizagem para encontrar operadores que minimizem os erros de predição do modelo selecionado. Por fim, a avaliação do modelo resultante é de responsabilidade do desenvolvedor.

O TPOT disponibiliza vários modelos de aprendizado de máquina como, por exem-

plo: Regressão Logística, *Gaussian Naive Bayes*, *Decision Tree*, *Extra Trees*, *Gradient Boosting*, *Bernoulli Naive Bayes*, *Multinomial Naive Bayes*, *Random Forest*, *K-Nearest Neighbors*, *Support Vector Classifier*, *XGBoost*, entre outros. Nas próximas subseções, são apresentados os embasamentos teóricos dos algoritmos *K-Nearest Neighbors*, *Decision Tree* e *Stochastic Gradient Descent*. A justificativa é que esses modelos foram os selecionados, pelo AutoML, para promover a classificação automática de áreas queimadas nos experimentos realizados nesta dissertação.

3.2.1 Modelo *K-Nearest Neighbors*

De acordo com Cover e Hart (1967), o modelo *K-Nearest Neighbors* (KNN) é um algoritmo de aprendizado de máquina supervisionado que usa dados previamente memorizados para classificar novos pontos de dados na classe-alvo, dependendo dos pontos disponíveis mais próximos. Os autores Hu et al. (2021) definem o KNN como sendo uma técnica de classificação baseada na proximidade entre pontos de dados no espaço de características. A premissa é que os pontos de dados com características semelhantes tendem a pertencer à mesma classe.

Suponha um conjunto de dados que contenha exemplos de treinamento. Cada exemplo de treinamento consiste em um vetor de características e uma classe associada. O vetor de características representa os atributos relevantes e a classe indica a categoria a qual o exemplo pertence. A primeira etapa de um algoritmo KNN consiste em receber um dado não classificado.

O próximo passo é escolher o valor de k, que represente o número de vizinhos mais próximos que serão considerados para classificar um novo exemplo. A escolha do valor de k é crítica, pois pode afetar o desempenho do algoritmo (SHARMA et al., 2022). Valores pequenos de k podem tornar o modelo sensível a ruído, podendo ocorrer o problema de *overfitting*, enquanto que valores muito grandes de k podem suavizar a decisão e reduzir a capacidade do modelo de se ajustar a padrões complexos, podendo ocorrer o problema de *underfitting*.

Para classificar um novo exemplo, o KNN baseia-se na distância entre a amostra de teste e as amostras de treinamento especificadas. A medida de distância pode ser calculada, por exemplo, por meio da distância Euclidiana e distância de Manhattan. Nas Equações 3.1 e 3.2, x_i e y_i, representam os componentes dos vetores x e y nas D dimensões do espaço de características. A distância Euclidiana é calculada como a raiz quadrada da soma dos quadrados das diferenças entre os componentes correspondentes dos vetores de características, conforme a Equação 3.1.

$$d(x, y) = \sqrt{\sum_{i=1}^{D}(x_i - y_i)^2} \tag{3.1}$$

Enquanto a distância de Manhattan é definida pela soma das diferenças absolutas entre as coordenadas dos pontos ao longo de todas as dimensões, conforme a Equação 3.2.

$$d(x, y) = \sum_{i=1}^{D} |x_i - y_i| \tag{3.2}$$

Após calcular as distâncias entre a amostra de teste e as amostras de treinamento, bem como identificar os k vizinhos mais próximos, a próxima etapa envolve a atribuição da classe prevista para o exemplo de teste. Para realizar essa classificação, aplica-se um critério de votação majoritária. Isso significa que a classe prevista para o exemplo de teste é determinada pela classe que é mais frequente entre os k vizinhos mais próximos. Essa abordagem de votação majoritária é fundamental no KNN, pois é usada para tomar decisões de classificação com base na vizinhança dos exemplos de treinamento. É um processo simples, mas eficaz, que explora a proximidade dos dados no espaço de características para realizar a classificação.

3.2.2 Modelo *Decision Tree*

O modelo *Decision Tree* é uma técnica no campo de aprendizado de máquina conhecida por sua abordagem conceitual simples e eficaz na solução de problemas de classificação, tornando-se amplamente adotada em diversas aplicações. Em Safavian e Landgrebe (1991), os autores destacam que as árvores de decisão possuem grande potencial no reconhecimento de padrões para a classificação de dados oriundos, por exemplo, de áreas como o sensoriamento remoto.

A árvore de decisão é uma representação que desdobra decisões e regras complexas em uma estrutura hierárquica, semelhante a um sistema de ramificações. No modelo *Decision Tree*, vários pontos de decisão, chamados de "nós", são criados. Em cada nó, a decisão segue por um caminho específico, chamado de "ramo", determinado pela comparação de um valor de atributo com um parâmetro obtido no treinamento do modelo. Essa abordagem visa simplificar uma decisão complexa, dividindo-a em decisões mais simples, com a expectativa de que a solução final se assemelhe à solução desejada (SAFAVIAN; LANDGREBE, 1991).

A árvore de decisão é construída a partir de um conjunto de dados de treinamento que consiste em exemplos rotulados. Cada exemplo de treinamento é composto por um conjunto de características e uma classe de saída. O objetivo principal da árvore de decisão é aprender a mapear as características de entrada para a classe de saída, de forma que, quando apresentada com um novo exemplo não rotulado, a árvore seja capaz de prever a classe a qual esse novo exemplo pertence.

O processo de construção da árvore de decisão envolve a seleção de características que melhor separam os exemplos de treinamento em classes distintas. Essa seleção é realizada com base em critérios como a entropia, o índice Gini ou a redução de erro. A ideia é escolher a característica que divide o conjunto de dados de treinamento de forma mais eficaz, minimizando a impureza nas classes resultantes.

Uma vez que a primeira divisão é feita, a árvore se ramifica, criando nós internos que representam novas decisões com base em outras características. Esses nós internos, por sua vez, têm ramos que levam a outros nós internos ou folhas da árvore. As folhas da árvore representam as classes de saída atribuídas aos exemplos após todas as decisões. A estrutura da árvore é construída de forma recursiva, dividindo os dados em subconjuntos menores até que os critérios de parada sejam atendidos, como um limite de profundidade máxima ou um número mínimo de exemplos por folha.

De acordo com Coffield et al. (2019), a interpretabilidade e simplicidade tornam o modelo *Decision Tree* mais transparente para aplicações de classificação de dados. Os autores enfatizam que a principal vantagem das árvores é sua capacidade de capturar relações complexas entre as características de entrada e a classe de saída. No entanto, é importante destacar que árvores de decisão são suscetíveis ao problema de sobreajuste (*overfitting*) caso não sejam devidamente regularizadas. Para mitigar esse risco, técnicas como a configuração apropriada de hiperparâmetros, como a profundidade máxima da árvore, tornam-se essenciais.

3.2.3 Modelo *Support Vector Machine* com otimização *Stochastic Gradient Descent*

O modelo *Support Vector Machine* (SVM) é aplicado em problemas de classificação. Especificamente, no caso linear, o modelo cria um hiperplano que divide o espaço das características (*features*) em regiões nas quais se encontram as amostras de cada classe. O hiperplano é escolhido de forma a maximizar a menor distância entre o mesmo e a amostra mais próxima no espaço das características. Os parâmetros desse

modelo são obtidos por meio de métodos de otimização (TRAPPENBERG, 2020).

O algoritmo *Stochastic Gradient Descent* (SGD) consiste em um método de otimização amplamente usado em aprendizado de máquina. Trata-se de uma variação do gradiente descendente tradicional, que desempenha um papel fundamental na otimização de funções de perda. Segundo Trappenberg (2020), no contexto do aprendizado de máquina, o ato de aprender envolve a busca por parâmetros que minimizem a função de perda.

Diferentemente do gradiente descendente tradicional, que calcula o gradiente da função de perda usando todo o conjunto de treinamento, o SGD adota uma estratégia de cálculo do gradiente para uma amostra aleatória, ou seja, um único ponto de dados (x_i, y_i), por vez. Essa abordagem proporciona uma notável aceleração no processo, especialmente em cenários que envolvem conjuntos de dados volumosos. No entanto, a inclusão de um componente aleatório no processo é uma característica inerente ao SGD.

Em Sra et al. (2012), os autores destacam que SGD pode ser considerado uma aproximação estocástica da otimização de gradiente descendente, uma vez que substitui o gradiente real (calculado a partir de todo o conjunto de dados) por uma estimativa do mesmo (calculada a partir de um subconjunto de dados selecionado aleatoriamente), reduzindo assim a carga computacional.

3.3 Métricas de avaliação

No contexto da análise de modelos de aprendizado de máquina voltados para problemas de classificação, existem diversas métricas que podem ser usadas para avaliação de desempenho. Na RSL, apresentada no Capítulo 2, pode-se observar que as métricas mais usadas foram a precisão, a revocação e o *F1-score*, respectivamente.

Para que seja possível medir o desempenho dos modelos de aprendizado de máquina e quantificar a qualidade das predições, é necessário a análise de métricas padrão para avaliar todos os modelos ou experimentos de maneira igualitária e poder comparar resultados obtidos. Com esse propósito, quatro métricas foram usadas para avaliação dos experimentos realizados nesta dissertação, são elas: taxa de acerto média, precisão, revocação e *F1-score*.

A matriz de confusão é um recurso fundamental na análise de modelos de aprendizado de máquina aplicados a problemas de classificação. Ela fornece uma representação visual do desempenho de um modelo em relação às classes de saída do problema

específico. Por meio da matriz de confusão, é possível avaliar o comportamento de um algoritmo ao categorizar os dados em diferentes classes, mostrando como suas previsões se comparam aos valores reais.

A matriz de confusão é derivada de um conjunto de dados de teste, no qual as instâncias já vêm acompanhadas de rótulos. Essa matriz estrutura as previsões do modelo em quatro categorias distintas, como ilustrado na Tabela 3.1. A diagonal principal dessa matriz, composta pelos Verdadeiros Positivos (VP) e Verdadeiros Negativos (VN), reflete as previsões corretas do modelo, enquanto que a diagonal secundária, constituída pelos Falsos Positivos (FP) e Falsos Negativos (FN), aponta as falhas na classificação.

Tabela 3.1 - Exemplo de matriz de confusão genérica para duas classes.

		Valor predito	
		Positivo (1)	Negativo (0)
Valor real	Positivo (1)	VP	FN
	Negativo (0)	FP	VN

- Verdadeiro Positivo (VP): Quando o modelo prediz que a classe é positiva e, ao verificar a resposta, vê-se que a classe realmente é positiva.
- Verdadeiro Negativo (VN): Quando o modelo prediz que a classe é negativa e, ao verificar a resposta, vê-se que a classe realmente é negativa;
- Falso Positivo (FP): Quando o método prediz que a classe é positiva, mas ao verificar a resposta, vê-se que a classe é negativa;
- Falso Negativo (FN): Quando o método prediz que a classe é negativa, mas ao verificar a resposta, vê-se que a classe é positiva;

Com base nesses quatro itens, a matriz de confusão é organizada conforme a Tabela 3.1, onde as linhas representam as classes reais e as colunas representam as classes previstas pelo modelo. A matriz mostra quantas instâncias pertencem a cada uma das quatro categorias. A partir da matriz, é possível calcular diversas métricas de avaliação. Abaixo são apresentadas as métricas de avaliação usadas na análise dos experimentos realizados nesta dissertação.

- Taxa de acerto média (τ): A taxa de acerto, também conhecida como acurácia, é uma métrica fundamental de avaliação de modelos de classificação.

Conforme a Equação 3.3, a métrica descreve a proporção de instâncias que foram classificadas corretamente pelo modelo em relação ao total de instâncias.

A taxa de acerto é calculada como um valor percentual, que pode variar de 0% (nenhuma instância classificada corretamente) a 100% (todas as instâncias classificadas corretamente). A taxa de acerto, individualmente, pode não ser suficiente para avaliar um modelo de aprendizado de máquina, especialmente quando há classes de saídas desbalanceadas. Para a análise mais abrangente, recomenda-se considerar também outras métricas de avaliação.

$$\begin{aligned} \tau &= \frac{\text{Número de Instâncias Classificadas Corretamente}}{\text{Total de Instâncias}} * 100\% \\ &= \frac{\text{VP + VN}}{\text{VP + VN + FP + FN}} * 100\% \end{aligned} \tag{3.3}$$

- Precisão: Essa métrica tem como objetivo identificar a porcentagem das amostras classificadas como positivas, isto é, quantas amostras foram classificadas de forma correta em relação ao número de classificações positivas. Conforme a Equação 3.4 é possível definir a precisão como sendo o número de verdadeiros positivos dividido pelo número de verdadeiros positivos mais o número de falsos positivos. Essa medida expressa o número de exemplos classificados como pertencentes a uma classe, que realmente são daquela classe, dividido pela soma entre esse número e o número de exemplos classificados nesta classe, mas que pertencem a outras classes.

$$\text{Precisão} = \frac{VP}{VP + FP} \tag{3.4}$$

- Revocação (*recall*): também conhecido na literatura como taxa de verdadeiro positivo ou sensibilidade, corresponde a proporção de instâncias reais positivas que foram corretamente previstas como positivas e é calculada como mostra a Equação 3.5.

$$\text{Revocação} = \frac{VP}{VP + FN} \tag{3.5}$$

- Medida F (*F1-score*): A métrica de avaliação *F1-score* combina a precisão P e a revocação R de um modelo de aprendizado de máquina, proporci-

onando um valor harmônico que sintetiza a qualidade global do modelo (RUSSELL, 2010), conforme a Equação 3.6. Essa métrica é recomendada quando os conjuntos de dados apresentam distribuições desiguais entre as classes de saída, permitindo uma avaliação mais robusta e equilibrada do desempenho da aplicação.

$$F1\text{-}score = 2 * \frac{P * R}{P + R} \tag{3.6}$$

Por fim, destaca-se que essas métricas são fundamentais para avaliar se o modelo tem um desempenho satisfatório, considerando a necessidade de minimizar tanto os FP quanto os FN, dependendo do contexto do problema. Portanto, as métricas de avaliação são ferramentas valiosas para ajustar, otimizar e comparar diferentes algoritmos de classificação, permitindo que os desenvolvedores tomem decisões para melhorar a eficácia dos modelos.

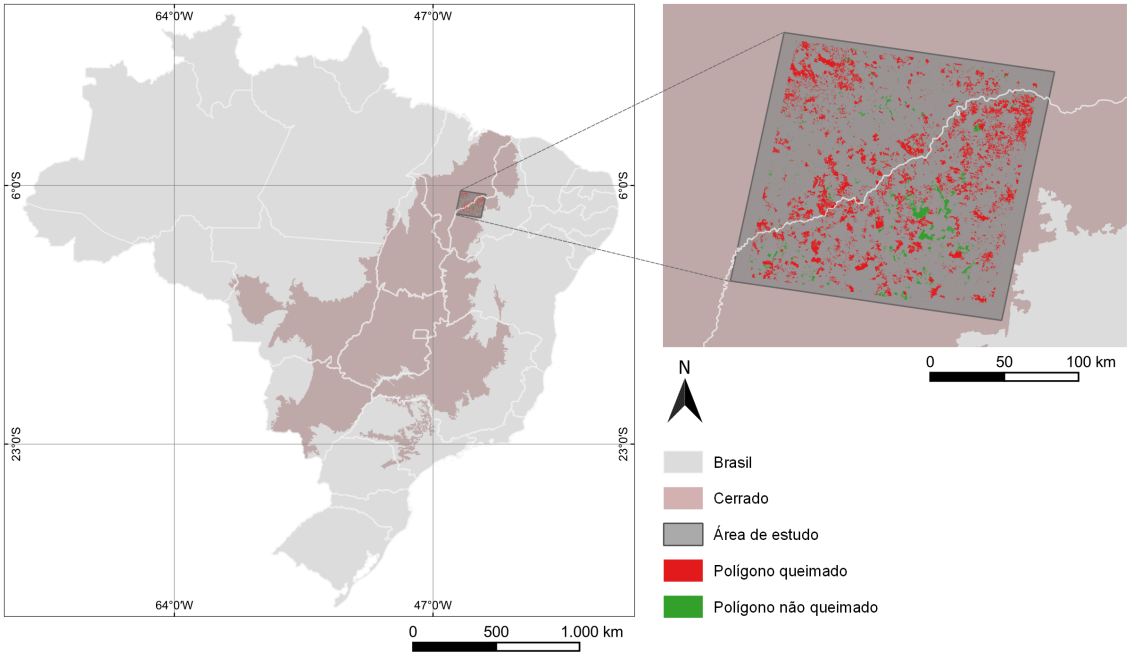
4 DESCRIÇÃO DO MÉTODO PROPOSTO PARA A CLASSIFICAÇÃO AUTOMÁTICA DE ÁREAS QUEIMADAS

Neste capítulo será apresentado o método proposto para a aplicação de aprendizado de máquina para promover a classificação automática de áreas queimadas por meio da análise de séries temporais. O capítulo descreve as etapas realizadas para transformar os conjuntos de dados, referentes aos polígonos, em variáveis de entrada para geração de um modelo capaz de estimar a classificação de novos pontos geográficos. O método apresentado abrange desde a obtenção dos dados até a geração de aprendizado de máquina automático.

4.1 Área de estudo

A Figura 4.1 representa a área de estudo selecionada neste trabalho, enfatizando a órbita-ponto 220-065 do satélite Landsat-8. Essa região encontra-se localizada nos estados do Maranhão e Piauí, sendo caracterizada pelo bioma Cerrado.

Figura 4.1 - Representação da área de estudo.



Área de estudo representada pela órbita-ponto (em cinza escuro), polígonos queimados (em vermelho) e polígonos não queimados (em verde).

Fonte: Elaborada pela autora.

O Cerrado é o segundo maior bioma em extensão territorial do Brasil, abrangendo aproximadamente dois milhões de km². Sua área engloba os estados de Minas Gerais, Mato Grosso, Mato Grosso do Sul, Goiás, Tocantins, Bahia, Maranhão, Piauí, São Paulo e Distrito Federal. O bioma possui uma vegetação tipicamente composta por árvores de pequeno porte, arbustos, gramíneas e diversas espécies de animais que habitam essa região (KLINK; MACHADO, 2005).

De acordo com Bandeira e Campos (2018), o Cerrado brasileiro possui uma importância significativa tanto para o meio ambiente quanto para a sociedade. Com uma diversidade de ecossistemas, paisagens e espécies animais e vegetais, o bioma é considerado o berço das águas do Brasil, abrigando as nascentes de rios importantes, como o São Francisco, o Tocantins e o Araguaia. A escolha dessa área de estudo deve-se à disponibilidade de dados e à necessidade de promover meios automatizados para a classificação de queimadas no Cerrado brasileiro.

4.2 Método proposto

O método proposto nesta dissertação fundamenta-se no processamento de três conjuntos de dados provenientes de situações reais. Esses conjuntos são compostos por polígonos que delimitam regiões no plano, representados por figuras geométricas. Esses polígonos foram auditados de forma a pertencer às classes "Queimada" ou "Não queimada". Os conjuntos de dados correspondem aos anos de 2018, 2019 e 2020, sendo disponibilizados pelo Programa Queimadas do INPE.

O propósito do método proposto consiste em desenvolver um modelo, baseado em aprendizado de máquina, com a capacidade de categorizar com alta precisão novas entradas que representem situações de queimadas e não queimadas. Os resultados gerados por esse modelo têm o potencial de auxiliar os auditores no processo de identificação e validação de incidências de queimadas.

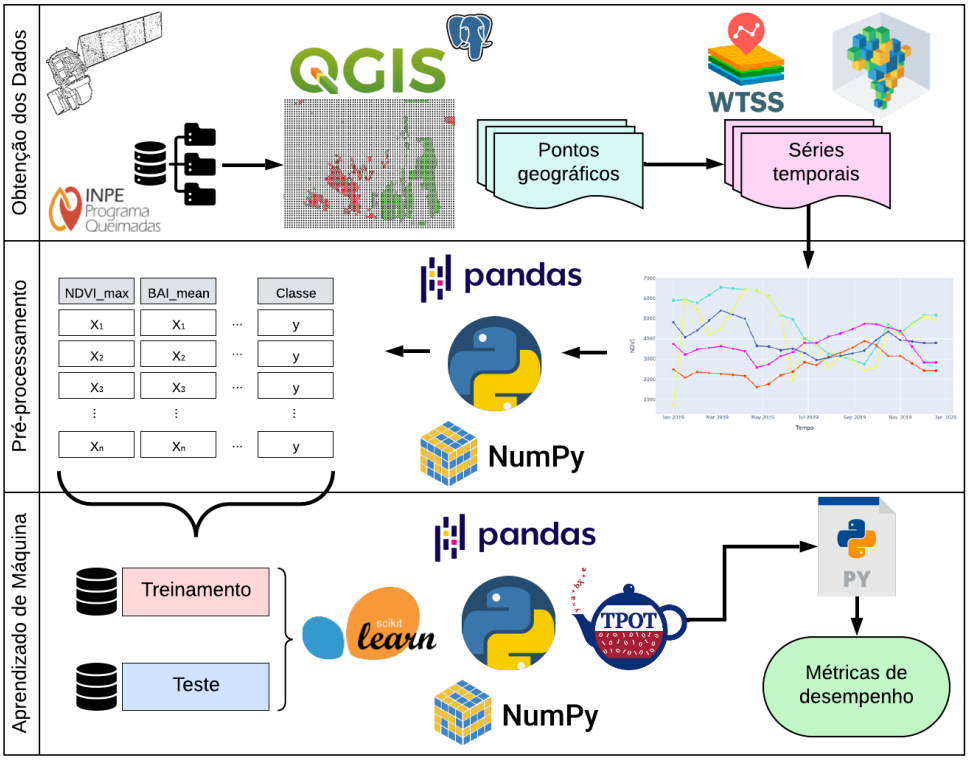
É relevante notar que, atualmente, o Programa Queimadas realiza a classificação de áreas queimadas de maneira semiautomática, destacando a importância do aprimoramento no processo de identificação e validação de ocorrências de queimadas, contribuindo para facilitar o tratamento da classificação por parte dos auditores.

A abordagem usada para promover a classificação automática de áreas queimadas foi baseada na análise de séries temporais de bandas e índices espectrais do satélite Landsat-8. Esta abordagem fundamenta-se na construção de variáveis estatísticas a partir das séries temporais históricas e, posteriormente, no uso dessas variáveis para

o treinamento de um modelo de aprendizado de máquina.

Inicialmente, os conjuntos de dados foram adquiridos e, posteriormente, amostras foram obtidas por meio da geração de pontos geográficos regularmente distribuídos. A partir desses pontos, as séries temporais foram obtidas usando o pacote Python *Web Time Series Service* (WTSS). No estágio de pré-processamento dos dados, foram empregadas estratégias para transformação dos conjuntos de dados brutos. A criação de variáveis estatísticas visa aprimorar a adequação do modelo de aprendizado de máquina durante o treinamento. Neste ponto, presume-se que os dados tenham sido convertidos em características relevantes, prontas para serem alimentadas em algoritmos de aprendizagem supervisionada. A etapa subsequente envolveu a execução do AutoML, a fim de determinar o modelo de aprendizado de máquina mais apropriado.

Figura 4.2 - Visão geral do método proposto para aplicação de aprendizado de máquina na classificação automática de áreas queimadas.



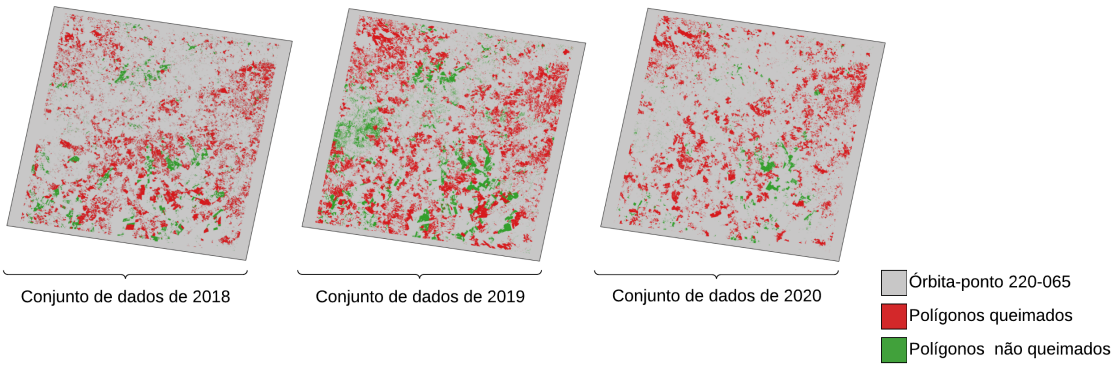
Fonte: Elaborada pela autora.

Por último, o algoritmo de classificação foi implementado com o intuito de criar um modelo capaz de destacar padrões e identificar áreas queimadas e não queimadas em novos conjuntos de dados. Como fase final, é realizada a avaliação dos resultados. A Figura 4.2 ilustra o método proposto para a classificação automatizada de áreas queimadas.

4.2.1 Obtenção dos conjuntos de dados

Neste trabalho, foram usados três conjuntos de dados, referentes aos anos de 2018, 2019 e 2020. Esses conjuntos de dados foram obtidos originalmente por meio de cenas do sensor *Operational Land Imager* (OLI), a bordo do satélite Landsat-8. A resolução espacial desse satélite é de 30 metros e o período de revisita é de 16 dias. Posteriormente, esses três conjuntos de dados foram auditados, pelo Programa Queimadas, como polígonos queimados e não queimados. Para o desenvolvimento deste trabalho, os conjuntos de dados foram disponibilizados na forma de um arquivo *shapefile* e dois arquivos CSV. A Figura 4.3 ilustra as projeções dos *shapefiles*, para os três anos de estudo.

Figura 4.3 - Representação dos conjuntos de polígonos para cada ano de estudo.



Área de estudo representada pela órbita-ponto (em cinza), polígonos queimados (em vermelho) e polígonos não queimados (em verde).

Fonte: Elaborada pela autora.

Primeiramente, os três arquivos foram abertos usando o QGIS. Após a abertura, houve a seleção dos atributos de interesse por meio de filtragem, seguida pela conversão dos arquivos para o formato *shapefile*. A Tabela 4.1 representa a estrutura

do cabeçalho após o filtro dos atributos.

Tabela 4.1 - Estrutura dos arquivos após o filtro dos atributos.

ID polígono	Geom	Satélite	Ano	Órbita-ponto	Data	Área	Classe

A estrutura dos arquivos é formada pelos atributos: “ID polígono”, que representa um identificador único para cada um dos polígonos auditados; “Geom”, que descreve a geometria espacial dos polígonos; “Satélite”, que se refere ao satélite Landsat-8; “Data”, que corresponde a data de passagem do satélite; “Ano”, que corresponde ao período de aquisição dos dados; “Órbita-ponto”, que é o identificador da órbita-ponto; “Área”, que representa o valor da área observada, em hectares; e a “Classe”, que é o resultado da classificação validada por um auditor.

Após essa transformação, os arquivos *shapefile* foram usados para criar e configurar um Banco de Dados Geográficos (BDG) por meio de ferramentas como o QGIS, pgAdmin e Postgres. Nesse contexto, os três arquivos *shapefile* foram importados e integrados ao BDG, permitindo um ambiente propício para criação de consultas SQL e análises geográficas.

4.2.2 Geração de pontos regulares

Esta etapa consiste em gerar pontos geográficos regulares, com foco principal na amostragem da área de estudo. Inicialmente, foi baixado um *shapefile*[1] contendo todas as órbitas-ponto do satélite Landsat-8. Na sequência, esse arquivo foi adicionado no BDG e foi realizada uma consulta SQL para filtrar somente a órbita-ponto 220-065.

No QGIS, para a geração de pontos geográficos regulares, foi construída uma grade de pontos, com base no *shapefile* da órbita-ponto 220-065. O tipo de grade foi “Ponto” e os espaçamentos, horizontal e vertical, foram de 0,001 graus, representando aproximadamente 110 metros de distância. Acredita-se que essa medida de espaçamento entre os pontos é adequada, pois possibilita a captura de áreas significativas dentro dos polígonos, de forma a se obter uma representação abrangente da região de interesse.

[1]Disponível em: https://www.usgs.gov/

A etapa seguinte consistiu em elaborar uma consulta SQL para selecionar os pontos geográficos regulares contidos dentro dos polígonos. A Tabela 4.2 apresenta a relação do total de polígonos dos conjuntos de dados originais, para os anos de 2018, 2019 e 2020, comparado a quantidade de polígonos que contém pontos geográficos em seu interior. Além disso, é possível identificar o total de pontos amostrais separados entre as classes "Queimada", representada pelo dígito 1, e "Não queimada", representada pelo dígito 0.

Tabela 4.2 - Representação da amostragem realizada por meio da grade regular.

Ano	Total de polígonos		Polígonos com pontos		Total de pontos	
	1	0	1	0	1	0
2018	20.499	7.739	17.682	6.122	330.063	84.471
2019	20.553	18.916	18.025	14.975	448.788	189.510
2020	12.897	5.799	11.342	4.367	293.455	48.095

4.2.3 Seleção de polígonos e pontos geográficos

Nesta etapa foram criados dois critérios para a seleção de polígonos e, posteriormente, de pontos geográficos pertencentes aos polígonos. Os critérios de seleção são:

- Critério A: foram selecionados os polígonos que contém mais de 10 pontos geográficos e menos de 100 pontos geográficos no interior de sua geometria.
- Critério B: foram selecionados os polígonos que contém mais de 100 pontos geográficos no interior da sua geometria.

Para os polígonos selecionados no Critério A, todos os pontos geográficos pertencentes ao polígono foram considerados. Entretanto, para os polígonos selecionados no Critério B, foi realizada uma amostragem dos pontos geográficos. Esta amostragem consistiu em eleger 10% do total de pontos de forma aleatória. Esse recurso foi usado devido a grande quantidade de pontos geográficos dentro de um mesmo polígono.

Após a finalização dessa etapa, surge a necessidade de adquirir as séries temporais associadas a cada um desses pontos geográficos selecionados. Essa ação desempenha um papel crucial no próximo passo do processo, que visa à representação desses pontos geográficos.

4.2.4 Aquisição das séries temporais

Para obter as séries temporais das bandas espectrais, índice espectral e máscara de nuvem, foi usado o pacote Python *Web Time Series Service*[2] (WTSS). O WTSS é um serviço desenvolvido pela equipe do *Brazil Data Cube*[3] (BDC) com o propósito de viabilizar a extração de séries temporais a partir de cubos de dados de Observação da Terra.

A coleção de dados usada neste estudo é denominada LC8-30-16D-STK, correspondente ao satélite Landsat-8. As buscas das séries temporais foram divididas em três partes. A primeira parte é referente aos dados do ano de 2018. A segunda parte, referente aos dados de 2019 e, por fim, a terceira parte, referente aos dados de 2020. Para ambas as buscas, a data inicial considerada foi 1 de janeiro e a data final foi 31 de dezembro. Ao término, as séries temporais foram salvas por meio de arquivos do tipo CSV. A Tabela 4.3 apresenta os atributos selecionados do produto de dados.

Tabela 4.3 - Atributos usados do produto de dados LC8-30-16D-STK.

Atributo	**Descrição**
NDVI	Índice espectral do NDVI
Banda 4	Banda espectral do *Red*
Banda 5	Banda espectral do NIR08
Banda 6	Banda espectral do SWIR16
Banda 7	Banda espectral do SWIR22
Fmask	Função para a máscara de nuvem
Data	Data de passagem do satélite

4.2.5 Pré-processamento das séries temporais

O pré-processamento de séries temporais é uma etapa fundamental quando se trata de aplicar técnicas de aprendizado de máquina a conjuntos de dados temporais. Essa fase desempenha um papel importante na garantia da qualidade dos dados e na preparação adequada para algoritmos de aprendizado de máquina, contribuindo significativamente para o desempenho da classificação.

Após a aquisição das séries temporais das bandas espectrais *Red*, NIR08, SWIR16 e SWIR22, bem como do NDVI, o próximo passo envolverá o pré-processamento dos dados coletados, que incluirá uma interpolação com base nas informações fornecidas

[2]Disponível em: https://github.com/brazil-data-cube/code-gallery/tree/master.
[3]Disponível em: http://www.brazildatacube.org/

pelas máscaras de nuvem.

A interpolação é uma técnica usada para preencher lacunas em conjuntos de dados temporais, permitindo obter uma visão mais contínua e suave das informações ao longo do tempo. Nesse contexto, a interpolação desempenha um papel fundamental na restauração de dados ausentes ou irregulares, contribuindo para a obtenção de uma representação mais precisa e completa das séries temporais.

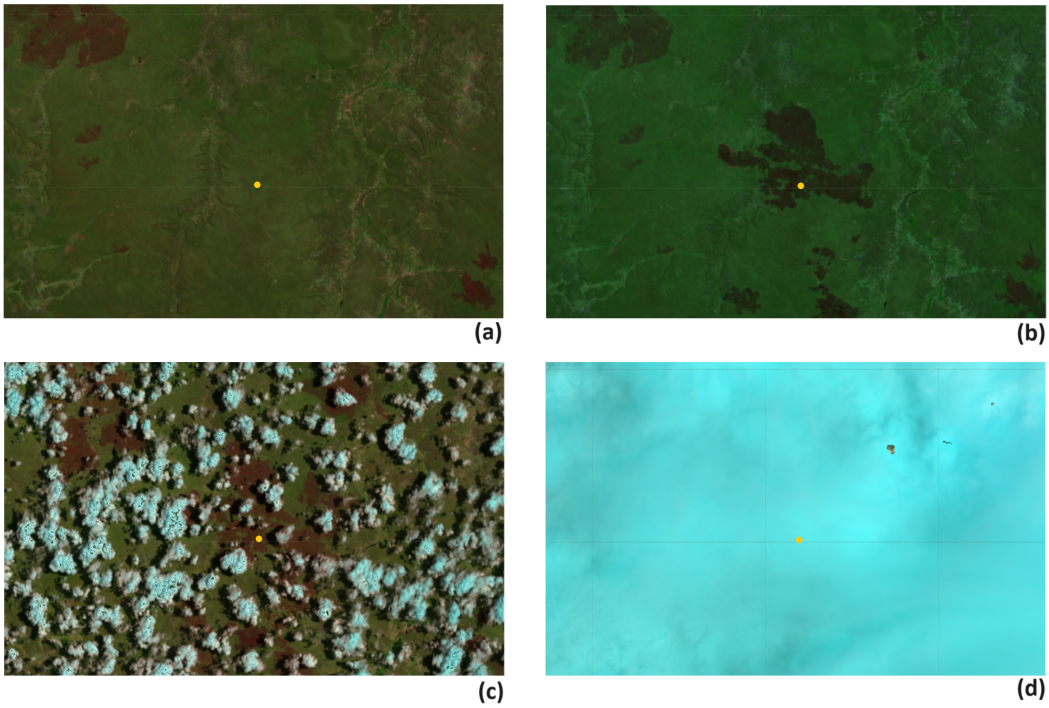
Para representar a importância da interpolação na fase de pré-processamento dos dados, foram ilustradas quatro imagens de cobertura da terra. Nas imagens é destacado um mesmo ponto geográfico. A Figura 4.4 (a) refere-se a data de 29/08/2018, neste período o ponto geográfico representa uma área de vegetação. A imagem da próxima passagem do satélite, dia 14/09/2018, é mostrada na Figura 4.4 (b). Nesta data observa-se a ocorrência de uma queimada. Figura 4.4 (c) refere-se a data de 30/09/2018, pode-se observar que o ponto geográfico representa uma cicatriz de queimada e a imagem possui a presença de várias nuvens. Por fim, 03/12/2018, observa-se que a imagem está completamente encoberta por nuvens. Essas imagens destacam a necessidade de interpolação para preencher lacunas e obter uma representação contínua e das séries temporais, possibilitando uma análise mais consistente ao longo do tempo.

A Tabela 4.4 descreve as diversas classes de máscaras de nuvem encontradas no produto de dados LC8-30-16D-STK. Essas classes são categorizadas com base nos valores dos pixels correspondentes. A tabela enumera as diferentes classes, como *"Clear Land"* (terra limpa), *"Clear Water"* (água limpa), *"Cloud Shadow"* (sombra de nuvem), *"Snow"* (neve), *"Cloud"* (nuvem) e *"No Data"* (sem dados). Cada classe é associada a um valor numérico específico, o qual é usado para representar a presença ou ausência desses elementos em determinados pixels. Essa representação codificada das classes de máscaras de nuvem foi usada para a interpretação das séries temporais de índices espectrais e do NDVI.

Tabela 4.4 - Classes de máscaras de nuvens do produto de dados LC8-30-16D-STK.

Classe	**Valor do pixel**
Clear Land	0
Clear Water	1
Cloud Shadow	2
Snow	3
Cloud	4
No Data	255

Figura 4.4 - Representação de imagens de cobertura da terra.



Imagens da cobertura da terra com destaque de um ponto (em amarelo) referente a coordenada geográfica 7°21' 36.0" S, 42°40' 48.0" W.

Fonte: Adaptada de *Brazil Data Cube* (2018).

No início da análise das séries temporais, foi identificado que estas eram constituídas por 23 pontos no tempo, cada um correspondendo a uma data de passagem do satélite, abrangendo o período de 1 de janeiro a 30 de dezembro de cada ano de estudo. A partir dessa constatação, levando em consideração a presença da máscara de nuvem denominada *"No Data"*, a qual indica a ausência de dados devido a obstruções, tornou-se essencial a aplicação da interpolação. Ao empregar essa técnica nesses pontos desprovidos de informações, os valores ausentes foram estimados com base nos dados vizinhos, possibilitando assim o preenchimento das lacunas nas séries temporais.

Posteriormente, a interpolação foi expandida para incluir as máscaras de nuvem *"Cloud"*, *"Cloud Shadow"* e *"Clear Water"*. Por meio da interpolação aplicada a esses pontos específicos, foi possível estimar os valores temporais correspondentes, proporcionando uma compreensão das variações ao longo do tempo, mesmo em

regiões impactadas por esses elementos atmosféricos. Para esse fim, foram elaboradas e implementadas 17 regras de interpolação de maneira sequencial, visando ao preenchimento das lacunas e a criação de uma representação mais completa das informações.

Para exemplificar a interpolação aplicada nas séries temporais das bandas espectrais *Red*, NIR08, SWIR16, SWIR22 e do índice espectral NDVI, suponha que a série temporal, do NDVI, de um ponto geográfico é representada pelo vetor $\overrightarrow{n}$ e que o vetor $\overrightarrow{m}$ reflete os dados de máscara de nuvem associados aos instantes de tempo da série temporal. Logo:

$$\overrightarrow{n} = [n_1,\ n_2,\ n_3,\ n_4,\ n_5,\ n_6,\ n_7,\ \ldots\ , n_{23}] \tag{4.1}$$

$$\overrightarrow{m} = [m_1,\ m_2,\ m_3,\ m_4,\ m_5,\ m_6,\ m_7,\ \ldots\ , m_{23}] \tag{4.2}$$

Para proporcionar uma ilustração concreta, considere os valores reais:

$$\overrightarrow{n} = [8550,\ 7974,\ -9999,\ 8548,\ 1616,\ 2512,\ 8154,\ \ldots\ , 8550] \tag{4.3}$$

$$\overrightarrow{m} = [0,\ 0,\ 255,\ 0,\ 4,\ 4,\ 0,\ \ldots\ ,\ 0] \tag{4.4}$$

O vetor $\overrightarrow{n}$ é formado por valores que representam a densidade da vegetação ao longo do tempo, com valores mais altos indicando uma vegetação mais densa e saudável. Na Equação 4.3, pode-se observar os valores específicos para cada período de tempo, onde o valor -9999 indica um dado ausente. Já o vetor $\overrightarrow{m}$ é fundamental para a identificação e correção da série temporal do NDVI, visto que, os diferentes valores na Equação 4.4 possuem significados distintos, conforme apresentado na Tabela 4.4.

A seguir, dando início ao procedimento de interpolação do vetor $\overrightarrow{n}$, as seguintes regras são aplicadas de forma sequencial:

Regra 1: Quantidade máxima de pontos inválidos.

Para que uma série temporal seja considerada válida, é necessário que, dos 23 pontos no tempo que a compõem, ou seja, componentes do vetor $\overrightarrow{m}$,

12 ou mais sejam do tipo *"Clear Land"*. Caso contrário, a série temporal é descartada. A exigência de uma quantidade mínima de pontos *"Clear Land"* em uma série temporal visa aumentar a qualidade e confiabilidade dos dados.

Regra 2: Quantidade máxima de pontos inválidos nas extremidades do vetor.

Se os componentes do vetor $\overrightarrow{m}$: $m_1 = m_2 = m_3 = m_4 = m_5 = 255$ ou os componentes $m_{19} = m_{20} = m_{21} = m_{22} = m_{23} = 255$, ou seja, se cinco itens em sequência, no início ou no final do vetor, são do tipo *"No Data"*, então a série temporal é descartada.

Regra 3: Preencher o início com o primeiro valor de boa qualidade.

Se os valores de m_1 até m_k pertencerem ao conjunto $\{255, 1, 2, 3, 4\}$, com $k \leqslant 5$ e $m_{k+1} = 0$, então substitui-se os valores de m_1 até m_k por 0 e substitui-se os valores de n_1 até n_k por n_{k+1}.

Exemplo:

$$\overrightarrow{n} = [-9999,\ -9999,\ 8000,\ \ldots] \tag{4.5}$$

$$\overrightarrow{m} = [255,\ 255,\ 0,\ \ldots] \tag{4.6}$$

Tornam-se:

$$\overrightarrow{n} = [8000,\ 8000,\ 8000,\ \ldots] \tag{4.7}$$

$$\overrightarrow{m} = [0,\ 0,\ 0,\ \ldots] \tag{4.8}$$

Regra 4: Preencher o final com o último valor válido.

Se o valor de m_k até m_{23} for igual a *"No Data"*, com $k \geq 20$, faz-se $m_{23} = m_{k-1},\ m_{22} = m_{k-1},\ \ldots\ , m_k = m_{k-1}$.

Exemplo:

$$\overrightarrow{n} = [8010,\ \ldots,\ 8020,\ -9999,\ -9999, -9999] \tag{4.9}$$

$$\overrightarrow{m} = [0,\ \ldots,\ 0,\ 255,\ 255,\ 255] \tag{4.10}$$

Tornam-se:

$$\overrightarrow{n} = [8010,\ \ldots,\ 8020,\ 8020,\ 8020, 8020] \tag{4.11}$$

$$\overrightarrow{m} = [0,\ \ldots,\ 0,\ 0,\ 0,\ 0] \tag{4.12}$$

Regra 5: Interpolação simples para ponto inválido entre pontos válidos iguais.

Se o valor de $m_k = m_{k+2} = \omega$, com $\omega \in \{0, 1, 2, 3, 4\}$ e $m_{k+1} = 255$, substitui-se m_{k+1} por ω e faz-se a interpolação simples para substituir n_{k+1}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.13}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 255,\ \omega,\ \ldots] \tag{4.14}$$

Torna-se:

$$\overrightarrow{n} = \left[\ \ldots,\ n_k,\ \frac{n_k + n_{k+2}}{2},\ n_{k+2}, \ldots\right] \tag{4.15}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ \omega,\ \omega,\ \ldots] \tag{4.16}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ -9999,\ 8010,\ \ldots] \tag{4.17}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 255,\ 0,\ \ldots] \tag{4.18}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8005,\ 8010,\ \ldots] \tag{4.19}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 0,\ \ldots] \tag{4.20}$$

Regra 6: Interpolação simples para ponto inválido antecedente a *"Clear Land"*.

Se o valor de $m_k = 0$ e $m_{k+2} = \omega$, com $\omega \in \{1, 2, 3, 4, 255\}$ e $m_{k+1} = 255$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_k, pois prefere-se preencher os pontos inválidos com informações mais representativas do estado da vegetação.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.21}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 255,\ \omega,\ \ldots] \tag{4.22}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_k,\ n_{k+2}, \ldots] \tag{4.23}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 0,\ \omega,\ \ldots] \tag{4.24}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ -9999,\ 2000,\ \ldots] \tag{4.25}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 255,\ 2,\ \ldots] \tag{4.26}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8000,\ 2000,\ \ldots] \tag{4.27}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 2,\ \ldots] \tag{4.28}$$

Regra 7: Interpolação simples para ponto inválido subsequente a *"Clear Land"*.

Se o valor de $m_k = \omega$ e $m_{k+2} = 0$, com $\omega \in \{1, 2, 3, 4, 255\}$ e $m_{k+1} = 255$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_{k+2}, pois prefere-se preencher os pontos inválidos com informações mais representativas do estado da vegetação.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.29}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 255,\ 0,\ \ldots] \tag{4.30}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+2},\ n_{k+2}, \ldots] \tag{4.31}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 0,\ 0,\ \ldots] \tag{4.32}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 2000,\ -9999,\ 8000,\ \ldots] \tag{4.33}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 255,\ 0,\ \ldots] \tag{4.34}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 8000,\ 8000,\ \ldots] \tag{4.35}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 0,\ 0,\ \ldots] \tag{4.36}$$

Regra 8: Interpolação simples para ponto inválido entre os pontos *"Cloud"* e *"Cloud Shadow"*.

Se o valor de $m_k = 2$, $m_{k+2} = 4$ e $m_{k+1} = 255$, então substitui-se m_{k+1} por 4 e faz-se a substituição de n_{k+1} por n_{k+2}. Analogamente, se o valor de $m_k = 4$, $m_{k+2} = 2$ e $m_{k+1} = 255$, então substitui-se m_{k+1} por 4 e faz-se a substituição de n_{k+1} por n_k.

A precedência da máscara de nuvem *"Cloud"* em detrimento da máscara de nuvem *"Cloud Shadow"* é fundamentada na similaridade espectral existente entre as sombras das nuvens e as superfícies de água. Esse fato pode resultar em uma omissão na estimativa da área queimada, uma vez que um classificador pode incorretamente identificar as áreas sombreadas como não afetadas pelo fogo (SOUSA et al., 2003).

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.37}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega_1,\ 255,\ \omega_2,\ \ldots] \tag{4.38}$$

Onde $\omega_1 \neq \omega_2$ e $\omega_1, \omega_2 \in \{2, 4\}$.

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ \alpha,\ n_{k+2}, \ldots] \tag{4.39}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega_1,\ 4,\ \omega_2,\ \ldots] \tag{4.40}$$

Onde $\alpha = n_k$ se $\omega_1 = 4$ ou $\alpha = n_{k+2}$ se $\omega_2 = 4$.

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 2000,\ -9999,\ 4000,\ \ldots] \tag{4.41}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 255,\ 4,\ \ldots] \tag{4.42}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 4000,\ 4000,\ \ldots] \tag{4.43}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 4,\ 4,\ \ldots] \tag{4.44}$$

Regra 9: Interpolação simples para ponto *"Cloud"* entre demais pontos válidos iguais.

Se o valor de $m_k = m_{k+2} = \omega$, com $\omega \in \{0, 1, 2, 3\}$ e $m_{k+1} = 4$, substitui-se m_{k+1} por ω e faz-se a interpolação simples para substituir n_{k+1}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.45}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 4,\ \omega,\ \ldots] \tag{4.46}$$

Torna-se:

$$\overrightarrow{n} = \left[\ \ldots,\ n_k,\ \frac{n_k + n_{k+2}}{2},\ n_{k+2}, \ldots\right] \tag{4.47}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ \omega,\ \omega,\ \ldots] \tag{4.48}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8050,\ -4000,\ 8000,\ \ldots] \tag{4.49}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 4,\ 0,\ \ldots] \tag{4.50}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8050,\ 8025,\ 8000,\ \ldots] \tag{4.51}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 0,\ \ldots] \tag{4.52}$$

Regra 10: Interpolação simples para ponto *"Cloud"* antecedente a *"Clear Land"*.

Se o valor de $m_k = 0$ e $m_{k+2} = \omega$, com $\omega \in \{1, 2, 3, 4\}$ e $m_{k+1} = 4$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_k.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.53}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 4,\ \omega,\ \ldots] \tag{4.54}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_k,\ n_{k+2}, \ldots] \tag{4.55}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 0,\ \omega,\ \ldots] \tag{4.56}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 4000,\ 2000,\ \ldots] \tag{4.57}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 4,\ 2,\ \ldots] \tag{4.58}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8000,\ 2000,\ \ldots] \tag{4.59}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 2,\ \ldots] \tag{4.60}$$

Regra 11: Interpolação simples para ponto *"Cloud"* subsequente a *"Clear Land"*.

Se o valor de $m_k = \omega$ e $m_{k+2} = 0$, com $\omega \in \{1, 2, 3, 4\}$ e $m_{k+1} = 4$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_{k+2}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.61}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 4,\ 0,\ \ldots] \tag{4.62}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+2},\ n_{k+2}, \ldots] \tag{4.63}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 0,\ 0,\ \ldots] \tag{4.64}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 4000,\ 8100,\ \ldots] \tag{4.65}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 4,\ 0,\ \ldots] \tag{4.66}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 8100,\ 8100,\ \ldots] \tag{4.67}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 0,\ 0,\ \ldots] \tag{4.68}$$

Regra 12: Interpolação simples para ponto *"Cloud Shadow"* entre demais pontos válidos iguais.

Se o valor de $m_k = m_{k+2} = \omega$, com $\omega \in \{0, 1\}$ e $m_{k+1} = 2$, substitui-se m_{k+1} por ω e faz-se a interpolação simples para substituir n_{k+1}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.69}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 2,\ \omega,\ \ldots] \tag{4.70}$$

Torna-se:

$$\overrightarrow{n} = \left[\ \ldots,\ n_k,\ \frac{n_k + n_{k+2}}{2},\ n_{k+2}, \ldots\right] \tag{4.71}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ \omega,\ \omega,\ \ldots] \tag{4.72}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8050,\ -4000,\ 8000,\ \ldots] \tag{4.73}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 2,\ 0,\ \ldots] \tag{4.74}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8100,\ 8150,\ 8200,\ \ldots] \tag{4.75}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 0,\ \ldots] \tag{4.76}$$

Regra 13: Interpolação simples para ponto *"Cloud Shadow"* antecedente a *"Clear Land"*.

Se o valor de $m_k = 0$ e $m_{k+2} = \omega$, com $\omega \in \{1, 2\}$ e $m_{k+1} = 2$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_k.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.77}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 2,\ \omega,\ \ldots] \tag{4.78}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_k,\ n_{k+2}, \ldots] \tag{4.79}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 0,\ \omega,\ \ldots] \tag{4.80}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 2000,\ 2000,\ \ldots] \tag{4.81}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 2,\ 2,\ \ldots] \tag{4.82}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8000,\ 2000,\ \ldots] \tag{4.83}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 2,\ \ldots] \tag{4.84}$$

Regra 14: Interpolação simples para ponto *"Cloud Shadow"* subsequente a *"Clear Land"*.

Se o valor de $m_k = \omega$ e $m_{k+2} = 0$, com $\omega \in \{1, 2\}$ e $m_{k+1} = 2$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_{k+2}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.85}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 2,\ 0,\ \ldots] \tag{4.86}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+2},\ n_{k+2}, \ldots] \tag{4.87}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 0,\ 0,\ \ldots] \tag{4.88}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 2000,\ 8000,\ \ldots] \tag{4.89}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 2,\ 0,\ \ldots] \tag{4.90}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 2000,\ 8000,\ 8000,\ \ldots] \tag{4.91}$$

$$\overrightarrow{m} = [\ldots,\ 2,\ 0,\ 0,\ \ldots] \tag{4.92}$$

Regra 15: Interpolação simples para ponto *"Clear Water"* entre pontos iguais a *"Clear Land"*.

Se o valor de $m_k = m_{k+2} = \omega$, com $\omega \in \{0\}$ e $m_{k+1} = 1$, substitui-se m_{k+1} por ω e faz-se a interpolação simples para substituir n_{k+1}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.93}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 1,\ \omega,\ \ldots] \tag{4.94}$$

Torna-se:

$$\overrightarrow{n} = \left[\ \ldots,\ n_k,\ \frac{n_k + n_{k+2}}{2},\ n_{k+2}, \ldots\right] \tag{4.95}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ \omega,\ \omega,\ \ldots] \tag{4.96}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 1000,\ 9000,\ \ldots] \tag{4.97}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 1,\ 0,\ \ldots] \tag{4.98}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8050,\ 9000,\ \ldots] \tag{4.99}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 0,\ \ldots] \tag{4.100}$$

Regra 16: Interpolação simples para ponto *"Clear Water"* antecedente a *"Clear Land"*.

Se o valor de $m_k = 0$ e $m_{k+2} = \omega$, com $\omega \in \{1\}$ e $m_{k+1} = 1$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_k.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.101}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 1,\ \omega,\ \ldots] \tag{4.102}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_k,\ n_{k+2}, \ldots] \tag{4.103}$$

$$\overrightarrow{m} = [\ \ldots,\ 0,\ 0,\ \omega,\ \ldots] \tag{4.104}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 1000,\ 1000,\ \ldots] \tag{4.105}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 1,\ 1,\ \ldots] \tag{4.106}$$

Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 8000,\ 8000,\ 1000,\ \ldots] \tag{4.107}$$

$$\overrightarrow{m} = [\ldots,\ 0,\ 0,\ 1,\ \ldots] \tag{4.108}$$

Regra 17: Interpolação simples para ponto *"Clear Water"* subsequente a *"Clear Land"*.

Se o valor de $m_k = \omega$ e $m_{k+2} = 0$, com $\omega \in \{1\}$ e $m_{k+1} = 2$, substitui-se m_{k+1} por 0 e faz-se a substituição de n_{k+1} por n_{k+2}.

Ou seja:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+1},\ n_{k+2}, \ldots] \tag{4.109}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 1,\ 0,\ \ldots] \tag{4.110}$$

Torna-se:

$$\overrightarrow{n} = [\ \ldots,\ n_k,\ n_{k+2},\ n_{k+2}, \ldots] \tag{4.111}$$

$$\overrightarrow{m} = [\ \ldots,\ \omega,\ 0,\ 0,\ \ldots] \tag{4.112}$$

Exemplo:

$$\overrightarrow{n} = [\ldots,\ 1000,\ 1000,\ 8000,\ \ldots] \tag{4.113}$$

$$\overrightarrow{m} = [\ldots,\ 1,\ 1,\ 0,\ \ldots] \tag{4.114}$$

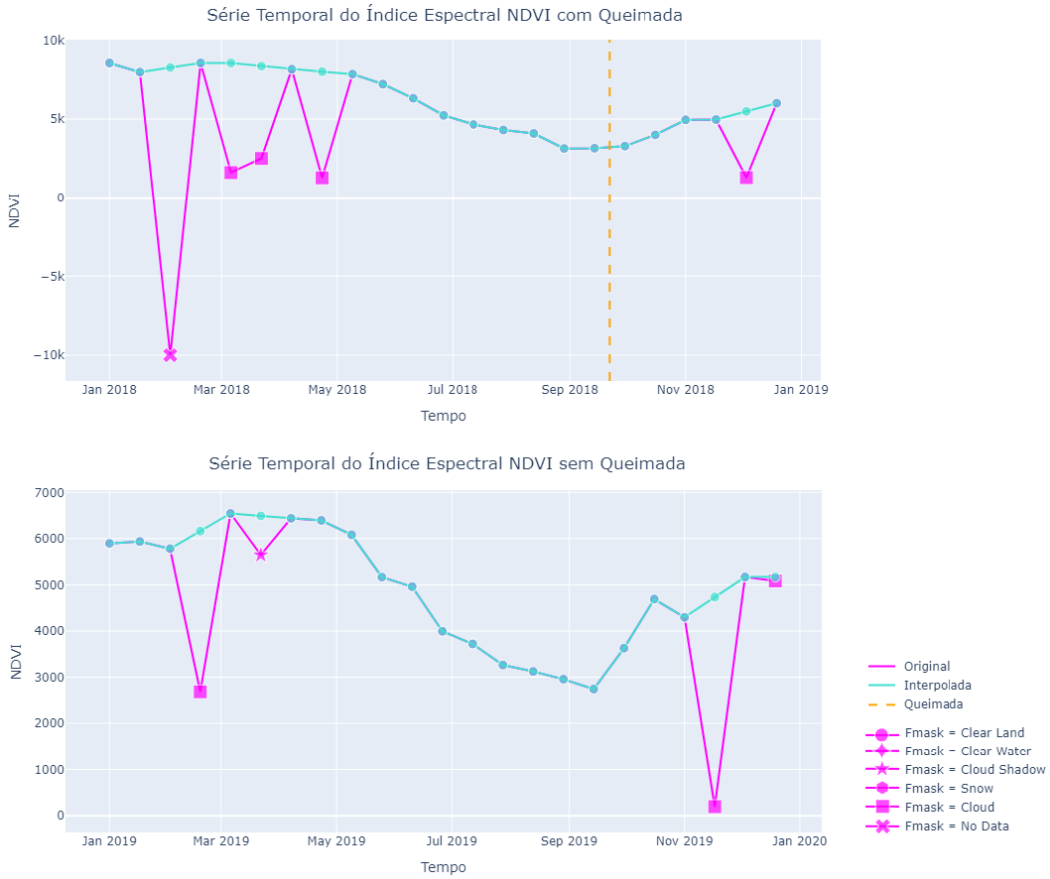
Tornam-se:

$$\overrightarrow{n} = [\ldots,\ 1000,\ 8000,\ 8000,\ \ldots] \tag{4.115}$$

$$\overrightarrow{m} = [\ldots,\ 1,\ 0,\ 0,\ \ldots] \tag{4.116}$$

A Figura 4.5 ilustra dois exemplos de interpolação realizada com duas séries temporais do NDVI. Nos gráficos é possível observar as curvas originais dos dados de NDVI, na cor rosa, e as curvas dos resultados da interpolação, na cor azul. Além disso, é possível identificar as máscaras de nuvem por meio dos marcadores que estão sobrepostos nas séries temporais. Os marcadores círculo, losango, estrela, hexágono, quadrado e "X" indicam *"Clear Land"*, *"Clear Water"*, *"Cloud Shadow"*, *"Snow"*, *"Cloud"* e *"No Data"*, respectivamente.

Figura 4.5 - Séries temporais NDVI, NDVI interpolado e Fmask.



Séries temporais do NDVI originais e interpoladas, marcadores do Fmask e ocorrência de queimada (em laranja).

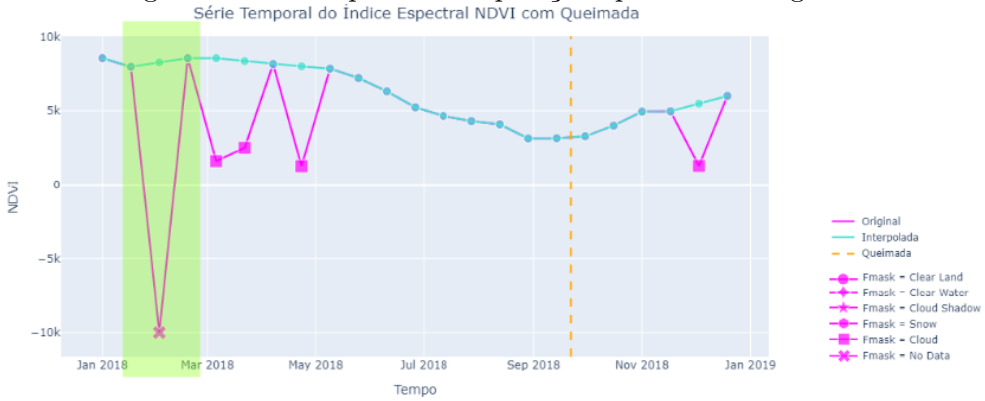
Fonte: Elaborada pela autora.

Na parte superior da Figura 4.5, observa-se uma ocorrência conhecida de queimada, indicada temporalmente pela linha vertical tracejada, na cor laranja. A série temporal original do NDVI, na cor rosa, teve início em janeiro de 2018 e fim em dezembro do mesmo ano. Ao longo do tempo, destaca-se que a série temporal possui 23 pontos no tempo e, pelo menos, 12 desses pontos são do tipo *"Clear Land"*. Para os pontos no tempo com máscara de nuvem diferente de *"Clear Land"*, por exemplo, *"No Data"* e *"Cloud"*, observam-se os resultados da interpolação da série temporal, representados na cor azul.

Na parte inferior da Figura 4.5, a série temporal original do NDVI, na cor rosa, teve início em janeiro de 2019 e fim em dezembro do mesmo ano. Ao longo do tempo, destaca-se que a série temporal possui 23 pontos no tempo e, pelo menos, 12 desses pontos, são do tipo *"Clear Land"*. Para os pontos com máscara de nuvem igual a *"Cloud"* e *"Cloud Shadow"*, observam-se os resultados da interpolação, representado na cor azul.

Na Figura 4.6, no retângulo, na cor amarela, destacam-se três pontos da série temporal do NDVI. Pode-se observar que o terceiro ponto da série temporal original possui a máscara de nuvem igual a "No Data" e que o valor do NDVI é de -9.999. Os pontos vizinhos, anterior e posterior, são do tipo "Clear Land". Neste caso, será aplicada a Regra 5: *Interpolação simples para ponto inválido entre pontos válidos iguais*. Dessa forma, será calculada a média dos valores NDVI, anterior e posterior, este resultado será atribuído ao ponto que anteriormente era "No Data", representado pelo ponto "Clear Land" da série temporal interpolada.

Figura 4.6 - Exemplo de interpolação aplicando a Regra 5.

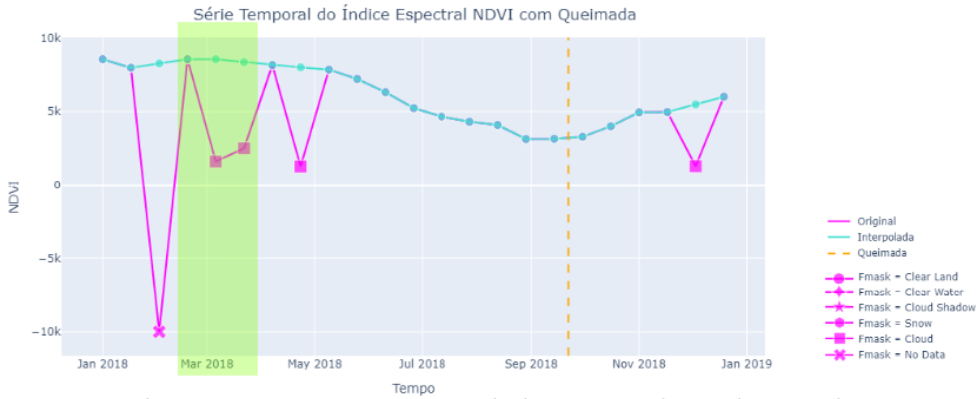


Séries temporais do NDVI originais e interpoladas, marcadores do Fmask e ocorrência de queimada (em laranja).

Fonte: Elaborada pela autora.

Na Figura 4.7, no retângulo, na cor amarela, destacam-se três pontos da série temporal do NDVI. Pode-se observar que o terceiro ponto da série temporal original possui a máscara de nuvem igual a "Cloud". O ponto anterior, é do tipo "Clear Land" e o ponto posterior também é do tipo "Cloud". Neste caso, será aplicada a Regra 10: *Interpolação simples para ponto "Cloud" antecedente a "Clear Land".* . Dessa forma, será atribuído o valor NDVI anterior ao ponto seguinte que, anteriormente, era "Cloud", representado pelo ponto "Clear Land" da série temporal interpolada.

Figura 4.7 - Exemplo de interpolação aplicando a Regra 5.



Séries temporais do NDVI originais e interpoladas, marcadores do Fmask e ocorrência de queimada (em laranja).

Fonte: Elaborada pela autora.

Após a conclusão da etapa de pré-processamento das séries temporais, as quantidades de pontos geográficos selecionados, agora representados por meio das séries temporais, são apresentados na Tabela 4.5.

Tabela 4.5 - Representação do total de pontos geográficos selecionados após o pré-processamento das séries temporais.

Ano	Total de pontos geográficos selecionados	
	Não queimados (0)	Queimados (1)
2018	14.785	46.895
2019	32.482	61.419
2020	3.110	17.051

A Tabela 4.5 mostra a distribuição entre pontos geográficos "Não queimados" (0) e

pontos geográficos “Queimados” (1) para os anos de 2018, 2019 e 2020. Em 2018, foram selecionados um total de 14.785 pontos “Não queimados” e 46.895 pontos “Queimados”. No ano seguinte, em 2019, o número de pontos “Não queimados” aumentou para 32.482, enquanto os pontos “Queimados” totalizaram 61.419. Já em 2020, houve uma diminuição significativa no total de pontos “Não queimados’, com apenas 3.110 pontos, em contraste com 17.051 pontos “Queimados”. Esses dados representam a base para análises subsequentes relacionadas a séries temporais e eventos de queimadas em diferentes anos. Somando essas categorias, pode-se observar que em 2018 houve um total de 61.680 amostras; em 2019, foram 93.901 amostras; e em 2020 obteve-se 20.161 amostras.

Em relação aos números apresentados na Tabela 4.5, observa-se que os três conjuntos de dados revelam variações nas quantidades de amostras entre as classes de “Queimada” e “Não queimada” ao longo dos anos de 2018, 2019 e 2020. No ano de 2018, o conjunto de dados apresenta as seguintes proporções entre as categorias, com “Queimada” representando 76% e “Não queimada” correspondendo a 24%. Em contraste, no conjunto de dados de 2019 as queimadas representam 65,4% e as não queimadas 34,6% dos dados. Já no conjunto de dados de 2020, observa-se um desbalanceamento mais acentuado, com a classe “Queimada” abrangendo 84,6% das amostras, enquanto a classe “Não queimada” está em minoria, com apenas 15,4%.

4.2.6 Geração de variáveis estatísticas

A geração de atributos é um processo fundamental na análise e modelagem de séries temporais, especialmente quando se trata de aplicar algoritmos de aprendizado de máquina. A geração de variáveis estatísticas baseia-se na criação de novos atributos por meio de transformações aplicadas ao conjunto de dados. O propósito é extrair recursos a fim de construir representações que potencializem o treinamento de modelos.

No contexto específico de séries temporais de bandas e índices espectrais relacionados a queimadas e não queimadas, a geração de variáveis estatísticas visa extrair informações-chave das séries temporais para melhorar a capacidade de distinção entre as duas classes.

Foi verificada a necessidade de redução de dimensionalidade devido à grande quantidade de dados, representados por meio das séries temporais. Nos estudos de Wang e Wang (2020) e Sánchez et al. (2021), foi discutido o potencial da combinação de índices espectrais na caracterização de áreas queimadas. Os autores destacam que

o conhecimento do domínio deve ser aplicado para orientar a geração de atributos, uma vez que cada variável gerada introduz complexidade computacional durante o treinamento de um modelo. Assim, a recomendação é evitar a adição de variáveis redundantes e altamente correlacionados.

Na Questão de Análise 1 (QA1), que investiga "Quais características têm sido usadas na literatura para a classificação de áreas queimadas por meio de aprendizado de máquina?", constatou-se que os índices espectrais NDVI, NBR1, NBR2 e BAI, juntamente com as bandas espectrais NIR, SWIR1, SWIR2 e *Red*, foram as características mais empregadas, na sua categoria, nos estudos da RSL. Nesta dissertação, algumas combinações de índices foram testadas e os resultados iniciais dos experimentos mostram que esse conjunto de índices apresentou desempenho promissor. A partir disso, procedeu-se à geração de estatísticas dos índices espectrais NDVI, NBR1, NBR2 e BAI, bem como as bandas espectrais NIR, SWIR1, SWIR2 e *Red*.

O próximo passo envolve a transformação das séries temporais de bandas e índices espectrais para a geração de variáveis estatísticas, que serão representações das séries associadas às áreas queimadas e não queimadas. As métricas calculadas incluem a média ($\bar{x}$), o desvio padrão (s), os valores mínimo e máximo, bem como os quartis (Q_1, Q_2 e Q_3).

A média aritmética ($\bar{x}$) é definida como a soma de todas as observações da variável x dividida pelo número de elementos N do conjunto de dados. A expressão genérica para encontrar a média aritmética é demonstrada na Equação 4.117.

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{4.117}$$

em que x_i é o valor na posição i no conjunto de dados.

O desvio padrão amostral (s) é uma medida que expressa o grau de dispersão de um conjunto de dados. Quanto maior o desvio padrão, maior a dispersão dos dados em torno da média. O desvio padrão amostral é calculado por meio da Equação 4.118.

$$
\begin{aligned}
s &= \sqrt{s^2} \\
&= \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N}}
\end{aligned}
\tag{4.118}
$$

O valor máximo em uma série temporal é o maior valor encontrado entre todas as observações da variável x, enquanto que o valor mínimo é o menor valor encontrado. Esses extremos são frequentemente úteis para compreender as variações nas séries temporais. As expressões genéricas para encontrar o valor máximo e mínimo são demonstradas nas Equações 4.119 e 4.120, respectivamente.

$$
\text{Máximo} = \max(x_1, x_2, \ldots, x_N) = \max_i x_i \tag{4.119}
$$

$$
\text{Mínimo} = \min(x_1, x_2, \ldots, x_N) = \min_i x_i \tag{4.120}
$$

Onde x_i representa o valor na posição i da série temporal.

Em estatística descritiva, os quartis são os valores resultantes de um conjunto de observações ordenadas de forma crescente, que dividem a distribuição em quatro partes iguais. O primeiro quartil, Q1, é o número que deixa 25% das observações abaixo e 75% acima, enquanto o terceiro quartil, Q3, deixa 75% das observações abaixo e 25% acima. Já Q2 é a mediana, deixa 50% das observações abaixo e 50% das observações acima.

Existem duas abordagens que podem ser usadas para calcular os quartis. As abordagens são conhecidas como método inclusivo[4] e método exclusivo[5]. Neste trabalho, para calcular os quartis foi usado o método exclusivo.

O processo de determinação dos quartis inclui duas etapas. Inicialmente, deve-se determinar a posição do quartil no conjunto de dados. Em um segundo momento,

[4]Método inclusivo: quando o conjunto de dados tem um número ímpar de elementos, o elemento correspondente ao Q_2 é incluído em ambas as metades do conjunto de dados para cálculo dos Q_1 e Q_3.

[5]Método exclusivo: quando o conjunto de dados tem um número ímpar de elementos, o elemento correspondente ao Q_2 não é incluído em nenhuma das metades do conjunto de dados para cálculo dos Q_1 e Q_3

deve-se calcular o valor do quartil. Quando o quartil coincide com um elemento do conjunto de dados, a posição é um valor inteiro k e, neste caso, o valor do quartil é imediato. Quando o quartil fica entre dois elementos, a sua posição é um valor não inteiro e, neste caso, é necessário calcular o seu valor fazendo a média dos valores nas posições adjacentes.

A posição dos quartis pode ser calculada matematicamente. Considere-se um conjunto de dados ordenado com n elementos $x_1 \leq x_2 \leq x_3 \leq \cdots \leq x_n$. A posição de Q_2 em um conjunto de dados com n elementos é dada pela Equação 4.121.

$$P_{Q_2} = \frac{n+1}{2} \tag{4.121}$$

A determinação das posições de Q_1 e Q_3 no conjunto de dados varia de acordo com a abordagem aplicada. O método utilizado neste trabalho foi o exclusivo. As Equações 4.122 e 4.123 apresentam os cálculos das posições dos primeiro e terceiro quartis, respectivamente.

$$P_{Q_1} = \begin{cases} K = \frac{n+2}{4}, \ para\ n\ par \\ K = \frac{n+1}{4}, \ para\ n\ ímpar \end{cases} \tag{4.122}$$

$$P_{Q_3} = \begin{cases} K = \frac{3n+2}{4}, \ para\ n\ par \\ K = \frac{3n+3}{4}, \ para\ n\ ímpar \end{cases} \tag{4.123}$$

Uma vez conhecida a posição dos quartis, a próxima etapa consiste em calcular o seu valor. O cálculo do valor dos quartis é equivalente nos métodos inclusivo e exclusivo. A Equação 4.124 apresenta a forma para calcular um quartil Q_p, dada a sua posição k no conjunto de dados.

$$Q_p = \begin{cases} x_k, \ para\ k\ inteiro \\ \frac{x_i + x_{i+1}}{2}, \ para\ k\ não\ inteiro \end{cases} \tag{4.124}$$

sendo $i < k < i + 1$.

4.2.7 Aplicação do aprendizado de máquina automatizado

Neste estudo, foi usado o TPOT, um gerador automatizado de modelos de aprendizado de máquina. O TPOT desempenha um papel fundamental ao investigar as

variedades de algoritmos e hiperparâmetros, com o objetivo de encontrar a configuração ideal para os conjuntos de dados em questão. Utilizando uma abordagem de busca automatizada, o TPOT procura identificar o modelo de aprendizado de máquina mais apropriado, levando em consideração os critérios de desempenho e precisão desejados.

Após a fase de exploração e otimização, realizada pelo TPOT, identificou-se o modelo de aprendizado de máquina mais indicado para os conjuntos de dados de cada experimento específico. Em seguida, procedeu-se à implementação do modelo preditivo selecionado. Essa etapa envolveu a configuração dos parâmetros do modelo, treinamento usando os dados disponíveis e validação para avaliar as métricas de desempenho.

4.2.7.1 Busca dos modelos de aprendizado de máquina

Para cada um dos experimentos realizados neste trabalho, inicialmente, foram determinados quais dados seriam usados para treinamento e teste do modelo. Para os experimentos de mesmo ano, a porcentagem do conjunto de dados para treinamento é de 75%, enquanto a porcentagem do conjunto de dados para teste é de 25%. Para os experimentos de anos distintos, um conjunto de dados completo é usado para treinamento e outro conjunto de dados completo, de outro ano, é usado para teste.

A função *Fit* inicializa o algoritmo de programação genética para encontrar o pipeline com a pontuação mais alta, com base na validação cruzada média de dobras $K = 10$. Em seguida, o pipeline é treinado em todo o conjunto de amostras fornecidas e a instância TPOT pode ser usada como um modelo adequado. Na sequência, o pipeline pode ser avaliado no conjunto de dados de teste por meio da função *Score*. A ferramenta retorna a pontuação do pipeline otimizado nos dados de teste usando a função de pontuação especificada pelo usuário. A função de pontuação padrão do TPOT, para problemas de classificação, é a precisão.

A saída dessas etapa consiste em seis modelos de classificação, que serão usados para prever a classe de novas instâncias sem rótulo. O TPOT gera um arquivo, em linguagem de programação Python, com os modelos resultantes e suas respectivas configurações de hiperparâmetros, conforme os algoritmos apresentados no Apêndice B. Os tópicos abaixo listam os experimentos realizados e apresentam o modelo de classificação, selecionado pelo TPOT, para cada experimento criado.

- Experimento 1: o modelo foi treinado e testado usando as variáveis estatís-

ticas do conjunto de dados de 2018. Primeiramente, o pipeline realiza a redução de dimensionalidade por meio da análise de componentes principais (do inglês, *Principal Component Analysis*) (PCA) e, em seguida, emprega o algoritmo KNN para a classificação. O parâmetro $n\_neighbors = 20$ determina o número de vizinhos mais próximos, $p = 1$ indica o uso da distância de Manhattan para medir a proximidade entre os pontos e $weights =$ "$uniform$" estabelece que todos os vizinhos têm igual peso. Essa combinação específica de etapas do pipeline e hiperparâmetros foi selecionada pelo TPOT durante a otimização automática, visando maximizar o desempenho no Experimento 1. A saída da execução do TPOT para o Experimento 1 é a apresentada no Algoritmo 1 do Apêndice B.

- Experimento 2: foram usadas as variáveis estatísticas do conjunto de dados de 2019 para treinamento e teste do modelo. Neste experimento, o pipeline consiste em uma combinação de dois modelos, um é o classificador *Multinomial Naive Bayes* com suavização e o outro é uma *Decision Tree*, de forma que o resultado do primeiro modelo é passado como entrada do seguinte. No algoritmo de *Decision Tree*, destaca-se a importância na definição do parâmetro $max\_depth = 10$, que serve como um limitador da profundidade da árvore. O desafio é encontrar o valor que equilibra adequadamente as capacidades de ajuste e de generalização, evitando *overfitting*. A saída da execução do TPOT para o Experimento 2 é a apresentada no Algoritmo 2 do Apêndice B.

- Experimento 3: o modelo foi treinado e testado usando as variáveis estatísticas do conjunto de dados de 2020. No Experimento 3, o processo inicia-se com a seleção iterativa das variáveis de entrada mais significativas por meio da aplicação do método *Recursive Feature Elimination* (RFE). O método consiste na seleção de atributos que parte do conjunto completo disponível e, em uma série de iterações, elimina de forma recursiva aqueles que são considerados menos relevantes, com base em sua importância estimada pelo estimador *Extra Trees Classifier*. Na sequência, aplica-se o classificador KNN aos dados resultantes da seleção de atributos. Destaca-se que os parâmetros selecionados para o modelo são os mesmos escolhidos no Experimento 1. A saída da execução do TPOT para o Experimento 3 é a apresentada no Algoritmo 3 do Apêndice B.

- Experimento 4: foram usadas as variáveis estatísticas do conjunto de dados de 2018 e 2019 para treinamento e teste do modelo, respectivamente. O

modelo resultante do experimento é o classificador KNN configurado com 99 vizinhos mais próximos e métrica de distância de Manhattan com $p = 1$. O modelo emprega uma abordagem de pesos uniformes, o que implica que todos os vizinhos têm igual peso na decisão de classificação. A saída da execução do TPOT para o Experimento 4 é a apresentada no Algoritmo 4 do Apêndice B.

- Experimento 5: foram usadas as variáveis estatísticas do conjunto de dados de 2019 e 2020 para treinamento e teste do modelo, respectivamente. O modelo selecionado é uma composição de dois estimadores, encapsulados em um pipeline. O primeiro é o classificador *Decision Tree* com a profundidade máxima da árvore limitada a 6, número mínimo de amostras por folha igual a 13 e número mínimo de amostras para dividir um nó igual a 10. O segundo é o classificador KNN, com 50 vizinhos mais próximos, métrica de distância Euclidiana $p = 2$ e ponderação igual a *"distance"*, o que significa que os vizinhos mais próximos têm um peso maior. Essa abordagem combina as características de ambos os modelos, aproveitando a capacidade de decisão *Decision Tree* e a flexibilidade do KNN para criar uma solução híbrida. A saída da execução do TPOT para o Experimento 5 é a apresentada no Algoritmo 5 do Apêndice B.

- Experimento 6: foram usadas as variáveis estatísticas do conjunto de dados de 2018 e 2019 para treinamento e as variáveis estatísticas do conjunto de dados de 2020 para teste do modelo. Primeiramente, o pipeline aplica uma transformação de escala nos dados de entrada. Em seguida, usa o classificador SVM combinado ao algoritmo de otimização SGD. A saída da execução do TPOT para o Experimento 6 é a apresentada no Algoritmo 6 do Apêndice B.

4.2.7.2 Execução do modelo de aprendizado de máquina selecionado

A etapa posterior a busca do modelo de aprendizado de máquina mais indicado para cada um dos experimentos consiste na implementação desses modelos nos conjuntos de dados formados pelas variáveis estatísticas das séries temporais.

O processo começa com a leitura dos conjuntos de dados rotulados, que variam conforme cada experimento. Os dados foram divididos em dois conjuntos distintos: um conjunto de treinamento e um conjunto de teste. O conjunto de treinamento é usado para treinar o modelo, enquanto o conjunto de teste é usado para avaliar o

desempenho do modelo posteriormente. A divisão é feita de forma aleatória, com 75% dos dados destinados ao treinamento e 25% para teste.

Foi aplicada a normalização por meio de um método denominado *LabelEncoder*. Esse processo é usado para converter valores categóricos em valores numéricos. No exemplo em questão, a variável de saída "Classe" foi transformada para o tipo numérico. A classe "Queimada" é representada pelo dígito 1, enquanto a classe "Não queimada" é representada pelo dígito 0. Quanto às variáveis de entrada, estas passaram por uma normalização.

Na sequência, o modelo de aprendizado de máquina que foi selecionado automaticamente usando a ferramenta AutoML é executado no conjunto de dados de treinamento e teste. A fim de avaliar o desempenho do modelo, é realizada uma validação cruzada com $k = 10$ do conjunto de teste. Isso significa que o conjunto de teste é dividido em 10 partes, e o modelo é treinado e testado 10 vezes, cada vez usando uma parte diferente como conjunto de teste e as outras partes como conjunto de treinamento. Isso ajuda a obter uma estimativa mais confiável do desempenho do modelo. Por último, para uma análise mais aprofundada do classificador, as matrizes de confusão foram geradas e as métricas de desempenho foram calculadas.

4.2.8 Avaliação de desempenho

A avaliação do desempenho de um algoritmo de classificação, por meio da análise da matriz de confusão e das métricas associadas, oferece uma visão completa da eficácia do modelo. Essa análise permite identificar tanto os pontos fortes quanto as áreas que podem ser aprimoradas, fornecendo informações valiosas para o ajuste do modelo e o aprimoramento de sua capacidade de classificação.

No campo do aprendizado de máquina, há diversas métricas disponíveis para avaliar a qualidade de modelos de regressão e classificação. Na Questão de Análise 3 (QA3), que investiga "Quais parâmetros de avaliação têm sido empregados na literatura para avaliar um método de classificação de áreas queimadas?", constatou-se que a precisão, a revocação e o *F1-score* são as métricas de desempenho mais recorrentes nos estudos selecionados na Revisão Sistemática da Literatura. Por esse motivo e considerando que essas métricas são tradicionais para avaliar um modelo de aprendizado supervisionado, neste estudo optou por adotá-las como indicadores de desempenho.

Em seu estudo, Bittencourt et al. (2020), cita que a importância de atingir uma taxa

de sucesso de pelo menos 95% como um critério-chave para avaliar a qualidade da classificação automática de áreas queimadas. Os autores argumentam que alcançar esse nível de precisão é crucial para as aplicações práticas desse processo, destacando a necessidade de modelos de classificação que possam fornecer resultados altamente confiáveis, o que é particularmente relevante em contextos relacionados à detecção e monitoramento de áreas queimadas.

5 AVALIAÇÃO DOS RESULTADOS

Neste capítulo, para a avaliação do método proposto de classificação automática de áreas queimadas, são apresentados os resultados obtidos em diferentes conjuntos de dados de entrada. Primeiramente, é apresentada a metodologia adotada na realização dos experimentos. Em seguida, é realizada uma análise individual de cada um dos cenários experimentais, aprofundando a discussão sobre suas particularidades e resultados. Por fim, conclui-se o capítulo com uma discussão geral que apresenta um panorama dos principais pontos abordados ao longo deste estudo.

5.1 Metodologia adotada na realização dos experimentos

Para responder à pergunta de pesquisa "*É possível determinar áreas queimadas por meio de séries temporais de índices espectrais referentes a pontos geográficos?*", foram realizados um total de seis experimentos distintos, cada um representando uma configuração única de conjunto de dados de entrada e modelo preditivo. Em todos os experimentos, foi aplicada a validação cruzada K-*fold* com 10 partições. Ao término da execuções, foram geradas as matrizes de confusão e calculadas as métricas de desempenho: taxa de acerto média, precisão, revocação e F1-*score*.

A validação cruzada com 10 partições foi a escolhida em (WANG et al., 2021), (BARRETO; ARMENTERAS, 2020) e (WANG; WANG, 2020). De acordo com os autores, ao dividir o conjunto de dados em 10 partições distintas, o método proporciona uma avaliação mais robusta do modelo, mitigando o impacto de possíveis variações na distribuição dos dados. A utilização de $k = 10$ permite uma abordagem mais abrangente, pois o conjunto de dados é repetidamente dividido em 10 partes, sendo que em cada iteração, 9 partes são usadas para treinamento e a parte restante para validação. Essa abordagem repetitiva contribui para uma avaliação mais confiável da capacidade de generalização do modelo, reduzindo o risco de *overfitting* e proporcionando uma visão mais representativa do desempenho médio do modelo em diferentes subconjuntos de dados.

Os experimentos foram executados em uma mesma máquina, com sistema operacional de 64 bits. A máquina usada apresenta as seguintes configurações técnicas: processador Intel® Core™ i5-6300U e memória RAM de 8 GB. Essas configurações técnicas possibilitaram as execuções da ferramenta TPOT e dos modelos de aprendizado de máquina selecionados.

5.1.1 Análise dos resultados do Experimento 1

No primeiro experimento deste estudo, foi usado o conjunto de dados referente ao ano de 2018 para treinamento e teste do modelo de classificação automática de áreas queimadas. Essa escolha foi feita com o objetivo de avaliar se o modelo apresenta um bom desempenho ao ser treinado e testado com amostras de mesmo ano. Considerando que treinar e testar com amostras de mesmo período anual possa gerar uma menor variabilidade nos dados, espera-se que as métricas de desempenho sejam superiores quando comparadas com experimentos treinados e testados com amostras de anos distintas.

A execução do AutoML TPOT resultou no modelo de classificação *K-Nearest Neighbors*. O tempo de execução do *framework* para indicar esse modelo foi de 3h28min. Neste experimento, a taxa de acerto média foi de 95,72% com desvio padrão de 0,71%. A Tabela 5.1 apresenta a matriz de confusão.

Tabela 5.1 - Matriz de confusão do Experimento 1.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	3413	317
Queimada	343	11347

A matriz de confusão compara as classes previstas pelo modelo de classificação com as classes reais dos dados, permitindo a análise de acertos e erros em cada categoria. Com base nos valores presentes na matriz de confusão, é possível calcular as métricas de desempenho: precisão, revocação e *F1-score*.

A Tabela 5.2 apresenta os resultados das métricas de desempenho do classificador. É possível destacar que a medida de precisão e de revocação (também conhecida como probabilidade de detecção), para as duas classes, apresentaram pequenas diferenças. Esse fato sugere que o preditor não é tendencioso para apenas uma das classes de saída. Outra observação importante é que todos os valores das métricas de desempenho foram superiores a 90%, muito próximo do valor desejado para se considerar um método automático de classificação de áreas queimadas como ótimo (95%).

Tabela 5.2 - Métricas de desempenho obtidas por meio do Experimento 1.

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	91	92	91
Queimada	97	97	97

5.1.2 Análise dos resultados do Experimento 2

No segundo experimento, foi usado o conjunto de dados referente ao ano de 2019 para treinamento e teste do modelo de classificação automática de áreas queimadas. Essa escolha foi feita com o objetivo de avaliar se o modelo apresenta um bom desempenho ao ser treinado e testado com amostras de mesmo ano.

A execução do AutoML TPOT resultou no modelo de classificação *Decision Tree*. O tempo de execução do *framework* para indicar esse modelo foi de 4h14min42s. A Tabela 5.3 apresenta a matriz de confusão. Neste experimento, a taxa de acerto média foi de 91,02% com desvio padrão de 0,54%.

Tabela 5.3 - Matriz de confusão do Experimento 2.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	7207	986
Queimada	1131	14152

A Tabela 5.4 apresenta os resultados das métricas de desempenho do classificador.

Tabela 5.4 - Métricas de desempenho obtidas por meio do Experimento 2.

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	86	88	87
Queimada	93	93	93

5.1.3 Análise dos resultados do Experimento 3

No terceiro experimento, foi usado o conjunto de dados referente ao ano de 2020 para treinamento e teste do modelo de classificação automática de áreas queimadas.

Essa escolha foi feita com o objetivo de avaliar se o modelo apresenta um bom desempenho ao ser treinado e testado com amostras de mesmo ano.

A execução do AutoML TPOT resultou no modelo de classificação *K-Nearest Neighbors*. O tempo de execução do *framework* para indicar esse modelo foi de 3h01min37s. A Tabela 5.5 apresenta a matriz de confusão. Neste experimento, a taxa de acerto média foi de 96,35% com desvio padrão de 0,54%.

Tabela 5.5 - Matriz de confusão do Experimento 3.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	671	117
Queimada	64	4189

A Tabela 5.6 apresenta os resultados das métricas de desempenho do classificador.

Tabela 5.6 - Métricas de desempenho obtidas por meio do Experimento 3 .

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	91	85	88
Queimada	97	98	98

5.1.4 Análise dos resultados do Experimento 4

No quarto experimento deste estudo, foi adotada uma abordagem diferenciada ao se utilizar dois conjuntos de dados distintos. No processo de treinamento do modelo preditivo, foi usado exclusivamente os dados do ano de 2018. Em contrapartida, para realizar o teste do modelo, recorreu-se ao conjunto de dados específico do ano de 2019. Esse experimento teve como objetivo avaliar o desempenho do modelo quando treinado e testado com amostras provenientes de período anual diferente. Essa análise ajudará a compreender a influência da temporalidade na classificação de áreas queimadas.

A execução do AutoML TPOT resultou no modelo de classificação *K-Nearest Neighbors*. O tempo de execução do *framework* para indicar esse modelo foi de 5h40min18s. A Tabela 5.7 apresenta a matriz de confusão. Neste experimento, a taxa de acerto média foi de 91,12% com desvio padrão de 2,07%.

Tabela 5.7 - Matriz de confusão do Experimento 4.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	28872	3610
Queimada	4723	56696

A Tabela 5.8 apresenta os resultados das métricas de desempenho do classificador.

Tabela 5.8 - Métricas de desempenho obtidas por meio do Experimento 4.

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	86	89	87
Queimada	94	92	93

5.1.5 Análise dos resultados do Experimento 5

No quinto experimento deste estudo, foi adotada uma abordagem diferenciada ao se utilizar dois conjuntos de dados distintos. No processo de treinamento do modelo preditivo, foi usado exclusivamente os dados do ano de 2019. Em contrapartida, para realizar o teste do modelo, recorreu-se ao conjunto de dados específico do ano de 2020. Esse experimento teve como objetivo avaliar o desempenho do modelo quando treinado e testado com amostras provenientes de período anual diferente. Essa análise ajudará a compreender a influência da temporalidade na classificação de áreas queimadas.

A execução do AutoML TPOT resultou no modelo de classificação *K-Nearest Neighbors*. O tempo de execução do *framework* para indicar esse modelo foi de 5h02min44s. A Tabela 5.9 apresenta a matriz de confusão. Neste experimento, a taxa de acerto média foi de 94,51% com desvio padrão de 1,91%.

Tabela 5.9 - Matriz de confusão do Experimento 5.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	2466	644
Queimada	462	16589

A Tabela 5.10 apresenta os resultados das métricas de desempenho do classificador.

Tabela 5.10 - Métricas de desempenho obtidas por meio do Experimento 5.

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	84	79	82
Queimada	96	97	97

5.1.6 Análise dos resultados do Experimento 6

No último experimento deste estudo, foram usados os três conjunto de dados. No processo de treinamento do modelo preditivo, foram usados os dados dos anos de 2018 e 2019, enquanto que, para teste do modelo, recorreu-se ao conjunto de dados do ano de 2020. Esse experimento teve como objetivo avaliar o desempenho do modelo quando treinado e testado com amostras provenientes de períodos anuais diferentes. Essa análise ajudará a compreender a influência da temporalidade na classificação de áreas queimadas.

A execução do AutoML TPOT resultou no modelo de classificação SVM treinado com o algoritmo de otimização SGD. O tempo de execução do *framework* para indicar esse modelo foi de 6h17min33s. A Tabela 5.11 apresenta a matriz de confusão. Neste experimento, a taxa de acerto média foi de 95,55% com desvio padrão de 1,78%.

Tabela 5.11 - Matriz de confusão do Experimento 6.

Classe real	**Classe predita**	
	Não queimada	**Queimada**
Não queimada	2538	572
Queimada	405	16646

Pode-se destacar que um modelo de aprendizado de máquina com taxa de acerto média de 95,55% e um desvio padrão de 1,78% pode ser considerado um bom resultado. Primeiramente, porque uma taxa de acerto média de 95,55% significa que o modelo está classificando corretamente a maioria das amostras de teste, indicando desempenho geral na tarefa de classificação. O desvio padrão de 1,78% indica a variabilidade dos resultados do modelo em diferentes partições dos dados de teste. Quanto menor

o desvio padrão, mais estável e consistente é o desempenho do modelo.

Com base nas métricas de desempenho apresentadas na Tabela 5.12, observa-se que o modelo de aprendizado de máquina obteve resultados promissores. A precisão do modelo é alta, com 97% para a classe "Queimada" e 86% para a classe "Não queimada". O *F1-score*, que combina precisão e revocação, é de 97% para "Queimada" e 84% para "Queimada". Essas métricas indicam que o modelo, mesmo que treinado com amostras de anos anteriores e testado com amostras de anos posteriores, é capaz de realizar uma boa classificação de áreas queimadas.

Tabela 5.12 - Métricas de desempenho obtidas por meio do Experimento 6.

Classe real	**Métricas de desempenho**		
	Precisão (%)	***Recall* (%)**	***F1-score* (%)**
Não queimada	86	82	84
Queimada	97	98	97

Considerando que o modelo está sendo aplicado para a classificação de áreas queimadas no Cerrado brasileiro, os resultados do Experimento 6 são promissores. O Cerrado é um bioma complexo, com características distintas e diversidade de padrões de queimadas. Portanto, alcançar uma taxa de precisão significativamente elevada nesse contexto é um forte indicativo de que o modelo está eficazmente aprendendo a discernir entre as diferentes classes relacionadas a áreas queimadas.

5.1.7 Discussões dos resultados

Os resultados obtidos no Experimento 1, conforme evidenciados pela matriz de confusão e métricas de desempenho apresentadas nas Tabelas 5.1 e 5.2, revelam um desempenho promissor do modelo de classificação, indicando uma boa capacidade do modelo em distinguir entre áreas queimadas e não queimadas. A precisão, a revocação e a medida *F1-score* para ambas as classes estão consistentemente acima de 90%, o que sugere uma eficácia geral do modelo.

A análise comparativa entre os Experimentos 2 e 4 revela aspectos relevantes no contexto da classificação automática de áreas queimadas. No Experimento 2, observou-se uma métrica de desempenho notável, com uma taxa média de acerto de 91,02% e um desvio padrão de 0,54%. Este experimento usou dados de treinamento e teste coletados no mesmo ano. Em contraste, o Experimento 4 apresentou uma taxa média de acerto ligeiramente superior, atingindo 91,12%, com um desvio padrão de

2,07%. O Experimento 4 empregou dados de treinamento e teste de anos distintos. Destaca-se que, embora o Experimento 4 tenha sido treinado com amostras de um ano diferente do conjunto de teste, este apresentou uma taxa de acerto média muito semelhante quando comparada a taxa de acerto média do Experimento 2.

A análise dos Experimentos 3, 5 e 6 revela uma consideração importante, já que todos eles utilizam o conjunto de dados de teste referente ao ano de 2020. A principal distinção entre esses experimentos está nos conjuntos de dados de treinamento empregados. No Experimento 3, alcançou-se o maior desempenho, com uma taxa média de acerto de 96,35% e um desvio padrão de 0,54%, empregando dados de treinamento e teste referentes ao ano de 2020. Em contraste, o Experimento 5 e o Experimento 6 adotaram uma abordagem diferente, usando amostras de anos distintos para treinamento e teste, visando responder à pergunta de pesquisa.

Embora o Experimento 3 tenha apresentado a maior taxa de acerto média, os Experimentos 5 e 6 alcançaram resultados muito semelhantes. Portanto, a capacidade dos Experimentos 5 e 6 de manter um desempenho comparável ao do Experimento 3, apesar da diferença temporal nos dados de treinamento e teste, destaca a promissora aplicabilidade de métodos de aprendizado de máquina na tarefa de identificação de áreas queimadas, mesmo em cenários com variações temporais. Esses resultados sugerem que os modelos desenvolvidos demonstraram uma capacidade de generalização e adaptação a variações temporais nas características das áreas queimadas.

Pode-se destacar que os conjuntos de dados dos anos de 2018, 2019 e 2020 não possuem as classes de saída “Queimada” e “Não queimada” equilibrados. Para todos os anos, a classe “Queimada” é majoritária. A maior desproporcionalidade é encontrada no conjunto de dados de 2020. No contexto de aprendizado de máquina, uma distribuição distorcida dos valores-alvo pode causar um viés de precisão nos algoritmos e afetar negativamente o desempenho dos modelos. Quando uma das classes é significativamente mais numerosa do que a outra, o modelo tende a favorecer a classe majoritária, resultando em um viés na predição em direção a essa classe. Observa-se que, para todos os experimentos realizados, as métricas de desempenho precisão, revocação e *F1-score* foram superiores para a classe “Queimada”, que é a majoritária, o que é esperado.

Por fim, as análises comparativas dos experimentos revelaram aspectos interessantes na classificação automática de áreas queimadas, considerando diferentes abordagens temporais nos conjuntos de treinamento e teste. Enquanto o Experimento 3 se destacou com a mais alta taxa de acerto média, os Experimentos 5 e 6 demonstraram

uma notável capacidade de manter um desempenho comparável, apesar das variações temporais nos dados. Isso sugere a promissora aplicabilidade do método proposto para identificar áreas queimadas em períodos anuais distintos.

6 CONCLUSÕES

A classificação de áreas queimadas desempenha um papel essencial no mapeamento de regiões que sofreram queimas, uma preocupação cada vez mais urgente em várias partes do mundo. A capacidade de identificar áreas afetadas pelo fogo, por meio de técnicas de sensoriamento remoto e aprendizado de máquina, oferece oportunidades significativas para ajudar na gestão e na preservação dos biomas brasileiros. A incorporação de abordagens de classificação automatizada pode promover a redução do tempo necessário para realizar a definição de áreas queimadas anualmente. No contexto econômico, a implantação de sistemas de classificação automática traz a expectativa de converter os trabalhos de auditoria em tratamentos de eventos duvidosos gerados por um classificador e, assim, minimizar custos e reduzir o tempo necessário para a atividade.

Como pode ser observado, há uma demanda relacionada a concepção de métodos de classificação ou regressão destinados a estimativa de áreas queimadas. Neste contexto, esta dissertação propôs o desenvolvimento de um método baseado em aprendizado de máquina para a classificação automática de áreas queimadas, por meio de análises de séries temporais de bandas espectrais e NDVI provenientes do satélite Landsat-8. A pergunta de pesquisa que orientou esse trabalho foi a seguinte: *"É possível determinar áreas queimadas por meio de séries temporais de índices espectrais referentes a pontos geográficos?"*. De maneira mais específica, o estudo investigou se um modelo de classificação supervisionada, treinado previamente com amostras de áreas queimadas de um ano específico, demonstraria a capacidade de reconhecer e classificar ocorrências de queimadas em períodos anuais subsequentes.

O método desenvolvido nesta pesquisa foi obtido após as diversas etapas apresentadas no Capítulo 4, incluindo a aquisição de dados e a geração de amostras a partir de pontos geográficos regularmente distribuídos. As séries temporais iniciais foram submetidas a um processo de pré-processamento, que incluiu transformar os dados em características relevantes para treinar os algoritmos de aprendizado de máquina. Foram realizados seis experimentos para avaliar a classificação de áreas queimadas, empregando AutoML para buscar modelos mais adequados. Por fim, os algoritmos de classificação foram implementados, visando obter um modelo capaz de identificar áreas queimadas e não queimadas em novos conjuntos de dados.

Na presente dissertação, realizou-se uma análise abrangente de diferentes abordagens para a classificação automática de áreas queimadas, considerando variações temporais nos conjuntos de treinamento e teste. Os experimentos executados reve-

laram *insights* relevantes no contexto de classificação de áreas queimadas. Embora o Experimento 3 tenha alcançado a mais alta taxa de acerto média, destacando-se com um desempenho notável, os Experimentos 5 e 6 demonstraram uma habilidade apreciável de manter um desempenho comparável, apesar das variações temporais nos dados, apontando para a promissora aplicabilidade de técnicas de aprendizado de máquina na classificação de áreas queimadas em períodos anuais distintos.

Em relação à distribuição das classes de saída, no conjunto de dados de 2020, observou-se um desbalanceamento mais acentuado em relação aos conjuntos de dados dos anos de 2018 e 2019, com a classe "Queimada" prevalecendo de forma significativa, abrangendo 84,6% das amostras, enquanto a classe "Não queimada" estava em minoria, com apenas 15,4%. Naturalmente, em todos os experimentos conduzidos as métricas de desempenho foram superiores para a classe mais representada, ou seja, "Queimada". Porém, as métricas de desempenho para a classe "Não queimada" também apresentaram bons resultados, demonstrando a robustez do método.

Os resultados alcançados no Experimento 6, como descritos na Subseção 5.1.6, são considerados satisfatórios para fomentar a discussão sobre a classificação automatizada de áreas queimadas no Cerrado brasileiro. Neste experimento, foi obtida uma taxa de acerto média de 95,55%, com um desvio padrão de 1,78%. A precisão para a classe "Queimada" foi de 97% e para a classe "Não queimada" foi de 86%. Esses resultados são promissores, uma vez que, de acordo com as referências apresentadas em Bittencourt et al. (2020), indicam um ideal almejado de precisão de 95%.

É importante destacar que a pergunta de pesquisa foi respondida com êxito com base nos resultados e nas conclusões obtidas ao longo do trabalho. Os resultados alcançados nos experimentos 4, 5 e 6 demonstraram que é possível classificar áreas queimadas em diferentes períodos anuais com taxas de acerto significativas. Portanto, esta dissertação responde de maneira afirmativa à pergunta de pesquisa, indicando que a aplicação de técnicas de aprendizado de máquina em séries temporais de índices espectrais é eficaz na identificação de áreas queimadas, mesmo diante das variações temporais anuais.

Por fim, de acordo com o presente estudo, pode-se concluir que o método desenvolvido, apresenta potencial para a concepção de sistemas de classificação automática de áreas queimadas. A avaliação de experimentos distintos demonstrou que é possível classificar conjuntos de diferentes períodos anuais, respondendo afirmativamente à pergunta de pesquisa. Sendo assim, este trabalho apresentou sua contribuição, uma vez que os resultados obtidos não apenas reforçam a aplicabilidade, mas também

o potencial das técnicas de aprendizado de máquina para abordar esse problema, mesmo em cenários desafiadores com variações temporais anuais.

6.1 Perspectivas para trabalhos futuros

Durante o desenvolvimento desse estudo, foram identificados alguns aspectos que podem servir como base para futuras investigações. Este trabalho concentrou-se na avaliação da capacidade de um modelo de aprendizado treinado com amostras de queimadas de um ano específico para classificar ocorrências de queimadas em períodos anuais subsequentes. No entanto, diversas perspectivas de estudos futuros podem ser consideradas, incluindo:

- Futuras investigações podem ampliar a abordagem atual adicionando novas variáveis para a classificação de áreas queimadas. Uma sugestão seria a incorporação de variáveis climáticas ao modelo, pois o entendimento da dinâmica das condições meteorológicas, dinâmica de chuvas, como padrões de precipitação, temperatura e umidade, pode proporcionar discussões importantes sobre os padrões temporais das queimas, para melhor discriminar as ocorrências de queimadas.

- A pesquisa em questão foi realizada considerando a órbita-ponto 220-065 do satélite Landsat-8. Uma perspectiva de pesquisa futura seria aplicar o método proposto em regiões para as quais não há amostras previamente validadas. Esta abordagem poderia estender a aplicabilidade do modelo para áreas geograficamente relacionadas, avaliando a robustez e a generalização do método em diferentes contextos geográficos.

- Elaborar um estudo combinando as séries temporais, de bandas e índices espectrais, com dados de focos ativos de fogo. A inclusão dessas informações possivelmente melhoraria a identificação de áreas queimadas.

- Realizar um estudo comparativo entre o desempenho dos modelos selecionados pelo AutoML com o modelo mais usado segundo a Revisão Sistemática da Literatura, no caso o *Random Forest*.

- Explorar a aplicação de Redes Neurais Convolucionais (CNNs) na classificação de áreas queimadas. Ao integrar essa arquitetura neural, que é especialmente hábil em aprender hierarquias de padrões, o modelo resultante pode apresentar uma capacidade aprimorada de discriminação e generalização.

REFERÊNCIAS BIBLIOGRÁFICAS

ALZUBAIDI, L.; ZHANG, J.; HUMAIDI, A. J.; AL-DUJAILI, A.; DUAN, Y.; AL-SHAMMA, O.; SANTAMARíA, J.; FADHEL, M. A.; AL-AMIDIE, M.; FARHAN, L. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. **Journal of Big Data**, v. 8, n. 53, 2021. 24, 26

ANDRADE, R. M.; SHIGUEMORI, E. H.; SANTOS, R. D. C. Convolutional neural network and LSTM applied to abnormal behaviour detection from highway footage. **Anais do Computer on the Beach**, v. 13, p. 051–058, 2022. 24

BANDEIRA, M. N.; CAMPOS, F. I. Bioma cerrado: relevância no cenário hídrico brasileiro. In: IX SIMPÓSIO NACIONAL DE CIÊNCIA E MEIO AMBIENTE. **Anais...** [S.l.]: 9, 2018. v. 2, p. 399–409. 38

BARRETO, J. S.; ARMENTERAS, D. Open data and machine learning to model the occurrence of fire in the ecoregion of "Llanos Colombo–Venezolanos". **Remote Sensing**, v. 12, n. 23, p. 3921, 2020. 21, 71

BITTENCOURT, O. O.; MORELLI, F.; SANTOS, C. A. J.; SANTOS, R. Evaluating classification models in a burned areas' detection approach. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS. **Proceedings...** [S.l.], 2019. p. 577–591. 2, 21

BITTENCOURT, O. O.; MORELLI, F.; SANTOS, C. A. J.; ; SANTOS, R. An approach to classify burned areas using few previously validated samples. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS. **Proceedings...** [S.l.], 2020. p. 239–254. 2, 14, 21, 68, 82

BUCZAK, A. L.; GUVEN, E. A survey of data mining and machine learning methods for cyber security intrusion detection. **IEEE Communications Surveys & Tutorials**, v. 18, n. 2, p. 1153–1176, 2015. 19

CHANDEL, A.; SARWAT, W.; NAJAH, A.; DHANAGARE, S.; AGARWALA, M. Evaluating methods to map burned area at 30-meter resolution in forests and agricultural areas of Central India. **Frontiers in Forests and Global Change**, v. 5, 2022. 20, 21

COELHO, M. H.; BITTENCOURT; OLIVEIRA, O.; MORELLI; FABIANO; SANTOS, R. Método para a classificação de áreas queimadas baseado em

aprendizado de máquina automatizado. **Computer on the Beach**, v. 13, p. 029–036, 2022. 2

COFFIELD, S. R.; GRAFE, C. A.; CHEN, Y.; SMYTH, P.; FOUFOULA, G. E.; RANDERSON, J. T. Machine learning to predict final fire size at the time of ignition. **International Journal of Wildland Fire**, v. 28, n. 11, p. 861–873, 2019. 2, 17, 31

COUGHLAN, R.; GIUSEPPE, F.; VITOLO, C.; BARNARD, C.; LOPEZ, P.; DRUSCH, M. Using machine learning to predict fire-ignition occurrences from lightning forecasts. **Meteorological Applications**, v. 28, n. 1, 2021. 21

COVER, T.; HART, P. Nearest neighbor pattern classification. **IEEE Transactions on Information Theory**, v. 13, n. 1, p. 21–27, 1967. 29

ESCUIN, S.; NAVARRO, R.; FERNANDEZ, P. Fire severity assessment by using NBR (Normalized Burn Ratio) and NDVI (Normalized Difference Vegetation Index) derived from Landsat TM/ETM images. **International Journal of Remote Sensing**, v. 29, n. 4, p. 1053–1073, 2008. 13, 14

FARASIN, A.; COLOMBAA, L.; GARZA, P. Double-step u-net: a deep learning-based approach for the estimation of wildfire damage severity through Sentinel-2 satellite data. **Applied Sciences**, v. 10, n. 12, p. 4332, 2020. 2

FEURER, M.; KLEIN, A.; EGGENSPERGER, K.; SPRINGENBERG, J.; BLUM, M.; HUTTER, F. Efficient and robust automated machine learning. **Advances in Neural Information Processing Systems**, v. 28, 2015. 27

HE, X.; ZHAO, K.; CHU, X. Automl: a survey of the state-of-the-art. **Knowledge-Based Systems**, v. 212, 2021. 27

HU, X.; BAN, Y.; NASCETTI, A. Uni–temporal multispectral imagery for burned area mapping with deep learning. **Remote Sensing**, v. 13, n. 8, p. 1509, 2021. 29

HUANG, Y.; JIN, Y.; SCHWARTZ, M.; THORNE, J. Intensified burn severity in California's northern coastal mountains by drier climatic condition. **Environmental Research Letters**, v. 15, 10 2020. 17

INSTITUTO NACIONAL DE PESQUIOSAS ESPACIAIS (INPE). **Portal do monitoramento de queimadas e incêndios**. São José dos Campos: INPE, 2023. 1

JAMAL, I.; IQBAL, N.; AHMAD, S.; KIM, D. H. Towards mountain fire safety using fire spread predictive analytics and mountain fire containment in IoT environment. **Sustainability**, v. 13, n. 5, 2021. 2, 17

JENSEN, J. R. **Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres**. [S.l.]: Parêntese, 2009. 1

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing systematic literature reviews in software engineering**. Durham: University of Durham, 2007. 5

KLINK, C. A.; MACHADO, R. B. A conservação do cerrado brasileiro. **Megadiversidade**, v. 1, n. 1, p. 147–155, 2005. 38

KLOMPENBURG, T.; KASSAHUN, A.; CATAL, C. Crop yield prediction using machine learning: a systematic literature review. **Computers and Electronics in Agriculture**, v. 177, oct 2020. 5

LESTARI, A. I.; LUHURKINANTI, D. L.; FITRIASARI, H. I.; HARWAHYU, R.; SARI, R. F. Machine learning approaches for burned area identification using Sentinel-2 in Central Kalimantan. **Journal of Applied Engineering Science**, v. 18, n. 2, p. 207–215, 2020. 14, 16

MITHAL, V.; NAYAK, G.; KHANDELWAL, A.; KUMAR, V.; NEMANI, R.; OZA, N. C. Mapping burned areas in Tropical Forests using a novel machine learning framework. **Remote Sensing**, v. 10, n. 1, p. 69, 2018. 19

MONARD, M. C.; BARANAUSKAS, A.; REZENDE, J. A. Conceitos sobre aprendizado de máquina. In: **REZENDE, S. O. (Ed.)**. Barueri-SP: Manole, 2003. p. 89–114. 25

MUHAMAD, L. J.; ALGEHYNE, E. A.; USMAN, S. S.; AHMAD, A.; CHAKRABORTY, C.; MOHAMMED, I. A. Supervised machine learning models for prediction of COVID-19 infection using epidemiology dataset. **SN Computer Science**, v. 2, n. 11, 2020. 24, 26

NEGRI, R. G.; LUZ, A. E. O.; FRERY, A. C.; CASACA, W. Mapping burned areas with multitemporalndash;multispectral data and probabilistic unsupervised learning. **Remote Sensing**, v. 14, n. 21, 2022. ISSN 2072-4292. 20

OLSON, R. S.; MOORE, J. H. Tpot: a tree-based pipeline optimization tool for automating machine learning. In: **WORKSHOP ON AUTOMATIC MACHINE LEARNING**. [S.l.: s.n.], 2016. p. 66–74. 2, 27, 28

OLSON, R. S.; URBANOWICZ, R. J.; ANDREWS, P. C.; LAVENDER, N. A.; KIDD, L. C.; MOORE, J. H. Automating biomedical data science through tree-based pipeline optimization. In: **SQUILLERO, G.; BURELLI, P.** Porto: Springer, 2016. p. 123–137. 24, 27

PEREIRA, A. A.; CARVALHO, L. M. T.; LIBONATI, R.; A., F. J. W.; MORELLI, F. Avaliação de nove índices espectrais quanto a separabilidade entre queimadas e diferentes alvos. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO. **Anais...** [S.l.], 2015. v. 17, p. 3105–3112. 15

PEREIRA, A. A.; LOBONATI, R.; RODRIGUES, J. A.; NOGUEIRA, J.; SANTOS, F. L. M.; O., D.; SANCHES, W.; ALVARADO, S. T.; PEREIRA, J. Multi-sensor, active fire-supervised, one-class burned area mapping in the Brazilian Savanna. **Remote Sensing**, v. 13, n. 19, p. 4005, 2021. 2, 14

RODRIGUES, J. A.; LIBONATI, R.; P., F. L.; A., S. Mapeamento de áreas queimadas em unidades de conservação da região serrana do Rio de Janeiro utilizando o satélite Landsat-8 durante a seca de 2014. **Anuário do Instituto de Geociências**, v. 41, n. 1, p. 318–327, 2018. 1

RUSSELL, S. J. **Artificial intelligence a modern approach**. [S.l.]: Pearson Education, 2010. 35

SAFAVIAN, S. R.; LANDGREBE, D. A survey of decision tree classifier methodology. **IEEE Transactions on Systems, Man and Cybernetics**, v. 21, n. 3, p. 660–674, 1991. 30

SÁNCHEZ, M. B.; TONINI, M.; MAPELLI, A.; FIORUCCI, P. Spatial assessment of wildfires susceptibility in Santa Cruz (Bolivia) using random forest. **Geosciences**, v. 11, n. 5, p. 224, 2021. 15, 16, 19, 61

SANTIAGO, L. A. N.; LOPES, R. S. Impacts on human health due to the emission of aerosols caused by burns. **Brazilian Journal of Development**, v. 7, n. 1, p. 9069–9075, jan 2021. 1

SANTOS, S. M. B.; DUVERGER, S. G.; GONÇALVES, A. B.; ROCHA, W. F.; A., V.; G., T. Remote sensing applications for mapping large wildfires based on machine learning and time series in Northwestern Portugal. **Fire**, v. 6, n. 2, p. 69, 2023. 20, 21

SANTOS, T. O.; ANDRADE FILHO, V. S.; ROCHA, V. M.; MENEZES, J. M. Os impactos do desmatamento e queimadas de origem antrópica sobre o clima da

Amazônia brasileira: um estudo de revisão. **Revista Geográfica Acadêmica**, v. 11, n. 2, p. 157–181, jan 2017. 1

SHARMA, L. K.; GUPTA, R.; NAUREEN, F. Assessing the predictive efficacy of six machine learning algorithms for the susceptibility of indian forests to fire. **International Journal of Wildland Fire**, v. 31, n. 8, p. 735–758, 2022. 29

SILVA, J.; A., J.; PENHA, A. P. Avaliação de índices espectrais e classificação normal bayes usando imagens OLI e TIRS para o mapeamento de áreas queimadas no Cerrado. **Revista Brasileira de Meio Ambiente**, v. 10, n. 3, 2022. 1

SOUSA, A. M. O.; PEREIRA, J. M. C.; SILVA, J. M. N. Evaluating the performance of multitemporal image compositing algorithms for burned area analysis. **International Journal of Remote Sensing**, v. 24, n. 6, p. 1219–1236, 2003. 50

SRA, S.; NOWOZIN, S.; WRIGHT, S. J. **Optimization for machine learning**. [S.l.]: Mit Press, 2012. 32

TRAPPENBERG, T. P. **Fundamentals of machine learning**. [S.l.]: Oxford University Press, 2020. 23, 32

VANDERHOOF, M. K.; HAEBAKER, T. J.; TESKE, C.; KU, A.; NOBLE, J.; PICOTTE, J. Mapping wetland burned area from Sentinel-2 across the southeastern United States and its contributions relative to Landsat-8 (2016–2019). **Fire**, v. 4, n. 3, p. 52, 2021. 15

VERAVERBEKE, S.; HARRIS, S.; HOOK, S. Evaluating spectral indices for burned area discrimination using MODIS/ASTER (MASTER) airborne simulator data. **Remote Sensing of Environment**, v. 115, n. 10, p. 2702–2709, 2011. 14

WANG, S.; WANG, Y. Quantifying the effects of environmental factors on wildfire burned area in the south central us using integrated machine learning techniques. **Atmospheric Chemistry and Physics**, v. 20, n. 18, p. 11065–11087, 2020. 21, 61, 71

WANG, S. C.; QIAN, Y.; LEUNG, L. R.; ZHANG, Y. Identifying key drivers of wildfires in the contiguous US using machine learning and game theory interpretation. **Earth's Future**, v. 9, n. 6, p. 1–21, 2021. 2, 17, 20, 21, 71

YU, Y.; MAO, J.; THORNTON, P. E.; NOTARO, M.; WULLSCHLEGER, S. D.; SHI X.AND HOFFMAN, F. M.; WANG, Y. Quantifying the drivers and

predictability of seasonal changes in African fire. **Nature Communications**, v. 11, n. 1, p. 1–8, 2020. 21

ZAGLIA, M. C. **Catalogação de imagens de satélites de sensoriamento remoto voltada para acesso a cubos de dados de observação da Terra**. Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2020. 1

ZHOU, Z. H. **Machine learning**. [S.l.]: Springer Nature, 2021. 23

APÊNDICE A - LISTA DE PUBLICAÇÕES CIENTÍFICAS

A Tabela A.1 apresenta informações sobre as publicações científicas usadas na RSL.

Tabela A.1 - Publicações científicas usadas na Revisão Sistemática da Literatura.

ID	Ano	Título do artigo	Base de dados	DOI
1	2015	Predicting burned areas of forest fires: an artificial intelligence approach	Scopus	10.4996/fireecology.1101106
2	2017	A data–driven approach to identify controls on global fire activity from satellite and climate observations (SOFIA V1)	Ambas as bases	10.5194/gmd-10-4443-2017
3	2018	Mapping burned areas in Tropical Forests using a novel machine learning framework	Ambas as bases	10.3390/rs10010069
4	2019	Machine learning to predict final fire size at the time of ignition	Ambas as bases	10.1071/WF19023
5	2019	Burned area detection and mapping using Sentinel–1 backscatter coefficient and thermal anomalies	Ambas as bases	10.1016/j.rse.2019.111345
6	2020	Quantifying the drivers and predictability of seasonal changes in Africa fire	*Web of Science*	10.1038/s41467-020-16692-w
7	2020	Observed changes in fire patterns and possible drivers over Central Africa	*Web of Science*	10.1088/1748-9326/ab9db2
8	2020	Machine learning approaches for burned area identification using Sentinel–2 in Central Kalimantan	Scopus	10.5937/jaes18-25495
9	2020	Comparison of treebased classification algorithms in mapping burned forest areas	Scopus	10.15292/ geodetski-vestnik.2020.03.348-360
10	2020	Evaluation of prescribed fires from Unmanned Aerial Vehicles (UAVs) imagery and machine learning algorithms	Scopus	10.3390/rs12081295

(Continua)

Tabela A.1 - Publicações científicas usadas na Revisão Sistemática da Literatura.

ID	Ano	Título do artigo	Base de dados	DOI
11	2020	Locating forest management units using remote sensing and geostatistical tools in North–Central Washington USA	Ambas as bases	10.3390/s20092454
12	2020	Double-Step U-Net: A deep learning based approach for the estimation of wildfire damage severity through Sentinel–2 satellite data	Ambas as bases	10.3390/app10124332
13	2020	The Landsat burned area algorithm and products for the conterminous United States	Ambas as bases	10.1016/j.rse.2020.111801
14	2020	Quantifying the effects of environmental factors on wildfire burned area in the south central US using integrated machine learning techniques	Ambas as bases	10.5194/acp-20-11065-2020
15	2020	Intensified burn severity in California's Northern Coastal mountains by drier climatic condition	Scopus	10.1088/1748-9326/aba6af
16	2020	Open data and machine learning to model the occurrence of fire in the ecoregion of “Llanos Colombo—Venezolanos"	Scopus	10.3390/rs12233921
17	2021	Machine learning estimation of fire arrival time from level–2 active fires satellite data	*Web of Science*	10.3390/rs13112203
18	2021	Using machine learning to predict fire ignition occurrences from lightning forecasts using random forest	Scopus	10.1002/met.1973

(Continua)

Tabela A.1 - Publicações científicas usadas na Revisão Sistemática da Literatura.

ID	Ano	Título do artigo	Base de dados	DOI
19	2021	Spatial assessment of wildfires susceptibility in Santa Cruz (Bolivia) using random forest	Ambas as bases	10.3390/geosciences11050224
20	2021	Towards mountain fire safety using fire spread predictive analytics and mountain fire containment in IoT environment	Ambas as bases	10.3390/su13052461
21	2021	Uni–temporal multispectral imagery for burned area mapping with deep learning	Ambas as bases	10.3390/rs13081509
22	2021	Machine learning methods and synthetic data generation to predict large wildfires	Ambas as bases	10.3390/s21113694
23	2021	Identifying key drivers of wildfires in the contiguous US using machine learning and game theory interpretation	Ambas as bases	10.1029/2020EF001910
24	2021	Patterns of mega–forest fires in east Siberia will become less predictable with climate warming	Scopus	10.1016/j.envadv.2021.100041
25	2021	A fully automatic, interpretable and adaptive machine learning approach to map burned area from remote sensing	Ambas as bases	10.3390/ijgi10080546
26	2021	Mapping wetland burned area from Sentinel–2 across the Southeastern United States and its contributions relative to Landsat–8 (2016–2019)	Ambas as bases	10.3390/fire4030052

(Continua)

Tabela A.1 - Publicações científicas usadas na Revisão Sistemática da Literatura.

ID	Ano	Título do artigo	Base de dados	DOI
27	2021	Multi–sensor, active Fire–supervised, One–class burned area mapping in the Brazilian Savanna	Ambas as bases	10.3390/rs13194005
28	2022	Mapping burned areas with multitemporal–multispectral data and probabilistic unsupervised learning	Scopus	10.3390/rs14215413
29	2022	Burned area classification based on rxtreme learning machine and Sentinel–2 Images	Ambas as bases	10.3390/app12010009
30	2022	Regional–scale burned area mapping in Mediterranean regions based on the multitemporal composite integration of Sentinel–1 and Sentinel–2 data	Ambas as bases	10.1080/15481603.2022.2128251
31	2022	Wildfire risk assessment in Liangshan Prefecture – China based on an integration machine learning algorithm	Ambas as bases	10.3390/rs14184592
32	2022	Evaluating methods to map burned area at 30–meter resolution in forests and agricultural areas of Central India	Ambas as bases	10.3389/ffgc.2022.933807
33	2022	Supervised machine learning approaches on multispectral remote sensing data for a combined detection of fire and burned area	Ambas as bases	10.3390/rs14030657
34	2022	Modern Pyromes: biogeographical patterns of fire characteristics across the Contiguous United States	Ambas as bases	10.3390/fire5040095
35	2022	Assessing the predictive efficacy of six machine learning algorithms for the susceptibility of Indian forests to fire	*Web of Science*	10.1071/WF22016

(Continua)

Tabela A.1 - Publicações científicas usadas na Revisão Sistemática da Literatura.

ID	Ano	Título do artigo	Base de dados	DOI
36	2022	Forest fire occurrence prediction in China based on machine learning methods	*Web of Science*	10.3390/rs14215546

Fonte: Elaborada pela autora.

APÊNDICE B - ALGORITMOS RESULTANTES DAS EXECUÇÕES DO AUTOML TPOT

```
import numpy as np
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline
from tpot.export_utils import set_param_recursive

# NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.9666882836143535
exported_pipeline = make_pipeline(
  PCA(iterated_power=3, svd_solver="randomized"),
  KNeighborsClassifier(n_neighbors=20, p=1, weights="uniform"))

# Fix random state for all the steps in exported pipeline
set_param_recursive(exported_pipeline.steps, 'random_state', 101)
exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 1: Código gerado pelo AutoML TPOT para o Experimento 1.

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline
from sklearn.tree import DecisionTreeClassifier
from tpot.builtins import StackingEstimator
from tpot.export_utils import set_param_recursive

# NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.9447213347532838
exported_pipeline = make_pipeline(
  StackingEstimator(estimator=MultinomialNB(alpha=100.0,
 fit_prior=False)),
  DecisionTreeClassifier(criterion="entropy", max_depth=10,
 min_samples_leaf=14, min_samples_split=10))

# Fix random state for all the steps in exported pipeline
set_param_recursive(exported_pipeline.steps, 'random_state', 101)
exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 2: Código gerado pelo AutoML TPOT para o Experimento 2.

```
import numpy as np
import pandas as pd
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.feature_selection import RFE
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline
from tpot.export_utils import set_param_recursive

# NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.9703042328042327
exported_pipeline = make_pipeline(
  RFE(estimator=ExtraTreesClassifier(criterion="gini",
 max_features=0.6000000000000001, n_estimators=100), step=0.05),
  KNeighborsClassifier(n_neighbors=20, p=1, weights="uniform"))

# Fix random state for all the steps in exported pipeline
set_param_recursive(exported_pipeline.steps, 'random_state', 101)
exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 3: Código gerado pelo AutoML TPOT para o Experimento 3.

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

#NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.9331225680933851
exported_pipeline = KNeighborsClassifier(n_neighbors=99, p=1,
 weights="uniform")

# Fix random state in exported estimator
if hasattr(exported_pipeline, 'random_state'):
    setattr(exported_pipeline, 'random_state', 101)

exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 4: Código para o Experimento 4.

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline
from sklearn.tree import DecisionTreeClassifier
from tpot.builtins import StackingEstimator
from tpot.export_utils import set_param_recursive

# NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.9073384539549207
exported_pipeline = make_pipeline(
 StackingEstimator(estimator=DecisionTreeClassifier(criterion="gini",
 max_depth=6, min_samples_leaf=13, min_samples_split=10)),
   KNeighborsClassifier(n_neighbors=50, p=2, weights="distance"))

# Fix random state for all the steps in exported pipeline
set_param_recursive(exported_pipeline.steps, 'random_state', 101)
exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 5: Código gerado pelo AutoML TPOT para o Experimento 5.

```
import numpy as np
import pandas as pd
from sklearn.linear_model import SGDClassifier
from sklearn.model_selection import train_test_split
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import MinMaxScaler
from tpot.export_utils import set_param_recursive

# NOTE: Make sure that the outcome column is labeled 'target' in
 the data file
tpot_data = pd.read_csv('PATH/TO/DATA/FILE',
 sep='COLUMN_SEPARATOR', dtype=np.float64)
features = tpot_data.drop('target', axis=1)
training_features, testing_features, training_target,
 testing_target =
train_test_split(features, tpot_data['target'], random_state=101)

# Average CV score on the training set was: 0.926314907468709
exported_pipeline = make_pipeline(
   MinMaxScaler(),
   SGDClassifier(alpha=0.0, eta0=0.01, fit_intercept=True,
 l1_ratio=0.25, learning_rate="constant", loss="hinge",
 penalty="elasticnet", power_t=100.0))

# Fix random state for all the steps in exported pipeline
set_param_recursive(exported_pipeline.steps, 'random_state', 101)
exported_pipeline.fit(training_features, training_target)
results = exported_pipeline.predict(testing_features)
```

Algoritmo 6: Código gerado pelo AutoML TPOT para o Experimento 6.

APÊNDICE C - VISÃO GERAL DOS MODELOS E SUAS MÉTRICAS DE DESEMPENHO

A Tabela C.1 apresenta informações sobre cada um dos experimentos realizados nesta dissertação.

Tabela C.1 - Tabela de desempenho dos modelos de aprendizado de máquina.

Experimento	Modelo	Média das métricas de desempenho		
		Precisão (%)	*Recall* (%)	*F1-score* (%)
1	KNN	94,0	94,5	94,0
2	*Decision Tree*	89,5	90,5	90,0
3	KNN	94,0	91,5	93,0
4	KNN	90,0	90,5	90,0
5	KNN	90,0	88,0	89,5
6	SVM	91,5	90,0	90,5