UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Paulo Júnior de Moraes Vasconcelos

Identificação de fungos anemófilos, em ambientes abertos, através de um nariz eletrônico e modelos de Inteligência Artificial

Recife

2022

Paulo Júnior de Moraes Vasconcelos

Identificação de fungos anemófilos, em ambientes abertos, através de um nariz eletrônico e modelos de Inteligência Artificial

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Área de Concentração: Inteligência Computacional

Orientador: Leandro Maciel Almeida

Recife

2022

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

V331i Vasconcelos, Paulo Júnior de Moraes
Identificação de fungos anemófilos, em ambientes abertos, através de um nariz eletrônico e modelos de inteligência artificial / Paulo Júnior de Moraes Vasconcelos. – 2022.
134 f.: il., fig., tab.

Orientador: Leandro Maciel Almeida.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2022.

Inclui referências.

1. Inteligência computacional. 2. Séries temporais. I. Almeida, Leandro Maciel (orientador). II. Título.

006.31 CDD (23. ed.) UFPE - CCEN 2022-162

Paulo Júnior de Moraes Vasconcelos

"**Identificação de fungos anemófilos, em ambientes abertos, através de um nariz eletrônico e modelos de Inteligência Artificial**"

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 22/06/2022.

Orientador: Leandro Maciel Almeida

BANCA EXAMINADORA

Prof. Dr. Paulo Salgado Gomes de Mattos Neto
Centro de Informática / UFPE

Prof. Dr. Reginaldo Gonçalves de Lima Neto
Departamento de Medicina Tropical / UFPE

Prof. Dr. Cleber Zanchettin
Centro de Informática / UFPE

Decido este trabalho à minha família e à minha esposa que, nesse período de muitos desafios e adversidades, foram o meu porto seguro.

AGRADECIMENTOS

Agradeço aos meus estimados amigos-irmãos pelos incentivos e apoios para me dedicar a essa fascinante e desafiadora guinada em minha carreira profissional. Em especial, gostaria de nominar Margarida Antunes, por me fazer acreditar nesse sonho e transformá-lo em realidade. Também gostaria de destacar a pessoa de Alexandre Falcão que, além de me incentivar e me apoiar, das mais variadas formas, com toda a sua vasta experiência e sabedoria, culminou por exercer uma função informal de coorientação nesta minha empreitada acadêmica.

RESUMO

Os fungos se dispersam na natureza através do ar atmosférico ou por outras vias, tais como água, insetos, humanos e animais. Os fungos que se dispersam pelo ar atmosférico são chamados de fungos anemófilos. Em indústrias, como a farmacêutica e a de alimentos, a preservação da qualidade do ar nos ambientes é um ponto importante para a garantia asséptica dos produtos. Os hospitais são ambientes que precisam de mais atenção em termos de monitoramento ambiental de áreas críticas. As questões de segurança alimentar em toda a cadeia global de fornecimento de alimentos tornaram-se fundamentais para promover a segurança da saúde pública e o sucesso comercial da indústria alimentícia global. Tecnologias e ferramentas de diagnóstico rápido e não invasivo, baseadas em *volatomics*, através de narizes eletrônicos, são muito promissoras. De forma sucinta, o trabalho dessa dissertação é classificar corretamente séries temporais. Os *Volatile Organic Compounds* (*VOC*)s emitidos pelas colônias de fungos anemófilos são lidos por um nariz eletrônico que os identifica e os armazena na forma de séries temporais. Essa pesquisa apresenta um estudo sobre os fungos anemófilos, explica o que são os *VOC*s, efetua uma breve introdução sobre séries temporais, evidencia e e debate a respeito do nariz eletrônico utilizado nos experimentos, explana sobre os motivos que levaram a escolher os modelos de IA para resolver o problema, analisa o funcionamento dos modelos de IA selecionados além de apresentar e explorar os dados evidenciados na pesquisa. Foram montados dois conjuntos de dados, o **"Placa"** que reúne os dados coletados das placas de textitPetri com colônias de fungos anemófilos e o **"Aberto"** que compila dados do ar de ambiente com a dispersão das colônias de fungos anemófilos em placas de textitPetri abertas. O classificador *MrSEQL* teve o melhor desempenho na base de dados **"Placa"**, alcançando uma precisão de 94,5% no conjunto de testes. O classificador *Arsenal* obteve os melhores resultados na base **("Placa+Aberto")**, obtendo uma precisão de 94,9% no conjunto de teste. No último experimento realizado, o treinamento foi realizado em **"Placa"** e o teste em **"Aberto"**. Os resultados foram insatisfatórios, com precisão máxima de apenas 58,8% no conjunto de teste. O comportamento dos fungos anemófilos é influenciado pelo ambiente (poluído, aberto, em mata, fechado, estéril etc.), temperatura, umidade, pH, entre outros. Diante disto, acredita-se estar justificado que os *VOC*s emitidos pelas colônias anemófilas em uma placa são diferentes dos emitidos pelas colônias anemófilas para o ambiente aberto. Como trabalhos futuros tem-se a expansão e a diversificação dos testes de fungos anemófilos visando cobrir uma gama mais ampla de problemas e patologias associadas a mais espécies de fungos anemófilos, bem como a expansão das bases de dados. No estudo de modelos de classificadores especializados para séries temporais, o objetivo é encontrar o modelo com os resultados mais satisfatórios e os menores custos computacionais.

Palavras-chaves: fungos anemófilos; nariz eletrônico; classificação de séries temporais.

ABSTRACT

Fungi disperse in nature through atmospheric air or by other routes, such as water, insects, humans, and animals. Fungi that disperse through atmospheric air are called anemophilous fungi. In industries, such as the pharmaceutical and food industries, preservation of the air quality in the environments is an important point for the aseptic assurance of the products. Hospitals are environments that need more attention in terms of environmental monitoring of critical areas. Food safety issues throughout the global food supply chain have become critical to promoting public health safety and the commercial success of the global food industry. Rapid, non-invasive diagnostic technologies and tools based on *volatomics* through electronic noses hold great promise. Briefly, the work of this dissertation is to correctly classify time series. The *VOC* emitted by colonies of anemophilous fungi are read by an electronic nose that identifies and stores them in the form of time series. This research presents a study about anemophilous fungi, explains what the *VOC*s are, makes a brief introduction about time series, evidences and debates about the electronic nose used in the experiments, explains about the reasons that led to choose the AI models to solve the problem, analyzes the operation of the selected AI models and presents and explores the data evidenced in the research. Two datasets were assembled, the **"Plate"** which compiles data collected from textitPetri plates with anemophilous fungi colonies and the **"Open"** which compiles ambient air data with the dispersion of anemophilous fungi colonies on open textitPetri plates. The *MrSEQL* classifier performed best on the **"Plate"** database, achieving an accuracy of 94.5% on the test set. The *Arsenal* classifier performed best on the **("Plate+Open")** database, achieving an accuracy of 94.9% on the test set. In the last experiment performed, training was performed on **"Plate"** and testing was performed on **"Open"**. The results were unsatisfactory, with maximum accuracy of only 58.8% on the test set. The behavior of anemophilous fungi is influenced by the environment (polluted, open, in the woods, closed, sterile, etc.), temperature, humidity, pH, among others. Given this, it is believed to be justified that the *VOC*s emitted by anemophilous colonies on a plate are different from those emitted by anemophilous colonies in the open environment. Future work includes the expansion and diversification of anemophilic fungal tests to cover a wider range of problems and pathologies associated with more anemophilic fungal species, as well as the expansion of databases. In the study of expert classifier models for time series, the goal is to find the model with the most satisfactory results and the lowest computational costs.

Keywords: airborne fungi; electronic nose; classification of time series.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACF	*Auto Correlation Function*
AM	Aprendizagem de Máquina
Arsenal	*Arsenal ensemble*
BOSS	*Bag of Symbolic Fourier Approximation Symbols*
CAWPE	*Cross-validation Accuracy Weighted Probabilistic Ensemble*
cBOSS	*Contractable Bag of Symbolic Fourier Approximation Symbols*
CNN	*Convolutional Neural Network*
ddNTPs	Dideoxinucleotídeos
DNA	Ácido Desoxirribonucleico
DrCIF	*Diverse Representation Canonical Interval Forest*
DFT	*Discret Fourier Transform*
EU	*European Union*
Flat-CODE	*Flat Collective of Transformation-based Ensembles*
FN	Falso Negativo
FP	Falso Positivo
GAP	*Global Average Pooling*
GDPR	*General Data Protection Regulation*
GPU	*Graphics Processing Unit*
HIVE-COTE V1	*Hierarchical Vote Collective of Transformation-based Ensembles V1*
HIVE-COTE V2	*Hierarchical Vote Collective of Transformation-based Ensembles V2*
IA	Inteligência Artificial
IoT	*Internet of Things*
kNN	*K-Nearest Neighbors Algorithm*
kNN-TSC	*kNN-Based Time-Series Classification*
MALDI-TOF MS	*Matrix-assisted laser desorption ionization-time of flight mass spectrometry*
MCB	*Multiple Coeficiente Binning*
ML	*Machine Learning*

Mr-SEQL	*Time Series Classification with multiple symbolic representations and SEQL*
MTS	*Multivariate Time Series*
MVOC	*Microbial Volatile Organic Compounds*
NGS	*Next Generation Sequencing*
NNDTW	*One-neares t-neighbor with dynamic time warping*
PAA	*Piecewise Aggregate Approximation*
PCR	*Polymerase Chain Reaction*
ppv	*proportion of positive values*
PS	*Power Spectrum*
PTFE	Politetrafluoretileno, *Teflon*
RF	*Randon Forest*
RF	*Receptive Field*
RISE	*Random Interval Spectral Forest*
ROCKET	*Random Convolutional Kernel Transform*
SAX	*Symbolic Aggregate approXimation*
SEQL	*SEQuence Learner*
SFA	*Symbolic Fourier Approximation*
ST-HESCA	*Shapelet-Transformed Heterogeneous Ensemble of Standard Classification Algorithms*
TDE	*Temporal Dictionary Ensemble*
TSC	*Time Series Classification*
TSF	*Time Series Forest*
UMAP	*Uniform Manifold Approximation and Projection*
VN	Verdadeiro Negativo
VOC	*Volatile Organic Compounds*
VP	Verdadeiro Positivo
WEASEL	*Word Extraction for Time Series Classification*

SUMÁRIO

1 INTRODUÇÃO

Segundo (MEZZARI et al., 2003), os fungos se dispersam na natureza através do ar atmosférico ou por outras vias, como água, insetos, humanos e animais. Os fungos que são dispersados através do ar atmosférico são denominados fungos anemófilos. Sendo assim, a microbiota fúngica anemófila pode ser semelhante ou diferente em cada cidade ou região. Os elementos fúngicos que são encontrados no ar atmosférico são os esporos (propágulos). São aeroalérgenos que, quando inalados, podem ser responsáveis por manifestações respiratórias alérgicas, como a asma e a rinite.

Segundo (MAGESTE et al., 2012), os fungos são microrganismos ubíquos (que estão ou existem ao mesmo tempo em toda parte; onipresentes), encontrados amplamente disseminados no ar, no solo e em ambientes aquáticos. São seres vivos absortivos com organização celular e com Ácido Desoxirribonucleico (DNA) delimitado por um envoltório nuclear. Os fungos que vivem no ar atmosférico são denominados anemófilos, sendo esse habitat o meio de dispersão mais utilizado e bem- sucedido desses microrganismos. Assim, dificilmente pode existir ambiente livre de contaminação fúngica, pois tais organismos têm o ar atmosférico como seu principal meio de dispersão e suportam grandes variações de temperatura, de umidade e de pH. Sendo assim, são facilmente encontrados em ambientes internos.

1.1 MOTIVAÇÃO

Na indústria farmacêutica, entre outras, como exemplo, a de alimentos, a preservação da qualidade do ar dos ambientes é um ponto importante para a garantia asséptica dos medicamentos ou alimentos que essas indústrias produzem. A contaminação ambiental pode trazer várias consequências como perda de produtos, afastamento de pessoal, dentre outros prejuízos, especialmente no que concerne ao consumo, por pessoas imunologicamente comprometidas, de medicamentos e/ou alimentos contaminados. As consequências podem ser danosas à saúde, tais quais: infecções, internamentos hospitalares, entre outros. Da mesma forma, pessoas que trabalham nestas indústrias, uma vez que estes medicamentos ou alimentos são produzidos em ambientes fechados. Portanto, a higiene industrial e um monitoramento microbiológico do ar asseguram boas condições higiênico-sanitárias, evitando danos à saúde dos consumidores de seus medicamentos e alimentos e e dos funcionários envolvidos na produção destes insumos (MAGESTE et al., 2012).

As questões de segurança alimentar em toda a cadeia global de fornecimento de alimentos se tornaram primordiais na promoção da segurança da saúde pública e do sucesso comercial da indústria alimentar global. À medida que as regulamentações alimentares e as expectativas dos consumidores continuam avançando em todo o mundo, apesar da tec-

nologia mais recente, ferramentas de detecção, regulamentações e educação do consumidor sobre segurança e qualidade dos alimentos, ainda há um aumento de surtos de doenças transmitidas por alimentos estragados. O desenvolvimento de narizes eletrônicos, como técnica não invasiva adequada para detecção de *VOC*s, tem sido aplicado para a análise de qualidade e segurança alimentar. A aplicação de *e-Nose* para detecção de patógenos tem sido bem-sucedida e superior aos métodos convencionais. O *e-Nose* oferece um método não invasivo, rápido e requer pouca ou nenhuma preparação da amostra, tornando-o ideal para uso como ferramenta de monitoramento *on-line* (BONAH et al., 2020).

As tecnologias e ferramentas de diagnóstico não invasivas rápidas baseadas em *volatolomics*, são uma grande promessa no controle de doenças infecciosas. No entanto, as ferramentas para identificar compostos orgânicos voláteis microbianos *VOC*s, discriminando-os dos patógenos humanos, ainda carecem de aprimoramento. A inteligência artificial é cada vez mais reconhecida como uma ferramenta essencial nas ciências da saúde. Algoritmos de IA foram aplicados para encontrar conjuntos de *VOC*s microbianos com poder de discriminação de patógenos (PALMA et al., 2018).

Nas últimas décadas, a importância dos bioaerossóis tem sido enfatizada, pois podem prejudicar a saúde das pessoas, levando ao aparecimento de patologias que vão de alergias a infecções disseminadas em pacientes suscetíveis. A determinação da composição e da concentração de microrganismos anemófilos de áreas internas e/ ou externas, em áreas críticas de hospitais tem sido enfatizada como extremamente necessária. Hospitais constituem ambientes que necessitam de maior atenção, no que diz respeito ao monitoramento ambiental das áreas críticas. A finalidade dessa ação é identificar possíveis fontes de contaminação/ disseminação de microorganismos e os possíveis agentes patogênicos envolvidos.

Por outro lado, em ambientes climatizados, o acúmulo de umidade e material orgânico em bandejas de ar-condicionado pode torná-las poderosas fontes dispersoras de bioaerossóis (MARTINS-DINIZ et al., 2005).

Conforme já citado, os elementos fúngicos que são encontrados no ar atmosférico são os esporos (propágulos). No que diz respeito ao ciclo de vida, os fungos são gerados de forma sexual ou assexual, o que muito ajuda no momento de isolar e identificar as espécies fúngicas. Por oportunismo, a partir da dispersão desses esporos por meio do vento, os fungos anemófilos acabam provocando patologias nos seres humanos. Tais fungos são encontrados frequentemente como componentes da microbiota transitória do homem e animais domésticos, como contaminantes de alimentos, deteriorantes de acervos, madeiras, em água doce e salgada e são responsáveis pela contaminação de diversos materiais (SOUZA; ANDRADE; LIMA, 2013).

Pesquisa desenvolvida por (ARROYO et al., 2020) apresenta um nariz eletrônico pessoal miniaturizado (39 mm $\times$ 33 mm), que é gerenciado por meio de um aplicativo desenvolvido para smartphone. O nariz eletrônico (*e-nose*) incorpora quatro sensores digitais de

gás de última geração. Esses sensores do tipo MOx incorporam um microcontrolador no mesmo pacote. Isso torna mais fácil integrá-los à eletrônica e melhora seu desempenho. Nesta pesquisa, a aplicação do dispositivo está focada na detecção de poluentes atmosféricos de forma a complementar as informações fornecidas pelas estações de referência. Um aplicativo móvel foi desenvolvido para fornecer serviços de classificação. Uma rede neural foi desenvolvida, treinada e integrada ao smartphone para processar as informações recuperadas do dispositivo *e-nose* (ARROYO et al., 2020).

É importante salientar o ineditismo do trabalho e a relevância para várias áreas que poderão ser beneficiadas com um método de detecção rápida, precisa e barata de fungos anemófilos potencialmente patógenos em comparação com os métodos existentes no mercado. Entre as áreas beneficiadas podem-se citar: portadores de manifestações respiratórias alérgicas, como asma e rinite; industrias que focam na preservação e controle da qualidade do ar, como a farmacêutica e a alimentícia; além de hospitais, uma vez que constituem ambientes que necessitam de maior atenção, no que diz respeito ao monitoramento ambiental das áreas críticas, manutenção de condicionadores de ar e sistemas de ar-condicionado.

1.2 DEFINIÇÃO DOS OBJETIVOS

Nesta seção, definem-se o objetivo da dissertação e os objetivos específicos.

1.2.1 Objetivo da dissertação

O objeto da dissertação é identificação de fungos anemófilos, através de um nariz eletrônico e modelos de Inteligência Artificial (IA), a partir da análise de culturas em placas de *Petri* (recipiente cilíndrico, achatado que pode ser de vidro ou plástico amplamente utilizado para cultura de microrganismos, possui duas partes uma base e uma tampa) e análise do ar do ambiente com a dispersão de *VOC*s e esporos desses fungos. A dispersão de *VOC*s e esporos deixa uma cultura aberta para o ambiente ao longo do tempo, coletam-se amostras de ar do ambiente para então verificarverificar se consegue identificação a presença dos fungos.

1.2.2 Objetivos específicos

Os objetivos específicos do trabalho são:

- Estudar culturas de fungos anemófilos em placas de *Petri* através de um nariz eletrônico;
- Analisar o ar do ambiente com a dispersão de *VOC*s e esporos de fungos anemófilos através de um nariz eletrônico;

- Testar experimentos com métodos de IA para a identificação dos fungos anemófilos estudados nos dois itens anteriores com vistas à classificação destes fungos.

1.3 ORGANIZAÇÃO DO TEXTO

O Capítulo 2, Fundamentação Teórica, corresponde a um breve estudo sobre os fungos anemófilos, a origem das colônias de fungos utilizadas no estudo, breves descrições dos fungos cobertos pelo trabalho, assinaturas de odor de espécies de fungos, uma introdução à Inteligência Artificial IA e uma breve discussão sobre séries temporais. O Capítulo 3, Revisão da Literatura, apresenta uma breve revisão da literatura sobre tema de interesse desta dissertação, abordando trabalhos relacionados às áreas de detecção e identificação de fungos e à classificação de séries temporais. No Capítulo 4, Materiais e Métodos, analisam-se o uso do nariz eletrônico e os métodos de aprendizagem de máquina testados para resolver o problema. O Capítulo 5 Resultados Experimentais, versa sobre os dados levantados durante a pesquisa, sobre as bases de dados propostas para solução do problema, sobre a exploração dos dados levantados, a metodologia de processamento dos experimentos e os resultados obtidos. Finalmente, como o próprio título evidencia, o Capítulo 6, Conclusões e Trabalhos Futuros, apresenta as conclusões e os trabalhos futuros que podem ser iniciados e/ou incentivados a partir do presente estudo.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, faz-se um breve estudo sobre os fungos anemófilos além de breves descrições dos fungos cobertos pelo trabalho, uma introdução à Inteligência Artificial IA e uma breve discussão sobre séries temporais.

2.1 FUNGOS ANEMÓFILOS

Uma excelente descrição sobre fungos microscópicos, entre esses os fungos anemófilos, é encontrada em (TROVÃO; PEREIRA, 2019), na qual se basiea muito desta seção.

2.1.1 O que são fungos?

Os fungos, dentre eles os anemófilos, são um grupo de seres vivos que, durante muito tempo, foram considerados como vegetais e, somente a partir de 1969, passaram a ser classificados como reino à parte, o reino Fungi (TROVÃO; PEREIRA, 2019).

Em (WHITTAKER, 1969) foi proposto um sistema de cinco reinos, criando concomitantemente o reino Fungi. Este novo sistema era baseado em três critérios principais: o nível de organização celular (diferenciação de células procariontes para eucariontes e uni e multicelularidade), modo de nutrição e interações nos ecossistemas (MOREIRA, 2014).

Os fungos apresentam um conjunto de características próprias que permitem a sua diferenciação das plantas: não possuem clorofila, não têm celulose nas suas paredes celulares (exceto em alguns fungos aquáticos) e não armazenam amido como substância de reserva. A presença de substâncias quitinosas na parede da maior parte das espécies fúngicas e a sua capacidade de depositar glicogênio os assemelham às células animais (TROVÃO; PEREIRA, 2019).

São seres vivos eucarióticos, com um só núcleo, como as leveduras, ou multinucleados, como os bolores. O seu citoplasma contém mitocôndrias e retículo endoplasmático rugoso, são heterotróficos e alimentam-se principalmente de matéria orgânica em decomposição - fungos saprófitos, ou de matéria orgânica viva - fungos parasitas (TROVÃO; PEREIRA, 2019).

Os componentes principais da parede celular são as hexoses e as hexoaminas, que formam mananas, ducanas e galactanas. Alguns fungos têm paredes celulares ricas em quitina (N-acetil glicosamina), outros possuem complexos polissacarídeos e proteínas, com predominância de cisteína. Os fungos apresentam uma variabilidade enzimática muito grande e, por isso, conseguem habitar nos mais variados substratos (TROVÃO; PEREIRA, 2019).

São aeróbios na sua grande maioria, mas se conhecem alguns fungos anaeróbicos estritos e facultativos. Podem ser uni ou multicelulares e reproduzem-se sexuada ou asse-

xuadamente, podendo apresentar ciclos parassexuados (TROVÃO; PEREIRA, 2019).

Dentre os gêneros mais frequentes de fungos anemófilos podem-se destacar, o *Aspergillus*, o *Fusarium*, o *Penicillium*, o *Alternaria* e o *Cladosporium* (TROVÃO; PEREIRA, 2019).

Em algumas ocasiões, esses microrganismos podem ser a causa de infecções em plantas, em humanos e em animais, na maioria das vezes como oportunistas. Como os esporos dos fungos se encontram no ar, é inevitável que se depositem na superfície de tudo o que lhe está exposto. Os fungos crescem onde existe matéria orgânica disponível, viva ou morta, geralmente adaptando-se ao calor e umidade. Água, solo, troncos, folhas, frutos, sementes, excrementos, insetos, alimentos frescos e processados, têxteis e outros produtos fabricados pelo homem constituem excelentes substratos para o seu desenvolvimento (TROVÃO; PEREIRA, 2019).

As leveduras constituem um grupo de microrganismos unicelulares que se reproduzem assexuadamente por gemulação ou por cissiparidade e promovem a fermentação alcoólica. São largamente encontradas na natureza: são comuns no solo, nas superfícies de órgãos dos vegetais, principalmente em flores e frutos, no trato intestinal de animais, em líquidos açucarados, e numa grande série de outros locais (TROVÃO; PEREIRA, 2019).

As colônias aveludadas ou pulverulentas, conhecidas por "bolores", são formadas por fungos de organização multicelular (fungos filamentosos). Os fungos filamentosos apresentam crescimento apical (longitudinal), reprodução por esporos e estruturas reprodutivas distintivas das suas células vegetativas. Podem ser parasitas, simbiontes (simbiose é associação de dois seres vivos, duas plantas ou uma planta e um animal, na qual ambos os organismos recebem benefícios, mesmo que em proporções desiguais) e, principalmente, saprófitos (que se alimentam absorvendo substâncias orgânicas normalmente provenientes de matéria orgânica em decomposição) (TROVÃO; PEREIRA, 2019).

2.1.2 Metodologias no estudo de fungos - Meios de cultura e obtenção de colônias

Os fungos são normalmente isolados por plaqueamento de uma amostra previamente recolhida (por exemplo a partir de solo, líquidos, superfícies e do ar) numa placa de *Petri* com meio de cultura próprio para o seu crescimento. O plaqueamento pode ser realizado, por exemplo, por diluição da amostra em água ou em solução salina de baixa concentração e posterior espalhamento na placa de meio de cultura (TROVÃO; PEREIRA, 2019).

De uma forma geral, a preparação de meios de cultura pode ser realizada pela adição dos seus constituintes em água destilada ou pela solubilização de meios comerciais nela liofilizados, seguido de esterilização em autoclave (durante cerca de 15 minutos, pressão de 1 ATM e temperatura de 121ºC) (TROVÃO; PEREIRA, 2019).

Após breve arrefecimento do meio, sem que ocorra a sua total solidificação (devido ao ágar), poderá ser adicionado um antibiótico para evitar o crescimento de bactérias inde-

Figura 1 – Câmera de fluxo (esquerda) e placas de *Petri* (direita)

Fonte: (TROVÃO; PEREIRA, 2019).

sejadas (exemplos: estreptomicina, cloranfenicol ou penicilina) e vertido cuidadosamente para placas de *Petri* (TROVÃO; PEREIRA, 2019).

Este procedimento deve ser realizado dentro de uma câmara de fluxo, previamente esterilizada (com álcool e exposto à luz ultravioleta por, pelo menos, 15 minutos) e em condições de assepsia, isto é, impedindo ao máximo que ocorram contaminações de origem externa, a Figura 1 mostra uma câmara de fluxo típica (TROVÃO; PEREIRA, 2019).

Após arrefecimento total do meio nas placas de *Petri*, recomenda-se um período de resguardo de cerca de 5 dias, com vista à confirmação da não existência de qualquer tipo de contaminação. Findo este período, as placas de meio podem ser utilizadas para a inoculação ou a propagação de colônias, a Figura 1 mostra placas (TROVÃO; PEREIRA, 2019).

A utilização de meios pobres é realizada também muitas vezes para induzir a formação de estruturas sexuais em culturas que não esporulam facilmente. Este estímulo baseia-se no princípio de que a concentração das fontes de carbono e de nitrogênio, e a temperatura de cultivo influenciam o tipo de reprodução (assexual vs. sexual) (TROVÃO; PEREIRA, 2019).

Dado que vários fungos são parasíticos ou endofíticos (vivem no interior) de espécies de plantas, uma alternativa ao cultivo em meios pobres, para induzir a esporulação, pode ser o seu cultivo em meio de cultura na superfície do qual colocam-se partes da planta que parasitam (por exemplo, agulhas de pinheiro). Esta técnica permite um crescimento "in situ" em laboratório, e, consequentemente, a visualização das estruturas reprodutoras. Estas modificações são necessárias para se formarem as estruturas reprodutivas de sobre-

vivência, que não seriam necessárias em um meio "rico" em nutrientes e que são, muitas vezes, essenciais para a correta identificação das diferentes colônias (TROVÃO; PEREIRA, 2019).

Para além da disponibilidade de nutrientes, há outros fatores externos que afetam o crescimento dos fungos, tais quais, nomeadamente o pH, a temperatura, a umidade e a luz (TROVÃO; PEREIRA, 2019).

Quanto ao pH, há espécies que vivem em ambientes com pH de 1,5 a 11. Para a maioria das espécies, o pH ótimo está na faixa de 5 a 7. Muitos fungos são ácido tolerantes e possuem alta capacidade tamponante. No que diz respeito à temperatura, a maioria dos fungos se desevolvem nas seguites faixas: psicrófilo de 5ºC a 20ºC, mesófilo de 20ºC a 30ºC e termófilo de 30ºC a 50ºC. Fora desses limites, ou seja, abaixo de 5ºC ou acima de 50ºC, poucos fungos se desenvolvem. Todos os fungos necessitam de umidade (água) para a absorção de nutrientes. Os esporos/conídios necessitam de água para germinarem. Em caso de escassez de umidade, o fungo é capaz de formar estruturas de resistência (clamidoconídio) e podem formar esporos.A luz visível (380-720 nm) não afeta o crescimento dos fungos, mas pode estimular a pigmentação. A luz tem efeito tanto na diferenciação de alguns fungos, quanto na reprodução sexual ou assexual.

No momento de isolamento de culturas, é salutar também levar em consideração a esporulação agressiva de algumas espécies, o que se traduz numa predominância destas face a outras com menores níveis de esporulação, e até mesmo em relação a espécies que apresentem crescimento mais lento. Esta problemática no isolamento pode, no entanto, ser resolvida por sucessivas diluições da amostra inicial e posterior plaqueamento, com vista à exposição das espécies menos abundantes (TROVÃO; PEREIRA, 2019).

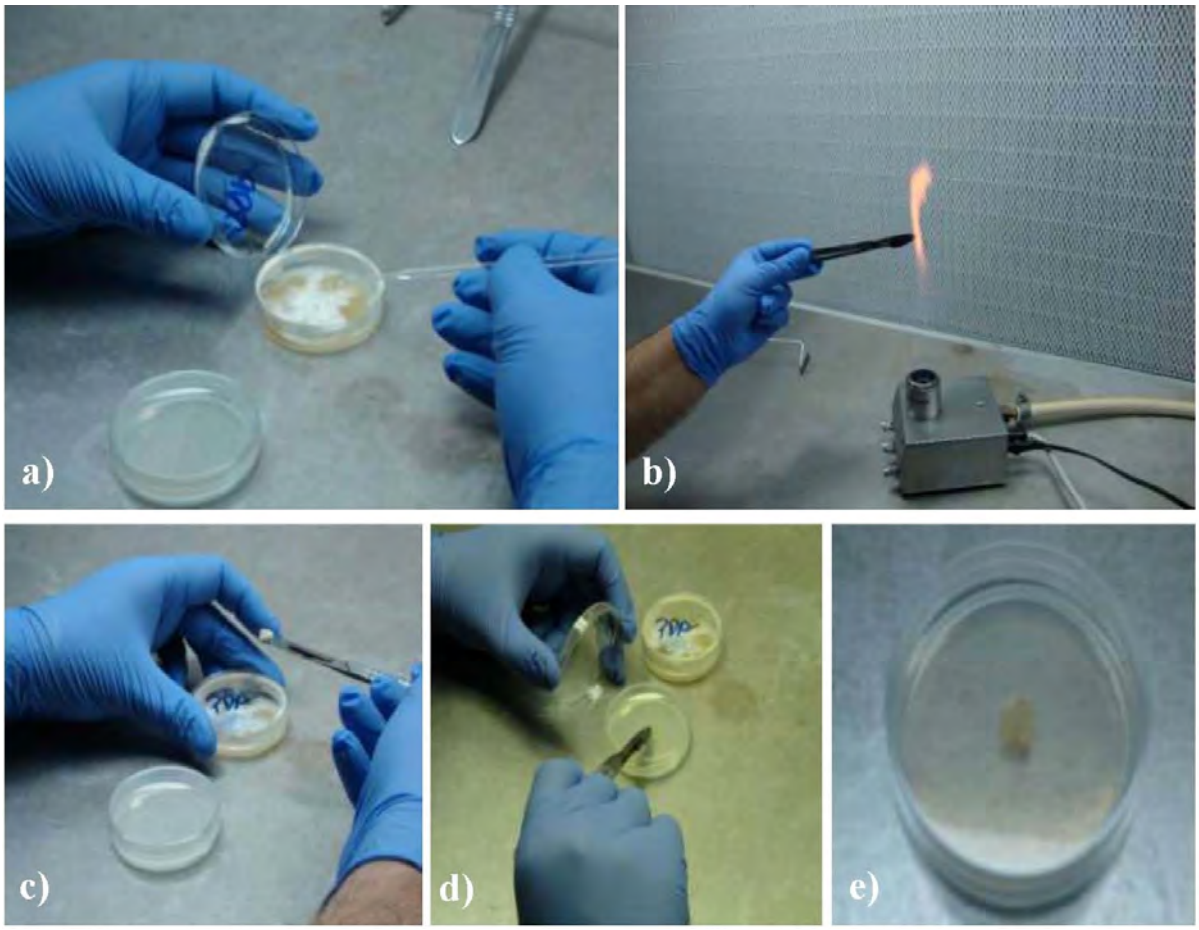
Os fungos podem ainda ser cultivados em culturas líquidas (sem adição de ágar), e este tipo de cultivo, normalmente, tem por finalidade a produção de elevadas quantidades de biomassa (biorreator), enzimas e antibióticos (TROVÃO; PEREIRA, 2019).

2.1.3 Propagação e preservação de culturas

Após a obtenção das diferentes culturas por plaqueamento, estas podem ser mantidas em culturas puras por propagação ("repicagem"). A repicagem consiste na inoculação de uma pequena parte do fungo em estudo em placas de *Petri* com meio fresco. Para a repicagem, são necessários bisturis ou ansa (alça) de inoculação, bico de *Bunsen* e *parafilm*, e tal procedimento deve ser realizar-se em condições de assepsia e dentro de uma câmara de fluxo (TROVÃO; PEREIRA, 2019).

Para realizar uma propagação, deve-se inicialmente colocar o bisturi ou a ansa (alça) de inoculação (não sendo descartável) na chama do bico de *Bunsen*, até que esta fique incandescente. Em seguida, e após o arrefecimento da mesma (para não destruir as estruturas reprodutoras por ação do calor), deve-se cortar um quadrado de cerca de 1cm por 1cm (no caso da utilização do bisturi; no caso de utilização de ansa (alça), uma passagem

Figura 2 – a) Repicagem com ansa de inoculação; b) Esterilização de bisturi; c) Repicagem com bisturi; d) Repicagem em novo meio de cultura; e) Novo repicado após propagação



Fonte: (TROVÃO; PEREIRA, 2019).

sobre a zona esporulante será suficiente) da colônia que se pretende propagar, e colocá-lo numa nova placa de *Petri* com a face contendo a colônia virada para o encontro ao ágar da nova placa de meio. (TROVÃO; PEREIRA, 2019).

Caso necessite repetir este procedimento, deve-se expor novamente o bisturi ou a ansa (alça) de inoculação à chama do bico de *Bunsen*, até que fique incandescente, de forma a esterilizar o instrumento de inoculação. Antes de executar uma nova repicagem, deve-se assegurar que o bisturi ou a ansa de inoculação foram devidamente arrefecidos, de forma a não provocar a morte da colônia pelo contato com o instrumento quente, conforme ilustrado na Figura 2 (TROVÃO; PEREIRA, 2019).

A conservação de culturas pode ser realizada seguindo variados procedimentos e consoante o tempo de conservação pretendido. Para conservar uma cultura durante vários meses, pode-se recorrer à sua refrigeração a 4^{o}C, enquanto que para uma conservação por maiores períodos de tempo (por exemplo em herbário), deve-se recorrer a outros procedimentos como, por exemplo, a liofilização ou a utilização de nitrogênio líquido (para informação relativa a outras técnicas de conservação ver (HUMBER, 1997)) (TROVÃO; PEREIRA, 2019).

A conservação de culturas a 4^oC pode ser realizada de duas maneiras. A primeira consiste na simples refrigeração de colônias com cerca de 5 a 7 dias diretamente a 4^oC, e a segunda, consiste na repicagem para a superfície de um tubo de meio sólido, com adição de óleo mineral até cobertura de toda a superfície da colônia e posterior refrigeração a 4^oC. Para reutilizar uma cultura conservada através deste método, simplesmente deve-se proceder à repicagem da colônia para uma placa de meio de cultura nova e incubar por um período de 5 a 7 dias, a cerca de 25^oC. Caso a colônia demonstre perda de vitalidade, é necessário usar o caldo glicosado (TROVÃO; PEREIRA, 2019).

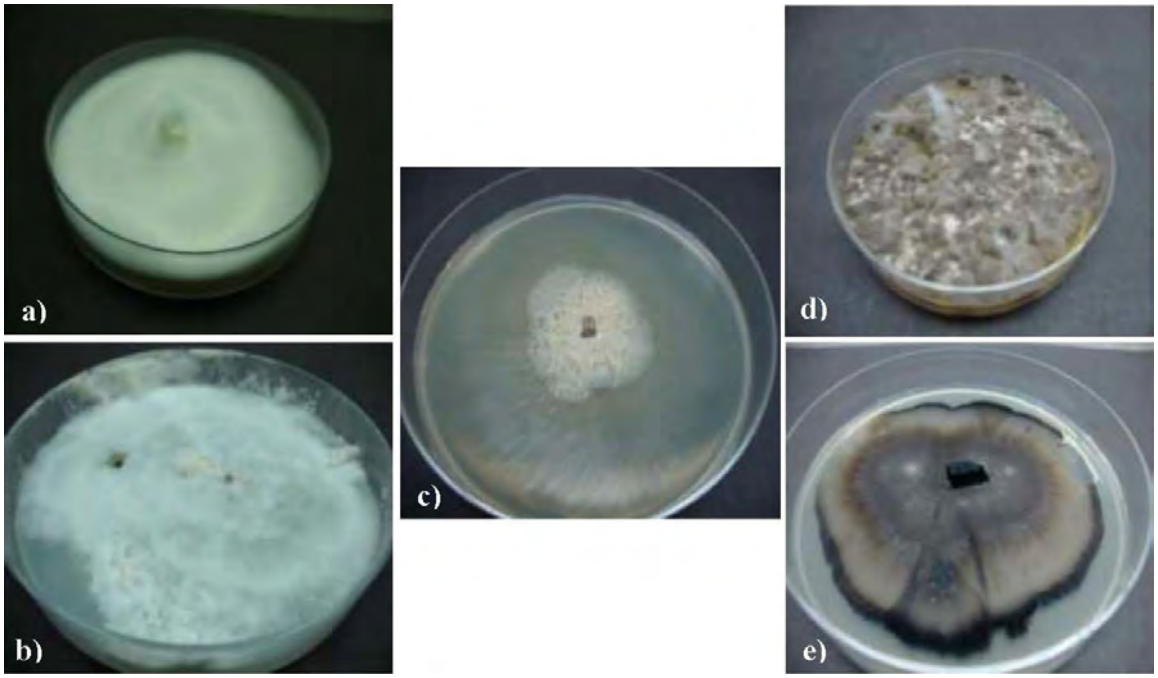
2.1.4 Textura, morfologia e topologia de culturas

Quando é necessário realizar a descrição de uma colônia ou realizar a sua avaliação com vista à sua identificação, três grandes características macroscópicas devem ser analisadas. Estas características são: a textura, a morfologia e a topologia das colônias. Além disso, caso as colônias libertem pigmento para o meio, alterando a sua cor, este deve também ser avaliado (TROVÃO; PEREIRA, 2019).

No que concerne à textura das colônias, de uma forma geral, podem-se observar 6 grandes tipos, aparências. São eles:

- **Lã de ovelha** (*"Cottony/ Woolly"*)
- **Veludo** (*"Velvety/ Suede-like"*)
- **Polvorosa ou em pó** (*"Powdery"*)
- **Granular ou arenosa** (*"Granular/ Sugary"*)
- **Glabra ou brilhante** (*"Glabrous/ Waxy"*)
- **Cremosa** (*"Creamy"*)

Figura 3 – Exemplos de tipos de textura de colônias: a) *"Cottony/ Woolly"* (aparência de lã de ovelha) b) *"Velvety/ Suede-like"* (aparência de veludo) c) *"Creamy"* (aparência cremosa) d) *"Powdery"* (aparência polvorosa ou em pó) e) *"Glabrous/ Waxy"* (aparência glabra ou brilhante)



Fonte: (TROVÃO; PEREIRA, 2019).

Em termos de morfologia de colônias, pode, de uma forma geral, ter 3 grandes tipos. São eles:

- Rugosas (descreve colônias que são enrugadas, dobradas e estriadas).
- Verrucosas (descreve colônias que são elevadas, enrugadas ou enroladas e com aspecto de verrugem).
- Umbiculadas (descreve colônias que se elevam num ponto leve, ou com elevação no meio, ou que apresentam forma de botões/ amontoados).

Em termos de topologia das colônias, podem-se analisar três propriedades, conforme ilustrado nas 3 tabelas a seguir. A Tabela 1 analisa a topologia das colônias em termos de forma, a Tabela 2 analisa a topologia das colônias em termos de elevação e a Tabela 3 analisa a topologia das colônias em termos de margem.

2.1.5 Micromorfologia básica de fungos

As hifas que compõem o corpo de um fungo podem ser de dois tipos, vegetativas (hifas que absorvem o alimento e que normalmente se encontram na superfície do ágar) ou aéreas (hifas que podem conter estruturas reprodutoras e que se estendem acima da superfície do ágar). As hifas aéreas podem servir de apoio a estruturas reprodutoras diferenciadas e são normalmente as responsáveis por características como a cor e a textura,

Tabela 1 – Topologia das colônias, em termos de forma

Forma	**Aparência**
Circular	
Irregular	
Filamentosa	
Rizóide	
Puntiforme	
Em fuso	

Fonte: (TROVÃO; PEREIRA, 2019).

Tabela 2 – Topologia das colônias, em termos de elevação

Elevação	**Aparência**
Plana	
Elevada	
Convexa	
Pulvinada	
Umbiculada	
Crateriforme	

Fonte: (TROVÃO; PEREIRA, 2019).

Tabela 3 – Topologia das colônias, em termos de margem

Margem	**Aparência**
Inteira	
Ondular	
Lobada	
Irregular	
Filamentosa	
Enrolada/ encaracolada	

Fonte: (TROVÃO; PEREIRA, 2019).

quando se visualiza a parte frontal da colônia. As hifas vegetativas são responsáveis pelas características verificadas na parte reversa. Ambas ajudam na identificação e distinção das diferentes espécies (TROVÃO; PEREIRA, 2019).

2.1.6 Comparação entre os principais métodos comerciais de identificação de microrganismos

Nesta subseção, serão discutidos, brevemente, os principais métodos e técnicas de identificação de microrganismos do mercado.

O monitoramento ambiental e o controle microbiológico em áreas críticas no processo de produção industrial são uma garantia de qualidade das indústrias, principalmente das que dependem de ambientes controlados, como a indústria de alimentos ou medicamentos. Além disso, o monitoramento ambiental é uma importante ferramenta para a avaliação da eficácia de medidas de controle de contaminação e para a identificação de ameaças específicas à qualidade e à segurança dos produtos fabricados. A detecção e a identificação microbiológica são essenciais para a adoção e a manutenção de medidas preventivas e corretivas relacionadas aos procedimentos operacionais, validação dos processos de limpeza e sanitização das instalações e treinamento de pessoal (ZAMPARETTE, 2017).

2.1.6.1 Métodos Bioquímicos

Estão disponíveis no mercado alguns *kits* como: *RapID™ ONE System (Oxoid-ThermoFisher, Cambridge, UK), BD BBL Crystal™ Identification System (Becton Dickinson, Sparks, MD, USA) e API®/ID32 (bioMerieux, Marcy-l'Etoile, France)* que possuem uma série de provas bioquímicas compiladas em um único painel de identificação padronizado para cada grupo de microrganismos (ZAMPARETTE, 2017).

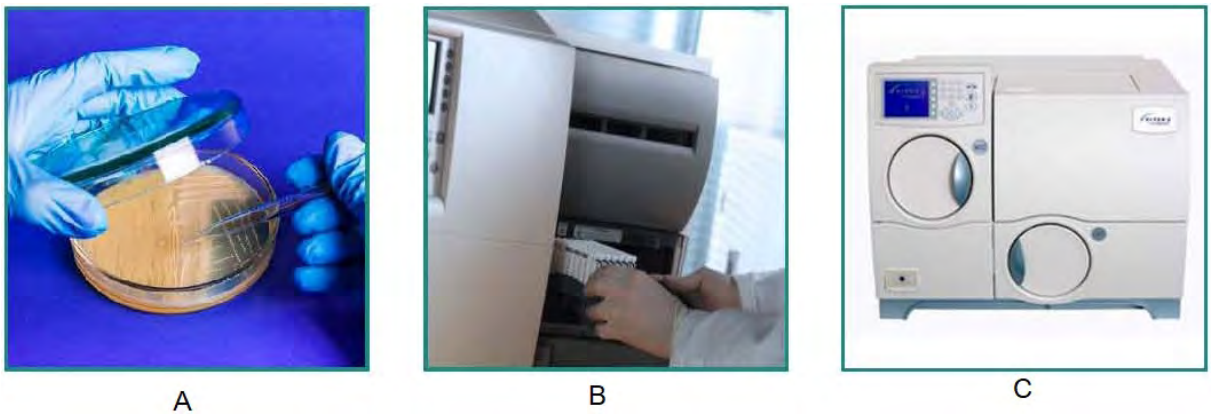
Portanto, para a utilização desses *kits* na identificação de microrganismos, é necessário conhecer o grupo de microrganismos ao qual a amostra pertence. Para a inoculação das amostras nesses *kits*, é necessário fazer uma suspensão do microrganismo e distribuí-la de maneira homogênea por todo o painel (ZAMPARETTE, 2017).

2.1.6.2 Método Automatizado

O método automatizado de identificação de microrganismos surgiu com o intuito de aperfeiçoar a identificação e diminuir a interferência e a dependência do operador. O aparelho mais conhecido é o *VITEK® Systems (bioMerieux, Marcy-l'Etoile, France)*. Essa metodologia faz a identificação microbiológica através de 64 provas colorimétricas dispostas em um cartão (ZAMPARETTE, 2017).

É um método dependente da cultura do microrganismo, cada cartão abrange um grupo de microrganismos diferentes, portanto crescimento em ágar seletivo ou coloração de *Gram* são necessários para a escolha dos cartões de identificação corretos. A leitura das provas

Figura 4 – (A) Cultura de microrganismo; (B) Preparação da suspenção de microrganismo e inoculação; (C) Interpretação dos resultados.



Fonte: (ZAMPARETTE, 2017).

é feita a cada 15 minutos com três comprimentos de ondas diferentes, conforme mostrado na Figura 4 (ZAMPARETTE, 2017).

2.1.6.3 Espectrometria de massas

Mais recentemente, a tecnologia *MALDI-TOF MS* atraiu rapidamente os microbiologistas pelo seu poderoso recurso de identificação microbiológica rápida. Esse método detecta e identifica proteínas pela determinação de seu peso molecular individual e de fragmentos específicos. Para a identificação de microrganismos na indústria, essa é ainda uma técnica dependente de cultura, porém, para o laboratório clínico, já é possível fazer a identificação de microrganismos a partir de amostras clínicas (ZAMPARETTE, 2017).

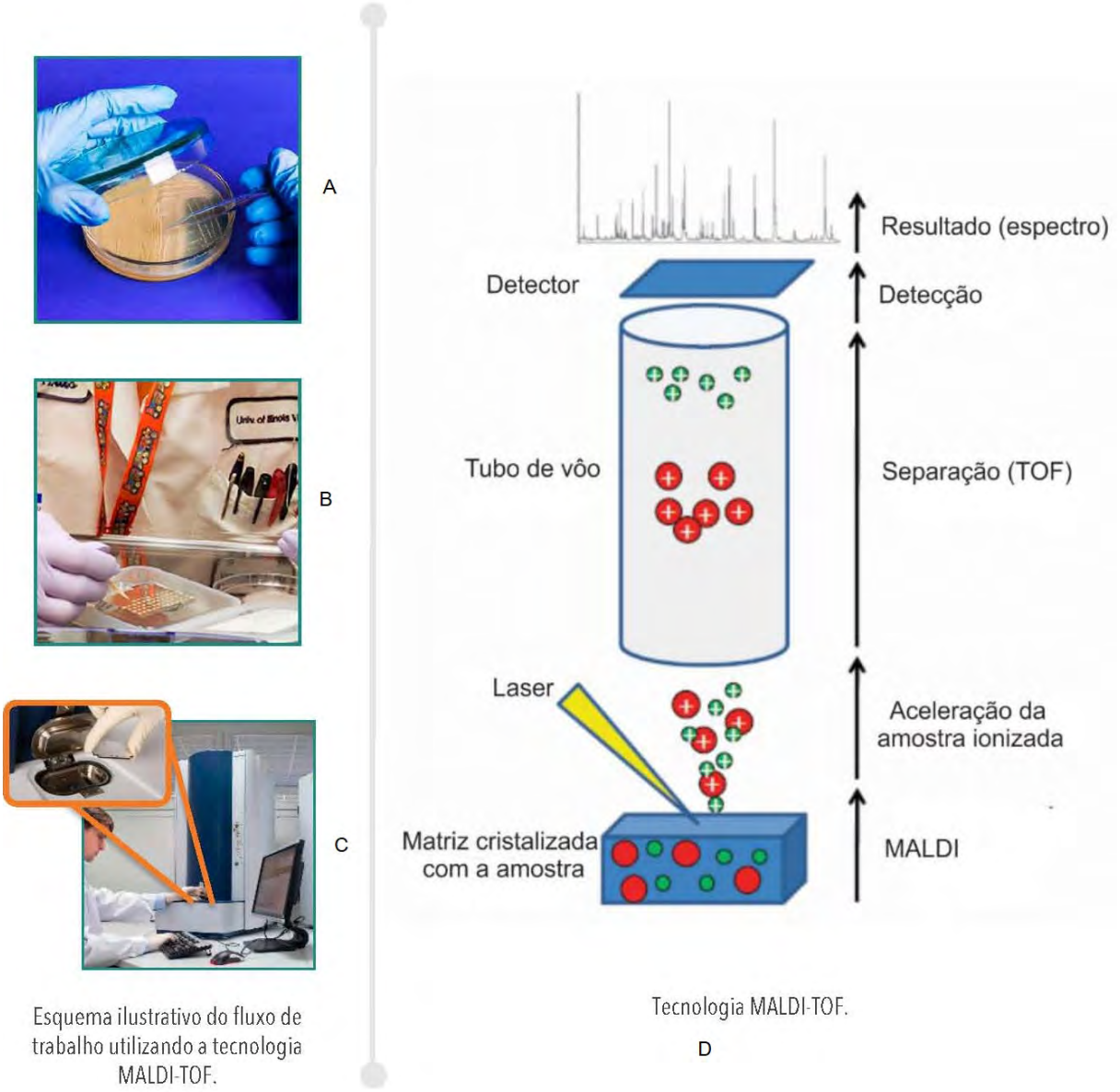
A primeira fase dessa metodologia é a preparação da suspensão de microrganismos, que, então, é colocada em uma placa junto com a matriz polimérica, sobre a qual um laser é irradiado e ocorre a ionização de várias moléculas. Essas moléculas são aspiradas por um tubo e ocorre a detecção de cada molécula através do tempo de chegada ao detector (*time of light*). Isso gera um gráfico que é específico para cada espécie microbiana e uma base de dados computadorizada interpreta e fornece o resultado, a Figura 5 ilustra o comentado (ZAMPARETTE, 2017).

2.1.6.4 Sequenciamento de DNA

Essa tecnologia consegue abranger, incluindo evolução e genômica comparativa, forense, epidemiologia, medicina aplicada para o diagnóstico e terapêutica. Atualmente, a necessidade de sequenciamento tem se tornado cada vez maior, haja vista a variedade de aplicações que essa tecnologia consegue abranger (ZAMPARETTE, 2017).

O método de Sanger é o procedimento tradicional de sequenciamento de DNA, criado em 1977. É um processo que envolve a síntese de um molde de DNA de um gene de inte-

Figura 5 – (A) Cultura de microrganismo; (B) Preparação da amostra; (C) Inoculação da amostra; (D) Metodologia *MALDI-TOF MS*.



Fonte: (ZAMPARETTE, 2017).

resse, por reação de *Polymerase Chain Reaction* (*PCR*), a Figura 6 mostra o procedimento (ZAMPARETTE, 2017).

O método de Sanger automatizado utiliza sequenciadores com eletroforese vertical em placa ou eletroforese em capilar, nos quais os Dideoxinucleotídeos (ddNTPs) são marcados com substância fluorescentes, conforme ilustrado na Figura 7 (ZAMPARETTE, 2017).

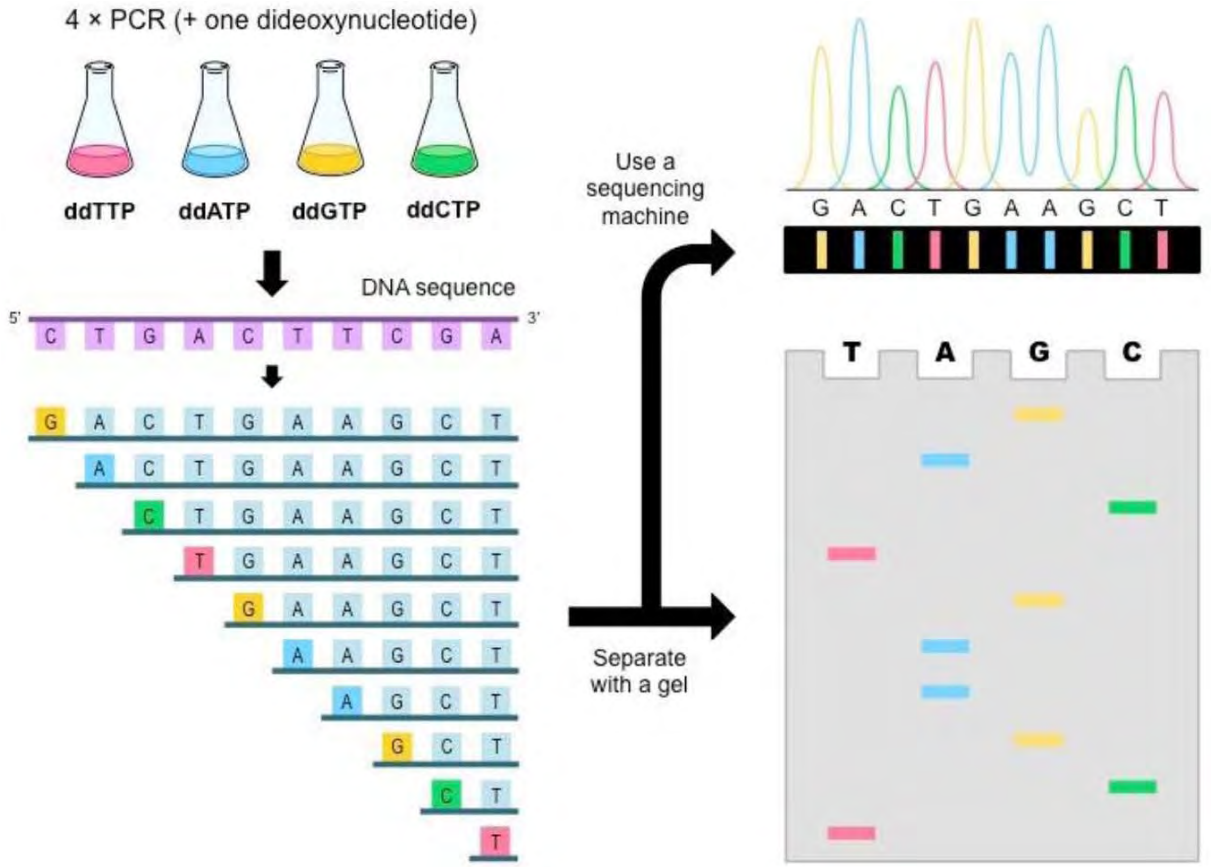
Em 2005, surgiu no mercado a tecnologia de sequenciamento de nova geração. Essa tecnologia engloba um número amplo de métodos para a preparação do molde de Ácido Desoxirribonucleico (DNA), sequenciamento, imagem e análise de dados. A combinação de protocolos específicos distingue uma tecnologia da outra e determina o tipo de dado produzido por cada plataforma (ZAMPARETTE, 2017).

Figura 6 – Esquema ilustrativo do fluxo de trabalho do sequenciamento Sanger.

Fonte: (ZAMPARETTE, 2017).

A imobilização do DNA em suportes sólidos, separados espacialmente, permite que milhares de bilhões de reações de sequenciamento sejam realizadas ao mesmo tempo. Essas plataformas conseguem gerar informações maiores que o sequenciamento de Sanger, com uma grande economia de tempo e custo. Essa maior eficiência advém do uso da clonagem in

Figura 7 – Esquema ilustrativo do método de Sanger automatizado.



Fonte: (ZAMPARETTE, 2017).

vitro e de sistemas de suporte sólido para as unidades de sequenciamento (ZAMPARETTE, 2017).

2.1.6.5 Compararação de metodologias

A Tabela 4 mostra as principais vantagens e desvantagens de cada método apresentado nas subseções acima (ZAMPARETTE, 2017).

Tabela 4 – Compararação de metodologias

Metodologia	***RapID™ ONE System, BBL Crystal™ e API®/ID32***	***VITEK® Systems***	***MALDI-TOF MS***	**Sequenciamento de DNA *Next Generation Sequencing* (*NGS*)**
Vantagens	. Diminuição do tempo de preparação; . Maior rapidez nos resultados; . Redução do uso de materiais.	. Automatizado; . Configuração simples; . Fácil manuseio; . Mínimo de reagentes requeridos; . Rastreabilidade da amostra; . Redução do manuseio do operador; . Interface intuitiva.	. Integração dos resultados para otimização do workflow; . Rastreabilidade completa; . Resultados rápidos; . Menor tempo de manipulação; . Capacidade para grandes volumes de trabalho (192 amostras/corrida).	. Identificação robusta e específica; . Permite identificar fungos, protozoários e algas; . Alta sensibilidade e precisão; . Precisa de uma quantidade mínima de DNA para possibilitar a identificação; . Permite identificar vários microrganismos em uma única amostra.
Desvantagens	. Dependente de cultura; . Laborioso; . Depende bastante do operador; . A interpretação dos resultados pode ser subjetiva.	. Dependente de cultura; . Alto custo dos consumíveis; . Resultados falso-positivos; . Resultados falso-negativos.	. Dependente de cultura; . Alto custo do aparelho; . Banco de dados são proprietários;	. Difícil implementação em laboratórios de microbiologia; . Alto custo dos consumíveis.

Fonte: (ZAMPARETTE, 2017).

Todas as metodologias convencionais da microbiologia utilizadas na identificação de microrganismos possuem um fator limitante que é a cultura. Além disso, os métodos que utilizam testes fenotípicos para identificação microbiana, além de demandar mais tempo para aferir um resultado, muitas vezes estes não são satisfatórios, devido ao número limitado de provas ou de resultados falso-negativos. O sistema *MALDI-TOF MS* mostrou ser um importante avanço na microbiologia, porém a aquisição do aparelho se justifica apenas se o laboratório apresentar uma grande demanda na rotina de identificação microbiológica, devido ao seu alto custo (ZAMPARETTE, 2017).

Neste panorama, o Sequenciamento de DNA torna-se uma importante ferramenta na identificação e rastreamento de microrganismos, especialmente em áreas estéreis, nas quais a presença de microrganismos não é permitida, principalmente por não depender de culturas e por ter alta escalabilidade (ZAMPARETTE, 2017).

2.2 ORIGEM DAS COLÔNIAS DE FUNGOS UTILIZADOS NO ESTUDO

As colônias de fungos utilizadas no estudo são repiques de colônias isoladas em (COUTO; MOTTA, 2021). Foram realizadas coletas em 11 ambientes cirúrgicos climatizados do Hospital das Clínicas da Universidade Federal de Pernambuco, através da técnica de sedimentação passiva, utilizando meio de cultura ágar *sabouraud* em placas de *Petri* estéreis. Após o crescimento dos fungos, fragmentos das colônias foram transferidos separadamente para a confirmação da pureza. Alguns isolados foram identificados por taxonomia clássica e outros por taxonomia molecular realizada no Laboratório de Biologia Molecular da Micoteca URM do Departamento de Micologia, utilizando o *kit* de extração *Wizard Genomic DNA Purification Kit*. Os isolados purificados, após a amplificação, foram sequenciados na plataforma de sequenciamento de Departamento de Genética, ambos os departamentos pertencentes ao Centro de Biociências da Universidade Federal de Pernambuco.

A limitação dessa dissertação que é de princípio de pesquisa e nesta etapa serão estudados apenas 5 gêneros de fungos *Aspergillus*, *Cladosporium*, *Fusarium*, *Penicillium* e *Rhizomucor*.

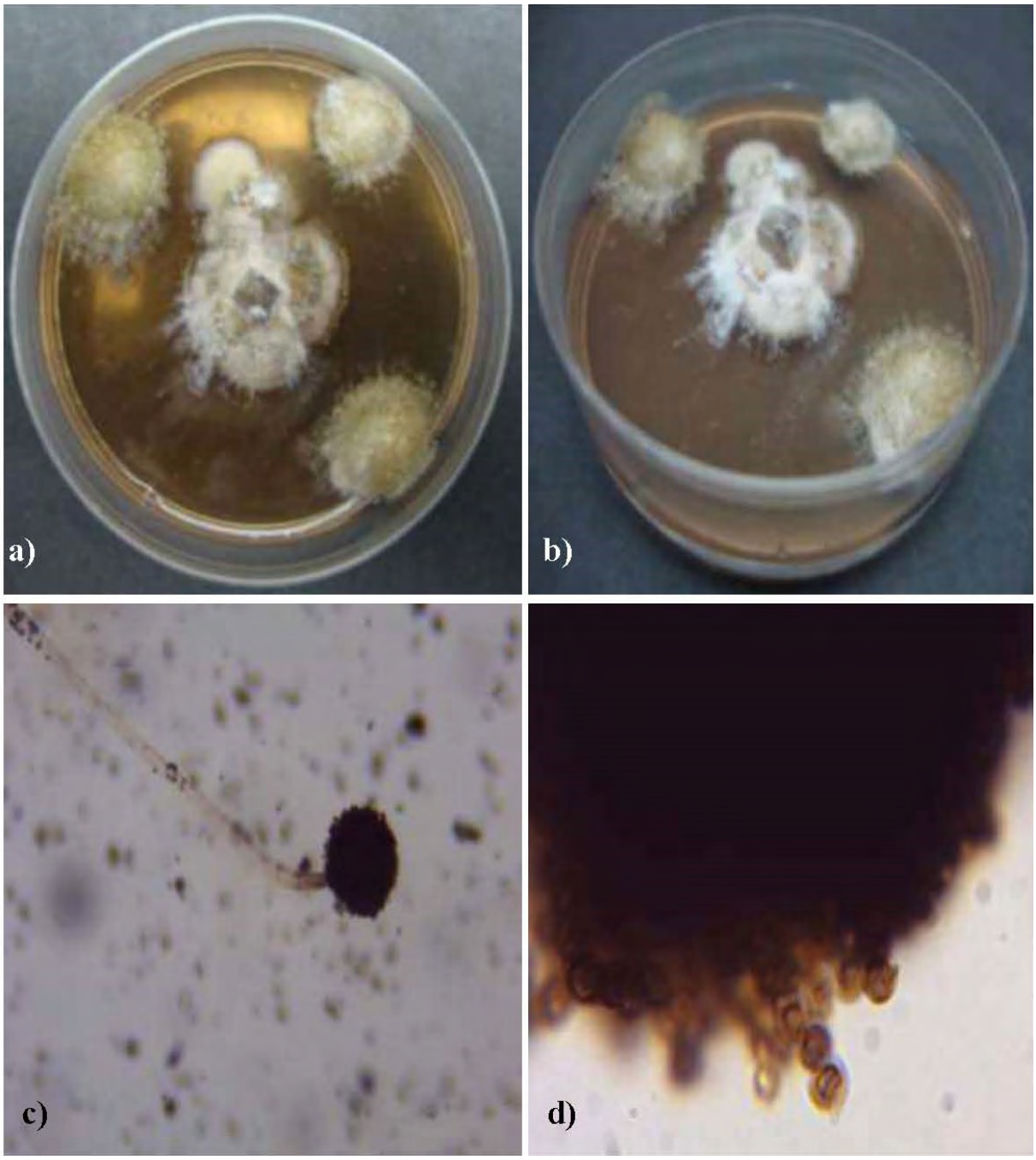
A seguir serão discutidos, de forma abreviada, esses 5 gêneros de fungos.

2.2.1 Breve descrição do Gênero *Aspergillus*

O gênero *Aspergillus* é um fungo cosmopolita, presente em diversos ambientes e composto por cerca de 250 espécies. A sua forma teleomórfica é conhecida por Emericella (TROVÃO; PEREIRA, 2019).

O gênero é caracterizado pelas vesículas em forma de aspersório (do latim aspergillum) nas quais se encontram os esporos. As suas espécies possuem a capacidade de crescer em altas pressões osmóticas, são fortemente aeróbicas e, como tal, são encontradas em quase todos os ambientes ricos em oxigénio (TROVÃO; PEREIRA, 2019).

Figura 8 – a) b) Exemplos de colônias de *Aspergillus sp.*; c) exemplo de canidióforo de *Aspergillus sp.*; d) exemplo de esporos de *Aspergillus sp.*

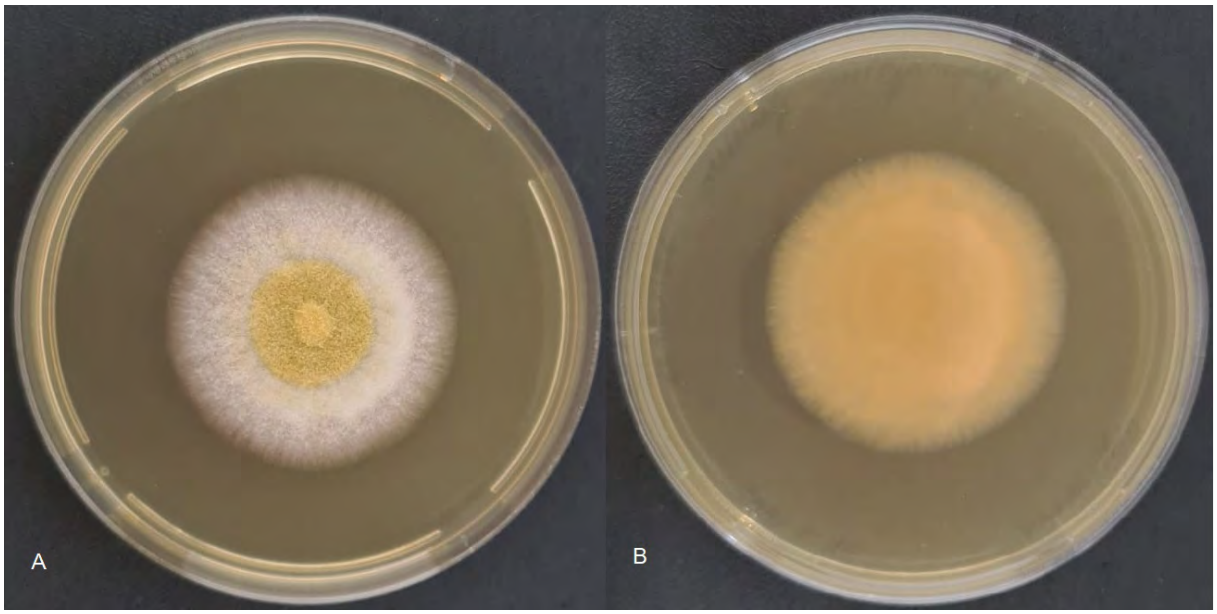


Fonte: (TROVÃO; PEREIRA, 2019).

Do ponto de vista ecológico, crescem em substratos ricos em carbono e são contaminantes comuns de produtos alimentares (bolor do pão e das frutas), podendo ainda infectar ou crescer na superfície de muitas plantas e árvores (TROVÃO; PEREIRA, 2019).

Do ponto de vista biotecnológico, são fontes de moléculas naturais aplicadas na indústria farmacêutica. A espécie *A. niger* é capaz de produzir ácido cítrico, contribuindo com 99% da produção mundial deste tipo de ácido. Algumas espécies são também utilizadas na produção de diversas enzimas com interesse comercial (TROVÃO; PEREIRA, 2019).

Figura 9 – Cultura em placa de *Petri* da espécie de fungo *Aspergillus flavus* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

Por outro lado, algumas espécies podem causar infecções em animais e em humanos como *A. flavus* e *A. fumigatus*, uma vez que produzem aflatoxina que é simultaneamente uma toxina e um carcinógeno. Adicionalmente algumas espécies podem causar o conjunto de micoses denominadas por aspergiloses. Outras espécies são ainda agentes patogênicos agrícolas, causando doenças em plantas como o milho (TROVÃO; PEREIRA, 2019).

A Figura 9 e a Figura 10 ilustram duas culturas em placa de *Petri* do gênero *Aspergillus*, nas quais as espécies *Aspergillus flavus* e *Aspergillus steynii*, respectivamente, são utilizadas nesse trabalho. As amostras foram repicadas pelo Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021).

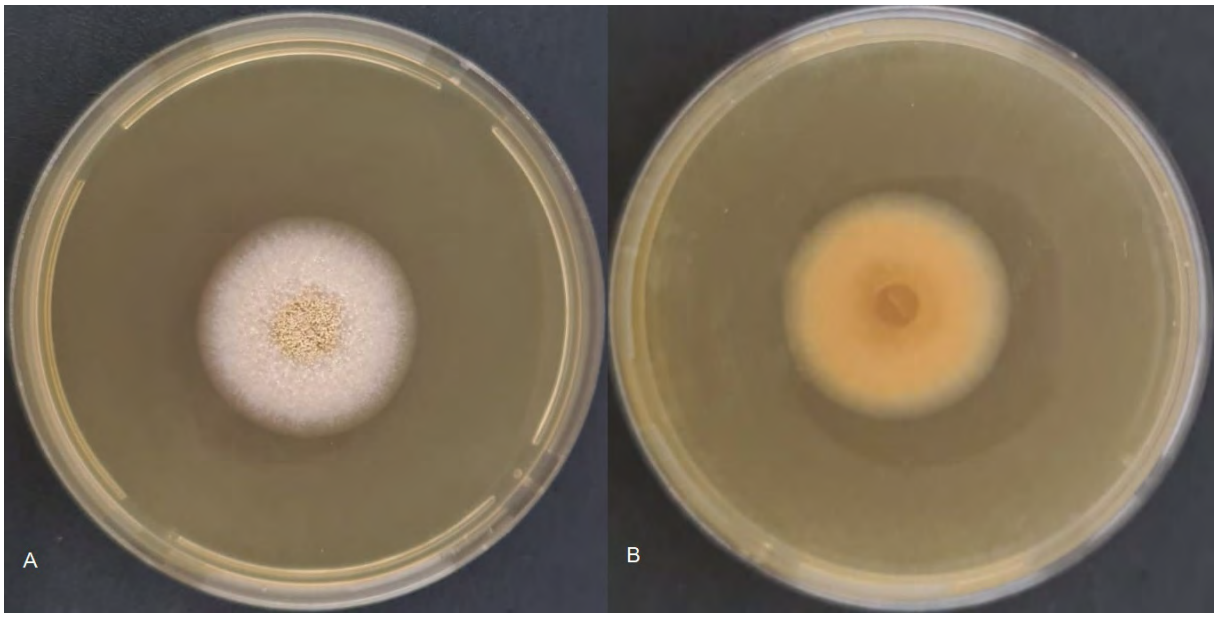
2.2.2 Breve descrição do Gênero *Cladosporium*

O gênero *Cladosporium* engloba mais de 750 espécies, representando um dos maiores grupos de fungos dematiáceos (de cor negra) conhecidos. O gênero é caracterizado pela presença de complexos de melanina em níveis elevados nas suas paredes celulares (TROVÃO; PEREIRA, 2019).

As suas espécies encontram-se amplamente distribuídas mundialmente e incluem os fungos mais comuns em ambientes internos e externos. Os esporos de *Cladosporium* são facilmente dispersos pelo vento e, no mais das vezes, são extremamente abundantes no ar, podendo, desta feita, crescer em superfícies mesmo quando há pouca umidade está presente (TROVÃO; PEREIRA, 2019).

Muitas espécies são altamente extremo tolerantes, crescendo em meios contendo até

Figura 10 – Cultura em placa de *Petri* da espécie de fungo *Aspergillus steynii* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

20% de NaCl. Por esta razão, são comuns em ambientes hipersalinos e extremos, onde muitos outros organismos não conseguem crescer. Os seus esporos são também muito frágeis, o que faz com que seja bastante difícil a preparação de amostras para microscopia de forma a manter a morfologia original das estruturas (TROVÃO; PEREIRA, 2019).

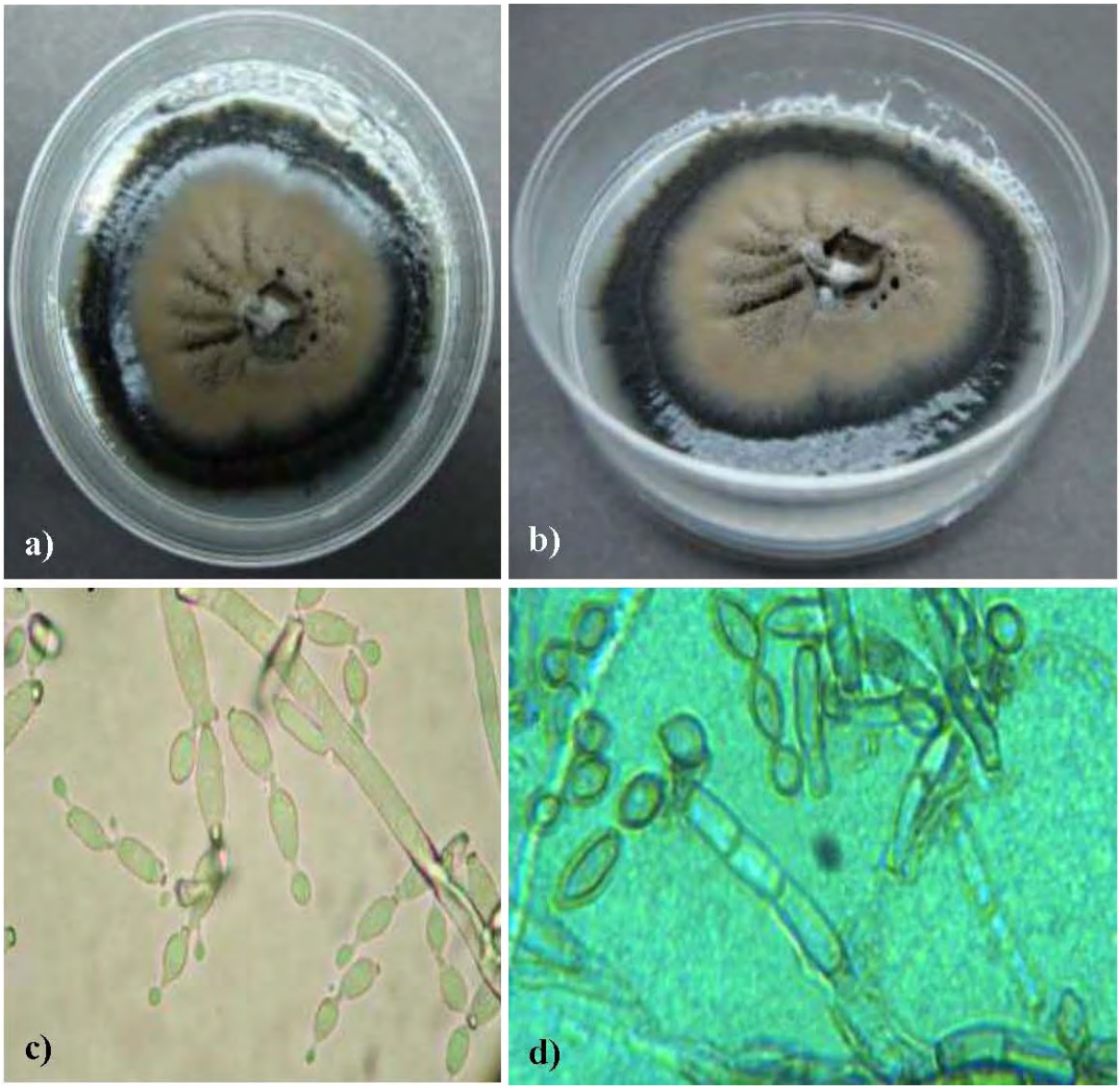
As espécies de *Cladosporium* raramente são patogênicas para humanos, mas há o relato de casos eventuais em que elas causaram infecções cutâneas e respiratórias. Os esporos são alergênicos que em grandes quantidades podem afetar severamente indivíduos asmáticos e pessoas com doenças respiratórias. As espécies de *Cladosporium* produzem vários compostos orgânicos voláteis (TROVÃO; PEREIRA, 2019).

A Figura 12 e a Figura 13 ilustram duas culturas em placa de *Petri* do gênero *Cladosporium*, as espécies *Cladosporium perangustum* e *Cladosporium vignae*, respectivamente, são utilizadas nesse trabalho. As amostras foram repicadas pelo Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021).

2.2.3 Breve descrição do Gênero *Fusarium*

O gênero *Fusarium* é um gênero vasto, que se encontra largamente distribuído em solos ou associado a várias espécies de plantas. A maior parte das suas espécies não causam doenças, no entanto algumas espécies produzem várias micotoxinas que afetam várias espécies de cultivo, os animais e o homem. São caracterizadas pela forma peculiar dos seus esporos em forma de "banana" e pela sua forma em fuso. O seu teleomorfo é conhecido como Gibberella (TROVÃO; PEREIRA, 2019).

Figura 11 – a) e b) Exemplos de colônias de *Cladosporium sp.*; c) e d) exemplos de conídios de *Cladosporium sp.*



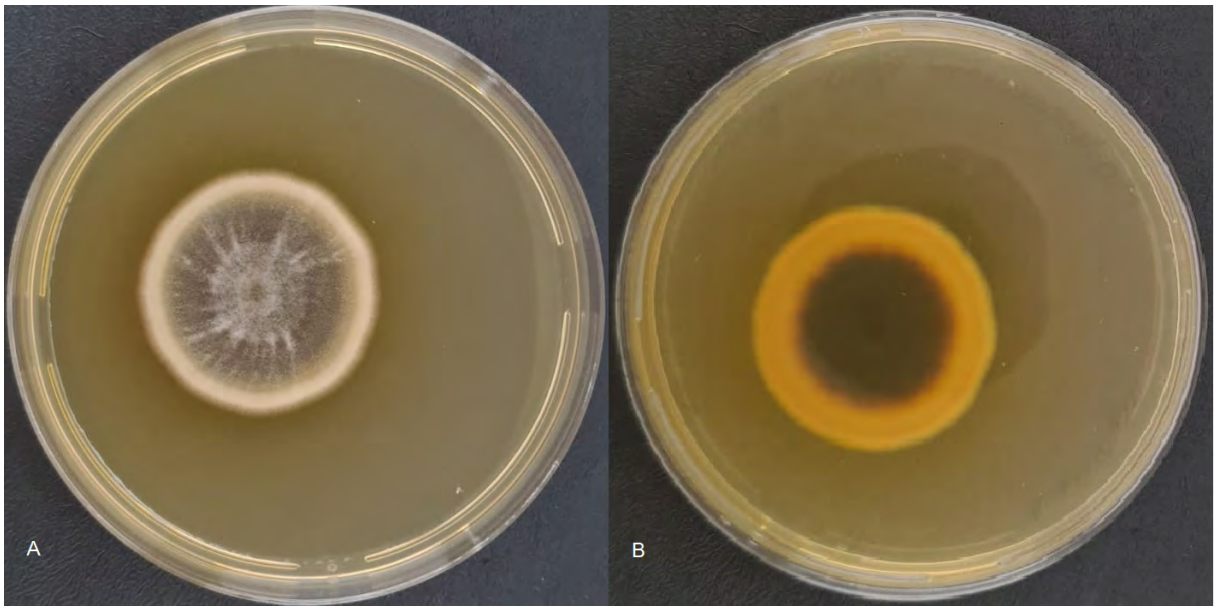
Fonte: (TROVÃO; PEREIRA, 2019).

O gênero inclui várias espécies patogênicas de plantas economicamente importantes, tais quais a cevada (ferrugem da cevada), o trigo, o milho e o pinheiro (cancro resinoso do pinheiro). A espécie *Fusarium oxysporum* afeta também o cultivo das bananeiras nos países do sul da América (TROVÃO; PEREIRA, 2019).

Algumas espécies podem causar infecções oportunistas em seres humanos, que podem ser cutâneas e/ou oculares. Historicamente, esse gênero foi cosiderado uma potencial arma biológica, dado o número de vítimas acometidas por ele na União Soviética, durante as décadas de 1930 e 1940, quando contaminação das farinhas de trigo demonstrou uma taxa de mortalidade de cerca de 60%. A toxina mais explorada com vista a este fim foi a T-2 dos tricotecenos (TROVÃO; PEREIRA, 2019).

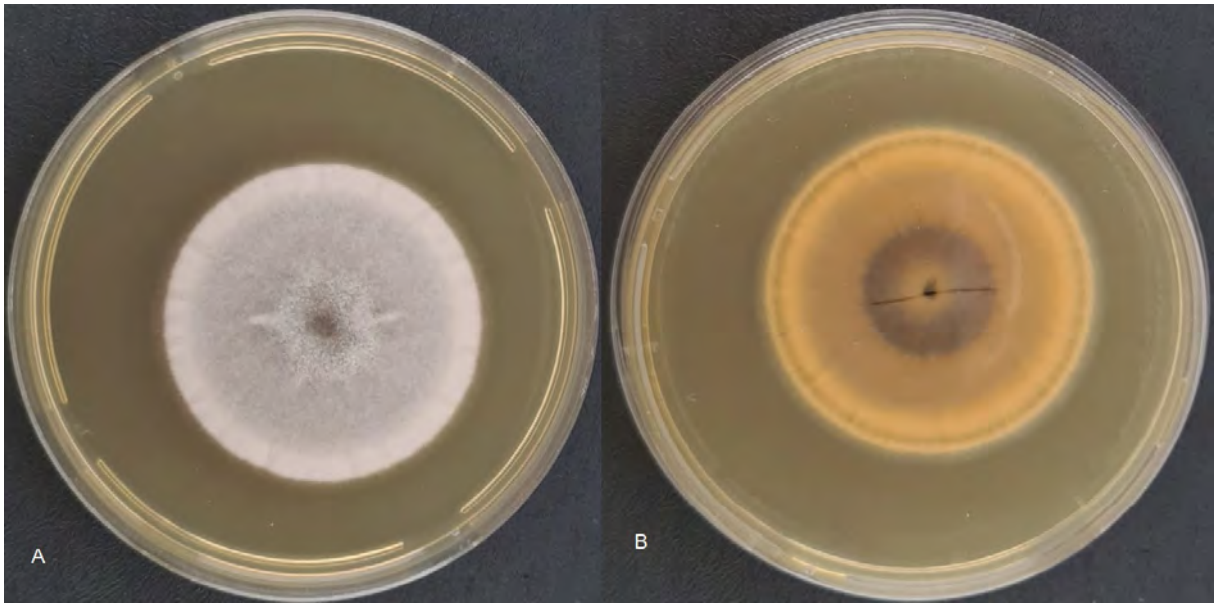
Foi também sugerido como agente biológico contra ervas daninhas, no entanto com

Figura 12 – Cultura em placa de *Petri* da espécie de fungo *Cladosporium perangustum* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

Figura 13 – Cultura em placa de *Petri* da espécie de fungo *Cladosporium vignae* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*

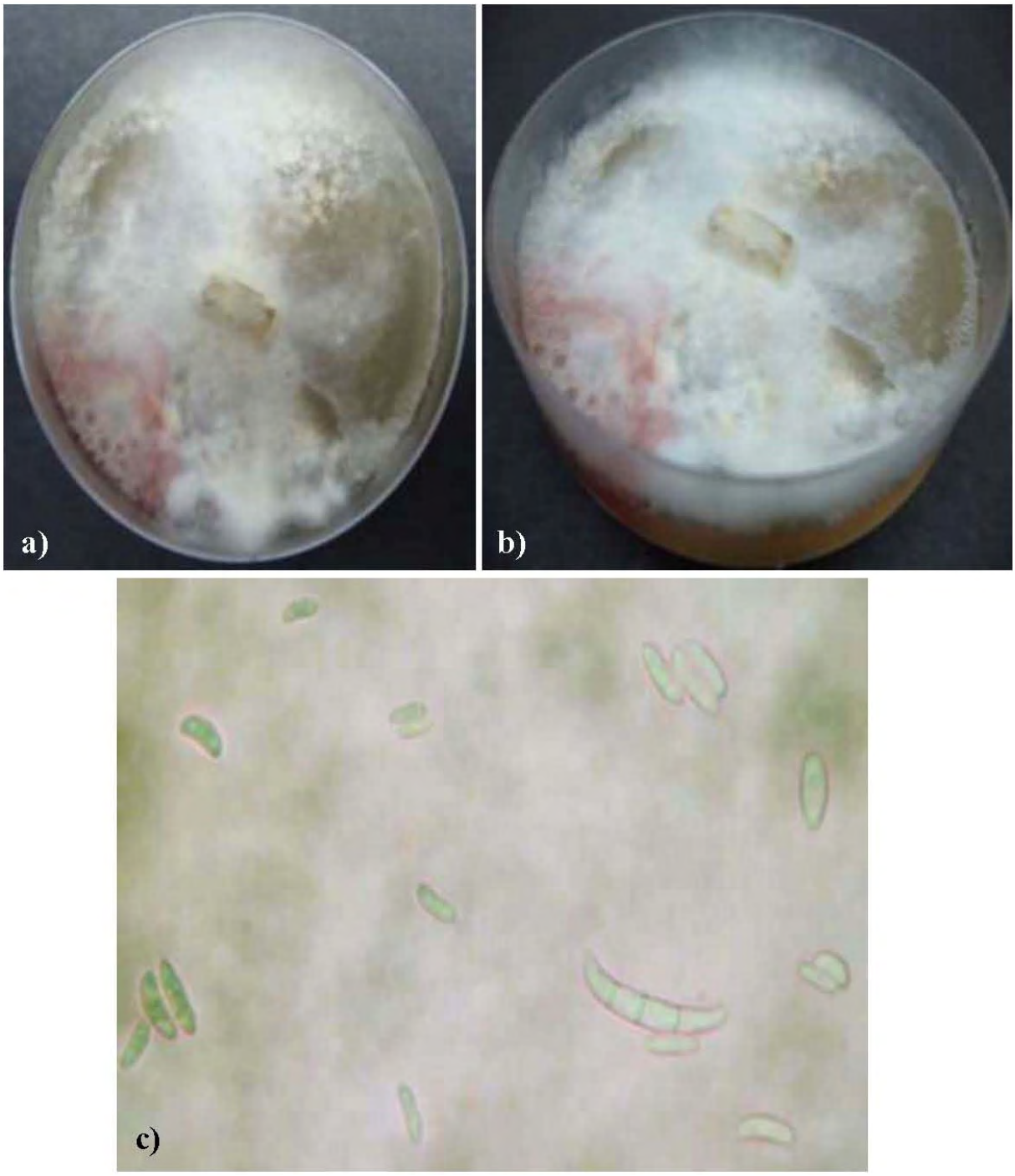


Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

alguma contestação e preocupação devido à sua rotulação como potencial arma de guerra biológica (TROVÃO; PEREIRA, 2019).

A Figura 15 e a Figura 16 ilustram duas culturas em placa de *Petri* do gênero *Fusarium*. Foram utilizadas nesse trabalho as espécies *Fusarium incarnatum* e *Fusarium pseudocircinatum*, respectivamente. As amostras foram repicadas pelo Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021).

Figura 14 – a) e b) Exemplos de colônias de *Fusarium sp*;. c) Exemplos de macrosporos e microsporos de *Fusarium sp.*

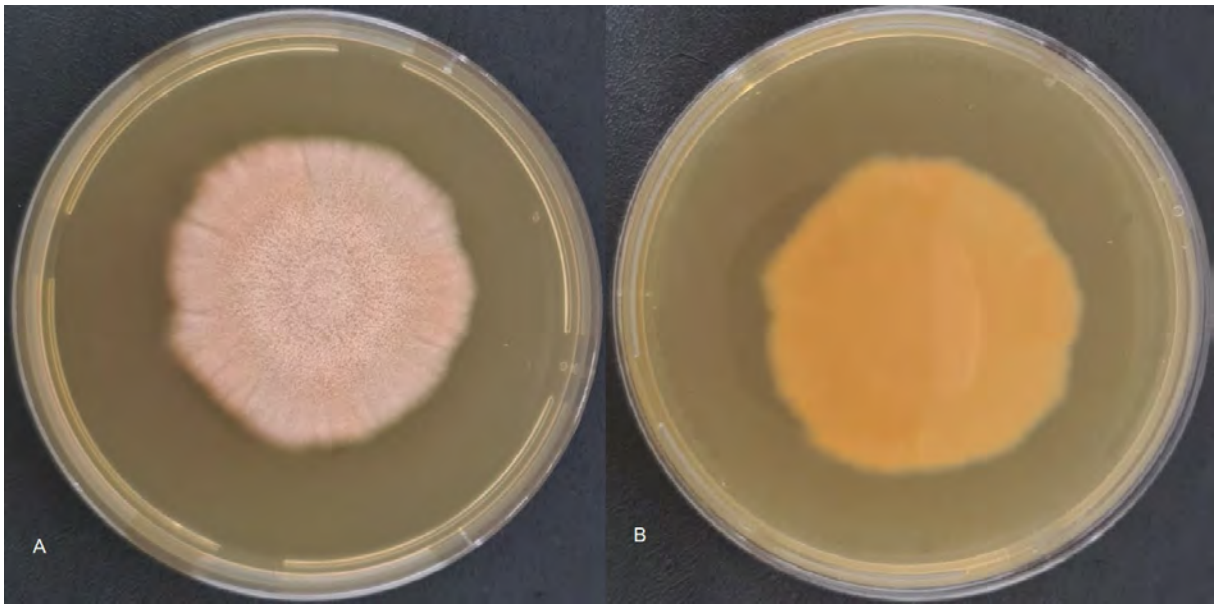


Fonte: (TROVÃO; PEREIRA, 2019).

2.2.4 Breve descrição do Gênero *Penicillium*

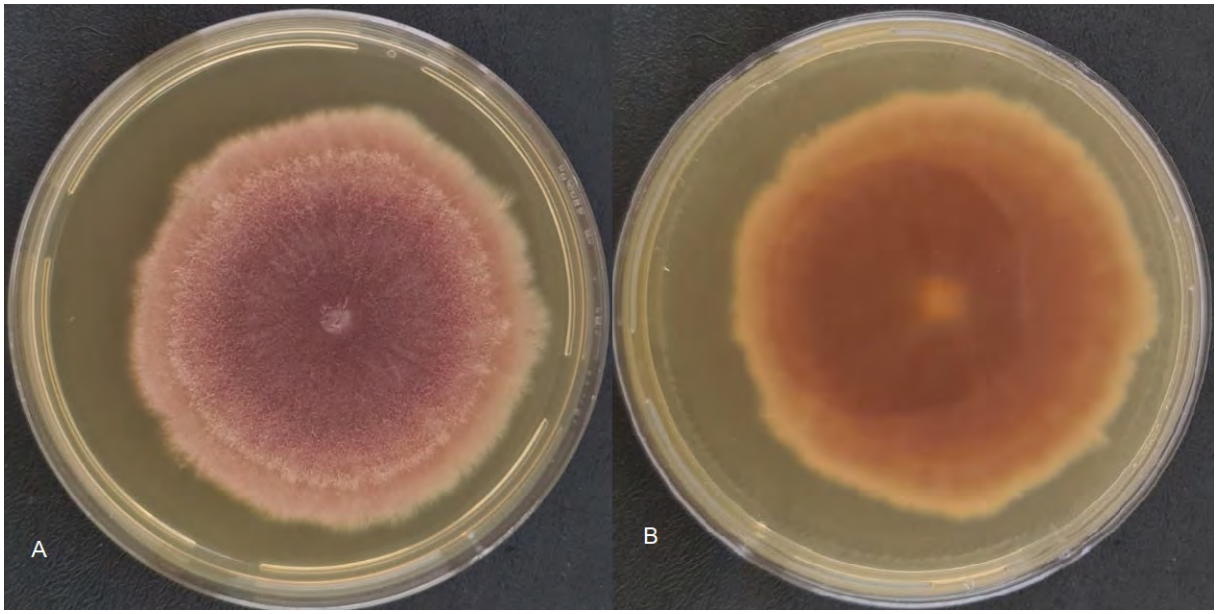
O gênero *Penicillium* é um fungo cosmopolita, presente em diversos ambientes e composto por cerca de 300 espécies. Os conidióforos deste gênero formam-se a partir do micélio em grande número e apresentam uma estrutura típicamente ramificada. Os conidiósporos são a sua principal via de dispersão e apresentam, muitas vezes, a cor verde. A reprodução

Figura 15 – Cultura em placa de *Petri* da espécie de fungo *Fusarium incarnatum* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

Figura 16 – Cultura em placa de *Petri* da espécie de fungo *Fusarium pseudocircinatum* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



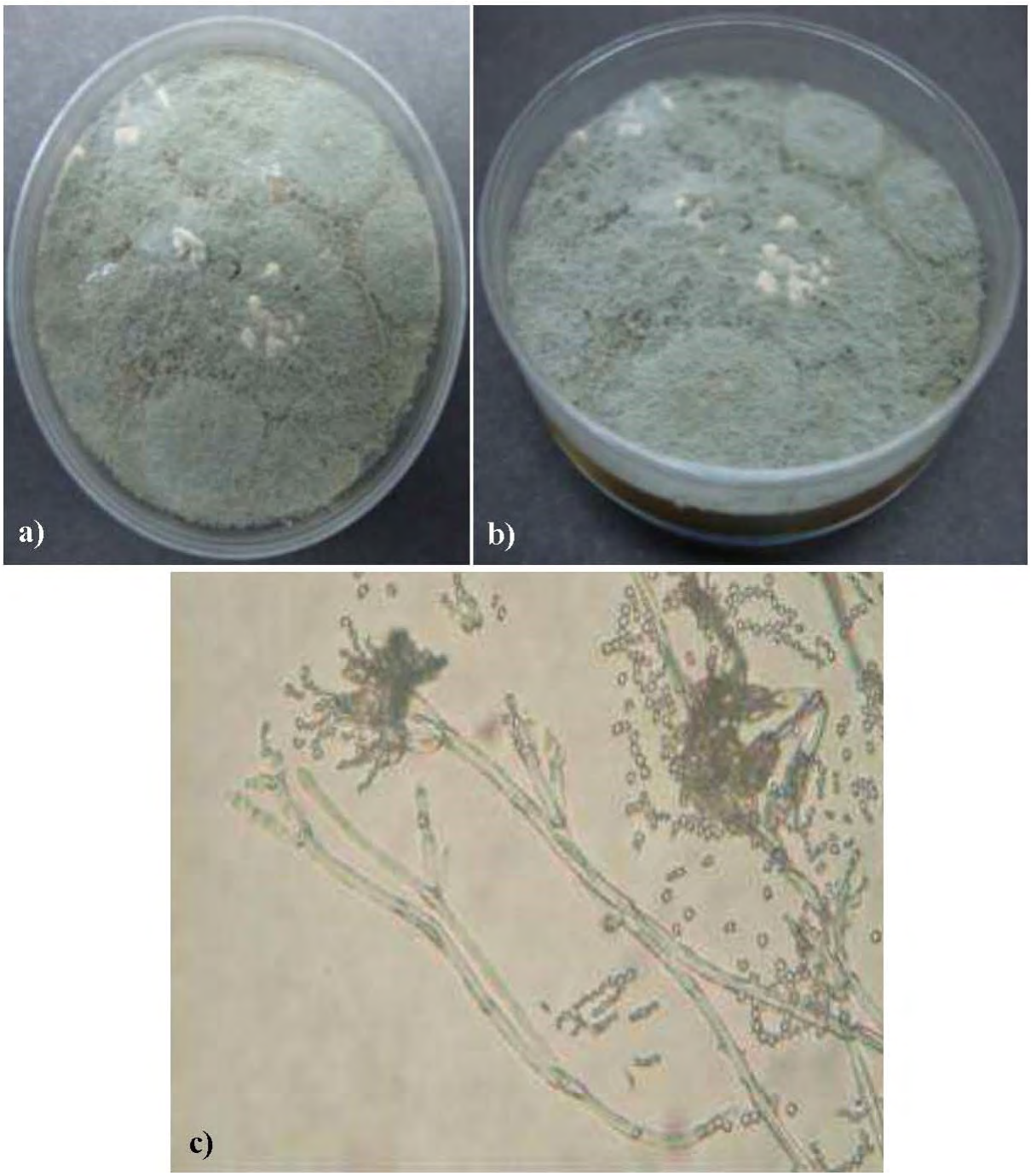
Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

sexual, quando ocorre, é realizada através da produção de ascósporos (TROVÃO; PEREIRA, 2019).

Do ponto de vista ecológico, são considerados fungos de solo que preferem climas frescos e moderados, com matéria orgânica disponível, mas também estão presentes no ar e em poeiras de ambientes internos (TROVÃO; PEREIRA, 2019).

Várias espécies do gênero *Penicillium* desempenham um papel essencial na produção de queijo, como, por exemplo, as espécies *Penicillium camemberti* e *Penicillium roqueforti*.

Figura 17 – a) e b) Exemplos de colônias de *Penicillium sp.*; c) exemplo de fíalide composta por ramificações, métula e conídios em *Penicillium sp.*

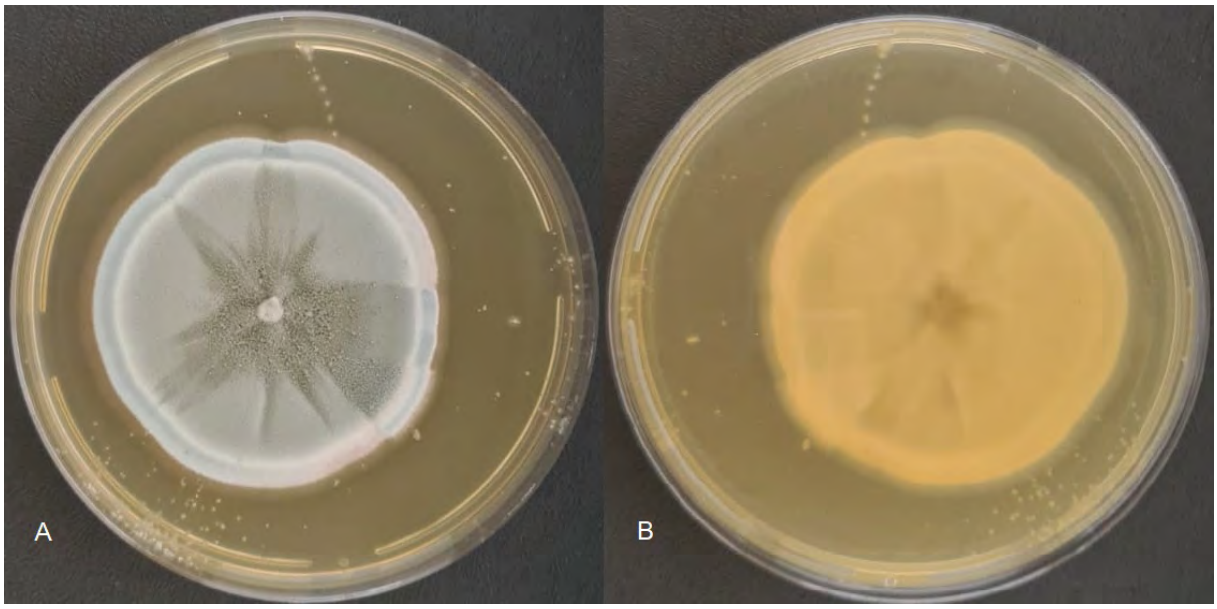


Fonte: (TROVÃO; PEREIRA, 2019).

Além da sua importância na indústria alimentar, possuem também um papel essencial na produção de enzimas e macromoléculas. Algumas espécies demonstram também potencial de uso na biorremediação (TROVÃO; PEREIRA, 2019).

O gênero inclui uma grande variedade de espécies principais produtoras de antibióticos. A penicilina, produzida por *P. chrysogenum* foi descoberta acidentalmente por Alexander Fleming em 1929. A sua potencial utilização como antibiótico foi testada no final da década de 1930 quando Howard Florey e Ernst Chain purificaram e concentraram o composto.

Figura 18 – Cultura em placa de *Petri* da espécie de fungo *Penicillium olsonii* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

Pelo enorme impacto deste antibiótico durante a Segunda Guerra Mundial, reduzindo o número de mortos por infecção de feridas, Fleming, Florey e Chain ganharam em conjunto o prémio Nobel de Medicina em 1945 (TROVÃO; PEREIRA, 2019).

Outras drogas com efeitos antimicóticos, como a griseofulvina, também foram extraídas de outras espécies, como por exemplo *P. griseofulvum*. Existem ainda espécies capazes de produzir moléculas com potencial aplicação na inibição de células tumorais e cancerígenas, pelo menos em testes laboratoriais (TROVÃO; PEREIRA, 2019).

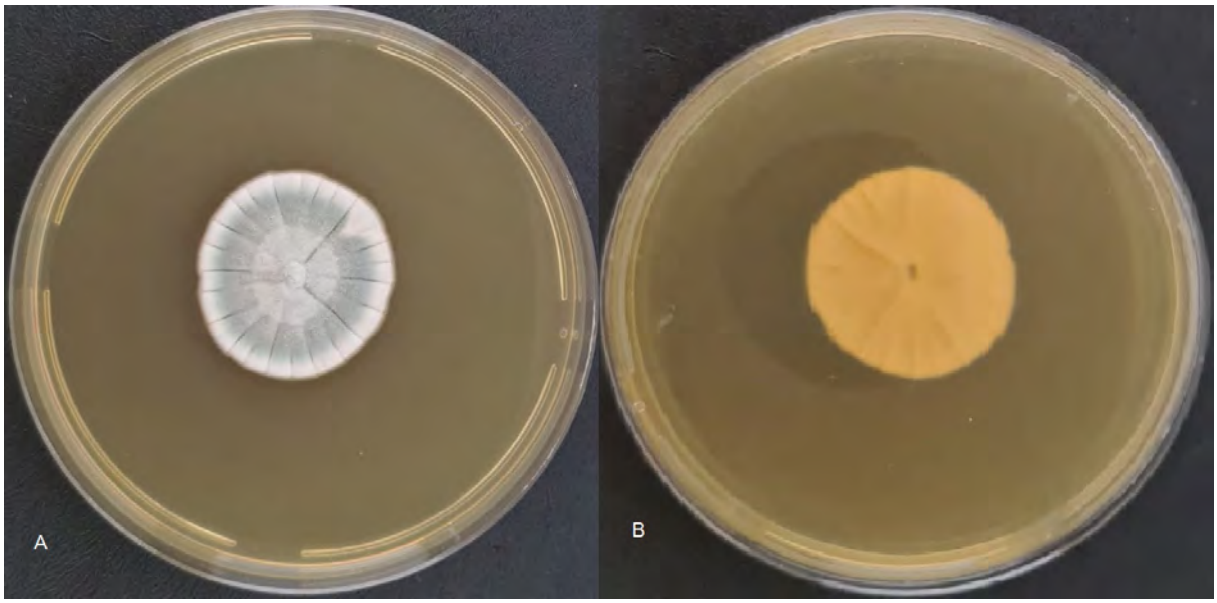
Não obstante, várias espécies são responsáveis pela degradação de alimentos processados e outras produzem várias micotoxinas severamente tóxicas. Algumas espécies afetam também algumas árvores Frutíferas tais como a macieira, a pereira e os citrinos. Outras espécies são ainda patogênicas para animais, tais como o *P. corylophilum*, o *P. fellutanum* ou o *P. implicatum*. Por fim, algumas espécies causam também alguns danos a maquinarias e aos materiais combustíveis e lubrificantes que estas utilizam (TROVÃO; PEREIRA, 2019).

A Figura 18 e a Figura 19 ilustram duas culturas em placa de *Petri* do gênero *Penicillium*. Utilizaram-se nesse trabalho as espécies *Penicillium olsonii* e *Penicillium steckii*, respectivamente. As amostras foram repicadas pelo Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021).

2.2.5 Breve descrição do Gênero *Rhizomucor*

Rhizomucor é um fungo filamentoso global encontrado em plantas e frutas em decomposição e no solo. *Rhizomucor spp.* são muitas vezes isolados da fermentação e composta-

Figura 19 – Cultura em placa de *Petri* da espécie de fungo *Penicillium steckii* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

Figura 20 – *Rhizomucor miehei*

Fonte: (MEDICINE, 2020).

gem de matéria orgânica, também são causas raras de infecções graves (e frequentemente fatais) em humanos. *Rhizomucor spp.*, exceção feita ao *Rhizomucor variabilis* são termofílicos por natureza e podem crescer a temperaturas tão altas quanto ˜54°C (MEDICINE, 2020).

O gênero *Rhizomucor* é composto por três espécies: *Rhizomucor pusillus*, *Rhizomucor miehei* e *Rhizomucor variabilis*. *Rhizomucor variabilis* é filogeneticamente muito próximo de *Mucor hiemalis*. A temperatura máxima de crescimento, o perfil de assimilação bioquímica, a dependência de tiamina e o diâmetro dos esporângios auxiliam na diferenciação das três espécies de *Rhizomucor*. *Rhizomucor miehei* é homotálico (autofértil), enquanto *Rhizomucor pusillus* é homo ou heterotálico (autoestéril) (MEDICINE, 2020).

Rhizomucor spp. estão entre os fungos que causam o grupo de infecções conhecidas como zigomicose. Embora o termo mucormicose tenha sido frequentemente usado para essa síndrome, atualmente a zigomicose é o termo mais comum para essa doença angioinvasiva. A invasão vascular que causa necrose do tecido infectado e a invasão perineural são as características mais severas dessas infecções. A zigomicose é frequentemente fatal (MEDICINE, 2020).

Existem poucos relatos de infecções humanas por *Rhizomucor spp.* Podem-se relatar as zigomicose cutânea, pulmonar, rinofacial que são provocadas pelo *Rhizomucor pusillus* em pacientes neutropênicos com neoplasias hematológicas e/ou diabetes mellitus. Digno de nota, embora raro, infecções cutâneas primárias devido ao *Rhizomucor variabilis* foram relatadas em indivíduos saudáveis. Ao contrário das raras infecções humanas, infecções em animais são comnuns especialmente no tocante ao aborto micótico bovino devido ao *Rhizomucor* (MEDICINE, 2020).

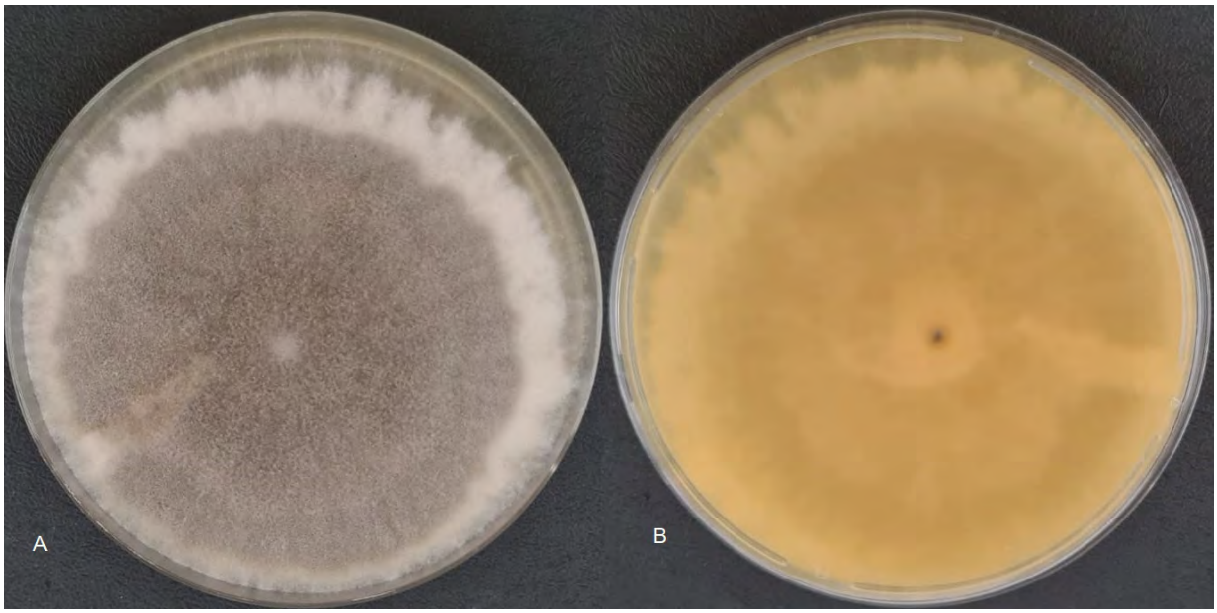
As colônias de *Rhizomucor* crescem muito rapidamente, preenchem a placa de *Petri* e amadurecem em 4 dias. A textura é tipicamente semelhante ao algodão doce. De frente, a cor da colônia é inicialmente branca e com o tempo muda de cinza para marrom amarelado. O reverso é de branco a pálido (MEDICINE, 2020).

A morfologia microscópica do *Rhizomucor* parece ser intermediária entre a de *Rhizopus* e *Mucor*. São visualizadas hifas largas não septadas ou esparsamente septadas, rizoides rudimentares, esporangióforos, esporângios e esporangiósporos. Os rizoides rudimentares, se existirem, são poucos em número e estão localizados em estolões entre os esporangióforos. Os esporangióforos são irregularmente ramificados e terminam com esporângios em seus ápices. Os esporângios (40-80 µm de diâmetro) são de cor marrom e formato redondo. A apófise está ausente. As columelas, por outro lado, são proeminentes e de formato esférico a piriforme. Esporangiósporos (3-4 µm de diâmetro) são pequenos, unicelulares e de formato redondo a elipsoidal. Os zigosporos, se presentes, são formados nas hifas aéreas. São redondos ou ligeiramente comprimidos e de cor castanha escura ou castanha enegrecida (MEDICINE, 2020).

O *Rhizomucor* difere do *Mucor* por crescer a 50-55°C e por ter rizoides e estolões; do *Rhizopus* por ter esporangióforos ramificados e rizoides que não surgem opostos aos esporangióforos; e do *Absidia* por ter esporângios globosos e esporangióforos que não são inchados onde se fundem com as columelas (MEDICINE, 2020).

Na Figura 21 mostra-se uma das culturas em placa de *Petri* da espécie *Rhizomucor*

Figura 21 – Cultura em placa de *Petri* da espécie de fungo *Rhizomucor pusillus* utilizada nesse trabalho. A - Vista do verso da placa de *Petri*, B - Vista do reverso da placa de *Petri*



Fonte: Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE.

pusillus utilizada nesse trabalho. A amostra foi repicada pelo Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021).

2.3 ASSINATURA DE ODOR DE ESPÉCIES DE FUNGOS

Os fungos filamentosos produzem uma ampla gama de *VOC*s. Álcoois, cetonas, terpenos, ésteres e os compostos de enxofre são os predominantes. A produção de voláteis é altamente dependente da espécie e do meio onde ela se encontra (KUSKE; ROMAIN; NICOLAS, 2005).

(GARCIA-ALCEGA et al., 2017) comenta que os avanços recentes nas técnicas analíticas abrem uma nova porta para a caracterização química do bioaerossol. Especificamente, a análise química de *MVOC*s pode ser uma avaliação confiável e rápida da natureza dos bioaerossóis ambientais, pois as comunidades microbianas expressam diferentes perfis de *MVOC*s, a depender do ambiente em que estejam. Além disso, foi demonstrado que os voláteis específicos da espécie podem servir como compostos marcadores para a detecção seletiva de espécies microbianas patogênicas em ambientes internos e externos. *MVOC*s são metabólitos secundários produzidos pela fermentação e são voláteis devido às suas propriedades físico-químicas (baixo peso molecular, baixo ponto de ebulição e alta pressão de vapor). Caracterizar e quantificar *MVOC*s também pode ser usado como uma abordagem *proxy* para estimar a concentração microbiana. A análise de *MVOC*s foi aplicada na área de saúde, por exemplo, para diagnosticar a doença de *Crohn* a partir da urina ou para detectar *Aspergillus fumigatus* invasivo a partir de amostras respiratórias. Esses

Figura 22 – Tabela de *MVOC*s - *Aspergillus niger*, *Cladosporium sp.*, *Mucor plumbeus* e *Penicillium spp.*

Concentrations of VOCs produced by contaminant fungi in 22 ml headspace vials containing strawberry jam agar

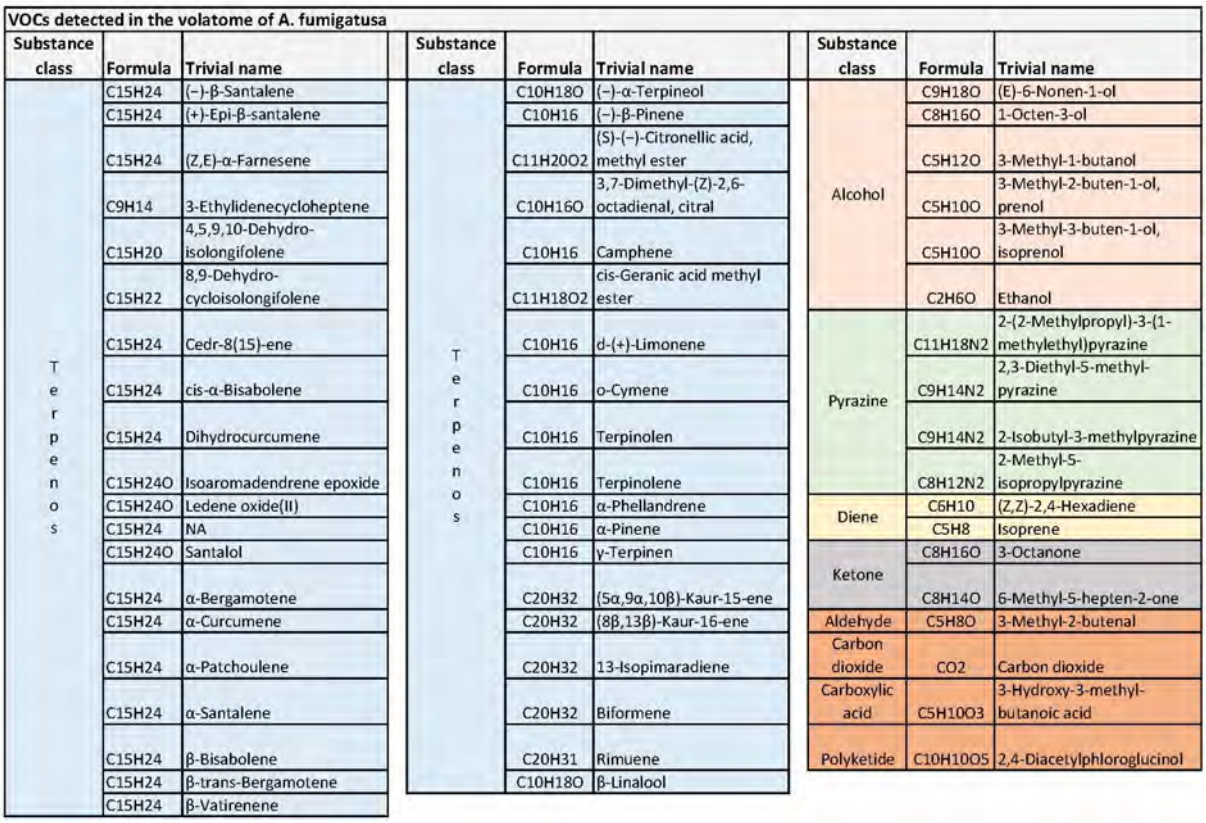
	VOC							
	3-methyl-1-butanol		2-pentanone		1,3-pentadiene		Styrene	
Incubation time (days) →	2	6	2	6	2	6	2	6
Fungal strain								
Non-inoculated strawberry jam agar	<20	<20	<4	<4	<0.9	<0.9	<2	<2
Aspergillus niger VH10	30	210	90	95	61	36	640	460
Emericella sp. VH12	<20	80	69	85	45	52	570	520
Cladosporium sp. 95/113	<20	<20	51	50	<0.9	<0.9	<2	<2
Mucor plumbeus AK1	5200	—[a]	66	—[a]	<0.9	—[a]	<2	—[a]
Penicillium sp. S645H	130	340[b]	110	51[b]	57	47[b]	670	540[b]
Penicillium sp. VH11	70	600	110	110	58	44[c]	630	540
Penicillium sp. VS13	<20	220	38	54	17	38	320	580
Penicillium sp. VS14	120	120	52	45	37	31	590	520
Penicillium sp. VS21	<20	190	33	45	33	45	530	610
Trichoderma sp. VS20	100	290	34	43	35	18[c]	710	540

Fonte: (NIEMINEN et al., 2008).

fungos também têm sido usados para a detecção de explosivos, drogas e em instalações de compostagem. A concentração de *MVOC*s no ambiente pode ser altamente variável entre locais devido a vários fatores, incluindo, entre outros, a fonte microbiana (substrato), a distância da fonte, as condições climáticas, a direção do vento e a topografia da paisagem.

A seguir, ilustram-se com exemplos de *MVOC*s de algumas espécies de gêneros de interesse desse trabalho nas Figura 22, Figura 23, Figura 24 e Figura 25.

Figura 23 – Tabela de *MVOC*s - *Aspergillus fumigatus*. Produzidos em meio de cultura Brian, em meio de cultura suplementado com ferro, em cultura aerada e em cultura com presença de alendronato



VOCs detected in the volatome of A. fumigatusa

Substance class	Formula	Trivial name
Terpenos	C15H24	(−)-β-Santalene
	C15H24	(+)-Epi-β-santalene
	C15H24	(Z,E)-α-Farnesene
	C9H14	3-Ethylidenecycloheptene
	C15H20	4,5,9,10-Dehydro-isolongifolene
	C15H22	8,9-Dehydro-cycloisolongifolene
	C15H24	Cedr-8(15)-ene
	C15H24	cis-α-Bisabolene
	C15H24	Dihydrocurcumene
	C15H24O	Isoaromadendrene epoxide
	C15H24O	Ledene oxide(II)
	C15H24	NA
	C15H24O	Santalol
	C15H24	α-Bergamotene
	C15H24	α-Curcumene
	C15H24	α-Patchoulene
	C15H24	α-Santalene
	C15H24	β-Bisabolene
	C15H24	β-trans-Bergamotene
	C15H24	β-Vatirenene

Substance class	Formula	Trivial name
Terpenos	C10H18O	(−)-α-Terpineol
	C10H16	(−)-β-Pinene
	C11H20O2	(S)-(−)-Citronellic acid, methyl ester
	C10H16O	3,7-Dimethyl-(Z)-2,6-octadienal, citral
	C10H16	Camphene
	C11H18O2	cis-Geranic acid methyl ester
	C10H16	d-(+)-Limonene
	C10H16	o-Cymene
	C10H16	Terpinolen
	C10H16	Terpinolene
	C10H16	α-Phellandrene
	C10H16	α-Pinene
	C10H16	γ-Terpinen
	C20H32	(5α,9α,10β)-Kaur-15-ene
	C20H32	(8β,13β)-Kaur-16-ene
	C20H32	13-Isopimaradiene
	C20H32	Biformene
	C20H31	Rimuene
	C10H18O	β-Linalool

Substance class	Formula	Trivial name
Alcohol	C9H18O	(E)-6-Nonen-1-ol
	C8H16O	1-Octen-3-ol
	C5H12O	3-Methyl-1-butanol
	C5H10O	3-Methyl-2-buten-1-ol, prenol
	C5H10O	3-Methyl-3-buten-1-ol, isoprenol
	C2H6O	Ethanol
Pyrazine	C11H18N2	2-(2-Methylpropyl)-3-(1-methylethyl)pyrazine
	C9H14N2	2,3-Diethyl-5-methyl-pyrazine
	C9H14N2	2-Isobutyl-3-methylpyrazine
	C8H12N2	2-Methyl-5-isopropylpyrazine
Diene	C6H10	(Z,Z)-2,4-Hexadiene
	C5H8	Isoprene
Ketone	C8H16O	3-Octanone
	C8H14O	6-Methyl-5-hepten-2-one
Aldehyde	C5H8O	3-Methyl-2-butenal
Carbon dioxide	CO2	Carbon dioxide
Carboxylic acid	C5H10O3	3-Hydroxy-3-methyl-butanoic acid
Polyketide	C10H10O5	2,4-Diacetylphloroglucinol

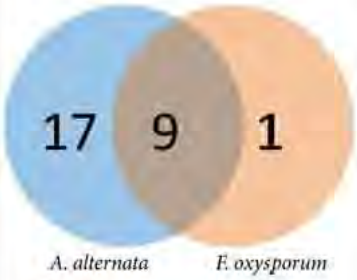
Fonte: (HEDDERGOTT; CALVO; LATGE, 2014).

Figura 24 – Tabela de *MVOC*s - *Fusarium spp.*

VOCs de fungos *Fusarium* altamente voláteis, crescendo em dois substratos de nutrientes: agar de sacarose de batata e grãos de trigo autoclavados		
VOCs	**VOCs**	**VOCs**
EtOH	2-Ethylfuran	Ethyl 3-methylbutanoate
Acetone	Methylcyclohexane	4-Methyloctane
Pentane	3-Methyl-1-butanol	3-Methylbutyl acetate
Methyl acetate	2-Methyl-1-butanol	2-Methylbutyl acetate
1-Propanol	1-Methyl-1H-pyrrol	1-Nonene
3-Methylpentane	4-Methylheptane	Nonane
Hexane	1,3,5-Cycloheptatriene	α-Pinene
2-Methylfuran	2-Methylpropyl acetate	β-Pinene
3-Methylfuran	Octane	β-Myrcene
AcOEt	Ethyl butanoate	2-Pentylfuran
iBuOH	2,4-Dimethylheptane	Limonene
4-Methyl-1,3-pentadiene	1,3,5-Trimethylcyclohexane	1-Undecene
3-Methylbutanal	2,4-Dimethyl-1-heptene	α-Terpinolene
2-Methylbutanal	Ethyl 2-methylbutanoate	Undecane
Heptane		

Fonte: (SAVELIEVA et al., 2016).

Figura 25 – Tabela da origem fúngica de *MVOC*s (acima do limite de quantificação (pmol placa^{-}1 h^{-}1)) coletadas em culturas solitárias de *Alternaria alternata* e *Fusarium oxysporum* cultivadas em meio rico em nutrientes (gelrite de extrato de malte).

Venn plot	VOC unique to *Alternaria alternata* (16)	VOC detected from both species (9)	VOC unique to *Fusarium oxysporum* (1)
	Di-epi-α-Cedrene		
	α-Cedrene		
	Thujopsene		
	(E)-β-Farnesene	Isobutanol[a]	
	unknown SQT #1	1-Butanol, 2-methyl[a]	
	β-Chamigrene[a]	β-Elemene	
	allo-Aromadendrene	β-Cedrene	
	Eremophilene[a]	β-Acoradiene	δ-Elemene
	α-Chamigrene	α-Himachalene	
	γ-Cadinene	unknown Sqt #2[b]	
	unknown Sqt #3	γ-Curcumene	
	Thujopsane-2β-ol	Germacrene D[b]	
	(+unknown Sqt #4)		
	Cedren-13-ol, 8-[a]		
	Widdrol		
	10-epi-γ-Eudesmol		

[a]absent on nutrient poor medium, [b]on nutrient poor medium only detected for *A. alternata*.

Fonte: (WEIKL et al., 2016).

2.4 BREVE INTRODUÇÃO ÀS SÉRIES TEMPORAIS

Segundo (MARTIN et al., 2016), as séries temporais (ou históricas) são conjuntos de medidas de uma mesma grandeza, relativas a vários períodos consecutivos. Ou seja, a série temporal é uma sucessão de valores de uma determinada variável observada em intervalos regulares de tempo. A variável de controle é o tempo e as séries temporais são ordenadas cronologicamente e se variar a ordem pode modificar a informação contida na série. Podem ser coletados em intervalos regulares de tempo, e podem ser observações diárias, mensais, trimestrais, anuais, entre outros. Como exemplo para as séries temporais, é possível citar o preço das ações, valores de exportações, Produto Interno Bruto (PIB), temperatura, vendas médias de determinado item, temperatura média, batimentos cardíacos, enfim, uma infinidade de séries históricas e ordenadas cronologicamente podem ser reconhecidas como séries temporais.

Também (MARTIN et al., 2016) cita quatro principais aplicações para a previsão por intermédio de séries temporais: planejamento econômico e de negócios, planejamento de produção, inventário e controle de produção e, por fim, controle e otimização de processos industriais. Realizar previsões por meio de um método estatístico reduz o grau de imprecisão sobre os valores futuros, o que auxilia na tomada de decisões. Ainda de acordo com (MARTIN et al., 2016), a demanda futura será uma projeção dos valores pretéritos e não sofre influência de outras variáveis. Pode-se dizer que uma das limitações da série temporal é considerar apenas as observações passadas para realizar as previsões, sem considerar as variações causais.

As séries temporais possuem três padrões básicos: tendência, sazonalidade e ciclo. A tendência ocorre quando se verifica que os dados crescem ou diminuem ao longo do tempo. Podem existir casos em que a tendência muda de direção, por exemplo, ir de uma tendência crescente para uma decrescente. Já as séries que permanecem constantes ao longo do tempo não possuem tendência (MARTIN et al., 2016).

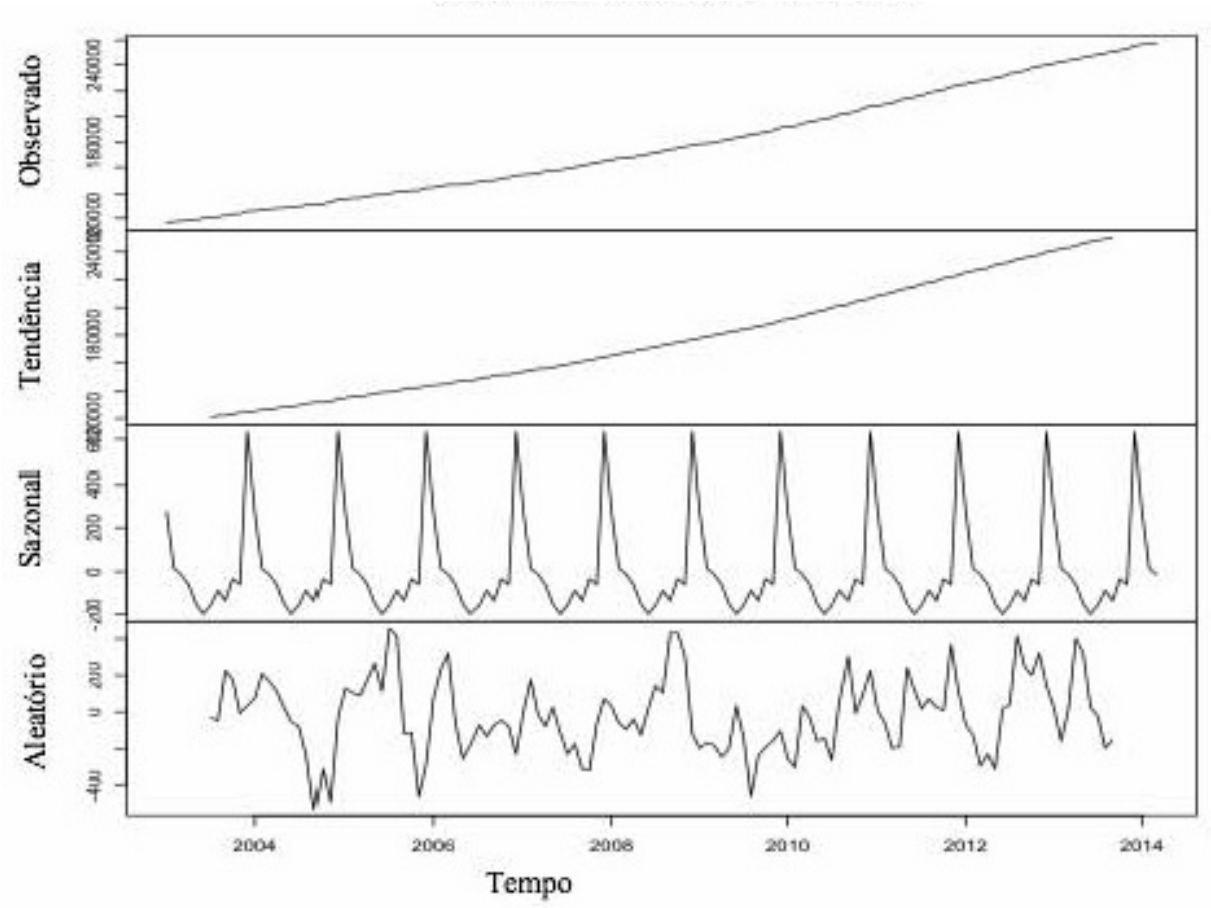
A sazonalidade ocorre quando a série sofre influência de fatores sazonais, por exemplo, o semestre, mês ou semana do ano, ou ainda de eventos, sendo o período sempre conhecido. Diferentemente da sazonalidade, as variações cíclicas ocorrem em períodos não conhecidos. As variações sazonais são movimentos cíclicos que se completam em um ano enquanto as variações cíclicas são movimentos cíclicos que se completam em período superior a um ano (MARTIN et al., 2016).

Existe outro componente, o irregular, pelo fato de haver algum movimento que não é explicável por tendência ou ciclos. Essas variações irregulares ocorrem por acaso e contribuem para aumento ou queda de valores da série, e a contribuição para o acontecimento pode ser, por exemplo, de sobretaxas alfandegárias ocasionais e até guerras (MARTIN et al., 2016).

A Figura 26 ilustra a decomposição de séries temporais.

As séries temporais também são encontradas nas áreas de engenharia, ciência, socio-

Figura 26 – Decomposição de séries temporais.



Fonte: (MARTIN et al., 2016).

logia e estatística, entre outras. Após a escolha de uma família apropriada de modelos, é, então, possível estimar parâmetros, verificar a adequação aos dados e possivelmente usar o modelo ajustado para melhorar nossa compreensão do mecanismo que gera a série. Uma vez desenvolvido um modelo satisfatório, ele pode ser usado de várias maneiras, dependendo do campo de aplicação particular (BROCKWELL; DAVIS, 2016).

O modelo pode ser utilizado simplesmente para fornecer uma descrição compacta dos dados. Pode-se, por exemplo, ser capazes de representar os dados de mortes acidentais, como a soma de uma tendência específica, e termos sazonais e aleatórios. Para a interpretação das estatísticas econômicas, tais como números de desemprego, é importante reconhecer a presença de componentes sazonais e removê-los para não os confundir com tendências de longo prazo. Este processo é conhecido como ajuste sazonal. Outras aplicações dos modelos de séries temporais incluem a separação (ou filtragem) do ruído dos sinais, previsão de valores futuros de uma série, como as vendas de vinho tinto ou os dados da população, testar hipóteses como o aquecimento global, usando dados de temperatura registrados, prever uma série a partir de observações de outra, por exemplo, prever vendas futuras usando dados de gastos com publicidade, e controlar valores futuros de uma série ajustando parâmetros. Os modelos de séries temporais também são úteis em estudos

de simulação. Por exemplo, o desempenho de um reservatório depende muito das entradas diárias aleatórias de água para o sistema. Se estes forem modelados como uma série cronológica, então pode-se usar o modelo ajustado para simular um grande número de sequências independentes de entradas diárias. Sabendo o tamanho e o modo de operação do reservatório, pode-se determinar a fração das sequências de entrada simuladas que fazem com que o reservatório fique sem água em um determinado período de tempo. Esta fração será, então, uma estimativa da probabilidade de esvaziamento do reservatório em algum momento do período em questão (BROCKWELL; DAVIS, 2016).

A principal característica que deve possuir uma série temporal é ser estacionária, ou seja, além de ser estocástico, o processo deve estar em equilíbrio em relação a uma média e com variância constante. Uma série não estacionária permite o estudo do seu comportamento apenas no período considerado, ou seja, não é possível utilizar para outros períodos, tornando-se de pouco valor para realizar previsões. Dificilmente as séries temporais são estacionárias, por exemplo, as séries financeiras apresentam tendências e, então, deve-se agir sobre os dados para convertê-las em estacionárias. A partir de diferenciações, é possível tornar a série estacionária (MARTIN et al., 2016).

Para analisar as séries temporais e verificar como as observações futuras são influenciadas pelas do passado, utilizam-se as funções de autocorrelação amostral (ACF) e autocorrelação amostral parcial (PACF). A ACF (ou correlograma) proporciona a estrutura de dependência linear da série, ou seja, como uma observação influência sobre as posteriores. Já a PACF (ou correlograma parcial) mostra o grau de associação linear direta entre observações separadas por k períodos. Através do correlograma, é possível verificar as seguintes propriedades das séries temporais: aleatoriedade, sazonalidade, correlação e estacionariedade. Já a tendência pode ser observada por meio do gráfico da série (MARTIN et al., 2016).

Outras áreas de aplicações mais próximas ao problema da dissertação são: visão computacional, matemática aplicada e processamento de sinais. Conforme comentado, uma série temporal é uma coleção de observações feitas sequencialmente ao longo do tempo. Em modelos de regressão linear com dados *cross-section*, a ordem das observações é irrelevante para a análise. Em séries temporais a ordem dos dados é fundamental. Uma característica muito importante deste tipo de dados é que as observações vizinhas são dependentes e o interesse é analisar e modelar essa dependência.

Na maioria das áreas, as aplicações de séries temporais são previsões de séries temporais, isto é, dado um recorte de tempo passado de uma série temporal, o problema é prever um tempo, um ciclo futuro. Já o problema dessa dissertação é o de classificação dessas séries temporais, ou seja, modelos de Aprendizagem de Máquina (AM) aprenderem com uma base séries temporais rotuladas e posteriormente serem capazes de classificar casos novos.

2.5 CONCLUSÕES E PRÓXIMOS PASSOS

Neste capítulo, foram discutidos os conhecimentos básicos acerca dos fungos anemófilos, da Inteligência Artificial, da Aprendizagem de Máquina e das Séries Temporais, afim de que seja possível conhecer os dados levantados na pesquisa com profundidade, bem como os fundamentos que regem os modelos que serão testados para resolver o problema da dissertação.

Os próximos passos serão revisar a literatura para situar esse trabalho em relação ao que está acontecendo em áreas correlatas.

3 REVISÃO DA LITERATURA

Este capítulo se dedica a uma breve revisão da literatura sobre o tema de interesse, abordando trabalhos relacionados nas áreas de detecção e identificação de fungos e classificação de séries temporais.

3.1 TRABALHOS RELACIONADOS EM DETECÇÃO E IDENTIFICAÇÃO DE FUNGOS

No artigo (MOTA; TEIXEIRA-SANTOS; RUFO, 2021), os autores realizaram uma revisão sistemática da literatura de acordo com as diretrizes PRISMA. Um total de 16 artigos atenderam aos critérios do estudo e foram incluídos na análise. Os resultados dos estudos revisados demonstraram que a detecção efetiva de fungos foi possível por meio de sistemas de narizes eletrônicos baseados em sensores, que podem realmente funcionar como uma ferramenta de triagem de *MVOC*s para diversas aplicações (MOTA; TEIXEIRA-SANTOS; RUFO, 2021).

Uma identificação rápida e eficaz de espécies de fungos é essencial para inúmeras aplicações, e os sistemas de narizes eletrônicos estão sendo propostos como alternativas adequadas às técnicas de identificação de fungos atualmente disponíveis. Assim, a revisão realizada em (MOTA; TEIXEIRA-SANTOS; RUFO, 2021) visa desvendar as informações publicadas sobre a identificação de fungos por sistemas de narizes eletrônicos.

Os resultados obtidos sugerem que os sistemas de narizes eletrônicos baseados em sensores podem não apenas rastrear diferentes gêneros de fungos, mas também identificar as espécies associadas. Essa tecnologia já foi experimentada em diversos campos, desde a indústria alimentícia até a prática clínica (MOTA; TEIXEIRA-SANTOS; RUFO, 2021).

Ao resumir esses resultados, a revisão pode acelerar a padronização de narizes eletrônicos na detecção e discriminação de fungos, permitindo uma triagem de amostras mais rápida e eficientemente (MOTA; TEIXEIRA-SANTOS; RUFO, 2021).

No trabalho (LOULIER et al., 2020), os autores pesquisaram sobre fungos e oomicetos que liberam *VOC*s em seus ambientes e que podem ser usados para detecção olfativa e identificação desses organismos por narizes eletrônicos (*e-Nose*). O objetivo do estudo foi pesquisar a emissão de *VOC*s usando um dispositivo *e-Nose* e identificar moléculas liberadas por meio de análise de microextração de fase sólida-cromatografia gasosa/ espectrometria de massa (SPME-GC/MS), para desenvolver um sistema de detecção para fungos e microrganismos semelhantes. Para tanto, culturas de oito fungos (*Armillaria gallica, Armillaria ostoyae, Fusarium avenaceum, Fusarium culmorum, Fusarium oxysporum, Fusarium poae, Rhizoctonia solani, Trichoderma asperellum*) e quatro oomicetos (*Phytophthora cactorum, P. cinnamomi, P. plurivora, P. ramorum*) foram testados com o sistema *e-Nose* e investigados por meio de SPME-GC/MS. As estirpes de *F. poae, R.*

solani e *T. asperellum* pareceram ser as mais odoríferas. Todas as espécies de fungos investigadas (exceto *R. solani*) produziram sesquiterpenos em quantidades variáveis, em contraste com as cepas de oomicetos testadas. Outras moléculas, como hidrocarbonetos alifáticos, álcoois, aldeídos, ésteres e derivados de benzeno, foram encontradas em todas as amostras. Os resultados sugeriram que as principais diferenças entre as respectivas faixas de emissão de *VOC*s das espécies testadas estão na produção de sesquiterpeno, com alguns fungos emitindo enquanto os oomicetos não liberam nenhuma ou quantidades menores de tais moléculas. O sistema de nariz eletrônico discriminou os odores emitidos por *P. ramorum, F. poae, T. asperellum e R. solani*, o que respondeu por mais de 88% da variância do PCA. Muitos fungos fitopatogênicos e espécies de oomicetos que causam tombamento e doenças de podridão radicular são conhecidos por emitir vários metabólitos secundários na forma de compostos orgânicos voláteis *VOC*s, que podem ser específicos de gênero ou espécie. Esses resultados preliminares da detecção de fungos e oomicetos tornam o dispositivo *e-Nose* adequado para outros projetos de sensores como uma ferramenta potencial para gerentes florestais, outros gerentes de plantas, bem como agências reguladoras, como serviços de quarentena (LOULIER et al., 2020).

Na pesquisa (HUNG; LEE; BENNETT, 2015), os autores explicam que todos os odorantes são *VOC*s, ou seja, compostos de baixo peso molecular que evaporam facilmente em temperaturas e pressões normais. *VOC*s fúngicos são relativamente pouco estudados em comparação com *VOC*s de origem bacteriana, vegetal ou sintética. Grande parte da pesquisa até hoje sobre *VOC*s fúngicos se concentrou em suas propriedades alimentares e de sabor, bem como no seu uso como indicadores indiretos do crescimento de fungos na agricultura ou no seu papel como semioquímicos para insetos. Além disso, a pesquisa de voláteis fúngicos também ocorreu para monitorar a deterioração, para fins de quimiotaxonomia, para uso em biofiltros e para biodiesel, afim de detectar doenças de plantas e animais, para "micofumigação", e garantindo a saúde das plantas. À medida que os métodos para a análise de moléculas em fase gasosa melhoraram, tornou-se evidente que os *VOC*s fúngicos são quimicamente mais variados e biologicamente mais ativos do que geralmente se imaginava. Em particular, há dados crescentes que mostram que os *VOC*s fúngicos frequentemente medem interações entre organismos dentro e através de diferentes nichos ecológicos. O objetivo da minirevisão é orquestrar dados sobre *VOC*s fúngicos obtidos de diferentes disciplinas, bem como chamar a atenção para a importância ecológica dos *VOC*s fúngicos na sinalização de diferentes espécies. Tecnologias e abordagens que são comuns em uma área de pesquisa são, muitas vezes, desconhecidas em outras, e o estudo de *VOC*s fúngicos se beneficiaria de mais discussões cruzadas entre as subdisciplinas (HUNG; LEE; BENNETT, 2015).

Dentre as técnicas de detecção de *VOC*s, são citados estudos arcaicos, como destilação a vapor, juntamente com extração líquido-líquido, concentração subsequente e, em seguida, identificação química laboriosa de *VOC*s concentrados individuais, muitas vezes

com a intenção de entender os aromas de cogumelos (CRONIN; WARD, 1971). Até métodos atuais como cromatografia gasosa-espectrometria de massas (GC-MS), variações no protocolo de identificação incluem espectrometria de massa de reação de transferência de prótons (PTR-MS) ou a espectrometria de massa de tubo de fluxo de íons selecionada (SIFT-MS) ou ainda acoplando uma técnica de extração por sorção de headspace com cromatografia gasosa-tempo de espectrometria de massa de voo (GC-TOFMS). A microextração em fase sólida (SPME) é um avanço importante. O "e-nose" é um dispositivo que utiliza a assinatura eletrônica exclusiva que diferentes compostos produzem quando interagem com várias superfícies eletrônicas e é útil para aplicações específicas com VOCs alvos conhecidos (HUNG; LEE; BENNETT, 2015).

No artigo (TAŞTAN; GÖKOZAN, 2019) os autores comentam que nos dias atuais, a poluição do ar é um grande problema de saúde ambiental mundial. A poluição do ar leva a efeitos adversos na saúde humana, no clima e nos ecossistemas. O ar está contaminado por gases tóxicos liberados pela indústria, por emissões veiculares e pelo aumento da concentração de gases nocivos e material particulado na atmosfera. A poluição do ar pode causar muitos problemas graves de saúde, como doenças respiratórias, cardiovasculares e de pele em humanos. Atualmente, onde a poluição do ar se tornou o maior risco à saúde ambiental, o interesse em monitorar a qualidade do ar está aumentando. Recentemente, tecnologias móveis, especialmente a Internet das Coisas, tecnologias de dados e aprendizado de máquina têm um impacto positivo na maneira como se pode gerenciar a saúde. Com a produção de dispositivos portáteis de medição da qualidade do ar baseados em *Internet of Things* (IoT) e com seu uso generalizado, as pessoas podem monitorar a qualidade do ar instantaneamente. No estudo, o *e-nose*, um sistema móvel de monitoramento da qualidade do ar em tempo real, com vários parâmetros do ar, como CO_2, CO, NO_2, temperatura e umidade, é proposto. O nariz eletrônico proposto é produzido com uma abordagem de código aberto, baixo custo, fácil instalação e faça você mesmo. Os dados de qualidade do ar medidos pelo conjunto de sensores GP2Y1010AU, MH-Z14, MICS-4514 e DHT22 podem ser monitorados através do controlador Wi-Fi ESP32 de 32 bits e da interface móvel desenvolvida pela plataforma Blynk IoT, e os dados recebidos são registrados em um servidor em nuvem. Após a avaliação dos resultados obtidos nas medições internas, foi demonstrado que a diminuição da qualidade do ar interno foi influenciada pelo número de pessoas na casa e pelas emissões naturais devido a atividades como dormir, limpar e cozinhar. No entanto, observa-se que mesmo a ventilação natural manual diária tem um efeito significativo na melhoria da qualidade do ar (TAŞTAN; GÖKOZAN, 2019).

3.2 TRABALHOS RELACIONADOS EM CLASSIFICAÇÃO DE SÉRIES TEMPORAIS

No artigo (DENG et al., 2013), os autores propõem um classificador de ensemble de árvores: *TSF*. O *TSF* emprega uma nova medida chamada de Ganho de entrada (entropia e distância) para identificar divisões de alta qualidade. O artigo mostra que o *TSF* usa

o ganho de entrada e também dois algoritmos *One-neares t-neighbor with dynamic time warping* (*NNDTW*). Usando uma estratégia de amostragem de características aleatórias, o *TSF* tem complexidade linear no comprimento da série temporal. Além disso, (DENG et al., 2013) propõem a curva de importância temporal para capturar as características informativas para *Time Series Classification* (*TSC*).

No artigo (LINES; TAYLOR; BAGNALL, 2018), os autores discorrem sobre uma avaliação experimental recente que avaliou 19 algoritmos de *TSC*, na qual foi descoberto que um classificador mais preciso do que todos os outros até aqui: o *Flat Collective of Transformation-based Ensembles* (*Flat-CODE*). *Flat-CODE* é um ensemble que combina 35 classificadores em quatro representações de dados. No entanto, a avaliação não considerou abordagens de aprendizagem profunda. *Convolutional Neural Network* (*CNN*)s são o estado da arte em muitos campos e levanta a questão: se as *CNN*s poderiam ser igualmente transformadoras para o *TSC* (LINES; TAYLOR; BAGNALL, 2018).

Os autores implementaram uma *CNN* de referência para *TSC* usando uma estrutura básica e usaram os resultados de uma *CNN* específica para *TSC* da literatura. Compararam ambos com o *Flat-CODE* e descobriram que o ensemble é significativamente mais preciso do que as duas *CNN*s. Esses resultados são consistentes, mas o *Flat-CODE* não está isento de deficiências. Os autores aprimoraram significativamente o ensemble, propondo uma nova estrutura hierárquica com votação probabilística, definindo e incluindo dois novos classificadores de ensemble construídos em espaços de recursos existentes e adicionando mais módulos para representam dois domínios de transformação adicionais. O classificador resultante, o *HIVE-COTE V1*, encapsula classificadores construídos em cinco representações de dados. Os autores demonstraram que o *HIVE-COTE V1* é significativamente mais preciso que *Flat-CODE* (e todos os outros algoritmos *TSC* dos quais se têem conhecimento) mais de 100 reamostras de 85 problemas de *TSC*, configunando o novo estado da arte para *TSC*. A análise adicional está inclusa na introdução e na avaliação de 3 novos estudos de caso, bem como na extensa experimentação em 1000 conjuntos de dados simulados de 5 tipos diferentes (LINES; TAYLOR; BAGNALL, 2018).

No artigo (LINES; TAYLOR; BAGNALL, 2018), os autores apresentam o *RISE* que se apóia em ideias de conjuntos firmados em árvores, como floresta aleatória e o intervalo de floresta de série temporal de classificador de recursos *TSF* (DENG et al., 2013). Como o *TSF*, as árvores são construídas em intervalos dos dados para construir um classificador de floresta aleatório. A principal diferença, no entanto, é que o *TSF* usa recursos de domínio do tempo calculando a média, a variância e a inclinação de cada intervalo, mas *RISE* extrai características espectrais sobre cada intervalo aleatório. Selecioam-se 500 intervalos aleatórios e calculam-se características espectrais para cada intervalo de forma independente. Treina-se uma árvore de decisão separada em cada conjunto de recursos e, em seguida, combinam-se as árvores em uma floresta. O ensemble resultante contém 500 classificadores básicos que são diversificados por meio da seleção de intervalo. Adici-

onalmente, a primeira árvore em *RISE* é um caso especial que usa toda a série (LINES; TAYLOR; BAGNALL, 2018).

No artigo (LEE et al., 2012), os autores descrevem que muitas aplicações interessantes envolvem previsões baseadas em uma sequência de séries temporais ou em um conjunto de sequências de séries temporais, que são chamadas de problemas de classificação de séries temporais. A pesquisa de análise de classificação anterior concentra-se predominantemente na construção de um modelo de classificação de instâncias de treinamento que envolvem atributos não temporais. A aplicação direta de técnicas tradicionais de análise de classificação a problemas de classificação de séries temporais requer a transformação de atributos de séries temporais em atributos não temporais, aplicando algumas operações estatísticas (por exemplo, média, soma, variância). No entanto, essa abordagem baseada em transformação estatística geralmente resulta em perda de informações e, por sua vez, põe em risco a eficácia da classificação. No estudo, (LEE et al., 2012) propõem uma técnica de classificação de séries temporais baseada na abordagem de classificação de *kNN*. Usando previsão de churn do setor de telecomunicações móveis, como aplicativo de avaliação, os resultados de avaliação empírica mostraram que a técnica de classificação de séries temporais baseada em *kNN-Based Time-Series Classification* (*kNN-TSC*) alcança melhor desempenho (medido por taxas de erros e alarmes falsos) do que a abordagem baseada em transformação estatística (LEE et al., 2012).

Os autores do artigo (MIDDLEHURST; VICKERS; BAGNALL, 2019) discutem que classificadores baseados em dicionário são uma família de algoritmos para *TSC* que se concentra na captura da frequência de ocorrências de padrões em uma série de tempo. O ensemble baseado em *Bag of Symbolic Fourier Approximation Symbols* (*BOSS*) foi considerado um dos melhores modelos de *TSC* em uma avaliação recente, bem como entre os melhores classificadores baseados em dicionário. No entanto, o *BOSS* não tem um bom escalonamento. Os autores avaliaram mudanças na forma como o *BOSS* escolhe classificadores para seu ensemble, substituindo sua pesquisa de parâmetro com seleção aleatória. Esta mudança permite fácil implementação da contratação (definindo um limite de tempo de construção para o classificador) e de verificação (salvando o progresso durante a construção dos classificadores). Foi alcançada uma redução significativa no tempo de construção sem uma significativa mudança na precisão, em média, quando comparada ao *BOSS*, criando um conjunto ponderado de tamanho fixo, selecionando os melhores resultados de um conjunto de parâmetros escolhido. Os experimentos foram conduzidos em conjuntos de dados do arquivo recentemente expandido de série temporal UCR. Foram demonstradas melhorias de usabilidade para *BOSS* randomizado com um estudo de caso usando um grande conjunto de dados de acústica de baleias para o qual o *BOSS* se mostrou inviável. Os autores chamaram esse conjunto de melhorias e avanços em relação ao *BOSS* e propõem o *cBOSS* (MIDDLEHURST; VICKERS; BAGNALL, 2019). Os autores (MIDDLEHURST; VICKERS; BAGNALL, 2019) citam que o *WEASEL* é um classificador baseado em dicioná-

rio que é uma extensão do *BOSS*, com resultados superiorres ao *BOSS*, mas que carrega alguns problemas do *BOSS*.

No artigo (NGUYEN et al., 2019), os autores comentam que a literatura de classificação de séries temporais se expandiu rapidamente na última década, com muitas novas abordagens de classificação publicadas a cada ano. A pesquisa focada em melhorar a precisão e a eficiência dos classificadores, com a interpretabilidade sendo um tanto negligenciada. Este aspecto dos classificadores tornou-se crítico para muitos domínios de aplicação e a introdução da legislação *General Data Protection Regulation* (*GDPR*) da *European Union* (EU) em 2018 é indicativo para enfatizar ainda mais a importância de algoritmos de aprendizagem interpretáveis. Atualmente, precisão de classificação de última geração é alcançada com modelos muito complexos baseados em grandes conjuntos (COTE) ou redes neurais profundas (FCN). Essas abordagens não são eficientes no que diz respeito ao tempo ou ao espaço, são difíceis de interpretar e não podem ser aplicadas a séries temporais de comprimento variável, exigindo pré-processamento da série original para um conjunto de comprimento fixo. Em (NGUYEN et al., 2019), os autores propõem novos modelos de classificação de séries temporais para abordar essas lacunas. A abordagem é baseada em representações simbólicas de séries temporais, algoritmos de mineração de sequência eficientes e modelos de classificação linear. Os modelos lineares são tão precisos quanto os modelos de aprendizado profundo, mas são mais eficientes em relação à execução tempo e memória, podendo trabalhar com séries temporais de comprimento variável e serem interpretados destacando as características simbólicas discriminativas na série temporal original. Os autores avançam no estado da arte na classificação de séries temporais, propondo novos algoritmos construídos usando três idéias-chave: (1) Múltiplas resoluções de representações simbólicas: combinamos representações simbólicas obtidas usando diferentes parâmetros, em vez de uma representação fixa (por exemplo, múltiplas representações *Symbolic Aggregate approXimation* (*SAX*)); (2) Múltiplas representações de domínio: combinamos representações simbólicas no tempo (por exemplo, *SAX*) e domínios de frequência (por exemplo, *Symbolic Fourier Approximation* (*SFA*)), para serem mais robustos em todos os tipos de problemas; (3) Eficiente navegação em um grande espaço de palavras simbólicas: estendemos uma sequência simbólica classificador (*SEQuence Learner* (*SEQL*)) para trabalhar com múltiplas representações simbólicas e usar sua estratégia gananciosa de seleção de recursos para filtrar com eficácia os melhores recursos para cada representação (NGUYEN et al., 2019).

Os autores (DEMPSTER; PETITJEAN; WEBB, 2020) discutem que a maioria dos métodos de classificação de séries temporais que atingem a precisão de ponta têm alta complexidade computacional, exigindo tempo de treinamento significativo, mesmo para conjuntos de dados pequenos, e são intratáveis para conjuntos de dados maiores. Além disso, muitos métodos existentes concentram-se em um único tipo de recurso, como forma ou frequência. Com base no recente sucesso de redes neurais convolucionais para classificação de séries

temporais, apresentam o *ROCKET*, baseado em classificadores lineares simples, usando kernels convolucionais aleatórios alcançam o estado da arte: precisão com uma fração do custo computacional dos métodos existentes. Usando este método, é possível treinar e testar um classificador em todos os 85 conjuntos de dados do Repositório UCR em menos de 2 horas, e é possível treinar um classificador em um grande conjunto de dados de mais de um milhão de séries temporais em aproximadamente 1 hora.

Os autores (MIDDLEHURST et al., 2021) apresentam o *HIVE-COTE V2* que é um metaensemble heterogêneo para classificação de séries temporais. *HIVE-COTE V2* forma seu conjunto a partir de classificadores de múltiplos domínios, incluindo shapelets independentes de fase, baseados em dicionários conjuntos de palavras e intervalos dependentes de fase. Desde que foi proposto pela primeira vez em 2016, o algoritmo manteve-se no estado da arte em termos de precisão na classificação de arquivo série temporal UCR. Com o tempo, ele foi atualizado de forma incremental, culminando em seu estado atual, *HIVE-COTE V1*. Durante esse tempo, foram propostos vários algoritmos que correspondem à precisão do *HIVE-COTE V1*. Os autores propõem abrangentes mudanças no algoritmo *HIVE-COTE V1* que melhoram significativamente sua precisão e usabilidade, apresentando esta atualização como *HIVE-COTE V2*. Apresentaram dois novos classificadores, o *Temporal Dictionary Ensemble* (*TDE*) e o *Diverse Representation Canonical Interval Forest* (*DrCIF*), que substituíram os classificadores anteriores do ensemble. Além disso, apresentaram o *Arsenal*, um conjunto de classificadores *ROCKET* como um novo constituinte *HIVE-COTE V2*. O *HIVE-COTE V2* é mais preciso do que o estado da arte atual em 112 conjuntos de dados univariados do repositório de dados UCR e 26 conjuntos de dados multivariados do repositório de dados UEA.

Os autores (FAWAZ et al., 2020), evidenciam que o artigo traz o aprendizado profundo na vanguarda da pesquisa em *TSC*. *TSC* é a área de aprendizado de máquina com a tarefa de categorizar (ou rotular) as séries temporais. As últimas décadas de trabalho nesta área levaram a um progresso significativo na precisão dos classificadores, com o estado da arte agora representado pelo algoritmo HIVE-COTE. Embora extremamente preciso, o HIVE-COTE não pode ser aplicado a muitos conjuntos de dados do mundo real devido à sua complexidade de tempo de alta formação em O($N^2.T^4$) para um conjunto de dados com N séries temporais de comprimento T. Por exemplo, o HIVE-COTE leva mais de 8 dias para aprender a partir de um pequeno conjunto de dados com (N=1500) séries temporais de curta duração (T=46). Enquanto isso, o aprendizado profundo tem recebido enorme atenção por causa de sua alta precisão e escalabilidade. Abordagens recentes para a aprendizagem profunda para *TSC* têm sido escaláveis, mas menos precisas do que o HIVE-COTE. Os pesquisadores apresentam o InceptionTime - um conjunto de modelos de redes neurais convolucionais profundas, inspirado na arquitetura Inception-v4. Os experimentos mostraram que o InceptionTime está no mesmo nível de HIVE-COTE em termos de precisão, embora seja muito mais escalável: não só pode aprender com

1500 séries temporais em uma hora, mas também pode aprender com séries temporais de 8 milhões em 13h, uma quantidade de dados que estão totalmente fora do alcance do HIVE-COTE.

3.3 CONCLUSÕES E PRÓXIMOS PASSOS

Neste capítulo foram apresentados trabalhos que utilizam os mesmos princípios dessa dissertação, detecção e identificação de fungos através de leituras com narizes eletrônicos e modelos de inteligência artificial para resolver esses problemas.

Os próximos passos serão apresentar o nariz eletrônico utilizado para analisar os VOCs emitidos pelas colônias de fungos, expor quais métricas serão utilizadas para mensurar os resultados, discutir os motivos que determinaram as escolhas dos modelos de IA testados para solucionar o problema e estudar o funcionamento de cada um dos modelos de IA que serão testados para solucionar o problema.

4 MATERIAIS E MÉTODOS

Neste capítulo será discutido o funcionamento do nariz eletrônico, as métricas utilizadas para mensurar os resultados da classificação de séries temporais, os motivos que determinaram as escolhas dos modelos de IA testados para solucionar o problema e os métodos de IA testados para resolver o problema.

4.1 NARIZ ELETRÔNICO

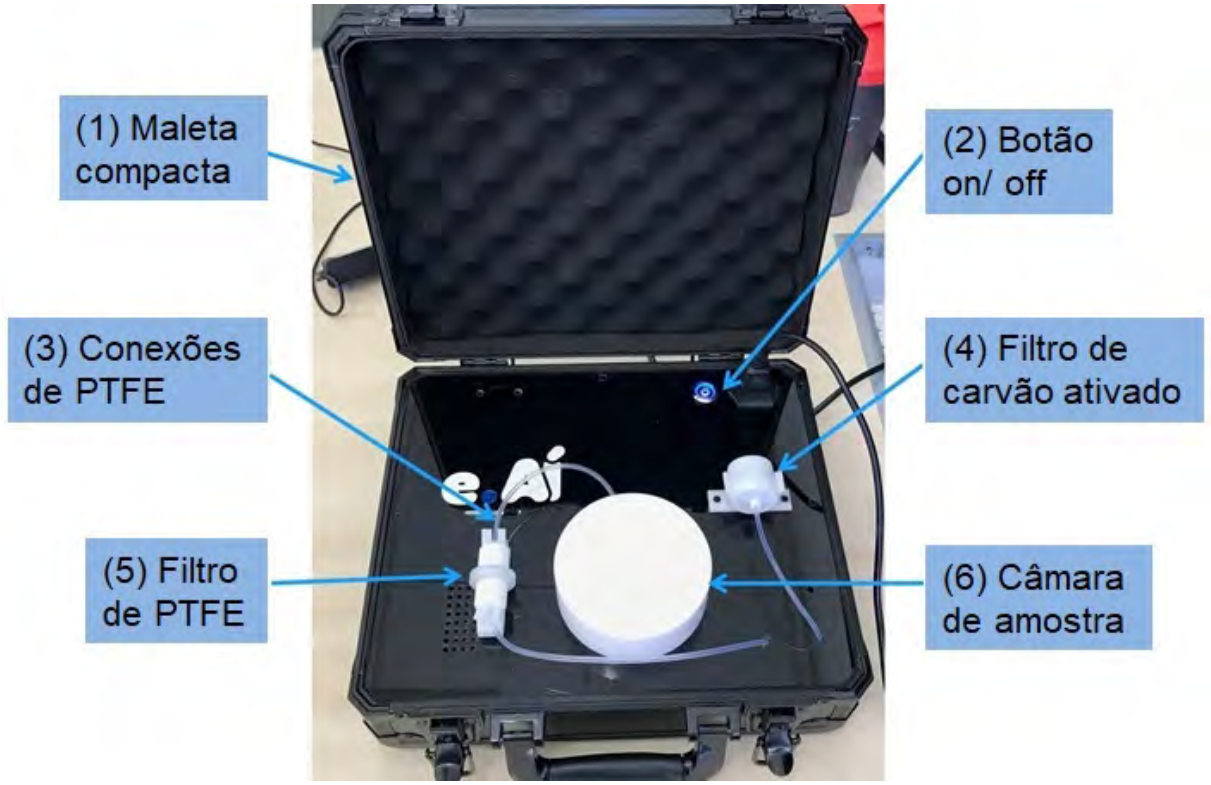
Um nariz eletrônico tenta imitar as três fases do sistema olfativo humano: detecção, processamento de sinais e reconhecimento/ interpretação de odores. Os sensores empregados em narizes eletrônicos não são específicos. Isso significa que os sensores não são seletivos em relação a um determinado composto químico, mas sensíveis a uma grande variedade de compostos, e um pouco mais a algumas famílias químicas, como solventes orgânicos, ácidos graxos, gases sulfurosos, etc., sabidamente presentes nos voláteis de interesse. Dessa forma, as respostas dos sensores produzem padrões característicos para cada mistura química apresentada à matriz de sensores. Ao apresentar muitos produtos químicos diferentes ao conjunto de sensores, um banco de dados de padrões ou base de assinaturas de odores é construído. Esta base de dados é utilizada para treinar o sistema de reconhecimento de padrões, e, posteriormente, permite reconhecer um conjunto de odores armazenados na memória (KUSKE; ROMAIN; NICOLAS, 2005).

Como um odor é uma mistura de compostos gasosos em concentrações relativamente baixas, o princípio do nariz eletrônico pode ser aplicado na detecção e monitoramento de qualquer mistura gasosa, mesmo que essa mistura não tenha odor detectável pelo olfato dos humanos. (KUSKE; ROMAIN; NICOLAS, 2005).

O problema de interesse desse trabalho é a classificação de compostos orgânicos voláteis *VOC* usando sensores compactos: o nariz eletrônico. O nariz eletrônico adotado é um aparelho portátil construído em parceria entre o Centro de Informática da Universidade Federal de Pernambuco (Cin da UFPE) e o Centro Regional de Ciências Nucleares do Nordeste. No caso particular desse trabalho, os *VOC*s analisados focarão nos odores de colônias de fungos para descobrir uma "assinatura de odor"para cada microrganismo e estabelecer novos métodos de detecção e identificação mais rápidos, baratos e precisos do que os disponíveis no mercado. Os dados coletados pelo nariz eletrônico são sinais elétricos gerados a partir da interação dos *VOC*s com as superfícies dos sensores, e são armazenados na forma de séries temporais.

Conforme mostrado na Figura 27, o nariz eletrônico foi construído com materiais para evitar contaminação por *VOC*s como o aço inoxidável e *PTFE*, o *teflon*. O equipamento requer 10 minutos de pré-aquecimento e uma purga de 1 minuto antes da primeira leitura

Figura 27 – Dispositivo de nariz eletrônico usado nos experimentos: (1) O nariz eletrônico é acondicionado em uma maleta compacta; (2) É acionado pelo botão liga-desliga; (3) Todas as conexões são feitas de *PTFE*; (4) Possui filtro de carvão ativado e (5) Filtro de *PTFE*; (6) Câmara de amostra também feita de *PTFE*



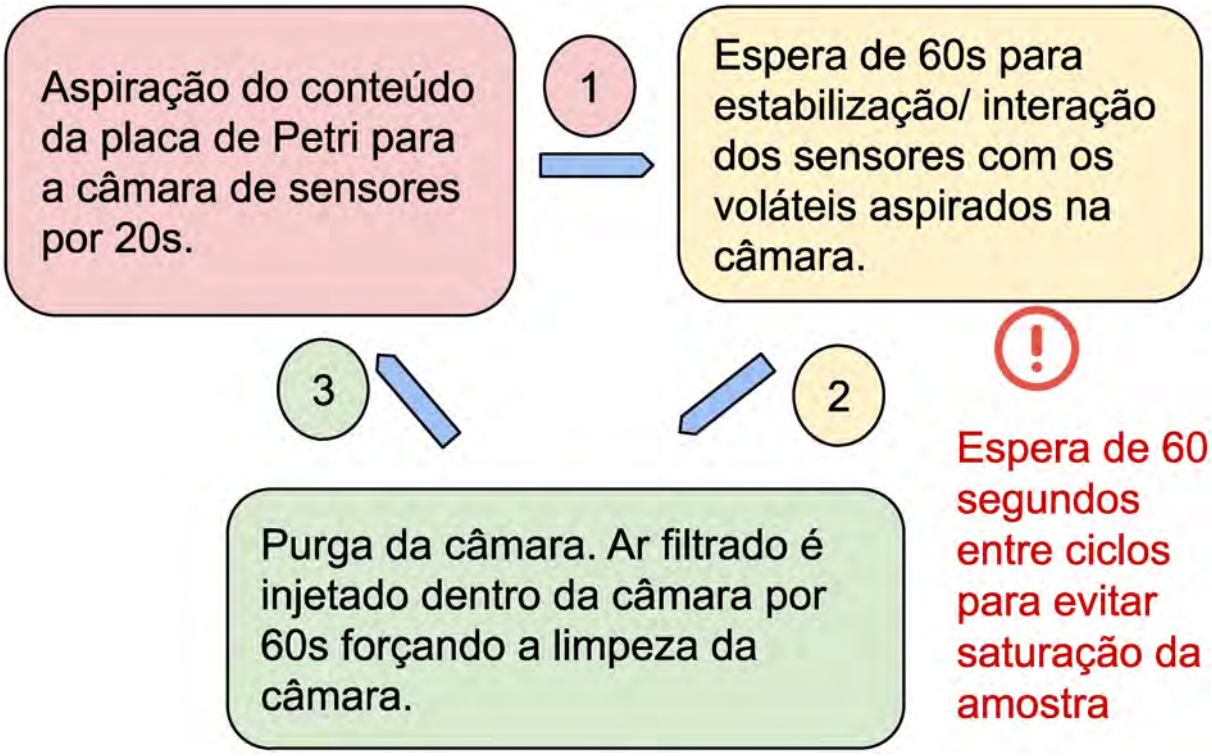
Fonte: Elaborado pelo autor.

e entre as leituras. São realizadas 3 leituras por segundo. Possui controle automático de tempos de injeção de amostra, limpeza (purga), coleta de dados e o intervalo entre as coletas.

O nariz eletrônico trabalha em ciclos de 3 etapas. A primeira etapa é a aspiração dos *VOC*s da câmara de amostra (projetada para encaixar uma placa de *Petri* aberta) para a câmara de sensores por um espaço de tempo. Os experimentos demonstraram que o tempo de 20 segundos é satisfatório. A segunda etapa é a interação/ estabilização dos *VOC*s com os sensores na câmara de sensores, também por um espaço de tempo. Os experimentos evidenciaram que o intervalo temporal de 60 segundos é o suficiente. A terceira e última etapa é a purga/ limpeza da câmara de sensores através da injeção de ar filtrado por carvão ativado, mais uma vez por um espaço de tempo. Os experimentos apontaram que o tempo de 60 segundos é satisfatório.

Outra variável programável no equipamento é o tempo entre ciclos, cuidado importantíssimo para evitar a diminuição ou parada de emissão de *VOC*s pela colônia de fungos estudada. Os experimentos delimitaram o intervalo de 60 segundos como satisfatório. O ciclo dos experimentos totaliza 200 segundos. Conforme mencionado acima, o equipamento realiza 3 leituras por segundo, portanto são gerados 600 pontos por ciclo. Também é possível configurar o número de ciclos por experimento. A Figura 28 ilustra o funcionamento

Figura 28 – Funcionamento dos ciclos do nariz eletrônico utilizado nos experimentos



Fonte: Elaborado pelo autor.

dos ciclos do nariz eletrônico utilizado nos experimentos desse trabalho.

Durante a fase de pesquisa, no caso das leituras em placa, foram realizados vários experimentos de 5 ciclos por leitura a 30 ciclos por leitura sem maiores problemas de perda de fluxo de *VOC*s. Contudo, como segurança, foi adotado um padrão de 10 ciclos por leitura.

Ainda durante a fase de pesquisa, e através de leituras abertas, foi adotado um padrão de 5 ciclos por leitura, seguindo as orientações do parceiro, Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE, que recomendou a observação da RESOLUÇÃO ANVISA RE Nº 09/2003 em conjunto com a NORMA TÉCNICA 001, que estabelecem um tempo de amostragem de 5 minutos a 15 minutos para "Método de Amostragem e Análise de Bioaerosol em Ambientes Interiores."(BRASIL, 2003).

O nariz eletrônico utilizado nos experimentos possui um conjunto de sensores que geram sinais elétricos representados em Ohms, fabricados pela Figaro Engineering Inc., Osaka, Japan, com as sensibilidades para as seguintes famílias de compoestos químicos:

- TGS826 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer um dos seguintes compostos: amônia, iso-butano, hidrogênio e etanol.

- TGS2611 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer um dos compostos: metano, iso-butano, hi-

drogênio, etanol.

- TGS2603 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer composto: aminas e série de enxofre.
- TGS813 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer um dos compostos: metano, propano e butano (gases combustíveis).
- TGS822 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer um dos compostos: etanol, metano, monóxido de carbono, isobutano, n-hexano, benzeno, acetona (vapores de solventes orgânicos).
- TGS2602 (Ohm): É um sensor de semicondutor de óxido de metal (SnO_2) de alta sensibilidade qualitativa para qualquer um dos compostos: amônia e H_2S (contaminantes do ar).
- TGS823 (Ohm): É um sensor de cerâmica (resistente a elevadas temperaturas) de alta sensibilidade qualitativa para qualquer um dos compostos: etanol, metano, monóxido de carbono, isobutano, n-hexano, benzeno, acetona (vapores de solventes orgânicos).

Sensores adicionais para coletar dados de temperatura (em ^{o}C), pressão (em kPa) e umidade (em %) completam o nariz eletrônico adotado.

4.2 MOTIVOS QUE DETERMINARAM AS ESCOLHAS DOS MODELOS DE INTELIGÊNCIA ARTIFICIAL TESTADOS PARA SOLUCIONAR O PROBLEMA DA DISSERTAÇÃO

Tanto a alta precisão quanto a interpretabilidade são desejáveis para classificadores. O *TSF* proposto por (DENG et al., 2013) aborda os desafios usando as duas estratégias a seguir. Em primeiro lugar, o *TSF* usa um critério de divisão denominado ganho de entrada, que combina o ganho de entropia e a medida de distância para identificar divisões de alta qualidade. Em segundo lugar, o *TSF* amostra aleatoriamente recursos O(M) de recursos O(M^2) e, assim, torna a complexidade computacional linear ao longo do comprimento temporal da série. Além disso, cada árvore no *TSF* é cultivada independentemente e, portanto, as técnicas de computação paralela podem ser aproveitadas para acelerar o *TSF* (DENG et al., 2013).

O *TSF* é um ensemble de árvores e não é fácil entendê-lo. No entanto, foi proposta a curva de importância temporal, calculada a partir do TSF, para capturar as características dos intervalos informados. A curva de importância temporal permite identificar importantes características temporais. O *TSF* usa recursos estatísticos simples, mas supera alternativas amplamente utilizadas (DENG et al., 2013).

Em resumo, *TSF* é um classificador de séries temporais preciso e eficiente, e pode fornecer insights sobre o caráter temporal estatísticas úteis para distinguir séries temporais de diferentes classes (DENG et al., 2013).

Segundo (LINES; TAYLOR; BAGNALL, 2018), o *TSF* foi selecionado para ser um dos cinco módulos porque, supera o problema do enorme espaço de recursos de intervalo, empregando uma abordagem de floresta aleatória com estatísticas resumidas de cada intervalo como recursos. Treinar uma única árvore envolve selecionar $\sqrt{m}$ intervalos aleatórios, gerando a média, o desvio padrão e a inclinação dos intervalos aleatórios para cada série. As árvores são treinadas nas $3\sqrt{m}$ características resultantes e a classificação é por maioria de votos.

Diante dos motivos discutidos acima, o *TSF* foi selecionado para tentar resolver o problema da dissertação.

No artigo (LINES; TAYLOR; BAGNALL, 2018), os autores dissertam sobre uma avaliação experimental recente que avaliou 19 algoritmos de *TSC* e descobriu-se que um foi mais preciso do que todos os outros: o *Flat-CODE*. *Flat-CODE* é um ensemble que combina 35 classificadores em quatro representações de dados. No entanto, ainda que abrangente, a avaliação não considerou abordagens de aprendizagem profunda. Redes neurais convolucionais (CNN) são o estado da arte em muitos campos e questionam se as *CNN*s poderiam ser igualmente transformadoras para o *TSC* (LINES; TAYLOR; BAGNALL, 2018).

Em (LINES; TAYLOR; BAGNALL, 2018), os autores implementaram uma *CNN* de referência para *TSC* usando uma estrutura simples e foram usados os resultados de uma *CNN* específica para *TSC* da literatura. Ambos foram comparados com o *Flat-CODE* e descobriu-se que o ensemble é mais preciso do que as duas *CNN*s. Esses resultados são consistentes, mas o *Flat-CODE* não está isento de deficiências. Os autores melhoraram significativamente o ensemble, propondo uma nova estrutura hierárquica, com votação probabilística, definindo e incluindo dois novos classificadores de ensemble construídos em espaços de recursos existentes e adicionando mais módulos para representarem dois domínios de transformação adicionais. O classificador resultante, o *HIVE-COTE V1*, encapsula classificadores construídos em cinco representações de dados. Os autores demonstraram que o *HIVE-COTE V1* é mais preciso que *Flat-CODE* (e todos os outros algoritmos *TSC* de que se tem conhecimento até o momento). Foi testado em mais de 100 reamostras de 85 problemas de *TSC* e é o novo estado da arte para *TSC*. Uma análise adicional está inclusa, através da introdução e avaliação de 3 novos estudos de caso e extensa experimentação em 1000 conjuntos de dados simulados de 5 tipos diferentes (LINES; TAYLOR; BAGNALL, 2018).

Uma grande deficiência do *Flat-CODE* não foi resolvida no *HIVE-COTE V1*: a complexidade computacinal é limitada pela *Shapelet-Transformed Heterogeneous Ensemble of Standard Classification Algorithms* (*ST-HESCA*) devido ao procedimento de extração de shapelet, e *Flat-CODE* que também contém classificadores construídos com base em

dados transformados em shapelet. Portanto, ambos os ensembles têm o mesmo tempo complexidade: O(n^2m^4) (LINES; TAYLOR; BAGNALL, 2018). Por exemplo o *HIVE-COTE V1* leva mais de 8 dias para aprender a partir de um pequeno conjunto de dados com (n=1500) séries temporais de curta duração (m=46) (FAWAZ et al., 2020), o que na prática tona sua utilização inviabilizada.

Contudo, os autores (LINES; TAYLOR; BAGNALL, 2018) propuseram o *RISE* e o selecionaram para ser um dos seus cinco módulos, porque o *RISE* resolveu dois problemas apresentado pelo *Flat-CODE*. Em primeiro lugar, a verdadeira função de autocorrelação é o inverso do espectro de potência, portanto, de muitas maneiras, duas transformações estão medindo a mesma coisa (embora em resoluções diferentes). Isso significa que quase metade dos constituintes em *Flat-CODE* são baseados em espectrais e podem estar causando um desequilíbrio dentro do ensemble. Em segundo lugar, a abordagem *Flat-CODE* para classificadores espectrais usa toda a série para *Power Spectrum* (*PS*) e transformações *Auto Correlation Function* (ACF). Isso pode causar a ofuscação de recursos discriminatórios incorporados, especialmente para séries longas, cujas características espectrais podem mudar com o tempo. É difícil estender essa abordagem a problemas de *TSC*, uma vez que, como o janelamento (por exemplo) aumenta enormemente o espaço de recursos e também introduz um parâmetro adicional, exigirá um nível adicional de validação cruzada para otimizar.

Diante dos motivos discutidos acima, o *RISE* foi selecionado para tentar resolver o problema da dissertação.

Em conformidade com os autores (LEE et al., 2012), a presente pesquisa lida com dados de séries temporais e concentra-se principalmente em fornecer a capacidade de pesquisa de similaridade em um conjunto de sequências de séries temporais. No entanto, o uso de dados de séries temporais pode ir além da busca por similaridade. Muitas aplicações interessantes envolvem previsões baseadas em uma sequência de séries temporais ou em um conjunto de sequências de séries temporais, chamadas de problemas de classificação de séries temporais. Neste estudo, propomos a técnica de classificação de séries temporais baseada em *kNN*, com base na abordagem de classificação do vizinho mais próximo. Usando previsão de churn do setor de telecomunicações móveis como aplicativo de avaliação, nossos resultados empíricos mostram que a técnica *kNN-TSC* proposta alcança melhor desempenho (medido por taxas de erros e alarmes falsos) do que a abordagem tradicional baseada em transformação estatística. Além disso, com o uso do método de média estratificada para combinação de decisão, a técnica *kNN-TSC* proposta pode lidar efetivamente com o problema de distribuição de classes assimétricas (desbalanceadas) (LEE et al., 2012).

O estudo de (LEE et al., 2012), portanto, traz várias contribuições para pesquisas e práticas de mineração de dados (ou especificamente análise de classificação). Primeiro, se destaca a importância dos problemas de classificação de séries temporais e se identificam

as limitações da pesquisa atual no tocante à análise de classificação. Em resposta, é proposta uma nova abordagem para resolver problemas de classificação de séries temporais e desenvolver a técnica de *kNN-TSC* com base na abordagem de classificação de k-vizinhos mais próximos. Os resultados da avaliação comparativa lançam luz sobre a viabilidade e aplicabilidade da proposta *kNN-TSC* e sugerem sua eficácia relativa, em comparação com a abordagem tradicional baseada em transformação estatística, que fora comumente usada em pesquisas anteriores. Em segundo lugar, foram formuladas duas estratégias de aprendizado para classificação de séries temporais (ou seja, estratégias de aprendizado baseadas em modelos e baseadas em instâncias para classificação de séries temporais). Essas duas estratégias de aprendizado fornecem uma base para o desenvolvimento futuro de técnicas de classificação de séries temporais. Por último, embora usemos a previsão de churn para nossos propósitos de avaliação empírica, a técnica proposta pode ser aplicada para dar suporte a uma série de aplicações, incluindo detecção de fraude e detecção de intrusão, nas quais os atributos de entrada de séries temporais estão envolvidos.

Diante dos motivos discutidos acima, o *kNN-TSC* foi selecionado para tentar resolver o problema desta dissertação.

O estudo de (MIDDLEHURST; VICKERS; BAGNALL, 2019) apresenta o *cBOSS*, uma versão mais escalável do classificador *BOSS* que usa um novo mecanismo de conjunto. A substituição do parâmetro de pesquisa por aleatoriamente os conjuntos de parâmetros selecionados proporcionam uma aceleração considerável, e a introdução da subamostragem para maior diversidade e votação ponderada significa que o *cBOSS* não é significativamente menos preciso do que o *BOSS*. A inclusão de um tamanho de conjunto fixo, a capacidade de contrair o tempo de construção e economizar progresso com check-point make o classificador mais robusto e previsível.

Foram criados de forma independente os resultados publicados para o classificador *WEASEL* e verificados os resultados em (SCHÄFER; LESER, 2017). *WEASEL* é significativamente melhor do que *BOSS* e *cBOSS*. No entanto, o classificador também é mais lento para construir dentro da média e tem um consumo de memória tão elevado quanto o do *BOSS*. Isso indica que é uma substituição *BOSS* adequada para conjuntos de dados menores, tem os mesmos problemas de escalabilidade que o *BOSS*. O *cBOSS* só é uma alternativa viável para grandes problemas se um classificador baseado em dicionário for recomendado (MIDDLEHURST; VICKERS; BAGNALL, 2019).

O *cBOSS* se adapta bem a problemas com dezenas de milhares de casos. Contudo, construir modelos com várias centenas de milhares de instâncias ainda pode causar um impasse nos requisitos de espaço e tempo. Expedientes simples, como subamostragem pode facilitar a construção de modelos em grandes dados, mas fazer isso automaticamente enquanto se mantém a precisão é um desafio. Além disso, comparativamente, o *cBOSS* faz uso intensivo de memória, uma vez que usa um classificador de vizinho mais próximo (MIDDLEHURST; VICKERS; BAGNALL, 2019).

Diante dos motivos apontados acima e já que as bases do problema não são grandes, estão na ordem das centenas tanto em comprimento das séries temporais quanto no número de instâncias, o *WEASEL* e o *cBOSS* foram selecionados para tentar resolver o problema desta dissertação.

No estudo de (NGUYEN et al., 2019), foram discutidas duas representações simbólicas de séries temporais (*SAX* e *SFA*) e foi proposta uma estrutura de classificação de séries temporais que pode utilizar ambas as representações em diferentes resoluções. Devido à sua capacidade de poda, o classificador principal (*SEQL*) pode navegar eficientemente no vasto espaço de palavras simbólicas. Aproximação simbólica em múltiplas resoluções é uma abordagem eficaz, em vez de tentar encontrar uma representação simbólica ótima. Além disso, os classificadores funcionam com diferentes representações simbólicas, assim, efetivamente, é possível aprender a partir de um domínio de múltiplo espaço de recursos sem a necessidade de incorporar vários algoritmos de aprendizagem. Esta característica tem grande potencial, pois o classificador pode teoricamente acomodar outras representações simbólicas no futuro. Na prática, essa flexibilidade pode significar que representações e resoluções podem ser escolhidas de acordo com o problema e aplicação domínio.

Os autores (NGUYEN et al., 2019) propuseram 8 algoritmos diferentes baseados em SEQL usando representações simbólicas SAX e SFA, entre eles o *Mr-SEQL*. Estes classificadores propostos foram testados com todo o UCR Time Series Archives e demonstraram que são fortemente competitivos contra o estado da arte, inclusive contra algoritmos complexos, como grandes conjuntos (COTE) ou métodos de aprendizagem profunda (FCN). Enquanto os ensembles e deep são bem conhecidos por suas precisões, eles também são notórios por sua alta demanda de recursos computacionais. Por outro lado, nossos métodos baseados em *SEQL* são mais eficientes devido à combinação eficaz de representações simbólicas e algoritmo de aprendizado de sequência. A complexidade de tempo e espaço dos algoritmos também permitem que eles sejam bem dimensionados para grandes conjuntos de dados. O resultado do melhor algoritmo *TSC* baseado em *SEQL*, o *Mr-SEQL* é um modelo linear, que permite interpretar a decisão de classificação, uma propriedade que é desejável para análise de séries temporais (NGUYEN et al., 2019).

Várias maneiras de interpretar os modelos resultantes também foram discutidas em (NGUYEN et al., 2019). Em particular, a interpretação da decisão de classificação no conhecido repositório de problemas UCR, bem como apresentando em um estudo de caso sobre testes de desempenho de atletas, pode se relacionar com as decisões algorítmicas ao conhecimento de domínio do mundo real. Também foram relatadas e discudas a precisão e a eficiência de tempo e espaço para todos os métodos avaliados durante o estudo de caso (NGUYEN et al., 2019).

Diante dos motivos expostos acima, o *Mr-SEQL* foi selecionado para tentar resolver o problema desta dissertação.

O estudo de (DEMPSTER; PETITJEAN; WEBB, 2020) mostra que os kernels convolu-

cionais são instrumentos que podem capturar muitos dos recursos usados pelos métodos existentes para classificação de séries temporais. Foi visto que, em vez de aprender os pesos do kernel, um grande número de kernels aleatórios - enquanto isolados apenas aproximando padrões relevantes - em combinação são extremamente eficazes para capturar padrões discriminativos em séries temporais.

Além disso, kernels aleatórios têm requisitos computacionais muito baixos, tornando aprendizado e classificação extremamente rápidos. O método proposto utiliza núcleos convolucionais randômicos para fins de transformação e classificação de séries temporais, *ROCKET*, atinge precisão de última geração com uma fração do custo computacional contra os custosos métodos de última geração existentes, além de poder ser dimensionado para milhões de séries temporais. Também foi mostrado que o *ROCKET* é significativamente mais preciso e fundamentalmente mais escalável do que vários métodos escaláveis recentemente propostos para classificação de séries temporais (DEMPSTER; PETITJEAN; WEBB, 2020).

O *ROCKET* faz uso da chave *proportion of positive values* (*ppv*) para resumir a saída de mapas de recursos, permitindo que um classificador pondere a prevalência de um padrão em uma determinada série temporal. Até o momento, o *ppv* não foi usado dessa maneira antes. Esse uso é mais eficaz do que um simples máximo aplicado em uma operação convencional de agrupamento máximo (DEMPSTER; PETITJEAN; WEBB, 2020).

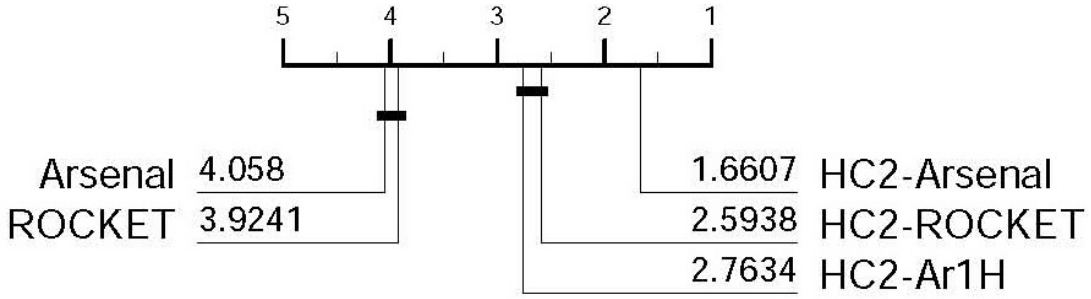
Os autores de (MIDDLEHURST et al., 2021) ponderam que o *ROCKET* é um classificador muito rápido, com precisão de última geração, e acreditam que é o desenvolvimento recente mais importante no campo.

Diante dos motivos evidenciados acima, o *ROCKET* foi selecionado para tentar resolver o problema desta dissertação.

No artigo (MIDDLEHURST et al., 2021), os autores dicutem que o *ROCKET* é um classificador muito rápido com precisão de última geração, e que acreditam ser um dos desenvolvimentos recentes mais importantes no campo. Representa uma classe diferente de abordagem e, como tal, é candidata à assimilação pelo coletivo. No entanto, surge um problema ao tentar incluir o *ROCKET* no *HIVE-COTE V2*: o regressor *Ridge* usado pelo *ROCKET* é difícil de configurar para produzir valores de probabilidade úteis para cada classe ao fazer previsões. O *CAWPE*, estrutura de conjunto do *HIVE-COTE V2*, usa probabilidades ponderadas e depende de classificadores para produzir uma distribuição representativa da força dos classificadores de confiança nas previsões. Uma solução para isso seria substituir o regressor *Ridge* por um classificador que produz estimativas de probabilidade representativas. No entanto, a experimentação com classificadores substitutos não produziu um algoritmo que fosse tão preciso quanto o regressor *Ridge* do *ROCKET* (MIDDLEHURST et al., 2021).

Para resolver este problema, a versão do *ROCKET* utilizada no *HIVE-COTE V2* é um conjunto de classificadores *ROCKET* menores. Esse ensemble de *ROCKET*s foi

Figura 29 – Diagrama de diferença crítica para os classificadores *Arsenal*, *ROCKET*, *HIVE-COTE V2*-Ar1H, *HIVE-COTE V2-ROCKET* e *HIVE-COTE V2-Arsenal* usando-os em 112 conjuntos de dados UCR. *HIVE-COTE V2*-Ar1H representa *HIVE-COTE V2* usando o classificador *Arsenal* com probabilidades geradas da mesma forma que *ROCKET*



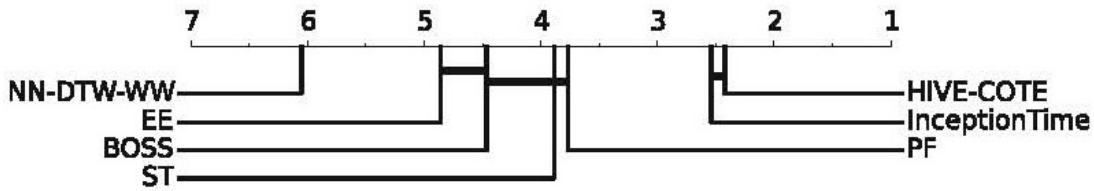
Fonte: (MIDDLEHURST et al., 2021).

denominado de *Arsenal*. Novos casos são classificados por maioria de votos. *Arsenal* é mais lento de construir do que o *ROCKET*, mas suas melhores probabilidades fazem é um candidato melhor para *HIVE-COTE V2*. A Figura 29, mostra ambas as versões do *ROCKET* e três versões de *HIVE-COTE V2*, uma versão do *HIVE-COTE V2* com *ROCKET*, outra do *HIVE-COTE V2* com *Arsenal* e uma versão *HIVE-COTE V2*-Ar1H representa *HIVE-COTE V2* usando o classificador *Arsenal* com probabilidades geradas da mesma forma que *ROCKET*. O *Arsenal* não gera melhoria alguma em relação ao *ROCKET* em termos de precisão, como também o *Arsenal* que usa o mesmo método para gerar probabilidades que o *ROCKET* não faz melhoria no *HIVE-COTE V2*. No entanto, a versão *HIVE-COTE V2*, incluindo uma versão inalterada do *Arsenal*, é melhor. Mesmo com probabilidades estimadas através de votos de um conjunto de pequenas dimensões, é feita uma grande diferença sobre não ter nenhum no *HIVE-COTE V2* (MIDDLEHURST et al., 2021).

Diante dos motivos discutidos acima, o *Arsenal* foi selecionado para tentar resolver o problema da dissertação.

Segundo o trabalho de (MIDDLEHURST et al., 2021) o *HIVE-COTE V2*, é um metaensemble de quatro classificadores muito diferentes, cada um deles é projetado para capturar diferentes características discriminatórias. Representa um novo estado da arte em termos de classificação de séries temporais, superando os melhores classificadores em problemas univariados e multivariados em termos de precisão. O estudo mostrou que *HIVE-COTE V2* é melhor do que qualquer um de seus constituintes, e que cada componente dá uma contribuição significativa para o desempenho geral. Acredita-se que sua força reside no fato de que muitos problemas têm características discriminatórias em vários domínios de dados; uma shapelet pode ser indicativo de um valor de classe, enquanto um padrão de repetição pode caracterizar outro. *HIVE-COTE V2* usa um conjunto simples, mas um esquema eficaz para combinar informações que foram demonstradas como melhores do que alternativas como empilhamento ou uma estratégia de seleção (MIDDLEHURST et al., 2021).

Figura 30 – Diagrama de diferença crítica mostrando o desempenho do InceptionTime em comparação com classificadores de última geração de dados de séries temporais



Fonte: (FAWAZ et al., 2020).

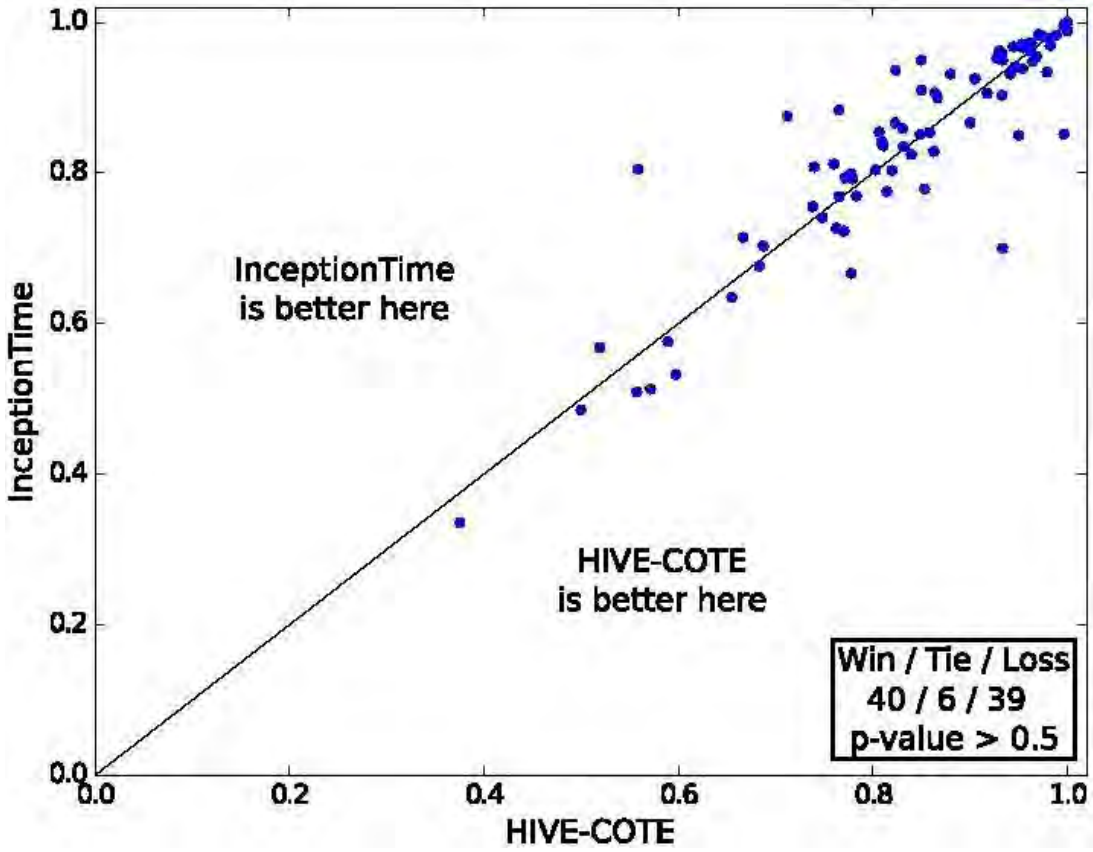
O ponto fraco do *HIVE-COTE V2* é que ele não se adapta bem a problemas muito grandes. Foi mostrado que para problemas com milhares de séries de comprimento na casa das dezenas de milhares, o tempo de construção pode ser excessivo, mas nada impede que possa ser feita pelo menos uma estimativa. Mesmo o *ROCKET*, que é de longe o algoritmo mais rápido, não escala bem com o aumento do número de instâncias.

Diante dos motivos discutidos acima e já que as bases do problema não são grandes, estão na ordem das centenas, tanto em comprimento das séries temporais quanto no número de instâncias, o *HIVE-COTE V2* foi selecionado para tentar resolver o problema da dissertação.

Segundo o trabalho de (FAWAZ et al., 2020), o aprendizado profundo para classificação de séries temporais ainda está atrás das redes neurais para reconhecimento de imagens em termos de estudos experimentais e projetos arquitetônicos. Em (FAWAZ et al., 2020), foi preenchida a lacuna apresentando o InceptionTime, baseado no recente sucesso de redes baseadas em Inception para várias tarefas de visão computacional. As redes foram reunidas para produzir novos resultados de última geração para o *TSC* nos 85 conjuntos de dados do arquivo UCR. A abordagem é altamente escalável, duas ordens de magnitude mais rápida do que os modelos atuais de última geração, como *HIVE-COTE V1*. A magnitude dessa velocidade é consistente em ambos os repositórios de Big Data *TSC*, bem como em séries de tempo mais longas com alta taxa de amostragem. Também foram investigados os efeitos na precisão geral de vários hiperparâmetros da arquitetura *CNN*. Para estes, a pesquisa foi muito além das práticas padrão para dados de imagem e projetamos redes com filtros longos. Foram realizados testes usando um conjunto de dados simulado e a investigação foi estruturada em termos da definição do campo receptivo de uma *CNN* para *TSC* (FAWAZ et al., 2020).

A Figura 30 ilustra o diagrama de diferença crítica com InceptionTime adicionado à mistura dos classificadores atuais de última geração para dados de séries temporais. Pode-se ver que o conjunto InceptionTime atinge precisão competitiva com o algoritmo líder da classe *HIVE-COTE V1*, um conjunto de 37 algoritmos *TSC* com um esquema de votação hierárquico. Enquanto os dois algoritmos compartilham o mesmo resultado no diagrama de diferença crítica, a paralelização de *Graphics Processing Unit* (*GPU*) trivial de modelos de aprendizado profundo torna o aprendizado de nosso modelo InceptionTime

Figura 31 – Gráfico de precisão mostrando como o modelo InceptionTime não é significativamente diferente do *HIVE-COTE V1*



Fonte: (FAWAZ et al., 2020).

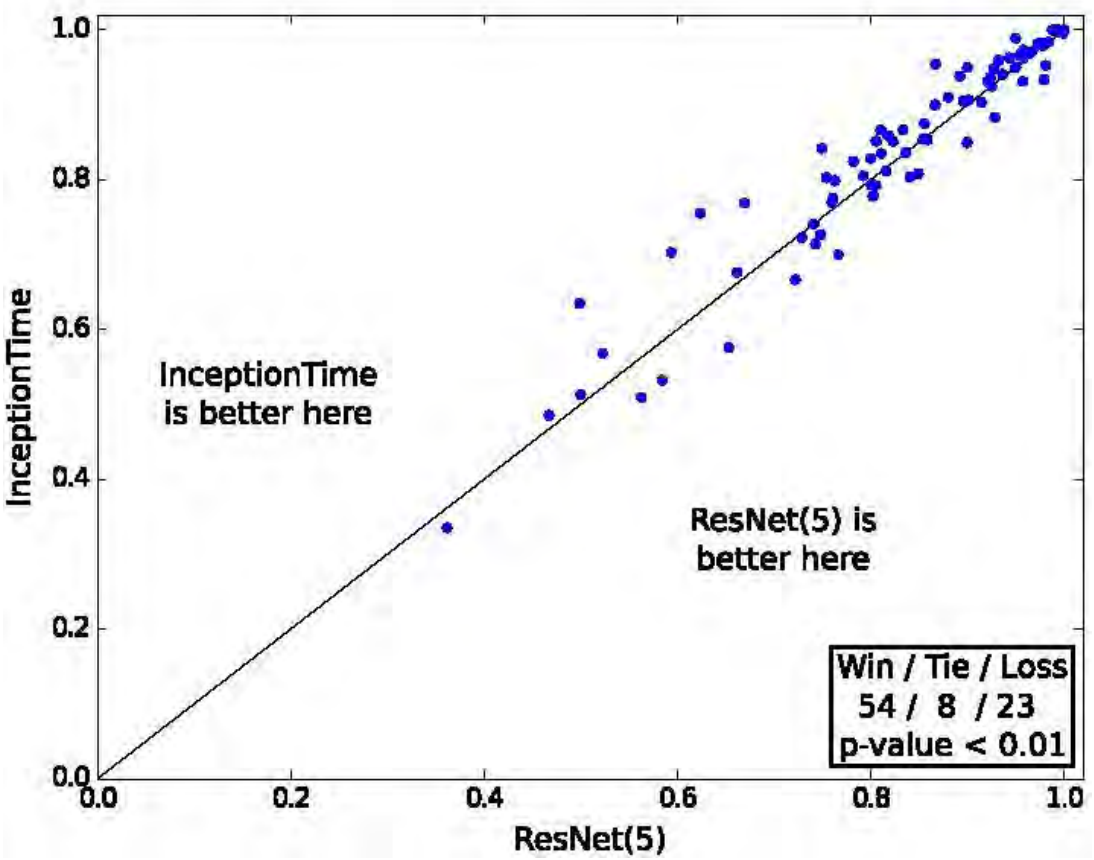
uma tarefa substancialmente mais fácil do que treinar os 37 classificadores diferentes de *HIVE-COTE V1*, cuja implementação não aproveita trivialmente o poder computacional das *GPU*s (FAWAZ et al., 2020).

Para visualizar melhor a diferença entre o InceptionTime e o *HIVE-COTE V1*, a Fig. 31 mostra o gráfico de precisão do InceptionTime em comparação com o *HIVE-COTE V1* para cada um dos 85 conjuntos de dados UCR. Os resultados mostram uma Vitória/ Empate/ Perda de 40/ 6/ 39 a favor do InceptionTime. No entanto, a diferença não é estatisticamente significativa como discutido anteriormente (FAWAZ et al., 2020).

O gráfico de precisão de pares na Fig. 32 compara o InceptionTime a um modelo que chamamos de ResNet (5), que é um conjunto de 5 redes ResNet diferentes. Verifica-se que o InceptionTime mostrou uma melhoria significativa em relação ao seu concorrente de rede neural, o melhor conjunto de aprendizado profundo anterior para TSC. Especificamente, os resultados mostram uma Vitória/ Empate/ Perda de 54/ 8/ 23 a favor de InceptionTime contra ResNet (5) com um valor $p < 0{,}01$, sugerindo que o ganho significativo no desempenho se deve principalmente às melhorias da arquitetura de rede de iniciação proposta (FAWAZ et al., 2020).

Diante dos motivos discutidos acima, o InceptionTime foi selecionado para tentar resolver o problema desta dissertação.

Figura 32 – Gráfico de precisão mostrando como o modelo InceptionTime supera significativamente a ResNet(5)



Fonte: (FAWAZ et al., 2020).

4.3 MÉTODOS DE INTELIGÊNCIA ARTIFICIAL TESTADOS PARA RESOLVER O PROBLEMA DA DISSERTAÇÃO

Como visto anteriormente na Seção 4.1, Nariz Eletrônico, e na Seção 2.3, Assinatura de Odores de espécies de fungos, o problema dessa dissertação é identificar um conjunto de *MVOC*s que são emitidos por culturas de fungos. A esse conjunto de *MVOC*s característicos de cada espécie de fungo é denominado assinatura de odor. Cada *MVOC* é um conjunto de séries temporais lida pelo nariz eletrônico e apresenta um perfil característico do fungo estudado nas condições de temperatura, pressão e umidade registradas pelo aparelho.

Conforme comentado na Seção 2.2, Origem das colônias de fungos utilizados no estudo, esse trabalho contempla 5 gêneros de fungos *Aspergillus*, *Cladosporium*, *Fusarium*, *Penicillium* e *Rhizomucor*.

Portanto, para solucionar esse problema de classificação, é necessário encontrar modelos de aprendizagem de máquina, especializados em classificar séries temporais, que apresentarem os melhores resultados.

(BHATTACHARYYA, 2021), apresenta referência a uma biblioteca *Python* especializada

em dados de séries temporais e faz alguns comentários relevantes sobre o assunto.

Os dados de séries temporais são amplamente usados para analisar diferentes tendências e sazonalidades de produtos ao longo do tempo, por vários setores. A *Sktime* é uma biblioteca/ estrutura *Python* unificada que fornece API para aprendizado de máquina com dados de séries temporais e ferramentas compatíveis com *Sklearn* para analisar, visualizar, ajustar e validar vários modelos de aprendizagem de séries temporais, como previsão de série temporal, regressão de série temporal e classificação. *Sktime* foi apresentado em um artigo de pesquisa chamado *sktime: A Unified Interface for Machine Learning with Time Series* para *NeurIPS* por um grupo de pesquisadores do *Alan Turing Institute*. (LÖNING et al., 2019).

A *Sktime* explora uma combinação de ambos os recursos de modelos de séries temporais populares e a biblioteca de aprendizado de *sci-kit*. A biblioteca *Sktime* usa modelos *Sklearn* na redução de vastos dados tabulares. Outros recursos incluem regressão de série temporal, classificação (multivariada e univariada), agrupamento de série temporal, anotações de série temporal, previsão, estimativa, transformação, conjuntos de dados, ferramentas de recurso e funções de utilidade (pré-processamento e plotagem) (LÖNING et al., 2019).

Os principais objetivos da biblioteca são fornecer:

- Interface padrão para construir diferentes tipos de tarefas de aprendizagem de série temporal usando recursos de aprendizagem do *sci-kit*.
- Aplicação de vários modelos de redução.
- Ferramentas de composição de modelo, ferramentas de avaliação de modelo e ferramentas de benchmarking comparativo.
- Interface para lidar com dados variados de séries temporais.

Dentre os modelos de classificação existentes na *Sktime* serão testados:

- *TSF*, da categoria intervalo
- *RISE*, da categoria intervalo
- *kNN-TSC*, da categoria distância
- *cBOSS*, da categoria dicionário
- *WEASEL*, da categoria dicionário
- *Mr-SEQL*, da categoria *shaplet*
- *ROCKET*, da categoria *kernel*
- *Arsenal*, da categoria *kernel*

- *HIVE-COTE V2*, da categoria hibrida

Além desses 9 modelos da biblioteca *Sktime*, será testado também o *InceptionTime* (FAWAZ et al., 2020).

Antes do aprofundamento sobre cada um dos 10 modelos de *Machine Learning* (*ML*) selecionados para resolver o problema em tela, é necessária a discussão de quais métricas serão utilizadas para mensurar o desempenho dos classificadores.

4.3.1 *Time Series Forest (TSF)*

A *TSF* é um modelo baseado em intervalo, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

Segundo (DENG et al., 2013), a *TSF* emprega uma combinação de ganho de entropia e medida de distância, referidos para obter o ganho de entrada (entropia e distância), para avaliar as divisões. Estudos experimentais mostram que o ganho de entrada melhora a precisão dos recursos de amostras aleatórias de *TSF*. A *TSF* atua em cada nó da árvore e tem complexidade computacional linear no período de tempo de série temporal e pode ser construída usando técnicas de computação paralela. A importância da curva temporal é proposta para capturar as características temporais úteis para a classificação. Estudos experimentais também mostram que a *TSF*, usando recursos simples, como média, desvio padrão e a inclinação, é computacionalmente eficiente e supera concorrentes fortes, como classificadores da vizinhança mais próxima com sincronização temporal dinâmica. A Figura 33 mostra um exemplo simplificado do funcionamento do modelo.

A *TSF* é uma modificação do modelo de *Randon Forest* (*RF*) para a configuração de série temporal:

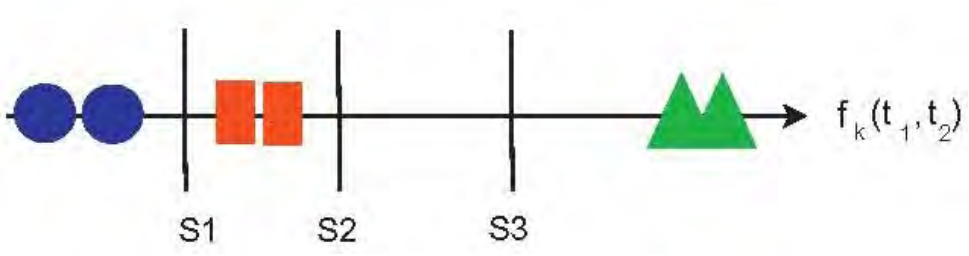
1. Divida a série em vários intervalos aleatórios,
2. Extraia características (média, desvio padrão e inclinação) de cada intervalo,
3. Treine uma árvore de decisão sobre os recursos extraídos,
4. *Ensemble* as etapas 1 - 3.

4.3.2 *Random Interval Spectral Forest (RISE)*

O *RISE* é um modelo baseado em intervalo, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

Em (LINES; TAYLOR; BAGNALL, 2018), os autores demonstram que o *RISE* baseia-se em ideias de conjuntos baseados em árvores, como *RF* e *TSF*. Como *TSF*, foram construídas árvores em intervalos dos dados para construir um classificador de *RF*. A principal

Figura 33 – O eixo x representa o valor de um recurso de intervalo. A figura mostra seis instâncias associadas a três classes (azul, vermelho e verde) e três divisões (S1, S2 e S3) produzindo o mesmo ganho de entropia. O ganho de entrada E é capaz de selecionar S3 como a melhor divisão.



Fonte: (DENG et al., 2013).

diferença, no entanto, é que enquanto a *TSF* usa recursos de domínio do tempo calculando a média, variância e inclinação de cada intervalo, o *RISE* extrai características espectrais sobre cada intervalo aleatório. São selecionados n intervalos aleatórios (por exemplo 500) e calculadas características espectrais para cada intervalo de forma independente. Treinados classificadores de árvores de decisão separadas em cada conjunto de recursos e, em seguida, combinadas as árvores em uma floresta, o classificador conjunto resultante contém 500 aprendizagens básicas que são diversificadas por meio de seleção de intervalo. Adicionalmente, a primeira árvore em *RISE* é um caso especial que usa toda a série.

Existem muitas opções para os recursos espectrais que podem ser usados no *RISE*. Pode-se usar recursos *PS*, recursos ACF ou uma combinação dos dois. A hipótese é que o melhor classificador será produzido através da combinação de recursos *PS* e ACF para um único classificador.

4.3.3 *kNN-Based Time-Series Classification (kNN-TSC)*

O *kNN-TSC* é um modelo baseado em distância, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

Segundo a pesquisa de (LEE et al., 2012), fica claro que a técnica de classificação de séries temporais baseada em k-vizinhos mais próximos *kNN-TSC* envolve dois componentes principais: uma medida de similaridade de séries temporais e um método de combinação de decisão (LEE et al., 2012).

O modelo de similaridade adotado por (LEE et al., 2012) foi proposto por (LIN; SHIM, 1995) e corresponde a um modelo de similaridade de séries temporais, baseado na abordagem de casamento de subsequências, que permite gaps não emparelhados, escala de amplitude e tradução de offset (deslocamento) (LIN; SHIM, 1995). No modelo, a semelhança de duas sequências de séries temporais é medida pela fração do comprimento de pares ordenados de tempo não sobrepostos de subsequências semelhantes para o comprimento total das duas sequências. Duas subsequências são consideradas semelhantes se

uma estiver dentro de um envelope de uma largura especificada desenhada ao redor da outra. Ao combinar subsequências em vez de sequências em si, algumas partes de sequências que são consideradas discrepantes ou lacunas podem ser deixadas de fora no processo de correspondência, e as subsequências correspondentes não precisam ser alinhadas ao longo do eixo do tempo. Além disso, a escala e o deslocamento das duas sequências podem ser ajustados adequadamente antes de determinar as subsequências de uma sequência que correspondem às subsequências da outra sequência. Devido à sua capacidade de endereçamento de outliers, escala de amplitude e tradução de offset (deslocamento) em sequências de séries temporais, o método de casamento de subsequência proposto por (LIN; SHIM, 1995) foi adotado para medir semelhanças de sequências de séries temporais na técnica *kNN-TSC* proposta(LEE et al., 2012). Para mais detalhes do modelo de similaridade, consultar (LIN; SHIM, 1995).

Usando a medida de similaridade descrita anteriormente, as instâncias de treinamento mais próximas de uma instância não classificada (isto é, instância cuja classe de decisão deve ser prevista) são selecionadas como as vizinhas mais próximas da instância alvo. Subsequentemente, um método de combinação de decisão que combina as classes de decisão conhecidas dos k vizinhos mais próximos é necessário para derivar a previsão geral para a instância de destino. Vários métodos de combinação de decisão têm sido propostos na literatura, incluindo métodos de votação majoritária, votação ponderada e média estratificada. O método de votação seleciona os k vizinhos mais próximos para a instância não classificada o_p e então atribui a classe de decisão para o_p como a classe de decisão majoritária dos k vizinhos mais próximos. O método de votação ponderada assume que os vizinhos que são mais semelhantes a o_p carregam mais pesos em seus votos. Nesse sentido, o método de votação ponderada calcula a soma da similaridade (para o_p) dos k vizinhos mais próximos que pertencem a cada classe de decisão e atribui o_p à classe de decisão que recebe a maior soma de similaridade (ou seja, votos ponderados) dos k vizinhos mais próximos (LEE et al., 2012).

Em aplicações de análise de classificação do mundo real, a distribuição de classes de decisão pode ser assimétrica (desbalanceada) em instâncias de treinamento. Essa distribuição de classe assimétrica (desbalanceada) também pode ser encontrada em aplicações de classificação de séries temporais. Os métodos de combinação de decisão acima mencionados (ou seja, votação majoritária e votação ponderada), normalmente usados pela classificação *kNN*, geralmente são incapazes de lidar com aplicações com distribuição de classe assimétrica (desbalanceada) porque nenhum ou apenas poucos dos k vizinhos mais próximos serão selecionados da decisão minoritária classe e, assim, levará a previsões de decisão favorecendo a classe de decisão majoritária.

(YANG et al., 1999) propôs um método de combinação de decisão (referido como o método de média estratificada no estudo de (LEE et al., 2012)) para resolver o problema de assimetria de classe evidenciado. Especificamente, em vez de selecionar k vizinhos mais

próximos para uma instância não classificada o_p, o método de média estratificada requer um número pré-especificado de vizinhos mais próximos para cada classe de decisão. Para um problema de classificação dicotômica (supondo que as duas classes de decisão sejam C_1 e C_2), este método seleciona k_1 vizinhos mais próximos das instâncias de treinamento pertencentes a C_1 e k_2 vizinhos mais próximos daquelas de C_2. Assim, a similaridade média (para o_p) dos k_1 vizinhos mais próximos e a dos k_2 vizinhos mais próximos são derivados independentemente para as duas classes de decisão, respectivamente. O método de média estratificada então atribui o_p à classe de decisão que recebe uma similaridade média maior do que a outra classe (LEE et al., 2012). Para mais detalhes sobre combinação de decisão ou método de média estratificada, consultar (YANG et al., 1999).

No estudo de (LEE et al., 2012), são implementamos os métodos de votação majoritária, votação ponderada e média estratificada como métodos alternativos de combinação de decisão para a técnica *kNN-TSC* proposta. São comparados empiricamente, usando previsão de churn do setor de telecomunicações móveis como aplicativo de avaliação, os efeitos dos três métodos de combinação de decisão na eficácia da classificação e é confirmada a média estratificada como método de combinação de decisão do modelo.

4.3.4 *Contractable Bag of Symbolic Fourier Approximation Symbols (cBOSS)*

O *cBOSS* é um modelo baseado em dicionário, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

No artigo (MIDDLEHURST; VICKERS; BAGNALL, 2019), os autores expõem os princípios de funcionamento dos classificadores que utilizam a frequência das palavras como base para encontrar características discriminatórias que são frequentemente referidas como classificadores baseados no dicionário. Esses classificadores têm semelhanças com abordagens baseadas em pacotes de palavras que são comumente usadas em visão computacional. Informalmente, uma abordagem baseada em dicionário será útil quando as características discriminatórias são padrões repetidos que ocorrem com mais frequência em uma classe, em seguida, outras classes.

Um único classificador base *BOSS* procede da seguinte maneira: para cada série, o classificador base *BOSS* extrai as janelas sequencialmente, normalizando a janela se o parâmetro p for verdadeiro. Em seguida, o classificador base *BOSS* aplica uma *Discret Fourier Transform* (*DFT*) às subséries resultantes, ignorando o primeiro coeficiente se p for verdadeiro. Os coeficientes *DFT* são truncados para inclusão apenas dos primeiros 1/2 termos de Fourier (reais e imaginários). As amostras truncadas são então discretizadas em valores possíveis de α usando um modelo chamado *Multiple Coeficiente Binning* (*MCB*). *MCB* envolve uma etapa de pré-processamento para encontrar os pontos de quebra de discretização estimando a distribuição de coeficientes de Fourier. Janelas consecutivas que produzem a mesma palavra são contadas apenas como uma única instância da palavra.

Uma função de distância *BOSS* sob medida é usada com um classificador de vizinho mais próximo para classificar novas instâncias. A função de distância é não simétrica, incluindo apenas a distância para recursos que não são zero no primeiro vetor de recurso fornecido. O classificador base *BOSS* tem quatro parâmetros: comprimento da janela w, comprimento da palavra l, se deve normalizar cada janela p e alfabeto tamanho α. O conjunto *BOSS* (também conhecido apenas como *BOSS*), avalia todos classificadores de base *BOSS* no intervalo w $\in \{10...m\}$, l $\in \{16, 14, 12, 10, 8\}$ e p $\in \{verdadeiro, falso\}$. Esta pesquisa de parâmetro é usada para determinar quais classificadores base são usados no conjunto. O tamanho do alfabeto é fixado em 4 para todos os experimentos. O número de tamanhos de janela é uma função do comprimento da série m. Todos os classificadores de base *BOSS* que possuírem uma precisão de treinamento dentro de 92% dos melhores, os classificadores de base de desempenho são mantidos para o *ensemble*. Esta dependência do comprimento da série e variabilidade do tamanho do conjunto são fatores que podem impactar significativamente a eficiência. A classificação de novas instâncias é, então, feita usando o voto da maioria do conjunto (MIDDLEHURST; VICKERS; BAGNALL, 2019).

O *cBOSS* é uma evolução do *BOSS* e se concentra principalmente na técnica de *ensemble* de classificadores, o que é computacionalmente cara e imprevisível. *Ensembling* foi mostrado ser um componente essencial do *BOSS*, resultando em significativamente mais precisão. Foi avaliada a possibilidade de substituição do mecanismo de conjunto do *BOSS* por um esquema mais estável e eficiente sem uma redução significativa na precisão. Foi verificado que a *randomização* completa, ou seja, a seleção aleatória, combinações de parâmetros para um número fixo de classificadores base funcionaram razoavelmente bem, mas em algumas bases apresentaram um desempenho muito ruim. Diante disso, foi mantida uma avaliação interna de cada membro possível por meio de validação cruzada *leave-one-out* nos dados do treinamento, em seguida, usa-se esse valor para selecionar e ponderar os votos do classificador. A principal diferença em relação ao *BOSS* é que não são determinados quais membros devem ser retidos após a pesquisa completa. Em vez disso, foi introduzido um novo parâmetro, k, um tamanho fixo de ensembles, que mantém uma lista de pesos. É disponibilizado, por meio de contratação, a opção de substituir o tamanho do ensemble k por um limite de tempo t. Por meio dessa mudança no processo de construção, o classificador é configurado desde o início da construção até o momento em que o tempo seja maior do que t. Mesmo com a ponderação, classificadores com parâmetros para um problema específico podem degradar o classificador geral. Por causa disso, foi definido um tamanho máximo de conjuntos para filtrá-los. Com qualquer classificador construído após este valor substitui-se o membro de menor precisão atual do conjunto se a precisão for maior. Para diversificar ainda mais o conjunto e aumentar a eficiência, é tomada uma subamostra de 70% selecionada aleatoriamente dos dados de treinamento para cada classificador indivídual. O espaço de parâmetro que se amostra aleatoriamente para o *cBOSS* é o mesmo que aquilo que o *BOSS* procura exaustivamente.

Para classificar novas instâncias, é adotado o esquema de ponderação exponencial usado na *CAWPE* para amplificar a pequena diferença em pesos, e resulta em um desempenho significativamente melhorado (MIDDLEHURST; VICKERS; BAGNALL, 2019). O *cBOSS* é descrito mais formalmente na Figura 34.

Figura 34 – Algorithm 1 - Descrição mais formal do cBOSS

Algorithm 1. cBOSS_build(A list of n cases length m, $\mathbf{T} = (\mathbf{X}, \mathbf{y})$)

Parameters: the ensemble size k, the max ensemble size s

Let w be window length, l be word length, p be normalise/not normalise and α be alphabet size.
Let $\mathbf{C}$ be a list of s BOSS classifiers $(\mathbf{c}_1, \ldots, \mathbf{c}_s)$
Let $\mathbf{E}$ be a list of s classifier weights $(\mathbf{e}_1, \ldots, \mathbf{e}_s)$
Let $\mathbf{R}$ be a set of possible BOSS parameter combinations
$i \leftarrow 0$
$lowest\_acc \leftarrow \infty, lowest\_acc\_idx \leftarrow \infty$
while $i < k$ AND $|\mathbf{R}| > 0$ **do**
  $[l, a, w, p] \leftarrow random\_sample(\mathbf{R})$
  $\mathbf{R} = \mathbf{R} \setminus \{[l, a, w, p]\}$
  $\mathbf{T}' \leftarrow$ subsample_data($\mathbf{T}$)
  $cls \leftarrow$ build_base_BOSS($\mathbf{T}', l, a, w, p$)
  $acc \leftarrow$ LOOCV(cls) { *train data accuracy*}
  if $i < s$ **then**
    if $acc < lowest\_acc$ **then**
      $lowest\_acc \leftarrow acc$, $lowest\_acc\_idx \leftarrow i$
    $c_i \leftarrow cls$, $e_i \leftarrow acc^4$
  else if $acc > lowest\_acc$ **then**
    $c_{lowest\_acc\_idx} \leftarrow cls$, $e_{lowest\_acc\_idx} \leftarrow acc^4$
    $[lowest\_acc, lowest\_acc\_idx] \leftarrow$ find_new_lowest_acc($\mathbf{C}$)
  $i \leftarrow i + 1$

Fonte: (MIDDLEHURST; VICKERS; BAGNALL, 2019).

4.3.5 *Word Extraction for Time Series Classification (WEASEL)*

O *WEASEL* é um modelo baseado em dicionário, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

Segundo (MIDDLEHURST; VICKERS; BAGNALL, 2019), o *WEASEL* é um classificador baseado em dicionário que é uma extensão do *BOSS*. O *WEASEL* é um único classificador em vez de um conjunto. Em *WEASEL* são concatenados histogramas para uma gama de valores de parâmetro de w e l, em seguida, executa um recurso seleção para reduzir o espaço de recursos. Como o *BOSS*, o *WEASEL* executa uma *DFT* em cada janela. Os coeficientes da *DFT* não são mais truncados e, em vez disso as características reais e imaginárias mais discriminativas são mantidas, conforme determinado por um teste

ANOVA F. Os valores retidos são então discretizados em palavras usando informação de ganho *binning*, semelhante à etapa *MCB* em *BOSS*. O *WEASEL* não remove palavras duplicadas adjacentes como o *BOSS* faz. A palavra e o tamanho da janela são usados como chaves para indexar o histograma. Um outro histograma é formado para bigramas. O número de recursos é reduzido usando um teste qui-quadrado após os histogramas para cada instância serem criados, removendo quaisquer palavras com pontuação abaixo um limite. O *WEASEL* usa um classificador de regressão logística para fazer as previsões para novos casos, além de realizar uma pesquisa de parâmetros p e a para um *range* reduzido de l e usa uma *10-fold cross-validation* para determinar o desempenho de cada conjunto. O tamanho do alfabeto α é fixado para 4 e o parâmetro *chi* é fixado para 2.

Os principais parâmetros do modelo são:

- *alphabet_size*: tamanho do alfabeto *chi2-threshold*: usado para seleção de recursos para selecionar as melhores palavras anova: seleciona os melhores coeficientes de Fourier l/2 além dos primeiros bigramas: usando bigramas de palavras *SFA*;
- *binning_strategy*: a estratégia de *binning* usada para discretizar em Palavras *SFA*.

O *WEASEL* desliza o comprimento da janela ao longo da série. A janela de comprimento w é encurtada para uma palavra de comprimento l tomando uma transformada de Fourier e mantendo os melhores coeficientes complexos l/2 usando um teste anova unilateral. Esses coeficientes l são então discretizados em possíveis símbolos alfa, para formar uma palavra de comprimento l. Um histograma de palavras para cada série é formado e armazenado. Para cada comprimento de janela, uma *bag* é criada e todas as palavras são unidas em um *bag-of-patterns*. Palavras de diferentes comprimentos de janela são discriminadas por diferentes prefixos. O ajuste envolve o treinamento de um classificador de regressão logística no único *bag-of-patterns* (LÖNING et al., 2019), (PONG, 2020).

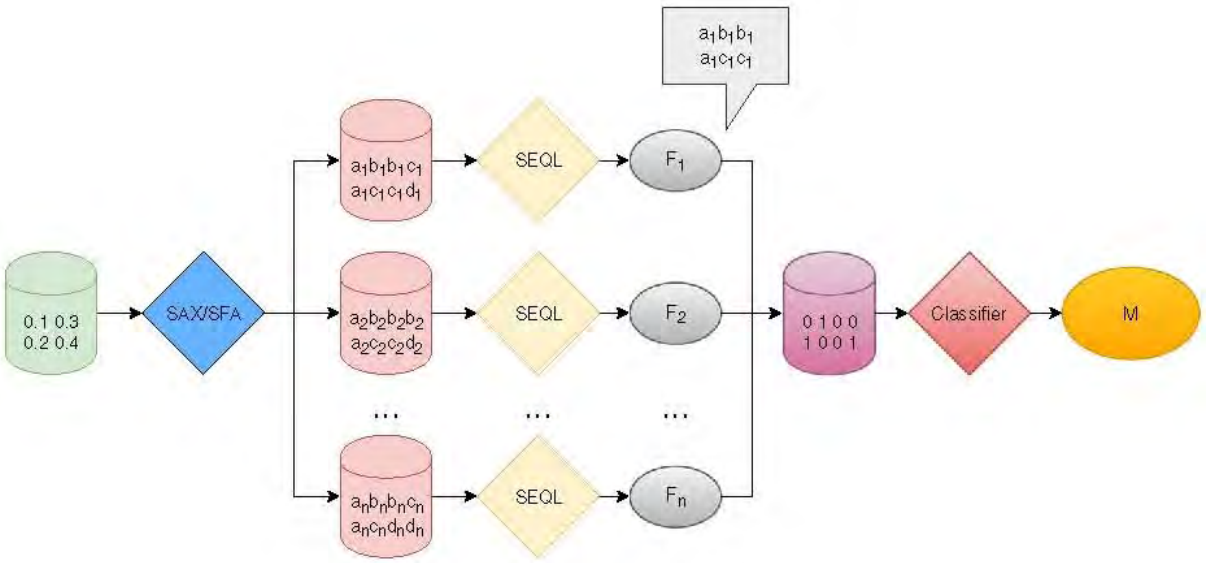
4.3.6 *Time Series Classification with multiple symbolic representations and SEQL (Mr-SEQL)*

O *Mr-SEQL* é um modelo baseado em *shaplet*, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

O *Mr-SEQL* é um software de classificação de séries temporais que utiliza modelos lineares (regressão logística) e múltiplas representações simbólicas de séries temporais (*SAX*, *SFA*) para fornecer um classificador de séries temporais preciso e interpretável (NGUYEN et al., 2019).

A *SAX* é um método de transformação para converter um vetor numérico em uma representação simbólica, ou seja, uma sequência de símbolos de um alfabeto predefinido a. *SAX* primeiro calcula a *Piecewise Aggregate Approximation* (*PAA*) de uma série temporal

Figura 35 – SEQL como método de seleção de recursos. Os recursos são selecionados de várias resoluções e/ou vários domínios de representações simbólicas e alimentadas a um modelo de regressão logística



Fonte: (NGUYEN et al., 2019).

no domínio do tempo e, em seguida, transforma essa aproximação em uma representação simbólica (NGUYEN et al., 2019).

A *PAA* reduz uma série temporal de comprimento L a um vetor de comprimento l (l < L é também o comprimento da sequência simbólica) dividindo a série temporal em segmentos iguais. Cada segmento é então substituído por seu valor médio (NGUYEN et al., 2019).

A *PAA* é então, seguida por uma etapa de discretização que substitui cada valor da *PAA* por um símbolo correspondente. O símbolo é selecionado no alfabeto com base no intervalo em que o valor cai. Existem intervalos, tantos quanto o tamanho do alfabeto. Cada intervalo está associado a um símbolo do alfabeto. Assumindo que a série temporal tem distribuição normal, os intervalos são divididos sob a distribuição normal (ou seja, N (0,1)) com probabilidade igual (NGUYEN et al., 2019).

A *SFA* também transforma uma série temporal em uma representação simbólica, mas desta vez usando o domínio da frequência para a discretização. As principais diferenças entre *SAX* e *SFA* são as opções de técnicas de aproximação e discretização. A *SFA* usa um método *DFT* para aproximar uma série temporal (NGUYEN et al., 2019).

O *SEQL* é um modelo de aprendizagem de sequência simbólica (por exemplo, pode tomar como sequências de entrada "abcd aabc ...") que percorre com eficiência um grande espaço de recursos e seleciona as subsequências mais discriminativas para um modelo linear, com base em um conjunto de dados de treinamento (NGUYEN et al., 2019). A Figura 35 apresenta a parte *SEQL* do modelo *Mr-SEQL*.

4.3.7 *Random Convolutional Kernel Transform (ROCKET)*

O *ROCKET* é um modelo baseado em *kernel*, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

O *ROCKET* transforma séries temporais usando um grande número de *kernels* convolucionais aleatórios, ou seja, *kernels* com comprimento, pesos, enviesamento, dilatação e preenchimento aleatórios. Os recursos transformados são usados para treinar um classificador linear. Para todos, exceto os maiores conjuntos de dados, é usado um classificador de regressão *Ridge*, que tem a vantagem de validação cruzada rápida para a regularização do hiperparâmetro (e nenhum outro hiperparâmetro). No entanto, como a regressão logística é treinada usando descida gradiente estocástica, é mais escalável para conjuntos de dados muito grandes. A regressão logística é usada quando o número de exemplos de treinamento é substancialmente maior do que o número de recursos (DEMPSTER; PETITJEAN; WEBB, 2020).

Quatro características distinguem o *ROCKET* das camadas convolutivas como as usadas em camadas convolutivas típicas de redes neurais e de trabalhos anteriores utilizando *kernels* convolucionais (incluindo *kernels* aleatórios) com séries temporais: (DEMPSTER; PETITJEAN; WEBB, 2020):

1. O *ROCKET* usa um grande número de *kernels*. Como existe apenas uma única "camada"de *kernels*, e como os pesos do *kernel* não são aprendidos, o custo de calcular o convoluções é baixo e é possível usar um grande número de *kernels* com relativamente pouca despesa computacional (DEMPSTER; PETITJEAN; WEBB, 2020).

2. O *ROCKET* usa uma grande variedade de *kernels*. Em contraste com as típicas de redes neurais convolucionais, nas quais é comum que grupos de *kernels* compartilhem o mesmo tamanho, dilatação e preenchimento, para o *ROCKET*, cada *kernel* tem comprimento, dilatação e preenchimento aleatórios, bem como pesos e enviesamento aleatórios (DEMPSTER; PETITJEAN; WEBB, 2020).

3. Em particular, o *ROCKET* faz uso chave da dilatação do *kernel*. Em contraste com o típico uso de dilatação em redes neurais convolucionais, nas quais a dilatação aumenta exponencialmente com profundidade, foi amostrada a dilatação aleatoriamente para cada *kernel*, produzindo uma grande variedade de dilatação do *kernel*, capturando padrões em diferentes frequências e escalas, que é crítico para o desempenho do método (DEMPSTER; PETITJEAN; WEBB, 2020).

4. Além de usar o valor máximo dos mapas de características resultantes (amplamente falando, semelhante ao *global max pooling*), o *ROCKET* usa um adicional e característica nova: a proporção de valores positivos (ou ppv). Isso permite um classificador

> para ponderar a prevalência de um determinado padrão dentro de uma série temporal. Isto é o único elemento da arquitetura do *ROCKET* que é mais crítico para sua precisão (DEMPSTER; PETITJEAN; WEBB, 2020).

Além disso, embora haja a combinação do *ROCKET* com as formas de regressão logística, de fato, uma rede neural convolucional de camada única com pesos de *kernel* aleatórios, em que os recursos transformados formam a entrada para uma camada *softmax* treinada, há importantes diferenças entre o *ROCKET* e as outras arquiteturas de rede neural. Em particular: (a) O *ROCKET* não usa uma camada oculta ou quaisquer não linearidades; (b) os recursos produzidos pelo *ROCKET* são independentes entre si (não há "conexões"entre os núcleos convolucionais); e (c) estritamente falando, o *ROCKET* não exige o uso de um classificador específico e, de fato, sugere-se o uso de classificadores diferentes (ou seja, o classificador de regressão *Ridge* ou regressão logística) em diferentes contextos (DEMPSTER; PETITJEAN; WEBB, 2020).

Em outras palavras, ainda conforme (DEMPSTER; PETITJEAN; WEBB, 2020), o *ROCKET* usa um grande número de *kernels* de convolução parametrizados aleatoriamente aplicados para cada instância. À medida que cada *kernel* é aplicado a uma série, o valor máximo e a proporção de valores positivos são registrados e concatenados em um vetor de recursos. Esses recursos são então usados para construir um classificador de regressão linear, com validação cruzada incorporada para selecionar o parâmetro alfa.

Os autores (MIDDLEHURST et al., 2021) comentam que, para cada *kernel* gerado, os parâmetros são selecionados a partir do seguinte espaço: O comprimento, l, é selecionado de modo que, $l \in \{7, 9, 11\}$ o valor de cada peso, w_i, é amostrado aleatoriamente a partir de uma distribuição normal $\sim N(0, 1)$, e são então centrados na média; *bias* b é amostrado a partir de uma distribuição uniforme $\sim U(-1, 1)$; a dilatação, a, é amostrada de uma escala exponencial até o comprimento da série; a decisão binária de preencher a série p é escolhida com igual probabilidade, se verdadeiro, a série é preenchida com zeros no início e no fim igualmente, de modo que o elemento do meio do *kernel* é aplicado a todos os pontos da série de entrada. O *stride* está sempre definido como 1. Para conjuntos de dados multivariados, cada *kernel* recebe um número aleatório de dimensões selecionadas aleatoriamente. O *kernel* para o caso multivariado ainda é um dimensional, mas com pesos diferentes para cada dimensão. O máximo e a proporção de valores positivos são calculados em todas as dimensões selecionadas.

4.3.8 *Arsenal ensemble*

O *Arsenal* é um modelo baseado em *kernel*, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

Os autores (MIDDLEHURST et al., 2021) ponderam que o *ROCKET* é um classificador muito rápido com precisão de última geração, e acreditam que é o desenvolvimento recente mais importante no campo. Representa uma classe diferente de abordagem e, como tal, é candidata à assimilação no coletivo. No entanto, surge um problema ao tentar incluir *ROCKET* no *HIVE-COTE V2*: o regressor *Ridge* usado pelo *ROCKET* é difícil de configurar para produzir valores de probabilidade úteis para cada classe ao fazer previsões. A estrutura *CAWPE* de conjunto de *HIVE-COTE V2* usa probabilidades ponderadas e depende de classificadores para produzir uma distribuição representativa da força dos classificadores de crença em previsões. Uma solução para isso seria substituir o regressor *Ridge* por um classificador que produz estimativas de probabilidades representativas. No entanto, a experimentação com classificadores de substituição adequados não produziu um modelo candidato que fosse tão preciso quanto o regressor *Ridge* para o *ROCKET*.

Para resolver este problema, os autores de (MIDDLEHURST et al., 2021) desenvolveram uma versão do *ROCKET* usada no *HIVE-COTE V2*, que é um *ensemble* de classificadores *ROCKET* menores. Denominaram este *ensemble* de *ROCKET*s como o *Arsenal*. Novos casos são classificados por maioria de votos. *Arsenal* é mais lento de construir do que *ROCKET*, mas suas melhores probabilidades fazem é um candidato melhor para *HIVE-COTE V2*. O *Arsenal* é descrito mais formalmente na Figura 36.

4.3.9 *Hierarchical Vote Collective of Transformation-based Ensembles (HIVE-COTE) V2*

O *HIVE-COTE V2* é um modelo híbrido, especializado em classificação de séries temporais que faz parte do acervo da biblioteca *Sktime* (LÖNING et al., 2019), (PONG, 2020).

O artigo (MIDDLEHURST et al., 2021) mostra que o *HIVE-COTE V2* substituiu três dos quatro classificadores que compunham o *HIVE-COTE V1*. Os novos módulos de componentes são: a transformação *Shapelet* baseada em *Shapelet Transform Classifier*; o conjunto baseado em convolução dos classificadores *ROCKET*, chamado de *Arsenal*; a representação baseada em dicionário *TDE*; e o *DrCIF* baseado em intervalo. Uma visão geral da estrutura atualizada do *HIVE-COTE V2* é exibida na Figura 37. Cada componente é treinado independentemente e, além do modelo final, é necessário produzir uma estimativa de sua precisão em dados não vistos. Para novos dados, cada módulo produz uma estimativa probabilística para cada classe. O controlador constrói uma distribuição inclinada através exponenciação (usando $= 4$ por padrão) para atenuar as diferenças nos classificadores e ponderado com a estimativa de precisão. Cada módulo do *HIVE-COTE V2* contém novos recursos e melhorias em relação às versões anteriores. Isso inclui um novo modelo com melhorias, extensões multivariadas e melhorias de contratação. Além disso, o método para estimar a precisão dos dados do treinamento foi melhorado. Genericamente existem três maneiras de estimar a precisão do teste dos dados no treinamento.

Figura 36 – Algorithm 2 - Descrição mais formal do Arsenal

Algorithm 2 Arsenal(A list of n cases of length m with d dimensions, $\mathbf{T} = (\mathbf{X}, \mathbf{y})$)

Parameters: the ensemble size, r and the number of kernels per classifier, k (default r=25 and $k = 2000$)

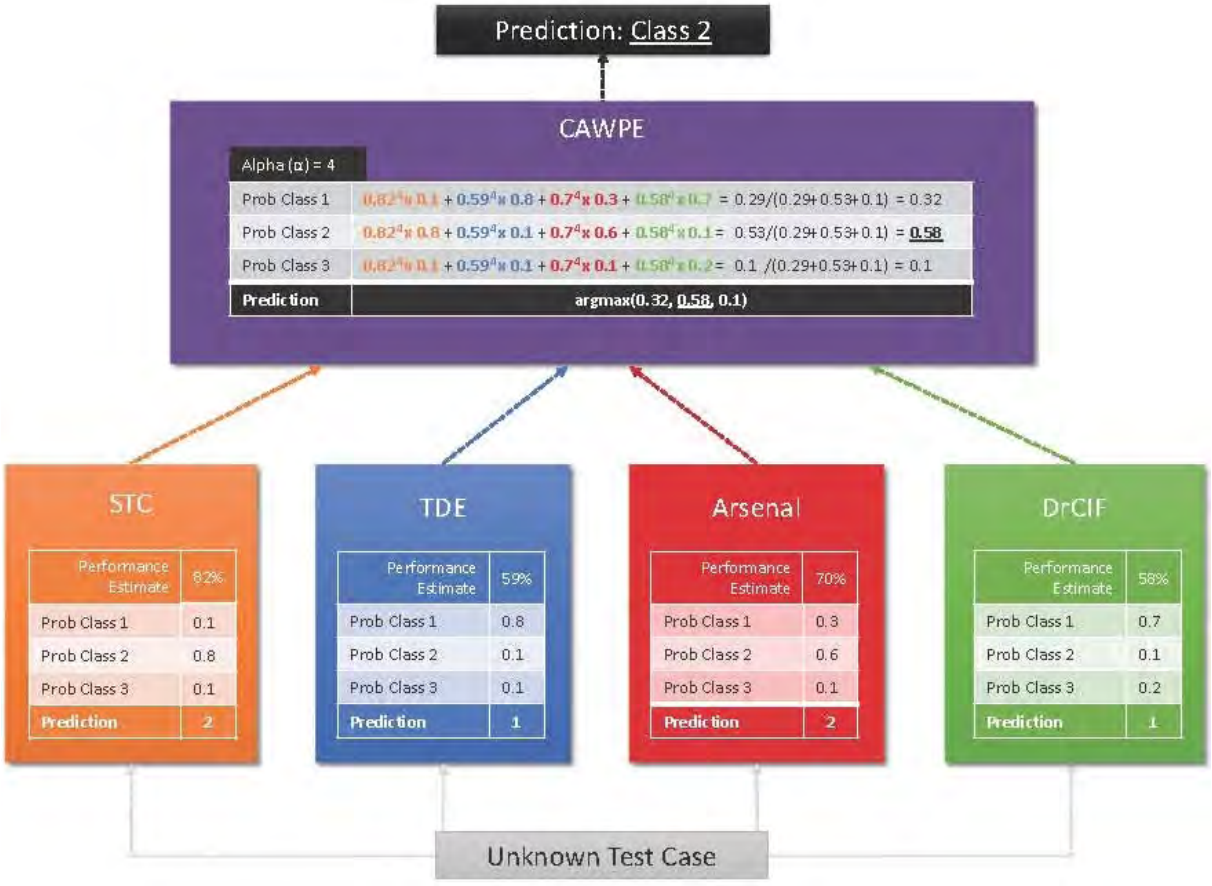
Let $\mathbf{F}$ be a list of ROCKET classifiers $(\mathbf{f}_1, \ldots, \mathbf{f}_r)$
for $i \leftarrow 1$ to r **do**
  Let $\mathbf{X}'$ be a n by $2k$ list of transformed cases
  for $j \leftarrow 1$ to k **do**
    $[l, w, b, a, p, O] \leftarrow randomKernelParameters(m, d)$
    for $t \leftarrow 1$ to n **do**
      $max \leftarrow -\infty$
      $ppv \leftarrow 0$
      for $g \leftarrow 1$ to $(m + (2p)) - ((l-1)a)$ **do**
        $v \leftarrow 0$
        for $c \leftarrow 1$ to $|O|$ **do**
          $v \leftarrow v + applyKernel(\mathbf{X}_t, g, O_c, l, w, b, a)$
        end for
        if $v > max$ **then**
          $max \leftarrow v$
        end if
        if $v > 0$ **then**
          $ppv \leftarrow ppv + 1$
        end if
      end for
      $\mathbf{X}'_{t,2j-1} \leftarrow max$
      $\mathbf{X}'_{t,2j} \leftarrow ppv/((m + (2p)) - ((l-1)a))$
    end for
  end for
  $f_i.buildRidgeClassifierCV([\mathbf{X}', y])$
end for

Fonte: (MIDDLEHURST et al., 2021).

Em primeiro lugar, pode ser avaliado o modelo final diretamente no treinamento dados. É provável que seja tendencioso e excessivamente otimista, especialmente se alguma forma de seleção de modelo ocorreu sem uma regularização cuidadosa. Em segundo lugar, pode ser feita a validação cruzada nos dados do treinamento, além da compilação final. Enquanto isso provavelmente será menos tendencioso (e geralmente pessimista), pois consome muito tempo. Em terceiro lugar, pode ser usada alguma forma de avaliação de validação incorporada na construção completa do modelo, como *bagging*. O *HIVE-COTE V1* usa uma mistura de abordagens com base na classificação.

O *HIVE-COTE V2* adota uma abordagem de *bagging* padronizada, o que é possível uma vez que todos os quatro classificadores no *HIVE-COTE V2* usam *ensemble*. No entanto, foi descoberto que, ao usar o desempenho *out of bagging* que produz estimativas aceitáveis de precisão do teste, os classificadores *bagging* eram significativamente menos precisos do que os classificadores construídos nos dados completos. Portanto, foi adotada uma abordagem híbrida. Em vez de construir 11 modelos totais para validação cruzada

Figura 37 – Uma visão geral da estrutura do conjunto de *HIVE-COTE V2* para um problema de três classes. Cada módulo é treinado de forma independente e produz uma estimativa da probabilidade de associação de cada classe para dados não vistos. A unidade de controle *CAWPE* combina estas probabilidades, ponderadas por uma estimativa da qualidade do módulo encontrada nos dados do treinamento.

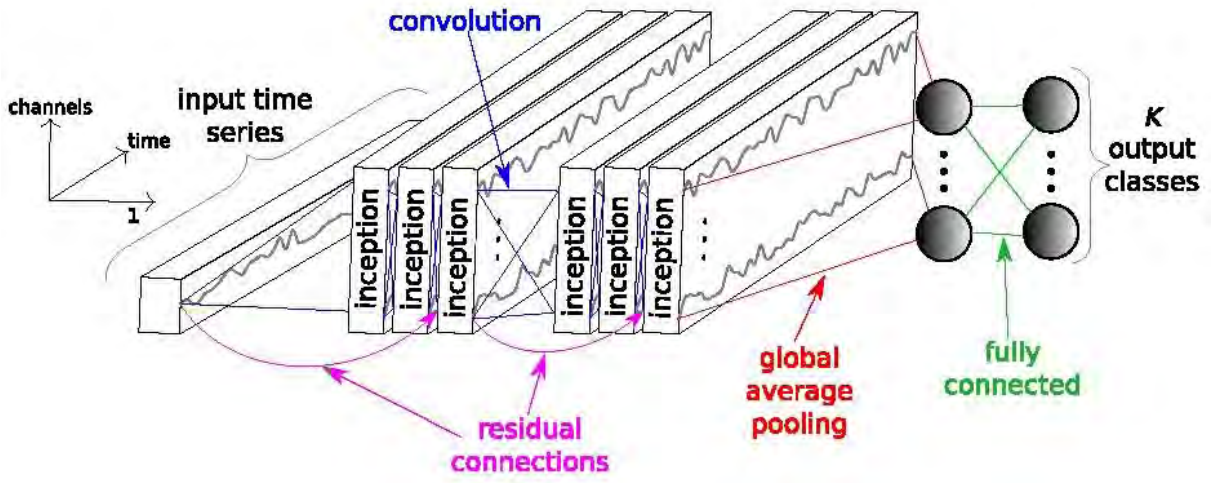


Fonte: (MIDDLEHURST et al., 2021).

de dez dobras ou um único modelo usando *bagging*, foi construído um modelo *bagging* para estimar a precisão para os classificadores que o exigem, e um modelo completo para prever novos casos (MIDDLEHURST et al., 2021).

O *HIVE-COTE V2* pode treinar cada componente simultaneamente. Mesmo assim, os componentes podem ser lentos em grandes problemas. Portanto, é permitido que o usuário configure o *HIVE-COTE V2* para que tenha um contrato de tempo. Se contraído, cada conjunto de componentes simplesmente constrói tantos classificadores básicos quanto possível no tempo fornecido. Este formulário simples em contrair é um primeiro x adequado. No entanto, surge um problema para grandes dados com contratos curtos: a construção de um único membro do conjunto pode exceder o contrato. Seria melhor que os componentes se autoconfigurassem quando isso é provável, por exemplo, por subamostragem de casos ou séries. *HIVE-COTE V2* é rosqueado, mas, atualmente, os próprios componentes não são. Como todos são conjuntos de classificadores básicos independentes, é, em princípio, fácil fazê-lo. Esta mudança inevitável para a GPU faz parte do plano de trabalho futuro (MIDDLEHURST et al., 2021).

Figura 38 – *InceptionTime* para classificação de séries temporais.



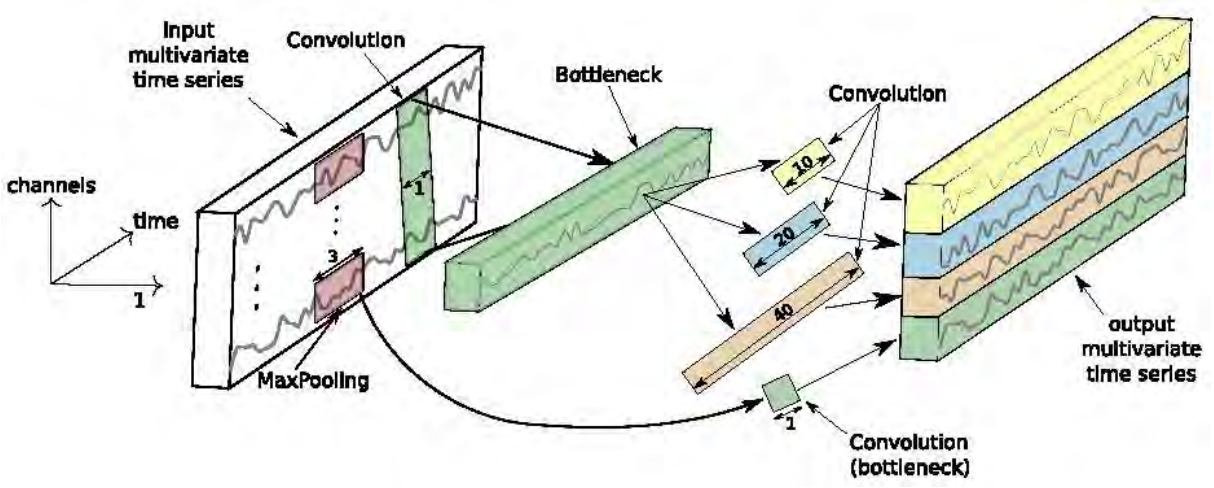
Fonte: (FAWAZ et al., 2020).

4.3.10 *InceptionTime*

A composição de um classificador *InceptionTime* contém dois blocos residuais diferentes. Para a rede *Inception*, cada bloco é composto por três módulos *Inception*, em vez de camadas tradicionais totalmente convolucionais. A entrada de cada bloco residual é transferida por meio de uma conexão linear de atalho para ser adicionada à entrada do próximo bloco, mitigando assim o problema do gradiente de desaparecimento, permitindo um fluxo direto do gradiente. Seguindo esses blocos residuais, é empregada uma camada *Global Average Pooling* (*GAP*) que calcula a média da série temporal multivariada de saída ao longo de toda a dimensão do tempo. Por fim, é usada uma camada *softmax* tradicional final totalmente conectada com um número de neurônios igual ao número de classes no conjunto de dados (FAWAZ et al., 2020). A Figura 38 descreve a arquitetura da rede *Inception*, mostrando 6 módulos *Inception* diferentes empilhados um após o outro.

Quanto ao módulo *Inception*, a Figura 39 ilustra os detalhes internos desta operação. Considerando a entrada como uma *Multivariate Time Series* (*MTS*) com dimensões M. O primeiro componente principal do módulo de Iniciação é chamado de camada de "gargalo". Esta camada realiza uma operação de filtros deslizantes de comprimento 1 com uma passada igual a 1. Isso transformará a série temporal de um *MTS* com dimensões de M para m *MTS* com dimensões $m \ll M$, reduzindo assim significativamente a dimensionalidade da série temporal, bem como a complexidade do modelo e mitigação de problemas de *overfitting* para pequenos conjuntos de dados. Observe que, para fins de visualização, A Figura 34 ilustra uma camada de gargalo com m = 1. Por fim, deve-se mencionar que esta técnica de gargalo permite que a rede *Inception* tenha filtros muito mais longos do que *ResNet* (quase dez vezes) com aproximadamente o mesmo número de parâmetros para ser aprendido, uma vez que sem a camada de gargalo, os filtros terão dimensões M comparadas para $m \ll M$ ao usar a camada de gargalo. O segundo componente principal do

Figura 39 – Detalhes internos da operação do módulo *Inception*.



Fonte: (FAWAZ et al., 2020).

módulo de iniciação desliza vários filtros de diferentes comprimentos simultaneamente na mesma série temporal de entrada. Por exemplo, na Figura 38, três convoluções diferentes com comprimento $l \in \{10, 20, 40\}$ são aplicados ao *MTS* de entrada, que é tecnicamente a saída da camada de gargalo. Além disso, a fim de tornar o modelo invariante a pequenas perturbações, é introduzida outra operação de *MaxPooling* paralela, seguida por uma camada de gargalo para reduzir a dimensionalidade. Na saída desliza-se uma janela *MaxPooling* que é computada, tomando o valor máximo nesta janela de série temporal. Finalmente, a saída de cada convolução paralela independente *MaxPooling* é concatenada para formar o *MTS* de saída. As últimas operações são repetidas para cada módulo da *Inception* individual da rede proposta (FAWAZ et al., 2020).

Ao empilhar vários módulos *Inception* e treinar os pesos (valores dos filtros) via retropropagação, a rede é capaz de extrair recursos hierárquicos latentes de várias resoluções graças ao uso de filtros com vários comprimentos. Para completar, especifica-se o número exato de filtros para nosso módulo de iniciação proposto: 3 conjuntos de filtros, cada um com 32 filtros $l \in \{10, 20, 40\}$ de comprimento com *MaxPooling* adicionado à mistura, assim tornando o número total de filtros por camada igual a $32 \times 4 = 128 =$ M - a dimensionalidade do *MTS* de saída. O valor padrão do tamanho do gargalo foi definido como m = 32 (FAWAZ et al., 2020).

O *InceptionTime* do "estado da arte" proposto é um conjunto de 5 redes *Inception*, com cada predição tendo um par de peso. Na verdade, durante a experimentação, foi observado que uma única rede *Inception* exibe alto desvio padrão na precisão. Provavelmente essa variabilidade vem tanto dos pesos inicializados aleatoriamente quanto do próprio processo de otimização estocástica. Devido a esta instabilidade de treinar apenas uma instância, potencialmente muito boa ou muito ruim, da rede *Inception*, surgiu a idéia de tirar proveito dessa instabilidade por meio de *ensembling*, criando *InceptionTime*. A seguinte equação explica o conjunto de previsões feitas por uma rede com inicializações diferentes (FAWAZ

et al., 2020):

$$\hat{y}_{i,c} = \frac{1}{n}\sum_{j=1}^{n} \sigma_c(x_i, \theta_j) | \forall c \in [1, C] \tag{4.1}$$

Com $\hat{y}_{i,c}$ denotando a probabilidade de saída do conjunto de ter a série temporal de entrada x_i pertencente à classe c , que é igual à saída logística σ_c calculado sobre os n modelos inicializados aleatoriamente. Quanto ao modelo proposto, o número de classificadores individuais escolhido, através de experimentos, foi igual a 5 (FAWAZ et al., 2020).

O conceito de *Receptive Field* (*RF*) é uma ferramenta essencial para a compreensão de *CNN*s em profundidade. Ao contrário de redes totalmente conectadas ou *Perceptrons Multi-Layer*, um neurônio em uma *CNN* depende apenas de uma região do sinal de entrada. Essa região no espaço de entrada é chamada de campo receptivo daquele neurônio específico. Para problemas de visão computacional, este conceito foi amplamente estudado, no qual os autores compararam os campos receptivos efetivos e teóricos de uma *CNN* para segmentação de imagens.

Para dados temporais, o campo receptivo pode ser considerado como um valor teórico que mede o campo de visão máximo de uma rede neural em um espaço unidimensional: quanto maior for, melhor se torna a rede (em teoria) na detecção de padrões mais longos. Agora é fornecida a definição de *RF* para dados de séries temporais, supondo que estivesse deslizando convoluções com um passo igual a 1. A fórmula para calcular a *RF* para uma rede de profundidade d com cada camada tendo um comprimento de filtro igual a k_i com $i \in [1, d]$ é (FAWAZ et al., 2020):

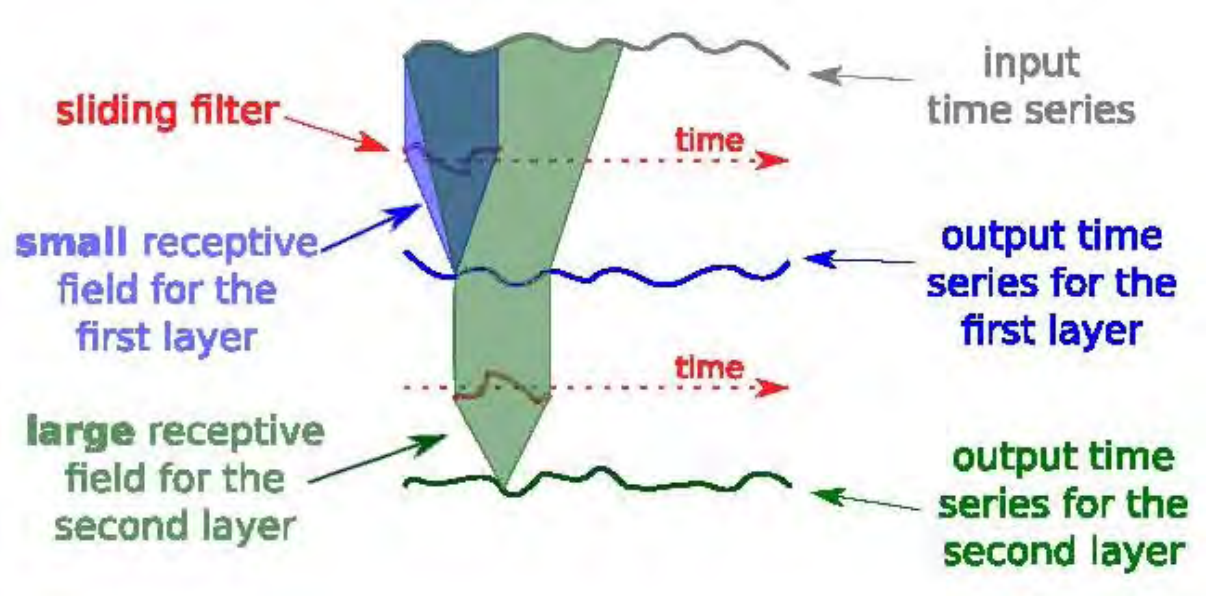
$$(1) + \sum_{i=1}^{d} (k_i - 1) \tag{4.2}$$

Ao analisar a Eq. 4.2, pode-se observar claramente que adicionar duas camadas ao conjunto inicial de d camadas aumentará apenas ligeiramente o valor de *RF*. Na verdade, neste caso, se o antigo *RF* valor é igual a *RF*', o novo valor *RF* será igual a RF' + 2 × (k - 1). Por outro lado, aumentando o comprimento do filtro k_i, $\forall i \in [1, d]$ por 2, o novo valor RF será igual a RF' + 2 × d. Isso é bastante esperado, pois ao aumentar o comprimento do filtro para todas as camadas, na verdade, está aumentando o *RF* para cada camada da rede. A Figura 40 ilustra o *RF* para um *CNN* de duas camadas (FAWAZ et al., 2020).

4.3.11 Resumo das características dos modelos de classificação

A Tabela 5 mostra as principais características dos 10 modelos de classificação de séries temporais testados.

Figura 40 – Ilustração de campo receptivo para uma CNN de duas camadas.



Fonte: (FAWAZ et al., 2020).

Tabela 5 – Resumo das características dos modelos de classificação

Classificador	**Características**
Time Series Forest (***TSF***)	*TSF* é uma modificação do modelo de *RF* para a configuração de série temporal. Estudos experimentais mostram que *TSF* usando recursos simples, como média, desvio padrão e a inclinação é computacionalmente eficiente e supera concorrentes fortes, como classificadores da vizinhança mais próxima com sincronização temporal dinâmica.
Random Interval Spectral Forest (***RISE***)	Como *TSF*, foram construídas árvores em intervalos dos dados para construir um classificador de *RF*. A principal diferença, no entanto, é que enquanto o *TSF* usa recursos de domínio do tempo calculando a média, variância e inclinação de cada intervalo, o *RISE* extrai características espectrais sobre cada intervalo aleatório. São selecionados n intervalos aleatórios (por exemplo 500) e calculadas características espectrais para cada intervalo de forma independente.

Continua

Tabela 5 – *Continuação*

Classificador	**Características**
*kNN-Based Time-Series Classification (**kNN-TSC**)*	A classificação baseada em vizinhos mais próximos é um tipo de aprendizagem baseada em instância ou aprendizagem não generalizante: O modelo de similaridade é especializado em séries temporais, baseado na abordagem de casamento de subsequências, que permite gaps não emparelhados, escala de amplitude e tradução de offset (deslocamento). O método de combinação de decisão adotado foi proposto por (YANG et al., 1999) como método de combinação de decisão referido em (LEE et al., 2012) como método de média estratificada e resolve o problema de classes assimétricas (desbalanceadas).
*Contcactable Bag of Symbolic Fourier Approximation Symbols (**cBOSS**)*	*cBOSS* é uma evolução do *BOSS* (para funcionamento do *BOSS* veja subseção *cBOSS*), se concentram principalmente na técnica de conjunto do classificador, o que é computacionalmente caro e imprevisível. *Ensembling* foi mostrado ser um componente essencial do *BOSS*, resultando em significativamente mais precisão. Foi avaliada a possibilidade de substituição o mecanismo de conjunto do *BOSS* por um esquema mais estável e eficiente sem uma redução significativa na precisão.
Word Extraction for Time Series Classification (**WEASEL**)	*WEASEL* é um classificador baseado em dicionário que é uma extensão do *BOSS* (para funcionamento do *BOSS* veja subseção *cBOSS*). *WEASEL* é um único classificador em vez de um *ensemble*.
*Time Series Classification with multiple symbolic representations and SEQL (**Mr-SEQL**)*	*Mr-SEQL* é um modelo de classificação de séries temporais que utiliza modelos lineares (regressão logística) e múltiplas representações simbólicas de séries temporais (*SAX*, *SFA*) para fornecer um classificador de séries temporais preciso e interpretável.

Continua

Tabela 5 – *Continuação*

Classificador	**Características**
*Random Convolutional kernel Transform (**ROCKET**)*	*ROCKET* transforma séries temporais usando um grande número de *kernels* convolucionais aleatórios, ou seja, *kernels* com comprimento, pesos, enviesamento, dilatação e preenchimento aleatórios. Os recursos transformados são usados para treinar um classificador linear. Para todos, exceto os maiores conjuntos de dados, usa-se um classificador de regressão *Ridge*, que tem a vantagem de validação cruzada rápida para a regularização do hiperparâmetro (e nenhum outro hiperparâmetro). No entanto, como regressão logística treinada usando descida gradiente estocástica é mais escalável para conjuntos de dados muito grandes, a regressão logística é usada quando o número de exemplos de treinamento é substancialmente maior do que o número de recursos.
***Arsenal** ensemble*	O *Arsenal* é uma versão do *ROCKET* usada no *HIVE-COTE V2*. É um conjunto de classificadores *ROCKET* menores. Novos casos são classificados por maioria de votos. *Arsenal* é mais lento de construir do que *ROCKET*, mas suas melhores probabilidades fazem é um candidato melhor para *HIVE-COTE V2*.

Continua

Tabela 5 – *Continuação*

Classificador	**Características**
Hierarchical Vote Collective of Transformation-based Ensembles (***HIVE-COTE V2***)	O *HIVE-COTE V2* substituiu três dos quatro classificadores que compõem o *HIVE-COTE V1*. Os módulos de componentes são: a transformação *Shapelet* baseada em *Shapelet Transform Classifier*; o *ensemble* baseado em convolução dos classificadores *ROCKET*, chamado de *Arsenal*; a representação baseada em dicionário *TDE*; e o *DrCIF* baseado em intervalo. *HIVE-COTE V2* adota uma abordagem de *bagging* padronizada, o que é possível uma vez que todos os quatro classificadores no *HIVE-COTE V2* usam *ensemble*. O *HIVE-COTE V2* pode treinar cada componente simultaneamente. Mesmo assim, os componentes podem ser lentos em grandes problemas. Portanto, é permitido que o usuário configure o *HIVE-COTE V2* para que tenha um contrato de tempo.
InceptionTime	A composição de um classificador de rede *Inception* contém dois blocos residuais diferentes. Para a rede *Inception*, cada bloco consiste em três módulos *Inception* em vez de camadas totalmente convolucionais tradicionais. A entrada de cada bloco residual é transferida por meio de uma conexão de atalho linear que será adicionada à entrada do próximo bloco, mitigando assim o problema do desaparecimento do gradiente, permitindo um fluxo gradiente direto. Seguindo esses blocos residuais, é utilizada uma camada *GAP*, que tem a função de realizar a média das séries temporais multivariadas de saída ao longo de toda a dimensão temporal. Finalmente, uma camada *softmax* totalmente conectada tradicional é usada com um número de neurônios igual ao número de classes no conjunto de dados.

Fonte: Organização do autor.

4.4 MÉTRICAS UTILIZADAS PARA MENSURAR OS RESULTADOS DA CLASSIFICAÇÃO DE SÉRIES TEMPORAIS

Na aplicação de modelos de *ML* a problemas reais, em geral, o conhecimento que se tem do domínio sendo investigado e provido unicamente pelo conjunto de exemplos, a partir do qual a indução de um modelo preditivo/ descritivo é então realizada. De maneira geral, pode-se afirmar que não existe técnica universal, ou seja, não é possível estabelecer *a priori* que um modelo de *ML* em particular será mais bem-sucedido na resolução de qualquer tipo de problema (FACELI et al., 2021).

Diversos algoritmos podem ser considerados para a solução de um dado problema. Ainda que um único algoritmo seja escolhido, pode ser necessário realizar ajustes em seus hiperparâmetros, o que leva a obtenção de múltiplos modelos para os mesmos dados (FACELI et al., 2021).

Os parágrafos anteriores evidenciam uma característica particular do domínio de *ML*: a necessidade de experimentação. De fato, a validação de qualquer nova técnica de *ML* proposta geralmente envolve a realização de experimentos controlados, em que se demonstre a sua efetividade na solução de diferentes problemas, representados por seus conjuntos de dados associados. Dessa forma, é recomendável seguir procedimentos que garantam a corretude, a validade e a reprodutibilidade dos experimentos realizados e, mais importante, das conclusões obtidas a partir de seus resultados (FACELI et al., 2021).

Essa avaliação experimental de um algoritmo de *ML* pode ser realizada segundo diferentes aspectos, tais como acurácia do modelo gerado, compreensibilidade do conhecimento extraído, tempo de aprendizado, requisitos de armazenamento do modelo, entre outros. Considerando os modelos preditivos, a discussão será concentrada em medidas relacionadas ao desempenho obtido nas predições realizadas. Muitos dos conceitos e procedimentos discutidos são facilmente generalizáveis a outros tipos de medidas de desempenho (FACELI et al., 2021).

A avaliação de um modelo de *ML* supervisionado é normalmente realizada por meio da análise do desempenho do preditor gerado pole modelo na rotulação de novos objetos, não apresentados previamente em seu treinamento (FACELI et al., 2021).

Em problemas de classificação binária, predições podem ter quatro possíveis classes:

- Verdadeiro Positivo (VP): quando o método diz que a classe é positiva e, ao verificar a resposta, vê-se que a classe era realmente positiva;
- Verdadeiro Negativo (VN): quando o método diz que a classe é negativa e, ao verificar a resposta, vê-se que a classe era realmente negativa;
- Falso Positivo (FP): quando o método diz que a classe é positiva, mas ao verificar a resposta, vê-se que a classe era negativa;

- Falso Negativo (FN): quando o método diz que a classe é negativa, mas ao verificar a resposta, vê-se que a classe era positiva;

Uma alternativa simples para visualizar o desempenho de um classificador é com o uso de uma matriz de confusão, ilustrada na Tabela 6. Essa matriz ilustra o número de predições corretas e incorretas em cada classe. Para um determinado conjunto de dados, as linhas dessa matriz representam as classes verdadeiras, e as colunas, as classes preditas pelo classificador. Logo, cada elemento m_{ij} de uma matriz de confusão M_c apresenta o número de exemplos da classe i classificados como pertencentes a classe j. Para k classes, Mc tem então dimensão k x k. A diagonal apresenta os acertos do classificador, enquanto os outros elementos correspondem aos erros cometidos nas suas predições. Por meio do exame dessa matriz, têm-se medidas quantitativas de quais classes o modelo de aprendizado tem maior dificuldade (FACELI et al., 2021).

Tabela 6 – Matriz de confusão de classificação binária

		Classe predita	
		Positiva	Negativa
Classe real	Positiva	VP	FN
	Negativa	FP	VN

Fonte: Elaborado pelo autor.

A Tabela 7 apresenta as fórmulas e as interpretações das principais métricas de avaliação binária de modelos de classificação de dados baseadas no Tabela 6 Matriz de confusão de classificação binária.

Tabela 7 – Principais métricas de avaliação binária de modelos de classificação de dados

Métrica	**Fórmula**	**Interpretação**
Acurácia	(VP+VN) / N	Avalia simplesmente o percentual de acertos, ou seja, a acurácia pode ser obtida pela razão entre a quantidade de acertos e o total de entradas.
Sensibilidade (S) ou recall	VP / (VP+FN)	Avalia a capacidade do método de detectar com sucesso resultados classificados como positivos.
Especificidade	VN / (FP+VN)	Avalia a capacidade do método de detectar com sucesso resultados classificados como negativos.
Precisão (P)	VP / (VP+FP)	Avalia a quantidade de verdadeiros positivos sobre a soma de todos os valores positivos.
F1-score	2 x (PxS) / (P+S)	É uma média harmônica calculada com base na precisão e na sensibilidade.

Fonte: Elaborado pelo autor.

Neste trabalho de *TSC*, serão analisadas 3 métricas, a acurácia para avaliar o acerto de forma global, a capacidade do modelo de identificar corretamente os casos positivos através da sensibilidade ou recall e a capacidade do modelo de identificar corretamente os casos negativos através da especificidade.

4.5 CONCLUSÕES E PRÓXIMOS PASSOS

Neste capítulo foram apresentados funcionamento do nariz eletrônico, métricas utilizadas para mensurar os resultados da classificação de séries temporais, motivos que determinaram as escolhas dos modelos de IA testados para solucionar o problema e os métodos de IA testados para resolver o problema.

Os próximos passos serão: apresentar os dados levantados durante a pesquisa, os conjuntos de dados propostos para solucionar o problema, a exploração dessas bases de dados, apresentação da metodologia adotada para o processamento dos experimentos e uma discussão sobre os resultados alcançados.

5 RESULTADOS EXPERIMENTAIS

Neste capítulo serão discutidos os dados levantados durante a pesquisa, assim como as bases de dados propostas para solução do problema, a exploração dos dados levantados, a metodologia de processamento dos experimentos e os resultados alcançados.

Para alcançar o objetivo do trabalho, foram realizadas diversas leituras no nariz eletrônico com esses gêneros fúngicos e rodados vários experimentos de IA. Conforme já evidenciado, serão abordados com mais profundidade nas seções abaixo, as leituras do nariz eletrônico, uma vez que são séries temporais que são a entrada para os métodos de IA que objetivam classificar os gêneros fúngicos estudados.

5.1 DADOS LEVANTADOS DURANTE A PESQUISA

Durante o período de pesquisa e leitura de dados, foram trabalhados vários repiques de espécies fúngicas fornecidas pelo parceiro Laboratório de Taxonomia e Biotecnologia do Departamento de Micologia da UFPE do trabalho de (COUTO; MOTTA, 2021). Entre outras espécies podem-se citar *Aspergillus flavus*, *Aspergillus steynii*, *Cladosporium perangustum*, *Cladosporium vignae*, *Fusarium incarnatum*, *Fusarium pseudocinatum*, *Penicillium olsonii*, *Penicillium steckii* e *Rhizomucor pusillus*. Como o objetivo é estudar os gêneros fúngicos, agregam-se as espécies em *Aspergillus* sp., *Cladosporium* sp., *Fusarium* sp., *Penicillium* sp. e *Rhizomucor* sp..

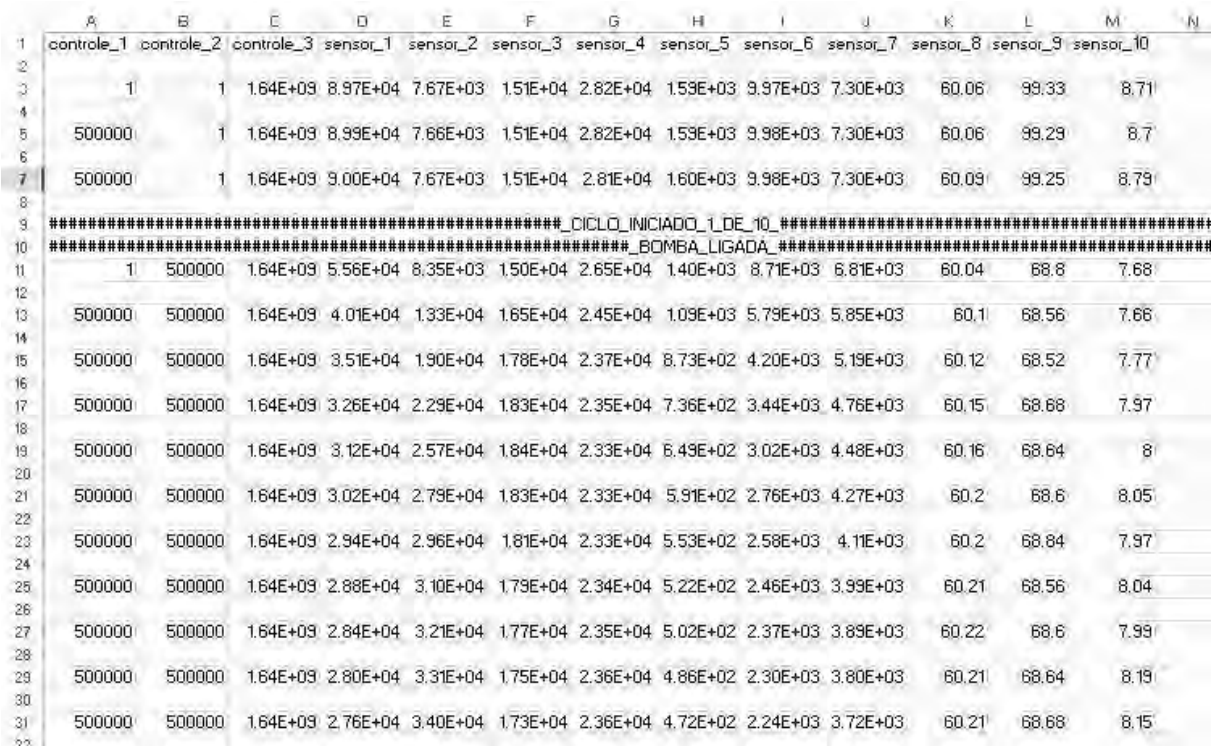
Outra variável de estudo foi se o *VOC* foi lido pelo nariz eletrônico direto da placa (quando denomina-se a leitura de "em placa") ou se o *VOC* foi lido pelo nariz eletrônico no ambiente aberto (quando denomina-se a leitura de "aberto").

A Figura 41 mostra um trecho de uma saída de leitura do nariz eletrônico. Os *labels* da linha 1 foram inseridos para fins didáticos, essa linha não faz parte da saída, a saída começa na linha 3.

A Figura 42 ilustra o ciclo leitura dos sinais dos *VOC*s da espéice *Aspergillus flavus* em placa, dos sensores do *e-Nose*, chama-se a atenção para os momentos do ciclo. O ponto (A), momento 0 segundos, indica o início do ciclo com a aspersão dos *VOC*s para câmara de sensores. O ponto (B), no momento 20 segundos, finaliza a primeira etapa e inicia a segunda etapa, a etapa de estabilização/ iteração dos *VOC*s com os sensores na câmara. O ponto (C), no momento 80 segundos, finaliza a segunda etapa e inicia a terceira e última etapa, a etapa de purga/ limpeza da câmara de sensores com a aspersão e passagem de ar filtrado em carvão ativado pelos sensores até o final do ciclo no ponto (D), no momento 140 segundos.

É relevante observar as mudanças de comportamento dos sinais dos sensores do momento (A) ao momento (B), com o início do ciclo, aspersão dos *VOC*s para câmara de

Figura 41 – Trecho de saída de uma leitura do nariz eletrônico.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	controle_1	controle_2	controle_3	sensor_1	sensor_2	sensor_3	sensor_4	sensor_5	sensor_6	sensor_7	sensor_8	sensor_9	sensor_10	
2														
3	1	1	1.64E+09	8.97E+04	7.67E+03	1.51E+04	2.82E+04	1.53E+03	9.97E+03	7.30E+03	60.06	99.33	8.71	
4														
5	500000	1	1.64E+09	8.99E+04	7.66E+03	1.51E+04	2.82E+04	1.53E+03	9.98E+03	7.30E+03	60.06	99.29	8.7	
6														
7	500000	1	1.64E+09	9.00E+04	7.67E+03	1.51E+04	2.81E+04	1.60E+03	9.98E+03	7.30E+03	60.09	99.25	8.79	
8														
9	##########_CICLO_INICIADO_1_DE_10_##########													
10	##########_BOMBA_LIGADA_##########													
11	1	500000	1.64E+09	5.56E+04	8.35E+03	1.50E+04	2.65E+04	1.40E+03	8.71E+03	6.81E+03	60.04	68.8	7.68	
12														
13	500000	500000	1.64E+09	4.01E+04	1.33E+04	1.65E+04	2.45E+04	1.09E+03	5.79E+03	5.85E+03	60.1	68.56	7.66	
14														
15	500000	500000	1.64E+09	3.51E+04	1.90E+04	1.78E+04	2.37E+04	8.73E+02	4.20E+03	5.19E+03	60.12	68.52	7.77	
16														
17	500000	500000	1.64E+09	3.26E+04	2.29E+04	1.83E+04	2.35E+04	7.36E+02	3.44E+03	4.76E+03	60.15	68.68	7.97	
18														
19	500000	500000	1.64E+09	3.12E+04	2.57E+04	1.84E+04	2.33E+04	6.49E+02	3.02E+03	4.48E+03	60.16	68.64	8	
20														
21	500000	500000	1.64E+09	3.02E+04	2.79E+04	1.83E+04	2.33E+04	5.91E+02	2.76E+03	4.27E+03	60.2	68.6	8.05	
22														
23	500000	500000	1.64E+09	2.94E+04	2.96E+04	1.81E+04	2.33E+04	5.53E+02	2.58E+03	4.11E+03	60.2	68.84	7.97	
24														
25	500000	500000	1.64E+09	2.88E+04	3.10E+04	1.79E+04	2.34E+04	5.22E+02	2.46E+03	3.99E+03	60.21	68.56	8.04	
26														
27	500000	500000	1.64E+09	2.84E+04	3.21E+04	1.77E+04	2.35E+04	5.02E+02	2.37E+03	3.89E+03	60.22	68.6	7.99	
28														
29	500000	500000	1.64E+09	2.80E+04	3.31E+04	1.75E+04	2.36E+04	4.86E+02	2.30E+03	3.80E+03	60.21	68.64	8.19	
30														
31	500000	500000	1.64E+09	2.76E+04	3.40E+04	1.73E+04	2.36E+04	4.72E+02	2.24E+03	3.72E+03	60.21	68.68	8.15	

Fonte: Elaborado pelo autor.

Obs: Os *labels* da linha 1 foram inseridos para fins didáticos, essa linha não faz parte da saída, começa na linha 3.

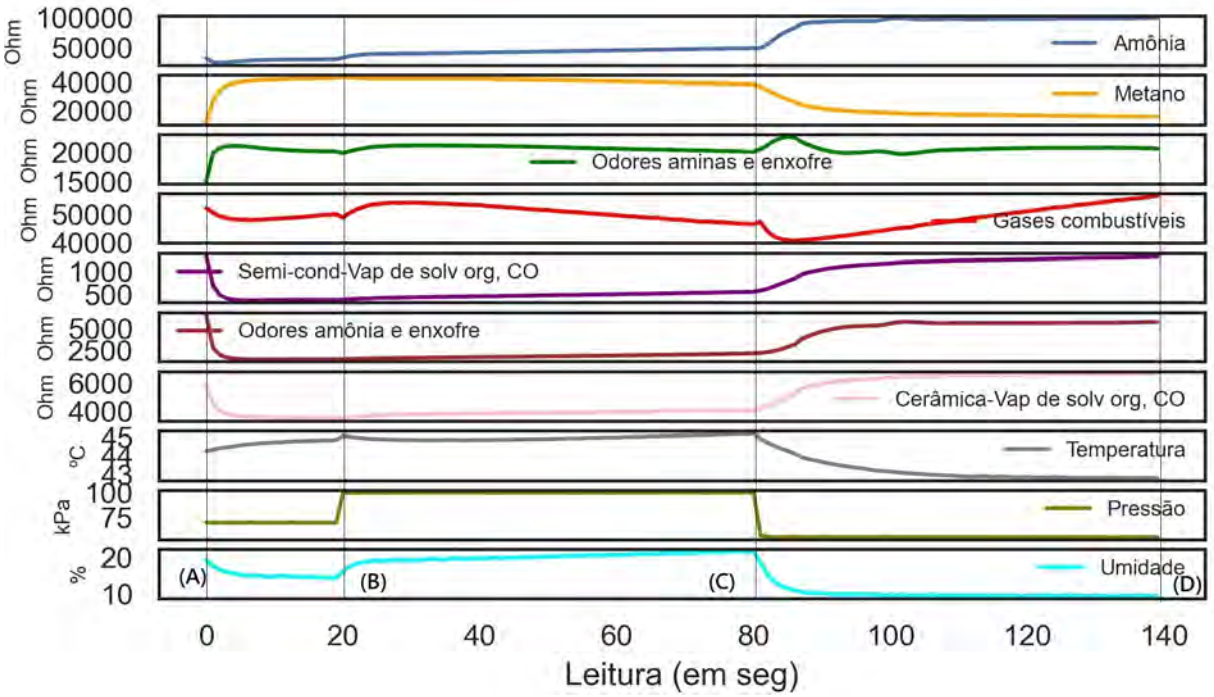
sensores. Em seguida, há uma estabilidade na maioria dos sinais dos sensores, caracterizando a etapa de estabilização/ iteração dos *VOC*s com os sensores na câmara até o momento (C), com o início da terceira e última etapa, a etapa de purga/ limpeza da câmara de sensores com a aspersão e passagem de ar filtrado em carvão ativado pelos sensores até o final do ciclo no ponto (D), quando se observa mudanças de comportamento dos sinais dos sensores, até o final do ciclo no momento (E).

Conforme visto nas Figura 41 e Figura 42, as leituras do nariz eletrônico são séries temporais que são a entrada para os métodos de IA que objetivam classificar os gêneros fúngicos estudados.

Outros dois conceitos que necessitam ser introduzidos são os de base por instância e base por ciclo. A base por instância é o empilhamento das instâncias limpas dos sensores, isto é, o empilhamento das features dos sensores pré-processadas e limpas. No caso da base por ciclos, primeiro são concatenadas todas as instâncias limpas dos sensores, isto é, os conjuntos das features dos sensores pré-processadas e limpas, para só depois os ciclos serem empilhados.

A Tabela 8 apresenta os quantitativos de instâncias x features e ciclos x features por base instância e base ciclo respectivamente para cada gênero fúngico levantado durante a pesquisa.

Figura 42 – Ciclo leitura dos sinais de *VOC*s da espéice *Aspergillus flavus* em placa, dos sensores do *e-Nose*, chama-se a atenção para os momentos do ciclo. O ponto (A), momento 0 segundos, indica o início do ciclo com a aspersão dos *VOC*s para câmara de sensores. O ponto (B), no momento 20 segundos, finaliza a primeira etapa e inicia a segunda etapa, a etapa de estabilização/ iteração dos *VOC*s com os sensores na câmara. O ponto (C), no momento 80 segundos, finaliza a segunda etapa e inicia a terceira e última etapa, a etapa de purga/ limpeza da câmara de sensores com a aspersão e passagem de ar filtrado em carvão ativado pelos sensores até o final do ciclo no ponto (D), no momento 140 segundos.



Fonte: Elaborado pelo autor.

Tabela 8 – Número de instâncias e ciclos por leitura de gênero fúngico levantados durante a pesquisa

Gênero fúngico	**Por instância**		**Por ciclo**	
	Instância	**Atributo**	**Ciclo**	**Atributo**
Aspergillus sp. em placa	7.424	10	40	2.340
Cladosporium sp. em placa	13.000	10	70	2.340
Fusarium sp. em placa	8.452	10	50	2.340
Penicillium sp. em placa	5.109	10	30	2.340
Rhizomucor sp. em placa	4.552	10	30	2.340
Cladosporium sp. aberto	563	10	5	2.340
Fusarium sp. aberto	1.213	10	10	2.340
Rhizomucor sp. aberto	1.189	10	10	2.340

Fonte: Elaborado pelo autor.

5.2 BASES DE DADOS PROPOSTAS PARA SOLUÇÃO DO PROBLEMA

Ao longo do processo de pesquisa, o aprofundamento do estudo dos dados foi progredindo e formularam-se cenários sobre como solucionar o problema. Chegou-se a 3 cenários. Inicialmente montou-se a base mais conservadora, que foi chamada de **Placa**, só com as leituras em placa, esta foi dividida em treino e teste e os experimentos foram realizados. Em seguida, deve-se escolher um dos dois caminhos possíveis: ou usar a base em placa como treinamento e reunir as leituras abertas em outra base como teste, nominando esta base como **Placa_TR_Aberto_TS ou Pl_TR_Ab_TS**. Ou formar uma outra base com as leituras em placa e leituras abertas juntas e dividi-la em treino e teste para rodar os modelos, esta outra base foi nominada de **Placa_Aberto ou Pl_Ab**. Cabe a observação de que foram agrupados os gêneros fúngicos que possuem leituras em placa e aberto (*Cladosporium* sp., *Fusarium* sp. e *Rhizomucor* sp.) para formar as bases **Placa_TR_Aberto_TS ou Pl_TR_Ab_TS** e **Placa_Aberto ou Pl_Ab**.

A Tabela 9 apresenta os quantitativos de instâncias x atributos e ciclos x atributos por base instância e base ciclo respectivamente para as bases de dados trabalhadas na pesquisa.

Tabela 9 – Número de instâncias e ciclos por base de dados utilizada na pesquisa

Base de dados	**Por instância**		**Por ciclo**	
	Instância	**Atributo**	**Ciclo**	**Atributo**
Placa	38.537	10	220	2.340
Placa_TR_Aberto_TS ou Pl_TR_Ab_TS	28.969	10	175	2.340
Placa_Aberto ou Pl_Ab	28.969	10	175	2.340

Fonte: Elaborado pelo autor.

5.3 EXPLORAÇÃO DOS DADOS LEVANTADOS

Nessa seção, serão comentados alguns aspectos relevantes que levaram ao conhecimento dos dados e contribuíram para entender o comportamento e os resultados dos modelos rodados.

A exploração dos dados dos sensores do nariz eletrônico das leituras das espécies fúngicas foi realizada através de rotinas escritas em *Python* e o formato das bases utilizado foi **por instância**. Os aspectos abordados foram:

- Leituras dos sensores do *e-Nose*;
- *Boxplots* das leituras dos sensores do *e-Nose*;
- *Boxplots* das leituras padronizadas dos sensores do *e-Nose*;

- Matriz de correlação linear das leituras dos sensores do *e-Nose*;
- Projeção *UMAP* das leituras dos sensores do *e-Nose*.

5.3.1 Leituras dos sensores do *e-Nose*

A leitura dos sensores do nariz eletrônico mostra o perfil das espécies fúngicas por sensor e o seu estudo dá uma ideia de quão diferentes ou iguais essas leituras são. Quanto mais diferentes forem esses perfis, maior é a chance dos modelos de IA conseguirem classificá-los corretamente. As leituras dos sensores do *e-Nose* são mostradas na Figura 43.

5.3.2 *Boxplots* das leituras dos sensores do *e-Nose*

Os *boxplots* das leituras dos sensores do *e-Nose* complementam os gráficos de perfis da Figura 43 das espécies fúngicas, explicitando as diferenças de escala, medianas, quartis, amplitude e até possíveis *outleirs* das leituras que não são facilmente perceptíveis nos gráficos de perfis por sensor. Os *boxplots* das leituras dos sensores do *e-Nose* são mostrados na Figuras 44 e 45.

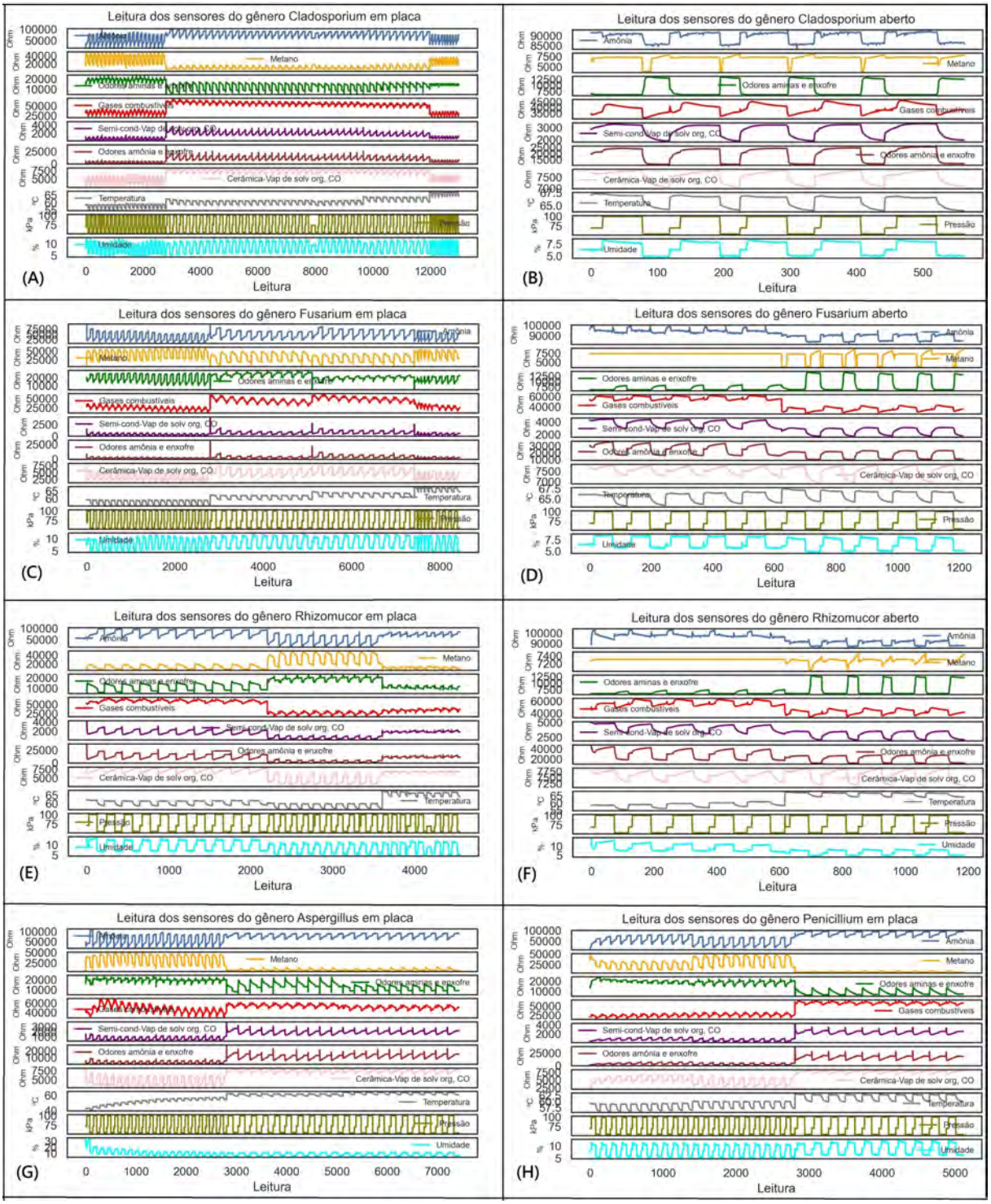
5.3.3 *Boxplots* das leituras padronizadas dos sensores do *e-Nose*

Os *boxplots* das leituras padronizadas dos sensores do *e-Nose* também complementam os gráficos de perfis da Figura 43 e os gráficos ds Figuras 44 e 45, *boxplots* das leituras dos sensores do *e-Nose* das espécies fúngicas, trazendo todas as leituras dos sensores para a mesma escala de leitura e explicitando ainda mais as diferenças de escala, medianas, quartis, amplitude e até possíveis outleirs das leituras que não são facilmente perceptíveis nos gráficos mostrados anteriormente. Os *boxplots* das leituras padronizadas dos sensores do *e-Nose* são mostrados na Figura 46.

5.3.4 Matriz de correlação linear das leituras dos sensores do *e-Nose*

A Matriz de correlação linear das leituras dos sensores do *e-Nose* por gênero fúngico indica se existe correlação linear entre os dados dos sensores. Quanto mais escuro é o vermelho, mais forte é a correlação linear direta e quanto mais escuro é o azul, mais forte é a correlação linear inversa. O ideal é não haver correlação linear entre os sensores, indicando independência entre sensores. Portanto, quanto mais claros forem os tons de vermelho e azul, próximos ao branco, isto é, ausência de correlação linear, melhor para o processo de classificação. As matrizes de correlações dos sensores do *e-Nose* estão mostradas na Figura 47.

Figura 43 – Leituras dos sensores do *e-Nose* por gênero fúngico. (A) *Cladosporium* sp. em placa, (B) *Cladosporium* sp. aberto, (C) *Fusarium* sp. em placa, (D) *Fusarium* sp. aberto, (E) *Rhizomucor* sp. em placa, (F) *Rhizomucor* sp. aberto, (G) *Aspergillus* sp. em placa, (H) *Penicillium* sp. em placa. A leitura dos sensores do nariz eletrônico mostra o perfil das espécies fúngicas por sensor e o seu estudo dá uma ideia de quão diferentes ou iguais essas leituras são. Quanto mais diferentes forem esses perfis, maior é a chance dos modelos de IA conseguirem classifica-los corretamente.

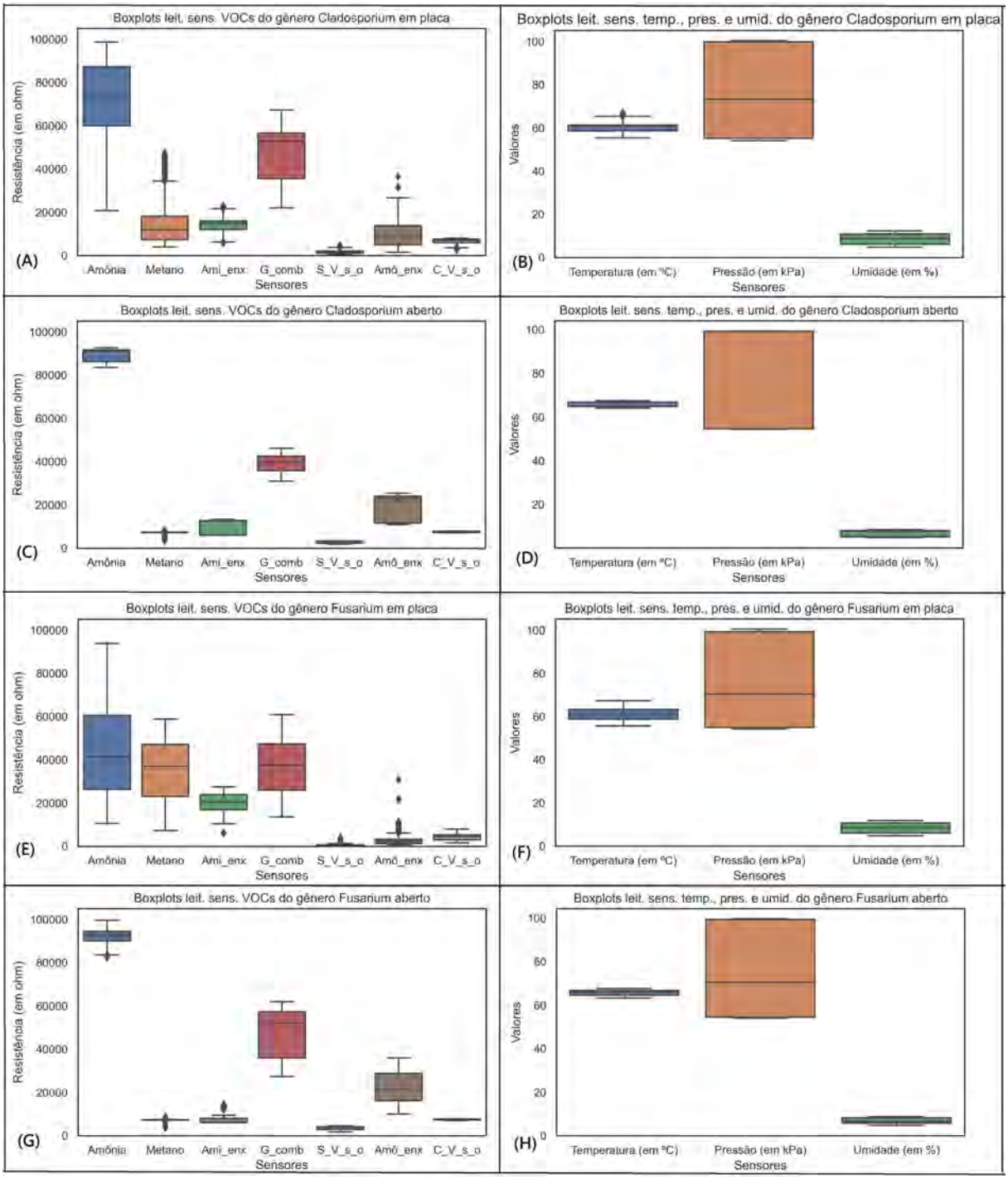


Fonte: Elaborado pelo autor.

5.3.5 Projeção *Uniform Manifold Approximation and Projection (UMAP)* das leituras dos sensores do *e-Nose*

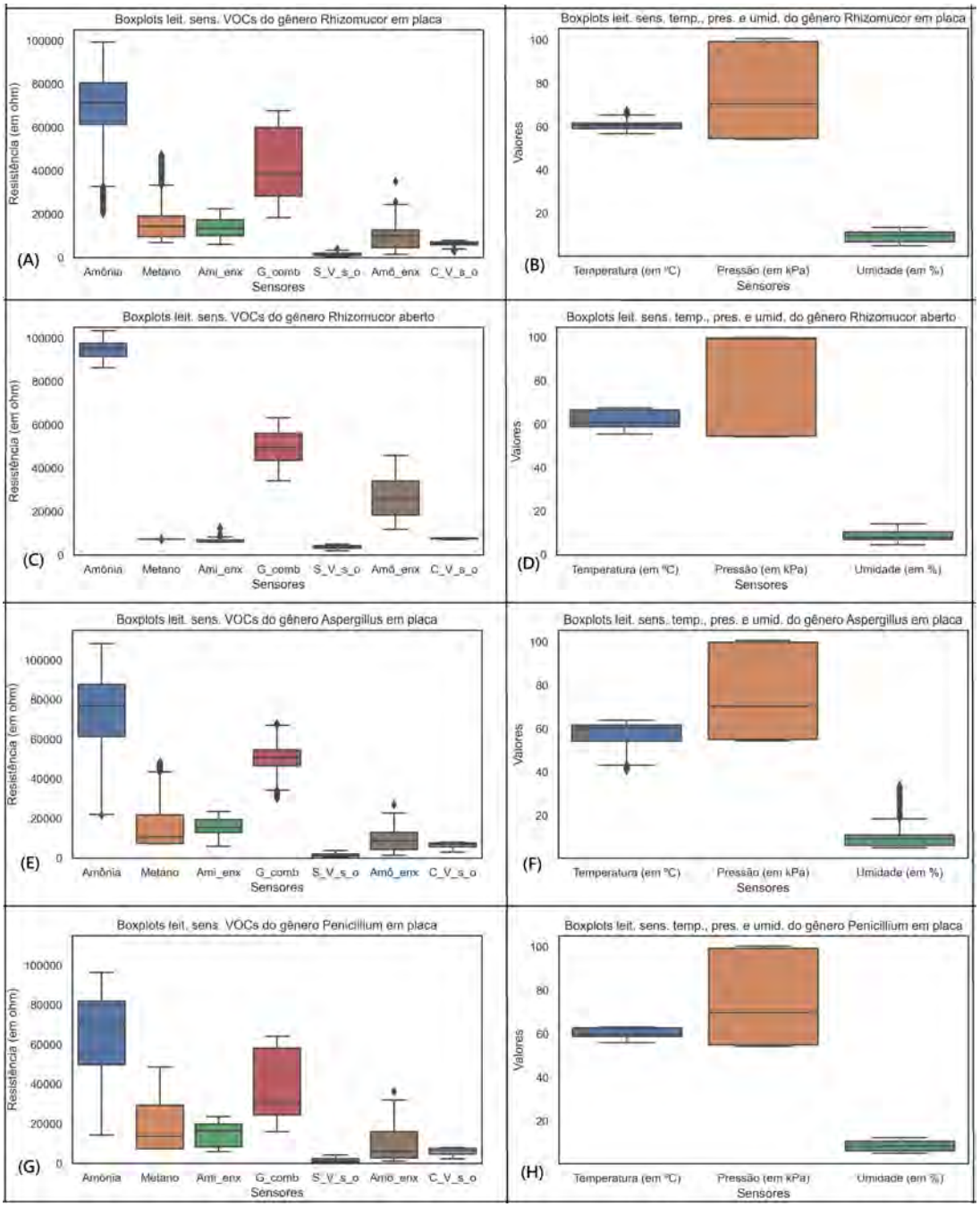
UMAP são técnicas de aproximação, redução de dimensão e projeção (visualização). Nesse trabalho foi utilizada a verificação de interdependência entre classes em conjuntos de dados grandes e complexos. A Projeção *UMAP* das leituras dos sensores do *e-Nose* por gênero fúngico dá uma indicação visual da sobreposição das classes das bases **Placa** e **Placa_Aberto** nas Figuras 48 e 49 respectivamente. Como não se observam grandes sobreposições das indicações das classes, é grande a probabilidade de bom desempenho dos modelos de classificação para essas bases. Para mais detalhes sobre as técnicas *UMAP*, consultar (MCINNES; HEALY; MELVILLE, 2018).

Figura 44 – *Boxplots* das leituras dos sensores do *e-Nose* por gênero fúngico. (A) *Boxplots* das leituras dos sensores de VOCs do *Cladosporium* sp. em placa, (B) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Cladosporium* sp. em placa, (C) *Boxplots* das leituras dos sensores de VOCs do *Cladosporium* sp. aberto, (D) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Cladosporium* sp. aberto, (E) *Boxplots* das leituras dos sensores de VOCs do *Fusarium* sp. em placa, (F) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Fusarium* sp. em placa, (G) *Boxplots* das leituras dos sensores de VOCs do *Fusarium* sp. aberto, (H) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Fusarium* sp. aberto. Os *boxplots* das leituras dos sensores do *e-Nose* complementam os gráficos de perfis da Figura 43 das espécies fúngicas explicitando as diferenças de escala, medianas, quartis, amplitude e até possíveis *outleirs* das leituras que não são facilmente perceptíveis nos gráficos de perfis por sensor.



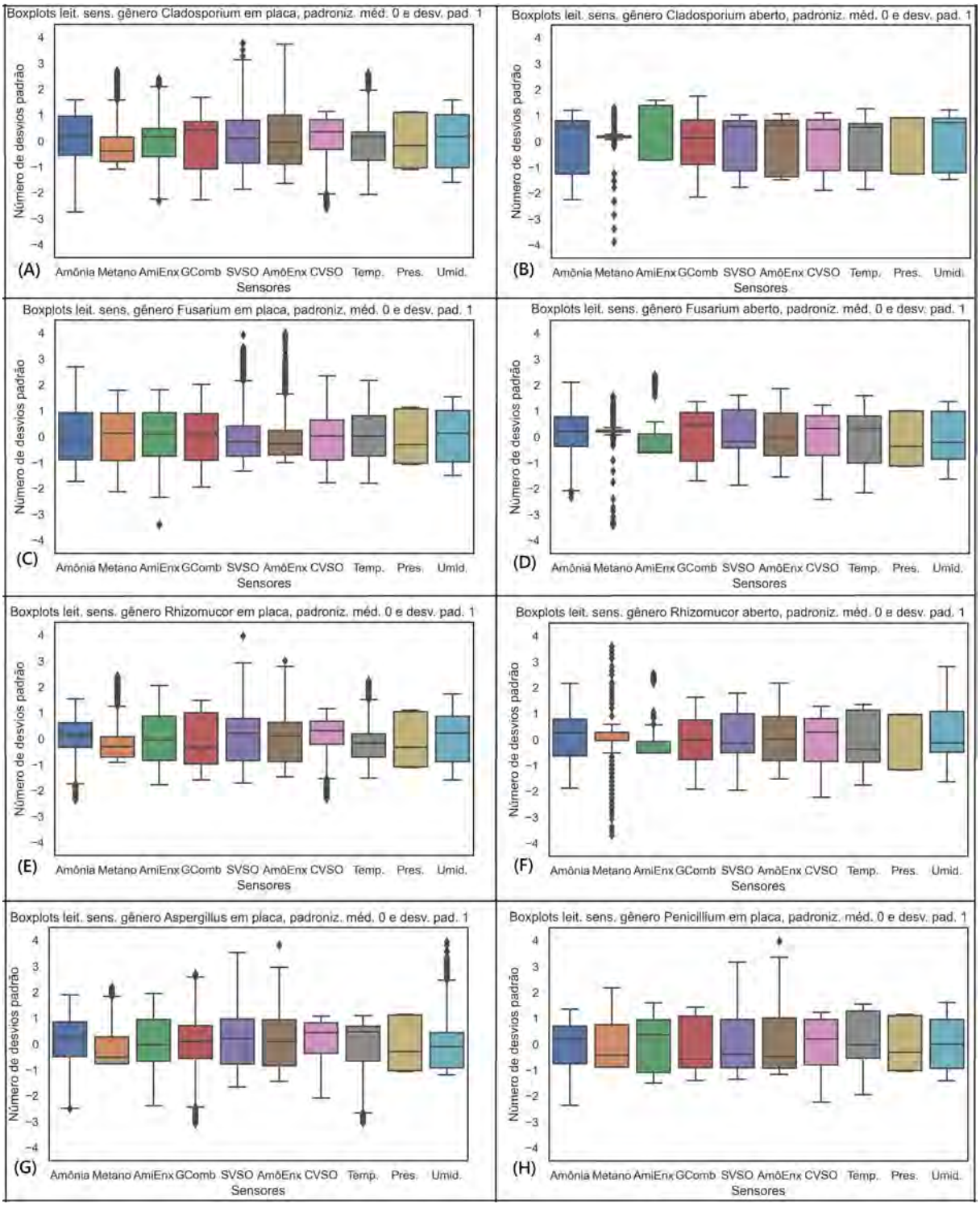
Fonte: Elaborado pelo autor.

Figura 45 – *Boxplots* das leituras dos sensores do *e-Nose* por gênero fúngico. (A) *Boxplots* das leituras dos sensores de VOCs do *Rhizomucor* sp. em placa, (B) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Rhizomucor* sp. em placa, (C) *Boxplots* das leituras dos sensores de VOCs do *Rhizomucor* sp. aberto, (D) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Rhizomucor* sp. aberto, (E) *Boxplots* das leituras dos sensores de VOCs do *Aspergillus* sp. em placa, (F) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Aspergillus* sp. em placa, (G) *Boxplots* das leituras dos sensores de VOCs do *Penicillium* sp. em placa, (H) *Boxplots* das leituras dos sensores de temperatura, pressão e umidade do *Penicillium* sp. em placa. Os *boxplots* das leituras dos sensores do *e-Nose* complementam os gráficos de perfis da Figura 43 das espécies fúngicas explicitando as diferenças de escala, medianas, quartis, amplitude e até possíveis *outleirs* das leituras que não são facilmente perceptíveis nos gráficos de perfis por sensor.



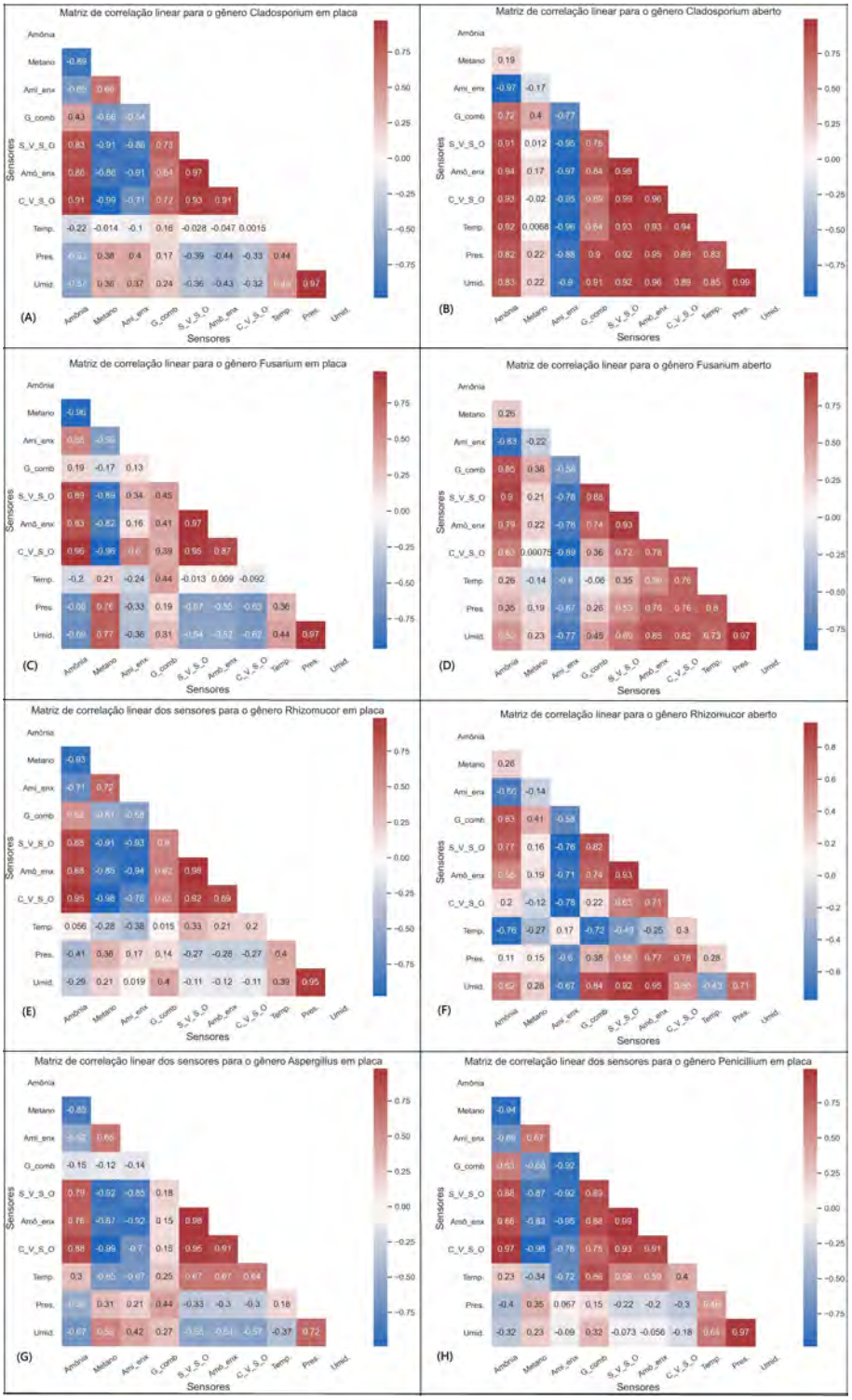
Fonte: Elaborado pelo autor.

Figura 46 – *Boxplots* das leituras padronizadas dos sensores do *e-Nose* por gênero fúngico. (A) *Cladosporium* sp. em placa, (B) *Cladosporium* sp. aberto, (C) *Fusarium* sp. em placa, (D) *Fusarium* sp. aberto, (E) *Rhizomucor* sp. em placa, (F) *Rhizomucor* sp. aberto, (G) *Aspergillus* sp. em placa, (H) *Penicillium* sp. em placa. Os *boxplots* das leituras padronizadas dos sensores do *e-Nose* também complementam os gráficos de perfis da Figura 43 e os gráficos ds Figuras 44 e 45, *boxplots* das leituras dos sensores do *e-Nose* das espécies fúngicas, trazendo todas as leituras dos sensores para a mesma escala de leitura e explicitando ainda mais as diferenças de escala, medianas, quartis, amplitude e até possíveis outleirs das leituras que não são facilmente perceptíveis nos gráficos mostrados anteriormente. Os *boxplots* das leituras padronizadas dos sensores do *e-Nose* são mostrados na Figura 46



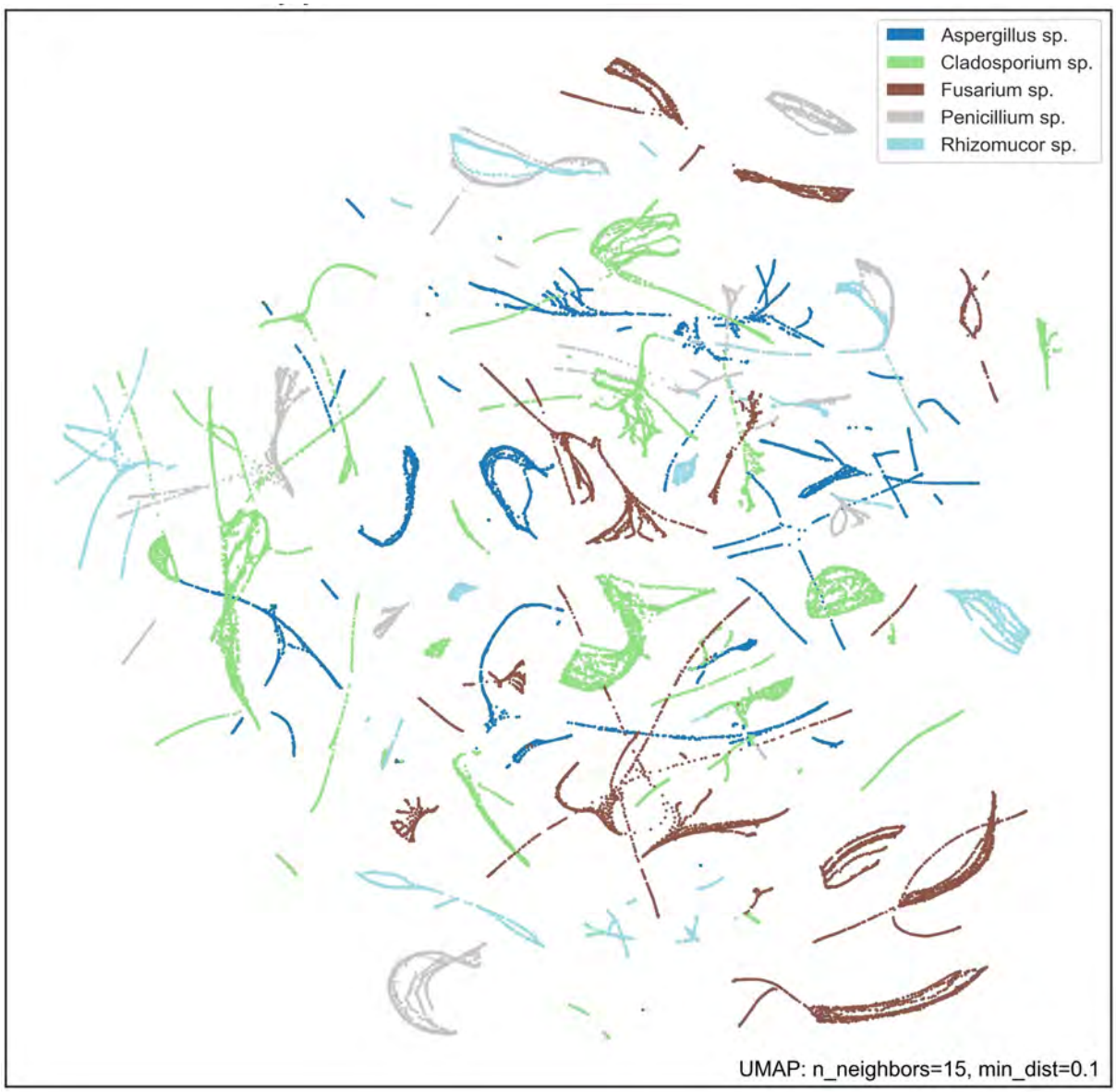
Fonte: Elaborado pelo autor.

Figura 47 – Matriz de correlação linear dos sensores do *e-Nose* por gênero fúngico. (A) *Cladosporium* sp. em placa, (B) *Cladosporium* sp. aberto, (C) *Fusarium* sp. em placa, (D) *Fusarium* sp. aberto, (E) *Rhizomucor* sp. em placa, (F) *Rhizomucor* sp. aberto, (G) *Aspergillus* sp. em placa, (H) *Penicillium* sp. em placa. O ideal é não haver correlação linear entre os sensores, indicando independência entre sensores. Portanto, quanto mais claros forem os tons de vermelho e azul, próximos ao branco, isto é, ausência de correlação linear, melhor para o processo de classificação.



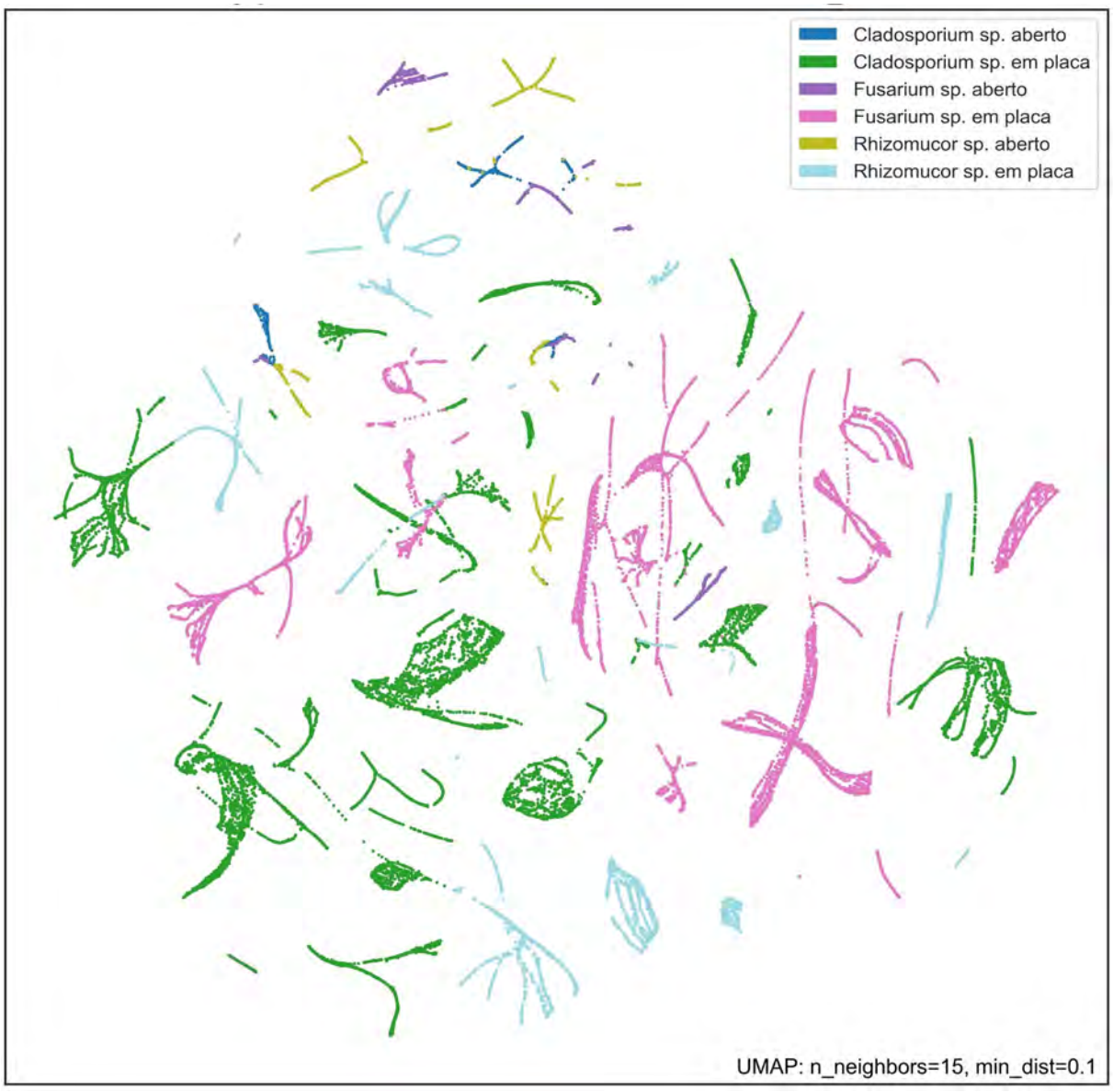
Fonte: Elaborado pelo autor.

Figura 48 – A Projeção UMAP das leituras dos sensores do *e-Nose* por classe da base **Placa**. A Projeção *UMAP* das leituras dos sensores do *e-Nose* por gênero fúngico dá uma indicação visual da sobreposição das classes. Como não se observam grandes sobreposições das indicações das classes, é grande a probabilidade de bom desempenho dos modelos de classificação para essas bases.



Fonte: Elaborado pelo autor.

Figura 49 – A Projeção *UMAP* das leituras dos sensores do *e-Nose* por classe da base **Placa_Aberto**. A Projeção *UMAP* das leituras dos sensores do *e-Nose* por gênero fúngico dá uma indicação visual da sobreposição das classes. Como não se observam grandes sobreposições das indicações das classes, é grande a probabilidade de bom desempenho dos modelos de classificação para essas bases.



Fonte: Elaborado pelo autor.

Figura 50 – Representação gráfica de uma base dividida em 70% para treino e 30% para teste pela técnica da biblioteca *Python scikit-learn, train_test_split.*

Fonte: Elaborado pelo autor.

5.4 METODOLOGIA DE PROCESSAMENTO DOS EXPERIMENTOS

Nesta seção, serão detalhados os procedimentos adotados nos experimentos de processamento dos 10 modelos de IA selecionados (conferir na Seção 4.3 Métodos de aprendizagem de máquina testados para resolver o problema) para resolver o problema de classificação (conferir na Seção 1.2 Definição dos objetivos).

Os experimentos foram processados através de códigos escritos em *Python* e o formato das bases utilizado foi **por ciclo**, porque esse formato é análogo à uma série temporal e portanto, mais adequado ao problema de estudo.

Para garantir que os resultados não são aleatórios, sem representatividade, foi adotada a validação cruzada *kFold* (abaixo será justificada a escola dessa metodologia) de cada processamento de modelo por 10 vezes nos casos das bases **Placa** e **Pl_TR_Ab_TS** e de 5 vezes no caso da base **Pl_Ab**. Essa limitação de 5 repetições na base **Pl_Ab** ocorreu em razão da existência de apenas 5 ciclos das coletas em aberto.

Para garantir que a validação cruzada kFold é a melhor solução de aleatorização dos resultados, foram realizados 3 experimentos de aleatorização de processamento dos modelos:

1. 10 Repetições dividindo a base em treino/ teste a 50%/ 50% aleatoriamente, utilizando a técnica da biblioteca *Python scikit-learn, train_test_split.*

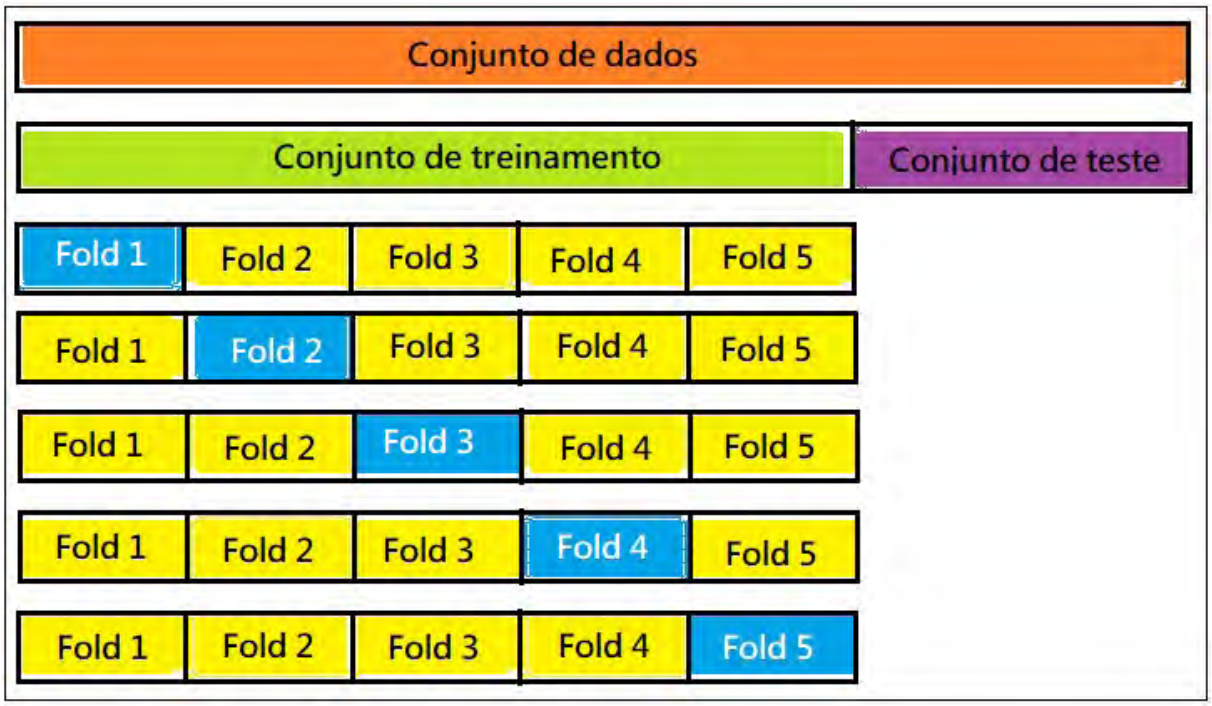
2. 10 Repetições dividindo a base em treino/ teste a 70%/ 30% aleatoriamente, utilizando a técnica da biblioteca *Python scikit-learn, train_test_split.*

3. Repetições utilizando técnica de validação cruzada, *kFold (StratifiedKFold)*, com k=10.

A técnica da biblioteca *Python scikit-learn, train_test_split*, consiste em dividir os instâncias de uma base qualquer em proporções de treino e teste. A Figura 50 ilustra a técnica *train_test_split.*

A técnica da biblioteca *Python scikit-learn*, validação cruzada *kFold* consiste em dividir uma base qualquer em k partes (*folders*) e rodar o modelo k vezes. Em cada rodada k-1 *folders* são o conjunto de treinamento e k *folder* é o conjunto de teste, até que o conjunto de teste tenha sido todos os *folders*. A Figura 51 ilustra a técnica de validação cruzada *kFold.*

Figura 51 – Representação gráfica da técnica de validação cruzada *kFold*.



Fonte: Elaborado pelo autor.

Através da aplicação de testes estatísticos de normalidade de Shapiro-Wilk e testes estatísticos de diferença de população média *One-way ANOVA* e ainda de testes estatísticos de diferença de distribuição de Kruskal-Wallis, chegou-se à conclusão de que os melhores resultados são alcançados com a técnica de validação cruzada *kFold*.

5.5 RESULTADOS ALCANÇADOS

Nesta seção serão apresentados os resultados alcançados nas três bases propostas para resolver o problema da dissertação.

5.5.1 Experimentos da base Placa

A Tabela 10 mostra os resultados das médias e desvios padrões das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento (em seg.).

Na Seção 5.2 Bases de dados propostas para solução do problema foi explicado que a base **Placa** reuniu só as leituras em placa.

Os resultados foram satisfatórios, com acurácias (avaliam os acertos de forma global) variando de 87.7% a 94,5%, com sensibilidades (que é a capacidade do modelo de identificar corretamente os casos positivos) variando de 86.0% a 93.6% e com especificidades (que é a capacidade do modelo de identificar corretamente os casos negativos) variando de 96.9% a 98,7%. Os desvios padrões foram baixos para as acurácias e sensibilidades dos

modelos e baixíssimas para as especificidades dos modelos. Esses baixos desvios padrões indicam que os resultados das métricas dos modelos foram homogêneos, sinalizando os funcionamentos adequados dos modelos, sem valores discrepantes.

Tabela 10 – Base **Placa** - Média e desvio padrão de 10 iterações da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)		Tempo proc. (seg)[1,2,3]	
	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.
MrSEQL	**94.5**	**5.2**	**93.6**	**5.9**	**98.7**	**1.3**	**256.7**	**3.1**
ROCKET	93.6	6.1	92.2	8.0	98.4	1.5	172.5	24.7
Arsenal	93.6	6.1	92.2	8.0	98.4	1.5	838.7	58.3
InceptionTime	92.7	5.7	91.0	8.2	98.1	1.5	1,577.1	9.2
HIVE-COTE V2	92.7	8.4	90.5	11.0	98.2	2.1	1,734.5	64.2
TSF	91.8	7.0	90.5	9.0	98.0	1.8	421.4	12.0
cBOSS	91.8	7.0	90.5	9.0	98.0	1.8	421.4	12.0
kNN	91.4	7.9	89.6	10.5	97.9	1.9	452.3	20.7
RISE	90.0	6.7	88.3	9.7	97.4	1.7	12.6	1.5
WEASEL	87.7	10.5	86.0	12.0	96.9	2.6	687.4	27.3

Fonte: Elaborado pelo autor.
1. Os classificadores Arsenal, cBOSS, HIVE-COTE 2.0, kNN, RISE, ROCKET, TSF e WEASEL foram rodados em notebook HP dv7-3085dx, processador Intel CORE i7, CPU 1.6Ghz, 6GB RAM.
2. O classificador MrSEQL foi rodado no Google Colab em Intel(R) Xeon(R) CPU 2.20GHz, 12GB RAM.
3. O classificador InceptionTime foi rodado no Google Colab em Tesla K80 GPU.

O classificador *Mr-SEQL* ficou em primeiro lugar nas 3 métricas, com os classificadores *ROCKET* e *Arsenal* empatados em segundo lugar.

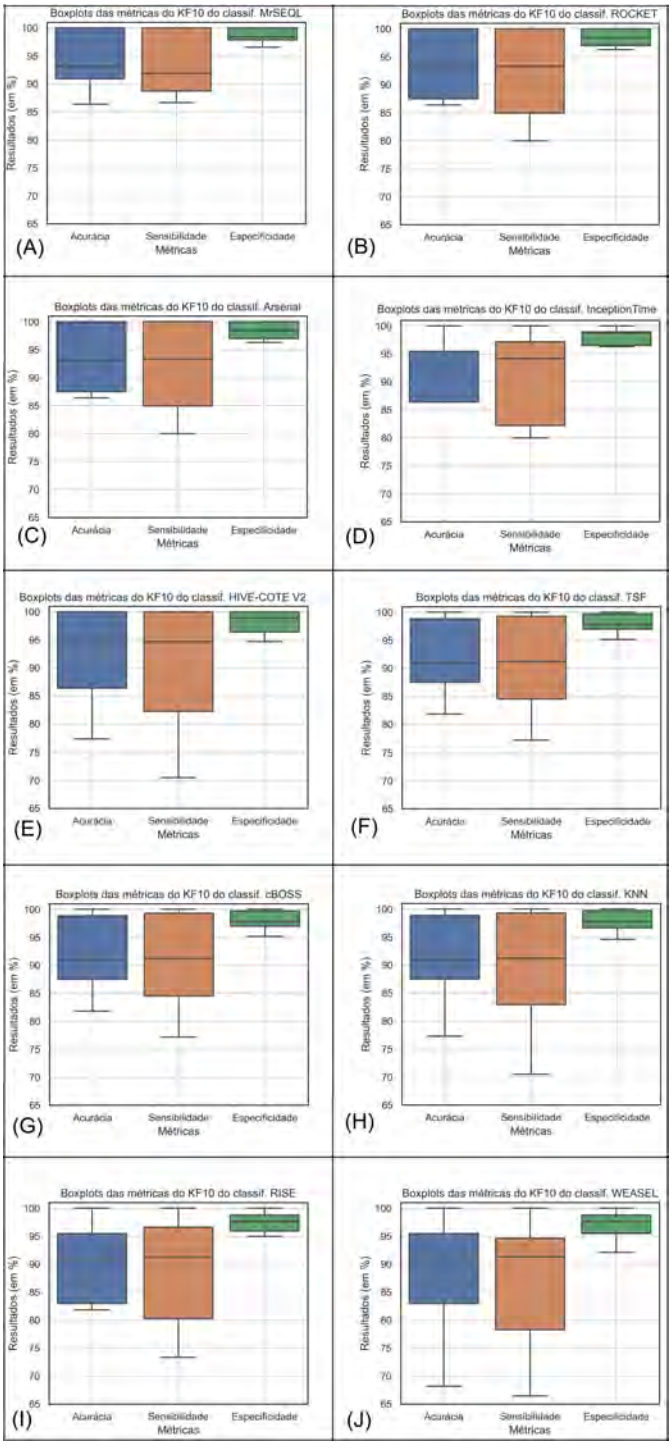
A Tabela 11 apresenta os intervalos de confiança, com nível de confiança de 95%, da acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) das 10 iterações da validação cruzada *kFold*.

Conforme mostrado na Tabela 11, os resultados das acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) entre os 10 modelos foi estatisticamente igual, variando apenas numericamente.

A Figura 52 mostra os *boxplots* das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 10 modelos de IA. Os gráficos mostram-se robustos, com quase todos os classificadores concentrando os resultados nos 3 primeiros quartis acima dos 87% para as acurácias, acima dos 82% para as sensibilidades e acima dos 95% para todas as especificidades.

Diante desse empate estatístico dos resultados, de um relativo maior custo computacional do *Mr-SEQL* (veja na observação 2 da Tabela 10 que o seu processamento é em máquina mais potente) e do estudo das matrizes de confusão mostradas na Figura 53 que

Figura 52 – Base **Placa** - *Boxplots* das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 10 modelos de IA. (A) Classificador *Mr-SEQL*, (B) Classificador *ROCKET*, (C) Classificador *Arsenal*, (D) Classificador *InceptionTime*, (E) Classificador *HIVE-COTE V2*, (F) Classificador *TSF*, (G) Classificador *cBOSS*, (H) Classificador *kNN*, (I) Classificador *RISE*, (J) Classificador *WEASEL*. Os gráficos mostram-se robustos, com quase todos os classificadores concentrando os resultados nos 3 primeiros quartis acima dos 87% para as acurácias, acima dos 82% para as sensibilidades e acima dos 95% para todas as especificidades. Os desvios padrões foram baixos para as acurácias e sensibilidades dos modelos e baixíssimas para as especificidades dos modelos. Esses baixos desvios padrões indicam que os resultados das métricas dos modelos foram homogêneos, sinalizando o funcionamento adequado dos modelos, sem valores discrepantes.



Fonte: Elaborado pelo autor.

Tabela 11 – Base **Placa** - Intervalos de confiança, com nível de confiança de 95%, da acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) de 10 iterações da validação cruzada *kFold*. Os resultados das acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) entre os 10 modelos foi estatisticamente igual, variando apenas numericamente.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)	
	Inferior	Superior	Inferior	Superior	Inferior	Superior
MrSEQL	91.5	97.5	90.3	96.8	97.1	100.0
ROCKET	90.4	96.9	88.7	95.8	96.8	100.0
Arsenal	90.4	96.9	88.7	95.8	96.8	100.0
InceptionTime	89.3	96.2	87.2	94.8	96.3	99.9
HIVE-COTE V2	89.3	96.2	86.7	94.4	96.4	99.9
TSF	88.2	95.4	86.6	94.4	96.1	99.8
cBOSS	88.2	95.4	86.6	94.4	96.1	99.8
kNN	87.7	95.1	85.5	93.6	96.0	99.8
RISE	86.0	94.0	84.0	92.5	95.3	99.5
WEASEL	83.4	92.1	81.4	90.6	94.6	99.2

Fonte: Elaborado pelo autor.

apresentam apenas 1 ou 2 erros que podem ser atribuídos às semelhanças sutis dos perfis dos *VOC*s dos sensores, cabe um olhar cuidadoso em relação à possibilidade de adotar como solução o **classificador *ROCKET* como a escolha inteligente** porque há uma pequena perda de desempenho nas métricas, mas em compensação há um bom ganho no tempo de execução.

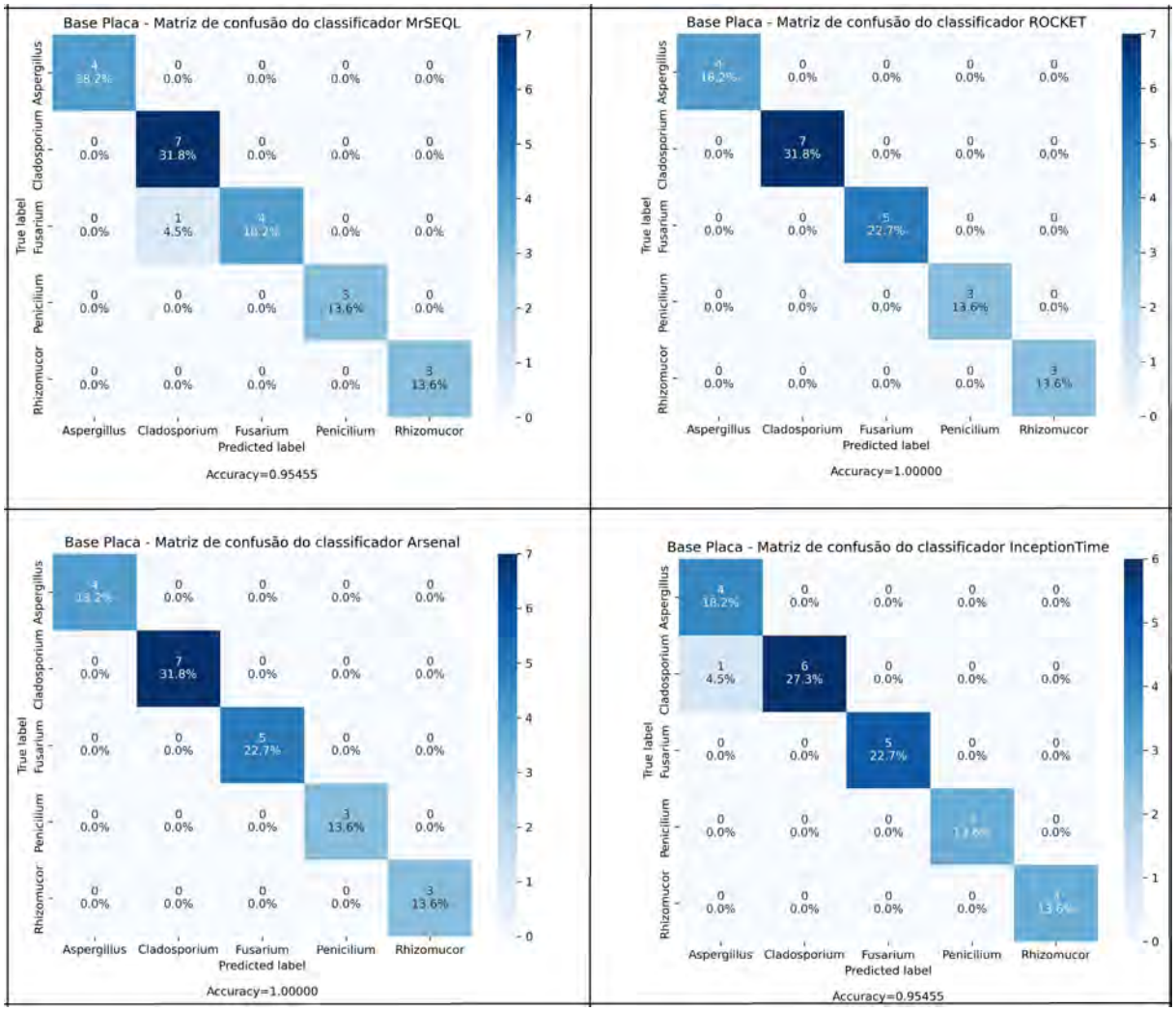
5.5.2 Experimentos da base Placa_TR_Aberto_TS

A Figura 12 mostra os resultados das médias e desvios padrões das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento (em seg.).

Na Seção 5.2, Bases de dados propostas para solução do problema, foi explicado que para formar a base **Placa_TR_Aberto_TS** foram usadas as leituras em placa como treinamento e as leituras abertas como teste.

Os resultados foram insatisfatórios, com acurácias (avaliam os acertos de forma global) variando de 36.4% a 58.8%, com sensibilidades (que é a capacidade do modelo de identificar corretamente os casos positivos) variando de 37.3% a 55.7% e com especificidades (que é a capacidade do modelo de identificar corretamente os casos negativos) variando de 69,0% a 77,6%. De maneira geral, os desvios padrões foram baixos para as acurácias e sensibilidades dos modelos e baixíssimas para as especificidades dos modelos. Contudo, alguns boxplots apresentam valores *outlaiers*. Esses baixos desvios padrões indicam que os resultados das métricas dos modelos foram homogêneos, sinalizando o funcionamento adequado dos modelos enquanto que os valores *outlaiers* indicam valores discrepantes.

Figura 53 – Base **Placa** - Matriz de confusão por classificador. (A) Classificador *Mr-SEQL*, (B) Classificador *ROCKET*, (C) Classificador *Arsenal*, (D) Classificador *InceptionTime*. As matrizes de confusão que apresentam apenas 1 ou 2 erros que podem ser atribuídos às semelhanças sutis dos perfis dos *VOC*s dos sensores.



Fonte: Elaborado pelo autor.
Obs: Classes: 0 - Aspergillus sp., 1 - Cladosporium sp., 2 - Fusarium sp., 3 - Penicillium sp., 4 - Rhizomucor sp.

O exame cuidadoso dos gráficos do estudo dos dados, na Seção 5.3, Exploração dos dados, mais espeficicamente nas subsecções 5.3.1, Leituras dos sensores do *e-Nose*, 5.3.2, *Boxplots* das leituras dos sensores do *e-Nose*, 5.3.3, *Boxplots* das leituras padronizadas dos sensores do *e-Nose* e 5.3.4 matriz de correlação lineardos sensores do *e-Nose*, já indicavam as incompatibilidades das classes das espécies fúngicas em placa e aberto.

Para corroborar as ponderações acima, o que se discutiu no Capítulo 1 Introdução, por (MAGESTE et al., 2012), indica que o comportamento dos fungos anemófilos é influenciado pelo ambiente (poluído, aberto, em mata, fechado, estéril, etc.), temperatura, umidade, pH, entre outros. Diante disso, acredita-se estar justificado que os *VOC*s emitidos pelas colônias de anemófilos em placa sejam diferentes dos *VOC*s emitidos por colônias de anemófilos para o ambiente aberto.

Tabela 12 – Base **Placa_TR_Aberto_TS** - Média e desvio padrão de 10 iterações da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)		Tempo proc. (seg)[1,2,3]	
	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.
Arsenal	**58.8**	**5.3**	**55.7**	**4.5**	**77.6**	**2.6**	**598.8**	**43.5**
ROCKET	52.8	6.5	50.7	5.4	74.5	3.2	117.8	10.3
TSF	47.2	4.1	43.7	4.8	75.8	1.9	1,433.5	44.2
HIVE-COTE V2	42.4	4.3	47.0	5.1	71.4	2.5	1,262.9	51.5
InceptionTime	42.0	7.4	41.0	7.0	68.3	3.4	1,087.9	22.8
cBOSS	41.2	4.2	40.7	5.4	69.8	2.0	285.8	26.3
MrSEQL	40.0	0.0	33.0	0.0	66.7	0.0	110.3	1.5
RISE	37.2	5.7	39.7	9.9	69.8	3.2	8.2	0.9
WEASEL	36.8	9.6	38.0	7.4	69.6	3.2	326.5	23.2
kNN	36.4	2.3	37.3	2.6	69.0	1.1	390.0	23.5

Fonte: Elaborado pelo autor.
1. Os classificadores Arsenal, cBOSS, HIVE-COTE 2.0, kNN, RISE, ROCKET, TSF e WEASEL foram rodados em notebook HP dv7-3085dx, processador Intel CORE i7, CPU 1.6Ghz, 6GB RAM.
2. O classificador MrSEQL foi rodado no Google Colab em Intel(R) Xeon(R) CPU 2.20GHz, 12GB RAM.
3. O classificador InceptionTime foi rodado no Google Colab em Tesla K80 GPU.

A Figura 54 mostra os *boxplots* das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 10 modelos de IA. Com resultados tão baixos, nem cabem comentários, apenas apresentam-se os gráficos como ilustração.

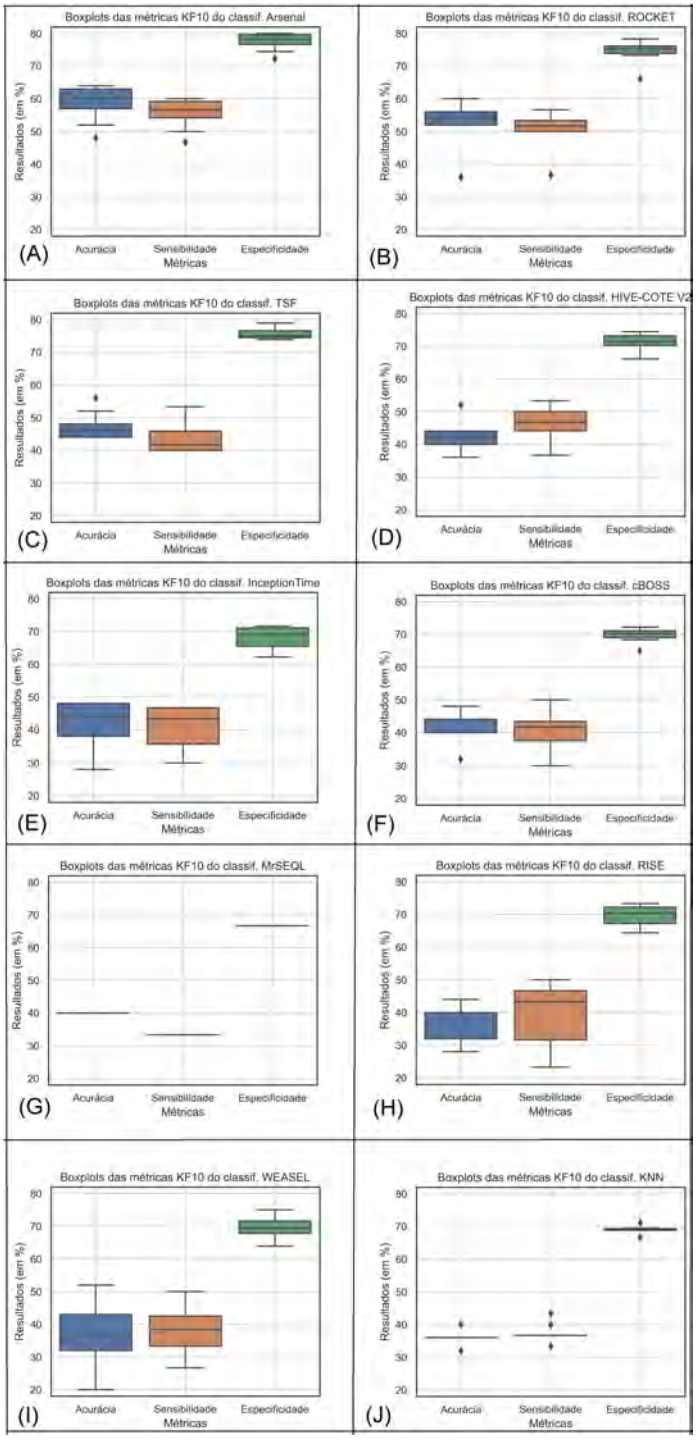
5.5.3 Experimentos da base Placa_Aberto

A Figura 14 mostra os resultados das médias e desvios padrões das 5 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento (em seg.).

Na Seção 5.2, Bases de dados propostas para solução do problema foi explando que, na formação da base **Placa_Aberto**, foram usadas as leituras em placa e aberto juntas.

Os resultados foram adequados, com acurácias (avaliam os acertos de forma global) variando de 82.9% a 94,9%, com sensibilidades (que é a capacidade do modelo de identificar corretamente os casos positivos) variando de 70,2% a 87,2% e com especificidades (que é a capacidade do modelo de identificar corretamente os casos negativos) variando de 96,4% a 99,0%. Os desvios padrões foram baixos para as acurácias, moderados para as sensibilidades e baixíssimos para as especificidades dos modelos. Esses baixos desvios padrões indicam que os resultados das métricas dos modelos foram homogêneos, sinalizando

Figura 54 – Base **Placa_TR_Aberto_TS** - *Boxplots* das 10 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 10 modelos de IA. (A) Classificador *Arsenal*, (B) Classificador *ROCKET*, (C) Classificador *TSF*, (D) Classificador *HIVE-COTE V2*, (E) Classificador *InceptionTime*, (F) Classificador *cBOSS*, (G) Classificador *Mr-SEQL*, (H) Classificador *RISE*, (I) Classificador *WEASEL*, (J) Classificador *kNN*. De maneira geral, os desvios padrões foram baixos para as acurácias e sensibilidades dos modelos e baixíssimas para as especificidades dos modelos. Contudo, alguns boxplots apresentam valores *outlaiers*. Esses baixos desvios padrões indicam que os resultados das métricas dos modelos foram homogêneos, sinalizando o funcionamento adequado dos modelos enquanto que os valores *outlaiers* indicam valores discrepantes.



Fonte: Elaborado pelo autor.

Tabela 13 – Base **Placa_TR_Aberto_TS** - Intervalos de confiança, con nível de confiança de 95%, da acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) de 10 iterações da validação cruzada *kFold*.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)	
	Inferior	Superior	Inferior	Superior	Inferior	Superior
Arsenal	38.5	79.1	35.2	76.2	60.4	94.8
ROCKET	32.2	73.4	30.0	71.3	56.5	92.5
TSF	26.6	67.8	23.2	64.1	58.1	93.5
HIVE-COTE V2	22.0	62.8	26.4	67.6	52.7	90.0
InceptionTime	21.6	62.4	20.7	61.3	49.1	87.5
cBOSS	20.9	61.5	20.4	50.9	50.8	88.7
MrSEQL	19.8	60.2	13.9	52.8	47.2	86.1
RISE	17.2	57.2	19.5	59.9	50.8	88.7
WEASEL	16.9	56.7	18.0	58.0	50.6	88.6
kNN	16.5	57.3	17.4	57.3	49.9	88.1

Fonte: Elaborado pelo autor.

o funcionamento adequado dos modelos, sem valores discrepantes.

O modelo *Arsenal* ficou primeiro lugar, com *ROCKET* em segundo e *kNN* em terceiro.

A Tabela 15 apresenta os intervalos de confiança, com nível de confiança de 95%, da acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) das 5 iterações da validação cruzada *kFold*.

Conforme mostrado na Tabela 15, os resultados das acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) entre os 9 primeiros modelos foi estatisticamente igual, variando apenas numericamente.

A Figura 55 mostra os *boxplots* das 5 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 5 modelos de IA. Os gráficos mostram-se robustos, com os dois modelos mais bem colocados concentrando os resultados nos 3 primeiros quartis acima dos 95% para as acurácias, acima dos 82% para as sensibilidades e acima dos 99% para as especificidades. Os demais modelos concentram os resultados nos 3 primeiros quartis acima dos 90% para as acurácias, acima dos 70% para as sensibilidades e acima dos 96% para as especificidades.

Diante desse empate estatístico dos resultados, do maior custo computacional do *Arsenal* e do estudo das matrizes de confusão mostradas na Figura 56, que apresentam apenas 1 ou 2 erros que podem ser atribuídos às sutis semelhanças dos perfis dos *VOC*s captados dos sensores, cabe um olhar cuidadoso e a possibilidade de adotar como solução o **classificador *ROCKET* como a escolha inteligente**, porque há uma pequena perda de desempenho nas métricas, mas em compensação há um grande ganho no tempo de execução.

Tabela 14 – Base **Placa_Aberto** - Média e desvio padrão de 5 iterações da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %), e do tempo de processamento.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)		Tempo proc. (seg)[1,2,3]	
	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.	Média	Desv. pad.
Arsenal	**94.9**	**5.5**	**87.2**	**11.3**	**99.0**	**1.2**	**635.5**	**43.1**
ROCKET	94.3	5.3	87.0	11.1	98.9	1.2	123.6	5.9
kNN	93.7	3.7	87.5	11.7	98.7	0.8	489.3	29.0
HIVE-COTE V2	93.7	6.5	85.0	13.2	98.7	1.5	1,298.5	6.4
MrSEQL	92.0	5.1	81.6	15.5	98.4	1.0	176.0	1.4
TSF	92.0	4.2	79.6	14.2	98.3	0.9	285.5	9.2
cBOSS	92.0	4.2	79.6	14.2	98.3	0.9	285.5	9.2
InceptionTime	90.3	6.3	80.5	10.9	98.0	1.4	1,139.0	24.2
RISE	86.9	5.6	74.3	6.4	97.2	1.3	8.8	0.8
WEASEL	82.9	7.3	70.2	13.5	96.4	1.7	577.7	105.6

Fonte: Elaborado pelo autor.

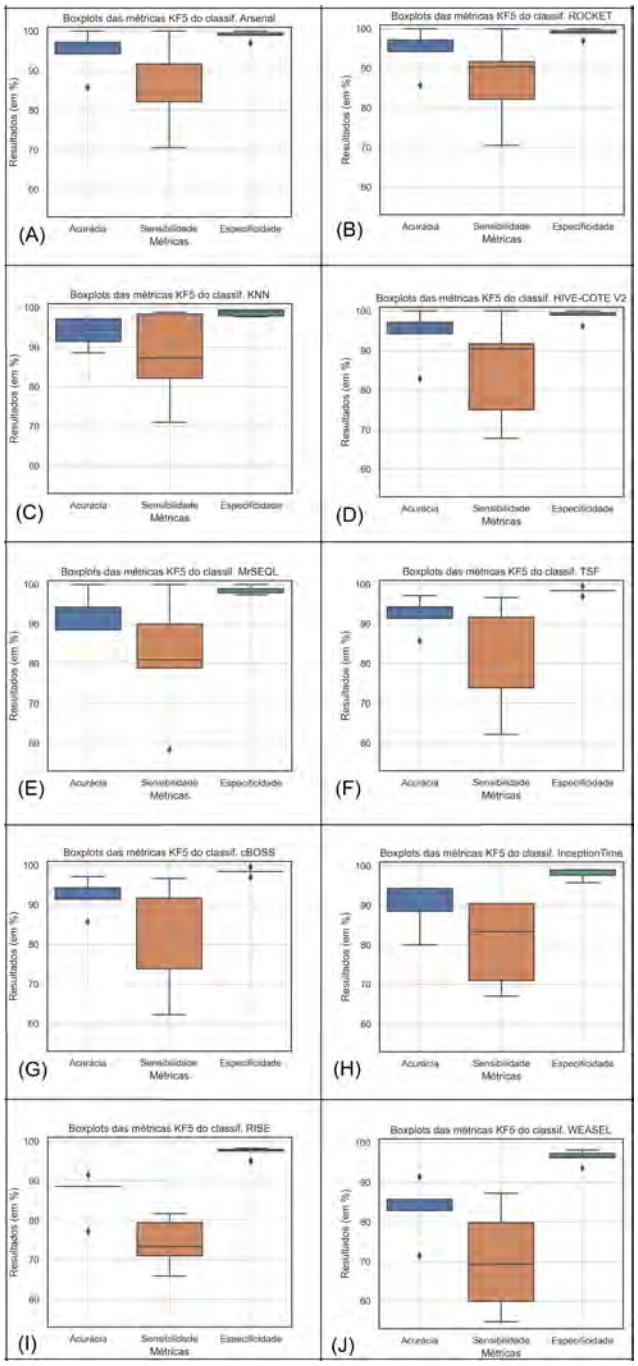
1. Os classificadores Arsenal, cBOSS, HIVE-COTE 2.0, kNN, RISE, ROCKET, TSF e WEASEL foram rodados em notebook HP dv7-3085dx, processador Intel CORE i7, CPU 1.6Ghz, 6GB RAM.
2. O classificador MrSEQL foi rodado no Google Colab em Intel(R) Xeon(R) CPU 2.20GHz, 12GB RAM.
3. O classificador InceptionTime foi rodado no Google Colab em Tesla K80 GPU.

Tabela 15 – Base **Placa_Aberto** - Intervalos de confiança, con nível de confiança de 95%, da acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) de 5 iterações da validação cruzada *kFold*. Os resultados das acurácia média (em %), sensibilidade média (em %) e especificidade média (em %) entre os 9 primeiros modelos foi estatisticamente igual, variando apenas numericamente.

Classificador	Acurácia (%)		Sensibilidade (%)		Especificidade (%)	
	Inferior	Superior	Inferior	Superior	Inferior	Superior
Arsenal	91.6	98.1	82.3	92.2	97.5	100.0
ROCKET	90.8	97.7	82.0	92.0	97.3	100.0
kNN	90.1	97.3	82.6	92.4	87.0	100.0
HIVE-COTE V2	90.1	97.3	79.7	90.3	97.0	100.0
MrSEQL	88.0	96.0	75.9	87.4	96.6	100.0
TSF	88.0	96.0	73.7	85.6	96.4	100.0
cBOSS	88.0	96.0	73.7	85.6	96.4	100.0
InceptionTime	85.9	94.7	78.6	86.3	96.0	100.0
RISE	81.9	91.9	67.8	80.7	94.8	99.7
WEASEL	77.3	88.4	63.4	77.0	93.6	99.1

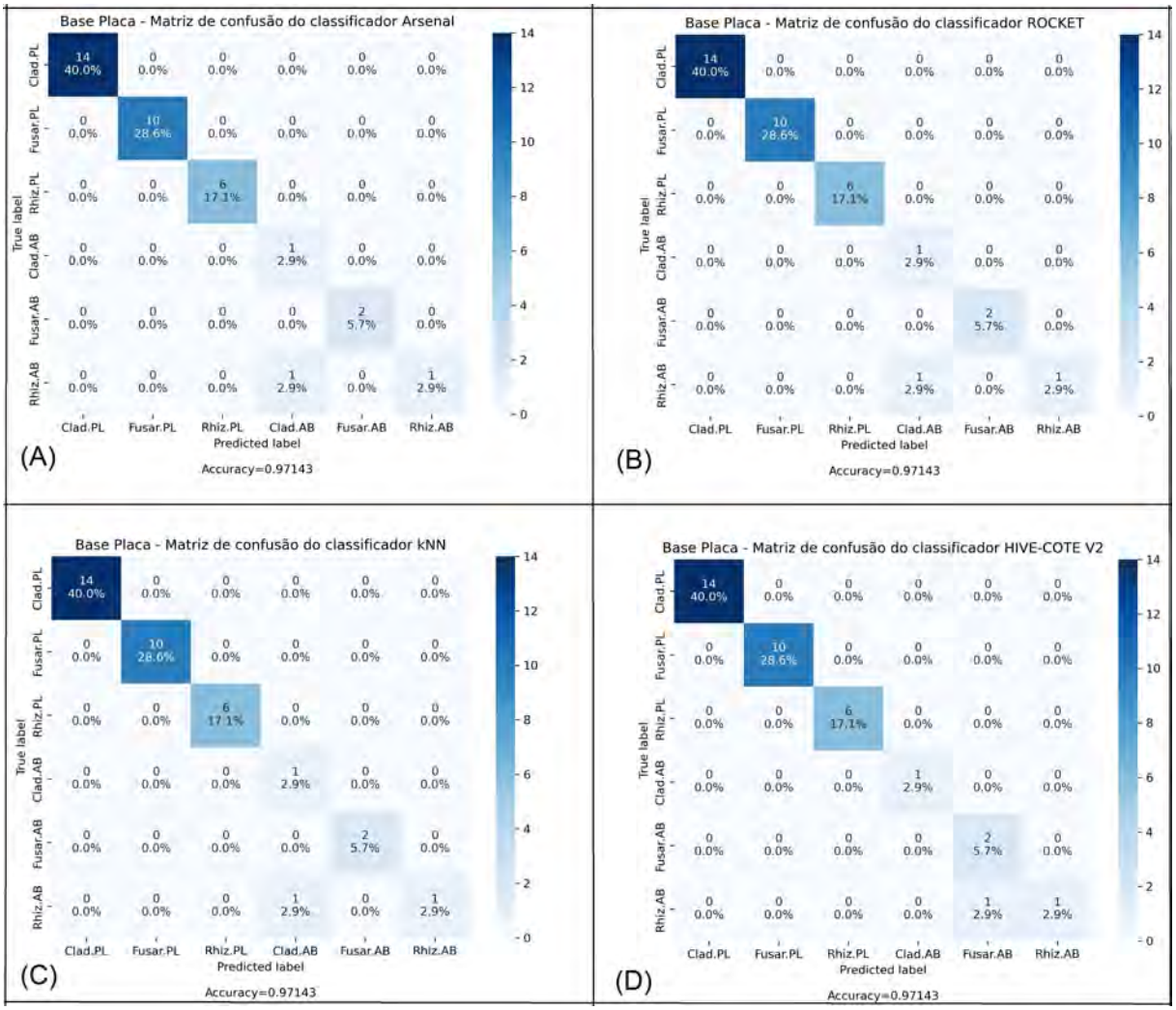
Fonte: Elaborado pelo autor.

Figura 55 – Base **Placa_Aberto** - *Boxplots* das 5 repetições da validação cruzada *kFold* das métricas acurácia (em %), sensibilidade (em %) e especificidade (em %) dos 10 modelos de IA. (A) Classificador *TSF*, (B) Classificador *RISE*, (C) Classificador *kNN*, (D) Classificador *cBOSS*, (E) Classificador *WEASEL*, (F) Classificador *Mr-SEQL*, (G) Classificador *ROCKET*, (H) Classificador *Arsenal*, (I) Classificador *HIVE-COTE V2*, (J) Classificador *InceptionTime*. Os gráficos mostram-se robustos, com os dois modelos mais bem colocados concentrando os resultados nos 3 primeiros quartis acima dos 95% para as acurácias, acima dos 82% para as sensibilidades e acima dos 99% para as especificidades. Os demais modelos concentram os resultados nos 3 primeiros quartis acima dos 90% para as acurácias, acima dos 70% para as sensibilidades e acima dos 96% para as especificidades. Os desvios padrões foram baixos para as acurácias, moderados para as sensibilidades e baixíssimos para as especificidades dos modelos.



Fonte: Elaborado pelo autor.

Figura 56 – Base **Placa_Aberto** - Matriz de confusão por classificador. (A) Classificador *Arsenal*, (B) Classificador *ROCKET*, (C) Classificador *kNN*, (D) Classificador *HIVE-COTE V2*. As matrizes de confusão que apresentam apenas 1 ou 2 erros que podem ser atribuídos às semelhanças sutis dos perfis dos *VOC*s dos sensores.



Fonte: Elaborado pelo autor.

Obs: Classes: 0 - Cladosporium sp. em placa, 1 - Fusarium sp. em placa, 2 - Rhizomucor sp. em placa, 3 - Cladosporium sp. aberto, 4 - Fusarium sp. aberto, 5 - Rhizomucor sp. aberto

6 CONCLUSÕES E TRABALHOS FUTUROS

6.1 CONCLUSÕES

Para alcançar o objetivo da dissertação e os objetivos específicos, foram realizados estudos sobre os fungos anemófilos, passando pela definição e importância destes fundos anemófilos, metodologias de estudo, principais meios de cultura utilizados, propagação e preservação de culturas, textura, morfologia e topologia de culturas, comparação entre principais métodos comerciais de identificação de microrganismos, origem das colônias utilizadas no estudo, estudo dos gêneros utilizados no trabalho, assinatura de odor dos fungos utilizados no trabalho e um breve estudo sobre séries temporais.

No Capítulo 3, Revição da Literatura, são tratadas duas áreas de interesse desse trabalho: a detecção e identificaçao de fungos e métodos de inteligência artificial para classificação de séries temporais.

Na Seção 3.1 Trabalhos Relacionados em detecção e identificação de fungos, são listados os artigos que apresentam narizes eletrônicos excutando, com resultados satisfatórios, a tarefa de indentificação de fungos. Alguns desses artigos mostram a evolução das técnicas e os prós e os contras dessas técnicas.

Conforme já discutido repetidas vezes nesse trabalho, os *VOC*s emitidos pelas colônias de fungos, utilizados para identificá-los, são armazenados na forma de séries temoprais, portanto o trabalho dessa dissertação é classificar corretamente essas séries temporais. A Seção 3.2, Trabalhos Relacionados em classificação de séries temporais, apresenta artigos com os princínpios de funcionamento dos principais modelos de inteligência artificial de classificação de séries temporais, dentre eles o "estado da arte"desses classificadores.

Em seguida, apresentou-se o nariz eletrônico utilizado nos experimentos. Foram debatidos os motivos que determinaram as escolhas dos modelos de inteligência artificial testados para resolver o problema da dissertação. Apresentaram-se os princípios de funcionamento de cada um dos 10 modelos de aprendizagem de máquina que foram testados para resolver o problema da dissertação. Também se discutiu sobre as métricas comumente utilizadas para esse tipo de problema.

No Capítulo 5, Resultados experimentais, foram apresentados os dados levantados durante a pesquisa e as bases de dados propostas para solucionar o problema. Apresentou-se também a exploração dos dados realizada para estudar e conhecer os dados. Abaixo são relatados os resultados das 3 bases que foram utilizadas para resolver o problema.

Inicialmente foram observados resultados satisfatórios da base **Placa_Aberto** que reúne as leituras em placa e aberto juntas. Com a métrica acurácia (variando de 82.9% a 94,9%), a sensibilidade (variando de 70,2% a 87,2%) e a especificidade (variando de 96,4% a 99,0%). O classificador *Arsenal* ficou primeiro lugar, com o *ROCKET* em segundo e

o *kNN* em terceiro. Foi visto também que as médias das 5 repetições das acurácias são estatisticamente iguais, o que levou a eleger, como **escolha inteligente o classificador *ROCKET*** que obteve uma acurácia média de apenas 0.6 ponto percentual menor do que o primeiro lugar, o classificador *Arsenal*, porém com uma diferença no tempo de processamento médio de 635.5 segundos do classificador *Arsenal* para um tempo médio de 123.6 segundos do classificador *ROCKET*.

A segunda base estudada foi a **Placa** que reúne apenas as leituras em placa. Os resultados também foram positivos, a métrica acurácia (variando de 87.7% a 94,5%), a sensibilidade (variando de 86.0% a 93.6%) e a especificidade (variando de 96.9% a 98,7%), com classificador *Mr-SEQL* em primeiro lugar nas 3 métricas, com os classificadores *ROCKET* e *Arsenal* empatados em segundo lugar. Também foi explanado que as médias das 10 repetições das acurácias são estatisticamente iguais, o que levou a eleger, como **escolha inteligente o classificador *ROCKET*** que obteve uma acurácia média de apenas 0.9 ponto percentual menor do que o primeiro lugar, o classificador *Mr-SEQL*, porém com uma diferença no tempo médio de processamento de 256.7 segundos (veja na observação 2 da Figura 10 para um tempo médio de processamento de 172.5 segundos do classificador *ROCKET*.

Finalmente foi estudada a base **Placa_TR_Aberto_TS** que reúne as leituras em placa como treinamento e as leituras abertas como teste. Os resultados foram insatisfatórios, a métrica acurácia (variando de 36.4% a 58.8%), a sensibilidade (variando de 37.3% a 55.7%) e a especificidade (variando de 69,0% a 77,6%). Na Seção 5.3, Exploração dos dados, mais espeficicamente nas subsecções 5.3.1, leituras dos sensores do *e-Nose*, 5.3.2, *Boxplots* das leituras dos sensores do *e-Nose*, 5.3.3, *Boxplots* das leituras padronizadas dos sensores do *e-Nose* e 5.3.4, matriz de correlação linear dos sensores do *e-Nose*, demonstraram as boas indicações do fracasso dessas classificações, e a discussão em 1, Introdução, por (MAGESTE et al., 2012), a respeito do comportamento dos fungos anemófilos que é influenciado pelo ambiente (poluído, aberto, em mata, fechado, estéril, etc.), temperatura, umidade, pH, entre outros, também vai ao encontro da justificativa de que os *VOC*s emitidos pelas colônias de anemófilos em placa sejam diferentes dos *VOC*s emitidos por colônias de anemófilos para o ambiente aberto.

Estando em princípio de pesquisa, um ou outro resultado adverso é natural. Mesmo assim, acredita-se que o objetivo desta dissertação foi plenamente resolvido pelo sucesso dos dois outros resultados.

Acredita-se, também, que os objetivos específicos foram atingidos. Realizaram-se dezenas de leituras de culturas de fungos anemófilos em placas de *Petri* com *e-Nose*, realizou-se também um pouco mais de uma dezena de leituras do ar do ambiente com a dispersão de *VOC*s e esporos de fungos anemófilos com *e-Nose*, sem problemas reportados e, finalmente, foram realizadas dezenas de experimentos com métodos de aprendizagem de máquina para identificação dos fungos anemófilos analisados pelo *e-Nose* em placas de

Petri e do ar do ambiente com a dispersão de *VOC*s e esporos de fungos anemófilos para classificação destes fungos. Os mais relevantes foram registrados nesse trabalho.

6.2 TRABALHOS FUTUROS

Como trabalhos futuros relevantes, podem-se citar:

- Ampliar e diversificar a testagem de fungos anemófilos em placa, como também a testagem do ar de ambientes com a dispersão de *VOC*s e esporos desses fungos anemófilos, objetivando cobrir uma gama maior de problemas e patologias associadas a mais espécies de fungos anemófilos, como também a ampliação das bases de dados, isto é, mais e melhores dados para minimizar os erros, implica em modelos mais bem treinados, que generalizam adequadamente para casos novos, que não são conhecidos pelos modelos.
- Aprofundar o estudo dos modelos de classificadores especializados em séries temporais. O objetivo é encontrar modelos com resultados mais satisfatórios com menores custos computacionais.
- Estudar o impacto de partes das séries temporais lidas pelo nariz eletrônico nos classificadores, isto é, testar modelos e possibilidades de classificação a partir de diferentes momentos da série temporal de um ciclo de leitura. Exemplos seriam as partes vistas na Figura 28, da Seção 4.1, nariz eletrônico e na Figura 42, da Seção 5.1, dados levantados durante a pesquisa, aspiração para a câmara de sensores, interação com os sensores e purga da câmara de sensores.
- Construir um produto, um aparelho, de baixo custo para medir a presença de fungos anemófilos em ambientes. Hoje, esse trabalho requer um processo oneroso, demorado e necessita da ajuda de laboratório especializado.

REFERÊNCIAS

ARROYO, P.; MELÉNDEZ, F.; SUÁREZ, J. I.; HERRERO, J. L.; RODRÍGUEZ, S.; LOZANO, J. Electronic nose with digital gas sensors connected via bluetooth to a smartphone for air quality measurements. *Sensors*, MDPI, v. 20, n. 3, p. 786, 2020.

BHATTACHARYYA, J. Guide to sktime – python library for time series data (compatible with sci-kit learn). *ANALYTICS INDIA MAGAZINE PVT LTD*, ANALYTICS INDIA MAGAZINE PVT LTD, 2021.

BONAH, E.; HUANG, X.; AHETO, J. H.; OSAE, R. Application of electronic nose as a non-invasive technique for odor fingerprinting and detection of bacterial foodborne pathogens: a review. *Journal of Food Science and Technology*, Springer, v. 57, n. 6, p. 1977–1990, 2020.

BRASIL. ResoluÇÃo anvisa re nº 09/2003, de 16 de janeiro de 2003. Agência Nacional de Vigilância Sanitária, 2003.

BROCKWELL, P. J.; DAVIS, R. A. *Introduction to time series and forecasting*. Third edition. Springer Science Business Media, LLC, 233 Spring Street, New York, NY 10013, USA: Springer, 2016.

COUTO, M. V. D. d. C.; MOTTA, C. M. d. S. Micota anemÓfila de centros cirÚrgicos do hospital das clÍnicas de pernambuco. 2021.

CRONIN, D.; WARD, M. K. The characterisation of some mushroom volatiles. *Journal of the Science of Food and Agriculture*, Wiley Online Library, v. 22, n. 9, p. 477–479, 1971.

DEMPSTER, A.; PETITJEAN, F.; WEBB, G. I. Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, Springer, v. 34, n. 5, p. 1454–1495, 2020.

DENG, H.; RUNGER, G.; TUV, E.; VLADIMIR, M. A time series forest for classification and feature extraction. *Information Sciences*, Elsevier, v. 239, p. 142–153, 2013.

FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. d. L. F. d. *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. Segunda edição. Travessa do Ouvidor, 11, Rio de Janeiro, RJ - CEP 20040-040, Brasil.: Grupo Editorial Nacional, 2021.

FAWAZ, H. I.; LUCAS, B.; FORESTIER, G.; PELLETIER, C.; SCHMIDT, D. F.; WEBER, J.; WEBB, G. I.; IDOUMGHAR, L.; MULLER, P.-A.; PETITJEAN, F. Inceptiontime: Finding alexnet for time series classification. *Data Mining and Knowledge Discovery*, Springer, v. 34, n. 6, p. 1936–1962, 2020.

GARCIA-ALCEGA, S.; NASIR, Z. A.; FERGUSON, R.; WHITBY, C.; DUMBRELL, A. J.; COLBECK, I.; GOMES, D.; TYRREL, S.; COULON, F. Fingerprinting outdoor air environment using microbial volatile organic compounds (mvocs)–a review. *TrAC Trends in Analytical Chemistry*, Elsevier, v. 86, p. 75–83, 2017.

HEDDERGOTT, C.; CALVO, A.; LATGE, J. The volatome of aspergillus fumigatus. *Eukaryotic Cell*, Am Soc Microbiol, v. 13, n. 8, p. 1014–1025, 2014.

HUMBER, R. A. Fungi: preservation of cultures. In: *Manual of techniques in insect pathology*. [S.l.]: Elsevier, 1997. p. 269–279.

HUNG, R.; LEE, S.; BENNETT, J. W. Fungal volatile organic compounds and their role in ecosystems. *Applied Microbiology and Biotechnology*, Springer, v. 99, n. 8, p. 3395–3405, 2015.

KUSKE, M.; ROMAIN, A.-C.; NICOLAS, J. Microbial volatile organic compounds as indicators of fungi. can an electronic nose detect fungi in indoor environments? *Building and environment*, Elsevier, v. 40, n. 6, p. 824–831, 2005.

LEE, Y.-H.; WEI, C.-P.; CHENG, T.-H.; YANG, C.-T. Nearest-neighbor-based approach to time-series classification. *Decision Support Systems*, Elsevier, v. 53, n. 1, p. 207–217, 2012.

LIN, R. A. K.-l.; SHIM, H. S. S. K. Fast similarity search in the presence of noise, scaling, and translation in time-series databases. In: CITESEER. *Proceeding of the 21th International Conference on Very Large Data Bases*. [S.l.], 1995. p. 490–501.

LINES, J.; TAYLOR, S.; BAGNALL, A. Time series classification with hive-cote: The hierarchical vote collective of transformation-based ensembles. *ACM Transactions on Knowledge Discovery from Data*, v. 12, n. 5, 2018.

LÖNING, M.; BAGNALL, A.; GANESH, S.; KAZAKOV, V.; LINES, J.; KIRÁLY, F. J. sktime: A unified interface for machine learning with time series. *arXiv preprint arXiv:1909.07872*, 2019.

LOULIER, J.; LEFORT, F.; STOCKI, M.; ASZTEMBORSKA, M.; SZMIGIELSKI, R.; SIWEK, K.; GRZYWACZ, T.; HSIANG, T.; ŚLUSARSKI, S.; OSZAKO, T. et al. Detection of fungi and oomycetes by volatiles using e-nose and spme-gc/ms platforms. *Molecules*, Multidisciplinary Digital Publishing Institute, v. 25, n. 23, p. 5749, 2020.

MAGESTE, J. D. O.; PEREIRA, T. C. D.; SILVA, G. A. D.; BARROS, R. A. M. D. Estudo da microbiota fúngica anemófila de uma indústria farmacêutica de juiz de fora–mg. *FACIDER-Revista Científica*, v. 1, n. 1, 2012.

MARTIN, A. C.; HENNING, E.; WALTER, O. M. F. C.; KONRATH, A. C. Análise de séries temporais para previsão da evolução do número de automóveis no município de joinville. *Revista ESPACIOS| Vol. 37 (Nº 06) Año 2016*, 2016.

MARTINS-DINIZ, J. N.; SILVA, R. A. M. d.; MIRANDA, E. T.; MENDES-GIANNINI, M. J. S. Monitoramento de fungos anemófilos e de leveduras em unidade hospitalar. *Revista de saúde pública*, SciELO Brasil, v. 39, p. 398–405, 2005.

MCINNES, L.; HEALY, J.; MELVILLE, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

MEDICINE, M. S. G. at Division of Infectious Diseases Heersink School of. *Rhizomucor Species*. 1720 2nd Ave South, Birmingham, AL 35294, USA.: The University of Alabama at Birmingham, 2020.

MEZZARI, A.; PERIN, C.; JÚNIOR, S. A. S.; BERND, L. A. G.; GESU, G. D. Os fungos anemófilos e sensibilização em indivíduos atópicos em porto alegre, rs. *Revista da Associação Médica Brasileira*, SciELO Brasil, v. 49, p. 270–273, 2003.

MIDDLEHURST, M.; LARGE, J.; FLYNN, M.; LINES, J.; BOSTROM, A.; BAGNALL, A. Hive-cote 2.0: a new meta ensemble for time series classification. *arXiv preprint arXiv:2104.07551*, 2021.

MIDDLEHURST, M.; VICKERS, W.; BAGNALL, A. Scalable dictionary classifiers for time series classification. In: SPRINGER. *International Conference on Intelligent Data Engineering and Automated Learning*. [S.l.], 2019. p. 11–19.

MOREIRA, C. Classificação de whittaker. *Revista de Ciência Elementar*, Casa das Ciências, v. 2, n. 4, 2014.

MOTA, I.; TEIXEIRA-SANTOS, R.; RUFO, J. C. Detection and identification of fungal species by electronic nose technology: A systematic review. *Fungal Biology Reviews*, Elsevier, v. 37, p. 59–70, 2021.

NGUYEN, T. L.; GSPONER, S.; ILIE, I.; O'REILLY, M.; IFRIM, G. Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. *Data mining and knowledge discovery*, Springer, v. 33, n. 4, p. 1183–1222, 2019.

NIEMINEN, T.; NEUBAUER, P.; SIVELÄ, S.; VATAMO, S.; SILFVERBERG, P.; SALKINOJA-SALONEN, M. Volatile compounds produced by fungi grown in strawberry jam. *LWT-Food Science and Technology*, Elsevier, v. 41, n. 10, p. 2051–2056, 2008.

PALMA, S. I.; TRAGUEDO, A. P.; PORTEIRA, A. R.; FRIAS, M. J.; GAMBOA, H.; ROQUE, A. C. Machine learning for the meta-analyses of microbial pathogens' volatile signatures. *Scientific Reports*, Nature Publishing Group, v. 8, n. 1, p. 1–15, 2018.

PONG, M. L. T. B. M. M. S. G. G. O. F. K. J. L. V. M. W. R. P. R. T. O. jesellier; Guzal Bulatova; Lovkush; Svea Marie Meyer; AidenRushbrooke; Patrick Schäfer; oleskiewicz; Y.-X. X. A. A. H. A. S. A. J. L. J. O. B. K. M. E. G. A. W. J. alan-turing-institute/sktime: v0.8.2. 2020.

SAVELIEVA, E. I.; GUSTYLEVA, L. K.; KESSENIKH, E. D.; KHLEBNIKOVA, N. S.; LEFFINGWELL, J.; GAVRILOVA, O. P.; GAGKAEVA, T. Y. Study of the vapor phase over fusarium fungi cultured on various substrates. *Chemistry & biodiversity*, Wiley Online Library, v. 13, n. 7, p. 891–903, 2016.

SCHÄFER, P.; LESER, U. Fast and accurate time series classification with weasel. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. [S.l.: s.n.], 2017. p. 637–646.

SOUZA, P. M. S. de; ANDRADE, S. L. de; LIMA, A. F. de. Pesquisa, isolamento e identificação de fungos anemófilos em restaurantes self-service do centro de maceió/al. *Caderno de Graduação-Ciências Biológicas e da Saúde-UNIT-ALAGOAS*, v. 1, n. 3, p. 147–154, 2013.

TAŞTAN, M.; GÖKOZAN, H. Real-time monitoring of indoor air quality with internet of things-based e-nose. *Applied Sciences*, MDPI, v. 9, n. 16, p. 3435, 2019.

TROVÃO, J.; PEREIRA, L. IntroduÇÃo ao estudo dos microfungos: Guia simples para a iniciação à identificação. 2019.

WEIKL, F.; GHIRARDO, A.; SCHNITZLER, J.-P.; PRITSCH, K. Sesquiterpene emissions from alternaria alternata and fusarium oxysporum: Effects of age, nutrient availability and co-cultivation. *Scientific reports*, Nature Publishing Group, v. 6, n. 1, p. 1–12, 2016.

WHITTAKER, R. H. New concepts of kingdoms of organisms: Evolutionary relations are better represented by new classifications than by the traditional two kingdoms. *Science*, American Association for the Advancement of Science, v. 163, n. 3863, p. 150–160, 1969.

YANG, Y.; CARBONELL, J. G.; BROWN, R. D.; PIERCE, T.; ARCHIBALD, B. T.; LIU, X. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems and their Applications*, IEEE, v. 14, n. 4, p. 32–43, 1999.

ZAMPARETTE, C. P. *GUIA COMPLETO: Comparação de Metodologias de Identificação de Microrganismos*. Florianópolis, SC, Brasil: Neoprospecta Microbiomes Tecnologies, 2017.