Pós-Graduação em Ciência da Computação

Poliana de Brito Santana Silva

IDENTIFICAÇÃO AUTOMÁTICA DE FRUTAS COM UM NARIZ ELETRÔNICO E MÉTODOS AVANÇADOS DE APRENDIZAGEM DE MÁQUINA

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
http://cin.ufpe.br/~posgraduacao

Recife
2020

Poliana de Brito Santana Silva

IDENTIFICAÇÃO AUTOMÁTICA DE FRUTAS COM UM NARIZ ELETRÔNICO E MÉTODOS AVANÇADOS DE APRENDIZAGEM DE MÁQUINA

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Área de Concentração: inteligência computacional
Orientador: Leandro Maciel Almeida

Recife
2020

Catalogação na fonte
Bibliotecária Arabelly Ascoli CRB4-2068

S586i Silva, Poliana de Brito Santana
Identificação automática de frutas com um nariz eletrônico e métodos avançados de aprendizagem de máquina / Poliana de Brito Santana Silva. – 2020.
118 f.: il. fig., tab

Orientador: Leandro Maciel Almeida
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn. Ciência da Computação. Recife, 2020.
Inclui referências e apêndice.

1. Nariz eletrônico. 2. Sensores. 3. Reconhecimento de padrões. 4. Compostos voláteis orgânicos. I. Almeida, Leandro Maciel. (orientador). II. Título.

006.31 CDD (22. ed.) UFPE-CCEN 2020-82

Poliana de Brito Santana Silva

“Identificação Automática de Frutas com um Nariz Eletrônico e Métodos Avançados de Aprendizagem de Máquina”

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.

Aprovado em: 30 de janeiro de 2020.

BANCA EXAMINADORA

_____

Prof. Dr. Fernando Maciano de Paula Neto
Centro de Informática/UFPE

_____

Prof. Dr. Frederico Duarte de Menezes
Instituto Federal de Pernambuco

_____

Prof. Dr. Leandro Maciel Almeida
Centro de Informática/UFPE
(**Orientador)**

Dedido este trabalho a minha família e a meu esposo que foram porto seguro perante as dificuldades durante este percurso.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer aos meus pais, Cicero e Rita, ao meu esposo, Allison, aos meu irmãos, Paulo e Ana Paula, aos meus sobrinhos, Miguel e Maitê, aos meus sogros, Eugênio e Aparecida e a minha cunhada Eumara, pelo apoio incondicional à realização do mestrado. Agradeço as mensagens de apoio, o carinho e os pensamentos positivos enviados mesmo de longe.

Um agradecimento especial ao meu esposo Allison, que esteve comigo lado a lado nessa jornada, que me levantou quando pensei que nada daria certo, que foi ao CIn comigo várias vezes só para eu fazer meus experimentos, que me apoia incondicionamente em qualquer decisão tomada, que é meu amigo, companheiro e parceiro. Só nos dois sabemos o quanto foi difícil deixar nossa cidade natal, Petrolina - PE, nossos familiares, nossa casa, para correr atrás dos nosso sonhos em uma cidade desconhecida, mas, foi algo que valeu a pena e que faria tudo novamente.

Gostaria de agradecer a uma pessoa que nos acolheu com todo amor em sua família e em seu lar, dona Josefa. Como foi maravilhoso, passar domingos e feriados em sua casa.

Quero agradecer as amigas que Recife(PE) me ofertou: Danny, Hellma, Sheila e Isabelle. Quem diria que uma simples aula de funcional para fugir da rotina, me traria amizades para o resto da vida. Agredeço muito a vocês meninas, por me acolherem, por me darem palavras de incentivo quando precisei, por cuidarem de mim nos momentos de fraqueza, por fazer os nossos momentos cheios de alegrias e ,claro, recheados de muitas risadas, comidas e diversão.

Agradeço ao meu orientador, Leandro Maciel, por toda atenção, por sua paciência e por acreditar em mim na conclusão desse trabalho.

RESUMO

Na década de 80, muitos protótipos foram construídos e testados com diferentes tipos de sensores químicos baseados no conceito de narizes eletrônicos. Atualmente, esses narizes estão disponíveis e podem ser aplicados em diversas áreas. Eles são capazes de produzir uma impressão digital para um odor específico, semelhante ao cérebro humano, quando um aroma é percebido, suas moléculas interagem com numerosos receptores, causando um sinal interpretado. Os narizes eletrônicos são como um sistema modular que detecta o odor associado a sensores que traduzem respostas químicas em sinais elétricos que são transmitidos e processados. Eles classificam odores conhecidos ou identificam os desconhecidos. Uma das áreas onde o nariz eletrônico está sendo aplicado é a da fruticultura. Atualmente, produtores de frutas, fornecedores e varejistas, tradicionalmente, utilizam testadores humanos para avaliar a qualidade dos seus frutos, mas, há algumas desvantagens em ter seres humanos realizando essas tarefas, devido a uma variedade de razões como fadiga, infecções, estado mental, subjetividade, variáveis individuais, entre outras. Sendo assim, se torna economicamente inviável investir financeiramente em treinamento para tarefas que duram relativamente pouco tempo. A fragrância da fruta é uma combinação complexa de dezena de aromas individuais que contribuem para caracterizar o aroma único de cada fruta. Essas características são de fundamental importância na escolha do consumidor, que julga a qualidade da fruta pelo visual (falta de manchas, cor, tamanho e textura), maturação, aroma e sabor. Os frutos liberam e produzem uma ampla variedade de Compostos Orgânicos Voláteis (VOCs), embora diferentes frutas compartilhem algumas características aromáticas, cada fruto possui um aroma distinto que depende da combinação específica de VOCs presentes nas misturas de aromas. Embora vários narizes eletrônicos comerciais estejam disponíveis no mercado, muitos são volumosos. Há de fato um número pequeno de narizes eletrônicos modernos, como o "Diagnose" do C-it da Holanda e o Artinose da SYSCA AG Alemanha, mas, os custos elevados desses produtos os tornam inviáveis para uma adoção generalizada. Esses narizes são compostos por uma matriz de sensores e podem conter entre 6 e 8 sensores. Pensando isso, foi produzido um prótotipo de nariz eletrônico de baixo custo baseado em um sensor de dióxido de estanho, esse equipamento reconhece os compostos orgânicos volatéis das frutas, gerando uma impressão digital para cada odor. As amostras das frutas foram coletadas em duas fases: frutas maduras intactas e cortadas. Com o auxílio de métodos de reconhecimento de padrões, essas frutas foram classificadas. O desempenho dos modelos foi avaliado através de um conjunto de métricas e técnicas estatísticas. Os resultados obtidos nessa pesquisa mostram-se satisfatórios e, com base nos resultados, é possível inferir que dados de um único sensor associado a técnicas de aprendizagem de máquina podem distinguir os compostos voláteis das frutas e, consequentemente, classificá-las.

Palavras-chaves: Nariz Eletrônico. Sensores. Odor. Reconhecimento de Padrões. Compostos Voláteis Orgânicos.

ABSTRACT

In the 1980, many prototypes were built and tested with different types of non-chemical chemical sensors. These data are currently available and can be used in several areas. They are able to produce a fingerprint for a specific odor, similar to the human brain, when an aroma is perceived, its molecules interact with numerous receptors, causing an interpreted signal. Electronic noses are like a modular system that allows detecting odors associated with sensors that translate chemical responses into electrical signals and are transmitted and processed. They classify known odors or identify unknown ones. One of the areas where the electronic nose is being applied is fruit growing. Currently, fruit growers, suppliers and retailers have traditionally used human testers to assess the quality of their fruit, but there are some advantages in humans performing these tasks, due to a varied variety of fatigue, illness, mental state, subjectivity, variables Individual, among others. Therefore, it becomes economically unfeasible to invest financially in training for tasks that last a short time. A fruit fragrance is a complex combination of ten individual aromas that contribute to characterize the unique aroma of each fruit. These characteristics are of fundamental importance to the consumer, who judges the quality of the fruit by its appearance (lack of spots, color, size and texture), ripeness, aroma and flavor. The fruits release and produce a variety of volatile organic compounds (VOCs), although different fruits with some aromatic characteristics, each fruit has a distinct aroma depending on the selected combination of VOCs present in the aroma mixtures. Emboravários narrates commercial electronics available in the market, many are bulky. There are a small number of modern electronic numbers, such as "Diagnosis" of C-it from the Netherlands and Artinose from SYSCA AG Germany, but the high costs of these products or those that are not feasible for a generalized application. These resources are composed of a matrix of sensors and can contain between 6 and 8 sensors. With this in mind, a low cost electronic nose prosthesis was produced, based on a carbon dioxide sensor, this equipment recognizes the volatile organic compounds in fruits, generating a fingerprint for each odor. The fruit bulbs were collected in two stages: ripe and intact ripe fruits. With the aid of pattern recognition methods, these fruits were classified. The performance of the models was evaluated through a set of metrics and statistical techniques. The results obtained in this research are satisfactory and, based on the results, it is possible to infer data from a single sensor associated with machine learning techniques that can differentiate volatile compounds from fruits and, consequently, classify them

Keywords: Electronic Nose. Sensors. Odor. Pattern Recognition. Organic Volatile Compounds.

LISTA DE FIGURAS

Figura 1 – Semelhança entre o sistema biológico olfativo e um nariz eletrônico . . 20
Figura 2 – Variação do volume molecular no filme do sensor, devido à absorção de moléculas odorantes e, consequentemente, variação nas resistências elétricas 35
Figura 3 – Linha de base sem alteração 36
Figura 4 – Presença do odor, alteração da linha base 36
Figura 5 – Presença de novo odor, nova alteração da linha de base 36
Figura 6 – Classificação dos métodos de detecção de gás. 38
Figura 7 – Impacto do valor de k no algoritmo K-NN 43
Figura 8 – Hiperplanos (a) com margem pequena e (b) com margem elevada . . . 45
Figura 9 – Visão Geral do Método Árvores Aleatórias 47
Figura 10 – Neurônio Artificial 49
Figura 11 – Exemplos de Função de Ativação 49
Figura 12 – Exemplo de RNA multicamadas típica 50
Figura 13 – Exemplo MLP 51
Figura 14 – Exemplo do papel dos neurônios das diferentes camadas de uma MLP . 52
Figura 15 – Exemplos de grades hiperparamétricas para (a) uma, (b) duas e (c) três dimensões. 54
Figura 16 – Imagem de entrada de um melão para sistema de visão computacional 60
Figura 17 – Imagem do melão em tons de cinza 60
Figura 18 – Melão detectado por análise de bordas 61
Figura 19 – Identificação das medidas D e d extraídas da imagem do melão 61
Figura 20 – Fluxograma ilustrativo da metodologia adotada 67
Figura 21 – Diagrama do Sistema Proposto 69
Figura 22 – Atividade do Sensor - Experimento Teste. (a) amostras de laranja. (b) amostras de banana. (c) amostras de uva 70
Figura 23 – Ilustração do Ciclo de um Grid-Search na busca de Parâmetros Ótimos 76
Figura 24 – Resposta do Sensor: Mudança de Resistência para a Banana Cortada . 77
Figura 25 – Resposta do Sensor: Mudança de Resistência para a Maça Cortada . . 78
Figura 26 – Resposta do Sensor: Mudança de Resistência para a Uva Cortada . . . 78
Figura 27 – Resposta do Sensor: Mudança de Resistência para a Laranja Cortada . 78
Figura 28 – Resposta do Sensor: Mudança de Resistência para a Banana Intacta . . 79
Figura 29 – Resposta do Sensor: Mudança de Resistência para a Maçã Intacta . . . 79
Figura 30 – Resposta do Sensor: Mudança de Resistência para a Laranja Intacta . . 79
Figura 31 – Score por parâmetro do KNN - Banco 1 (Frutas Cortadas) 82
Figura 32 – Score por parâmetro do RF - Banco 1 (Frutas Cortadas) 83

Figura 33 – Score por parâmetro do SVM - Banco 1 (Frutas Cortadas) 83
Figura 34 – Score por parâmetro do MLP - Banco 1 (Frutas Cortadas) 83
Figura 35 – Score por parâmetro do GBOOST - Banco 1 (Frutas Cortadas) 84
Figura 36 – Score por parâmetro do LIGTH GBM - Banco 1 (Frutas Cortadas) . . 84
Figura 37 – Score por parâmetro do KNN - Banco 2 (Frutas Intactas) 84
Figura 38 – Score por parâmetro do RF - Banco 2 (Frutas Intactas) 85
Figura 39 – Score por parâmetro do SVM - Banco 2 (Frutas Intactas) 85
Figura 40 – Score por parâmetro do MLP - Banco 2 (Frutas Intactas) 85
Figura 41 – Score por parâmetro do GBOOST - Banco 2 (Frutas Intactas) 86
Figura 42 – Score por parâmetro do LIGTH GBM - Banco 2 (Frutas Intactas) . . . 86
Figura 43 – Comparação dos Algoritmos - Banco 1 (Frutas Cortadas) - Valores de p-value 93
Figura 44 – Comparação dos Algoritmos - Banco 2 (Frutas Intactas) - Valores de p-value 93
Figura 45 – Comparação dos Algoritmos - Banco 1 (Frutas Cortadas) 94
Figura 46 – Comparação dos Algoritmos - Banco 2 (Frutas Intactas) 94

LISTA DE TABELAS

Tabela 1 – Exemplos de aplicações utilizando narizes eletrônico 16
Tabela 2 – Alguns narizes, modelos e tecnologias eletrônicos comercialmente disponíveis 23
Tabela 3 – Exemplo de aplicativos baseados no setor para Nariz Eletrônico 25
Tabela 4 – A composição típica de compostos orgânicos voláteis (VOC) em cada estágio da maturidade da banana 31
Tabela 6 – Compostos voláteis identificados na casca e polpa de uvas Moscatel . . 33
Tabela 7 – Compostos voláteis identificados na laranja pêra 34
Tabela 8 – Funções de *kernel* 46
Tabela 9 – Matriz de Confusão de problema contendo duas classes 55
Tabela 10 – Recursos, aplicações, faixa de detecção de partículas e preço do sensor Figaro TGS 822 68
Tabela 11 – Datas da coleta das frutas cortadas e das intactas 71
Tabela 12 – Principais Parâmetros Variados no Algoritmo para a Busca em Grade . 74
Tabela 13 – Resultados dos Melhores parâmetros para cada Algoritmo - Banco 1 (Frutas Cortadas) 80
Tabela 14 – Resultados dos Melhores parâmetros para cada Algoritmo - Banco 2 (Frutas Intactas) 81
Tabela 15 – Resultados Métricas de Avaliação - Banco 1 (Frutas Cortadas) 86
Tabela 16 – Resultados Métricas de Avaliação - Banco 2 (Frutas Intactas) 88
Tabela 17 – Matriz de confusão para todos os classificadores - Banco 1 (Frutas Cortadas) 90
Tabela 18 – Matriz de confusão para todos os classificadores - Banco 2 (Frutas Intactas) 91

LISTA DE ABREVIATURAS E SIGLAS

ABC	Colônia de Abelhas Artificiais
AM	Aprendizagem de Máquina
AUCROC	Area Under the Receiver Operating Characteristics (Área Sob a Curva ROC)
BAW	Bulk Acoustic Wave (Sensor de Ondas Acústicas a Granel)
CG/MS	Cromatografia Gasosa Com Espectrometria De Massa
CNT	Nanotubo De Carbono
CP	Polímeros condutores
CSA	Sensores Colorimétricos
DFA	Análise Fatorial Discriminante
FN	Falso Negativo
FP	Falso Positivo
GBM	Gradient Boosting Machine
GC	Cromatografia Gasosa
GF	Gráfeno
HS-SPME	Headspace Solid Phase Microextraction
IC	Insuficiência Cardiaca
IPC	Polímeros Intrinsecamente condutores
KNN	K-Nearest Neighbors (k- Vizinhos Mais Próximos)
LDA	Análise Discriminante Linear
LGBM	LightGBM
LLE	Incorporação Linear Local
MLP	Multilayer Perceptron (Perceptron Multicamadas)
MOS	Semicondutores De Óxido Metálico

OTAN	Organização do Tratado do Atlântico do Norte
PCA	Análise de Componentes Principais
PLS	Mínimos Quadrados Parciais
PLSR	Regressão de Mínimos Quadrados Parciais
RF	Random Forest (Florestas Aleatórias)
RNA	Redes Neurais Artificiais
SAW	Surface Acoustic Wave Sensor (Sensor de Ondas Acústicas)
SVM	Máquinas de Vetores de Suporte
VC	Visão Computacional
VN	Verdadeiro Negativo
VOC	Composto Orgânico Volátil
VP	Verdadeiro Positivo

SUMÁRIO

1 INTRODUÇÃO 16
1.1 OBJETIVO GERAL 18
1.2 OBJETIVOS ESPECÍFICOS 18
1.3 ORGANIZAÇÃO DO TRABALHO 18

2 NARIZ ELETRÔNICO 20
2.1 DESCRIÇÃO DO NARIZ ELETRÔNICO 21
2.2 DESENVOLVIMENTO CONCEITUAL DO NARIZ ELETRÔNICO 22
2.3 APLICAÇÕES DO NARIZ ELETRÔNICO 25
2.3.1 Medicina 27
2.3.2 Qualidade de alimentos 28
2.3.3 Qualidade de bebidas 29
2.4 CARACTERISTICAS QUÍMICAS DOS VOLÁTEIS DE FRUTAS 30
2.4.1 Bioquímica da banana 30
2.4.2 Bioquímica da maçã 32
2.4.3 Bioquímica da uva 33
2.4.4 Bioquímica da laranja 34
2.5 SENSORES 35
2.5.1 Características e funcionamento de um sensor 35
2.5.2 Tipos de sensores 38
2.5.2.1 Sensores de óxidos metálicos (MOS) 38
2.5.2.2 Sensores de polímeros condutores (CP) 39
2.5.2.3 Sensores colorimétricos (CSA) 40
2.5.2.4 Sensores de ondas acústicas 40
2.5.2.5 Sensores com nanomateriais 41
2.6 MÉTODOS DE RECONHECIMENTO DE PADRÕES 42
2.6.1 K-vizinhos mais próximos (KNN) 42
2.6.2 Máquinas de vetor de suporte (SVM) 44
2.6.3 Florestas aleatóricas (RF) 46
2.6.4 Redes neurais artificiais (RNA) 47
2.6.4.1 Multilayer perceptron (MLP) 51
2.6.5 Xgboost 52
2.6.6 Lightgbm 53
2.6.7 Grid search 54
2.7 AVALIAÇÃO DE CLASSIFICADORES E MÉTRICAS 55
2.7.1 Validação cruzada - K-fold 55

2.7.2 Métricas . . . 55
2.7.2.1 Matriz de confusão . . . 55
2.7.2.2 Métricas de qualidade . . . 56
2.8 COMPARAÇÃO DE CLASSIFICADORES . . . 57
2.8.1 Teste de Kruskal-Wallis . . . 57

3 TRABALHOS RELACIONADOS . . . 59
3.1 IDENTIFICAÇÃO AUTOMÁTICA DE FRUTAS . . . 59
3.1.1 Visão computacional na fruticultura . . . 59
3.1.2 Narizes eletrônicos na fruticultura . . . 62
3.1.2.1 Bananas . . . 63
3.1.2.2 Maçã . . . 64
3.1.2.3 Uvas . . . 65
3.1.2.4 Laranjas . . . 65

4 MATERIAIS E MÉTODOS . . . 67
4.1 CONSTRUÇÃO DO EQUIPAMENTO . . . 67
4.2 PROTOCOLO DE EXPERIMENTAÇÃO . . . 69
4.3 COLETA DAS AMOSTRAS . . . 71
4.4 FORMAÇÃO DA BASE DE DADOS . . . 72
4.5 PROCEDIMENTOS DE PROCESSAMENTO DOS DADOS . . . 73
4.6 BUSCA EM GRADE - VARIAÇÃO DE HIPERPARÂMETROS . . . 74
4.7 PROCEDIMENTO DE AVALIAÇÃO DOS DADOS . . . 76

5 RESULTADOS . . . 77
5.1 RESPOSTAS GERADAS PELO SENSOR . . . 77
5.2 RESPOSTA GERADAS PELA BUSCA EM GRADE . . . 80
5.2.1 Resultados das classificações e discursões . . . 86
5.2.2 Comparação dos modelos . . . 92
5.2.3 Comparação com outros trabalhos . . . 94

6 CONCLUSÃO E TRABALHOS FUTUROS . . . 96
6.1 TRABALHOS FUTUROS . . . 97

REFERÊNCIAS . . . 98

APÊNDICE A – CÓDIGO PHYTON METÓDOS AVANÇADOS DE CLASSIFICAÇÃO . . . 103

1 INTRODUÇÃO

Pesquisas sobre métodos alternativos de sensoriamento olfativo já percorreram um longo caminho desde sua introdução em 1982. Avanços na tecnologia de sensores de aroma, eletrônica, bioquímica e inteligência artificial tornaram possível desenvolver dispositivos capazes de medir e caracterizar aromas voláteis liberados de uma infinidade de fontes para inúmeras aplicações. Esses equipamentos, conhecidos como narizes eletrônicos, foram desenvolvidos para imitar o sistema oftativo de mamíferos dentro de um instrumento projetado para obter medições reprodutíveis, permitindo identificações e classificações de misturas de aromas ao mesmo tempo que elimina a fadiga do operador (WILSON; BAIETTO, 2009).

O termo nariz eletrônico é geralmente confundido com um equipamento que é capaz de identificar odores como um nariz humano, sendo, de fato, esta definição equivocada. Os narizes eletrônicos normalmente são constituídos de uma matriz de sensores químicos baseados em materiais de óxido metálico. Mas, como os narizes humanos, esses dispositivos não são capazes de identificar as substâncias separadamente em uma determinada amostra. Os dados produzidos por esses sensores podem fazer uma analogia com a impressão digital, uma vez que, é muito difícil de encontrar duas substâncias que gerem a mesma resposta para todo conjunto de sensores, sendo possível classificar substâncias de acordo com suas características (VOSS, 2019). Esses instrumentos são constituídos em quatro partes: matriz de sensores, unidade de processamento de sinais, banco de dados e reconhecimento de padrões. Essas quatro partes simulam, respectivamente, a aquisição de informações por neurônios sensoriais do receptor olfatório humano, a codificação do nervo olfatório (bulbo), a memória cerebral e processamento de informações pelo sistema olfativo humano (HUANG et al., 2018).

Atuamente, esses dispositivos vêm sendo usados em diversas áreas, sendo cada vez mais empregados como alternativas aos métodos tradicionais, tendo aplicação na área médica, qualidade de alimentos, qualidade de bebidas, qualidade do ar e na agricultura . O interesse nessa tecnologia tem crescido e se tornado evidente o surgimento de trabalhos e artigos sobre o tema, tanto para desenvolvimento quanto para o aprimoramento de equipamentos nos diversos setores da indústria (VOSS, 2019). A tabela 1 resume algumas das várias aplicações possíveis para o nariz eletrônico.

Tabela 1 – Exemplos de aplicações utilizando narizes eletrônico

Aplicação	**Referências**
Área Médica	Voss et al. (2012), Lorwongtragool et al. (2014), Huang et al. (2018)

Qualidade de Alimentos	Gómez et al. (2006), Rajamaki et al. (2006), Garcia, Aleixandre e Horrillo (2005)
Qualidade de Bebidas	Raggazzo-Sanchez et al. (2008), Mutabil et al. (2017) Xiao, Wang e Wang (2017)
Qualidade do ar	Bagula et al. (2012), Abraham e Pandian (2013, Peterson et al. (2017)

Fonte: A Autora (2020)

Embora vários narizes eletrônicos comerciais estejam disponíveis no mercado, muitos são volumosos. Há de fato um número pequeno de narizes eletrônicos modernos, como o "Diagnose" do C-it da Holanda e o Artinose da SYSCA AG Alemanha, mas, os custos elevados desses produtos os tornam inviáveis para uma adoção generalizada. Esses narizes são compostos por uma matriz de sensores podendo conter de 6 até 8 sensores (WILSON; BAIETTO, 2009).

Uma das áreas onde o nariz eletrônico está sendo aplicado é a da fruticultura. Atualmente, produtores de frutas, fornecedores e varejistas tradicionalmente utilizam testadores humanos para avaliar a qualidade dos seus frutos, mas há algumas desvantagens em ter seres humanos realizando essas tarefas, devido a uma variedade de razões como fadiga, infecções, estado mental, subjetividade, variáveis individuais, entre outras. Sendo assim economicamente inviável investir financeiramente em treinamento para tarefas que duram relativamente pouco tempo (BAIETTO; WILSON, 2015).

O odor da fruta é uma combinação complexa de dezena de aromas individuais que contribuem para caracterizar o aroma único de cada fruta. Essas características são de fundamental importância na escolha do consumidor, que julga a qualidade da fruta pelo visual (falta de manchas, cor, tamanho e textura), maturação, aroma e sabor (TANG et al., 2010). Os frutos liberam e produzem uma ampla variedade de Compostos Orgânicos Voláteis (VOCs), embora diferentes frutas compartilhem algumas características aromáticas, cada fruto possui um aroma distinto que depende da combinação específica de VOC's presentes nas misturas de aromas. O advento dos dispositivos de nariz eletrônico ofereceu ferramentas alternativas para classificar frutas e outros alimentos perecíveis usando medidas qualitativas e quantitativas mais consistentes de caracteríistivas aromáticas que evitam as opiniões subjetivas altamente variáveis de classificadores humanos (BAIETTO; WILSON, 2015).

Com isso, esse trabalho apresenta o estudo e o desenvolvimento de um prótotipo de um nariz eletrônico de baixo custo baseado em um sensor de dióxido de estanho a fim de classificar diferentes tipo de frutas em dois estágios: frutas maduras intactas e cortadas. Além disso, é necessário o estudo das diferentes ferramentas para processamento

e classificação dos dados gerados pelo sensor, a fim de obter uma classificação precisa das frutas nos dois estágios.

1.1 OBJETIVO GERAL

Desenvolver uma solução para a classificação de frutas utilizando nariz eletrônico dotado de sensores e métodos avançados de inteligência artificial, afim de classificar quatro tipos de frutas (banana, maçã, laranja e uva) de forma automática, divididas em dois estágios: frutas maduras intactas e cortadas, através da presença dos gases exalados por elas.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos epecíficos do trabalho estão descritos a seguir.

- Construir um nariz eletrônico de baixo custo, baseado em um sensor de óxido metálico.
- Coletar e formar a base de dados, com dados adquiridos durante os dois estágios.
- Identificar e implementar os algoritmos adequados para a classificação das frutas nos dois estágios.
- Demonstrar a capacidade do equipamento em classificar frutas maduras intactas e cortadas.
- Comparar os resultados obtidos do equipamento desenvolvido com aqueles encontrados na literatura.

1.3 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está dividido nos seguintes capítulos:

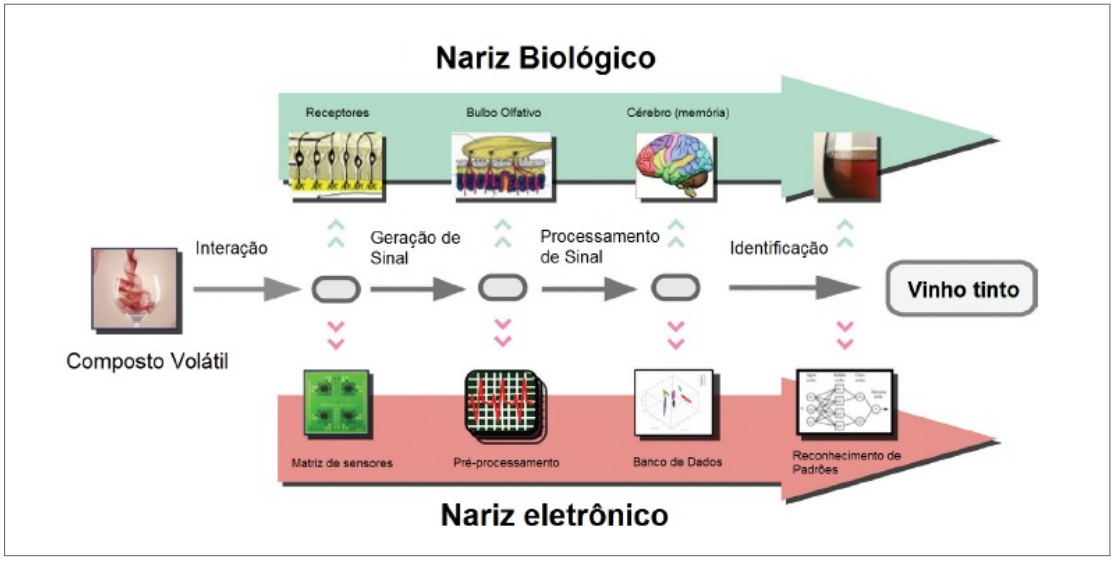
- Capítulo 1: Este capítulo apresentou uma introdução ao estudo abordado e tratará dos objetivos dessa dissertação.
- Capítulo 2: Apresenta a revisão da literatura referente ao assunto abordado.
- Capítulo 3: Descreve os trabalhos relacionados e o estado da arte de aplicações de narizes eletrônicos na fruticultura.
- Capítulo 4: Este capítulo apresenta os materiais e métodos adotados para alcançar os objetivos propostos.
- Capítulo 5: Apresenta os resultados obtidos além de realizar uma discursão.

- Capítulo 6: Nesta capítulo é apresentada uma breve conclusão obtida no desenvolvimento deste trabalho, mostrando suas contribuições e os possíveis trabalhos futuros que tomarão essa dissertação como base.

2 NARIZ ELETRÔNICO

Um Nariz Eletrônico é um dispositivo projetado para detectar e discriminar odores complexos usando um conjunto de sensores químicos. São baseados no processo biológico de interpretação do odor, trabalhando de maneira similar, com sensores que substituem os receptores do nariz humano, transmitindo um sinal a um programa que o processará e simulará a interpretação cerebral(LISBOA; PAGE; GUY, 2009) . A figura 1 demonstra essa similaridade. No Nariz Biológico, a amostra é reconhecida pelos receptores, que por sua vez geram um sinal que é enviado para o bulbo olfativo. O bulbo processa esses sinais, enviando-os para o cérebro que, através da mémoria identifica a amostra. Para o Nariz Eletrônico o processo é similar, na primeira etapa o contato com a amostra é realizado pela matriz de sensores, que realizará a geração do sinal que será pré-processado por algoritmos especifícos e geram dados que são armazenados em um banco de dados. Por fim, a identificação da amostra é feita através de algoritmos de reconhecimento de padrões (VOSS, 2019)

Figura 1 – Semelhança entre o sistema biológico olfativo e um nariz eletrônico



Fonte: (VOSS, 2019)

Desde meados dos anos 80, o conceito de Nariz Eletrônico tem sido discutido por muitos autores e vários protótipos foram criados e testados para diferentes tipos de sensores químicos (MARTINS, 2011). Atualmente, esses dispositvos vem sendo usados em diversas áreas, sendo cada vez mais empregados como alternativas aos métodos tradicionais, tendo aplicações na área médica, agricultura, qualidade de alimentos, qualidade de bebidas e qualidade do ar (VOSS, 2019).

O Nariz Eletrônico é um instrumento que pode aprender e reconhecer quase todos

os compostos ou combinações de compostos. Esse dispositivo é capaz de produzir uma "impressão digital"para um odor específico (LISBOA; PAGE; GUY, 2009).

É uma ferramenta que pode ser usada para monitorar a segurança, qualidade de processo, com respostas rápidas que poderiam levar dias para serem apresentadas por outros procedimentos, tais como a olfatometria (medida de odores), que consiste na utilização do olfato, poderosa ferramenta sensorial do ser humano como um instrumento de análise laboratorial (QUADROS et al., 2010). A parte mais complicada do Nariz Eletrônico é a tecnologia de sensores a serem utilizados para captura dos aromas. Qualquer sensor reversível, ou seja, aquele que seu material retorna ao estado inicial após detecção, e que, reaja a uma substância química ou gasosa, tem a competência para ser desenvolvido para uso no formato de nariz eletrônico(MARTINS, 2011).

2.1 DESCRIÇÃO DO NARIZ ELETRÔNICO

Os sistemas de reconhecimento de odores são constituídos por um *Hardware* e um *Software*. Neles, um sistema de amostragem coleta uma parcela de odor, levando-a ao conjunto de sensores, que convertem as variáveis químicas em sinais elétricos que são conduzidos a um computador onde são processados e interpretados. Portanto, o Nariz Eletrônico é composto por um arranjo de sensores, por um sistema de aquisição de amostras e por um sistema de processamento e classificação de odores. Durante o processo, uma amostra gasosa é injetada sobre o conjunto de sensores. O sinal é digitalizado e alimenta um computador, quando então, o sistema de processamento identifica os compostos (LISBOA; PAGE; GUY, 2009).

Alguns métodos de processamento são utilizados para analisar os dados com a finalidade de reproduzir funções típicas do sistema olfativo, por exemplo, a percepção de um odor e sua classificação, através de comparações com estímulos similares percebidos no passado. Muitas técnicas de reconhecimento de padrões são utilizadas para esse propósito, por exemplo, analises de funções discriminatórias, algoritmos diversos e redes neurais artificiais (MARTINS, 2011).

A caracterização química de um conjunto de sensores determina um conjunto de respostas, superpondo diferentes sensibilidades dos sensores individuais, ou seja, produz respostas a muitos compostos com estruturas moleculares diferentes (LISBOA; PAGE; GUY, 2009).

A parte mais importante de um nariz eletrônico é o método de amostragem de coleta. O sistema mais simples possível é a amostragem manual de *headspace*, onde o material aromático é armazenado em um volume fechado. Os compostos voláteis são retirados do recipiente usando uma seringa e injetados na câmara do sensor, mantida a uma temperatura constante, e purgados com um gás de referência limpo (geralmente o ar ambiente filtrado) após as leituras do sensor (WILSON; BAIETTO, 2009).

2.2 DESENVOLVIMENTO CONCEITUAL DO NARIZ ELETRÔNICO

Os primeiros estudos abrangendo as medições de aromas foram realizados na década de 1920 por Zwaardemaker e Hogewind. Eles sugeriram que os odores pudessem ser identificados medindo a carga elétrica desenvolvida por um fino spray de água que continha o odorante em solução, mas não conseguiram transformá-lo em um instrumento utilizável. O primeiro dispositivo real a medir aromas foi desenvolvido em 1954 por Hartman e seus colegas. Eles descrevem um sensor eletroquímico composto por microeletrodo, um simples fio de platina de 0,8 mm de diâmetro que media o fluxo de corrente por um milivoltímetro sensível. A primeira ideia de que o instrumento poderia operar com vários elementos sensíveis e que algumas substâncias diferentes de revestimentos de eletrodo poderiam ser aptos a fornecer respostas diferenciais com diferentes compostos veio também de Hartman (WILSON; BAIETTO, 2009).

Em 1960 os sensores de gás baseados em óxidos de metal (MOS) foram usados pela primeira vez. Eles eram usados como alarmes de gás doméstico no Japão sob os nomes de Taguchi ou Figaro. Moncrieff (1961), trabalhou no conceito de que diferentes materiais de revestimento, tais como cloreto de polivinil, gelatina e gorduras vegetais, eram capazes de fornecer informações diferentes e complementares para a discriminação de aromas simples e complexos. Seus estudos limitaram-se ao uso de um único resistor sensível à temperatura, mas mostraram que uma matriz com seis termistores, provida de seis revestimentos diferentes, poderia distinguir uma grande variedade de aromas (G.SUJATHA et al., 2012).

O princípio de sensores de BAW foi apresentado por King em 1964 (G.SUJATHA et al., 2012). Já em 1965, surgem dois novos grupos que publicaram estudos e experimentos com dispositivos de olfação. Buck, Allen e Dalton (1965) estudaram a modulação da condutividade como resposta à diferenciação de buquê de aromas, enquanto Dravnieks e Trotter (1965) utilizaram a modulação do potencial de contato para monitorar aromas. Devido a falta de instrumentos analíticos, esses estudos foram considerados apenas uma primeira abordagem à avaliação de aromas. Toda via, cerca de 20 anos depois, a ideia de um instrumento de nariz eletrônico com um sistema inteligente de sensores de matriz química para classificar aroma resultou de estudos de Persaud e Dodd (1982) e Ikegami e Kaneyasu (1985). Nesse período, o desenvolvimento de computadores e sensores eletrônicos tornou conceitualmente possível a obtenção de um dispositivo eletrônico capaz de imitar o sistema olfativo de mamíferos.

A palavra "Nariz Eletrônico" foi cunhada em 1988 por Gardner e Bartlett, que logo depois o definiram como *"Um instrumento que compreende uma série de sensores químicos eletrônicos com especificidade parcial e sistema de reconhecimento de padrões apropriado, capaz de reconhecer odores simples ou complexos"*. O primeiro Workshop Avançado de Processamento de Informações Quimiossensorial foi realizado em 1991 durante uma sessão da Organização do Tratado do Altântico do Norte (OTAN), que foi inteiramente dedicada ao tema de olfato arficial (WILSON; BAIETTO, 2009).

Desde então, o interesse na tecnologia de sensores tem crescido, e é evidente o surgimento de trabalhos e artigos sobre o tema, tanto para o desenvolvimento, quanto para o aprimoramento de equipamentos nos diversos setores da indústria. No presente momento, a atenção dos pesquisadores é mais focada em diminuir o tempo de análise dos narizes eletrônicos, aumentar a sensibilidade de configuração de medida, reduzir o tamanho do dispositivo e validar os métodos empregados (VOSS, 2019).

Alguns Narizes Eletrônico estão comercialmente disponíves e têm uma vasta gama de aplicações em vários mercados e indústrias, desde fabricação industrial, controle de qualidade, proteção ambiental, segurança, aplicações farmacêuticas, médicas e de diagnósticos. Estes sensores representam uma ampla diversidade de tipos de sensores baseados em tecnologias diferentes, incluindo instrumentos com matrizes de sensores de tecnologia única e instrumentos de tecnologia combinadas que consistem em narizes eletrônicos trabalhando em conjunto com instrumentos puramente analíticos clássicos (WILSON; BAIETTO, 2009). Um resumo de alguns dos narizes eletrônicos mais utilizados com fabricantes, modelos disponíveis e base tecnológica estão listados na tabela 2.

Tabela 2 – Alguns narizes, modelos e tecnologias eletrônicos comercialmente disponíveis

Tipo de Instrumento	**Fabricante**	**Modelos Produzidos**	**Base Tecnologica**
Tecnologia única (apenas sensores de nariz eletrônico)	Airsense Analytics	i-Pen, PEN2, PEN3	MOS sensors
	Alpha MOS	FOX 2000, 3000, 4000	MOS sensors
	Applied Sensor	Air quality module	MOS sensors
	Chemsensing	ChemSensing Sensor array	Colorimetric optical
	CogniScent Inc.	ScenTrak	Dye polymer sensors
	CSIRO	Cybernose	Receptor-based array
	Dr. Födisch AG	OMD 98, 1.10	MOS sensors
	Forschungszentrum Karlsruhe	SAGAS	SAW sensors
	Gerstel GmbH Co.	QSC	MOS sensors
	GSG Mess- und Analysengeräte	MOSES II	Modular gas sensors

Continua na próxima página

Tabela 2 – continuação da página anterior

Tipo de Instrumento	Fabricante	Modelos Produzidos	Base Tecnologica
	Illumina Inc.	oNose	Fluorescence optical
	Microsensor Systems Inc	Hazmatcad, Fuel Sniffer, SAW MiniCAD mk II	SAW sensors
	Osmetech Plc	Aromascan A32S	Conducting polymers
	Sacmi	EOS 835, Ambiente	Gas sensor array
	Scensive Technol.	Bloodhound ST214	Conducting polymers
	Smiths Group plc	Cyranose 320	Carbon black-polymers
	Sysca AG	Artinose	MOS sensors
	Technobiochip	LibraNose 2.1	QMB sensors
Tecnologia combinada (nariz eletrônico + outros tipos)	Airsense Analytics	GDA 2	MOS, EC, IMS, PID
	Alpha MOS	RQ Box, Prometheus	MOS, EC, PID, MS
	Electronic Sensor Technology	ZNose 4200, 4300, 7100	SAW, GC
	Microsensor Syst.	Hazmatcad Plus	SAW, EC
		CW Sentry 3G	SAW, EC
	Rae Systems	Area RAE monitor	CB, O2, EC, PID
		IAQRAE	Thermistor, EC, PID, CO2, humidity
	RST Rostock	FF2, GFD1	MOS, QMB, SAW

Adaptado de: (WILSON; BAIETTO, 2009)

O uso dessa tecnologia cresceu rapidamente à medida que novas aplicações foram descobertas. A quantidade de narizes eletrônicos vendidos por vários fabricantes depende bastante da base tecnólogica individual dos instrumentos, custo por unidade e a necessidade específica (F.ROCK; N.BARSAN; WEIMAR, 2008). No ano de 1997, existia cerca de 500 unidades de equipamentos analitícos de mesa vendidos no mundo inteiro, com o valor

aproximado de 30 milhões de euros. Nos últimos anos, a Applied Sensor Company vendeu a maioria das unidades de seu nariz eletrônico, usado principalmente para manter a qualidade do ar ambiente, detectar odores, compostos volatéis e dióxido de carbono nos espaços de conveniência (WILSON; BAIETTO, 2009).

2.3 APLICAÇÕES DO NARIZ ELETRÔNICO

Os sistemas de nariz eletrônicos foram desenvolvidos para serem usados em inúmeras aplicações em diversos processos de produção industrial. Uma ampla variedade de indústrias utilizam os narizes eletrônicos para uma diversificada gama de aplicações, tais como, controle de qualidade de matérias-primas e manufaturados, prazo de validade, avaliação de autencidade de produtos, classificação de perfumes, detecção de patógenos microbianos e estudos de avaliação ambiental (WILSON; BAIETTO, 2009). A tabela 3 mostra alguns exemplos individuais de aplicações do nariz eletrônico em cada uma dessas indústrias. Logo a seguir será discutido com mais detalhe alguns estudos em diversas áreas de aplicação.

Tabela 3 – Exemplo de aplicativos baseados no setor para Nariz Eletrônico

Setor industrial	**Área de aplicação**	**Tipos e exemplos de uso específico**
Agricultura	Proteção de culturas; Tempo e armazenamento da colheita; Produtos de carne, frutos do mar e peixes; Produção vegetal; Doenças pré e pós-colheita.	Segurança interna,fornecimento seguro de alimentos , amadurecimento das culturas, tratamento de preservação, frescura, contaminação, deterioração, seleção de cultivares, características das variedades, diagnóstico de doenças de plantas, identificação de pragas.
Transporte Aéreo	Segurança pública e bem-estar; Segurança de passageiros e pessoal.	Detecção de materiais explosivos e inflamáveis

Continua na próxima página

Tabela 3 – continuação da página anterior

Setor industrial	**Área de aplicação**	**Tipos e exemplos de uso específico**
Cosméticos	Produtos de aplicação pessoal ; Aditivos para fragrâncias	Desenvolvimento de perfume e colônia Aprimoramento do produto, apelo do consumidor
De Meio Ambiente	Monitoramento da qualidade do ar e da água; Controle de qualidade do ar interno ; Regulamentos de redução da poluição .	Detecção de poluição, efluentes, derramamentos tóxicos Emissões de mau cheiro, gases tóxicos / perigosos Controle de liberações de poluição por fontes pontuais
Comida e Bebida	Prevenção de fraude do consumidor; Avaliações de controle de qualidade; Maturação, contaminação de alimentos; sabor, características olfativas.	Confirmação de ingredientes, padrões de conteúdo; Reconhecimento de marca, consistência do produto; Condição comercializável, deterioração, prazo de validade; Avaliações de variedade de produtos
Fabricação	Controles de processamento; Uniformidade do produto; Segurança, condições de trabalho.	Características e consistência do produto; Características de aroma e sabor; Alarmes de incêndio, detecção de vazamentos de gases tóxicos.
Médicos e Clínico	Identificação de patógenos; Detecção de patógenos ou doenças; Condições fisiológicas	Seleção de tratamento do paciente, prognósticos; Diagnóstico de doenças, distúrbios metabólicos Estado nutricional, falência de órgãos.

Continua na próxima página

Tabela 3 – continuação da página anterior

Setor industrial	Área de aplicação	Tipos e exemplos de uso específico
Militares	Segurança pessoal e populacional; Segurança civil e militar.	Armas biológicas e químicas; Detecção de materiais explosivos
Farmacêutico	Contaminação, pureza do produto; Variações nas misturas de produtos.	Controle de qualidade da pureza do medicamento; Consistência e uniformidade da formulação.
Regulatório	Proteção do consumidor Proteção Ambiental	Segurança do produto, características perigosas; Testes de contaminação do ar, da água e do solo.
Pesquisa Científica	Botânica, estudos ecológicos Engenharia, propriedades do material Microbiologia, patologia	Quimiotaxonomia, funções do ecossistema; Projeto de máquinas, processos químicos; Identificação de micróbios e metabolitos

Adaptado de: (WILSON; BAIETTO, 2009)

2.3.1 Medicina

Nesta seção serão descrito alguns trabalhos existentes na literatura no âmbito da medicina.

A medicina moderna enfrenta o problema e o desafio de alcançar diagnósticos efetivos da doença por meio de detecção precoce de patogéneses ou condições de doença, a fim de facilitar a aplicação de tratamentos rápidos. A análise química de amostras biológicas humanas, como ar exalado, sangue, urina, suor e pele, é o meio mais habitual de diagnosticar a maioria das condições patogênicas (CHAMBAZ; DEFAYE; MADANI, 1973).

Um ampla gama de compostos voláteis orgânicos são produzidos por espécies microbianas patogênicas, e o potencial diagnóstico do reconhecimento de patógenos através da análise de metabólitos microbianos secundários foi reconhecido e considerado teoricamente possível já na decáda de 1960. No entanto, ainda é muito caro o uso de analisadores químicos de VOC's, como GC ou GC/MS, eles requerem profisionais altamente qualificados e consomem tempo a ponto de impedir o diagnósticos precoces (WILSON; BAIETTO, 2009).

Diferentes compostos voláteis orgânicos (VOC′s) podem ser encontrados no odor do corpo humano e podem ser usados como biomarcadores de várias doenças. A detecção pre-

coce de insuficiência cardíaca (IC) por meio de triagem periódica fornece uma aplicação de tratamento precoce. Por esse motivo, Voss et al. (2012) em seus estudo, desenvolveram um método não invasivo para identificar a IC, aplicando um nariz eletrônico que fornece uma impressão aromática da doença com base em análise de gases voláteis do suor da superfície da pele. Um aplicador especial com o chip sensor foi desenvolvido para ser aplicado diretamente na superfície da pele. 27 pacientes com IC descompensada HF (DHF), 25 pacientes com IC compensada HF(CHF, idade média de 70,72 ± 12,02) e 28 controles (CON) foram incluídos no primeiro estudo piloto. A análise de componentes principais (PCA) foi realizada em combinação com a análise de função discriminante (DA) para discriminar entre os grupos de pacientes. O DHF foi separado do CHF com uma precisão de 87%, enquanto o CON foi discriminado com sucesso do CHF em 85%.

Lorwongtragool et al. (2014) fizeram um nariz eletrônico vestível para análise de odores nas axilas. Nele, é proposto usando um conjunto de sensores químicos de baixo custo integrado a um sistema de comunicação sem fio, o ZigBee. Foi utilizada uma matriz de nanotubos de carbono (CNT's) por sensor de polímetro com base na tecnologia de impressão a jato de tinta. O conjunto de sensores responde a uma variedade de odores complexos, permitindo a classificação de diferentes odores da axila e quantidade de voláteis liberados em função do nível de higiene da pele em diferentes atividades.

Huang et al. (2018) utilizaram um conjunto de sensores químicos e um técnica de aprendizagem de máquima com o objetivo de desenvolver um teste de respiração para detecção de câncer de pulmão. Eles realizaram um estudo prospectivo para registrar casos de câncer de pulmão e controles não tumorais entre 2016 e 2018 e analisaram amostras de ar alveolar usando matrizes de sensores de nanotubos de carbono. No estudo, eles totalizaram 117 casos e 199 controles, dos quais 72 indivíduos foram excluídos devido a câncer em outro local. Os sujeitos inscritos entre 2016 e 2017 foram utilizados para validação interna. O modelo foi validado com indivíduos recrutados em 2018. A precisão do diagnóstico foi avaliada usando os relatórios patológicos como padrão de referência. Na validação externa, as áreas sob a curva de características operacionais do receptor (AUCs) foram de 0,91 (IC 95% = 0,79 ± 1,00) por análise discriminante linear e 0,90 (IC 95% = 0,80 ± 0,99) pela técnica de máquina de vetores de suporte.

2.3.2 Qualidade de alimentos

Nesta seção serão descritos alguns dos trabalhos presentes na literatura que utilizam sensores de gás no âmbito da qualidade de alimentos.

Gómez et al. (2006) utilizaram o nariz eletrônico E PEN2 (10 sensores de óxido de metal diferentes) para estudar a produção volátil de tomates verdes maduros, meio maduros, maduros e super maduros com análise discriminante linear (LDA) e análise de componentes principais (PCA). Os resultados evidenciam que o nariz eletrônico pode distinguir os estados de maturação dos tomates e classificá-los com 100% de confiabilidade em cada

grupo de maturação. Eles também avaliaram o mesmo nariz eletrônico por sua eficiência de monitorar mudanças na produção volátil de tangerina durante diferentes tratamentos de armazenamento. Nessa situação, o prazo de validade do armazenamento foi melhor distinguido usando LDA do que PCA. No entanto, a correlação entre os valores medidos e previstos da qualidade dos frutos mostrou desempenho preditivo razoavelmente bom usando saídas de sensor não analisadas.

Rajamäki et al. (2006) mostraram em seu estudo a aplicabilidade de um nariz eletrônico para controle de qualidade de cortes de frango embalados em atmosfera modificada com diferentes regimes de temperatura. Seus resultados foram comparados com aqueles obtidos por análise microbiológicas, sensoriais e de *headspace*. O nariz foi capaz de diferenciar com clareza os pacotes de frangos de corte com a deterioração de embalagens novas, mais cedo ou ao mesmo tempo em que as alterações sensoriais indicaram uma deterioração significativa. Com a associação ao nariz eletrônico, as contagens de *Enterobacteriaceae* e bacterias produtoras de sulfeto de hidrogênio foram mais consistentes, mostranto que o nariz foi capaz de detectar até mesmo sinais precoces de deteriorização em carne embalada em atmosfera modificada.

Garcia, Aleixandre e Horrillo (2005) usaram sensores semicondutores de óxido de estanho, alguns dos quais foram dopados com catalizadores de metal como Cr e In, para identificação de presuntos estragados durante o processo de cura. Uma boa classificação de dois tipos de presuntos ibéricos (mimados e intectos) foi obtida através dos métodos estatístivos da análise de componentes principais (PCA) e da rede neural probabilística (PNN).

2.3.3 Qualidade de bebidas

Nesta seção serão descritos alguns dos trabalhos presentes na literatura, que utilizam sensores de gás no âmbito da qualidade de bebidas.

Ragazzo-Sanchez et al. (2008) usaram um nariz eletrônico para analisar 21 diferentes bebidas alcoólicas após um procedimento de desidratação e descoolização. Para a classificação foram utilizadas a análise de componentes principais (PCA) e análise fatorial discriminante (DFA) permitiram identificar com clareza as diferenças entre essas várias bebidas independetemente do seu teor de etanol. O PCA mostrou que a discriminação da amostra foi realizada de acordo com o conteúdo em compostos aromáticos. A qualidade da discriminação foi avaliada usando o método de exclusão, que permitiu quantificar a porcentagem de amostras classificadas corretamente para um número determinado de sensores.

Mutalib et al. (2017) utilizaram um nariz eletrônico desenvolvido pela International Islamic University Malaysia (IIUM) para detecção rápida da concentração de etanol (EtOH) em bebidas. Esse nariz eletrônico é amplamente usado na análise de alimentos. Os resultados do estudo mostram que o dispositivo pode ser utilizado para detecção rápida da

concentração de etanol em várias bebidas, como bebidas alcoólicas, bebidas isotônicas, refrigerantes e sucos de frutas de diferentes marcas vendidas na Malásia.

Xiao, Wang e Wang (2017) desenvolveram um nariz eletrônico portátil para identificar vinhos de arroz com diferentes idades. As operações do nariz eletrônico foram controladas por um aplicativo especial para smartphone.O nariz foi composto por 12 sensores de óxidos metálicos (MOS) e os valores das respostas dos sensores foram transmitidos ao smartphone por meio de um módulo de comunicação sem fio. Análise de componentes principais (PCA), a incorporação linear local (LLE) e a análise discriminante linear (LDA) foram aplicadas para a classificação dessas amostras de vinho. O LDA com base nos dados de características da área obtidos no TIOP-AIOP provou ser uma ferramenta poderosa e mostrou os melhores resultados de classificação. A regressão de mínimos quadrados parciais (PLSR) e a máquina de vetores de suporte (SVM) foram aplicadas para as previsões de idades marcadas e SVM (R2 = 0,9942) superando o PLSR.

2.4 CARACTERISTICAS QUÍMICAS DOS VOLÁTEIS DE FRUTAS

O aroma da fruta é frequentemente a característica mais valiosa que determina a qualidade do fruto e a escolha do consumidor, pois o aroma é ,geralmente, o melhor indicador do sabor da fruta. Os frutos produzem e liberam uma ampla variedade de compostos voláteis orgânicos (VOC's) que compõem seus aromas característicos com ésteres, terpenóides, lactonas e derivados de aminoácidos, ácidos graxos e compostos fenólicos sendo de classes dominantes de voláteis orgânicos representados nos aromas das frutas. Embora diferentes frutas compartilhem algumas características aromáticas, cada fruto possui um aroma distinto que depende da combinção específica de VOC's presentes na mistura de aromas (BAIETTO; WILSON, 2015). As sessões seguintes mostram os principais VOC's gerados pelas frutas usadas nos experimentos.

2.4.1 Bioquímica da banana

O aroma da banana tem cerca de 250 VOC's os compostos que são característicos do odor da banana são os ésteres voláteis, como acetato de isoamilo e acetato de isobutilo, que tendem a aumentar durante o amadurecimento (BAIETTO; WILSON, 2015). Chen et al. (2018) estudaram a composição dos compostos orgânicos voláteis nos estágios de amadurecimento da banana para a análise foi usado um sistema de espectrometria de massa de dessorção-gás-cromatografia-massa (TD-GC-MS). A maturidade da banana foi definida em quatro estágios : verde, meio maduro, totalmente maduro e maduro demais. A tabela 4 apresenta a ánalise dos resultados obtidos na pesquisa, mostrando os compostos orgânicos voláteis encontrados em cada estágio de maturidade da banana.

Tabela 4 – A composição típica de compostos orgânicos voláteis (VOC) em cada estágio da maturidade da banana

Compostos orgânicos voláteis.	**Verde**	**Meio maduro**	**Totalmente maduro**	**Maduros demais**
Alcanos				
Isobutano	*	*	*	*
Butano	*	*	*	*
Pentano	*	*	*	*
Ciclopentano	*	*		
2-Pentanona		*	*	*
1,3-Butadieno, 2-metil-			*	*
Ciclobutano, metil-			*	
Biciclo [4.2.0] octa-1,3,5-trieno				*
Álcoois				
Álcool etílico		*	*	*
1-Propanol, 2-metil-		*	*	*
1-Butanol		*	*	
1-butanol, 3-metil-		*	*	*
2-pentanol			*	*
Ésteres				
Ácido fórmico, éster etílico				*
Acetato de etilo		*	*	*
acetato de n-propilo		*	*	*
Ácido acético, éster 2-metilpropílico		*	*	*
Ácido butanóico, éster etílico		*	*	*
Ácido acético, éster butílico		*	*	
2-pentanol, acetato		*	*	*
1-Butanol, 3-metil-, acetato		*	*	*
Ácido butanóico, éster butílico			*	*

Ácido butanóico, éster 1-metilbutílico				
Ácido butanóico, éster 2-metilpropílico		*	*	*
Éster 3-metil-, 2-metilpropílico do ácido butanóico		*	*	*
Éster 3-metil-, butílico do ácido butanóico		*	*	
Éster 3-metil-, 3-metilbutílico do ácido butanóico		*	*	*

Adaptado de: (CHEN et al., 2018)

2.4.2 Bioquímica da maçã

Utilizando uma técnica de enriquecimento dos voláteis do *headspace* em Porapak Q e eluição com hexano, Janzanntti, Fraco e Lancas (2000) detectaram 84 compostos voláteis na maçã Fuji, por cromatografia gasosa de alta resolução, dos quais trinta foram identificados por cromatografia gasosa-espectrometria de massas, aliada aos índices de Kovats. A tabela 2.4.2 mostra os principais fragmentos de massas dos compostos vólateis identificados na maçã Funji.

Compostos Voláteis identificados na Maça Fuji

Compostos Voláteis - Maçã Fuji	
Butanoato de metila	Acetato de isobutila
2-metil butanoato de metila	Hexanal
Butanoato de etila	Propionato de propila
Acetato de butila	2-metil butanoato de etila
Acetato de 2-metil butila	Butanoato de propila
Valerato de etila	Propionato de butila
Acetato de amila	Hexanoato de metila
2-metil butanoato de propila	Butanoato de butila
Hexanoato de etila	Acetato de hexila
Acetato de ciclo hexila	2-metil butanoato de butila
Hexanoato de propila	Propanoato de hexila
2-metil butanoato de amila	Hexanoato de butila

Butanoato de hexila	2-metil butanoato de hexila
Hexanoato de isoamila	Hexanoato de hexila
Octanoato de isoamila	−farneseno

Adaptado de: (JANZANNTTI; FRACO; LANCAS, 2000)

O estudo mostra que a classe química predominante de compostos voláteis da maçã Fuji foi a dos ésteres, compreendendo, butanoato de metila, acetato de isobutila, 2-metil butanoato de metila, butanoato de etila, propionato de propila, acetato de butila, 2-metil butanoato de etila, acetato de 2-metil butila, butanoato de propila, valerato de etila, propionato de butila, acetato de amila, hexanoato de metila, 2-metil butanoato de propila, butanoato de butila, hexanoato de etila, acetato de hexila, acetato de ciclo hexila, 2-metil butanoato de butila, hexanoato de propila, propionato de hexila, 2-metil butanoato de amila, hexanoato de butila, butanoato de hexila, 2-metil butanoato de hexila, hexanoato de isoamila, hexanoato de hexila e octanoato de isoamila. O terpeno µ-farneseno e o aldeído hexanal também foram identificados entre os compostos voláteis.

2.4.3 Bioquímica da uva

O aroma da uva contêm muitos compostos voláteis, incluindo monoterpenos, C_{13} norisoprenóides, álcoois, ésteres e compostos cabonilicos. As variedades da uva podem ser classificadas em duas categorias aromáticas e não aromáticas. Os terpenóides são importantes voláteis em uvas vermelhas e brancas (BAIETTO; WILSON, 2015).

A fim de obter uma técnica apropriada para estudar os compostos voláteis da uva, Palomo, Maroto e Coello (2005) desenvolveram um método HS-SPME para detectar os compostos voláteis da polpa e da casca da uva Moscartel. Dezesseis compostos volátes foram quantificados em polpa e casca. Os terpenos, principalmente linalol, geranional e nerol, têm sido os voláteis presentes na maior concentração, pois esses componentes contribuem em maior parte para o aroma das uvas e dos vinhos. A tabela 6 mostra os compostos orgânicos voláteis identificados na casca e na polpa de uvas Moscatel.

Tabela 6 – Compostos voláteis identificados na casca e polpa de uvas Moscatel

Compostos Voláteis - Casca	**Compostos Voláteis - Polpa**
Hexanal	Hexanal
(E) -2-hexenal	(E) -2-hexenal
1-hexanol	1-hexanol
(E) -2-hexen-1-ol	(Z) -3-hexen-1-ol
furano de óxido de trans- linalol	(E) -3-hexen-1-ol

furano de óxido de cis- linalol	(E) -2-hexen-1-ol
Benzaldeído	furano de óxido de trans- linalol
Linalool	furano de óxido de cis- linalol
-Terpineol	Benzaldeído
pirano de óxido de trans- linalol	Linalool
pirano do óxido de cis- linalol	α-Terpineol
Citronelol	óxido trans- linalol pirano
Nerol	óxido de cis- linalol pirano
Geraniol	Citronelol
Álcool benzílico	Nerol
2-feniletanol	Geraniol
	Álcool benzílico
	Álcool feniletílico

Adaptado de: (PALOMO; MAROTO; COELLO, 2005)

2.4.4 Bioquímica da laranja

O Brasil é um dos maiores produtos mundiais de laranja, e suco de laranja concentrando a maior produção no estado de São Paulo, onde existem fábricas de processamento dos frutos para produção de suco entre outros produtos como pectina, óleos essesciais e farelo de polpa cítrica. Durante esses processos, compostos voláteis são emitidos, Pinheiro (2015) apresentou em seu estudo uma ánalise dos compostos orgânicos voláteis emitidos pela laranja pêra. Quantificou-se por cromatografia gasosa com detector de ionização por cham (DIC). A tabela 7 mostra os principais compostos emitidos pela laranja pêra.

Tabela 7 – Compostos voláteis identificados na laranja pêra

Composto Volátel	**Fórmola Molecular**
Furfural	$C_5H_40O_2$
α-pineno	$C_{10}H_{16}$
β-pineno	$C_{10}H_{16}$
3-careno	$C_{10}H_{16}$
Para-cimeno	$C_{10}H_{14}$
D-limoneno	$C_{10}H_{16}$

Adaptado de: (PINHEIRO, 2015)

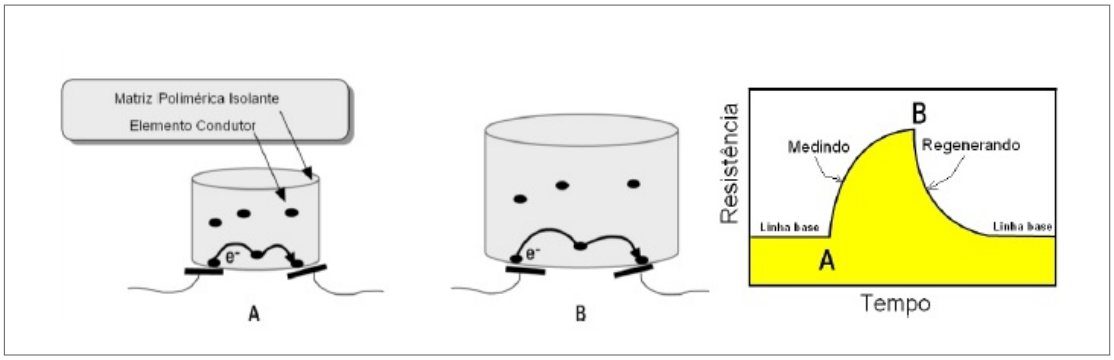
2.5 SENSORES

Narizes Eletrônico utilizam um conjunto de sensores que imitam receptores oftativos e são responsáveis por captar estímulos de compostos químicos voláteis que compõem o odor característico de uma amostra. Assim o conjunto de sensores pode ser considerado o coração e o componente mais importante do nariz eletrônico (WILSON; BAIETTO, 2009). Diferente da maioria dos sensores químicos existentes, projetados para detectar compostos químicos específicos, os sensores em um nariz eletrônico não são específicos a nenhum vapor ou gás. Portanto, usando um grupo de diferentes sensores que reajam a diversos compostos, podem ser identificados vapores, gases e misturas de gases a partir da comparação a padrões de resposta disponíveis no computador. Uma linha de base de um ar limpo é estabelecida e seus desvios registrados como mudanças na resistência dos sensores. Usando um *Software* para análise, as respostas dos diferentes sensores podem ser comparadas e agrupadas de modo a permitir a identificação e quantificação dos diferentes compostos químicos presentes na amostra, a partir dos padrões registrados (REITENBACH, 2016).

2.5.1 Características e funcionamento de um sensor

Os sensores são normalmente condutores que dispõe de propriedades químicas que modificam a resistência quando exposta aos vapores ou gases. O nariz eletrônico é composto por uma matriz de sensores, onde, cada sensor é composto por um filme polimérico carregado com carbono, depositado sobre um par dos eletrodos. Quando o filme é exposto a uma mudança na atmosfera ocorre uma mudança de sinais elétricos, implicando em mudança da resistência, a figura 2 traz uma representação desse tipo de sensor.

Figura 2 – Variação do volume molecular no filme do sensor, devido à absorção de moléculas odorantes e, consequentemente, variação nas resistências elétricas

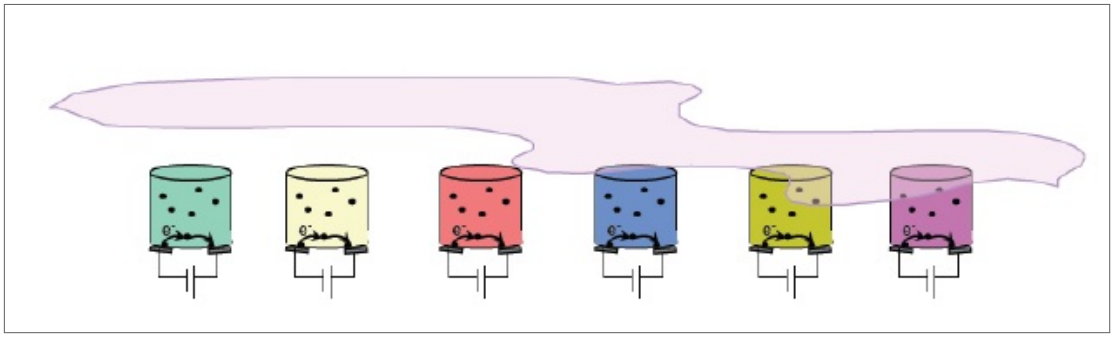


Fonte: (MARTINS, 2011)

Segundo Miller, todos os filmes poliméricos em um grupo de eletrodos (sensores) detêm uma resistência inicial (sua linha base). Caso não ocorra nenhuma mudança na composição do ar, os sensores permanecem na resistência da linha base e a mudança percentual é

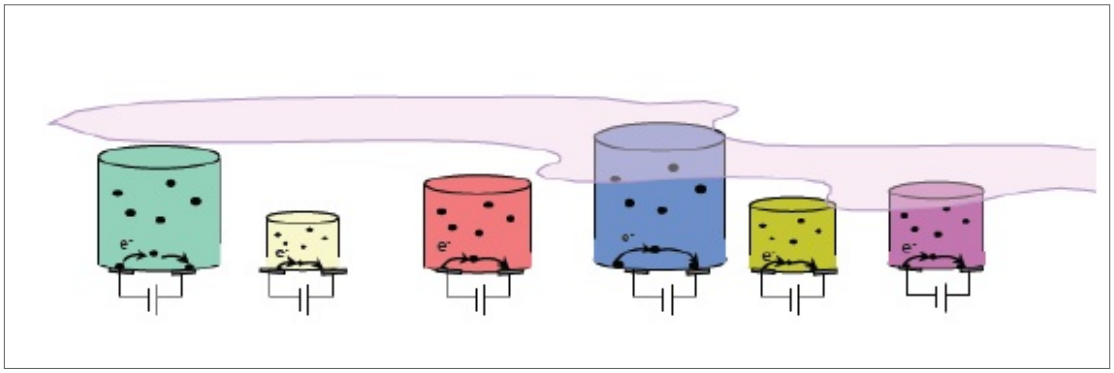
zero (figura 3). Se o nariz ”cheirar” algo, cada polímero terá mudanças em seus tamanhos e, consequentemente, em sua resistência, causando uma impressão padrão da mudança (figura 4). Se outro composto vier a ser identificado no ar, as respostas dos filmes poliméricos dos sensores serão diferentes (figura 5) (MARTINS, 2011).

Figura 3 – Linha de base sem alteração



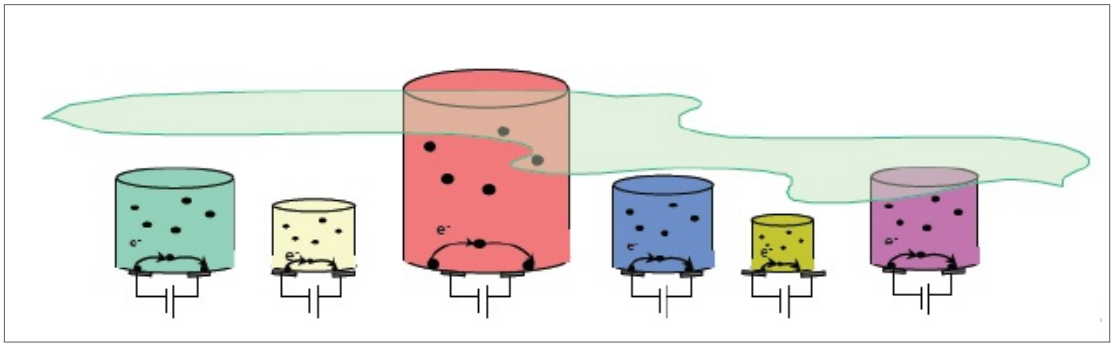
Fonte: (MARTINS, 2011)

Figura 4 – Presença do odor, alteração da linha base



Fonte: (MARTINS, 2011)

Figura 5 – Presença de novo odor, nova alteração da linha de base



Fonte: (MARTINS, 2011)

Cada filme é feito de um polímero diferente, por isso, cada um reage a cada substância de uma maneira ligeiramente diferente. Enquanto as variações na condutividade em um

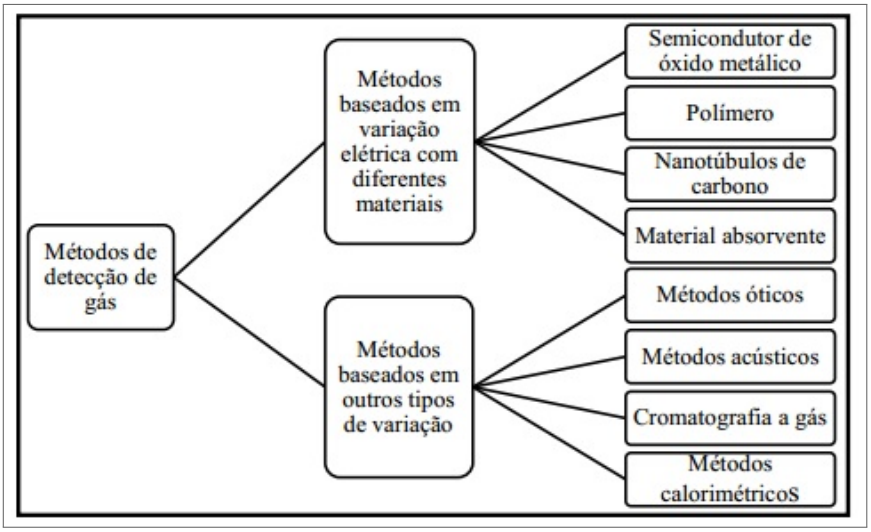
único polímero não seriam suficientes para identificar um analito, as diversas mudanças nos diferentes sensores produzem um padrão bem distinto, permitindo sua identificação (MARTINS, 2011).

Segundo Reitenbach (2016), para avaliar a performace de um sensor de gás, alguns indicadores podem ser avaliados. Dentre eles, merecem destaque:

1. *sensibilidade: o valor mínimo de concentração volumétrica do gás de interesse detectável;*
2. *seletividade: habilidade do sensor de gás em identificar um gás específico em uma mistura de gases;*
3. *tempo de resposta: período entre o tempo em que a concentração de gás alcança um valor específico para gerar um sinal de alerta;*
4. *consumo de energia;*
5. *reversibilidade: o material do sensor deve retornar ao seu estado inicial após a detecção;*
6. *capacidade adsortiva (também afeta a sensibilidade e seletividade);*
7. *custo de fabricação*

Os sensores mais utilizados são os de óxido metálicos, polímeros condutores e piezoselétricos. E o que diferencia os instrumentos no mercado atual é o tipo e o modo de como são produzidos. Uma representação da classificação de métodos de detecção de gases, divididos em duas categorias principais: métodos baseados em variação elétrica com diferente materiais e métodos baseados em outros tipos de variação, pode ser vista na figura 6.

Figura 6 – Classificação dos métodos de detecção de gás.



Fonte: (LIU et al., 2012)

2.5.2 Tipos de sensores

2.5.2.1 Sensores de óxidos metálicos (MOS)

O tipo mais empregado de sensores de gás são os sensores de óxido de metálicos (MOS). Foram utilizados pela primeira vez comercialmente em 1960 como alarmes de gás doméstico no Japão. O uso mais recente desse tipo de sensor inclue aplicativos em diferentes processos industriais. Basicamente, um sensor de óxido metálico é constituído de um tubo de suporte de cerâmica contendo uma espiral de aquecimento, geralmente composta de platina. O revestimento mais utilizado é o dióxido de estanho (SnO_2), dopados com pequenas quantidades de aditivos metálicos catalíticos (também chamados de sensores Taguchi) (WILSON; BAIETTO, 2009). Além do dióxido de estanho outros componentes podem ser usados (por exemplo, TiO_2, ZnO, ZrO_2).

Os sensores de óxidos metálicos demonstram boa sensibilidade aos vapores orgânicos (ppm ou mesmo ppb) para uma grande abrangência de compostos químicos. Apresentam baixa seletividade, isto é, todos os sensores podem responder a um mesmo composto volátil, mas em magnitudes diferentes, arranjos desse tipo de sensor devem ser empregados. Na maioria das vezes o dispositivo também contém um elemento de aquecimento, visto que as taxas de reação são aumentadas em altas temperaturas, esses sensores são aquecidos geralmente entre 175 e 425 °C (MARTINS, 2011).

As reações de oxirredução nos sensores de óxidos metálicos ocorrem de acordo com a interação com o odor, assim elétrons são gerados e uma variação na resistência pode ser mensurada. “O oxigênio do ar é dissolvido em uma rede de semicondutores”, levando a resistência elétrica a um nível de fundo (quando em equilíbrio estável). Durante a medição, as moléculas voláteis são adsorvidas na superfície do semicondutor onde reagem

(redução/oxidação), com espécies de oxigênio dissolvido provocando uma nova resposta dependente da natureza da molécula e do tipo de óxido metálico usado na preparação do sensor, o tempo de resposta dos sensores varia de 10 a 120 segundos (REITENBACH, 2016).

A escolha do material semicondutor é quem determina a sensibilidade e a seletividade dos sensores do tipo MOS. Modificações são induzidas pela dopagem de semicondutores com catalisadores de metal nobre (por exemplo, Pt, Pd, Al e Au), modulando a temperatura operacional ou por introdução de gradientes térmicos/ciclos. Alterar o tamanho da partícula e a espessura do filme de semicondutores tem também sido testado com a mesma finalidade, como também o revestimento de um sensor de gás com uma membrana semipermeável, com espessura variando com aumento da seletividade. Sensores dopados apresentam maior sensibilidade para compostos orgânicos voláteis oxigenados (eg alcoóis, cetonas, etc) do que alifáticos aromáticos ou composto clorados. A sensibilidade dos sensores de SnO_2 para gases, como o bezeno e tolueno, aumentam com a dopagem de Pt e Pd (ALBERT et al., 2000).

Por causa da dependência logarítima da resposta do sensor com a concentração dos voláteis, a perca de sensibilidade surge (para compostos de baixa volatidade de aroma) na presença de alta concentração de espécies detectáveis, como o etanol.

No mercado, os modelos mais recentes são capazes de corrigir o desvio que possibilitam o monitoramento da temperatura e umidade com dispositivo de controle de temperatura, tornando possível lidar com a intoxicação, pois permitem uma melhor regeneração do sensor após cada analíse. Esses sensores têm a vantagem de não saturarem em compostos químicos, como é o caso dos odores, isto é devido a reação de oxidação na superfície (REITENBACH, 2016).

2.5.2.2 Sensores de polímeros condutores (CP)

Os sensores de Polímeros Condutores (CP) atuam com base nas alterações na resistência elétrica provocadas pela adsorção de gases na superfície do sensor. Desde a década de 1980, esses polímeros electroactivos condutores têm atraído muito interesse para o uso com narizes eletrônicos, particulamente porque têm sensibilidades elevadas, os tempos de respostas curtos, são facilmente sintetizados, possuem boas propriedades mecânicas e são sobreturo úteis porque operam à temperatura ambiente (WILSON; BAIETTO, 2009).

Os sensores de polímeros condutores orgânicos, também chamados de polímeros intrinsecamente condutores (ICP), consistem em um substrato, geralmente silício, um par de eletrodos banhados a ouro e um revestimento de polímero orgânico condutor como elemento sensor (SCHALLER; BOSSET; ESCHER, 1998). A sensibilidade dos polímeros condutores aos compostos volatéis orgânicos é medida como mudanças na resistência elétrica. Esses polímeros são geralmente sintetizados por oxidação química ou eletroquímica dos monômeros correspondentes. Polipirrol, polianilina e politofeno são os monômeros de re-

vestimento do sensor mais utilizados. A presença de um sistema pi-electron conjugado que se estende sobre todo o polímero é a característica comum dos materiais poliméricos condutores (WILSON; BAIETTO, 2009).

A alta suscetibilidade à umidade ambiente é um dos pricipais pontos fracos dos polímeros condutores, embora a absorção da água nos filmes poliméricos possa ter uma papel importante no mecanismo de sensibilidade do gás (ARCHER et al., 1989).

Outro desafio desta tecnologia está em melhorar a reprodutibilidade na fabricação de sensores de polímeros condutores. No entanto, os sensores CP apresentam respostas lineares e seletividade maior em comparação aos sensores MOS. Em comparação com os sensores MOS, nenhum efeito a envenenamento com compostos contento enxofre ou ácidos fracos tem sisdo observado. Eles mostram respostas mais rápidas e as recuperações da linha de base, e não precisam de altas temperaturas de operação (MARTINS, 2011).

Harper (2001) mostrou em um estudo comparativo de sensores que um instrumento Alpha MOS foi o menos sensitivo ao vapor de água em comparação com dois sensores baseados em polímeros condutores, um da empresa AromaScan e o outro da empresa Neotronics. Considerando que o sistema da AromaScan era de difícil operação, devido ao complexo controle de umidade relativa das amostras e do gás de arraste, o sistema da Neutronics, embora de mais facil operação, mostrou uma sensibilidade reduzida.

2.5.2.3 Sensores colorimétricos (CSA)

Os componentes de arrajos de sensores colorimétricos (CSA) são inerentes à nano escala. Tais arranjos atuam como nariz óptico-eletrônicos, usando um substâncias que mudam de cor, cujas mudanças de cores são baseadas numa ampla gama de interações intermoleculares. Os arranjos de sensores colorimétricos foram desenvolvidos em anos recentes e apresenta imagens digitalizadas antes e após exposição, as diferenças resultantes fornecem uma impressão digital para qualquer composto orgânico volátil (VOC) ou mistura de odorantes.

Quando comparados as demais tecnologias eletrônicas de narizes eletrônicos, o resultado é um enorme aumento no poder discriminatório dos compostos odorantes. Eles têm sido utilizados na detecção de importantes analitos biológico, como por exemplo aminas, ácidos carboxilicos e tióis, com sensibilidades elevadas (ppb). Esse tipo de sensor não responde a mudanças de umidade devido à hidrofobicidade das tinturas e membranas (REITENBACH, 2016).

2.5.2.4 Sensores de ondas acústicas

Os sensores de ondas acústicas usam uma onda mecânica (acústica) como mecanismo de detecção. Como a onda acústica se propaga através ou sobre a superfície do material de revestimento do sensor, quaisquer alteração nas características do caminho de propagação, devido à sorção de compostos voláteis orgânicos (VOCs), afetam e/ou amplitude da onda.

São constituidos de um substrato piezoelétrico, geralmente quartzo (SiO_2), niobato de lítio ($LiNbO_3$), tantalita de lítio ($LiTaO_3$) ou óxido de zinco, dopado com um material de sorção adequado. Estes dispositivos são pequenos, baratos e sensíveis a praticamente todos os gases, particulamente à massa de compostos adsorvidos cujo peso molecular é um dos parâmetros fundamentais para identificação do analitico (WILSON; BAIETTO, 2009).

Embora os dispositivos de onda acústicas estivessem em uso comercial há mais de 60 anos, principalmente no setor de telecomunicação, o primeiro relatório de um sensor de ondas acústicas foi em 1979 (WOHLTJEN; DESSY, 1979).

Os dois principais sensores de ondas acústicas são os sensores de SAW e BAW. A diferença entre esses sensores é o modo de oscilação, na superfície e no volume, respectivamente, a determinação da gama disponível de oscilação: SAW opera em 5-10 MHz, enquanto o BAW em 5-30 MHz. Os dispositivos de SAW são mais sensíveis, mais instáveis e exigem um controle de alta tecnologia set-up (MARTINS, 2011).

A sensibilidade do sensor de ondas acústicas aos VOC's é determinada pelos tipos de revestimento absorventes usados nos sensores. Materiais diferentes têm sido utilizados para este propósito dentre eles: filmes monocamadas, moléculas anexadas à superfície e camadas de uma ampla variedade de filmes poliméricos (WILSON; BAIETTO, 2009).

2.5.2.5 Sensores com nanomateriais

Pesquisas têm sido desenvolvidas com base em aplicações da nanotecnologia para uso de sensores oftométricos. Esses nano materiais podem aprimorar a condutividade dos sensores e tornar mais simples a sua fabricação, são bastante utilizados para a detecção de gases perigosos em pequenas concentração na indústria aeroespacial (REITENBACH, 2016).

Os Nanotubos de Carbono (CNTs), nos últimos anos, se tornaram alvo de grandes pesquisas, especialmente no desenvolvimento e avaliação de sensores e biossensores para diversas aplicações em eletroanálitica. Isso ocorreu, particularmente, devido as suas propriedades físicas e químicas notáveis, como elevada área superficial, alta resistência mecânica, excelente condutividade elétrica e térmica, facilidade em promover a transferência eletrônica e diminuição do potencial de trabalho (eletrocatálise), além da possibilidade de modificações para se ancorar nanoparticulas metálicas/óxido metálico sobre sua superfície (FILHO, 2014).

Esses equipamentos detêm propriedades elétricas e são extremamente sensíveis a pequenas quantidades de gases como álcool, amônia, dióxido de carbono e óxidos de nitrogênio a temperatura ambiente, ao passo que outros materiais como óxidos metálicos necessitam ser aquecidos para operar normalmente. A alta sensibilidade suprime a necessidade de tecnologias como pré-concentração e contribuem com a vantagem do peso e de configuração simplificada (LIU et al., 2012).

O Grafeno (GF) atualmente é considerado um dos materiais mais finos do mundo, o qual apresenta uma camada única de átomos de carbono dispostos a partir de ligações hexagonais em uma estrutura bidimensional (2D). A ampliação dos estudos do GF começou após publicação de dois artigos científicos publicados por NOVOSELOV e colaboradores nos anos de 2004 e 2005. Após estas publicações, o GF passou a ser amplamente explorado em uma vasta gama de aplicações, dentre elas, o armazenamento e geração de energia, fabricação de sensores e em nano-disposotivos elétricos, devido principalmente às suas propriedade eletrônicas (alta mobilidade de elétrons e boa condutividade) (FILHO, 2014).

As principais limitações do uso do GF são da natureza experimental. O primeiro problema é como conectar eletricamente uma única folha de GF, e em segundo lugar como reduzir ou previnir a agregação de folhas de GF à sua configuração de menor energia (grafite), devido as fortes interações entre as folhas de GF. Os problemas da agregação do GF podem limitar significativamente o desempenho dos dispositivos eletroquímicos baseados neste material. Por exemplo, o uso de derivados do GF em baterias e supercapacitadores apresenta uma redução na área ativa e diminuição no espaçamento entre as folhas de GF individuais à medida que o número de camadas aumenta, reduzindo as capacidades de intercalação de Li e, consequentemente, o desempenho do dispositivo. Na atualidade uma área de pesquisa promissora e em amplo desenvolvimento é a procura de novos métodos de preparação do GF, objetivando-se obter folhas de GF de alta qualidade (livre de impurezas, grande área superficial e volume) e boa repetibilidade (FILHO, 2014).

2.6 MÉTODOS DE RECONHECIMENTO DE PADRÕES

Com os dados numéricos de determinado domínio, os sinais captados pelos sensores do nariz eletrônico podem ser processados por sistemas de reconhecimento e classificação. As seções 2.6.1, 2.6.2, 2.6.3, 2.6.4.1, 2.6.5 e 2.6.6 mostram os métodos de aprendizagem de máquina utilizados nesse trabalho para reconhecer e classificar as frutas.

2.6.1 K-vizinhos mais próximos (KNN)

O método KNN(*K-Nearest Neighbours*) é bastante utilizado por ter um conceito simples e ser considerado de fácil implementação. Além de ter resultados satisfatórios, esse classificador é baseado em aprendizado por analogia, cada registro novo a ser inserido na base para classificação é comparado com os registros que já estão na base de treinamento que são semelhantes a esse novo registro (VOSS, 2019).

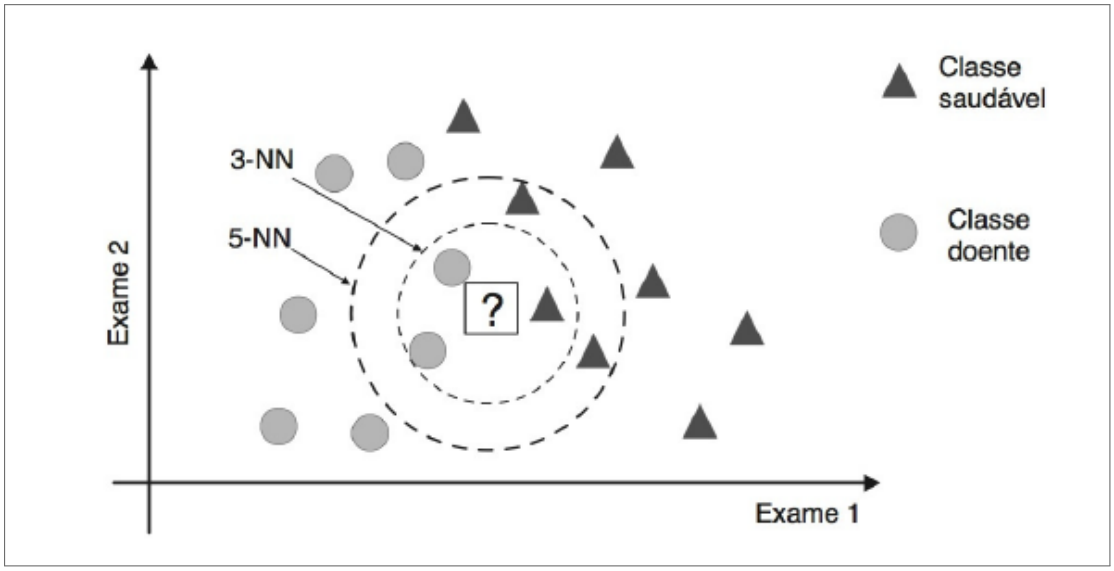
Para cada um dos registro na base de dados, existem n atributos para representá-lo, podendo assim, cada registro ser representado por um ponto no espaço n-dimensional. Para cada novo registro, ainda desconhecido, o classificador pesquisa no espaço n-dimensional as tuplas de treinamento(k) que estão mais próximas desse novo registro, sendo assim, os k registros são denominados "k-vizinhos mais próximos"do novo registro desconhecido. A

vizinhança (proximidade) entre os pontos no espaço é definida por uma distância métrica, um exemplo utilizado é a distância euclidiana. A distância euclidiana de dois pontos X_1 e X_2, em que $X_1 = (X_1 1, X_1 2,...,X_1 n)$ e $X_2 = (X_2 1, X_2 2,...,X_2 n)$ é definida pela equação 2.1 (VOSS, 2019).

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n}(X_l i - X_2 i)^2} \tag{2.1}$$

A figura 7 representa um exemplo ilustrativo simples do uso do algoritmo dos vizinhos mais próximos para diferentes valores de k. Nesse exemplo, considera-se um conjunto de dados cujos objetos são indivíduos que podem ser classificados em saudáveis ou doentes, e o espaço de entrada é definido por dois atributos que representam o resultado de dois exames. O ponto representado por "?" é o ponto de teste, ou seja, o indivíduo a ser classificado os outros pontos são objetos em que a classe é conhecida: saúdavel, representada por um triângulo, ou doente, representada por um círculo. Pra $k = 3$, o objetivo do teste seria classificado, como pertencendo à classe doente, enquanto $k = 5$ o objeto de teste seria classificado como pertencendo à classe saudável (FACELI et al., 2011).

Figura 7 – Impacto do valor de k no algoritmo K-NN



Fonte: (FACELI et al., 2011)

A escolha do valor mais apropriado de k para um problema de decisão específico pode não ser trivial. A definição do valor de k é feita pelo usuário. Frequentemente, o valor de k é pequeno e ímpar: k= 3,5.... Em problemas de classificação, não é usual utilizar $k = 2$ ou valores pares, para evitar empates. Duas estratégias referidas na literatura para estimar k, consistem em: estimar k por validação cruzada ou associar um peso à contribuição de cada vizinho (FACELI et al., 2011).

O KNN é um algoritmo considerado preguiçoso por ser baseado em memória, seu processo de aprendizado consiste apenas em memorizar objetos. Outro problema do k-NN

está relacionado à dimensionalidade dos exemplos. O espaço definido pelos atributos de um problema cresce exponencialmente com o número de atributos, ou seja, o número de atributos define o número de dimensões do espaço. Em problemas de alta dimensionalidade, o ponto que está mais perto de outro pode estar muito distante (FACELI et al., 2011).

2.6.2 Máquinas de vetor de suporte (SVM)

Segundo Faceli et al. (2011), as máquinas de vetores de suporte (*support vector machines* - SVMs) vêm recebendo bastante atenção da comunidade que utiliza as técnicas de aprendizagem de máquina. Seus resultados são comparáveis e muitas vezes superiores aos obtidos por outras técnicas populares de aprendizado, como por exemplo o uso de redes neurais artificiais. As SVMs são baseadas na teoria do aprendizado estatístico, desenvolvida por Vapnik(1995), e estabelece princípios para obtenção de classificadores com boa capacidade de generalização.

Sabe-se que um classificador nada mais é do que uma função que, para cada entrada x_i (padrão apresentado) gera uma saída y_i (em um problema de classificação, a classe à qual o padrão pertence). A teoria do aprendizado estatístico estabelece uma série de restrições matemáticas que guiam a escolha do melhor classificador f, dado o conjunto F de todos os classificadores gerados por um algoritmo de aprendizagem de máquinas (SANTOS, 2014)

A escolha desse classificador é fundamentada na minimização do risco esperado, que mede a capacidade de generalização de f. Um dos limites estabelecidos pela teoria de aprendizado estatístico, relaciona o risco esperado de uma função com o seu risco empírico (mede o desempenho do classificador nos dados de treinamento) e um termo de capacidade.

$$R(f) \leq R_{emp}(f) + \sqrt{\frac{h(ln(\frac{2n}{h}) + 1) - ln(\frac{\theta}{4})}{n}} \tag{2.2}$$

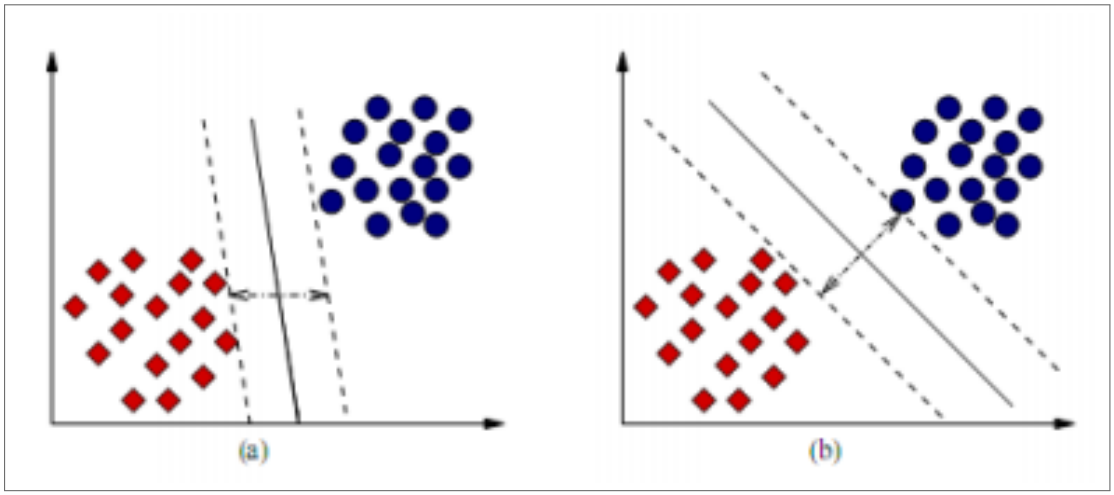
onde a quantidade de exemplos do conjunto de treinamento é dado por n, $\theta\epsilon[0, 1]$ e h representa a dimensão. Vapnik-Charvonenki mede a capacidade do conjunto de funções F de forma que quanto maior seu valor, mais complexas são as funções de classificação que podem ser induzidas a partir de F. Deste modo, pode-se afirmar que a minimização do risco esperado depende de um f que minimize o risco empírico e que pertença a uma classe de funções com baixo h (LORENA; CARVALHO, 2007).

Apesar de útil para iniciar o entendimento sobre SVMs. a equação mostrada possui algumas limitações práticas, como a difícil determinação de h. Deste modo, uma nova equação foi inserida, empregando o conceito de margem:

$$R(f) \leq R_p(f) + \sqrt{\frac{c}{n}(\frac{R^2}{p^2}log^2(\frac{n}{p}) + log(\frac{1}{\theta})} \tag{2.3}$$

onde R_p representa o risco marginal (erro no conjunto de treinamento), c é uma constante, e p é uma margem, a qual tem relação com a distância dos exemplos à fronteira de decisão induzida. Um maior valor de margem diminui o termo de capacidade, mas pode aumentar o erro marginal, a medida que um valor de margem baixo aumenta o termo de capacidade e diminui o erro marginal. Logo, deve-se buscar um equilíbrio entre a maximização da margem e a obtenção de um erro marginal baixo. Assim, o objetivo de um classificador linear é descobrir um hiperplano que tenha margem elevada e cometa poucos erros marginais (hiperplano ótimo) (LORENA; CARVALHO, 2007). A figura 8 apresenta dois possíveis hiperplanos: (a) um hiperplano com margem pequena e (b) um hiperplano com margem elevadas (hiperplano ótimo).

Figura 8 – Hiperplanos (a) com margem pequena e (b) com margem elevada



Fonte: (SANTOS, 2014)

Um hiperplano não é suficiente para separar as classes no tratamento de dados não linearmente separáveis. As SVMs lidam com esse tipo de problema de forma semelhante a Rede Neural RBF (Radial Basis Function Neural Network): baseando-se no teorema de Cover, mapeiam (mapeamento denotado por Φ) o conjunto de treinamento de seu espaço original para um novo espaço, o espaço de características, com dimensão maior e linearmente separável utilizando uma transformação não-linear,de forma que os dados possam ser resolvidos por uma SVM linear, a qual encontra o hiperplano com maior margem de separação, garantindo uma boa generalização (SANTOS, 2014).

As funções de *kernel* são responsáveis por realizar a transformação das amostras do espaço original para o novo espaço. Desta maneira, uma função de *kernel* K recebe dois pontos x_i e x_j do espaço de entradas e estima o produto escalar desses dados no espaço de características : $K(x_i, x_j) = \Phi(x_i).\Phi(x_j)$. Há vários tipos de *kernel* com seus respectivos parâmetros que devem ser determinados pelo usuário. Na prática os mais utilizados são: Polinomial, o Gaussiano (RBF) e o Sigmóide (SANTOS, 2014). A tabela 8 apresenta os principais tipos de *kernel* e suas particularidades .

Tabela 8 – Funções de *kernel*

Tipo de Kernel	**Função** $k(x_i, x_j)$	**Parâmetros**
Polinomial	$(\delta(x_i.x_j) + k)^d$	δ, k e d
Gaussiano	$exp(-\sigma \parallel x_i.x_j \parallel^2)$	σ
Sigmoidal	$tanh(\sigma(x_i, x_j) + k)$	δ e k

Fonte: (SANTOS, 2014)

2.6.3 Florestas aleatóricas (RF)

O *Random Forest*(RF) é considerado um método de classificação do tipo *ensemble*, esse tipo combina uma série de r modelos ou classificadores, M_1, M_2,..., M_r, tendo como objetivo obter uma classificação composta melhorada, M^*. Métodos *ensemble* são tendenciosnados a serem mais precisos do que classificadores individuais. Considerando um método *ensemble*, os classificadores base podem cometer erros, mas, o conjunto só irá classificar de forma errada uma instância ou registro se mais da metade dos classificadores base também estiverem errados (VOSS, 2019).

O RF é um conjunto de classificadores que são árvores de decisão, onde, cada árvore depende dos valores de um vetor aleatório de atributos em cada nó para determinar a divisão de cada árvore de decisão individual. Esse vetor aleatório é amostrado de forma independente e com a mesma distribuição para todas as árvores na floresta(conjunto de árvores que formam o *random forest*). Durante o processo de classificação, cada árvore faz seu voto e a classe mais popular é retornada (VOSS, 2019).

O *overfitting*, termo utilizado para definir quando um determinado modelo estatístico se ajusta muito bem ao conjunto de treinamento e se torna ineficaz para prever novos resultados, não é um problema para as florestas aleatórias. O erro de generalização de uma floresta converge e a precisão depende do poder de classificação dos classificadores individuais e uma medida de dependência entre eles. Como esse método considera menos atributos para cada divisão, ele se torna eficiente em bancos de dados grandes (VOSS, 2019).

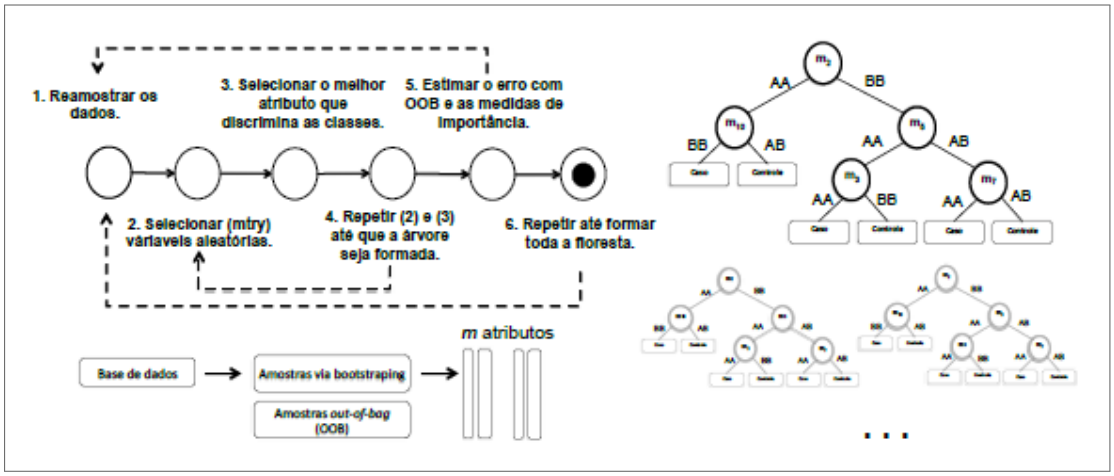
Construção do Modelo

Para cada árvore de decisão do modelo, um conjunto de treinamento diferente é gerado. Sendo estabelecidos distintamente dois subconjuntos X_G^1 e X_G^2. Definimos com X_G^1 o subconjunto das amostras utilizadas para construção das árvores, enquanto X_G^2 é composto das amostras para estimação do erro, chamados tecnicamente de amostras *out-of-bag*. As árvores são construídas por particionamento recursivo, utilizando observações do conjunto de treinamento, via *booststraping* com reposição dos dados. Quando os nós filhos estão homogênios o particionamento recursivo é finalizado, ou seja, quando contêm apenas classes de um mesmo grupo. Para cada divisão de um nodo da árvore é selecionado um conjunto aleatório de m preditores, considerando apenas o conjunto de treinamento.

Segundo o critério GI, cada variável é avaliada, considerando a habilidade de separação das amostras. Ao nodo é atribuído o preditor com menor GI, iniciando do nodo raiz. Ao final da construção das árvores não há podas (ARAUJO, 2013).

Dois parâmetros são relevantes para a formação do modelo RF, são eles: *ntree* e *mtry*. O *ntree* é o parâmetro que define a quantidade de árvores de decisão a ser formada, sendo relevante pois define o custo computacional do modelo, quanto maior o número de árvores, maior o custo de execução. Para cada árvore e a cada divisão dos nodos, a RF escolhe aleatóriamente *mtry* variáveis, que corresponde a um subconjunto das váriaveis do conjunto original de dados. Com base no índice GI, é escolhido o melhor preditor do subconjunto, que separa as duas classes de acordo com a homogeneidade. Este parâmetro é muito importante para controlar o *bias-variance trade-off*, de modo que as árvores sejam produzidas com baixa correlação. O modelo de Florestas Aleatórias finaliza o seu processo de construção após formação de todas as árvores. Para agrupar os votos e tomar uma decisão final, o modelo considera o voto majoritário sem pesos, isto é, são supervisionados os votos e a decisão final é aquela que recebe o maior número de votos (ARAUJO, 2013). A Figura 32 mostra os passos para a contrução do modelo RF.

Figura 9 – Visão Geral do Método Árvores Aleatórias



Fonte: (ARAUJO, 2013)

2.6.4 Redes neurais artificiais (RNA)

A redes neurais artificiais tem como base o modelo do cérebro humano. No cotidiano, o ser humano realiza diferentes eventos ao mesmo tempo que requerem a devida atenção e processa informações variadas de forma a tomar ações convenientes. A complexidade de ações simples como, por exemplo, caminhar, é evidenciada pela dificuldade encontrada em ensinar robôs a realizá-la (FACELI et al., 2011).

Os projetos iniciais em RNA buscavam compreender o cérebro e utilizar o conhecimento obtido para desenvolver sistemas de aprendizado biologicamente plausíveis, logo, as RNAs são baseadas em modelos abstratos de como se sabe que o cérebro e os neurônios

funcionam. Elas são inspiradas em estudos sobre o sistema nervoso humano (FACELI et al., 2011).

O cérebro humano consiste, basicamente, em neurônios que são ligados a outros neurônios por fios que são chamados axônios. Eles, por sua vez, são responsáveis por levar impulsos nervosos de um neurônio para outro, sempre que os neurônios são estimulados. Um neurônio é ligado em axônios de outros neurônios por dentritos, que são extensões do corpo da célula. O ponto de encontro entre o dentrito e um axônio é chamado de sinapse. Segundo neurologistas, o cérebro humano aprende mudando a força da conexão sináptica entre os neurônios em estimulação repetida pelo mesmo impulso (VOSS, 2019).

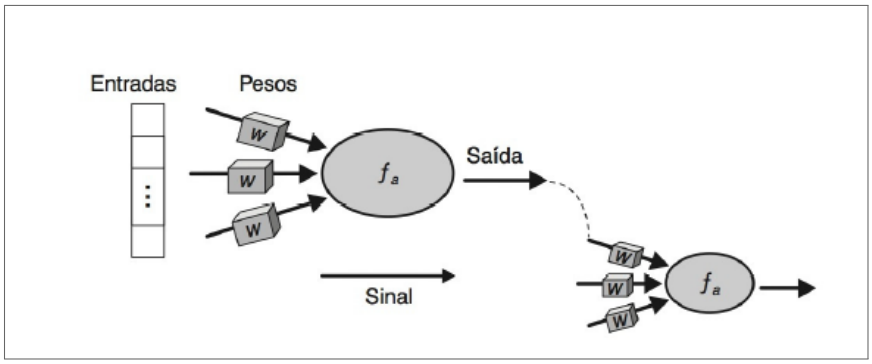
As RNAs são sistemas computacionais distribuídos compostos de unidades de processamento simples, densamente interconectadas. Essas unidades são conhecidas como neurônios artificiais, que computam funções matemáticas. Essas unidades estão organizadas em uma ou mais camadas interligadas por um grande número de conexões, geralmente unidirecionais (que indica uma única direção). Na maioria das arquiteturas, essas conexões, que simulam as sinapses biológicas (contato entre a terminação de um axônio e o dendrito de outro neurônio), possuem pesos associados, que ponderam a entrada recebida por cada neurônio da rede. Esses pesos, dependendo do comportamento da conexão, podem assumir valores positivos ou negativos, respectivamente. Os pesos têm seus valores ajustados em um processo de aprendizado e codificam o conhecimento adquirido pela rede. Portanto, dois aspetos básicos podem caracterizar uma RNA: arquitetura e aprendizado. A arquitetura está relacionada ao tipo e número de unidades de processamento e a forma como os neurônios estão conectados, o aprendizado diz respeito às regras utilizadas para ajuste dos pesos da rede e que informação é utilizada pelas regras (FACELI et al., 2011).

Arquitetura

Segundo Faceli et al. (2011) o neurônio é a unidade de processamento fundamental de uma RNA, onde cada unidade de processamento desempenha um papel simples. Cada terminal de entrada, dentrito, recebe um valor. Esses valores são ponderados e combinado por uma função matemática f_a, onde, várias e diferentes funções podem ser utilizadas nesse processo. A figura 10 representa um objeto x com d atributos, representado pelo vetor $x = [x_1, x_2, x_3, ...x_d]^t$ e um neurônio com d terminais de entrada com os seguintes pesos, representados na forma vetorial: $W = [w_1, w_2, w_3, ...w_d]$. A entrada total recebida do neurônio u é definida pela equação 2.4.

$$u = \sum_{d}^{j=1} x_j wj \tag{2.4}$$

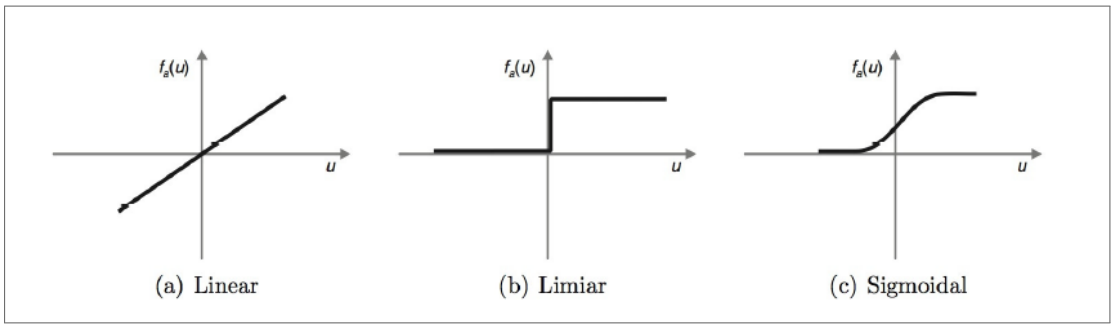
Figura 10 – Neurônio Artificial



Fonte: (FACELI et al., 2011)

A definição para a saída de um neurônio é dada pela função de ativação. Na literatura, várias funções têm sido propostas, entre elas as funções linear, limiar, sigmoidal. Na função linear, o seu uso implica em retornar como saída o valor de u. Na função limiar, o valor do limiar define quando o resultado da função limiar será igual a 1 ou 0, pode-se empregar o valor -1. Se a soma das entradas recebidas ultrapassar o limiar recebido, o neurônio se torna ativo (saída + 1). Na função sigmoidal, podem ser utilizadas diferentes inclinações, ela representa uma aproximação continua e diferenciada da função limiar. (FACELI et al., 2011). A figura 11 mostra exemplos das funções de ativação.

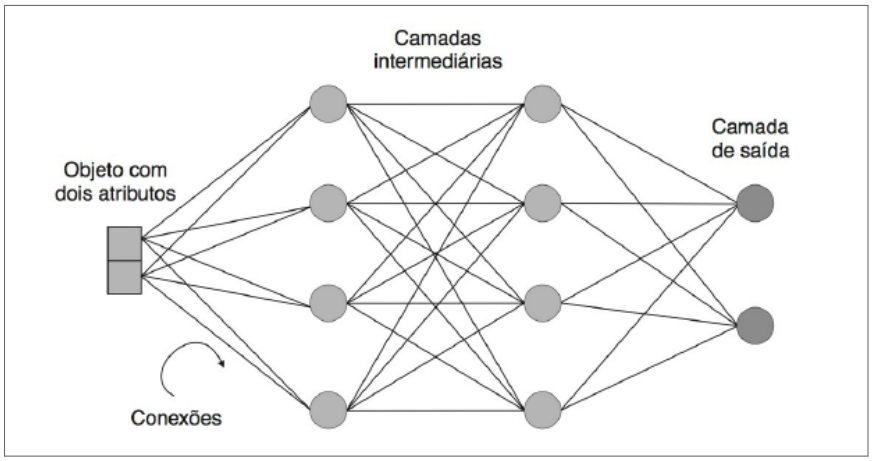
Figura 11 – Exemplos de Função de Ativação



Fonte: (FACELI et al., 2011)

Os neurônios de uma RNA podem estar dipostos em uma ou mais camadas, onde, um neurônio com mais de uma camada pode receber em seus terminais de entrada as saídas dos neurônios da camada interior e/ou enviar seu valor de saída para neurônios da camada seguinte. Uma rede com mais de uma camada de neurônios recebe o nome de rede multicamadas. A figura 12 recebe como entrada dois atributos e gera dois valores em sua saída.

Figura 12 – Exemplo de RNA multicamadas típica



Fonte: (FACELI et al., 2011)

Aprendizado

Como dito anteriormente, vários algoritmos têm sido propostos para ajuste dos parâmetros de um RNA. O ajuste de parâmetro pode ser entendido principalmente como a definição dos valores dos pesos associados às conexões da rede que fazem com que o modelo obtenha melhor desempenho, esse desempenho é medido pela acurácia preditiva. Eses algoritmos são formados por um conjunto de regras bem definidas que especificam quando e como deve ser alterado o valor de cada peso (FACELI et al., 2011). Podem ser divididos em quatro grupos:

1. Correção de erro: utilizado em aprendizado supervisionado, procura ajustar os pesos da RNA de forma a reduzir os erros cometidos pela rede.

2. Hebbiano: Baseados na regra Hebb, são usados em aprendizado não supervisionado, que diz que, se dois neurônios estão simultaneamente ativos, a conexão entre eles deve ser reforçada.

3. Competitivo: Promovem uma competição entre os neurônios para definir qual ou quais devem ter seus pesos ajustados. Os que vencem a competição em geral são os que respondem mais fortemente ao objeto apresentado aos seus terminais de entrada.

4. Termodinâmico : Algoritmos estocásticos baseados em princípios observados na metalurgia.

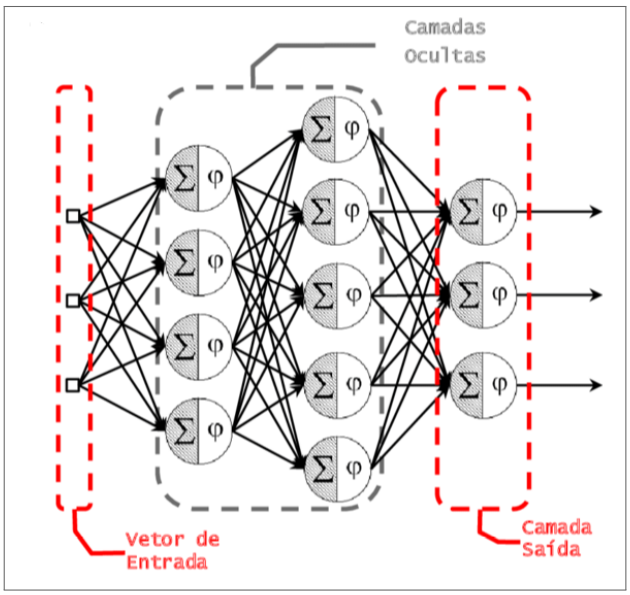
A seção 2.6.4.1 mostra um exemplo de RNA que é bastante utilizada nos projetos que envolvem aprendizagem de máquina.

2.6.4.1 Multilayer perceptron (MLP)

As RNAs são utilizadas para resolver problemas não linearmente separáveis, com isso, a alternativa mais utilizada é adicionar uma ou mais camadas intermediárias. Com uma camada, é possivel resolver qualquer função contínua e, com duas camadas intermediárias, é possível realizar a aproximação de qualquer função (FACELI et al., 2011).

As redes do tipo perceptron multicamadas (MLP - *multilayer perceptron*) podem ter uma ou mais camadas intermediárias de neurônios e uma camada de saída. As MLPs são, de modo geral, completamente conectadas, ou seja, os neurônios de uma camada l estão conectados a todos os neurônios da camada l + 1. Caso cada camada l seja a primeira camada intermediária, cada um de seus neurônios estará conectado a todos os atributos de entrada x^j, j = 1..., d, para um objeto de entrada x. Nas camadas intermediárias, as redes multicamadas utilizam funções de ativação não lineares, um exemplo de uma dessas funções é a sigmoidal. Essas redes realizam em cada neurônio uma função específica (FACELI et al., 2011). A figura 13 mostra um exemplo típico de MLP.

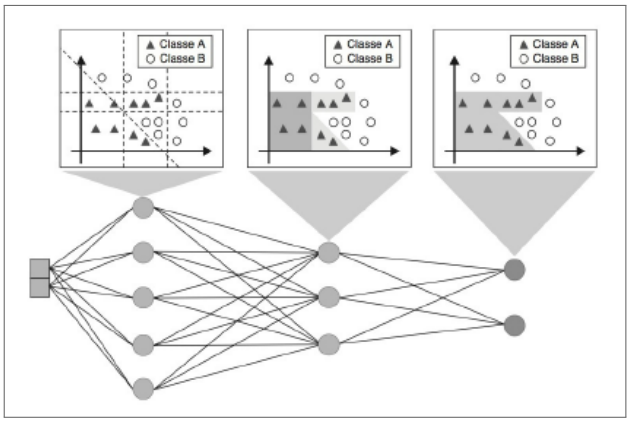
Figura 13 – Exemplo MLP



Fonte: (OLIVEIRA, 2013)

O processo se torna cada vez mais complexo quando o processamento avança de uma camada intermediária para a camada seguinte. Na primeira camada, cada neurônio aprende uma função que define um hiperplano, o qual divide o espaço de entrada em duas partes e cada neurônio da camada seguinte combina um grupo de hiperplanos definidos pelos neurônios da camada anterior, formando regiões convexas (FACELI et al., 2011). A figura 14 mostra um exemplo do papel de cada neurônio.

Figura 14 – Exemplo do papel dos neurônios das diferentes camadas de uma MLP



Fonte:(FACELI et al., 2011)

No exemplo da figura 14, pode-se observar que para cada neurônio da camada de saída da MLP existe uma associação a uma das classes presentes no conjunto de dados. Um vetor y = $[y_1, y_2, ..., y_k]^t$ pode representar os valores gerados pelos neurônios de saída para um dado objeto de entrada, sendo que k é o número de neurônios da camada de saída e o número de classes do problema. Um possível erro de classificação da rede se dá pela comparação entre o vetor de saída dos neurônios da camada de saída e o vetor de valores desejados para essa saída. Esse erro de classificação ocorre quando o neurônio de uma outra classe é o que produz o valor de saída mais elevado. A rede classifica um objeto de forma correta quando o valor de saída mais elevado produzido pela rede é aquele gerado pelo neurônio de saída que corresponde à classe correta do objeto. A rede não tem condições de prever a classe do objeto quando nenhum neurônio produz um valor elevado ou o valor elevado é produzido por mais de um neurônio (FACELI et al., 2011).

2.6.5 Xgboost

O XGBoost é um algoritmo e uma variante eficiente e escalonável da *Gradient Boosting Machine* (GBM). Ele tem sido utilizado em várias competições de aprendizado de máquina com resultados vencedores. Possui características como a facilidade de uso, facilidade de paralelização e precisão preditiva impressionantes, além de, possuir alta capacidade de lidar com espaço de recursos altamente diversificado e complexo, especialmente em situações onde a distribuição de classes é altamente desequilibrada (MUSTAPHA; SAEED, 2016).

Segundo o tutorial do algoritmo XGBoost, o termo significa “Extreme Gradient Boosting”, onde o termo “Gradient Boosting” tem origem no artigo Greedy Function Approximation: A Gradient Boosting Machine , de Friedman. Ele é utilizado para problemas de aprendizado supervisionado. O XGBoost não pode lidar com recursos categóricos por si só, mas apenas aceita valores numéricos semelhantes ao Random Forest, seção 2.6.3. O Xgboost é um *ensemble* de k árvores de classificação e regressão (CART - Classification

and Regression Trees). Ele possui uma grande variedade de parâmetros ajustáveis para um desempenho ideal (MUSTAPHA; SAEED, 2016).

Esse algoritmo de aprendizado de máquina trabalha em vetores numéricos, existe a conversão de variáveis categóricas no vetor numérico. Basicamente, o XGBoost depende da atribuição de pesos para a observação. Ele possui vários recursos: processamento rápido, validação cruzada, poda de árvores, altamente flexível. O ajuste excessivo é controlado pelo modelo XGBoost para o melhor desempenho, o que torna este modelo muito bom (BANSAL; KAUR, 2018).

2.6.6 Lightgbm

O LightGBM(LGBM) é um algoritmo de classificação e pode trabalhar com recursos categóricos. O LGBM usa um algoritmo especial para encontrar o valor dividido dos recursos categóricos (MEDIUM, 2017).

O tamanho dos dados está aumentando dia a dia e está ficando difícil para os algoritmos tradicionais de ciência de dados fornecerem resultados mais rápidos. O Light GBM é prefixado como 'Light' devido à sua alta velocidade. Ele pode lidar com o grandes bases de tamanho variados e utiliza menos memória para ser executado . Outro motivo pelo qual o Light GBM é popular é porque ele se concentra na precisão dos resultados . O LGBM também suporta o aprendizado da GPU e, portanto, os cientistas de dados estão usando amplamente o LGBM para o desenvolvimento de aplicativos de ciência de dados (MEDIUM, 2017).

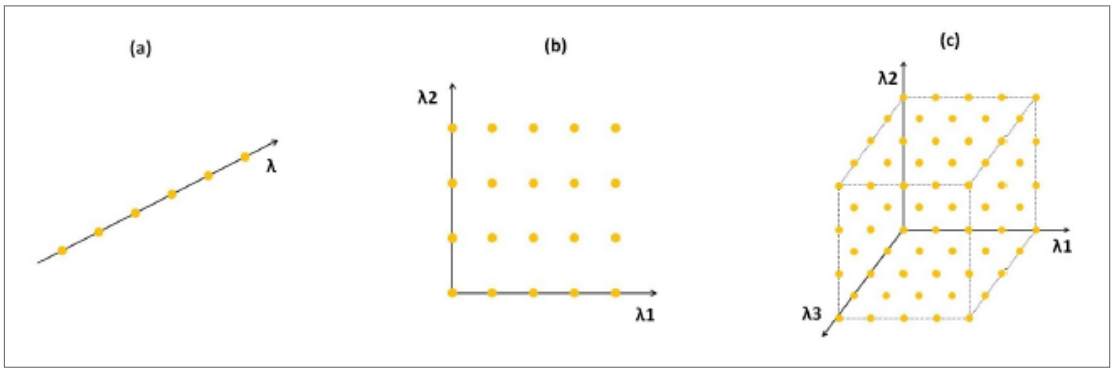
O LGBM é um algoritmo que se baseia em árvores de decisão que aplica um boosting gradiente. Ele possui em alta velocidade e eficiência no treinamento, baixo uso de memória, melhor acurácia, suporte para aprendizado paralelo, e capaz de lidar com grandes escalas de dados. O boosting presente no LGBM consiste em utilizar várias cópias do classificador para aprenderem sobre o conjunto de dados de forma interativa. Para classificar um resultado, é dado um peso para cada predição desses classificadores e então é a calculada a predição final. Cada tupla de treinamento, formado por um dado e sua classe, possui um peso. Assim que o primeiro classificador executa o processo de aprendizado, os pesos são alterados de modo a aumentar a atenção para as tuplas classificadas erroneamente por esse classificador. Com isso, o próximo terá maiores chances de acertar o que o primeiro classificador e assim sucessivamente. O último classificador impulsionado (boosted), combina os votos de cada classificador individualmente, onde os pesos de cada classificador é uma função de sua acurácia (FREITAS, 2019).

Em problemas de classificação para múltiplas classes, o LGBM pode utilizar algumas métricas, e uma delas é através da função objetivo softmax. Essa função retorna um vetor que representa a distribuição de probabilidades de uma lista de potenciais resultados. A soma das probabilidades desse vetor é igual a 1 (FREITAS, 2019).

2.6.7 Grid search

O Grid Search, ou Busca em Grade, é um método utilizado na otimização hiperparamétrica de modelos com um número reduzido de hiperparâmetros. Essa técnica usa uma forma de resolução simples e direta, além de, ser considerada de fácil implementação e paralelização. Cada hiperparâmetro do grid search é delimitado em torno de um intervalo particular de busca. Para cada hiperparâmetro, é estabelecido uma resolução de grade que determina a quantidade de pontos candidatos à serem considerados. Uma estrutura de grade é gerada no espaço hiperparamétrico, ela é formada pelo conjunto de $\mathbf{T}$ vetores que compõem todos os candidatos, $\lambda_{1:\mathbf{T}}$, sendo estes avaliados nos treinamentos do modelo de aprendizado, por meio uma função de perda (£), resultando na avaliação $\mathbf{f}$. A finalidade de todo este procedimento é achar o melhor ponto de configuração (λ^*), entre todas as configurações ($\lambda_{1:\mathbf{T}}$) experimentadas. Após delimitar um intervalo do espaço hiperparamétrico, o método de busca em grade possui grande chance de atingir um valor ótimo, ou muito próximo do ótimo(JUNIOR, 2018). A figura 15 mostra 3 exemplos de grades hiperparamétricas para diferentes dimensões.

Figura 15 – Exemplos de grades hiperparamétricas para (a) uma, (b) duas e (c) três dimensões.



Fonte: (JUNIOR, 2018)

Esse método busca os melhores parâmetros através da análise dos resultados obtidos com a execução do algoritmo para o intervalo de parâmetros. A função de kernel RBF possui dois parâmetros que podem sofrer o grid search que são: C e γ, não se sabe previamente qual ou quais os melhores parâmetros para o problema. O grid search pode ser utilizado em conjunto com a validação cruzada, seção 2.7.1. A validação cruzada diminui a probabilidade dos conjuntos de teste e treinamento serem particionados de forma inadequada, o que, manualmente, pode acontecer (OLIVEIRA, 2010).

O problema da utilização do método grid search está no crescimento exponencial de treinamentos necessários, à medida que o número de hiperparâmetros a serem estimados aumenta (JUNIOR, 2018).

2.7 AVALIAÇÃO DE CLASSIFICADORES E MÉTRICAS

2.7.1 Validação cruzada - K-fold

Segundo Tan, Steinbach e Kumar (2006) a validação cruzada *K-fold* é um dos métodos mais aplicados. Nessa validação os dados iniciais são divididos aleatoriamento em K subconjuntos mutuamente exclusivos chamadas de dobras, D_1, D_2, ..., D_k, cada um com tamanho aproximadamente igual. O treinamento e o teste são realizados K vezes. Na interação i, a partição D_i é reservada como o conjunto de testes e as partições restantes são usadas para treinar o modelo. Cada amostra é usada o mesmo número de vezes para treinamento e uma única vez para teste. Para classificação, a estimativa de precisão é o número geral de classificações corretas das k iterações, dividido pelo número total de tuplas nos dados iniciais. Habitualmente, o k adotado é 5 ou 10. O caso especial, quando $k = u$, onde u é o tamanho do conjunto de dados, é chamado de validação cruzada leave-one-out, onde o conjunto de testes compreende um único ponto e os dados restantes são usados para fins de treinamento.

2.7.2 Métricas

Após o treinamento os diferentes métodos de aprendizagem deverão ser avaliados para que possam ser comparados. Para realizar a avaliação algumas métricas comumente utilizadas em problemas de classificação serão utilizadas.

2.7.2.1 Matriz de confusão

Segundo, Marques (2017), a matriz de confusão é uma ferramenta importante, pois permite analisar de forma rápida o desempenho de cada sistema. Os valores que constituem a matriz são obstidos fornecendo os segmentos do conjunto de teste ao método de classificação e comparando sua predição com a classe correta de cada segmento.

Tabela 9 – Matriz de Confusão de problema contendo duas classes

	Positivo	Negativo
Positivo	VP Verdadeiro Positivo	FP Falso Positivo
Negativo	FN Falso Negativo	VN Verdadeiro Negativo

Fonte: A Autora (2020)

Os valores são classificados em quatro possíveis opções, são elas:

1. Verdadeiro Positivo (VP): segmentos que pertencem a classe positiva e foram classificados como positivos.

2. Falso Positivo (FP): segmentos que pertencem a classe negativa e foram classificados como positivos.

3. . Falso Negativo (FN): segmentos que pertencem a classe positiva e foram classificados como negativos.

4. Verdadeiro Negativo (VN): segmentos que pertencem a classe negativa e foram corretamente classificados como negativos.

2.7.2.2 Métricas de qualidade

1. Acurácia : Indica a quantidade de acertos feitos pelo modelo em relação ao todo informado assim o quanto um resultado se aproxima do real.

2. Precisão: Tem como objetivo identificar quantas amostras foram classificadas positivamente. Em outras palavras, é um medida do quão exato é a classificação para as amostras positivas. Ela é definida como o número de verdadeiros positivos dividido pela soma de veradeiros positivos mais falso positivos (MARQUES, 2017).

$$P = \frac{VP}{VP + FP} \tag{2.5}$$

3. Recall : Quantidade de segmentos positivos que foram classificados corretamente dividido pela soma de verdadeiros positivos mais falso negativos (MARQUES, 2017).

$$R = \frac{VP}{VP + FN} \tag{2.6}$$

4. F1 Score: Marques (2017), define o F1 scores, como sendo uma métrica que considera tanto a taxa de precisão quanto o recall. O seu valor tradicional é a medida harmônica entre a precisão e o recall multiplicado pela contaste 2.

$$F1 = 2 * \frac{PxR}{P + R} \tag{2.7}$$

5. Area Under the Receiver Operating Characteristic (AUCROC) : Para cada segmento apresentado ao classificador esse irá ter uma saída a classe predita para aquele segmento. Quanto mais próximo a saída estiver do 0 maior a certeza do classificador de que aquele segmento pertece a classe negativa e quanto mais próximo de 1 da classe positiva. A curva ROC funciona através da variação de um limiar de classificação que vai de 0 a 1 e mede a certeza que o método de classificação tem para cada um dos limiares. Quanto mais próximo a curva esiver do canto superior esquerdo maior

é a certeza do classificador sobre as classificações feitas. A área embaixo da curva mede exatamente esse nível de certeza do método a ser avaliado. A linha tracejada é a representação de um classificador aleatório (MARQUES, 2017).

2.8 COMPARAÇÃO DE CLASSIFICADORES

2.8.1 Teste de Kruskal-Wallis

O Portal Action, portal estatítico elaborado Estatcamp, descreve que o teste de Kruskal-Wallis é um teste não paramétrico que tem como finalidade comparar três ou mais populações. Ele é usado para testar a hipótese nula de que todas as populações possuem função de distribições iguais contra a hipótese alternativa de que ao menos duas das populações possuem funções de distribução diferentes .

Hipóteses

- As N amostras aleatórias $X_{J_1}, X_{J_2}, X_{J_3}, ..., X_{J_n}$ com $J = 1, 2, 3, ..., k$ mutuamente independentes.
- Para cada $j\epsilon 1, ..., k$ as n_j variáveis aleatórias $X_{J_1}, X_{J_2}, X_{J_3}, ..., X_{J_n}$ são uma amostra aleatória de um distribuição contínua com função de distribuição F_J.
- As funções de de distribuição $F_1, F_2, ..., F_K$ se relacionam através da relação $F_j(t) = F(t - \tau_j), -\infty < \infty$, para $J = 1, 2, ..., k$ em que F é uma distribuição contínua com mediana desconhecida. Nesse caso, a hipótese nula H_0 de interesse é a de que não há diferença entre os efeitos $\tau_1, ..., \tau_k$, isto é : $H_0 : \tau_1 = \tau_2 = ... = \tau_k$, essa hipótese nula garante que cada função de distribuição é igual.

Passos para realização do teste

1. Estabelecer as hipóteses
$$\begin{cases} H_0 : \tau 1 = \tau 2 = ... = \tau_k \\ \quad H_1 : \tau_1, \tau_2, ..., \tau_k. \end{cases}$$
2. Ordenar de forma crescente de magnitude os valores do conjunto de dados e associar a cada valor seu posto (rank) correspondente.
3. Calcular o valor da estatística H. Em seguida, fixar o nível de significância α.
$$H = \frac{\frac{12}{n(n+1)} \sum_{i=1}^{K} n_i (R_j - \frac{N+1}{2})^2}{1 - \frac{\sum_{j=1}^{\alpha} t_j^3 - t_j}{N^3 - N}} = \frac{(\frac{12}{N(N+1)} \sum_{i=1}^{K} \frac{R_j^2}{n_i}) - 3(N+1)}{1 - \frac{\sum_{j=t}^{\alpha} t_j^3 - t_j}{N^3 - N}}$$
4. Encontrar os valores críticos referentes ao nível de significância fixado. Calculando os valores Q_α de modo que $P[H > Q_\alpha] = \alpha$ (sob H_0)
5. Se $H_{obs} > Q_\alpha$ rejeita-se a hipótese nula que as amostras provém de populações igualmente distribuídas.

6. O p-valor é calculado : $P - valor = P[X^2_{K-1} \geq H|H_0]$

Quando a hipótese nula H_0 é rejeitada no teste de Kruskal-Wallis, indica que ao menos um dos grupos é diferente dos demais. Para determinar essa diferença existe um procedimento simples, onde é testada a diferença entre todos os pares de grupos. Testa-se a significância dos pares de diferenças através da seguinte desigualdade : $[R_i. - R_j.] \geq Z(\frac{\alpha}{K(K-1)})\sqrt{\frac{N(N+1)}{12}(\frac{1}{n_i} + \frac{1}{n_j})}$ em que

- n_i e n_j são os tamanhos da amostra dos grupos i e j respectivamente;
- $N = n_1 + n_2 + .. + n_k$ o número total de elementos considerados.
- R_i e R_j é o efeito dos postos dos grupos i e j.
- $[R_i. - R_j.]$ é a diferença observada;
- $Z(\frac{\alpha}{K(K-1)})\sqrt{\frac{N(N+1)}{12}(\frac{1}{n_i} + \frac{1}{n_j}}$ é a diferença crítica.

3 TRABALHOS RELACIONADOS

3.1 IDENTIFICAÇÃO AUTOMÁTICA DE FRUTAS

Assim como outros produtos agrícolas, as frutas estão sujeitas a danos e perdas ao longo de todas as fases de sua produção, porém, é na pós-colheita que se concentram os maiores prejuizos, que são devidos principalmente ao processo, embalagem, manuseio e transportes inadequados. Tudo isso porde vir a produzir alterações mecânicas, fisiológicas ou patológicas, degradando a qualidade do produto (CARVALHO et al., 201).

A automação na agricultura entra em jogo para aumentar a produtividade, a qualidade e o crescimento econômico do país. A seleção e classificação de frutas merecem atenção, é um processo importante para os produtores que afeta a avaliação da qualidade das frutas no mercado. Embora elas possam ser feitas pelo ser humano elas são lentas, trabalhosas, propensas a erros e tediosas. Portanto, passa a ser uma alternativa desejada um sistema automatizado de classificação de frutas. Nos últimos anos, os pesquisadores desenvolveram numerosos algoritmos para seleção e classificação de frutas usando a visão computacional e narizes eletrônicos associados a métodos inteligentes de aprendizagem de máquina. As características de cor, textura, morfologia, tamanho, maturação, aroma e sabor são utilizadas para identificar as doenças, maturidade e classes dos frutos (TIAN et al., 2019).

As sessões seguintes mostrarão alguns trabalhos da literatura no âmbito da fruticultura utilizando visão computacional e narizes eletrônicos.

3.1.1 Visão computacional na fruticultura

A visão computacional faz com que, de modo artificial, a máquina consiga enxergar o objeto de estudo. Nessa tecnologia, uma câmera e um computador são utilizados em vez do olho humano, conseguindo assim, realizar a identificação, o rastreamento e a medição de alvos para posterior processamento de imagens. Com o desenvolvimento da visão computacional, seu uso tem se tornado cada vez maior no campo da automação agrícola e seu papel se torna cada vez mais fundamental nessa área (TIAN et al., 2019).

Os sistemas baseados em visão computacional (VC) pode identificar, por exemplo, hortaliças a partir de informações pré-definidas e armazenadas em um banco de dados. Esses sistemas podem ser aplicados na agricultura e, um exemplo de aplicação nesse domínio, é a classificação de hortaliças de acordo com o formato, peso, teor de açucar, defeito, tipo cor. (CALIXTO et al., 2016).

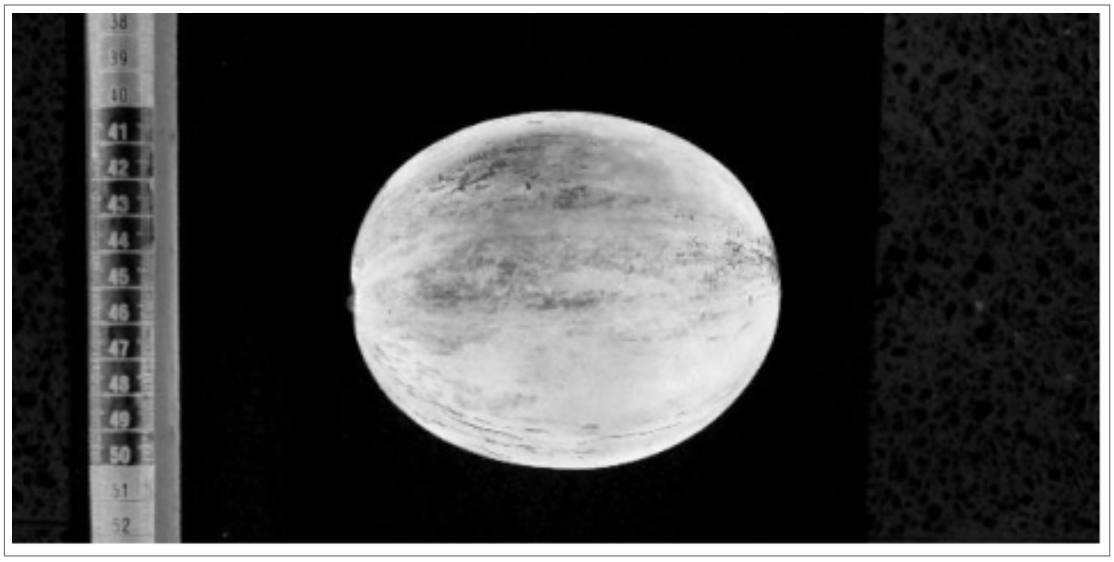
A figura 16 mostra o exemplo de imagem de entrada de um melão em um sistema de cores RGB (*Red*, *Green* e *Blue*) para um sistema com base em visão computacional.

Figura 16 – Imagem de entrada de um melão para sistema de visão computacional

Fonte: (CALIXTO et al., 2016)

A figura 17 mostra o exemplo de imagem de entrada do melão após transformação para uma imagem em tons de cinza.

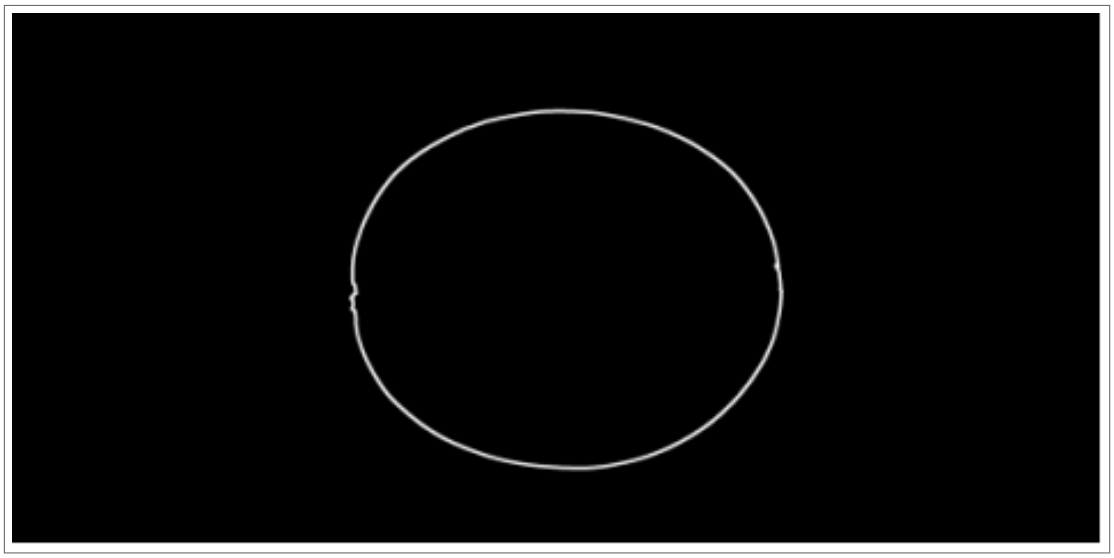
Figura 17 – Imagem do melão em tons de cinza



Fonte: (CALIXTO et al., 2016)

A figura 18 mostra a identificação do melão através da função cvFindContours da OpenCv. Essa função tem em seu principal objetivo detectar contornos em imagens.

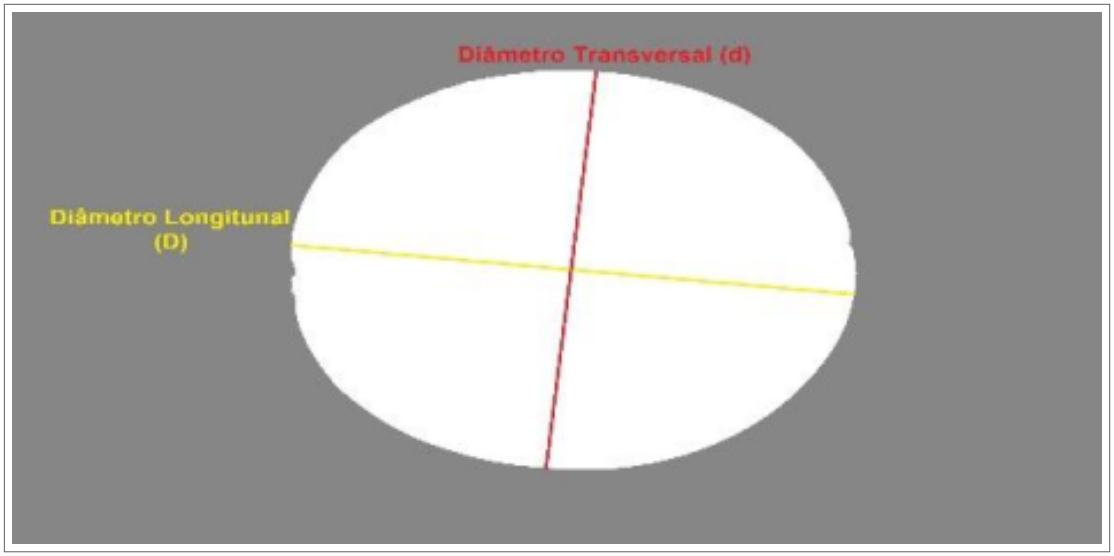
Figura 18 – Melão detectado por análise de bordas



Fonte: (CALIXTO et al., 2016)

A figura 18 mostra a identificação do diâmetro longitunal e do diâmetro transversal para o melão.

Figura 19 – Identificação das medidas D e d extraídas da imagem do melão



Fonte: (CALIXTO et al., 2016)

A classificação automática de produtos agrícolas possui maior eficiência quando comparado com o sistema manual, que é lento, tedioso, trabalhoso e propenso a erros. É necessário que existam sistemas de triagem automática que possa identificar esses produtos com base em suas características. A visão computacional baseia-se nessas características e busca entregar ferramentas para essa utilização (SEEMA; KUMAR; GILL, 2015).

3.1.2 Narizes eletrônicos na fruticultura

As frutas produzem uma ampla gama de compostos orgânicos voláteis (VOC's) que proporciona seus aromas exclusivamente distintos e contribuem para características únicas de sabor. O aroma e o sabor das frutas são caracteristícas de importância fundamental para a aceitação do consumidor nos mercados comerciais. A qualidade das frutas é avaliada pelos consumidores principalmente por sua percepção da aceitabilidade com base em características que incluem apelo visual (falta de manchas, cor, tamanho e textura), maturação, aroma e sabor. Os produtores, fornecedores e varejistas de frutas utilizam e confiam em testadores de painéis humanos (profissionais que julgam a qualidade das frutas com base nas características visuais e aromáticas), tradicionamente usados para avaliar a qualidade da fruta em mercados frescos. O advento dos dispositivos eletrônicos (narizes eletrônicos) trouxe ferramentas alternativas, usando medidas qualitativas e quantitativas mais consistentes das características aromáticas que evitam as opiniões subjetivas altamente variáveis dos painéis humanos. São instrumentos eficazes em discriminar misturas complexas de voláteis de frutas, eficaz em análise de aromas frutados e eficientes para substituir métodos caros convencionais (BAIETTO; WILSON, 2015).

O interesse inicial no uso de narizes eletrônicos como método não destrutivo para estudar as características dos fruto foi demostrado por Benady et al. (1995) que desenvolveram uma máquina sensora com um único sensor de gás semicondutor, localizada dentro de uma pequena xícara colocada na superfície de frutos de três diferentes cultivares de melão. O sensor usado foi o sensor TGS822 da Figaro. O instrumento poderia discriminar entre frutos maduros e imaturos com precisão de 90,2 % e classificar frutos em três categorias (imaturos, meio maduros e totalmente maduros) com uma precisão de 83 %.

Voss (2019) desenvolveu um protótipo de nariz eletrônico para identificar a maturação do pêssego Eragil no ciclo de crescimento pré-colheita. Foram estudados modelos com os 13 sensores semicondutores de óxido metálico (MOS) e a redução para 6 sensores com o auxílio do teste *Chi*-quadrado de Pearson, a fim de comparação. Para o treinamento e construção do modelo foram utilizadas amostras com 4 estágios de crescimento. Na etapa de testes das amostras, a taxa de erro foi de 1,92%, utilizando o método *Random Forest*, com análise discriminante linear. Os resultados finais da pesquisa foram satisfatórios, nas comparações quantitivas (erros) e qualitativas dadas pelo equipamento.

Zakaria et al. (2012) desenvolveram um estudo para classificação de mangas Harumanis em níveis de maturidade e maturação. Foi usada a fusão de dados de um nariz eletrônico e um sensor acústico. Três grupos de amostra, com dois tempos de colheita direntes (semana 7 e semana 8), foram avaliados pelo nariz e seguidos pelo sensor acústico. A Análise de Componentes Principais (PCA) e a Análise Discriminante Linear (LDA) foram capazes de discriminar a manga colhida nas semanas 7 e 8, com base unicamente no aroma e nos gases voláteis liberados nas mangas. Porém, quando seis grupos diferentes de diferentes níveis de maturidade e maturação foram combinados em uma análise de

classificação, o PCA e o LDA não conseguiram discriminar a diferença de idade das mangas Harumanis. Apenas quatro grupos foram observados usando o LDA, ao em vez de seis, o PCA mostrou apenas dois grupos distintos. Para aprimorar a classificação dos níveis de maturidade com o LDA um técnica de fusão de dados de baixo nível doi aplicada. No entando, nehuma melhoria significativa foi observada usando PCA com a técnica de fusão. Trabalhos adicionais usando uma rede neural híbrida LDA-Competitive Learning foram realizados para validar a técnica de fusão e classificar as amostras. Verificou-se que o LDA-CLNN também foi melhorado significativamente quando a fusão de dados foi aplicada.

3.1.2.1 Bananas

A.Sanaeifar et al. (2014) aplicaram um nariz eletrônico para detectar alterações de compostos voláteis orgânicos (VOC's) da banana durante o processo de amadurecimento. No estudo, para detecção dos odores foram utilizados os sensores MQ-3, ,Q-5, MQ-9, MQ131, MQ-135 e MQ-136. Os dados foram pré-processados através das estapas de identificação, manipulação e determinação da linha de base, compressão e normalização. Na ánalise dos dados pós pré-processamento foram usadas as técnicas PCA, LDA, SIMCA e SVM. Os resultados mostram que o nariz pode distinguir entre diferentes estágios de amadurecimento. Os resultados obtidos pelo PCA, categorizados pela maturação da banana, indicam que os dois primeiros componentes, PC1 e PC2, contém 82% da variação de dados. O primeiro componente principal, PC1, explica 58% da variação total, enquanto 24% da variância total é explicada pelo PC2. Os resultados obtidos pela LDA, também categorizados pela maturação da banana, permitem uma classificação, com 97,6% da variância total dos dados obtidos por LD1 e LD2 que representaram 85,7% e 11,9% da variância, respectivamente. A precisão da classificação obtida pelo método LDA com validação cruzada foi de 97,3%. O método SIMCA, obteve uma precisão de classificação de 92%, e a análise SVM teve uma taxa de sucesso alta na classificação de amadurecimento da banana, com precisões de treinamento e validação de 100 e 98,66%, respectivamente.

Chen et al. (2018) desenvolveram um nariz eletrônico que tem potencial para monitorar a variação dos COV's produzidos na banana durante o processo de maturação. Para a construção do dispositivo os sensores de MOS da Figaro foram utilizados, sendo eles : TGS2600, TGS2602, TGS2603, TGS2610, TGS2612 e TGS2620. Um sistema de câmera para monitorar a cor da casca da fruta também foi utilizado como recurso para identificação. O sistema duplo nariz/câmera apresenta melhor medida de separabilidade de classe nos 4 estágios de maturação da banana: imaturos, meio maduros, maduros e supermaduros. O PCA e o LDA foram aplicados as respostas dos sensores do nariz eletrônico para redução dos dados, e verificação de eventuais agrupamentos dos diferentes estágios de maturação da fruta. O LDA apresentou variabilidade de 94,11% para os dois componentes principais e o PCA 93,92%. Os dois métodos de redução de dimensiona-

lidade foram utilizados como entrada para 2 algoritmos de classificação, KNN e SVM. As respostas dos sensores MOS e as respostas do sistema de câmera foram agregados e utilizados no métodos de classificação. Isso permitiu uma perfeita classificação para todos os classificadores independente da imputação (PCA ou LDA). Portanto, em comparação com outros trabalhos, os dois sistemas agregados incrementam a precisão do sistema como um todo.

3.1.2.2 Maçã

Os danos nas maças podem causar deterioração das frutas, reduzir o valor econômico e dar origem a preocupações com a qualidade e a segurnaça dos alimentos. Pensando nisso, Li, Heinemann e Irudayaraj (2007) apresentam no estudo deles a aplicação de um nariz eletrônico (Cyranose 320) e zNose no desenvolvimento de um sistema não destrutivo, rápido e econômico para detecção rápida de deterioração em maças. Os principais compostos associados ao aroma de maçã foram identificados por cromatografia gasosa e espectrometria de massa, e as diferenças foram observadas após 6 dias de exposição a danos induzidos artificialmente na forma de um corte. Maças saudáveis e danificadas foram mantida em frascos de vidro de 2 L por 6 horas para pré-concentração antes da medição. Os dados de alta dimensão foram compactados por análise de componentes principais (PCA) e mínimos quadrados parciais (PLS). Os modelos de análise discriminante linear (LDA) e análise variada canônica (AVE) foram desenvolvidos com base nos dados compactados. Experimentos mostraram que o nariz eletrônico e o zNose foram capazes de detectar efetivamente as diferenças voláteis entre maças não danificadas e danificadas quatro ou mais dias após o corte. O modelo PLS-LDA produziu as melhores taxas de classificações coretas : 85% no nariz eletrônico e 96% no zNose.

Zhu et al. (2020) usaram nariz eletrônico e língua eletrônica acopladas à cromatografia gasosa-espectrometria de massa (SPME/GC-MS) para diferenciar sabores e outros indicadores de qualidade entre cinco variedades da maça ('Ralls', 'Jonagold', 'Orin', 'Indo' e 'Hanfu'). Os dados de saída do nariz eletrônico foram análisados pela análise linear discriminante (LDA), o aroma de 'Orin' foi bastante diferente de outras variedades. Haviam diferenças óbvias em acidez, salinidade e umami entre as cinco variedades de maçãs na análise da língua eletrônica. Um total de 45 compostos voláteis foram identificados e suas respectivas concentrações quantificadas usando SPME/GC-MS para as cinco variedades da maçã, e mostraram que os compostos mais abundates nas maçãs eram ésteres. Pela análise de componentes principais (PCA), butanoato de hexila, (E) -2-hexenal e α-farneseno foram os importantes compostos voláteis nas maçãs. A análise hierárquica por cluster (HCA), de acordo com a análise eletrônica de nariz e língua eletrônica e outros indicadores de qualidade, mostrou que o sabor de 'Hanfu' estava intimamente relacionado a 'Jonagold', enquanto o sabor de 'Indo' e 'Ralls' estava intimamente relacionado a 'Orin'. O método de análise colaborativa de combinação de nariz eletrônico e língua eletrônica

teve significância potencial na avaliação da diferença de sabor.

3.1.2.3 Uvas

Yamuna et al. (2011) descobriram que o lado do dossel (norte versus sul e leste versus oeste) tem efeito sobre o aroma do aroma emitido pelas frutas (e vinho) do franco Cabernet. Os dados de três datas de amostragem por estação foram processados usando dois narizes eletrônicos diferentes (um Cyranose 320 e um zNose 730) com base em duas estações de crescimento. Esses dados foram comparados com avaliações físico-químicas e sensoriais de aroma de vinho. Análises estatísticas univariadas e multivariadas geralmente indicavam que os índices físio-químicos da uva não podiam diferenciar de forma consistente (p>0,05) entre os lados do dossel ao longo das estações de crescimento e datas de amostragem. Já os dois narizes eletrônicos forneceram discrimicações completa dos lados do dossel para uvas e vinho usando análise discriminante caônica. O nariz eletrônico baseado em ondas acústicas de superfície explicou <50% da variação para uvas e <60% para o vinho usando o primeiro componente principal, em comparação com >80% para o nariz a base de polímero condutores. A avaliação sensorial do aroma do vinho diferenciou os lados do dossel em três das quatro avaliações.

McKellar et al. (2005) com o objetivo de fornecer informações úteis sobre atributos de qualidade, utilizaram o nariz eletrônico Fox 3000(EN) para testar à capacidade de caracterizar vinhos de frutas produzidos em Ontário. Oito vinhos de frutas (mirtilo, cereja, framboesa, groselha, sabugueiro, amora, maçã e pêssego) e quatro vinhos (tinto, Chardonnay, Riesling e gelo) foram obtidos em um mínimo de cinco vinícolas de Ontário. As réplicas de cada amostra de vinho foram secas em filtros de membrana para remover o etanol e foram analisadas pelo nariz eletrônico (Fox 3000 (EN)). Foi possível separar completamente cada variedade de vinho com base nas diferenças entre vinícolas, no entranto, quando todos os dados do vinho foram reunidos, a classificação por variedade foi ruim (58,7% corretamente classificadas). Alguns erros foram revelados na análise de diferentes variedades de vinho de uma única vinícola. Com base na posição no mapa de análise discriminante, os vinhos podem ser separados em quatro grupos distintos, com taxa de acerto de 79,9%. Os vinhos tintos e brancos foram bem separados um do outro (92,2% de acerto), assim como os vinhos de frutas e uvas (75,9%). Os resultados mostra que o nariz Fox 300 (EN) pode discriminar vinhos de frutas e uvas em agrupamentos naturais e úteis.

3.1.2.4 Laranjas

Natale et al. (2001) aplicaram um nariz eletrônico baseado em ressonadores de quartzo de modo cisalhamento de espessura (TSMR), para avaliar a qualidade de laranjas e maçãs pós-colheita. O nariz utilizado na pesquisa é um dos protótipos da série LibraNose (Universidade de Roma Tor Vergata e Technobiochip). Para análise dos dados ferramentas quimiométricas foram utilizadas, como análise de componentes principais (PCA) e

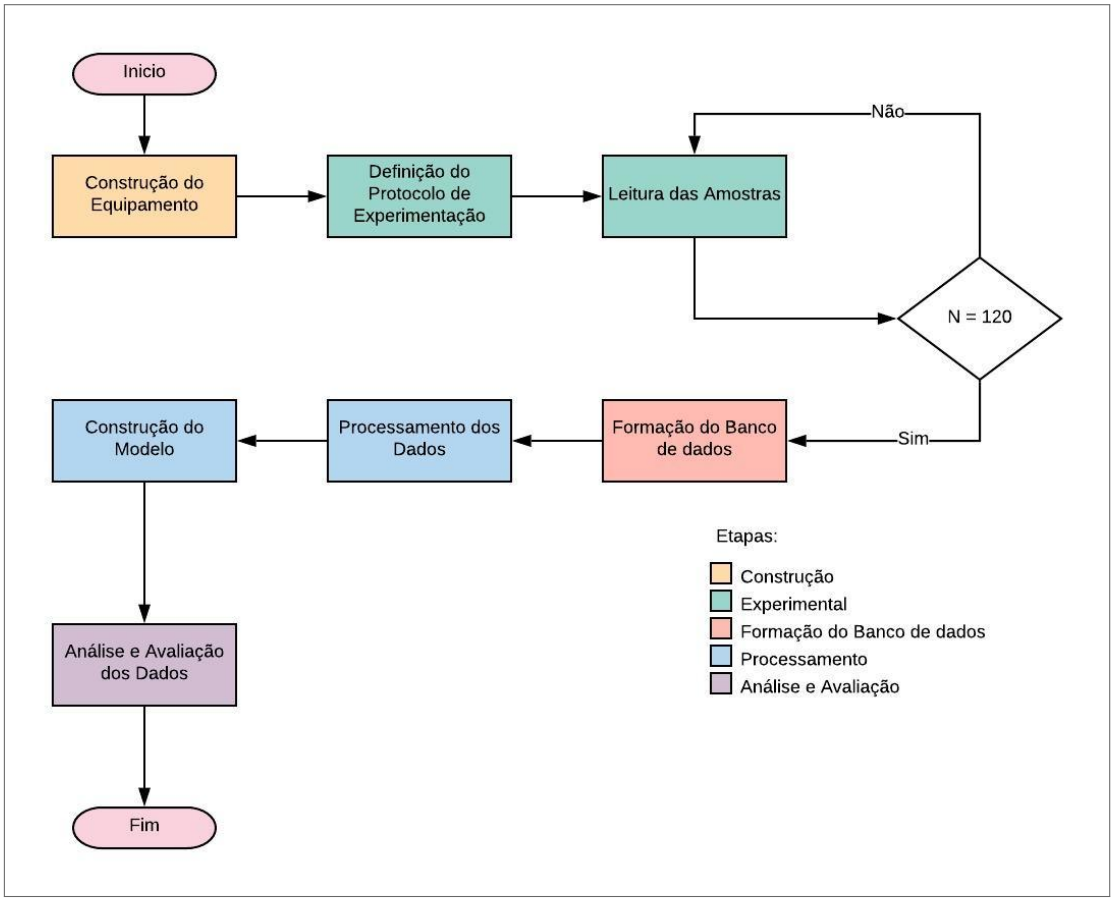
mínimos quadrados parcial (PLS). O PLS também foi usado para análise discriminante (PLS-DA), afim de encontrar uma representação de dados adequada, onde o objetivo da separação de classe supervisionada é maximizado. Os resultados demostraram que o nariz eletrônico possui sensibilidade e resolução suficientes para distinguir entre as várias classes e prever corretamente a quantidade de defeitos (para maças) e dias de armazenamento (para laranja).

Gomez et al. (2006) avaliaram a capacidade do nariz eletrônico PEN2, fornecido por (WMA Airsense Analysentechnik GmbH) Schwerin, Alemanha, para monitorar a mudança na produção volátil de tangerina durante diferentes datas de colheita. O nariz eletrônico portátil PEN2 possui uma matriz de 10 sensores de óxido de metal diferentes posicionados em uma pequena câmara (V = 1,8 mL). Análise do componente principal(PCA) e análise discriminante linear (LDA) foram utilizados para investigar se o nariz eletrônico era capaz de distinguir entre diferentes datas de coleta (estados de maturação). A análise de cargas foi utilizada para identificar os sensores responsáveis pela discriminação no arquivo de padrões atual. Os resultados obtidos provam que o nariz eletrônico pode discriminar com sucesso diferentes datas de colheita na tangerina usando a análise LDA. Porém usando a análise PCA, o nariz não foi capaz de detectar uma clara diferença no perfil volátil da tangerina. Durante a validação externa, foi utilizada a LDA para classificar adequadamente 92% do total de amostras.

4 MATERIAIS E MÉTODOS

A metodologia executada neste trabalho se divide basicamente em cinco etapas: contrução/parametrização do equipamento, estabelecimento do protocolo experimental, formação e análise e avaliação dos dados. A figura 20 ilustra o fluxograma referente à metodologia utilizada, que será descrita com mais detalhes nas seções subsequentes, no qual N é o número de amostras para cada fruta.

Figura 20 – Fluxograma ilustrativo da metodologia adotada



Fonte: A autora (2020)

4.1 CONSTRUÇÃO DO EQUIPAMENTO

Esta etapa pode ser subdividida em demostração do sensor utilizado e construção do hardware do dispositivo.

O sensor utilizado é um semicondutor de dióxido de estanho (SnO_2) da Figaro, o TGS 822. Esse sensor possui alta sensibilidade aos vapores de solventes orgânicos, bem como, outros voláteis. Ele também possui sensibilidade a uma variedade de gases combustíveis, como o monóxido de carbono, tornando-o um bom sensor de uso geral. Na presença de um gás detectável, a condutividade do sensor aumenta dependendo da concentração de

gás no ar. Mais especificamente, a resistividade do sensor aumenta na presença de ar e diminui na presença dos gases sensíveis. A tabela 10 mostra os recursos, aplicações, faixa de detecção de partículas, conforme os *datasheets* do fabricante, e o preço do sensor TGS 822.

Tabela 10 – Recursos, aplicações, faixa de detecção de partículas e preço do sensor Figaro TGS 822

Recursos	**Aplicações**
Alta sensibilidade a vapores de solventes orgânicos, como etanol	Detectores de álcool no ar expirado
Alta estabilidade e confiabilidade por um longo período	Detectores/Alarmes de vazamento de gás
Longa vida e Baixo Custo	Detectores de solventes para fábricas, Lavanderias a seco e Semicondutores.
Usa circuito elétrico simples	

Escopo de Detecção de Concentração	**Custo**
50-5000ppm Metano, etanol, iso-butano, CO, acetona, benzeno e n-hexano	R$ 36,03[a]

[a] https://www.soselectronic.com/products/figaro/tgs-822-7719
Preço obtido em novembro de 2019

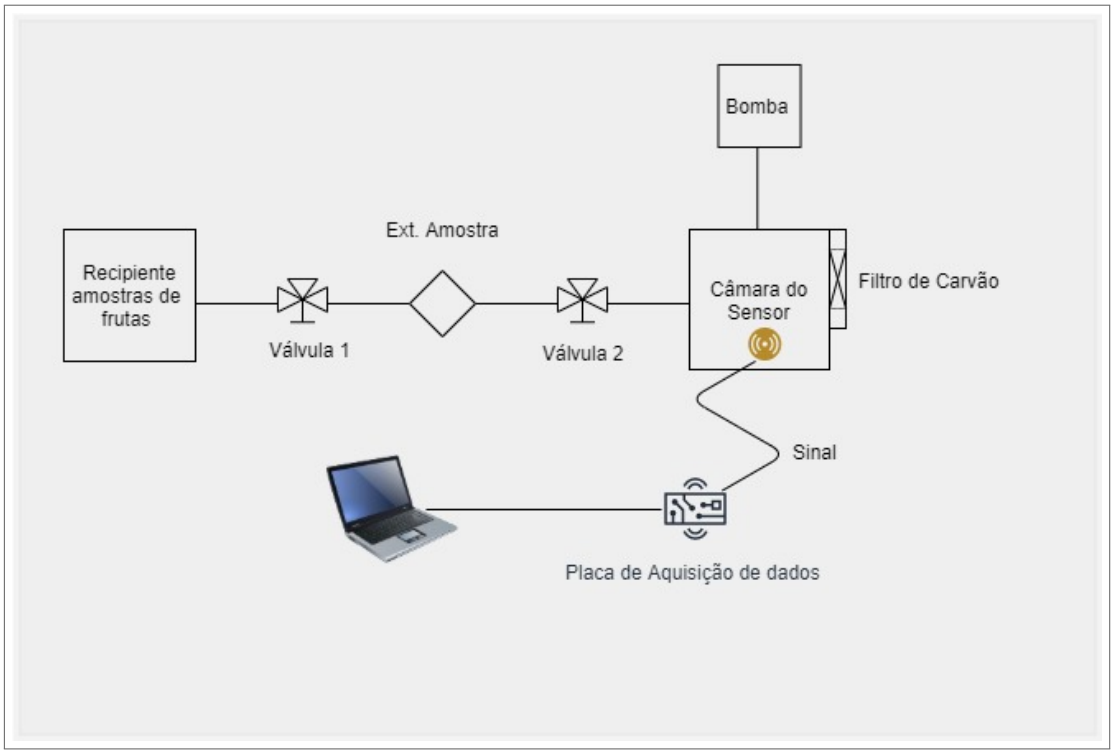
Fonte: A autora (2020)

O sensor de gás da família (TGS) foi escolhido por ser uma das mais utilizadas para construção de narizes eletrônicos, devido ao menor custo, confiabilidade e a fácil aquisição em comparação com as demais.

O nariz eletrônico utilizado nesse estudo foi desenvolvido pelo Centro Regional de Ciências Nucleares (NE) em parceria com o Centro de Informática (CIn) da UFPE. A figura 21 mostra a configuração do nariz eletrônico, que compreende um sensor de dióxido de estanho da FIGARO, o TGS822, uma placa de aquisição de dados, um notebook, um recipiente de amostra de frutas, uma bomba de ar, duas válvulas para controle de extração da amostra, um filtro de carvão ativado e uma câmara de armazenamento do sensor. A fruta é armazenada no recipiente, contendo um tubo de saída, controlado pela válvula 1. O sensor é colocado na câmara, um dos tubos da câmara do sensor é a via de entrada, que é controlada pela a válvula 2, para amostras das frutas. Outro tubo é conectado à bomba de ar para realizar a purga (limpeza) do sensor com uma sucção de ar interno e um filtro de carvão ativado na entrada da câmara. Uma placa de aquisição de dados fica conectada ao sensor e ao computador, que por sua vez, possui um *software* instalado para

identificação do sinal do sensor. Os resultados da percepção do sensor são mostrados e armazenados no computador.

Figura 21 – Diagrama do Sistema Proposto



Fonte: A autora (2020)

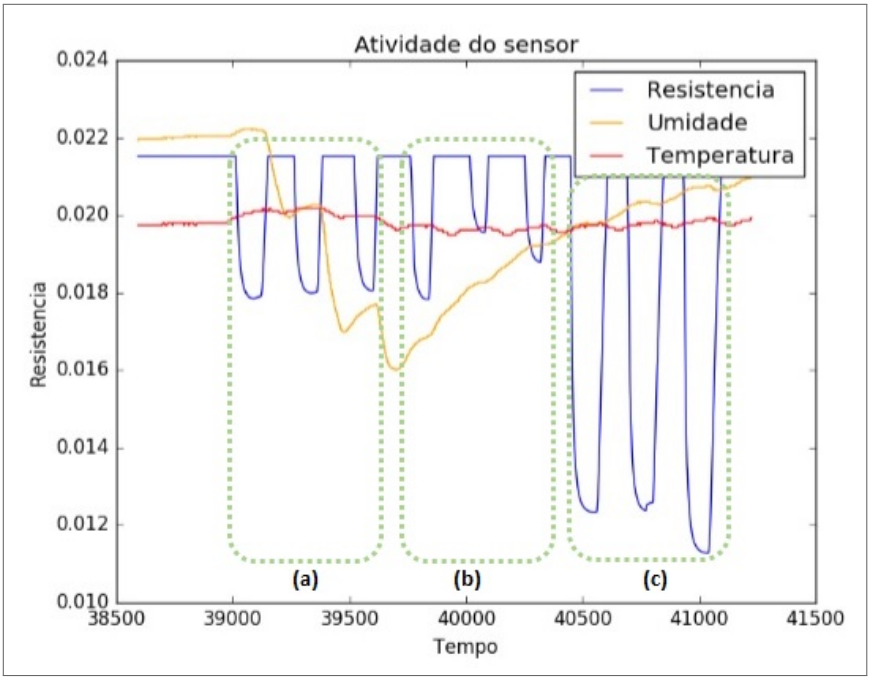
4.2 PROTOCOLO DE EXPERIMENTAÇÃO

A fim de avaliar o comportamento do sensor, foram realizados experimentos testes para verificar se o sensor era capaz de distinguir três tipos de frutas (banana, uva e laranja). Surgiram algumas dificuldades. Inicialmente, em relação a dinâmica da coleta e a escolha dos recipientes a serem utilizados para armazenar as frutas, pois, o material precisava ter a mínima adsorção, processo pelo qual átomos, moléculas ou íons ficam retidos na superfície de sólidos. O material escolhido foi o vidro, pelo seu baixo índice de adsorção de aroma. Após várias tentativas para escolher o volume ideal para o sensor conseguir detectar o aroma de cada fruta e o tempo de contado do sensor com ele, chegou-se a um padrão de coleta que será apresentado a seguir. A figura 22 exibe o resultado satisfatório na coleta teste, pois, revela que o sensor TGS 822 sozinho foi capaz de criar uma impressão de aroma para cada fruta.

Nesse experimento, utilizou-se três tipos de frutas (banana, uva e laranja). Cada fruta foi cortada e armazenada em recipientes de vidro por um tempo de aproximadamente 6 horas. Após esse tempo de armazenamento, utilizando um aparelho de sucção descatável, foram retiradas 3 amostras de cada fruta, cada uma correspondendo a 10 ml de aroma. Em seguida cada amostra é injetada no recipiente onde encontra-se o sensor. Após 60 segundos

dessa interação, é realizada uma purga, repetindo esse processo para cada amostra de fruta.

Figura 22 – Atividade do Sensor - Experimento Teste. (a) amostras de laranja. (b) amostras de banana. (c) amostras de uva



Fonte: A autora (2020)

Além dos experimentos com as frutas, foi realizado um experimento no laboratório do Instituto Federal de Pernambuco (IFPE), onde foi testada a sensibilidade do sensor a três tipos de cetonas: Acetofenona, Isopropanona e Isoforona. O conteúdo foi colocado em contato com o sensor e foram precisas pequenas quantidades de amostras das cetonas para a detecção.

Com base nas amostras coletadas anteriormente, foi definida a dinâmica de coleta usada nos experimentos desse estudo, compreendendo cinco passos, são eles:

1. Espera-se o sensor alcançar uma média de resistência de 34000 Ω (linha de Base).
2. A válvula 1 é aberta e coleta-se 10 ml do aroma da fruta.
3. A válvula 1 é fechada e, em seguida, a válvula 2 é aberta e o conteúdo é injetado na câmara onde está localizado o sensor.
4. O aroma é mantido em contato com o sensor por 1 minuto após a injeção.
5. É Feita uma purga (limpeza) do sensor com a sucção de ar interno e filtro de carvão ativado na entrada da câmara para o sensor chegar a linha de base inicial.
6. Volta ao passo 1.

O sinal utilizado como dado de entrada para os algoritmos é a variação da resistência do sensor no tempo que passa em contato com o aroma da fruta, ela é medida em ohms (Ω). O ohm é definido como a resistência de um corpo de modo que uma diferença de potencial de 1 volt através do corpo suceda em uma corrente de 1 ampère através dele, ou seja, a relação entre a tensão e uma corrente sobre um elemento (SBRUZZI; PLENTZ, 2018).

4.3 COLETA DAS AMOSTRAS

Os experimentos aconteceram na Universidade Federal de Pernambuco (UFPE) na cidade de Recife-PE. As frutas (banana, maçã, uva e laranja) usadas nos experimentos foram adquiridas no mercado local da região. As análises compreendem dois estados das frutas (frutas maduras intactas e cortada) afim de analisar a classificação pelo sensor TGS822.

Os experimentos foram divididos em dois momentos: coleta das frutas cortadas e coleta das frutas intactas. Todos os experimentos aconteceram numa sala do centro de informática da Universidade Federal de Pernambuco, com temperatura em torno de 23ºC. A tabela 11 mostra as datas de coleta de cada fruta para cada momento.

Tabela 11 – Datas da coleta das frutas cortadas e das intactas

Frutas Cortadas	
Fruta	**Data da Coleta**
Banana	01/05/2019
	02/05/2019
Maçã	06/05/2019
	09/05/2019
	11/05/2019
	14/05/2019
	16/05/2019
	20/05/2019
Laranja	07/08/2019
Uva	18/08/2019
Frutas Intactas	
Fruta	**Data da Coleta**
Banana	29/09/2019
Laranja	05/10/2019
Maçã	12/10/2019

Fonte: A autora (2020)

O equipamento foi mantido no mesmo local. Para todos os experimentos, para a coleta

das frutas cortadas, foram usados 100 g de cada fruta. Elas ficaram armazenadas 8 horas antes do início da coleta para uma melhor distribuição do aroma pelo recipiente. A coleta foi realizada com um aparelho de sucção descartável de 10 ml de capacidade No tubo de saída do recipiente de armazenamento das frutas, foi usado um filtro PTFE/TF 0.22, para garantir que a amostra só contesse o aroma da fruta. Para cada coleta, um aparelho de sucção e um filtro eram utilizados. O mesmo método de coleta e armazenamento foi usado para as frutas intactas.

O sensor necessita de um tempo para alcançar a estabilização de sinal. Nos experimentos, foi adotada uma tensão máxima de 34000 Ω e os resultados foram obtidos entre 6 e 8 minutos, compreendidos pela utilização do filtro de carvão ativado na entrada de ar do sensor.

Algumas dificuldades foram encontradas na coleta das amostras, a primeira foi referente ao tempo de purga (limpeza) do sensor na coleta da maça. Como podemos observar, foram 6 dias para coletar todas as amostras dessa fruta.No primeiro dia, foi observado que o sensor, após uma quantidade de amostras (20 amostras), demorava a voltar a sua linha de base (34000 Ω). Após um pesquisa, observou-se que a maçã possui enzimas que entram em atividade após o corte e interagem com o oxigênio, provocando a oxidação da fruta, esse processo é um mecanismo de defesa contra fungos e bactérias. Devido ao processo de limpeza do sensor ser feita com ar (oxigênio), estávamos provocando mais interações dessas enzimas. Uma solução para interromper esse processo de oxidação seria o ajuste na curva ou utilização de uma atmosfera de baixo teor de oxigênio. Com isso, para a limpeza total do sensor, ele precisou ficar ligado por mais de 24 horas.

A segunda dificuldade foi na coleta da uva intacta. Apesar de ter passado o mesmo tempo que as outras frutas no recipiente antes da coleta, o sensor não conseguiu reagir ao aroma exalado pela mesma. Esse comportamento pode ser explicado, pois, apesar da concentração de compostos vólateis da uva encontrar-se na casca, as variedades podem ser divididas em categorias aromáticas (uvas roxas) e pouco aromáticas (WILSON; BAIETTO, 2009). A uva utilizada nos experimento era da espécie Itália, uva verde.

Para cada fruta, coletas de 120 amostras eram feitas, nos dois estágios. Com isso, a base de dados das frutas cortadas compreende um total de 480 amostras e a base de dados das frutas intactas compreende um total de 360 amostras.

4.4 FORMAÇÃO DA BASE DE DADOS

Nesse trabalho, dois banco de dados foram utilizados com o objetivo de analisar a capacidade do nariz eletrônico, formado de um único sensor, ser capaz de distinguir entre várias frutas de duas maneiras: frutas maduras cortadas (Banco 1) e frutas maduras intactas (Banco 2).

Para a formação das bases de dados, foram desconsiderados o tempo de pré-aquecimento do sensor para chegar a linha de base e o tempo de retorno a linha de base entre uma

amostra e outra. Uma amostra corresponde a 10 ml do aroma da fruta que é injetada no recipente onde encontra-se o sensor. Os dados compreendem os 60 segundos de interação do sensor com a amostra do aroma. Para as etapas de treinamento e teste, foram utilizados 70% e 30% dos registros, respectivamente.

Como dito anteriormente, a base de dados composta pelas frutas cortadas possui 480 amostras (banana, maça, laranja e uva), e a base de dados das frutas intactas possui 380 amostras (banana, maça, laranja) . Para cada fruta foi atribuído um rótulo ficando distribuído da seguinte maneira:

- Classe 1 - Banana
- Classe 2 - Maça
- Classe 3 - Laranja
- Classe 4 - Uva

O banco de dados 1 foi organizado em uma tabela com 61 colunas. A primeira coluna corresponde a atribuição das classes (frutas) e as demais correspondem a frequência da resistência no intervalo de 60 segundos. Foram coletadas 120 amostras para cada classe de fruta, totalizando 480 linhas na base de dados. O banco de dados 2 segue a mesma organização, porém, com apenas 3 classes de frutas, totalizando 61 colunas e 360 linhas. O valor das linhas refere-se a mudança de resistência do sensor no tempo de um minuto em contato com a amostra, ou seja, os dados de entrada do sistema são variações de resistência do sensor durante o período.

4.5 PROCEDIMENTOS DE PROCESSAMENTO DOS DADOS

No processamento dos dados, seis algoritmos de classificação supervisionada foram utilizados: o KNN, que é o algoritmo mais simples, intuitívo e de fácil implementação para classificação (FACELI et al., 2011); o SVM, possui as vantagens de gerar fronteiras lineares ou não-lineares e o uso da representação em espaço e estados (VOSS, 2019); o RF, por causa de sua classificação precisa e robustez ao ruído em comparação com classificadores únicos, além de ser computacionalmente mais leve que outros métodos de conjunto de árvores (VOSS, 2019). O XGboost, um algoritmos baseado em árvore de decisão que usa uma estrutura de aumento de gradiente e desde sua introdução, não foi só creditado como vencedor de inúmeras competições Kaggle(Plataforma de competições de Data Science), mas também por sua força motriz para diversas aplicações na indústria de ponta (GOMES, 2019); o LightGBM, que assim como o XGBoost, tem sido muito utilizado em competições do Kaggle, também é baseado em árvores de decisão; e a MLP, com uma simples camada oculta.

O KNN e o SVM foram selecionados, pois, como mostrado em (TANG et al., 2010) eles obtiveram resultados satisfatórios para classsificação de três tipos de frutas.

Alguns trabalhos mostram que o uso do RF em narizes eletrônicos pode melhorar a taxa de precisão na regressão/classificação (VOSS, 2019).

O XGBoost e LightGBM foram selecionados por serem algoritmos que tem se destacados em competições do Kaggle, como dito anteriormente.

Em cada um dos algoritmos citados acima, foi utilizada uma técnica de otimização de hiperparâmetros, a busca em grade (*Grid Search*). A estimação do valor ótimo para o(s) hiperparâmetro(s) é fundamental para a obtenção de um modelo com melhor desempenho de predição.

A etapa de construção do modelo envolve fazer o algoritmo de classificação utilizar a base de dados para treinar e aprender sobre os dados coletados. Assim, é construído um modelo ideal que é responsável por predizer os novos dados inseridos no banco.

Todos os algoritmos foram implementados utilizando a linguagem Python, que possui uma elevada gama de bibliotecas e funções que auxiliam no processo de construção dos modelos. Os algoritmos foram executados na plataforma Google Colab.

4.6 BUSCA EM GRADE - VARIAÇÃO DE HIPERPARÂMETROS

Uma boa parte dos algoritmos de aprendizagem de máquina (AM) possui parâmetros cujos valores podem ser especificados pelo usuário. Em geral, esse valores para parâmetros livres influenciam diretamente no desempenho de modelos induzidos, o que pode ser entendido como uma deficiência das técnicas de AM. O objetivo do processo de ajuste desses parâmetros pode ser visto como encontrar os melhores valores dos parâmetros para um determinado conjunto de dados (ROSSI, 2009).

Uma das técnicas utilizadas para a definição do melhor conjunto de parâmetros é a busca em grade (*Grid Search*), um método heurístico que percorre todo o espaço de parâmetro definido pelo o usuário. Por isso, antes da avaliação dos classificadores foi realizada uma etapa preliminar de variação dos principais parâmetros de cada algoritmo avaliado por meio do método de busca em grade em conjunto com a validação cruzada. Nesse trabalho foi adotado para o k-fold o K= 10. Os principais parâmetros variados nos algoritmos do estudo estão descritos na tabela 12:

Tabela 12 – Principais Parâmetros Variados no Algoritmo para a Busca em Grade

Algoritmo	**Parâmetros Variados**
KNN	weights : ['uniform', 'distance'], metric : ['euclidean', 'manhattan'], leaf_size:[1,3,5], algorithm:['auto','kd_tree']

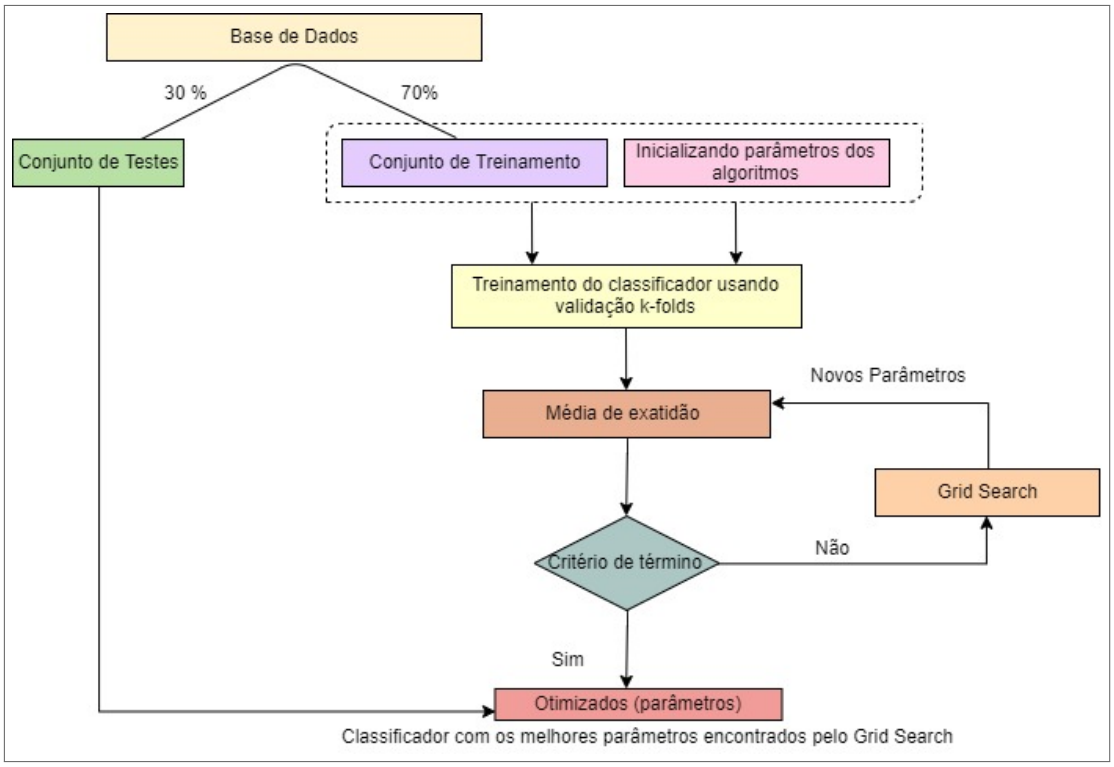
RF	{criterion: ['entropy', 'gini'], n_estimators: [25, 50, 100], bootstrap: [False, True], max_depth: [3, 5, 10], max_features: ['auto', 0.1, 0.2, 0.3]}
SVM	C: [0.1,1,10,100], gamma: [1,0.1,0.001]
MLP	{ hidden_layer_sizes : np.arange(5, 20), activation: ['tanh', 'relu'], solver: ['adam', 'sgd'], alpha: [0.0001, 0.05], learning_rate: ['constant','adaptive'], learning_rate_init:[0.001,0.01,0.1],}
XGBOOST	max_depth: [10,30,50], min_child_weight : [1,3,5], n_estimators: [200, 100], learning_rate: [0.05, 0.1, 0.16], gamma: [0.5, 1, 1.5], subsample: [0.6, 0.8, 1.0]
LIGHTGBM	max_depth: [25,50, 75], learning_rate : [0.01,0.05,0.1], num_leaves: [31,127], feature_fraction: [0.5, 1.0], bagging_fraction: [0.75, 0.95], reg_alpha: [0.1, 0.5]

Fonte: A autora (2020)

Para as duas bases de dados: frutas cortadas e frutas intactas, foi utilizada a mesma variação dos parâmetros.

A figura 23 ilustra o comportamento da busca em grade realizada no estudo, onde testes sucessivos na busca de parâmetros ótimos são realizados. A base de dados foi dividida em um conjunto de teste e em um conjunto de treinamento. O algoritmo é executado e os valores possíveis para os parâmetros de cada algoritmo são testados.

Figura 23 – Ilustração do Ciclo de um Grid-Search na busca de Parâmetros Ótimos



Fonte: A autora (2020)

4.7 PROCEDIMENTO DE AVALIAÇÃO DOS DADOS

O método *K-fold cross-validation* foi utilizado para validação dos resultados obtidos pelo melhor classificador da busca em grade. A validação cruzada *K-fold* é uma técnica comumente empregada, que a partir de um conjunto de X exemplos, realiza a partição em K conjuntos (*folds*) de tamanhos iguais. Para cada conjunto, um classificador é treinado nos outros conjuntos. Portanto, a validação do modelo antes da etapa de teste irá auxiliar no processo de verificar quais os parâmetros que ajustam melhor na construção de um classificador que gere uma resposta mais satisfatória para o tipo de conjunto de dados que envolve o problema. Assim, o objetivo é construir um modelo ótimo de classificação para a base de dados (VOSS, 2019).

Após isso, o modelo é testado com um conjunto de teste, ou seja, dados desconhecidos para o determinado modelo. Se o resultado com a base de teste é satisfatória, o modelo pode ser considerado aceito, caso contrário, tem-se possivelmente um caso de sobre-ajuste ou overfitiing, que é quando o modelo se ajusta demais e obtém respostas satisfatórias para a base de treino, mas, para novos dados é ineficiente.

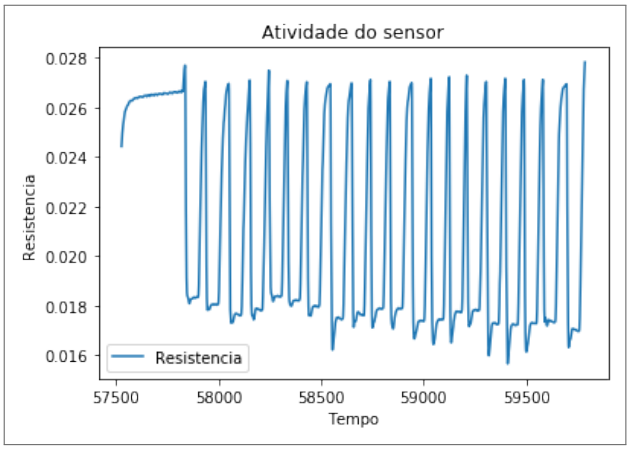
5 RESULTADOS

Neste capítulo serão apresentados os resultados obtidos, mostrando as respostas geradas pelo sensor utilizado(seção 5.1) e pela busca em grade (seção 5.2). Também serão descritos nesse capítulo os resultados obtidos com os métodos de classificação, tanto para validação, quanto para teste, dos dois bancos utilizados além das métricas e matrizes de confusão (seção 5.2.1). Por último, serão apresentadas as comparações dos modelos (seção 5.2.2) e com outros trabalhos presentes na literatura (seção 5.2.3).

5.1 RESPOSTAS GERADAS PELO SENSOR

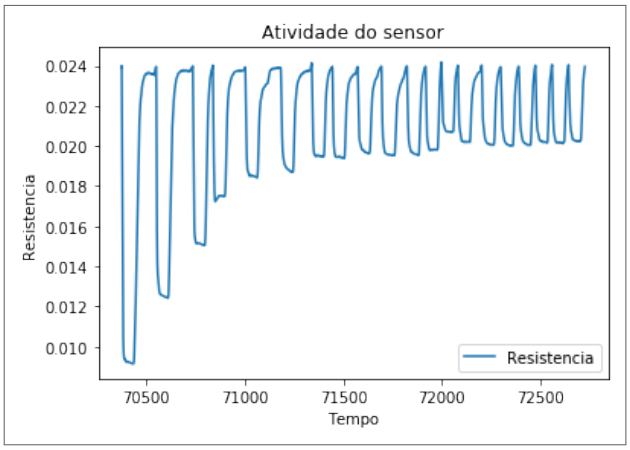
O sensor utilizado nos experimentos é um semicondutor de dióxido de estanho, o TGS822 da Figaro. O registro da modificação da resistência do sensor, quando exposto as frutas dos experimentos, pode ser visto abaixo: Nas figuras 24, 25, 27 a 26 o sensor foi exposto as frutas cortadas. Enquanto nas figuras 28, 29 30 as frutas intactas, mostrando a mudança de resistência, ou seja, a atividade do sensor.

Figura 24 – Resposta do Sensor: Mudança de Resistência para a Banana Cortada



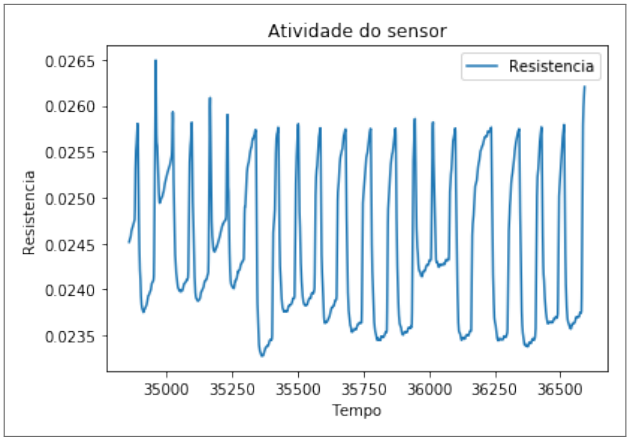
Fonte: A autora (2020)

Figura 25 – Resposta do Sensor: Mudança de Resistência para a Maça Cortada



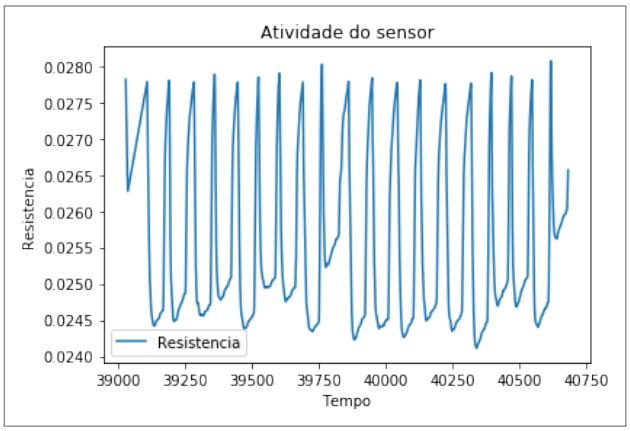
Fonte: A autora (2020)

Figura 26 – Resposta do Sensor: Mudança de Resistência para a Uva Cortada



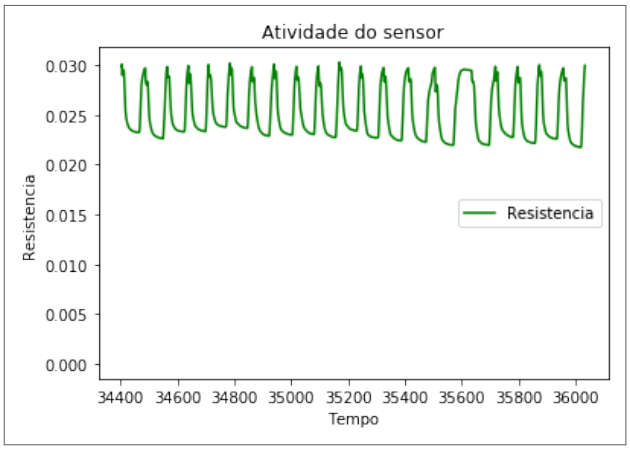
Fonte: A autora (2020)

Figura 27 – Resposta do Sensor: Mudança de Resistência para a Laranja Cortada



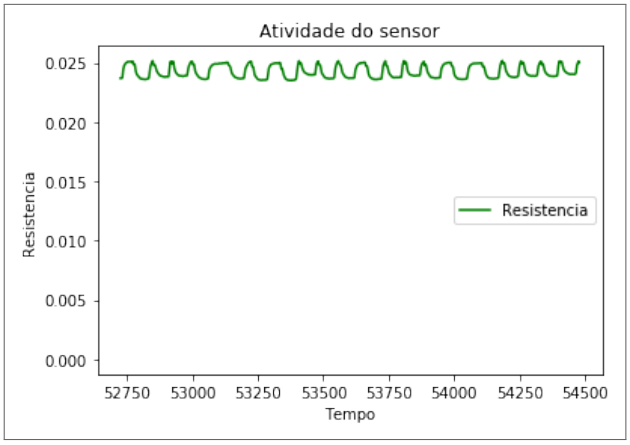
Fonte: A autora (2020)

Figura 28 – Resposta do Sensor: Mudança de Resistência para a Banana Intacta



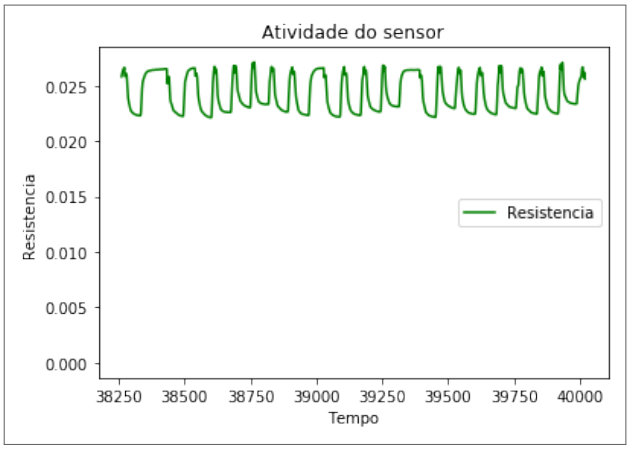
Fonte: A autora (2020)

Figura 29 – Resposta do Sensor: Mudança de Resistência para a Maçã Intacta



Fonte: A autora (2020)

Figura 30 – Resposta do Sensor: Mudança de Resistência para a Laranja Intacta



Fonte: A autora (2020)

Essa variação na resistência é o que chamamos de impressão aromática da fruta. Ela nos mostra como o sensor reaje a cada aroma. Podemos perceber que as respostas do sensor para as três frutas intactas são distintas entre si, o que nos mostra que a banana, maça e a laranja intacatas possuem impressões aromáticas diferentes então espera-se uma precisão de classificação muito boa, para essa base de dados.

5.2 RESPOSTA GERADAS PELA BUSCA EM GRADE

Conforme foi discutido na seção 5.2, antes da avaliação dos classificadores, foi realizada uma etapa preliminar de variação dos principais parâmetros de cada algoritmo avaliado por meio do método de busca em grade em conjunto com a validação cruzada. Esse método, após a busca, nos retorna o melhor conjunto de parâmetros, entre os definidos, para a base de dados em questão. As tabelas 13 e 14 demostram os resultados dos melhores parâmetros dados pela pesquisa em grade para cada algoritmo utilizado nesse estudo, para o banco 01 e para o banco 2.

Tabela 13 – Resultados dos Melhores parâmetros para cada Algoritmo - Banco 1 (Frutas Cortadas)

Algoritmo	**Melhor Parâmetro**
KNN	{'algorithm': 'auto', 'leaf_size': 1, 'metric': 'euclidean', 'n_jobs': -1, 'n_neighbors': 4, 'weights': 'uniform'}
RF	{'bootstrap': True, 'criterion': 'entropy', 'max_depth': 10, 'max_features': 'auto', 'n_estimators': 50}
SVM	{'C': 1, 'gamma': 1, 'kernel': 'linear'}
MLP	{'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': 9, 'learning_rate': 'constant', 'learning_rate_init': 0.01, 'solver': 'adam'}

XGBOOST	{'gamma': 0.5, 'learning_rate': 0.1, 'max_depth': 10, 'min_child_weight': 1, 'n_estimators': 200, 'subsample': 0.8}
LIGTH GBM	{'bagging_fraction': 0.75, 'feature_fraction': 0.5, 'learning_rate': 0.1, 'max_depth': 25, 'n_estimators': 200, 'num_leaves': 31, 'reg_alpha': 0.1}

Fonte: A autora (2020)

Tabela 14 – Resultados dos Melhores parâmetros para cada Algoritmo - Banco 2 (Frutas Intactas)

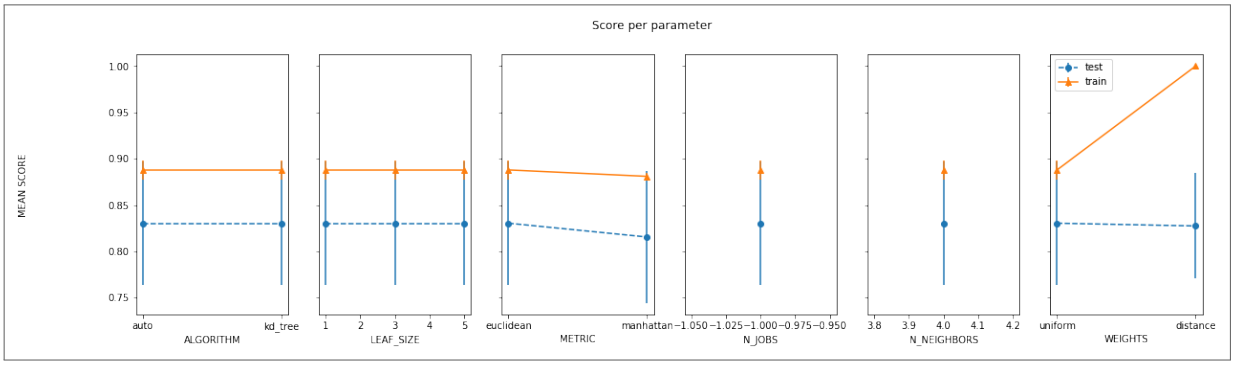
Algoritmo	**Melhor Parâmetro**
KNN	{'algorithm': 'auto', 'leaf_size': 1, 'metric': 'euclidean', 'n_jobs': -1, 'n_neighbors': 4, 'weights': 'distance'}
RF	{'bootstrap': True, 'criterion': 'entropy', 'max_depth': 5, 'max_features': 0.3, 'n_estimators': 25}
SVM	{'C': 0.1, 'gamma': 1, 'kernel': 'linear'}
MLP	{'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': 19, 'learning_rate': 'constant', 'learning_rate_init': 0.001, 'solver': 'adam'}

XGBOOST	{'gamma': 0.5, 'learning_rate': 0.05, 'max_depth': 10, 'min_child_weight': 1, 'n_estimators': 200, 'subsample': 1.0}
LIGTH GBM	{'bagging_fraction': 0.75, 'feature_fraction': 0.5, 'learning_rate': 0.01, 'max_depth': 25, 'n_estimators': 200, 'num_leaves': 31, 'reg_alpha': 0.1}

Fonte: A autora (2020)

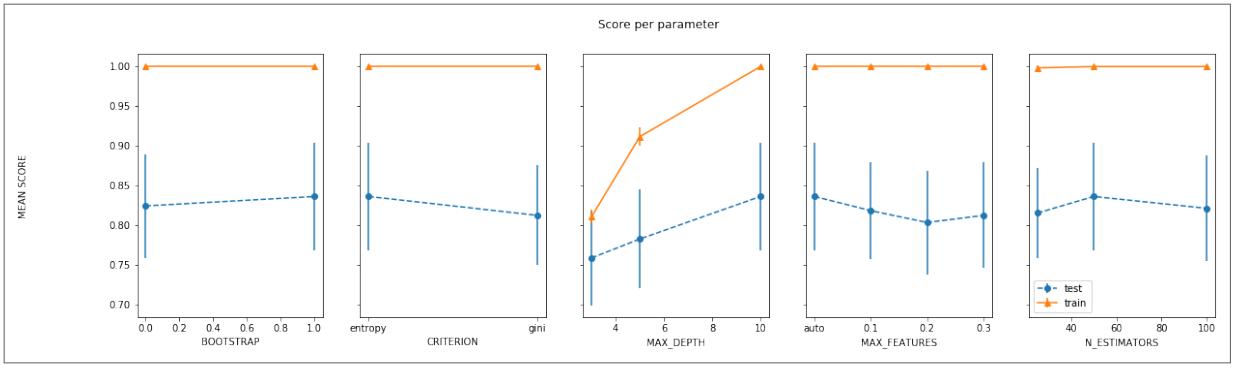
Um processo de validação cruzada é realizado para determinar o valor do hiper parâmetro definido que fornece os melhores níveis de precisão. Os resultados da média dos scores por parametro para cada algoritmo pode ser visto abaixo. Banco 1: Figuras 31, 32, 33, 34, 35, 36 e Banco 2: Figuras 37, 38, 39, 40, 41, 42. Podemos observar a variação dos *scores* por parâmetros e a média desses scores, tanto no conjunto de treinamento como no conjunto de teste do *cross validation*.

Figura 31 – Score por parâmetro do KNN - Banco 1 (Frutas Cortadas)



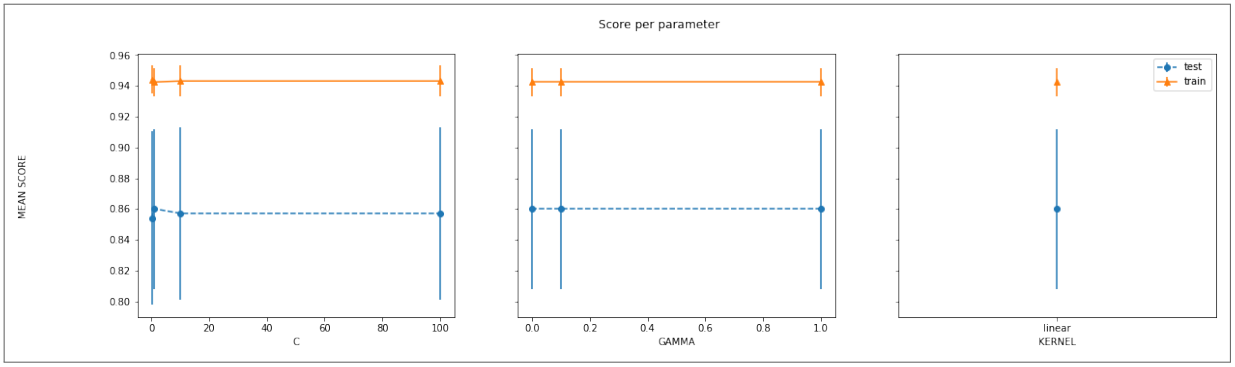
Fonte: A autora (2020)

Figura 32 – Score por parâmetro do RF - Banco 1 (Frutas Cortadas)



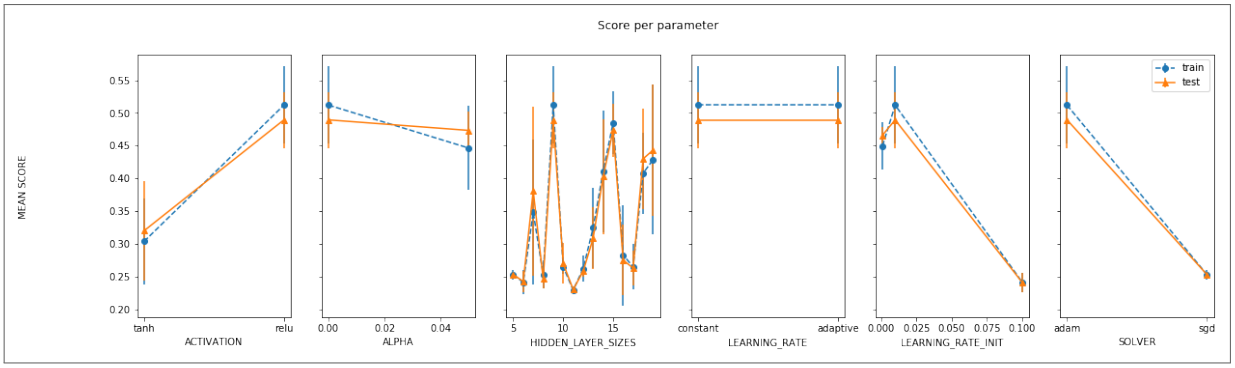
Fonte: A autora (2020)

Figura 33 – Score por parâmetro do SVM - Banco 1 (Frutas Cortadas)



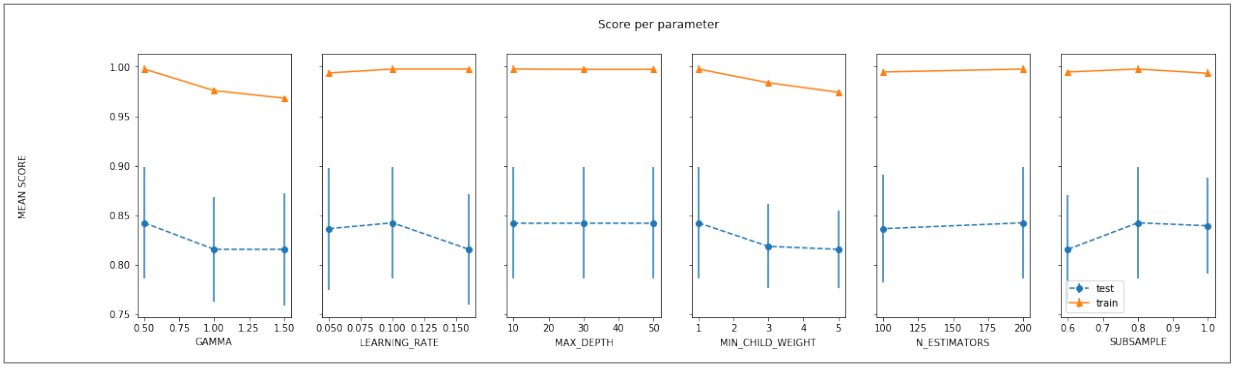
Fonte: A autora (2020)

Figura 34 – Score por parâmetro do MLP - Banco 1 (Frutas Cortadas)



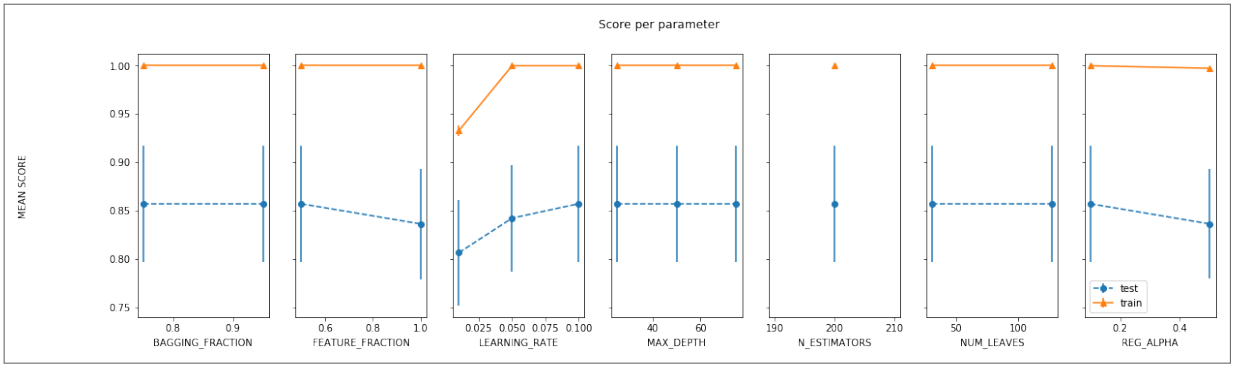
Fonte: A autora (2020)

Figura 35 – Score por parâmetro do GBOOST - Banco 1 (Frutas Cortadas)



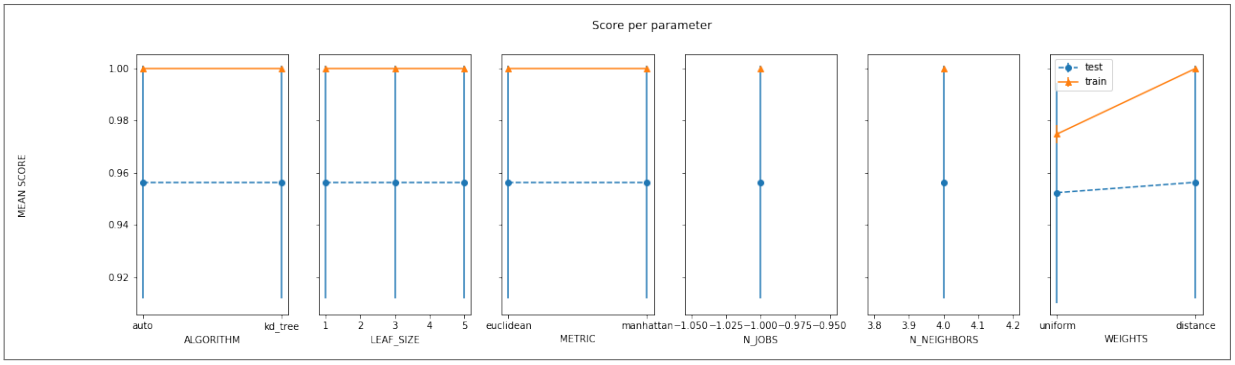
Fonte: A autora (2020)

Figura 36 – Score por parâmetro do LIGTH GBM - Banco 1 (Frutas Cortadas)



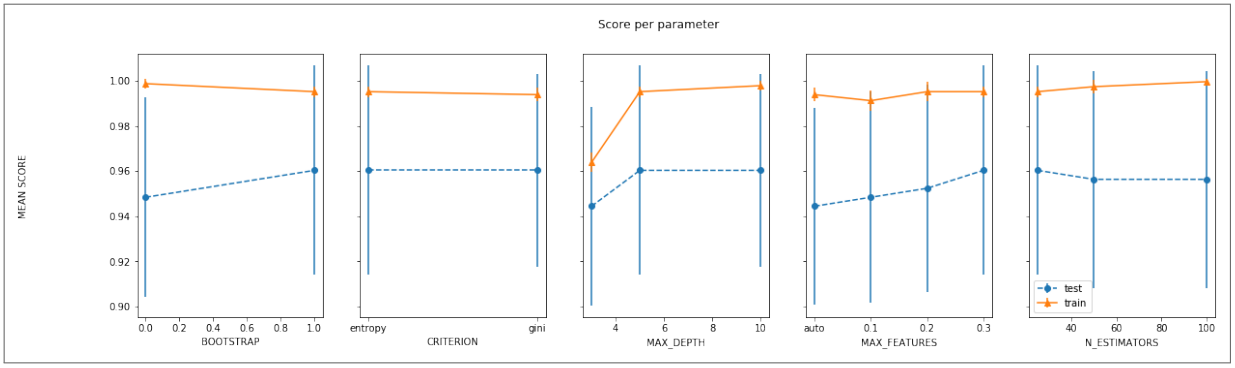
Fonte: A autora (2020)

Figura 37 – Score por parâmetro do KNN - Banco 2 (Frutas Intactas)



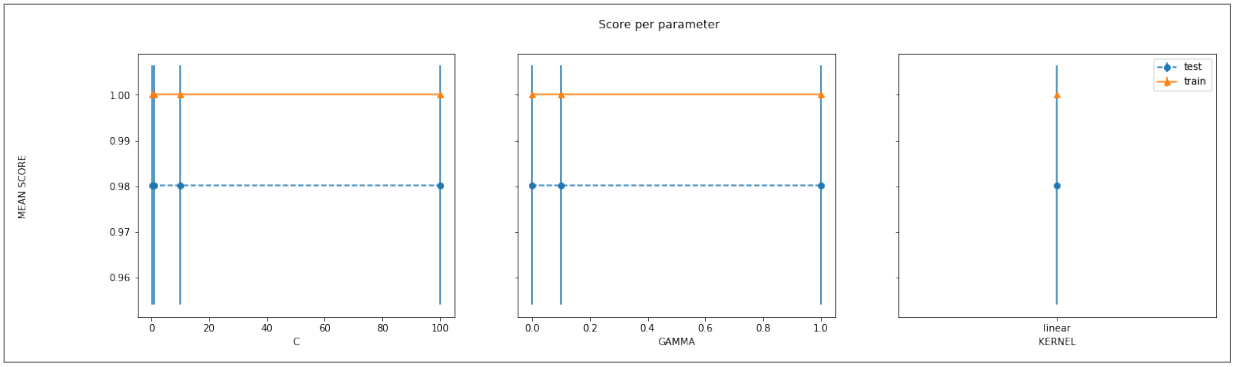
Fonte: A autora (2020)

Figura 38 – Score por parâmetro do RF - Banco 2 (Frutas Intactas)



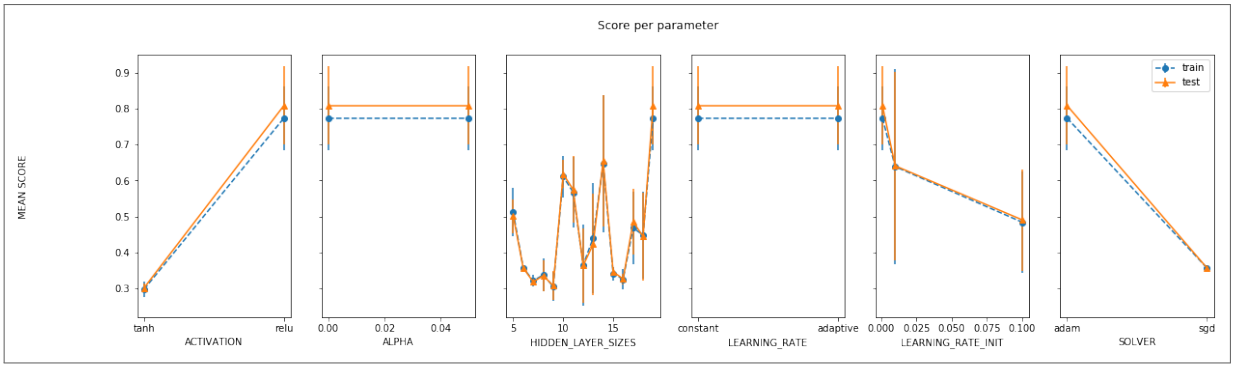
Fonte: A autora (2020)

Figura 39 – Score por parâmetro do SVM - Banco 2 (Frutas Intactas)



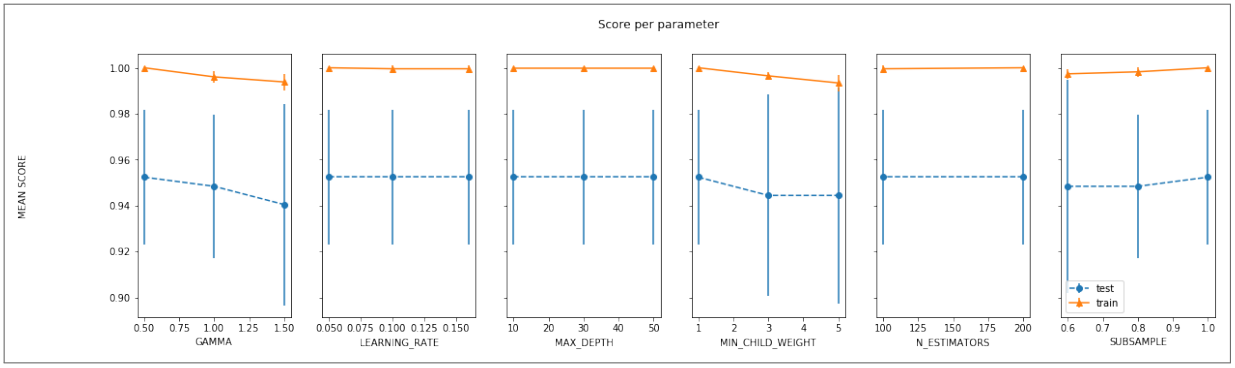
Fonte: A autora (2020)

Figura 40 – Score por parâmetro do MLP - Banco 2 (Frutas Intactas)



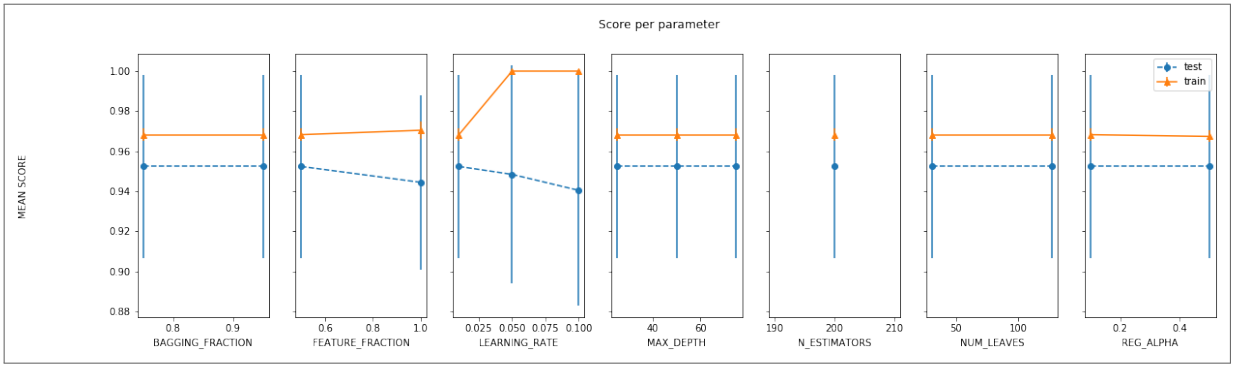
Fonte: A autora (2020)

Figura 41 – Score por parâmetro do GBOOST - Banco 2 (Frutas Intactas)



Fonte: A autora (2020)

Figura 42 – Score por parâmetro do LIGTH GBM - Banco 2 (Frutas Intactas)



Fonte: A autora (2020)

5.2.1 Resultados das classificações e discursões

A fim de investigar quais dos algoritmos têm melhor performance para classificar as respostas do sensor para cada fruta, alguns classificadores que estão sendo utilizados na literatura e que estão com destaques em competições foram utilizados. Após passar pelo processo de busca do melhor parâmetro, cada melhor classificador, apontado pelo o grid search, foi treinado e testado sobre os dois bancos de dados utilizados nesta pesquisa. As tabelas 15 e 16 apresentam os resultados dessas avaliações para as duas bases de dados, as métricas de avaliação utilizadas são: Acurácia treinamento, Acurácia de Teste, Precisão, Recall, F - Score e A curva ROC.

Tabela 15 – Resultados Métricas de Avaliação - Banco 1 (Frutas Cortadas)

KNN				
Acurácia Treinamento	89,28 %	**Acurácia de Teste**	81,25 %	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**

1-Banana	0.80	1.00	0.89	0.99
2- Maça	0.86	0.56	0.68	0.90
3- Laranja	0.90	0.86	0.88	0.95
4 - Uva	0.70	0.81	0.75	0.94
RF				
Acurácia Treinamento	100 %	**Acurácia de Teste**	82,63 %	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.85	1.00	0.92	0.99
2- Maça	0.77	0.68	0.72	0.92
3- Laranja	0.87	0.79	0.83	0.97
4 - Uva	0.79	0.84	0.82	0.96
SVM				
Acurácia Treinamento	95,23 %	**Acurácia de Teste**	86,11%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.92	0.94	0.93	0.99
2- Maça	0.78	0.91	0.84	0.94
3- Laranja	0.89	0.79	0.84	0.94
4 - Uva	0.87	0.81	0.84	0.94
MLP				
Acurácia Treinamento	47,61\%	**Acurácia de Teste**	43,05%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.70	1.00	0.82	0.93
2- Maça	0.29	0.79	0.42	0.60
3- Laranja	0.00	0.00	0.00	0.50
4 - Uva	0.00	0.00	0.00	0.50
XGBOOST				
Acurácia Treinamento	99,70%	**Acurácia de Teste**	86,80%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.85	0.94	0.89	0.99
2- Maça	0.81	0.74	0.77	0.95
3- Laranja	0.95	0.86	0.90	0.97
4 - Uva	0.86	0.94	0.90	0.98
LIGTH GBM				
Acurácia Treinamento	100%	**Acurácia de Teste**	83,33%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.84	0.91	0.88	0.99
2- Maça	0.71	0.74	0.72	0.92
3- Laranja	0.95	0.81	0.88	0.96

4 - Uva	0.82	0.88	0.85	0.97

Fonte : A autora (2020)

Tabela 16 – Resultados Métricas de Avaliação - Banco 2 (Frutas Intactas)

KNN				
Acurácia Treinamento	100%	**Acurácia de Teste**	98,14%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	1.00	0.95	0.97	0.99
2- Maça	1.00	1.00	1.00	1.00
3- Laranja	0.95	1.00	0.97	0.99
RF				
Acurácia Treinamento	99,60%	**Acurácia de Teste**	99,07%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	1.00	0.98	0.99	1.00
2- Maça	1.00	1.00	1.00	1.00
3- Laranja	0.97	1.00	0.99	1.00
SVM				
Acurácia Treinamento	100%	**Acurácia de Teste**	98,14%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	1.00	0.95	0.97	1.00
2- Maça	1.00	1.00	1.00	1.00
3- Laranja	0.95	1.00	0.97	0.99
MLP				
Acurácia Treinamento	67,06%	**Acurácia de Teste**	65,74%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.54	1.00	0.70	0.74
2- Maça	0.94	1.00	0.97	0.99
3- Laranja	0.00	0.00	0.00	0.50
XGBOOST				
Acurácia Treinamento	100%	**Acurácia de Teste**	98,14%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.98	0.98	0.98	1.00
2- Maça	1.00	1.00	1.00	1.00
3- Laranja	0.97	0.97	0.97	1.00
LIGTH GBM				

Acurácia Treinamento	97,22%	**Acurácia de Teste**	97,22%	
Classe	**Precisão**	**Recall**	**F1-Score**	**AUCROC**
1-Banana	0.98	0.98	0.98	0.99
2- Maça	1.00	1.00	1.00	1.00
3- Laranja	0.97	0.97	0.97	0.99

Fonte : A autora (2020)

Fazendo uma análise dos resultados dos algoritmos para cada base de dados, podemos observar que os resultados das métricas no banco 1 (Fruta Cortada), para a classe da maçã, foi a que obteve os resultados mais baixos, a métrica recall dessa classe, para todos os algoritmos, exceto o SVM, foi o que teve valor mais baixo, o que indica que existe uma alto valor de falsos positivos para essa classe. Comparado com os resultados dessa mesma classe para o banco 2 (Fruta Intacta), podemos observar que a maçã obteve os melhores resultados de recall em todos os algoritmos. A explicação para essa diferença nos resultados se dá porque o tecido epidérmico (casca) das maçãs produz uma maior quantidade de compostos orgânicos voláteis do que tecidos carnosos internos (pericarpo) (WILSON; BAIETTO, 2009), então, como a base 2 é composta pela percepção do sensor a fruta intacta, os resultados da maça têm uma representatividade melhor que no banco 1.

A laranja no banco de dados 1, foi a segunda classe com resultado baixo, em comparação com as outras classes. A banana foi a classe que obteve os melhores resultados do recall em todos os algoritmos.

Podemos observar que a base de dados 1 nos trouxe várias diferenças nas métricas para cada classe, enquanto no banco de dados 2 composto por dados de apenas três frutas como já explicado anteriormente, todas as classes obtiveram respostas satisfatórias para todos os algoritmos. Esse resultado satisfatório no banco 2, era esperado vendo que, as amostras das três frutas possuem impressões digitais dos odores bem distintas entre si, com isso esperava-se uma pecisão de classificação boa para essa base de dados.

Na base de dados 2, podemos observar que os resultados referente a acurácia tanto em treinamento como em teste para os algoritmos KNN, SVM e XGboost foram satisfatórias, variando apenas os valores das métricas.

Referente a acurácia no conjunto de teste (classificador é colocado em contato a amostras desconhecidas) podemos observar que para o banco de dados 1, o classificador que obteve melhor acurácia foi o XGBoost com 86,80%, seguido pelo SVM com acurácia de 86,11%. Na base de dados 2, o RF foi classificador que obteve melhor acurácia no conjunto de teste, 99,07%.

O algoritmo MLP foi o que obteve os piores resultados de classificação. Apesar de ter sido feita uma variaçao nos parâmetros do grid search, geralmente a MLP precisa de uma

quantidade considerável de dados para que consiga aprender mais com os dados e obter um melhor resultado de classificação.

Como descrito anteriormente, 30% do conjunto total de amostras foi usado para teste, totalizando 144 amostras para o banco 1, sendo 35 da banana, 34 da maça, 43 da laranja e 32 da uva. No banco 2, um total de 108 amostras, sendo 41 da banana, 30 da maçã e 37 da laranja. As tabelas 17 e 18 mostram as matrizes de confusão obtidas através do conjunto de amostra de teste classificadas para os quatro tipos de fruta (banco 1) e três tipos de fruta (banco 2) para cada algoritmo classificador.

Tabela 17 – Matriz de confusão para todos os classificadores - Banco 1 (Frutas Cortadas)

KNN				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	35	0	0	0
2- Maça	6	19	3	6
3- Laranja	0	1	37	5
4 - Uva	3	2	1	26
RF				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	35	0	0	0
2- Maça	4	23	4	3
3- Laranja	0	5	34	4
4 - Uva	2	2	1	27
SVM				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	33	2	0	0
2- Maça	1	31	0	1
3- Laranja	0	6	34	3
4 - Uva	1	1	4	26
MLP				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	35	0	0	0
2- Maça	7	27	0	0
3- Laranja	0	43	0	0
4 - Uva	8	24	0	0
XGBOOST				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	33	2	0	0

2- Maça	5	25	2	2
3- Laranja	0	3	37	3
4 - Uva	1	1	0	30
LIGHT GBM				
	1 - Banana	2- Maça	3 - Laranja	4 - Uva
1-Banana	32	3	0	0
2- Maça	5	25	2	2
3- Laranja	0	4	35	4
4 - Uva	1	3	0	28

Fonte : A autora (2020)

Tabela 18 – Matriz de confusão para todos os classificadores - Banco 2 (Frutas Intactas)

KNN			
	1-Banana	2- Maça	3- Laranja
1-Banana	39	0	2
2- Maça	0	30	0
3- Laranja	0	0	37
RF			
	1-Banana	2- Maça	3- Laranja
1-Banana	40	0	1
2- Maça	0	30	0
3- Laranja	0	0	37
SVM			
	1-Banana	2- Maça	3- Laranja
1-Banana	39	0	2
2- Maça	0	30	0
3- Laranja	0	0	37
MLP			
	1-Banana	2- Maça	3- Laranja
1-Banana	41	0	0
2- Maça	0	30	0
3- Laranja	35	2	0
XGBOOST			
	1-Banana	2- Maça	3- Laranja
1-Banana	40	0	1
2- Maça	0	30	0

3- Laranja	1	0	36
LIGHT GBM			
	1-Banana	2- Maça	3- Laranja
1-Banana	39	0	2
2- Maça	0	30	0
3- Laranja	1	0	39

Fonte : A autora (2020)

A tabela 17 mostra que a banana foi a classe que teve a quantidade de amostra melhor classificada em todos os algoritmos, apenas no XGboost e no LightGBM que ela teve duas e três amostras classificadas incorretamente no total de 35.

Podemos observar também que, dependendo do algoritmo, a classe da laranja foi a que obteve mais amotras classificadas incorretamente, no KNN, 6 das 43 amostras foram classificadas incorretamente, no SVM, mesmo sendo o que obteve melhor acúracia de classificação em teste, podemos observar que 9 das 43 amostras da laranja foram classificadas incorretamente. O Light GBM classificou 8 amostras da laranja incorretamente. O algoritmo que não conseguiu classificar nenhuma amostra da laranja corretamente foi o MLP. Para a maçã, o algoritmo KNN classificou corretamente apenas 19 das 35 amostras. A uva foi pior classificada em dois algoritmos além da MLP, o SVM e o KNN que classificou apenas 26 amostras das 32.

Na na tabela 18, composta pelas frutas intactas, a classe da banana teve apenas 2 amostras das 41 classificadas incoretamente nos algoritmos KNN, SVM e Light GBM. A maçã foi classificada corretamente por todos os algoritmos. O MLP foi o algoritmo que pior classificou a laranja, classificando corretamente apenas 2 das 37 amostras.

Em suma, a matriz de confusão nos mostra a análise da performace do modelo, o quanto de cada classe o modelo está acertando. No banco 1, vimos que os modelos classificaram de forma errada algumas classe, mas, se comparamos ao banco 2, onde todos os modelos tiveram desempenhos bons, podemos ver que o fato da a uva compor o banco 1, mostrou que os classificadores tiveram dificuldades na classificação. Provavelmente com uma massa de dados maior, seria possível ter um desempenho mais satisfatório para o conjunto de teste.

5.2.2 Comparação dos modelos

O teste estatístico de Kruskal-Wallis foi aplicado sobre os resultados obtidos no conjunto de treinamento executados durante a aplicaçao da validação cruzada para. Esse teste foi aplicado a fim de saber se há diferença estatística significante entre dois modelos. Logo, se a hipótese nula H_0 é verdadeira, de forma que não existe diferença significativa entre

os algoritmos utilizados. Se H_0 é rejeitada, então a hipótese alternativa H_1 é aceita. A $H1$ considera, portanto, que os algoritmos possuem diferença significativa.

O teste foi aplicado para os dois bancos de dados com um nível de significância $\alpha = 0.05$, o que indica que o resultado é 99,95% confiável. Para os bancos de dados 1 e 2, os resultados dos testes de hipótese, analisando o p-value, com $\alpha = 0.05$ foi de que as distribuições são diferentes, logo, rejeitou-se H_0. As Figuras 43 e 44 mostram os resultados dos p-value da comparação dos pares de algoritmos. Em laranja são os algoritmos considerados estatisticamente diferentes.

Figura 43 – Comparação dos Algoritmos - Banco 1 (Frutas Cortadas) - Valores de p-value

	KNN	RandomForest	SVM	RedeNeural	XGB	LGB
KNN	1	0.404084	0.471171	0.000155267	0.568415	0.969656
RandomForest	0.404084	1	0.111465	0.000155267	0.848752	0.342883
SVM	0.471171	0.111465	1	0.000155267	0.169552	0.568855
RedeNeural	0.000155267	0.000155267	0.000155267	1	0.000150871	0.000154381
XGB	0.568415	0.848752	0.169552	0.000150871	1	0.541835
LGB	0.969656	0.342883	0.568855	0.000154381	0.541835	1

Fonte : A autora (2020)

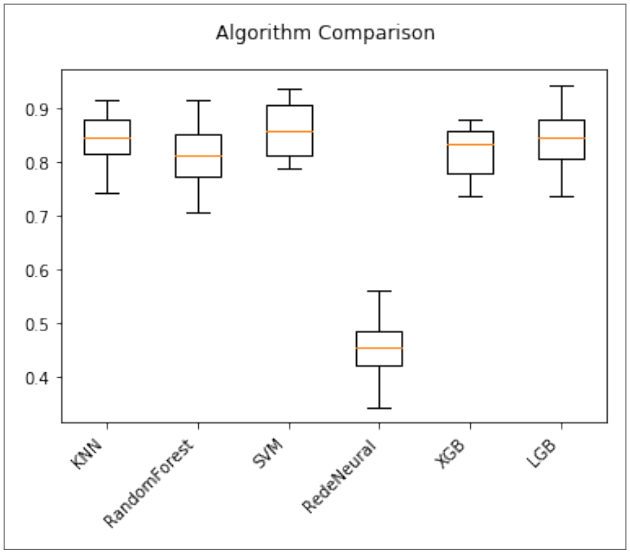
Figura 44 – Comparação dos Algoritmos - Banco 2 (Frutas Intactas) - Valores de p-value

	KNN	RandomForest	SVM	RedeNeural	XGB	LGB
KNN	1	0.968482	0.159123	0.000403549	0.391427	0.875379
RandomForest	0.968482	1	0.221682	0.000347354	0.391805	0.783823
SVM	0.159123	0.221682	1	0.000125654	0.0271707	0.118993
RedeNeural	0.000403549	0.000347354	0.000125654	1	0.000198533	0.000477507
XGB	0.391427	0.391805	0.0271707	0.000198533	1	0.670495
LGB	0.875379	0.783823	0.118993	0.000477507	0.670495	1

Fonte : A autora (2020)

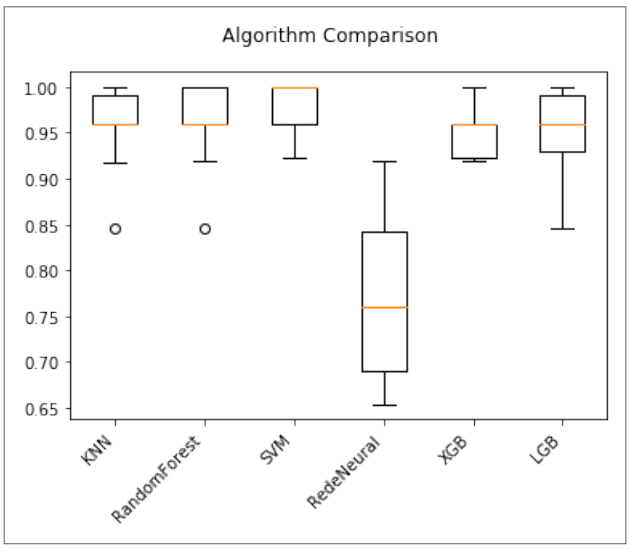
Para a interpretação e melhor visualização da variação das acurácias na valiação cuzada,foram gerados os gráficos boxplot dos dados para cada base. Em um boxplot são apresentados 5 estatísticas: o mínimo, o máximo, o primeiro quartil (Q1), a mediana, o terceiro quartil (Q3). O grande objetivo é analisar a distribuição dos dados. A observação desse gráfico vai nos mostrar as conclusões a respeito de: centro dos dados (a média ou mediana) das acurácias para cada algoritmo, a amplitude dos dados (máximo - mínimo) das acurácias, a simetria ou assimetria do conjunto de dados e a presença de outliers (os pontos fora das linhas desenhadas).

Figura 45 – Comparação dos Algoritmos - Banco 1 (Frutas Cortadas)



Fonte : A autora (2020)

Figura 46 – Comparação dos Algoritmos - Banco 2 (Frutas Intactas)



Fonte : A autora (2020)

Realizando a análise das figuras, 45 e 46 podemos observar que o algoritmo que possuiu a melhor média de acurácia para os dois bancos de dados foi o SVM. É possível observar também, que existem diferentes variabilidades das acurácias entre os algoritmos utilizados.

5.2.3 Comparação com outros trabalhos

Na literatura, são poucos os trabalhos que utilizam narizes eletrônicos para classificação de diferentes frutas. Adak e Yumusak (2016) usaram o nariz eletrônico MOSES II, para classificar frutas (morango, limão, cereja e melão). Na fase de treinamento, a rede neural com duas camadas ocultas foi otimizada usando algoritmo de colônia de abelhas artificial

(ABC), assim melhorando o desempenho da classificação. Os dados foram fornecidos a duas redes neurais diferentes, cada uma treinada separadamente com retropropagação (PA) e ABC. Obtendo um desenpenho de 60 % para a rede neural treinada com PA e 76,39 % para a rede neural treinada com ABC. Esse desempenho mostra que a rede neural treinada com ABC é bem-sucedida na classificação dos aromas.

Tang et al. (2010) desenvolveram um protótipo de um nariz eletrônico portátil, que compreende um conjunto de oito sensores (TGS2620, TGS826, TGS822, TGS825, TGS2602) disponíveis no mercado, uma interface de aquisição de dados PCB e um microprocessador. Os resultados indicaram que o protótipo proposto é capaz de identificar a fragrância de de três frutas (limão, banana e lichia). O protótipo proposto e o *software* de verificação foram testados e obtiveram uma precisão de classificação superior a 95%.

A fim de comparação, o dispositivo apresentado nesta pesquisa, além de ser composto por apenas um sensor, o que tornou ser um dispositivo de baixo custo, possui uma boa classificação e diferente dos outros estudos, foi capaz de classificar, com o auxilio de métodos avançados de aprendizagem de máquina, não só a fruta aberta, como também a fruta intacta. Obtendo 86,80% nos melhores dos casos para classificar as frutas maduras cortadas e uma acurácia de 98,14% para frutas maduras intactas. Esse protótipo mostrou não ser necessário um conjunto de sensores para detectar variedades de frutas. Demonstrando o seu diferencial em comparação com outras pesquisas, o nariz eletrônico foi capaz de produzir uma impressão de aromas para cada fruta, em seus estágios, ainda que cada fruta possua sua diferença de compostos orgânicos voláteis.

Diante dos resultados obtivos e do potencial apresentado pelo modelo computacional utilizado nesse trabalho, algumas evoluções são possíveis tendo-o como base. A tecnologia de narizes eletrônico vem ocupando espaço no mercado por sua portabilidade. O desenvolvimento desses aparelhos associados as técnicas de aprendizado de máquinas se tornam revolucionários na indústria da fruticultura. Será possível torná-lo reprodutível em outros cenários ou ambientes para que a sua escalabilidade dentro da fruticultura se torne mais vasta e acessível possível.

6 CONCLUSÃO E TRABALHOS FUTUROS

Os instrumentos de nariz eletrônico receberam considerável atenção no campo da tecnologia de sensores durante os últimos vinte anos, isso se deve em grande parte devido a descoberta de inúmeras aplicações derivadas de pesquisas em diversos campos das ciências aplicadas. Aplicações recentes de tecnologias de ponta eletrônica surgiram através de avanços no projeto de sensores, melhorias materiais, inovação de software e progresso no projeto de microcircuitos e integração de sistemas. O surgimento de novos tipos e matrizes de sensores de narizes eletrônicos, baseada em diferentes princípios e mecanismos de detecção, está estreitamente correlacionada com a expansão de novas aplicações. Os narizes eletrônicos fornecem uma infinidade de benefícios para uma variedade de indústrias comerciais, incluindo os setores agrícola, biomédico, cosmético, ambiental, alimentício, industrial, militar, e vários campos de pesquisa científica. Os avanços melhoraram os atributos, a uniformidade e a consistência do produto (BAIETTO; WILSON, 2015).

Com o objetivo de classificar quatro tipo de frutas (Banana, maçã, uva e laranja) de forma automática em dois estágios : frutas maduras intactas e cortadas, atráves da presença dos gases exalados por elas, o protótipo do nariz eletrônico de baixo custo, composto apenas por um sensor, o TGS822 da Figaro, obteve-se êxito na classificação das frutas nos seus dois estágios. Assim, é possível ter um dispositivo de baixo custo para classificar diferentes espécies de frutas.

Como apresentado nessa dissertação, a uva intacta não conseguiu ser detectada pelo sensor TGS822 da figaro. Isso se deu porque apesar dos compostos orgânicos voláteis da uva estarem em grande parte na casca, esse fruta é subdividida em duas espécies, as aromáticas (uvas roxas) e as pouco aromáticas, no caso do trabalho em questão, a uva utilizada foi a verde. O que não tirou o êxito da classificação das demais frutas, das 360 amostras dos três tipo de frutas, o resultado de acerto total na validação chegou a o máximo e no teste obteve acurácia de 99,07%.

As 480 amostras das 4 frutas cortadas, com resultados de acerto total na validação de 99,70 % e de 86,80% no seu melhor classificador, demostraram a adequação do nariz eletrônico para a classificação de variedades de frutas cortadas.

De modo geral, cinco dos seis classificadores apresentaram acurácias de teste consideradas boas ou ótimas. Apenas a MLP teve um resultado não satisfatório com acurácia de teste no banco 01 de 43,05% e no banco 2 de 65,74%. Uma possível solução para isso, pode ser a obtenção futura de mais dados para os bancos utilizados, já que na MLP espera uma massa de dados grande para melhorar o processo de aprendizagem.

O equipamento utilizado na pesquisa, mostrou-se promissor, demostrando ter sensibilidade para a deteção de COVs emitidos pelas frutas em dois estágios: frutas maduras intactas e cortadas. Portanto, o trabalho mostra que apenas com um único sensor é possí-

vel criar as impressões de aromas da frutas e, com métodos avançados de aprendizagem de máquinas, classificá-las. Esse trabalho também mostra que não é necessário uma matriz de sensores para obter os objetivos dessa dissertação.

6.1 TRABALHOS FUTUROS

Esse trabalho pode ser ampliado em:

- Ampliação da base de dados para obter melhor aprendizado em algoritmos que tiveram resultados não satisfatórios.
- Realizar estudos sobre técnicas de pré-processamento do sinal obtido pelo sensor, visando extrair informações relevantes para aprimorar a classificação de frutas.
- Realizar reconstrução de modelos, treinando com outras espécies de frutas.
- Utilizar o modelo para o monitoramento de estágio de maturação das frutas já classificadas nesse trabalho.
- Automatização do processo da obtenção dos dados.
- Utilizar outros métodos avançados de aprendizagem de máquina para verificar o comportamento da classificação.
- Construir novos modelos de narizes eletrônicos com outros sensores.

REFERÊNCIAS

ADAK, M. F.; YUMUSAK, N. Classification of e-nose aroma data of four fruit types by abc-based neural network. *PMC - US National Library of Medicine National Institutes of Health Search databaseSearch term Search*, v. 16, p. 304, 2016.

ALBERT, K. J.; LEWIS, N. S.; SCHAUER, C. L.; SOTZING, G. A.; STITZEL, S. E.; VAID, T. P.; WALT, D. R. Cross-reactive chemical sensor arrays. *ACSPublications*, v. 100, p. 2595–2626, 2000.

ARAUJO, G. S. *Uso de Random Forests e Redes Biológicas na Associação de Polimorfismos à Doença de Alzheimer*. Dissertação (Mestrado) — Universidade Federal de Pernabuco - Centro de Informática, Recife - PE, 2013.

ARCHER, P.; CHADWICK, A.; MIASIK, J.; M.TAMIZI; WRIGHT, J. Kinetic factors in the response of organometallic semiconductor gas sensors. *ScienceDirect*, v. 488, p. 379–392, 1989.

A.SANAEIFAR; MOHTASEBI, S.; VARNAMKHSTI, M. G.; AHMADI, H.; ROGADO, J. S. L. Development and application of a new low cost electronic nose for the ripeness monitoring of banana using computational techniques (pca, lda, simca, and svm). *Czech Academy of Agricultural Sciences,*, 2014.

BAIETTO, M.; WILSON, A. Electronic-nose applications for fruit identification, ripeness and quality grading. *PMC - US National Library of Medicine National Institutes of Health Search databaseSearch term Search*, v. 15, p. 899–931, 2015.

BANSAL, A.; KAUR, S. Extreme gradient boosting based tuning for classification in intrusion detection systems. *Springer Nature Singapore*, 2018.

BENADY, M.; SIMON, J.; CHARLES, D.; MILES, G. E. Fruit ripeness determination by electronic sensing or aromatic volatiles. *Trans. ASAE.*, p. 251–257, 1995.

BUCK, T.; ALLEN, F.; DALTON, J. V. *Detection of chemical species by surface effects on metals and semiconductors. In Surface effects in detection*. Washington, D.C., USA: Bell Telephone Laboratories, 1965.

CALIXTO, R. R.; ARAGãO, M. F.; RODRIGUES, A. B.; NETO, L. G. P.; CAVALCANTE, T. S. Sistema de visão computacional para classificação de melão amarelo de acordo com o formato. *SBrT*, 2016.

CARVALHO, J. N.; GURJAO, E. C.; MATA, M. E. R. M. M. C.; DUARTE, M. E. M. ClassificaÇÃo automÁtica de frutas por anÁlise de imagem – o caso da manga tommy atkins. *WTA 2014 – VIII Workshop de Tecnologia Adaptativa*, 201.

CHAMBAZ, E. M.; DEFAYE, G.; MADANI, C. Application of gas chromatography, mass spectrometry, and computer methods in clinical biochemistry. *ACSPublications*, v. 45, p. 1099–1106, 1973.

CHEN, L. Y.; WU, C. C.; CHOU, T. I.; CHIU, S. W.; TANG, K. T. Development of a dual mos electronic nose/camera system for improving fruit ripeness classification. *PMC - US National Library of Medicine National Institutes of Health Search databaseSearch term Search*, v. 18, n. 10, p. 3256, 2018.

DRAVNIEKS, A.; TROTTER, P. J. Polar vapour detector based on thermal modulation of contact potential. *IOPScience - Journal of Scientific Instruments*, v. 42, p. 624–627, 1965.

FACELI, K.; LORENA, A.; GAMA, J.; CARVALHO, A. C. P. L. F. *Inteligência Artificial Uma Abordagem de Apredizado de Máquina*. Rio de Janeiro: LTC- Livros Técnos e Científicos Editora Ltda, 2011. ISBN 978-85-216-1880-5.

FILHO, L. C. S. F. *Desenvolvimento de sensores eletroquímicos Utilizando Nanomateriais para a Determinação de Espécies de Interesse Farmacêutico e Ambiental*. 193 p. Dissertação (Mestrado) — Universidade Federal de São Carlos, São Carlos - SP, 2014.

FREITAS, M. A. *Classificador LGBM Aplicado na Resolução do Desafio PLAsTiCC do Telescópio LSST*. 68 p. Dissertação (Mestrado) — Universidade Federal de Uberlândia, Uberlândia, 2019.

F.ROCK; N.BARSAN; WEIMAR, U. Electronic nose: Current status and future trends. *ACS Publications*, v. 108, n. 2, p. 705–725, 2008.

GARCIA, M.; ALEIXANDRE, M.; HORRILLO, M. Electronic nose for the identification of spoiled iberian hams. *IEEExplore Digital Library*, p. 537–540, 2005.

GOMES, P. C. T. Conheça o algoritmo xgboost. *DataGeeks*, 2019.

GOMEZ, A. H.; WANG, J.; HU, G.; PEREIRA, A. G. Electronic nose technique potential monitoring mandarin maturity. *ScienceDirect*, v. 113, n. Issue 1, p. 347–353, 2006.

G.SUJATHA; DHIVYA, N.; AYYADURAI, K.; D.THYAGARAJAN. Advances in electronic - nose technologies. *International Journal of Engineering Research and Applications (IJERA)*, v. 2, n. Issue4, p. 1541–1546, 2012.

GóMEZ, A.; HU, G.; WANG, J.; PEREIRA, A. G. Evaluation of tomato maturity by electronic nose. *ScienceDirect*, v. 54, n. Issue1, p. 44–52, 2006.

HARPER, W. J. The strengths and weaknesses of the electronic nose. *Headspace Analysis of Foods and Flavors*, v. 16, n. 4, p. 59–71, 2001.

HUANG, C. H.; ZENG, C.; WANG, Y. C.; .PENG, H. Y.; LIN, C. S.; CHANG, C. J.; YANG, H. Y. A study of diagnostic accuracy using a chemical sensor array and a machine learning technique to detect lung cancer. *PublMed.gov - US National Library of Medicine National Institutes of Health*, v. 18, p. 2845–2018, 2018.

IKEGAMI, A.; KANEYASU, M. Olfactory detection using integrated sensors. proceedings of the 3rd international conference on solid-state sensors and actuators. New York, NY, USA, p. 136–139, 1985.

JANZANNTTI, N. S.; FRACO, M. R. B.; LANCAS, F. M. Identificação de compostos voláteis de maçãs (malus domestica) cultivar fuji, por cromatografia gasosa-espectrometria de massas. *SciELO Network Meeting - Internacional Conference*, v. 20, n. 2, p. 164–171, 2000.

JUNIOR, W. J. de A. *Métodos De Otimização Hiperparamétrica: Um Estudo Comparativo Utilizando Árvores De Decisão E Florestas Aleatórias Na Classificação Binária*. 60 p. Dissertação (Mestrado) — Universidade Federal de Minas Gerais, Belo Horizonte, 2018.

LI, C.; HEINEMANN, P. H.; IRUDAYARAJ, J. Detection of apple defects using an electronic nose and znose. *Transactions of the ASABE*, v. 50, n. 4, p. 1417–1425, 2007.

LISBOA, H.; PAGE, T.; GUY, C. Gestão de odores: fundamentals of electronic nose. *SciELO*, v. 14, n. 1, p. 1–18, 2009.

LIU, X.; CHENG, S.; LIU, H.; HU, S.; ZHANG, D.; NING, H. A survey on gas sensing technology. *MDPI*, v. 12, n. 7, p. 9635—-9665, 2012.

LORENA, A.; CARVALHO, A. Uma introdução às support vector machines. *RITA - Revista de Informática Teória e Aplicada*, v. 14, 2007.

LORWONGTRAGOOL, P.; SOWADE, E.; WATTHANAWISUTH, N.; BAUMANN, R. R.; KERDCHAROEN, T. A novel wearable electronic nose for healthcare based on flexible printed chemical sensor array. *SemanticScholar*, v. 14, p. 19700—-19712, 2014.

MARQUES, V. G. O. *Avaliação do desempenho das redes neurais convolucionais na detecção de ovos de esquistossomose*. Dissertação (Mestrado) — UNIVERSIDADE FEDERAL DE PERNAMBUCO - CENTRO DE INFORMÁTICA, Recife, 2017.

MARTINS, G. *Identificação de plásticos comerciais por meio de um nariz eletrônico baseado em polímeros condutores*. Dissertação (Mestrado) — USP - Universidade de São Paulo, São Paulo, 2011.

MCKELLAR, R. C.; RUPASINGHE, H. V.; LU, X.; KNIGHT, K. P. The electronic nose as a tool for the classification of fruit and grape wines from different ontario wineries. *Journal of the Science of Food and Agriculture*, v. 85, p. Issue 14, 2005.

MEDIUM. *What is LightGBM, How to implement it? How to fine tune the parameters?* 2017. Disponível em: <https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-pa
Acesso em: 04 jan. 2020.

MUSTAPHA, I. B.; SAEED, F. Bioactive molecule prediction using extreme gradient boosting. *Molecules*, 2016.

MUTALIB, N. A. A.; JASWIR, I.; AKMELIAWATI, R.; IBRAHIM, A. M.; ASLAM, M.; OCTAVIANTI, F. Rapid detection of ethanol in beverages using iium-fabricated electronic nose. *International Food Research Journal, Faculty of Food Science and Technology*, v. 24, p. n.suppl, 2017.

NATALE, C. D.; MACAGNANO, A.; MARTINELLI, E.; PAOLESSE, R.; PROIETTI, E.; D'AMICO, A. The evaluation of quality of post-harvest oranges and apples by means of an electronic nose. *ScienceDirect*, Issues 1-3, n. 78, p. 26–31, 2001.

OLIVEIRA, A. I. de. *ESTUDO DA RELAÇÃO ENTRE O CAMPO MAGNÉTICO E A INTENSIDADE DE RAIOS CÓSMICOS NO MEIO INTERPLANETÁRIO VIA REDES NEURAIS*. Tese (Doutorado), 02 2013.

OLIVEIRA, G. M. de. *Máquina de Vetores Suporte: estudo e análise de parâmetros para otimização de resultado*. 46 p. Dissertação (Mestrado) — Universidade Federal de Pernambuco, Recife, 2010.

PALOMO, E. S.; MAROTO, M. C. D.; COELLO, M. S. P. Rapid determination of volatile compounds in grapes by hs-spme coupled with gc–ms. *ScienceDirect*, v. 66, n. 5, p. 1152–1157, 2005.

PERSAUD, K.; DODD, G. Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose. *Nature - international Journal os science*, v. 299, n. Issue4, p. 352—-355, 1982.

PINHEIRO, C. L. *Estudo e Identificação de Compostos Orgânicos Voláteis Emitidos durante o Processo de Secagem de Bagaço de Laranja*. Dissertação (Mestrado) — UNIVERSIDADE ESTADUAL DE CAMPINAS-FACULDADE DE ENGENHARIA QUÍMICA-DEPARTAMENTO DE ENGENHARIA DE PROCESSOS, Campinas, 2015.

QUADROS, M.; CARMO, G. N. R.; FILHO, P. B.; HOINASKI, L.; OLIVEIRA, V. V.; CAMARA, V.; SCHIRMER, W. N.; GODKE, M. Metodologias olfatométricas para avaliação do impacto odorante. *LCQAr*, 2010.

RAGAZZO-SANCHEZ, J.; CHALIER, P.; CHEVALIER, D.; .CALDERON-SANTOYO, M.; GHOMMIDH, C. Identification of different alcoholic beverages by electronic nose coupled to gc. *ScienceDirect*, v. 134, p. 43–48, 2008.

RAJAMäKI, T.; ALAKOMI, H. L.; RITVANEN, T.; SKYTTä, E.; SMOLANDER, M.; AHVENAINEN, R. Application of an electronic nose for quality assessment of modified atmosphere packaged poultry meat. *ScienceDirect*, v. 17, n. Issue1, p. 5–13, 2006.

REITENBACH, A. *DESENVOLVIMENTO DE NARIZ ELETRÔNICO PARA DETECÇÃO DE COMPOSTOS VOLÁTEIS NA CERVEJA*. Dissertação (Mestrado) — Universidade de Santa Catarina - Departamento de Engenharia de Química e de Engenharia de Alimentos, Santa Catarina, 2016.

ROSSI, A. L. D. *Ajuste de parâmetros de técnicas de classificação por algoritmos bioinspirados*. Dissertação (Mestrado) — UNIVERSIDADE DE SAO PAULO, São Paulo, 2009.

SANTOS, R. H. S. *"Predição da resposta de Pacientes a Terapias Anti-HIV Através de Aprendizagem de Máquina*. 104 p. Dissertação (Mestrado) — Universidade Federal de Pernabuco - Centro de Informática, Recife - PE, 2014.

SBRUZZI, G.; PLENTZ, R. D. M. Bases da estimulação elétrica neuromuscular. *Profisio*, 2018.

SCHALLER, E.; BOSSET, J. O.; ESCHER, F. 'electronic noses' and their application to food. *ScienceDirect*, v. 31, n. 4, p. 305–316, 1998.

SEEMA; KUMAR, A.; GILL, G. Automatic fruit grading and classification system using computer vision: A review. *IEEE Xplore*, 2015.

TAN, P. N.; STEINBACH, M.; KUMAR, V. *Introduction to Data Mining*. [S.l.]: Pearson Addison Wesley, 2006, 2006. ISBN 0321321367, 9780321321367.

TANG, K. T.; CHIU, S. W.; PAN, C. H.; HSIEH, H. Y.; LIANG, Y. S.; LIU, S. C. Development of a portable electronic nose system for the detection and classification of fruity odors. *PMC - US National Library of Medicine National Institutes of Health Search databaseSearch term Search*, v. 10, p. 9179–9193, 2010.

TIAN, H.; WANG, T.; LIU, Y.; QIAO, X.; LI, Y. Computer vision technology in agricultural automation—a review. *ScienceDirect*, p. 2214–3173, 2019.

VOSS, A.; WITT, K.; FISCHER, C.; REULECKE, S.; POITZ, W.; KECHAGIAS, V.; SURBER, R.; FIGULLA, H. Smelling heart failure from human skin odor with an electronic nose. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, v. 45, p. 4034—-4037, 2012.

VOSS, H. G. J. *Desenvolvimento de um Nariz Eletrônico Aplicado à Determinação do Estágio da Maturação em Pessegueiros*. Dissertação (Mestrado) — Universidade Estadual de Ponta Grossa, Ponta Grossa, 2019.

WILSON, A.; BAIETTO, M. Applications and advances in electronic-nose technologies. *MDPI*, v. 9, n. 7, p. 5099–5148, 2009.

WOHLTJEN, H.; DESSY, R. Surface acoustic wave probe for chemical analysis. i. introduction and instrument description. *ACSPublications*, v. 51, p. 1458–1464, 1979.

XIAO, Z. W. snd X.; WANG, J.; WANG, H. Identification of the rice wines with different marked ages by electronic nose coupled with smartphone and cloud storage platform. *PMC - US National Library of Medicine National Institutes of Health*, v. 17, n. 11, p. 2500, 2017.

YAMUNA, S.; DEVARAJAN, B. W.; ZOECKLEIN, K.; GARDNER, D. M. Electronic nose evaluation of the effects of canopy side on cabernet franc (vitis vinifera l.) grape and wine volatiles. *American Journal of Enology and Viticulture*, v. 63, p. 73–80, 2011.

ZAKARIA, A.; SHAKAFF, A. Y. M.; MASNAN, M. J.; SAAD, F. S. A.; ADOM, A. H.; AHMAD, M. N.; JAAFAR, M. N.; ABDULLAH, A. H.; KAMARUDIN, L. M. Improved maturity and ripeness classifications of magnifera indica cv. harumanis mangoes through sensor fusion of an electronic nose and acoustic sensor. *PMC - US National Library of Medicine National Institutes of Health Search databaseSearch term Search*, v. 12, p. 6023–6048, 2012.

ZHU, D.; REN, X.; WEI, L.; CAO, X.; GE, Y.; LIU, H.; LI, J. Collaborative analysis on difference of apple fruits flavour using electronic nose and electronic tongue. *ScienceDirect*, v. 260, 2020.

APÊNDICE A – CÓDIGO PHYTON METÓDOS AVANÇADOS DE CLASSIFICAÇÃO

```

Importacao das Bibliotecas


# Commented out IPython magic to ensure Python compatibility.
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from sklearn.model_selection import GridSearchCV, StratifiedKFold
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
!pip install scikit-plot
from scikitplot.metrics import plot_roc
import matplotlib.pyplot as plt
from sklearn.model_selection import cross_val_score
from sklearn.ensemble import RandomForestClassifier
# %matplotlib inlin
from scikitplot.metrics import plot_roc
import matplotlib.pyplot as plt
from sklearn import svm
import xgboost as xgb
import lightgbm as lgb
from sklearn import tree

"""Carregar Banco de dados"""

from google.colab import files
def getLocalFiles():
    _files = files.upload()
    if len(_files) >0:
       for k,v in _files.items():
         open(k,'wb').write(v)
getLocalFiles()

X = df.drop('Classe', axis=1)

Y = df['Classe']

"""Divisaodo Banco de dados 70% treinamento 30% testes"""

```

```
43 X_train, X_test, y_train, y_test = train_test_split(X,Y, random_state=7,
       test_size=0.3 )

45 CV = StratifiedKFold(10, random_state=42)

47 """Grid Search - KNN"""

49 knn = KNeighborsClassifier()

51 par_grid_knn = {'n_neighbors':[4],
                  'weights' : ['uniform', 'distance'],
53                'metric' : ['euclidean', 'manhattan'],
                  'leaf_size':[1,3,5],
55                'algorithm':['auto', 'kd_tree'],'n_jobs':[-1]}

57 grid = GridSearchCV(knn, par_grid_knn, cv=CV,n_jobs=-1, return_train_score=
       True, scoring='accuracy')

59 proba_KNN = grid.fit(X_train,y_train).predict_proba(X_test)

61 grid.best_estimator_

63 grid.best_score_

65 grid.best_index_

67 grid.best_params_

69 resultados = pd.DataFrame(grid.cv_results_)
   resultados.head()
71
   scores = cross_val_score(grid, X_train, y_train, cv = CV)
73
   predicao = grid.predict(X_test)
75
   print("Acuracia Grid _ KNN: Treinamento", grid.score(X_train, y_train),"
       Teste" ,grid.score(X_test, y_test))
77 print("Clasification report:", classification_report(y_test, predicao))
   print("Confussion matrix:", confusion_matrix(y_test, predicao))
79
   """Plot curva ROC"""
81
   plot_roc(y_test, proba_KNN)
83 plt.show()

85 """Plot Varia  o dos Resultados"""
```

```
#def plot_search_results(grid):
"""
Params:
    grid: A trained GridSearchCV object.
"""
## Results from grid search
results = grid.cv_results_
means_test = results['mean_test_score']
stds_test = results['std_test_score']
means_train = results['mean_train_score']
stds_train = results['std_train_score']

## Getting indexes of values per hyper-parameter
masks=[]
masks_names= list(grid.best_params_.keys())
for p_k, p_v in grid.best_params_.items():
    masks.append(list(results['param_'+p_k].data==p_v))

params=grid.param_grid

## Ploting results
fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',figsize
    =(20,5))
fig.suptitle('Score per parameter')
fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
pram_preformace_in_best = {}
for i, p in enumerate(masks_names):
    m = np.stack(masks[:i] + masks[i+1:])
    pram_preformace_in_best
    best_parms_mask = m.all(axis=0)
    best_index = np.where(best_parms_mask)[0]
    x = np.array(params[p])
    y_1 = np.array(means_test[best_index])
    e_1 = np.array(stds_test[best_index])
    y_2 = np.array(means_train[best_index])
    e_2 = np.array(stds_train[best_index])
    ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='test')
    ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='train' )
    ax[i].set_xlabel(p.upper())


plt.legend()
plt.show()

"""Grid Search - Random Florest"""

rfc = RandomForestClassifier(random_state=7)
```

```
133
par_grid_rf = {
135             "criterion": ['entropy', 'gini'],
                "n_estimators": [25, 50, 100],
137             "bootstrap": [False, True],
                "max_depth": [3, 5, 10],
139             "max_features": ['auto', 0.1, 0.2, 0.3]
}
141
classificador_rf = GridSearchCV(rfc, par_grid_rf, cv=CV,n_jobs=-1,
    return_train_score=True, scoring='accuracy')
143
proba_RF = classificador_rf.fit(X_train,y_train).predict_proba(X_test)
145
grid.best_estimator_
147
grid.best_score_
149
grid.best_index_
151
grid.best_params_
153
resultados = pd.DataFrame(grid.cv_results_)
155 resultados.head()

157 scores1 = cross_val_score(classificador_rf, X_train, y_train, cv = CV)

159 predicao1 = classificador_rf.predict(X_test)

161 print("Acuracia RF: Treinamento", classificador_rf.score(X_train, y_train)
    ," Teste" ,classificador_rf.score(X_test, y_test))
    print("Clasification report:", classification_report(y_test, predicao1))
163 print("Confussion matrix:", confusion_matrix(y_test, predicao1))

165 """Plot curva ROC"""

167 plot_roc(y_test, proba_RF)
    plt.show()
169
"""Plot Varia  o dos Resultados"""
171
#def plot_search_results(grid):
173     """
        Params:
175         grid: A trained GridSearchCV object.
        """
177     ## Results from grid search
```

```
    results = classificador_rf.cv_results_
    means_test = results['mean_test_score']
    stds_test = results['std_test_score']
    means_train = results['mean_train_score']
    stds_train = results['std_train_score']

    ## Getting indexes of values per hyper-parameter
    masks=[]
    masks_names= list(classificador_rf.best_params_.keys())
    for p_k, p_v in classificador_rf.best_params_.items():
        masks.append(list(results['param_'+p_k].data==p_v))

    params=classificador_rf.param_grid

    ## Ploting results
    fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',
        figsize=(20,5))
    fig.suptitle('Score per parameter')
    fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
    pram_preformace_in_best = {}
    for i, p in enumerate(masks_names):
        m = np.stack(masks[:i] + masks[i+1:])
        pram_preformace_in_best
        best_parms_mask = m.all(axis=0)
        best_index = np.where(best_parms_mask)[0]
        x = np.array(params[p])
        y_1 = np.array(means_test[best_index])
        e_1 = np.array(stds_test[best_index])
        y_2 = np.array(means_train[best_index])
        e_2 = np.array(stds_train[best_index])
        ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='test
            ')
        ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='train'
            )
        ax[i].set_xlabel(p.upper())

    plt.legend()
    plt.show()

"""Grid Search - SVM"""

classifier = svm.SVC(random_state=100, probability=True)

par_grid_SVM = {'C': [0.1,1,10,100], 'gamma': [1,0.1,0.001], 'kernel': ['
    linear']}

CV = StratifiedKFold(10, random_state=42)
```

```
221
classificador_SVM = GridSearchCV(classifier , par_grid_SVM,refit=True,
    verbose=3, cv=CV, n_jobs=-1, return_train_score=True, scoring='accuracy'
    , )
223
proba_SVM = classificador_SVM.fit(X_train,y_train).predict_proba(X_test)
225
grid.best_estimator_
227
grid.best_score_
229
grid.best_index_
231
grid.best_params
233
resultados = pd.DataFrame(grid.cv_results_)
235 resultados.head()

237 scores2 = cross_val_score(classificador_SVM, X_train, y_train, cv = CV)

239 predicao2 = classificador_SVM.predict(X_test)

241 print("Acuracia SVM: Treinamento", classificador_SVM.score(X_train,
    y_train)," Teste" ,classificador_SVM.score(X_test, y_test))
    print("Clasification report:", classification_report(y_test, predicao2))
243 print("Confussion matrix:", confusion_matrix(y_test, predicao2))

245 """Plot curva ROC"""

247 plot_roc(y_test, proba_SVM)
    plt.show()
249
"""Plot varia  o dos resultados"""
251
#def plot_search_results(grid):
253     """
        Params:
255         grid: A trained GridSearchCV object.
        """
257     ## Results from grid search
        results = classificador_SVM.cv_results_
259     means_test = results['mean_test_score']
        stds_test = results['std_test_score']
261     means_train = results['mean_train_score']
        stds_train = results['std_train_score']
263
        ## Getting indexes of values per hyper-parameter
```

```
265     masks=[]
        masks_names= list(classificador_SVM.best_params_.keys())
267     for p_k, p_v in classificador_SVM.best_params_.items():
            masks.append(list(results['param_'+p_k].data==p_v))
269
        params=classificador_SVM.param_grid
271
        ## Ploting results
273     fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',
            figsize=(20,5))
        fig.suptitle('Score per parameter')
275     fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
        pram_preformace_in_best = {}
277     for i, p in enumerate(masks_names):
            m = np.stack(masks[:i] + masks[i+1:])
279         pram_preformace_in_best
            best_parms_mask = m.all(axis=0)
281         best_index = np.where(best_parms_mask)[0]
            x = np.array(params[p])
283         y_1 = np.array(means_test[best_index])
            e_1 = np.array(stds_test[best_index])
285         y_2 = np.array(means_train[best_index])
            e_2 = np.array(stds_train[best_index])
287         ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='test
                ')
            ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='train'
                 )
289         ax[i].set_xlabel(p.upper())

291     plt.legend()
        plt.show()
293

"""Grid Search MLP"""
295

MLP =  MLPClassifier(random_state=42, max_iter=1000, early_stopping=True)
297

par_grid_mlp = {
299     'hidden_layer_sizes': np.arange(5, 20),
        'activation': ['tanh', 'relu'],
301     'solver': ['adam', 'sgd'],
        'alpha': [0.0001, 0.05],
303     'learning_rate': ['constant','adaptive'],
        'learning_rate_init':[0.001,0.01,0.1],
305 }

307 CV = StratifiedKFold(10, random_state=42)
```

```
309 classificador_mlp = GridSearchCV(MLP, par_grid_mlp, cv=CV,n_jobs=-1,
        return_train_score=True, scoring='accuracy')

311 proba = classificador_mlp.fit(X_train,y_train).predict_proba(X_test)

313 grid.best_estimator_

315 grid.best_score_

317 grid.best_index_

319 grid.best_params_

321 resultados = pd.DataFrame(grid.cv_results_)
    resultados.head()
323
    predicao3 = classificador_mlp.predict(X_test)
325
    print("Acuracia MLP: Treinamento", classificador_mlp.score(X_train,
        y_train)," Teste" ,classificador_mlp.score(X_test, y_test))
327 print("\nClasification report:\n", classification_report(y_test, predicao3)
        )
    print("Confussion matrix:\n", confusion_matrix(y_test, predicao3))
329
    """Plot Curva ROC"""
331
    plot_roc(y_test, proba)
333 plt.show()

335 """Plot varia  o dos resultados"""

337 #def plot_search_results(grid):
    """
339 Params:
        grid: A trained GridSearchCV object.
341 """
    ## Results from grid search
343 results = classificador_mlp.cv_results_
    means_test = results['mean_test_score']
345 stds_test = results['std_test_score']
    means_train = results['mean_train_score']
347 stds_train = results['std_train_score']

349 ## Getting indexes of values per hyper-parameter
    masks=[]
351 masks_names= list(classificador_mlp.best_params_.keys())
    for p_k, p_v in classificador_mlp.best_params_.items():
```

```
353     masks.append(list(results['param_'+p_k].data==p_v))

355 params=classificador_mlp.param_grid

357 ## Ploting results
    fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',figsize
        =(20,5))
359 fig.suptitle('Score per parameter')
    fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
361 pram_preformace_in_best = {}
    for i, p in enumerate(masks_names):
363     m = np.stack(masks[:i] + masks[i+1:])
        pram_preformace_in_best
365     best_parms_mask = m.all(axis=0)
        best_index = np.where(best_parms_mask)[0]
367     x = np.array(params[p])
        y_1 = np.array(means_test[best_index])
369     e_1 = np.array(stds_test[best_index])
        y_2 = np.array(means_train[best_index])
371     e_2 = np.array(stds_train[best_index])
        ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='train')
373     ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='test' )
        ax[i].set_xlabel(p.upper())
375
    plt.legend()
377 plt.show()

379 """Grid Search - XGBOOST"""

381 model = xgb.XGBClassifier()

383 param_dist = {"max_depth": [10,30,50],
              "min_child_weight" : [1,3,5],
385           "n_estimators": [200, 100],
                "learning_rate": [0.05, 0.1, 0.16],
387             'gamma': [0.5, 1, 1.5],
                 'subsample': [0.6, 0.8, 1.0]}
389
    CV = StratifiedKFold(10, random_state=42)
391
    classificador_xg = GridSearchCV(model, param_dist, cv=CV,n_jobs=-1,
        return_train_score=True, scoring='accuracy')
393 proba_xg = classificador_xg.fit(X_train,y_train).predict_proba(X_test)

395 grid.best_estimator_

397 grid.best_score_
```

```
399 grid.best_index_

401 grid.best_params_

403 resultados = pd.DataFrame(grid.cv_results_)
    resultados.head()
405
    scores4 = cross_val_score(classificador_xg, X_train, y_train, cv = CV)
407
    predicao4 = classificador_xg.predict(X_test)
409
    print("Acuracia GBOOST: Treinamento", classificador_xg.score(X_train,
        y_train)," Teste" ,classificador_xg.score(X_test, y_test))
411 print("Clasification report:", classification_report(y_test, predicao4))
    print("Confussion matrix:", confusion_matrix(y_test, predicao4))
413
    """Plot Curva ROC"""
415
    plot_roc(y_test, proba_xg)
417 plt.show()

419 """Plot varia  o dos resultados"""

421 #def plot_search_results(grid):
        """
423     Params:
            grid: A trained GridSearchCV object.
425     """
        ## Results from grid search
427     results = classificador_xg.cv_results_
        means_test = results['mean_test_score']
429     stds_test = results['std_test_score']
        means_train = results['mean_train_score']
431     stds_train = results['std_train_score']

433     ## Getting indexes of values per hyper-parameter
        masks=[]
435     masks_names= list(classificador_xg.best_params_.keys())
        for p_k, p_v in classificador_xg.best_params_.items():
437         masks.append(list(results['param_'+p_k].data==p_v))

439     params=classificador_xg.param_grid

441     ## Ploting results
        fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',
            figsize=(20,5))
```

```
443     fig.suptitle('Score per parameter')
        fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
445     pram_preformace_in_best = {}
        for i, p in enumerate(masks_names):
447         m = np.stack(masks[:i] + masks[i+1:])
            pram_preformace_in_best
449         best_parms_mask = m.all(axis=0)
            best_index = np.where(best_parms_mask)[0]
451         x = np.array(params[p])
            y_1 = np.array(means_test[best_index])
453         e_1 = np.array(stds_test[best_index])
            y_2 = np.array(means_train[best_index])
455         e_2 = np.array(stds_train[best_index])
            ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='test
                ')
457         ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='train'
                )
            ax[i].set_xlabel(p.upper())
459
        plt.legend()
461     plt.show()

463 """Grid Search LGBM"""

465 lg = lgb.LGBMClassifier(silent=False, random_state=7)

467 param_dist_lg = {"max_depth": [25,50, 75],
                 "learning_rate" : [0.01,0.05,0.1],
469              "num_leaves": [31,127],
                 "n_estimators": [200],
471              'feature_fraction': [0.5, 1.0],
                 'bagging_fraction': [0.75, 0.95],
473              'reg_alpha': [0.1, 0.5],
                }
475
CV = StratifiedKFold(10, random_state=42)
477
classificador_lg = GridSearchCV(lg, param_dist_lg, cv=CV,n_jobs=-1,
    return_train_score=True, scoring='accuracy')
479 proba_lg = classificador_lg.fit(X_train,y_train).predict_proba(X_test)

481 grid.best_estimator_

483 grid.best_score_
    grid.best_index_
485 grid.best_params_
```

```
487 resultados = pd.DataFrame(grid.cv_results_)
    resultados.head()
489
    scores5 = cross_val_score(classificador_lg, X_train, y_train, cv = CV)
491
    predicao5 = classificador_lg.predict(X_test)
493
    print("Acuracia MLP: Treinamento", classificador_lg.score(X_train, y_train
        )," Teste" ,classificador_lg.score(X_test, y_test))
495 print("Clasification report:", classification_report(y_test, predicao5))
    print("Confussion matrix:", confusion_matrix(y_test, predicao5))
497
    """Plot curva ROC"""
499
    plot_roc(y_test, proba_lg)
501 plt.show()

503 """Plot varia  o de Resultados"""

505 #def plot_search_results(grid):
        """
507     Params:
            grid: A trained GridSearchCV object.
509     """
        ## Results from grid search
511     results = classificador_lg.cv_results_
        means_test = results['mean_test_score']
513     stds_test = results['std_test_score']
        means_train = results['mean_train_score']
515     stds_train = results['std_train_score']

517     ## Getting indexes of values per hyper-parameter
        masks=[]
519     masks_names= list(classificador_lg.best_params_.keys())
        for p_k, p_v in classificador_lg.best_params_.items():
521         masks.append(list(results['param_'+p_k].data==p_v))

523     params=classificador_lg.param_grid

525     ## Ploting results
        fig, ax = plt.subplots(1,len(params),sharex='none', sharey='all',
            figsize=(20,5))
527     fig.suptitle('Score per parameter')
        fig.text(0.04, 0.5, 'MEAN SCORE', va='center', rotation='vertical')
529     pram_preformace_in_best = {}
        for i, p in enumerate(masks_names):
531         m = np.stack(masks[:i] + masks[i+1:])
```

```
        pram_preformace_in_best
533     best_parms_mask = m.all(axis=0)
        best_index = np.where(best_parms_mask)[0]
535     x = np.array(params[p])
        y_1 = np.array(means_test[best_index])
537     e_1 = np.array(stds_test[best_index])
        y_2 = np.array(means_train[best_index])
539     e_2 = np.array(stds_train[best_index])
        ax[i].errorbar(x, y_1, e_1, linestyle='--', marker='o', label='test
            ')
541     ax[i].errorbar(x, y_2, e_2, linestyle='-', marker='^',label='train'
            )
        ax[i].set_xlabel(p.upper())
543
    plt.legend()
545 plt.show()

547 """Teste de Kruskall"""

549 #prepare models
KNN = KNeighborsClassifier(algorithm='auto', leaf_size=1, metric='euclidean
    ',
551                            metric_params=None, n_jobs=-1, n_neighbors=3, p
                               =2,
                           weights='uniform')
553 RFC = RandomForestClassifier(bootstrap=True, class_weight=None, criterion='
    entropy',
                             max_depth=10, max_features=0.3, max_leaf_nodes
                                =None,
555                          min_impurity_decrease=0.0, min_impurity_split=
                                None,
                             min_samples_leaf=1, min_samples_split=2,
557                          min_weight_fraction_leaf=0.0, n_estimators
                                =100,
                             n_jobs=None, oob_score=False, random_state=
                                seed,
559                          verbose=0, warm_start=False)
SVMC = SVC(C=1, cache_size=200, class_weight=None, coef0=0.0,
561            decision_function_shape='ovr', degree=3, gamma=1, kernel='linear
                ',
               max_iter=-1, probability=False, random_state=seed, shrinking=
                   True,
563            tol=0.001, verbose=False)
MLP = MLPClassifier(activation='tanh', alpha=0.05, batch_size='auto',
    beta_1=0.9,
565                     beta_2=0.999, early_stopping=True, epsilon=1e-08,
                        hidden_layer_sizes=(100,), learning_rate='constant',
```

```
                        learning_rate_init=0.01, max_iter=1000, momentum=0.9,
                        n_iter_no_change=10, nesterovs_momentum=True, power_t
                            =0.5,
                        random_state=seed, shuffle=True, solver='sgd', tol
                            =0.0001,
                        validation_fraction=0.1, verbose=False, warm_start=
                            False)
XGB = XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                        colsample_bynode=1, colsample_bytree=1, gamma=0.5,
                        learning_rate=0.1, max_delta_step=0, max_depth=10,
                        min_child_weight=1, missing=None, n_estimators=200,
                            n_jobs=1,
                        nthread=None, objective='multi:softprob', random_state=
                            seed,
                        reg_alpha=0, reg_lambda=1, scale_pos_weight=1, seed=
                            None,
                        silent=None, subsample=0.8, verbosity=1)
LGB = LGBMClassifier(bagging_fraction=0.75, boosting_type='gbdt',
    class_weight=None,
                         colsample_bytree=1.0, feature_fraction=0.5,
                         importance_type='split', learning_rate=0.05, max_depth
                             =25,
                         min_child_samples=20, min_child_weight=0.001,
                             min_split_gain=0.0,
                         n_estimators=200, n_jobs=-1, num_leaves=31, objective=
                             None,
                         random_state=seed, reg_alpha=0.1, reg_lambda=0.0,
                             silent=False,
                         subsample=1.0, subsample_for_bin=200000,
                             subsample_freq=0)

models = []
models.append(('KNN', KNN))
models.append(('RandomForest', RFC))
models.append(('SVM', SVMC))
models.append(('RedeNeural', MLP))
models.append(('XGB', XGB))
models.append(('LGB', LGB))

# evaluate each model in turn
results = []
names = []
scoring = 'accuracy'

kfold = model_selection.StratifiedKFold(n_splits=30, random_state=seed)

for name, model in models:
```

```
    cv_results = model_selection.cross_val_score(model, X_train, y_train,
        cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)

# Teste de hipotese analisando o p-value
stat, p = stats.kruskal(results[0],results[1],results[2],results[3],
                        results[4],results[5])
alpha = 0.05
if p > alpha:
    print('Same distributions (fail to reject H0)')
else:
    print('Different distributions (reject H0)')
    print('Comparison stats', stat)

name_algorithms = ['KNN', 'RandomForest', 'SVM', 'RedeNeural', 'XGB', 'LGB'
    ]

list_final= []

for i in range(len(results)):
    list_1 = []
    for j in range(len(results)):
        result = stats.kruskal(results[i], results[j])
        list_1.append(result.pvalue)
    list_final.append(list_1)

def _color_highlight(val):
    color = ''
    if abs(val) >= 0.001 and abs(val) <= 0.05:
        color = 'orange'
    return 'background-color: %s' % color

board_kruskal = pd.DataFrame(list_final, columns=name_algorithms, index=
    name_algorithms)

board_kruskal.style.applymap(_color_highlight)

name_algorithms = ['KNN', 'RandomForest', 'SVM', 'RedeNeural', 'XGB', 'LGB'
                   ]

for i in range(len(results)):
    for j in range(i+1, len(results)):
        result = stats.kruskal(results[i], results[j])
```

```
645             print('Comparacao ' + name_algorithms[i] + ' | ' + name_algorithms[
                    j] + ' ->', result)
        print('\n')
647
KNN.fit(X_train, y_train)
649 RFC.fit(X_train, y_train)
SVMC.fit(X_train, y_train)
651 MLP.fit(X_train, y_train)
XGB.fit(X_train, y_train)
653 LGB.fit(X_train, y_train)

655 Y_test_prediction_KNN = KNN.predict(X_test)
Y_test_prediction_RFC = RFC.predict(X_test)
657 Y_test_prediction_SVMC = SVMC.predict(X_test)
Y_test_prediction_MLP = MLP.predict(X_test)
659 Y_test_prediction_XGB = XGB.predict(X_test)
Y_test_prediction_LGB = LGB.predict(X_test)
661
fig = plt.figure()
663 fig.suptitle('Algorithm Comparison')
ax = fig.add_subplot(111)
665 plt.boxplot(results)
ax.set_xticklabels(name_algorithms, rotation=45, ha='right')
667 plt.show()
```