UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA QUÍMICA E DE ALIMENTOS

Flávia Letícia Ribeiro

Predição do processo de secagem por *spray dryer* do suco concentrado de acerola utilizando inteligência artificial

Florianópolis
2020

Flávia Letícia Ribeiro

Predição do processo de secagem por *spray dryer* do suco concentrado de acerola utilizando inteligência artificial

Trabalho Conclusão do Curso de Graduação em Engenharia de Alimentos do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Bacharel em Engenharia de Alimentos.
Orientador: Prof. Dr. Ricardo Antônio Francisco Machado

Florianópolis
2020

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Ribeiro, Flávia Letícia
Predição do processo de secagem por spray dryer do suco concentrado de acerola utilizando inteligência artificial / Flávia Letícia Ribeiro ; orientador, Ricardo Antônio Francisco Machado, 2020.
79 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia de Alimentos, Florianópolis, 2020.

Inclui referências.

1. Engenharia de Alimentos. 2. spray dryer. 3. inteligência artificial. 4. machine learning. 5. secagem de acerola. I. Francisco Machado, Ricardo Antônio. II. Universidade Federal de Santa Catarina. Graduação em Engenharia de Alimentos. III. Título.

Flávia Letícia Ribeiro

Predição do processo de secagem por *spray dryer* do suco concentrado de acerola utilizando inteligência artificial

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Engenharia de Alimentos” e aprovado em sua forma final pelo Departamento de engenharia Química e de Alimentos.

Florianópolis, 30 de novembro de 2020.

Prof. Dr. João Borges Laurindo
Coordenador do Curso

Banca Examinadora:

Prof. Dr. Ricardo Antônio Francisco Machado
Orientador

Prof.ª Dr.ª
Alcilene Rodrigues Monteiro Fritz

Prof., Dr.
Bruno Augusto Mattar Carciofi

Aos amores da minha vida, meus pais e irmãos.

AGRADECIMENTOS

À Deus, pela força para chegar até aqui e a coragem para sempre buscar novos desafios.

Aos meus pais, que são a razão de tudo e os principais responsáveis pela pessoa que me tornei, pelos meus valores e pela minha formação.

Aos meus irmãos, Júlia e Felipe, por estarem do meu lado em todos os momentos.

Ao meu namorado, Luiz Philipi Machado, por todo o apoio, paciência e parceria durante o desenvolvimento deste trabalho.

À todos os professores que passaram pela minha trajetória como estudante e contribuíram para que me tornasse a profissional que sou. Em especial ao professor Ricardo Machado, pela orientação e contribuições para o meu trabalho e formação.

À HarboR Informática Industrial pela oportunidade de me aproximar um pouco mais da indústria e colocar em prática tudo que aprendi nos anos de graduação, em especial aos líderes Paulo Narciso, Túlio Duarte e Elisângela Catapan, por serem referências do profissional que busco me tornar.

Aos membros da banca, Bruno Mattar e Alcilene Monteiro, pela disponibilidade.

À Universidade Federal de Santa Catarina e ao Departamento de Engenharia Química e de Alimentos, pelo apoio físico, técnico e psicológico necessários para a minha formação.

RESUMO

Frente ao panorama atual, a busca por garantia da qualidade, aliada à redução de custos, diminuição de falhas operacionais e prevenção de defeitos vem aumentando gradativamente. Alinhado a isto, a quarta revolução industrial tem como característica a utilização de sistemas computacionais para atender aos requisitos ágeis e dinâmicos de produção e melhorar a eficácia de processos industriais. A inteligência artificial vem ganhando espaço neste meio com o uso de ferramentas para auxiliar na tomada de decisão, a fim de aumentar a produtividade e diminuir a intervenção humana. O processo de secagem de alimentos está entre uma das operações mais complexas e criteriosas. Um controle eficiente deve ser empregado nesta etapa a fim de reduzir a degradação dos compostos bioativos. Com isso, no presente trabalho foram avaliados os modelos baseados em inteligência artificial na previsão do teor de vitamina C ao final do processo de secagem por atomização do suco concentrado de acerola. Para isso, um conjunto de dados foi disponibilizado por indústria local. Para a aplicação dos algoritmos utilizou-se algumas técnicas de tratamento dos dados, tais como detecção de *outliers*, substituição de valores faltantes, normalização e extrapolação dos dados. Os modelos *K-Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), Rede Neural Artificial (RNA), *Random Forest* (RF) e *Stacked Generalization* (SG) foram implementados em Python para avaliar a assertividade de cada modelo frente ao conjunto de dados. Como resultado, os modelos apresentaram boa capacidade de predição e a aplicação de modelos de inteligência artificial mostrou-se viável para a utilização em processos industriais.

Palavras-chave: Aprendizado de máquina. Secagem por atomização. Inteligência artificial. Ácido ascórbico.

ABSTRACT

In view of the current scenario, the search for quality assurance, combined with cost reduction, reduction of operational failures and prevention of defects has been gradually increasing. In line with this, the fourth industrial revolution is characterized by the use of computer systems to meet agile and dynamic production requirements and improve the efficiency of industrial processes. Artificial intelligence has been gaining ground in this zone with the use of tools to assist in decision making, in order to increase productivity and decrease human intervention. The food drying process is among one of the most complex and thorough operations. Efficient control must be employed at this stage in order to reduce the degradation of bioactive compounds. Thus, this study sought to evaluate the artificial intelligence models in the prediction of vitamin C content at the end of the spray drying process of concentrated acerola juice. For this purpose, a data set was obtained from a specific industry in the acerola sector. For the application of the algorithms, some data processing techniques were used, such as outlier detection, replacement of missing values, normalization and extrapolation of the data. K-Nearest Neighbors Algorithm (KNN), Support Vector Machine (SVM), Artificial Neural Network (RNA), Random Forest (RF) and Stacked Generalization (SG) models were implemented in Python to evaluate the assertiveness of each model against the data set. As a result, the models showed good predictive capacity and the application of artificial intelligence models proved to be viable for use in industrial processes.

Keywords: Machine Learning. Spray drying. Artificial intelligence. Ascorbic acid.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AA	Ácido L-ascórbico
AD	Árvore de Decisão
AM	Aprendizado de Máquina
CLP	Controlador Lógico Programável
DHA	Ácido L-dehidroascórbico
IA	Inteligência Artificial
KNN	*K-Nearest Neighbors*
RF	*Random Forest*
RNA	Rede Neural Artificial
SE	Sistema Especialista
SG	*Stacked Generalization*
SVM	*Support Vector Machine*

SUMÁRIO

1 INTRODUÇÃO

Os rápidos avanços nos métodos de industrialização e informatização estimularam um progresso no desenvolvimento da tecnologia. A quarta revolução industrial, também conhecida como Indústria 4.0, teve inicio na Alemanha em 2011 e tem como objetivo atingir um maior nível de eficiência operacional e produtiva através da automatização (SCHWAB, 2019).

A quarta revolução industrial tem como característica a utilização de sistemas computacionais para atender aos requisitos ágeis e dinâmicos de produção e melhorar a eficácia e eficiência de toda a indústria (LU, 2017).

Com a incorporação dos avanços científico e tecnológicos, termos como Inteligência Artificial (IA), *Data Science*, *Big Data*, Internet das Coisas, *Machine Learning* e tantos outros, passaram a ser utilizados e estudados a fim de transformar a maneira como máquinas se comunicam e utilizam as informações para otimizar o processo de produção, tornando-o mais econômico, ágil e autônomo (ZENG; MARTINEZ, 2000).

A Inteligencia artificial ganhou um espaço cada vez maior neste setor. A possibilidade de tomar decisões a partir de dados para aumentar a produtividade e diminuir a intervenção humana vêm sendo estudada e possui uma atuação importante. Entretanto, ainda há muito o que ser explorado em aplicações industriais.

Uma das principais vantagens da Inteligência Artificial é o fato de que a tecnologia tem a capacidade de aprender com seus próprios erros, sem a necessidade do intermédio de uma pessoa responsável para configurá-la. As falhas decorrentes da atuação humana, influenciadas por fatores externos, são praticamente nulas com a utilização de IA. O conceito de aprendizado de máquinas é um dos ramos que se encaixam dentro deste universo, denominado *Machine Learning* (TEIXEIRA, 2019).

A indústria, no geral, está passando por um período de transformação. São inúmeros os processos onde pode-se aplicar os conceitos de IA para otimizar a performance do mesmo. Modelos matemáticos são comumente utilizados na otimização de processos. Entretanto esses modelos, embora muito precisos, precisam lidar com a complexidade dos princípios fenomenológicos existentes, como as transferências de calor e massa, e a necessidade de parâmetros nem sempre disponíveis para a sua resolução.

O processo de secagem de alimentos está entre um dos processos mais complexos e criteriosos. Um controle eficiente deve ser empregado nesta operação a fim de reduzir o consumo de energia e a degradação dos compostos bioativos, uma vez que a utilização de temperaturas elevadas podem comprometer a qualidade do produto (SANTOS *et al.*, 2020).

No processo de secagem da acerola não é diferente. Sendo um produto com alto teor de vitamina C, a acerola tem um destaque pelo seu valor nutricional. A vitamina C

possui uma cinética de degradação influenciada por diversos fatores, o controle desses fatores para obter um produto com alto teor de ácido ascórbico é um desafio para a industria de alimentos (TEIXEIRA; MONTEIRO, 2006).

Diferentes parâmetros podem ser atribuídos ao processo de secagem, a definição das melhores especificações dependerão dos equipamentos utilizados no processo (AGUIRRE; GASPARINO FILHO, 1999). A secagem por *spray dryer* começou a ser utilizada na década de 20 devido a sua disponibilidade de equipamentos, viabilidade econômica e boa qualidade e estabilidade do produto final e vem sendo amplamente utilizado nas indústrias alimentícia e farmacêutica (JAYASUNDERA *et al.*, 2011).

A secagem por atomização possui como vantagem o controle da uniformidade das partículas, disponibilidade para alterar as condições de operação sem interromper o processo, custo relativamente baixo e baixo tempo de residência do produto, apresentando uma baixa agressividade aos produtos termossensíveis (SILVA, 2017).

Dessa forma, a qualidade do produto, a melhoria das condições de operação e o custo energético têm sido tema de estudo cada vez mais frequentes entre os pesquisadores da área de engenharia de alimentos (FREIRE, 2011).

Com isso, o presente trabalho tem por finalidade estudar os modelos de *Machine Learning* para identificar e aplicá-los no processo de secagem por *spray dryer* do suco concentrado de acerola, avaliando o teor de vitamina C ao final do processo.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é propor o uso de ferramentas de inteligência artificial como métodos de modelagem do sistema de secagem do suco de acerola para predizer o teor de vitamina C na saída do *spray dryer*.

1.1.2 Objetivos Específicos

Para que o objetivo geral seja alcançado, os seguintes objetivos específicos foram definidos:

a. Analisar a capacidade de predição dos modelos KNN, SVM, RNA e *Random Forest*;

b. Aplicar o método de *Stacked Generalization* para geração de um modelo com melhor coeficiente de determinação, erro quadrático médio e erro absoluto médio;

c. Analisar a performance e assertividade dos modelos aplicados utilizando métricas de avaliação;

d. Validar o modelos desenvolvido a partir de um novo conjunto de dados.

2 REVISÃO BIBLIOGRÁFICA

2.1 ACEROLA

2.1.1 Aspectos Gerais

A aceroleira é um arbusto de porte mediano cujo desenvolvimento e produção são favorecidos em climas tropicais e subtropicais, com produção de três a mais safras por ano concentrando-se nas estações da primavera e verão, tendo como média de 26ºC a temperatura ideal para seu cultivo (JUNQUEIRA; PIO, 2004). A partir do terceiro ou quarto ano pós plantio, plantas adultas chegam à produção anual de 40 kg por unidade correspondendo à produtividade de 16 toneladas por hectare (RITZINGER, R.; RITZINGER, C., 2004).

O fruto da aceroleira é constituído por uma película fina externa, um mesocarpo que diz respeito à polpa, e endocarpo, que corresponde a três caroços unidos. Durante o processo de maturação, a acerola apresenta tonalidades que variam de acordo com à degradação da clorofila e à síntese de antocianinas e carotenoides, podendo variar do verde ao vermelho (RIBEIRO; SERAVALLI, 2007).

Segundo a Embrapa (2012), a acerola teve seu cultivo intensificado devido a descoberta do seu alto teor de vitamina C em 1946. Através desta descoberta, o plantio comercial da aceroleira iniciou-se na América Central chegando ao Brasil pouco tempo depois.

Atualmente o Brasil é considerado o maior produtor, consumidor e exportador mundial de acerola, com uma área de plantio de aproximadamente 6.000 hectares. Entre os principais estados brasileiros produtores de acerola, Pernambuco representa 21.351 toneladas da produção nacional, seguido pelo Ceará, com 7.578, Sergipe com 5.427 e São Paulo, com 3.907 toneladas (IBGE, 2017). Com destaque para as regiões no trópico-árido do Nordeste brasileiro, onde possui cerca de 3.100 ha e uma posição de destaque de 14 polos de irrigação em constante desenvolvimento, devido as condições do solo e clima que favorecem o cultivo e permitem o plantio durante boa parte do ano (GONZAGA NETO *et al.*, 2012).

O consumo e produção em expansão dessa fruta deve-se, basicamente, ao seu teor de ácido ascórbico que, em algumas variedades, alcança até 5.000 mg/100 g de polpa, chegando a ser 100 vezes superior ao da laranja e 10 vezes ao da goiaba, frutas tidas como as de mais alto conteúdo desta vitamina (GONZAGA NETO *et al.*, 2012).

2.1.2 Características e Propriedades

A composição química e as propriedades da acerola variam de acordo com a espécie, condições ambientais e, também, com o estágio de maturação da fruta (VENDRAMINI; TRUGO, 2000). A vitamina C e as demais características atribuídas

à qualidade da acerola, como coloração, peso e tamanho dos frutos, teor de sólidos solúveis e pH do suco, sofrem influência pela desuniformidade genética dos pomares, precipitações pluviais, temperatura, altitude, adubação, irrigação e a ocorrência de pragas e doenças (NOGUEIRA *et al.*, 2002).

Ns Tabela 1 são apresentados valores para ácido ascórbico de diferentes estudos onde avaliaram cultivares em diferentes estágios de maturação. Os resultados apresentados referem-se ao estágio final, onde o fruto está maduro.

Tabela 1 – Valores médios de vitamina C de diferentes cultivares de acerolas, segundo diferentes autores.

Autores	**Cultivar**	**Vitamina C (mg 100g^{-1})**
Costa *et al.* (2011)	Junco	970,06 [a]
Luciana De Siqueira Oliveira *et al.* (2012)	Cereja	1642 $\pm$ 0,08
Luciana De Siqueira Oliveira *et al.* (2012)	Roxinha	1293 $\pm$ 0,14
Nasser e Zonta (2014)	Okiwana	2580,00
Patrício Ferreira Batista *et al.* (2018)	Okiwana	2337,18 $\pm$ 82,73
Patrício Ferreira Batista *et al.* (2018)	Sertaneja	2075,13 $\pm$ 9,95

[a] Resultados em $mg.100mL^{-1}$

Vendramini e Trugo (2000) caracterizaram a acerola *(Malpíghia punicifolia L.)* ressaltando as modificações presentes em cada estágio de maturação. Como resultado, os autores observaram uma redução de 50% do ácido ascórbico do estágio verde para o vermelho, esse fator ocorre devido a oxidação bioquímica. Os sólidos solúveis e os açucares totais apresentaram um aumento de 7,8 para 9,2 $^{\circ}$Brix e de 3,3 a 4,4% p/p, respectivamente. Entretanto, as análises indicaram que o pH permaneceu praticamente constante, indicando que as diferenças no grau de dissociação entre o ácido ascórbico e outros ácidos orgânicos produziram um equilíbrio líquido constante. Os resultados são apresentados na Tabela 2.

Tabela 2 – Caracterização da Acerola em diferentes estágios de maturação.

Características[a]	**Imaturo (Verde)**	**Intermediário (Amarelo)**	**Maduro (Vermelho)**
Vitamina C [b]	2164	1065	1074
Proteína	1,2	0,9	0,9
Cinza	0,4	0,4	0,4
Umidade	91,0	92,4	92,4
Acidez titulável [c]	18,2	15,6	34,4
pH	3,7	3,6	3,7
Sólidos solúveis	7,8	7,7	9,2
Açúcares redutores	3,3	4,2	4,4
Açúcares não redutores	1,1	0,1	nd [d]
Açúcares totais	4,3	4,3	4,4

[a] Os resultados estão em g/100 g de amostra, exceto conforme indicado abaixo.

[b] Resultados em mg/100g de amostra

[c] Resultados em ml de NaOH 0,1 N / 100g de amostra.

[d] nd, não detectado

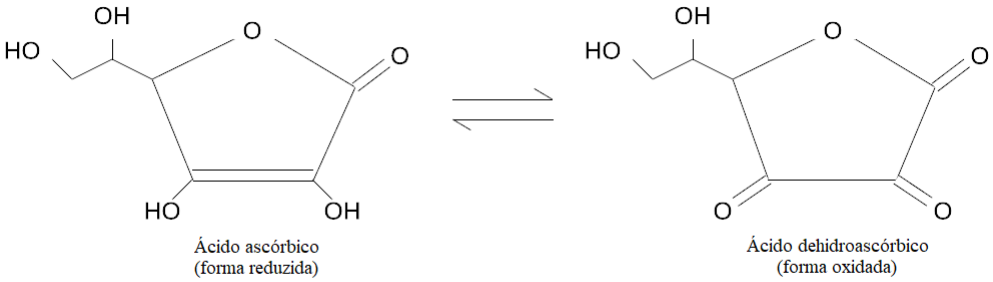
Fonte – Extraído de Vendramini e Trugo (2000).

2.1.2.1 Vitamina C

Segundo Grzybowski e Pietrzak (2013) a vitamina C vem sendo aplicada como um componente de diversos medicamentos e cosméticos dermatológicos desde 1928. Presente em grande número de frutas e vegetais, o ácido ascórbico é uma vitamina hidrossolúvel, de forma cristalina, conhecida por suas propriedades antioxidantes e suas aplicações na terapia do câncer e alterações inflamatórias da pele (NÓBREGA, 2012).

Chamamos de ácido ascórbico, ou Ácido L-ascórbico (AA), a vitamina C em sua forma reduzida. Enquanto que em sua forma oxidada, temos o Ácido L-dehidroascórbico (DHA) (NÓBREGA, 2012). O ácido L-ascórbico é um composto biologicamente ativo, instável, que pode ser oxidado a ácido L-dehidroascórbico de forma fácil e reversível (TEIXEIRA; MONTEIRO, 2006).

Figura 1 – Estrutura química da vitamina C.



Fonte – Toralles *et al.* (2008)

Derivado da hexose a partir da glicose e da galactose, o ácido ascórbico é sintetizado por vegetais e pela maioria dos animais. Entretanto, o homem não possui a enzima L-gulonolactona oxidase que participa da biossíntese da vitamina C, sendo necessária a ingestão desta vitamina pela dieta alimentar. Segundo a ANVISA (2005) a ingestão diária recomendada de vitamina C para adultos é 45 mg/dia e para crianças entre 30-35 mg/dia.

Entre uma das suas funções, o ácido ascórbico tem a capacidade de ceder e receber elétrons, o que lhe confere um papel essencial como antioxidante. Além disso, ele está envolvido na síntese e manutenção do colágeno e neurotransmissores, como a norepinefrina obtida a partir da dopamina e a serotonina. O consumo de vitamina C também pode promover resistência a infecções e ajudar em processos de cicatrização (MANELA-AZULAY *et al.*, 2003).

Moraes *et al.* (2018) avaliou o processo de cicatrização em peixes utilizando uma alimentação com ração não suplementada (controle) e suplementada com 100, 200 e 500 mg de vitamina C/kg de ração. Os resultados mostraram que a vitamina C promoveu a aceleração da cicatrização através do aumento da proliferação de células das mucosas, do acúmulo de colágeno, da remodelação tecidual e da formação de escamas, levando à redução proporcional da área lesada.

Além disso, estudos epidemiológicos apontam que a vitamina C pode atuar na inibição da formação de nitrosaminas cancerígenas colaborando para prevenção do câncer e no tratamento da doença, na diminuição do risco de doenças cardiovasculares, no tratamento da hipertensão e na redução da incidência de cataratas (BLOOM *et al.*, 2015) (SOLDI *et al.*, 2019).

Por esses fatores, a ingestão de vitamina C torna-se essencial na alimentação, sendo as frutas as maiores fontes encontradas na natureza, com destaque para laranja, acerola, tangerina, goiaba, limão e cupuaçu. Entretanto, diversas variáveis contribuem

para a degradação da fruta in natura, como sazonalidade, condições climáticas, colheita e condições de estocagem. A fim de aumentar sua disponibilidade no mercado, produtos como suco e polpa são desenvolvidos a partir da fruta e aumentando, assim, a vida útil do produto (TEIXEIRA; MONTEIRO, 2006).

2.1.2.1.1 Degradação de Vitamina C

Por tratar-se de um composto orgânico, o ácido ascórbico pode sofrer diversas alterações físico-químicas. Fatores como pH, temperatura, umidade e luz podem alterar a estabilidade desta vitamina, acarretando na redução do seu teor nos alimentos (FABRÍCIO, 2018).

As reações de degradação da vitamina C em sucos de fruta são predominantemente de natureza não-enzimática, e podem ser aeróbicas ou anaeróbicas. Em condições aeróbicas, o ácido ascórbico é transformado em ácido L-dehidroascórbico que passa a ácido 2,3-dicetogulônico produzindo, finalmente, hidroxifurfural. Já em condições anaeróbicas, ele decompõe-se em ácido 2,5-dihidro-2-furanóico que passa a dióxido de carbono e furfural. O furfural sofre polimerização como um aldeído ativo e pode se combinar com aminoácidos, contribuindo para o escurecimento do suco (TANAKA, 2007).

O processamento de sucos e polpas de frutas e suas condições de estocagem têm grande influência na perda da vitamina C. Os principais fatores que podem afetar a degradação do ácido ascórbico incluem o tipo de processamento, condições de estocagem, tipo de embalagem, oxigênio, luz, catalisadores metálicos, enzimas e pH, sendo primordial um controle de processo durante toda a cadeia produtiva (TEIXEIRA; MONTEIRO, 2006). Uma das técnicas amplamente empregada é a conservação de alimentos pelo controle de umidade, utilizando diferentes secadores industriais para obter um produto desidratado.

2.2 SECAGEM

Segundo McCabe *et al.* (1995) o processo de secagem é a operação na qual um liquido é removido de um material sólido na forma de vapor, por meio de um mecanismo de vaporização térmica ou sublimação (liofilização). Consequentemente, ocorrerá a diminuição do crescimento microbiano e redução da velocidade de reações químicas e bioquímicas. Além disso, a baixa disponibilidade de água dificultará a ação de enzimas sobre os alimentos (NÓBREGA, 2012).

O processo de secagem utiliza ar quente para realizar a transferência de calor para o alimento e vaporização da água presente no meio. A capacidade do ar para eliminar a água depende principalmente de sua temperatura e umidade relativa (CELESTINO, 2010) (NÓBREGA, 2012).

Segundo McCabe *et al.* (1995) e Nóbrega (2012) o processo é realizado em duas etapas primordiais:

- Transferência de energia do ambiente para evaporar a umidade superficial. Essa etapa é dependente das condições externas ao alimento, como temperatura, umidade do ar, fluxo do ar e pressão;
- Transferência de massa do interior do alimento para a superfície do material, que depende das condições física do sólido, temperatura e teor de umidade.

Diferentes parâmetros podem ser atribuídos ao processo de secagem, a definição das melhores especificações dependem dos equipamentos utilizados no processo. Os secadores mais utilizados industrialmente são os secadores de esteiras, pneumáticos, de secagem por atomização (*spray dryers*), de leito fluidizado, de cilindro rotativo, a vácuo e por micro-ondas. Os fatores principais para a seleção do tipo de secador a ser utilizado incluem a natureza do produto, as condições de operação e fatores econômicos (AGUIRRE; GASPARINO FILHO, 1999).

A ampla utilização desta operação unitária é explicada pelas suas inúmeras vantagens, dentre elas, podemos destacar:

a. Aumento da vida útil do produto;
b. Facilidade de transporte e comercialização do produto final;
c. Redução de perdas pós colheita;
d. Baixo custo operacional;
e. Alto valor nutritivo: o valor alimentício do produto concentra-se devido a perda de água.

2.2.1 Secador do tipo *spray dryer*

Devido a disponibilidade de equipamentos, viabilidade econômica e boa qualidade e estabilidade do produto final, a secagem por *spray drying* começou a ser utilizada a partir da década de 20 com o desenvolvimento de leite e sabão em pó. Atualmente, tal processo vem sendo amplamente utilizado nas indústrias alimentícia e farmacêutica (JAYASUNDERA *et al.*, 2011).

Dentre as vantagens do processo de atomização, podemos citar o controle da uniformidade das partículas, disponibilidade para alterar as condições de operação sem interromper o processo, alto rendimento e custo de processo relativamente baixo. Além disso, o processo por *spray dryer* apresenta uma baixa agressividade a produtos termossensíveis, como a vitamina C e alguns compostos bioativos, tal fator deve-se ao baixo tempo de residência dos produtos na câmara de secagem (SILVA, 2017).

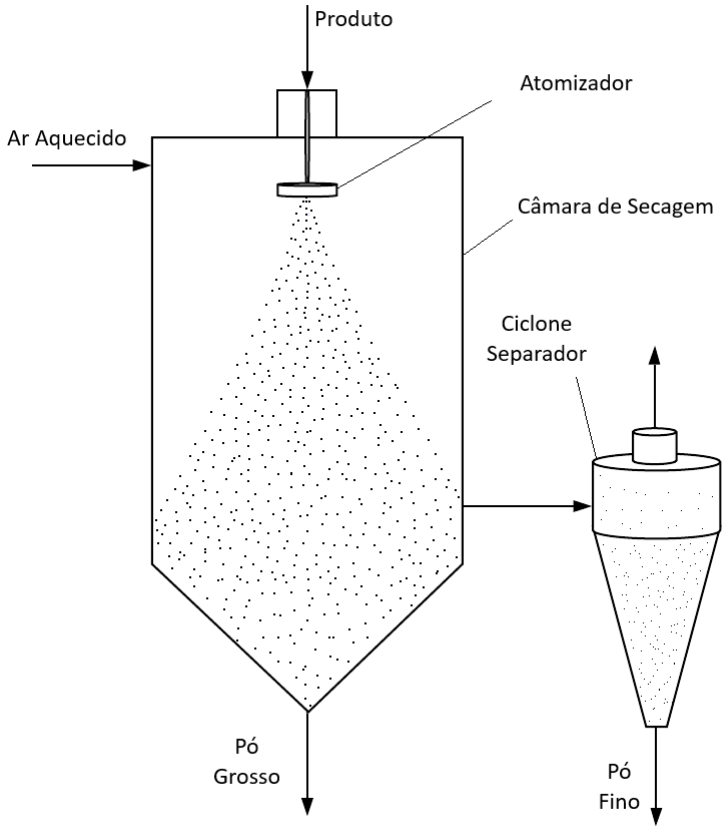
Esse sistema de secagem consiste de um atomizador, uma fonte de ar quente, uma câmara de secagem e um separador, conforme Figura 2. O processo ocorre em três etapas fundamentais.

Na primeira fase, o fluído a ser secado é disperso como gotículas através do atomizador, produzindo uma área superficial por toda a câmara de secagem.

Na segunda, ocorre o contato dessas gotículas com uma corrente de ar aquecido, que ao entrar em contato com o ar suficientemente quente para suprir o calor latente de vaporização, ocorre a evaporação quase instantânea da água presente na superfície (MEZHERICHER *et al.*, 2010).

Na terceira etapa, ocorre a evaporação do solvente e a formação da partícula sólida. A evaporação superficial da gotícula conduz à formação de uma camada de material seco externa, por meio desta camada, o liquido situado no interior da gotícula propaga-se para o exterior. Por fim, o produto é transportado por uma corrente de ar para o ciclone e posteriormente coletado (OLIVEIRA, O. W.; PETROVICK, 2010).

Figura 2 – Sistema de secagem por aspersão



Fonte – Adaptado de LABMAQ (2020).

2.2.1.1 Fatores que influenciam no rendimento do processo

A qualidade do produto final está diretamente relacionada com as condições de processo, *design* do equipamento e característica do material de entrada. Desta forma, todas as variáveis devem ser controladas durante o processo de desidratação a fim de obter o melhor rendimento, teor de umidade adequado, minimização da aderência de partículas na câmara de secagem e adequação aos limites especificados (KESHANI *et al.*, 2015).

Como fatores associados ao produto, podemos citar a viscosidade do suco, conteúdo de sólidos solúveis, tensão superficial e fluxo de alimentação. A concentração de sólidos presentes no alimento exerce grande influência sobre a eficiência da operação, visto que baixas concentrações necessitam que uma grande quantidade de solvente seja eliminada, aumentando o custo do processo. Além disso, é um dos fatores determinante no tamanho da partícula e densidade do produto final (OLIVEIRA, N. d. M. S. *et al.*, 2002).

Além da concentração de sólidos, a formação de partículas a partir do atomizador está diretamente relacionada com a viscosidade do produto. Deste modo, quando temos um produto com baixa densidade, menos energia ou menor pressão são aplicadas ao processo. Em contra partida, elevada viscosidade requer uma maior exposição à atomização e dificulta a formação das gotas, resultando em partículas de forma irregular (OLIVEIRA, N. d. M. S. *et al.*, 2002) (ENGEL, 2017).

Em relação ao processo, podemos destacar o tipo e mecanismo de funcionamento do atomizador e as propriedades do ar de secagem (velocidade, temperatura de entrada e de saída, pressão). A temperatura do ar de entrada é um fator determinante na qualidade do produto obtido. O aumento na temperatura do material de entrada facilita o processo de secagem, pois normalmente reduz a tensão superficial e a viscosidade, facilitando a formação de gotículas (OLIVEIRA, N. d. M. S. *et al.*, 2002).

Fazaeli *et al.* (2012) avaliou os efeitos das condições de processo nas propriedades do pó de amora preta. Utilizando um secador por pulverização e maltodextrina como coadjuvante de processo, os autores observaram que uma maior temperatura de entrada de ar provocou um aumento no rendimento e na solubilidade, diminuindo a densidade, umidade e atividade de água. O aumento da taxa de fluxo de ar apresentou um efeito positivo no rendimento e na densidade e um efeito negativo sobre as demais propriedades.

2.2.2 Influência da secagem em compostos bioativos de alimentos

Produtos submetidos ao processo de secagem sofrem constantes modificações que ocasionam em mudanças na qualidade do produto final. Determinadas alterações influenciam diretamente no sabor, aroma, textura, cor e valor nutricional dos alimentos

(SANTOS *et al.*, 2020).

Essas alterações são muito relevantes ao se considerar compostos bioativos presentes nos alimentos, tais como compostos fenólicos e vitamina C, no qual podem ser significantemente perdidos ou reduzidos devido as condições de processo. Dessa forma, a melhoria das condições de operação e o custo energético têm sido tema de estudo cada vez mais frequentes entre os pesquisadores da área (FREIRE, 2011)

Marete *et al.* (2009) estudaram os efeitos da temperatura sobre a extração de compostos fenólicos e cor da erva medicinal *Tanacetum parthenium*. Os resultados apontaram forte relação entre a quantidade de compostos fenólicos totais e cor desejável. Em temperaturas acima de 70ºC, observou-se que a enzima polifenoloxidase foi inativada, o que resultou em extratos mais coloridos e com alto teor de compostos fenólicos totais.

MOREIRA *et al.* (2010) avaliaram a retenção do ácido ascórbico e antocianinas durante a secagem do resíduo da acerola na secagem por atomização, levando em consideração temperatura de entrada, proporção de adjuvante e substituição de maltodextrina por goma de cajueiro. Como resultado, os autores observaram que os grau de retenção de ácido ascórbico e antocianina foram prejudicados pelo aumento da temperatura de entrada e favorecida por temperaturas mais baixas e maior proporção de adjuvante.

Nóbrega (2012) avaliou a secagem do resíduo da acerola em secador convectivo de bandejas sob condições controladas de temperatura (60, 70 e 80ºC), velocidade do ar (4, 5 e 6 m/s) e espessura do material (0,5, 0,62 e 0,75 cm) para estudar o impacto da secagem sobre as características físico-químicas, cor, concentração de compostos bioativos selecionados e atividade antioxidante do produto final. Os resultados demostraram que houve uma diminuição da concentração de compostos fenólicos, antocianinas, carotenoides, proantocianidinas e ácido ascórbico ocasionada pela secagem nas condições estudadas. No entanto, devido a concentração final desses compostos detectada no produto desidratado, a caracterização colorimétrica e a estabilidade microbiológica alcançada, concluiu-se que o pó do resíduo de acerola trata-se de um ingrediente com elevado potencial bioativo.

2.3 INTELIGÊNCIA ARTIFICIAL

Segundo Russell e Norvig (2004), a inteligência artificial é uma área da computação que visa usar métodos e dispositivos que simulem a capacidade humana de pensar ou resolver problemas. A IA é uma ciência que teve origem na segunda guerra mundial com a necessidade de desenvolver tecnologias voltada para a indústria bélica. Após algum tempo surgiram novas linhas de pesquisas da IA, umas das mais importantes é a IA para a área biológica que iniciou o conceito de redes neurais artificiais.

Warrem Macculloch e Walter Pitts (1943) sugeriram um modelo de neurônios artificiais, no qual, cada neurônio era caracterizado por “ligado” ou “desligado”, desse modo, o estado de um neurônio era analisado como, “equivalente em termos concretos a uma proposição que definia seu estímulo adequado”, assim, eles demonstraram que qualquer função computacional pode ser calculada por uma série de neurônios interligados. (RUSSELL; NORVIG, 2004).

Sete anos depois, Alan Turing articulou uma visão completa da IA em seu artigo de 1950 “Computing Machinery and Intelligence” com aplicação do Teste de Turing, onde sugeria um teste baseado na impossibilidade de distinguir entre seres humanos e máquinas. O computador passa no teste se, ao responder algumas perguntas realizadas por um humano, o mesmo não conseguir distinguir quem respondeu seus questionamentos (TURING, 1950).

Em 1969, a Universidade de Stanford desenvolveu o programa DENDRAL para desenvolver soluções capazes de encontrar as estruturas moleculares orgânicas a partir da espectrometria de massa das ligações químicas presentes em uma molécula desconhecida. O DENDRAL, sistema desenvolvido por Edward Feigenbaum, Bruce Buchanan e Joshua Lederberg, foi capaz de solucionar o problema devido ao seu modo automático de tomar decisões e teve sua importância para o desenvolvimento de programas inteligentes, pois tratou-se do primeiro sistema bem-sucedido de conhecimento intensivo (RUSSELL; NORVIG, 2004).

Após esse estudo, Feigenbaum aprofundou-se nos estudos de IA para avaliar outras aplicações na área do conhecimento humano. Foi então que surgiu o MYCIN, um sistema especialista para diagnosticar infecções sanguíneas, que apresentou resultados condizentes quando comparados com especialistas da área, e resultados superiores quando comparados com médicos recém formados (FEIGENBAUM *et al.*, 1970).

Nos tempos atuais, o campo de IA abrange uma variedade de subcampos, desde áreas de uso geral, como na robótica, reconhecimento de padrões em imagens de satélites, imagens médicas, sistemas de apoio ao diagnóstico médico, sistemas de ensino-aprendizagem e em diversos sistemas de controle industrial (GAMA *et al.*, 2011).

Na medicina, Piccolo *et al.* (2002) avaliaram a validade da dermatoscopia digital

comparando os diagnósticos de um dermatologista com 5 anos de experiência com os de um clínico com treinamento mínimo nesse campo e comparando esses resultados com os obtidos por meio de diagnósticos por computador. Um banco de dados com 341 lesões cutâneas melanocíticas e não melanocíticas foram incluídas e imagens digitais de todas as lesões foram analisadas usando software baseado em uma rede neural artificial treinada. Como resultado, os autores concluíram que a análise feita pela rede neural pode melhorar a precisão diagnóstica do melanoma quando comparado com o diagnóstico de um médico inexperiente. Além disso, o diagnóstico por computador pode representar uma ferramenta útil para a triagem do melanoma, principalmente em centros não experientes em dermatoscopia.

Já Zellweger *et al.* (2018) utilizando o algoritmo Basel-MPA e Stuckey *et al.* (2018) os conceitos de *Machine Learning* aplicaram IA na cardiologia, onde obtiveram melhores discriminações entre pacientes que tinham ou não angiografia documentada de doença arterial coronariana. Com isso, substituíram ferramentas de alto custo, invasivas e não-invasivas, principalmente em pacientes com pouco risco de doenças arteriais, apresentando alto índice de confiança, sensibilidade e especificidade comparável a outros testes funcionais.

Na área de alimentos, CHEN *et al.* (2007) utilizou rede neural para classificação de carcaça de aves. Um total de 236 carcaças de frango foram analisadas. Entre elas, 99 eram saudáveis e 137 não. Cada carcaça foi escaneada com uma sonda de espectrofotômetro Vis/NIR por 6 vezes: 3 vezes para cada velocidades de linha (60 e 90 aves por min). Foram obtidos um total de 1.416 espectros de reflectância, sendo, 594 saudáveis e 822 não-saudáveis. Das 236 galinhas, 126 foram escaneadas sob a luz ambiente (um total de 756 espectros) e 110 foram escaneadas no ambiente escuro (um total de 660 espectros). Os resultados do experimento mostraram que os modelos de redes neurais classificaram as carcaças saudáveis ou não-saudáveis com precisão média superior a 94%. Os melhores resultados foram obtidos com uma velocidade de manilha de 90 aves/min e detecção em ambiente escuro. As precisões foram de 96,0% para classificar carcaças saudáveis e 98,9% para carcaças não-saudáveis.

Leal *et al.* (2017) avaliaram as diferentes proporções de frações de carboidratos e proteínas de capim-braquiarão (Brachiaria brizantha (Hochst) cv. Marandu), com o objetivo de compreender a influência de cada fração na nutrição animal, para com isso, otimizar o uso da forragem e melhorar o manejo dos animais. Visando a dificuldade atual em quantificar, em campo, valores de matéria seca, fibras, lignina, teores de proteína e carboidratos, os autores justificaram o estudo do fracionamento da forragem por ser um método que obtém resultados relevantes, beneficiando o sistema planta-animal, uma vez que torna possível compreender a participação de cada fração na nutrição animal e obter um aproveitamento adequado das pastagens. No desenvolvimento do trabalho, o autor comparou as respostas laboratoriais de fibras, proteína bruta e suas

frações com as respostas de uma rede neural artificial (RNA) para obter um modelo com correlação de todas as variáveis envolvidas para predizer os valores nutricionais da planta. Com isso, seria possível reduzir o tempo em análises laboratoriais. Como resultado concluíram que a RNA foi capaz de predizer os teores de proteína bruta, fibra em detergente neutro e detergente ácido, apresentando valores muito próximos às análises realizadas em laboratório. Além disso, o tempo de avaliação da RNA é relativamente baixo quando comparado ao tempo para realizar as analises laboratoriais.

Sistematizando e automatizando tarefas manuais e intelectuais, a IA torna-se uma área de aplicação abrangente (RUSSELL; NORVIG, 2004). Por muitos anos, esta área foi vista como uma área teórica com aplicação em problemas simplórios e desafiadores, mas de pouco valor prático. Os problemas práticos que precisavam de computação eram resolvidos pela codificação em alguma linguagem de programação (RUSSELL; NORVIG, 2004).

Na década de 70, houve uma maior disseminação do uso de técnicas de computadores baseadas em IA para a solução de problemas reais. Muitas vezes, esses problemas eram tratados computacionalmente necessitando de conhecimento de especialistas de um dado domínio, por exemplo, no domínio da medicina, que era então codificado por regras lógicas ao consultar um médico (GAMA *et al.*, 2011). Essa abordagem é conhecida como Sistemas Especialistas (SE).

2.3.1 Sistemas Especialistas

Estes sistemas atuam como colaboradores na tomada de decisão em áreas dominadas por especialistas humanos. Um Sistema Especialista (SE) condensa este conhecimento armazenado para auxiliar na resolução de problemas do usuário, atuando em áreas e tarefas bem definidas (MENDES, 1997). Estruturalmente, todo SE é constituído de duas partes principais: a Base de Conhecimento, que contém o conhecimento heurístico e fatual sobre o domínio de aplicação, e a Máquina de Inferência, que utiliza a base de conhecimento para construir a linha de raciocínio que leva à solução do problema (RUSSELL; NORVIG, 2004).

O processo para adquirir uma base de conhecimento geralmente envolvia entrevistas com os especialistas para descobrir quais regras eles utilizavam ao tomar uma decisão. Entretanto, esse processo possuía várias limitações, como a subjetividade de acordo com o especialista entrevistado. Nas últimas décadas, com a crescente complexidade dos problemas a serem tratados computacionalmente e do volume de dados gerados por diferentes setores, evidenciou-se a necessidade de ferramentas computacionais mais sofisticadas e autônomas, reduzindo a necessidade de intervenção humana e dependência de um profissional da área (GAMA *et al.*, 2011).

Um sistema especialista atual apresenta uma arquitetura com três critérios: base de regras, memória de trabalho e motor de inferência. A base de regra e a memória de

trabalho formam a base de conhecimento do SE, e representam o conhecimento sobre o domínio. O motor de inferência é o mecanismo de controle do sistema que avalia e aplica as regras de acordo com as informações da memória. A memória de trabalho deve seguir um modelo de representação de conhecimento, ou seja, possuir uma linguagem formal e uma descrição matemática. A base de regras contém condições que representam "perguntas" onde envolvem variáveis a serem instanciadas e algum tipo de inferência (GAMA *et al.*, 2011).

Os sistemas especialistas devem ter algumas capacidades essenciais nos dias de hoje, tais como interagir com os usuários de forma amigável, trabalhar com incertezas, oferecer explicações quanto ao seu raciocínio e continuar aprendendo (MENDES, 1997).

Por tratar-se de sistemas dotados de inteligência e conhecimento, os benefícios advindos da utilização da técnica de sistema especialista se diferem daqueles obtidos pelos sistemas tradicionais. Dentre as vantagens, Mendes (1997) destaca:

a. Estende as facilidades de tomada de decisão para diversas áreas;

b. Melhora a produtividade e desempenho de seus usuários, considerando que o sistema fornece um vasto conhecimento, que em condições normais, demandaria mais tempo para analisá-lo. Dessa forma, todo o processo de análise e tomada de decisão torna-se muito mais eficiente;

c. Reduz o grau de dependência à falta de um especialista e a gestão de conhecimento. Um problema enfrentado por muitas empresas é a concentração de informações em determinados funcionários, uma vez em que ocorre algum imprevisto pode prejudicar o andamento das atividades. Ao registrar o conhecimento de colaboradores nos sistemas especialistas, promove-se uma significativa redução no grau de dependência entre empresa e presença física do colaborador.

Um exemplo simples de SE foi o desenvolvimento de conjunto de regras para definir quais clientes de um determinado supermercado deveriam receber a propaganda de um novo produto utilizando dados de compras passados dos clientes cadastrados na base de dados do supermercado. Esse processo de indução de uma hipótese (ou aproximação de função) a partir de uma experiência passada denomina-se Aprendizado de Máquina (do inglês, *"Machine Learning"*) (GAMA *et al.*, 2011).

2.4 APRENDIZADO DE MÁQUINA

Atividades como memorizar, observar e explorar situações para aprender fatos, melhorar habilidades motoras ou cognitivas por meio de práticas de organizar conhecimento é considerada essencial para um comportamento inteligente (STUART *et al.*,

1997). Mitchell (1997) define Aprendizado de Máquina (AM) como a capacidade de melhorar o desempenho na realização de alguma tarefa por meio de experiência.

Em AM, computadores são treinados para aprender com eventos anteriores. Para isso, empregam um princípio de inferência denominada indução, no qual obtêm-se conclusões a partir de um conjunto de exemplos. Assim, esses algoritmos aprendem a induzir uma função ou hipótese capaz de resolver um problema a partir de dados que representam instâncias do problema raiz (GAMA *et al.*, 2011).

2.4.1 Indução de Hipóteses

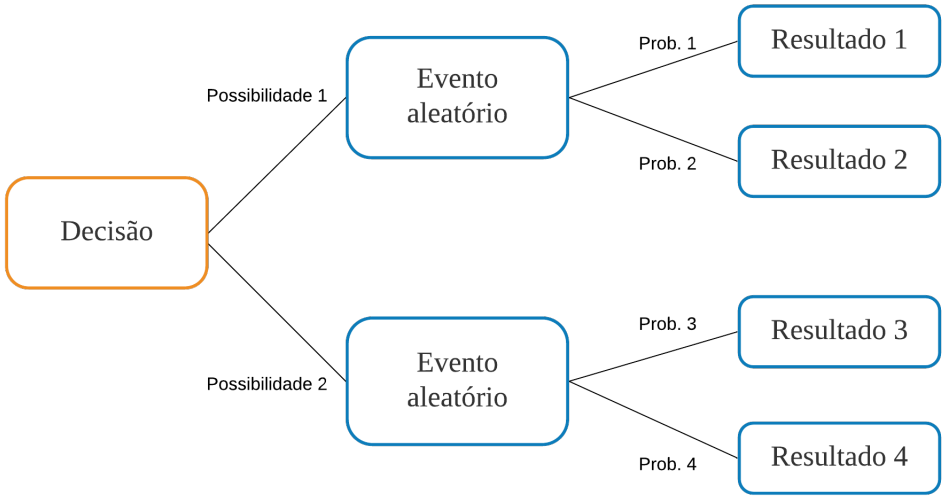
Em um modelo de *Machine Learning* denominamos atributo meta os valores que serão estimados utilizando os valores dos demais atributos, denominados de entrada ou previsores. Um requisito importante para um algoritmo de AM é que seja capaz de lidar com dados imperfeitos. Muitos conjuntos de dados apresentam algum tipo de problema, como presença de ruídos, dados inconsistentes, dados faltantes ou redundantes. Neste caso, são aplicadas algumas ferramentas para pré-processamento dos dados visando minimizar a ocorrência deste problema (GAMA *et al.*, 2011).

Ao induzir uma hipótese, é necessário que ela também seja validada para outros dados do mesmo domínio. Com esta finalidade, após a realização do pré-processamento, os dados são divididos em dois conjuntos: treino e teste. O conjunto de treino será aquele utilizado para a criação e treinamento do modelo, enquanto que o conjunto de teste será utilizado para validação do aprendizado (STUART *et al.*, 1997).

Essa propriedade de validação da hipótese é chamada de capacidade de generalização. Quando uma hipótese apresenta baixa capacidade de generalização, pode significar um super ajustamento aos dados (*overfitting*), neste caso, é dito que a hipótese memorizou ou se especializou nos dados de treinamento. Caso contrário, o algoritmo pode induzir hipóteses que apresentem baixa taxa de acerto, configurando uma condição de sub ajustamento (*underfitting*). Este caso pode ocorrer quando os dados de treinamento são poucos representativos ou estamos utilizando um modelo muito simples que não captura os padrões existentes, ou ainda, o conjunto de dados ser esparso ou apresentar ruído significativo (MONARD; BARANAUSKAS, 2003).

Ao aprender a partir de um conjunto de dados de treinamento, o algoritmo de AM procura por uma hipótese no espaço de possíveis hipóteses, onde cada algoritmo possui uma representação diferente para descrever uma hipótese indutiva. A representação utilizada define a preferência ou viés de representação do algoritmo (GAMA *et al.*, 2011). Na Figura 3 podemos visualizar o viés de representação de uma árvore de decisão.

Figura 3 – Representação de uma árvore de decisão.



Fonte – Adaptado de Gama *et al.* (2011).

Além do viés de representação, cada algoritmo também possui um viés de busca, que trata-se da forma como o algoritmo busca a hipótese que melhor se ajusta aos dados do treinamento. Ele define a forma como as hipóteses serão pesquisadas no espaço de hipóteses. Como exemplo, em uma árvore de decisão, o viés de busca é a preferência de uma árvore com poucos nós (STUART *et al.*, 1997).

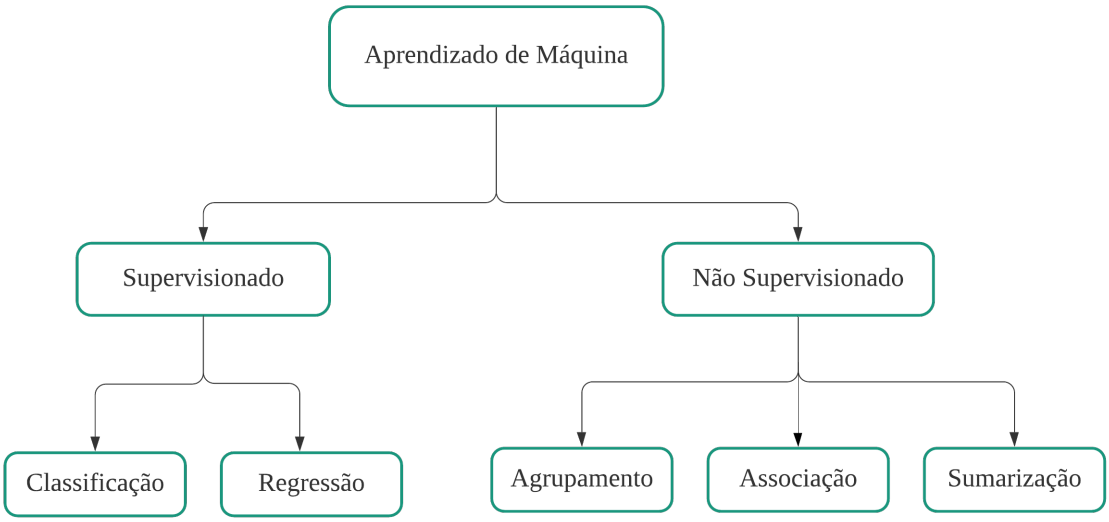
2.4.2 Paradigmas de Aprendizado de Máquina

Os algoritmos de AM diferem de acordo com sua aplicação. Segundo bacstanlar2014introduction temos dois tipos de aprendizado de máquinas:

- Não supervisionado: quando o conjunto de dados possui apenas entradas. Neste caso, não se tem rótulos dos dados. Tratam-se de modelos descritivos, onde o objetivo é explorar ou descrever um conjunto de dados;
- Supervisionado: quando o conjunto de dados contém tanto as entradas quanto as saídas, então estamos tratando do aprendizado supervisionado, ou ainda, aprendizado rotulado ou aprendizado com professor. Neste caso, conhece-se a saída para cada exemplo e podemos avaliar a capacidade da hipótese induzida predizer os valores de saída para novos modelos. Os modelos preditivos seguem esse paradigma.

Na Figura 4 está representada a hierarquia de aprendizado mostrando suas aplicações. Neste trabalho será utilizado o modelo supervisionado de regressão.

Figura 4 – Hierarquia de Aprendizado.



Fonte – Adaptado de Gama *et al.* (2011).

2.5 MODELOS PREDITIVOS

Um algoritmo de AM preditivo é uma função onde, dado um conjunto de exemplos rotulados, constrói-se um estimador. O *output* toma valores em um domínio conhecido. Tem-se um problema de classificação se o domínio for um conjunto de valores nominais, neste caso, o estimador será um classificador. Se o domínio for um conjunto infinito e ordenado de valores temos um problema de regressão, onde será gerado um regressor (GAMA *et al.*, 2011).

Dado um conjunto de observações de pares $D = \{(x_i, f(x_i)), i = 1, \ldots, n\}$ em que f representa uma função desconhecida, um algoritmo de AM preditivo aprende uma aproximação de $\hat{f}$ da função desconhecida f, onde é possível estimar o valor de f para novas observações de x (STUART *et al.*, 1997).

O algoritmo do tipo classificador cria uma fronteira de decisão que separa os exemplos da classe 1 da classe 2. Dado um conjunto de dados com duas classes, se os exemplos da classe 1 forem linearmente separáveis da classe 2, a fronteira de decisão será uma reta. Caso contrário, será necessária uma combinação de retas. Já para exemplos com mais de 2 classes, são utilizados planos de separação (PEDREGOSA *et al.*, 2011). Diferentes algoritmos podem encontrar diferentes fronteiras de decisão e diferenças nos conjuntos de treinamento podem gerar variações na ordem de apresentação dos exemplos durante o processo, corroborando para que um mesmo algoritmo de AM encontre fronteiras diferentes em cada novo treino (GAMA *et al.*, 2011).

2.5.1 Modelos de Regressão

2.5.1.1 Rede Neural Artificial

O processo de aprendizagem humano ocorre através do processamento de informações diversas, essas informações são recebidas e processadas pelo cérebro humano. Diariamente, executamos tarefas que requerem a ação de diferentes componentes, como: memória, aprendizado e coordenação física (GAMA *et al.*, 2011). A realização dessas tarefas é permitida pela complexidade da nossa estrutura biológica, onde conta um sistema capaz de transmitir sinais entre as diferentes partes do corpo e coordenar as suas ações voluntárias e involuntárias: o sistema nervoso (RUSSELL; NORVIG, 2004).

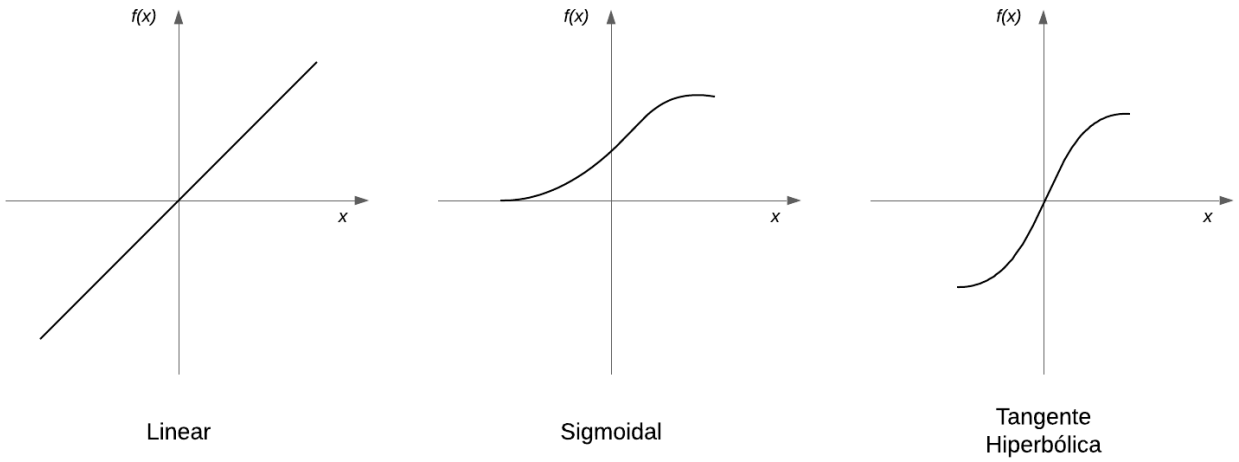
Neste contexto, as redes neurais artificiais surgiram com a inspiração de reproduzir o funcionamento do sistema nervoso e simular a capacidade de aprendizado do cérebro humano. Na biologia, o cérebro humano é composto por um grande número de neurônios, podendo variar de 10 a 500 bilhões. Estes encontram-se organizados em módulos, onde cada módulo possui cerca de 500 redes neurais. Os neurônios possuem conexões entre si que possibilita a relação entre centenas ou milhares de outros neurônios. Essas conexões permitem com que o cérebro humano seja capaz de processar informações com alta velocidade e realizar diversas tarefas ao mesmo tempo (GAMA *et al.*, 2011).

Já na tecnologia, os neurônios artificiais computam funções matemáticas, esses neurônios ficam dispostos em uma ou mais camadas interligadas por conexões, assim como no cérebro humano. Assim, a RNA é capaz de reconhecer padrões, ou seja, possui a capacidade de aprender por meio de exemplos e de generalizar a informação aprendida, gerando um modelo não-linear (SOARES *et al.*, 2015).

Durante o processo de aprendizado supervisionado, a rede realiza um ajustamento dos pesos entre as conexões de processamento, segundo uma determinada lei de aprendizagem, até que o erro entre os padrões de saída gerados pela rede alcance um valor mínimo desejado (BRAGA, 2000).

Nas RNAs, cada entrada do neurônio receberá um valor, esses valores serão ponderados e combinados por uma função f_a. A saída da função será a resposta do neurônio de camada posterior. Diversas funções podem ser utilizadas em problemas de RNA, dentre elas, damos um destaque para a função linear, sigmoidal e tangente hiperbólica.

Figura 5 – Exemplos de funções de ativação para RNA

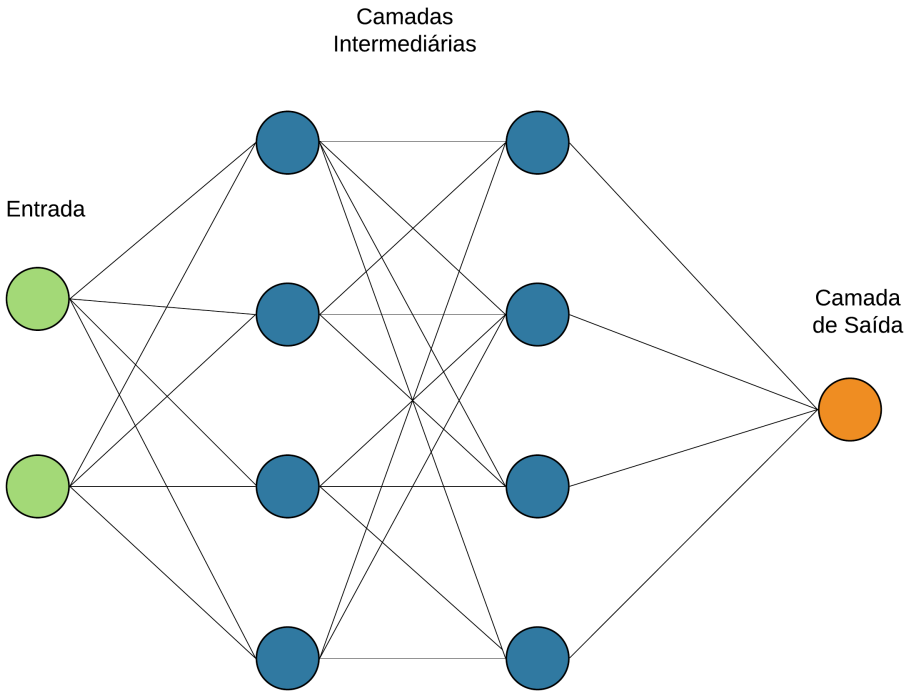


Fonte – Adaptado de Gama *et al.* (2011).

A arquitetura da rede neural artificial é dividida em camadas. Os neurônios podem estar disposto em uma ou mais camadas. Em redes com número de camadas superior a 2, um mesmo neurônio pode receber valores de saída da camada anterior, ou enviar informações para a camada seguinte, esta denominamos de camada intermediária ou camada oculta.

A Figura 6 ilustra uma RNA com 3 camadas, onde possui duas camadas intermediárias e gera um valor na camada de saída.

Figura 6 – Exemplos de uma RNA com três camadas



Fonte – Adaptado de Gama *et al.* (2011).

Boeri *et al.* (2013) otimizaram as condições de secagem a fim de obter os parâmetros de secagem ótimos para a redução do consumo energético. Os autores realizaram uma simulação utilizando uma rede neural formada por uma camada de entrada, com 4 neurônios, uma camada oculta com 9 neurônios e uma camada de saída formada por 1 neurônio. Sendo as variáveis de entrada: tempo, umidade relativa, velocidade e temperatura e a saída sendo o custo energético. Foram utilizadas as funções de transferência tangente sigmoidal e linear para as camadas oculta e de saída, respectivamente. Com isso, os autores encontraram os melhores parâmetros para obter o menor consumo energético.

Já Soares *et al.* (2015) avaliaram o desempenho das redes neurais artificiais na predição da produtividade da cultura do milho. Como variáveis na camada de entrada, os autores avaliaram o índice de área foliar, matéria verde total, altura de planta e quantidade de planta por metro quadrado. Essas variáveis foram inseridas para predizer a produção de grãos. Cada arquitetura foi treinada 10 vezes, escolhendo-se, ao final do treinamento, aquela com menor erro relativo médio e menor variância em relação aos dados de validação. A partir dos resultados do erro quadrático médio, pode-se concluir que as RNAs são eficientes, podendo ser utilizadas como ferramenta para estimar a produtividade de grãos da cultura do milho.

Além disso, Susama Chokphoemphun e Suriya Chokphoemphun (2018) projetaram um modelo de rede neural artificial de múltiplas camadas para prever a proporção de umidade do arroz durante o processo de secagem em um secador de leito fluidizado. Os modelos foram projetados com diferentes números de camadas ocultas (1-3) e números de nós de neurônios (2-12) na camada oculta para encontrar o melhor modelo. O melhor desempenho para a previsão da taxa de umidade da secagem do arroz foi para a estrutura de 3–2–2–1, que teve um coeficiente de determinação de regressão (R^2) de 0,995 e um erro quadrático médio (MSE) de $1,988 \times 10^{-4}$.

2.5.1.2 Máquinas de Vetores de Suporte

As máquinas de vetores de suporte (do inglês, *Support Vector Machine* (SVM)) são técnicas embasadas na teoria do aprendizado estatístico, onde visa estabelecer condições matemáticas que permitam a escolha de um classificador *f'* com bom desempenho para os conjuntos de treinamento e teste, ou seja, busca-se uma função *f'* capaz de classificar os dados de treinamento com maior assertividade (LORENA; CARVALHO, 2003). Dados as entradas de treinamento, as SVMs utilizam regressão para encontrar um hiperplano que melhor separa as classes de entradas. Uma vez que a máquina de vetores de suporte tenha sido treinada, ela é capaz de avaliar novas entradas em relação ao hiperplano divisor e classificá-las entre as categorias (GAMA *et al.*, 2011).

Este modelo vêm sendo utilizado em diversas tarefas de reconhecimento de padrões, obtendo resultados superiores aos alcançados por outras técnicas de aprendizado em várias aplicações, ele é especialmente indicado em dados onde uma ou mais das condições a seguir se manifestam (VON ZUBEN; ATTUX, 2013):

- Baixa amostragem;
- Altos níveis de ruídos nos dados;
- Dados com muitas entradas.

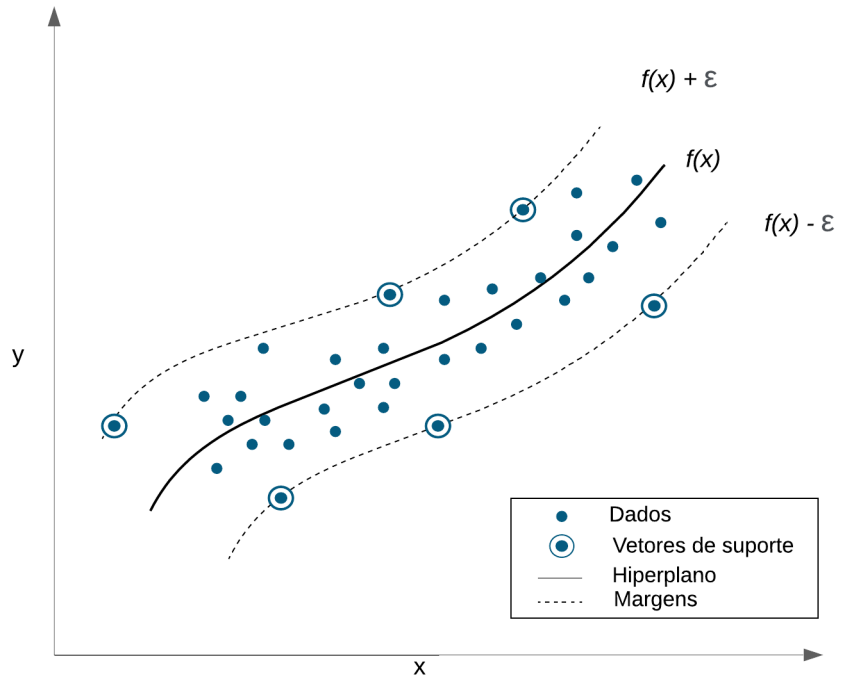
Inicialmente, a técnica de *Support Vector Machine* foi introduzida para modelos de classificação. Entretanto, os conceitos de SVM podem ser generalizados para se tornarem aplicáveis a problemas de regressão. Seguindo os conceitos do SVM, a técnica de regressão treina usando uma função de perda simétrica, que penaliza igualmente as estimativas erradas. Uma das principais vantagens desta abordagem é que sua complexidade computacional não depende da dimensionalidade do espaço de entrada. Além disso, possui excelente capacidade de generalização, com alta precisão de predição (AWAD; KHANNA, 2015).

O algoritmo SVM é um método baseado em kernel, onde tenta encontrar o melhor hiperplano de regressão com o menor risco estrutural em um espaço de alta

dimensão (YEH *et al.*, 2011). O kernel K é uma função que recebe dois pontos x_i e x_j no espaço de entradas e calcula o produto escalar desses objetos no espaço de características (GAMA *et al.*, 2011). Alguns dos kernels mais utilizados são os polinomiais, os de função base radial (RBF) e os sigmoidais.

O algoritmo mais utilizado do tipo SVM é o ϵ-svm, no qual localiza o hiperplano com um ϵ como função de perda. Na Figura 7 é apresentado um exemplo do procedimento para um modelo linear.

Figura 7 – Ilustração do procedimento realizado pelo algoritmo SVM



Fonte – Adaptado de Gama *et al.* (2011).

O modelo proposto vem sendo empregado nas mais variadas aplicações. Dando um destaque para a área de engenharia de processos, Das e Akpinar (2018) estabeleceram um modelo preditivo para os valores de coeficiente de transferência de calor por convecção de diferentes métodos de secagem, a fim de determinar qual método de secagem de peras seria mais rápido. Os autores avaliaram a secagem natural e a secagem com ar quente. Três modelos de kernel diferentes foram usados na regressão da máquina de vetor de suporte. Como resultado, a taxa de secagem da pera foi examinada para ambos os sistemas e foi observado que a pera havia secado mais cedo no sistema de secagem com ar aquecido e o kernel polinomial foi determinado como o melhor modelo para estimar os valores do coeficiente de transferência de calor.

Younis *et al.* (2019) aplicaram o modelo de SVM, rede neural artificial (RNA) e a regressão do processo gaussiano (GPR) para otimização do processo de remoção do amargor da casca de Mosambi. O objetivo da otimização foi remover o amargor de maneira econômica e oportuna, mantendo o máximo possível de polifenóis e atividade antioxidante. A otimização resultou na redução do tempo de imersão em cerca de 1 hora para render quase o mesmo polifenol total e atividade antioxidante. Assim, a otimização usando diferentes ferramentas de aprendizado de máquina mostrou que polifenóis máximos podem ser retidos mantendo o sabor em uma faixa aceitável. O pó de casca otimizado apresentou boas propriedades funcionais, portanto, pôde ser incorporado na cadeia alimentar humana.

2.5.1.3 Vizinhos mais próximos

Outro método muito utilizado em projetos de *Machine Learning* é o algoritmo dos vizinhos mais próximos (do inglês, *Nearest Neighbors*). Nesta técnica, cada objeto representa um ponto definido no espaço de entrada. A partir dessas entradas, é possível calcular as distâncias entre cada ponto por meio de uma função objetivo. Sendo assim, o algoritmo classifica um novo objeto com base nos experimentos do conjunto de treinamento mais próximos a ele, considerando *k* vizinhos (GAMA *et al.*, 2011).

A regressão de vizinhos mais próximos usa pesos uniformes, ou seja, cada ponto na vizinhança local contribui uniformemente para a classificação de um ponto de consulta. Em algumas circunstâncias, pode ser vantajoso ponderar os pontos de forma que os pontos próximos contribuam mais para a regressão do que os pontos distantes (PEDREGOSA *et al.*, 2011).

Embora seja um algoritmo relativamente simples, uma vez que ele trabalha com a memorização dos objetos de aprendizagem, ele é aplicável mesmo em problemas mais complexos. Além disso, o algoritmo é naturalmente incrementável, pois ao adquirir novos exemplos de treinamento, basta armazena-los na memória. (GAMA *et al.*, 2011).

Taur *et al.* (2019) utilizaram o modelo KNN para prever a taxa de teor de umidade no processo de secagem de uma industria têxtil. As entradas do modelos incluíram

o tipo de fibra, largura, peso, densidade, configuração de temperatura da máquina e velocidade da correia transportadora. Para selecionar um valor k apropriado, os autores fizeram uma varredura de valores k com o mesmo conjunto de dados de treinamento. A escolha do algoritmo de busca de vizinhos foi automática e a melhor escolha foi decidida pelo algoritmo fornecido por Pedregosa *et al.* (2011). Como resultado, os autores obtiveram um coeficiente de determinação de 0,9597 e um erro quadrático médio de 0,0811 quando k foi igual a 7.

Já Khaled *et al.* (2020) descreveram a cinética de secagem de fatias de frutos de caqui durante a secagem a vácuo e a secagem a ar quente sob diferentes temperaturas de 50 °C, 60 °C e 70 °C, utilizando rede neural artificial de várias camadas, máquina de vetor de suporte e vizinhos k-mais próximos (k-NN). Como resultado, os modelos RNA, SVM e k-NN apresentaram valores de R^2 de 0,9994, 1,0000, 0,932, respectivamente. A validação dos modelos indicaram boa concordância entre os valores previstos obtidos a partir dos métodos e os dados experimentais de umidade.

2.5.1.4 *Random Forest*

Breiman (2001) propôs o modelo de florestas aleatórias para a construção de um conjunto de preditores com um conjunto de árvores de decisão que crescem em subespaços de dados selecionados aleatoriamente. Uma Árvore de Decisão (AD) usa como estratégia a divisão do problema em casos mais simples. As soluções dos sub-problemas podem ser combinadas para produzir uma solução do problema complexo, ou seja, realiza a divisão do espaço em subespaços e cada subespaço é ajustado usando diferentes modelos (GAMA *et al.*, 2011). Uma árvore de decisão trata-se de um grafo acíclico constituído de nós, onde em cada nó da árvore o algoritmo escolhe o atributo dos dados que melhor particiona o conjunto de amostras em subconjuntos (QUINLAN, 2014).

Algumas vantagens das ADs são a compreensibilidade das regras de classificação produzidas, a facilidade de manutenção, flexibilidade e velocidade de treinamento. Por outro lado, são pouco robustas à exemplos de elevada dimensionalidade e apresentam certa instabilidade com a ocorrência de pequenas variações no conjunto de treinamento (LIBRALON, 2007).

Em uma Floresta Aleatória, o algoritmo irá escolher N conjuntos de amostras do conjunto de dados original e então realizar os cálculos com base nas amostras selecionadas, para definir qual dessas será utilizada no primeiro nó. Em seguida, serão escolhidas duas ou mais variáveis excluindo aquelas selecionadas anteriormente e o processo de escolha se repetirá. Desta forma, a árvore será construída até o último nó. O mesmo ocorre para a criação de todas as árvores que irão compor a floresta (CÁNOVAS-GARCÍA *et al.*, 2017).

Um preditor de regressão *Random Forest* pode ser expresso como:

$$\hat{f}_{RF}^{C} = \frac{1}{C}\sum_{i=1}^{C} T_i(x) \tag{1}$$

Onde x é a variável de entrada vetorial, C é o número de árvores, e $T_i(x)$ é uma árvore de regressão única construída com base em um subconjunto de variáveis de entrada e as amostras inicializadas (AHMAD *et al.*, 2018). Para a validação do modelo de regressão, cada árvore criada irá apresentar o seu resultado, e o resultado final será a média dos valores previstos em cada uma (GAMA *et al.*, 2011).

Tacchella *et al.* (2017) estudou a combinação de previsões feitas por seres humanos com as de um algoritmo de aprendizado de máquina sobre a progressão da esclerose múltipla em um conjunto de pacientes. Por meio de um conjunto de dados com 525 registros clinicos de 84 pacientes com Esclerose Múltipla. Parâmetros como idade, tempo para concluir uma tarefa, escore clínico ou presença/ausência de cada sintoma foi observado em cada visita realizada pelo paciente, onde foi registrado se o paciente estava no estágio RR (remitente-recorrente) ou SP (progressiva secundária) após 180, 360 e 720 dias. Foi elaborado um modelo de classificação supervisionada onde 0 significa "ainda na fase RR"e 1 indica "transição para a fase SP, utilizando a abordagem *Random Forest*. Em paralelo ao desenvolvimento do modelo, 42 estudantes dos últimos anos do curso de Medicina em Roma avaliaram 50 prontuários, extraídos aleatoriamente do mesmo conjunto de dados usado para aprendizado de máquina e estimaram a probabilidade de o paciente progredir para a fase SP dentro de 180, 360 e 720 dias. Em seguida, foi realizada uma integração das previsões humanas e computacionais em uma previsão híbrida, que combina o raciocínio clínico humano com a abordagem de classificação dos algoritmos. Como resultado, observou-se uma melhora significativa da capacidade preditiva quando as previsões foram combinadas, ou seja, a previsão híbrida homem-máquina produz melhores prognósticos do que algoritmos de aprendizado de máquina ou grupos de seres humanos isolados.

2.5.2 *Stacked Generalization*

Segundo Breiman (1996) SG é um método para formar combinações lineares de diferentes preditores e fornecer um modelo com maior precisão. Este algoritmo é um esquema para estimar e corrigir o erro de um generalizador que foi treinado em um conjunto de aprendizagem específico, reduzindo os vieses de cada modelo (WOLPERT, 1992). Ele consiste em empilhar as previsões de cada estimador individual e usar como entrada para um estimador final para calcular a previsão. Este estimador final é treinado por meio de validação cruzada (PEDREGOSA *et al.*, 2011).

Em um primeiro momento, os modelos iniciais são treinados com o conjunto de treino, em seguida, um combinador é treinado para fazer uma previsão final com base

nas previsões dos modelos básicos. Esses conjuntos empilhados tendem a superar qualquer um dos modelos individuais (BOEHMKE; GREENWELL, 2019).

Ma *et al.* (2018) relatam que com apenas um único algoritmo, o desempenho de previsão pode atingir um limite superior mesmo com parâmetros ideais. Com isso, desenvolveram um conjunto de algoritmos de aprendizado de máquina usando a abordagem de generalização empilhada para estimar a dose de varfarina. Inicialmente, um conjunto de dados com 6256 usuários crônicos de varfarina foi treinado com 8 modelos de *Machine Learning*, incluindo RNA, SVM, *Random Forest* e KNN, os erros absolutos de cada modelo variaram entre 8,5% e 10%. Ao aplicar a técnica de empilhamento, o resultado obtido foi de 8,32% para o erro absoluto.

Já Wu *et al.* (2019) desenvolveram dois modelos baseados em empilhamento onde analisou-se duas propriedades dinâmicas de uma gota de água, o ângulo de contato e ligações de hidrogênio. Os dois modelos consistiam em um conjunto de *Random Forest*, RNA, SVR, KNN e *Kernel Ridge Regression*. Os valores de erro quadrático médio de ambos os modelos sugeriram que o resultado final foi mais preciso em comparação com os modelos individuais.

3 MATERIAIS E MÉTODOS

Um projeto de *machine learning* consiste em 6 etapas fundamentais: obtenção do conjunto de dados, análise estatística, pré processamento, separação dos dados, escolha e seleção de hiperparâmetros, avaliação e validação do modelo.

3.1 OBTENÇÃO DO CONJUNTO DE DADOS

Os dados utilizados para o desenvolvimento do trabalho foram adquiridos por meio de uma parceria realizada com uma indústria do setor de alimentos. A respectiva indústria desenvolve produtos e coprodutos da acerola, entre eles, está a obtenção do pó de acerola. Este subproduto é comercializado como ingrediente do suplemento alimentar rico em vitamina C.

Conforme já destacado na revisão bibliográfica, é de primordial importância a etapa de levantamento das variáveis para o desenvolvimento do modelo. Desta forma, buscou-se trabalhar com as variáveis que influenciam diretamente no processo de secagem.

Durante a etapa de coleta de dados, as amostras foram tomadas diretamente de um único secador e um único produto, eliminando-se assim interferências como diferenças de capacidade de secagem de demais secadores, diferença de especificação e condições de processo para cada produto. Os lotes produzidos entre o período de 01/01/2019 à 11/06/2020 foram utilizados para o treinamento do modelo.

As variáveis associadas ao sistema de secagem são divididas em dois grupos: propriedades intrínsecas e variáveis de processo.

1. Propriedades Intrínsecas: São as variáveis relacionadas ao produto. Antes de iniciar a secagem, o suco de acerola passa por uma formulação onde é elaborado uma mistura com demais coadjuvantes. Nesta etapa são realizados alguns ajustes da qualidade do produto. Todas as medições deste grupo foram coletados de forma manual.

    - Brix (B): Dentro esses ajustes, está o teor de sólidos solúveis. Esta medida de concentração serve também como um parâmetro indicativo para a etapa de secagem, uma vez que altas concentrações podem interferir na fluidez do produto.
    - pH inicial (pH): Outro fator importante no inicio do processo é o pH. Antes de iniciar o processo, deve-se conhecer o pH da mistura.
    - Vitamina C Inicial (V_i): Para obter o rendimento do processo, o teor de vitamina C foi medido após a formulação do produto concentrado, sendo expressa em porcentagem.

- Vitamina C Final (V_f): Ao final do processo de secagem, as amostra foram enviada ao laboratório onde foi medido o teor de vitamina C final. Essa variável foi utilizada como resposta no modelo desenvolvido.

2. Variáveis de processo: As variáveis de processo foram coletadas a partir de um Controlador Lógico Programável (CLP) integrado com um sistema para coletas e análises de dados. Todas as temperaturas foram expressas em graus Celsius.

    - Vazão do produto (F): Vazão de entrada do produto na câmara de secagem.
    - Temperatura do Tanque de Alimentação (T_{tq}): Temperatura média de cada lote a qual as amostras foram submetidas no tanque de alimentação anterior à câmara de secagem.
    - Temperatura de entrada do ar na câmara de secagem (T_{ear1}): Temperatura média por lote durante a injeção de ar quente na câmara de secagem.
    - Temperatura de saída do ar da câmara de secagem (T_{sar1})
    - Temperatura de saída do produto da câmara (T_{sp1})
    - Temperatura de entrada do ar no ciclone (T_{ear2})
    - Temperatura de saída do produto no ciclone (T_{sp2})
    - Rotação do atomizador (RA)
    - Amperagem do atomizador (AA)
    - Temperatura da câmara de secagem (T_1): Média da temperatura interna da câmara de secagem no topo da mesma.
    - Pressão da câmara de secagem (P_1): Média da pressão interna na câmara de secagem durante o processamento
    - Umidade (U): Para obter a curva de secagem, a umidade do produto é medida em intervalos de tempo pré definidos ao longo do processo. Para o desenvolvimento deste projeto, foi considerado apenas a umidade final para cada lote.

Com isso, obteve-se um conjunto e dados com 16 variáveis e 321 registros.

O desenvolvimento de todas as etapas a seguir foram realizadas utilizando o Google Colab (GOOGLE, 2020), que trata-se de um serviço na nuvem gratuito hospedado pelo Google para incentivar a pesquisa de Aprendizado de Máquina e Inteligência Artificial. Ele permite a criação de códigos utilizando bibliotecas Python.

3.2 ANÁLISE ESTATÍSTICA

Para a análise dos dados, o conjunto de dados foi importado no formato .csv e avaliou-se a distribuição dos dados obtidos, por meio dos valores de amplitude, desvio

padrão e intervalo interquartil. Essa etapa foi fundamental para analisar a presença de dados ausentes e *outliers*.

Uma matriz de correlação foi gerada para descrever a associação entre as variáveis do conjunto de análise, onde obteve-se uma tabela mostrando coeficientes de correlação entre variáveis, exibindo os valores de correlação de Pearson.

O coeficiente de correlação de Pearson (ρ) quantifica a força de associação linear entre duas variáveis e pode assumir valores entre -1 a 1. Valores próximos a 1 representa uma correlação forte e positiva entre duas variáveis, ou seja, as variáveis são diretamente dependentes. Já valores que tendem para -1 representam uma correlação forte negativa e indicam que as variáveis são inversamente dependentes.entre as variáveis (GALARÇA *et al.*, 2010). Para obter o coeficiente de Pearson, o seguinte calculo é necessário:

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2}$$

A análise de correlação foi utilizada para definir os parâmetros a serem utilizados no modelo, visando considerar as variáveis de maior correlação para o treinamento.

3.3 PRÉ PROCESSAMENTO DE DADOS

Devido ao desempenho dos modelos de inteligência artificial estarem ligados diretamente aos dados utilizados, a qualidade de dados é uma das principais preocupações em *Machine Learning*. Frequentemente, os dados apresentam diversos problemas, tais como grande quantidade de valores desconhecidos, ruídos, desproporcionalidade da variável de saída, entre outros. Se os problemas presentes nos dados forem identificados e tratados antes dos dados serem fornecidos a um algoritmo de extração de conhecimento, então espera-se que o conhecimento extraído seja mais representativo e mais preditivo (BATISTA *et al.*, 2003). As técnicas aplicadas nesta etapa visam preparar os dados para que a fase de aprendizagem seja mais efetiva. Desse modo, quatro técnicas foram utilizadas para diminuir os ruídos no conjunto de dados, sendo elas:

3.3.1 Detecção de *outliers*

Uma das técnicas gráficas mais utilizadas para analisar um conjunto de dados univariados é o boxplot, ou diagrama de caixa, proposto por Tukey (1977). O boxplot é um gráfico utilizado para auxiliar na identificação de valores discrepantes de um conjunto de dados (chamados de *outliers*) e sua construção baseia-se na divisão do conjunto de dados em quatro sub-intervalos, cada um com 25% das observações (NAVIDI, 2012).

O primeiro intervalo é delimitado pelo menor valor do conjunto de dados e Q1 (quartil1). A partir do menor ponto do conjunto de dados até o segmento vertical, encontram-se 5% dos valores do conjunto de dados, do segmento vertical até o 1º segmento do quadro amarelo, encontram-se 20% dos valores do conjunto de dados, totalizando 25% dos valores. O segundo intervalo está situado entre Q1 e Q2 (quartil2 – mediana). Neste intervalo encontram-se mais 25% dos valores do conjunto de dados e é representado pela 1ª parte da caixa amarela. Já o 3º intervalo é a diferença entre Q2 e Q3 (quartil3), onde encontram-se mais 25% dos valores do conjunto de dados. E por último, o quarto intervalo é delimitado pelo Q3 e o valor máximo (HARBOR, 2017).

Para a identificação de *outliers* foi utilizada o intervalo interquartil (IQR), onde apresenta a diferença entre Q3 e Q1. Uma regra bastante utilizada afirma que um dado é um *outlier* quando ele é maior que 1.5$\times$ IQR acima do terceiro quartil ou abaixo do primeiro quartil (HUBERT; VANDERVIEREN, 2008).

Os *outliers* de cada coluna foram substituídos por valores nulos, que serão tratados posteriormente

3.3.2 Valores ausentes

Os valores não existentes oriundos de coletas que não foram realizadas na produção ou da remoção de outliers foram tratados utilizando o algoritmo *KNN Imputer*.

Segundo Batista *et al.* (2003) este método obtém resultados superiores quando comparados aos métodos de substituição pela média, mediana ou moda.

O *KNN Imputer* substitui os valores ausentes de cada amostra usando o valor médio dos vizinhos mais próximos encontrados no conjunto de treinamento. Os vizinhos são encontrados utilizando a distância euclidiana. As características dos vizinhos são calculadas uniformemente ou ponderadas pela distância de cada vizinho (PEDREGOSA *et al.*, 2011).

3.3.3 Extrapolação dos dados

Uma das desvantagens encontradas ao construir um modelo de AM é trabalhar com dados que não possuem uma boa distribuição. Em situações onde os valores de treinamento possuem poucas informações em uma determinada faixa, o modelo apresentará maior erro ao tentar prever um valor deste intervalo. Por isso, têm-se à necessidade de um grande conjunto de informações experimentais para o treinamento.

A extrapolação de dados é um método matemático que busca estimar, além do intervalo de observação original, o valor de uma variável com base em sua relação com outras variáveis.

Tsen *et al.* (1996) e Stuart *et al.* (1997) propuseram uma metodologia para trabalhar com dados desbalanceados, gerando um conjunto extenso de dados experimentais suficiente para o treinamento do modelo. De forma geral, a metodologia

assume um modelo capaz de capturar as tendências do processo. A expressão para a geração dos dados aumentados é dada por:

$$f_i^a = \sum_{k=1}^{N} w_{ik} \left[f^e(m_{1k}^e, m_{2k}^e ...) + \sum_{j=1}^{M} \frac{\partial f}{\partial m_j} (m_j^a - m_{kj}^e) \right] \tag{3}$$

Onde $m_1, m_2, ..., m_M$ representem M variáveis independentes do conjunto, f representa a variável de saída, m_{kj}^e representa pontos experimentais (N), $f_a(ma_1, ma_2, ...)$ é a expressão para geração dos dados aumentados e w_{ik} é um fator de ponderação inversamente proporcional à distância entre os pontos experimentais e é dado por:

$$w_{ik} = \frac{\left(\sqrt{\sum_{j=1}^{M} (m_j^a - m_{jk}^e)^2} \right)^{-1}}{\sum_{i=1}^{N} \left(\sqrt{\sum_{j=1}^{M} (m_j^a - m_{jk}^e)^2} \right)^{-1}} \tag{4}$$

Seguindo a Equação (3) e Equação (4) um algoritmo foi implementado em Python para a geração dos dados aumentados.

3.3.4 Normalização

A normalização dos dados consiste em transformar os valores dos atributos de seus intervalos originais para uma mesma escala, como, por exemplo, [-1, 1] ou [0, 1] (BATISTA *et al.*, 2003). A padronização de um conjunto de dados é um requisito comum para muitos estimadores de aprendizado de máquina, pois muitos modelos podem ter um comportamento negativo ao utilizar recursos em diferentes escalas (PEDREGOSA *et al.*, 2011).

Muitos elementos usados na função objetivo de um algoritmo de aprendizagem assumem que todos os recursos estão centrados em torno de zero e têm variância na mesma ordem. Se uma característica tem uma variância com uma ordem superior a outras, ela pode dominar a função objetivo e tornar o estimador incapaz de aprender com outras características (PEDREGOSA *et al.*, 2011).

Com isso, aplicou-se o método de padronização de *z-score* onde cada variável quantitativa terá um valor z definido pela Equação (5), onde u é a média das amostras de treinamento e s é o desvio padrão.

$$z = \frac{(x - u)}{s} \tag{5}$$

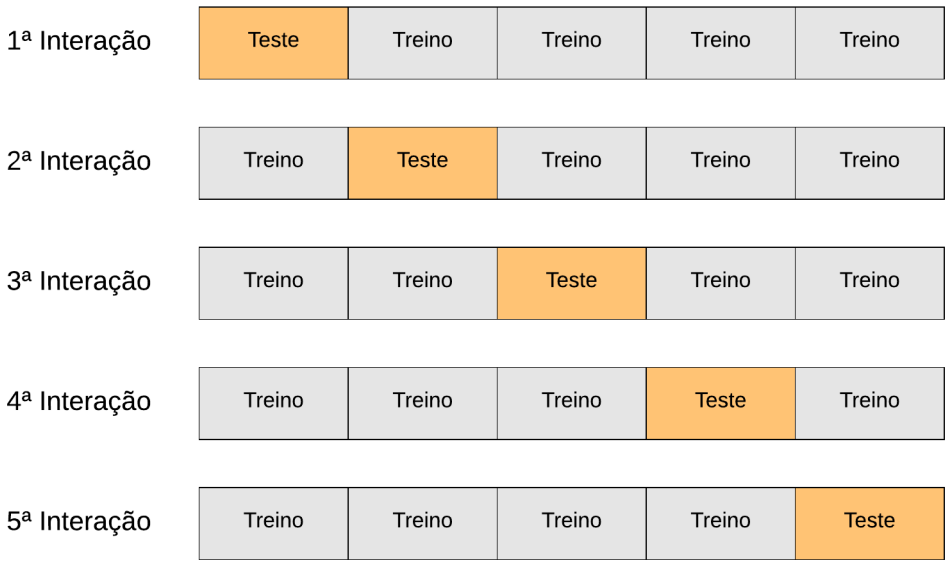
3.4 VALIDAÇÃO CRUZADA

A validação cruzada é uma técnica para avaliar a capacidade de generalização de um modelo, a partir de um conjunto de dados. Antes de treinar o modelo deve-

se separá-lo de forma aleatória em base de treino e teste. O conjunto de teste será utilizado para validar o aprendizado.

Neste caso, o conjunto de exemplos são aleatoriamente divididos em k conjuntos menores (*fold*) de tamanho aproximadamente igual. O modelo é treinado usando $k-1$ das partições como dados de treinamento e a hipótese induzida é testada no *fold* restante. Este processo é repetido k vezes, cada vez considerando um *fold* diferente (MONARD; BARANAUSKAS, 2003).

Figura 8 – Exemplo de uma validação cruzada com k=5



Fonte – Autora

3.5 APLICAÇÃO DOS MODELOS

O desempenho de todo o modelo é baseado nos valores de hiperparâmetros especificados, tais como a função objetivo utilizada, taxa de aprendizado, número de variáveis, etc. Com isso, o *Grid Search* é o processo de otimização de hiperparâmetros para determinar os valores ideais para um determinado modelo (PEDREGOSA *et al.*, 2011).

Os hiperparâmetros são especificados usando um valor mínimo (limite inferior), valor máximo (limite superior) e número de etapas. O desempenho de cada combinação é avaliado usando algumas métricas de desempenho. O objetivo é identificar a melhor combinação de hiperparâmetros para que cada modelo possa prever dados desconhecidos com precisão (SYARIF *et al.*, 2016).

Buscando aplicar esse método, foi utilizada a ferramenta *GridSearchCV* existente na biblioteca Pandas, utilizando os seguintes parâmetros:

1. Rede Neural Artificial: Neste trabalho foi utilizada uma rede neural estática do tipo multicamadas.

    - Número de Camadas Ocultas e Neurônios (*hidden_layer_sizes*): Apesar da seleção de neurônios e camadas intermediárias serem baseadas em tentativa e erro, Heaton (2008) estabeleceu três regras para serem utilizadas como ponto de partida para a escolha do número de neurônio:
        - O número de neurônios ocultos deve estar entre o tamanho da camada de entrada e o tamanho da camada de saída;
        - O número de neurônios ocultos deve ser 2/3 do tamanho da camada de entrada, mais o tamanho da camada de saída;
        - O número de neurônios ocultos deve ser menor que o dobro do tamanho da camada de entrada.

      Com isso, cinco parâmetros foram utilizados para a geração das camadas ocultas: (10,), (15,), (10,10), (10,10,10), (8,10,8). Ou seja, foram geradas cinco arquiteturas diferentes. Na primeira será gerada uma rede neural com 1 camada oculta e 10 neurônios. Na segunda, constará de uma camada com 15 neurônios. A terceira e a quarta arquitetura terá 10 neurônios com 2 e 3 camadas, respectivamente. A última arquitetura terá 8 neurônios na primeira e na última camada oculta, enquanto que a segunda terá 10 neurônios. A função de ativação determinará o peso de cada neurônio em sua respectiva camada;
    - Função de Ativação (*activation*): foram utilizadas as funções *identity, logistic, tanh e relu*, onde *identity* retorna $f(x) = x$, *logistic*, é a função sigmóidal logística e retorna $f(x) = \frac{1}{(1+exp(-x))}$. *Tanh*, retorna a função tangente hiperbólica de x e *relu* retorna $f(x) = max(0, x)$;
    - Solver: Para a otimização de pesos, foi utilizado os parâmetros *lbfgs, sgd* e *adam*;
    - Alfa: O parâmetro de penalidade foi alterado entre os seguintes valores [0,0001; 0,05; 0,1; 0,5; 1;10;100];
    - Taxas de aprendizagem (*learning_rate*): foram utilizadas as taxas constante, adaptativa e escala inversa;
    - Epsilon: variou-se a estabilidade numérica entre e^{-8} e e^{-5}.

2. K- Vizinhos mais próximos:

- Número de vizinhos (*n_neighbors*): Variou-se entre 2 à 30 vizinhos mais próximos;
- Função de peso (*weights*): Variou entre peso uniforme, onde todos os pontos são ponderados igualmente, e pelo inverso da distância, onde vizinhos mais próximos terão uma maior influência que vizinhos mais distantes.
- Algoritmo (*algorithm*): *auto, ball_tree, kd_tree* e *brute*;
- Tamanho da folha (*leaf_size*): Variando entre 30 à 100, o tamanho da folha pode afetar a velocidade de construção e consulta do modelo, bem como a memória necessária para armazenar o mesmo;
- Parâmetro de potência (*p*). Variando entre 1 ou 2.

3. *Random Forest*:

   - Critério (*criterion*): A função para medir a qualidade de uma divisão. Podendo ser o MSE (erro médio quadrático) ou MAE (erro médio absoluto);
   - O número de árvores na floresta (*n_estimators*): Variou-se entre 4 à 33 árvores;
   - Profundidade máxima da árvore (*max_depth*): Variou-se entre 4 à 11;
   - O número de variáveis a serem considerados ao procurar a melhor divisão (*max_features*) variando entre 5 á 15;

4. Máquina de vetores de suporte:

   - Kernel: *linear, rbf* e *sigmoid*;
   - Gamma: Coeficiente de kernel (*scale, auto*);
   - Parâmetro de regularização (C) com variação entre 1 e 100;
   - Epsilon: a especificação do tubo variou entre 0,1; 0,2 e 0,5.

A partir do código implementado (ANEXO 1), obteve-se os melhores estimadores para cada modelo. Os resultados obtidos foram combinados utilizando a técnica de *Stacked Generalization* (SG) com um modelo de regressão linear para obter o modelo final.

3.6 VALIDAÇÃO DOS MODELOS

Aprendizado de Máquina é uma ferramenta poderosa onde existem diversos algoritmos com desempenhos satisfatórios para um mesmo conjunto de dados (MONNARD; BARANAUSKAS, 2003). Com isso, deve-se entender o desempenho e as limitações de cada modelo utilizando metodologias de avaliação.

A avaliação de um algoritmo de AM supervisionado é normalmente realizada por meio da análise de desempenho do preditor na análise de novos objetos (MONARD; BARANAUSKAS, 2003). Ao aplicar um algoritmo em um conjunto de dados, deve-se aplicar o modelo em um conjunto exemplo diferente e real para avaliar o desempenho do modelo.

Para isso, um novo conjunto de dados foi gerado com os lotes produzidos a partir do dia 11/06/2020. Os novos dados, com 16 variáveis e 48 registros, foram utilizados para a validação do modelo.

3.6.1 Métricas de desempenho

Avaliar o algoritmo de aprendizado de máquina é uma parte essencial do projeto. As métricas de desempenho de um modelo fornecem uma indicação da qualidade do ajuste e, portanto, uma medida de quão bem as amostras foram previstas (PEDREGOSA *et al.*, 2011). As principais métricas utilizadas para modelos de regressão são: o erro quadrático médio, o erro absoluto médio e o coeficiente de determinação.

1. Coeficiente de Determinação (R^2)

   Representa a proporção da variância na variável dependente que é previsível a partir das variáveis independentes, ou seja, é uma medida estatística de quão bem as previsões de regressão se aproximam dos pontos de dados reais. Um R^2 próximo à 1 indica que as previsões de regressão se ajustam perfeitamente aos dados. Valores negativos de R^2 podem ocorrer quando o modelo selecionado não se ajusta ao hiperplano. Neste caso, o modelo selecionado não interage adequadamente os dados.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_1)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_1)^2} \tag{6}$$

2. Erro Médio Absoluto (MAE): Trata-se da média da diferença entre os valores originais e os valores previstos. Ele nos dá a medida de quão longe as previsões estão do valor real. Matematicamente, é representado como:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{7}$$

3. Erro Médio Quadrático (MSE): Semelhante ao erro médio absoluto, entretanto o MSE realiza a média do quadrado da diferença entre os valores originais e os valores previstos. A vantagem do MSE em relação ao MAE está na penalização do maior erro. À medida que tomamos o quadrado do erro, o efeito de erros

maiores torna-se mais pronunciado do que o erro menor, portanto, o modelo agora pode se concentrar mais nos erros maiores.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{8}$$

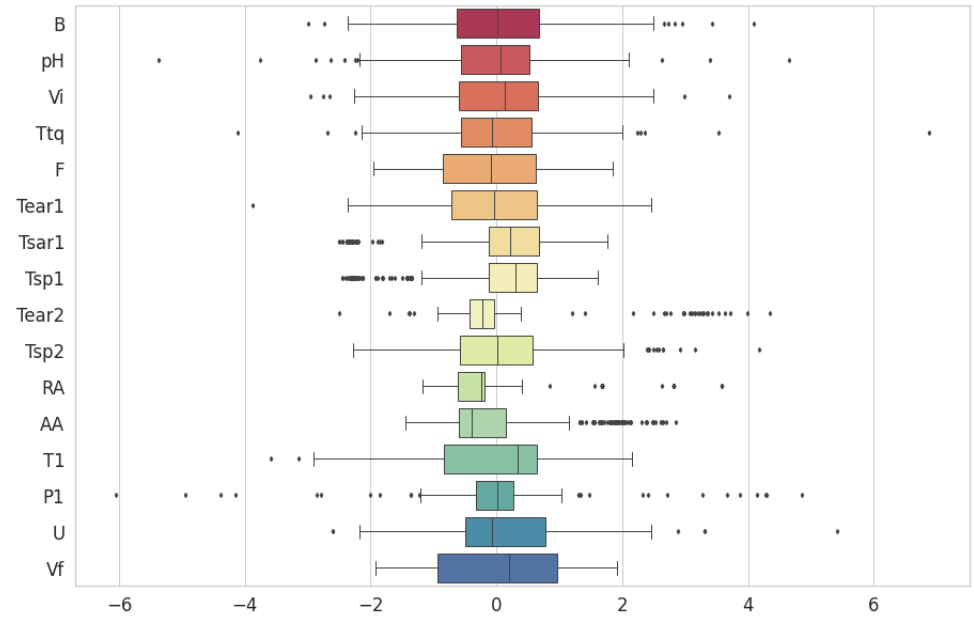
4 RESULTADOS E DISCUSSÃO

4.1 ANÁLISE ESTATÍSTICA

4.1.1 Análise Univariada

Na Figura 9 é apresentado um BoxPlot com os dados normalizados, utilizado para a visualização de *outliers* e análise de variabilidade.

Figura 9 – Boxplot do conjunto de dados normalizado.



A partir da análise do *boxplot* e dos valores de amplitude e intervalo interquartil exibidos na Tabela 3 pode-se observar que os parâmetros temperatura do tanque de alimentação (T_{tq}), temperatura de entrada de ar (T_{ear2}) e saída do produto no ciclone (T_{sp2}) apresentam maior presença de *outliers* e maior variação no processo. Em contrapartida, a vazão do produto, umidade e o teor de vitamina C final apresentaram menor dispersão, indicando serem as variáveis com maior controle.

Tabela 3 – Análise estatística para as diferentes variáveis.

Parâmetro	**Amplitude**	**Desvio Padrão**	**Distância interquartil**
Brix	9,890	1,397	1,810
pH inicial	2,860	0,285	0,310
Vitamina c inicial	2,992	0,450	0,567
Temperatura tanque de alimentação	36,497	3,324	3,718
Vazão	0,139	0,036	0,053
Temperatura ar de entrada	4,600	0,727	0,994
Temperatura ar de saída câmara	8,722	2,046	1,631
Temperatura saída produto câmara	8,282	2,037	1,581
Temperatura entrada ar ciclone	31,725	4,640	1,797
Temperatura produto saída ciclone	32,435	5,030	5,790
Rotação atomizador	12,441	2,618	1,127
Amperagem atomizador	2,109	0,491	0,371
Temperatura câmara secagem	14,330	2,505	3,668
Pressão câmara secagem	20,318	1,866	1,109
Umidade final	1,900	0,237	0,300
Vitamina c final	2,710	0,707	1,340

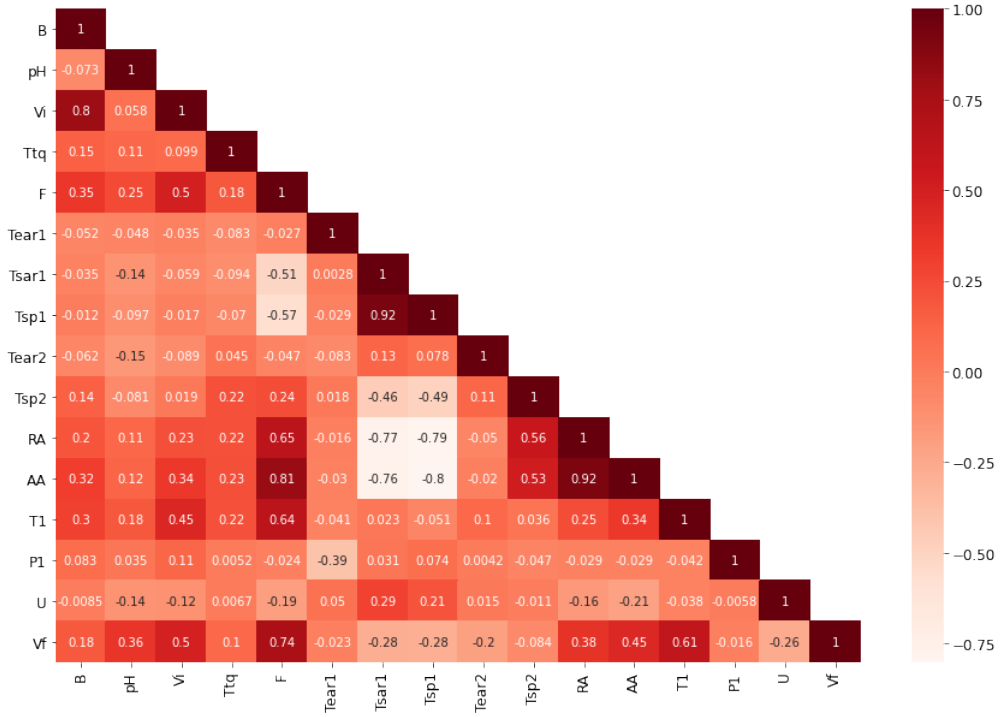
Com os valores de desvio padrão e amplitude pode-se observar que o processo industrial apresenta pouca variabilidade de dados. Todas as variáveis utilizadas possuem uma concentração de dados dentro da faixa estabelecida. Isso deve-se ao controle de processo utilizado na indústria em questão. Esse fator contribui diretamente para um bom desempenho dos modelos testados.

4.1.2 Análise Multivariada

A influência das variáveis, como mencionado no tópico 2.2.1.1, é um fator determinante ao selecionar os parâmetros de processo para obter um produto com maior qualidade. A Figura 10 apresenta uma matriz de correlação de Pearson das variáveis utilizadas no presente trabalho.

Analisando os coeficientes de Pearson observa-se que a vazão do produto (F) foi uma das variáveis com maior influência no processo, apresentando uma correlação

Figura 10 – Matriz de correlação entre as variáveis



forte com a amperagem do atomizador (AA), vitamina C final (V_f) e a temperatura interna da câmara (T_1) com valores de 0,81, 0,74 e 0,64, respectivamente. Além disso, apresentou-se uma correlação negativa com a temperatura de saída do produto da câmara (T_{sp1}) de -0,57, demonstrando que essas variáveis são inversamente proporcionais.

Outro fator determinante no processo é a rotação do atomizador (RA), no qual apresentou dependência direta com a vazão do produto (F) e a temperatura de saída do produto do ciclone (T_{sp2}), 0,65 e 0,56, respectivamente. Em paralelo, uma correlação negativa foi observada quando comparado com a temperatura de saída do ar e do produto da câmara (T_{sar1} e T_{sp1}) com valores próximos à -0,77 e -0,79, respectivamente. Indicando que quanto menor a rotação do atomizador, maior será as temperatura de saídas da câmara.

Os demais parâmetros apresentaram correlação baixa entre as mesmas. Esse fato é observado devido as condições ideais de operação. A pouca dispersão dos dados dificulta a análise de correlação entre as variáveis. Desta forma, optou-se por desenvolver um modelo utilizando todas as variáveis existentes no conjunto de dados.

4.2 APLICAÇÃO DOS MODELOS

A partir do método implementado, os hiperparâmetros foram definidos de acordo com o melhor desempenho do treinamento, avaliando o coeficiente de determinação

Tabela 4 – Hiperparâmetros obtidos a partir do método *GridSearchcv*.

Parâmetros	**Modelos**			
	RNA	**SVM**	**KNN**	**RF**
Camadas Ocultas e neurônios	(10,10,10)	-	-	-
Função de Ativação	Tanh	-	-	-
Solver	adam	-	-	-
Alfa	1	-	-	-
Taxas de aprendizagem	adaptativa	-	-	-
Epsilon	e^{-8}	0,2	-	-
Kernel	-	linear	-	-
Gamma	-	auto	-	-
Parâmetro de regularização (C)	-	61	-	-
Número de vizinhos	-	-	8	-
Função de peso	-	-	*distance*	-
Algoritmo	-	-	*kd_tree*	-
Tamanho da folha	-	-	40	-
Parâmetro de potência	-	-	1	-
Critério	-	-	-	MSE
Número de árvores na floresta	-	-	-	20
Profundidade máxima da árvore	-	-	-	10
Número de variáveis para divisão	-	-	-	7

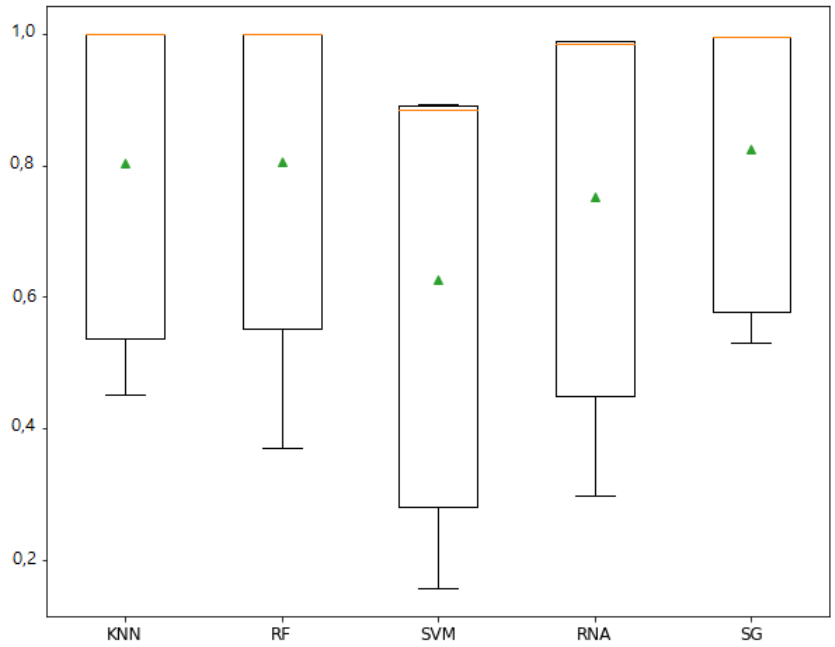
médio.

A Tabela 4 mostra que a melhor arquitetura da rede neural foi aquela com três camadas intermediárias e 10 neurônios em cada. Para isso, utilizou-se a tangente hiperbólica como função de ativação e uma taxa de aprendizado adaptativa e obteve-se um coeficiente de determinação médio equivalente à 0,877.

Com um R^2 de 0,905, o *Random Forest* utilizou-se uma estrutura com 20 árvores de decisão e a qualidade para a divisão foi utilizando o erro quadrático médio. O KNN utilizou o inverso da distância entre os 8 vizinhos mais próximos aplicando o algoritmo *kd_tree*, onde busca-se reduzir o número necessário de cálculos de distância e obteve-se o R^2 de 0,903. A fim de buscar uma função capaz de classificar os dados de treinamento com maior assertividade, o SVM apresentou um melhor resultado quando utilizado a função linear com perda equivalente a 0,2 com R^2 equivalente a 0,814.

Com a utilização dos hiperparâmetros selecionados, os modelos foram submetidos ao teste com o conjunto de dados onde foram divididos em 10 *folds*. A Figura 11 apresenta o diagrama de caixas com a distribuição do coeficiente de determinação para cada modelo.

Figura 11 – Resultado da variação do R^2 com k-fold=10.



A mediana, indicada pela linha laranja na Figura 11 nos leva a concluir que ao utilizar os modelos selecionados para predizer os dados de teste, a tendência central dos modelos variaram em torno de 0,9 e 1,0. Observa-se que os modelos tiveram uma similaridade na distribuição dos resultados, com exceção do modelo de SVM que apresentou resultadores inferiores quando comparado aos demais. Esse comportamento pode ser explicado pela escolha dos hiperparâmetros. Segundo Cawley e Talbot (2010) a otimização dos parâmetros deve ser realizada buscando minimizar a complexidade computacional. Para o SVM, a função que apresentou melhor resultado neste trabalho, foi a função linear. A escolha da função kernel é de vital importância para o desempenho do modelo, o uso de diferentes funções possibilita a construção de máquinas de aprendizagem com diferentes tipos de superfícies de decisão.

Estudos recentes apontam que as técnicas de otimização não-linear apresentam resultados superiores à abordagem linear. Entretanto, a utilização destas funções podem acarretar em um modelo computacionalmente caro e robusto, requerendo alta performance de máquina e elevado tempo de processamento (SANTOS *et al.*, 2002) (MALDONADO *et al.*, 2011)(TEHRANY *et al.*, 2015).

Tabela 5 – Avaliação dos modelos desenvolvidos

Modelo	**MSE**	**MAE**	R^2
RF	0,110	0,157	0,902
RNA	0,139	0,196	0,872
KNN	0,112	0,155	0,900
SVM	0,186	0,275	0,809
SG	0,099	0,163	0,911

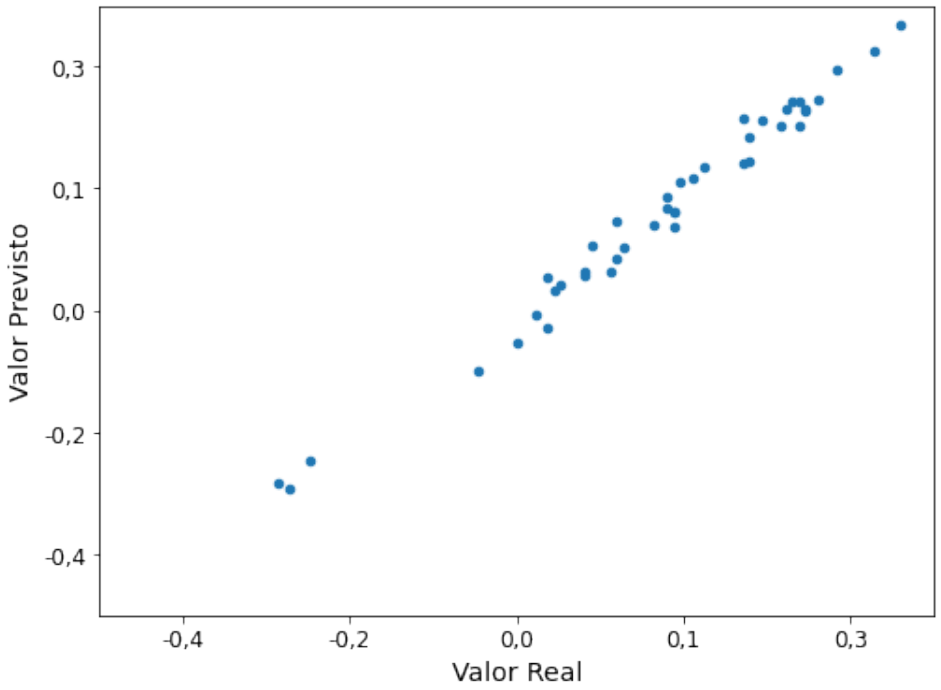
Analisando os modelos individuais, o SVM apresentou maior erro quadrático médio e erro absoluto, respectivamente 0,186 e 0,275. Enquanto que o KNN e o RF apresentaram um erro menor e maior coeficiente de determinação.

A técnica de *Stacked Generalization* apresentou uma redução no erro quadrático médio e um aumento do R^2, corroborando com os estudos de Boehmke e Greenwell (2019), Ma *et al.* (2018) e Wu *et al.* (2019) onde os modelos desenvolvidos indicaram um resultado final mais preciso quando comparado com os modelos individuais. Devido a sua capacidade de reduzir os vieses de cada modelo individual e melhorar a performance dos mesmos, tomou-se o modelo SG como o modelo final deste trabalho.

4.3 VALIDAÇÃO DOS MODELOS

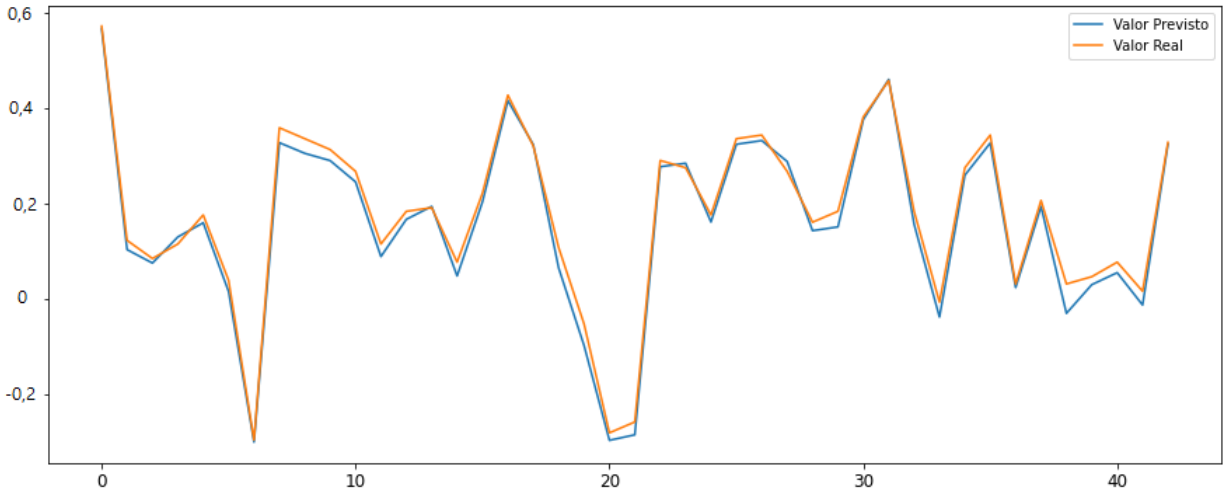
Ao aplicar o modelo final em um novo conjunto de dados, os resultados obtidos demonstram um correlação positiva entre o valor predito e o valor real, conforme Figura 12.

Figura 12 – Dispersão dos dados reais *versus* preditos do teor de vitamina C.



A Figura 13 apresenta a distribuição dos pontos experimentais e dos pontos preditos pelo modelo. Nota-se que os valores preditos apresentaram valores próximos ao valor real.

Figura 13 – Comparativo entre os valores preditos e valores experimentais



Ao analisar o desempenho do preditor com novos dados, obteve-se o coeficiente de determinação próximo a 0,98. O erro quadrático médio e o erro absoluto apresentaram valores de 0,000542 e 0,0189, respectivamente. Esses resultados indicam que o modelo gerado apresentou valores muito próximos ao valor real, seguindo a tendência dos dados de produção.

De forma geral, observa-se que o modelo final apresentou boa capacidade de generalização, sem a ocorrência de *overfitting* e *underfitting*, demonstrando a viabilidade de sua aplicação no ambiente de produção ao predizer o teor de vitamina C final do pó de acerola.

5 CONSIDERAÇÕES FINAIS

A acerola, quando submetida ao processo de secagem, pode sofrer constantes modificações que ocasionam em mudanças na qualidade do produto final. Essas alterações influenciam diretamente nos compostos bioativos, como a vitamina C. Sendo assim, a melhoria das condições de operação têm sido tema de estudo cada vez mais frequente a fim de reduzir a perda de vitamina durante o processo.

Alinhado a isto, o presente trabalho buscou avaliar os modelos de inteligência artificial na predição do teor de vitamina C da acerola ao final do processo de secagem por atomização. Como resultado, os modelos propostos apresentaram boa capacidade de predição, com o coeficiente de determinação variando entre 0,815 à 0,912. O desempenho de cada modelo depende diretamente de sua forma de aplicação e dos hiperparâmetros selecionados. Ao final, foi implementado a técnica de SG para combinar o comportamento de cada modelo e obter um resultado mais assertivo. Com um erro quadrático médio de 0,000542 g/100g entre o valor predito e o valor real, a aplicação de modelos de inteligência artificial mostrou-se viável para a utilização em processos industriais.

Com o desenvolvimento e implementação deste trabalho pode-se concluir que o desempenho dos modelos foram favoráveis na predição do teor de vitamina C. Os resultados obtidos indicam que os valores preditos ficaram muito próximos dos dados experimentais, mostrando ser possível a utilização de sistemas especialistas no ambiente de produção, a fim de torná-lo mais automatizado e eficaz. O coeficiente de determinação do modelo final foi de 0,98, havendo a possibilidade desta taxa ser melhorada à medida que novas amostras são incorporadas ao conjunto de dados.

Contudo, ainda há espaço para melhorias significativas na metodologia e técnica de predição. Fatores como a seleção de variáveis e otimização de hiperparâmetros podem ser explorado a fim de obter um modelo com melhor resultado. O modelo SVM apresentou MSE equivalente à 18,7%, esse resultado poderia ser melhorado com a utilização de uma função polinomial para definição do hiperplano. Já a rede neural artificial pode apresentar melhores resultados com a variação do número de neurônios e camadas ocultas.

De forma geral, os resultados indicam que os objetivos esperados foram alcançados, provando que a inteligência artificial é um ramo com potencial para ser estudado e implementado nas indústrias de alimentos. A utilização de dados reais de produção tornou os resultados mais próximos à realidade, pois além de utilizar dados de uma empresa que possui controle de qualidade criterioso, a mesma consta de automatizações de processo que permitiram a aquisição de dados confiáveis, fato que influenciou diretamente no bom desempenho do projeto.

REFERÊNCIAS

AGUIRRE, JM; GASPARINO FILHO, J. Desidratação de Frutas e Hortaliças: Manual Técnico do Instituto de Tecnologia de Alimentos. **Campinas: ITAL**, 1999.

AHMAD, Muhammad Waseem; REYNOLDS, Jonathan; REZGUI, Yacine. Predictive modelling for solar thermal energy systems: A comparison of support vector regression, random forest, extra trees and regression trees. **Journal of cleaner production**, Elsevier, v. 203, p. 810–821, 2018.

ANVISA. Resolução, RDC nº 269, de 22 de setembro de 2005. **Aprova o Regulamento Técnico sobre a Ingestão Diária Recomendada de Proteına, Vitaminas e Minerais. ANVISA**, 2005.

AWAD, Mariette; KHANNA, Rahul. Support vector regression. *In*: EFFICIENT learning machines. [*S.l.*]: Springer, 2015. P. 67–80.

BATISTA, Patrício Ferreira; LIMA, Maria Auxiliadora Coêlho de; ALVES, Ricardo Elesbão; FAÇANHA, Rafaela Vieira. Bioactive compounds and antioxidant activity in tropical fruits grown in the lower-middle São Francisco Valley. **Revista Ciência Agronômica**, SciELO Brasil, v. 49, n. 4, p. 616–623, 2018.

BATISTA *et al.* **Pré-processamento de dados em aprendizado de máquina supervisionado**. 2003. Tese (Doutorado) – Universidade de São Paulo.

BLOOM, Maria Inez Tenório; SANTOS, TMP; ATAIDE-SILVA, Thays; VASCONCELOS, Sandra Mary Lima. Ingestão de vitaminas e minerais em uma amostra de hipertensos de um municıpio da região nordeste do Brasil. **Rev Bras Nutr Clin**, v. 30, n. 2, p. 154–8, 2015.

BOEHMKE, Brad; GREENWELL, Brandon M. **Hands-on machine learning with R**. [*S.l.*]: CRC Press, 2019.

BOERI, Camila N; SILVA, Fernando J Neto da; FERREIRA, Jorge AF. Otimização dos parâmetros de secagem para minimização do custo energético num secador convectivo de alimentos. **Proceeding Series of the Brazilian Society of Computational and Applied Mathematics**, v. 1, n. 1, 2013.

BRAGA, A de P. **Redes neurais artificiais: teoria e aplicações**. [*S.l.*]: Livros Técnicos e Cientıficos, 2000.

BREIMAN, Leo. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.

BREIMAN, Leo. Stacked regressions. **Machine learning**, Springer, v. 24, n. 1, p. 49–64, 1996.

CÁNOVAS-GARCÍA, Fulgencio; ALONSO-SARRÍA, Francisco; GOMARIZ-CASTILLO, Francisco; OÑATE-VALDIVIESO, Fernando. Modification of the random forest algorithm to avoid statistical dependence problems when classifying remote sensing imagery. **Computers & Geosciences**, Elsevier, v. 103, p. 1–11, 2017.

CAWLEY, Gavin C; TALBOT, Nicola LC. On over-fitting in model selection and subsequent selection bias in performance evaluation. **The Journal of Machine Learning Research**, JMLR. org, v. 11, p. 2079–2107, 2010.

CELESTINO, Sonia Maria Costa. Princıpios de secagem de alimentos. **Embrapa Cerrados-Documentos (INFOTECA-E)**, Planaltina, DF: Embrapa Cerrados, 2010., 2010.

CHEN, Y.R.; PARK, B.; HUFFMAN, R.W.; NGUYEN, M. Classification of on-line poultry carcasses with backpropagation neural networks. **Journal of Food Process Engineering**, v. 21, p. 33–48, jan. 2007. DOI: 10.1111/j.1745-4530.1998.tb00437.x.

CHOKPHOEMPHUN, Susama; CHOKPHOEMPHUN, Suriya. Moisture content prediction of paddy drying in a fluidized-bed drier with a vortex flow generator using an artificial neural network. **Applied Thermal Engineering**, Elsevier, v. 145, p. 630–636, 2018.

COSTA, Ana Carolina Sousa; LIMA, Maria Auxiliadora Coêlho de; ALVES, Ricardo Elesbão; ARAÚJO, AL de S; BATISTA, PF; ROSATTI, SR; RISTOW, NC. Caracterização fisico-quımica de acerola e dos resıduos do processamento em dois estádios de maturação. *In*: IN: SIMPÓSIO BRASILEIRO DE PÓS-COLHEITA DE FRUTAS, HORTALIÇAS E FLORES, 3 . . . EMBRAPA Semiárido-Artigo em anais de congresso (ALICE). [*S.l.*: *s.n.*], 2011.

DAS, Mehmet; AKPINAR, Ebru Kavak. Investigation of pear drying performance by different methods and regression of convective heat transfer coefficient with support vector machine. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 8, n. 2, p. 215, 2018.

ENGEL, Bruno. Emprego de Spray Dryer na indústria de alimentos: Uma breve revisão. **Revista Jovens Pesquisadores**, v. 7, n. 2, p. 02–11, 2017.

FABRÍCIO, Dayane Silva. **Determinação de vitamina C em suco de frutas in natura e industrializados por cromatografia líquida e titulação iodométrica**. 2018. B.S. thesis – Universidade Tecnológica Federal do Paraná.

FAZAELI, Mahboubeh; EMAM-DJOMEH, Zahra; ASHTARI, Ahmad Kalbasi; OMID, Mahmoud. Effect of spray drying conditions and feed composition on the physical properties of black mulberry juice powder. **Food and bioproducts processing**, Elsevier, v. 90, n. 4, p. 667–675, 2012.

FEIGENBAUM, Edward; BUCHANAN, Bruce; LEDERBERG, Joshua. On generality and problem solving: A case study using the DENDRAL program. **Machine Intelligence**, v. 6, set. 1970.

FREIRE, Luziany Adyja da Costa. **Montagem e operação de um secador pneumático tipo flash**. 2011. Diss. (Mestrado) – Universidade Federal do Rio Grande do Norte.

GALARÇA, Simone Padilha; LIMA, Cláudia Simone Madruga; SILVEIRA, Gustavo da; RUFATO, Andreia De Rossi. Correlação de Pearson e análise de trilha identificando variáveis para caracterizar porta-enxerto de Pyrus communis L. **Ciência e Agrotecnologia**, SciELO Brasil, v. 34, n. 4, p. 860–869, 2010.

GAMA, J.; FACELI, K.; LORENA, A.C.; DE CARVALHO, A.C.P.L.F. **Inteligência artificial: uma abordagem de aprendizado de máquina**. [*S.l.*]: Grupo Gen - LTC, 2011. ISBN 9788521618805. Disponível em: https://books.google.com.br/books?id=4DwelAEACAAJ.

GONZAGA NETO, L; SOARES, JM; CHOUDHURY, MM; LEAL, IM. A cultura da acerola. **Brasılia: Embrapa-SPI**, 2012.

GOOGLE. **Google Colaboratory**. Mar. 2020. Disponível em: https://colab.research.google.com/.

GRZYBOWSKI, Andrzej; PIETRZAK, Krzysztof. Albert Szent-Györgyi (1893-1986): the scientist who discovered vitamin C. **Clinics in dermatology**, Elsevier, v. 31, n. 3, p. 327–331, 2013.

HARBOR, Informática Industrial. **Box Plot: você conhece essa ferramenta?** Nov. 2017. Disponível em: https://www.harbor.com.br/harbor-blog/2017/07/17/box-plot-voce%20-conhece-esta-ferramenta/.

HEATON, Jeff. **Introduction to neural networks with Java**. [*S.l.*]: Heaton Research, Inc., 2008.

HUBERT, Mia; VANDERVIEREN, Ellen. An adjusted boxplot for skewed distributions. **Computational statistics & data analysis**, Elsevier, v. 52, n. 12, p. 5186–5201, 2008.

IBGE. **Cesno Agro 2017 - Acerola Brasil.** [*S.l.*: *s.n.*], 2017. https://censos.ibge.gov.br/agro/2017/templates/censo_agro/resultadosagro/agricultura.html?localidade=0&tema=76215.

JAYASUNDERA, Mithila; ADHIKARI, Benu; HOWES, Tony; ALDRED, Peter. Surface protein coverage and its implications on spray-drying of model sugar-rich foods: solubility, powder production and characterisation. **Food Chemistry**, Elsevier, v. 128, n. 4, p. 1003–1016, 2011.

JUNQUEIRA, KP; PIO, R. VALE, MR do; RAMOS, JD Cultura da acerola. **Lavras: UFLA**, 2004.

KESHANI, Samaneh; DAUD, Wan Ramli Wan; NOUROUZI, MM; NAMVAR, Farideh; GHASEMI, Mostafa. Spray drying: An overview on wall deposition, process and modeling. **Journal of Food Engineering**, Elsevier, v. 146, p. 152–162, 2015.

KHALED, Alfadhl Yahya; KABUTEY, Abraham; SELVI, Kemal Çağatay; MIZERA, Čestmır; HRABE, Petr; HERÁK, David. Application of Computational Intelligence in Describing the Drying Kinetics of Persimmon Fruit (Diospyros kaki) During Vacuum and Hot Air Drying Process. **Processes**, Multidisciplinary Digital Publishing Institute, v. 8, n. 5, p. 544, 2020.

LABMAQ. **LABMAQ**. 2020. Disponível em: `http://www.labmaqdobrasil.com.br`. Acesso em: 1 ago. 2020.

LEAL, Danilo; FRANÇA, Aldi; OLIVEIRA, Leonardo; CORRÊA, Daniel; ARNHOLD, Emmanuel; FERREIRA, Reginaldo; BASTO, Débora; BRUNES, Ludmilla. Fracionamento de carboidratos e proteínas da Brachiaria híbrida 'Mulato II' sob adubação nitrogenada e regime de cortes. **Archivos de Zootecnia**, v. 66, p. 181, jan. 2017.

LIBRALON, Giampaolo Luiz. **Investigação de combinações de técnicas de detecção de ruıdo para dados de expressão gênica**. 2007. Tese (Doutorado) – Universidade de São Paulo.

LORENA, Ana Carolina; CARVALHO, ACPLF. Introdução às máquinas de vetores suporte (Support Vector Machines). **Laboratório de Inteligência Computacional, ICMC/USP, São Carlos**, n. 192, 2003.

LU, Yang. Industry 4.0: A survey on technologies, applications and open research issues. **Journal of industrial information integration**, Elsevier, v. 6, p. 1–10, 2017.

MA, Zhiyuan; WANG, Ping; GAO, Zehui; WANG, Ruobing; KHALIGHI, Koroush. Ensemble of machine learning algorithms using the stacked generalization approach to estimate the warfarin dose. **PloS one**, Public Library of Science San Francisco, CA USA, v. 13, n. 10, e0205872, 2018.

MALDONADO, Sebastián; WEBER, Richard; BASAK, Jayanta. Simultaneous feature selection and classification using kernel-penalized support vector machines. **Information Sciences**, Elsevier, v. 181, n. 1, p. 115–128, 2011.

MANELA-AZULAY, Mônica; MANDARIM-DE-LACERDA, Carlos Alberto; PEREZ, Maurıcio de Andrade; FILGUEIRA, Absalom Lima; CUZZI, Tullia. Vitamina C. **Anais brasileiros de dermatologia**, SciELO Brasil, v. 78, n. 3, p. 265–272, 2003.

MARETE, Eunice N; JACQUIER, Jean Christophe; O'RIORDAN, Dolores. Effects of extraction temperature on the phenolic and parthenolide contents, and colour of aqueous feverfew (Tanacetum parthenium) extracts. **Food chemistry**, Elsevier, v. 117, n. 2, p. 226–231, 2009.

MCCABE, WL; SMITH, JC; HARRIOTT, P. **Unit operations of Chemical Engineering, 5th International ed**. [*S.l.*]: McGraw-Hill Chemical Engineering Series, Singapore: McGraw Hill, 1995.

MENDES, Raquel Dias. INTELIGÊNCIA ARTIFICIAL: SISTEMAS ESPECIALISTAS NO GERENCIAMENTO DA INFORMAÇÃO. pt. **Ciência da Informação**, scielo, v. 26, jan. 1997. ISSN 0100-1965. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-19651997000100006&nrm=iso.

MEZHERICHER, M; LEVY, A; BORDE, I. Spray drying modelling based on advanced droplet drying kinetics. **Chemical Engineering and Processing: Process Intensification**, Elsevier, v. 49, n. 11, p. 1205–1213, 2010.

MITCHELL, T.M. **Machine Learning**. [*S.l.*]: McGraw-Hill, 1997. (McGraw-Hill International Editions). ISBN 9780071154673. Disponível em: https://books.google.com.br/books?id=EoYBngEACAAJ.

MONARD, Maria Carolina; BARANAUSKAS, José Augusto. Conceitos Sobre Aprendizado de Máquina. *In*: SISTEMAS Inteligentes Fundamentos e Aplicações. 1. ed. Barueri-SP: Manole Ltda, 2003. P. 89–114.

MORAES, Julieta; FREITAS, Josafá; BOZZO, Fabiana; MORAES, Flávio; MARTINS, Maurıcio. A suplementação alimentar com vitamina C acelera a evolução do processo cicatricial em Piaractus mesopotamicus (Holmberg, 1887). **Boletim do Instituto de Pesca**, v. 29, n. 1, p. 57–67, 2018.

MOREIRA, GERMANO ÉDER GADELHA;
DE AZEREDO, HENRIETTE MONTEIRO CORDEIRO;
DE MEDEIROS, MARIA DE FÁTIMA DANTAS; DE BRITO, Edy Sousa;
DE SOUZA, ARTHUR CLÁUDIO RODRIGUES. Ascorbic acid and anthocyanin retention during spray drying of acerola pomace extract. **Journal of food processing and preservation**, Wiley Online Library, v. 34, n. 5, p. 915–925, 2010.

NASSER, Maurıcio Dominguez; ZONTA, Augusto. Caracterização de frutos de genótipos de aceroleira em função de estádios de maturação. **Tecnologia & Ciência Agropecuária, João Pessoa**, v. 8, n. 5, p. 76–78, 2014.

NAVIDI, William. **Probabilidade e estatıstica para ciências exatas**. [*S.l.*]: AMGH Editora, 2012.

NÓBREGA, Erly Maria Medeiros de Araújo. **Secagem do resıduo de acerola (Malphigia emarginata DC): estudo do processo e avaliação do impacto sobre o produto final**. 2012. Diss. (Mestrado) – Universidade Federal do Rio Grande do Norte.

NOGUEIRA, Rejane Jurema Mansur Custódio;
MORAES, José Antônio Proença Vieira de; BURITY, Hélio Almeida;
SILVA JUNIOR, Josué Francisco da. Efeito do estádio de maturação dos frutos nas caracterısticas fısico-quımicas de acerola. **Pesquisa Agropecuária Brasileira**, SciELO Brasil, v. 37, n. 4, p. 463–470, 2002.

OLIVEIRA, Luciana De Siqueira; MOURA, Carlos Farley Herbster;
DE BRITO, Edy Sousa; MAMEDE, Rosa Virgınia Soares;
DE MIRANDA, Maria Raquel Alcântara. Antioxidant metabolism during fruit development of different acerola (Malpighia emarginata DC) clones. **Journal of Agricultural and Food Chemistry**, ACS Publications, v. 60, n. 32, p. 7957–7964, 2012.

OLIVEIRA, Nelma de Mello Silva; NASCIMENTO, Luiz Carlos do;
FIORINI, João Evangelista. Isolamento e identificação de bactérias facultativas mesofılicas em carnes frescas bovinas e suınas. **Hig. aliment**, p. 68–74, 2002.

OLIVEIRA, Olivia Werner; PETROVICK, Pedro Ros. Secagem por aspersão (spray drying) de extratos vegetais: bases e aplicações. **Revista brasileira de farmacognosia. São Paulo, SP. Vol. 20, n. 4 (Ago./Set. 2010), p. 641-650**, 2010.

PEDREGOSA, F. *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PICCOLO, Domenico; FERRARI, Angela; PERIS, Ketty; DIADONE, R;
RUGGERI, Benedetta; CHIMENTI, Sergio. Dermoscopic diagnosis by a trained clinician vs. a clinician with minimal dermoscopy training vs. computer-aided diagnosis of 341 pigmented skin lesions: A comparative study. **The British journal of dermatology**, v. 147, p. 481–6, out. 2002. DOI: 10.1046/j.1365-2133.2002.04978.x.

QUINLAN, J Ross. **C4. 5: programs for machine learning**. [*S.l.*]: Elsevier, 2014.

RIBEIRO, Eliana Paula; SERAVALLI, Elisena AG. **Quımica de alimentos**. [*S.l.*]: Editora Blucher, 2007.

RITZINGER, Rogério; RITZINGER, CHSP. **Acerola-aspectos gerais da cultura**. [*S.l.*]: Embrapa Mandioca e Fruticultura Tropical Cruz das Almas, 2004.

RUSSELL, S.; NORVIG, P. **Inteligência artificial**. [*S.l.*]: CAMPUS - RJ, 2004. ISBN 9788535211771. Disponível em: https://books.google.com.br/books?id=wBMvAAAACAAJ.

SANTOS, Eulanda Miranda dos *et al.* Teoria e aplicação de Support Vector Machines à aprendizagem e reconhecimento de objetos baseado na aparência. Universidade Federal de Campina Grande, 2002.

SANTOS, Newton Carlos; BARROS, Sâmela Leal; ALMEIDA, Raphael Lucas Jacinto; MONTEIRO, Shênia Santos; NASCIMENTO, Amanda Priscila Silva; SILVA, Virgínia Mirtes de Alcântara; GOMES, Josivanda Palmeira; LUIZ, Márcia Ramos; VIEIRA, Danise Medeiros. Evaluation Degradation of Bioactive Compounds of Fruit Physalis (P. peruviana) During the Drying Process. **Research, Society and Development**, v. 9, e102911678, jan. 2020. DOI: 10.33448/rsd-v9i1.1678. Disponível em: https://rsdjournal.org/index.php/rsd/article/view/1678.

SCHWAB, Klaus. **A quarta revolução industrial**. [*S.l.*]: Edipro, 2019.

SILVA, Raissa Henrique. **Secagem do extrato da casca de berinjela (Solanum melongena L.) por atomização em spray dryer com adição de adjuvantes**. 2017. B.S. thesis – Universidade Federal do Rio Grande do Norte.

SOARES, Fátima Cibele; ROBAINA, Adroaldo Dias; PEITER, Marcia Xavier; RUSSI, Jumar Luis. Predição da produtividade da cultura do milho utilizando rede neural artificial. **Ciência Rural**, SciELO Brasil, v. 45, n. 11, p. 1987–1993, 2015.

SOLDI, Luiz Ricardo *et al.* Avaliação do efeito do uso de menadiona e ácido ascórbico associadas ao ferro em células tumorais de linhagem 4T1 e leucócitos in vitro. Universidade Federal do Triângulo Mineiro, 2019.

STUART, Giane; MACHADO, Ricardo; OLIVEIRA, José V de; ULLER, Angela C; LIMA, Enrique L. HYBRID ARTIFICIAL NEURAL NETWORK APPLIEDTO MODELING

SCFE OF BASIL AND ROSEMARY OILS. **Food Science and Technology**, SciELO Brasil, v. 17, n. 4, p. 501–505, 1997.

STUCKEY, Thomas *et al.* Cardiac Phase Space Tomography: A novel method of assessing coronary artery disease utilizing machine learning. **PLOS ONE**, v. 13, e0198603, ago. 2018. DOI: 10.1371/journal.pone.0198603.

SYARIF, Iwan; PRUGEL-BENNETT, Adam; WILLS, Gary. SVM parameter optimization using grid search and genetic algorithm to improve classification performance. **Telkomnika**, Ahmad Dahlan University, v. 14, n. 4, p. 1502, 2016.

TACCHELLA, Andrea; ROMANO, Silvia; FERRALDESCHI, Michela; SALVETTI, Marco; ZACCARIA, Andrea; CRISANTI, Andrea; GRASSI, Francesca. Collaboration between a human group and artificial intelligence can improve prediction of multiple sclerosis course: a proof-of-principle study. **F1000Research**, v. 6, p. 2172, dez. 2017. DOI: 10.12688/f1000research.13114.1.

TANAKA, Deise Luciane. Influência da desidratação por spray dryng sobre o teor ácido ascórbico no suco de acerola (Malpighia ssp). Universidade Estadual Paulista (UNESP), 2007.

TAUR, Ke-Haur; DENG, Xiang-Yun; CHOU, Mi-Huo; CHEN, Jing-Wei; LEE, Yi-Hsiu; WANG, Wen-June. A study on Machine Learning Approaches for Predicting and Analyzing the Drying Process in the Textile Industry. *In*: IEEE. 2019 International Automatic Control Conference (CACS). [*S.l.*: *s.n.*], 2019. P. 1–5.

TEHRANY, Mahyat Shafapour; PRADHAN, Biswajeet; MANSOR, Shattri; AHMAD, Noordin. Flood susceptibility assessment using GIS-based support vector machine model with different kernel types. **Catena**, Elsevier, v. 125, p. 91–101, 2015.

TEIXEIRA, João. **O que é inteligência artificial**. [*S.l.*]: E-Galáxia, 2019.

TEIXEIRA, Mirella; MONTEIRO, Magali. Degradação da vitamina C em suco de fruta. **Alimentos e Nutrição**, v. 17, n. 2, p. 219–227, 2006.

TORALLES, Ricardo Peraça; VENDRUSCOLO, João Luiz; VENDRUSCOLO, Claire Tondo; DEL PINO, Francisco Augusto Burkert; ANTUNES, Pedro Luiz. Determinação das constantes cinéticas de degradação do

ácido ascórbico em purê de pêssego: efeito da temperatura e concentração. **Food Science and Technology**, SciELO Brasil, v. 28, n. 1, p. 18–23, 2008.

TSEN, Andy Yen-Di; JANG, Shi Shang; WONG, David Shan Hill; JOSEPH, Babu. Predictive control of quality in batch polymerization using hybrid ANN models. **AIChE Journal**, Wiley Online Library, v. 42, n. 2, p. 455–465, 1996.

TUKEY, John W. **Exploratory data analysis**. [*S.l.*]: Reading, MA, 1977. v. 2.

TURING, A. M. I. - COMPUTING MACHINERY AND INTELLIGENCE. **Mind**, v. LIX, n. 236, p. 433–460, out. 1950. ISSN 0026-4423. DOI: 10.1093/mind/LIX.236.433. eprint: https://academic.oup.com/mind/article-pdf/LIX/236/433/30123314/lix-236-433.pdf. Disponível em: https://doi.org/10.1093/mind/LIX.236.433.

VENDRAMINI, Ana L; TRUGO, Luiz C. Chemical composition of acerola fruit (Malpighia punicifolia L.) at three stages of maturity. **Food Chemistry**, Elsevier, v. 71, n. 2, p. 195–198, 2000.

VON ZUBEN, Fernando J; ATTUX, Romis RF. Máquinas de vetores-suporte. **Departamento de Computação Aplicada, Faculdade de Engenharia Elétrica e de Computação, Unicamp**, 2013.

WOLPERT, David H. Stacked generalization. **Neural networks**, Elsevier, v. 5, n. 2, p. 241–259, 1992.

WU, Dongrui; LIN, Chin-Teng; HUANG, Jian; ZENG, Zhigang. On the functional equivalence of TSK fuzzy systems to neural networks, mixture of experts, CART, and stacking ensemble regression. **IEEE Transactions on Fuzzy Systems**, IEEE, 2019.

YEH, Chi-Yuan; HUANG, Chi-Wei; LEE, Shie-Jue. A multiple-kernel support vector regression approach for stock market price forecasting. **Expert Systems with Applications**, Elsevier, v. 38, n. 3, p. 2177–2186, 2011.

YOUNIS, Kaiser; AHMAD, Saghir; OSAMA, Khwaja; MALIK, Mudasir A. Optimization of de-bittering process of mosambi (Citrus limetta) peel: Artificial neural network, Gaussian process regression and support vector machine modeling approach. **Journal of Food Process Engineering**, Wiley Online Library, v. 42, n. 6, e13185, 2019.

ZELLWEGER, Michael; TSIRKIN, Andrew; VASILCHENKO, Vasily; FAILER, Michael; DRESSEL, Alexander; KLEBER, Marcus; RUFF, Peter; MÄRZ, Winfried. A new non-invasive diagnostic tool in coronary artery disease: artificial intelligence as an essential element of predictive, preventive, and personalized medicine. **EPMA Journal**, v. 9, ago. 2018. DOI: 10.1007/s13167-018-0142-x.

ZENG, Xinchuan; MARTINEZ, Tony R. Distribution-balanced stratified cross-validation for accuracy estimation. **Journal of Experimental & Theoretical Artificial Intelligence**, Taylor & Francis, v. 12, n. 1, p. 1–12, 2000.

ANEXO A – CÓDIGO FONTE

Código A.0.1 – Detecção de *outliers*

```
def outlier (col):
  Q1 =  dados[col].quantile(0.25)
  Q3 = dados[col].quantile(0.75)
  IQR = Q3 - Q1
  dados[col].where(dados[col] > Q1-1.5*IQR , float('NaN'), inplace=True)
  dados[col].where(dados[col] < Q3+1.5*IQR , float('NaN'), inplace=True)
```

Código A.0.2 – Trabalhando com valores nulos

```
from sklearn.impute import KNNImputer
imputer = KNNImputer(n_neighbors=10)
dados=pd.DataFrame(imputer.fit_transform(dados),columns=dados.columns)
```

Código A.0.3 – Padronização de dados

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scalery = StandardScaler()
X = pd.DataFrame(scaler.fit_transform(X),columns=X.columns)
y = pd.DataFrame(scalery.fit_transform(y.values.reshape(-1, 1)),columns=['vitamina c final'])
```

Código A.0.4 – Extrapolação dos dados

```
import random
import math
xe=X.values
nvar=15
nexp=314

xam=[0]*(nvar+1)
xa=[0]*(nvar+1)
dados_gerados=[]
soma.append(0.0)
b = 0.0
a=[]
w=[]
soma=[]
faux=[0]*(nvar+1)
max_value=dados.values.max(axis=0)
min_value=dados.values.min(axis=0)
xam=(max_value+min_value)/2

for jj in range(1,501,1):
  aax=random.uniform(0,1)
  for col in range(0,nvar+1,1):
    xa[col]=xam[col]+aax*(xam[col]-min_value[col])
  dif=[[0]*(nvar+1) for i in range(nexp)]
  for linha in range(0,nexp,1):
    for col in range(0,nvar,1):
      dif[linha][col] = (xa[col]-xe[linha][col])*(xa[col]-xe[linha][col])
  for linha in range(0,nexp,1):
    for col in range(0,nvar+1,1):
      soma[linha]=soma[linha]+dif[linha][col]
    a.append(1.0 / math.sqrt(soma[linha]))
    b = b+a[linha]
  for linha in range(0,nexp,1):
    w.append(a[linha]/b)
  for linha in range(0,nexp,1):
    for col in range(0,nvar+1,1):
      faux[col]=(xa[col] + w[linha] * y[linha])

  dados_gerados.append(faux)
```

Código A.0.5 – Otimização de parâmetros para cada modelo

```
from sklearn.model_selection import GridSearchCV

class EstimatorSelectionHelper:

    def __init__(self, models, params):
        if not set(models.keys()).issubset(set(params.keys())):
            missing_params = list(set(models.keys()) - set(params.keys()))
            raise ValueError("Some estimators are missing parameters: %s" % missing_params)
        self.models = models
        self.params = params
        self.keys = models.keys()
        self.grid_searches = {}

    def fit(self, X, y, cv=10, n_jobs=10, verbose=1, scoring=None, refit=False):
        for key in self.keys:
            print("Running GridSearchCV for %s." % key)
            model = self.models[key]
            params = self.params[key]
            gs = GridSearchCV(model, params, cv=cv, n_jobs=n_jobs,
                              verbose=verbose, scoring=scoring, refit=refit,
                              return_train_score=True)
            gs.fit(X,y)
            self.grid_searches[key] = gs

    def score_summary(self, sort_by='mean_score'):
        def row(key, scores, params):
            d = {
                 'estimator': key,
                 'min_score': min(scores),
                 'max_score': max(scores),
                 'mean_score': np.mean(scores),
                 'std_score': np.std(scores),
            }
            return pd.Series({**params,**d})

        rows = []
        for k in self.grid_searches:
            print(k)
            params = self.grid_searches[k].cv_results_['params']
            scores = []
            for i in range(self.grid_searches[k].cv):
                key = "split{}_test_score".format(i)
                r = self.grid_searches[k].cv_results_[key]
                scores.append(r.reshape(len(params),1))

```

```
46              all_scores = np.hstack(scores)
47              for p, s in zip(params,all_scores):
48                  rows.append((row(k, s, p)))
49
50          df = pd.concat(rows, axis=1).T.sort_values([sort_by], ascending=False)
51
52          columns = ['estimator', 'min_score', 'mean_score', 'max_score', 'std_score']
53          columns = columns + [c for c in df.columns if c not in columns]
54
55          return df[columns]
56
57  models1 = {
58
59      'RandomForestRegressor': RandomForestRegressor(random_state=42),
60      'SVR': SVR(),
61      'KNN': KNeighborsRegressor(),
62      'MPL': MLPRegressor(random_state=42, max_iter=10000),
63
64  }
65
66  params1 = {
67
68      'RandomForestRegressor': {
69          'criterion': ['mse', 'mae'],
70          'n_estimators': np.arange(4, 33, 4).tolist(),
71          'max_depth': np.arange(4, 11, 1).tolist(),
72          'max_features': np.arange(5, 15, 1).tolist(),
73
74          },
75      'SVR': {
76          'C':            np.arange( 1, 100+1, 10 ).tolist(),
77          'kernel':       ['linear', 'rbf', 'sigmoid'],
78          'gamma':['scale', 'auto'],
79          'epsilon':[0.1, 0.2, 0.5],
80          },
81      'KNN': {'n_neighbors': np.arange( 2, 30+1, 2 ).tolist(),
82              'weights':['uniform', 'distance'],
83              'algorithm': ['auto', 'ball_tree', 'kd_tree', 'brute'],
84              'leaf_size': np.arange( 30, 100+1, 10 ).tolist(),
85              'p': [1,2],
86              },
87      'MPL': {
88          'hidden_layer_sizes': [(10,),(15,) ,(10,10), (10,10,10),(8,10,8)],
89          'activation': ['tanh', 'relu','identity', 'logistic'],
90          'solver': ['sgd', 'adam'],
91          'alpha': [0.0001, 0.05, 0.1, 0.5, 1,10,100],
92          'learning_rate': ['constant','adaptive', 'invscaling'],
93          'epsilon':[1e-8,1e-7,1e-6,1e-5]
94          },
```

```
95  }
96
97  helper1 = EstimatorSelectionHelper(models1, params1)
98  helper1.fit(X, y, scoring='r2', n_jobs=10)
99  helper1.score_summary()
```

Código A.0.6 – Seleção e aplicação dos modelos

```
from sklearn.model_selection import RepeatedKFold
from numpy import mean
from numpy import std
from matplotlib import pyplot

# get a stacking ensemble of models
def get_stacking():
  # define the base models
  level0 = list()
  level0.append(('KNN', KNeighborsRegressor(
                algorithm='kd_tree',leaf_size=40,n_neighbors=8,weights='distance',p=1)))
  level0.append(('RF', RandomForestRegressor(
                random_state=42,criterion='mse',max_depth=10,max_features=7,n_estimators=20)))
  level0.append(('SVM', SVR(
                epsilon=0.2,C=61,gamma='auto',kernel='linear')))
  level0.append(('RNA', MLPRegressor(
                activation='tanh',random_state=42, max_iter=1000,alpha=1,epsilon=1e-8,
                hidden_layer_sizes=(10,10,10),learning_rate='adaptive',solver='adam')))
  # define meta learner model
  level1 = LinearRegression()
  # define the stacking ensemble
  model = StackingRegressor(estimators=level0, final_estimator=level1, cv=10)
  return model
  print(level0)
# get a list of models to evaluate
def get_models():
  models = dict()
  models['KNN'] = KNeighborsRegressor(
                    algorithm='kd_tree',leaf_size=40,n_neighbors=8,weights='distance',p=1)
  models['RF'] = RandomForestRegressor(
                    random_state=42,criterion='mse',max_depth=10,max_features=7,n_estimators=20)
  models['SVM'] = SVR(
                    epsilon=0.2,C=61,gamma='auto',kernel='linear')
  models['RNA'] = MLPRegressor(
                    activation='tanh',random_state=42, max_iter=1000,alpha=1,epsilon=1e-8,
                    hidden_layer_sizes=(10,10,10),learning_rate='adaptive',solver='adam')

  models['SG'] = get_stacking()

  return models

# evaluate a given model using cross-validation
def evaluate_model(model, X, y):
  # cv = RepeatedKFold(n_splits=10, n_repeats=3, random_state=1)
  scores = cross_val_score(model, X, y, scoring='r2', cv=10, n_jobs=10, error_score='raise')
```

```
46    return scores
47
48  # get the models to evaluate
49  models = get_models()
50  # evaluate the models and store results
51  results, names = list(), list()
52  for name, model in models.items():
53    scores = evaluate_model(model, X, y)
54    results.append(scores)
55    names.append(name)
56    print('>%s %.3f (%.3f)' % (name, mean(scores), std(scores)))
57  # plot model performance for comparison
58  pyplot.boxplot(results, labels=names, showmeans=True)
59  pyplot.show()
```