

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

GABRIEL BALIEIRO BORTOLETI

**USO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA PARA PREDIÇÃO E
CLASSIFICAÇÃO DE VARIÁVEIS DE INTERESSE EM PROCESSO INDUSTRIAL
DE ABATE DE FRANGO**

**FRANCISCO BELTRÃO
2021**

GABRIEL BALIEIRO BORTOLETI

**USO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA PARA PREDIÇÃO E
CLASSIFICAÇÃO DE VARIÁVEIS DE INTERESSE EM PROCESSO INDUSTRIAL
DE ABATE DE FRANGO**

**Use of machine learning algorithms for prediction and classification of
variables of interest in industrial chicken slaughter process**

Trabalho de conclusão de curso de graduação
apresentado como requisito para obtenção do título de
Bacharel em Engenharia Química da Universidade
Tecnológica Federal do Paraná (UTFPR).
Orientador: Prof. Dr. Claiton Zanini Brusamarello.

FRANCISCO BELTRÃO

2021

FOLHA DE APROVAÇÃO

GABRIEL BALIEIRO BORTOLETI

USO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA PARA PREDIÇÃO E CLASSIFICAÇÃO DE VARIÁVEIS DE INTERESSE EM PROCESSO INDUSTRIAL DE ABATE DE FRANGO

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Engenharia Química da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 14 de Maio de 2021

Prof. Claiton Zanini Brusamarello, Dr
Universidade Tecnológica Federal do Paraná - UTFPR

Prof.^(a) Camila Nicola Boeri Di Domenico, Dr.
Universidade Tecnológica Federal do Paraná - UTFPR

Prof. Douglas Junior Nicolin, Dr.
Universidade Tecnológica Federal do Paraná - UTFPR

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

FRANCISCO BELTRÃO

2021

AGRADECIMENTOS

Primeiramente gostaria de agradecer a Deus e a minha família por terem me permitido chegar até aqui e terem sido parte imprescindível da minha caminhada.

A todos meus professores, que através de seus ensinamentos fizeram-me chegar aonde estou. Dentre todos, gostaria de agradecer especialmente aos professores: Ricardo e Mara, da UTFPR do câmpus de Toledo, aos professores Douglas, Camila, Ana Paula Romio, Ana Paula Oliveira, Fernanda, Maiquiel, Guilherme, Mauro, Michele e Vilmar do câmpus de Francisco Beltrão, e tantos outros do qual contribuíram para minha caminhada.

Gostaria de agradecer imensamente ao professor Claiton Zanini Brusamarello pela paciência, pela oportunidade, pelo tempo de trabalho e pelos ensinamentos tanto profissionais como pessoais.

Entre tantos amigos que vieram e que foram, desejo-lhes sorte e felicidade acima de tudo e gostaria de agradecer de coração à: Ingo, Joyce, Viviane, Vanessa, Daniella, Gabriela, Marília, Jaqueline, Gustavo, Lucas e Jennifer por toda a amizade durante todos estes anos.

Gostaria de agradecer também à Bruno e a Giovani que além de amigos exemplares, muito me ensinaram também sobre a vida, e a eles eu devo muito.

Aos meus pais por nunca desistirem e sempre me apoiarem em todos os momentos de minha vida, lhes devo tudo.

Por fim à minha querida Crisllainy, gostaria de agradecer pelos momentos juntos nessa caminhada e por toda a companhia e momentos de alegria que têm me proporcionado, obrigado por estar ao meu lado.

Estes anos foram excepcionais graças a todos vocês.

RESUMO

No processo industrial de abate de frango, a etapa mais importante e de mais atenção exigida é a etapa do resfriamento, é nela que se define o teor de absorção de água final, a temperatura final, as características físicas da carne e controla o índice microbiológico do produto. Nesta etapa, ocorrem processos de transferência de massa e energia, num regime transiente com volume de controle sem um formato fixo e com diferentes variáveis de processo que influenciam no resultado final, logo os cálculos para se estimar a temperatura, absorção de água final na carcaça e classificar quais das variáveis do processo são as mais influentes, acabam sendo complicados e envolvem diversas aproximações que tornam o resultado final não condizente com o real obtido industrialmente. O foco deste estudo é utilizar métodos de inteligência artificial, mais exatamente, sistema de aprendizado de máquina para realizar a predição e classificação das variáveis envolvidas no processo de resfriamento da carne de frango. Objetivou-se prever os valores de absorção de água na carcaça ao final do processo, utilizando-se dados obtidos da tese de Sant'anna em 2008, aplicando três diferentes algoritmos, entre eles: *K-Star*, Otimização Sequencial Mínima para Regressão (SMOReg) e Redes Neurais Artificiais. Ao total realizaram-se 206 testes envolvendo os três algoritmos. Para a validação dos resultados foram utilizados o coeficiente de correlação (R) e o erro médio quadrado (MSE). Para o algoritmo *K-Star*, a predição dos resultados e a classificação das variáveis apresentaram resultados de 50,11% de correlação e a capacidade de classificar duas das três variáveis mais importantes. Para o algoritmo SMOReg, a função que se destacou com o melhor resultado foi a Função de Base Radial (RBF) com 99,87% de correlação e a capacidade de classificar as três variáveis mais importantes. Para o algoritmo das Redes Neurais Artificiais, a melhor configuração de sua estrutura obteve um resultado de 99,09% de correlação e a capacidade de classificar as três variáveis mais importantes. Por fim, analisando os resultados individuais do coeficiente de correlação (R) e a capacidade de classificação das variáveis dos três métodos, o que apresentou o melhor resultado foi o algoritmo SMOReg para ambas as tarefas.

Palavras-chave: avicultura industrial; aprendizado de máquina; *k-star*; smoreg; redes neurais artificiais.

ABSTRACT

In the industrial chicken slaughtering process, the most important and most required step is the cooling step, where the final water absorption content, the final temperature, as the physical characteristics of the meat and controls, the microbiological index of the product is defined. In this step, mass and energy transfer processes take place, in a transient regime with control volume without a fixed format and with different process variables that influence the final result, so the calculations to estimate the temperature, final water absorption in the carcass and classify which of the process variables are the most influential, end up being complicated and involve several approximations that end up making the final result not to be consistent with the real industrially processed. The focus of the study is to use artificial intelligence methods, more precisely, machine learning system to perform the prediction and classification of the variables involved in the chicken meat cooling process. The objective was to predict the water absorption values in the carcass at the end of the process, using data obtained from Sant'anna's thesis in 2008, applying three different algorithms, among them: *K-Star*, Minimum Sequential Optimization for Regression (SMOReg) and Artificial Neural Networks. In total, 206 tests were carried out involving the three algorithms. For the validation of the results, the correlation coefficient (R) and the mean square error (MSE) were used. For the *K-Star* algorithm, a prediction of the results and a classification of the results variables of 50.11% correlation and an ability to classify two of the three most important variables. For the SMOReg algorithm, a function that stands out with the best result for a Radial Base function (RBF) with 99.87% correlation and an ability to classify as three most important variables. For the Artificial Neural Networks algorithm, the best configuration of its structure obtained a result of 99.09% correlation and an ability to classify as three most important variables. Finally, analyzing the individual results of the correlation coefficient (R) and the ability to classify the variables of the three methods, the one that presents the best result was the SMOReg algorithm for both tasks.

Keywords: industrial poultry; machine learning; *k-star*; smoreg; artificial neural networks.

LISTA DE FIGURAS

Figura 1 - Dados de produção anual de frango	20
Figura 2 - Processo do abate de frango	22
Figura 3 - Representação de um pré-chiller e um chiller industrial	27
Figura 4 - Tanques de resfriamento em abatedouro	30
Figura 5 - Vertentes do aprendizado de máquina	37
Figura 6 - Efeitos do parâmetro C	47
Figura 7 - Três valores diferentes de expoente para a função polinomial	49
Figura 8 - Comportamento da função de acordo com variação do ω e σ	51
Figura 9 - Fluxograma processo WEKA	62
Figura 10 - Filtro de normalização de dados	64
Figura 11 - Separação dos dados	65
Figura 12 - Esquema da rede neural artificial aplicada neste trabalho.	66
Figura 13 - Esquema algoritmo K*	68
Figura 14 - Representação do algoritmo SMOReg	69
Figura 15 - Sequência de passos para classificação	74
Figura 16 - Resultado melhor configuração K-Star	77
Figura 17 - Melhor resultado SMOReg Polinomial Kernel	79
Figura 18 - Representação da mudança de parâmetros Gama	82
Figura 19 - Melhor resultado SMOReg RBF Kernel	83
Figura 20 - Melhor resultado SMOReg PUK	86
Figura 21 - Esquematização da rede neural com melhor configuração	92
Figura 22 - Melhor resultado RNA	92
Figura 23- Junção dos melhores resultados dos algoritmos	94
Figura 24 - Conjunto de dados trabalhados	119

LISTA DE QUADROS

Quadro 1 - Resultados encontrados de busca	17
Quadro 2 - Conjunto de Dados	63
Quadro 3 - Configurações testadas da rede neural	67
Quadro 4 - Valores de GB para testes aplicados	69
Quadro 5 - Funções Kernel	71
Quadro 6 - Parâmetros de teste SMOReg	72
Quadro 7 - Resultados testes K-Star	75
Quadro 8 - Resultados SMOReg Polinomial	78
Quadro 9 - Resultados testes SMOReg RBF Kernel	80
Quadro 10 - Resultados testes SMOReg PUK	84
Quadro 11 - Resultados RNA de uma camada	87
Quadro 12 - Resultados RNA de 2 camadas	88
Quadro 13 - Comparação dos melhores resultados gerais	93
Quadro 14 - Explicação física das variáveis mais influentes	96
Quadro 15 - Resultados classificação K-Star	97
Quadro 16 - Resultados classificação SMOReg	98
Quadro 17 - Testes classificação Redes Neurais	99
Quadro 18 - Resultados dos valores individuais K-Star	114
Quadro 19 - Resultados individuais SMOReg Kernel	115
Quadro 20 - Resultados individuais SMOReg RBF Kernel	116
Quadro 21 - Resultados individuais SMOReg PUK	117
Quadro 22 - Resultados individuais RNA melhor configuração	118

LISTA DE ABREVIATURAS

ABPA	Associação Brasileira de Proteína Animal
AM	Aprendizado de Máquina
IA	Inteligência Artificial
KKT	Condições de Karush Kuhm Tucker
KNN	Modelo de Vizinho Mais Próximo (<i>K Nearest Neighbour</i>)
K*	<i>K-Star</i>
MAPA	Ministério da Agricultura, Pecuária e Abastecimento
MLP	Rede Neural <i>Multilayer Perceptron</i>
MSE	Erro Médio Quadrado (<i>Mean Square Error</i>)
PUK	Função Kernel Universal de Pearson VII
R	Coefficiente de Correlação
RBF	Função Kernel de Base Radial
RNA	Rede Neural Artificial
SMO	Algoritmo de Otimização Mínima Sequencial (<i>Sequential Minimal Optimization</i>)
SMOReg	Algoritmo de Otimização Mínima Sequencial para Regressão
SMV	Máquina de Vetor de Suporte (<i>Support Vector Machine</i>)
WEKA	<i>Waikato Environment for Knowledge Analysis</i>

SUMÁRIO

1	INTRODUÇÃO	13
2	OBJETIVOS	16
2.1	OBJETIVO GERAL	16
2.2	OBJETIVOS ESPECÍFICOS	16
3	FUNDAMENTAÇÃO TEÓRICA	17
3.1	ANÁLISE DOS TÓPICOS APRESENTADOS	17
3.2	AVICULTURA DE CORTE	18
3.3	PROCESSO DE ABATE DE FRANGO	22
3.3.1	Resfriamento da Carne	24
3.3.2	Resfriamento em Tanques Contínuos	26
3.3.2.1	<i>Pré-Chiller</i>	27
3.3.2.2	<i>Chiller</i>	29
3.3.3	Variáveis Influenciadoras nos Efeitos de Resfriamento e Absorção	30
3.3.3.1	Resfriamento da Carcaça	31
3.3.3.2	Absorção de Água da Carcaça	31
3.3.4	Controle de Qualidade Industrial no Processo de Resfriamento	33
3.3.5	Uso dos Dados no Processo Industrial	34
3.4	INTELIGÊNCIA ARTIFICIAL E APRENDIZADO DE MÁQUINA	34
3.4.1	Inteligência Artificial (IA)	34
3.4.2	Aprendizado de Máquina (AM)	35
3.4.3	Métodos de Aprendizado de Máquina	36
3.4.4	Vantagens ao se utilizar o Aprendizado de Máquina	38
3.4.5	Desvantagens ao se utilizar o Aprendizado de Máquina	38
3.4.6	Cooperação entre Seres Humanos e Máquinas	40
3.4.7	O Uso de Aprendizado de Máquina no Cenário Industrial e o Crescente uso de Softwares	40
3.5	CLASSIFICADORES BASEADOS EM INSTÂNCIAS E MODELO “ <i>K NEAREST NEIGHBOUR</i> ” (KNN)	42
3.5.1	Vantagens e Desvantagens de Métodos de Aprendizados Baseados em Instâncias	43
3.5.2	Algoritmo <i>K-Star</i> (K^*)	44
3.5.3	Vantagens e Desvantagens do K^*	44

3.6	MODELO DE MÁQUINA DE VETOR DE SUPORTE (SVM)	45
3.6.1	Definição	45
3.6.2	Treinamento de um SVM	47
3.6.2.1	Parâmetro C (<i>Cost</i>)	47
3.6.3	Diferentes Tipos de Função Kernel	48
3.6.4	Vantagens e Desvantagens do Uso de SVM	51
3.6.5	Algoritmo de Otimização Mínima Sequencial (SMO)	52
3.6.6	Treinamento de um Algoritmo SMO	53
3.6.7	Vantagens e Desvantagens do Algoritmo SMO	54
3.7	REDES NEURAIS ARTIFICIAIS (RNA'S)	54
3.7.1	Definição	54
3.7.2	Construção e Componentes de uma Rede Neural Artificial	56
3.7.3	Classificação de Diferentes Tipos de Rede Neural Artificial	57
3.7.4	Vantagens e Desvantagens de uma Rede Neural Artificial	58
3.7.5	Rede Neural Artificial <i>Multilayer Perceptron</i> (MLP)	59
3.7.6	Aplicação do Algoritmo das Redes Neurais Artificiais no Ramo da Avicultura de Corte	60
3.8	SOFTWARE “ <i>WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS</i> ” (<i>WEKA</i>)	61
4	PROCEDIMENTOS METODOLÓGICOS	63
4.1	PRÉ-PROCESSAMENTO DOS DADOS	63
4.2	TESTES EXPERIMENTAIS	65
4.2.1	REDES NEURAIS ARTIFICIAIS	65
4.2.2	Algoritmo <i>K-Star</i> (K^*)	67
4.2.3	Algoritmo de Otimização Mínima Sequencial para Regressão (SMOReg)	69
4.3	AVALIAÇÃO E VALIDAÇÃO DOS RESULTADOS	72
4.3.1	Coeficiente de Correlação	72
4.3.2	Erro Médio Quadrado	73
4.4	CLASSIFICAÇÃO DAS VARIÁVEIS	73
5	RESULTADOS E DISCUSSÕES	75
5.1	PREDIÇÃO DO VALOR DA ABSORÇÃO	75
5.1.1	Seleção da melhor configuração para o algoritmo <i>K-Star</i> (K^*)	75
5.1.2	Seleção para a melhor configuração para o algoritmo SMOReg.	77

5.1.2.1	Kernel Polinomial	78
5.1.2.2	Função Kernel de Base Radial (RBF)	80
5.1.2.3	Kernel Universal de Pearson VII (PUK)	84
5.1.3	Redes Neurais Artificiais	87
5.2	DEFINIÇÃO DO MELHOR MODELO	93
5.3	CLASSIFICAÇÃO DAS VARIÁVEIS MAIS INFLUENTES NA ABSORÇÃO DE ÁGUA NAS CARCAÇAS DE FRANGO	95
5.3.1	<i>K-Star</i>	97
5.3.2	SMOReg	98
5.3.3	Redes Neurais Artificiais	99
5.4	COMPARAÇÃO E SELEÇÃO DO ALGORITMO MAIS EFICIENTE	100
5.5	COMPARAÇÃO COM A LITERATURA	101
6	CONCLUSÃO	103
	REFERÊNCIAS	105

1 INTRODUÇÃO

O setor avícola brasileiro possui um elevado grau de organização e coordenação, isso o leva a ter uma vantagem sobre outros países quando se refere à produção da carne de frango (PINOTTI, 2005). A produção brasileira de carne de frango atingiu no ano de 2019 um total de 13,245 milhões de toneladas (ficando atrás apenas dos Estados Unidos e da China no mercado mundial). Deste total produzido, 68% foi direcionado ao mercado interno e 32% foi direcionado à exportações (sendo o Brasil o maior exportador de carne de frango do mundo). O resultado dessa exportação da carne de frango gerou uma receita de 6,994 milhões de dólares ao total (ABPA, 2020).

O processo de abate de frango industrial se resume nas seguintes etapas: recebimento, pendura, insensibilização, sangria, escaldagem, depenagem, cortes e extrações, lavagem da carcaça e por fim o resfriamento (SANT'ANNA, 2008). Os métodos de resfriamento são utilizados para se atingir parâmetros desejados do produto (carcaça) antes de serem embalados para venda.

A etapa de resfriamento no processo é considerada uma das mais importantes, pois o não atendimento da temperatura ou da absorção determinadas pela legislação, Portaria nº 210, do Ministério da Agricultura e do Abastecimento que diz que a absorção da água nas carcaças de aves submetidas ao pré-resfriamento por imersão, não deverá ultrapassar a 8% de seus pesos (BRASIL, 1998). Isso pode gerar perdas financeiras para o cliente ou para a indústria, além de problemas microbiológicos envolvidos no processo (KLASSEN, 2008). Entre os diferentes métodos existentes para o resfriamento, dois se destacam: o resfriamento com ar forçado e o resfriamento com água, sendo este último o mais utilizado, por sua rapidez e eficiência (LORENZETTI et al., 2018; CARCIOFI E LAURINDO, 2010).

Segundo Carciofi (2005), o resfriamento das carcaças pela indústria é um problema de engenharia de grande complexidade, primeiramente devido ao número de variáveis envolvidas no processo do resfriamento da carcaça e o fato de todas essas variáveis apresentarem uma influência sobre o valor final da absorção de água na mesma. Outra justificativa é a contínua variação da massa da carcaça ou da temperatura do meio onde acontecerá o resfriamento, envolvendo questões de transferência de massa e de energia simultaneamente onde o volume de controle não

possui dimensões constantes, podendo variar de carcaça para carcaça, assim não sendo possível utilizar somente coordenadas cartesianas, esféricas ou cilíndricas para toda a carcaça, e sim uma junção destas três. Devido a estes fatores mencionados, são utilizadas estimativas ou valores médios para os cálculos de transferência de massa e energia, como retratado no trabalho feito por Bailone e Roça (2017), que utilizaram um valor médio para o peso das carcaças, sendo que esta é uma das variáveis que pode influenciar na absorção da água na mesma.

Visto que cálculos manuais de balanço de massa, energia ou outros tipos de balanço se tornam muito difíceis para este tipo de processo industrial, têm-se recorrido a técnicas computacionais baseadas em modelagem matemática, como as Redes Neurais Artificiais, para a resolução deste tipo de problema. Estas técnicas computacionais aplicadas ao processo industrial de frango são retratados nos trabalhos de Müller (2019), Pinheiro e Pasa (2020), Reali (2004) e Klassen (2008), em que todos utilizaram Redes Neurais Artificiais para se atingir seus objetivos. Embora já se tenha um número relativamente aceitável de trabalhos envolvendo inteligência artificial aliada ao processo global de abate de frango, o uso de diferentes técnicas é muito restrito e frequentemente se mantém em apenas Redes Neurais Artificiais, se tendo pouco conhecimento de outras técnicas que possam ser aplicadas com o mesmo intuito no respectivo processo industrial.

As técnicas computacionais mencionadas exigem uma breve preparação do conjunto de dados a serem estudados para que os algoritmos possam realizar a classificação do conjunto de maneira automática. Muitos destes algoritmos podem não fornecer uma explicação compreensível sobre a razão pela qual uma classe obteve determinado peso. Logo, há a necessidade de se fazer uma análise física/técnica de cada peso encontrado pelo algoritmo e procurar otimizar então o processo de acordo com a variável mais influente nele, aliando assim a inteligência artificial com o parâmetro físico do processo.

A partir desta, o analista pode aferir a acurácia do algoritmo e utilizar o mesmo modelo criado para a análise de novos dados. Dentre essas técnicas, pode-se citar a mineração de dados e o aprendizado de máquina, que empregam conceitos de inteligência artificial para tomar decisões baseadas em treinamentos prévios (FERNANDES, 2019).

A inteligência artificial (IA) é um produto da combinação de modelos matemáticos sofisticados, aliados à computação, permitindo o desenvolvimento de

algoritmos complexos capazes de simular a inteligência humana (MARQUES et al., 2019). Para trabalhar-se com estes diferentes tipos de algoritmos, existem atualmente alguns programas disponíveis, como o *MATLAB* (*MathWorks Inc*, 197-), mais conhecido no meio acadêmico. Porém muitos deles exigem chave de acesso, tornando assim o processo oneroso financeiramente. Uma alternativa se torna a procura por softwares de livre acesso, como o *WEKA* (*University of Waikato*, 2014), que será utilizado no presente trabalho.

O *WEKA* é capaz de trabalhar com vários tipos de algoritmos em sua interface, sendo cada tipo designado para uma função específica, como classificação, associação e clusterização. Para dados numéricos, utiliza-se os algoritmos de classificação e regressão.

Já existem pesquisas realizadas envolvendo Inteligência Artificial e o processo industrial de frango de corte, principalmente para predição de valores de absorção e temperatura da carcaça, como exemplo, os trabalhos feitos por Klassen et al. (2009), e Reali (2004), porém apenas um tipo de algoritmo é comumente utilizado dentre estes trabalhos, que são as redes neurais artificiais (RNA's).

Atualmente é muito escasso o número de trabalhos publicados que utilizam diferentes tipos de algoritmos de aprendizado de máquina além das redes neurais para predição de valores envolvendo diferentes características do processo de abates de frangos. Para que o uso de novas técnicas de inteligência artificial sejam implantadas em escala industrial para processos de resfriamento por imersão (visto que no Brasil, as indústrias não utilizam o método de resfriamento por ar forçado (KLASSEN et al., 2009)), são necessários mais trabalhos de pesquisas envolvendo tais algoritmos.

O presente trabalho visa otimizar o processo da avicultura industrial por meio da utilização de algoritmos de aprendizado de máquina através de ferramentas computacionais (Softwares). Esta otimização é feita em decorrência da predição do valor final da variável de saída de interesse, assim como a influência de cada variável de entrada (definida manualmente) no processo. Assim, com antecedência, tomar-se-ão os planos de ações de quais adaptações podem ser realizadas no processo para melhoria do mesmo, podendo assim conciliar-se a área de engenharia de processos com a área de inteligência artificial.

2 OBJETIVOS

2.1 OBJETIVO GERAL

O presente trabalho tem como objetivo a predição de valores de absorção de água na carcaça de frango e a classificação de suas variáveis em um processo industrial de abate utilizando algoritmos de aprendizado de máquina através do software WEKA (versão 3.8.4).

2.2 OBJETIVOS ESPECÍFICOS

- Testar os três seguintes modelos computacionais para a predição da absorção de água no processo de resfriamento de carne de frango:
 - *Redes Neurais Artificiais do tipo Multilayer perceptron;*
 - *K-Star (K*);*
 - *Sequential Minimal Optimization for Regression (SMOReg);*
- Ajustar os parâmetros de cada algoritmo durante os testes visando obter o melhor desempenho;
- Apresentar os modelos com melhor e pior desempenho;
- Validar o melhor método através dos parâmetros estatísticos: Coeficiente de Correlação (R) e Erro Quadrado Médio (MSE);
- Classificar as variáveis mais influentes no processo através dos melhores e piores configurações dos modelos computacionais testados;
- Comparar com dados já existentes na literatura;
- Analisar estatisticamente e tecnicamente os dados de predição e classificação pelo modelo mais aceito;

3 FUNDAMENTAÇÃO TEÓRICA

No capítulo a seguir será abordada toda a base teórica dos assuntos utilizados no presente trabalho, obtida através de pesquisas na literatura nacional e estrangeira para corroborar com as informações apresentadas pelo próprio autor. Na primeira seção será abordada uma breve análise dos temas inclusos no presente trabalho, seguindo com a avicultura de corte, sua história e sua produção, posteriormente será feita a entrada no tópico do processo industrial de abate de frango, seus métodos, assim como a etapa do resfriamento da carcaça. Após sair do contexto da avicultura industrial, serão abordados, então, tópicos de inteligência artificial, aprendizado de máquina e os respectivos algoritmos utilizados no presente trabalho.

3.1 ANÁLISE DOS TÓPICOS APRESENTADOS

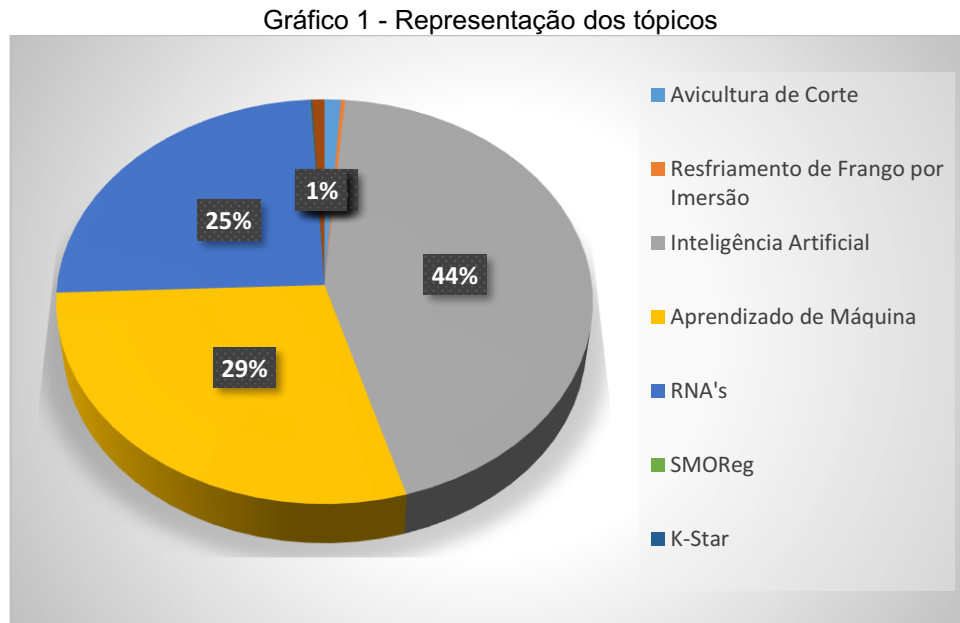
Existe uma variedade muito alta de estudos já feitos retratando alguns dos temas abordados no presente trabalho, através de uma pesquisa no portal “Google Scholar” para se encontrar, entre eles: Artigos, dissertações ou teses, citações, patentes, livros, entre outros, pôde se encontrar uma alta gama de resultados dos temas estudados. Conciliando os resultados encontrados em inglês e português, os resultados obtidos estão representados no Quadro 1:

Quadro 1 - Resultados encontrados de busca

Tema	Resultados Encontrados
Avicultura de Corte	32.800
Resfriamento de Frango por Imersão	8.050
Inteligência Artificial	1.354.900
Aprendizado de Máquina	889.000
RNA's	756.700
SMOReg	1.060
<i>K-Star</i>	3.455
WEKA	25.000

Fonte: Autoria própria (2020).

Representando-os graficamente, têm-se então:



Fonte: Autoria própria (2020).

Um fator imprescindível de se ressaltar é o enorme decréscimo nos resultados encontrados através da comparação de cada tema. Atualmente, o número de trabalhos publicados envolvendo a predição de uma variável no resfriamento de frangos por imersão, incluindo tanto trabalhos em inglês quanto em português, não chegam a 10 no total, desde o ano de 1978 até o ano de 2020. Quando se trata de classificação de variáveis, o número de trabalhos não apresenta grande aumento, ficando também na casa da dezena.

Por fim, não foram encontrados pelo autor, trabalhos envolvendo predição de valores da variável de interesse no processo da avicultura industrial utilizando algoritmos diferentes das Redes Neurais Artificiais, tampouco envolvendo uma comparação dos 3 algoritmos trabalhados: RNA's, SMOReg e *K-Star*.

3.2 AVICULTURA DE CORTE

A prática da avicultura em geral é tão antiga quanto a história da humanidade. Porém, quando se trata da avicultura de corte, foi apenas em 1975 que a mesma se consolidou no mundo como uma das mais importantes fontes de proteína animal. No

Brasil, a primeira referência de chegada destas aves em território nacional remete à época das colonizações portuguesas, onde relata-se na carta de Pero Vaz de Caminha a comprovação de que os índios locais não conheciam este tipo de aves. Até o final do século XIX, a carne animal era escassa de se encontrar no comércio e também relativamente custosa. Independentemente de seu continente, foi à partir do século XX que o comércio e atividades produtoras das mesmas se desenvolveu em maior escala (EMBRAPA, 2011; COSTA, 2011).

Os primeiros passos da avicultura brasileira foram dados por produtores familiares, onde a produção de aves era uma das responsáveis pela geração de renda da propriedade. Inicialmente voltada à subsistência, prevendo a comercialização apenas dos excedentes a avicultura tornou-se rapidamente comercial pouco antes de 1930, principalmente após a fundação da Cooperativa Avícola de São Paulo e de outras associações cooperativas avícolas, conforme afirmado por Arashiro (1989) e Malavazzi (1982). O desenvolvimento da avicultura se efetivou na década de 70, com a entrada de empresas processadoras no mercado e especialistas no processo de produção do frango. Transformações tecnológicas, técnicas de produção intensiva e o desenvolvimento de genética adaptada contribuíram para o avanço da atividade (ZEN et al., 2015).

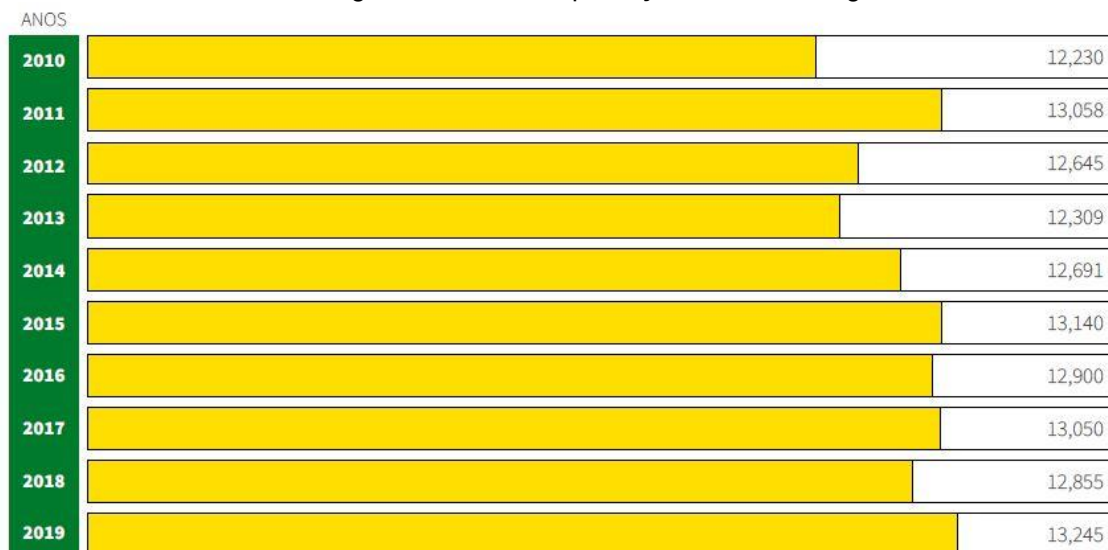
O sistema de produção integrada continua sendo empregado no Brasil desde então. Este sistema é constituído de uma parceria formada entre os produtores e agroindústria para produzir animais de maior qualidade. A indústria é responsável por oferecer os animais, rações, medicamentos, transporte dos mesmos e de seus insumos, além da assistência técnica necessária à produção. O produtor por sua vez, participa com as instalações, equipamentos, água e energia elétrica, além de criar e engordar os animais até chegado o momento do abate (DIAS, 2016).

Segundo Araújo (2007), as vantagens para os produtores rurais são: diminuição de desembolsos financeiros, segurança de venda dos produtos, menor dificuldade na obtenção de financiamentos agropecuários, garantia de assistência técnica, maior comodidade, maior possibilidade de especialização, utilização de produtos e de mão de obra próprios, elevando sua renda e diminuindo seus custos. Já as vantagens para as empresas integradoras são: garantia de sua matéria-prima no momento certo, terceirização da produção, que acarreta (a) diminuição dos recursos financeiros aplicados, (b) pulverização de riscos, (c) diminuição de encargos sociais e (d) maior ganho financeiro.

Para Malavazzi (1982), um dos fatores essenciais que contribuíram para o aumento da produção avícola durante a década de 70 até os dias atuais foi a consolidação propriamente dita da exploração avícola econômica industrial. Além deste, outro fator essencial foi o início das atividades de exportação de frangos de corte, inteiro e em partes, através do grupo Sadia e de outros grupos do ramo avícola.

Embora a atividade de exportação brasileira tenha começado apenas em 1975, estima-se que somente dez anos depois começaram as exportações de partes de frango e, desde então, sua participação tenha aumentado até os dias atuais (EMBRAPA, 2011). Segundo dados do relatório anual da Associação Brasileira de Proteína Animal (ABPA) para 2019, a produção brasileira de carne de frango foi de 13,245 milhões de toneladas, a maior desde 2010 como mostra a Figura 1:

Figura 1 - Dados de produção anual de frango



Fonte: ABPA (2020).

Desta produção anual, os estados que mais se destacaram no processo de abate de frango foram: Paraná (34,69%), Santa Catarina (15,40%) e Rio Grande do Sul (14,32%). No mercado mundial, o Brasil se firmou em segundo lugar como maior produtor de carne de frango, atrás apenas dos Estados Unidos (com 19.941 milhões de toneladas produzidas) (ABPA, 2020).

Quando se trata de exportação, o Brasil está no topo da lista (com 4.214 mil toneladas), seguido pelos Estados Unidos, que exportou 3,261 mil toneladas (ABPA, 2020). A manutenção da liderança brasileira nas exportações teve como um dos fatores importantes a ausência de casos de gripe aviária no país no decorrer dos

últimos anos. Entre os anos de 2014 e 2015, os Estados Unidos, principal concorrente brasileiro no mercado internacional, tiveram casos de Influenza aviária (sendo considerado o pior surto da doença no país, matando 50 milhões de aves segundo o departamento de agricultura dos Estados Unidos - USDA), fazendo com que os importadores optassem por diminuir a demanda do maior produtor de carne de frangos no mundo (OLIVA, 2015; RAMOS et al., 2017).

A avicultura foi um dos primeiros setores do agronegócio a investir na produção em escala, com o intuito de continuar atendendo ao crescente aumento da população brasileira e sua consequente demanda pelo mercado alimentício. A carne de frango foi o produto que apresentou a maior evolução de consumo em território brasileiro desde 1970. De acordo com Mendes e Padilha Junior (2007): *“Nenhum alimento no Brasil experimentou expansão tão grande de consumo aparente quanto a carne de frango. De menos de 3 quilos no início dos anos 70 ultrapassou os 32,9 kg/hab/ano em 2005, ou seja, um crescimento em torno de 1200% em 35 anos, o que corresponde a um aumento anual de 7,6%”*. Em 2019 segundo a ABPA, o consumo da carne de frango foi de 42,84 Kg/hab/ano, que indicou um crescimento em torno de 30% em 14 anos, em torno de 2,1% ao ano.

Atualmente, o mercado alimentício brasileiro aponta uma liderança para o consumo de carne de aves no país, devido à melhoria na renda da população, ao baixo preço em relação a outras carnes, à agregação de valor e a diversificação das linhas de produção para agradar o consumidor final. De todos estes fatores, o mais influente é certamente o preço desta carne em relação à outras (EMBRAPA, 2011).

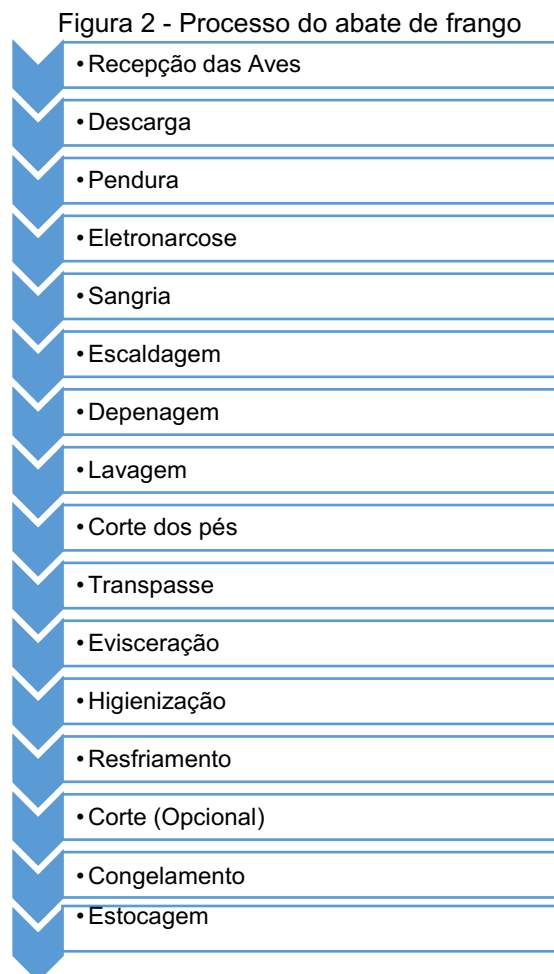
As previsões extraídas do relatório da Organização para a Cooperação e Desenvolvimento Econômico – OCDE, e da Organização das Nações Unidas para a Alimentação e a Agricultura - FAO (OECD/FAO, 2017), empregam que o Brasil superaria 23% das exportações globais de carne (que englobam: bovina, suína e aves) até 2017, movimentando um montante em torno de 8,7 milhões de toneladas. Já o comércio de aves aumentaria 1,9% em 2018 e alcançaria 13,3 milhões de toneladas. Apesar de ser o principal exportador global de frango, o país exportaria menos frango por causa de embargos de acesso aos tradicionais destinos de exportação (MUNDO AGRO, 2018).

O Departamento de Agricultura dos EUA estimou, em suas primeiras projeções, que a exportação de carne de frango no Brasil aumentaria perto de 2,5% em 2019, porém, sendo um valor pequeno se comparado com o aumento de 4,18%

da média mundial (MUNDO AGRO, 2018). Porém, de acordo com números do boletim anual feito pela Associação Brasileira de Proteína Animal - ABPA (2020), que expõe dados de 2019 no território nacional, houve um acréscimo de 6,4% no aumento da exportação da carne de frango no Brasil (6,994 mil ton em 2019 contra 6,571 mil ton em 2018, que acarreta um crescimento de 0,423 mil ton em um ano). Portanto, houve uma melhora acima do previsto para a atividade de exportação de frango brasileira.

3.3 PROCESSO DE ABATE DE FRANGO

Nos próximos parágrafos é descrito o processo industrial de abate de aves, de acordo com Carciofi (2005), Klassen (2008), Barzotto (2013) e Sarcinelli e Venturini (2007), onde relata-se desde a chegada das aves ao abatedouro até o processo de estocagem do produto final. Na Figura 2 está resumido todo o processo de abate de frango em suas principais etapas:



Fonte: Autoria própria (2020).

A primeira etapa do processo do abate da ave acontece no galpão de espera, na recepção dos lotes. Neste lugar, as aves chegam acondicionadas em gaiolas plásticas e são mantidas em repouso através de um sistema de ventiladores (às vezes utiliza-se aspersão com água), que têm por finalidade diminuir a mortalidade no transporte/espera, bem como minimizar a desidratação.

Após a recepção, tem-se a descarga das aves. Elas são pesadas e colocadas próximas ao processo de “pendura”, que é feito em ganchos de aço inoxidável numa “nórea” de transporte. Esta nórea consiste em um trilho, no qual as aves são penduradas pelos pés, e então são transportadas ao interior do abatedouro para o processamento.

Dentro do abatedouro as aves inicialmente passam pela insensibilização por meio de descarga elétrica (eletronarcole) em um tanque de água contendo um eletrodo, outro eletrodo toca o gancho, as aves então têm suas cabeças mergulhadas no tanque. Esse processo dura em torno de 7 segundos e o objetivo é diminuir o sofrimento da ave, assim como o índice de descarte em virtude de lesões e hemorragias internas causadas pelo fato das aves se debaterem ao serem encaminhadas aos processos seguintes; também tem como objetivo inibir parcialmente as reações bioquímicas post-mortem, atuando na maciez do peito.

Na sangria, é efetuado um corte no pescoço do frango, podendo ser de forma mecânica ou manual, que percorre um túnel por cerca de três minutos, tempo este necessário para uma sangria eficiente. A sangria manual muitas vezes é exigência para exportação.

As duas etapas subsequentes têm por intuito remover as penas do frango: A escaldagem é o processo no qual a ave passa imersa em um tanque com água quente (geralmente entre 50°C e 70°C) provocando a troca térmica, que ocasiona a abertura dos folículos da pele onde estão fixadas as penas facilitando a depenagem (deve-se ter atenção para a escaldagem para esta não ultrapassar 2 minutos, pois pode ocorrer o cozimento da carcaça e um tempo muito menor pode não causar a abertura dos folículos). A depenagem é etapa onde as penas são retiradas mecanicamente, normalmente através de paletas giratórias de borracha.

Posteriormente é feita uma lavagem, que se destina a higienização das carcaças no intuito de diminuir a carga microbiana superficial. Após a sua liberação,

há o corte dos pés, que são classificados e destinados para exportação ou fábrica de farinhas.

Através do transpasse é feita a troca de nóreas. As carcaças, então, seguem para uma sequência de operações chamada de evisceração, que consiste na:

- Extração da cloaca;
- Corte abdominal;
- Corte da pele do pescoço;
- Eventração (exposição das vísceras);
- Separação das vísceras em comestíveis e não comestíveis (o coração,

a moela e o fígado são comestíveis, logo seguem para industrialização, enquanto as demais vísceras vão para a fábrica de farinhas);

- Retirada da cabeça;
- Lavagem das carcaças;
- Extração da traqueia.

Submetida a uma higienização posterior, as carcaças então seguem para o processo de resfriamento, que será discutido subseqüentemente. Após este processo de resfriamento, as carcaças são redirecionadas para o congelamento (ou então para o processo de corte de suas partes, caso a carcaça não seja vendida inteira) e posterior estocagem do produto final.

3.3.1 Resfriamento da Carne

É definido como resfriamento (ou pré-resfriamento) o processo de abaixamento da temperatura das carcaças de aves, que é realizado imediatamente após as etapas de evisceração e lavagem. Após o abate, a carcaça do frango tem de ser submetida ao resfriamento para reduzir e manter sua temperatura abaixo do valor que irá garantir uma alta qualidade e segurança do produto (JAMES et al., 2005; CARCIOFI, 2005).

Em âmbito nacional, é necessário realizar a redução de temperatura da carcaça (de 40°C que é a temperatura média inicial do frango ao chegar no processo de abate, até em torno de 5 a 7°C que é a temperatura que a mesma deve atingir ao final do *chiller*), logo após os primeiros instantes do abate, para evitar fatores como o crescimento de microrganismos que podem reduzir a vida útil do produto (OLIVO, 2006; CARCIOFI e LAURINDO, 2007). A legislação europeia define que a temperatura

final da carcaça deve ser de 4°C, enquanto para os EUA, sua legislação impõe que a temperatura deve ser de 4,4°C em até 4h post mortem (JAMES et al., 2005).

O processo básico industrial para abate de frango é similar em muitas empresas, seguindo o modelo discutido anteriormente. Em cada frigorífico, as atividades são realizadas de acordo com a capacidade produtiva de cada empresa e/ou sua respectiva demanda, podendo existir algumas adaptações. Estas, porém, devem seguir a padronização exigida pelo Regulamento Técnico da Inspeção Tecnológica e Higiênico-Sanitária de Carne de Aves do Ministério da Agricultura Pecuária e Abastecimento (MAPA). No regulamento consta que o teor de absorção de água nas carcaças de frango submetidas ao pré-resfriamento não deve exceder 8% de seu peso. O limite de absorção exigido na Europa é imposto apenas para carcaças vendidas inteiras, e não para cortes. Quando a carcaça passa pelo processo de corte, calcula-se uma relação entre os percentuais de umidade e proteína, com limites especificados para cada produto, o que não se aplica à legislação brasileira (KLASSEN (2008); CARCIOFI e LAURINDO (2007); BRASIL (1998)).

Devido a fatores relacionados com a estética e segurança da carne, a etapa de pré-resfriamento é obrigatória pela legislação. Dentre esses métodos, existem dois que se destacam:

- Imersão em tanques contínuos com água gelada (*chillers*);
- Resfriamento por ventilação forçada em câmara (*air-chillers*).

O sistema de resfriamento com ar é mais adequado do ponto de vista sanitário e microbiológico, devido a água que envolve as carcaças possui uma elevada carga microbiana, que pode levar a uma contaminação do produto. Porém, o processo de resfriamento com água é mais rápido, mais eficiente e mais viável economicamente. Segundo o manual da Sociedade Americana de Engenheiros de Aquecimento, Refrigeração e Ar Condicionado - ASHRAE, *chillers* de imersão são mais rápidos que *air-chillers*, além de impedirem a desidratação, pois durante a imersão ocorre absorção de água pelas carcaças (CARCIOFI, 2005).

Calcular os custos de operação de diferentes métodos de resfriamento não é uma tarefa fácil e reduzir a perda de peso pela evaporação é um fator crítico para o sucesso econômico do processo de resfriamento de frangos (JAMES et al., 2006).

Lillard (1982) em seu trabalho, considerou o processo de resfriamento por imersão, nos Estados Unidos, como sendo um método muito mais rentável. Pedersen (1979) também em seu trabalho, calculou os custos relativos de cinco diferentes tipos

de processo de resfriamentos aplicados na Dinamarca. Quando apenas o custo energético fora aplicado, o método de resfriamento por imersão teve custo 1/5 menor que o sistema de resfriamento por ar. Porém, quando os custos da água e de tratamento da mesma foram aplicados, o método por imersão acabou se tornando cinquenta vezes mais caro que o método por ar forçado. Muitas vezes esses parâmetros não são analisados quando se compara a eficiência entre esses dois métodos.

Em território europeu, o método de resfriamento por imersão acaba tendo algumas restrições impostas pela legislação vigente, o que acaba resultando em uma preferência no uso do método de resfriamento por ar forçado. Atualmente, o método mais adotado pelas empresas brasileiras e americanas é o resfriamento por imersão em água, que basicamente consiste na passagem da carcaça por tanques contínuos, preenchidos com gelo (CARCIOFI, 2005; CARCIOFI e LAURINDO, 2007).

3.3.2 Resfriamento em Tanques Contínuos

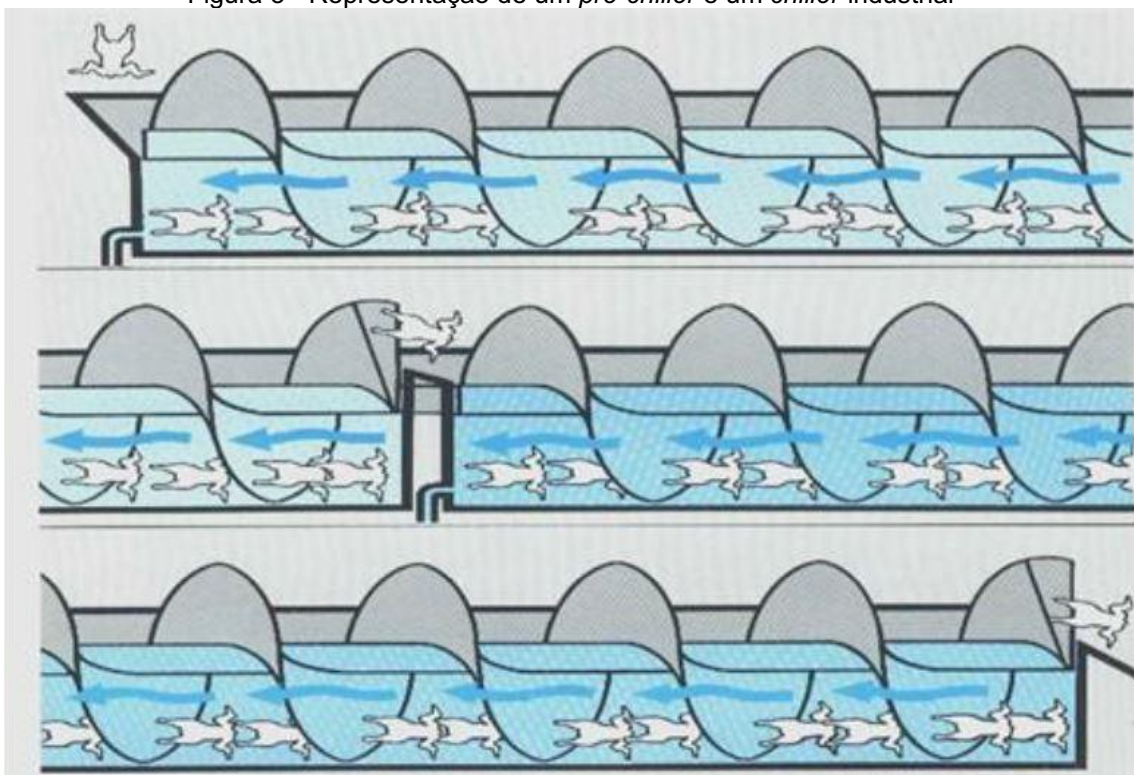
O resfriamento retarda o desenvolvimento microbiológico e diminui a degradação natural do músculo decorrente da atividade enzimática. Segundo Klassen (2008): “*A infinidade de produtos de carne disponíveis hoje no mercado, in natura e processados, o padrão de qualidade e os longos prazos de validade não existiriam sem os processos de resfriamento*”.

O resfriamento neste tipo de processo é rápido e eficiente, podendo ser aplicado numa ampla faixa de produtos. Porém, deve se atentar, pois um resfriamento muito rápido resulta no encurtamento das fibras musculares, que podem causar uma redução da maciez da carne, enquanto um resfriamento demasiadamente lento produz músculos pálidos, flácidos e exudativos (efeito *pale, soft and exudative* - PSE). Sabe-se também que quando bem operado, o sistema de resfriamento por imersão reduz a contagem total de bactérias nas carcaças de frango, tornando-se um método válido pelo ponto de vista higiênico, que é o principal ponto analisado quando comparado com o método de resfriamento a ar (TERUEL et al., 2003; KLASSEN, 2008; NEVES FILHO, 1978; BRESSAN e BERAQUET, 2004).

A carcaça leva em torno de 60 minutos para ter sua temperatura reduzida até o ponto desejado (em torno de 5 a 7°C). Durante o resfriamento, a maior temperatura na carcaça é encontrada no centro do peito do frango devido a sua menor taxa de resfriamento neste ponto (CARCIOFI e LAURINDO, 2007).

Quando a carcaça sai dos processos de evisceração, inspeção e ducha de lavação, ela passa por um disco circular giratório, onde então irá cair no tanque de pré-resfriamento que é constituído de roscas sem-fim e pás adaptadas para promover o movimento contínuo das carcaças, onde recebe um fluxo de água constante em sentido contracorrente da carcaça. O resfriamento por *chillers* contínuos envolve no mínimo (definido pela portaria nº 201 do Ministério da Agricultura e do Abastecimento) dois tanques construídos de aço inoxidável (para sanitização dos alimentos) que são chamados de: *pré-chillers* e *chillers* (OLIVO, 2006; MAPA, 1998).

Figura 3 - Representação de um *pré-chiller* e um *chiller* industrial



Fonte: MAPA (2014).

3.3.2.1 Pré-Chiller

É o primeiro tanque em que é destinada a carcaça, a qual se encontra em torno de 40°C. Este valor de temperatura é suficiente para estabelecer uma diferença de temperatura significativa com o meio refrigerante, pois o primeiro tanque possui a temperatura da água inferior a 16°C, conforme exigência da legislação. Esta configuração estabelece a força motriz necessária à transferência de calor (CARCIOFI, 2005; OLIVO, 2006).

O tamanho do *pré-chiller* varia de acordo com a disponibilidade de espaço na planta industrial, assim como a demanda de produto exigido. Basicamente, é constituído de um tanque semicircular de aço inox provido de um transportador helicoidal acoplado a um eixo central, onde o movimento das carcaças é feito em sentido contrário ao fluxo de água e tem sua agitação também promovida por injeção de ar comprimido por meio de orifícios acoplados na parte inferior e ao longo do tanque, a alimentação dos orifícios é feita por tubos plásticos acoplados a um injetor. Estes orifícios, ao permitirem a entrada do ar (que deve ser previamente tratado e seguir padrões da legislação vigente) promovem uma maior agitação da água. Isso ocasiona um aumento nas taxas de resfriamento e de absorção de água pelas carcaças. O borbulhamento também influencia na redução dos canais preferenciais no meio, e sua distribuição de ar deve ser homogênea por todo o comprimento dos *chillers*. (CARCIOFI, 2005; KLASSEN, 2008; NEVES FILHO, 1978).

A água que corre pelo tanque é previamente resfriada em um compartimento externo e bombeada até o ponto do tanque que corresponde à saída das carcaças, o fluxo é controlado por válvulas e após a passagem pelo tanque, a água é descartada (CARCIOFI, 2005; NEVES FILHO, 1978). Deve-se ter uma renovação mínima e constante de água neste equipamento. Segundo Klassen (2008), é necessária a adição de 1,5 L de água por carcaça durante o resfriamento no *pré-chiller*.

Para acelerar o processo de troca térmica entre o frango e a água, utilizam-se alguns artifícios como camisas com circulação de uma solução de propilenoglicol numa temperatura abaixo de 0°C, ou adiciona-se gelo em escamas em pontos dispersos no equipamento, esta que é a grande ferramenta para se alcançar o maior gradiente de temperatura nos tanques. O propilenoglicol, por ser um anticongelante barato, seguro e inerte para as tubulações, é preferido entre os demais. Ele é utilizado normalmente numa concentração de 30%, que permite a solução chegar a -20°C sem congelar (KLASSEN, 2008).

Além da redução da temperatura, esse estágio também promove a lavagem da carcaça, removendo assim o sangue restante, a microflora contaminante e outras matérias orgânicas que podem influenciar na qualidade final da carne. É também realizada a hidratação da carcaça, pois nesta temperatura a água penetra facilmente a pele e em outros tecidos subcutâneos. Esta hidratação é importante para reidratar a carne e proteger a mesma durante os processos subsequentes (congelamento, armazenamento e distribuição), evitando assim a sua desidratação superficial,

podendo causar alterações de cor e sabor do produto final (OLIVO, 2006; CARCIOFI, 2005).

O sistema de resfriadores contínuos por imersão deve portar de equipamentos de mensuração que proporcionem o controle e registro constante da temperatura da água nos tanques e o volume de água renovada no sistema. Ao final de cada período de trabalho (geralmente oito horas), os tanques refrigeradores devem ser completamente esvaziados, lavados e higienizados (BRASIL, 1998; SOUZA JÚNIOR, 2009).

A portaria nº 210 do MAPA estabelece o método de análise de gotejamento (*drip test*) como parâmetro para controle da absorção de água na carcaça. Se a quantidade resultante (em % de peso), com todas as partes comestíveis na embalagem, ultrapassar o valor limite de 6%, então é dito que a carcaça absorveu um nível excessivo de água no processo de pré-resfriamento (OLIVO, 2006).

3.3.2.2 *Chiller*

Após a etapa de pré-resfriamento, as carcaças seguem para o tanque de resfriamento (*chiller*) principal. Ambos são parecidos em estrutura, porém o *chiller* possui um comprimento maior, assim como proporciona um maior tempo de residência para as carcaças. A portaria nº 210 do MAPA estabelece que a temperatura da água residente nos tanques *chiller* não devem ser superiores a 4°C, sendo utilizados revestimentos de etilenoglicol, amônia ou outros componentes no tanque, visando o controle desta temperatura (BRASIL, 1998; OLIVO, 2006).

Bishop (1972) estudou diferentes tipos de resfriadores por imersão e indicou adicionar-se 1 kg de gelo para cada kg de carcaça, necessitando o resfriamento de 35 a 40 minutos. Já o tempo de permanência das carcaças no primeiro tanque (*pré-chiller*), não deve ultrapassar 30 minutos. A temperatura das carcaças deve ser igual ou inferior a 7°C no final da etapa de pré-resfriamento, tolerando-se temperatura de até 10°C, desde que as carcaças sejam encaminhadas imediatamente ao congelamento. Visando garantir a qualidade das carcaças de frango, a portaria permite que a água de renovação do sistema de pré-resfriamento por imersão seja hiperclorada para combater a carga microbiana presente na mesma, podendo-se utilizar no máximo 5 ppm (partes por milhão) de cloro livre. A renovação de água no processo de resfriamento deve ser constante, sendo que a renovação no último

tanque deve ser de 1,0 L por carcaça para aquelas com peso até 2,5 kg, 1,5 L para carcaças com peso entre 2,5 kg a 5 kg ou ainda, 2,0 L de água para carcaças com peso superior a 5 kg. Utiliza-se para a renovação da água (que está com temperatura próxima de 1°C) trocadores de calor, normalmente de placas, para poderem resfriar a mesma (KLASSEN, 2008; BRASIL, 1998).

A água e a carcaça correrão em sentido contracorrente no tanque para ocorrer um maior contato físico, permitindo maior resfriamento e limpeza das carcaças. O tempo de gotejamento das carcaças desde a saída do processo de resfriamento até serem embaladas ou processadas, influencia na quantidade de água retida por elas. Quanto mais tempo ficarem gotejando, mais água perdem. O que determina este tempo é a distância a ser percorrida na nória, ou seja, a distância entre as diversas seções de processamento (KLASSEN, 2008; OLIVO, 2006). Na Figura 4 está representado o maquinário envolvido na etapa do processo de resfriamento em um abatedouro de frango:

Figura 4 - Tanques de resfriamento em abatedouro



Fonte: Marel (2018).

3.3.3 Variáveis Influenciadoras nos Efeitos de Resfriamento e Absorção

Como já discutido anteriormente, calcular o fenômeno de transferência de energia e de massa em um volume de controle sem forma específica não é uma tarefa

fácil. Todavia alguns autores conseguiram contornar este problema, porém, segundo Klassen (2008), as previsões que se consegue obter são aproximadas e muitas vezes baseadas em ajustes por correlação de dados empíricos, e com muitas idealizações.

3.3.3.1 Resfriamento da Carcaça

James et al. (2005) ao citar Veerkamp and Hofman (1974) em seu trabalho, relata que os mesmos conseguiram produzir equações para a taxa de remoção de calor para um sistema de imersão de alta e média mistura e também para um sistema de refrigeração a ar de alta velocidade.

Neves Filho (1978) resolveu um modelo bastante simplificado da temperatura após o resfriamento, utilizando de muitas idealizações e eliminando do modelo parâmetros importantes como borbulhamento e adição de gelo. Seus resultados dentro destas idealizações foram precisos, porém como dito anteriormente, seu modelo levou em conta apenas alguns parâmetros presentes no processo, como o tempo de imersão, temperatura da água, espessura do músculo e características dos materiais.

Wang e Sun (2001) avaliaram a performance do resfriamento de carne cozida por um sistema de refrigeração com ar lento, com ar em alta velocidade e por imersão em água, seus cálculos foram feitos utilizando o método dos elementos finitos. Para diferentes geometrias de produtos (ainda que idealizados e não condizentes perfeitamente com a geometria atual), foram determinadas as condições ideais de operação e qual dos métodos seria o mais adequado.

3.3.3.2 Absorção de Água da Carcaça

Brant (1963) indicou em seu trabalho que a temperatura da água de resfriamento parece apresentar um efeito menor que a agitação da água, dimensão da ave, temperatura de escaldagem e de pré-resfriamento.

Andersen (1974) discutiu diferentes aspectos no resfriamento por imersão onde a absorção da água pela carcaça é influenciada por fatores como: tempo de imersão, agitação das carcaças, temperatura da água, assim como cortes ou incisões ocasionadas durante a etapa de evisceração.

De acordo com a legislação nacional vigente, os mais importantes fatores que afetam o índice de absorção de água em carcaças durante a etapa de resfriamento são a temperatura da água nos resfriadores (ambos *chiller* e *pré-chiller*), o tempo de permanência das carcaças nos resfriadores, o tipo de corte abdominal, o borbulhamento no sistema e outros fatores menos significativos, como velocidade no abate e temperatura ambiente (BRASIL, 1998).

Young e Smith (2004) relataram em seu trabalho que a absorção relativa de água pelas carcaças de menor porte foram maiores que a absorção de água pelas carcaças maiores, e que o tempo de imersão da carcaça é a variável mais importante relacionada com a absorção de água.

James et al. (2005) relacionou a influência do tempo de imersão, temperatura da água e quantidade de gelo sobre diversos parâmetros do resfriamento das carcaças, entre eles sendo a temperatura final e a quantidade de água absorvida pelas mesmas.

Obdam (2005) descreveu outros fatores relevantes que podem variar a absorção de água pelas carcaças, dentre eles também atentou para o tamanho ou peso das aves (quanto menor a ave, maior é a porcentagem de absorção pela mesma), o próprio processo de abate que pode deixar o comprimento de pele do pescoço menor, que irá ocasionar numa facilidade para a entrada de água adicional na carcaça e o comprimento da linha de gotejamento antes das etapas de embalagem ou corte das partes que. Se o comprimento for muito curto, pode-se reter mais água nas cavidades da carcaça.

Carciofi e Laurindo (2007) afirmaram que os parâmetros determinantes na quantidade de absorção pelas carcaças de aves durante o processo de resfriamento por imersão são a temperatura da água, pressão hidrostática, agitação do meio e o tempo de imersão, sendo que o tempo de permanência das carcaças nos *chillers* é pouco variado industrialmente e tem um valor próximo a 60 minutos. Estes autores, em especial, propuseram um modelo empírico para descrever a variação de massa das carcaças. Eles consideraram as características do material e o tempo de imersão, descritos pela Equação 1:

$$\frac{dW}{dt} = K_0 e^{\beta W} (W_m - W) \quad (1)$$

onde “W” é a fração de água na carcaça [Kg/Kg], “ W_m ” é a fração máxima de água na carcaça [Kg/Kg], “ K_0 ” é uma constante experimental [min^{-1}], “ β ” é outra constante experimental [Kg/Kg] e “t” é o tempo de imersão [min]. Os erros encontrados pelos autores na aplicação deste modelo para dados experimentais ficaram entre 5 e 8%.

Klassen (2008) ao citar Dyer (2007) em seu trabalho descrevem: “ *ser de grande importância a temperatura e o tempo de escaldagem na capacidade de retenção de água das carcaças. Como os tanques de escaldagem possuem um tamanho fixo o tempo passa a ser função apenas da velocidade da linha. Os autores também comentam: “sobre os efeitos do borbulhamento, tanto buscando aumentar o coeficiente de transferência de calor do frango para a água, como também influenciando na retenção de umidade.”* Ainda para Klassen (2008), muitos dos fenômenos presentes teriam que ser desconsiderados e muitas idealizações assumidas para se obter um modelo, que de qualquer maneira seria extremamente complexo, de solução muito difícil, e de resposta duvidosa em função das idealizações.

3.3.4 Controle de Qualidade Industrial no Processo de Resfriamento

O sistema de controle da absorção de água em carcaças de aves submetidas ao resfriamento por imersão deve ser eficaz e seguro, sem afetar a qualidade do produto final. Os métodos oficiais para este controle são: o Método de Controle Interno, realizado durante o próprio processo industrial pela Inspeção Federal local e pela própria indústria. O outro método é o Método do Gotejamento, para controle de absorção de água em carcaças congeladas recém submetidas ao resfriamento (CARCIOFI, 2005).

Para o Método de Controle Interno, a diferença positiva obtida na massa das carcaças entre a saída e a entrada do resfriador é calculada e indicada de maneira percentual, sobre o valor da massa na entrada. Segundo o MAPA, a quantidade de água (em percentagem da massa total) da carcaça deve-se ter o limite máximo de 8% da massa total. O Método do Gotejamento é utilizado para determinar a quantidade de água resultante do descongelamento das carcaças resfriadas. Se a quantidade de água resultante (em percentagem da massa da carcaça, com todos(as) os(as) miúdos/partes comestíveis na embalagem) ultrapassar o valor limite de 6%,

considera-se que a carcaça absorveu um excesso de água durante o resfriamento (CARCIOFI, 2005; BRASIL, 1998).

3.3.5 Uso dos Dados no Processo Industrial

No ambiente industrial muitas são as dúvidas com relação a como o ganho de massa pelas carcaças ocorre e quais são os fatores que mais o influenciam. Até os dias de hoje, os parâmetros observados são, em geral, influências decorrentes da temperatura da água, da agitação do meio e das próprias características da carcaça. Diferentemente da transferência de calor, onde o interesse é voltado para o controle da temperatura do centro do músculo peitoral da carcaça, na transferência de massa, busca-se controlar a quantidade total de água absorvida, não importando a maneira como ela se distribui internamente nas carcaças (CARCIOFI, 2005).

Até os tempos atuais, não foram encontrados trabalhos na literatura tratando da dinâmica de absorção de água pelas carcaças durante o resfriamento por imersão, que retratem fielmente como a mesma ocorre. Segundo Neves Filho (1978), como a carcaça de frango apresenta uma forma irregular, considera-se que as transferências de calor (consequentemente de massa) ocorrem perpendicularmente a uma placa infinita cuja espessura seria a do peito da carcaça. Essa aproximação obtida pelo autor têm apoio de trabalhos realizados anteriormente ao mesmo, como Andersen (1974), Loguinov e Civatseva (1973 e 1974), Vacineck e Toledo (1973).

3.4 INTELIGÊNCIA ARTIFICIAL E APRENDIZADO DE MÁQUINA

3.4.1 Inteligência Artificial (IA)

A IA é um ramo da ciência da computação que se propõe em desenvolver sistemas capazes de simular a capacidade humana de percepção de problemas, identificando seus componentes e, com base nisso, resolver este problema tomando/propondo diferentes decisões (LOBO, 2018). Segundo Gomes (2010): *“O homem tem a capacidade única de raciocínio e, durante milhares de anos, ele procurou entender como o pensamos: isto é, como um mero punhado de matéria pôde compreender, perceber, prever e manipular um mundo muito maior e muito mais*

complexo que ele próprio. O campo da inteligência artificial vai ainda mais além: ele tenta não apenas compreender, mas também construir entidades inteligentes”.

O termo “Inteligência Artificial” foi constituído em 1955 por John McCarthy, um professor de matemática da Faculdade de Dartmouth, nos Estados Unidos. O mesmo posteriormente organizou uma conferência seminal do assunto no ano seguinte (BRYNJOLFSSON e MCAFEE, 2017). Segundo Fernandes (2019), a inteligência artificial pode ser definida pelo “estudo de agentes que recebem percepções do ambiente e executam ações”. Estes agentes executam seus respectivos comandos que lhes foram atribuídos, de maneira a se obter o melhor resultado desejado. O campo da inteligência artificial discute, por exemplo, a capacidade de agentes físicos (máquinas) ou lógicos (programas de computador) tomarem decisões com base em dados captados ou alimentados por meio de intervenção humana.

Ao passar dos anos, o ramo da IA têm se desenvolvido drasticamente devido à melhora da capacidade de processamento dos computadores e do crescimento/acúmulo de banco de dados (conhecidos como *big data*). Entretanto, a capacidade de utilização das tecnologias envolvendo IA tem se mantido limitada em determinadas áreas de pesquisa, como por exemplo reconhecimento de imagens, reconhecimento de voz e respostas de comandos. Essa limitação se deve pelo fato da IA estar, até o momento, aquém da capacidade de agir similarmente às mínimas funções do cérebro humano, não tendo a capacidade de substituir o mesmo, principalmente em funções como motivação, consciência e controle próprios (LU et al., 2018).

3.4.2 Aprendizado de Máquina (AM)

O aprendizado de máquina (AM), do inglês: *machine learning*, pode ser resumido como a alteração do programa de controle, baseando-se nas suas próprias experiências realizadas através de algoritmos próprios (ARTERO, 2009). A coisa mais importante para se compreender sobre o aprendizado de máquina é que ele representa um processo diferente de se criar um programa (ou *software*), a máquina irá aprender como agir através de um exemplo (AM) ao invés de ser explicitamente programada para resolver os mesmos (criação de um *software*) (BRYNJOLFSSON e MCAFEE, 2017).

O AM é utilizado para ensinar máquinas a como lidar com dados mais eficientemente. Haverá momentos em que, após se analisar algum tipo de banco de dados, nós humanos não conseguiremos interpretar nenhum padrão ou extrair nenhuma informação do mesmo, neste caso é utilizado o AM. Muitas indústrias, desde a médica, até a militar, vêm utilizando o AM para extrair informações relevantes de diferentes bancos de dados e aprender com os mesmos (DEY, 2016).

O AM abrange um campo de pesquisa que foca na teoria, na performance e nas propriedades de sistemas de aprendizado e algoritmos. É um campo altamente interdisciplinar que inclui ideias de IA, teorias de otimização de processos, teoria da informação, estatística, ciência cognitiva, controle de processo, engenharia, matemática e outros campos. Devido ao seu alto alcance de aplicações, o AM tem sido imposto em quase todo o território científico, no qual tem trazido grande impacto. Algumas de suas recentes aplicações foram melhoramento de motores, sistemas de reconhecimento, informática, mineração de dados, juntamente com o controle de sistemas autônomos (QIU et al., 2016).

3.4.3 Métodos de Aprendizado de Máquina

Existem diferentes tipos de métodos relacionados ao aprendizado de máquina, cada um com seu propósito específico. Esses métodos estão retratados na Figura 5:

Figura 5 - Vertentes do aprendizado de máquina



Fonte: Autoria própria (2020).

Dentre todas as vertentes que o aprendizado de máquina inclui, as mais conhecidas e mais comumente utilizadas são o aprendizado supervisionado e o aprendizado não-supervisionado. Dey (2016) e Ge et al. (2017) em seus trabalhos explicam resumidamente como são estes dois tipos de aprendizado:

- **Aprendizado Supervisionado**

Os algoritmos de AM relacionados ao aprendizado supervisionado são aqueles que necessitam de assistência externa, isso é, os comandos são incluídos manualmente no algoritmo através de ajuda humana. Estes algoritmos lidam com amostras de dados previamente rotulados, discretos ou contínuos. O conjunto de dados de entrada são divididos entre conjunto de treinamento e conjunto de teste, este último tendo como função prever ou classificar uma variável que é representada como a variável de saída (que é o que se deseja analisar). Todo o algoritmo irá aprender algum tipo de padrão do conjunto de dados de treinamento e aplicar este padrão no restante do conjunto de dados que servirá para a predição ou classificação.

- Aprendizado Não-Supervisionado

Os algoritmos relacionados ao aprendizado não-supervisionado não necessitam de ajuda externa através do implante de códigos manuais e adquirem pouca informação dos dados apresentados, sendo este método mais utilizado quando se trabalha com dados dos quais não existem rotulações (não são previamente separados em grupos). O principal foco deste tipo de aprendizado é explorar os dados e encontrar estruturas escondidas entre eles, quando um novo conjunto de dados é implantado, o algoritmo utiliza informações previamente obtidas para reconhecer a classe de cada dado e possivelmente realizar a tarefa de predição ou classificação.

3.4.4 Vantagens ao se utilizar o Aprendizado de Máquina

Os algoritmos de AM são utilizados por serem poderosos e com alta capacidade de predições. Devido ao fato destes algoritmos possuírem uma capacidade de aumentar sua eficácia com maiores quantidades de dados a serem treinados, a crescente disponibilidade de tais dados recentemente têm trazido interesse renovado neste tipo de trabalho com algoritmos de AM (BURRELL, 2016).

Segundo Brynjolfsson e McAfee (2017), o porquê do AM ser uma área em crescimento se deve por dois fatores: O primeiro deles é pelo fato de nós humanos sabermos mais do que podemos dizer, ou seja, não conseguimos explicar exatamente como nós somos capazes de realizar muitas coisas (como por exemplo reconhecer um rosto ou fazermos jogadas inteligentes em jogos como Xadrez), anteriormente ao AM, esta inabilidade de explorar nosso conhecimento significava que não podíamos automatizar muitas tarefas, porém agora com o AM é possível. O segundo fator é devido aos algoritmos de AM serem excelentes na arte de aprender. Eles podem obter uma performance além da capacidade humana em uma larga escala de atividades, incluindo detecção de fraudes ou diagnóstico de doenças.

3.4.5 Desvantagens ao se utilizar o Aprendizado de Máquina

Uma presunção comum ao se trabalhar com AM é que seus algoritmos podem aprender melhor quando se trabalham com mais dados e conseqüentemente podem prover resultados mais acurados. Porém, conjuntos de dados massivos (*Big Data*)

impõem uma variedade de desafios, pois estes algoritmos mais comuns não são designados a se trabalhar com tais informações. Muitos algoritmos são designados a trabalhar com conjunto de dados menores, assumindo que eles possam caber em sua memória, ou então para que seja possível atingir o tempo de processamento necessário para o treinamento. O *Big Data* quebra estes paradigmas devido ao seu tamanho, fazendo com que algoritmos comuns sejam inutilizados ou então reduzam drasticamente sua performance. Uma válvula de escape para este problema está sendo trabalhada recentemente com técnicas (como o aprendizado por reforço ou o aprendizado profundo, também conhecido como *Deep Learning*) para fazerem com que algoritmos de AM possam trabalhar envolvendo os *Big Data* (L'HEUREUX et al., 2017).

Outro fator para se atentar ao se trabalhar com AM, é que a explicação e a interpretação física dos resultados são frequentemente tarefas difíceis, pois muitos modelos não são interpretáveis ou possuem uma baixa interpretabilidade, significando que os humanos encontram uma dificuldade decifrando como os sistemas chegaram às suas decisões. Um conceito que pode ser relacionado é a justificção: ela explica por que a decisão é boa ou ruim, porém pode não explicar o porquê de como essa decisão foi feita. Ao contrário da explicação, a justificção pode ser produzida para sistemas não-interpretáveis, porém a explicação tem se mostrado importante para a aceitação e satisfação em diversos estudos (BIRAN e COTTON, 2017).

Devido a estes fatores apresentados anteriormente, três tipos de riscos majoritários podem ser enfrentados ao se trabalhar com AM:

O primeiro é que as máquinas podem ter vieses ocultos que não são derivados da intenção humana ao se trabalhar com o algoritmo, mas sim da própria máquina ao analisar os dados fornecidos para treinar o sistema.

O segundo é que ao contrário de sistemas tradicionais construídos em lógicas explícitas, podem existir algoritmos, como as redes neurais, que lidam com verdades estatísticas ao invés de fatos reais (ou verdades literais).

O terceiro é que sistemas de AM estão suscetíveis a cometer erros e inevitavelmente irão, e, quando acontecer, diagnosticar e corrigir exatamente o ponto de falha pode se tornar uma tarefa difícil de ser realizada (BRYNJOLFSSON e MCAFEE, 2017).

3.4.6 Cooperação entre Seres Humanos e Máquinas

Como discutido anteriormente, embora estes modelos consigam acurácias de predições incríveis, sua estrutura não linear os faz altamente não transparentes, não expondo claramente qual parâmetro no conjunto de dados de entrada os faz chegar em suas decisões, assim é dificultosa a tarefa de se analisar estes algoritmos e cabem aos humanos realizá-la.

Sistemas de AM dificilmente irão reproduzir toda a tarefa, processo ou modelo de negócio em uma empresa, mas frequentemente eles irão complementar as atividades humanas, tornando seu trabalho mais valioso. Supondo-se que para a realização de uma atividade fossem necessárias 10 etapas, apenas uma ou duas poderiam ser automatizadas, enquanto o resto seria mais proveitosa se feita por humanos. A regra mais efetiva para uma divisão de funções é não distribuir toda a função para a máquina e sim aliá-la à capacidade humana de utilizá-la (SAMEK et al., 2017; BRYNJOLFSSON e MCAFEE, 2017).

3.4.7 O Uso de Aprendizado de Máquina no Cenário Industrial e o Crescente uso de Softwares

O AM tem se tornado uma tecnologia emergente em diversos campos e tem uso tanto comercial quanto não-comercial. Em particular, o rápido crescimento das aplicações do “*Deep Learning*” (traduzindo: aprendizado profundo, método de busca intensa em bancos de dados) têm aumentado o desenvolvimento do AM. Conseqüentemente, o “*Deep Learning*” tem se tornado um tópico de pesquisa prioritário para organizações e empresas tecnológicas (SIANAKI et al., 2019).

Nas últimas décadas, um grande acúmulo de dados tem sido atingido no processo industrial devido ao alto uso de sistema de controle de processos. Enquanto atualmente se torna cada vez mais dificultoso realizar cálculos manuais em processo mais complexos, a modelagem, o monitoramento, a previsão e o controle de processos orientados a dados têm recebido muita atenção nos últimos anos e possui um importante papel em construir modelos de dados de processo, baseados em quais informações tem sido extraídas, identificando novos padrões nos dados e realizando as predições mais facilmente para novos conjuntos de dados, assim como suas decisões podem ser feitas mais rapidamente e efetivamente.

Através da análise de padrões nos dados de processo e sua relação entre as variáveis, informações úteis podem ser extraídas baseadas em quais modelos estatísticos podem então ser desenvolvidos dependendo de sua aplicação, sendo entre essas aplicações utilizadas atualmente, monitoramento de processo, diagnóstico de falhas, clusterização, detecção de variáveis principais ou de variáveis qualitativas, entre outras. Então tem-se que a principal meta da mineração destes dados e sua respectiva análise é extrair informação útil dos dados de processo e transferir a mesma para melhoramento e entendimento, assim como suporte para tomada de decisões no processo.

Lançar tecnologias digitais em processos de fabricação está originando a quarta revolução industrial, comumente chamada de indústria 4.0. Esta é baseada no extensivo uso do CPS (Sistema Cyber-Físico) e foca em melhorar significativamente a produtividade de fabricações no século 21, através da coleção e análise de dados em tempo real (BRIK et al., 2019).

As duas técnicas mais largamente utilizadas industrialmente, constituindo 80 a 90% das aplicações industriais, são ambas as técnicas de aprendizado supervisionado e não-supervisionado. Um típico exemplo de uma técnica de regressão (aprendizado supervisionado) industrial é a predição da performance de processo, como por exemplo seu tempo de realização, no qual o conjunto de dados de entrada possuem variáveis de processo gravadas rotineiramente, sendo entre elas temperatura e pressão (GE et al., 2017).

Atualmente, muitas ferramentas e softwares de mineração de dados e AM estão disponíveis para uso, como o RapidMiner, KNIME, R-Programming, Orange, NLTK, entre outros. Estes *softwares* provêm uma gama de métodos e algoritmos que procuram auxiliar numa melhor análise dos dados. Suas ferramentas possuem função de clusterização, visualização de dados, análise regressiva, árvores de decisão, análise preditiva, mineração de textos, entre outros. Um dos *softwares* mais difundidos em pesquisas atualmente é o *software WEKA*, que será discutido no tópico 5.7 do presente trabalho (NAIK e SAMANT, 2016).

3.5 CLASSIFICADORES BASEADOS EM INSTÂNCIAS E MODELO “*K NEAREST NEIGHBOUR*” (KNN)

Em métodos de aprendizado baseado em instâncias (do inglês: *Instance Based Learning* – IBL), o trabalho é realizado quando chega a hora de classificar uma nova instância, e não quando o conjunto de treinamento é processado. Logo, a diferença entre esse método e os outros é o tempo no qual o “aprendizado” ocorre. Estes métodos são considerados “preguiçosos” (por isso chamados de “*lazy learning*” - aprendizado preguiçoso), adiando o seu trabalho o máximo possível, enquanto outros métodos procuram uma resolução mais rápida, produzindo uma generalização assim que os dados são vistos. Neste método de aprendizado, é encontrada uma instância de treino mais próxima (podendo ou não ser em distâncias euclidianas) da instância de teste provida para classificação. Realizada a predição de sua classe através desta distância, se muitas instâncias de treino estiverem próximas, a primeira instância encontrada é usada para a classificação. Esse método pode ser chamado de método de classificação do vizinho mais próximo. Às vezes, mais de um vizinho pode ser usado e a classe majoritária dos vizinhos mais próximos (ou a média ponderada da distância, se a classe é numérica) é atribuída à nova instância. Esse método constitui a base do modelo de vizinho mais próximo (ou KNN – *K Nearest Neighbour*).

Os modelos baseados em instâncias armazenam as instâncias de treino em tabelas e fazem interpolações através destas. Este aprendizado é apto a ocorrer rapidamente de um conjunto de dados muito pequeno, enquanto outros métodos requerem uma representação razoável de cada sentença antes de serem introduzidas. O modelo baseado em instâncias consegue fazer predições úteis com um exemplo de instância por classe, sua performance de classificação frequentemente excede 75% do máximo possível após aceitar apenas 25% de um conjunto de dados completo.

Os algoritmos de aprendizado baseados em instâncias não constroem descrições conceituais para suas instâncias treinadas (apenas para as instâncias já pré-existentes). Porém, este pode ser interpretado através da similaridade e da determinada função de classificação utilizada pelo algoritmo para definir a nova classe da instância (VIJAYARANI et al., 2013; WITTEN et al., 2011; HERNÁNDEZ, 2015).

Segundo Cleary (1995), existem três componentes que descrevem todos os algoritmos de aprendizado baseados em instâncias, são eles:

- Funções de Similaridade: Calculam uma similaridade entre as instâncias de treino “a” e as instâncias ao seu redor; essas funções são dadas em valores numéricos.
- Função de Classificação: Esta por sua vez obtém os resultados das funções de similaridade e os registros de desempenho de classificação das instâncias e então geram uma classificação para a instância “a”.
- Atualizador de Descrição Conceitual: Este em questão obtém dados da performance de classificação e decide quais instâncias incluir na base de dados e quais instâncias da nova base de dados devem ser usados para uma nova classificação.

Para algoritmos mais simples, após uma instância ser classificada, ela sempre é levada para a base de dados das instâncias, juntamente com sua correta classificação. Para algoritmos mais complexos, pode haver uma filtragem de quais instâncias serão adicionadas à base de dados para reduzir os requisitos computacionais para o armazenamento e melhorar quanto sua tolerância para dados que apresentarem muitos ruídos (dados inconsistentes).

3.5.1 Vantagens e Desvantagens de Métodos de Aprendizados Baseados em Instâncias

A principal vantagem adquirida ao empregar um método de aprendizado baseado em instâncias é que a função alvo será aproximada localmente assim como o algoritmo KNN, pois sua função objetivo é aproximada localmente para cada consulta ao sistema. Este aprendizado pode concorrentemente resolver problemas múltiplos e lidar sucessivamente com mudanças na área do problema.

As desvantagens ao empregar este método incluem o alto requerimento computacional para armazenar o conjunto de dados completo. A maioria dos dados ruidosos aumentam a necessidade de armazenamento desnecessariamente e

também são frequentemente mais vagarosos para avaliações que outros algoritmos similares, embora atualmente haja filtros para aumentar sua rapidez (VIJAYARANI et al., 2013).

3.5.2 Algoritmo *K-Star* (K^*)

O algoritmo *K-Star* (K^*) é um algoritmo de aprendizado baseado em instâncias similar ao *K-Nearest Neighbour* (KNN), onde novos dados de instâncias “x” são designados às classes que ocorrem mais frequentemente em torno dos pontos mais próximos (chamados pontos “k”). O principal lema do algoritmo K^* , e outros métodos de aprendizados baseados em instâncias, é que instâncias similares terão classes similares. O algoritmo K^* utiliza medidas entrópicas, tendo em vista a probabilidade de mudar uma ocasião para outra, escolhendo aleatoriamente entre todas as mudanças imagináveis. Utilizar essa distância entrópica pode prover um certo número de benefícios para a função utilizada, como a capacidade de lidar com valores de atributos reais e valores desconhecidos, ela também ajuda a entender a probabilidade que um ponto “a” irá chegar em “b” via um trajeto randômico no espaço em questão. A classificação com o K^* é feita juntando-se as probabilidades de comparação de uma nova instância para todos os membros de uma categoria. Isto deve ser feito com todas as categorias adjacentes para finalmente selecionar qual destas terá a maior probabilidade de classificação. A complexidade de uma mudança de um caso para outro é calculada através de duas etapas. Primeiro, caracteriza-se um arranjo limitado de mudanças que pode ocorrer entre uma instância em outra. Nesse ponto, altera-se um exemplo (a) para (b) com a assistência do programa em um agrupamento limitado de alterações começando em (a) e terminando em (b), levando em conta intermináveis tipos de caminhos e conjunto de mudanças predefinidas. Sua otimização será realizada sobre o percentual do parâmetro de proporção de mistura (*blending ratio*) (VIJAYARANI et al., 2013; HERNÁNDEZ, 2015; KUMAR e HIWARKAR, 2019).

3.5.3 Vantagens e Desvantagens do K^*

As vantagens de uso do K^* é que ele provê uma aproximação consistente de trabalho com valores reais, simbólicos e de lidar com valores ocultos. A distância entrópica auxilia a obter então as instâncias mais similares possíveis. As suas

limitações estão basicamente no seu longo tempo de treinamento e na sua dificuldade de entender seus pesos e suas funções de proximidade que são altamente probabilísticas (NEELAVATHI e JAGATHEESAN, 2016).

Muitos autores têm utilizado o K^* para diferentes problemas de classificação, como por exemplo Madhusudana et al. (2016), que o utilizaram para estudar características do histograma do sinal de vibração de ferramentas de corte. Madera e Varona (2016) o utilizaram para comparação de eficiência com outros algoritmos conhecidos de mineração de dados, e, Devasena (2017) o utilizou para classificação de dados multivariados de um repositório. Alguns resultados têm se mostrado promissores, embora ainda haja uma grande heterogeneidade envolvendo este algoritmo e sua performance. Algumas vezes o resultado pode não ser relevante, outrora pode ser o melhor algoritmo testado, ou então o pior. Estas diferenças podem se dar devido à composição heterogênea dos conjuntos de dados testados nos estudos em questão (HERNÁNDEZ, 2015). Para o processo de abate industrial de frango, não foram encontrados pelo autor trabalhos publicados utilizando o algoritmo K^* para tarefas de predição e classificação de variáveis.

3.6 MODELO DE MÁQUINA DE VETOR DE SUPORTE (SVM)

3.6.1 Definição

Os Vetores de Suporte de Máquina (do inglês: *Support Vector Machine* - SVM), é um método de classificação introduzido em 1992 por Boser, Guyon e Vapnik. É uma ferramenta computacional passível de resolver problemas de reconhecimento e regressão de padrões. Ela possui capacidade de fornecer excelente desempenho de generalização, além de demonstrar também que é eficiente para várias aplicações do mundo real. Recentemente os SVM para regressão têm sido aplicados em diversos campos de estudo, como controle ótimo de processos (SYARIF et al., 2016), predição de parâmetros (DENG et al., 2020) e análises de regressão de intervalos (QUINTERO et al. 2019). Até o momento, não foram encontrados trabalhos que utilizam o uso de SVM para processo de abate de frango. Para se obter a melhor performance do algoritmo de SVM, alguns parâmetros (chamados de Hiperparâmetros) devem ser sintonizados cuidadosamente. São estes parâmetros os relacionados com a função Kernel escolhida, o valor de perda da função (épsilon) e o valor de complexidade C .

A solução dos parâmetros do modelo correspondem a um problema de otimização convexo: os SVM utilizam um subconjunto mínimo dos dados de treinamento (chamados de vetores de suporte) para definir a melhor superfície de decisão entre duas classes (CROWLEY, 2016; BEN-HUR e WESTON, 2010; SHEVADE et al., 2000; ABAKAR e YU, 2014).

Os SVM foram os primeiros algoritmos baseados em Kernel. Eles utilizam uma função Kernel para transformar o espaço dos dados de entrada em um espaço de recurso de alta dimensão no qual procura por um hiperplano de separação. As funções Linear, Polinomial, RBF, entre outras, são exemplos das muitas funções que são comumente utilizadas para se transformar o espaço de entrada em um outro espaço desejado. Os SVM são baseados no princípio de “Estrutura de Minimização de Riscos” levando em conta a probabilidade de classificar incorretamente os padrões ainda a serem vistos para uma distribuição de probabilidade fixa, mas desconhecida dos dados. Ele usa um hiperplano separador linear para criar um classificador. Ainda assim, não é fácil separar linearmente alguns problemas no espaço original, mas pode facilmente transformar não-linearmente o espaço original em um espaço de alta dimensão, onde é essencial encontrar um hiperplano linear de separação ótimo (BAITHARU et al., 2016).

Adendo a esse algoritmo está o conceito de programação quadrática. Ele é um tipo especial de otimização de problemas matemáticos, reduzindo ao mínimo ou maximizando uma função quadrática de múltiplas variáveis sujeita a restrições lineares em tais variáveis. Um dos métodos conhecidos deste tipo de programação, é o método dos multiplicadores de Lagrange, o qual permite otimizar um problema de múltiplas variáveis através de determinadas condições (QUINTERO et al., 2019).

Alguns algoritmos de programação quadrática tradicionais não são adequados para problemas extensos devido a alguns problemas de complexidade. Tentativas foram feitas para desenvolver métodos que superem alguns ou todos esses problemas, um desses métodos é o *chunking*. Seu princípio é operar em um subconjunto de tamanho fixo do conjunto de treinamento por vez. Esse subconjunto é chamado de “conjunto de trabalho” e o subproblema de otimização é resolvido com relação às variáveis correspondentes aos exemplos no conjunto de trabalho, encontrando um conjunto de vetores de suporte para o conjunto de trabalho atual (SHEVADE et al., 2000).

3.6.2 Treinamento de um SVM

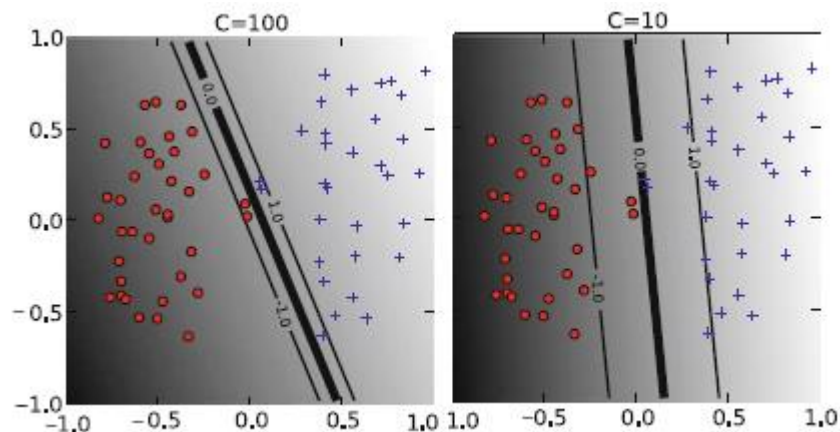
Treinar um SVM se resume em encontrar a maior margem do hiperplano para separar os dados, fazendo isto através da sintonização dos parâmetros. Dentre todos os presentes, o mais importante e utilizado é o parâmetro de complexidade C .

3.6.2.1 Parâmetro C (*Cost*)

Este parâmetro C determina entre: criar uma margem suave no hiperplano, permitindo que mínimos erros de treinamento aconteçam (numa escala controlada), ou forçar margens rígidas para permitir flexibilidade na separação das classes, diminuindo assim os erros.

Aumentar-se o valor de C significa aumentar o custo do erro da classificação (visa diminuir o erro, diminuindo-se a margem), então os dois pontos mais próximos ao hiperplano interferem em sua orientação e o hiperplano aproxima-se de vários outros pontos flexibilizando-se. Se diminuirmos o valor de C aqueles pontos tornam-se margens de erro; o hiperplano muda de orientação, e provê uma margem maior sendo propenso a mais erros de classificação.

Figura 6 - Efeitos do parâmetro C



Fonte: Ben-Hur e Weston (2010).

Os parâmetros do Kernel possuem um efeito importante também no limite de decisão. O grau da função Polinomial Kernel e o parâmetro de largura no Kernel de comportamento Gaussiano, controlam a flexibilidade do classificador resultante.

Uma das dúvidas mais frequentes ao se utilizar esse algoritmo é: Qual função utilizar com os determinados dados? Não existe uma resposta certa. A primeira orientação é utilizar o máximo de funções Kernel possíveis para ver qual se ajusta melhor com os dados em questão. O “procedimento padrão” para se testar as funções Kernel seria aplicar uma função Linear primeiro, e então ver se é possível otimizar a performance do algoritmo com uma função Kernel não-linear. A função Kernel Linear provê uma base útil para saber se os dados são facilmente ajustáveis, pois este tipo de função apresenta apenas um parâmetro para ser sintonizado (a constante de margem de erro). A partir do momento que o resultado usando a Kernel Linear está pronto, ele serve como base para saber se é necessário ou não realizar uma otimização com um Kernel não-linear (Kernel Polinomiais ou que utilizam Gaussianas) (BEN-HUR e WESTON, 2010).

Os SVM são capazes de aplicar técnicas de classificação e regressão a grandes conjuntos de dados, tendo em conta as variáveis inerentes ao modelo com que se busca um hiperplano ótimo para a separação das classes. Segundo Quintero et al. (2019), neste processo se incluem:

- Aplicação de um mecanismo indutivo que busca a minimização do risco;
- Utilização de separadores lineares;
- Utilização de separadores não-lineares.

3.6.3 Diferentes Tipos de Função Kernel

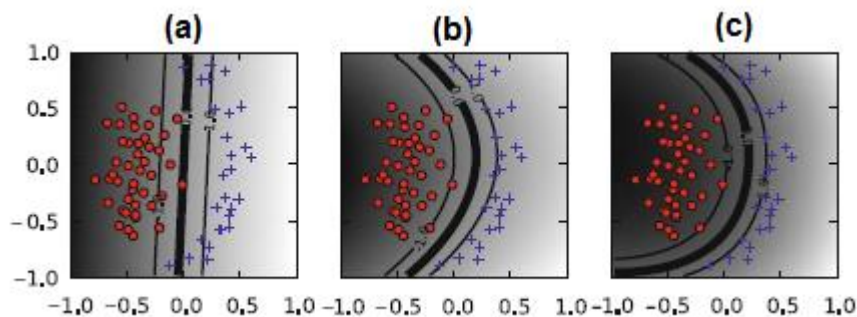
Uma função Kernel transforma os dados de treinamento para que uma superfície de decisão não-linear seja transformada em uma equação linear em um número maior de dimensões. As funções discriminantes lineares podem fornecer classificadores muito eficientes de duas classes, pelo fato de as classes de recursos poderem ser separadas por uma superfície de decisão linear. Para muitos casos, é mais fácil separar as classes com uma função linear se for possível transformar os dados dos recursos em um espaço com um maior número de dimensões. Uma forma de fazer isso é transformar os dados com a função Kernel, elas podem ser aplicadas sobre gráficos, conjuntos, caracteres e textos (CROWLEY, 2016).

- Função Kernel Polinomial

A função Kernel polinomial é uma função Kernel cujo seu expoente seja maior ou igual a dois, ao contrário, ela será linear quando a ordem for igual a 1. Cada ponto do conjunto de dados possui impacto no valor da função Kernel, assim como o expoente escolhido que irá definir o comportamento da superfície de decisão do conjunto de dados, funções com expoentes muito altos tendem a sobrecarregar o conjunto de dados e acabar afetando a performance do algoritmo. Ela é uma função Kernel popular, amplamente utilizada em reconhecimento de fala e outras aplicações de AM (LIMA, 2014; CROWLEY, 2016). O parâmetro envolvido exclusivamente na Função Polinomial é:

- Grau da Função: Altera o comportamento da superfície de decisão através do expoente escolhido, permitindo uma flexibilidade maior entre a separação o conjunto de dados. Na Figura 7 estão representados 3 casos distintos onde os expoentes são: 1, 2 e 5 respectivamente:

Figura 7 - Três valores diferentes de expoente para a função polinomial



Fonte: Adaptado de Ben-Hur e Weston (2010).

- Função Kernel de Base Radial (RBF)

A função Kernel RBF também possui a característica de resolver problemas que não são originalmente linearmente separáveis, através do mapeamento de alta dimensão (apenas funções Kernel Linear não possuem essa capacidade). Em boa parte dos problemas de classificação, o Kernel RBF apresenta um desempenho melhor que o Kernel Linear ou o Polinomial, e por ter menos parâmetros, é tido como a primeira opção muitas vezes ao se selecionar um Kernel. Contudo ele pode se tornar difícil de modelar a ponto de se chegar a um valor ótimo para seus parâmetros. Este Kernel é bastante utilizado pelas suas características de

mapear dados de entrada em espaços altamente dimensionais e ser um Kernel favorável quando se há pouco ou nenhum conhecimento da distribuição dos dados em questão, uma aplicação onde este Kernel é bastante utilizado é em visão computacional, que estuda e descreve sistemas de visão artificial implementados por hardware ou software (VIDHYA, 2013; NOVAKOVIC e VELJOVIC, 2011). O Parâmetro envolvido exclusivamente na função RBF é:

- Gama (γ): Também conhecido como parâmetro de largura do Kernel RBF, ele controla a flexibilidade do classificador ou seja, a similaridade entre os pontos. Quanto maior for o gama, menor é a similaridade entre eles, quanto menor o gama, maior a similaridade (pontos distantes se tornam similares).
- Função Kernel Universal VII de Pearson (PUK)

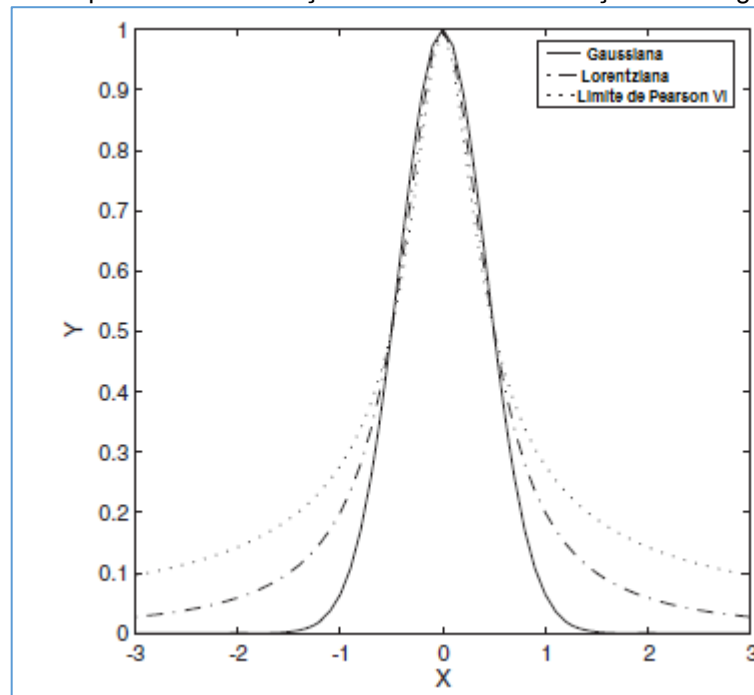
A função PUK foi aplicada como uma função de SVM que satisfazia as condições do teorema de Mercer para problemas de análise funcional. O teorema de Mercer, em análise funcional, é uma representação de uma função quadrática positiva-definida (função quadrática sobre um espaço vetorial que possui o mesmo sinal, positivo ou negativo, de cada vetor não-nulo deste espaço) resultante da soma de uma sequência convergente de funções adjacentes.

A função PUK possui uma excelente flexibilidade e possibilidade de se alterar facilmente entre um pico de função Gaussiano, um pico Lorentziano, ou outros, através da adaptação dos seus parâmetros. É possível então se utilizar a função PUK para substituir as outras funções antes mencionadas como uma função Kernel (ABAKAR e YU, 2014). Os parâmetros envolvidos exclusivamente na função PUK são:

- Ômega (ω): Variando o parâmetro ômega, é possível alterar a flexibilidade da curva de uma forma Gaussiana (ômega aproximado ao infinito) para uma forma Lorentziana (ômega igual a 1). Outros valores de ômega geram uma ampla gama de formas de pico (intermediárias).
- Sigma (σ): Em geral, o sigma serve como um indicador para a suavidade do modelo: um valor grande sugere uma relação quase linear, enquanto uma largura

(muito) pequena indica que um modelo polinomial de ordem superior se ajusta aos dados disponíveis.

Figura 8 - Comportamento da função de acordo com variação do ω e σ



Fonte: Adaptado de Üstün et al. (2006).

Segundo Ben-Hur e Weston (2010), seus experimentos apresentaram que as funções Kernel Gaussianas normalmente possuem uma acurácia e tempo de convergência melhores do que funções polinomiais, se os dados forem normalizados corretamente e um bom valor do parâmetro de largura do pico da função for escolhido.

3.6.4 Vantagens e Desvantagens do Uso de SVM

Um modelo de SVM é passível de prever com uma boa acurácia na maioria dos casos, entretanto, uma das principais razões para a popularidade do SVM é sua habilidade de modelar uma relação não-linear complexa selecionando uma função Kernel cabível (ABAKAR e YU, 2014).

Ben-Hur e Weston (2010) afirmam que há duas vantagens no uso de funções Kernel em SVM:

- A habilidade de gerar limites de decisão não-lineares usando métodos designados para classificadores lineares.

- O uso das funções Kernel permite o usuário a aplicar um classificador para dados que possuem nenhum espaço dimensional fixo em uma representação de espaço vetorial.
- Em muitas aplicações, classificadores não-lineares apresentam uma melhor acurácia. Ainda assim, classificadores lineares possuem suas vantagens, uma delas é prover algoritmos mais simples que trabalham bem com uma boa gama de exemplos.

Os mesmos autores também apresentaram algumas complicações para o uso do SVM, uma delas é que cabe ao humano por trás do algoritmo tomar certas decisões, como a maneira correta de se pré-processar os dados, qual Kernel usar e a sintonização de parâmetros de ambos função Kernel e do algoritmo. Escolhas feitas erroneamente podem ocasionar uma redução severa da performance final do algoritmo.

Para Deng et al. (2020) entretanto, o SVM pode solucionar problemas de modelos de dados de altas dimensões sobre condições finitas e possuir uma forte habilidade de generalização

Syarif et al. (2016) afirma que o SVM apresenta uma vantagem que é não ser afetados por mínimos locais, e posteriormente não sofrer pela recorrência à alta dimensionalidade através do uso dos vetores de suporte. Também ressalta que a performance depende altamente da sintonização dos parâmetros e da seleção da função Kernel correta.

3.6.5 Algoritmo de Otimização Mínima Sequencial (SMO)

O algoritmo SMO (do inglês: *Sequential Minimal Optimization*), ao contrário dos algoritmos SVM previamente discutidos, que trabalham com programação quadrática incessantemente, o SMO utiliza a programação quadrática analítica como apenas um passo de sua aplicação. O SMO gasta a maioria do seu tempo avaliando a função de decisão do que realizando a programação quadrática. Ele pode explorar bancos de dados que contenham um número substancial de elementos zeros, e apresentam bom desempenho com estes tipos de bancos de dados, sendo eles binários ou não-binários.

O algoritmo SMO resolve rapidamente o problema de programação quadrática sem nenhum armazenamento extra de matriz e sem aplicar uma rotina numérica iterativa para cada subproblema (PLATT, 2000).

3.6.6 Treinamento de um Algoritmo SMO

Alguns problemas quadráticos provenientes de métodos de SVM não podem ser solucionados usando métodos tradicionais de programação quadrática, especialmente para problemas de tamanhos maiores. Diferentes algoritmos foram propostos para resolver este problema. O algoritmo SMO o resolve iterativamente, selecionando o conjunto de trabalho e otimizando a função objetivo de acordo com ele, resolvendo seus subproblemas analiticamente. O processo se repete iterativamente até as condições de Karush-Kuhn-Tucker (KKT), que são condições particulares de igualdade e desigualdade, serem resolvidas para todas as instâncias de treino. Posteriormente um algoritmo chamado “SMOReg” foi proposto por Smola e Schölkopf para se resolver problemas de regressão utilizando o método do SMO convencional (AL - SHAMERY e AL – GASHAMY, 2018).

O algoritmo SMOReg utiliza um método de solução iterativo, onde ele divide a função objetivo em uma série de subproblemas e os resolve analiticamente. Pelo fato da limitação de equidade linear envolvendo os multiplicadores de Lagrange, o menor problema possível envolve dois multiplicadores e assim é possível resolver o problema analiticamente.

O funcionamento do algoritmo é possível de se simplificar em 3 passos:

- Encontrar um multiplicador de Lagrange que viola as condições KKT para o problema de otimização;
- Encontrar um segundo multiplicador e então otimizar o par de multiplicadores;
- Repetir os dois passos anteriores até a convergência do resultado.

Quando todos os multiplicadores de Lagrange satisfazem as condições de KKT (sem uma tolerância definida manualmente), o problema é resolvido. Embora

este algoritmo seja garantido para se convergir, métodos Heurísticos são utilizados para se escolher o par de multiplicadores para acelerar a taxa de convergência.

O algoritmo SMO usual apresenta um problema que é causado pelo uso de um valor único de limite. Através das condições de KKT para problemas duplos, dois parâmetros de limite são empregados para a resolução do problema principal. Estes parâmetros são empregados nos métodos de SMO desenvolvidos recentemente e que são mais robustos e mais rápidos que o SMO original, um destes métodos é o SMOReg (VITKAR, 2013).

3.6.7 Vantagens e Desvantagens do Algoritmo SMO

A vantagem do SMO está presente no fato de que se pode utilizar dois multiplicadores de Lagrange para se resolver o problema analiticamente, assim, uma iteração interna inteira devido à otimização numérica do QP é evitada. O código do algoritmo pode ser expresso em um “pequeno” código em C, ao invés de se optar por um banco de dados de código de programação quadrática que são muito extensos. A solução de cada subproblema é feita tão rapidamente que o problema geral de programação quadrática também é resolvido da mesma forma.

Porém, problemas muito extensos de AM utilizando SMO ou SVM não podem ser resolvidos em máquinas comuns devido à alta manipulação com matrizes e seu alto requisito computacional, então são necessários ambientes com maquinário adequado para a resolução dos mesmos, o que pode ocasionar num custo maior de equipamento (PLATT, 2000).

3.7 REDES NEURAIS ARTIFICIAIS (RNA'S)

3.7.1 Definição

Tendo como base central a ideia de simular o cérebro humano em seu funcionamento e em sua capacidade cognitiva, criou-se um modelo computacional que simulasse o comportamento do mesmo. Para isso, foram criados neurônios artificiais similares aos humanos, que foram interligados para formar redes que se mostraram capazes de fazer tarefas (cognitivas e/ou criativas), antes restritas ao cérebro humano.

O processo de treinamento do cérebro para realizar o aprendizado de coisas novas é feito através de um processo chamado “aprendizado por reforço”. Cada vez que um impulso sensorial particular passa através de uma sequência de sinapses, estas sinapses tornam-se mais capazes de transmitir o mesmo impulso da próxima vez, desta maneira, constituem-se os caminhos neurais (as sequências de sinapses) relativas àquele aprendizado. As Redes Neurais artificiais simulam este processo através da modificação dos pesos sinápticos para que a rede possa apresentar um comportamento similar ao aprendizado por reforço.

A ideia básica de uma rede neural é que o aprendizado seja possível pelo reconhecimento de padrões que possam levar a decisões concretas: sua estrutura básica irá receber diferentes variáveis, modificá-las e passá-las adiante para outra estrutura igual ou semelhante até que a rede chegue ao seu fim. São usadas diferentes camadas de diferentes tipos de neurônios (também chamados de *perceptrons*) para permitir que a rede neural aperfeiçoe seus cálculos e consiga resolver o problema em questão com base em uma biblioteca de treinamento que foi utilizada como modelo. Assim, temos uma grande rede que se comunica e modifica seus pesos e valores através de tentativa e erro para garantir que a tarefa seja realizada corretamente com a maior acurácia e precisão possível (MÜLLER, 2019; KOPIER et al., 2019).

Numa RNA, os neurônios representam as funções para processamento de informações. Uma RNA pode representar qualquer sistema linear ou não linear, mesmo quando nenhum modelo físico ou equação matemática é necessária e seu treinamento estabelece a correlação entre as variáveis de saída e de entrada. As RNA's têm se tornado um dos métodos de AM mais populares devido às suas vantagens sobre os outros, incluindo seu fácil treinamento, estrutura adaptativa e seus parâmetros de treino flexíveis. As RNA's são comumente utilizadas na resolução de problemas complexos, onde o comportamento das variáveis não é rigorosamente conhecido (FLECK et al., 2016; ABDELKHALEK et al., 2019; LI et al., 2017).

O primeiro modelo artificial de um neurônio biológico foi apresentado pela primeira vez em 1943 por Warren McCulloch e Walter Pitts, porém o primeiro trabalho demonstrando ligação direta com o aprendizado de redes artificiais de que se tem notícia foi apresentado por Donald Hebb no seu livro “The Organization of Behavior: A Neuropsychological Theory” (HEBB, 1949). Em seu trabalho, Hebb propôs uma teoria para explicar o aprendizado em neurônios biológicos baseada no reforço das

ligações sinápticas entre neurônios excitados. Deste trabalho surgiu-se uma regra que leva seu próprio nome e é utilizada em algoritmos de aprendizagem de máquina (BRAGA, 2007).

3.7.2 Construção e Componentes de uma Rede Neural Artificial

Para se implementar uma RNA deve-se definir previamente, algumas variáveis, dentre as quais: o número de nós na camada de entrada (estes nós correspondem ao número de variáveis de entrada; o número de camadas escondidas e seu respectivo número de neurônios a serem colocados em cada camada; o número de neurônios na camada de saída (FLECK et al., 2016).

De acordo com Braga (2007) um modelo básico de RNA possuirá os seguintes componentes:

- Conjunto de sinapses: Conexões entre os neurônios da RNA (cada conexão possui um peso sináptico);
- Integrador: Realiza as somas dos sinais de entrada da RNA, ponderados pelos pesos sinápticos;
- Função de ativação: Restringe a amplitude do valor de saída de um neurônio;
- Bias: Valor aplicado externamente a cada neurônio e tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação.

O modelo de cada unidade da rede pode incluir uma não-linearidade na sua saída, a qual deve ser reduzida. A função de ativação representa o efeito que a camada interna e o estado atual de ativação exercem na definição do próximo estado de ativação da unidade. Existem diversos tipos de funções de ativação, sendo as mais populares:

- Função limiar: Normalmente restringe a saída da RNA em valores binários [0,1]. A saída do neurônio assume valor 0 quando seu resultado for negativo, e 1 caso seja positivo.

- Função linear: Pode apresentar duas situações: se a região linear de operação é mantida sem entrar em saturação, surge um combinador linear; caso contrário a função linear se reduz à função limiar, se o fator de amplificação da região linear é infinitamente grande.
- Função tangente hiperbólica: pelo fato da função apresentar valores de ativação apenas no intervalo $[0,1]$, ela é substituída pela função tangente hiperbólica, que preserva a forma sigmoideal da função, podendo assumir valores positivos e negativos (FLECK et al., 2016).

O desenvolvimento do método de aprendizagem da RNA passará então por duas etapas, o treinamento e o teste. O processo de treinamento (ou aprendizado) ocorrerá a partir de uma base de dados provida ao algoritmo enquanto o processo de teste (validação) ocorrerá após o modelo ser treinado, utilizando dados que não foram previamente fornecidos para o treinamento do algoritmo, para evitar viciar o mesmo. Por isso muitas vezes os processos de aprendizagem de algoritmos têm seus conjuntos de dados separados em porcentagens de treino e de validação de modelo.

Para cada variável do conjunto de dados, sua distribuição deve ser qualitativa e quantitativa o suficiente para garantir um bom treinamento. Se a mesma for muito mal distribuída, o modelo treinado só terá um bom potencial de predição para uma região mínima local. Para uma predição numérica, as variáveis dependentes são normalmente as propriedades mais difíceis de se obter de forma simples. As variáveis independentes, pelo contrário, são facilmente medidas e tem relação com as variáveis dependentes (LI et al., 2017).

3.7.3 Classificação de Diferentes Tipos de Rede Neural Artificial

As redes neurais podem ser classificadas, de modo geral e simplificado, quanto à sua forma de aprendizado (Macroestrutura), sua arquitetura (Mesoestrutura), e suas funções de ativação utilizadas em seu interior (Microestrutura) (SOUZA et al., 2019).

Em geral, pode-se identificar três classes de arquiteturas de rede fundamentais: redes alimentadas com camada única (*feedforward*), redes alimentadas diretamente com múltiplas camadas (*multilayer perceptron*) e redes recorrentes (FLECK et al., 2016). Quanto às camadas, existem duas arquiteturas principais: as arquiteturas em camada única e em múltiplas camadas.

A rede de uma única camada é chamada de “rede de *perceptrons*” e praticamente não é mais utilizada recorrentemente, devido ao estudo de Minsky e Papert que provou no final da década de 60 que este tipo de rede só conseguia classificar corretamente problemas linearmente separáveis. Rummelhart, no início da década de 80, desenvolveu o algoritmo de “*backpropagation*” para o treinamento de redes de múltiplas camadas, que viria então a substituir o método de camada única e se tornar o mais utilizado até os dias atuais (KOPILER et al., 2019).

Com o alto desenvolvimento da computação, outros tipos de RNA's estão surgindo, como a rede neural profunda (também chamada de *Deep Neural Network* – DNN) que despertou interesse da comunidade científica devido à sua forte capacidade de aprendizado e inclusão do método de *deep learning*, que é um dos mais aconselháveis para se trabalhar com big data (LI et al., 2017).

3.7.4 Vantagens e Desvantagens de uma Rede Neural Artificial

Recentemente o uso de redes neurais para cognição de objetos cresceu enormemente. Existem várias razões para esse forte interesse, primeiramente pela crescente necessidade do uso de AM nas indústrias, pela disposição de melhores conjuntos de dados numéricos ou fotográficos para treinar os algoritmos e também a melhora de hardware para se realizar o processo de treinamento, permitindo o uso de algoritmos cada vez mais extensos e robustos para as atividades requeridas (visando melhorar o problema do *big data* mencionado anteriormente) (BUSCEMA et al., 2018).

De acordo com Haykin (2009) o uso das redes neurais oferece as seguintes capacidades:

- Não-linearidade: Um neurônio artificial pode ser linear ou não-linear, mas a não-linearidade é distribuída por toda a rede.
- Adaptabilidade: As redes neurais possuem a capacidade de adaptar seus pesos sinápticos a modificações do meio ambiente.
- Resposta a evidências: Uma rede neural pode ser projetada para fornecer informações sobre a decisão tomada, possibilitando rejeitar padrões ambíguos e, por consequência, melhorar o desempenho de classificação da rede.

- Informação contextual: Cada neurônio da rede é potencialmente afetado pelas atividades de todos os outros neurônios na rede. Dessa forma, a informação contextual é tratada naturalmente pela rede.
- Tolerância a falhas: Para se assegurar que uma rede neural seja de fato tolerante a falhas pode ser necessário adotar medidas corretivas no projeto do algoritmo utilizado para treiná-la.
- Análise neurobiológica: Os neurobiólogos encaram as RNAs como ferramentas de pesquisa para a interpretação de fenômenos neurobiológicos.

Uma das principais limitações deste algoritmo é a sua lentidão. Para não perder as variações do espaço de busca, o algoritmo o percorre de forma lenta. Na década de 90 passou a ser usado o termo de momento, que potencializa o treinamento, e também o auxilia a escapar de tomadas de decisões baseadas em mínimos locais (KOPILER et al., 2019). Outra das maiores dificuldades encontradas no uso das redes neurais é a escolha da melhor arquitetura, uma vez que esse processo é experimental e demanda um grande tempo de execução (FLECK et al., 2016).

Segundo Pinheiro e Pasa (2020): *“A tomada de decisão, no meio industrial, é um importante fator para obtenção de vantagens competitivas. Nesse sentido, as Redes Neurais Artificiais (RNA) podem ser utilizadas como ferramentas de apoio no processo de tomada de decisão. São ferramentas de grande importância em um sistema de planejamento, pois direcionam a melhor forma de aproveitar todos os recursos disponíveis pela empresa”*.

3.7.5 Rede Neural Artificial *Multilayer Perceptron* (MLP)

As redes MLP são as mais estudadas e aplicadas em problemas de classificação, cerca de 95% das aplicações encontradas na literatura utilizam esse tipo de modelo neural (BISSACOT, 2015).

Nas redes MLP cada neurônio efetua o produto do seu vetor de entrada pelo vetor de pesos associados a cada entrada e o resultado é então comparado a um valor de limiar. A Equação 2 representa este processo:

$$y_k = (FA) \times \left(\sum_{i=1}^n (y_i w_{ki}) + b_k \right) \quad (2)$$

onde: FA é a função de ativação, y_i é o termo de entrada, w_{ki} é o peso sináptico entre o neurônio “i” e o neurônio “k”, b_k é o valor de limiar (também conhecido como bias) e y_k é a saída calculada pelo neurônio i.

Em caso de disparo (quando se atinge um resultado desejado), o valor do produto escalar é utilizado como variável independente de uma função de ativação, cuja variável dependente constituirá a saída do neurônio. A função sigmoide é a mais comumente utilizada como função de ativação e os neurônios são conectados em modo acíclico, isso significa que neurônios de uma camada são ligados unicamente a neurônios da camada seguinte (FLECK et al., 2016; KOPILER et al., 2019; LI et al., 2017).

Dentre os algoritmos utilizados para treinamento de redes MLP, o mais conhecido e amplamente utilizado é o algoritmo *backpropagation*, que é também conhecido como método do gradiente descendente. O algoritmo procura minimizar o erro obtido ajustando pesos e limiares de ativação em busca dos pontos de mínimo da superfície de erro. O treinamento através do algoritmo de BP repete-se por um certo número de vezes (chamadas de épocas de treinamento). O mesmo ocorre em duas etapas: a etapa direta (ou *forward*), na qual se define a saída da rede para um determinado padrão de entrada ou conjunto de padrões de entrada; e a etapa reversa (ou *backward*) que ocorre posteriormente por quantas vezes forem definidas, que utiliza a diferença entre a saída de rede e a saída desejada e o erro da rede para atualizar os parâmetros livres (BISSACOT, 2015).

3.7.6 Aplicação do Algoritmo das Redes Neurais Artificiais no Ramo da Avicultura de Corte

Segundo Pinheiro e Pasa (2020), diversos trabalhos utilizaram redes neurais artificiais (principalmente na configuração MLP) na gestão da produção de frangos de corte. Dentre eles podem-se citar os trabalhos de: Bastiani et al. (2018) que estudaram a aplicação de algoritmos de mineração de dados na gestão da produção de frangos de corte por meio do software *WEKA*; Johasen et al. (2019) realizaram a previsão do

peso de abate de frangos de corte utilizando modelos dinâmicos de redes neurais artificiais e Amrael et al. (2017) analisaram o peso de frangos de corte com base em redes neurais artificiais e visão de máquina.

3.8 SOFTWARE “*WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS*” (*WEKA*)

O *Waikato Environment for Knowledge Analysis* – *WEKA* é um software de aprendizado que contempla uma série de algoritmos de preparação de dados, de aprendizagem de máquina (mineração dos dados) e de validação de resultados. O *WEKA* foi desenvolvido na Universidade de Waikato, na Nova Zelândia. Seu sistema está escrito em Java e distribuído sobre os termos de GNU, *General Public License*, (licença pública) e é possível rodá-lo em qualquer sistema operacional. O pacote *WEKA* contém uma coleção de ferramentas de visualização e algoritmos para análise de dados e modelagem preditiva, juntamente com interfaces gráficas do usuário para facilitar o acesso a essa funcionalidade. Uma limitação da ferramenta é a sua escalabilidade, uma vez que suas versões atuais limitam o volume de dados a ser manipulado à dimensão de memória principal. Mesmo assim, é possível minerar bases de dados relevantes, tornando o pacote atrativo para um grande número de aplicações (componentes do *WEKA* vem sendo comumente utilizados em diferentes projetos de pesquisa, como análise econômica ou empresarial) (SILVA, 2009; SINGHAL e JENA, 2013; KALMEGH, 2015).

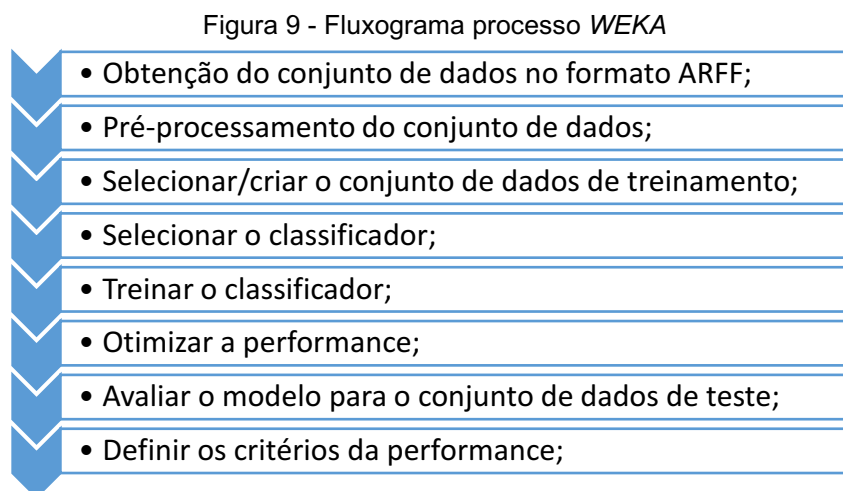
O *WEKA* fornece implementações de algoritmos de AM que podem ser facilmente aplicados à diferentes conjuntos de dados, também inclui uma variedade de ferramentas para transformar os mesmos (como os algoritmos para discretização). O programa permite pré-processar o conjunto de dados, alimentá-lo em um esquema de aprendizado e analisar o classificador e seu desempenho resultantes, sem a necessidade de escrever nenhum código de programação auxiliar. Ele foi designado para que fosse possível testar rapidamente novos métodos existentes em diferentes conjuntos de dados de forma flexível, assim como suas ferramentas de pré-processamento. Os usuários podem comparar estes diferentes métodos e avaliar/identificar qual é o mais apropriado para o problema em questão. Todos os algoritmos recebem seus dados de entrada na forma de uma tabela de relações

simples no formato ARFF (*Attribute-Relation File Format*). As vantagens do WEKA incluem:

- Estar disponível gratuitamente sob a GNU, *General Public License*;
- Sua portabilidade, devido a estar implementada na linguagem de programação Java (utilizada na maioria dos programas existentes atualmente) e por rodar em qualquer máquina disponível;
- Sua compilação de ferramentas de análise de dados e técnicas de modelagem;
- Sua facilidade de utilização devido à interface gráfica e intuitiva do programa;

O ambiente de trabalho inclui métodos para todos os problemas padrão de mineração de dados: regressão, classificação, clusterização, mineração de regras de associação e seleção de atributos. Todas as técnicas de AM do WEKA partem do princípio de que todos os dados estão disponíveis em formato de texto simples e relacionados, onde cada ponto de entrada é descrito por um número fixo de atributos (numéricos ou nominais, apesar de alguns outros atributos também serem aceitos), então conhecer os dados é parte integrante do trabalho e são fornecidos muitos recursos de visualização de dados e ferramentas de pré-processamento dos mesmos (WITTEN et al., 2011; KALMEGH, 2015).

Na Figura 9 está representado um fluxograma de um exemplo de processo no WEKA:



Fonte: Adaptado de Kulkarni e Kulkarni (2016).

4 PROCEDIMENTOS METODOLÓGICOS

No capítulo a seguir serão abordadas as metodologias utilizadas no presente trabalho para se aplicar os algoritmos de AM, como o pré-processamento dos dados, a sintonização de parâmetros, a escolha da melhor configuração para cada algoritmo e a posterior avaliação dos mesmos de acordo com os parâmetros de validação dos resultados. Na primeira seção se encontra o pré-processamento dos dados, posteriormente na seção 2 serão abordados os testes com os algoritmos, na seção 3 a avaliação e validação dos resultados e por fim na seção 4 a classificação das variáveis mais influentes.

4.1 PRÉ-PROCESSAMENTO DOS DADOS

Os dados para o presente trabalho foram obtidos do estudo de processo de abate de frango industrial feito por Sant'anna (2008) em sua dissertação. O conjunto de dados foi separado em dez variáveis de entrada (dados para treinamento e validação) e uma variável de saída (que é o valor a ser predito pelo algoritmo), totalizando 11 variáveis. O conjunto de dados está representado na seção de anexos do presente trabalho. Os dados foram alimentados no software *WEKA* para serem utilizados nos algoritmos propostos neste trabalho.

Quadro 2 - Conjunto de Dados

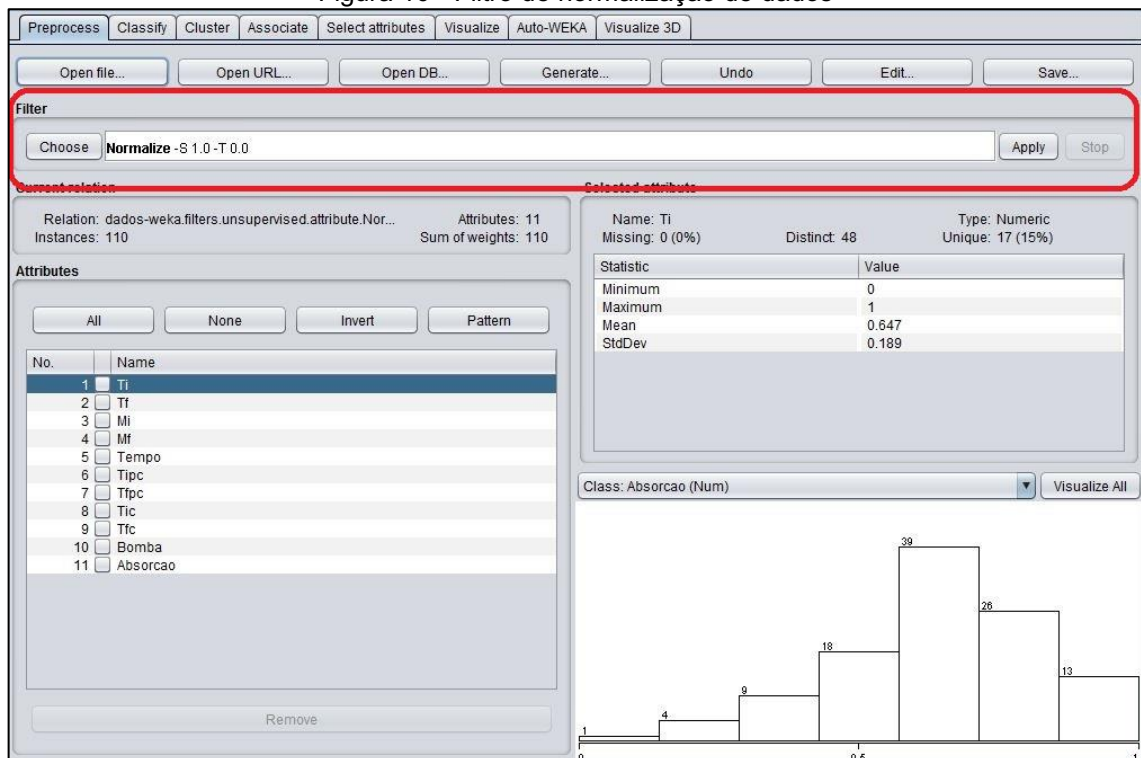
Variáveis de Entrada
Temperatura Inicial da Carcaça (Ti)
Temperatura Final da Carcaça (Tf)
Massa Inicial da Carcaça (Mi)
Massa Final da Carcaça (Mf)
Tempo de Resfriamento (Tempo)
Temperatura Inicial do <i>Pré-Chiller</i> (Tipc)
Temperatura Final do <i>Pré-Chiller</i> (Tfpc)
Temperatura Inicial do <i>Chiller</i> (Tic)
Temperatura Final do <i>Chiller</i> (Tfc)
Vazão do Borbulhamento (Bomba)
Variável de Saída
Absorção de Água na Carcaça (Absorção)

Fonte: Autoria própria (2020).

Como as variáveis de entrada para os algoritmos apresentam uma não-linearidade, é necessário que haja uma normalização de seus valores em intervalos entre 0 e 1, pois assim o algoritmo apresentará um melhor desempenho ao trabalhar com a sua respectiva função de transferência para o conjunto de dados a serem avaliados. O próprio software *WEKA* oferece a opção de normalização de dados dentro de seus comandos de pré-processamento, logo não é necessário realizar a normalização manualmente ao aplicar os algoritmos trabalhados. Dentre os algoritmos, existe um que oferece a possibilidade de normalização ao serem aplicados os seus testes (caso das Redes Neurais Artificiais). Tal opção foi desabilitada.

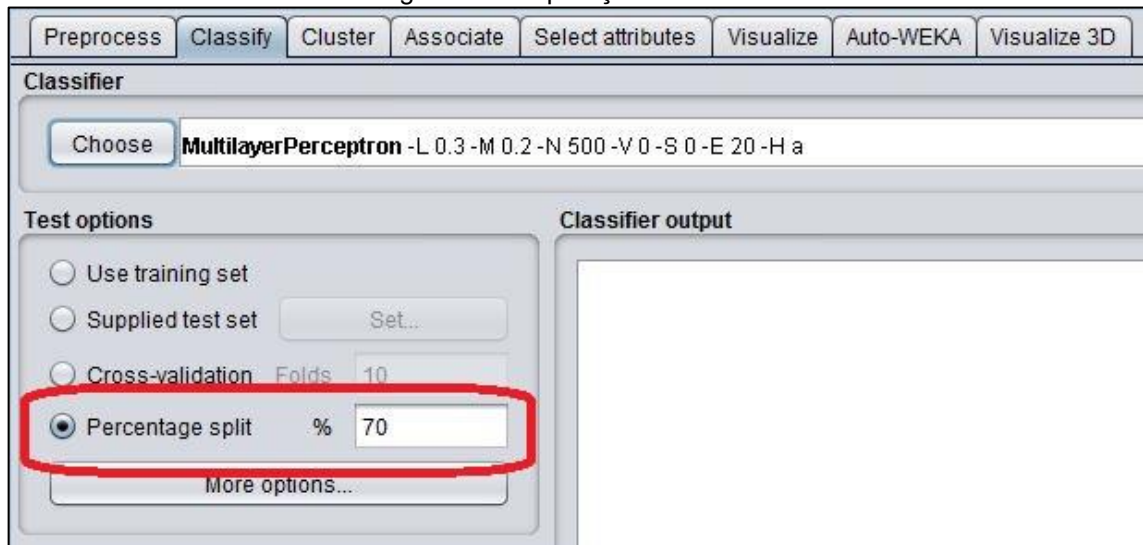
Os dados foram separados em dois grupos ao total, sendo: um grupo para treinamento do algoritmo constituindo 70% dos dados totais e um grupo de teste constituído dos 30% restante dos dados. Tal separação foi definida com base em pesquisas na literatura, como o trabalho feito por Morariu et al. (2017), estudando o *Multilayer Perceptron*, que apresentou porcentagens de separação semelhantes para validação dos dados. A representação da interface do software com a aplicação da normalização e da separação dos dados estão representadas nas Figuras 10 e 11:

Figura 10 - Filtro de normalização de dados



Fonte: WEKA (2020).

Figura 11 - Separação dos dados



Fonte: WEKA (2020).

4.2 TESTES EXPERIMENTAIS

4.2.1 REDES NEURAIS ARTIFICIAIS

O primeiro algoritmo a ser testado neste trabalho foi o das redes neurais do tipo *Multilayer Perceptron*, assim como diferentes arquiteturas e suas respectivas camadas escondidas. Todas as camadas da rede neural serão treinadas por um algoritmo de *backpropagation* (mais indicado para problemas não-lineares) e utilizaram uma função padrão de ativação sigmoide nas camadas ocultas, que é a única função possível disponibilizada no software utilizado para se trabalhar com classes numéricas, e está sendo representada pela Equação 3:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

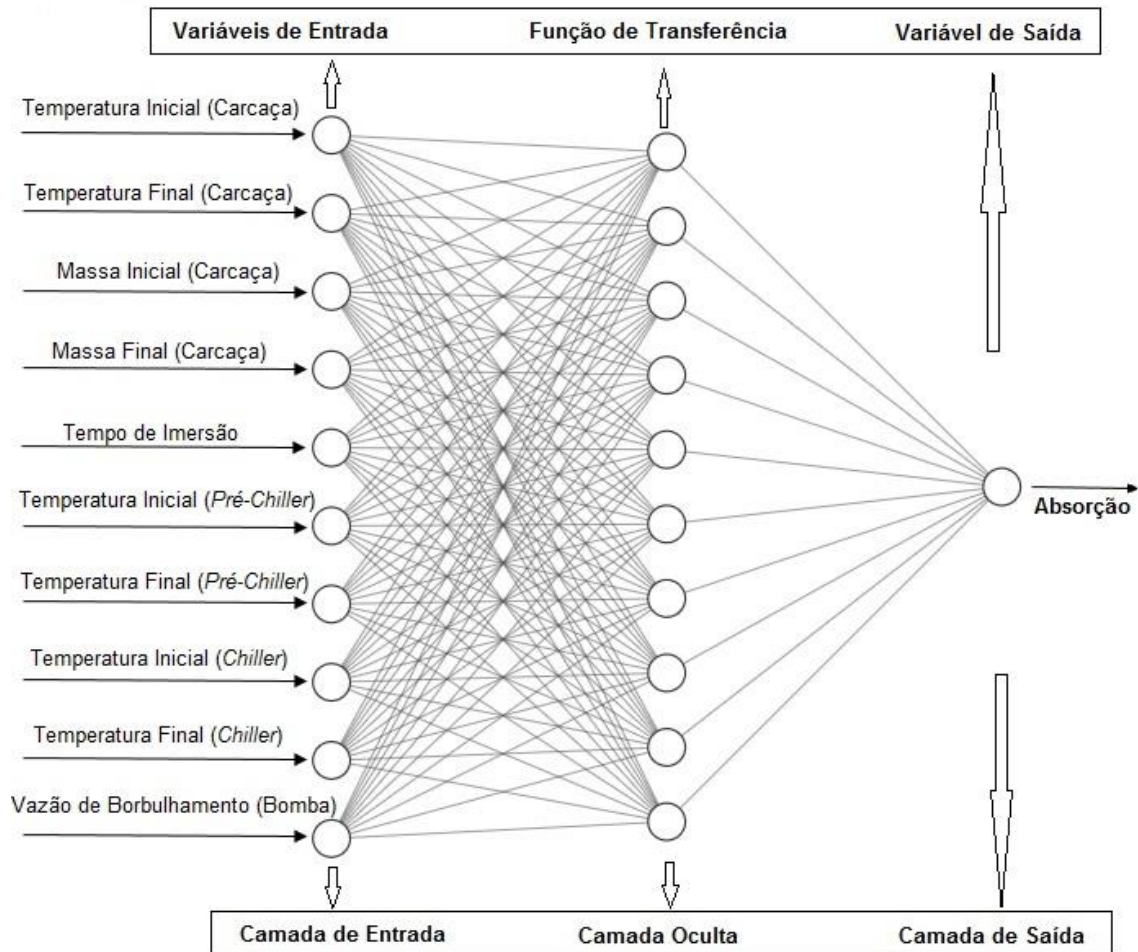
onde “x” é o resultado do somatório da multiplicação entre os valores de entrada, seus respectivos pesos, somados a uma constante (chamada “bias”) que serve para ajustar a saída junto com a soma ponderada das entradas no neurônio.

Para a função de ativação da camada de saída, foi utilizada uma função linear, que também é a função padrão disponibilizada pelo programa, e está representada na Equação 4:

$$f(x) = ax \quad (4)$$

A representação de uma rede neural multicamadas está representada na Figura 12:

Figura 12 - Esquema da rede neural artificial aplicada neste trabalho.



Fonte: Autoria própria (2020).

Os parâmetros alterados para a configuração da rede neural neste trabalho foram o número de épocas (*TrainingTime*), o número de camadas presentes na rede e o número de neurônios em cada camada. As configurações testadas estão representadas no Quadro 3:

Quadro 3 - Configurações testadas da rede neural

Num. de camadas ocultas	Num. de neurônios por camada oculta	Épocas
1	5	500, 1000, 1500, 2000, 2500, 3000
1	10	500, 1000, 1500, 2000, 2500, 3000
1	15	500, 1000, 1500, 2000, 2500, 3000
1	20	500, 1000, 1500, 2000, 2500, 3000
2	5 / 5	500, 1000, 1500, 2000, 2500, 3000
2	5 / 10	500, 1000, 1500, 2000, 2500, 3000
2	5 / 15	500, 1000, 1500, 2000, 2500, 3000
2	5 / 20	500, 1000, 1500, 2000, 2500, 3000
2	10 / 5	500, 1000, 1500, 2000, 2500, 3000
2	10 / 10	500, 1000, 1500, 2000, 2500, 3000
2	10 / 15	500, 1000, 1500, 2000, 2500, 3000
2	10 / 20	500, 1000, 1500, 2000, 2500, 3000
2	15 / 5	500, 1000, 1500, 2000, 2500, 3000
2	15 / 10	500, 1000, 1500, 2000, 2500, 3000
2	15 / 15	500, 1000, 1500, 2000, 2500, 3000
2	15 / 20	500, 1000, 1500, 2000, 2500, 3000
2	20 / 5	500, 1000, 1500, 2000, 2500, 3000
2	20 / 10	500, 1000, 1500, 2000, 2500, 3000
2	20 / 15	500, 1000, 1500, 2000, 2500, 3000
2	20 / 20	500, 1000, 1500, 2000, 2500, 3000

Fonte: Autoria própria (2020).

4.2.2 Algoritmo *K-Star* (K^*)

O segundo algoritmo trabalhado é o K^* , que utilizou uma função de entropia como medida de distância entre dois pontos, (a) e (b). A função utilizada neste algoritmo (chamada de função K^*), embora estritamente não seja uma função própria de distância, visto que $K^*(a|a)$ (a função K^* aplicada do ponto “a”, até o próprio ponto “a”) seja diferente de zero e sua representação seja não simétrica, pode ser aplicada pela Equação 5:

$$K^*(b|a) = -\log_2 P^*(b|a) \quad (5)$$

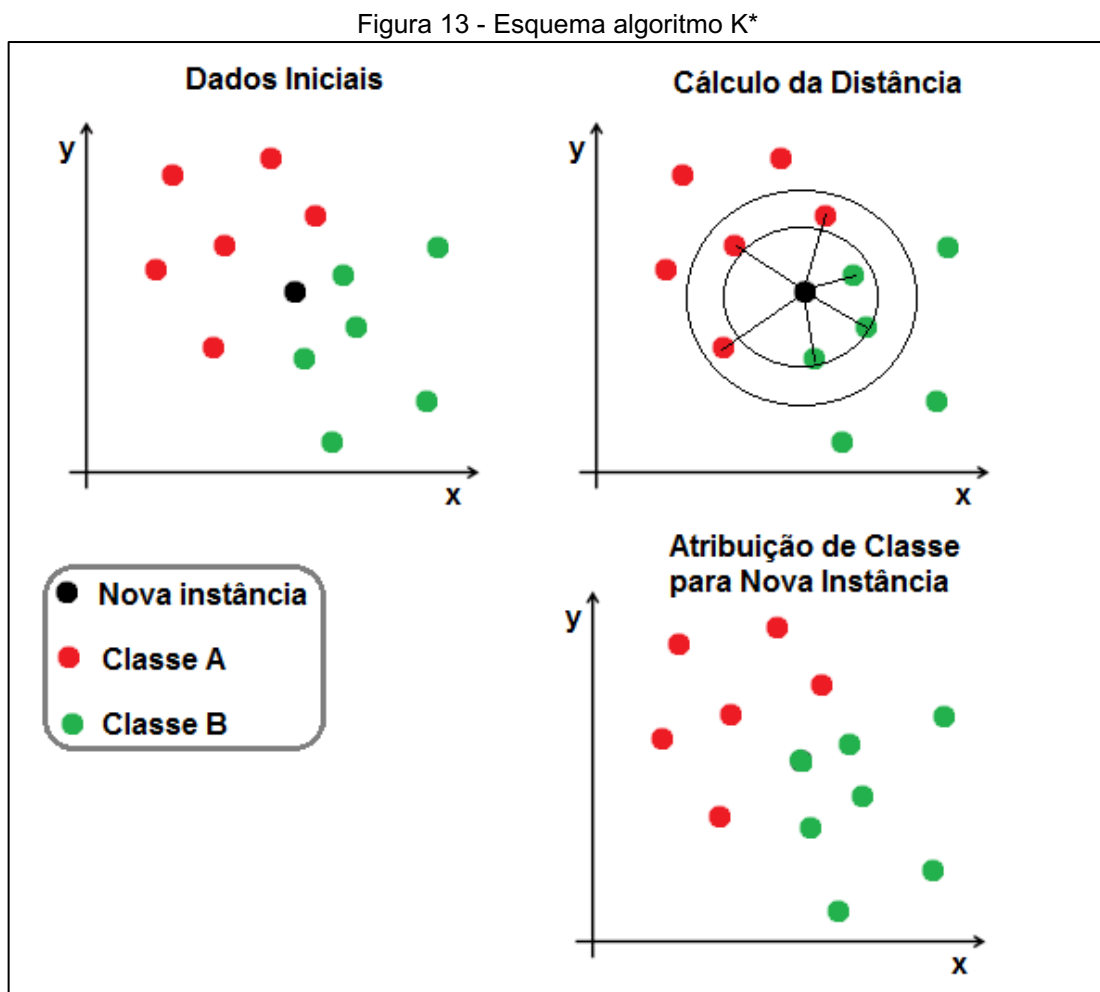
onde “a” e “b” são as instâncias e a função de probabilidade P^* é definida como a probabilidade de todos os caminhos da instância “a” para a instância “b”:

$$P^*(b|a) = \frac{x}{\sqrt{2x - x^2}} \left(\frac{1 - \sqrt{2x - x^2}}{1 - x} \right)^i \quad (6)$$

$$i = |a - b| \quad (7)$$

onde “x”, neste caso, é uma constante arbitrária de proporcionalidade da função entre as duas instâncias.

Uma representação esquemática do algoritmo K^* está representada na Figura 13:



Fonte: Autoria própria (2020).

O parâmetro configurado para os testes com o algoritmo K^* foi o *GlobalBlend* (GB), onde este parâmetro define a importância dos vizinhos próximos da nova

instância e é representado entre 0% e 100%, sendo que em 0% são apenas considerados os vizinhos de menores distâncias para a nova instância, e 100% são considerados todos os vizinhos independente da distância. Nos testes realizados o valor de GB variou entre 0 e 100%, como representado no Quadro 4:

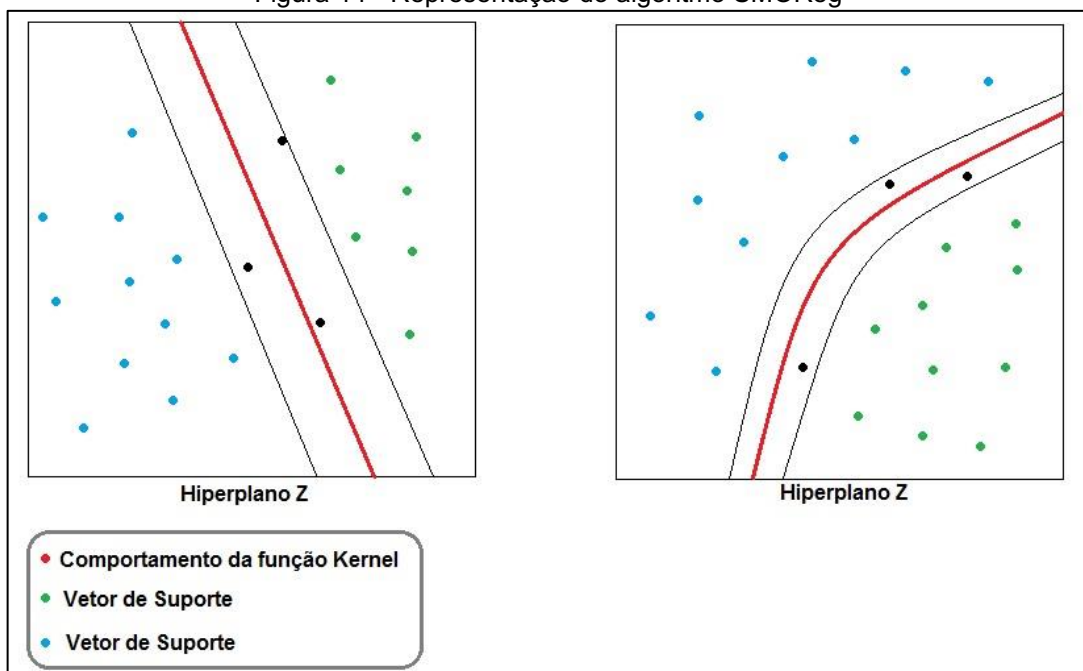
Iteração	Valor do GB
1º	0
2º	5
3º	25
4º	50
5º	75
6º	100

Fonte: Autoria própria (2020).

4.2.3 Algoritmo de Otimização Mínima Sequencial para Regressão (SMOReg)

O terceiro e último algoritmo utilizado foi o “SMOReg”, que aplicou o método de otimização mínima sequencial (SMO) em suporte de vetor de máquina para trabalhos de regressão envolvendo valores numéricos. Uma esquematização do algoritmo está representada na Figura 14:

Figura 14 - Representação do algoritmo SMOReg



Fonte: Autoria própria (2020).

Primeiramente, o algoritmo usa o método SMO para contar o limite de multiplicadores de Lagrange, determinando-os através das equações:

$$L = \max(0, a_2 - a_1), \quad H = \min(C, C + a_2 - a_1) \quad (8)$$

onde a_i são os multiplicadores de Lagrange e C é o parâmetro de complexidade.

Se o valor alvo y_1 for similar a y_2 no conjunto de dados, sendo que este é representado por $\{(x_i, y_i)\}$, temos então que o seguinte limite é aplicado para a_2 :

$$L = \max(0, a_2 - a_1 - C), \quad H = \min(C, a_2 - a_1) \quad (9)$$

A segunda diferenciação da função objetivo pode ser expressa então por:

$$\eta = K(x_1, x_1) + K(x_2, x_2) - 2 \times K(x_1, x_2) \quad (10)$$

onde K é a respectiva função Kernel.

Em condições normais, a função objetivo terá um valor positivo. Desta forma, o método SMO contará o novo mínimo da forma do limite a seguir:

$$a_2^{novo} = a_2 + \frac{y_2(E_1 - E_2)}{\eta} \quad (11)$$

onde E é o erro de saída para determinado padrão.

O próximo passo é encontrar o limite mínimo, cortando-o da linha de mínimos, que será limitado até a extremidade da linha de máximos:

$$a_2^{novo, cortado} = \begin{cases} H & \text{se } a_2^{novo} \geq H; \\ a_2^{novo} & \text{se } L < a_2^{novo} < H \\ L & \text{se } a_2^{novo} \leq L \end{cases} \quad (12)$$

Considerando o valor $s = y_1 y_2$ com o valor a_1 contando para o valor recém-obtido de a_2 , têm-se então:

$$a_1^{novo} = a_1 + s(a_2 - a_2^{novo,cortado}) \quad (13)$$

As funções Kernel utilizadas para os testes realizados no presente trabalho estão denotadas no Quadro 5:

Quadro 5 - Funções Kernel	
Função Kernel	Equação
Kernel Polinomial	$k(\vec{x}, \vec{z}) = (\vec{z}^T \vec{x} + c)^n$
Função de Base Radial (RBF)	$g(\vec{X}) = \sum_{n=1}^N \vec{W}_n^T f(\ \vec{X} - \vec{X}_n\)$
Kernel de Pearson Universal (PUK)	$f(x) = \frac{H}{[1 + (2 \times (x - x_0) \sqrt{2\frac{1}{\omega} - \frac{1}{\sigma}})]^\omega}$

Fonte: Autoria própria (2020).

Os parâmetros alterados neste algoritmo variaram de acordo com a função Kernel utilizada, sendo para a função polinomial: Parâmetro de complexidade C e o expoente n da função. Para a função Kernel RBF: Parâmetro de complexidade C e o termo Gama da função. Para a função Kernel PUK: Parâmetro de complexidade C, termo Ômega e termo Sigma da função. No Quadro 6 estão representados todos os valores testados no trabalho para os parâmetros abordados:

Quadro 6 - Parâmetros de teste SMOReg

Parâmetro	Valores
Função Kernel Polinomial	
C	1 / 10 / 100 / 1000
Expoente	1 / 2 / 3 / 4
Função Kernel RBF	
C	1 / 10 / 100 / 1000
Gama	0,001 / 0,01 / 0,02 / 0,05 / 0,1 / 0,2 / 0,5 / 0,7 / 1
Função Kernel PUK	
C	1 / 10 / 100 / 1000
Ômega	0,1 / 0,5 / 1 / 2 / 5 / 10 / 30
Sigma	0,5 / 1 / 2 / 5 / 10 / 20 / 30

Fonte: Autoria própria (2020).

4.3 AVALIAÇÃO E VALIDAÇÃO DOS RESULTADOS

4.3.1 Coeficiente de Correlação

O melhor algoritmo e sua configuração serão definidos pelo coeficiente de correlação (R) que indica a porcentagem em que os valores preditos estão relacionados com o valor real. Sua função é representada por:

$$R = \frac{S_{pA}}{\sqrt{S_p S_A}} \quad (14)$$

$$S_{pA} = \sum_i^n \frac{(p_i - \bar{p})(a_i - \bar{a})}{n - 1} \quad (15)$$

$$S_p = \sum_i^n \frac{(p_i - \bar{p})^2}{n - 1} \quad (16)$$

$$S_a = \sum_i^n \frac{(a_i - \bar{a})^2}{n - 1} \quad (17)$$

onde “n” e “i” se referem ao número de predições, “p” e “a” são os valores preditos e atuais, respectivamente e “ \bar{p} ” e “ \bar{a} ”, são os valores médios preditos e atuais, respectivamente.

4.3.2 Erro Médio Quadrado

Outro parâmetro utilizado para a definição do melhor algoritmo será pelo erro quadrático médio, também chamado de “*mean square error*” (MSE), que é o principal e mais utilizado critério de aceitação para este tipo de estudo e pode ser descrito pela Equação 18:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{i,Alg} - y_{i,exp})^2 \quad (18)$$

onde “n” é o número de experimentos utilizados, “ $y_{i,Alg}$ ” é o resultado previsto pela algoritmo testado e “ $y_{i,exp}$ ” é o resultado obtido experimentalmente. Então têm-se que este tipo de critério avalia a distância do erro médio entre o valor predito e o atual.

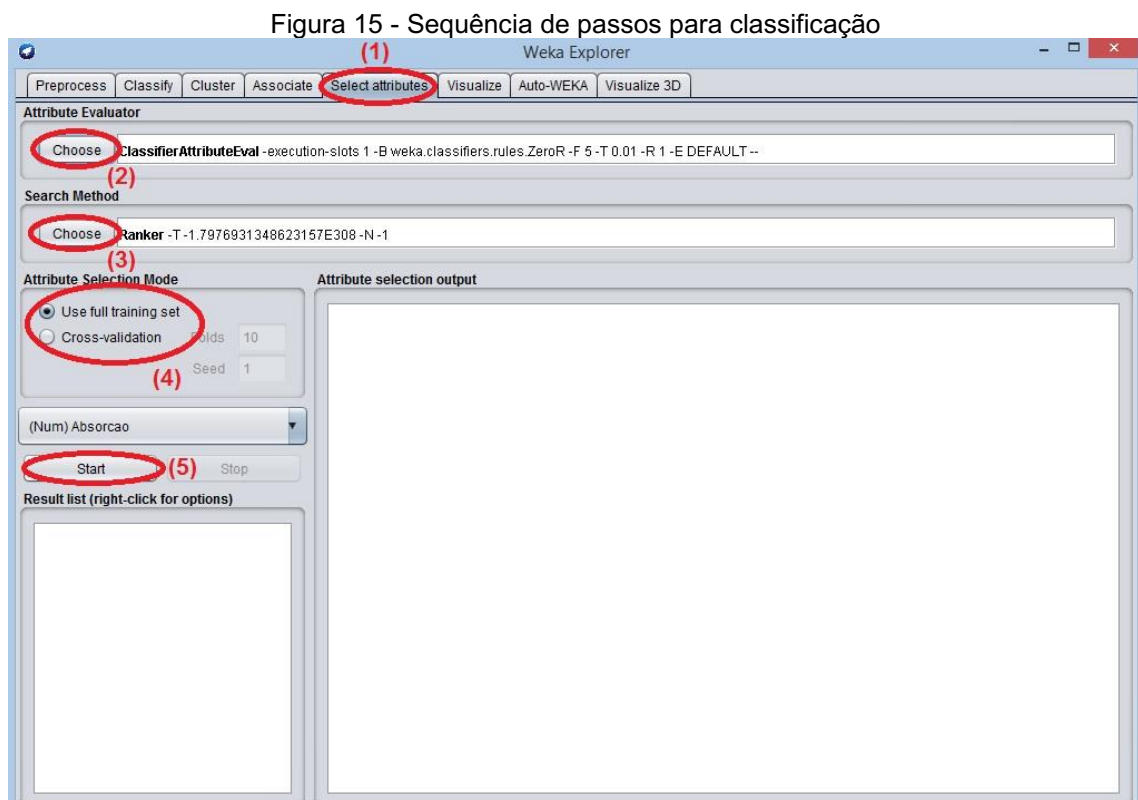
Logo, a configuração com o maior R e menor MSE será analisada, para escolha do melhor algoritmo testado neste trabalho.

4.4 CLASSIFICAÇÃO DAS VARIÁVEIS

Posteriormente foi realizada a classificação das variáveis de entrada que mais influenciam na variável de saída estudada para o processo em questão. O *software* disponibiliza a função de classificação de variáveis em sua interface, apenas sendo necessário selecionar o algoritmo responsável por classificar as mesmas. No presente trabalho, foram utilizados os mesmos algoritmos tanto para a predição quanto para a classificação. Os parâmetros individuais dos algoritmos e seus respectivos valores ótimos também foram mantidos, para se obter seu melhor resultado possível em cada teste de classificação.

O método de busca utilizado neste trabalho foi o método *Ranker*, que consiste em apresentar uma ordem de importância das variáveis de acordo com sua avaliação através dos algoritmos trabalhados. Para a classificação das variáveis foi utilizado todo o conjunto de dados, sem utilização de separação em porcentagens de validação ou de validação cruzada.

Para a realização da classificação, deve-se acessar a aba do software designada para a função (1), escolher o algoritmo a ser testado e avaliar seus parâmetros (2), selecionar o método de busca (3), selecionar a opção de uso do conjunto de dados (4) e então iniciar o teste (5). Esta sequência está representada na Figura 15:



Fonte: WEKA (2020).

5 RESULTADOS E DISCUSSÕES

Na seção a seguir serão apresentados os resultados dos testes realizados para a predição e classificação das variáveis, primeiramente para cada algoritmo, analisando-os e comparando-os com trabalhos que apresentam os mesmos algoritmos utilizados em termos de performance, posteriormente serão comparados os resultados dos testes entre si para a definição do melhor algoritmo em ambas as tarefas de predição e classificação.

5.1 PREDIÇÃO DO VALOR DA ABSORÇÃO

5.1.1 Seleção da melhor configuração para o algoritmo *K-Star* (K^*)

Após a aplicação dos dados no software *WEKA*, mediante mudança dos parâmetros indicados (neste caso o valor do *GlobalBlend*), foram realizados 6 testes no total em busca de se obter o melhor resultado da predição da variável de saída. As configurações e seus respectivos resultados estão retratados no Quadro 7:

Quadro 7 - Resultados testes *K-Star*

Teste	GlobalBlend	R	MSE	Tempo (s)
1	0	0,5011	3,5348	0,08
2	5	0,4364	3,7400	0,04
3	25	0,3970	3,7512	0,04
4	50	0,3688	3,8072	0,04
5	75	0,3744	3,8306	0,04
6	100	0,2267	4,2185	0,04

Fonte: Autoria própria (2020).

A partir da Tabela 7 é possível identificar a melhor e a pior configuração para o algoritmo do *K-Star* de acordo com os parâmetros analisados. O teste 1 apresentou o melhor resultado obtido, onde para um *GlobalBlend* de 0, obteve-se o R de 0,5011; MSE de 3,5348 e um tempo computacional necessário para a realização do teste, de 0,08 segundos. Este resultado apesar de ter sido o melhor entre os apresentados, ainda possui um baixo nível de aceitação, visto que o nível de correlação entre os valores preditos e os valores reais estão apenas em 50%. Se comparados a outros

estudos envolvendo predição de valores por inteligência artificial, é possível encontrar estudos onde o algoritmo *K-Star* pode apresentar uma eficiência maior.

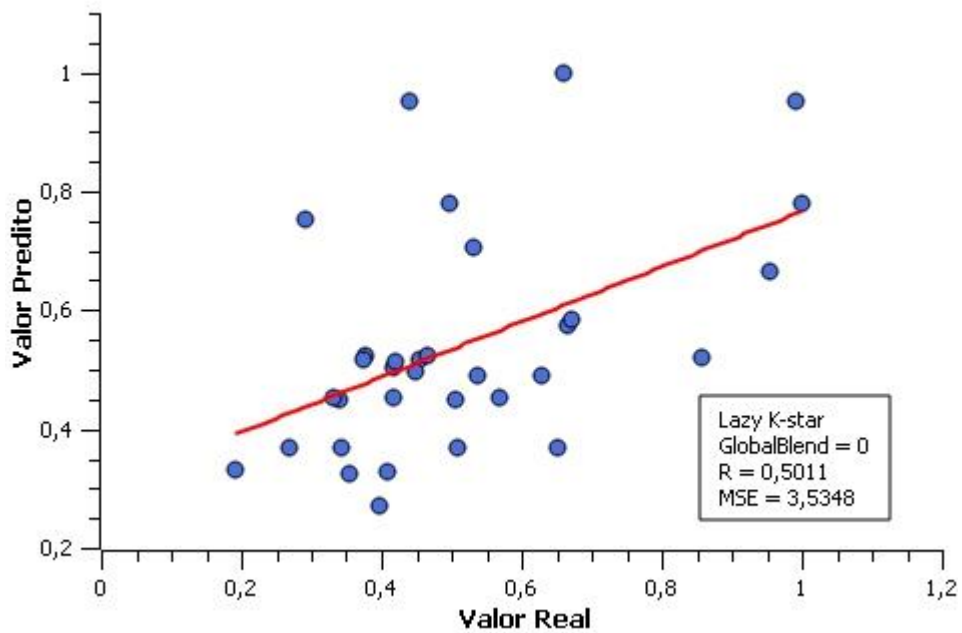
O estudo realizado por Ravikumar et al. (2019), analisou o diagnóstico de falhas de rolos no sistema de transporte de correias através deste algoritmo e obtiveram um resultado de 91% de eficácia, resultado este que foi consideravelmente maior que o do presente trabalho.

Assim como o trabalho de Sharma et al. (2017), que estudou o monitoramento da condição do rolamento de rolos usando sinal sonoro, obtendo resultados de 88% de eficácia (dependendo do parâmetro utilizado e do processamento dos dados).

Mahmood e Hussein (2013) estudaram o sistema de detecção de intrusos em redes digitais utilizando algoritmos de aprendizado de máquina e redução de conjunto de recursos, onde o mesmo obteve o resultado de 98% até 99% de eficácia. Este último sendo o melhor dentre todos os trabalhos apresentados anteriormente e consideravelmente melhor que o resultado apresentado pelo presente trabalho.

O teste 6 apresentou o pior resultado dentre os testes realizados, com um R de apenas 0,2267 e um MSE de 4,2185, sendo seu tempo computacional de 0,04 segundos para a realização do teste com o modelo. O resultado da eficiência do modelo foi altamente dependente do valor do parâmetro *GlobalBlend*, onde para o presente trabalho o melhor valor utilizado foi de 0, o que está de acordo com o que diz o livro da IFIP (em português: Federação Internacional para Processamento de Informação): “Artificial Intelligence Applications and Innovations” de 2018, onde relata que os trabalhos com menores valores de *GlobalBlend* obtiveram resultados melhores. Porém, esse resultado também depende do conjunto de dados estudado, como o estudo feito por Sajad (2019), utilizando PCA e algoritmos de aprendizado de máquina para detecção de câncer de pulmão, e que apresentou um melhor resultado de eficácia para um *GlobalBlend* entre 60 e 80; e também o estudo feito por Sahu et al. (2015) classificando estados oculares relacionados a estudos da medicina, e que obteve uma melhor eficiência com um *GlobalBlend* de 39 e 42.

A representação do valor predito normalizado pelo valor real normalizado, para o melhor resultado obtido está retratada na Figura 16, onde os pontos azuis são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 16 - Resultado melhor configuração *K-Star*

A partir da Figura 16, percebe-se que não há uma boa linearidade entre os valores preditos e os valores reais, ou seja, o algoritmo teve dificuldades em prever o resultado da absorção em relação ao valor que foi obtido experimentalmente, isso se deve ao fato dos pontos estarem distantes da reta que impõe uma linearidade nos resultados. Isso implica que o teste não obteve um bom coeficiente de correlação e nem obteve um baixo erro entre seus dados, como provado na Tabela 7. Logo para o conjunto de dados estudados, o algoritmo *K-Star* não obteve um bom desempenho para a predição dos valores da absorção da água na carcaça de frango. A tabela de comparação entre os valores reais e os preditos (ambos sem estarem normalizados) para cada iteração, assim como sua respectiva diferença, podem ser encontrados no apêndice A.

5.1.2 Seleção para a melhor configuração para o algoritmo SMOReg.

Os testes com o conjunto de dados para o algoritmo SMOReg gerou um total de 80 testes que foram divididos de acordo com as 3 funções Kernel utilizadas neste trabalho, sendo elas: Polinomial Kernel, RBF e PUK.

5.1.2.1 Kernel Polinomial

Para a função Kernel Polinomial, foram realizados 8 testes onde através da variação dos parâmetros C e expoente da função. Foram encontrados os seguintes resultados que estão representados no Quadro 8:

Quadro 8 - Resultados SMOReg Polinomial

C = 1				
Teste	Grau da Função	R	MSE	Tempo (s)
7	1	0,2260	4,3231	0,02
8	2	0,3691	3,8244	0,04
C = 10				
Teste	Grau da Função	R	MSE	Tempo (s)
9	1	0,4893	3,5415	0,02
10	2	0,8482	1,6249	0,02
C = 100				
Teste	Grau da Função	R	MSE	Tempo (s)
11	1	0,9969	0,0438	0,02
12	2	0,9731	0,2434	0,01
C = 1000				
Teste	Grau da Função	R	MSE	Tempo (s)
13	1	0,9981	0,0175	0,01
14	2	0,9810	0,1860	0,01

Fonte: Autoria própria (2020).

Para esta função, o parâmetro C que obteve o melhor e o pior resultados, respectivamente, foram o C = 1000 e C = 1. Devido à relação direta deste com os dados estudados (maior proximidade dos limites da função Kernel com os dados estudados à medida que o C aumenta). O teste que apresentou o melhor resultado para esta função Kernel foi o teste 13, onde se obteve um R de 0,9981 e um MSE de 0,0175 com um tempo computacional exigido de 0,01 segundos. Esse resultado é considerado bom, visto que aproxima com quase perfeição dos valores reais de acordo com seu coeficiente de correlação. O expoente da função que apresentou o melhor resultado foi o de valor 1, para que a função Kernel apresentasse um melhor comportamento para separação dos dados no hiperplano Z. O teste que apresentou

o pior resultado o teste 7, com um R de 0,226 e um MSE de 4,3231 e um tempo computacional exigido de 0,02 para se construir o modelo do teste.

Ao comparar este valor com outros trabalhos realizados para a predição de valores com o algoritmo SMOReg utilizando a função Kernel Polinomial, os resultados obtidos neste trabalho apresentam melhor performance.

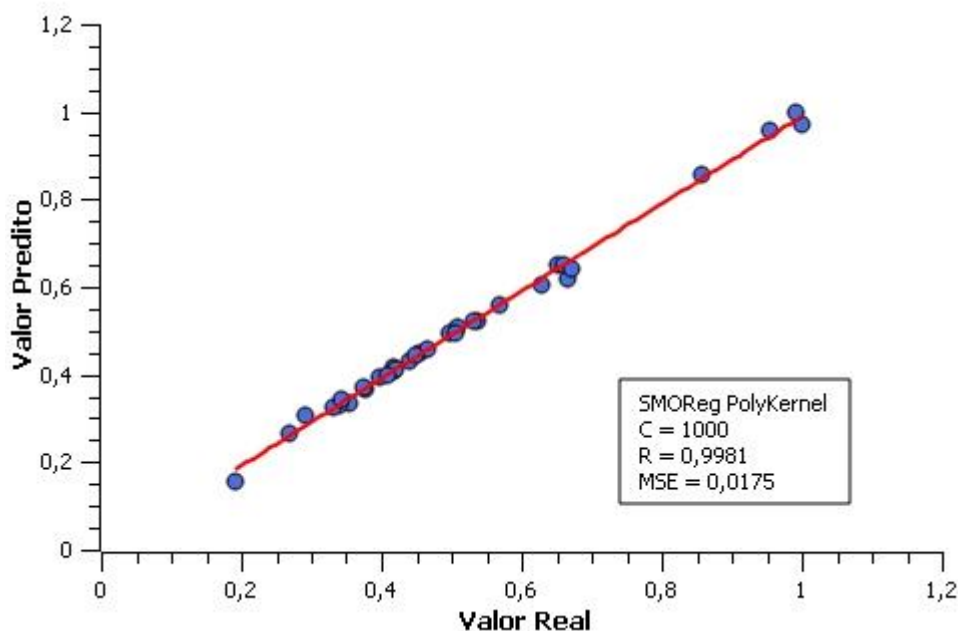
Abakar e Yu (2014) estudaram em seu trabalho a predição de valores da tenacidade de fios comparando-os suas propriedades em relação a suas fibras, e obtiveram um valor de 93% de eficácia utilizando o algoritmo SMOReg.

Nuanmeesri e Poomhiran (2019) em seu trabalho, analisaram o melhoramento da conversação do Chatbot através das técnicas de análise e classificação de sentimentos, e obtiveram resultados entre 79% até 89% através da variação dos parâmetros no algoritmo.

Além dos autores mencionados previamente, o estudo feito por Baitharu et al. (2016) em seu trabalho analisou a comparação das funções Kernel para algoritmos de suporte de vetor de máquina utilizando estudos com diabetes e obtiveram um resultado de 65% de eficácia com este tipo de função Kernel.

A Figura 17 mostra a relação entre o valor predito e o valor real (normalizados) para o melhor resultado obtido deste método, onde os pontos azuis são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 17 - Melhor resultado SMOReg Polinomial Kernel



Fonte: Autoria própria (2020).

Através da figura acima é possível notar a alta linearidade entre os valores reais e preditos, pelo fato dos pontos estarem próximos e alinhados entre si, que significa que o algoritmo conseguiu prever os valores com boa *performance*. Isto se deve ao alto valor do coeficiente de correlação e pelo baixíssimo valor de MSE (o que se espera em um resultado). Logo, para este conjunto de dados, este método utilizando a função Kernel polinomial apresentou um resultado promissor e satisfatório. A tabela de comparação entre os valores reais e os preditos (sem estarem normalizados) para cada instância, e suas diferenças, podem ser encontrados no apêndice B.

5.1.2.2 Função Kernel de Base Radial (RBF)

Para a função Kernel RBF foram realizados um total de 36 testes envolvendo a variação do parâmetro C e do parâmetro Gama, os resultados dos testes estão representados no Quadro 9:

Quadro 9 - Resultados testes SMOReg RBF Kernel

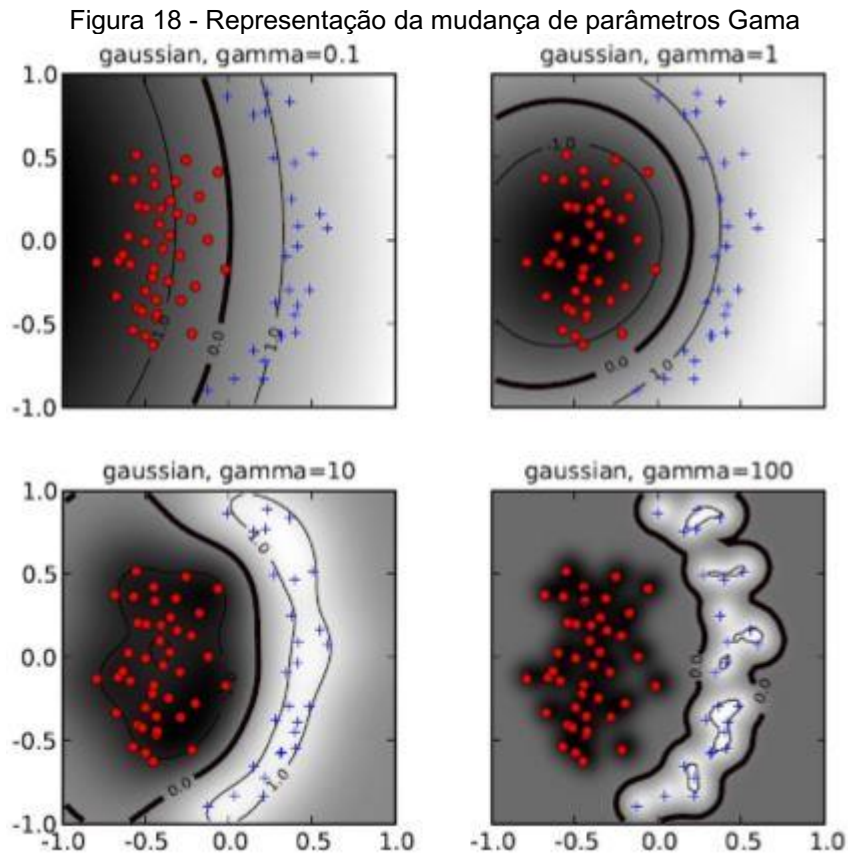
C = 1				
Teste	Gama	R	MSE	Tempo (s)
15	0,001	0,1713	4,4821	0,01
16	0,01	0,1734	4,4796	0,02
17	0,02	0,1756	4,4071	0,02
18	0,05	0,1847	4,4058	0,02
19	0,1	0,1908	4,2762	0,02
20	0,2	0,2248	4,1820	0,02
21	0,5	0,2727	4,1433	0,02
22	0,7	0,2763	4,1156	0,02
23	1	0,2612	4,0950	0,02

C = 10				
Teste	Gama	R	MSE	Tempo (s)
24	0,001	0,1726	4,4808	0,02
25	0,01	0,1750	4,3564	0,02
26	0,02	0,2028	4,2994	0,02
27	0,05	0,2479	4,2382	0,01
28	0,1	0,3082	4,1002	0,02
29	0,2	0,3397	3,9386	0,02
30	0,5	0,4070	3,6271	0,02
31	0,7	0,4227	3,5487	0,01
32	1	0,4256	3,5074	0,02
C = 100				
Teste	Gama	R	MSE	Tempo (s)
33	0,001	0,1732	4,3668	0,02
34	0,01	0,2682	4,1898	0,01
35	0,02	0,3426	4,0477	0,01
36	0,05	0,5216	3,3849	0,02
37	0,1	0,7076	2,5847	0,01
38	0,2	0,8578	1,7876	0,01
39	0,5	0,9519	1,0189	0,01
40	0,7	0,9556	0,8707	0,02
41	1	0,9412	0,9421	0,01
C = 1000				
Teste	Gama	R	MSE	Tempo (s)
42	0,001	0,2591	4,2197	0,02
43	0,01	0,7532	2,4964	0,02
44	0,02	0,9343	1,1580	0,01
45	0,05	0,9961	0,0610	0,01
46	0,1	0,9977	0,0256	0,01
47	0,2	0,9987	0,0130	0,01
48	0,5	0,9915	0,0791	0,01
49	0,7	0,9817	0,1622	0,01
50	1	0,9592	0,3474	0,01

Fonte: Autoria própria (2020).

Para essa função, o teste que obteve o melhor resultado foi o teste 47. Tendo em vista que para os experimentos realizados com a função Kernel polinomial com C igual a 1000 obtiveram o melhor resultado, pode-se dizer que este valor para este parâmetro é o melhor a ser definido. Para o valor de Gama de 0,2, o resultado atingido foi satisfatório e foi corroborado pelo trabalho de Nuanmeesri e Poomhiran (2019), onde os mesmos obtiveram o melhor valor de eficiência do algoritmo ao usarem um

valor de Gama de 0,1 (sua eficácia final foi de 93% com esse Gama, e também utilizando um alto valor do parâmetro C assim como no presente trabalho). Logo, para este conjunto de dados, o comportamento da curva Gaussiana foi o melhor obtido para a separação dos dados visto que com um valor de 0,2 de Gama, a função é ligeiramente curvilínea como mostra a Figura 18:



Fonte: Ben-Hur e Weston (2008).

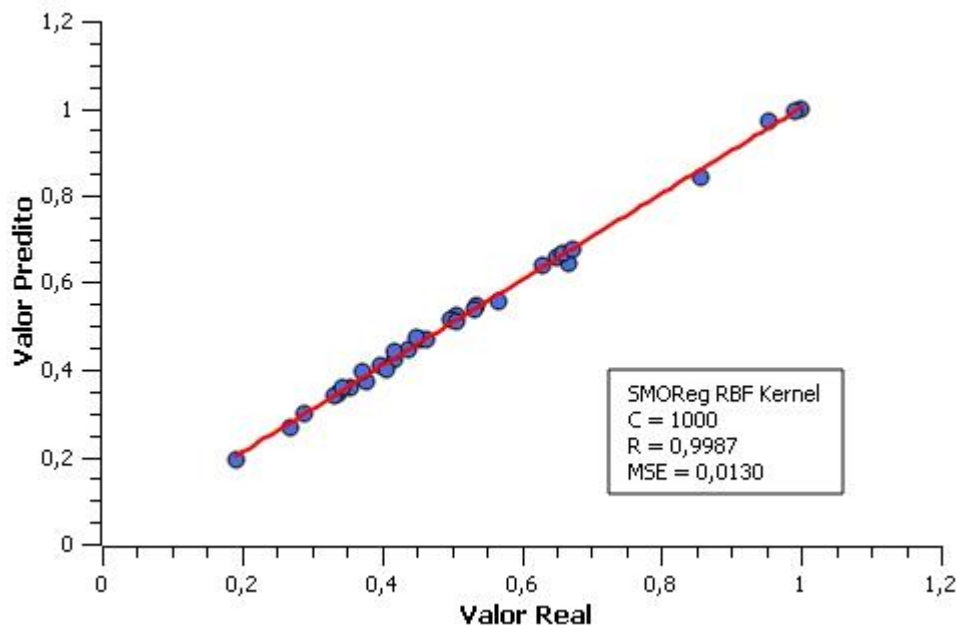
O melhor resultado obtido então (teste 47) apresentou um R de 0,9987 e um MSE de 0,0130 com um tempo computacional exigido de 0,01 segundos. Assim como o resultado da função Kernel Polinomial, este resultado também foi considerado excelente, inclusive maior que o da função Kernel Polinomial. Ao comparar com outros trabalhos de inteligência artificial que utilizam também o mesmo tipo de algoritmo (SMOReg) e de função Kernel, o resultado obtido no presente trabalho atingiu maiores valores. Ao comparar-se o resultado deste trabalho com o resultado obtido por Abakar e Yu (2014), que foi de 98%, assim como os trabalhos de Nuanmeesri e Poomhiran (2019) e de Baitharu et al. (2016) que obtiveram um resultado de 93% e de 77%

respectivamente, o resultado obtido pelo presente trabalho se mostrou melhor, portanto mais recomendado para se aplicar para este tipo de conjunto de dados.

O pior resultado foi obtido no teste 15, com um parâmetro $C = 1$ (assim como na função Kernel polinomial, visto que quanto menor o C , maior a margem de limite mínimo entre a função Kernel e os vetores de suporte), e um parâmetro Gama de 0,001, que indica um comportamento da Gaussiana quase linear, dificultando separar corretamente os vetores de suporte no hiperplano Z . O resultado para o teste 15 obteve um R de 0,1713, com um MSE de 4,4821 e um tempo computacional exigido de 0,01 segundos.

O gráfico da relação entre os valores preditos e os valores reais (normalizados) para o melhor resultado obtido está representado na Figura 19, onde os pontos azuis são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 19 - Melhor resultado SMOReg RBF Kernel



Fonte: Autoria própria (2020).

Assim como para a função Kernel polinomial, houve uma alta linearidade entre os valores preditos e os valores reais devido ao alto valor de R e o baixo MSE através da configuração testada (Isto quer dizer que o algoritmo conseguiu prever o resultado do ponto experimental com uma boa performance, culminando em pontos mais lineares). Portanto pode-se dizer que a função Kernel RBF também apresentou

um resultado satisfatório e ainda mais promissor que a função anterior, sendo mais indicada ainda para este conjunto de dados. A tabela com os resultados individuais (não-normalizados) para cada instância pode ser encontrada no apêndice C.

5.1.2.3 Kernel Universal de Pearson VII (PUK)

Para a função PUK foram realizados um total de 36 testes envolvendo a variação do parâmetro C e dos parâmetros Ômega e Sigma simultaneamente. Os resultados dos testes estão representados no Quadro 10:

Quadro 10 - Resultados testes SMOReg PUK

C = 1					
Teste	Ômega	Sigma	R	MSE	Tempo (s)
51	0,1	1	0,2438	4,1812	0,02
52	0,5	0,5	0,2682	4,0998	0,02
53	0,5	1	0,2663	4,0889	0,02
54	1	1	0,2654	4,0647	0,02
55	2	2	0,2682	4,1099	0,02
56	5	5	0,1915	4,2523	0,02
57	10	10	0,1751	4,3861	0,02
58	20	20	0,1732	4,4618	0,02
59	30	30	0,1715	4,4441	0,02
C = 10					
Teste	Ômega	Sigma	R	MSE	Tempo (s)
60	0,1	1	0,3051	3,8263	0,03
61	0,5	0,5	0,3237	3,7718	0,02
62	0,5	1	0,4275	3,6508	0,02
63	1	1	0,4580	3,5615	0,02
64	2	2	0,4052	3,6092	0,02
65	5	5	0,3216	4,0522	0,02
66	10	10	0,2241	4,2977	0,01
67	20	20	0,1740	4,4066	0,02
68	30	30	0,1709	4,4020	0,02

C = 100					
Teste	Ômega	Sigma	R	MSE	Tempo (s)
69	0,1	1	0,3061	3,8228	0,02
70	0,5	0,5	0,3871	3,5498	0,02
71	0,5	1	0,5740	2,8009	0,02
72	1	1	0,6757	2,2934	0,01
73	2	2	0,9300	1,2190	0,02
74	5	5	0,7306	2,4411	0,02
75	10	10	0,4075	3,8123	0,01
76	20	20	0,2422	4,2551	0,02
77	30	30	0,2164	4,3339	0,02

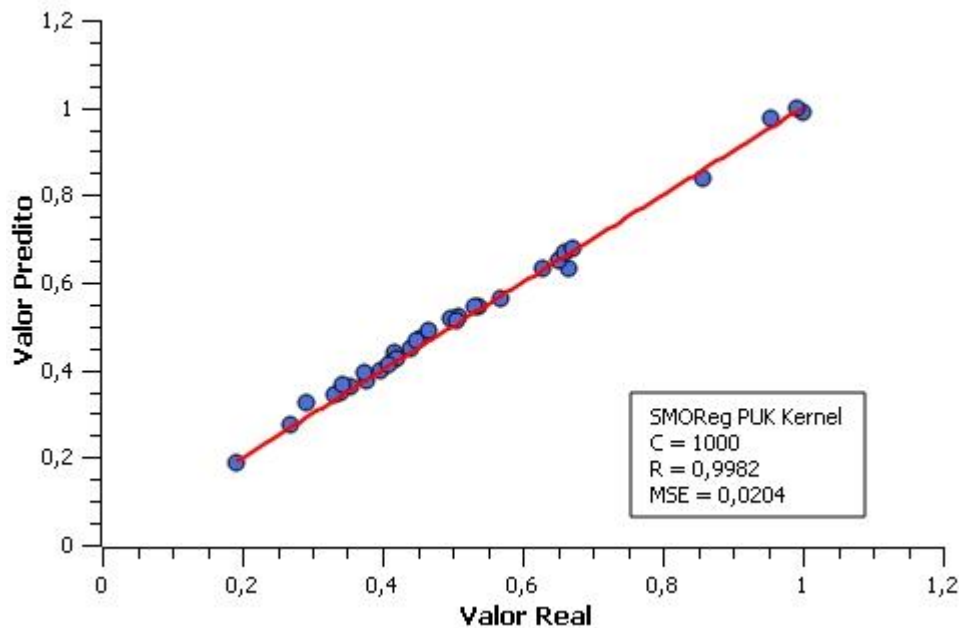
C = 1000					
Teste	Ômega	Sigma	R	MSE	Tempo (s)
78	0,1	1	0,3061	3,8228	0,02
79	0,5	0,5	0,3871	3,5498	0,01
80	0,5	1	0,5804	2,7686	0,01
81	1	1	0,6958	2,1480	0,02
82	2	2	0,9618	0,3377	0,01
83	5	5	0,9982	0,0204	0,01
84	10	10	0,9794	0,5396	0,03
85	20	20	0,6200	3,0636	0,01
86	30	30	0,3848	3,9069	0,01

Fonte: Autoria própria (2020).

Para esta função, a configuração que apresentou o melhor resultado foi o teste 83, onde se ajustou o parâmetro C igual a 1000 e os parâmetros Ômega igual a 5 e Sigma igual a 5, novamente visto que um maior valor de C apresenta melhores resultados, não foi diferente para esta função assim como os valores de ômega e sigma influenciaram no comportamento final da função para a separação dos vetores. Os valores de ômega e sigma foram decididos arbitrariamente, há estudos na área que usam valores diversos para ambos os parâmetros e também atingiram-se bons resultados, como o estudo feito por Abakar e Yu (2014), que utilizaram valores de ômega e sigma como 85 e 9,1 respectivamente, e com isso obtiveram uma eficácia de 98%, assim como o estudo feito por Nuanmeesri e Poomhiran (2019), que obtiveram uma eficácia de 92% utilizando um ômega e sigma de 1 em ambos. O teste 83 apresentou um R de 0,9982 e um MSE de 0,0204, portanto foi novamente um bom resultado para correlacionar os valores preditos e reais. O pior resultado para a configuração testada deste algoritmo foi obtido no teste 68, seu R obtido foi de 0,1709, com um MSE de 4,4020 e um tempo computacional exigido de 0,02 segundos.

A Figura 20 demonstra a relação entre os valores preditos e os valores reais (normalizados) para o melhor resultado obtido está representado abaixo, onde os pontos azuis são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 20 - Melhor resultado SMOReg PUK



Através da figura acima, pode-se analisar que há novamente uma alta linearidade entre os valores preditos e os reais, ou seja, o valor da absorção predita pelo algoritmo está se assimilando ao valor real obtido no experimento. Novamente devido ao alto valor de R e baixo valor de MSE, esta função também se mostrou eficiente para a predição dos valores do conjunto de dados testado, logo uma função mais linear tende a prever mais corretamente os resultados obtidos experimentalmente. A tabela com os resultados individuais (não-normalizados) para cada iteração pode ser encontrada no apêndice D.

Por fim, ao comparar-se o melhor resultado de cada função do método tal, o melhor resultado geral foi o apresentado pela função RBF. Portanto, este seria o mais indicado entre os três pois seu coeficiente de correlação é o maior dentre eles e seu MSE é o menor ($R = 0,0987$ e $MSE = 0,0130$).

5.1.3 Redes Neurais Artificiais

Para o algoritmo das Redes Neurais do tipo MLP, foram realizados 120 testes, onde os seus parâmetros alterados foram o número de épocas (*Epoch/TrainingTime*), o número de camadas internas (máximo = 2) e o número de neurônios em cada camada. Os testes desenvolvidos estão representados nos Quadros 10 e 11:

Quadro 11 - Resultados RNA de uma camada

Número de Camadas = 1					
TrainingTime = 500					
Teste	Num. De Neurônios	R	MSE	Tempo (s)	
87	5	0,9058	1,3891	0,03	
88	10	0,9360	0,5774	0,02	
89	15	0,8661	1,0952	0,01	
90	20	0,9353	2,5616	0,03	
TrainingTime = 1000					
Teste	Num. De Neurônios	R	MSE	Tempo (s)	
91	5	0,9602	1,0445	0,02	
92	10	0,9778	0,2803	0,02	
93	15	0,9012	1,2197	0*	
94	20	0,9621	0,3363	0*	
TrainingTime = 1500					
Teste	Num. De Neurônios	R	MSE	Tempo (s)	
95	5	0,9796	0,3940	0,02	
96	10	0,9794	0,2300	0,02	
97	15	0,9269	1,1546	0,03	
98	20	0,9581	0,3365	0,02	
TrainingTime = 2000					
Teste	Num. De Neurônios	R	MSE	Tempo (s)	
99	5	0,9843	0,3104	0,02	
100	10	0,9831	0,1845	0,02	
101	15	0,9417	0,6775	0,03	
102	20	0,9627	0,2935	0,02	
TrainingTime = 2500					
Teste	Num. De Neurônios	R	MSE	Tempo (s)	
103	5	0,9825	0,2957	0,02	
104	10	0,9839	0,1805	0,02	
105	15	0,9333	0,7086	0,01	
106	20	0,9756	0,2039	0,01	

TrainingTime = 3000

Teste	Num. De Neurônios	R	MSE	Tempo (s)
107	5	0,9712	0,4690	0,02
108	10	0,9909	0,0888	0,02
109	15	0,9467	0,4695	0,02
110	20	0,9821	0,1665	0,02

* = inferior a 0,01 segundo

Fonte: Autoria própria (2020).

Quadro 12 – Resultados RNA de 2 camadas**Número de Camadas = 2****TrainingTime = 500**

Teste	Num. De Neurônios	R	MSE	Tempo (s)
111	5 / 5	0,6279	3,6363	0,02
112	5 / 10	0,8303	1,5763	0,02
113	5 / 15	0,7657	2,0535	0,01
114	5 / 20	0,8082	2,5262	0,02
115	10 / 5	0,6736	2,9002	0,02
116	10 / 10	0,8547	1,9636	0,02
117	10 / 15	0,8845	1,4132	0,01
118	10 / 20	0,8822	1,9193	0,03
119	15 / 5	0,6225	3,2131	0*
120	15 / 10	0,7750	6,4918	0,02
121	15 / 15	0,8748	2,1141	0,03
122	15 / 20	0,8841	1,0457	0,02
123	20 / 5	0,7804	1,9895	0,02
124	20 / 10	0,7990	1,8926	0*
125	20 / 15	0,9179	0,7829	0*
126	20 / 20	0,9261	0,9532	0*

TrainingTime = 1000

Teste	Num. De Neurônios	R	MSE	Tempo (s)
127	5 / 5	0,7928	1,9533	0,01
128	5 / 10	0,8786	1,6631	0,02
129	5 / 15	0,8308	1,4634	0,01
130	5 / 20	0,8837	1,3731	0,01
131	10 / 5	0,8041	1,4244	0*
132	10 / 10	0,8770	1,1088	0,02
133	10 / 15	0,9435	0,5467	0,02
134	10 / 20	0,9559	0,4997	0,02
135	15 / 5	0,9038	1,4571	0,02
136	15 / 10	0,9228	0,6183	0,01
137	15 / 15	0,9219	0,9008	0,02
138	15 / 20	0,9463	0,4275	0,01

139	20 / 5	0,9359	0,6466	0,02
140	20 / 10	0,8898	0,8593	0,02
141	20 / 15	0,9455	0,5001	0,02
142	20 / 20	0,9549	0,3997	0,02

TrainingTime = 1500

Teste	Num. De Neurônios	R	MSE	Tempo (s)
143	5 / 5	0,8228	2,4847	0,02
144	5 / 10	0,9008	1,0252	0,02
145	5 / 15	0,8207	1,3995	0,02
146	5 / 20	0,8916	0,9390	0,01
147	10 / 5	0,8127	1,4371	0,02
148	10 / 10	0,8868	0,9308	0,02
149	10 / 15	0,9723	0,2533	0,02
150	10 / 20	0,9359	0,5433	0,01
151	15 / 5	0,8312	1,2778	0,01
152	15 / 10	0,9140	0,6961	0,01
153	15 / 15	0,9502	0,3890	0,02
154	15 / 20	0,9563	0,4072	0,02
155	20 / 5	0,9397	0,5992	0*
156	20 / 10	0,9097	1,0685	0,02
157	20 / 15	0,9581	0,6172	0,02
158	20 / 20	0,9633	0,3078	0,02

TrainingTime = 2000

Teste	Num. De Neurônios	R	MSE	Tempo (s)
159	5 / 5	0,7808	1,5977	0,01
160	5 / 10	0,9474	0,5592	0,01
161	5 / 15	0,7976	1,5630	0,01
162	5 / 20	0,8608	1,2463	0,02
163	10 / 5	0,8490	1,2452	0,02
164	10 / 10	0,8212	1,3048	0,02
165	10 / 15	0,9577	0,3492	0*
166	10 / 20	0,9205	0,9019	0,02
167	15 / 5	0,8550	1,0889	0,01
168	15 / 10	0,9350	0,5618	0,01
169	15 / 15	0,9566	0,3436	0,01
170	15 / 20	0,9652	0,3528	0,02
171	20 / 5	0,9373	0,6390	0,02
172	20 / 10	0,9540	1,2717	0,02
173	20 / 15	0,9796	0,2161	0,02
174	20 / 20	0,9734	0,2726	0,02

TrainingTime = 2500

Teste	Num. De Neurônios	R	MSE	Tempo (s)
175	5 / 5	0,8515	1,2515	0,01
176	5 / 10	0,9537	0,4513	0,01
177	5 / 15	0,8747	0,9512	0*
178	5 / 20	0,8942	0,8212	0*
179	10 / 5	0,8490	1,8312	0*
180	10 / 10	0,8725	1,4886	0*
181	10 / 15	0,9533	0,5573	0*
182	10 / 20	0,9269	0,6841	0,02
183	15 / 5	0,8932	0,9851	0,02
184	15 / 10	0,9490	0,4438	0*
185	15 / 15	0,9617	0,3341	0,02
186	15 / 20	0,9693	0,2609	0,02
187	20 / 5	0,9331	0,6292	0*
188	20 / 10	0,9621	0,4007	0,01
189	20 / 15	0,9864	0,1549	0*
190	20 / 20	0,9770	0,1827	0,01

TrainingTime = 3000

Teste	Num. De Neurônios	R	MSE	Tempo (s)
191	5 / 5	0,9348	0,5223	0,01
192	5 / 10	0,9441	0,4448	0*
193	5 / 15	0,8260	1,5896	0,01
194	5 / 20	0,9297	0,7707	0,02
195	10 / 5	0,8577	1,5954	0*
196	10 / 10	0,8466	1,2481	0*
197	10 / 15	0,9478	0,4702	0,01
198	10 / 20	0,9357	0,5778	0,02
199	15 / 5	0,8920	1,0445	0,02
200	15 / 10	0,9492	0,4834	0,02
201	15 / 15	0,9581	0,3601	0,02
202	15 / 20	0,9578	0,3652	0,02
203	20 / 5	0,9185	0,7211	0,02
204	20 / 10	0,9652	0,4579	0,02
205	20 / 15	0,9843	0,1461	0,02
206	20 / 20	0,9765	0,1871	0,01

* = inferior à 0,01 segundo

Fonte: Autoria própria (2020).

Para as Redes Neurais Artificiais, a configuração que apresentou o melhor resultado de eficácia foi a apresentada no teste 108. O valor obtido para o R foi de 0,9909 e o MSE obtido foi de 0,0888 com um tempo computacional exigido de 0,02

segundos. O pior resultado obtido foi apresentado no teste 119, o coeficiente de correlação juntamente com o MSE foram de 0,6225 e 3,2131 respectivamente. Esta configuração necessitou de um tempo computacional inferior a 0,01 segundos para ser montada. O melhor valor de R encontrado no presente trabalho em testes com as RNAs, se comparado com trabalhos envolvendo o uso de RNA's para predição e classificação de variáveis envolvendo carcaça de frango, apresentou um resultado melhor quando comparado ao estudo feito por Klassen (2008), o qual obteve um resultado final em torno de 95% para a predição da massa e temperatura final da carcaça.

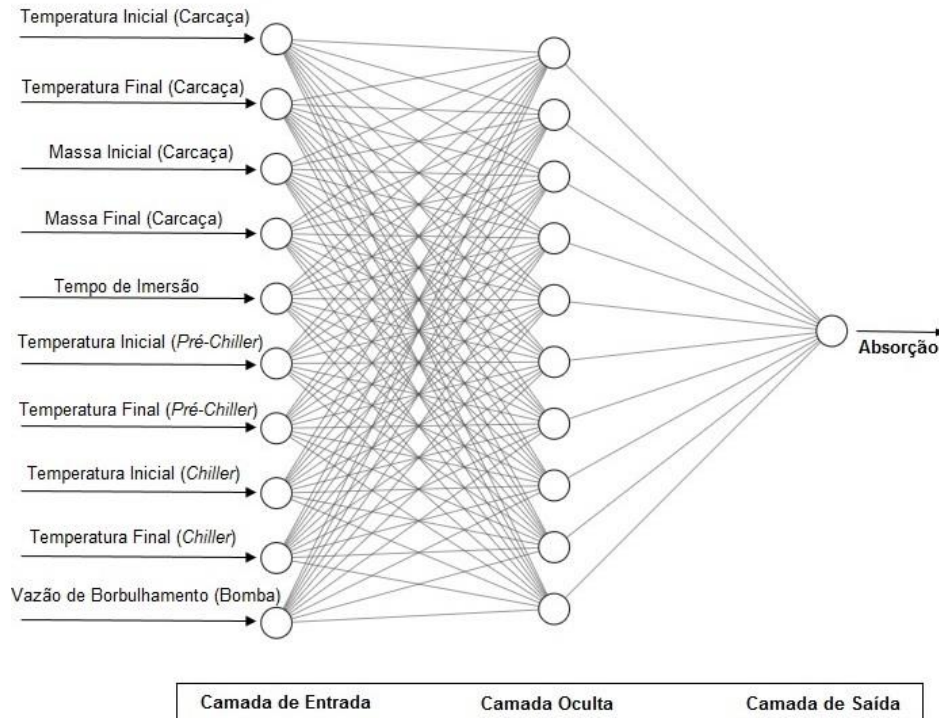
Para a temperatura, por outro lado, o método das RNA's não obteve um resultado satisfatório, visto que este variou entre 8% e 42%, resultado inferior que o apresentado pela predição da massa.

Ao comparar o resultado do presente trabalho com o trabalho feito por Saritas e Yasar (2019) que estudaram a comparação de performance entre diferentes algoritmos para classificação de dados, e que apresentou uma eficácia de 86%, tem-se que o resultado apresentado pelo presente trabalho se mostrou sendo mais eficiente.

Assim como ao se comparar com o resultado do estudo feito por Silva e Schimidt (2015) que trabalhou com a análise de componentes principais (PCA) na modelagem de oxigênio dissolvido, e que apresentaram resultados de 46% e 76% (após o uso de PCA), tem-se que este resultado também foi inferior ao do presente trabalho.

A configuração da rede treinada que obteve o melhor resultado, está representada na Figura 21:

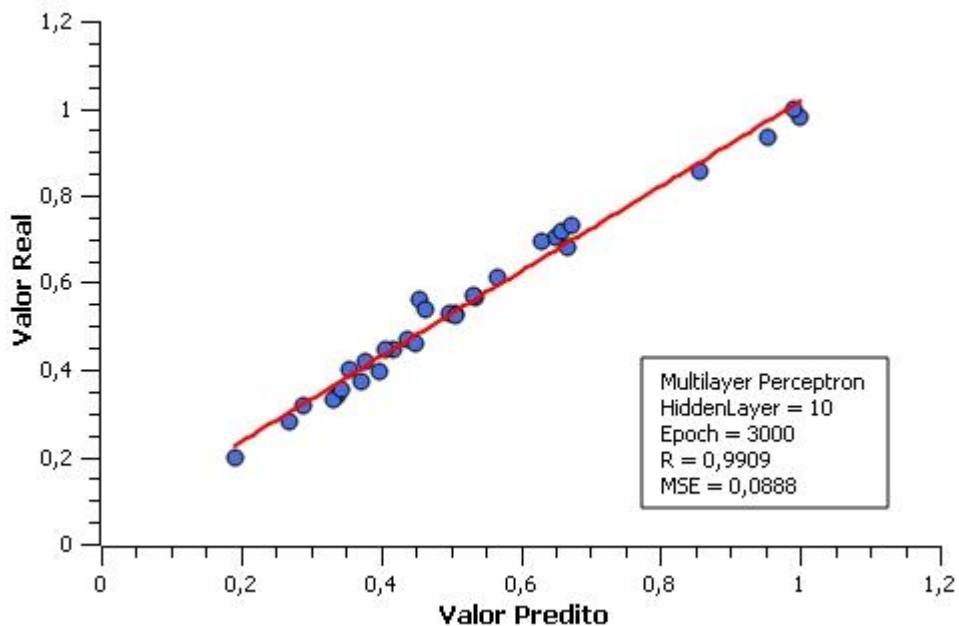
Figura 21 - Esquemática da rede neural com melhor configuração



Fonte: Autoria própria (2020).

A relação entre os valores preditos e os valores reais para o melhor resultado obtido está representado na Figura 22, onde os pontos azuis são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 22 - Melhor resultado RNA



Fonte: Autoria própria (2020).

Através da Figura 22, é possível analisar que embora haja um coeficiente de correlação menor que o algoritmo anterior, ainda há uma boa linearidade, pois seu R ainda é considerado bom, assim como seu MSE. Isso indica que o algoritmo apresentou uma boa capacidade de prever os valores experimentais através de sua aplicação no conjunto de dados. Portanto, da mesma maneira que o algoritmo SMOReg, as RNA's são recomendáveis para se trabalhar com este tipo de conjunto de dados estudado. A tabela com os resultados individuais (não-normalizados) para cada instância pode ser encontrada no apêndice E.

O que pode ser percebido pelos resultados das RNAs através das Tabelas 10 e 11, é que este método apresentou o maior número de resultados próximos entre si, assim como resultados elevados em sua maioria, então é possível afirmar com isso que o método das RNA's é mais constante em suas análises, evitando valores de saída muito discrepantes entre os algoritmos analisados por este trabalho como verificado nos algoritmos anteriores.

5.2 DEFINIÇÃO DO MELHOR MODELO

Após a análise de cada algoritmo, é necessário definir qual dentre eles obteve os melhores resultados. Foram, então, analisados os resultados da melhor configuração de cada um dos algoritmos, os quais estão dispostos no Quadro 13:

Quadro 13 - Comparação dos melhores resultados gerais

Teste	Algoritmo	R	MSE	Tempo (s)
1	<i>K-Star</i>	0,5011	3,5348	0,08
13	SMOReg	0,9987	0,0130	0,01
108	RNA	0,9909	0,0888	0,02

Fonte: Autoria própria (2020).

Através da tabela acima é possível perceber que dentre os algoritmos utilizados neste trabalho, o melhor dentre eles foi o SMOReg, pois apresentou o maior R (pouco acima da RNA), porém com um MSE menor, o que acarreta em menos erros durante seus testes.

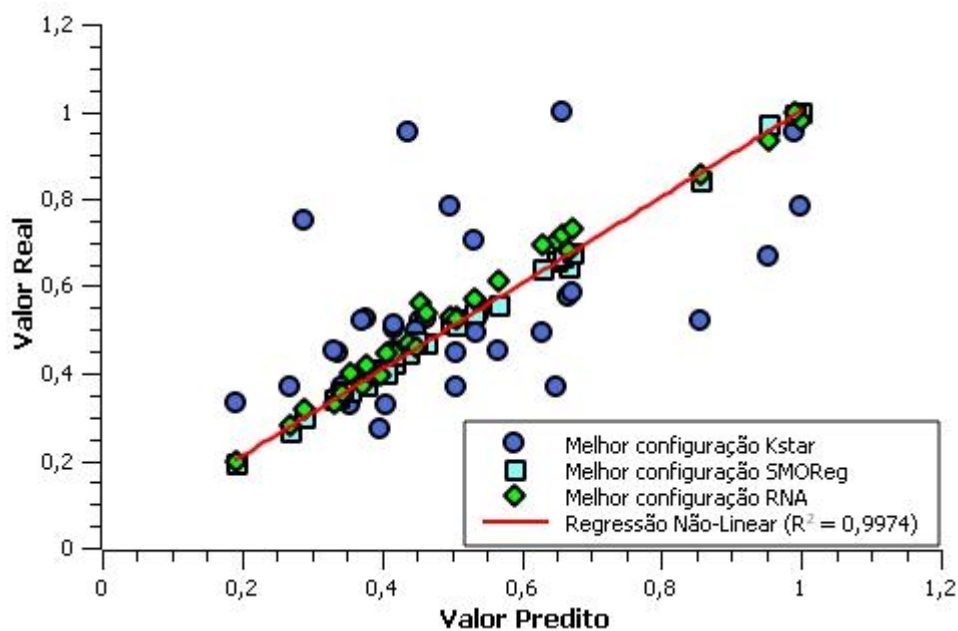
Um dos motivos do SMOReg ter apresentado melhores resultados foi o fato dele ser um algoritmo que trabalha com várias funções diferentes assim como é

possível alterar o comportamento dessas funções dentro do hiperplano através da variação de seus parâmetros. Porém o método SMOReg não apresenta resultados constantes em seus testes, visto que uma mínima mudança em um de seus parâmetros pode diminuir drasticamente o valor do R. Portanto, o melhor teste quando se trata de constância se mostrou o algoritmo das RNA's, pois seus resultados estavam muito próximos entre si para qualquer mudança de parâmetro em seus testes.

O algoritmo que obteve o pior resultado foi o *K-Star*. Correlacionando seu resultado com trabalhos já realizados utilizando o mesmo algoritmo, como o trabalho feito por Diniz et al. (2013) e por Hernández (2015) que apresentaram eficiências melhores que outros algoritmos aqui testados, pode-se chegar à conclusão de que o *K-Star* não é o algoritmo mais indicado para se trabalhar com este tipo de conjunto de dados numéricos (visto que o mesmo algoritmo apresentou bons resultados envolvendo conjunto de dados de imagens), podendo ser devido ao número de variáveis de entrada, ao tamanho do conjunto dos dados ou por se tratar de um problema de regressão cujo o algoritmo apresentaria uma dificuldade em se trabalhar.

Os valores reais e preditos normalizados para os três algoritmos juntos estão representados na Figura 23, onde os diferentes pontos são os resultados obtidos e a reta vermelha é a reta de normalização entre os pontos:

Figura 23- Junção dos melhores resultados dos algoritmos



Fonte: Autoria própria (2020).

Como já foi discutido anteriormente, é possível ver que o comportamento do SMOReg é o mais linear, seguido pelas RNA's que também apresentaram um bom comportamento, e o do *K-star* o mais destoante do restante. Dentre os fatores que podem ser levados em conta para esses resultados estão o conjunto de dados, a finalidade do algoritmo e a facilidade do mesmo em se trabalhar com o problema requerido.

5.3 CLASSIFICAÇÃO DAS VARIÁVEIS MAIS INFLUENTES NA ABSORÇÃO DE ÁGUA NAS CARÇAÇAS DE FRANGO

Utilizando as configurações do melhor resultado e pior para cada algoritmo afim de compará-los, foram realizados então os testes para a seleção das três variáveis mais influentes no processo de absorção de água pela carcaça através do próprio software do WEKA. Os valores obtidos para estes testes estão representados primeiramente para cada algoritmo e então estes serão comparados entre si.

Alguns trabalhos na literatura estudaram a relação das variáveis com a absorção de água e seus resultados serão utilizados como base para corroborar com os resultados apresentados no presente trabalho, são estes trabalhos: “Uso de redes neurais artificiais para a modelagem da temperatura e da retenção de água no processo de resfriamento de carcaças de frangos por imersão” realizado por Klassen (2008), “Estudo do resfriamento de carcaças de frango em *chiller* de imersão em água” realizado por Carciofi (2005), “Modeling the water uptake by chicken carcasses during cooling by immersion” realizado por Martins et al. (2011), “Análise dos fatores que afetam a temperatura e absorção de água de carcaças de frango em *chiller* industrial” realizado por Sant'anna (2008) e “Water uptake by poultry carcasses during cooling by water immersion” realizado por Carciofi e Laurindo (2007).

As principais variáveis encontradas nos trabalhos citados anteriormente que mais influenciam no processo foram: Massa inicial da carcaça, temperatura inicial do pré-*chiller*, vazão de borbulhamento, temperatura inicial/final no *chiller* e tempo de resfriamento. Os motivos pelos quais estes trabalhos citados apresentam estas variáveis como as mais influentes, estão representados no Quadro 14:

Quadro 14 - Explicação física das variáveis mais influentes

Variável	Explicação Física de sua influência
Massa inicial da carcaça	A massa inicial afeta a absorção através de seu volume de controle, pois quanto maior este for, mais lentamente seu processo de resfriamento ocorrerá e, portanto, podendo ser prejudicial ao produto
Temperatura Inicial do Pré-Chiller	A temperatura inicial do pré- <i>chiller</i> tem sua influência por ser a primeira etapa do processo, visto que os 10 primeiros minutos influenciam mais durante toda etapa do resfriamento, será nesta etapa que ocorrerá a maior absorção da carcaça
Vazão de Borbulhamento	Melhora a circulação das carcaças nos tanques, diminuindo caminhos preferenciais dos mesmo e aumentando o grau de agitação do meio fazendo com que haja uma melhor transferência de energia
Temperatura Inicial/Final no Chiller	Acarreta a temperatura final com que o produto sairá, que deverá ser menor que 4°C, respeitando a legislação e também evitando fatores microbiológicos que possam afetar o produto.
Tempo de Resfriamento	O tempo de resfriamento condiz com o quanto a carcaça absorverá de água, quanto maior o tempo da carcaça no tanque, maior será seu valor de absorção, porém a carcaça não deve passar o valor limite imposto pela legislação.

Fonte: Autoria própria (2020).

A pressão da coluna d'água, a temperatura ambiente e a quantidade de gelo nos tanques também foram variáveis consideradas no processo pela literatura, mas não foram incluídas no presente trabalho.

Os valores do coeficiente ao lado da variável, são índices de importância para o resultado final da classificação, ou seja, quanto maior o índice mais importante é a variável. Este índice varia de 0 a 1 e o limite de aceitação definido no presente trabalho foi o menor valor do coeficiente dentre os resultados previstos das melhores configurações, que foi de 0,27 (variável de tempo na melhor configuração da RNA). Valores abaixo de 0,27 foram classificados como abaixo do limite de aceitação e, portanto, foram marcados com a cor amarela.

5.3.1 K-Star

Para o algoritmo *K-Star* foram utilizadas as configurações dos testes 1 e 6, que foram os testes que apresentaram o melhor e o pior resultado respectivamente. Os resultados de suas classificações estão representados no Quadro 15:

Quadro 15 - Resultados classificação *K-Star*

Melhor Configuração		Pior Configuração	
Coeficiente	Variável	Coeficiente	Variável
0,5538	Tempo	0,0022	Bomba
0,5479	Tfpc	0,0016	Tfc
0,5380	Tfc	0,0008	Tic
0,5359	Tic	0,0003	Ti
0,5200	Tipc	0,0003	Tempo
0,5124	Bomba	0,0002	Tipc
0,2537	Mf	0,0002	Tfpc
0,1973	Ti	0,0000	Mi
0,1661	Mi	-0,0001	Mf
0,0982	Tf	-0,0002	Tf
Legenda:			
Previsto pela literatura e acima do limite de aceitação			
Previsto pela literatura mas abaixo do limite de aceitação			
Não previsto pela literatura			

Fonte: Autoria própria (2020).

Através da tabela acima, é possível analisar que apesar da predição dos valores da absorção de água do algoritmo *K-Star* não obterem um bom resultado, o mesmo obteve êxito em prever a maioria das variáveis mais importantes segundo a literatura, visto que classificou 2 das 3 variáveis corretas dentre as previstas, suas variáveis mais influentes foram o tempo de permanência da carcaça nos tanques, a temperatura final do *pré-chiller*, e a temperatura final do *chiller*, duas delas comprovadas pela literatura que apresentam maior influência. Segundo as variáveis apresentadas pelo algoritmo, deve-se então priorizar no controle do tempo em que as aves estão passando pelo processo de imersão, assim como realizar o controle da temperatura do *pré-chiller* para que este não esteja demasiadamente alto ou baixo demais e a temperatura final do *chiller*, para que o produto final não saia da temperatura exigida. Analisando o resultado da sua pior configuração, embora este apresentasse variáveis previstas como influentes pela literatura, seu índice ficou muito

abaixo do limite de aceitação, o que invalidou a confiança do mesmo para esta configuração.

5.3.2 SMOReg

Para o algoritmo SMOReg foram utilizadas as configurações dos testes 47 e 68, que apresentaram o melhor e pior resultado para a predição respectivamente. Os testes para a classificação com estas configurações estão representados no Quadro 16:

Quadro 16 - Resultados classificação SMOReg

Melhor Configuração		Pior Configuração	
Coefficiente	Variável	Coefficiente	Variável
0,3835	Tfc	0,3014	Bomba
0,3749	Bomba	0,1172	Tfc
0,3618	Tic	0,1119	Tic
0,2637	Tempo	0,0458	Tipc
0,2588	Tfpc	0,0204	Tempo
0,1556	Mi	0,0115	Ti
0,0753	Ti	0,0067	Mi
0,0646	Tipc	-0,0135	Tfpc
0,0635	Tf	-0,0163	Tf
0,0047	Mf	-0,0419	Mf
Legenda:			
Previsto pela literatura e acima do limite de aceitação			
Previsto pela literatura mas abaixo do limite de aceitação			
Não previsto pela literatura			

Fonte: Autoria própria (2020).

Através da tabela acima, é possível perceber que da mesma forma que o algoritmo SMOReg apresenta um bom desempenho para a predição da absorção, ele também apresentou para a classificação das variáveis, pois classificou corretamente três das variáveis mais influentes previstas pela literatura e todas estas acima do limite de aceitação. As variáveis mais influentes segundo este algoritmo foram a temperatura final e inicial do *chiller* e a vazão de borbulhamento no tanque. Segundo a classificação deste algoritmo, então, deve-se priorizar no processo a temperatura inicial e final do *chiller* para controlar a temperatura final do produto (prioritariamente deve haver um gradiente positivo de temperatura), a vazão da bomba para controlar o borbulhamento para evitar aglomeração de carcaças, caminhos preferenciais e

também melhorar a circulação do meio refrigerante. Analisando o resultado da sua pior configuração, o mesmo algoritmo também apresentou as três variáveis anteriores como sendo as mais influentes, porém com sua ordem de influência alterada, e apenas uma destas variáveis com o índice aceitável, as outras apresentaram um índice com uma certa distância do valor mínimo para aceitação. Isto reforça a afirmação de que ele é um método que apresenta resultados muito discrepantes uns dos outros em certas ocasiões.

5.3.3 Redes Neurais Artificiais

Para o algoritmo das RNA's foram utilizadas as configurações dos testes 108 e 119, que foram os testes que apresentaram o melhor e o pior desempenho respectivamente para a predição da absorção da água. Os resultados da classificação para estas configurações estão representados no Quadro 17:

Quadro 17 - Testes classificação Redes Neurais

Melhor Configuração		Pior Configuração	
Coefficiente	Variável	Coefficiente	Variável
0,368	Tfc	0,357	Bomba
0,321	Bomba	0,340	Tfc
0,270	Tempo	0,242	Tic
0,268	Tic	0,236	Tempo
0,257	Ti	0,236	Mi
0,252	Mi	0,232	Mf
0,236	Mf	0,223	Ti
0,232	Tf	0,220	Tf
0,213	Tfpc	0,220	Tfpc
0,188	Tipc	0,209	Tipc
Legenda:			
Previsto pela literatura e acima do limite de aceitação			
Previsto pela literatura mas abaixo do limite de aceitação			
Não previsto pela literatura			

Fonte: Autoria própria (2020).

Através da tabela acima, é possível analisar que as redes neurais também apresentaram um bom trabalho prevendo as variáveis mais influentes no processo, assim como na predição dos valores da absorção, visto que conseguiu predizer todas as 3 variáveis mais influentes com sucesso. Suas variáveis mais influentes foram a

temperatura final do *chiller*, a vazão de borbulhamento e o tempo de imersão. Segundo este algoritmo então, deve-se realizar um controle maior sobre a temperatura do *chiller* acarretando então na transferência de calor para o produto final, controlar a vazão da bomba assim como apresentado no algoritmo SMOReg e também controlar o tempo de imersão para não ultrapassar o limite necessário. Outro ponto para se destacar ao analisar a tabela, é que se percebe que a variável do tempo de imersão apresentou um valor de coeficiente com uma leve diferença da variável subsequente que seria a temperatura inicial do *chiller*, esta que também é uma variável prevista pela literatura como sendo influente. Analisando sua pior configuração, percebe-se que esta apresentou o melhor desempenho dentre os três algoritmos, predizendo duas das três variáveis influentes com um valor de aceitação acima do limite mínimo, a última variável também foi predita corretamente, porém seu valor de aceitação foi menor que o mínimo exigido. Portanto a rede apresentou uma classificação perfeita das variáveis, ela apresentou um índice de classificação constante de suas variáveis, onde se mostrou um método eficiente e confiável.

5.4 COMPARAÇÃO E SELEÇÃO DO ALGORITMO MAIS EFICIENTE

Através das análises individuais discutidas anteriormente, para as melhores configurações testadas, todos os algoritmos utilizados no presente trabalho obtiveram um bom resultado em avaliar as 3 variáveis mais importantes em relação à absorção da água na carcaça. O algoritmo SMOReg e as Redes Neurais previram com eficiência as variáveis: a vazão na bomba, a temperatura no *chiller*, e o tempo de imersão, que corrobora com o que é apresentado pela literatura, discutida anteriormente. O algoritmo *K-Star* previu corretamente duas das três variáveis mais influentes, resultado este que ainda se mostrou aceitável. Porém, a baixa confiabilidade demonstrada pelas configurações testadas para a predição da absorção (visto que apresentou um baixo R e um alto MSE), faz do *K-Star* um algoritmo menos confiável do que os outros, acarretando na falta de integridade de seus resultados para a predição.

Através dos testes envolvendo as piores configurações de cada algoritmo, os que mais se destacaram também foram as RNA's e o SMOReg. A primeira apresentou 3 variáveis condizentes com a literatura, duas destas variáveis com valores acima do limite de aceitação, reforçando então a afirmação de que este algoritmo é o mais

constante entre os testados. O SMOReg conseguiu classificar todas as variáveis condizentes com a literatura, porém com apenas uma variável acima do limite de aceitação, o que a torna um método menos constante do que o previamente abordado. O algoritmo *K-Star* conseguiu prever corretamente todas as variáveis condizentes com a literatura, porém seu índice extremamente baixo não demonstrou confiabilidade em seus resultados.

Portanto, comparando-se os três algoritmos, o que apresentou o melhor desempenho (considerando apenas o valor do coeficiente de importância da variável) foi o algoritmo SMOReg, sendo este então, o mais recomendado para se realizar a classificação das variáveis do que o algoritmo das RNA's e *K-Star*. Isto pode ser explicado pela facilidade deste algoritmo em se trabalhar com soluções simultâneas de subproblemas da função objetivo, aplicados a um conjunto de dados com diversas variáveis de entrada.

5.5 COMPARAÇÃO COM A LITERATURA

Através de pesquisas feitas pelo autor em trabalhos na literatura, foi encontrado um baixo número de trabalhos que envolvam predição da absorção de água utilizando inteligência artificial. É possível destacar o trabalho de Klassen (2008) que utilizou as Redes Neurais aplicadas ao modelo de Carciofi e Laurindo (2007) e obteve o melhor resultado de correlação com um valor de 99,44% para a retenção de massa no *pré-chiller*. Esse resultado se aproximou bastante da correlação obtida no presente trabalho, que serve para ressaltar que o uso das RNAs se aplica em diversos casos com uma boa performance. Para os outros algoritmos não foram encontrados trabalhos envolvendo processo de abate industrial de frango, tampouco predição da absorção de água do mesmo.

Em relação à classificação de variáveis mais impactantes no processo, alguns estudos podem ser destacados, como o estudo feito por Carciofi (2005) utilizando modelagens matemáticas provenientes de balanços de massa e energia para analisar o comportamento do processo através de diferentes parâmetros, o estudo feito por Klassen (2008) que utilizou as Redes Neurais também para classificar as variáveis mais impactantes no processo, e por fim o próprio trabalho de Sant'anna (2008) que testou no próprio ambiente industrial, o impacto de cada componente do processo,

onde classificou as variáveis mais importantes através de cada resultado obtido na linha de produção.

Contudo, não foram obtidos por nenhum meio trabalhos em que se relacionem o processo de classificação das variáveis mais influentes com os presentes métodos computacionais estudados neste trabalho.

6 CONCLUSÃO

No presente trabalho, o conjunto de dados obtidos da tese de Sant'anna (2008) foram previamente normalizados para melhor adaptação aos algoritmos, sendo estes: *K-star*, SMOReg e Redes Neurais Artificiais. No total foram realizados 206 testes envolvendo os três algoritmos, onde separou-se o melhor teste de cada algoritmo.

O melhor resultado do algoritmo SMOReg apresentou uma correlação de 99,87%, um erro quadrático médio de 0,0130 e tempo computacional de 0,01 segundos. A função Kernel utilizada foi a função RBF, com o valor de parâmetro gama sendo escolhido como 0,2. O resultado foi considerado bom, visto que aproximou ao valor de 100% de correlação, o que indica que o algoritmo fez o trabalho da predição dos valores de maneira satisfatória. Quanto à classificação dos dados, o algoritmo que classificou mais corretamente as variáveis influentes de acordo com a literatura foi também o algoritmo SMOReg, dentre as variáveis classificadas estavam: Temperatura Final e Inicial do *Chiller* e a Vazão da Bomba.

O melhor resultado das Redes Neurais Artificiais apresentou uma correlação de 99,09%, um erro quadrático médio de 0,0888 e um tempo computacional de 0,02 segundos. Sua arquitetura foi de uma camada oculta apenas. O número de neurônios utilizados na camada oculta foi de 10 neurônios, que coincide com o número de variáveis de entrada da rede, números maiores e menores de neurônios pioraram sua performance. O resultado apresentado pelo algoritmo foi considerado bom, já que seu valor de correlação também se aproximou de 100%. As Redes Neurais também apresentaram um bom comportamento na classificação das variáveis (embora estas tenham apresentado coeficientes menores que no SMOReg), que classificou corretamente as variáveis: Temperatura Final do *Chiller*, Vazão da Bomba e Tempo de Resfriamento.

O melhor resultado do algoritmo *K-star* apresentou uma correlação de 50,11% dos valores, e um erro médio quadrado de 3,53 e um tempo de 0,08 segundos. Seu parâmetro de GlobalBlend utilizado foi zero. O resultado obtido não foi suficiente para aceitação, visto que uma correlação de 50,11% é um valor muito baixo se comparado com outros trabalhos envolvendo o mesmo algoritmo e que apresentaram resultados próximos de 90% de correlação. Para a classificação dos dados, o algoritmo *K-star*

novamente não apresentou um resultado considerado satisfatório, prevendo incorretamente uma variável (Temperatura final do Pré-*Chiller*) dentre as três. Logo, não se considerou um algoritmo propenso a se trabalhar com este conjunto de dados, ao contrário dos outros dois estudados.

Dentre os três algoritmos, dois se mostraram promissores para o estudo em questão (SMOReg e Redes Neurais Artificiais). O algoritmo que prevaleceu ao final do trabalho foi o algoritmo SMOReg, tanto para predição, quanto para classificação. Entretanto, é importante atentar-se para o algoritmo das Redes Neurais que também apresentou boas correlações. Dentre eles, o único algoritmo que não apresentou resultado satisfatório foi o *K-star*. Porém, é necessário considerar também a natureza do conjunto de dados, e que estes também podem não se aplicar de maneira satisfatória em determinado algoritmo, fazendo-o ter uma performance abaixo do esperado para sua capacidade.

Por fim é possível afirmar que o presente trabalho contribuiu de forma inovadora para a área de abate de frango industrial e da inteligência artificial, utilizando-se de algoritmos que não foram previamente testados para a predição de valores da absorção de água na carcaça e também na classificação de variáveis envolvendo todo o processo da mesma. Assim, foi possível obter uma base de dados maior em uma área de pesquisa ainda muito pouco difundida e trabalhada, para que futuramente possa haver um maior aproveitamento industrial dos artifícios tratados no presente trabalho.

- Sugestões para trabalhos futuros:
 - Trabalhar com um conjunto maior de dados coletados industrialmente, para analisar se há uma melhora na performance dos algoritmos;
 - Trabalhar com outros algoritmos já existentes na plataforma WEKA, assim como trabalhar com uma configuração diferente dos algoritmos testados.
 - Aplicar métodos de otimização aos algoritmos e ao conjunto de dados, fazendo possivelmente uma redução de variáveis quando necessária.
 - Aplicar o uso de aprendizado de máquina para estudar o racionamento do uso da água no processo de abate de frango, de acordo com o trabalho sugerido por Bailone e Roça (2017).

REFERÊNCIAS

- AL - SHAMERY, E. S.; AL – GASHAMY, H. A. Enhanced Evolutionary Sequential Minimal Optimization Model for Inflation Prediction. **International Journal of Engineering & Technology**, v. 7, n. 4.19, p. 788, 2018.
- ABAKAR, K. A. A.; YU, C. Performance of SVM based on PUK kernel in comparison to SVM based on RBF kernel in prediction of yarn tenacity. **Indian Journal of Fibre and Textile Research**, v. 39, n. 1, p. 55–59, 2014.
- ABDELKHALEK, H. S.; MEDHAT, H.; ZIEDAN, I.; AMAL, M. Simulation and prediction for a satellite temperature sensors based on artificial neural network. **Journal of Aerospace Technology and Management**, v. 11, p. 1–14, 2019.
- AMRAEL, S.; MEHDIZADEH, S. A.; SALARI, S. Broiler weight estimation based in machine vision an artificial neural network. **British Poultry Science**, p. 200–205, 2017.
- ANDERSEN, B. C. Engineering Aspect of Poultry Chilling. **Scandinavian Refrigeration**, p. 60–64, 1974.
- ARASHIRO, O. **A história da avicultura do Brasil**. São Paulo: Ed. Gessulli, 1989.
- ARAÚJO, M. Agronegócios: conceitos e dimensões. **Fundamentos de agronegócios**, p. 160, 2007.
- ARTERO, A. O. **Inteligência artificial: teoria e prática**. São Paulo: Livraria da Física, 2009.
- ASSOCIAÇÃO BRASILEIRA DE PROTEÍNA ANIMAL - ABPA. **Relatório Anual**. 2020.
- BAILONE, R. L.; ROÇA, R. O. Tendências no processamento de frangos de corte: Uso racional da água. **Engenharia Sanitaria e Ambiental**, v. 22, n. 1, p. 65–72, 2017.
- BAITHARU, T. R.; SUBHENDU, K. P.; DHAL, S. K. Comparison of Kernel Selection for Support Vector Machines Using Diabetes Dataset. **Journal of Computer Sciences and Applications**, Vol. 3, 2016, Pages 181-184, v. 3, n. 6, p. 181–184, 2016. Disponível em: <<http://pubs.sciepub.com/jcsa/3/6/14/>>. .
- BARZOTTO, P. C. Estudo De Riscos Ambientais Na Indústria Frigorífica – Processos Abate Frango. , p. 69, 2013.
- BASTIANI, M.; SANTOS, J. A. A.; SCHMIDT, C A P SEPULVEDA, G. P. L. Application of data mining algorithms in the management of the broiler production. **Geintec**, 2018.
- BEN-HUR, A.; WESTON, J. A User's Guide to Support Vector Machines. **Data**

Mining Techniques for the Life Sciences, Methods in Molecular Biology. v. 609, 2010. LLC.

BIRAN, O.; COTTON, C. Explanation and Justification in Machine Learning: A Survey. **IJCAI Workshop on Explainable Artificial Intelligence (XAI)**, p. 8–13, 2017.

BISHOP, E. Developments in The Chilling of Poultry. **Refregeration and Air Conditioning**, p. 36–47, 1972.

BISSACOT, A. C. G. Estudo Comparativo entre Regressão Logística Binária e Redes Neurais Artificiais na Avaliação dos Resultados Clássicos de Hosmer, Lemeshow e Sturdivant. , p. 106, 2015.

BRAGA, A. DE P. **Redes neurais artificiais: teoria e aplicações**. 2º ed. 2007.

BRANT, A. W. Chilling Poultry - A Review. **Poultry Processing and Marketing**, p. 14–22, 1963.

BRASIL. Ministério da Agricultura Pecuária e Abastecimento. Portaria N 210 de 10 de Novembro de 1998. Regulamento técnico de inspeção tecnológica e higiênico sanitária de carne de aves. **Diário Oficial da União**, v. 1998, 1998.

BRESSAN, M. C.; BERAQUET, N. J. Tratamentos de pré-resfriamento e resfriamento sobre a qualidade de carne de peito de frango. **Ciência e Tecnologia de Alimentos**, v. 24, n. 2, p. 230–235, 2004.

BRIK, B.; BETTAYEB, B.; SAHNOUN, M.; DUVAL, F. Towards predicting system disruption in industry 4.0: Machine learning-based approach. **Procedia Computer Science**, v. 151, n. 2018, p. 667–674, 2019. Elsevier B.V. Disponível em: <<https://doi.org/10.1016/j.procs.2019.04.089>>. .

BRYNJOLFSSON, B. Y. E.; MCAFEE, A. Business of Artificial. **Harvard Business Review**, p. 1–20, 2017.

BURRELL, J. How the machine ‘thinks’: Understanding opacity in machine learning algorithms. **Big Data and Society**, v. 3, n. 1, p. 1–12, 2016.

BUSCEMA, P. M.; MASSINI, G.; BREDA, M.; et al. Artificial neural networks. **Studies in Systems, Decision and Control**, v. 131, p. 11–35, 2018.

CARCIOFI, B. A. M. **Estudo do resfriamento de carcaças de frango em chiller de imersão em água**, 2005.

CARCIOFI, B. A. M.; LAURINDO, J. B. Water uptake by poultry carcasses during cooling by water immersion. **Chemical Engineering and Processing: Process Intensification**, v. 46, n. 5, p. 444–450, 2007.

CARCIOFI, B. A. M.; LAURINDO, J. B. Experimental results and modeling of poultry carcass cooling by water immersion. **Ciência e Tecnologia de Alimentos**, v. 30, n.

2, p. 447–453, 2010.

CLEARY, J. G. K*: An Instance-based Learner Using an Entropic Distance Measure. **Elsevier**, v. 5, p. 1–14, 1995. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9781558603776500220>>. .

COSTA, S. **The saga of the Brazilian poultry industry: How Brazil has become the world's largest exporter of chicken meat**. 2011.

CROWLEY, J. L.; M, E. M. Intelligent Systems : Reasoning and Recognition Introduction to Bayesian Recognition. , v. 2009, p. 1–8, 2016.

DENG, Z. HUA; YIN, H. QING; JIANG, X.; et al. Machine-learning-assisted prediction of the mechanical properties of Cu—Al alloy. **International Journal of Minerals, Metallurgy and Materials**, v. 27, n. 3, p. 362–373, 2020.

DEVASENA, C. L. **EFFECTIVENESS PREDICTION OF MEMORY BASED CLASSIFIERS FOR THE CLASSIFICATION OF MULTIVARIATE DATA SET**, 2017. Sphoorthy Engineering College.

DEY, A. Machine Learning Algorithms: A Review. **International Journal of Computer Science and Information Technologies**, v. 7, n. 3, p. 1174–1179, 2016. Disponível em: <www.ijcsit.com>. .

DIAS, E. M. R. ESTUDO DE CASO SOBRE CRÉDITOS DE IMPOSTOS PAGOS EM UMA EMPRESA DO SETOR AVÍCOLA. , p. 1–48, 2016.

DINIZ, F. A.; NETO, F. M. M.; JÚNIOR, F. DAS C. L.; FONTES, L. M. O. RedFace: um sistema de reconhecimento facial baseado em técnicas de análise de componentes principais e autofaces. **Revista Brasileira de Computação Aplicada**, v. 5, n. 1, p. 42–54, 2013.

EMBRAPA. **Sonho, Desafio e Tecnologia: 35 anos de contribuição da Embrapa suínos e aves**. 2011.

FERNANDES, F. T. Perspectivas do uso de mineração de dados e aprendizado de máquina em saúde e segurança no trabalho. , v. 6369, p. 1–12, 2019.

FLECK, L.; TAVARES, M. H. F.; EYNG, E.; HELMANN, A. C.; ANDRADE, M. A. DE M. Redes Neurais Artificiais: Princípios Básicos Artificial Neural Networks: Basic Principles. **Revista Eletrônica Científica Inovação e Tecnologia**, v. 1, n. 13, p. 47–57, 2016. Disponível em: <<https://periodicos.utfpr.edu.br/recit/article/viewFile/4330/Leandro>>. .

GE, Z.; SONG, Z.; DING, S. X.; HUANG, B. Data Mining and Analytics in the Process Industry: The Role of Machine Learning. **IEEE Access**, v. 5, p. 20590–20616, 2017.

GOMES, D. DOS S. Inteligência Artificial : Conceitos e Aplicações. , p. 234–246, 2010.

HAYKIN, S. **Neural networks and learning**. 3^o ed. Ontario, Canada: Pearson Prentice Hall, 2009.

HEBB, D. **The Organization of Behavior: A Neuropsychological Theory**. 1949.

HERNÁNDEZ, D. C. T. An Experimental Study of K* Algorithm. **International Journal of Information Engineering and Electronic Business**, v. 7, n. 2, p. 14–19, 2015.

JAMES, C.; VINCENT, C.; DE ANDRADE LIMA, T. I.; JAMES, S. J. The primary chilling of poultry carcasses-a review. **International Journal of Refrigeration**, v. 29, n. 6, p. 847–862, 2005.

JOHASEN, S. V; BENDTSEN, J. D.; MOGENSEN, J. Broiler slaughter weight forecasting using dynamic neural network models. **IEEE 6th International Conference (ICIEA)**, 2019.

KALMEGH, S. Analysis of WEKA Data Mining Algorithm REPTree , Simple Cart and RandomTree for Classification of Indian News. **International Journal of Innovative Science, Engineering & Technology**, v. 2, n. 2, p. 438–446, 2015.

KLASSEN, T. **Uso de redes neurais artificiais para a modelagem da temperatura e da retenção de água no processo de resfriamento de carcaças de frango por imersão**, 2008.

KLASSEN, T.; MARTINS, T. D.; FILHO, L. C.; DA SILVA, E. A. Modelagem do sistema de resfriamento por imersão de carcaças de frangos utilizando redes neurais artificiais. **Acta Scientiarum - Technology**, v. 31, n. 2, p. 201–205, 2009.

KOPIER, A. A.; NAVARRO, V.; SILVA, A. L.; et al. Redes Neurais Artificiais e suas aplicações no setor elétrico. , v. 9, p. 27–33, 2019. Disponível em: <<http://www.fsma.edu.br/RESA27>>. .

KULKARNI, E. G.; KULKARNI, R. B. WEKA Powerful Tool in Data Mining General Terms. **International Journal of Computer Applications**, v. 5, n. Rtdm, p. 975–8887, 2016.

KUMAR, S.; HIWARKAR, T. COMPARATIVE ANALYSIS TO PREDICTING STUDENT ' S PERFORMANCE USING K -STAR. , v. 6, n. 1, p. 204–210, 2019.

L'HEUREUX, A.; GROLINGER, K.; ELYAMANY, H. F.; CAPRETZ, M. A. M. Machine Learning with Big Data: Challenges and Approaches. **IEEE Access**, v. 5, p. 7776–7797, 2017. IEEE.

LI, H.; ZHANG, Z.; LIU, Z. Application of artificial neural networks for catalysis: A review. **Catalysts**, v. 7, n. 10, 2017.

LILLARD, H. S. Improving Chilling Systems for Poultry. **Food Technol** 36, p. 58–67, 1982.

LIMA, R. Avaliação Do Algoritmo Svm Na Detecção De Comportamentos Suspeitos Em Cenas De Vídeo. , 2014. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/6469/1/PG_COADS_2014_2_04.pdf>. .

LOBO, L. C. Inteligência artificial, o Futuro da Medicina e a Educação Médica. **Revista Brasileira de Educação Médica**, v. 42, n. 3, p. 3–8, 2018.

LORENZETTI, E.; PUTON, B. M. S.; FERNANDES, I. A.; et al. Water absorption and dripping of chicken breast and carcasses during pre-cooling in an industrial system. **Poultry Science**, v. 97, n. 12, p. 4462–4469, 2018.

LU, H.; LI, Y.; CHEN, M.; KIM, H.; SERIKAWA, S. Brain Intelligence: Go beyond Artificial Intelligence. **Mobile Networks and Applications**, v. 23, n. 2, p. 368–375, 2018. Mobile Networks and Applications.

MADERA, J.; VARONA, I. L. DE. STUDY OF THE PERFORMANCE OF THE K * ALGORITHM IN INTERNATIONAL STUDY OF THE PERFORMANCE OF THE K * ALGORITHM IN. , n. May 2017, 2016.

MADHUSUDANA, C. K.; KUMAR, H.; NARENDRANATH, S. Condition monitoring of face milling tool using K-star algorithm and histogram features of vibration signal. **Engineering Science and Technology, an International Journal**, v. 19, n. 3, p. 1543–1551, 2016. Karabuk University. Disponível em: <<http://dx.doi.org/10.1016/j.jestch.2016.05.009>>. .

MAHMOOD, D. Y.; HUSSEIN, M. Intrusion Detection System Based on K-Star Classifier and Feature Set Reduction. **IOSR Journal of Computer Engineering**, v. 15, n. 5, p. 107–112, 2013.

MALAVAZZI, G. **Manual de criação de frangos de corte**. São Paulo: Nobel, 1982.

MAREL, M. W. How to chill poultry in a processing plant? Disponível em: <<https://www.viv.net/articles/news/how-to-chill-poultry-in-a-processing-plant>>. Acesso em: 21/6/2020.

MARQUES, E.; FILHO, D. S.; FERNANDES, F. D. A.; et al. Artigo de Revisão Inteligência Artificial em Cardiologia : Conceitos , Ferramentas e Desafios – “ Quem Corre é o Cavalo , Você Precisa ser o Jôquei ”. , 2019.

MENDES, J. T. G.; PADILHA JUNIOR, J. B. **Agronegócio: uma abordagem econômica**. São Paulo: Pearson Prentice Hall, 2007.

MORARIU, D. I.; GRATULESCU, R. G.; BREAZU, M. THE WEKA MULTILAYER PERCEPTRON CLASSIFIER. **International Journal of Advanced Statistics and IT&C for Economics and Life Sciences**, v. 7, 2017.

MÜLLER, T. B. **APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA AVICULTURA COMERCIAL**, 2019. UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL.

MUNDOAGRO. A revista do Avisite: A cadeia avícola analisada do ovo ao consumidor. , p. 72, dez. 2018.

NAIK, A.; SAMANT, L. Correlation Review of Classification Algorithm Using Data Mining Tool: WEKA, Rapidminer, Tanagra, Orange and Knime. **Procedia Computer Science**, v. 85, n. Cms, p. 662–668, 2016. The Author(s). Disponível em: <<http://dx.doi.org/10.1016/j.procs.2016.05.251>>. .

NEELAVATHI, C.; JAGATHEESAN, S. M. Improving Spam Mail Filtering Using Classification Algorithms With Partition Membership Filter. , p. 1–4, 2016.

NEVES FILHO, L. **Estudo de Algumas Variáveis no Resfriamento Contínuo por Imersão de Frango**, 1978. Universidade Estadual de Campinas.

NOVAKOVIC, J.; VELJOVIC, A. C. Support vector classification: Selection of kernel and parameters in medical diagnosis. **2011 IEEE 9th International Symposium on**, p. 465–470, 2011.

NUANMEESRI, S.; POOMHIRAN, L. Improving Responsiveness Conversation of Thai Chatbot through Sentiment Analysis Classification Techniques. **International Journal of Engineering and Advanced Technology**, v. 9, n. 2, p. 3733–3737, 2019.

OBDAM, J. **Resfriamento de carcaça de aves em ar ou água – implicações microbiológicas e de qualidade da carne**, 2005.

OECD/FAO. **OECD/FAO (2017), OECD-FAO Agricultural Outlook 2017-2026**, OECD Publishing, Paris. http://dx.doi.org/10.1787/agr_outlook-2017-en. 2017.

OLIVA, F. Exportações brasileiras crescem em volume e na receita em reais. **Anuário 2016 da Avicultura Industrial**. p.54–58, 2015. Itu.

OLIVO, R. **O mundo do frango: cadeia produtiva da carne de frango**. Criciúma, 2006.

PEDERSEN, R. Advantages and disadvantages of various methods for the chilling of poultry. **Landbrugsministeriets Slakteri-og Konserverlaboratorium**, 1979. Copenhagen.

PINHEIRO, T. C.; PASA, L. A. MANAGEMENT OF THE PRODUCTION OF BROILER CHICKENS THROUGH ARTIFICIAL NEURAL. , p. 1–15, 2020.

PINOTTI, R. N. Análise Comparativa Dos Mecanismos De Governança Das Redes Agroindustriais Avícola Da Macrorregião De Ribeirão Preto-Sp E De Santa Catarina. **Dissertação de Mestrado**, 2005.

PLATT, J. C. Fast Training of Support Vector Machines using Sequential Minimal Optimization. , 2000.

QIU, J.; WU, Q.; DING, G.; XU, Y.; FENG, S. A survey of machine learning for big

data processing. **Eurasip Journal on Advances in Signal Processing**, v. 2016, n. 1, 2016. EURASIP Journal on Advances in Signal Processing. Disponível em: <<http://dx.doi.org/10.1186/s13634-016-0355-x>>. .

QUINTERO, C. D. B.; MEJÍA, J. H. M.; ARBOLEDA, H. D. J. **APLICACIÓN DE UN ALGORITMO DE OPTIMIZACIÓN SECUENCIAL MÍNIMA (SMOreg)**, 2019. Universidad cooperativa de Colombia.

RAMOS, S.; MACLACHLAN, M.; MELTON, A. Impacts of the 2014–2015 highly pathogenic avian influenza outbreak on the U.S. poultry sector. **Economic Research Service, USDA**, v. LDPM-282-0, p. 282–284, 2017. Disponível em: <<https://www.ers.usda.gov/publications/pub-details/?pubid=86281>>. .

RAVIKUMAR, S.; KANAGASABAPATHY, H.; MURALIDHARAN, V. Fault diagnosis of self-aligning troughing rollers in belt conveyor system using k-star algorithm. **Measurement: Journal of the International Measurement Confederation**, v. 133, p. 341–349, 2019. Elsevier Ltd. Disponível em: <<https://doi.org/10.1016/j.measurement.2018.10.001>>. .

REALI, E. H. **Utilização de inteligência artificial (redes neurais artificiais) no gerenciamento da produção de frangos de corte**, 2004.

SAHU, M.; NAGWANI, N. K.; VERMA, S.; SHIRKE, S. An Incremental Feature Reordering (IFR) Algorithm to Classify Eye State Identification Using EEG. **Advances in Intelligent Systems and Computing**, 2015.

SAJAD, F. Effect Of Principal Component Analysis In Lung Cancer detection using Machine Learning Techniques. , , n. May, p. 6783–6791, 2019.

SAMEK, W.; WIEGAND, T.; MÜLLER, K.-R. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. , 2017. Disponível em: <<http://arxiv.org/abs/1708.08296>>. .

SANT'ANNA, V. **Análise dos fatores que afetam a temperatura e absorção de água de carcaças de frango em chiller industrial**, 2008. Universidade Federal do Rio Grande do Sul.

SARCINELLI, M. F.; VENTURINI, K. S. **ABATE DE AVES**, 2007. Universidade Federal do Espírito Santo - UFES.

SARITAS, M. M.; YASAR, A. Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification. **International Journal of Intelligent Systems and Applications in Engineering**, 2019. Disponível em: <<http://xlink.rsc.org/?DOI=C5TC02043C>>. .

SHARMA, R. K.; SUGUMARAN, V.; KUMAR, H.; AMARNATH, M. Condition monitoring of roller bearing by K-star classifier and K-nearest neighborhood classifier using sound signal. **SDHM Structural Durability and Health Monitoring**, v. 12, n. 1, p. 1–16, 2017.

SHEVADE, S. K.; KEERTHI, S. S.; BHATTACHARYYA, C.; MURTHY, K. R. K. Improvements to the SMO algorithm for SVM regression. **IEEE Transactions on Neural Networks**, v. 11, n. 5, p. 1188–1193, 2000.

SIANAKI, O. A.; YOUSEFI, A.; TABESH, A. R.; MAHDAVI, M. Machine learning applications: The past and current research trend in diverse industries. **Inventions**, v. 4, n. 1, 2019.

SILVA, M. P. DOS S. Mineração de Dados - Conceitos , Aplicações e Experimentos com Weka. **Dados**, v. 11, p. 10–18, 2009. Disponível em: <<http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Minera??o+de+Dados+-+Conceitos,+Aplica??es+e+Experimentos+com+Weka#0>>. .

SILVA, S. R.; SCHIMIDT, F. Redução de variáveis de entrada de redes neurais artificiais a partir de dados de análise de componentes principais na modelagem de oxigênio dissolvido. **Química Nova**, v. 39, n. 3, p. 273–278, 2015.

SINGHAL, S.; JENA, M. A Study on WEKA Tool for Data Preprocessing , Classification and Clustering. **International Journal of Innovative Technology and Exploring Engineering**, v. 2, n. 6, p. 250–253, 2013. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.687.799&rep=rep1&type=pdf>>. .

SOUZA, D. A. DE; SOUZA, E. T. E. DE; SANTOS, F. J. DOS; MACEDO, M. DA S.; JÚNIOR, V. D. L. REDES NEURAS ARTIFICIAIS: Identificação de danos em elementos estruturais. , p. 1–11, 2019.

SOUZA JÚNIOR, L. C. T. **Avaliação microbiológica de carcaças de frango e água em um sistema de pré-resfriamento por imersão em 8 horas e 16 horas**, 2009. Universidade Estadual Paulista.

SYARIF, I.; PRUGEL-BENNETT, A.; WILLS, G. SVM parameter optimization using grid search and genetic algorithm to improve classification performance. **Telkomnika (Telecommunication Computing Electronics and Control)**, v. 14, n. 4, p. 1502–1509, 2016.

TERUEL, B.; CORTEZ, L.; NEVES FILHO, L. Estudo comparativo do resfriamento de laranja valência com ar forçado e com água. **Ciência e Tecnologia de Alimentos**, v. 23, n. 2, p. 174–178, 2003.

ÜSTÜN, B.; MELSSSEN, W. J.; BUYDENS, L. M. C. Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel. **Chemometrics and Intelligent Laboratory Systems**, v. 81, n. 1, p. 29–40, 2006.

VIDHYA, M. Efficient classification of portscan attacks using support vector machine. In: Green High Performance Computing (ICGHPC). **2013 IEEE International Conference on**, p. 1–5, 2013.

VIJAYARANI, M. S.; MUTHULAKSHMI, M. M.; PROFESSOR, A. Comparative Analysis of Bayes and Lazy Classification Algorithms. **International Journal of**

Advanced Research in Computer and Communication Engineering, v. 2, n. 8, 2013. Disponível em: <www.ijarcce.com>. .

VITKAR, S. Trend Analysis and Prediction of Air and Water Pollutants using Regression algorithm SMOreg. **JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN COMPUTER ENGINEERING**, v. 2, n. 0975–6760, p. 487–492, 2013.

WANG, L.; SUN, D. Evaluation of performance of slow air, air blast and water immersion cooling methods in the cooked meat industry by the finite element method. **Journal of Food Engineering**, p. 1–12, 2001.

WITTEN, I. H.; FRANK, E.; HALL, M. A. **Data Mining: Practical Machine Learning Tools and Techniques, Third Edition (The Morgan Kaufmann Series in Data Management Systems)**. 2011.

YOUNG, L. L.; SMITH, D. P. Moisture Retention by Water- and AirChilled Chicken Broilers During Processing and Cutup Operations. **Poultry Science**, v. 119–122, 2004.

ZEN, S. DE; IGUNA, M. D.; ORTELAN, C. B.; SANTOS, V. H. S. DOS; FELLI, C. B. Evolução da avicultura no Brasil Mercado internacional. **Informativo Cepea**, p. 1–3, 2015. Disponível em: <www.cepea.esalq.usp.br>. .

APÊNDICE A – Resultados Melhor Configuração *K-Star*

Quadro 18 - Resultados dos valores individuais *K-Star*

Instância	Valor Atual	Valor Predito	Diferença
1	8,543	4,517	-4,026
2	5,660	3,934	-1,727
3	4,374	8,268	+3,894
4	5,342	4,273	-1,069
5	9,971	6,791	-3,180
6	6,644	5,005	-1,639
7	3,767	4,565	+0,798
8	6,484	3,202	-3,282
9	9,520	5,800	-3,720
10	9,885	8,267	-1,618
11	4,167	3,933	-0,234
12	3,537	2,826	-0,711
13	6,274	4,274	-2,001
14	5,062	3,202	-1,860
15	1,915	2,883	+0,968
16	4,157	4,378	+0,221
17	4,528	4,504	-0,024
18	6,577	8,699	+2,122
19	4,967	6,791	+1,825
20	3,377	3,896	+0,519
21	3,718	4,506	+0,788
22	3,957	2,358	-1,599
23	4,182	4,459	+0,277
24	4,636	4,563	-0,073
25	2,685	3,216	+0,531
26	3,309	3,934	+0,624
27	3,426	3,202	-0,224
28	4,485	4,323	-0,162
29	4,063	2,848	-1,215
30	2,894	6,536	+3,642
31	6,707	5,087	-1,621
32	5,051	3,896	-1,154
33	5,311	6,130	+0,818

APÊNDICE B – Resultados Melhor Configuração SMOReg Kernel Polinomial

Quadro 19 - Resultados individuais SMOReg Kernel

Instância	Valor Atual	Valor Predito	Diferença
1	8,543	8,719	+0,176
2	5,66	5,678	+0,017
3	4,374	4,369	-0,005
4	5,342	5,317	-0,025
5	9,971	9,908	-0,063
6	6,644	6,299	-0,346
7	3,767	3,723	-0,044
8	6,484	6,614	+0,13
9	9,52	9,742	+0,223
10	9,885	10,188	+0,303
11	4,167	4,165	-0,001
12	3,537	3,408	-0,129
13	6,274	6,17	-0,105
14	5,062	5,185	+0,122
15	1,915	1,59	-0,325
16	4,157	4,225	+0,068
17	4,528	4,554	+0,026
18	6,577	6,649	+0,073
19	4,967	5,044	+0,078
20	3,377	3,366	-0,012
21	3,718	3,771	+0,053
22	3,957	4,034	+0,076
23	4,182	4,188	+0,007
24	4,636	4,674	+0,038
25	2,685	2,685	-0,001
26	3,309	3,299	-0,011
27	3,426	3,51	+0,084
28	4,485	4,517	+0,032
29	4,063	4,046	-0,017
30	2,894	3,108	+0,214
31	6,707	6,531	-0,176
32	5,051	5,037	-0,013
33	5,311	5,322	+0,011

APÊNDICE C – Resultados Melhor Configuração SMOReg Kernel RBF

Quadro 20 - Resultados individuais SMOReg RBF Kernel

Instância	Valor Atual	Valor Predito	Diferença
1	8,543	8,251	-0,292
2	5,66	5,48	-0,18
3	4,374	4,389	+0,016
4	5,342	5,373	+0,031
5	9,971	9,811	-0,161
6	6,644	6,32	-0,324
7	3,767	3,662	-0,105
8	6,484	6,454	-0,03
9	9,52	9,538	+0,018
10	9,885	9,75	-0,135
11	4,167	4,161	-0,005
12	3,537	3,505	-0,032
13	6,274	6,284	+0,009
14	5,062	5,123	+0,061
15	1,915	1,882	-0,033
16	4,157	4,337	+0,18
17	4,528	4,619	+0,091
18	6,577	6,553	-0,024
19	4,967	5,062	+0,095
20	3,377	3,388	+0,011
21	3,718	3,859	+0,141
22	3,957	4,002	+0,045
23	4,182	4,16	-0,021
24	4,636	4,597	-0,039
25	2,685	2,622	-0,064
26	3,309	3,353	+0,044
27	3,426	3,529	+0,103
28	4,485	4,631	+0,147
29	4,063	3,932	-0,131
30	2,894	2,932	+0,038
31	6,707	6,61	-0,097
32	5,051	5,029	-0,022
33	5,311	5,292	-0,02

APÊNDICE D – Resultados Melhor Configuração SMOReg PUK

Quadro 21 - Resultados individuais SMOReg PUK

Instância	Valor Atual	Valor Predito	Diferença
1	8,543	8,217	-0,326
2	5,66	5,517	-0,144
3	4,374	4,379	+0,006
4	5,342	5,337	-0,005
5	9,971	9,652	-0,319
6	6,644	6,2	-0,444
7	3,767	3,693	-0,074
8	6,484	6,373	-0,111
9	9,52	9,521	+0,001
10	9,885	9,777	-0,108
11	4,167	4,183	+0,016
12	3,537	3,527	-0,01
13	6,274	6,193	-0,082
14	5,062	5,085	+0,023
15	1,915	1,854	-0,061
16	4,157	4,292	+0,135
17	4,528	4,634	+0,106
18	6,577	6,525	-0,051
19	4,967	5,06	+0,093
20	3,377	3,396	+0,019
21	3,718	3,838	+0,12
22	3,957	3,897	-0,06
23	4,182	4,175	-0,007
24	4,636	4,778	+0,142
25	2,685	2,687	+0,002
26	3,309	3,349	+0,04
27	3,426	3,569	+0,143
28	4,485	4,561	+0,076
29	4,063	4,014	-0,049
30	2,894	3,193	+0,299
31	6,707	6,646	-0,061
32	5,051	5,01	-0,041
33	5,311	5,348	+0,037

APÊNDICE E – Resultados Melhor Configuração RNA

Quadro 22 - Resultados individuais RNA melhor configuração

Instância	Valor Atual	Valor Predito	Diferença
1	8,543	8,035	-0,508
2	5,660	5,728	+0,068
3	4,374	4,410	+0,036
4	5,342	5,329	-0,013
5	9,971	9,190	-0,781
6	6,644	6,386	-0,259
7	3,767	3,934	+0,167
8	6,484	6,607	+0,123
9	9,520	8,789	-0,731
10	9,885	9,398	-0,488
11	4,167	4,179	+0,013
12	3,537	3,759	+0,222
13	6,274	6,524	+0,25
14	5,062	4,984	-0,078
15	1,915	1,878	-0,037
16	4,157	4,175	+0,018
17	4,528	5,282	+0,753
18	6,577	6,759	+0,182
19	4,967	4,972	+0,005
20	3,377	3,203	-0,174
21	3,718	3,498	-0,220
22	3,957	3,735	-0,222
23	4,182	4,205	+0,023
24	4,636	5,068	+0,432
25	2,685	2,644	-0,041
26	3,309	3,120	-0,190
27	3,426	3,333	-0,093
28	4,485	4,330	-0,155
29	4,063	4,172	+0,109
30	2,894	2,978	+0,084
31	6,707	6,859	+0,152
32	5,051	4,905	-0,146
33	5,311	5,377	+0,066

ANEXO A – Apresentação dos Dados Trabalhados

Figura 24 - Conjunto de dados trabalhados

Ti (°C)	Tf (°C)	Mi (g)	Mf (g)	Tempo Resf. (min)	Tipc (°C)	Tfpc (°C)	Tic (°C)	Tfc (°C)	Bomba (m3/h)	Abs (%)
35,8	2,6	1102	1198	60	3	1,7	1,1	1,8	220	8,7114
36,9	3,5	1129	1166	60	3	1,7	1,1	1,8	220	3,2772
36,8	2,7	1043	1090	60	3	1,7	1,1	1,8	220	4,5062
35	2,8	1112	1158	60	3	1,7	1,1	1,8	220	4,1367
36,5	3,2	1149	1244	60	3	1,7	1,1	1,8	220	8,2681
36,9	3	1166	1217	60	3	1,7	1,1	1,8	220	4,3739
35,8	2,5	1195	1254	60	3	1,7	1,1	1,8	220	4,9372
35	2,9	1115	1158	60	3	1,7	1,1	1,8	220	3,8565
34,8	4,2	1133	1245	60	3	1,7	1,1	1,8	220	9,8853
36,3	3,3	1110	1183	60	3	1,7	1,1	1,8	220	6,5766
36,6	5,7	811	881	41	7,9	4,2	3,5	3	285	8,6313
35,9	8	1054	1098	41	7,9	4,2	3,5	3	285	4,1746
37	6,7	983	1033	41	7,9	4,2	3,5	3	285	5,0865
36,5	9,4	1093	1160	41	7,9	4,2	3,5	3	285	6,1299
37,2	7,4	1127	1180	41	7,9	4,2	3,5	3	285	4,7028
37,4	8	1092	1150	41	7,9	4,2	3,5	3	285	5,3114
36	7,4	1059	1122	41	7,9	4,2	3,5	3	285	5,949
36,2	8,5	1076	1133	41	7,9	4,2	3,5	3	285	5,2974
34,5	6	1014	1070	41	7,9	4,2	3,5	3	285	5,5227
35,6	6,2	984	1050	41	7,9	4,2	3,5	3	285	6,7073
35,3	4,2	993	1030	48	2,3	1,7	1,3	2,8	252	3,7261
36,4	3,8	997	1066	48	2,3	1,7	1,3	2,8	252	6,9208
36,5	4,6	994	1041	48	2,3	1,7	1,3	2,8	252	4,7284
35,5	3,8	1077	1135	48	2,3	1,7	1,3	2,8	252	5,3853
34,6	4,9	988	1022	48	2,3	1,7	1,3	2,8	252	3,4413
34	4,1	1111	1166	48	2,3	1,7	1,3	2,8	252	4,9505
33,5	4	1000	1091	48	2,3	1,7	1,3	2,8	252	9,1
32	4	1015	1080	48	2,3	1,7	1,3	2,8	252	6,4039
34	3,8	1000	1036	48	2,3	1,7	1,3	2,8	252	3,6
34	4,1	1031	1066	48	2,3	1,7	1,3	2,8	252	3,3948
37,1	4,5	1200	1250	61	6,2	2	2,2	1,6	175	4,1667
35,7	3,4	963	1015	61	6,2	2	2,2	1,6	175	5,3998
36,7	4,2	1144	1189	61	6,2	2	2,2	1,6	175	3,9336
36,5	3,5	1049	1088	61	6,2	2	2,2	1,6	175	3,7178
37,2	3,4	1048	1095	61	6,2	2	2,2	1,6	175	4,4847
36,5	3,9	1118	1155	61	6,2	2	2,2	1,6	175	3,3095
36,8	5,1	1206	1255	61	6,2	2	2,2	1,6	175	4,063
36,7	4,2	1274	1310	61	6,2	2	2,2	1,6	175	2,8257
36,2	4,4	1060	1120	61	6,2	2	2,2	1,6	175	5,6604
39,5	5,4	1244	1288	61	6,2	2	2,2	1,6	175	3,537
39,7	6,8	1060	1108	50	8,5	2,9	4	3,9	170	4,5283
39,4	7,3	1211	1260	50	8,5	2,9	4	3,9	170	4,0462
38,3	7,6	1112	1207	50	8,5	2,9	4	3,9	170	8,5432
39,4	9,1	1320	1359	50	8,5	2,9	4	3,9	170	2,9545
39	9	1174	1237	50	8,5	2,9	4	3,9	170	5,3663
37	9,5	1276	1319	50	8,5	2,9	4	3,9	170	3,3699
38,6	8,3	1160	1209	50	8,5	2,9	4	3,9	170	4,2241
37,6	7,2	1129	1180	50	8,5	2,9	4	3,9	170	4,5173
38,2	7,5	1238	1315	50	8,5	2,9	4	3,9	170	6,2197
36,6	7,4	1129	1170	50	8,5	2,9	4	3,9	170	3,6315
37,9	4,2	1359	1422	49	5	2	1,2	2,1	200	4,6358
39,4	3,5	1031	1068	49	5	2	1,2	2,1	200	3,5887
38	3,2	1053	1098	49	5	2	1,2	2,1	200	4,2735
38,3	3,8	1067	1124	49	5	2	1,2	2,1	200	5,3421

36,8	4,5	1161	1214	49	5	2	1,2	2,1	200	4,565
36,9	5,5	1105	1175	49	5	2	1,2	2,1	200	6,3348
37,7	3	1020	1084	49	5	2	1,2	2,1	200	6,2745
38,7	3,1	999	1039	49	5	2	1,2	2,1	200	4,004
39,2	3,5	1048	1092	49	5	2	1,2	2,1	200	4,1985
37,5	2,9	1003	1046	49	5	2	1,2	2,1	200	4,2871
38,5	5,3	1066	1102	47	2	1	1,2	2,2	190	3,3771
38,6	5,9	1155	1188	47	2	1	1,2	2,2	190	2,8571
38,7	5,8	1138	1204	47	2	1	1,2	2,2	190	5,7996
39,4	4,7	1052	1088	47	2	1	1,2	2,2	190	3,4221
39,7	5,9	1201	1224	47	2	1	1,2	2,2	190	1,9151
37	5	1050	1098	47	2	1	1,2	2,2	190	4,5714
38,8	4,9	1015	1047	47	2	1	1,2	2,2	190	3,1527
38,2	5,7	1078	1120	47	2	1	1,2	2,2	190	3,8961
37,8	5,2	1124	1231	47	2	1	1,2	2,2	190	9,5196
38,7	5,6	1089	1144	47	2	1	1,2	2,2	190	5,0505
35,3	3,6	947	990	47	3,8	1,9	1,6	1	165	4,5407
35,4	3,7	925	965	47	3,8	1,9	1,6	1	165	4,3243
37	2,9	903	963	47	3,8	1,9	1,6	1	165	6,6445
36,1	3,2	919	965	47	3,8	1,9	1,6	1	165	5,0054
35,6	3,7	1017	1045	47	3,8	1,9	1,6	1	165	2,7532
34,8	3,4	866	902	47	3,8	1,9	1,6	1	165	4,157
35,5	4,2	1080	1109	47	3,8	1,9	1,6	1	165	2,6852
36	3,5	1022	1067	47	3,8	1,9	1,6	1	165	4,4031
34,8	3,2	1013	1058	47	3,8	1,9	1,6	1	165	4,4422
36	4,1	1156	1199	47	3,8	1,9	1,6	1	165	3,7197
37,5	3,8	969	993	43	4,2	3,1	2,5	1,4	215	2,4768
37,8	5	1117	1152	43	4,2	3,1	2,5	1,4	215	3,1334
38,3	3,4	933	955	43	4,2	3,1	2,5	1,4	215	2,358
37,4	4,1	1059	1082	43	4,2	3,1	2,5	1,4	215	2,1719
37	3,9	1133	1175	43	4,2	3,1	2,5	1,4	215	3,707
37,1	5,2	1121	1151	43	4,2	3,1	2,5	1,4	215	2,6762
38,1	5,2	1086	1109	43	4,2	3,1	2,5	1,4	215	2,1179
37,6	3,8	1112	1159	43	4,2	3,1	2,5	1,4	215	4,2266
36,2	3,1	935	972	43	4,2	3,1	2,5	1,4	215	3,9572
37,1	3,5	1048	1087	43	4,2	3,1	2,5	1,4	215	3,7214
38,1	4,9	1047	1099	40	5,5	2,1	4,4	2,1	380	4,9666
38	4,7	992	1053	40	5,5	2,1	4,4	2,1	380	6,1492
37,6	4,4	1043	1147	40	5,5	2,1	4,4	2,1	380	9,9712
38,2	4,9	1126	1183	40	5,5	2,1	4,4	2,1	380	5,0621
38	5,3	1095	1166	40	5,5	2,1	4,4	2,1	380	6,484
38,3	5,2	979	1032	40	5,5	2,1	4,4	2,1	380	5,4137
38,5	6	1093	1128	40	5,5	2,1	4,4	2,1	380	3,2022
37,9	4,7	1196	1258	40	5,5	2,1	4,4	2,1	380	5,184
38,1	5,2	1080	1117	40	5,5	2,1	4,4	2,1	380	3,4259
38,4	5	1016	1085	40	5,5	2,1	4,4	2,1	380	6,7913
37	3,7	933	989	40	5,1	2,5	2	2,1	330	6,0021
37,1	2,8	1002	1072	42	5,1	2,5	2	2,1	330	6,986
36,8	2,3	885	921	42	5,1	2,5	2	2,1	330	4,0678
36,5	2,7	940	1002	42	5,1	2,5	2	2,1	330	6,5957
37,1	3,2	1168	1212	42	5,1	2,5	2	2,1	330	3,7671
37,5	2,9	1066	1123	42	5,1	2,5	2	2,1	330	5,3471
37	3,4	1100	1146	42	5,1	2,5	2	2,1	330	4,1818
37,3	3,5	1099	1148	42	5,1	2,5	2	2,1	330	4,4586
36,8	3,2	933	960	42	5,1	2,5	2	2,1	330	2,8939
36,5	3,9	1029	1077	42	5,1	2,5	2	2,1	330	4,6647

Fonte: Sant'anna (2008).