



ROBSON ROSSERRANI DE LIMA

**ESTUDO COMPARATIVO DE TÉCNICAS DE INTELIGÊNCIA
COMPUTACIONAL PARA ESTIMAÇÃO DOS
REQUERIMENTOS ENERGÉTICOS DE GADO BOVINO**

LAVRAS – MG

2018

ROBSON ROSSERRANI DE LIMA

**ESTUDO COMPARATIVO DE TÉCNICAS DE INTELIGÊNCIA COMPUTACIONAL
PARA ESTIMAÇÃO DOS REQUERIMENTOS ENERGÉTICOS DE GADO BOVINO**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Engenharia de Sistemas e Automação, área de concentração em Engenharia de Sistemas e Automação, para a obtenção do título de Mestre.

Prof. DSc. Danton Diego Ferreira

Orientador

Prof. DSc. Mateus Pies Gionbelli

Coorientador

LAVRAS – MG

2018

**Ficha catalográfica elaborada pelo Sistema de Geração de Ficha Catalográfica da Biblioteca
Universitária da UFLA, com dados informados pelo próprio autor.**

Lima, Robson Rosserrani de

Estudo Comparativo de Técnicas de Inteligência Computacional para Estimação dos Requerimentos Energéticos de Gado Bovino / Robson Rosserrani de Lima. – Lavras : UFLA, 2018.

93 p. : il.

Dissertação (mestrado acadêmico)–Universidade Federal de Lavras, 2018.

Orientador: Prof. DSc. Danton Diego Ferreira.

Bibliografia.

1. Inteligência Computacional. 2. Requerimento Energético de Bovinos. 3. Redes Neurais. I. Ferreira, Danton Diego. II. Gionbelli, Mateus Pies. III. Título.

ROBSON ROSSERRANI DE LIMA

**ESTUDO COMPARATIVO DE TÉCNICAS DE INTELIGÊNCIA COMPUTACIONAL
PARA ESTIMAÇÃO DOS REQUERIMENTOS ENERGÉTICOS DE GADO BOVINO
COMPARATIVE STUDY OF COMPUTATIONAL INTELLIGENCE TECHNIQUES
FOR ESTIMATION OF ENERGY REQUIREMENTS OF CATTLE**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Engenharia de Sistemas e Automação, área de concentração em Engenharia de Sistemas e Automação, para a obtenção do título de Mestre.

APROVADA em 30 de Agosto de 2018.

Prof. DSc. Danton Diego Ferreira UFLA

Prof. DSc. Augusto Santiago Cerqueira UFJF

Prof. DSc. Wilian Soares Lacerda UFLA

Prof. DSc. Elias San Vito UFMT

Prof. DSc. Danton Diego Ferreira
Orientador

Prof. DSc. Mateus Pies Gionbelli
Co-Orientador

**LAVRAS – MG
2018**

À Juscicleide. Sua força, apoio e companheirismo foram essenciais para a conclusão deste trabalho. Obrigado por todo o tempo do meu lado, nunca me deixando desistir.

AGRADECIMENTOS

Esta dissertação de mestrado é o resultado de uma jornada de vinte anos. Desde o primeiro dia na escola até a defesa deste trabalho, diversas pessoas e profissionais participaram desta conquista e devo agradecimento a todos.

Agradeço principalmente à meus pais, Renato e Sueli, que dedicaram suas vidas a dar a mim e minha irmã a melhor educação que puderam. Agradeço por cada dia de trabalho deles, cada puxão de orelha para não deixar de fazer a lição, cada palavra de incentivo, sempre visando me oferecer as oportunidades que muitas vezes não tiveram, me ajudando a construir um futuro melhor.

À todos meus professores, desde às “tias” que me ensinaram as letras e números na pré-escola, passando pelos excelentes professores do ciclo fundamental e médio. Aos professores do ensino superior, em especial aos professores Bruno, Dimitri, Leonardo e Danton, que foram meus orientadores e me ajudaram a ver a beleza e importância da carreira acadêmica e facilitaram muito este caminho com todo seu apoio e conhecimento.

À Juscicleide, que nestes dois anos de mestrado aguentou meu estresse, minhas queixas, minha falação, e me ajudou a superar cada uma dessas dificuldades e compartilhou verdadeiramente de cada sucesso e alegria.

Obrigado à minha irmã, Renata, por sempre me proporcionar muitas gargalhadas e sempre ser prestativa a tudo que preciso, sempre dando um jeito em meio a tantas tarefas, que sempre cumpre com maestria.

À todos os amigos da graduação que continuaram dando apoio durante a pós, e aos novos amigos do PPGESisA que sempre proporcionaram um ótimo ambiente de trabalho.

Ao professor Mateus, meu coorientador e idealizador deste trabalho, e à todos os demais professores que contribuíram direta e/ou indiretamente para o desenvolvimento desta dissertação.

*Não sou nada.
Nunca serei nada.
Não posso querer ser nada.
À parte isso, tenho em mim todos os sonhos do mundo.
(Álvaro de Campos - Heterônimo de Fernando Pessoa)*

RESUMO

O Brasil é o segundo maior exportador de carne bovina do mundo e dados apontam para um aumento crescente no consumo e exportação, demandando um aumento na produção. Devido às questões ambientais, torna-se importante que esse acréscimo ocorra sem que se aumente significativamente a área de produção. A chave para isso reside em conhecimento preciso das exigências nutricionais e da composição dos alimentos fornecidos ao gado brasileiro. As técnicas baseadas em Inteligência Computacional (IC) vêm ganhando espaço nas ciências animais e se mostrando ferramentas eficazes para as mais diversas aplicações. Nesta Dissertação de Mestrado foram desenvolvidos modelos capazes de estimar os requerimentos energéticos do gado bovino, uma das bases para elaboração de dietas eficientes para os animais. Foram desenvolvidas nesse trabalho técnicas baseadas em Redes Neurais Artificiais (RNA) e Sistemas de Inferência *Fuzzy* (SIF), alicerces da IC, como alternativas aos métodos clássicos. As técnicas abordadas foram: Redes Perceptron Multicamadas (MLP) com treinamento *backpropagation* baseado no algoritmo Gradiente Conjugado Escalonado (SCG), Rede Neural de Regressão Generalizada (RNRG) e Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS). Todas as técnicas foram empregadas para a estimação do Consumo de Energia Metabolizável (CEM) a partir de um banco de dados de 840 animais oriundos de 31 estudos. Os resultados foram analisados estatisticamente e comparados entre si, com o método de modelagem clássico, baseado em Regressões Múltiplas (RM) e com as equações sugeridas pelo Sistema BR-CORTE. Os parâmetros utilizados para a modelagem foram: gênero, grupo genético, sistema de alimentação, Peso Corporal Vazio Médio (PCVZm) e Ganho de Peso de Corpo Vazio (GPCVZ). Os modelos propostos foram capazes de superar o desempenho médio da técnica clássica em cinco das oito métricas utilizadas para avaliar estatisticamente os modelos resultantes, apresentando ganho significativo pelo Teste de Tukey em três. A melhor metodologia encontrada foram os MLP, com resultados superiores em relação a correlação linear, índice de determinação e coeficiente de correlação de concordância. Os resultados encontrados atestam a validade das metodologias de IC para predição de requerimentos energéticos em bovinos de corte.

Palavras-chave: Inteligência Computacional. Redes Neurais. Sistemas de Inferência Fuzzy. Requerimento Energético de Bovinos. Consumo de Energia Metabolizável.

ABSTRACT

Brazil is the second largest exporter of beef in the world and the increase in consumption and exports, will demand an increase in production. Due to environmental issues, it is important that this increase occurs without significantly increasing the production area. The key to achieve this goal lies in an accurate knowledge of the nutritional requirements and the food composition offered to the cattle. Techniques based on Computational Intelligence (CI) are spreading in animal sciences and shown as effective tools for the most diverse applications. In this Master's Dissertation, were developed models capable of estimating the energy requirements of cattle, one of the bases for the elaboration of efficient diets for the animals. Techniques based on Artificial Neural Network (ANN) and Fuzzy Inference Systems (FIS), foundations of CI, were developed in this work as alternatives to classical methods. The techniques covered were: Multilayer Perceptron (MLP) Networks with Backpropagation training based on Squared Conjugate Gradient (SCG), Generalized Regression Neural Networks (GRNN) and Adaptive Neuro-Fuzzy Inference Systems (ANFIS). All techniques were used to estimate the Metabolizable Energy Intake (MEI) from a database of 840 animals from 31 studies. The results were analyzed statistically and compared with each other, with the classical modeling method, based on Multiple Regressions (MR) and with the equations suggested by BR-CORTE System. The parameters used for the modeling were: gender, genetic group, feeding system, Average Empty Body Weight (AEBW) and Empty Body Weight Gain (EBWG). The proposed models were able to overcome the average performance of classical technique in five of the eight metrics used to statistically evaluate the resulting models, presenting a significant gain by the Tukey Test in three. The best methodology was the MLP, with superior results in relation to linear correlation, determination index and concordance correlation coefficient. The results confirm the validity of CI methodologies for the prediction of energy requirements in beef cattle.

Keywords: Computational Intelligence. Neural Networks. Fuzzy Inference Systems. Energy Requirements of Cattle. Metabolizable Energy Intake.

LISTA DE FIGURAS

Figura 3.1 – Arquitetura de um neurônio artificial <i>Perceptron</i> com cinco parâmetros de entrada.	32
Figura 3.2 – Arquitetura geral de uma rede MLP com duas camadas escondidas de tamanhos H e O , respectivamente, e um neurônio de saída linear.	32
Figura 3.3 – Arquitetura de um neurônio artificial RBF.	34
Figura 3.4 – Arquitetura geral de uma rede RBF do tipo Rede Neural de Regressão Generalizada (RNRG) com uma camada escondida de tamanho H e um neurônio de saída linear.	34
Figura 3.5 – Arquitetura típica de um ANFIS com n entradas.	35
Figura 4.1 – Curvas de treinamento e validação para os modelos Lineares baseados no MMQ.	42
Figura 4.2 – Curvas de treinamento e validação para os modelos MLP com 2 neurônios na camada escondida.	43
Figura 4.3 – Curvas de treinamento e validação para os modelos MLP com 3 neurônios na camada escondida.	44
Figura 4.4 – Curvas de treinamento e validação para os modelos MLP com 4 neurônios na camada escondida.	44
Figura 4.5 – Curvas de treinamento e validação para os modelos MLP com 5 neurônios na camada escondida.	45
Figura 4.6 – Curvas de treinamento e validação para os modelos MLP com 6 neurônios na camada escondida.	45
Figura 4.7 – Curvas de treinamento e validação para os modelos MLP com 7 neurônios na camada escondida.	46
Figura 4.8 – Curvas de treinamento e validação para os modelos MLP com 8 neurônios na camada escondida.	46
Figura 4.9 – Curvas de treinamento e validação para os modelos MLP com 9 neurônios na camada escondida.	47
Figura 4.10 – Curvas de treinamento e validação para os modelos MLP com 10 neurônios na camada escondida.	47
Figura 4.11 – Curvas de treinamento e validação para os modelos RNRG.	48
Figura 4.12 – Curvas de treinamento e validação para os modelos ANFIS.	49

Figura 4.13 – Desempenho dos modelos MLP-SCG para um conjunto de dados de Validação igual a 30% e uma camada escondida.	51
Figura 4.14 – Teste de Comparação Múltipla de Médias para a arquitetura MLP-SCG com 1 camada escondida e número de neurônios variando de [2, 10].	52
Figura 4.15 – Desempenho dos modelos MLP-SCG para um conjunto de dados de Validação igual a 30% e duas camadas escondidas.	53
Figura 4.16 – Teste de Comparação Múltipla de Médias para a arquitetura MLP-SCG com 2 camadas escondidas.	54
Figura 4.17 – Teste de Comparação Múltipla de Médias para os resultados de Correlação Linear.	62
Figura 4.18 – Teste de Comparação Múltipla de Médias para os resultados do Índice de Determinação.	62
Figura 4.19 – Teste de Comparação Múltipla de Médias para os resultados de Acurácia.	63
Figura 4.20 – Teste de Comparação Múltipla de Médias para os resultados de Coeficiente de Correlação de Concordância.	63
Figura 4.21 – Teste de Comparação Múltipla de Médias para os resultados do Erro Médio Quadrático de Predição.	64
Figura 1 – Arquitetura de um neurônio artificial Perceptron.	78
Figura 2 – Arquitetura geral de uma rede MLP com duas camadas escondidas de tamanhos H e O, respectivamente, e uma camada de saída linear.	79
Figura 3 – (a) Função Logística Padrão. (b) Função Tangente Hiperbólica. (c) Função Linear Pura.	82
Figura 4 – Arquitetura típica de um ANFIS com duas entradas.	90

LISTA DE TABELAS

Tabela 3.1 – Parâmetros categóricos	23
Tabela 3.2 – Pesos de corpo vazio à maturidade para diferentes grupos genéticos/classes sexuais de bovinos de corte.	28
Tabela 3.3 – Modelos adotados pelo Sistema BR-CORTE para estimar as exigências de energia para animais criados em sistema de confinamento	30
Tabela 3.4 – Modelos adotados pelo Sistema BR-CORTE para estimar as exigências de energia para animais criados em sistema de pastejo	30
Tabela 4.1 – Correlação Médio das Estimativas - Consumo de Energia Metabolizável - Modelos ANFIS em relação ao número de Agrupamentos.	57
Tabela 4.2 – Parâmetros ajustados para as FP de saída (Eq. 3.25) do melhor modelo ANFIS	58
Tabela 4.3 – Parâmetros ajustados para as FP de entrada (Eq. 3.24) do melhor modelo ANFIS	59
Tabela 4.4 – Desempenho comparativo médio dos métodos de estimação do CEM para 30% de dados de Validação e cinco parâmetros de entrada em relação à Correlação Linear (ρ), Coeficiente de Determinação (r^2), Acurácia (A_p) e Coeficiente de Correlação de Concordância ($\hat{\rho}_c$).	60
Tabela 4.5 – Desempenho comparativo médio dos métodos de estimação do CEM para 30% de dados de Validação e cinco parâmetros de entrada em relação à Raiz do Erro Médio Quadrático de Predição e suas decomposições.	61
Tabela 4.6 – Tempo médio, em segundos, de ajuste dos parâmetros para cada método de estimação do CEM, com 70% de dados de Treino e cinco parâmetros de entrada.	65
Tabela 4.7 – Número de operações matemáticas básicas realizadas por cada modelo. . .	66
Tabela 4.8 – Número de chamadas de funções e operações lógicas exclusivas de cada modelo.	66

LISTA DE ABREVIACOES

AD Anlise Discriminante

ANFIS Sistemas de Inferncia Neuro-*Fuzzy* Adaptativos

AS Agrupamento Subtrativo

CCC Coeficiente de Correlao de Concordncia

CEM Consumo de Energia Metabolizvel

CNES *California Net Energy System*

EMQ Erro Mdio Quadrtico

EMQP Erro Mdio Quadrtico de Predio

FCM Agrupamento *Fuzzy C-Means*

FP Funes de Pertinncia

GPCVZ Ganho de Peso de Corpo Vazio

IC Inteligncia Computacional

MAPA Ministrio da Agricultura, Pecuria e Abastecimento

MLP Perceptron Multicamadas

MMQ Mtodo dos Mnimos Quadrados

PCVZeq Peso de Corpo Vazio Equivalente

PCVZm Peso Corporal Vazio Mdio

RBC Raciocnio Baseado em Casos

RBF Funes de Base Radial

REMQ Raiz do Erro Médio Quadrático

RM Regressões Múltiplas

RNA Redes Neurais Artificiais

RNP Rede Neural Probabilística

RNRG Rede Neural de Regressão Generalizada

SCG Gradiente Conjugado Escalonado

SIF Sistemas de Inferência *Fuzzy*

SUMÁRIO

1	INTRODUÇÃO	13
2	REFERENCIAL TEÓRICO	15
2.1	Inteligência Computacional (IC) para Regressão	15
2.2	Inteligência Computacional nas Ciências Animais	18
3	METODOLOGIA E EXECUÇÃO DO PROJETO	22
3.1	Banco de Dados	22
3.2	Pré-processamento	24
3.3	Abordagem Clássica	25
3.3.1	Modelos BR-CORTE	25
3.3.2	Modelos Lineares	30
3.4	Técnicas de IC Propostas	31
3.4.1	Redes Neurais Perceptron Multicamadas (MLP)	31
3.4.2	Redes de Funções de Base Radial (RBF)	33
3.4.3	Sistemas de Inferência <i>Fuzzy</i> (SIF)	35
3.5	Comparações	37
4	RESULTADOS	42
4.1	Definição dos Tamanhos dos Conjuntos de Treinamento e Teste	42
4.2	Regressão Linear Múltipla - Método dos Mínimos Quadrados (MMQ)	50
4.3	Redes Perceptron Multicamadas (MLP) - Treinamento Gradiente Conjugado Escalonado (SCG)	51
4.4	Rede Neural de Regressão Generalizada (RNRG)	56
4.5	Sistemas de Inferência Neuro-<i>Fuzzy</i> Adaptativos (ANFIS)	57
4.6	Comparações de Desempenho	60
5	CONCLUSÕES	67
	REFERÊNCIAS	69
	APENDICE A – Consumo de Energia Metabolizável (CEM)	75
	APENDICE B – Redes Neurais Perceptron Multicamadas (MLP)	78
	APENDICE C – Sistemas de Inferência Neuro-<i>Fuzzy</i> Adaptativos (ANFIS)	89

1 INTRODUÇÃO

O Brasil é o segundo maior exportador de carne bovina do mundo, estando apenas atrás da Índia. No ano de 2015 foram abatidas 33,9 milhões de cabeças de gado bovino em todo país, nesse mesmo ano a arroba do boi gordo estava sendo comercializada a R\$ 143,60 e foram consumidas, somente no território nacional, 7188 mil toneladas (MAPA/AGE, 2015). Dados do Ministério da Agricultura, Pecuária e Abastecimento (MAPA), apontam ainda para o período de 2015/25 um aumento de 23,3% na produção, 17,8% no consumo e de 37,4% nas exportações (MAPA/AGE, 2015).

Para comportar esse crescimento, o setor enfrenta vários desafios, tanto ambientais, como o aumento da produção sem oferecer riscos aos biomas nativos devido ao aumento da área de manejo e resíduos, quanto econômicos, como diminuir os gastos com o manejo. Os maiores custos associados à atividade, hoje, dizem respeito a gastos com alimentação. Um caminho que pode contribuir para a solução desses desafios é o aumento da eficiência energética dos animais, isto é, a formulação de dietas que, com uma mesma unidade de alimento, consigam produzir maior quantidade de produto para consumo humano e com menos resíduos. Para isso, é necessário conhecer com precisão necessidades nutricionais e os nutrientes fornecidos pelos alimentos disponíveis aos animais.

As dietas são formuladas com base na estimativa dos requerimentos energéticos do gado, mais especificamente na estimativa do Consumo de Energia Metabolizável (CEM), calculado através de modelos múltiplos que empregam uso de regressões lineares e não-lineares (RM). O Sistema BR-CORTE (VALADARES FILHO et al., 2016) apresenta um conjunto de modelos e equações baseados em RM para o cálculo do CEM e serve como base para as estimativas dos requerimentos energéticos em bovinos. Entretanto, a técnica oferece limitações quanto ao número de parâmetros abordados, quanto à capacidade de detectar efeitos de interação entre os parâmetros e quanto ao tempo necessário para criação e atualização de novos modelos. Técnicas de Inteligência Computacional (IC) como Redes Neurais Artificiais (RNA) e Sistemas de Inferência *Fuzzy* (SIF) podem superar essas limitações e se apresentam, neste contexto, como boas alternativas às técnicas convencionais.

O objetivo desta Dissertação foi investigar a viabilidade de utilização de técnicas de Inteligência Computacional (IC) como ferramentas para a construção de modelos capazes de estimar os requerimentos energéticos de gado bovino e compará-las com a técnica tradicional. Definiu-se como objetivos específicos para este trabalho:

- Aplicação da técnica tradicional utilizada para estimação dos requerimentos energéticos baseada em Regressões Múltiplas (RM), representada pelo conjunto de equações disponíveis pelo Sistema BR-CORTE (VALADARES FILHO; MARCONDES; CHIZZOTTI, 2010) e pela construção de modelos lineares utilizando o Método dos Mínimos Quadrados (MMQ);
- Criação de modelos para a estimação com Redes Neurais Artificiais (RNA) do tipo *feed-forward*, investigando as melhores arquiteturas para as redes e utilizando diferentes variações da técnica, a citar:
 - RNA Perceptron Multicamadas (MLP) com algoritmo de treinamento do tipo *Back-propagation*;
 - RNA com neurônios do tipo Funções de Base Radial (RBF) de arquitetura Rede Neural de Regressão Generalizada (RNRG);
- Criação de modelos para a estimação com Sistemas de Inferência *Fuzzy* (SIF) do tipo Sugeno de Primeira Ordem, ajustados por arquitetura Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS).
- Comparação estatística dos modelos propostos pelas técnicas clássica e de IC;
- Definição do melhor modelo de estimação.

O texto encontra-se organizado da seguinte forma: o Capítulo 2 traz o Referencial Teórico; o Capítulo 3 descreve o banco de dados e metodologia; o Capítulo 4 descreve quais são os resultados encontrados com a aplicação dos métodos propostos e a comparação com a abordagem clássica. O Capítulo 5 apresenta as conclusões para o trabalho e propostas para trabalhos futuros. Por fim são apresentadas as Referências Bibliográficas e os Apêndices com as definições e formulações complementares.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta um conjunto de trabalhos que visam demonstrar o desempenho das técnicas de Inteligência Computacional (IC) em diversos campos de estudo. O texto está organizado de forma a primeiro apresentar o desempenho das técnicas de IC quando comparadas à abordagens estatísticas clássicas, principalmente em problemas que envolvem regressão de dados. Posteriormente discute-se o desempenho destas abordagens em diversos campos das Ciências Animais, porém não limitando-se à aplicações de regressão ou comparação. Em ambos casos, pode-se observar o bom desempenho das técnicas de IC.

2.1 Inteligência Computacional (IC) para Regressão

Algumas das técnicas estatísticas tradicionais mais utilizadas para problemas de predição e classificação são as Regressões Múltiplas (RM), Análise Discriminante (AD) e a Regressão Logística, devido à sua metodologia já bem estabelecida. Entretanto, as técnicas de IC tem ganhado popularidade como métodos alternativos a essas técnicas tradicionais, o que levou a vários estudos fazendo comparações entre tais abordagens em uma grande variedade de aplicações. Esta Seção mostrará uma breve revisão de alguns trabalhos que correlacionam e comparam redes neurais MLP ou RBF e modelos ANFIS com a abordagem estatística clássica em problemas de regressão.

Na área de contabilidade e finanças, o trabalho de Duliba (1991) comparou modelos RNA treinados pelo algoritmo Mínimo Erro Quadrado (WIDROW; HOFF, 1960), com quatro tipos de modelos de regressão para predizer o desempenho financeiro de um grupo de companhias de transporte. Em seu trabalho, os modelos RNA foram capazes de superar os efeitos aleatórios dos modelos de regressão, mas não os efeitos fixos.

Em saúde e medicina, Razi e Athappilly (2005) realizaram uma análise comparativa da acurácia de predição de modelos de redes neurais, regressão não-linear e Árvores de Regressão e Classificação (CART) utilizando um grande conjunto de dados sobre hábitos de fumantes para predizer o número de dias de cama devido à doença. Os modelos RNA e CART obtiveram as melhores predições comparados aos de regressão quando os preditores eram variáveis binárias (ou categóricas) e a variável dependente era contínua. Entretanto, nem as RNA, nem os modelos CART mostraram vantagens claras de um sobre o outro. Para esta aplicação, ambos modelos podem ser usados para predição e devem fornecer resultados mais acurados em relação à regressão.

Gu, Lam e Dhurjati (1996) usaram Regressão Quadrática para comparar os resultados de redes neurais no melhoramento da eficiência no desenvolvimento de processos de fermentação. Seus resultados mostraram que diferentes tamanhos de redes neurais dentro de um intervalo obtiveram predições igualmente boas usando a técnica da Parada Antecipada, enquanto as regressões quadráticas são sensíveis ao tamanho do conjunto de dados.

Finnie, Wittig e Desharnais (1997) usaram modelos de rede neural para estimar esforços de desenvolvimento de *software*. A melhora das técnicas de estimação disponíveis para gerentes de projetos facilita o controle mais efetivo do tempo e despesas em desenvolvimento de *software*. Este artigo comparou três técnicas de estimação usando pontos de função como uma estimativa do tamanho do sistema. Os modelos considerados foram baseados em análises de regressão, RNA e Raciocínio Baseado em Casos (RBC). Apesar dos modelos de regressão terem um desempenho baixo no conjunto de dados de 299 projetos, ambos RNA e RBC mostraram algum valor para estimar esforços de desenvolvimento de *software*. Além disso, Heiat (2002) comparou RNA e regressão para estimar a quantidade de esforço necessário para desenvolver um sistema de informação. Os resultados do estudo mostraram que, quando se usa dados de linguagem de terceira geração, a análise de regressão e redes neurais tem desempenho quase idêntica e que, combinando-se linguagens de terceira e quarta geração, as redes neurais obtêm um desempenho melhor em termos do erro absoluto médio percentual.

Lee e Um (2000) utilizaram Análises de Regressões Múltiplas (RM) e métodos de RNA para prever as relações entre parâmetros de soldagem e a geometria dos grânulos em soldagem a arco. Eles descobriram que o erro de predição das redes neurais foi menor que o da análise RM.

Li et al. (2001) compararam regressões e redes neurais para prever a potência produzida por parques eólicos e obtiveram melhores desempenhos com redes neurais do que com modelos de regressão.

A aplicabilidade de RNA para estimação dos custos de construção foi testado por Kim, An e Kang (2004) que usaram RM, RNA e RBC para estimação dos custos da construção residencial coreana. Os resultados mostraram que, apesar dos modelos de RNA terem um desempenho melhor para a acurácia da estimação, o procedimento para se treinar a melhor rede neural foi lento devido ao processo de tentativa e erro. Também nesse tema, Setyawati, Sahirman e Creese (2002), Setyawati, Creese e Sahirman (2003) concluíram que as redes neurais supera-

ram os modelos de regressão linear dados os mesmos conjuntos de treinamento e as mesmas variáveis.

Yilmaz e Kaynar (2011) utilizaram redes MLP, RBF e ANFIS para predição do potencial de inchaço em solos argilosos e compararam com modelos baseados em RM. Seus resultados mostraram que as redes neurais obtiveram acurácia semelhante aos modelos ANFIS, porém as redes RBF obtiveram um desempenho superior a MLP, ANFIS e RM. Os autores concluíram que o uso de IC é uma boa ferramenta para minimizar as incertezas nos projetos de engenharia do solo.

O uso de redes neurais em *marketing* estão associadas principalmente a estimativa de preço e elasticidades de publicidade, bem como, de forma mais geral, previsão dos efeitos das mudanças nas variáveis de *marketing*. Hruschka (1993) compara a previsão da resposta do mercado com base em dados sobre uma marca de consumo e funções de resposta do mercado representadas por redes neurais artificiais com modelos econométricos, principalmente com base em medidas de erro. Ele encontrou melhores resultados utilizando RNA mesmo com apenas uma unidade escondida em comparação com os modelos de regressão. Entretanto, ele reportou que talvez o treinamento das redes neurais estivessem com problema de *overfitting*.

Dutta, Shekhar e Wong (1994) usou redes neurais para predizer a frequência de compra de um produto e seus resultados não foram muito encorajadores, com apenas uma rede sendo capaz de superar os modelos de regressão dentro de um conjunto de 14 modelos treinados.

O trabalho de XiangJun e Al-Hashimi (2012) buscou investigar as capacidades de predição e estimação de modelos ANFIS e de regressão não-linear. Os autores utilizaram dez conjuntos de dados não-lineares com diferentes formas e tamanhos e os resultados apontaram que o método ANFIS foi praticamente perfeito para a estimação, enquanto para predição os modelos não-lineares obtiveram um desempenho levemente superior.

2.2 Inteligência Computacional nas Ciências Animais

Como demonstrado na seção anterior, as técnicas de IC vem sendo utilizadas nas mais diversas áreas do conhecimento para resolução de uma grande variedade de problemas. Entretanto, dentro das Ciências Animais, essas técnicas ainda são pouco exploradas e vêm ganhando espaço apenas recentemente. Sua utilização está principalmente para a resolução de problemas envolvendo predição em séries temporais e agrupamento de dados (FERNÁNDEZ et al., 2006).

No que diz respeito à predição de requerimentos energéticos em bovinos, área em que este trabalho se concentra, existe apenas um estudo disponível utilizando técnicas de IC. Gionbelli et al. (2016) utilizam uma RNA de arquitetura MLP para o cálculo do CEM. Seus resultados mostraram que a rede MLP foi capaz de predições mais precisas em comparação com a técnica de regressão linear múltipla.

Para crescimento animal, Potokar et al. (2015) utilizaram dados de 5893 carcaças de touros para treinar e validar RNA com o objetivo de prever o grau de conformação e gordura em touros, divididos em 14 categorias. Os resultados mostraram uma taxa de correspondências entre os valores reais e preditos de 73,6% para conformação e 64,9% para gordura. Os autores correlacionaram a taxa de sucesso com a quantidade de representantes de cada categoria e avaliaram que as RNA mostraram potencial para esse tipo de predição.

Em seu trabalho, Behzadi e Aslaminejad (2010) fizeram comparações entre técnicas de regressão estatística tradicionais e Perceptron Multicamadas (MLP) para predição do crescimento de ovelhas Baluchi. Seus resultados mostraram que as redes neurais foram capazes de produzir curvas de crescimento mais descritivas que a gerada pelo melhor modelo de regressão não-linear e com predições mais acuradas, concluindo que os modelos MLP são uma ferramenta valiosa para a predição do ganho de peso dessas ovelhas.

Raja, Ruhil e Gandhi (2012) compararam MLP e RM como ferramentas para predição do peso corporal em cabras *Attappady Black* a partir de medidas corporais de 834 animais. Os coeficientes de correlação entre os pesos corporais reais e preditos pelos modelos MLP variaram entre 90,27% e 93,69% e com baixo valor de Raiz do Erro Médio Quadrático (REMQU). Os autores concluíram que as redes neurais eram ferramentas melhores para a predição do peso corporal de cabras que a análise RM.

Mottaghitlab et al. (2010) utilizaram um tipo de rede neural auto-organizável para prever a eficiência energética e de alimentação em perus submetidos à dietas com diferentes níveis de energia e amino ácidos. Os autores relatam uma alta aproximação entre os valores

observados e reais, dando destaque ao poder dos modelos e os impactos de sua utilização na predição de características econômicas, requerimentos nutricionais e melhoria do desempenho de produção de aves domésticas.

A utilização de MLP e uma técnica híbrida entre RNA e SIF – chamada simplesmente de Sistema Neuro-*Fuzzy* (SNF) – foi investigada por Shahinfar et al. (2012) para a predição de valores genéticos em gado leiteiro iraniano, utilizando um banco de dados de vacas *Holstein*. Eles utilizaram a melhor predição linear disponível para estimar os valores para produção de leite e gordura, e compararam os resultados com os obtidos pelas técnicas de IC. Para produção de leite, os modelos MLP e SNF alcançaram correlação de 0,92 e 0,93, respectivamente. As correlações para o rendimento de gordura foram ambas de aproximadamente 0,93. A previsão conjunta para os dois parâmetros foram 0,93 (leite) e 0,93 (gordura) para o MLP e 0,94 (leite) e 0,95 (gordura) para o SNF.

Ehret et al. (2015) examinaram diferentes arquiteturas de MLP não-lineares alimentadas com várias estruturas genômicas para avaliar as habilidades de predizer características do leite usando dados de polimorfismos de nucleotídeo único (PNU) de gado leiteiro. Como referência eles utilizaram uma rede MLP linear equivalente a um modelo de Melhor Predição Linear Não-Viesada Genômica (GBLUP). Os resultados mostraram uma pequena vantagem dos modelos GBLUP sobre os MLP não-lineares, apontando a importância da seleção de parâmetros para esse tipo de análise. Ainda assim, avaliaram que as RNA são ferramentas poderosas para capturar não-linearidades em predições genômicas.

Ventura et al. (2012) utilizaram redes MLP treinadas pelo algoritmo de *Levenberg-Marquardt* (LEVENBERG, 1944; MARQUARDT, 1963) com dados de 19240 animais de gado *Tabapuã* nascidos entre 1976 e 1995 em 152 fazendas brasileiras para predizer os valores genéticos para peso aos 205 dias. Entretanto, o banco de dados não possuía os valores alvos, requeridos para o treinamento das redes, fazendo-se necessário o uso da Melhor Predição Linear Não-Viesada (BLUP) para criar esses alvos. Os resultados apontaram alta correlação entre ambas predições, porém o autor ressalta que as diferenças entre a ordenação dos resultados pode apresentar algum risco no uso de RNA e pontua a dependência da metodologia dos modelos BLUP.

Bhosale e Singh (2015) realizaram um trabalho também na linha de comparação entre os métodos estatísticos convencionais e técnicas de IC. Eles compararam redes MLP treinadas com quatro diferentes algoritmos e RM para predizer a produção de leite em primeira lactação

de 305 dias usando registros mensais da produção de leite de 443 vacas *Frieswal*. Os modelos MLP obtiveram um melhor desempenho que os modelos RM, com uma precisão de 85,07% já no 126º dia de lactação. Os autores também apontam para um aumento do Coeficiente de Determinação (r^2) dos modelos com o aumento dos registros de produção de leite. Grzesiak et al. (2003) realizaram um trabalho semelhante, utilizando MLP e RM para a predição da mesma variável, porém utilizando dados de 902 vacas da raça *Black-and-White* com várias proporções de genes *Holstein-Friesian* no genótipo. Seus resultados não apresentaram diferença ($P > 0,05$) entre as duas técnicas, mas os autores apontaram as redes neurais como métodos alternativos para a predição da produção de leite.

O trabalho de Fernández et al. (2007) utilizou RNA para fazer a predição da produção de leite de cabra uma semana à frente, baseado nas produções anteriores e atuais. Utilizaram um conjunto de 35 cabras leiteiras da raça *Murciano-Granadina* utilizando 22 para treinamento e 13 para validação dos modelos. A melhor rede treinada foi selecionada e comparada com outros modelos preditivos, obtendo sempre os melhores resultados.

Sousa et al. (2016) desenvolveram um classificador baseado em Sistemas de Inferência *Fuzzy* (SIF) para integrar fatores climáticos e animais para avaliar individualmente o nível de estresse térmico em bovinos *Nelore* em terminação de confinamento. O sistema foi construído baseado em conhecimento especialista e os resultados mostraram baixa correspondência entre o Sistemas de Inferência *Fuzzy* (SIF) e o Índice de Temperatura-Umididade (29,3%), mas grande consistência (83,2%) entre o valor médio da temperatura retal durante o dia real e a resposta do classificador *Fuzzy*. Seus resultados apresentaram melhores estimativas do nível de estresse térmico quando comparado ao índice tradicional e Sistemas *Fuzzy* anteriores.

Tavares et al. (2016) utilizaram SIF para determinar o índice de conforto térmico de zebuínos e a elaboração do zoneamento bio-climático para bovinos de corte. Eles utilizaram séries históricas de 30 anos com informações mensais médias de 156 estações. Para os autores, o uso de IC se mostrou uma ferramenta vantajosa na determinação do índice de conforto térmico e zoneamento bio-climático.

Deb et al. (2015) compara o desempenho de MLP com duas camadas escondidas de RM para predição da motilidade pós-congelamento de sêmen de touros de gado leiteiro e do efeito combinado de alelos microssatélites específicos de Y na motilidade pós-congelamento. Os resultados encontrados apontaram para uma menor precisão da predição da motilidade, que foi atribuída à maior variabilidade no conjunto de dados. O trabalho também mostrou uma

maior acurácia (34,88%) e menor erro ($EMQ = 8,4353$) dos modelos MLP em comparação com a RM (32,04% e $EMQ = 8,6168$), validando a metodologia para a predição de motilidade.

Com o objetivo de avaliar a sustentabilidade da pecuária bovina em ambientes tropicais, Santos et al. (2017) descrevem um sistema de apoio à decisão baseado na abordagem de regras *Fuzzy* chamado Rancho Sustentável do Pantanal. O sistema desenvolvido permitiu obtenção de resultados claros para diferentes níveis de atributos, dimensões e índices gerais, essenciais para o diagnóstico e processo de tomada de decisão, segundo os autores. Eles pontuam também que a mesma abordagem pode ser utilizada para outros ecossistemas usando informações específicas daquela região.

Chaparro et al. (2011) desenvolveram um sistema *Fuzzy* para determinar os níveis de infecção no pasto em qualquer época do ano na região dos Pampas. A comparação entre as simulações e modelos obtidos por outros trabalhos de campo, segundo os autores do texto, alcançou resultados satisfatórios.

Simsek et al. (2015) utilizou duas abordagens para geração de ANFIS, Particionamento de Rede (PR) e Agrupamento Subtrativo (AS), para estimar as concentrações de nutrientes no escoamento de sistemas de confinamento. A metodologia também foi comparada com um trabalho anterior (CEMEK; RAHMAN; RAHMAN, 2013) que utilizou RNA. Os resultados mostraram que os modelos ANFIS de ambas abordagens previram com sucesso a concentração de nutrientes no escoamento superficial, com a metodologia PR ligeiramente melhor. Em comparação com os modelos RNA, a abordagem ANFIS se mostrou superior.

Ghazanfari, Nobari e Tahmoorespur (2011) utilizaram com sucesso modelos MLP para prever a produção de ovos de galinha utilizando modelos com duas camadas escondidas. Os autores encontraram valores elevados do Coeficiente de Determinação ($r^2 > 0.94$) validando a metodologia como uma ferramenta eficaz e precisa para prever a produção de ovos.

3 METODOLOGIA E EXECUÇÃO DO PROJETO

Este capítulo apresenta o banco de dados e trata da metodologia utilizada para o pré-processamento dos dados, aplicação dos modelos de RM e elaboração dos modelos de IC. São apresentadas brevemente algumas informações do funcionamento das técnicas e posteriormente os parâmetros utilizados para execução de cada uma delas.

3.1 Banco de Dados

Foi utilizado um conjunto de dados do Sistema BR-CORTE (VALADARES FILHO et al., 2016) para elaboração e comparação dos modelos aqui propostos e implementados. O BR-CORTE é um compilado de informações e recomendações com tabelas sobre as exigências nutricionais de zebuínos puros e cruzados, além de tabelas com composição de alimentos. O sistema atual está em sua terceira versão e disponibiliza um *software* para elaboração de dietas e cálculo das exigências nutricionais dos animais.

O banco de dados contém dados de 31 experimentos comparativos de abate (GARRETT; MEYER; LOFGREEN, 1959), com informações individuais de 840 animais. O conjunto de dados foi composto por 558 touros, 175 novilhos e 107 novilhas; 544 Zebu de raça pura e 296 mestiços; e 672 animais de confinamento e 168 de pastagem.

O banco de dados possui cinco parâmetros que são usados para a elaboração dos modelos, sendo eles: gênero, sistema de alimentação, grupo genético, Peso Corporal Vazio Médio (PCVZm) (peso da carcaça do animal, excluídos parasitas, dejetos, gases, etc.) e Ganho de Peso de Corpo Vazio (GPCVZ) (taxa de ganho de peso diária desejada). Este parâmetros são considerados independentes e são utilizados pois, além de serem mais fáceis de se obter, apresentam relevância, do ponto de vista biológico e fisiológico, para a determinação das exigências energéticas dos animais. Os três primeiros parâmetros (gênero, sistema de alimentação e grupo genético) são categóricos, enquanto PCVZm e GPCVZ são numéricos. O banco de dados também contém o valor medido para o Consumo de Energia Metabolizável (CEM), que é utilizado pelas técnicas como valor alvo (ou saída desejada). A seguir, a Tabela 3.1 apresenta os valores assumidos pelos parâmetros categóricos, tanto nominal quanto numericamente, como retirados do banco de dados.

Tabela 3.1 – Parâmetros categóricos

Parâmetros	Valor Nominal	Codificação Numérica	
Gênero	Macho	1	
	Castrado	2	
	Fêmea	3	
Alimentação	Confinamento	1	
	Pastagem	2	
Grupo Genético	Zebuínos de raça pura	1	
	Zebuínos cruzados com raças Taurinas de Leite	2	
	Zebuínos cruzados com raças Taurinas de Corte (Continental)	3	
	Zebuínos cruzados com raças Taurinas de Corte (Britânico)	4	

Fonte: Autor

Devido ao baixo número de exemplos para os grupos genéticos 3 e 4 e em concordância com a abordagem tradicional, esses foram unificados em um único bloco com codificação numérica 3.

É prática comum dividir-se o conjunto de dados (quando suficientemente grande) em subconjuntos distintos que são usados para o treinamento e validação do modelo. O primeiro conjunto tem como objetivo representar o problema de forma que o algoritmo de treinamento (ou técnica de modelagem) possa ajustar os parâmetros do modelo corretamente. O segundo conjunto funciona como uma métrica de qualidade: uma vez encontrados os parâmetros, o modelo resultante é testado nesse subconjunto, se as predições encontradas são acuradas, então o modelo é válido.

Para cada treinamento realizado neste trabalho o banco de dados foi reordenado por um vetor de números inteiros pseudo-aleatórios distribuídos uniformemente com tamanho igual a 840 (número total de animais) no intervalo [1, 840]. Com esse sorteio, espera-se que os conjuntos de treinamento e validação posteriormente construídos sejam representativos do todo.

Os conjuntos de treinamento e validação foram gerados variando-se a porcentagem de animais considerados dentro do intervalo de 10 a 90%, sendo um conjunto o complemento do outro, assim, por exemplo, quando o conjunto de treinamento contém 60% dos dados ($n = 504$ animais) o conjunto de validação contém 40% ($n = 336$ animais). Esta variação tem o intuito de investigar o efeito do número de dados nas estimações. Os mesmos conjuntos são utilizados para treinar e validar todos os métodos.

3.2 Pré-processamento

Após a divisão dos dados nos subconjuntos de treino e teste, o único tratamento feito nos dados foi a normalização, que corresponde a um importante passo no processo de elaboração dos modelos de IC. Ela permite que todos os parâmetros sejam apresentados dentro de uma mesma escala, impedindo dessa forma que os algoritmos deem preferência a algum parâmetro exclusivamente devido ao seu valor absoluto durante o processo de treinamento (THEODORIDIS; KOUTROUMBAS, 2008; LECUN et al., 2012).

A despeito de existirem diversas formas de normalização dos dados, como as sugeridas em Theodoridis e Koutroumbas (2008), este trabalho adotou como regra de normalização o mapeamento dentro do intervalo $[-1, 1]$. Este tipo de normalização possui algumas vantagens, como possuir média zero e coincidir com os limites da segunda derivada das funções de ativação da camada escondida dos modelos *backpropagation*, o que pode auxiliar no treinamento deste tipo de algoritmo como apontado por LeCun et al. (2012). A Equação (3.1), a seguir, mostra como foi feita a normalização.

$$\mathbf{y} = \frac{(y_{max} - y_{min}) \cdot (\mathbf{x} - x_{min})}{(x_{max} - x_{min})} + y_{min} \quad (3.1)$$

em que, \mathbf{y} , y_{max} e y_{min} representam o vetor de parâmetros normalizado, o valor máximo esperado para o vetor normalizado (1, neste caso) e o valor mínimo para o vetor normalizado (-1, neste caso), respectivamente. Os valores \mathbf{x} , x_{max} e x_{min} são o vetor de parâmetros original, o valor máximo desse vetor e seu valor mínimo, respectivamente. Como usada nesse trabalho, a Equação (3.1) pode ser reescrita como:

$$\mathbf{y} = 2 \cdot \frac{(\mathbf{x} - x_{min})}{(x_{max} - x_{min})} - 1 \quad (3.2)$$

Vale notar que a Equação (3.2) é aplicada ao subconjunto de treinamento, portanto, os valores de x_{max} e x_{min} assumidos por essas variáveis são os correspondentes a esses dados e devem ser utilizados em todas as normalizações subsequentes, no que diz respeito ao parâmetro de interesse. Assim, a normalização dos dados de validação utiliza os valores x_{min} e x_{max} dos dados de treinamento.

3.3 Abordagem Clássica

Como apresentado no Apêndice A, para a determinação do Consumo de Energia Metabolizável (CEM) é necessário estimar-se os valores de Energia Líquida de Manutenção (EL_m) e as eficiências de uso da energia de manutenção e ganho, k_m e k_g , respectivamente.

A seguir serão apresentados os modelos de regressão adotados pelo Sistema BR-CORTE (VALADARES FILHO et al., 2016), que atualmente constitui as bases das recomendações energéticas para o gado brasileiro e tem sua metodologia já bem estabelecida e amplamente aceita. Vale notar que cada um deles foi testado em cada variável categórica, apresentadas na Tabela 3.1, ou seja, é verificado se existe diferença significativa caso o modelo seja aplicado para diferentes grupos genéticos, sexos e sistemas de alimentação, sendo então criados a partir do mesmo modelo diferentes conjuntos de variáveis, que se ajustam ao grupo desejado.

3.3.1 Modelos BR-CORTE

O Sistema BR-CORTE, como apresentado na Seção 3.1, é um compilado de informações e recomendações nutricionais para bovinos. Um de seus capítulos traz a metodologia para o cálculo das exigências energéticas do gado. A seguir são apresentados os modelos assumidos no BR-CORTE e os parâmetros obtidos.

I. Exigências de Energia Líquida para Manutenção (EL_m)

A EL_m é calculada através de uma regressão não-linear exponencial da produção de calor ($PCalor$) em função do Consumo de Energia Metabolizável (CEM). Nesse ponto, a EL_m é admitida como sendo o intercepto (β_0) do modelo (FERRELL; JENKINS, 1998).

$$PCalor = \beta_0 \cdot e^{(\beta_1 \cdot CEM)} \quad (3.3)$$

Onde $PCalor$ é a produção de calor, CEM é o consumo de energia metabolizável e β_0 e β_1 são os parâmetros do modelo.

As Equações (3.4)-(3.7) apresentam os modelos recomendados pelo Sistema.

- Animais em Confinamento:

- Zebuínos:

$$PCalor = 0,0749 \cdot e^{(3,8684 \cdot CEM)} \quad (3.4)$$

- Cruzados de Corte:

$$PCalor = 0,0749 \cdot e^{(4,0612 \cdot CEM)} \quad (3.5)$$

- Cruzados de Leite:

$$PCalor = 0,0749 \cdot e^{(4,1487 \cdot CEM)} \quad (3.6)$$

- Animais em Pastejo:

$$PCalor = 0,0749 \cdot e^{(4,1986 \cdot CEM)} \quad (3.7)$$

II. Exigência de Energia metabolizável para Manutenção (EM_m)

A obtenção da EM_m é condicionada ao cálculo da eficiência de utilização da energia metabolizável para manutenção (k_m). Estudos realizados por Marcondes et al. (2013) mostraram que efeitos da eficiência de uso da energia metabolizável para ganho (k_g) e do Ganho de Peso de Corpo Vazio (GPCVZ) afetam k_m , mostrando que o desempenho dos animais afetam os requisitos de manutenção. Por tanto, o modelo adotado para a estimação do valor de k_m incorpora esses elementos.

$$EM_m = EL_m / k_m \quad (3.8)$$

$$k_m = [(\beta_0 + \beta_1 \cdot k_g + \beta_2 \cdot GPCVZ) \cdot \theta] \quad (3.9)$$

onde β_0 , β_1 e β_2 são os parâmetros da equação e θ é um fator de ajuste para o sistema de criação (pastejo ou confinamento). As Equações (3.10)-(3.13) apresentam os modelos obtidos pelo sistema.

- Animais em Confinamento:

- Zebuínos:

$$k_m = [(0,513 + 0,173 \cdot k_g + 0,100 \cdot GPCVZ) \cdot 1] \quad (3.10)$$

- Cruzados de Corte:

$$k_m = [(0,513 + 0,173 \cdot k_g + 0,073 \cdot GPCVZ) \cdot 1] \quad (3.11)$$

- Cruzados de Leite:

$$k_m = [(0,513 + 0,173 \cdot k_g + 0,010 \cdot GPCVZ) \cdot 1] \quad (3.12)$$

- Animais em Pastejo:

$$k_m = [(0,513 + 0,173 \cdot k_g + 0,100 \cdot GPCVZ) \cdot 0,92] \quad (3.13)$$

III. Peso de Corpo Vazio Equivalente ($PCVZ_{eq}$)

Para determinação do $PCVZ_{eq}$, antes deve-se determinar o Peso de Corpo Vazio à Maturidade ($PCVZ_{mat}$). Estabelece-se o $PCVZ_{mat}$ com base na relação exponencial entre o conteúdo corporal de gordura e o $PCVZ$ dos animais, com 25% de gordura corporal como ponto de maturidade. O modelo é apresentado na Equação (3.14).

$$GC = \beta_0 \cdot e^{(\beta_1 \cdot PCVZ)} \quad (3.14)$$

onde GC é o conteúdo de gordura corporal, $PCVZ$ é o peso de corpo vazio e β_0 e β_1 são os parâmetros da equação. A Tabela 3.2 apresenta as recomendações para diferentes grupos genéticos e sexos.

Tabela 3.2 – Pesos de corpo vazio à maturidade para diferentes grupos genéticos/classes sexuais de bovinos de corte.

Grupo Genético	Classe Sexual	Peso à Maturidade ($PCVZ_{mat}$)
Zebuínos	Macho não Castrado	517
	Macho Castrado	433
	Fêmea	402
Cruzados de Corte	Macho não Castrado	560
	Macho Castrado	482
	Fêmea	417
Cruzados de Leite	Macho não Castrado	616
	Macho Castrado	532
	Fêmea	493

Fonte: Adaptado de Valadares Filho et al. (2016).

Tomando-se então um valor de referência, como sugerido pelo NRC (2000), o Peso de Corpo Vazio Equivalente pode ser determinado pela Equação 3.15

$$PCVZ_{eq} = (PCVZ/PCVZ_{mat}) \cdot PCVZ_{ref} \quad (3.15)$$

Em que $PCVZ_{eq}$ é o peso de corpo vazio equivalente, $PCVZ$ é o peso de corpo vazio, $PCVZ_{ref}$ é o peso de corpo vazio referência e $PCVZ_{mat}$ é o peso de corpo vazio à maturidade. O $PCVZ_{ref}$ adotado na edição atual do BR-CORTE é de 517 Kg.

IV. Exigência de Energia Líquida para Ganho (EL_g)

A EL_g é modelada pela relação da energia retida no corpo em função do $PCVZ_{eq}$ e do ganho de peso de corpo vazio desejado:

$$EL_g = \beta_0 \cdot PCVZ_{eq}^{0,75} \cdot GPCVZ^{\beta_1} \quad (3.16)$$

Onde EL_g é a exigência de energia líquida para ganho de peso, $PCVZ_{eq}^{0,75}$ é chamado peso de corpo vazio metabólico equivalente, $GPCVZ$ é o ganho de peso de corpo vazio e β_0 e β_1 são os parâmetros do modelo. São adotadas duas equações para determinar a EL_g , uma para animais em confinamento (3.17) e outra para animais em pastejo (3.18).

$$EL_g = 0,061 \cdot PCVZ_{eq}^{0,75} \cdot GPCVZ^{1,035} \quad (3.17)$$

$$EL_g = 0,052 \cdot PCVZ_{eq}^{0,75} \cdot GPCVZ^{1,062} \quad (3.18)$$

V. Exigência de Energia Metabolizável para Ganho (EM_g)

A EM_g foi calculada seguindo os modelos propostos por Marcondes et al. (2013) para a estimação da Eficiência de utilização da energia metabolizável para ganho (k_g), assim tem-se:

$$EM_g = EL_g/k_g \quad (3.19)$$

$$k_g = \frac{0,327}{0,539 + ER_{prot}} \quad (3.20)$$

Onde k_g é a eficiência de utilização da energia metabolizável para ganho e ER_{prot} é a proporção da energia retida na forma de proteína. O modelo fica então condicionado à correta estimação da ER_{prot} . Neste trabalho, adotou-se o modelo potencial sugerido por Marcondes et al. (2013):

$$ER_{prot} = 1,140 \left(\frac{ER}{GPCVZ} \right)^{-1,137} \quad (3.21)$$

Onde ER_{prot} é a proporção da energia retida na forma de proteína, ER é a energia retida ou a exigência de energia líquida para ganho (EL_g) e $GPCVZ$ é ganho de peso de corpo vazio.

VI. Resumo dos modelos Propostos pelo Sistema BR-CORTE

As tabelas 3.3 e 3.4 apresentam todas as equações obtidas. As tabelas dizem respeito aos animais criados em sistema de confinamento e pastejo, respectivamente, e quando faz-se necessário, são apresentadas as equações separadas em termos do grupo genético. Nenhum dos modelos utilizados nesse trabalho apresentou diferença quanto à classe sexual, excetuando-se o parâmetro $PCVZ_{mat}$, apresentado na Tabela 3.2.

Tabela 3.3 – Modelos adotados pelo Sistema BR-CORTE para estimar as exigências de energia para animais criados em sistema de confinamento

Item	Grupo Genético	Equações
$PCVZeq$	Todos	$(PCVZ/PCVZmat) \cdot 517$
EL_M	Todos	$0,075 \cdot PCVZ^{0,75}$
EL_g	Todos	$0,061 \cdot PCVZeq^{0,75} \cdot GPCVZ^{1,035}$
ER_{prot}	Todos	$1,140 \cdot \left(\frac{EL_g}{GPCVZ}\right)^{-1,137}$
k_g	Todos	$\frac{0,327}{0,539+ER_{prot}}$
k_m	Zebuínos	$[(0,513 + 0,173 \cdot k_g + 0,100 \cdot GPCVZ) \cdot 1]$
	Cruzados de Corte	$[(0,513 + 0,173 \cdot k_g + 0,073 \cdot GPCVZ) \cdot 1]$
	Cruzados de Leite	$[(0,513 + 0,173 \cdot k_g + 0,010 \cdot GPCVZ) \cdot 1]$
EM_m	Todos	EL_m/k_m
EM_g	Todos	EL_g/k_g
EM_{total} ou CEM	Todos	$EM_m + EM_g$

Fonte: Adaptado de Valadares Filho et al. (2016).

Tabela 3.4 – Modelos adotados pelo Sistema BR-CORTE para estimar as exigências de energia para animais criados em sistema de pastejo

Item	Grupo Genético	Equações
$PCVZeq$	Todos	$(PCVZ/PCVZmat) \cdot 517$
EL_M	Todos	$0,075 \cdot PCVZ^{0,75}$
EL_g	Todos	$0,052 \cdot PCVZeq^{0,75} \cdot GPCVZ^{1,062}$
ER_{prot}	Todos	$1,140 \cdot \left(\frac{EL_g}{GPCVZ}\right)^{-1,137}$
k_g	Todos	$\frac{0,327}{0,539+ER_{prot}}$
k_m	Todos	$[(0,513 + 0,173 \cdot k_g + 0,100 \cdot GPCVZ) \cdot 0,92]$
EM_m	Todos	EL_m/k_m
EM_g	Todos	EL_g/k_g
EM_{total} ou CEM	Todos	$EM_m + EM_g$

Fonte: Adaptado de Valadares Filho et al. (2016).

3.3.2 Modelos Lineares

Uma segunda metodologia para estimação do CEM baseada na abordagem estatística foi considerada. Utilizou-se um modelo puramente linear para caracterizar a relação entre as variáveis utilizadas e o CEM, representado pela Equação (3.22).

$$CEM = w_0 + w_1 \cdot G + w_2 \cdot S + w_3 \cdot R + w_4 \cdot PCVZm + w_5 \cdot GPCVZ \quad (3.22)$$

onde w_0-w_5 são os parâmetros do modelo, G representa o gênero do animal, S é o sistema de alimentação e R é o grupo genético. Utilizou-se o Método dos Mínimos Quadrados (MMQ) para a obtenção dos parâmetros, que podem ser calculados pela Equação (3.23).

$$\hat{\mathbf{W}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} \quad (3.23)$$

onde $\hat{\mathbf{W}}$ é o vetor de parâmetros estimado, \mathbf{X} é uma matriz construída com as variáveis de estudo, \mathbf{y} é o vetor com os valores desejados, isto é, os valores reais do CEM. O símbolo $'$ denota uma matriz transposta e a operação $()^{-1}$ denota a inversão de uma matriz.

3.4 Técnicas de IC Propostas

Esta Subseção tratará da metodologia utilizada para aplicação das técnicas de IC propostas, apresentadas no Capítulo 1. Todas as técnicas foram implementadas via *software* MatLab e cada variação utilizada (em termos de arquitetura, variação de dados, etc.) foi compilada 100 vezes.

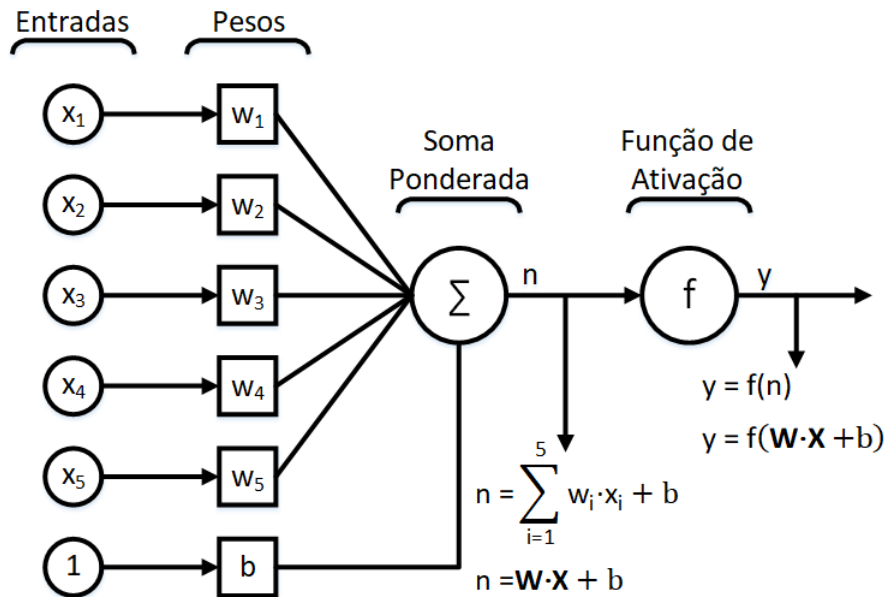
3.4.1 Redes Neurais Perceptron Multicamadas (MLP)

A arquitetura geral das redes MLP implementadas nesse trabalho pode ser vista na Figura 3.2, e sua unidade básica, o *Perceptron*, na Figura 3.1. Tratam-se de redes com cinco parâmetros de entrada, uma ou duas camadas escondidas e um neurônio linear de saída único.

A Figura 3.1 mostra com detalhe o processamento feito pelo neurônio artificial *Perceptron*. Cada um dos $R = 5$ parâmetros de um exemplo são multiplicados pelos seus respectivos pesos sinápticos ($w_i, i = 1, 2, \dots, 5$) e somados com um valor de viés, b , que pode ser incorporado como mais um peso correspondente a um parâmetro de entrada constante igual a um. O resultado dessa soma é a saída “não ativada”, n , que então é apresentada a uma função $f(\cdot)$ para gerar a saída do neurônio, y .

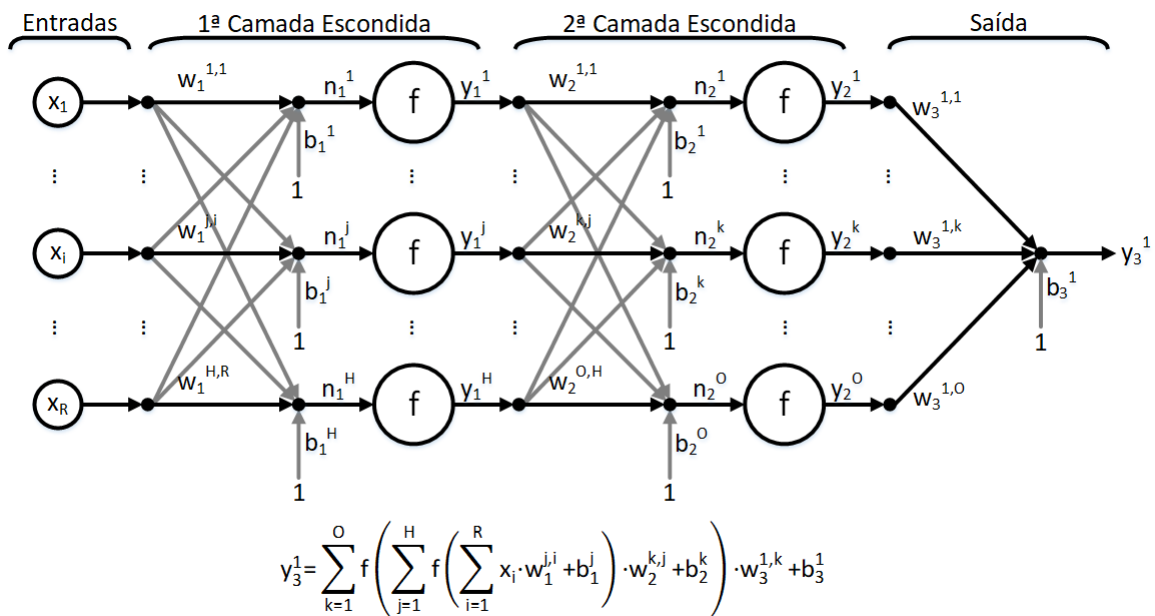
Na Figura 3.2 os *perceptrons* são apresentados de forma simplificada. A unidade de soma é mostrada apenas como o ponto de acoplamento das setas (que nesse caso indicam o sentido dos dados), dando-se destaque apenas para as funções de ativação, $f(\cdot)$. O neurônio de saída por possuir uma função de saída linear pura ($y(x) = x$), tem essa unidade omitida.

Figura 3.1 – Arquitetura de um neurônio artificial *Perceptron* com cinco parâmetros de entrada.



Fonte: Autor.

Figura 3.2 – Arquitetura geral de uma rede MLP com duas camadas escondidas de tamanhos H e O , respectivamente, e um neurônio de saída linear.



Fonte: Autor.

Primeiramente foram construídos modelos com apenas uma camada intermediária, com um máximo de 10 neurônios ($H \in [2, 10]$). Após a construção desses modelos, determinou-se a melhor arquitetura, em relação ao número de neurônios, e foi adicionada uma segunda

camada escondida com um máximo, também, de 10 neurônios ($O \in [2, 10]$), com o intuito de se investigar se uma segunda camada é capaz de agregar desempenho aos modelos.

Para essas redes o método de treinamento utilizado foi o Gradiente Conjugado Escalonado (SCG) (MØLLER, 1993) e o desempenho foi dado em termos do Erro Médio Quadrático (EMQ). Este algoritmo se enquadra na família de algoritmos *backpropagation* e tende a realizar menos cálculos e apresentar taxas de convergências maiores que os demais métodos baseados em gradientes, além de possibilitar a utilização de computação paralela para efetuação dos seus cálculos, o que aumenta ainda mais sua velocidade.

Os parâmetros de treinamento utilizados foram:

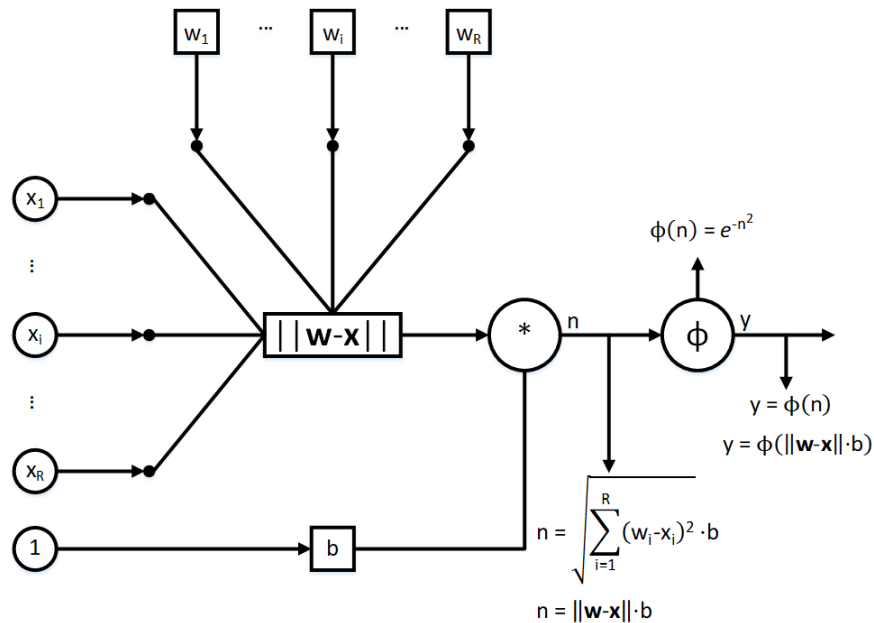
- Máximo de Épocas: 500;
- Tempo Máximo de treinamento: infinito;
- Performance desejada: 0 (erro nulo);
- Gradiente mínimo: 10^{-6} ;
- Sigma: $\sigma = 5 \cdot 10^{-5}$;
- Lambda: $\lambda = 5 \cdot 10^{-7}$.

O parâmetro σ determina a mudança de peso para a aproximação da segunda derivada durante o treinamento. Já λ é um Multiplicador de Lagrange e utilizado como parâmetro para regular a indefinição da Matriz Hessiana.

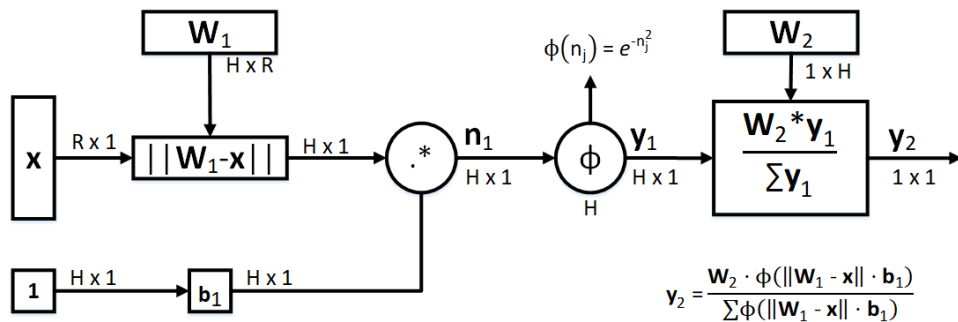
3.4.2 Redes de Funções de Base Radial (RBF)

A arquitetura utilizada para as Redes RBF foi a Rede Neural de Regressão Generalizada (RNRG) (SPECHT, 1991), um tipo de Rede Neural Probabilística (RNP) (SPECHT, 1967; SPECHT, 1990) que utiliza neurônios do tipo RBF. A arquitetura de um neurônio desse tipo e a forma geral da rede são apresentadas nas Figuras 3.3 e 3.4, a seguir.

Figura 3.3 – Arquitetura de um neurônio artificial RBF.



Fonte: Autor.

Figura 3.4 – Arquitetura geral de uma rede RBF do tipo Rede Neural de Regressão Generalizada (RNRG) com uma camada escondida de tamanho H e um neurônio de saída linear.

Fonte: Autor

A rede representada na Figura 3.4, possui duas camadas, a primeira é composta por neurônios do tipo RBF e funciona exatamente como uma Rede RBF convencional (BROOMHEAD; LOWE, 1988) fazendo um mapeamento não-linear em um novo espaço. A segunda camada é uma camada linear levemente modificada, que normaliza as saídas da primeira camada antes de apresentar ao neurônio linear, que realiza uma classificação linear no espaço gerado pela primeira camada.

As RNRG foram implementadas com o número de parâmetros de entrada R igual a cinco, tal qual as Redes MLP. A primeira camada possui tantos neurônios quanto o número de exemplos no vetor de entrada \mathbf{X} . Os pesos da primeira camada, \mathbf{W}_1 , são iniciados como $\mathbf{W}_1 = \mathbf{X}^T$ e os valores de viés, \mathbf{b}_1 , são iniciados como $\mathbf{b}_1 = 0.8326/\sigma$, onde a constante σ é

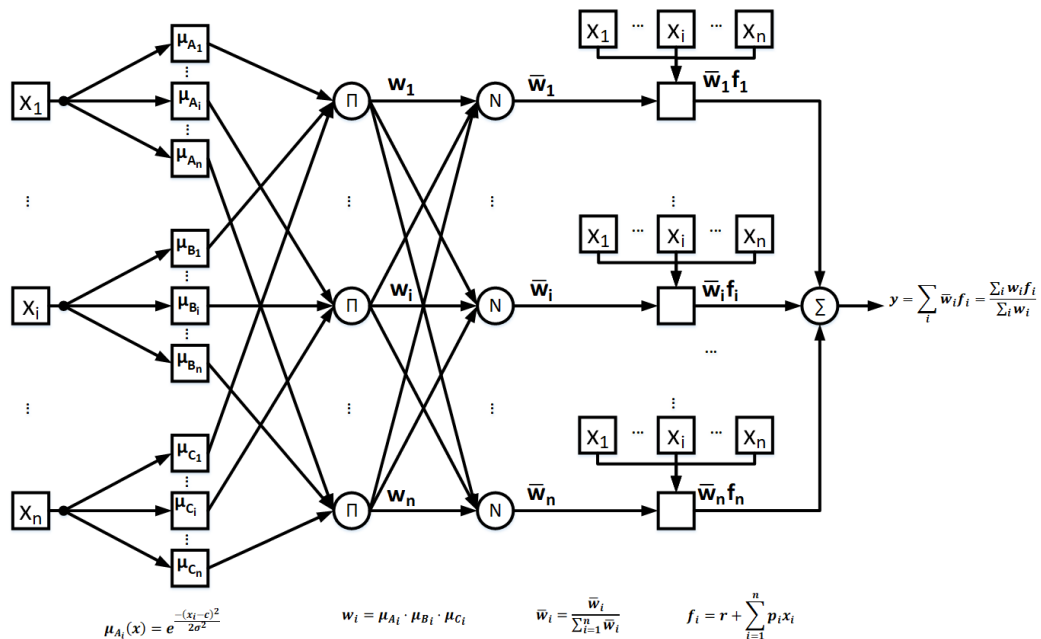
chamada constante de espalhamento é foi definida como $\sigma = 0,1$. Isso significa que quando o vetor de entrada tem distância zero em relação ao vetor de pesos, a saída dessa camada é um, quando a distância é igual a σ , a saída dessa camada é 0,5. A segunda camada tem os pesos iniciados com os valores de alvo, $\mathbf{W}_2 = \mathbf{T}$. O desempenho do modelo foi dado em termos do EMQ.

Essa abordagem para determinação dos pesos segue o seguinte raciocínio: suponha um vetor de entrada \mathbf{p} próximo de \mathbf{p}_i , um dos vetores utilizados na construção da rede, portanto um dos pesos da primeira camada. Esse vetor de entrada \mathbf{p} produz uma saída próxima de 1 na primeira camada, isto leva a uma saída da camada 2 perto do alvo t_i , um dos usados para formar a camada 2.

3.4.3 Sistemas de Inferência Fuzzy (SIF)

Os SIF utilizados nesse trabalho são do tipo Sugeno (também chamados Takagi-Sugeno) (SUGENO, 1985; TAKAGI; SUGENO, 1985) de primeira ordem, isto é, utilizam funções lineares como Funções de Pertinência (FP) de saída. O sistema foi convertido para a arquitetura Sistemas de Inferência Neuro-Fuzzy Adaptativos (ANFIS), que tem sua estrutura geral apresentada na Figura 3.5.

Figura 3.5 – Arquitetura típica de um ANFIS com n entradas.



Fonte: Autor.

O processo de criação e ajuste de um ANFIS envolve a geração de um SIF inicial e sua conversão para a nova arquitetura. Para essa tarefa, combinou-se dois métodos de Agrupamento *Fuzzy*: o Agrupamento Subtrativo (AS) (CHIU, 1994) e o Agrupamento *Fuzzy C-Means* (FCM) (BEZDEK, 1981). O AS é utilizado para estimar o número de agrupamentos presente no banco de dados e então usado como parâmetro para o FCM que gera um modelo SIF inicial. Os parâmetros utilizados em cada técnica foram:

I. Agrupamento Subtrativo (AS):

- Raio de Influência do Agrupamento: 0,5;
- Normalização: Mapeamento no intervalo [0, 1] (Equação 3.1);
- Fator de Achatamento: 1,25;
- Razão de Aceitação: 0,5;
- Razão de Rejeição: 0,15.

II. Agrupamento *Fuzzy C-Means* (FCM):

- Tipo de SIF: Sugeno;
- Número de Agrupamentos: Algoritmo AS;
- Expoente da Matriz de Partição *Fuzzy*: 2;
- Número Máximo de Iterações: 100;
- Melhoria Mínima da função custo: 10^{-5} .

O algoritmo FCM gera então um SIF com FP de entrada do tipo gaussianas e FP de saída do tipo lineares. Essas funções são apresentadas nas Equações (3.24) e (3.25) a seguir.

$$f(x_i) = e^{-\frac{(x_i-c)^2}{2\sigma^2}} \quad (3.24)$$

$$f(\mathbf{x}) = r + \sum_{i=1}^n p_i \cdot x_i \quad (3.25)$$

Onde x_i é o i -ésimo parâmetros de entrada e os valores c , σ , r , p_i são as constantes a serem ajustadas pelo treinamento do ANFIS. O SIF gerado também tem por características para cada agrupamento *Fuzzy*: associar uma FP Gaussiana à cada variável de entrada; ter uma regra;

associar uma FP Linear à cada variável de saída. Além disso, o sistema tem como propriedades ter apenas uma saída; não ter regras compartilhadas, isto é, regras diferentes não usam a mesma FP de saída; e cada regra tem peso um.

Esse sistema então tem seus parâmetros ajustados pelo algoritmo de treinamento do ANFIS, uma abordagem híbrida entre o Método dos Mínimos Quadrados (MMQ) e o gradiente descendente (JANG; SUN; MIZUTANI, 1997). O algoritmo foi configurado da seguinte forma:

I. Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS)

- Número máximo de iterações: 10;
- Performance desejada: 0 (erro nulo);
- Passo inicial: 0,01;
- Taxa de decréscimo do Passo: 0,9;
- Taxa de acréscimo do Passo: 1.1;
- Método de otimização: Híbrido.

3.5 Comparações

Esta subseção apresenta as métricas utilizadas para validação e comparação entre as técnicas propostas e a metodologia clássica.

I. Intramétodos

Cada execução de uma mesma metodologia foi comparada com as demais em termos do Índice de Correlação Linear de Pearson. A correlação de Pearson entre duas variáveis X e Y pode ser calculada pela Equação (3.26):

$$\rho = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{\text{cov}(X, Y)}{\sqrt{\text{var}(X) \cdot \text{var}(Y)}} \quad (3.26)$$

Onde x_i e y_i são os valores medidos de ambas variáveis; e \bar{x} e \bar{y} são as médias aritméticas de ambas variáveis. Os valores que cada uma das variáveis assume são a estimativa obtida por um determinado modelo e os valores reais presentes no banco de dados.

Calculados os índices de correlação para todas execuções, cada uma das arquiteturas assumidas pela técnica são comparadas entre si utilizando o Teste de Tukey da Diferença Honestamente Significativa (TUKEY, 1949), um teste para comparação múltipla de médias. O procedimento para realização do teste pode ser feito como se segue.

Suponha k observações independentes, Y_1, \dots, Y_k , de uma distribuição normal com média μ e variância σ^2 . A faixa dinâmica para esse conjunto de observações é dado por:

$$w = \max(Y_i) - \min(Y_i). \quad (3.27)$$

Suponha agora a estimativa s^2 da variância σ^2 , baseada nos $N - k$ graus de liberdade e independente de Y_i , onde N é o número total de observações. A razão w/s é chamada de Amplitude Studentizada e é denotada por:

$$q(k, N - k) = \frac{w}{s}, \quad (3.28)$$

onde q é um valor tabelado.

Para tamanhos de amostras iguais (dados balanceados), o Teste de Tukey declara duas médias significativamente diferentes se o valor absoluto de duas diferenças amostrais ultrapassar

$$TSD = q_\alpha(k, N - k) \sqrt{\frac{QME}{n}}, \quad (3.29)$$

onde QME é o quadrado médio dos resíduos e n é o número de réplicas do nível. Rejeita-se a igualdade da média de dois níveis se $|\bar{y}_i - \bar{y}_j| > TSD$. Um intervalo de confiança de $1 - \alpha$ para a diferença entre todos os pares das médias é dado como:

$$\bar{y}_i - \bar{y}_j - TSD \leq \mu_i - \mu_j \leq \bar{y}_i - \bar{y}_j + TSD, \quad i \neq j \quad (3.30)$$

II. Entre métodos

Uma vez definida cada arquitetura para os diferentes métodos avaliados (RM, MLP, RBF, AN-FIS) são comparados os 100 modelos gerados de cada método.

A avaliação da adequação de um modelo para um determinado propósito deve ser feita através da combinação de várias análises estatísticas. Portanto, foram utilizadas diversas métricas para comparação entre os diferentes métodos analisados, como sugerido por Tedeschi (2006). Além do Índice de Correlação Linear de Pearson (ρ), definido na Equação (3.26), os modelos serão avaliados levando-se em conta seus desempenhos em termos do Coeficiente de Determinação (r^2), Acurácia (A_ρ) (LIAO, 2003), Coeficiente de Correlação de Concordância (CCC)($\hat{\rho}_c$) (LIN, 1989) e do Erro Médio Quadrático de Predição (EMQP) (BIBBY; TOUTENBURG, 1977) e suas decomposições: o Viés Médio, Viés Sistemático e Erros Aleatórios.

Cada uma dessas métricas são brevemente discutidas e tem suas formulações apresentadas a seguir. Todas as formulações foram realizadas pelo *software Model System Evaluation* (MES, v 3.1.17, <http://nutritionmodels.com/mes.html>) (TEDESCHI, 2006).

a) Coeficiente de Determinação (r^2):

O Coeficiente de Determinação r^2 é uma medida estatística de quão bem as previsões de regressão se aproximam dos pontos de dados reais. Um r^2 de 1 indica que as previsões de regressão se ajustam perfeitamente aos dados. Seu cálculo é feito através da Equação 3.31.

$$r^2 = 1 - \frac{\sum_{i=1}^n (Y_i - f_i(X_1, \dots, X_p))^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} \quad (3.31)$$

Onde $f_i(X_1, \dots, X_p)$ é um vetor de n previsões feitas a partir de n pontos sobre todas as variáveis e Y_i é o vetor de valores observados das variáveis sendo preditas. \bar{Y} é o valor médio do vetor de valores observados, Y_i .

b) Acurácia (A_p):

A Acurácia denota o quão próximo um valor medido é de um padrão ou valor real. Ela pode ser entendida também como uma descrição dos erros sistemáticos, isto é, erros inerentes ao sistema utilizado. Uma forma de se quantificar a acurácia é:

$$A_p = \frac{4\sigma_1\sigma_2 - \rho(\sigma_1^2 + \sigma_2^2)}{(2 - \rho)(\sigma_1^2 + \sigma_2^2) + (\mu_2 - \mu_1)^2} \quad (3.32)$$

Onde os índices 1 e 2 fazem referência aos vetores de valores observados e preditos, respectivamente, σ_i é o desvio padrão, ρ é a correlação linear, σ_i^2 é a variância e μ_i é o valor médio.

c) Coeficiente de Correlação de Concordância (CCC) ($\hat{\rho}_c$):

O Coeficiente de Correlação de Concordância (CCC), também conhecido como Índice de Reprodutibilidade, tem sido sugerido para explicar simultaneamente ambos acurácia e precisão. Ele pode ser calculado como:

$$\hat{\rho}_c = \frac{2 \cdot \sigma_{f(X_1, \dots, X_p)Y}}{\sigma_Y^2 + \sigma_{f(X_1, \dots, X_p)}^2 + (\bar{Y} - \bar{f}(X)1, \dots, X_p)^2} \quad (3.33)$$

Onde Y são os dados observados, $f(X_1, \dots, X_p)$ são os valores preditos, σ e σ^2 denotam o desvio padrão e a variância, respectivamente, e \bar{Y} e \bar{f} são os valores médios de Y e $f(X_1, \dots, X_p)$, respectivamente. A Equação (3.33) pode ser expandida em dois termos, como mostrado na Equação (3.34), onde o primeiro termo representa a precisão e é equivalente à correlação linear (ρ , Eq. (3.26)), e o segundo termo (C_b) é o fator de correção de viés, que indica o quão longe a linha de regressão se distancia da inclinação unitária, indicada nas Equações (3.35)-(3.37).

$$\hat{\rho}_c = \rho \times C_b \quad (3.34)$$

$$C_b = \frac{2}{\left[v + \frac{1}{v} + \mu^2\right]} \approx A_\rho \quad (3.35)$$

$$v = \frac{\sigma_1}{\sigma_2} \quad (3.36)$$

$$\mu = \frac{\mu_1 - \mu_2}{\sqrt{\sigma_1 \sigma_2}} \quad (3.37)$$

d) Erro Médio Quadrático de Predição (EMQP):

O EMQ avalia a precisão da regressão linear ajustada usando a diferença entre os valores observados (Y_i) e os valores preditos pela regressão (\hat{Y}_i). Em contraste, EMQP consiste na diferença entre os valores observados (Y_i) e os valores preditos pelo modelo ($f(X_1, \dots, X_p)_i$) em vez do valor predito pela regressão.

$$EMQP = \frac{\sum_{i=1}^n (Y_i - f(X_1, \dots, X_p)_i)^2}{n} \quad (3.38)$$

Mosbaek e Theil (1962) introduziram métodos para decompor as fontes de variação do EMQP para analisar a adequação do modelo. A Equação (3.38) pode ser expandida e solucionada para medidas conhecidas de regressão linear ao invés de pares de dados individuais, como observado na Equação (3.39).

$$EMQP = (\bar{f}(X_1, \dots, X_p) - \bar{Y})^2 + \sigma_{f(X_1, \dots, X_p)}^2 + \sigma_Y^2 - 2 \cdot \rho \cdot \sigma_{f(X_1, \dots, X_p)} \cdot \sigma_Y \quad (3.39)$$

Onde \bar{f} e \bar{Y} denotam a média dos dados preditos e observados, respectivamente, σ^2 é a variância, ρ é a correlação linear e σ é o desvio padrão. Rearranjos adicionais produzem uma forma interpretável do EMQP com três termos.

$$EMQP = (\bar{f}(X_1, \dots, X_p) - \bar{Y})^2 + (\sigma_{f(X_1, \dots, X_p)} - \rho \cdot \sigma_Y)^2 + (1 - r^2) \cdot \sigma_Y^2 \quad (3.40)$$

Os termos da Equação (3.40) representam erros na tendência central (viés médio), erros devidos a regressão (viés sistemático) e erros devidos a perturbações (erros aleatórios). Atenção deve ser dada ao último termo de erro, pois pode ser um bom indicador de falta de ajuste.

4 RESULTADOS

Este Capítulo apresenta os resultados individuais de cada técnica e uma comparação entre cada abordagem. Os resultados apresentados, a não ser explicitamente dito o contrário, são referentes à Correlação Linear (ρ , Eq. (3.26)) média entre os conjuntos de validação e os conjuntos de estimativas do CEM obtidos pelas 100 execuções de cada modelo.

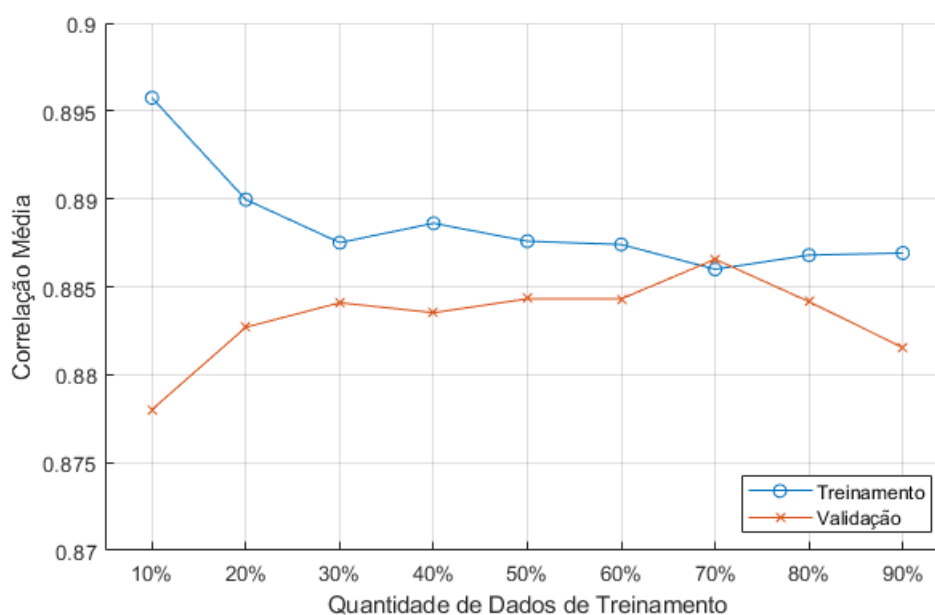
4.1 Definição dos Tamanhos dos Conjuntos de Treinamento e Teste

Os valores observados do CEM presentes no banco de dados são dependentes da realização de experimentos de abate comparativo (GARRETT; MEYER; LOFGREEN, 1959), que são muito caros de se realizar. Portanto, uma técnica que necessite de um número reduzido de exemplos e que ainda assim seja capaz de realizar boas previsões é preferível.

Como apresentado na seção 3.1, as técnicas foram testadas utilizando conjuntos de treino e validação com tamanhos que variaram de 10% a 90% do número total de dados. Os gráficos a seguir mostram o desempenho médio de cada técnica em termos da correlação linear (Eq. (3.26)). São apresentadas as curvas de desempenho para o conjunto de treinamento e teste.

I. Regressão Linear Múltipla - Método dos Mínimos Quadrados (MMQ)

Figura 4.1 – Curvas de treinamento e validação para os modelos Lineares baseados no MMQ.



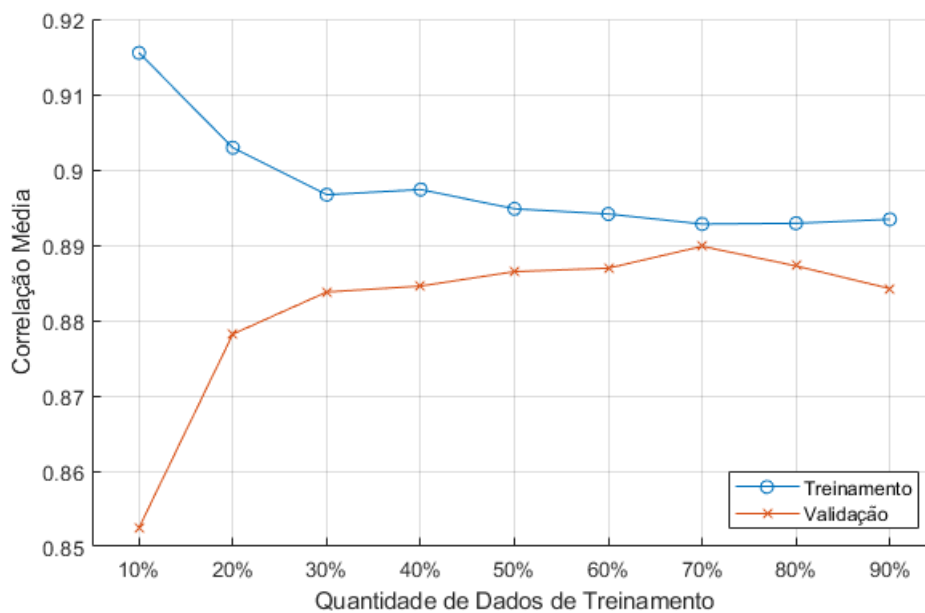
Fonte: Autor

Pela Figura 4.1, observa-se que o melhor desempenho médio de treinamento foi para o conjunto com 10% dos dados, com uma tendência de queda do desempenho com o aumento do número de dados. Entretanto, observa-se o comportamento contrário para o desempenho médio dos conjuntos de validação, que tem um aumento no desempenho até 70% dos dados utilizados para treinamento (30% para validação) e uma posterior queda para os grupos com 80 e 90%.

II. Redes Perceptron Multicamadas (MLP)

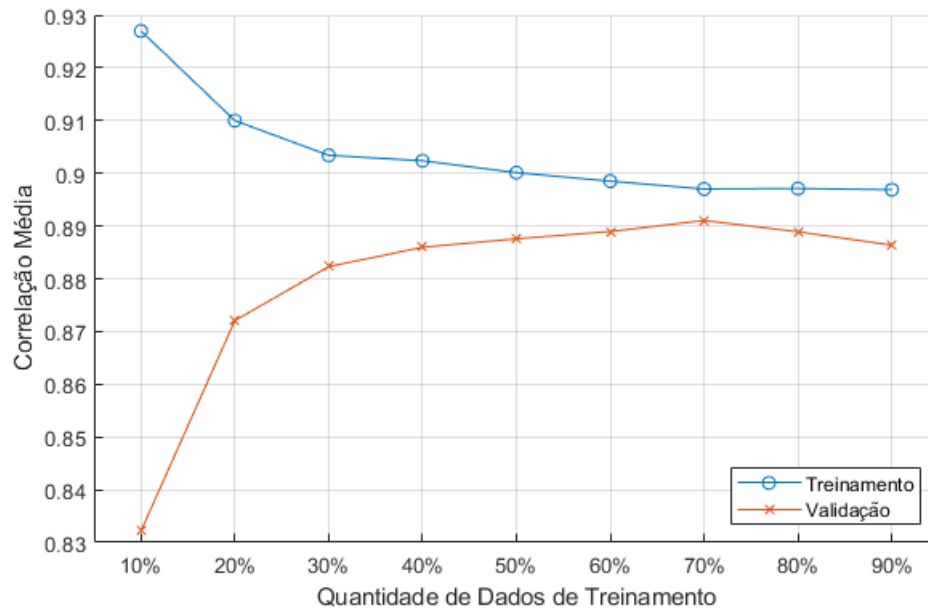
As Figuras 4.2-4.10 apresentam os desempenhos médios para os conjuntos de treinamento e validação das arquiteturas MLP, com cada figura apresentando os resultados para uma quantidade específica de neurônios na camada escondida.

Figura 4.2 – Curvas de treinamento e validação para os modelos MLP com 2 neurônios na camada escondida.



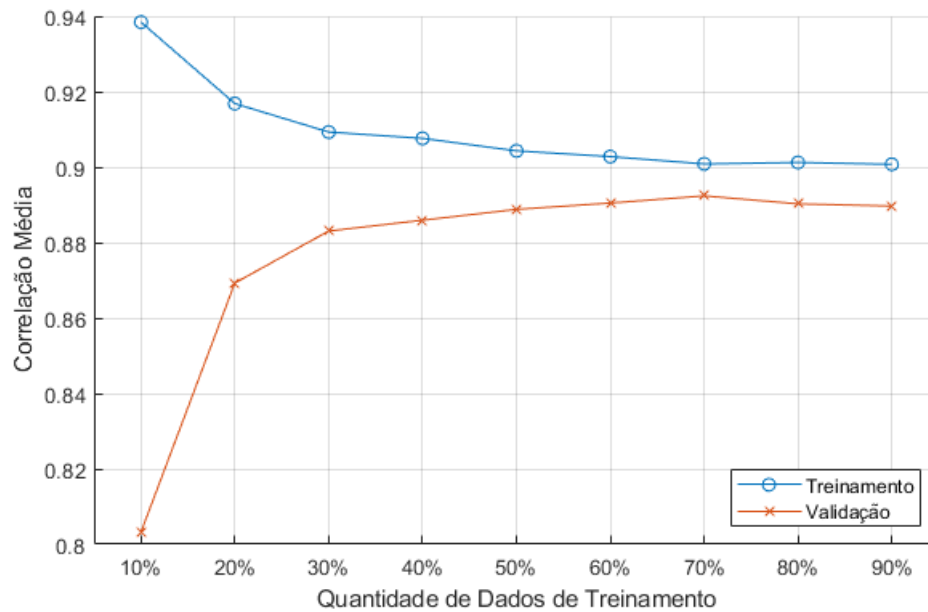
Fonte: Autor

Figura 4.3 – Curvas de treinamento e validação para os modelos MLP com 3 neurônios na camada escondida.



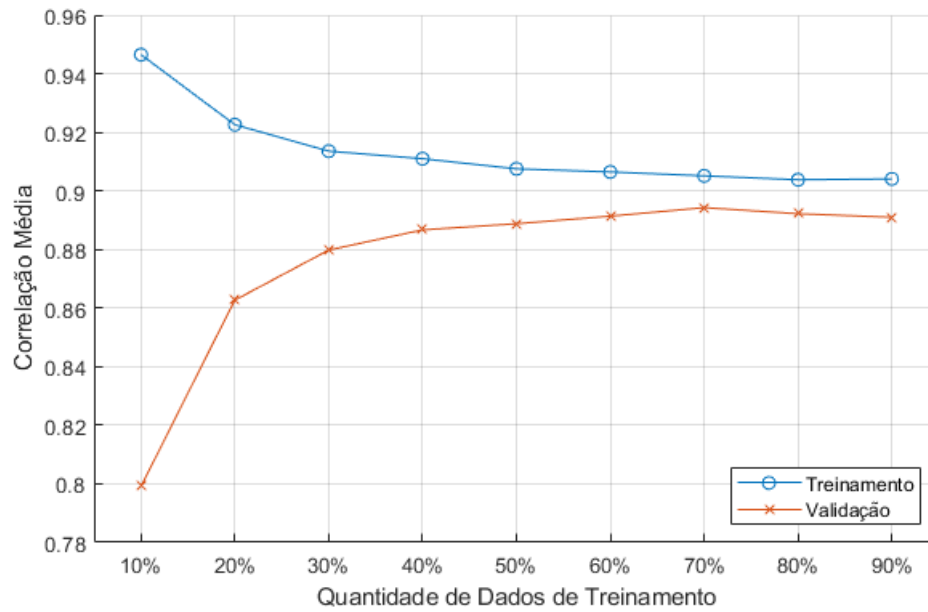
Fonte: Autor

Figura 4.4 – Curvas de treinamento e validação para os modelos MLP com 4 neurônios na camada escondida.



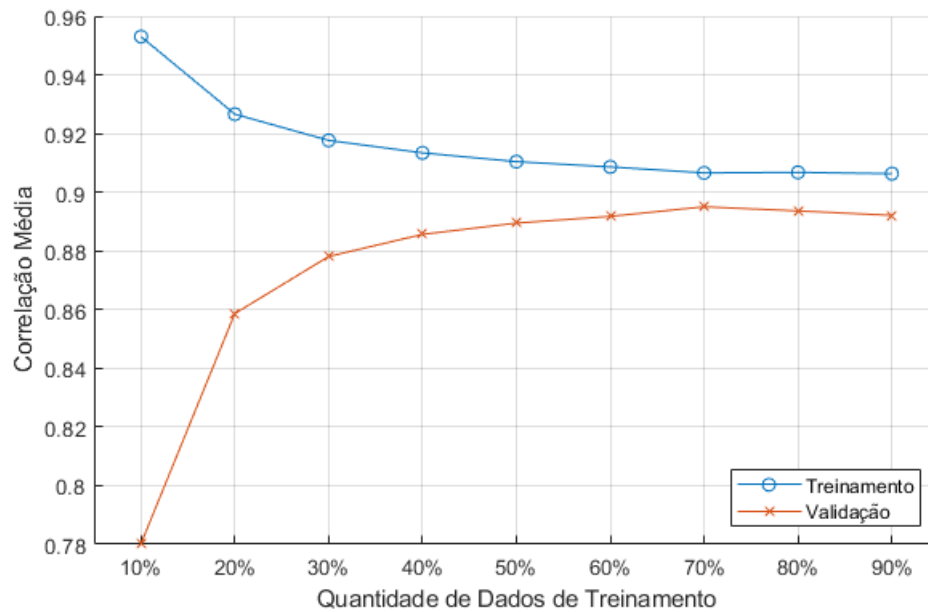
Fonte: Autor

Figura 4.5 – Curvas de treinamento e validação para os modelos MLP com 5 neurônios na camada escondida.



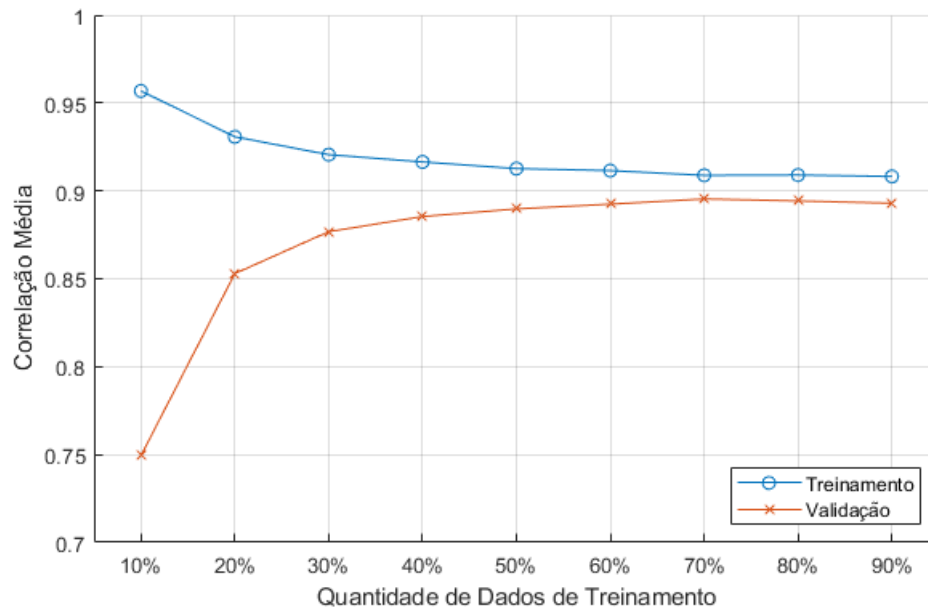
Fonte: Autor

Figura 4.6 – Curvas de treinamento e validação para os modelos MLP com 6 neurônios na camada escondida.



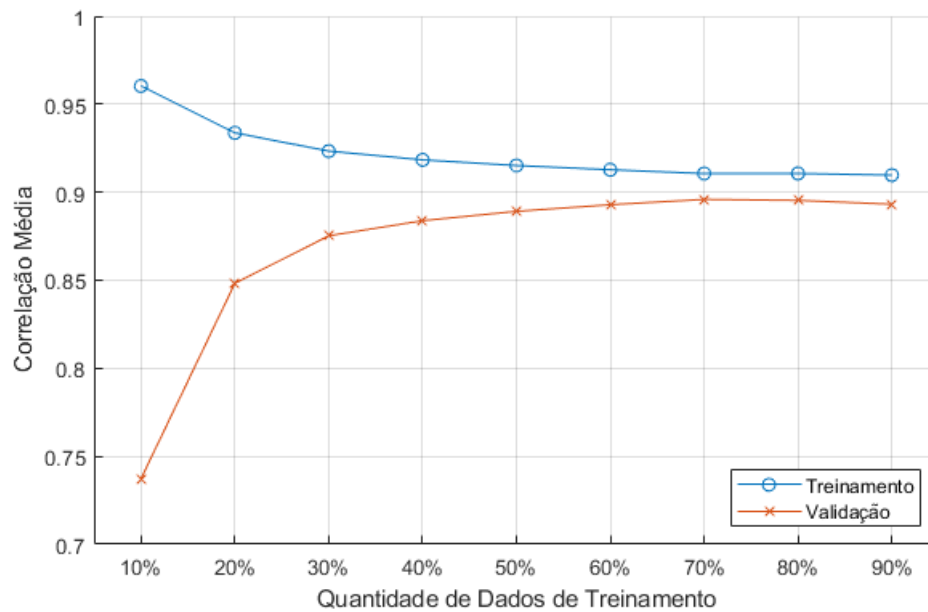
Fonte: Autor

Figura 4.7 – Curvas de treinamento e validação para os modelos MLP com 7 neurônios na camada escondida.



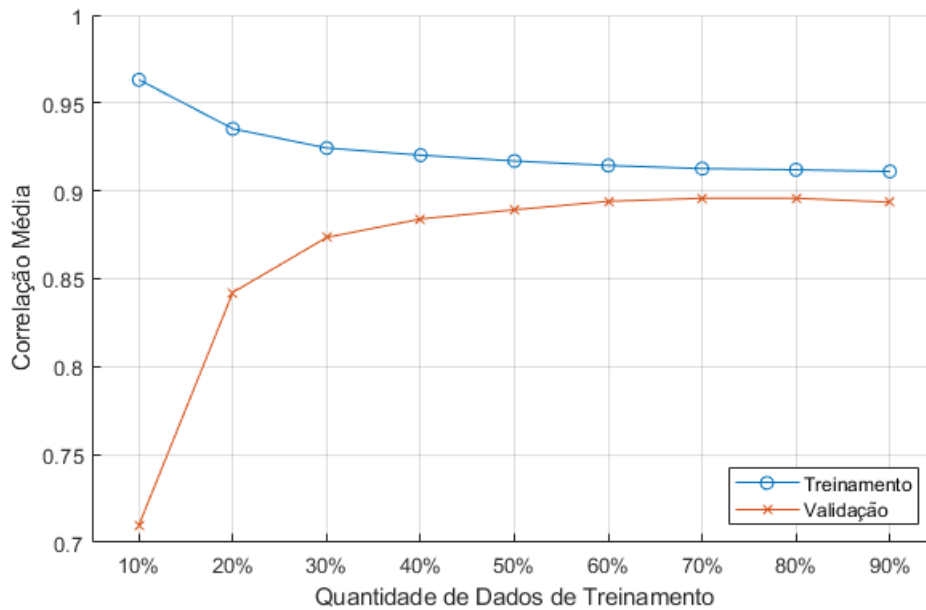
Fonte: Autor

Figura 4.8 – Curvas de treinamento e validação para os modelos MLP com 8 neurônios na camada escondida.



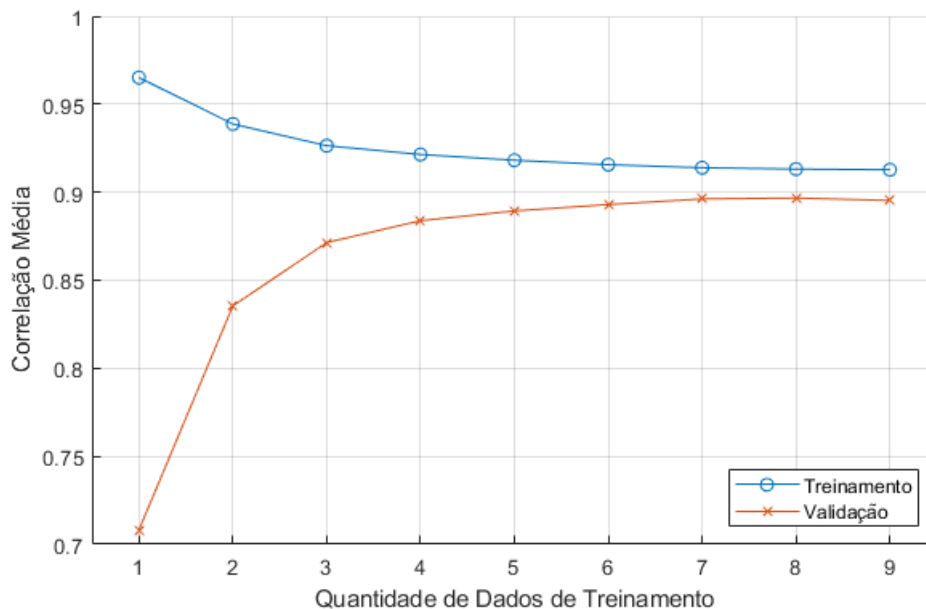
Fonte: Autor

Figura 4.9 – Curvas de treinamento e validação para os modelos MLP com 9 neurônios na camada escondida.



Fonte: Autor

Figura 4.10 – Curvas de treinamento e validação para os modelos MLP com 10 neurônios na camada escondida.



Fonte: Autor

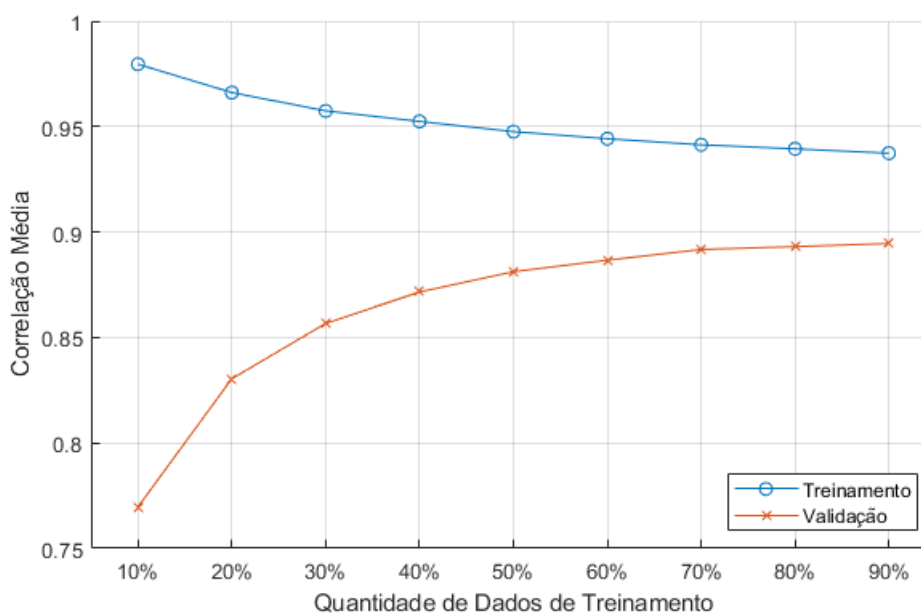
As curvas para as redes MLP apresentam uma tendência muito semelhante ao comportamento apresentado pela técnica MMQ, com uma queda do desempenho de treinamento para um conjunto maior de dados e um aumento no desempenho do conjunto de validação. Vale notar

que para os desempenhos de validação, os valores máximos são alcançados entre 70% e 80% do banco de dados utilizado para o treinamento (30% e 20% para validação, respectivamente).

III. Redes RBF - Arquitetura: Rede Neural de Regressão Generalizada (RNRG)

A análise do desempenho das redes RNRG em relação ao número de dados utilizados para o treinamento, mostrou uma tendência de aumento no desempenho dos grupos de validação e um decréscimo no desempenho de treinamento, assim como as demais técnicas até agora apresentadas, como pode ser visto na Figura 4.11

Figura 4.11 – Curvas de treinamento e validação para os modelos RNRG.

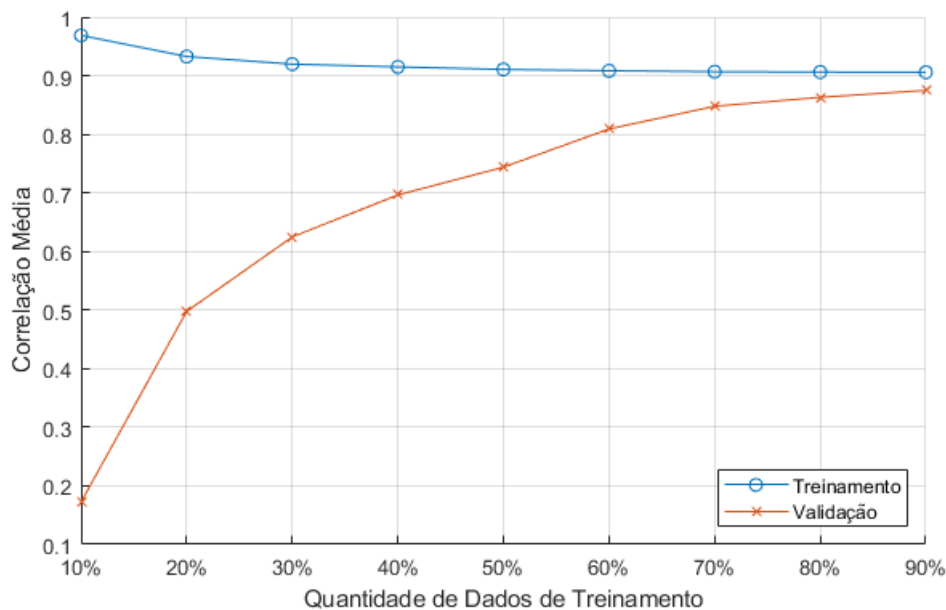


Fonte: Autor

III. Sistemas de Inferência Neuro-Fuzzy Adaptativos (ANFIS)

Os modelos ANFIS confirmaram os comportamentos anteriores, com decréscimo do desempenho de treinamento e aumento do desempenho de validação a medida que se utiliza mais dados para treinamento dos modelos. O desempenho específico desta técnica pode ser visto na Figura 4.12.

Figura 4.12 – Curvas de treinamento e validação para os modelos ANFIS.



Fonte: Autor

IV. Considerações Finais

As técnicas MMQ e MLP apresentaram os melhores desempenhos do conjunto de validação quando se usa em torno de 70%-80% dos dados para a elaboração dos modelos. As técnicas RNRG e ANFIS apresentaram uma tendência constante de aumento no desempenho de validação a medida que mais dados eram usados no treinamento dos algoritmos, com um acréscimo menos expressivo a partir da utilização de 60% dos dados.

Dado o fato de que a maior parte dos testes apresentarem um desempenho médio do conjunto de validação superior para um número de dados de treinamento igual a 70% do total de dados (Figs. 4.1-4.7), essa quantidade foi escolhida como padrão para as demais comparações.

4.2 Regressão Linear Múltipla - Método dos Mínimos Quadrados (MMQ)

Como apresentado na subseção 3.3.2, utilizou-se o MMQ como representante da abordagem estatística tradicional. Foi considerado um modelo linear para descrever a relação entre as variáveis de estudo e o CEM, descrito pela Equação (3.22).

O desempenho médio, em termos da correlação linear, encontrado pelas 100 regressões realizadas foi de $\rho = 0,8866$, com o melhor modelo obtendo um desempenho de $\rho = 0,9182$ e o pior modelo um desempenho de $\rho = 0,8521$. A seguir são apresentados os desempenhos para todas as métricas consideradas, os resultados são referentes à média \pm desvio padrão.

- Índice de Correlação Linear de Pearson: $\rho = 0,8866 \pm 0,0125$;
- Coeficiente de Determinação: $r^2 = 0,7862 \pm 0,0222$;
- Acurácia: $A_p = 0,9867 \pm 0,0077$;
- Coeficiente de Correlação de Concordância: $\hat{\rho}_c = 0,8748 \pm 0,0144$;
- Erro Médio Quadrático de Predição: $EMQP = 0,0280 \pm 0,0031$;
 - Viés Médio = $0,0001 \pm 0,0002$;
 - Viés Sistemático = $0,0001 \pm 0,0002$;
 - Erros Aleatórios = $0,0277 \pm 0,0031$.

Os parâmetros ajustados para o melhor modelo foram:

$$CEM = 0,1215 + 0,0333 \cdot G + 0,0013 \cdot S + 0,0669 \cdot R + 0,3582 \cdot PCVZm + 0,6725 \cdot GPCVZ \quad (4.1)$$

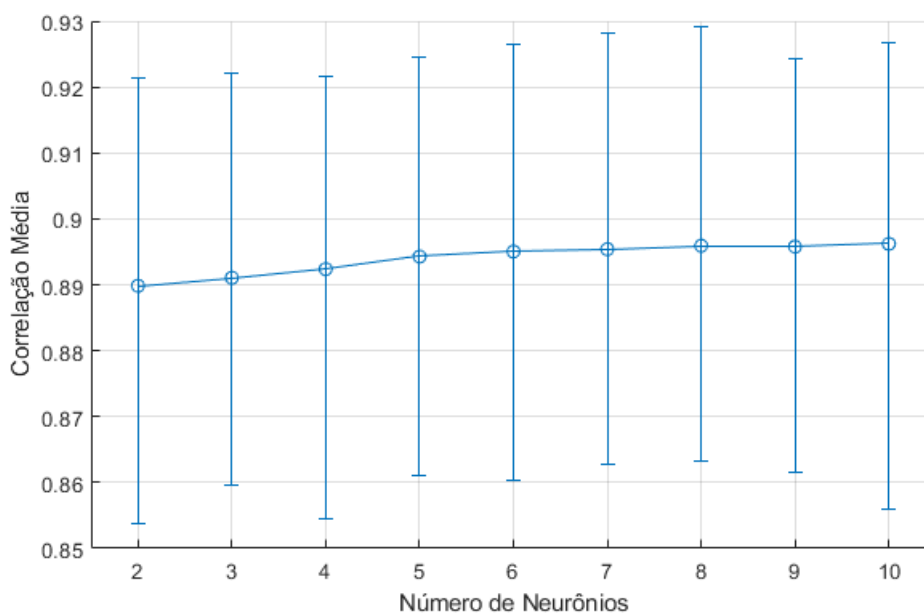
onde G representa o gênero do animal, S é o sistema de alimentação, R é o grupo genético, $PCVZm$ é o Peso Corporal Vazio Médio e $GPCVZ$ é o Ganho de Peso de Corpo Vazio.

Vale lembrar que a Equação 4.1 não pode ser usada diretamente apenas substituindo-se os respectivos valores de entrada, sendo necessário normalizá-los com a Equação 3.2 e “des-normalizar” a saída com a esta mesma equação, mas invertendo-se os limites.

4.3 Redes Perceptron Multicamadas (MLP) - Treinamento Gradiente Conjugado Escalonado (SCG)

O Algoritmo SCG foi implementado no *software* MatLab utilizando as configurações descritas na subseção 3.4.1. A Figura 4.13 apresenta os resultados de Validação para os modelos MLP treinados com apenas uma camada escondida. Os resultados são referentes à correlação média do conjunto de validação das 100 execuções de cada arquitetura e os limites inferiores e superiores são referentes ao pior e melhor desempenho, respectivamente.

Figura 4.13 – Desempenho dos modelos MLP-SCG para um conjunto de dados de Validação igual a 30% e uma camada escondida.

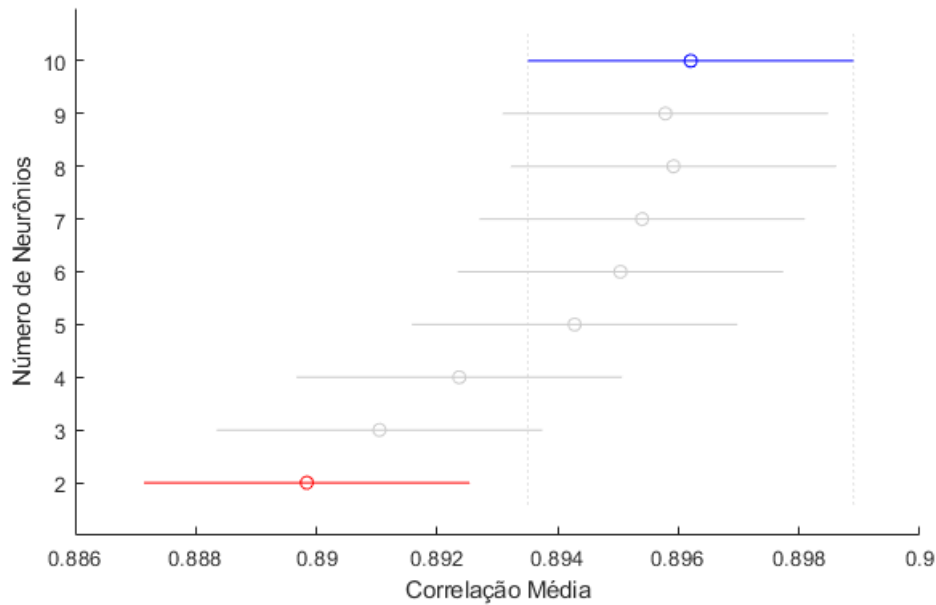


Fonte: Autor

Todos os modelos alcançaram desempenho médio, em relação ao índice de correlação linear, superiores à 0,889, com todos os piores desempenhos de cada arquitetura alcançando desempenhos superiores à 0,85. Os redes MLP com apenas dois neurônios na camada escondida tiveram o pior desempenho médio e absoluto, com correlações $\rho = 0,8898$ e $\rho = 0,8536$, respectivamente. O melhor desempenho médio foi alcançado pelos modelos com 10 neurônios na camada escondida com $\rho = 0,8962$, enquanto o melhor desempenho absoluto foi para um modelo com oito neurônios e desempenho $\rho = 0,9292$.

As arquiteturas tiveram seus desempenhos comparados pelo Teste de Tukey da Diferença Honestamente Significativa, utilizando-se um nível de confiança de 95%. Os resultados dessa comparação são apresentados na Figura 4.14.

Figura 4.14 – Teste de Comparação Múltipla de Médias para a arquitetura MLP-SCG com 1 camada escondida e número de neurônios variando de [2, 10].



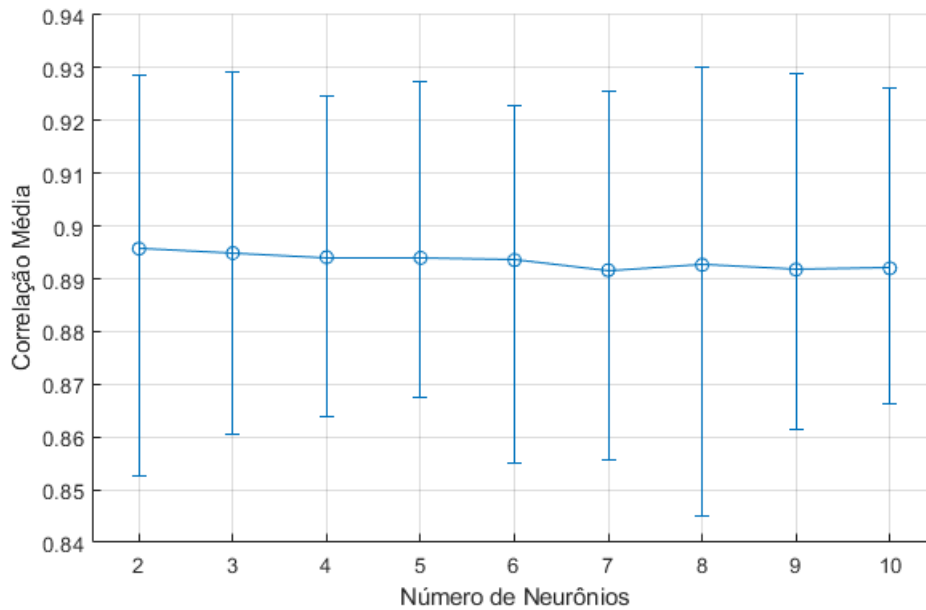
Fonte: Autor

A figura dá destaque ao modelo com dez neurônios (10N), que possui a maior média. O grupo em vermelho, dois neurônios (2N), são os modelos que apresentaram diferença significativa da arquitetura 10N, enquanto os grupos em cinza não apresentaram diferença estatística em relação à arquitetura de melhor desempenho (10N).

Como praticamente todos os modelos não apresentaram diferença estatística que justifique a escolha de algum em detrimento de outro para os testes realizados, a arquitetura escolhida é a com 10 neurônios na camada escondida, por possuir o maior desempenho absoluto.

Antes da determinação da arquitetura final, um teste foi realizado com as arquiteturas de 10 neurônios, acrescentando-se uma segunda camada escondida com um número de neurônios variando-se entre 2-10 neurônios. A Figura 4.15 mostra o desempenho com os modelos de duas camadas escondidas.

Figura 4.15 – Desempenho dos modelos MLP-SCG para um conjunto de dados de Validação igual a 30% e duas camadas escondidas.

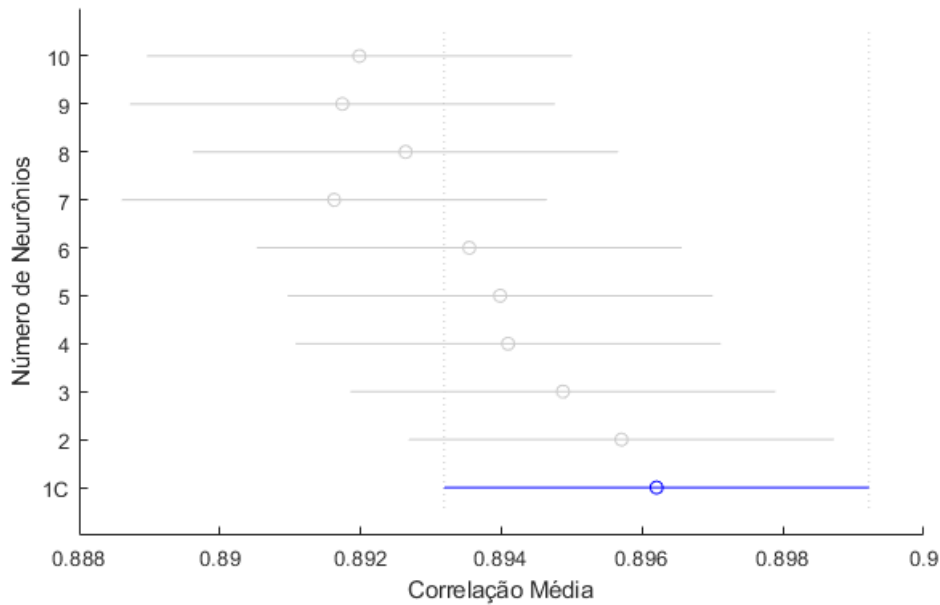


Fonte: Autor

O melhor desempenho médio para os modelos com duas camadas escondidas é para a arquitetura que contém dois neurônios na segunda camada, com $\rho = 0,8957$, levemente inferior à sua contraparte de apenas uma camada escondida com dez neurônios. O resultado do Teste de Tukey, com significância de 0,05, confirma que de fato o acréscimo de uma segunda camada escondida não agrega desempenho aos modelos, como pode ser observado na Figura 4.16.

Na Figura 4.16, o item '1C' corresponde ao modelo com apenas uma camada escondida, enquanto os demais valores (2-10) dizem respeito ao número de neurônios na segunda camada escondida, mantendo-se a primeira fixa com 10 neurônios.

Figura 4.16 – Teste de Comparação Múltipla de Médias para a arquitetura MLP-SCG com 2 camadas escondidas.



Fonte: Autor

Dados os resultados encontrados, a arquitetura escolhida como a melhor dentre as testadas para a técnica MLP-SCG é a com apenas uma camada escondida e 10 neurônios. A seguir são apresentados os desempenhos para todas as métricas consideradas, os resultados são referentes à média \pm desvio padrão das 100 execuções.

- Índice de Correlação Linear de Pearson: $\rho = 0,8962 \pm 0,0128$;
- Coeficiente de Determinação: $r^2 = 0,8033 \pm 0,0229$;
- Acurácia: $A_p = 0,9930 \pm 0,0045$;
- Coeficiente de Correlação de Concordância: $\hat{\rho}_c = 0,8899 \pm 0,0135$;
- Erro Médio Quadrático de Predição: $EMQP = 0,0258 \pm 0,0032$;
 - Viés Médio = $0,0001 \pm 0,0002$;
 - Viés Sistemático = $0,0002 \pm 0,0003$;
 - Erros Aleatórios = $0,0255 \pm 0,0032$.

Os modelos implementados por esse tipo de arquitetura implementam função de estimação descrita pela Equação 4.2:

$$y(x_1, x_2, x_3, x_4, x_5) = \sum_{j=1}^{10} w_j^2 \left(\frac{1}{1 - e^{-2(\sum_{i=1}^5 x_i \cdot w_{i,j}^1)}} - 1 \right) + b^2 \quad (4.2)$$

Onde x_i é o i -ésimo parâmetro de entrada, $w_{i,j}^1$ é o peso da primeira camada escondida correspondente ao i -ésimo parâmetro de entrada e ao j -ésimo neurônio, b_j^1 é o viés da primeira camada correspondente ao j -ésimo neurônio, w_j^2 é o peso da camada de saída correspondente à j -ésima saída da camada escondida e b^2 é o viés da camada de saída.

A Equação 4.3, a seguir, é o modelo que obteve o melhor desempenho dentre os 100 modelos com dez neurônios, com correlação $\rho = 0,9268$.

$$\begin{aligned} CEM = & 0,7517 \left(\frac{1}{1 - e^{-2(-1,9588G - 0,4271S - 0,3080R + 1,5158PCVZm + 1,4772GPCVZ + 1,7269)}} - 1 \right) - \\ & 0,8545 \left(\frac{1}{1 - e^{-2(-1,7912G + 1,1476S - 1,2010R + 0,1045PCVZm - 0,7723GPCVZ + 1,4126)}} - 1 \right) - \\ & 1,1678 \left(\frac{1}{1 - e^{-2(-0,4096G + 0,1787S + 0,2903R - 0,7760PCVZm - 0,6339GPCVZ - 0,8225)}} - 1 \right) + \\ & 0,3191 \left(\frac{1}{1 - e^{-2(0,6751G + 0,7463S - 1,4169R - 2,0133PCVZm - 1,1153GPCVZ - 1,0994)}} - 1 \right) - \\ & 0,3484 \left(\frac{1}{1 - e^{-2(-2,5866G + 0,1163S + 1,0320R - 3,3333PCVZm + 0,9284GPCVZ - 0,3197)}} - 1 \right) - \\ & 1,1212 \left(\frac{1}{1 - e^{-2(2,0911G - 1,3537S + 1,2070R + 0,3527PCVZm + 0,6823GPCVZ - 0,6668)}} - 1 \right) + \\ & 0,3897 \left(\frac{1}{1 - e^{-2(-1,1326G - 0,2220S + 0,6877R - 0,7397PCVZm + 2,0056GPCVZ - 0,2145)}} - 1 \right) - \\ & 0,4382 \left(\frac{1}{1 - e^{-2(-1,4901G - 0,7965S - 1,3996R + 1,4499PCVZm + 0,5144GPCVZ - 1,5958)}} - 1 \right) - \\ & 0,2893 \left(\frac{1}{1 - e^{-2(-1,8515G + 0,4774S - 1,0384R - 0,1781PCVZm + 0,9342GPCVZ - 1,1644)}} - 1 \right) - \\ & 0,0542 \left(\frac{1}{1 - e^{-2(-1,8010G + 0,2566S + 1,1904R - 0,2394PCVZm - 0,7755GPCVZ - 2,2200)}} - 1 \right) - \\ & 0,8801; \end{aligned} \quad (4.3)$$

onde G representa o gênero do animal, S é o sistema de alimentação, R é o grupo genético, $PCVZm$ é o Peso Corporal Vazio Médio e $GPCVZ$ é o Ganho de Peso de Corpo Vazio.

A Equação 4.3, assim como a Equação 4.1, deve ser usada após a normalização das entradas. E o resultado final deve ser desnormalizado.

4.4 Rede Neural de Regressão Generalizada (RNRG)

A arquitetura utilizada para o treinamento das Redes Funções de Base Radial (RBF) foi a Rede Neural de Regressão Generalizada (RNRG). Essa abordagem possui uma estrutura fixa, dependente apenas dos pares de entrada/saída utilizados na sua construção, não sendo necessário uma retro-propagação dos erros, o que faz com que seu desenvolvimento seja muito rápido. Foram desenvolvidas 100 redes distintas, construídas a partir de 100 divisões aleatórias do banco de dados.

A correlação média entre as estimativas da rede e os dados de CEM utilizados na validação das 100 execuções foi de $\rho = 0,8917 \pm 0,0134$, com a melhor rede obtendo um desempenho de $\rho = 0,9299$, em relação ao conjunto de validação, e a pior rede com um desempenho de $\rho = 0,8559$. Pode-se observar que o desvio padrão dessa abordagem é pequeno, indicando um desempenho semelhante, independente do subconjunto de “treino”. Esse fato deve-se principalmente no modo de determinação dos pesos dessa rede, onde desde que se apresente um conjunto de dados representativo e escolha-se um fator de espalhamento adequado, pontos semelhantes no espaço de parâmetros geram saídas semelhantes. Os demais resultados são:

- Índice de Correlação Linear de Pearson: $\rho = 0,8917 \pm 0,0134$;
- Coeficiente de Determinação: $r^2 = 0,7954 \pm 0,0239$;
- Acurácia: $A_p = 0,9873 \pm 0,0071$;
- Coeficiente de Correlação de Concordância: $\hat{\rho}_c = 0,8804 \pm 0,0157$;
- Erro Médio Quadrático de Predição: $EMQP = 0,0268 \pm 0,0033$;
 - Viés Médio = $0,0002 \pm 0,0002$;
 - Viés Sistemático = $0,0001 \pm 0,0001$;
 - Erros Aleatórios = $0,0265 \pm 0,0033$.

Como foram utilizados 70% dos dados, a primeira camada dessa rede tem 588 elementos e um neurônio de saída. Devido à complexidade de se apresentar a equação implementada por esses modelos e a quantidade de parâmetros envolvida, um total de 4116, não serão apresentados os pesos desta rede.

4.5 Sistemas de Inferência Neuro-Fuzzy Adaptativos (ANFIS)

Foram elaborados 100 modelos utilizando-se a técnica Sistemas de Inferência Neuro-Fuzzy Adaptativos (ANFIS). Como se utilizou na análise de dados uma abordagem híbrida entre as técnicas Agrupamento Subtrativo (AS) e Agrupamento *Fuzzy C-Means* (FCM), a arquitetura ficou condicionada ao conjunto de dados de treinamento utilizado em cada modelo. A Tabela 4.1 apresenta como ficaram distribuídos os 100 modelos em termos de número de agrupamentos e o desempenho médio de cada um deles.

Tabela 4.1 – Correlação Médio das Estimativas - Consumo de Energia Metabolizável - Modelos ANFIS em relação ao número de Agrupamentos.

Número de Agrupamentos	Frequência	Desempenho Médio
7 Agrupamentos	5	0,7448 ± 0,2241
8 Agrupamentos	23	0,8657 ± 0,0864
9 Agrupamentos	31	0,8349 ± 0,1575
10 Agrupamentos	23	0,8469 ± 0,1452
11 Agrupamentos	8	0,8652 ± 0,0691
12 Agrupamentos	2	0,8887 ± 0,0112
13 Agrupamentos	3	0,8867 ± 0,0156
14 Agrupamentos	4	0,8898 ± 0,0206
15 Agrupamentos	1	0,8666 ± 0,0000
Comportamento Médio	100	0,8478 ± 0,1307

Fonte: Autor

Pode-se observar que o comportamento predominante do treinamento foi ser dividido em nove grupos, fato que ocorreu em 31% dos testes, seguido da divisão em oito e dez grupos, com 23% de ocorrência cada. O agrupamento menos comum foi o de 15 grupos, com apenas 1 ocorrência. De fato, se realizarmos o agrupamento seguindo a mesma metodologia, porém com o todos os dados (treinamento + teste), o banco se organiza em nove agrupamentos, fato testado em outras 100 execuções do algoritmo, sugerindo que esse é o comportamento padrão. Os demais desempenhos para as métricas consideradas são apresentados a seguir. Os resultados são referentes à média ± desvio padrão dos 31 agrupamentos.

- Índice de Correlação Linear de Pearson: $\rho = 0,8349 \pm 0,1575$;
- Coeficiente de Determinação: $r^2 = 0,7210 \pm 0,1905$;
- Acurácia: $A_p = 0,9718 \pm 0,0809$;
- Coeficiente de Correlação de Concordância: $\hat{\rho}_c = 0,8224 \pm 0,1736$;

- Erro Médio Quadrático de Predição: $EMQP = 0,0806 \pm 0,2189$;
- Viés Médio = $0,0008 \pm 0,0031$;
- Viés Sistemático = $0,0432 \pm 0,1953$;
- Erros Aleatórios = $0,0366 \pm 0,0246$.

Os modelos com 9 agrupamentos, como discutido na Subseção 3.4.3, geram arquiteturas ANFIS que possuem 9 FP para cada variável de entrada e saída, e 9 regras. O melhor modelo ANFIS treinado obteve desempenho $\rho = 0,9159$; os parâmetros ajustados para este modelo são apresentados nas Tabelas 4.2 e 4.3.

Tabela 4.2 – Parâmetros ajustados para as FP de saída (Eq. 3.25) do melhor modelo ANFIS

FP	Parâmetros Ajustados
1	$p_1 = -0,0152$; $p_2 = -0,0170$; $p_3 = -0,0148$; $p_4 = +0,4729$; $p_5 = +0,4175$; $r = +0,0152$
2	$p_1 = +0,6179$; $p_2 = -1,2982$; $p_3 = +0,1700$; $p_4 = +0,3263$; $p_5 = +0,3806$; $r = -0,6179$
3	$p_1 = -0,1225$; $p_2 = -0,0764$; $p_3 = +0,0559$; $p_4 = +0,1009$; $p_5 = +1,0145$; $r = +0,1225$
4	$p_1 = -0,0153$; $p_2 = -0,0620$; $p_3 = -0,0151$; $p_4 = -0,0010$; $p_5 = +1,1336$; $r = +0,0153$
5	$p_1 = -3,3811$; $p_2 = +5,9990$; $p_3 = -0,5637$; $p_4 = +0,6785$; $p_5 = +0,3974$; $r = +3,3811$
6	$p_1 = -0,0329$; $p_2 = -17,4501$; $p_3 = +0,2219$; $p_4 = +0,3863$; $p_5 = +0,9476$; $r = -16,9837$
7	$p_1 = +41,0263$; $p_2 = +0,1639$; $p_3 = +0,4358$; $p_4 = -0,2647$; $p_5 = +0,4390$; $r = -40,3824$
8	$p_1 = -0,0146$; $p_2 = -0,1579$; $p_3 = +0,1221$; $p_4 = +0,7182$; $p_5 = +0,5202$; $r = +0,0146$
9	$p_1 = -2,1029$; $p_2 = +4,1489$; $p_3 = -4,6776$; $p_4 = +0,3508$; $p_5 = +0,7190$; $r = +2,1029$

Fonte: Autor

Tabela 4.3 – Parâmetros ajustados para as FP de entrada (Eq. 3.24) do melhor modelo ANFIS

Variável Associada	FP	Parâmetros Ajustados
x_1 :Alimentação	1	$\sigma = 0,1484; c = -0,9963$
	2	$\sigma = 0,1479; c = -0,9929$
	3	$\sigma = 0,1388; c = -0,9967$
	4	$\sigma = 0,1515; c = -0,9952$
	5	$\sigma = 0,1410; c = -0,9955$
	6	$\sigma = 0,5422; c = +0,9890$
	7	$\sigma = 0,5508; c = +0,9735$
	8	$\sigma = 0,1438; c = -0,9979$
	9	$\sigma = 0,1504; c = -0,9891$
x_2 :Genética	1	$\sigma = 0,2187; c = -0,9527$
	2	$\sigma = 0,2109; c = -0,9558$
	3	$\sigma = 0,5025; c = +0,9282$
	4	$\sigma = 0,2076; c = -0,9766$
	5	$\sigma = 0,1963; c = -0,9616$
	6	$\sigma = 0,1984; c = -0,9865$
	7	$\sigma = 0,2484; c = -0,4518$
	8	$\sigma = 0,4208; c = +0,4259$
	9	$\sigma = 0,2120; c = -0,9381$
x_3 :Sexo	1	$\sigma = 0,1528; c = -0,9907$
	2	$\sigma = 0,3208; c = -0,0232$
	3	$\sigma = 0,3131; c = -0,0487$
	4	$\sigma = 0,1547; c = -0,9889$
	5	$\sigma = 0,5601; c = +0,9709$
	6	$\sigma = 0,1425; c = -0,9881$
	7	$\sigma = 0,1439; c = -0,9764$
	8	$\sigma = 0,1562; c = -0,9798$
	9	$\sigma = 0,3186; c = -0,0361$
x_4 :PCVZm	1	$\sigma = 0,1543; c = +0,2331$
	2	$\sigma = 0,1301; c = +0,1096$
	3	$\sigma = 0,1249; c = +0,0595$
	4	$\sigma = 0,1424; c = -0,1790$
	5	$\sigma = 0,1824; c = -0,4441$
	6	$\sigma = 0,1999; c = -0,4496$
	7	$\sigma = 0,0956; c = +0,0289$
	8	$\sigma = 0,0951; c = +0,2743$
	9	$\sigma = 0,1258; c = -0,0797$
x_5 :GPCVZ	1	$\sigma = 0,1677; c = -0,1213$
	2	$\sigma = 0,1295; c = -0,2307$
	3	$\sigma = 0,1539; c = -0,0401$
	4	$\sigma = 0,1593; c = -0,4480$
	5	$\sigma = 0,1213; c = -0,5721$
	6	$\sigma = 0,1675; c = -0,6892$
	7	$\sigma = 0,1283; c = -0,6232$
	8	$\sigma = 0,1618; c = -0,2164$
	9	$\sigma = 0,1454; c = -0,6222$

Fonte: Autor

Tal qual nas abordagens anteriores, não basta somente aplicar os parâmetros acima nas FP correspondentes, sendo necessário a normalização das variáveis de entrada e desnormalização do resultado. As forças de ativação de cada regra (w_i), devem ser calculadas como descrito no Apêndice C.

4.6 Comparações de Desempenho

Esta subseção apresentará os resultados das comparações entre todas as metodologias investigadas. Serão comparados os desempenhos em relação às estatísticas apresentadas no Item 3.5, o tempo de treinamento/elaboração dos modelos e número de operações necessárias para aplicação do modelo.

Os resultados referentes ao conjunto de equações disponível pelo Sistema BR-CORTE provém da sua aplicação aos 100 subconjuntos de treino e validação utilizados por todas as demais técnicas.

I. Desempenho Estatístico

As Tabelas 4.4 e 4.5 mostram um comparativo dos modelos em relação às métricas consideradas. Além dos valores absolutos das decomposições do EMQP, foram acrescentadas na Tabela 4.5 as contribuições percentuais de cada componente, uma vez que essa forma de apresentação facilita a identificação das fontes de erro.

Tabela 4.4 – Desempenho comparativo médio dos métodos de estimação do CEM para 30% de dados de Validação e cinco parâmetros de entrada em relação à Correlação Linear (ρ), Coeficiente de Determinação (r^2), Acurácia (A_p) e Coeficiente de Correlação de Concordância ($\hat{\rho}_c$).

Técnica	Estatística de Comparação			
	ρ -Eq. 3.26	r^2 -Eq. 3.31	A_p -Eq. 3.32	$\hat{\rho}_c$ -Eq. 3.34)
MLP-SCG	0,8962 ± 0,0128	0,8033 ± 0,0229	0,9930 ± 0,0045	0,8899 ± 0,0135
RNRG	0,8917 ± 0,0134	0,7954 ± 0,0239	0,9873 ± 0,0071	0,8804 ± 0,0157
ANFIS	0,8349 ± 0,1575	0,7210 ± 0,1905	0,9718 ± 0,0809	0,8224 ± 0,1736
BR-CORTE	0,8745 ± 0,0125	0,7649 ± 0,0219	0,9922 ± 0,0043	0,8677 ± 0,0123
MMQ	0,8866 ± 0,0125	0,7862 ± 0,0222	0,9867 ± 0,0077	0,8748 ± 0,0144

Fonte: Autor

Tabela 4.5 – Desempenho comparativo médio dos métodos de estimação do CEM para 30% de dados de Validação e cinco parâmetros de entrada em relação à Raiz do Erro Médio Quadrático de Predição e suas decomposições.

Estatística de Comparação				
Técnica	<i>EQMP</i> (Eq. 3.38)	Viés Médio (Eq. 3.40)	Viés Sistemático (Eq. 3.40)	Erros Aleatórios (Eq. 3.40)
MLP-SCG	$0,0258 \pm 0,0032$	$0,0001 \pm 0,0002$ (0,39%)	$0,0002 \pm 0,0003$ (0,76%)	$0,0255 \pm 0,0032$ (98,84%)
RNRG	$0,0268 \pm 0,0033$	$0,0002 \pm 0,0002$ (0,75%)	$0,0001 \pm 0,0001$ (0,37%)	$0,0265 \pm 0,0033$ (98,88%)
ANFIS	$0,0806 \pm 0,2189$	$0,0008 \pm 0,0031$ (0,99%)	$0,0432 \pm 0,1953$ (53,60%)	$0,0366 \pm 0,0246$ (45,41%)
BR-CORTE	$10,4506 \pm 1,0561$	$0,3149 \pm 0,1664$ (3,01%)	$1,3795 \pm 0,4517$ (13,20%)	$8,7562 \pm 0,9108$ (83,79%)
MMQ	$0,0280 \pm 0,0031$	$0,0001 \pm 0,0002$ (0,36%)	$0,0001 \pm 0,0002$ (0,36%)	$0,0277 \pm 0,0031$ (98,93%)

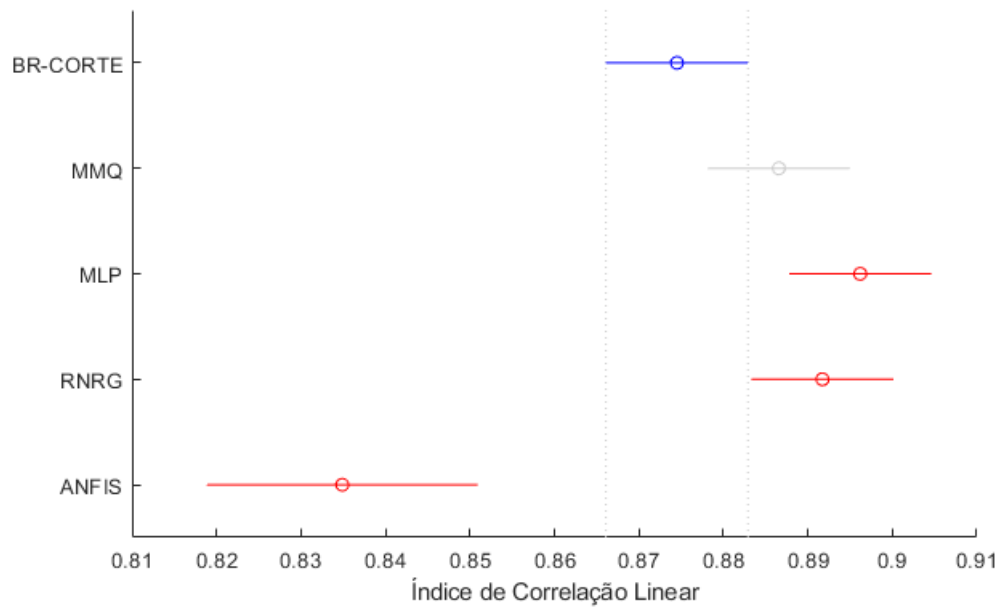
Fonte: Autor

Das oito métricas utilizadas, as técnicas de Inteligência Computacional (IC) superaram o desempenho da metodologia clássica em cinco, ficando atrás somente nas decomposições do EMQP em termos de contribuição percentual ao erro total. A técnica MMQ obteve as menores contribuições de Viés Médio e Sistemático e a maior porcentagem de Erros Aleatórios, porém, seu desempenho absoluto para o EMQP foi inferior às técnicas MLP e RNRG.

A técnica MLP teve os melhores desempenhos médios nas estatísticas de Correlação Linear, Coeficiente de Determinação, Acurácia, Coeficiente de Correlação de Concordância e Erro Médio Quadrático de Predição. A técnica ANFIS, entretanto, ficou com os piores desempenhos para Correlação Linear, Coeficiente de Determinação, Acurácia e Coeficiente de Correlação de Concordância. O conjunto de equações BR-CORTE, teve o pior desempenho em termos do EMQP.

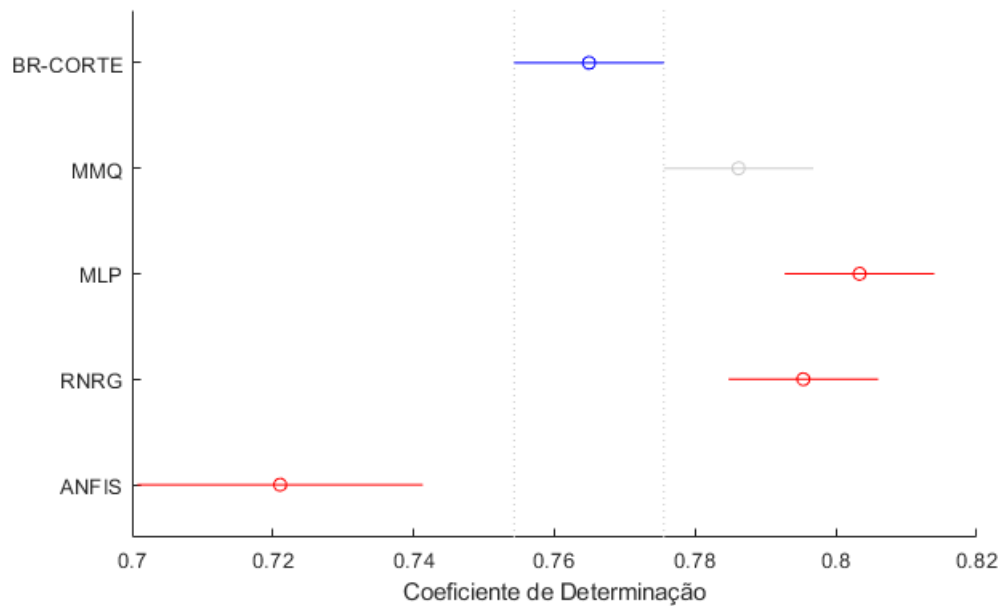
Nas Figuras 4.17-4.21, a seguir, são apresentados os resultados do Teste de Tukey para os valores de Correlação Linear, Índice de Determinação, Acurácia, Coeficiente de Correlação de Concordância e Erro Médio Quadrático de Predição. Nas figuras, o valor em azul indica a média sendo testada, os valores vermelhos são aqueles que apresentam diferença estatística e os valores em cinza são aqueles que não apresentam diferença. Em todas as figuras a média sendo testada é a dos resultados referentes à técnica BR-CORTE.

Figura 4.17 – Teste de Comparação Múltipla de Médias para os resultados de Correlação Linear.



Fonte: Autor

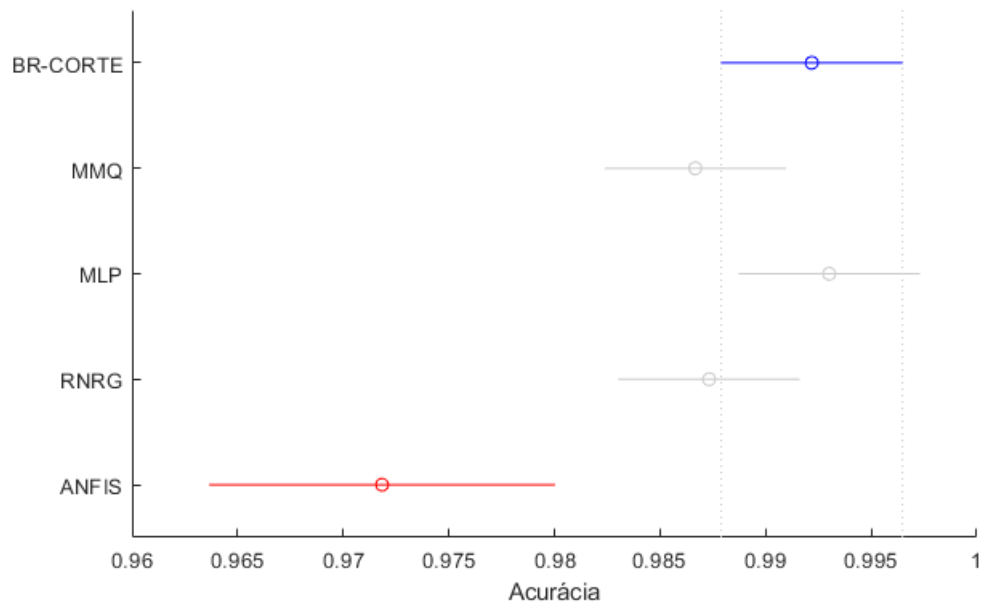
Figura 4.18 – Teste de Comparação Múltipla de Médias para os resultados do Índice de Determinação.



Fonte: Autor

Para os resultados de Correlação Linear e Coeficiente de Determinação, os resultados MLP e RNRG apresentaram ganho sobre o método clássico de estimação do CEM, enquanto os modelos lineares calculado pelo MMQ não apresentou diferença e a técnica ANFIS obteve um desempenho significativamente menor.

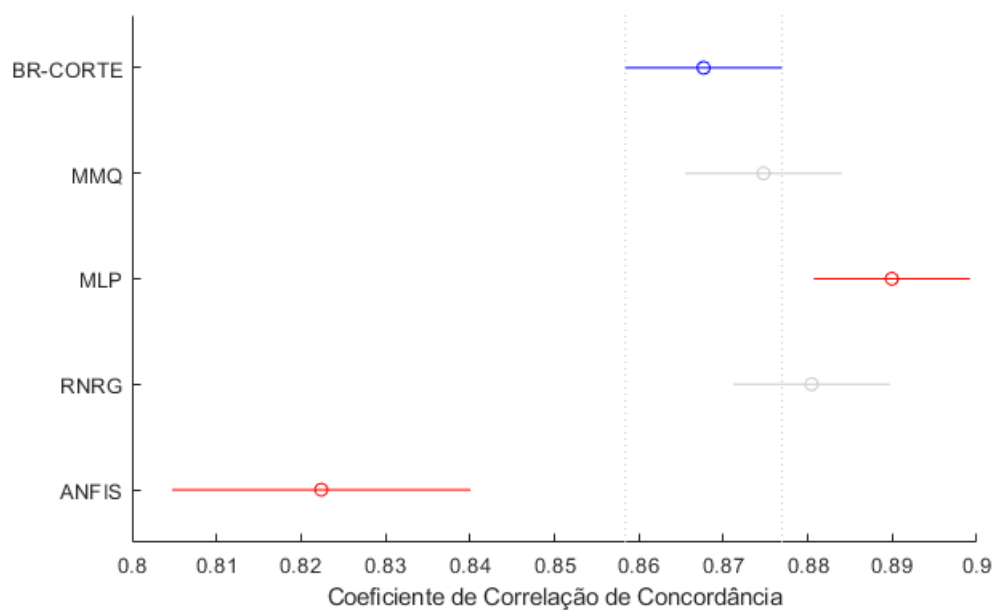
Figura 4.19 – Teste de Comparação Múltipla de Médias para os resultados de Acurácia.



Fonte: Autor

Para a métrica de Acurácia, apenas a técnica ANFIS apresentou diferença significativa em relação à metodologia BR-CORTE, com um desempenho inferior. Os demais modelos não apresentaram diferença estatística.

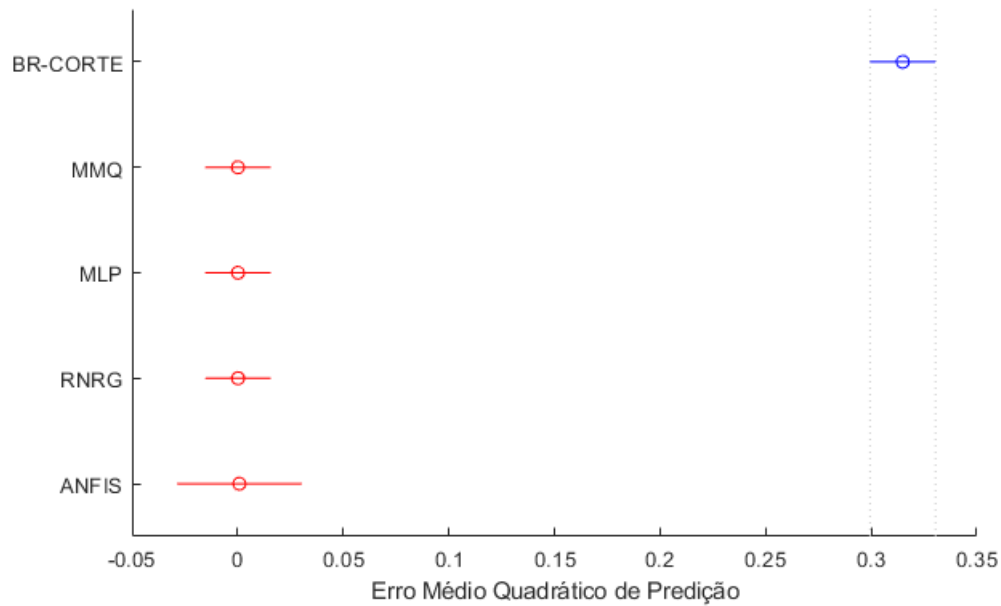
Figura 4.20 – Teste de Comparação Múltipla de Médias para os resultados de Coeficiente de Correlação de Concordância.



Fonte: Autor

Para os valores do Coeficiente de Correlação de Concordância, os modelos MLP apresentaram um ganho em relação à abordagem tradicional, enquanto a técnica ANFIS teve um desempenho inferior. Os modelos RNRG e MMQ não apresentaram ganho significativo.

Figura 4.21 – Teste de Comparação Múltipla de Médias para os resultados do Erro Médio Quadrático de Predição.



Fonte: Autor

Por fim, para o EMQP, todas as técnicas abordadas apresentaram ganho em relação ao Sistema BR-CORTE. Entre si, os modelos de IC e a técnica MMQ não apresentaram diferença estatística.

II. Tempo de Treinamento/Elaboração dos Modelos

Um fator importante que deve ser levado em consideração ao escolher-se qual método aplicar é o tempo necessário para a construção dos modelos. A Tabela 4.6 apresenta o tempo médio para o ajuste dos parâmetros de um modelo (chamada da função de treinamento) para cada técnica de IC considerada. As configurações do computador utilizado foram:

- Sistema Operacional: Windows 10 Home Single Language (64 bits);
- Processador : Intel^(R) Core^(TM) i5-4210U, CPU 2 núcleos de 1.70GHz cada (64 bits);
- Memória RAM : 8 GB;
- Memória Cache : 1280 MB.

Tabela 4.6 – Tempo médio, em segundos, de ajuste dos parâmetros para cada método de estimação do CEM, com 70% de dados de Treino e cinco parâmetros de entrada.

Técnica	Tempo [s]
MLP-SCG	$0,8162 \pm 0,1015$
RNRG	$0,0406 \pm 0,0204$
ANFIS	$0,7754 \pm 0,1130$
MMQ	$0,0001 \pm 0,0003$

Fonte: Autor

A técnica com menor tempo para treinamento foi o MMQ, levando aproximadamente 0,0001 segundos para o cálculo dos parâmetros. Este tempo deve-se ao fato desse método ser exato, dependente apenas do cálculo de uma matriz pseudo-inversa (Eq. (3.23)).

O método com maior tempo para seu ajuste foram as redes MLP. O algoritmo de treinamento, ainda que otimizado para fazer menos avaliações da função custo, ainda requer várias chamadas desta função, o cálculo de gradientes e a retro-propagação dos erros para a atualização dos pesos.

O método RNRG teve o segundo melhor desempenho, com um tempo médio em torno de 0,04 segundos para o ajuste de seus parâmetros. Este tempo deve-se ao modo como essas redes são construídas, como os pesos são atribuídos de acordo com os pares entrada/saída usados no design da rede, não há avaliações de uma função de desempenho ou cálculo de gradientes.

Os modelos ANFIS tiveram o segundo pior desempenho, com um tempo médio para treinamento de 0,7754 segundos. Este tempo pode ser dividido em duas componentes, uma responsável pelo tempo necessário para criação do SIF inicial e uma responsável pelo ajuste dos parâmetros, já utilizando a arquitetura ANFIS. Para os modelos analisados neste trabalho, essas componentes foram iguais a 0,2382 e 0,5372 segundos, respectivamente.

III. Número de Operações para Aplicação dos Modelos Finais

Uma outra forma de se comparar as técnicas e inferir sobre a complexidade computacional do método é calculando-se o número de operações necessárias para se processar um exemplo.

As Tabelas 4.7 e 4.7 apresentam o número de operações realizadas em cada técnica para se processar um vetor de entrada com 5 parâmetros e utilizando as configurações descritas nas subseções anteriores. Elas apresentam as operações matemáticas mais básicas e as operações exclusivas de cada técnica, respectivamente.

Tabela 4.7 – Número de operações matemáticas básicas realizadas por cada modelo.

Técnica	Adições	Multiplicações	Divisões	Raízes
MLP-SCG	71	60	–	–
RNRG	8820	4116	1	588
ANFIS	85	49	9	–
MMQ	5	5	–	–
BR-CORTE	4	8	5	–

Fonte: Autor

Tabela 4.8 – Número de chamadas de funções e operações lógicas exclusivas de cada modelo.

Técnica	$tansig(\cdot)$	RBFs	FP Gaussianas	Comparações
MLP-SCG	10	–	–	–
RNRG	–	588	–	–
ANFIS	–	–	45	–
BR-CORTE	–	–	–	18

Fonte: Autor

O estimador que realiza o maior número de operações é o RNRG devido ao grande número de neurônios que fazem parte dessa arquitetura, sempre igual ao número de exemplos utilizados no *design*.

Os modelos BR-CORTE por não levarem em conta nas suas equações as informações de sistema de alimentação, grupo genético e sexo diretamente, fazem operações lógicas para determinar qual conjunto de fórmulas utilizar, no pior caso sendo necessárias 18 comparações.

Os modelos MMQ são os que realizam o menor número de operações uma vez que assumem o modelo mais simples para a estimação do CEM.

5 CONCLUSÕES

Esta dissertação de mestrado buscou investigar a viabilidade de utilização de técnicas de Inteligência Computacional (IC) como ferramentas para a construção de modelos capazes de estimar os requerimentos energéticos de gado bovino e compará-las com a técnica tradicional. Utilizou-se modelos baseados em Redes Neurais Artificiais (RNA) e Sistemas de Inferência *Fuzzy* (SIF) como alternativas aos modelos atuais baseados no uso de Regressões Múltiplas (RM), aqui representados pelo Sistema BR-CORTE, um compilado de informações e recomendações para a nutrição de bovinos, e pela Regressão Linear Múltipla calculada pelo Método dos Mínimos Quadrados (MMQ).

Os resultados foram obtidos a partir dos dados de 840 animais, com informações sobre a raça, sexo, grupo genético, Peso Corporal Vazio Médio (PCVZm) e Ganho de Peso de Corpo Vazio (GPCVZ), subdivididos em grupos de 70% dos dados para elaboração dos modelos e 30% para validação. As arquiteturas utilizadas foram as Redes Neurais Perceptron Multicamadas (MLP) treinadas pelo algoritmos Gradiente Conjugado Escalonado (SCG), Rede Neural de Regressão Generalizada (RNRG) e Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS), comparadas com os modelos sugeridos pelo Sistema BR-CORTE, um compilado de informações e recomendações para a nutrição de bovinos, e um modelo de Regressão Linear Múltipla calculado pelo Método dos Mínimos Quadrados (MMQ).

A comparação entre os modelos propostos e a abordagem clássica demonstrou a efetividade da utilização das ferramentas de IC no campo da estimação de requerimento energético em bovinos, com desempenhos superiores em cinco das oito métricas utilizadas.

O destaque fica com a metodologia Perceptron Multicamadas (MLP) que mostrou um desempenho estatístico superior nas métricas Índice de Correlação Linear, Coeficiente de Determinação e Coeficiente de Correlação de Concordância, além de possuir o maior valor médio de acurácia, apesar de não apresentar um ganho significativo pelos resultados do Teste de Tukey. Apesar desta técnica apresentar uma complexidade maior, com maior tempo de execução e número de cálculos, o ganho estatístico representa um ganho em relação à metodologia clássica. Assim os modelos de rede neural do tipo MLP treinados com o algoritmo SCG e 10 neurônios na camada escondida, podem ser utilizados como um complemento à metodologia atual.

Apesar dos modelos gerados pelas técnicas propostas nesse trabalho apresentarem dificuldade de interpretação do ponto de vista de adequação à conceitos biológicos, o fato de serem facilmente implementados e atualizados, fazem com que eles sejam candidatos perfeitos para

integrar sistemas de produção de gado de corte e *softwares* capazes de rapidamente apresentarem os requerimentos energéticos para bovinos.

Deste trabalho espera-se ainda a derivação de resultados adicionais que irão comparar o desempenho de modelos de RM Não-Lineares elaborados sobre o mesmo conjunto de dados que os modelos de IC, eliminando-se assim quaisquer dúvidas sobre a validade da metodologia proposta. Além disso, está em fase de elaboração um *software* que seja capaz de realizar as estimativas do CEM através das técnicas propostas neste trabalho.

REFERÊNCIAS

- BAZARAA, M. S.; SHETTY, C. M. **Nonlinear Programming**. [S.l.]: John Wiley & Sons, Inc., 1993.
- BCNRM. **Beef Cattle Nutrient Requirements Model - BCNRM. Nutrient Requirements of Beef Cattle, 8th Revised Edition**. [S.l.]: National Academies Press, 2016.
- BEHZADI, M. R. B.; ASLAMINEJAD, A. A. A comparison of neural network and nonlinear regression predictions of sheep growth. **Journal of Animal and Veterinary Advances**, v. 9, p. 2128–2131, 2010.
- BEZDEK, J. C. **Pattern Recognition with Fuzzy Objective Function Algorithms**. 1. ed. Springer US, 1981. (Advanced Applications in Pattern Recognition). ISBN 978-1-4757-0452-5, 978-1-4757-0450-1. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=98c46549b2e9025e9ad06d361f55d4c8>>.
- BHOSALE, M. D.; SINGH, T. P. Comparative study of feed-forward neuro-computing with multiple linear regression model for milk yield prediction in dairy cattle. **Current Science**, v. 108, n. 12, p. 2257, jun. 2015.
- BIBBY, J.; TOUTENBURG, H. Prediction and improved estimation in linear models. **Biometrical Journal**, Wiley, v. 20, n. 7-8, p. 826–826, 1977.
- BROOMHEAD, D. S.; LOWE, D. Multivariable functional interpolation and adaptive networks. **Complex Systems**, v. 2, p. 321–355, 1988.
- CEMEK, B.; RAHMAN, S.; RAHMAN, A. Artificial neural network for predicting nutrients concentration in runoff from beef cattle feedlot. **Environmental Engineering & Management Journal (EEMJ)**, v. 12, n. 12, 2013.
- CHAPARRO, M. A. E. et al. Estimation of pasture infectivity according to weather conditions through a fuzzy parametrized model for the free-living stage of ostertagia ostertagi. **Ecological Modelling**, v. 222, n. 11, p. 1820 – 1832, 2011. ISSN 0304-3800. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0304380011001426>>.
- CHIU, S. L. Fuzzy model identification based on cluster estimation. **Journal of Intelligent & fuzzy systems**, IOS Press, v. 2, n. 3, p. 267–278, 1994.
- CLAUSIUS, R. Ueber die bewegende kraft der wärme und die gesetze, welche sich daraus für die wärmelehre selbst ableiten lassen. **Annalen der Physik und Chemie**, Wiley, v. 155, n. 3, p. 368–397, 1850.
- DEB, R. et al. Designing of an artificial neural network model to evaluate the association of three combined y-specific microsatellite loci on the actual and predicted postthaw motility in crossbred bull semen. **Theriogenology**, v. 83, n. 9, p. 1445 – 1450, 2015. ISSN 0093-691X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0093691X15000291>>.
- DULIBA, K. A. Contrasting neural nets with regression in predicting performance in the transportation industry. In: **Proceedings of the Twenty-Fourth Annual Hawaii International Conference on System Sciences**. [S.l.]: IEEE Comput. Soc. Press, 1991.

DUTTA, S.; SHEKHAR, S.; WONG, W. Y. Decision support in non-conservative domains: generalization with neural networks. **Decision Support Systems**, Elsevier, v. 11, n. 5, p. 527–544, 1994.

EHRET, A. et al. Application of neural networks with back-propagation to genome-enabled prediction of complex traits in holstein-friesian and german fleckvieh cattle. **Genetics Selection Evolution**, v. 47, n. 1, p. 22, Mar 2015. ISSN 1297-9686. Disponível em: <<http://dx.doi.org/10.1186/s12711-015-0097-5>>.

FERNÁNDEZ, C. et al. Neural networks for animal science applications: Two case studies. **Expert Systems with Applications**, v. 31, n. 2, p. 444 – 450, 2006. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417405002800>>.

FERNÁNDEZ, C. et al. Weekly milk prediction on dairy goats using neural networks. **Neural Computing and Applications**, v. 16, n. 4-5, p. 373–381, maio 2007.

FERRELL, C. L.; JENKINS, T. G. Body composition and energy utilization by steers of diverse genotypes fed a high-concentrate diet during the finishing period: Ii. angus, boran, brahman, hereford, and tuli sires. **Journal of animal science**, v. 76, p. 647–657, fev. 1998. ISSN 0021-8812.

FERRELL, C. L.; OLTJEN, J. W. ASAS CENTENNIAL PAPER: Net energy systems for beef cattle—concepts, application, and future models. **Journal of Animal Science**, Oxford University Press (OUP), v. 86, n. 10, p. 2779–2794, oct 2008.

FINNIE, G. R.; WITTIG, G. E.; DESHARNAIS, J.-M. A comparison of software effort estimation techniques: Using function points with neural networks, case-based reasoning and regression models. **Journal of Systems and Software**, Elsevier BV, v. 39, n. 3, p. 281–289, dec 1997.

GARRETT, W. N.; MEYER, J. H.; LOFGREEN, G. P. The comparative energy requirements of sheep and cattle for maintenance and gain1. **Journal of Animal Science**, Oxford University Press (OUP), v. 18, n. 2, p. 528–547, may 1959.

GHAZANFARI, S.; NOBARI, K.; TAHMOORESPUR, M. Prediction of egg production using artificial neural network. **Iranian Journal of Applied Animal Science**, v. 1, n. 1, p. 11–16, 2011. ISSN 2251-628X. Disponível em: <http://ijas.iaurasht.ac.ir/article_513431.html>.

GIONBELLI, M. P. et al. Use of artificial neural networks to improve estimation of energy requirements of cattle. In: **Energy and protein metabolism and nutrition**. [S.l.: s.n.], 2016. p. 359.

GRZESIAK, W. et al. A comparison of neural network and multiple regression predictions for 305-day lactation yield using partial lactation records. **Canadian Journal of Animal Science**, v. 83, n. 2, p. 307–310, 2003. Disponível em: <<https://doi.org/10.4141/A02-002>>.

GU, Z.; LAM, L. H.; DHURJATI, P. S. Feature correlation method for enhancing fermentation development: A comparison of quadratic regression with artificial neural networks. **Computers & Chemical Engineering**, Elsevier BV, v. 20, p. S407–S412, jan 1996.

HEIAT, A. Comparison of artificial neural network and regression models for estimating software development effort. **Information and Software Technology**, Elsevier BV, v. 44, n. 15, p. 911–922, dec 2002.

- HRUSCHKA, H. Determining market response functions by neural network modeling: a comparison to econometric techniques. **European Journal of Operational Research**, Elsevier, v. 66, n. 1, p. 27–35, 1993.
- JANG, J.-S. Anfis: adaptive-network-based fuzzy inference system. **IEEE transactions on systems, man, and cybernetics**, IEEE, v. 23, n. 3, p. 665–685, 1993.
- JANG, J.-S. R.; SUN, C.-T.; MIZUTANI, E. **Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence**. [S.l.]: prentice hall, 1997.
- KIM, G.-H.; AN, S.-H.; KANG, K.-I. Comparison of construction cost estimating models based on regression analysis, neural networks, and case-based reasoning. **Building and Environment**, Elsevier BV, v. 39, n. 10, p. 1235–1242, oct 2004.
- LECUN, Y. A. et al. Efficient BackProp. In: **Lecture Notes in Computer Science**. [S.l.]: Springer Berlin Heidelberg, 2012. p. 9–48.
- LEE, J.; UM, K. A comparison in a back-bead prediction of gas metal arc welding using multiple regression analysis and artificial neural network. **Optics and Lasers in Engineering**, Elsevier BV, v. 34, n. 3, p. 149–158, sep 2000.
- LEVENBERG, K. A method for the solution of certain non-linear problems in least squares. **Quarterly of Applied Mathematics**, American Mathematical Society (AMS), v. 2, n. 2, p. 164–168, jul 1944.
- LI, S. et al. Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation. **Journal of Solar Energy Engineering**, ASME International, v. 123, n. 4, p. 327, 2001.
- LIAO, J. J. Z. An improved concordance correlation coefficient. **Pharmaceutical Statistics**, Wiley, v. 2, n. 4, p. 253–261, 2003.
- LIN, L. I.-K. A concordance correlation coefficient to evaluate reproducibility. **Biometrics**, JSTOR, v. 45, n. 1, p. 255, mar 1989.
- LOFGREEN, G. P.; GARRETT, W. N. A system for expressing net energy requirements and feed values for growing and finishing beef cattle. **Journal of Animal Science**, Oxford University Press (OUP), v. 27, n. 3, p. 793, 1968.
- MAPA/AGE. **Projeções do agronegócio - Brasil 2014/15 a 2024/25**. Ministério da Agricultura, Pecuária e Abastecimento - Assessoria de Gestão Estratégica, 2015. Disponível em: <<http://www.agricultura.gov.br/>>.
- MARCONDES, M. I. et al. Predicting efficiency of use of metabolizable energy to net energy for gain and maintenance of nellore cattle. **Journal of Animal Science**, v. 91, p. 4887–4898, 2013.
- MARCONDES, M. I. et al. Using growth and body composition to determine weight at maturity in nellore cattle. **Animal Production Science**, CSIRO Publishing, v. 56, n. 7, p. 1121, 2016.

MARQUARDT, D. W. An algorithm for least-squares estimation of nonlinear parameters. **SIAM Journal on Applied Mathematics**, SIAM, v. 11, n. 2, p. 431–441, 1963. Disponível em: <<http://scitation.aip.org/getabs/servlet/GetabsServlet?prog=normal&id=SMJMAP000011000002000431000001&idtype=cvips&gifs=yes>>.

MØLLER, M. F. A scaled conjugate gradient algorithm for fast supervised learning. **Neural Networks**, v. 6, n. 4, p. 525 – 533, 1993. ISSN 0893-6080. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0893608005800565>>.

MOSBAEK, E. J.; THEIL, H. Economic forecasts and policy. **Econometrica**, JSTOR, v. 30, n. 3, p. 620, jul 1962.

MOTTAGHITALAB, M. et al. Predicting caloric and feed efficiency in turkeys using the group method of data handling-type neural networks. **Poultry Science**, v. 89, n. 6, p. 1325, 2010. Disponível em: <<http://dx.doi.org/10.3382/ps.2009-00490>>.

NRC. **Nutrient Requirements of Beef Cattle**. [S.l.]: National Academies Press, 1984.

NRC. **Nutrient Requirements of Beef Cattle**. [S.l.]: National Academies Press, 1996.

NRC. **Nutrient Requirements of Beef Cattle**. [S.l.]: National Academies Press, 2000.

POTOKAR, M. Čandek et al. An attempt to predict conformation and fatness in bulls by means of artificial neural networks using weight, age and breed composition information. **Italian Journal of Animal Science**, v. 14, n. 1, p. 3198, 2015. Disponível em: <<http://dx.doi.org/10.4081/ijas.2015.3198>>.

RAJA, T.; RUHIL, A.; GANDHI, R. Comparison of connectionist and multiple regression approaches for prediction of body weight of goats. **Neural Computing and Applications**, v. 21, n. 1, p. 119–124, fev. 2012. Disponível em: <<https://link.springer.com/article/10.1007/s00521-011-0637-z>>.

RAZI, M.; ATHAPPILLY, K. A comparative predictive analysis of neural networks (NNs), nonlinear regression and classification and regression tree (CART) models. **Expert Systems with Applications**, Elsevier BV, v. 29, n. 1, p. 65–74, jul 2005.

SANTOS, S. A. et al. A fuzzy logic-based tool to assess beef cattle ranching sustainability in complex environmental systems. **Journal of Environmental Management**, v. 198, p. 95–106, 2017. ISSN 0301-4797. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0301479717304243>>.

SETYAWATI, B. R.; CREESE, R. C.; SAHIRMAN, S. Neural networks for cost estimation (part 2). **AACE International Transactions**, American Association of Cost Engineers, p. ES141, 2003.

SETYAWATI, B. R.; SAHIRMAN, S.; CREESE, R. Neural networks for cost estimation. **AACE International Transactions**, p. EST131–EST139, 01 2002.

SHAHINFAR, S. et al. Prediction of breeding values for dairy cattle using artificial neural networks and neuro-fuzzy systems. **Computational and Mathematical Methods in Medicine**, v. 2012, ago. 2012.

SIMSEK, H. et al. Estimation of nutrient concentrations in runoff from beef cattle feedlot using adaptive neuro-fuzzy inference systems. **Neural Network World**, Czech Technical University in Prague, Faculty of Transportation Sciences, v. 25, n. 5, p. 501, 2015.

SMITH, J. A. B.; LIVESTOCK, A. R. C. G. B. T. C. on the Nutrient Requirements of F.; RUMINANTS, A. R. C. G. B. W. P. on the Nutrient Requirements of. Book; Book/Illustrated. **The nutrient requirements of farm livestock. No.2, Ruminants : summaries of estimated requirements.** [S.l.]: London : Agricultural Research Council, 1965. Prepared by the Working Party on Nutrient Requirements of Ruminants, chairman J.A.B. Smith, on behalf of the Technical Committee on the Nutrient Requirements of Farm Animals.

SOUSA, R. V. de et al. Development and evaluation of a fuzzy logic classifier for assessing beef cattle thermal stress using weather and physiological variables. **Computers and Electronics in Agriculture**, v. 127, p. 176 – 183, 2016. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169916303933>>.

SPECHT, D. A general regression neural network. **IEEE Transactions on Neural Networks**, Springer US, Cambridge, MA, USA, v. 2, n. 6, p. 568–576, jan 1991. ISSN 0304-3800. High Pressure Processing Special Issue Section. Disponível em: <<https://ieeexplore.ieee.org/document/97934/>>.

SPECHT, D. F. Generation of polynomial discriminant functions for pattern recognition. **IEEE Transactions on Electronic Computers**, Springer US, Cambridge, MA, USA, EC-16, n. 3, p. 308–319, jun 1967. ISSN 0304-3800. High Pressure Processing Special Issue Section. Disponível em: <<https://ieeexplore.ieee.org/document/97934/>>.

SPECHT, D. F. Probabilistic neural networks. **Neural Networks**, Springer US, Cambridge, MA, USA, v. 3, n. 1, p. 109–118, jan. 1990. ISSN 0304-3800. High Pressure Processing Special Issue Section. Disponível em: <<https://ieeexplore.ieee.org/document/97934/>>.

SUGENO, M. **Industrial Applications of Fuzzy Control.** [S.l.]: Elsevier Science Ltd, 1985. ISBN 0444878297.

TAKAGI, T.; SUGENO, M. Fuzzy identification of systems and its applications to modeling and control. **IEEE transactions on systems, man, and cybernetics**, IEEE, n. 1, p. 116–132, 1985.

TAVARES, G. F. et al. Bioclimatic zoning for beef cattle in brazil with the aid of intelligent systems. **JABB-Online Submission System**, v. 4, n. 4, p. 116–123, 2016.

TEDESCHI, L. O. Assessment of the adequacy of mathematical models. **Agricultural Systems**, v. 89, n. 2, p. 225 – 247, 2006. ISSN 0308-521X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0308521X05002568>>.

THEODORIDIS, S.; KOUTROUMBAS, K. **Pattern Recognition, Fourth Edition.** 4th. ed. [S.l.]: Academic Press, 2008. ISBN 1597492728, 9781597492720.

TUKEY, J. W. Comparing individual means in the analysis of variance. **Biometrics**, JSTOR, v. 5, n. 2, p. 99, jun 1949.

VALADARES FILHO, S. C. et al. Nutrient requirements of zebu and crossbred cattle - BR-CORTE. In: **Exigências Nutricionais de Zebuínos Puros e Cruzados - BR-CORTE.** [S.l.]: Editora Federal de Viçosa, 2016.

VALADARES FILHO, S. d. C.; MARCONDES, M. I.; CHIZZOTTI, M. L. **Nutrient requirements of Zebu beef cattle - BR-CORTE**. 2. ed. [S.l.]: Federal Univeristy of Viçosa, Department of Animal Science, 2010. ISBN 9788590604150.

VENTURA, R. V. et al. Uso de redes neurais artificiais na predição de valores genéticos para peso aos 205 dias em bovinos da raça tabapuã. **Arquivo Brasileiro de Medicina Veterinária e Zootecnia**, FapUNIFESP (SciELO), v. 64, n. 2, p. 411–418, apr 2012. ISSN 0102-0935. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0102-09352012000200022&nrm=iso>.

WIDROW, B.; HOFF, M. Adaptive switching techniques. **IRE WESCON Conv. Rec**, v. 4, p. 96–104, 1960.

XIANGJUN, W.; AL-HASHIMI, M. M. Y. The comparison of adaptive neuro-fuzzy inference system (ANFIS) with nonlinear regression for estimation and prediction. In: **2012 International Conference on Information Technology and e-Services**. [S.l.]: IEEE, 2012.

YAGER, R. R.; FILEV, D. P. Generation of fuzzy rules by mountain clustering. **Journal of Intelligent & Fuzzy Systems**, IOS Press, v. 2, n. 3, p. 209–219, 1994.

YILMAZ, I.; KAYNAR, O. Multiple regression, ANN (RBF, MLP) and ANFIS models for prediction of swell potential of clayey soils. **Expert Systems with Applications**, Elsevier BV, v. 38, n. 5, p. 5958–5966, may 2011.

APÊNDICE A – Consumo de Energia Metabolizável (CEM)

A determinação correta das exigências de energia para bovinos em crescimento e terminação é um dos fatores mais importantes de um sistema de alimentação (VALADARES FILHO et al., 2016). Utiliza-se como base para as estimativas das exigências energéticas o trabalho de Lofgreen e Garrett (1968), mais tarde conhecido como *California Net Energy System* (CNES) que definiu as bases das recomendações dos sistemas norte americanos subsequentes (NRC, 1984; NRC, 1996; NRC, 2000; BCNRM, 2016). O resumo do Sistema CNES pode ser representado pela equação abaixo:

$$EM_t = EM_m + EM_g = (EL_m/k_m) + (EL_g/k_g) \quad (1)$$

Onde EM_t é a exigência de energia metabolizável total, EM_m é a exigência de energia metabolizável para manutenção, EM_g é a exigência de energia metabolizável para ganho, EL_m é a exigência de energia líquida para manutenção, k_m é a eficiência de utilização da EM_m , EL_g é a exigência de energia líquida para ganho e k_g é a eficiência de utilização da EM_g .

Uma vez feitas as estimativas de exigências energéticas pela Equação 1, é necessário a determinação das frações apresentadas, que pode ser feito pelos resultados de experimentos de abate comparativo (GARRETT; MEYER; LOFGREEN, 1959). Nesses experimentos são medidas a energia metabolizável consumida (EM_t) e a energia retida na forma de tecidos no corpo do animal (EL_g). A energia líquida gasta com manutenção (EL_m) e as ineficiências de uso de energia de manutenção e ganho ($(1 - k_m)e(1 - k_g)$, respectivamente) representam a energia consumida que foi transformada em calor e são estimadas por diferença com base na Primeira Lei da Termodinâmica (CLAUSIUS, 1850). A representação dessas relações pode ser vista nas equações abaixo.

$$EM_t = CEM = ER + PCalor \quad (2)$$

$$ER = EL_g \quad (3)$$

$$PCalor = EL_m + ((1 - k_m) \cdot EM_m) + ((1 - k_g) \cdot EM_g) \quad (4)$$

Onde CEM é o consumo de energia metabolizável, ER é a energia retida e $PCalor$ é a produção de calor.

Como o valor da $PCalor$ é determinado por diferença e três das variáveis apresentadas na Equação 4 são oriundas da $PCalor$, utiliza-se de modelos matemáticos para estimar os valores de EL_m , k_m e k_g .

Exigências de Energia para Manutenção

Exigências de Energia Líquida para Manutenção

A exigência de energia líquida para manutenção (EL_m) pode ser entendida como toda a produção de calor do animal no estado de jejum absoluto e está correlacionada com o atendimento das funções basais, como a manutenção da homeotermia, circulação, respiração, manutenção de sistemas enzimáticos e síntese de tecidos, além do atendimento de atividades voluntárias como ruminar e caminhar (GARRETT; MEYER; LOFGREEN, 1959).

A EL_m pode ser obtida diretamente através da utilização de técnicas respirométricas, onde o animal, em jejum absoluto, é mantido no interior de uma câmara e tem toda a sua produção de gás carbônico e metano e seu consumo de oxigênio mensurados (FERRELL; OLTJEN, 2008). Partindo-se do princípio que todo o calor produzido é oriundo da oxidação metabólica dos compostos orgânicos, onde é necessário o consumo de oxigênio para produzir gás carbônico, torna-se possível estimar a EL_m (SMITH; LIVESTOCK; RUMINANTS, 1965)

Exigências de Energia Metabolizável para Manutenção

A utilização do valor encontrado para a EL_m é limitada e não permite aplicação prática na formulação de dietas, uma vez que animais de produção não se encontram em estado de jejum, assim, torna-se necessário calcular a exigência de manutenção em uma forma mais aplicável. Nesse contexto, utiliza-se o conceito da EM_m , que pode ser entendida como o ponto em que todo o consumo de energia metabolizável é utilizado para a produção de calor, ou seja, não há retenção de energia no corpo (LOFGREEN; GARRETT, 1968).

Exigências de Energia para Ganho

Peso de Corpo Vazio Equivalente

A aplicação de mensurações de desempenho como o ganho médio diário é essencial em sistemas de predição de exigências nutricionais. Porém, tão importante quanto ter noção do ganho médio diário dos animais, é entender a composição que um ganho apresenta (MARCONDES et al., 2016). Uma das formas de inferir sobre a composição do ganho dos animais é através do peso corporal à maturidade, uma vez que quanto mais próximo deste peso, maior será a tendência a depositar maiores teores de gordura em relação à proteína no peso de corpo vazio.

O NRC (2000) sugere que seja feita uma correção de peso dos animais de diferentes tamanhos corporais ou pesos à maturidade, o que geraria um valor que equivaleria todos os animais, sendo conhecido como Peso de Corpo Vazio Equivalente (PCVZeq). Assim, pode-se calcular o PCVZeq a partir do valor de peso à maturidade para cada tipo de animal e um valor de referência de peso de corpo vazio.

Exigência de Energia Líquida para Ganho

A exigência de energia líquida para ganho de peso (EL_g) pode ser entendida como toda a energia que é retida no peso de corpo vazio dos animais na forma de proteína ou gordura (GARRETT; MEYER; LOFGREEN, 1959). Sendo assim, a composição do ganho de corpo vazio é a principal determinante das exigências de energia para ganho de peso. Nesse sentido, o que determina a composição do ganho de corpo vazio não é o peso corporal absoluto, mas o peso relativo ao peso à maturidade do animal, sendo que quanto mais próximo ao peso à maturidade, maior é a deposição de gordura em relação à proteína, o que tende a aumentar as exigências de ganho de peso (MARCONDES et al., 2016).

Exigências de Energia Metabolizável para Ganho

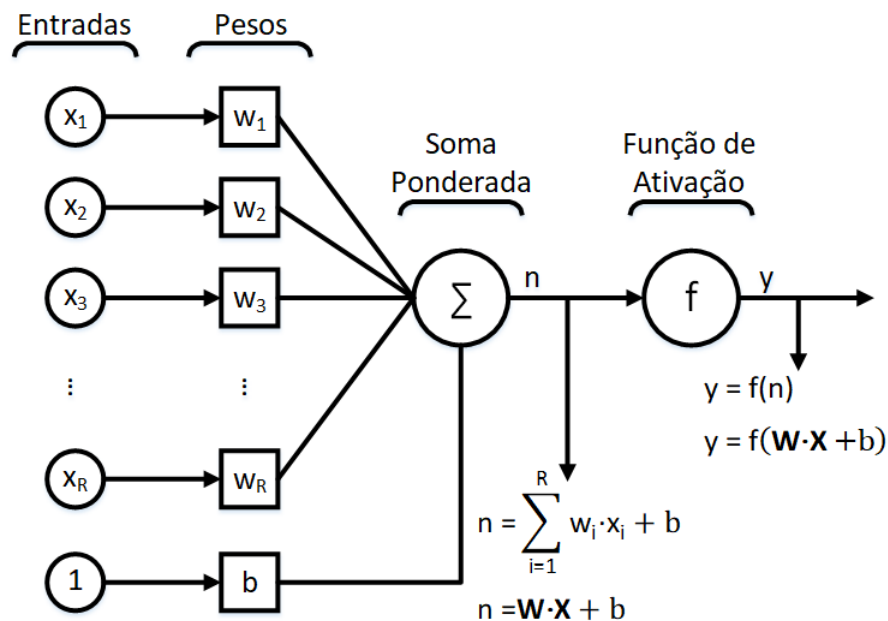
Da mesma forma que a EL_m , a EL_g não apresenta aplicação direta na formulação de dietas, necessitando ser convertida para a forma de exigência de energia metabolizável para ganho (EM_g). Assim, para converter a EL_g em EM_g , torna-se necessário conhecer a eficiência de utilização da energia metabolizável para ganho de peso (k_g).

APÊNDICE B – Redes Neurais Perceptron Multicamadas (MLP)

Arquitetura

O Perceptron Multicamadas (MLP) consiste em um sistema de neurônios artificiais chamados *Perceptrons* (Figura 1) interconectados e organizados em camadas, como pode ser visto na Figura 2, que representam um modelo de mapeamento não-linear entre um vetor de entrada e um vetor de saída. Os neurônios são conectados por pesos e seus sinais de saída são uma função da soma das entradas, modificadas por uma função de ativação não-linear (THEODORIDIS; KOUTROUMBAS, 2008). A arquitetura de um MLP é variável, mas em geral consiste em várias camadas de tamanhos variados. A camada de entrada não realiza um papel computacional, servindo apenas para passar o vetor de entrada à rede, por isso alguns autores não a contabilizam como uma camada realmente, o que será feito neste trabalho. A rede pode ter uma ou mais camadas escondidas e uma camada de saída. As Figuras 1 e 2 apresentam a arquitetura de um neurônio artificial do tipo Perceptron e a arquitetura geral de uma rede MLP com duas camadas escondidas e uma saída linear, respectivamente.

Figura 1 – Arquitetura de um neurônio artificial Perceptron.



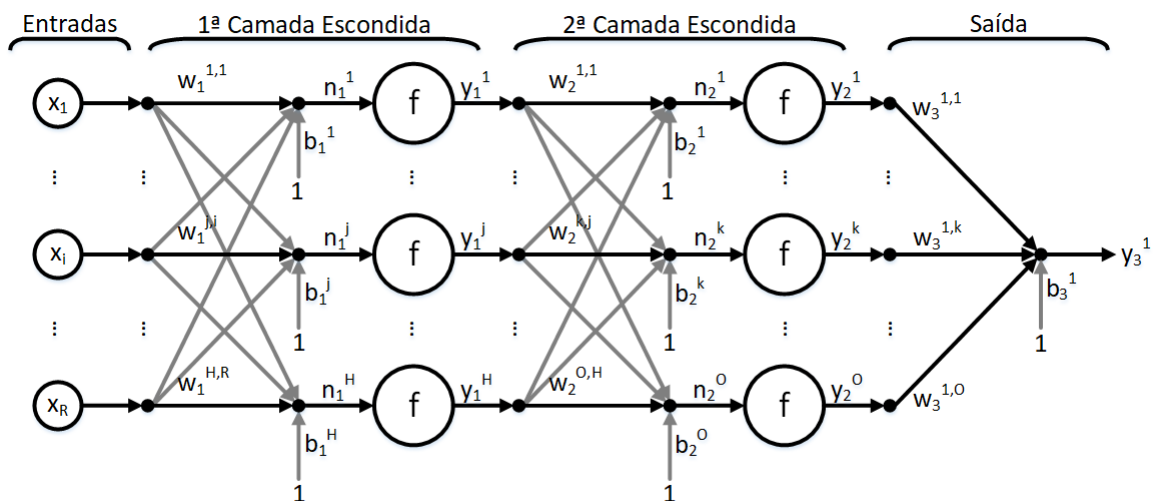
Fonte: Autor.

Na Figura 1, x_i representa o i -ésimo elemento do vetor de entrada $\mathbf{x} = [x_1, x_2, \dots, x_R]^T$, com R elementos; w_i representa o i -ésimo elemento do vetor de pesos $\mathbf{w} = [w_1, w_2, \dots, w_R]$, também com R elementos; b é o valor de viés (também chamado de bias ou tendência); n é a saída não ativada e y é a saída ativada pela função de ativação $f(\cdot)$. A função implementada pelo Perceptron é descrita pelas Equações 5 e 6 (representação matricial).

$$y = f\left(\sum_{i=1}^R w_i \cdot x_i + b\right) \quad (5)$$

$$y = f(\mathbf{w} \cdot \mathbf{x} + b) \quad (6)$$

Figura 2 – Arquitetura geral de uma rede MLP com duas camadas escondidas de tamanhos H e O , respectivamente, e uma camada de saída linear.



Fonte: Autor.

Na Figura 2, cada elemento do vetor de entrada \mathbf{x} é apresentado a H neurônios na primeira camada conectados por seus respectivos pesos sinápticos $\mathbf{W}_1 = [\mathbf{w}_1^1, \mathbf{w}_1^2, \dots, \mathbf{w}_1^R]$, onde \mathbf{w}_1^i é o i -ésimo vetor de pesos correspondente ao i -ésimo elemento de \mathbf{x} e é construído como $\mathbf{w}_1^i = [w_1^{1,i}, w_1^{2,i}, \dots, w_1^{H,i}]$; é feita a soma ponderada das entradas e seus pesos com o respectivo valor de viés, $\mathbf{b}_1 = [b_1^1, b_1^2, \dots, b_1^H]$ e o resultado, $\mathbf{n}_1 = [n_1^1, n_1^2, \dots, n_1^H]$, “é ativado” pela correspondente função de ativação, \mathbf{f} , gerando a saída da rede $\mathbf{y}_1 = [y_1^1, y_1^2, \dots, y_1^H]$. A saída da primeira camada é então apresentada como entrada para a segunda camada, que realiza as operações de forma análoga. O processo continua até que a saída final seja gerada, no caso da Figura 2, a terceira camada. A formulação completa é apresentada a seguir pelas Equações 7-17.

$$n_1^i = \sum_{j=1}^R (w_1^{i,j} \cdot x_j) + b_1^i, i = 1, 2, \dots, H; \quad (7)$$

$$y_1^i = f(n_1^i), i = 1, 2, \dots, H; \quad (8)$$

$$n_2^i = \sum_{j=1}^H (w_2^{i,j} \cdot y_1^j) + b_2^i, i = 1, 2, \dots, O; \quad (9)$$

$$y_2^i = f(n_2^i), i = 1, 2, \dots, O; \quad (10)$$

$$y_3^1 = \sum_{j=1}^O (w_3^{1,j} \cdot y_2^j) + b_3^1; \quad (11)$$

As Equações 7-11 podem ser escritas matricialmente como:

$$\mathbf{n}_1 = \mathbf{W}_1 \cdot \mathbf{x} + \mathbf{b}_1; \quad (12)$$

$$\mathbf{y}_1 = \mathbf{f}(\mathbf{n}_1); \quad (13)$$

$$\mathbf{n}_2 = \mathbf{W}_2 \cdot \mathbf{y}_1 + \mathbf{b}_2; \quad (14)$$

$$\mathbf{y}_2 = \mathbf{f}(\mathbf{n}_2); \quad (15)$$

$$\mathbf{y}_3 = \mathbf{W}_3 \cdot \mathbf{y}_2 + \mathbf{b}_3; \quad (16)$$

Substituindo as Equações 12-15 na Equação 16, obtemos a expressão final de uma rede MLP como a representada na Figura 2:

$$\mathbf{y}_3 = \mathbf{W}_3 \cdot \mathbf{f}_2(\mathbf{W}_2 \cdot \mathbf{f}_1(\mathbf{W}_1 \cdot \mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2) + \mathbf{b}_3; \quad (17)$$

Ao selecionar um conjunto apropriado de pesos e funções de ativação, foi mostrado que a Equação 17, implementada pelo MLP pode aproximar qualquer função suave e mensurável

entre os vetores de entrada e saída. A prova pode ser encontrada em Theodoridis e Koutroumbas (2008).

O MLP possui a capacidade de aprender através do treinamento por um conjunto de dados de treinamento, que consiste em uma série de entradas e os vetores de saída associados a cada entrada. Durante o treinamento é apresentado repetidamente ao MLP os dados de treinamento e os pesos na rede são ajustados até que o mapeamento entrada-saída desejado ocorra. Este tipo de treinamento, que requer um vetor de saída associado às entradas é chamado de treinamento supervisionado. Durante o treinamento a saída do MLP, para um dado vetor de entrada, pode não ser igual à saída desejada. Um sinal de erro é então definido pela diferença entre a saída atual e a desejada. O treinamento usa a magnitude desse sinal de erro para determinar qual o nível em que os pesos da rede devem ser ajustados de forma que o erro geral da rede seja reduzido. Existem vários algoritmos que podem ser usados para treinar uma rede MLP. Uma vez treinado com um conjunto de dados de treinamento adequadamente representativo, o MLP pode generalizar para novos dados de entrada não vistos.

Funções de Ativação

São as funções de ativação não-lineares que dão à rede neural MLP suas capacidades não-lineares. Algumas das formas mais comuns de funções de ativação são as funções da família sigmoide que são funções monotônicas que possuem assíntotas em algum valor finito a medida que aproxima-se de $\pm\infty$. Os exemplos mais comuns são a função logística padrão e a tangente hiperbólica, descritas pelas Equações 18 e 19, respectivamente, e mostradas na Figura 3. Sigmoides que são simétricas em relação à origem (ver Figura 3b) são preferíveis pois elas produzem saídas (que são entradas para a próxima camada) que são na média próximas de zero (LECUN et al., 2012).

$$f(x) = \frac{1}{(1 + e^{-x})}; \quad (18)$$

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}; \quad (19)$$

Dada limitações computacionais e/ou de tempo de processamento, a função Tangente Hiperbólica pode ser substituída por uma função matematicamente equivalente, chamada Tangente Hiperbólica Sigmoidal, de cálculo mais simples e conseqüentemente rápido. Apesar de

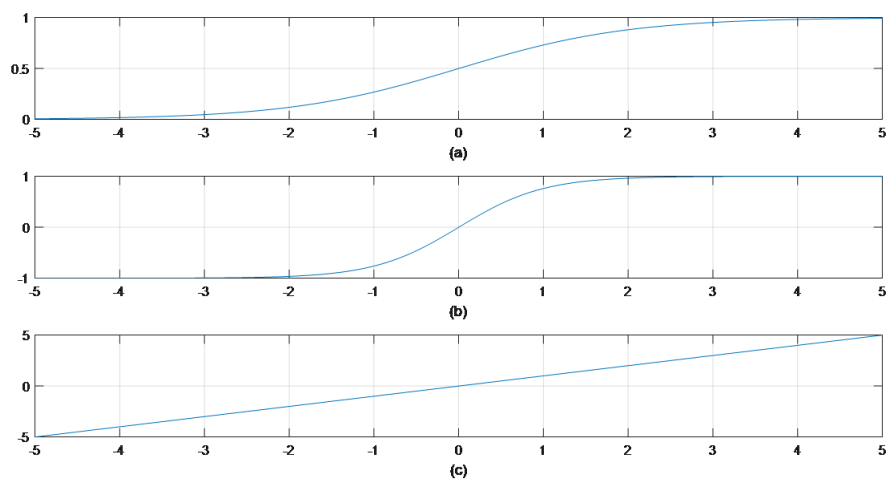
apresentar diferenças numéricas em relação à Tangente Hiperbólica, ela é uma boa alternativa para redes neurais, onde o tempo de processamento é um limitante e a forma exata da função, resguardadas as propriedades matemáticas, não é tão importante. A Equação 20 apresenta a formulação para a Tangente Hiperbólica Sigmoidal.

$$f(x) = \text{tansig}(x) = \frac{2}{1 - e^{-2x}} - 1 \quad (20)$$

Para aproximação de funções, as funções de ativação da camada de saída devem ser lineares, permitindo quaisquer saídas reais para o resultado da rede. A função mais simples é a linear pura, que simplesmente copia a sua entrada para a saída, como pode ser visto na Equação 21 e na Figura 3c.

$$f(x) = x; \quad (21)$$

Figura 3 – (a) Função Logística Padrão. (b) Função Tangente Hiperbólica. (c) Função Linear Pura.



Fonte: Autor.

Algoritmos de Treinamento

Treinamento *Backpropagation* baseado no Método do Gradiente Conjugado Escalonado (SCG)

Em problemas de aproximação utilizando um número finito de dados amostrados e definido um modelo de aproximação $\hat{g}(\cdot, \theta)$, a distância entre a função a ser aproximada e sua aproximação $dist(g(\cdot), \hat{g}(\cdot, \theta))$ é uma função apenas do vetor de parâmetros $\theta \in \mathfrak{R}^P$. Tomando a norma euclidiana como medida de distância, temos a seguinte expressão:

$$J(\theta) = \frac{1}{N} \sum_{l=1}^N (g(\mathbf{x}) - \hat{g}(\mathbf{x}, \theta))^2 \quad (22)$$

O funcional $J : \mathfrak{R}^P \rightarrow \mathfrak{R}$ é denominado superfície de erro do problema de aproximação, pois pode ser interpretado como uma hipersuperfície localizada “acima” do espaço de parâmetros \mathfrak{R}^P , sendo que cada ponto $\theta \in \mathfrak{R}^P$ corresponde uma “altura” $J(\theta)$. O termo funcional corresponde a toda função $f : X \subset \mathfrak{R}^n \rightarrow \mathfrak{R}$, e por isso o problema de minimizar $J(\theta)$ torna-se um problema de minimização funcional.

Dada a superfície de erro, o problema de aproximação passa a ser um problema de otimização cuja solução é o vetor $\theta^* \in \mathfrak{R}^P$ que minimiza $J(\theta)$, ou seja,

$$\theta^* = \arg \min_{\theta \in \mathfrak{R}^P} J(\theta) \quad (23)$$

Para a maioria dos modelos de aproximação $\hat{g}(\cdot, \theta)$, o problema de otimização apresentado na Equação 23 tem a desvantagem de ser não-linear e não-convexo, mas as vantagens de ser irrestrito e permitir a aplicação de conceitos de cálculo variacional na obtenção da solução θ^* . Estas características impedem a existência de uma solução analítica, mas permitem obter processos iterativos de solução, a partir de uma condição inicial θ_0 , na forma:

$$\theta_{i+1} = \theta_i + \alpha_i \mathbf{d}_i, \quad i \geq 0, \quad (24)$$

onde $\theta_i \in \mathfrak{R}^P$ é o vetor de parâmetros, $\alpha_i \in \mathfrak{R}^+$ é um escalar que define o passo de ajuste e $\mathbf{d}_i \in \mathfrak{R}^P$ é a direção de ajuste, todos definidos na iteração i . Os algoritmos de otimização são aplicados na obtenção da direção de ajuste do processo iterativo dado pela equação 24.

Antes de apresentar o Algoritmo SCG, será apresentado o método dos gradientes conjugados e um resultado intermediário chamado Método das Direções Conjugadas.

Método das Direções Conjugadas

A Equação 24 é chamada também de lei de adaptação e, havendo convergência, a solução ótima $\theta^* \in \mathfrak{R}^P$ pode ser expressa na forma:

$$\theta^* = \alpha_0 \mathbf{d}_0 + \alpha_1 \mathbf{d}_1 + \dots = \sum_i \alpha_i \mathbf{d}_i \quad (25)$$

Assumindo por hipótese que o conjunto $\{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{P-1}\}$ forma uma base de \mathfrak{R}^P e $\alpha = (\alpha_0 \dots \alpha_{P-1})^T$ é a representação de θ^* nesta base, então é possível obter θ^* em P iterações da Equação 24 na forma

$$\theta^* = \alpha_0 \mathbf{d}_0 + \alpha_1 \mathbf{d}_1 + \dots + \alpha_{P-1} \mathbf{d}_{P-1} = \sum_{i=0}^{P-1} \alpha_i \mathbf{d}_i \quad (26)$$

Dada uma matriz \mathbf{A} simétrica de dimensão $P \times P$, as direções $\mathbf{d}_i \in \mathfrak{R}^P, i = 0, \dots, P-1$, são ditas serem \mathbf{A} -conjugadas ou \mathbf{A} -ortogonais se: $\mathbf{d}_j^T \mathbf{A} \mathbf{d}_i = 0$, para $i \neq j$ e $i, j = 0, \dots, P-1$.

Se a matriz \mathbf{A} for também definida positiva, o conjunto de P direções \mathbf{A} -conjugadas forma uma base do \mathfrak{R}^P . Dessa forma os coeficientes $\alpha_j^*, j = 1, \dots, P-1$, podem ser determinados pelo procedimento descrito abaixo.

Dada uma matriz \mathbf{A} simétrica, definida positiva e de dimensões $P \times P$, multiplicando-se a Equação 26 à esquerda por $\mathbf{d}_j^T \mathbf{A}$, com $0 \leq j \leq P-1$, resulta:

$$\mathbf{d}_j^T \mathbf{A} \theta^* = \sum_{i=0}^{P-1} \alpha_i^* \mathbf{d}_j^T \mathbf{A} \mathbf{d}_i, \quad i = 1, \dots, P-1 \quad (27)$$

Escolhendo as direções $\mathbf{d}_i \in \mathfrak{R}^P$, como sendo \mathbf{A} -conjugadas, é possível aplicar os resultados acima para obter:

$$\alpha_i^* = \frac{\mathbf{d}_j^T \mathbf{A} \theta^*}{\mathbf{d}_j^T \mathbf{A} \mathbf{d}_j}, \quad j = 1, \dots, P-1 \quad (28)$$

É necessário eliminarmos θ^* da expressão 28, e para que isso seja feito são necessários duas hipóteses adicionais:

1. Suponha que o problema é quadrático, ou seja: $J(\theta) = \frac{1}{2} \theta^T \mathbf{Q} \theta - \mathbf{b}^T \theta$, então no ponto de mínimo θ^* , é válida a expressão:

$$\nabla J(\theta^*) = 0 \Rightarrow \mathbf{Q} \theta^* - \mathbf{b} = 0 \Rightarrow \mathbf{Q} \theta^* = \mathbf{b}. \quad (29)$$

2. Suponha que $\mathbf{A} = \mathbf{Q}$. Sendo assim a Equação 28 resulta em:

$$\alpha_j^* = \frac{\mathbf{d}_j^T \mathbf{b}}{\mathbf{d}_j^T \mathbf{A} \mathbf{d}_j}, \quad j = 1, \dots, P-1 \quad (30)$$

e o ponto de mínimo é dado por:

$$\theta^* = \sum_{j=0}^{P-1} \frac{\mathbf{d}_j^T \mathbf{b}}{\mathbf{d}_j^T \mathbf{A} \mathbf{d}_j} \mathbf{d}_j \quad (31)$$

Assumindo solução iterativa com θ^* sendo expresso na forma da Equação 31, os coeficientes $\alpha_j^*, j = 1, \dots, P-1$, são dados por:

$$\theta^* = \theta_0 + \alpha_0^* \mathbf{d}_0 + \alpha_1^* \mathbf{d}_1 + \dots + \alpha_{P-1}^* \mathbf{d}_{P-1}, \quad (32)$$

$$\alpha_j^* = -\frac{\mathbf{d}_j^T \mathbf{Q}(\theta^* - \theta_0)}{\mathbf{d}_j^T \mathbf{Q} \mathbf{d}_j}, \quad j = 1, \dots, P-1. \quad (33)$$

Na j -ésima iteração, e levando-se em conta a Equação 32, obtém-se:

$$\alpha_j^* = -\frac{\mathbf{d}_j^T \nabla J(\theta_j)}{\mathbf{d}_j^T \mathbf{Q} \mathbf{d}_j}, \quad j = 1, \dots, P-1, \quad (34)$$

e a lei de ajuste do método das direções conjugadas é dada por:

$$\theta_{i+1} = \theta_i - \frac{\mathbf{d}_i^T \nabla J(\theta_i)}{\mathbf{d}_i^T \mathbf{Q} \mathbf{d}_i} \mathbf{d}_i \quad (35)$$

Método do Gradiente Conjugado

Antes de se aplicar a lei de ajuste dada pela Equação 35, é necessário obter as direções \mathbf{Q} -conjugadas $\mathbf{d}_i \in \mathfrak{R}^P, i = 0, \dots, P-1$. Uma maneira de determinar estas direções é tomá-las na forma (BAZARAA; SHETTY, 1993):

$$\begin{cases} \mathbf{d}_0 = -\nabla J(\theta_0) \\ \mathbf{d}_{i+1} = -\nabla J(\theta_{i+1}) + \beta_i \mathbf{d}_i \quad i \geq 0 \\ \text{com } \beta_i = \frac{\nabla J(\theta_{i+1})^T \mathbf{Q} \mathbf{d}_i}{\mathbf{d}_i^T \mathbf{Q} \mathbf{d}_i} \end{cases} \quad (36)$$

Método do Gradiente Conjugado Escalonado (SCG)

O Método do Gradiente Conjugado Escalonado (SCG) desenvolvido por (MØLLER, 1993) é um método de segunda ordem, isto é, que utiliza informações sobre a derivada segunda da função custo (Eq. 22), que combina a abordagem desenvolvida pelo Método de Levenberg-Marquardt (LEVENBERG, 1944; MARQUARDT, 1963) com os métodos de gradiente conjugados. A ideia é substituir o procedimento busca unidimensional da taxa de aprendizagem, que envolve um grande número de avaliações da função ou sua derivada, por um escalonamento do passo de ajuste α .

Quando não estamos tratando com problemas quadráticos a matriz \mathbf{Q} deve ser aproximada pela matriz hessiana calculada no ponto θ_i , a Equação 34 torna-se:

$$\alpha_j^* = -\frac{\mathbf{d}_j^T \nabla J(\theta_j)}{\mathbf{d}_j^T \nabla^2 J(\theta_j) \mathbf{d}_j} \quad (37)$$

A ideia utilizada por Moller é estimar o termo determinado $s_j = \nabla^2 J(\theta_j) \mathbf{d}_j$ do método do gradiente conjugado por uma aproximação da forma:

$$s_j = \nabla^2 J(\theta_j) \mathbf{d}_j \approx \frac{\nabla J(\theta_j + \sigma_j \mathbf{d}_j) - \nabla J(\theta_j)}{\sigma_j}, \quad 0 < \sigma_j \ll 1. \quad (38)$$

A aproximação tende, no limite, ao valor de $\nabla^2 J(\theta_j) \mathbf{d}_j$. Combinando esta estratégia com a abordagem de gradiente conjugado e Levenberg-Marquardt, obtém-se um algoritmo diretamente aplicável ao treinamento de redes MLP. Isso é feito da seguinte maneira:

$$s_j = \frac{\nabla J(\theta_j + \sigma_j \mathbf{d}_j) - \nabla J(\theta_j)}{\sigma_j} + \lambda_j \mathbf{d}_j \quad (39)$$

Seja δ_j o denominador da Equação 37, resulta:

$$\delta_j = \mathbf{d}_j^T s_j \quad (40)$$

O ajuste do parâmetro λ_j a cada iteração e a análise do sinal de δ_j permitem determinar se a Hessiana é definida-positiva ou não.

A aproximação quadrática $J_{quad}(\theta)$, utilizada pelo algoritmo, nem sempre representa uma boa aproximação para $J(\theta)$, uma vez que λ_j escalona a matriz hessiana de maneira artificial. Um mecanismo para aumentar e diminuir λ_j é necessário para fornecer uma boa aproximação, mesmo quando a matriz for definida-positiva. Defina:

$$\Delta_j = \frac{J(\theta_j) - J(\theta_j + \alpha_j \mathbf{d}_j)}{J(\theta_j) - J_{quad}(\alpha_j \mathbf{d}_j)} = \frac{2\delta_j [J(\theta_j) - J(\theta_j + \alpha_j \mathbf{d}_j)]}{\mu_j^2}, \quad (41)$$

onde $\mu_j = -\mathbf{d}_j^T \nabla J(\theta_j)$.

O termo Δ_j represent uma medida da qualidade da aproximação $J_{quad}(\theta)$ em relação a $J(\theta_j + \alpha_j \mathbf{d}_j)$ no sentido de que quanto mais próximo de 1 estiver Δ_j , melhor é a aproximação. O algoritmo do Gradiente Conjugado Escalonado resultante é representado a seguir.

1. Escolha um vetor de parâmetros inicial θ_0 e escalares $0 < \sigma \ll 1$ e $\lambda_0 = 0$;

2. Faça: $p_0 = r_0 = -\nabla J(\theta_0)$; $j = 0$; sucesso = 1;

3. Se sucesso = 1, calcule a informação de segunda ordem:

$$\sigma_j = \frac{\sigma}{\|\mathbf{d}_j\|}; s_j = \frac{\nabla J(\theta_j + \sigma_j \mathbf{d}_j) - \nabla J(\theta_j)}{\sigma_j} + \lambda_j \mathbf{d}_j; \delta_j = p_j^T s_j;$$

4. Se $\delta_j \leq 0$, então faça a matriz hessiana definida-positiva:

$$\lambda_j^N = 2 \left[\lambda_j - \frac{\delta_j}{\|\mathbf{d}_j\|^2} \right]; \delta_j = \delta_j + \left(\lambda_j^N - \lambda_j \|\mathbf{d}_j\|^2 \right); \lambda_j = \lambda_j^N;$$

5. Calcule a taxa de ajuste:

$$\mu_j = p_j^T r_j; \alpha_j = \mu_j / \delta_j;$$

6. Calcule o parâmetro de comparação:

$$\Delta_j = \frac{2\delta_j [J(\theta_j) - J(\theta_j + \alpha_j \mathbf{d}_j)]}{\mu_j^2};$$

7. Se $\Delta_j \geq 0$ (o erro pode ser reduzido), então atualize o vetor de pesos:

$$\theta_{j+1} = \theta_j + \alpha_j \mathbf{d}_j; r_{j+1} = -\nabla J(\theta_{j+1});$$

8. Se $(j \bmod P) = 0$, então reinicialize o algoritmo:

$$p_{j+1} = r_{j+1}$$

Senão, defina uma nova direção conjugada:

$$\beta_j = \frac{\|r_{j+1}\|^2 - r_{j+1}^T r_j}{\mu_j}; p_{j+1} = r_{j+1} + \beta_j p_j$$

9. Se $\Delta_j \geq 0,75$, então faça $\lambda_j = 0,5\lambda_j$;

10. Se $\Delta_j \leq 0,25$, então faça $\lambda_j = 4\lambda_j$;
Senão, uma redução no erro não é possível: sucesso = 0;
11. Se $r_{j+1} > \varepsilon$, onde $\varepsilon \rightarrow 0$, então faça: $j = j + 1$; retorne ao passo 3;
12. Senão, o procedimento de ajuste chegou ao fim e θ_{j+1} é o ponto de mínimo.

APÊNDICE C – Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS)

Sistema de Inferência Neuro-*Fuzzy* Adaptativo

Uma das dificuldades das técnicas *Fuzzy* está em encontrar Funções de Pertinência (FP) e regras adequadas. Esse processo é basicamente baseado em uma exaustiva busca de tentativa e erro. É aqui que as técnicas de RNA podem proporcionar grande ajuda. Redes Neurais treinadas com algoritmos de aprendizagem corretos podem ser aplicadas a sistemas *fuzzy*. Alguns algoritmos eficientes para treinar RNA com essa finalidade foram desenvolvidos para ajudar na melhoria do ajuste de SIF. Jang (JANG, 1993; JANG; SUN; MIZUTANI, 1997) conduziu alguns dos primeiros trabalhos em sistemas neuro-*fuzzy*, com aplicações em controle de processos, porém, sua aplicabilidade se estendeu para áreas como análise de dados, classificação, detecção de imperfeições e suporte a tomada de decisões.

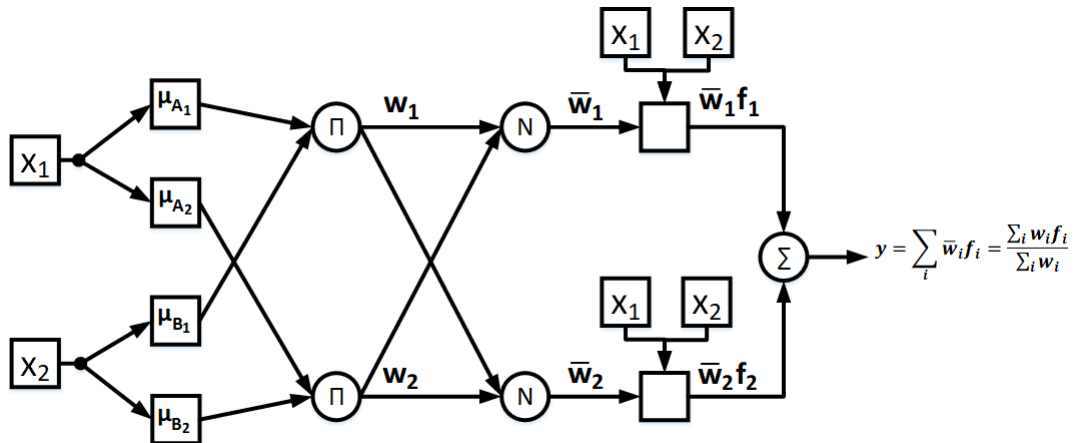
Jang desenvolveu um tipo de sistema especialista de inferência hibridizando técnicas neurais e *fuzzy* que trabalham como um SIF do tipo Sugeno chamado Sistemas de Inferência Neuro-*Fuzzy* Adaptativos (ANFIS) (JANG, 1993; JANG; SUN; MIZUTANI, 1997). Ambos modelos, ANFIS e RNA, possuem estruturas com algumas similaridades, porém as conexões no ANFIS apenas indicam o fluxo de informação entre os neurônios e não possuem pesos associados a elas. A arquitetura de um ANFIS possui cinco camadas, onde as camadas um e quatro consistem de neurônios adaptativos, enquanto as demais camadas, dois, três e cinco, são constituídas de neurônios fixos. Os neurônios adaptativos possuem parâmetros que são apropriadamente atualizados a cada iteração do algoritmo de treinamento, enquanto os neurônios fixos não possuem parâmetros (JANG, 1993; JANG; SUN; MIZUTANI, 1997).

Assuma um conjunto de regras típica de um modelo fuzzy Sugeno de primeira ordem, como as a seguir:

$$\begin{aligned} \text{Regra 1 : Se } x_1 \text{ é } A_1 \text{ e } x_2 \text{ é } B_1, \text{ então } f_1 &= p_1x_1 + q_1x_2 + r_1 \\ \text{Regra 2 : Se } x_1 \text{ é } A_2 \text{ e } x_2 \text{ é } B_2, \text{ então } f_2 &= p_2x_1 + q_2x_2 + r_2 \end{aligned} \quad (42)$$

onde x_1 e x_2 são as entradas, A_i e B_i são os conjuntos fuzzy, f_i são as saídas dentro da região fuzzy especificada pelas regras, p_i , q_i e r_i são os parâmetros consequentes. A arquitetura de um ANFIS que representa esse sistema é mostrada na Figura 4, onde um círculo indica um neurônio fixo e os retângulos indicam neurônios adaptativos.

Figura 4 – Arquitetura típica de um ANFIS com duas entradas.



Fonte: Arquivo do autor.

As camadas de um ANFIS funcionam como descrito a seguir.

- **Camada 1:** essa camada é chamada de camada de fuzzificação, onde cada um de seus neurônios é do tipo adaptativo. A saída de cada nó i dessa camada é o grau de pertinência fuzzy das entradas que pode ser expressado como:

$$O_{1,i} = \mu_{A_i}(x), \quad i = 1, 2 \quad (43)$$

$$O_{1,i} = \mu_{B_{i-2}}(x), \quad i = 3, 4 \quad (44)$$

Onde x e y são as entradas do nó i e A_i e B_{i-2} são as variáveis linguísticas associadas com esse neurônio. Assim, $O_{1,i}$ é o grau de pertinência em um dos conjunto fuzzy A_1, A_2, B_1, B_2 .

Usualmente, são escolhidas MF $\mu_A(x)$ como sendo funções em formato de sino com valores máximo de 1 e mínimo 0, como a descrita pela Equação 45:

$$\mu_A(x) = e^{-\frac{(x_i - a_i)^2}{2b_i^2}} \quad (45)$$

Onde a_i e b_i são o conjunto de parâmetros a ser ajustado e são conhecidos como parâmetros antecedentes.

- **Camada 2:** essa camada é chamada de camada de regras, com neurônios fixos cujas saídas são o produto de todas os sinais de chegada e podem ser representadas como:

$$O_{2,i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2 \quad (46)$$

Nesta camada, cada neurônio representa a força de ativação de uma determinada regra. Qualquer operador do tipo ‘E’ pode ser usado, além do produto.

- **Camada 3:** esta é chamada de camada de normalização, os neurônios dessa camada também são fixos e indicados como círculos. O i -ésimo neurônio calcula a razão entre a i -ésima força de ativação e a soma de todas as forças de ativação. As saídas são referidas como forças de ativação normalizadas, \bar{w}_i , e são calculadas como:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2 \quad (47)$$

- **Camada 4:** é chamada de camada de defuzzificação e seus neurônios são do tipo adaptativos. A saída de cada neurônio nessa camada é o produto da força de ativação normalizada por um polinômio de primeira ordem (para um sistema Sugeno do tipo 1). A função dos neurônios é descrita pela Equação 48.

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (48)$$

- **Camada 5:** esta camada é composta por um único neurônio de somatório e é do tipo fixo. Ele simplesmente calcula a saída total como uma soma de todos os sinais de entrada:

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (49)$$

O ANFIS pode ser treinado utilizando um algoritmo de treinamento híbrido que utiliza o método dos mínimos quadrados e o gradiente descendente, proposto por Jang (JANG; SUN; MIZUTANI, 1997). Na fase de avanço – *forward* – o algoritmo usa o método dos mínimos quadrados para identificar os parâmetros consequentes (p_i , q_i , r_i) da camada 4. Na fase de volta – *backward* – os erros são propagados e os parâmetros antecedentes (a_i , b_i) são atualizados utilizando o método do gradiente descendente. Trata-se portanto de um treinamento supervisionado, e o conjunto de dados deve alimentar o ANFIS com as entradas e as saídas desejadas.

Agrupamento *Fuzzy*

Agrupamento Subtrativo (AS)

O AS pode ser utilizado quando não se tem ideia do número de agrupamentos, o algoritmo cria interativamente novos grupos, baseados nos seus raios de influência e na densidade de dados em torno de um possível centro de grupo (CHIU, 1994). O método assume que cada dado é um centro de agrupamento em potencial e calcula uma medida de probabilidade de cada ponto definir um centro. Os passos para construção dos agrupamentos segue a seguinte sequência:

- 1 Seleciona o dado com o maior potencial para ser o primeiro centro de grupo;
- 2 Remove todos os dados na vizinhança do primeiro centro (determinado pelo raio de influência), para determinar o próximo agrupamento e a localização do seu centro;
- 3 Repete esse processo até que todos os dados estejam dentro do raio de influência de um centro de grupo.

O método AS é uma extensão da Clusterização de Montanha proposta por Yager (YAGER; FILEV, 1994).

Agrupamento *Fuzzy C-Means* (FCM)

O FCM é uma técnica de agrupamento onde cada dado pertence a um grupo com algum grau que é especificado por uma nota de associação. Essa técnica foi originalmente introduzida por Jim Bezdek (BEZDEK, 1981) como uma melhoria a métodos anteriores. O algoritmo fornece um método que mostra como agrupar dados que preenchem algum espaço multidimensional em um número específico de grupos diferentes. O FCM é baseado na minimização da função custo definida pela Equação 50, mostrada a seguir.

$$J_m = \sum_{i=1}^D \sum_{j=1}^N \mu_{ij}^m \|x_i - c_j\|^2 \quad (50)$$

Onde:

- D é o número total de dados;
- N é o número de clusters;

- m é o expoente da matriz de partição *fuzzy* que controla o nível de sobreposição fuzzy, com $m > 1$. Sobreposição *fuzzy* se refere ao quão “nebulosas” são as fronteiras entre os clusters, isso é, o número de dados que tem pertinência significativa em mais de um grupo;
- x_i é o i -ésimo dado;
- c_j é o centro do j -ésimo grupo;
- μ_{ij} é o grau de pertinência de x_i no j -ésimo grupo. Para um dado, x_i , a soma dos valores de pertinência de todos os grupos é um.

Os passos para construção dos agrupamentos segue a seguinte sequência:

- 1 Aleatoriamente inicializa os valores de pertinência aos grupos, μ_{ij} ;
- 2 Calcula os centros dos grupos pela Equação 51, a seguir:

$$c_j = \frac{\sum_{i=1}^D \mu_{ij}^m x_i}{\sum_{i=1}^D \mu_{ij}^m} \quad (51)$$

- 3 Atualiza μ_{ij} de acordo com a Equação 52, a seguir:

$$\mu_{ij} = \frac{1}{\sum_{k=1}^N \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (52)$$

- 4 Calcula a função objetivo, J_m ;
- 5 Repete os passos 2-4 até J_m melhorar menos que um limiar mínimo especificado ou até o número máximo de iterações.