

FUNDAÇÃO UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL FACULDADE DE ENGENHARIAS, ARQUITETURA E URBANISMO E GEOGRAFIA

MESTRADO EM ENGENHARIA ELÉTRICA

Algoritmo Modificado de Treinamento de Redes Neurais de Função de Base Radial com Avaliação Direta de Métricas De Desempenho de Modelos Auto-Associativos para Validação de Leitura de Sensores

Marco Aurélio Duarte Alves

Campo Grande MS 29 de janeiro de 2019



FUNDAÇÃO UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL FACULDADE DE ENGENHARIAS, ARQUITETURA E URBANISMO E GEOGRAFIA

MESTRADO EM ENGENHARIA ELÉTRICA

Algoritmo Modificado de Treinamento de Redes Neurais de Função de Base Radial com Avaliação Direta de Métricas De Desempenho de Modelos Auto-Associativos para Validação de Leitura de Sensores

Marco Aurélio Duarte Alves

Orientador: Prof. Dr. João Onofre Pereira Pinto Coorientador: Prof. Dr. Luigi Galotto Júnior

> Dissertação apresentada à Universidade Federal de Mato Grosso do Sul na Faculdade de Engenharias, Arquitetura e Urbanismo e Geografia, como requisito para a obtenção do grau de Mestre em Engenharia Elétrica.

Campo Grande MS 29 de janeiro de 2019

Algoritmo Modificado de Treinamento de Redes Neurais de Função de Base Radial com Avaliação Direta de Métricas De Desempenho de Modelos Auto-Associativos para Validação de Leitura de Sensores

Marco Aurélio Duarte Alves

'Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Engenharia Elétrica, Área de Concentração em Energia Elétrica, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Mato Grosso do Sul.'

Banca Examinadora:

Prof. Dr. João Onofre Pereira Pinto Orientador

Prof. Dr. Cristiano Quevedo Andrea

Prof. Dr. Luiz Eduardo Borges da Silva

Dr. Mário César Mello Massa de Campos

Prof. Dr. Raymundo Cordero García

Campo Grande MS 29 de janeiro de 2019

A Deus e à minha família.

AGRADECIMENTOS

Agradeço e dedico este trabalho primeiramente à Deus pela fé, força e perseverança que me mantêm prosseguindo na busca pelo conhecimento; e a minha família, especialmente minha esposa e minhas filhas, pelo apoio, paciência, tolerância e compreensão nos momentos de minha ausência.

Agradeço aos meus professores do mestrado, em especial, aos meus orientadores: professor Dr. João Onofre Pereira Pinto, não só pelas indispensáveis orientações, mas também por sempre me apoiar e incentivar; e ao professor Dr. Luigi Galotto Júnior pelas contribuições, correções e disponibilidade.

Aos professores Dr. Raymundo Cordero Garcia, Dr. Cristiano Quevedo Andrea, Dr. Luiz Eduardo Borges da Silva e ao Dr. Mário César Mello Massa de Campos por aceitarem o convite para participar da banca e pelas valiosas contribuições.

Aos colegas e professores participantes do laboratório de pesquisa BATLAB da Universidade Federal de Mato Grosso do Sul que de alguma forma ou de outra contribuíram para que eu chegasse até aqui.

RESUMO

Este trabalho apresenta o uso da rede neural artificial função de base radial para estimar as leituras de sensores, explorando a redundância analítica via auto associação. No entanto, a fim de garantir o bom desempenho da rede, o processo de treinamento e otimização foi modificado. No algoritmo de treinamento convencional, embora os critérios de parada, tal como soma dos erros quadráticos, sejam atingidos, uma ou mais métricas de desempenho individuais, incluindo: i) precisão; ii) robustez; iii) espalhamento e iv) filtragem da rede neural pode não serem satisfatórias. A modificação do algoritmo de treinamento se baseia em buscar garantir que uma ou mais métricas sejam atingidas. O trabalho descreve o algoritmo proposto incluindo toda a base matemática. Um conjunto de dados de uma bomba de injeção de água de uma unidade processamento de petróleo e gás é usado para treinar uma rede RBF usando o algoritmo convencional e o modificado para avaliar o desempenho de ambos. Além disso, o modelo AAKR é usado para o mesmo conjunto de dados. Por fim, um estudo comparativo dos modelos desenvolvidos é feito para cada uma das métricas de desempenho, bem como para a eficácia global, demonstrando a principal vantagem da abordagem proposta que consistem na obtenção de resultados de estimação equivalentes ou superiores ao AAKR com tempo de execução muito menor e a desvantagem de ter maior complexidade durante o treinamento do modelo.

Palavras-Chave: Rede Neural de Função de Base Radial, Regressão Kernel Auto-Associativa, Detecção de Falta, Robustez, Precisão, Espalhamento, Filtragem, Teste de Hipóteses.

ABSTRACT

This work presents the use of radial basis function artificial neural network to estimate the sensors readings, exploring the analytical redundancy via auto association. However, in order to guarantee good performance of the network the training and optimization process was modified. In the conventional training algorithm, although the stop criteria, such as summed squared error, is reached, one or more of the individual performance metrics, including: i) accuracy; ii) robustness; iii) spillover and iv) filtering of the neural network may not be satisfactory. The modification of the training algorithm is based on seeking to ensure that one or more metrics are met. This work describes the proposed algorithm including all the mathematical foundation. A data set of a water injection pump for an oil and gas processing unit is used to train an RBF network using the conventional and the modified algorithm and the performance of both will be evaluated. Furthermore, AAKR model is used to the same dataset. Finally, a comparison study of the developed models will be done for each of the performance metrics, as well as for the overall effectiveness, demonstrating the main advantage of the proposed approach is to obtain estimation results equivalent or superior to the AAKR with shorter runtime and the disadvantage of having greater complexity during model training.

Keywords: Radial Basis Function Neural Network, Auto-Associative Kernel Regression, Fault Detection, Robustness, Accuracy, Spillover, Filtering, Statistical Hypothesis Testing.

LISTA DE FIGURAS

Número		Página
Figura 2.1	Arquitetura de Rede Neural Função de Base Radial	
Figura 2.2	Exemplos de funções de base radial	
Figura 2.3	Função de Base Radial tipo Gaussiana	
Figura 2.4	Fluxograma do treinamento convencional da RBF	
Figura 2.5	Distribuição de referência para um teste de hipóteses	
Figura 3.1	Fluxograma de Treinamento RBF Modificado	
Figura 3.2	Diagrama físico simplificado da bomba A-T2620	
Figura 4.1	Precisão comparativa entre AAKR e RBF	
Figura 4.2	Sensibilidade (robustez) comparativa entre AAKR e RBF	
Figura 4.3	Sensibilidade (espalhamento) comparativa entre AAKR e RBF	
Figura 4.4	Filtragem comparativa entre AAKR e RBF	
Figura 4.5	Representação gráfica Robustez x Precisão do Grupo 2	
Figura 4.6	Mapeamento da Robustez x Espalhamento (Precisão) do Grupo 2	
Figura 4.7	Dados de leituras dos sensores do Grupo 2	
Figura 4.8	(a) Dados reais, medidos e estimados com <i>drift</i> . (b) Resíduos	
Figura 4.9	Dados medidos e estimados em cada modelo para o Grupo 2	40

LISTA DE TABELAS

Número		Página
Tabela 3.1	Pesos dos casos de teste considerados	
Tabela 3.2	Grupos de sensores dos dados reais.	
Tabela 4.1	Resultados médios para todos os grupos	
Tabela 4.2	Teste de Hipótese por métrica entre ARF e AF	
Tabela 4.3	Teste de Hipótese por métrica entre AF e AAKR	
Tabela 4.4	Teste de Hipótese por métrica entre ARF e AAKR	
Tabela 4.5	Teste de Hipótese por métrica entre RF e AAKR.	
Tabela 4.6	Tempo decorrido (ms) por cada modelo para estimação	44
Tabela 7.1	Matriz de pesos IW para o Grupo 1 do caso RBF AF.	
Tabela 7.2	Matriz de pesos W para o Grupo 1 do caso RBF AF	
Tabela 7.3	Vetor de <i>bias</i> w ₀ para o Grupo 1 do caso RBF AF	
Tabela 7.4	Matriz de pesos IW para o Grupo 2 do caso RBF AF.	
Tabela 7.5	Matriz de pesos W para o Grupo 2 do caso RBF AF	
Tabela 7.6	Vetor de <i>bias</i> w ₀ para o Grupo 2 do caso RBF AF	
Tabela 7.7	Matriz de pesos IW para o Grupo 3 do caso RBF AF.	
Tabela 7.8	Matriz de pesos W para o Grupo 3 do caso RBF AF	
Tabela 7.9	Vetor de <i>bias</i> w ₀ para o Grupo 3 do caso RBF AF	
Tabela 7.10	Matriz de pesos IW para o Grupo 4 do caso RBF AF.	
Tabela 7.11	Matriz de pesos W para o Grupo 4 do caso RBF AF	
Tabela 7.12	Vetor de <i>bias</i> w ₀ para o Grupo 4 do caso RBF AF	55
Tabela 7.13	Matriz de pesos IW para o Grupo 5 do caso RBF AF.	55
Tabela 7.14	Matriz de pesos W para o Grupo 5 do caso RBF AF	55
Tabela 7.15	Vetor de <i>bias</i> w_0 para o Grupo 5 do caso RBF AF	
Tabela 7.16	Matriz de pesos IW para o Grupo 6 do caso RBF AF.	
Tabela 7.17	Matriz de pesos W para o Grupo 6 do caso RBF AF	
Tabela 7.18	Vetor de <i>bias</i> w ₀ para o Grupo 6 do caso RBF AF	
Tabela 8.1	Matriz de pesos IW para o Grupo 1 do caso RBF RF	
Tabela 8.2	Matriz de pesos W para o Grupo 1 do caso RBF RF.	
Tabela 8.3	Vetor de <i>bias</i> w ₀ para o Grupo 1 do caso RBF RF	
Tabela 8.4	Matriz de pesos IW para o Grupo 2 do caso RBF RF	

Tabela 8.5	Matriz de pesos W para o Grupo 2 do caso RBF RF.	58
Tabela 8.6	Vetor de <i>bias</i> w ₀ para o Grupo 2 do caso RBF RF	58
Tabela 8.7	Matriz de pesos IW para o Grupo 3 do caso RBF RF	59
Tabela 8.8	Matriz de pesos W para o Grupo 3 do caso RBF RF.	59
Tabela 8.9	Vetor de <i>bias</i> w ₀ para o Grupo 3 do caso RBF RF	59
Tabela 8.10	Matriz de pesos IW para o Grupo 4 do caso RBF RF	59
Tabela 8.11	Matriz de pesos W para o Grupo 4 do caso RBF RF.	60
Tabela 8.12	Vetor de <i>bias</i> w ₀ para o Grupo 4 do caso RBF RF	60
Tabela 8.13	Matriz de pesos IW para o Grupo 5 do caso RBF RF	60
Tabela 8.14	Matriz de pesos W para o Grupo 5 do caso RBF RF.	60
Tabela 8.15	Vetor de <i>bias</i> w ₀ para o Grupo 5 do caso RBF RF	61
Tabela 8.16	Matriz de pesos IW para o Grupo 6 do caso RBF RF	61
Tabela 8.17	Matriz de pesos W para o Grupo 6 do caso RBF RF	61
Tabela 8.18	Vetor de <i>bias</i> w ₀ para o Grupo 6 do caso RBF RF	61
Tabela 9.1	Matriz de pesos IW para o Grupo 1 do caso RBF ARF	62
Tabela 9.2	Matriz de pesos W para o Grupo 1 do caso RBF ARF.	62
Tabela 9.3	Vetor de <i>bias</i> w ₀ para o Grupo 1 do caso RBF ARF	62
Tabela 9.4	Matriz de pesos IW para o Grupo 2 do caso RBF ARF	63
Tabela 9.5	Matriz de pesos W para o Grupo 2 do caso RBF ARF.	63
Tabela 9.6	Vetor de <i>bias</i> w ₀ para o Grupo 2 do caso RBF ARF	63
Tabela 9.7	Matriz de pesos IW para o Grupo 3 do caso RBF ARF	64
Tabela 9.8	Matriz de pesos W para o Grupo 3 do caso RBF ARF.	64
Tabela 9.9	Vetor de <i>bias</i> w ₀ para o Grupo 3 do caso RBF ARF	64
Tabela 9.10	Matriz de pesos IW para o Grupo 4 do caso RBF ARF	65
Tabela 9.11	Matriz de pesos W para o Grupo 4 do caso RBF ARF.	65
Tabela 9.12	Vetor de <i>bias</i> w ₀ para o Grupo 4 do caso RBF ARF	65
Tabela 9.13	Matriz de pesos IW para o Grupo 5 do caso RBF ARF	65
Tabela 9.14	Matriz de pesos W para o Grupo 5 do caso RBF ARF.	66
Tabela 9.15	Vetor de <i>bias</i> w ₀ para o Grupo 5 do caso RBF ARF	66
Tabela 9.16	Matriz de pesos IW para o Grupo 6 do caso RBF ARF	66
Tabela 9.17	Matriz de pesos W para o Grupo 6 do caso RBF ARF.	67
Tabela 9.18	Vetor de <i>bias</i> w ₀ para o Grupo 6 do caso RBF ARF	67
Tabela 10.1	Matriz de pesos IW para o Grupo 1 do caso RBF ARSFF	68
Tabela 10.2	Matriz de pesos W para o Grupo 1 do caso RBF ARSFF	68

Tabela 10.3	Vetor de <i>bias</i> w ₀ para o Grupo 1 do caso RBF ARSFF
Tabela 10.4	Matriz de pesos IW para o Grupo 2 do caso RBF ARSFF
Tabela 10.5	Matriz de pesos W para o Grupo 2 do caso RBF ARSFF
Tabela 10.6	Vetor de <i>bias</i> w ₀ para o Grupo 2 do caso RBF ARSFF
Tabela 10.7	Matriz de pesos IW para o Grupo 3 do caso RBF ARSFF70
Tabela 10.8	Matriz de pesos W para o Grupo 3 do caso RBF ARSFF70
Tabela 10.9	Vetor de <i>bias</i> w_0 para o Grupo 3 do caso RBF ARSFF
Tabela 10.10	Matriz de pesos IW para o Grupo 4 do caso RBF ARSFF71
Tabela 10.11	Matriz de pesos W para o Grupo 4 do caso RBF ARSFF
Tabela 10.12	Vetor de bias w_0 para o Grupo 4 do caso RBF ARSFF
Tabela 10.13	Matriz de pesos IW para o Grupo 5 do caso RBF ARSFF71
Tabela 10.14	Matriz de pesos W para o Grupo 5 do caso RBF ARSFF
Tabela 10.15	Vetor de bias w_0 para o Grupo 5 do caso RBF ARSFF
Tabela 10.16	Matriz de pesos IW para o Grupo 6 do caso RBF ARSFF72
Tabela 10.17	Matriz de pesos W para o Grupo 6 do caso RBF ARSFF
Tabela 10.18	Vetor de <i>bias</i> w_0 para o Grupo 6 do caso RBF ARSFF

SUMÁRIO

1.	INTRODUÇÃO	14
1.1.	CONTEXTUALIZAÇÃO DO PROBLEMA	14
1.2.	REVISÃO BIBLIOGRÁFICA (ESTADO DA ARTE)	14
1.3.	MOTIVAÇÃO E JUSTIFICATIVA DO TRABALHO	16
1.4.	OBJETIVO GERAL	16
1.5.	ORGANIZAÇÃO DO TRABALHO	16
2.	FUNDAMENTAÇÃO TEÓRICA	18
2.1.	REDES NEURAIS DE FUNÇÃO DE BASE RADIAL	18
2.2.	MÉTRICAS DE DESEMPENHO DE MODELOS AUTO-ASSOCIATIVOS	23
2.2.1.	Erro Quadrático Médio	23
2.2.2.	Sensibilidade	23
2.2.3.	Filtragem	24
2.3.	TESTE DE HIPÓTESES	24
3.	MÉTODO DE TREINAMENTO PARA RBF COMO MODELO	Ο
AUT	O-ASSOCIATIVO	26
3.1.	FUNÇÃO CUSTO PROPOSTA	26
3.2.	CASOS DE TESTE	27
3.3.	A METODOLOGIA DE TREINAMENTO	28
3.4.	TESTE DE HIPÓTESES PARA DUAS AMOSTRAS COM VARIÂNCIAS	
DESIG	GUAIS	30
3.5.	DADOS REAIS PARA VALIDAÇÃO DO MODELO	30
4.	RESULTADOS	33
4.1.	RESULTADOS DE MÉTRICAS	33
4.2.	ANÁLISE TEMPORAL	37
4.3.	RESULTADOS DOS TESTES DE HIPÓTESES	41
4.3.1.	Entre RBF ARF e RBF AF	41
4.3.2.	Entre RBF AF e AAKR	42
4.3.3.	Entre RBF ARF e AAKR	42
4.3.4.	Entre RBF RF e AAKR	43

4.4.	COMPLEXIDADE COMPUTACIONAL	44
5.	CONCLUSÕES E SUGESTÕES DE TRABALHOS FUTUROS	. 45
6.	REFERÊNCIAS	. 49
7.	APÊNDICE A – PARÂMETROS DAS REDES NEURAIS DO	
CAS	SO RBF AF	. 51
8.	APÊNDICE B – PARÂMETROS DAS REDES NEURAIS DO	
CAS	SO RBF RF	. 57
9.	APÊNDICE C – PARÂMETROS DAS REDES NEURAIS DO	
CAS	SO RBF ARF	. 62
10.	APÊNDICE D – PARÂMETROS DAS REDES NEURAIS DO	
CAS	SO RBF ARSFF	. 68

1. INTRODUÇÃO

1.1. CONTEXTUALIZAÇÃO DO PROBLEMA

Em qualquer planta ou processo químico moderno centenas ou até milhares de variáveis - tais como vazão, temperatura, pressão, níveis ou suas combinações - são rotineiramente medidas e registradas automaticamente para fins de controle de processos, otimização *on-line*, ou avaliação econômica do processo. Computadores modernos e sistemas de aquisição de dados facilitam a coleta e o processamento de um grande volume de dados, frequentemente amostrados com uma frequência da ordem de minutos ou até milissegundos.

O uso de computadores não só permite que os dados sejam obtidos com maior frequência, mas também resulta na eliminação de erros presentes no registro manual. Isso por si só melhorou muito a precisão e a validação dos dados do processo. O aumento da quantidade de informações, no entanto, pode ser explorado para melhorar ainda mais a precisão e a consistência dos dados do processo por meio de uma verificação e tratamento sistemáticos dos dados, visto que medições de processos são inevitavelmente corrompidas por erros durante a medição, processamento e transmissão do sinal medido [1].

1.2. REVISÃO BIBLIOGRÁFICA (ESTADO DA ARTE)

Diversos trabalhos foram realizados nas últimas décadas para melhorar a confiabilidade dos processos industriais [2], [3], em controladores [4], atuadores [5] e sensores [6], [7]. Normalmente, todos os equipamentos podem ser analisados usando as medições dos processos, mas a detecção de faltas em sensores mostrou ter uma complexidade extra, porque estes são os provedores de medição. A validação de sensores é um assunto importante em sistemas de monitoramento e controle baseado em sensores. Na indústria de petróleo e gás natural, como em muitas outras plantas químicas modernas, medições confiáveis são fundamentais para garantir o ponto de operação ideal e seguro da planta e, em alguns casos, faturamento confiável.

A chamada redundância física é a encontrada em sistemas tradicionais de monitoramento e diagnóstico de faltas, na qual mais de um sensor realiza uma mesma medida. Apesar de que esta técnica seja ainda utilizada por apresentar resultados confiáveis, ela apresenta algumas desvantagens, tais como custo e disponibilidade de espaço para instalação de sensores, além de ser mais um equipamento a ser incluído no programa de manutenção [8]. A fim de não implicar em incremento de componentes à planta, além de reduzir os custos, utiliza-se a chamada redundância analítica, que utiliza um modelo matemático para estimar as leituras dos sensores.

Por isso, muitos anos de projetos de pesquisa e desenvolvimento foram realizados com o objetivo de aprimorar as ferramentas de diagnóstico, especialmente para a indústria de petróleo e gás [9], bem como para os mais diversos tipos de indústrias. Muitas técnicas que utilizam redundância analítica em medições foram amplamente testadas e aprovadas [10]. A Regressão Kernel Auto-Associativa (AAKR – *Auto-Associative Kernel Regression*) mostrou ter a melhor generalização em muitos problemas [11] e também é fácil de treinar ou adaptarse a novos dados, uma vez que é uma técnica não paramétrica baseada em memória. No entanto, a principal desvantagem do AAKR provavelmente é o alto esforço computacional, em tempo real, necessário para avaliar cada estimativa.

A Rede Neural Função de Base Radial (RBF) [12] é uma das técnicas possíveis que poderiam ser aplicadas nas mesmas aplicações de AAKR. Mas a melhoria do algoritmo de treinamento é uma questão fundamental para permitir que os modelos RBF alcancem métricas de desempenho similares dos modelos AAKR. Isso porque o objetivo principal do modelo auto-associativo não é uma estimativa precisa, mas a possibilidade de estimar corretamente mesmo em condições de falta devido à redundância analítica explorada pelo princípio da auto-associação.

Este trabalho apresenta o treinamento de modelos de RBF usando métricas de desempenho como metas, ao invés de apenas erros. A abordagem proposta será comparada com o treinamento tradicional e com um modelo AAKR como referência de qualidade para a aplicação da detecção de faltas em sensores.

1.3. MOTIVAÇÃO E JUSTIFICATIVA DO TRABALHO

A principal motivação deste estudo é possibilitar o uso técnicas de baixo esforço computacional, em tempo real, neste caso Rede Neural Artificial de Função de Base Radial, para a implementação de modelo baseado em dados para monitoramento e diagnóstico de sensores e equipamentos com desempenho equivalente aos modelos não paramétricos baseados em memória.

Com melhor capacidade de generalização em diversos problemas, AAKR, tem o inconveniente em demandar grande esforço computacional em tempo real. Em contraponto à AAKR, a RBF possui menor esforço computacional, porém, ambas constroem aproximações locais para mapeamentos de entrada-saída não-lineares.

1.4. OBJETIVO GERAL

Desenvolver e avaliar um algoritmo de treinamento para uma Rede Neural Artificial de Função de Base Radial que garanta melhores desempenhos em auto-associação usando métricas de desempenho.

1.5. ORGANIZAÇÃO DO TRABALHO

No Capítulo 1, discutiu-se o desenvolvimento de um algoritmo modificado de treinamento de redes neurais de função de base radial com avaliação direta de métricas de desempenho de modelos auto-associativos para validação de leitura de sensores. No Capítulo 2 tem-se a fundamentação teórica do trabalho onde será discutido inicialmente sobre redes neurais artificiais de função de base radial. Em seguida, explana-se sobre as métricas de desempenho de modelos auto-associativos utilizadas com a devida base matemática. Por fim, uma breve apresentação dos principais pontos de um teste de hipóteses. Já no Capítulo 3 será apresentada a metodologia de treinamento proposta com a utilização das métricas de desempenho, bem como as características do teste de hipóteses utilizado. O Capítulo 4 contém os resultados obtidos com dados reais de uma planta industrial, apresentando o comparativo das métricas de desempenho para cada caso, a análise temporal dos resultados e dos testes de

hipóteses. Finalmente um breve comparativo do esforço computacional do modelo utilizando RBF e AAKR. No Capítulo 5, as conclusões e sugestões de trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a fundamentação teórica das redes neurais de função de base radial, as métricas de desempenho de modelos auto-associativos e teste de hipóteses, haja vista que neste trabalho a rede neural de função de base radial será utilizada como modelo auto-associativo com a utilização das métricas de desempenho cujos resultados serão comparados e avaliados através da utilização de teste de hipóteses.

2.1. REDES NEURAIS DE FUNÇÃO DE BASE RADIAL

As redes neurais função de base radial (RBF) de configuração mais básica, envolvem três camadas com papéis totalmente diferentes, sendo camada de entrada, uma intermediária ou oculta e a camada de saída, conforme ilustração da Figura 2.1.



Figura 2.1 Arquitetura de Rede Neural Função de Base Radial

Fonte: Próprio autor

O aprendizado de uma rede RBF é equivalente a ajustar uma superfície não-linear ao conjunto de dados, em um espaço multidimensional, de acordo com algum critério estatístico. O processo de generalização equivale a usar esta superfície multidimensional para interpolar outros pontos que não pertençam ao conjunto de treinamento, mas estejam em sua vizinhança.

A camada de entrada representa o espaço de entrada conectando a rede ao seu ambiente. A camada intermediária aplica uma transformação não linear do espaço de entrada, de modo a obter um espaço oculto com alta dimensionalidade, através de funções de base radial, como por exemplo, a gaussiana, a multi-quadrática, a multi-quadrática inversa ou a thin-plate spline, conforme ilustra a Figura 2.2. A camada de saída aplica uma transformação linear do espaço oculto para gerar as saídas da rede ao padrão (sinal) de ativação aplicado à camada de entrada [13].



Figura 2.2 Exemplos de funções de base radial

Fonte: Próprio autor

A função de base radial do tipo Gaussiana, que é definida conforme equação (1), é a mais utilizada em aplicações práticas, e é a escolhida para ser empregada neste trabalho. O parâmetro σ corresponde ao desvio-padrão da função gaussiana, de forma que, ele define a distância euclidiana média (largura da função radial, *spread*) que mede o espalhamento dos dados representados pela função de base radial em torno do seu centro.

$$\varphi(v) = e^{\left(-\frac{v^2 \cdot \ln(2)}{\sigma^2}\right)} \tag{1}$$

Em que:

- *v* é entrada da função que virá de uma norma euclidiana.
- σ é a largura da função radial.

A Figura 2.3 apresenta uma função de base radial do tipo Gaussiana centrada em 0 (zero) e com largura (σ ou *spread*) igual a 1 (um).



Fonte: Próprio autor

Através da equação (2) temos que a resposta da função de ativação de uma rede neural RBF a um vetor de entrada x depende da norma (distância) euclidiana [13] [14] entre x e centro da função de Base Radial associada ponderada pela linha j da matriz IW. A saída da função de ativação vai de 0 (para entradas distantes do centro) para 1 (para entradas próximas ao centro).

$$a(\mathbf{x})_j = \varphi(\|\mathbf{I}\mathbf{W}_j - \mathbf{x}\|)$$
(2)

Em que:

- $a(\mathbf{x})_i$ é a *j*-ésima saída da camada oculta.
- *IW_j* é o vetor de pesos de cada entrada para o neurônio *j*.

A equação (3) apresenta a *k*-ésima saída da rede neural de função de Base Radial, denotada por $y(\mathbf{x})_k$, w_{0k} é o termo de polarização ou *bias* do neurônio de saída *j* e w_{kj} é o peso sináptico entre a saída *j* e o neurônio *k*.

$$y(x)_{k} = \sum_{j=1}^{m} w_{kj} \cdot a(x)_{j} + w_{0k}$$
(3)

A Figura 2.4 ilustra o fluxograma do treinamento convencional da RBF, no qual o usuário entra com os parâmetros da sua arquitetura, como número máximo de neurônios e *spread*, de seu desempenho, sendo o erro quadrático médio (MSE – *Mean Square Error*) desejado, além dos dados de treinamento. O algoritmo convencional cria iterativamente uma rede, inicialmente com 1 (um) neurônio na camada oculta, calcula a saída da camada oculta, MSE, os pesos e *bias* das camadas ocultas e de saída, caso o usuário escolher 1 neurônio como parâmetro da RBF encerra-se o treinamento, caso contrário, neurônios são adicionados à rede sempre um por vez, recalculando-se, em cada iteração, o MSE, os pesos e *bias* das camadas ocultas e de saída, o u seja, o algoritmo convencional busca a minimização do erro quadrático médio.



Figura 2.4 Fluxograma do treinamento convencional da RBF

Fonte: Próprio autor

Muitos métodos foram propostos para o treinamento da RBF, sendo todos eles buscando a minimização do erro de estimação. A maioria deles são classificados como híbridos por serem compostos em duas partes: A primeira fase, o número de RBF e seus parâmetros são estabelecidos por métodos não supervisionados. Na segunda fase de treinamento, os pesos dos neurônios na camada de saída são ajustados. Devido às saídas dos neurônios na camada intermediária serem um vetor linearmente separável, os pesos podem ser determinados através de métodos lineares supervisionados, de rápida convergência [15].

2.2. MÉTRICAS DE DESEMPENHO DE MODELOS AUTO-ASSOCIATIVOS

A qualidade de cada modelo auto-associativo pode ser definida usando medidas de performance, entre elas destacam-se: erro quadrático médio, sensibilidade e filtragem. Elas podem mostrar se o modelo pode melhorar a resposta do sistema em condições normais de operação e se o modelo pode melhorar a resposta em condições de falta.

2.2.1. Erro Quadrático Médio

O erro quadrático médio na equação (4), compara a saída estimada com a saída real sem ruído. Em (4), n é o número de observações, \hat{y}_{kj} é a k-ésima estimação do sinal j e y_{kj} é a k-ésima observação.

$$Acc_{j} = \frac{1}{n} \sum_{k=1}^{n} \left(\hat{y}_{kj} - y_{kj} \right)^{2}$$
(4)

Essa métrica é bem conhecida e é útil para determinar a precisão do modelo. Esta precisão é fundamental para garantir que o sistema de controle mantenha a mesma resposta usando as estimativas do modelo como um sinal de realimentação no laço de controle.

2.2.2. Sensibilidade

A sensibilidade é uma métrica para definir a melhoria da resposta do modelo na presença de faltas. Considere que x_{ki}^{drift} é a *k*-ésima observação de x_i com drift gerado artificialmente e \hat{y}_{kj}^{drift} é a *k*-ésima observação estimada de y_j para as mesmas entradas com *drift*. Este *drift* pode ser aditivo ou multiplicativo alterando o deslocamento ou a amplitude do sinal original. Assim, o denominador da equação (5) representa um *drift* simulado na entrada *i* e o numerador é a propagação da falta para a estimativa do sinal *j*.

$$S_{ij} = \frac{\sum_{k=1}^{n} |\hat{y}_{kj}^{drift} - \hat{y}_{kj}|}{\sum_{k=1}^{n} |x_{ki}^{drift} - x_{ki}|}$$
(5)

Onde, $S_{ij} = Robustez$, para: $i = j e S_{ij} = Espalhamento$; para: $i \neq j$. Observe que o resultado é uma matriz que mostra o efeito na saída *j* devido a uma falta na entrada *i*.

A sensibilidade ideal deve ser zero para que os efeitos da falta possam ser completamente eliminados. Quando i = j é chamada de robustez e, caso contrário, é chamada de espalhamento.

Vale ressaltar que, de acordo com o modelo a ser avaliado, a sensibilidade pode estar relacionada em função do desvio simulado na entrada, de forma que, desvios de grande magnitude podem ser mais atenuados do que os pequenos. Portanto, um erro em forma de rampa pode ser adicionado como desvio nas entradas para melhor representar a sensibilidade.

2.2.3. Filtragem

Como os modelos têm uma propriedade para filtrar o ruído, uma equação semelhante à da sensibilidade pode ser usada, mas um ruído branco (wn) é usado em vez de um drift, como mostrado em (6).

$$S_{ij} = \frac{\sum_{k=1}^{n} |\hat{y}_{kj}^{wn} - \hat{y}_{kj}|}{\sum_{k=1}^{n} |x_{ki}^{wm} - x_{ki}|}$$
(6)

2.3. TESTE DE HIPÓTESES

Através da teoria da probabilidade, pode-se inferir sobre quantidades de interesse de uma população a partir de uma amostra observada de um experimento científico. Alguns exemplos são: testar se uma determinada teoria científica deve ser descartada, verificar se um lote de lâmpadas deve ser devolvido por falta de qualidade, entre outros. Por intermédio do rigor matemático, a inferência estatística pode ser utilizada para auxiliar a tomada de decisões nas mais variadas áreas.

Chama-se de Hipótese Estatística qualquer afirmação feita a respeito da função de probabilidade de uma ou mais variáveis aleatórias ou a respeito da população da qual a amostra em estudo foi extraída e o procedimento de tomada de decisão sobre a hipótese é chamado de Teste de Hipóteses (TH).

A Hipótese Nula, denotada ao longo deste trabalho por H_0 , é a hipótese estatística a ser testada. Quando a Hipótese Nula é rejeitada temos a conclusão do teste, denotada por H_1 e chamada de Hipótese Alternativa.

O teste de hipóteses para decidir se H_0 será rejeitada é construído a partir de uma função T = T(X) de uma amostra aleatória X = (X1, X2, ...Xn). Essa função T(X) é chamada estatística de teste e está associada à distribuição de probabilidade do estimador do parâmetro que se deseja testar.

A Região Crítica, do teste de hipóteses, denotada por R, é formada pelo conjunto de valores, determinados com base na estatística de teste, que levam à rejeição de H_0 e Região de Aceitação é aquela cujo conjunto de valores não levam a rejeição da Hipótese Nula. O valor limite que delimita as regiões de aceitação e de rejeição para o teste estatístico é chamado de Valor Crítico do teste.

A probabilidade de se rejeitar H_0 , quando H_0 é verdadeira, é chamada de nível de significância do teste.

A Figura 2.5 apresenta um exemplo de distribuição de referência para um teste de hipóteses unilateral à esquerda, ou também chamado de unicaudal à esquerda. Nessa figura observa-se o valor crítico e as regiões crítica e de aceitação.



Figura 2.5 Distribuição de referência para um teste de hipóteses

Fonte: Próprio autor

Maiores detalhes sobre o teste de hipóteses utilizado neste trabalho estão descritos em 3.4.

3. MÉTODO DE TREINAMENTO PARA RBF COMO MODELO AUTO-ASSOCIATIVO

Uma técnica de validação cruzada é frequentemente usada para otimizar o modelo da RBF com base no erro de validação. Essa é uma boa prática para minimizar a incerteza do modelo, evitando soluções com sobreajuste ou tendenciosas. No entanto, a proposta é desenvolver um algoritmo que permita considerar também as outras medidas de performance dos modelos auto-associativos como a sensibilidade e a filtragem.

Diferentemente do algoritmo convencional de treinamento da RBF, que minimiza o erro, este algoritmo, o qual é proposto para a RBF sendo utilizada como modelo autoassociativo, tem como objetivo minimizar uma função custo, que é composta pela soma ponderada das métricas de desempenho de modelos auto-associativos, ou seja, precisão, robustez, espalhamento e filtragem.

3.1. FUNÇÃO CUSTO PROPOSTA

O algoritmo proposto tem como objetivo treinar a RBF quando a mesma for utilizada como modelo auto-associativo. Assim, a base do algoritmo proposto visa ter controle sobre as métricas de desempenho durante o processo de treinamento. Para tal, em vez de utilizar apenas a soma do erro quadrático como a função custo a ser minimizada durante o treinamento, propõe-se utilizar uma função custo a qual é função de tais métricas. A Equação (7) mostra a função custo (*L*) proposta, a qual possui 4 parâmetros de projeto (α , β , $\gamma \in \delta$), que definem os pesos para cada métrica de desempenho Precisão (*Acc*), Robustez (*Rob*), Espalhamento (*Spill*) e filtragem (*Filt*), respectivamente.

$$L = \alpha.Acc + \beta.Rob + \gamma.Spill + \delta.Filt$$
(7)

onde: { $(\alpha, \beta, \gamma \in \delta) \in \mathbb{R} | (\alpha, \beta, \gamma \in \delta) \ge 0$ } e $(\alpha + \beta + \gamma + \delta = 1)$, de forma que esta função custo terá sempre valores pertencente ao conjunto dos números reais não negativos e a soma dos 4 parâmetros de projeto serão iguais a 1.

A utilização da função custo com os devidos pesos para cada métrica, no algoritmo de treinamento da RBF, permitirá que o projetista tenha a liberdade de escolher qual (is) métrica (s) de desempenho o modelo auto-associativo priorizará. Por exemplo, se desejado, o modelo poderá simplesmente focar na filtragem, ignorando as demais métricas, caso os pesos dos parâmetros sejam iguais a $\alpha = 0$, $\beta = 0$, $\gamma = 0$ e $\delta = 1$.

3.2. CASOS DE TESTE

Como os parâmetros α , β , γ e δ são valores contínuos, no intervalo de 0 (zero) a 1 (um), há um número infinito de possibilidades de escolha desses parâmetros de projeto (α , β , γ e δ) da função custo *L*. No entanto, para mostrar a efetividade do algoritmo proposto foi suficiente limitar em 4 (quatro) casos a função custo *L*, ou seja, a quatro conjunto de parâmetros. Tais casos foram nomeados de acordo com a métrica priorizada, como segue:

• AF (*Accuracy Focused*): Focado na Precisão (método de treinamento convencional da RBF);

- RF (Robustness Focused): Focado na Robustez;
- ARF (Accuracy and Robustness Focused): Focado na Precisão e Robustez;

• **ARSFF** (*Accuracy, Robustness, Spillover and Filtering equally Focused*): Precisão, Robustez, Espalhamento e Filtragem igualmente focada;

Na Tabela 3.1, são apresentados os pesos de cada caso de teste considerando as métricas de desempenho.

Caso	α alfa	β beta	γ Gama	δ Delta
AF	1	0	0	0
RF	0	1	0	0
ARF	0,5	0,5	0	0
ARSFF	0,25	0,25	0,25	0,25

Tabela 3.1Pesos dos casos de teste considerados.

Fonte: Próprio autor

3.3. A METODOLOGIA DE TREINAMENTO

Neste trabalho o algoritmo de treinamento convencional da RBF foi modificado para minimizar a função custo L, ao invés do erro quadrático médio (MSE). Para obter a função custo com melhor desempenho, foi necessário variar os valores de *spread* (entre 0,1 e 100) e limitar o número máximo de neurônios da camada oculta em (N - 1), sendo N o número de sensores do grupo.

A Figura 3.1 representa o fluxograma deste algoritmo modificado de treinamento da RBF, no qual os blocos na cor vermelha estão enfatizando o que foi modificado em relação ao algoritmo convencional de treinamento da RBF.

A execução do algoritmo modificado se dá, para cada grupo de sensores, inicialmente, quando o usuário entra com os valores dos 4 parâmetros de projeto (α , β , $\gamma \in \delta$), além dos dados de treinamento e teste. Posteriormente, o algoritmo modificado cria iterativamente uma rede, com *spread* e número de neurônios na camada oculta iguais a 0,1 e 1, respectivamente, e:

- a) com os dados de treinamento, calcula a (s) saída (s) da camada oculta, atribui o valor de 1 para a variável (i) definida como sendo o número de neurônios desta camada e calcula os pesos e *bias*. Com os dados de teste, calcula as métricas de desempenho e a função custo *L*. Se o valor da função custo da iteração atual for menor que o valor na iteração anterior armazena a RBF treinada nesta iteração. Incrementa 1 na variável (i);
- b) Enquanto o valor da variável (*i*) for menor ou igual que o número máximo de neurônios da camada oculta (N − 1), com os dados de treinamento, calcula os pesos e bias. Com os dados de teste, calcula as métricas de desempenho e a função custo *L*. Se o valor da função custo da iteração atual for menor que o valor na iteração anterior armazena a RBF treinada nesta iteração. Incrementa 1 na variável (*i*); quando o valor da variável (*i*) for maior que o número máximo de neurônios da camada oculta (N − 1), então, acrescenta 0,1 no valor do *spread*;

c) Enquanto o *spread* for menor ou igual a 100 repete-se os itens a) e b), caso contrário, encerra-se o treinamento, ficando armazenada a RBF com a função custo *L* minimizada.





Fonte: Próprio autor

3.4. TESTE DE HIPÓTESES PARA DUAS AMOSTRAS COM VARIÂNCIAS DESIGUAIS

Nesta dissertação, para testar a diferença entre as médias populacionais $(\mu_1 - \mu_2)$ das métricas obtidas a partir da resposta da RBF treinada para cada grupo de sensores da bomba, considerando-as como distribuições normais com as variâncias $\sigma_1^2 e \sigma_2^2$ desconhecidas, usaremos a estatística de teste t não combinado para duas amostras com variâncias desiguais (Teste t de Welch) para o teste de hipóteses, conforme equação (8).

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - d_0}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$
(8)

Em que:

- $\bar{x}_1 \in \bar{x}_2$ são as médias amostrais das amostras 1 e 2, respectivamente.
- d_0 é a diferença da média populacional hipotética.
- $S_1^2 \in S_2^2$ são as variâncias amostrais das amostras 1 e 2, respectivamente.
- $n_1 e n_2$ são os tamanhos das amostras 1 e 2, respectivamente.

Em todos os casos que serão apresentados testes de hipóteses com o nível de significância de 5%.

3.5. DADOS REAIS PARA VALIDAÇÃO DO MODELO

Foram utilizados dados de 35 sensores, contendo 65.412 medidas com taxa de amostragem de 1 minuto, de uma das quatro bombas de injeção de água de uma unidade processamento de petróleo e gás da Petrobras, sendo que esta unidade recebe gás e condensado de diversas plataformas offshore.

A Figura 3.2 ilustra o diagrama físico simplificado da bomba A-T2620 cujos dados serão utilizados.



Figura 3.2 Diagrama físico simplificado da bomba A-T2620.

Fonte: Adaptado de [16].

Os dados recebidos foram processados e agrupados de acordo com a matriz dos coeficientes de correlação linear e sua função no processo. O processamento de dados e o agrupamento de sensores não fazem parte desse trabalho, portanto, eles não serão detalhados aqui.

As informações de cada sensor, como o número do grupo a que pertence, descrição, Tag e unidade de medida, são apresentadas na Tabela 3.2.

N° Grupo	Descrição	Tag	Uni.
	A-T2620 - Pressão sucção água de injeção	301072_T26_PSI_206	barg
	P-T2620 - Número de partidas da bomba	301072_T26_PT2620_N_PARTIDAS	NI
1	P-T2621 -Temperatura da Carcaça	301072_T26_TI_217	°C
	P-T2620 - Temperatura óleo lubrificante	301072_T26_TI_218	°C
	A-T2620 - Vazão água de injeção	301072_T26_FIC_201	m³/h
	A-T2620 - Vazão água de injeção	301072_T26_FSI_202	m³/h
	P-T2620 - Rotação bomba de injeção	301072_T26_KE_211	rpm
	P-T2621 - Potência hidráulica bomba de injeção de água	301072_T26_PH_PT2621_CALC	kW
2	P-T2623 - Pressão descarga	301072_T26_PI_201	barg
	A-T2620 - Pressão descarga água de injeção	301072_T26_PI_204	barg
	P-T2620 - Head - BOMBA	301072_T26_PT2620_HEAD_CALC	m
	P-T2621 - Deslocamento axial (1)	301072_T26_ZI_201	mm
	P-T2621 - Deslocamento axial (2)	301072_T26_ZI_202	mm
	P-T2621 – Temp. Mancal Axial do Lado Ativo	301072_T26_TI_201	°C
	P-T2621 – Temp. Mancal Axial do Lado Inativo	301072_T26_TI_202	°C
2	P-T2621 – Temp. Mancal Radial Lado Não Acoplado	301072_T26_TI_203	°C
3	P-T2621 – Temp. Mancal Radial do Lado Acoplado	301072_T26_TI_204	°C
	PM-T2621 - Temp. Mancal Radial do Lado Acoplado	301072_T26_TI_205	°C
	PM-T2621 – Temp. Mancal Radial Lado Não Acoplado	301072_T26_TI_206	°C
	PM-T2621 - Temperatura do Enrolamento Fase U	301072_T26_TI_207	°C
	PM-T2621 - Temperatura do Enrolamento Fase V	301072_T26_TI_208	°C
4	PM-T2621 - Temperatura do Enrolamento Fase W	301072_T26_TI_209	°C
	T26 - Temperatura de Entrada de Água de Refrigeração	301072_T26_TI_215	°C
	T26 - Temperatura de Saída de Água de Refrigeração	301072_T26_TI_216	°C
	P-T2621 – Vibr. Mancal Radial Lado Não Acoplado (X)	301072_T26_VIX_204	μm
	P-T2621 - Vibração Mancal Radial Lado Acoplado (X)	301072_T26_VIX_206	μm
	PM-T2621 - Vibração Mancal Radial Lado Acoplado (X)	301072_T26_VIX_208	μm
F	PM-T2621 – Vibr. Mancal Radial Lado Não Acoplado (X)	301072_T26_VIX_210	μm
5	P-T2621 – Vibr. Mancal Radial Lado Não Acoplado (Y)	301072_T26_VIY_203	μm
	P-T2621 - Vibração Mancal Radial Lado Acoplado (Y)	301072_T26_VIY_205	μm
	PM-T2621 - Vibração Mancal Radial Lado Acoplado (Y)	301072_T26_VIY_207	μm
	PM-T2621 – Vibr. Mancal Radial Lado Não Acoplado (Y)	301072_T26_VIY_209	μm
	P-T2620 - DP filtro óleo lubrificante	301072_T26_PDI_204	bar
6	P-T2620 - Pressão óleo lubrificante	301072_T26_PI_200	barg
	P-T2620 - Pressão óleo lubrificante	301072_T26_PI_202	barg

Tabela 3.2Grupos de sensores dos dados reais.

Fonte: Próprio autor

4. RESULTADOS

Os modelos foram treinados para os seis grupos pré-definidos na Tabela 3.2, utilizando todos os casos de treinamento considerados na Tabela 3.1. Além disso, um modelo AAKR otimizado é apresentado como uma referência para comparação das métricas de desempenho e resultados de estimativa.

São apresentados os resultados comparando as métricas de performance calculadas para cada modelo, estimação no domínio do tempo em condições normais e de falta, os testes de hipóteses entre os casos RBF AF e RBF ARF e entre RBF ARF e AAKR, ao final apresentado o tempo de processamento requerido por cada modelo.

4.1. RESULTADOS DE MÉTRICAS

A Figura 4.1 mostra a precisão alcançada para todos os grupos com os quatro casos de treinamento definidos pela Tabela 3.1, além do AAKR. Nesta figura a barra em vermelho atrás de todos os 4 casos de cada grupo representa a precisão de AAKR, essa mesma representação ocorre nas figuras que apresentam os resultados da robustez, espalhamento e filtragem.

Nota-se que dentre os 4 casos de redes neurais RBF que foram treinadas, o caso RBF AF é o melhor para alcançar resultados com precisão, inclusive até se comparado com AAKR nos grupos 2, 3, 4, 5 e 6. Além disso, a Figura 4.2, Figura 4.3 e Figura 4.4 mostram que o caso RBF AF foi um dos piores casos para a maioria dos grupos, considerando os desempenhos de robustez, espalhamento e filtragem.

Os modelos RBF mostraram ter melhor sensibilidade quando comparados ao AAKR. O caso RBF AF é o treinamento convencional da RBF que pode resultar em melhor precisão e pior sensibilidade conforme o esperado, uma vez que as outras métricas não são observadas durante este processo.



Figura 4.1 Precisão comparativa entre AAKR e RBF.



Figura 4.2 Sensibilidade (robustez) comparativa entre AAKR e RBF.

Fonte: Próprio autor



Figura 4.3 Sensibilidade (espalhamento) comparativa entre AAKR e RBF.

Fonte: Próprio autor

Resultados semelhantes foram observados para as métricas de filtragem.



Figura 4.4 Filtragem comparativa entre AAKR e RBF.

Fonte: Próprio autor

Os resultados médios para todos os grupos estão resumidos na Tabela 4.1 tanto para os casos da RBF quanto AAKR.

Caso	Precisão	Robustez	Espalhamento	Filtragem	L	Neuro.	Spread
AF	0,014	0,581	0,132	0,210	0,014	4.8	45,8
RF	0,343	0,010	0,010	0,011	0,010	1.0	3.2
ARF	0,075	0,100	0,077	0,082	0,087	2.2	3,9
ARSFF	0,170	0,038	0,033	0,036	0,069	1.3	3.7
AAKR	0,015	0,265	0,131	0,198			

Tabela 4.1Resultados médios para todos os grupos.

Fonte: Próprio autor

Aleatoriamente selecionando o grupo 2 verifica-se graficamente através da Figura 4.5 e da Figura 4.6 a diferença de desempenho entre todos os casos RBF e o AAKR.



Figura 4.5 Representação gráfica Robustez x Precisão do Grupo 2.

Fonte: Próprio autor
A Figura 4.5 apresenta o equilíbrio entre Precisão e Robustez, o que deixa claro que tais métricas são inversamente relacionadas, de modo que um modelo muito robusto não é muito preciso e vice-versa.

A Figura 4.6 é mais completa, pois está sendo apresentada a relação entre Robustez e Espalhamento incluindo um círculo para cada caso com sua área proporcional à precisão. Claramente, o caso RBF AF atinge a melhor precisão, pois a área do círculo é menor, mas a pior robustez, enquanto o caso RBF RF é o oposto. Nota-se também que o caso RBF ARF apresentou o desempenho mais próximo do que o modelo AAKR.



Figura 4.6 Mapeamento da Robustez x Espalhamento (Precisão) do Grupo 2.

4.2. ANÁLISE TEMPORAL

As métricas de desempenho indicam quantitativamente as características esperadas da resposta das estimativas dos modelos. Observando no domínio do tempo, permite entender melhor o funcionamento efetivo dos modelos avaliados. Arbitrariamente os modelos para o Grupo 2 são usados para apresentar esses resultados.

A Figura 4.7 apresenta as 65.412 medidas do período de tempo observado, cuja taxa de amostragem é de 1 minuto.



Figura 4.7 Dados de leituras dos sensores do Grupo 2.

Fonte: Próprio autor

Para observar a resposta de cada modelo aleatoriamente escolheu-se o sensor cuja Tag é *301072_T26_KE_211*, pertencente ao Grupo 2, que recebeu um *drift* negativo crescente em seu valor a partir da medida 1000 minutos da amostra apresentada na Figura 4.8 que contém 3272 medidas. Este sinal com *drift* está representado na Figura 4.8 (a) com a descrição "Medido c/ Drift".



Figura 4.8 (a) Dados reais, medidos e estimados com *drift*. (b) Resíduos.

Fonte: Próprio autor

A Figura 4.8 (a) mostra os valores reais, medidos com o *drift* e o estimado por cada modelo, já a Figura 4.8 (b), o resíduo de cada modelo com um limiar fixo, definido arbitrariamente, para detecção de faltas.

Observa-se na Figura 4.8 (b) que o tempo de resposta de alarme entre os modelos RBF ARF e AAKR é muito semelhante e ocorreu em torno do tempo 1050 minutos, para um limiar que varia entre -1 e 1%.

A Figura 4.9 mostra os valores estimados dos sensores do Grupo 2 para os modelos RBF AF, RBF ARF e AAKR.



Figura 4.9 Dados medidos e estimados em cada modelo para o Grupo 2.

Fonte: Próprio autor

A Figura 4.9 mostra a grande similaridade de desempenho entre os modelos AAKR e RBF ARF, expressando graficamente o que foi apresentado na Tabela 4.1, Figura 4.5 e Figura 4.6.

De acordo com os gráficos da Figura 4.9, há uma similaridade significativa da resposta dos modelos AAKR e RBF ARF.

4.3. RESULTADOS DOS TESTES DE HIPÓTESES

A partir dos casos obtidos utilizando o algoritmo modificado para treinamento das redes neurais RBF, tem-se interesse de verificar se as médias, de cada métrica, de cada grupo de sensores são melhores para um ou outro caso de treinamento. Logo, pode-se dizer que a média de uma métrica de desempenho para o caso RBF ARF é menor que em RBF AF ou que a média de uma métrica de desempenho para o caso RBF ARF é menor que em AAKR?

Os testes de hipóteses foram divididos em quatro comparativos:

- entre os casos RBF ARF e RBF AF;
- entre os casos RBF AF e AAKR;
- entre os casos RBF RF e AAKR;
- entre os casos RBF ARF e AAKR;

4.3.1. Entre RBF ARF e RBF AF

Neste teste de hipóteses, unicaudal à esquerda, entre as médias de cada métrica para os casos RBF ARF e RBF AF construiu-se a Tabela 4.2.

H_0 : $\mu_{ARF} = \mu_{AF}$	V	alor	Her (A. e. Nede H			
H_1 : $\mu_{\rm ARF} < \mu_{\rm AF}$	Crítico	Calculado	Hipotese Nula H_0			
Precisão	-2,015	3,361	Não Rejeita			
Robustez	-2,015	-5,474	Rejeita			
Espalhamento	-2,015	-1,369	Não Rejeita			
Filtragem	-2,015	-2,819	Rejeita			
Fonte: Próprio autor						

Tabela 4.2Teste de Hipótese por métrica entre ARF e AF.

Através dos resultados apresentados na Tabela 4.2, tem-se que os valores calculados são menores que o crítico, para as métricas Robustez e Filtragem, com isso, conclui-se que a média destas métricas do caso RBF ARF é menor que a do caso RBF AF, para a população representada por uma amostra de 6 medidas que são as métricas resultantes de cada grupo de sensores da bomba A-T2620 com um nível de significância de 5%. O

mesmo não se pode dizer para a Precisão e o Espalhamento, pois os testes falharam em rejeitar a hipótese nula H_0 , visto que os valores calculados foram superiores aos valores críticos, ou seja, eles ficaram situados na região de aceitação.

4.3.2. Entre RBF AF e AAKR

Neste teste de hipóteses, unicaudal a esquerda, entre as médias de cada métrica para os casos RBF AF e AAKR construiu-se a Tabela 4.3.

H_0 : $\mu_{AF} = \mu_{AAKR}$	Valor Crítico Calculado		Himátogo Nulo II
H_1 : $\mu_{AF} < \mu_{AAKR}$			Hipotese Nula H_0
Precisão	-2,015	-0,018	Não Rejeita
Robustez	-2,015	3,712	Não Rejeita
Espalhamento	-2,015	0,020	Não Rejeita
Filtragem	-2,015	0,395	Não Rejeita

Tabela 4.3Teste de Hipótese por métrica entre AF e AAKR.

Fonte: Próprio autor

Através dos resultados apresentados na Tabela 4.3, tem-se que os valores calculados são maiores que o crítico em todas as métricas, ou seja, o teste falhou em rejeitar a hipótese nula H_0 . Diante disso, com 5% de significância não se pode dizer que o caso RBF AF tem médias de métricas melhores que AAKR.

4.3.3. Entre RBF ARF e AAKR

Neste teste de hipóteses, unicaudal a esquerda, entre as médias de cada métrica para os casos RBF ARF e AAKR construiu-se a Tabela 4.4.

H_{θ} : $\mu_{ARF} = \mu_{AAKR}$	V	alor	Her (A Nede H
H_1 : $\mu_{ARF} < \mu_{AAKR}$	Crítico Calculado		Hipotese Nula H_0
Precisão	-2,015	4,003	Não Rejeita
Robustez	-2,015	-3,110	Rejeita
Espalhamento	-2,015	-1,532	Não Rejeita
Filtragem	-2,015 -2,647		Rejeita

Tabela 4.4Teste de Hipótese por métrica entre ARF e AAKR.

Fonte: Próprio autor

De forma semelhante aos testes de hipóteses entre RBF ARF e RBF AF, nos testes de hipóteses entre RBF ARF e AAKR, conforme resultados na Tabela 4.4, os valores calculados são menores que o crítico, para as métricas Robustez e Filtragem, com isso, concluise que a média destas métricas do caso RBF ARF é menor que a do caso AAKR, para a população representada por uma amostra de 6 medidas que são as métricas resultantes de cada grupo de sensores da bomba A-T2620 com um nível de significância de 5%. O mesmo não se pode dizer para a Precisão e o Espalhamento, pois os testes de hipóteses falharam em rejeitar a hipótese nula H_0 , visto que os valores calculados foram superiores aos valores críticos, ou seja, eles ficaram situados na região de aceitação.

4.3.4. Entre RBF RF e AAKR

Neste teste de hipóteses, unicaudal a esquerda, entre as médias de cada métrica para os casos RBF RF e AAKR construiu-se a Tabela 4.5.

H_0 : $\mu_{\rm RF} = \mu_{\rm AAKR}$	Valor					
H_1 : $\mu_{\rm RF} < \mu_{\rm AAKR}$	$_{\rm F} < \mu_{\rm AAKR}$ Crítico Calculado		Hipotese Nula H_0			
Precisão	-2,015	4,003	Não Rejeita			
Robustez	-2,015	-7,428	Rejeita			
Espalhamento	-2,015	-8,714	Rejeita			
Filtragem	-2,015	-8,234	Rejeita			
Fonto: Próprio autor						

Tabela 4.5Teste de Hipótese por métrica entre RF e AAKR.

Fonte: Próprio autor

Nestes testes de hipóteses, nota-se que através dos resultados apresentados na Tabela 4.5 os valores calculados são menores que o crítico para as métricas Robustez, Espalhamento e Filtragem, com isso, conclui-se que **a média destas métricas do caso RBF RF é menor que a do caso AAKR, para a população representada por uma amostra de 6 medidas que são as métricas resultantes de cada grupo de sensores da bomba A-T2620 com um nível de significância de 5%**. O mesmo não se pode dizer para a Precisão, pois o teste de hipóteses falhou em rejeitar a hipótese nula H_0 , visto que o valor calculado foi superior ao valor crítico, ou seja, ele ficou situado na região de aceitação.

Diante desses resultados, nota-se a importância de se ter um algoritmo de treinamento de redes neurais de função de base radial, para um modelo auto-associativo, que permita a priorização de uma ou mais métricas de desempenho. Um exemplo é o caso RBF AF, algoritmo convencional da RBF, que não foi possível afirmar para nenhuma métrica que o seu desempenho tenha sido melhor que o de um modelo não paramétrico baseado em memória, diferentemente de RBF AF, nos casos RBF ARF e RBF RF foi possível afirmar que para algumas métricas o desempenho do modelo foi superior ao AAKR.

4.4. COMPLEXIDADE COMPUTACIONAL

O tempo de processamento necessário para executar os modelos RBF AF, RBF ARF e AAKR também foi comparado. A Tabela 4.6 mostra o tempo necessário (em milissegundos) para cada modelo estimar os valores de todos os sensores em cada grupo usando um banco de dados de 3272 amostras e usando a mesma unidade de processamento (hardware e software).

Tabela 4.6

.6 Tempo decorrido (ms) por cada modelo para estimação.

ID do Grupo	2	3	6		
AAKR	374.38	299.41	262.5		
AF	8.18	11.18	6.99		
ARF	7.43	7.12	6.65		
Nº Neurônios	9	6	3		
Eantas Duáncia autor					

Fonte: Próprio autor

Nota-se que os modelos de estimação AF e ARF utilizaram, em média, apenas 2,5% do tempo utilizado pelo modelo AAKR.

Esse tempo pode ser diferente dependendo do sistema operacional e da otimização do algoritmo. No entanto, a diferença entre o tempo decorrido nos casos de RBF e AAKR é suficiente para enfatizar a limitação apresentada no início deste trabalho.

5.CONCLUSÕES E SUGESTÕES DE TRABALHOS FUTUROS

Neste trabalho, foi proposto um algoritmo de treinamento das redes neurais de função de base radial como uma alternativa para treinamento de RBF quando utilizada para estimar as leituras de sensores, explorando a redundância analítica via auto-associação. A expectativa inicial seria a obtenção de modelos com a RBF com resultados compatíveis com o AAKR que é um dos métodos mais utilizados para esta aplicação. Apesar dos excelentes resultados, a desvantagem do AAKR consiste na alta complexidade computacional, pelo fato de ser não-paramétrico baseado em memória. A RBF como método paramétrico e com certa semelhança ao AAKR pelo uso de uma função gaussiana de distâncias euclidianas (comum aos 2 métodos) poderia ser uma alternativa ao AAKR desde que possa alcançar métricas de desempenho também compatíveis.

Os 4 casos apresentados com diferentes valores para os parâmetros (α , β , $\gamma \in \delta$) da função custo *L* são exemplos de como podem ser exploradas as medidas de desempenho de cada modelo auto-associativo, simplesmente modificando os pesos de cada métrica através da soma ponderada utilizada na função custo *L*. Esta flexibilidade de priorização de uma métrica de desempenho no treinamento do modelo auto-associativo permite na prática a sua utilização em diversas condições de aplicação. Por exemplo, quando se quer usar um processo que exige precisão, treina-se focando na precisão do modelo, da mesma forma para robustez, espalhamento e filtragem. Especificamente para a aplicação de monitoramento de sensores usualmente realizado com AAKR, a configuração de parâmetros que mais permitiram a RBF se aproximar em desempenho foram $\alpha = 1$, $\beta = 1$, $\gamma = 0$ e $\delta = 0$, ou seja, o treinamento denominado ARF com foco na precisão e na robustez igualmente ponderados.

Os testes realizados com dados reais de uma bomba de injeção de água de uma unidade processamento de petróleo e gás, foram identificados 6 grupos correlacionados entre 35 sensores. Estes foram treinados com AAKR e com a RBF em 4 diferentes parâmetros de treinamento. Com estes testes foram levantadas as mesmas métricas para todos os casos e os resultados foram comparados por sensor e por grupo. Testes de hipótese para comparação de médias amostrais foram utilizados para permitir extrapolar a comparação à população de casos representados pela amostra utilizada. Foi possível confirmar com 5% de significância, de que há melhoria no treinamento da RBF para as métricas de interesse. Também foi comparado com testes de hipótese se há equivalência entre o AAKR e a RBF com treinamento ARF e foi observado desempenho equivalente e até melhor para algumas métricas.

Adicionalmente, para quantificar o esforço computacional foi realizada a medição do tempo de execução de todos os métodos no mesmo computador. Estes resultados de tempo de execução da RBF apresentaram aproximadamente 2,5% do despendido pelos modelos nãoparamétricos baseados em memória, sendo muito mais rápidas como esperado. A AAKR é mais simples e rápida para o treinamento, portanto, se fosse comparar o tempo de treinamento com o da RBF, é possível que fossem mais compatíveis. Isto não foi realizado por não ser o objetivo do trabalho obter um método com menor tempo de treinamento, mas um método com menor tempo durante sua execução. Apesar deste aspecto não ter sido mensurado neste trabalho, é interessante destacar esta característica que auxilia na compreensão dos resultados obtidos que podem ser interpretados como uma "migração" de complexidade da execução para o treinamento.

Desta forma, os resultados de tempo e desempenho atenderam às expectativas iniciais permitindo que a RBF alcance desempenhos compatíveis ao AAKR com tempo de execução muito menor. Portanto, a proposta deste trabalho de fato poderá ser aplicada no monitoramento e diagnóstico de instrumentos de medição em processos industriais como alternativa aos atualmente realizados com AAKR, com a vantagem do tempo de processamento sem perda de qualidade de estimações e com a desvantagem da maior complexidade do treinamento do modelo.

As contribuições deste trabalho podem ser resumidas nos seguintes itens:

- Proposição de um algoritmo de treinamento para redes neurais de função de base radial que permita priorizar no treinamento dos modelos autoassociativos métricas de performance, como a sensibilidade e a filtragem.
- Utilização de técnicas de baixo esforço computacional, em tempo real, neste caso Rede Neural Artificial de Função de Base Radial, para a implementação de modelo baseado em dados para monitoramento e diagnóstico de sensores

CAPÍTULO 5 – Conclusões e Sugestões de Trabalhos Futuros

e equipamentos com desempenho equivalente aos modelos não paramétricos baseados em memória.

- Revisão bibliográfica das principais técnicas de modelagem utilizadas no monitoramento de sensores.
- Publicação de artigos sobre utilização das métricas de desempenho no treinamento de RBF e comparação com AAKR no WCEAM 2018 [17] e IECON 2018 [18].

Entre as sugestões de trabalhos futuros estão:

- Reconciliação de dados através de testes básicos, de modelos fenomenológicos (*First Principle Model*) e técnicas estatísticas com o objetivo de melhorar a precisão das medições, reduzindo o efeito de erros aleatórios nos dados.
- Criação de algoritmos de treinamento que permitam a RBF se tornar adaptativa para processos com características variantes no tempo. A variação da correlação entre os sensores dos processos que possuem estas características torna o modelo treinado incorreto em questão de meses ou semanas. Nestes casos, a própria AAKR necessita se tornar adaptativa e pode ser vantajosa pelo fato de ter mais simplicidade no treinamento.
- Detecção de regime permanente também é necessária em aplicações de treinamento adaptativo, pelo fato dos modelos auto-associativos não explorarem a correlação dinâmica das variáveis e portanto não realizarem o monitoramento durante os transitórios.
- Realizar a comparação da RBF com outras técnicas bastante aceitas além do AAKR, como o MSET (Multivariate State Estimation Thecnique), o LWR (Locally Weighted Regression) e o AANN (Auto-Associative Neural Networks). Sendo esta última AANN também paramétrica, mas com algumas características diferentes da RBF.

- Criação de modelos que permitam monitorar também o regime transitório, utilizando possivelmente uma complementação com as teorias de observadores de estados.
- Realizar trabalhos de otimização dos parâmetros (α, β, γ e δ) de modo a obter os melhores resultados de alarmes verdadeiros com minimização dos alarmes falsos. Isto porque este é o objetivo final dos modelos auto-associativos: detectar uma falta no menor tempo possível e evitar alarmes falsos no maior tempo possível.

6. REFERÊNCIAS

- [1] S. Narasimhan e C. Jordache, Data reconciliation and gross error detection: An intelligent use of process data, Elsevier, 1999.
- [2] F. Kristjanpoller, A. Crespo Marquez, M. López-Campos, P. Viveros e T. Grubessich, "Reliability assessment methodology for multiproduct and flexible industrial process," 2016.
- [3] F. D. Maio, P. Baraldi, E. Zio e R. Seraoui, "Fault Detection in Nuclear Power Plants Components by a Combination of Statistical Methods," *IEEE Transactions on Reliability*, vol. 62, pp. 833-845, 12 2013.
- [4] M. Kettunen e P. Zhang, "An embedded fault detection, isolation and accommodation system in a model predictive controller for an industrial benchmark process," *Computers* & *Chemical Engineering*, vol. 32, pp. 2966-2985, 2008.
- [5] M. A. Sharif e R. I. Grosvenor, "Fault diagnosis in industrial control valves and actuators," em IMTC/98 Conference Proceedings. IEEE Instrumentation and Measurement Technology Conference. Where Instrumentation is Going (Cat. No.98CH36222), 1998.
- [6] J. Hines e D. Garvey, "Development and Application of Fault Detectability Performance Metrics for Instrument Calibration Verification and Anomaly Detection," *Journal of Pattern Recognition Research*, vol. 1, pp. 2-15, 1 2006.
- [7] J. M. T. Lee, J. Y. Kim, J. Shin, J. Kim e C. Yeom, "Sensor drift detection in SNG plant using auto-associative kernel regression," em 2017 IEEE International Systems Engineering Symposium (ISSE), 2017.
- [8] D. Theilliol, H. Noura e J.-C. Ponsart, "Fault diagnosis and accommodation of a threetank system based on analytical redundancy," *ISA transactions*, vol. 41, pp. 365-382, 2002.

- [9] L. Galotto, A. D. M. Brun, R. B. Godoy, F. R. R. Maciel e J. O. P. Pinto, "Data based tools for sensors continuous monitoring in industry applications," em 2015 IEEE 24th International Symposium on Industrial Electronics (ISIE), 2015.
- [10] P. Baraldi, F. D. Maio, D. Genini e E. Zio, "Comparison of Data-Driven Reconstruction Methods For Fault Detection," *IEEE Transactions on Reliability*, vol. 64, pp. 852-860, 9 2015.
- [11] L. Galotto, J. O. P. Pinto, B. Ozpineci, L. C. Leite e L. E. S. Borges, "Sensor Compensation in Motor Drives using Kernel Regression," em 2007 IEEE International Electric Machines Drives Conference, 2007.
- [12] N. Sundararajan, P. Saratchandran e L. Y. Wei, "A Review of Radial Basis Function (RBF) Neural Networks," em *Radial Basis Function Neural Networks with Sequential Learning*, WORLD SCIENTIFIC, 1999, pp. 1-19.
- [13] S. S. Haykin, Neural Networks: A Comprehensive Foundation, 2 edition ed., Prentice Hall, 1999.
- [14] C.-T. Chen, Linear system theory and design, Oxford University Press, Inc., 1998.
- [15] C. M. Bishop e others, Neural networks for pattern recognition, Oxford university press, 1995.
- [16] BATLAB Laboratório de Pesquisa e Desenvolvimento, "Relatório Técnico Final do Projeto Sistema de Monitoramento e Diagnóstico de Sensores," UFMS - Universidade Federal de Mato Grosso do Sul, Campo Grande, 2015.
- [17] M. A. Duarte Alves, L. Galotto, J. O. P. Pinto, H. Teixeira e M. C. M. Campos, "Modified Training and Optimization Method of Radial Basis Function Neural Network for Metrics Performance Guarantee in the Auto Association of Sensor Validation Tool," em WCEAM - 2018: The 13th World Congress on Engineering Asset Management, Stavanger, Norway, 2018.
- [18] M. A. Duarte Alves, L. Galotto, J. O. P. Pinto, R. C. Garcia, H. Teixeira e M. C. M. Campos, "RBF Neural Networks Modeling Methodology Compared to Non-Parametric Auto-Associative Models for Condition Monitoring Applications," em *IECON 2018 -*44th Annual Conference of the IEEE Industrial Electronics Society, Washington D.C., USA, 2018.

7. APÊNDICE A – PARÂMETROS DAS REDES NEURAIS DO CASO RBF AF

Foram treinadas 6 redes neurais de função de base radial, sendo uma para cada grupo de sensores, para o caso RBF AF que foi descrito na Tabela 3.1.

A RBF do **Grupo 1** ficou com **3** neurônios na camada oculta, **4** na saída e *spread* de **2,1**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.1, Tabela 7.2, e Tabela 7.3, respectivamente.

Tabela 7.1 Matriz de pesos **IW** para o Grupo 1 do caso RBF AF.

1,2408	0,8110	3,5012	0,3161			
0,0334	-1,4597	-0,7646	-0,8206			
1,2480	0,6045	1,0636	-5,1803			
Fonte: Próprio autor						

Tabela 7.2

7.2 Matriz de pesos **W** para o Grupo 1 do caso RBF AF.

-0,3213	-0,1830	1,5139			
-1,3193	-3,3333	-1,6769			
4,9053	-0,2773	3,1610			
-1,0992	-1,4188	-7,9248			
Fonte: Próprio autor					

Tabela 7.3 Vetor de *bias* w_0 para o Grupo 1 do caso RBF AF.

0,2701	
1,9214	
-0,5612	
1,0075	
Fonte: Próprio a	utor

A RBF do **Grupo 2** ficou com **8** neurônios na camada oculta, **9** na saída e *spread* de **78,9**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.4, Tabela 7.5, e Tabela 7.6, respectivamente.

Tabel	a 7.4	Matriz de pesos IW para o Grupo 2 do caso RBF AF.						
0,4871	0,9876	0,2097	0,5167	0,2010	0,1919	0,1926	0,6772	0,6671
-4,3538	-4,0960	-4,6810	-4,4963	-4,5156	-4,6174	-4,6776	-0,6357	-2,2072
0,4866	1,0543	0,2071	0,5674	0,1830	0,2033	0,2354	-0,4169	-0,3974
0,3859	0,7452	0,2034	0,4808	0,1867	0,2638	0,2692	0,6170	0,6150
-4,3407	-4,0974	-1,0119	-4,4949	-4,5368	-4,5907	-4,6853	-1,1694	-2,2388
-0,4552	0,0088	0,2096	0,0181	0,2431	0,8035	0,8131	-0,0139	0,1091
0,3677	0,8601	0,2153	0,4860	0,3257	0,2693	0,2742	-0,2724	-0,2523
-4,2580	-4,0355	-4,8199	-4,4956	-4,6329	-4,6190	-4,6862	-1,6791	-2,2223

Fonte: Próprio autor

Tabela 7.5Matriz de pesos W para o Grupo 2 do caso RBF AF.

-16918,49	-799,21	-544,29	19779,10	-115,91	-4484,23	2306,17	778,51
26123,86	344,95	4682,16	-28612,30	-100,17	3781,55	-5931,75	-285,74
37,10	-622,44	180,27	28,98	1202,86	-3,58	-243,41	-582,83
-17256,07	680,51	7640,35	19718,45	-242,15	-2421,50	-7488,30	-636,16
7534,54	13,38	-23210,99	-11239,60	-197,83	-2096,72	29104,23	88,63
-974,48	416,95	6384,30	632,65	-283,14	2478,22	-8300,77	-356,72
-1069,20	418,50	6012,45	695,51	-295,03	2393,23	-7810,16	-349,54
-1253,27	4433,21	2514,12	1727,82	-155,12	-75,49	-2918,46	-4288,98
5772,46	-4436,66	-6097,03	-2137,54	132,70	212,30	2265,70	4279,21

Tabela 7.6	Vetor de <i>bias</i> w ₀ para o Grupo 2 do caso RBF AF.
	-4,4072
	-1,0402
	0,1733
	1,3791
	2,0276
	0,2792
	1,4698
	14,7213
	9,9991
	Fonte: Próprio autor

CAPÍTULO 7 – Apêndice A – Parâmetros das redes neurais do caso RBF AF

A RBF do **Grupo 3** ficou com **5** neurônios na camada oculta, **6** na saída e *spread* de **40,3**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.7, Tabela 7.8, e Tabela 7.9, respectivamente.

Tabela 7.7Matriz de pesos IW para o Grupo 3 do caso RBF AF.

1,0374	1,2471	-0,1586	1,2759	0,6840	0,5866
-4,9230	-4,8282	-0,1586	-4,2859	-4,8630	-5,0373
0,1433	0,0967	6,2993	0,3782	0,0983	0,2276
0,1433	0,0967	6,2993	0,0761	0,2154	0,1736
0,4414	0,3864	-0,1586	0,3782	0,4497	0,3473

Fonte: Próprio autor

Tabela 7.8Matriz de pesos W para o Grupo 3 do caso RBF AF.

284,5629	-43,5942	-1075,1365	1057,3927	-258,8350
1981,3106	161,6778	-3541,9643	3569,5556	-2104,0013
22,7742	-7,2428	5,2904	178,6639	-197,9577
-244,1177	-44,0098	3778,4482	-3783,0662	280,2473
-1035,4243	-171,9905	-11,7631	-0,5396	1183,2439
-963,3342	-134,2820	1206,1978	-1199,2531	1096,7534

Tabela 7.9	Vetor de <i>bias</i> w ₀ para o Grupo 3 do caso RBF AF.	
	33,5966	
	-54,9081	
	1,2003	
	10,0545	
	26,1724	
	-13,9763	
	Fonte: Próprio autor	

CAPÍTULO 7 - Apêndice A - Parâmetros das redes neurais do caso RBF AF

A RBF do **Grupo 4** ficou com **4** neurônios na camada oculta, **5** na saída e *spread* de **100**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.10, Tabela 7.11, e Tabela 7.12, respectivamente.

Tabela 7.10Matriz de pesos IW para o Grupo 4 do caso RBF	AF.
--	-----

0,0074 -0,0504 -0,0078 -1,4675 -0,209 -0,0851 -0,0756 -0,0914 -0,4763 -0,209 0,0074 -0,0504 -0,0078 -1,5247 -0,827	-5,7876	-5,5339	-5,6916	-1,5247	-4,2879
-0,0851 -0,0756 -0,0914 -0,4763 -0,209 0,0074 -0,0504 -0,0078 -1,5247 -0,827	0,0074	-0,0504	-0,0078	-1,4675	-0,2098
0,0074 -0,0504 -0,0078 -1,5247 -0,827	-0,0851	-0,0756	-0,0914	-0,4763	-0,2098
	0,0074	-0,0504	-0,0078	-1,5247	-0,8272

Fonte: Próprio autor

Tabela 7.11Matriz de pesos W para o Grupo 4 do caso RBF AF.

-414,9254	-2338,9828	129,7314	2661,3252
-425,1108	-2188,6622	50,8463	2548,9041
-452,0487	-3081,5856	233,0224	3259,8556
-87,6409	-7815,3886	7312,4443	588,6678
8,0177	12420,3289	-673,2647	-11752,6796

Tabela 7.12	Vetor de <i>bias</i> w_0 para o Grupo 4 do caso RBF AF.	
	-40,2885	
	10,7810	
	37,2989	
	0,2878	
	-2,9091	
	Fonte: Próprio autor	

CAPÍTULO 7 - Apêndice A - Parâmetros das redes neurais do caso RBF AF

A RBF do **Grupo 5** ficou com 7 neurônios na camada oculta, 8 na saída e *spread* de **39,9**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.13, Tabela 7.14, e Tabela 7.15, respectivamente.

Tabela 7.13Matriz de pesos IW para o Grupo 5 do caso RBF AF.

0,8883	1,1363	0,3722	0,2990	1,1622	0,9614	0,5050	0,3419
-4,1814	-4,4666	-4,6207	-4,6499	-4,2721	-4,1252	-4,6656	-4,6858
-0,2331	0,9068	0,4562	0,4547	-0,1268	0,4355	0,2330	0,3554
-0,1789	0,1080	0,5337	0,5422	-0,2123	0,4307	0,5415	0,5948
-0,1835	0,0318	0,5896	0,6463	-0,3242	0,1701	0,6084	0,6861
-0,1172	0,1458	0,4003	0,5371	-0,1745	0,5027	0,5472	0,6841
-0,1607	0,1779	0,5167	0,5138	-0,3366	0,5628	0,5345	0,5659

Tabela 7.14Matriz de pesos W para o Grupo 5 do caso RBF AF.

-							
	1065,0096	47,1475	-1083,3478	-8858,2381	4098,0001	-554,6530	5303,0068
	-44,9893	-33,0752	1247,5380	-3685,1882	939,5137	625,1766	948,5499
	55,7128	-52,5127	-141,0809	5337,6723	-407,9362	-5430,2197	630,3018
	-295,1175	-54,9471	358,2821	728,6407	398,5926	834,2618	-1973,6240
	25,5378	-21,8031	100,5753	8094,5490	-2519,3460	174,4297	-5854,6845
	-247,4145	-41,4421	-331,6759	2298,7462	-3365,1948	362,0242	1320,2614
	123,0978	-29,2963	-555,7166	-1623,2310	1005,9550	-687,9466	1767,1980
	-459,0889	-55,0067	274,8890	-2423,1977	-75,7365	4612,1348	-1884,7035

Fonte: Próprio autor

Tabela 7.15	Vetor de <i>bias</i> w_0 para o Grupo 5 do caso RBF AF.
	-11,5037
	0,4778
	4,5351
	-0,1831
	-1,2747
	1,2192
	-1,6464
	6,3213
	Fonte: Próprio autor

CAPÍTULO 7 - Apêndice A - Parâmetros das redes neurais do caso RBF AF

A RBF do **Grupo 6** ficou com 2 neurônios na camada oculta, 3 na saída e *spread* de **13,4**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 7.16, Tabela 7.17, e Tabela 7.18, respectivamente.

Tabela 7.16Matriz de pesos IW para o Grupo 6 do caso RBF AF.

-10,6889 -12,6113 -12,6295					
3,3871 3,0272 3,0296					
Fonte: Próprio autor					

51,3857	63,8866	
-22,9792	-7,2425	
-27,5033	-11,7345	
Fonte: Próprio autor		

Matriz de pesos W para o Grupo 6 do caso RBF AF.

Tabela 7.18

Tabela 7.17

Vetor de *bias* w_0 para o Grupo 6 do caso RBF AF.

-66,6325 10,8212 15,6720 Fonte: Próprio autor

8.APÊNDICE B – PARÂMETROS DAS REDES NEURAIS DO CASO RBF RF

Foram treinadas 6 redes neurais de função de base radial, sendo uma para cada grupo de sensores, para o caso RBF RF que foi descrito na Tabela 3.1.

A RBF do **Grupo 1** ficou com **1** neurônios na camada oculta, **4** na saída e *spread* de **0,1**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.1, Tabela 8.2, e Tabela 8.3, respectivamente.

0,7331	0,6045	0,3323	0,2160			
Fonte: Próprio autor						

Tabela 8.2 Matriz de pesos W para o Grupo 1 do caso RBF RF. 1,3059 1,8671 0,3954 0,7703 Fonte: Próprio autor

 Tabela 8.3
 Vetor de *bias* w_0 para o Grupo 1 do caso RBF RF.

 0,1146

 -0,3137

 0,1159

 -0,1661

A RBF do **Grupo 2** ficou com **1** neurônios na camada oculta, **9** na saída e *spread* de **3,8**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.4, Tabela 8.5, e Tabela 8.6, respectivamente.

Tabel	a 8.4	Matriz de pesos IW para o Grupo 2 do caso RBF RF.						
-4,3538	-4,0960	-4,6810	-4,4963	-4,5156	-4,6174	-4,6776	-0,6357	-2,2072
			Fonte	e: Próprio	autor			
Tabe	la 8.5	Matriz o	ie pesos V	V para o G	rupo 2 do	caso RBF	FRF.	
				-4,9766				
				-4,8157				
				-5,3493				
				-5,2536				
				-5,9993				
				-5,3812				
				-5,4719				
				-1,6715				
			_	-2,7003				
			Fonte	e: Próprio	autor			
Tab	ela 8.6	Vetor of	de <i>bias</i> w ₀	para o Gr	upo 2 do c	caso RBF	RF.	
				0 1947				
				0.3105				
				0,2118				
				0,2085				
				0,2343				
				0,2150				
				0,2144				
				0,0277				
				0,0405				
			Fonte	e: Próprio	autor			

A RBF do **Grupo 3** ficou com **1** neurônios na camada oculta, **6** na saída e *spread* de

58

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.7, Tabela 8.8, e Tabela 8.9, respectivamente.

Tabela 8.7	Ma	triz de pes	sos IW par	a o Grupo	3 do caso	RBF RF.
	-4,9230	-4,8282	-0,1586	-4,2859	-4,8630	-5,0373
-	Fonte: Próprio autor					

Tabela 8.8

Matriz de pesos W para o Grupo 3 do caso RBF RF.

-5,2038 -5,1353 -0,3085 -4,5120 -5,2395 -5,1515 Fonte: Próprio autor

 Tabela 8.9
 Vetor de $bias w_0$ para o Grupo 3 do caso RBF RF.

 0,1048
 0,1261

 0,1181
 0,1360

 0,1634
 0,1812

Fonte: Próprio autor

A RBF do **Grupo 4** ficou com **1** neurônios na camada oculta, **5** na saída e *spread* de **2,2**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.10, Tabela 8.11, e Tabela 8.12, respectivamente.

 Tabela 8.10
 Matriz de pesos IW para o Grupo 4 do caso RBF RF.

 -5,6152
 -5,4317
 -5,3758
 -0,4763
 -3,9138

Tabela 8.11	Matriz de pesos W para o Grupo 4 do caso RBF RF.
	-7,0749
	-6,8660
	-6,8080
	0,2945
	-5,1809
	Fonte: Próprio autor

Tabela 8.12	Vetor de <i>bias</i> w_0 para o Grupo 4 do caso RBF RF.
	0,1120
	0,1059
	0,0979
	-0,4059
	-0,1022
	Fonte: Próprio autor

A RBF do **Grupo 5** ficou com 1 neurônios na camada oculta, 8 na saída e *spread* de 4,1.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.13, Tabela 8.14, e Tabela 8.15, respectivamente.

Та	abela 8.13	Ma	triz de pes	sos IW par	ra o Grupo	5 do caso	RBF RF.	
•	-4,3034	-4,6192	-4,7311	-4,7080	-4,3754	-4,6694	-4,7342	-4,7132
-				Fonte: Pro	óprio autor	•		
Т	abela 8.14	l Ma	atriz de pe	sos W par	a o Grupo	5 do caso	RBF RF.	
				-4,4	275			
				-4,8	332			
				-4,9	358			
				-4,9	243			
				-4,5	232			
				-4,7	790			
				-4,9	491			
				-4,9	322			
					, ,			

Tabela 8.15	Vetor de <i>bias</i> w_0 para o Grupo 5 do caso RBF RF.
	0,1439
	0,2407
	0,2227
	0,2211
	0,1596
	0,2402
	0,2183
	0,2086
	Fonte: Próprio autor

CAPÍTULO 8 - Apêndice B - Parâmetros das redes neurais do caso RBF RF

A RBF do **Grupo 6** ficou com **1** neurônios na camada oculta, **3** na saída e *spread* de **5,8**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 8.16, Tabela 8.17, e Tabela 8.18, respectivamente.

Tabela 8.16Matriz de pesos IW para o Grupo 6 do caso RBF RF.

-10,4667 -12,4963 -12,4838 Fonte: Próprio autor

Tabela 8.17 Matriz de pesos W para o Grupo 6 do caso RBF RF. -10,5436 -12,6367 -12,6379 Fonte: Próprio autor

Tabela 8.18 Vetor de *bias* w₀ para o Grupo 6 do caso RBF RF. -0,0397 0,0751 0,0739 Fonte: Próprio autor

9. APÊNDICE C – PARÂMETROS DAS REDES NEURAIS DO CASO RBF ARF

Foram treinadas 6 redes neurais de função de base radial, sendo uma para cada grupo de sensores, para o caso RBF ARF que foi descrito na Tabela 3.1.

A RBF do **Grupo 1** ficou com **3** neurônios na camada oculta, **4** na saída e *spread* de **1,2**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.1, Tabela 9.2, e Tabela 9.3, respectivamente.

Tabela 9.1	Matriz de peso	iz de pesos IW para o Grupo 1 do caso 1			
	0,0261	-1,4597	-0,8865	-0,5557	
	-0,0478	0,8110	2,8918	0,2330	
	1,7594	0,6045	1,7948	-4,8866	

Fonte: Próprio autor

Tabela 9.2

Matriz de pesos **W** para o Grupo 1 do caso RBF ARF.

-0,2896	-0,6314	1,6117			
-2,3046	0,0313	-0,2717			
-0,7321	3,3144	2,7100			
-0,7455	-0,0768	-7,1804			
Fonte: Próprio autor					

Tabela 9.3 Vetor de *bias* w_0 para o Grupo 1 do caso RBF ARF.

_	0,3475	
	0,7955	
	-0,0052	
	0,2642	
Fonte	: Próprio	autor

A RBF do **Grupo 2** ficou com **4** neurônios na camada oculta, **9** na saída e *spread* de **4,8**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.4, Tabela 9.5, e Tabela 9.6, respectivamente.

Tabela 9.4	Matriz de pesos IW para o Grupo 2 do caso RBF ARF.						
-4,3745 -4,0754	-4,8199	-4,4951	-5,7239	-4,6026	-4,6864	-1,6039	-2,4803
-1,6911 -1,2476	0,2036	-1,1972	0,1651	1,2294	1,2321	11,9138	11,3978
0,7451 1,0814	0,2042	0,5684	0,1699	0,0084	0,0144	-0,7373	-0,7021
-0,8816 -1,0610	0,2145	-0,6377	0,2616	0,5648	0,5515	1,7985	1,7534

Fonte: Próprio autor

Tabela 9.5Matriz de pesos W para o Grupo 2 do caso RBF ARF.

	-3,9973	-1,0760	1,9635	-1,5684
	-5,2159	-2,0905	1,5082	-2,8260
	-6,0909	-1,0551	-0,6331	-0,6153
	-3,4086	0,2319	2,0598	-0,5491
	-4,5516	1,0097	0,6389	0,7189
	-3,4743	2,4549	0,3709	1,6844
	-3,5138	2,5065	0,4146	1,6912
	-2,5785	10,9902	-2,7213	2,1070
	-3,1892	10,9620	-2,4467	2,2338
-				

Fonte: Próprio autor

Tabela 9.6

9.6 Vetor de *bias* w_0 para o Grupo 2 do caso RBF ARF.

-0,4580
0,9700
1,2509
-1,2606
-0,8393
-1,3092
-1,3543
0,9720
0,6484
Fonte: Próprio autor

A RBF do **Grupo 3** ficou com **2** neurônios na camada oculta, **6** na saída e *spread* de **2**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.7, Tabela 9.8, e Tabela 9.9, respectivamente.

Tabela 9.7	Matriz de pesos IW para o Grupo 3 do caso RBF ARF.					
-	-4,9230	-4,8282	-0,1586	-4,2859	-4,8630	-5,0373
	0,1433	0,0967	6,2993	0,0761	0,0983	0,1079
-			Fonte: Pro	óprio autor	•	
Tabela 9.8	Mat	triz de pes	os W para	o Grupo 3	3 do caso 1	RBF ARF.
			-5,8263	-0,0887		
			-5,7687	-0,0710		
			0,0015	6,5914		
			-5,0091	0,0496		
			-5,8694	0,0045		
			-5,7942	0,0061		
			Fonte: Pro	óprio autor	•	
Tabela 9.9) Ve	etor de <i>bia</i>	s w ₀ para o	o Grupo 3	do caso R	BF ARF.
			0,1	025		
			0,1	234		
			-0,1	598		

A RBF do **Grupo 4** ficou com **2** neurônios na camada oculta, **5** na saída e *spread* de

0,1281 0,1573 0,1754 Fonte: Próprio autor

5,5.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.10, Tabela 9.11, e Tabela 9.12, respectivamente.

Tabela 9.10	Matriz de pesos IW para o O	Grupo 4 do caso RBF ARF				
	-5,7876 -5,5339 -5,6916	-1,5247 -4,2879				
	1,6138 1,6949 1,7379	3,7170 2,8768				
	Fonte: Próprio	o autor				
Tabela 9.11	Matriz de pesos W para o C	Grupo 4 do caso RBF ARF.				
	-5,9080 2,	1420				
	-5,6893 2,	3831				
	-5,5540 2,	5942				
	5,1695 6,	1333				
	-2,6729 3,	1063				
	Fonte: Próprio autor					
Tabela 9.12	Vetor de <i>bias</i> w_0 para o Gr -0.4124	rupo 4 do caso RBF ARF.				
	-0.5454					
	-0,6605					
	-3,6391					
	-1,3282					
	-1,3282	r				

CAPÍTULO 9 – Apêndice C – Parâmetros das redes neurais do caso RBF ARF

A RBF do **Grupo 5** ficou com **1** neurônios na camada oculta, **8** na saída e *spread* de

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.13, Tabela 9.14, e Tabela 9.15, respectivamente.

4,2.

Та	bela 9.13	Matriz de pesos IW para			o Grupo	o Grupo 5 do caso RBF ARF.		
	-4,3034	-4,6192	-4,7311	-4,7080	-4,3754	-4,6694	-4,7342	-4,7132

Tabela 9.14	Matriz de pesos W para o Grupo 5 do caso RBF ARF.
	-4,4284
	-4,8338
	-4,9365
	-4,9249
	-4,5240
	-4,7798
	-4,9498
	-4,9330
	Fonte: Próprio autor
	-

Tabela 9.15 Vetor de *bias* w_0 para o Grupo 5 do caso RBF ARF.

-		_
	0,1449	
	0,2417	
	0,2237	
	0,2221	
	0,1606	
	0,2412	
	0,2194	
	0,2097	
Fonte	: Próprio	autor

A RBF do **Grupo 6** ficou com **1** neurônios na camada oculta, **3** na saída e *spread* de **5,8**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 9.16, Tabela 9.17, e Tabela 9.18, respectivamente.

Tabela 9.16Matriz de pesos IW para o Grupo 6 do caso RBF ARF.

-10,4667 -12,4963 -12,4838 Fonte: Próprio autor

Tabela 9.17	Matriz de pesos W para o Grupo 6 do caso RBF ARF.
	-10,5436
	-12,6367
	-12,6379
	Fonte: Próprio autor
	_

CAPÍTULO 9 – Apêndice C – Parâmetros das redes neurais do caso RBF ARF

Tabela 9.18 Vetor de *bias* w_0 para o Grupo 6 do caso RBF ARF. -0,0397 0,0751 0,0739 Fonte: Próprio autor

10.APÊNDICE D – PARÂMETROS DAS REDES NEURAIS DO CASO RBF ARSFF

Foram treinadas 6 redes neurais de função de base radial, sendo uma para cada grupo de sensores, para o caso RBF ARSFF que foi descrito na Tabela 3.1.

A RBF do **Grupo 1** ficou com **1** neurônios na camada oculta, **4** na saída e *spread* de **1,4**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.1, Tabela 10.2, e Tabela 10.3, respectivamente.

Tabela 10.1	Matriz de pesos IW	/ para o Grupo 1	do caso RBF ARSFF

0,0334	-1,4597	-0,7646	-0,8206
	Fonte: Pro	óprio auto	r

Tabela 10.2Matriz de pesos W para o Grupo 1 do caso RBF ARSFF.

-0,1162 -2,3895 -1,9231 -0,6346 Fonte: Próprio autor

Tabela 10.3

0.3 Vetor de *bias* w₀ para o Grupo 1 do caso RBF ARSFF.

0,1948 0,8539 1,0343 0,1493 Fonte: Próprio autor A RBF do **Grupo 2** ficou com 2 neurônios na camada oculta, 9 na saída e *spread* de 4,1.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.4, Tabela 10.5, e Tabela 10.6, respectivamente.

Tabela 10.4Matriz de pesos IW para o Grupo 2 do caso RBF ARSFF.								
-4,3748	-4,0913	-4,8199	-4,4949	-5,6895	-4,5658	-4,6812	-1,2884	-2,3049
-1,6911	-1,2476	0,2036	-1,1972	0,1651	1,2294	1,2321	11,9138	11,3978
Fonte: Próprio autor								

Tabela 10.5Matriz de pesos W para o Grupo 2 do caso RBF ARSFF.

-4,7411	-1,7592
-4,5860	-1,4581
-5,0997	-0,0096
-5,0021	-1,2660
-5,7448	-0,0731
-5,1132	0,9302
-5,2002	0,9368
-1,5484	12,0234
-2,5228	11,6539
	• •

Fonte: Próprio autor

Tabela 10.6

10.6 Vetor de *bias* w_0 para o Grupo 2 do caso RBF ARSFF.

0,2027	
0,3173	
0,2130	
0,2145	
0,2364	
0,2122	
0,2116	
-0,0201	
-0,0057	
 Dućania	

A RBF do **Grupo 3** ficou com **2** neurônios na camada oculta, **6** na saída e *spread* de **2,2**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.7, Tabela 10.8, e Tabela 10.9, respectivamente.

Tabela 10.7	Matriz de pesos IW para o Grupo 3 do caso RBF ARSFF.					
	-4,9230	-4,8282	-0,1586	-4,2859	-4,8630	-5,0373
	0,0027	0,0967	6,2993	0,0761	0,2154	0,1079
			Fonte: Pró	prio autoi	[
Tabela 10.8	Matri	iz de peso	s W para o	o Grupo 3	do caso R	BF ARSFF.
			-5,6738	-0,0900	•	
			-5,6127	-0,0721		
			0,0161	6,5586		
			-4,8860	0,0475		
			-5,7142	0,0036		
			-5,6354	0,0054		
			Fonte: Pró	prio autor	ſ	
Tabela 10.9	Vete	or de <i>bias</i>	w ₀ para o	Grupo 3 d	lo caso RE	BF ARSFF.
			0,1	035		
	0,1243					
-0,1723						
			0,1	289		
			0,1	581		
			0.1	761		

0,1761 Fonte: Próprio autor

A RBF do **Grupo 4** ficou com 1 neurônios na camada oculta, 5 na saída e *spread* de

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.10, Tabela 10.11, e Tabela 10.12, respectivamente.

4,5.

Tabela 10.10	Matriz de pesos IW para o Grupo 4 do caso RBF ARSFF.
	-5,6152 -5,4317 -5,3758 -0,4763 -3,9138
	Fonte: Próprio autor
Tabela 10.11	Matriz de pesos W para o Grupo 4 do caso RBF ARSFF.
	-7,1855
	-7,1384
	-7,1436
	0,0100
	-5,1435
	Fonte: Próprio autor
Tabala 10.12	Voten de him en nom e Crune 4 de esse DDE ADSEE
Tabela 10.12	vetor de <i>blas</i> w_0 para o Grupo 4 do caso RBF ARSFF.
	0,3765
	0,3699
	0,3627
	-0,4039
	0,0862
	Fonte: Próprio autor

CAPÍTULO 10 – Apêndice D – Parâmetros das redes neurais do caso	RBF	ARSEF
CAT II O LO IO - Apendice D - I arametros das redes neurais do caso	KDI.	AKSIT

A RBF do **Grupo 5** ficou com **1** neurônios na camada oculta, **8** na saída e *spread* de

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.13, Tabela 10.14, e Tabela 10.15, respectivamente.

4,1.

Tabela 10	.13	Matriz	z de pesos	IW para	o Grupo 5	do caso R	BF ARSF	F.
-4,3	3034	-4,6192	-4,7311	-4,7080	-4,3754	-4,6694	-4,7342	-4,7132

Tabela 10.14	Matriz de pesos W para o Grupo 5 do caso RBF ARSFF.
	-4,4275
	-4,8332
	-4,9358
	-4,9243
	-4,5232
	-4,7790
	-4,9491
	-4,9322
	Fonte: Próprio autor

CAPÍTULO 10 – Apêndice D – Parâmetros das redes neurais do caso RBF ARSFF

Tabela 10.15 Vetor de *bias* w_0 para o Grupo 5 do caso RBF ARSFF.

0,1439
0,2407
0,2227
0,2211
0,1596
0,2402
0,2183
0,2086
Fonte: Próprio autor

A RBF do **Grupo 6** ficou com **1** neurônios na camada oculta, **3** na saída e *spread* de **5,8**.

A matriz **IW** dos pesos da camada oculta, a matriz **W** dos pesos da camada de saída e o vetor w_0 de *bias* da camada de saída estão na Tabela 10.16, Tabela 10.17, e Tabela 10.18, respectivamente.

Tabela 10.16	Matriz de pesos IW	para o Grupo 6 do caso	RBF ARSFF
	1	1 1	

-10,4667 -12,4963 -12,4838 Fonte: Próprio autor
Tabela 10.17	Matriz de pesos W para o Grupo 6 do caso RBF ARSFF.
	-10,5436
	-12,6367
	-12,6379

CAPÍTULO 10 - Apêndice D - Parâmetros das redes neurais do caso RBF ARSFF

Fonte: Próprio autor

Tabela 10.18 Vetor de bias w₀ para o Grupo 6 do caso RBF ARSFF.

> -0,0397 0,0751 0,0739 Fonte: Próprio autor