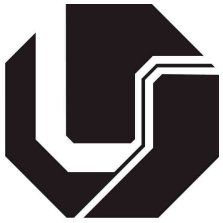
UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



ANÁLISE DE DESEMPENHO DE REDES NEURAIS ARTIFICIAIS DO TIPO MULTILAYER PERCEPTRON POR MEIO DO DISTANCIAMENTO DOS PONTOS DO ESPAÇO DE SAÍDA

JOSÉ RICARDO GONÇALVES MANZAN

SETEMBRO
2016

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

ANÁLISE DE DESEMPENHO DE REDES NEURAIS ARTIFICIAIS DO TIPO MULTILAYER PERCEPTRON POR MEIO DO DISTANCIAMENTO DOS PONTOS DO ESPAÇO DE SAÍDA

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia perante os membros da banca examinadora, como requisito parcial para obtenção do título de Doutor em Ciências.

Keiji Yamanaka, Phd (UFU) - Orientador
Shigueo Nomura, Dr (UFU) - Co-orientador
Edilberto Pereira Teixeira, Dr (UNIUBE)
Hugo Leonardo Pereira Rufino, Dr (IFTM)
Igor Santos Peretta, Dr (UFU)
Marcelo Rodrigues de Souza, Dr (UFU)

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

M296a
2016

Manzan, José Ricardo Gonçalves, 1984-
Análise de desempenho de redes neurais artificiais do tipo multilayer perceptron por meio do distanciamento dos pontos do espaço de saída / José Ricardo Gonçalves Manzan. - 2016.
129 f. : il.

Orientador: Keiji Yamanaka.
Tese (doutorado) - Universidade Federal de Uberlândia, Programa de Pós-Graduação em Engenharia Elétrica.
Inclui bibliografia.

1. Engenharia elétrica - Teses. 2. Reconhecimento de padrões - Teses. 3. Redes neurais (Computação) - Teses. I. Yamanaka, Keiji. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

CDU: 621.3

ANÁLISE DE DESEMPENHO DE REDES NEURAIS ARTIFICIAIS DO TIPO MULTILAYER PERCEPTRON POR MEIO DO DISTANCIAMENTO DOS PONTOS DO ESPAÇO DE SAÍDA

JOSÉ RICARDO GONÇALVES MANZAN

Tese apresentada por José Ricardo Gonçalves Manzan à Universidade Federal de Uberlândia como requisito parcial para obtenção do título de Doutor em Ciências.

Keiji Yamanaka, Phd	Darizon Alves de Andrade, Phd
Orientador	Coordenador do Curso de Pós-Graduação

Aos meus pais, Antônio e Geralda pelas primeiras lições de vida e pela formação continuada permanente cuja metodologia dos bons exemplos, me ensina a cada dia a ser uma pessoa melhor. A minha querida e amada esposa Ana Paula. Seu apoio incondicional, sua paciência e compreensão, são verdadeiras provas de amor e companheirismo. Eu não teria conseguido superar as dificuldades sem você ao meu lado.

Agradecimentos

À Deus que esteve comigo em todos os momentos, principalmente naqueles em que eu já não acreditava que iria conseguir concluir o curso. À ele que é o mestre dos mestres, o doutor dos doutores.

Ao meu orientador Professor Keiji Yamanaka pela confiança, paciência e compreensão. Obrigado por aceitar minhas dificuldades e limitações, por me permitir continuar nesse projeto de pesquisa e por me animar nos momentos de grande dificuldade. Obrigado também pelas colaborações durante a pesquisa, me ajudando a enxergar caminhos em meio às tempestades

Ao verdadeiro amigo, Professor Igor Santos Peretta. Suas visões, percepções, críticas e sugestões colaboraram de forma fundamental para o desenvolvimento desse trabalho. Suas instigações conseguem motivar a continuidade de qualquer pesquisa.

Aos meus amados pais, que serão sempre meus eternos professores. Por todos os ensinamentos de vida. Por terem me guiado nos meus primeiros passos e por estarem servindo de referência para todos os outros passos que ainda terei que tomar. Por toda a dedicação, amor e carinho.

À minha amada esposa Ana Paula. Seu apoio durante essa caminhada trouxe tranquilidade e paz. Além disso, suas palavras de conforto e aconselhamento me ajudaram a amadurecer e a enxergar os obstáculos de outra forma. Obrigado pela paciência, compreensão, amor e carinho.

Aos amigos do IFTM, da Paróquia São Geraldo Magela (em Uberaba) e aos meus familiares pelas orações para que eu concluísse o curso.

À Cinara Fagundes Paranhos Mattos, por desenvolver seu trabalho com seriedade e competência, sem deixar de lado valores como amizade, caridade e simpatia.

Resumo

MANZAN, José R. G. *Análise de Desempenho de Redes Neurais Artificiais do tipo Multilayer Perceptron por meio do distanciamento dos pontos do espaço de saída*, Uberlândia, Faculdade de Engenharia Elétrica - UFU, 2015.

As Redes Neurais Artificiais (RNAs) do tipo Multilayer Perceptron (MLP) são amplamente conhecidas e utilizadas numa grande gama de aplicações relacionadas ao Reconhecimento de Padrões (RP). Naturalmente, vários estudos são realizados no intuito de melhorar o desempenho dessa ferramenta. Esses estudos buscam abordagens diversas como a melhoria do algoritmo de treinamento, a determinação de topologias ideais para cada problema, a inicialização dos pesos sinápticos de treinamento e o tratamento dos padrões de entrada da rede. Nesse contexto, o espaço de saída das MLPs não tem sido explorado como forma de melhorar o seu desempenho. Este trabalho consiste num estudo sobre a influência do distanciamento dos pontos do espaço de saída no desempenho de MLPs em tarefas de RP. O aumento da distância dos pontos do espaço de saída, que são os alvos da rede, é obtido pela utilização de vetores-alvo bipolares e ortogonais (VBO). A condição de ortogonalidade implica no aumento da distância euclidiana, o que não ocorre com vetores convencionais VCs que são não ortogonais. O presente estudo mostra uma análise matemática do algoritmo de treinamento denominado *backpropagation*, relacionando sua dedução a partir da função erro com a dedução alternativa a partir da função distância euclidiana. Nessa análise, a hipótese de que o uso de VBOs melhora o desempenho de redes do tipo MLP é demonstrada por meio da redução da suscetibilidade de classificação incorreta em relação ao uso dos (VC). Neste trabalho, também são mostradas análises experimentais na classificação de dígitos manuscritos, íris humana e signos da linguagem de sinais australiana. Uma das análises avaliou estatisticamente a propensão a erros de classificação de redes do tipo MLP treinadas com VCs e VBOs. Os resultados confirmaram as conclusões obtidas com a análise matemática. Em outra análise experimental, foi avaliado o desempenho das redes do tipo MLP treinadas com VCs e VBOs, desde a conclusão do primeiro ciclo de treinamento até os ciclos finais do treinamento. Os resultados indicam três aspectos importantes. Um deles é o grande aumento nas taxas de classificação de padrões nos primeiros ciclos de treinamento. O outro aspecto é a menor suscetibilidade ao efeito do *overfitting* em redes MLP treinadas com VBOs. E o terceiro consiste na obtenção de taxas significativas de desempenho com pouco treinamento e também com pouco esforço computacional. Por fim, o trabalho também realizou um estudo sobre a robustez das redes MLP diante da alteração do número de neurônios da camada intermediária e o valor da taxa de aprendizagem inicial. Foi constatado que as redes treinadas com VBOs apresentam pouca suscetibilidade a essas alterações de parâmetros, ao contrário do que ocorre com as redes treinadas com vetores convencionais.

Palavras-chave: Reconhecimento de padrões, vetores-alvo, vetores bipolares convencionais, vetores bipolares ortogonais, esforço computacional.

Abstract

MANZAN, José R. G. *Advanced analysis of using new target vectors on high performance MLPs*, Uberlândia, Faculty of Electric Engineering - UFU, 2012.

Artificial Neural Networks (ANN) of Multilayer Perceptron (MLP) type are widely known and used in a wide range of applications related to Pattern Recognition (PR). Studies have been conducted in order to improve the performance of that tool. They search for different approaches such as the improvement of the training algorithm, the determination of optimal topologies for each problem, the initialization of synaptic training weights, and the treatment of the network input patterns. In this context, the output of MLPs hasn't been exploited in order to improve its performance. This work is a study on the influence of the distance of the output space points in the performance of MLPs in PR tasks. The gap widening of the output points, which are the network targets, is obtained by using target bipolar and orthogonal vectors (VBO). The orthogonality condition implies in the Euclidean gap widening, which does not occur with conventional vectors VCs that are not orthogonal. This study shows a mathematical analysis of the training algorithm called backpropagation, relating its deduction from the error function with the alternative deduction from the Euclidean distance function. The assumption that the use of VBOs improves the MLP type network performance is demonstrated by the reduction of the misclassification susceptibility in relation to the use of (VCs). This work also shows experimental analysis on classification of manuscripts digits, human iris and signs of the Australian sign language. The propensity to networks misclassification of MLP type trained with VCs and VBOs is statistically evaluated. The results confirmed the findings obtained in the mathematical analysis. In another experimental analysis, we have evaluated the performance of the MLP type networks trained with VCs and VBOs, from the completion of the first training cycle to the training end cycles. The results show three important findings. Firstly, the great increase in patterns classification rates in the first training cycles. The second aspect is less susceptible to the effect of the overfitting in MLP networks trained with VBOs. The third one deals with the achievement of significant rates of performance with little training, and with little computational effort. Finally, the study also conducted a study on the robustness of the MLP networks before the change in the number of neurons of the intermediate layer, and the value of the initial learning rate. It was found that the networks trained with VBOs have little susceptibility to these changes of parameters, unlike what happens with the networks trained with conventional vectors.

Keywords: Pattern recognition, target-vectors, conventional bipolar vectors, orthogonal bipolar vectors, computational effort.

Sumário

Lista de Figuras **xii**

Lista de Tabelas **xiv**

Lista de Abreviaturas **xv**

1 Introdução **15**
1.1 Considerações iniciais 15
1.2 Motivação e originalidade 16
1.3 Objetivo da tese 18
1.4 Estrutura do trabalho 19

2 Fundamentos teóricos de Reconhecimento de Padrões e Redes Neurais Artificiais **21**
2.1 Fundamentos de reconhecimento de padrões 21
2.1.1 Conceitos de um sistema de reconhecimento de padrões 21
2.1.2 Etapas de um sistema de reconhecimento de padrões 23
2.1.3 Técnicas para classificação de padrões 25
2.2 Técnicas de Redes Neurais Artificiais 27
2.2.1 Neurônio biológico 28
2.2.2 Neurônio artificial 29
2.2.3 Rede neural artificial 29
2.2.4 Funções de ativação 35

3 Redes Neurais Artificiais do tipo Multilayer Perceptron **40**
3.1 Arquitetura e características 40
3.2 Treinamento de redes Perceptron multicamadas – algoritmo *backpropagation* . . 42
3.2.1 Propagação de um padrão no algoritmo de treinamento 42
3.2.2 Retropropagação do erro 44
3.3 Principais mecanismos de melhoria de desempenho de redes Multilayer Perceptron 45
3.3.1 Maximização do conteúdo de informação 46
3.3.2 Função de ativação 46
3.3.3 Normalização das entradas 48
3.3.4 Inicialização dos pesos sinápticos 48
3.3.5 Taxa de aprendizagem 49
3.3.6 Parada antecipada do treinamento – *Early stopping* 50

3.3.7 Termo Momentum 51

4 Estatística utilizada no trabalho 53
4.1 Teste de Kolmogorov-Smirnov 54
4.2 Teste de Mann-Whitney 56

5 Conceitos matemáticos envolvidos na pesquisa 59
5.1 Produto interno e distância euclidiana de vetores no espaço R^n 59
5.2 Ângulo e ortogonalidade entre vetores no espaço R^n 60

6 Discussão matemática do uso de alvos ortogonais em redes Multilayer Perceptron 62
6.1 Definição de vetores-alvo 62
6.2 Algoritmo de geração de vetores bipolares ortogonais 63
6.3 Observações sobre os vetores-alvo 67
6.4 O algoritmo *backpropagation* e os efeitos da distância euclidiana no desempenho da rede – discussão matemática 69

7 A redução da suscetibilidade ao erro de classificação de redes Multilayer Perceptron com o uso de alvos ortogonais 74
7.1 Dados experimentais 74
7.1.1 Dígitos manuscritos 74
7.1.2 Íris humana 75
7.1.3 Signos da linguagem australiana de sinais 76
7.2 Planejamento experimental e estatístico 78
7.3 Resultados experimentais 81
7.4 Discussão 90

8 O comportamento do desempenho de redes Multilayer Perceptron com o uso de alvos ortogonais 92
8.1 Vetores-alvo usados nos experimentos 92
8.2 Procedimento experimental 93
8.2.1 Parâmetros analisados 93
8.2.2 Topologia e taxa de aprendizagem inicial 95
8.2.3 Planejamento estatístico 96
8.3 Resultados experimentais 96
8.4 Discussão 105

9 A robustez de redes do tipo MLP com a utilização de VBOs 107
9.1 Robustez, a topologia e a taxa de aprendizagem inicial 107
9.2 Procedimento experimental 108
9.2.1 Planejamento experimental e estatístico 108
9.3 Resultados experimentais 112
9.4 Discussão 117
9.5 Conclusão 119

10 Publicações do Trabalho 120

11 Conclusão **121**

Referências Bibliográficas **123**

Lista de Figuras

1.1 Figura ilustrativa da originalidade do trabalho - parte 1 19
1.2 Figura ilustrativa da originalidade do trabalho - parte 2 20

2.1 Exemplo de separador de classes desejado 22
2.2 Exemplo de separador de classes próximo do ideal 23
2.3 Ilustração das etapas em um sistema de reconhecimento de padrões - Fonte: (Duda, Hart, & Stork, 2001) 24
2.4 Ilustração das principais técnicas de reconhecimento de padrões 26
2.5 Ilustração de um neurônio biológico - Fonte: (Silva, Spatti, & Flauzino, 2010) . 28
2.6 Ilustração de um neurônio artificial - Fonte: (Silva et al., 2010) 30
2.7 Arquitetura de redes de camadas simples - Fonte: (L. V. Fausett & Hall, 1994) . 31
2.8 Função lógica OU 32
2.9 Função lógica XOR 33
2.10 Arquitetura de redes multicamadas com realimentação - Fonte: (Silva et al., 2010) 33
2.11 Arquitetura de rede em reticulado - Fonte: (Silva et al., 2010) 34
2.12 Função de ativação degrau - Fonte: (Silva et al., 2010) 35
2.13 Função de ativação degrau bipolar - Fonte: (Silva et al., 2010) 36
2.14 Função de ativação degrau rampa - Fonte: (Silva et al., 2010) 37
2.15 Função de ativação logística binária - Fonte: (Silva et al., 2010) 37
2.16 Função de ativação logística bipolar - Fonte: (Silva et al., 2010) 38
2.17 Função de ativação tangente hiperbólica bipolar - Fonte: (Silva et al., 2010) . . . 38
2.18 Função de ativação gaussiana - Fonte: (Silva et al., 2010) 39

3.1 Arquitetura de uma rede do tipo MLP - Fonte: (Haykin, 2008) 40
3.2 Função tangente hiperbólica ideal 47
3.3 Função logística padrão 47
3.4 Ilustração da regra do *early stopping* 51

4.1 Esboço da região de aceitação e de rejeição para o Teste de Mann-Whitney . . . 57

5.1 Ângulo entre vetores 61
5.2 Ilustração de vetores ortogonais 61

6.1 Distância euclidiana de vetores-alvo 68
6.2 Ilustração de regiões de convergência 72
6.3 Ilustração de regiões de convergência com alvos distantes 73

7.1 Esquema de determinação da Média do Tipo 3 82

7.2 Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - dígitos manuscritos 84
7.3 Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - íris humana 85
7.4 Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - signos australianos 85

8.1 Ilustração dos parâmetros analisados 95
8.2 Desempenho máximo global 100
8.3 Desempenho obtido após o primeiro ciclo 101
8.4 Desempenho médio obtido nos cinco primeiros ciclos 101
8.5 Ponto de parada *overfitting* 101
8.6 Desempenho obtido no ponto de parada 102
8.7 Desempenho máximo obtido antes do ponto de parada 102
8.8 Desempenho máximo obtido depois do ponto de parada 102
8.9 Desempenho médio obtido antes do ponto de parada 103
8.10 Desempenho médio obtido depois do ponto de parada 103
8.11 Média de desempenho em todos os ciclos - dígitos manuscritos 104
8.12 Média de desempenho em todos os ciclos - íris humana 104
8.13 Média de desempenho em todos os ciclos - signos australianos 105

9.1 Combinações de parâmetros de treinamento para dígitos manuscritos 108
9.2 Combinações de parâmetros de treinamento para íris humana 109
9.3 Combinações de parâmetros de treinamento para signos australianos 109
9.4 Média de desempenho para todos os ciclos e todas as combinações de parâmetros 111
9.5 Média de desempenho para todos os ciclos e todas as combinações de parâmetros 113
9.6 Coeficientes de variação dos desempenhos coletados durante os treinamentos . . 114
9.7 Desempenho no ciclo 1 115
9.8 Desempenho no ciclo 10 115
9.9 Desempenho no ciclo 20 115
9.10 Desempenho no ciclo 30 117
9.11 Desempenho no ciclo 40 117
9.12 Desempenho no ciclo 50 117

Lista de Tabelas

4.1 Ilustração de Tabela de cálculo para o valor de D_n 55

7.1 Parâmetros obtidos com algoritmo genético - dígitos manuscritos 79
7.2 Parâmetros obtidos com algoritmo genético - íris humana 79
7.3 Parâmetros obtidos com algoritmo genético - signos australianos 80
7.4 Média do Tipo 3 - dígitos manuscritos 83
7.5 Média do Tipo 3 - íris humana 83
7.6 Média do Tipo 3 - signos australianos 84
7.7 Teste para normalidade de Kolmogorov-Smirnov - dígitos manuscritos 86
7.8 Teste para normalidade de Kolmogorov-Smirnov - íris humana 86
7.9 Teste para normalidade de Kolmogorov-Smirnov - signos australianos 86
7.10 Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - dígitos manuscritos 87
7.11 Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - íris humana 88
7.12 Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - signos australianos 89
7.13 Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - dígitos manuscritos 90
7.14 Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - íris humana 90
7.15 Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - signos australianos 90

8.1 Parâmetros de treinamento da MLP indicados pelo algoritmo genético 96
8.2 Teste estatístico de Mann-Whitney para a comparação de médias - dígitos manuscritos 97
8.3 Médias de desempenho obtidas com o uso de VBC10, VNO16 e VBO16 - dígitos manuscritos 97
8.4 Comparação simultânea das médias - dígitos manuscritos 98
8.5 Teste estatístico de Mann-Whitney para a comparação de médias - íris humana . 98
8.6 Médias de desempenho obtidas com o uso de VBC71, VNO128 e VBO128 - íris humana 99
8.7 Comparação simultânea das médias - íris humana 99
8.8 Teste estatístico de Mann-Whitney para a comparação de médias - signos australianos 99

8.9 Médias de desempenho obtidas com o uso de VBC95, VNO128 e VBO128 - signos australianos 100
8.10 Comparação simultânea das médias - signos australianos 100

9.1 Número de experimentos com desempenho superior a 70% - dígitos manuscritos 116
9.2 Número de experimentos com desempenho superior a 70% - íris humana 116
9.3 Número de experimentos com desempenho superior a 70% - signos australianos 116

Lista de Siglas

RNA Rede neural artificial
MLP Multilayer Perceptron
VBO Vetor bipolar ortogonal
VBN Vetor binário
VBC Vetor bipolar convencional
VNO Vetor não ortogonal
RP Reconhecimento de padrão
VC Vetor convencional
AG Algoritmo Genético
RBF Radial Basis Function

Capítulo 1

Introdução

1.1 Considerações iniciais

Segundo (Kruse et al., 2013) é atribuído a inteligência computacional, o conjunto de técnicas computacionais cuja abordagem tem inspiração na natureza, capazes de resolver problemas complexos nos quais as abordagens tradicionais são ineficazes. Essas técnicas utilizam o poder de processamento dos computadores em cálculos de operações matemáticas, para permitir que os algoritmos consigam desempenhar tarefas de classificação de padrões, de aproximação de funções, de previsão de séries temporais e de otimização. Cada ferramenta tem um processo de aprendizagem computacional, na qual o sistema se apropria de informações dos dados para conseguir "aprender". Essa importante área da computação tem ganhado grande destaque no meio científico graças à sua capacidade de resolver problemas complexos. A aprendizagem profunda (*Deep Learning*) é um dos exemplos que elucidam essa realidade. Resultados promissores em problemas de recuperação de imagem têm sido obtidos por meio da aplicação de Redes Neurais Convolucionais (Wan et al., 2014). A utilização de Máquinas de Vetores de Suporte e Regressão Logística permite aplicações diversas, como a detecção de transtorno depressivo (Shimizu et al., 2015). Utiliza-se a aprendizagem computacional baseada na inferência bayesiana para a predição de informações relacionadas às proteínas e suas interações (Birlutiu, d'Alche Buc, & Heskes, 2014). Outra aplicação recente é a separação entre ruído e fala em

sons de microfones (Healy, Yoho, Wang, & Wang, 2013). Também há importantes resultados de aplicação da inteligência computacional na área forense (Muda, Choo, Abraham, & Srihari, 2014). Parte desse conjunto de técnicas já foi usada com sucesso na detecção de pedestres aptos a atravessarem ruas, com o objetivo de se evitarem acidentes (Xu et al., 2012).

As redes neurais artificiais (RNAs) constituem um dos tipos de ferramentas da inteligência computacional mais conhecidas, amplamente utilizadas e reconhecidas pela comunidade científica. Sua aplicação em problemas de reconhecimento de padrões (RP) é bastante consolidada. Nesse contexto há várias aplicações de RNAs relacionadas ao reconhecimento de padrões (Wan et al., 2014) (Birlutiu et al., 2014) (Muda et al., 2014) (Xu et al., 2012) (Huang, Huang, Song, & You, 2015).

Existem vários modelos de RNAs, e um dos mais difundidos é o de redes Multilayer Perceptrons (MLP). Alguns exemplos mais recentes de aplicações desse tipo de RNA que poderiam ser destacados são: detecção de doenças dos movimentos do olhos (Jesús, Ortiz-Rodriguez, Mariaca-Gaspar, & Tovar, 2013); predição de risco de terremoto (Ashwini, Devi, & Gangashetty, 2012); detecção de falhas em rolamentos do motor de indução (Zarei, 2012); reconhecimento de carta de controle (Ranaee & Ebrahimzadeh, 2013); reconhecimento de face (Danisman, Bilasco, Martinet, & Djeraba, 2013); classificação de sinais de eletromiografia (Sahin & Sahin, 2012); classificação de fonemas (Sivaram & Hermansky, 2012) e solução de equações diferenciais (Rudd & Ferrari, 2015).

1.2 Motivação e originalidade

As redes do tipo MLP são amplamente utilizadas em diversas aplicações, conforme mencionado anteriormente. Contudo, estudos apontam que essa ferramenta apresenta alguns inconvenientes em tarefas de classificação de padrões. Um deles é a longa duração da fase de treinamento (Hamedi, Salleh, Astaraki, Noor, & Harris, 2014) que requer um grande esforço computacional (Mikolov, Kombrink, Burget, Cernocky, & Khudanpur, 2011) (C.-H. Lee, Chang, Kuo, & Chang, 2012). Outro problema é a determinação dos parâmetros de treina-

mento, como a taxa de aprendizagem, o tamanho da camada oculta e a escolha dos pesos sinápticos iniciais (Lawrence, Burns, Back, Tsoi, & Giles, 2012). Esses parâmetros têm impacto significativo no desempenho das redes do tipo MLP. Existem muitos estudos que apontam metodologias para determinação dos parâmetros (Isa & Mamat, 2011) (Sivaram & Hermansky, 2012) (Samal, Panda, & Das, 2015) (Kim, 2005) (Wang, Chang, & Du, 2002). No entanto, não existe um método exato para uma boa escolha de parâmetros.

Há vários estudos destinados à melhoria do desempenho de MLPs no reconhecimento de padrões. Algumas investigações têm como objeto de estudo o algoritmo de treinamento (Kim, 2005) (C.-M. Lee, Yang, & Ho, 2006) (Castro & Braga, 2013). Também há estudos que buscam a determinação da topologia ideal da MLP (Costa, Braga, & de Menezes, 2003) (Samal et al., 2015). Outra proposta é uma nova metodologia de escolha dos pesos sinápticos iniciais (Wang et al., 2002). Podem-se destacar resultados obtidos por meio da detecção de sensibilidade do sinal para evitar a interferência de ruídos no desempenho da MLP (C.-M. Lee et al., 2006). A essas pesquisas, somam-se contribuições como a adição do termo de regularização esparso para o custo de entropia cruzada e atualização dos parâmetros da rede para minimizar o custo conjunto (Sivaram & Hermansky, 2012), a adoção do termo de regularização para a determinação dos pesos de saída (Iosifidis, Tefas, & Pitas, 2015) e a implementação de sistemas híbridos com o uso de funções RBF (Isa & Mamat, 2011).

Um estudo que avaliou as metodologias usadas na competição para o reconhecimento de dígitos manuscritos (Cireşan, Meier, Gambardella, & Schmidhuber, 2012) aponta que os melhores resultados conseguidos nessa competição foram obtidos com o aumento do número de camadas ocultas, de neurônios das camadas ocultas, de dados deformados para evitar o fenômeno do *overfitting* e pelo efeito do uso de poderosas placas de processamento gráfico para aumentar a velocidade de processamento computacional.

Dentre as pesquisas dedicadas a melhoraria do desempenho de redes do tipo MLP, foram encontradas apenas duas pesquisas com foco no espaço de saída. Uma delas, analisa o comportamento de funções lineares e não lineares para a delimitação de fronteiras (Ruck, Rogers, Kabrisky, Oxley, & Suter, 1990). A outra pesquisa, indica a utilização do gradiente

para fornecer o limiar para a função discriminante linear na camada de saída. Cada padrão de entrada é utilizado para determinação de um discriminante linear (Hwang, Choi, Oh, & Marks, 1991). Dessa forma, está claro que a maioria das pesquisas, utiliza abordagens voltadas para o processamento dos padrões de entrada, para o algoritmo de treinamento, para a otimização de parâmetros de treinamento, para a topologia e para a hibridização com outros sistemas.

Alguns estudos preliminares publicados pelo autor e pelos orientadores mostram que o uso de vetores bipolares ortogonais (VBOs) como vetores-alvo em redes do tipo MLP melhora a habilidade de generalização da RNA (Nomura, Yamanaka, Katai, Kawakami, & Shiose, 2005) (Nomura, Yamanaka, Katai, Kawwakami, & Shiose, 2004) (Manzan, Yamanaka, & Nomura, 2011) (Hallinan, 1991) (Manzan, Nomura, Yamanaka, Carneiro, & Veiga, 2012a) (Manzan, Nomura, & Filho, 2014) (Manzan, Nomura, & Yamanaka, 2011a) (Manzan, Nomura, & Yamanaka, 2011b) (Nomura, Manzan, & Yamanaka, 2011) (Manzan, Nomura, & Yamanaka, 2012). Esses estudos mostram uma melhoria no desempenho de classificação em problemas com padrões complexos muito degradados e com grande quantidade de ruídos. A utilização de vetores-alvo do tipo VBO implica aumento da distância euclidiana entre pontos no espaço de saída. Por causa da ortogonalidade, eles têm a maior distância euclidiana possível um do outro. As figuras 1.1 e 1.2 ilustram a originalidade da proposta deste trabalho.

Este trabalho realiza uma discussão matemática sobre o desempenho de redes do tipo MLP treinadas com alvos convencionais e alvos ortogonais. Também realiza análises experimentais detalhadas do seu desempenho por meio da utilização de dígitos manuscritos do repositório internacional UCI Machine Learning (Lichman, 2013). Além da análise de a propensão a erros de classificação, também foram analisados o desempenho geral, o desempenho nos primeiros ciclos de treinamento, antes e depois do ponto de *overfitting*.

1.3 Objetivo da tese

Para a definição do objetivo deste trabalho, três aspectos mencionados anteriormente são levados em consideração. O primeiro deles é a importância das RNAs do tipo MLP em

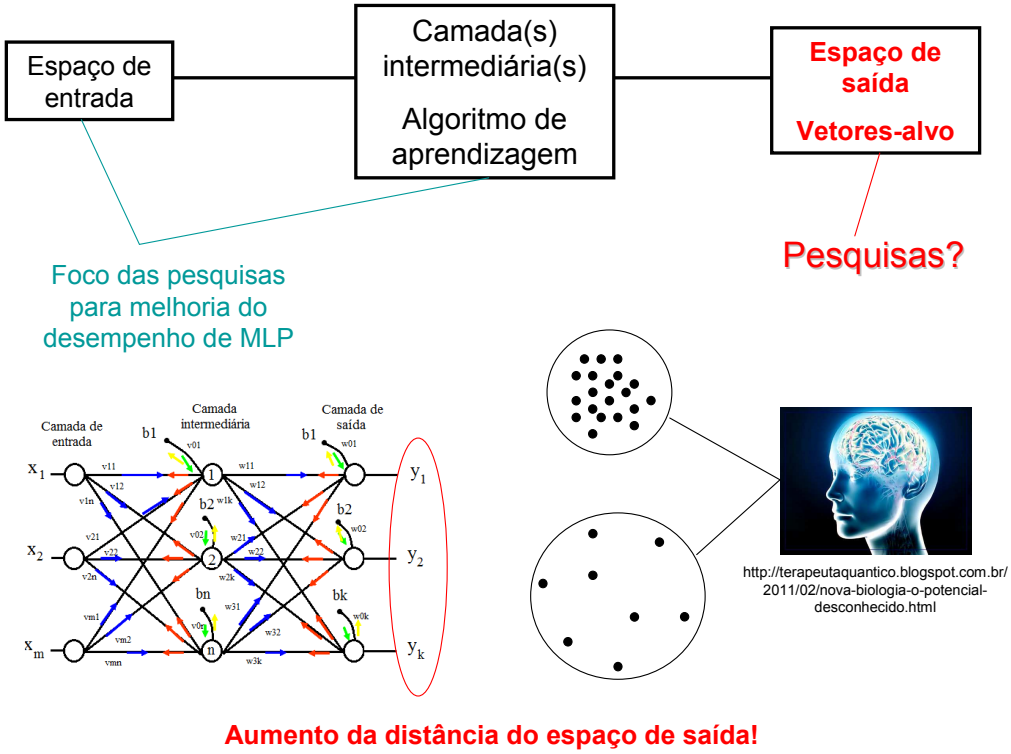


Figura 1.1: Figura ilustrativa da originalidade do trabalho - parte 1

tarefas de RP. O segundo consiste na sua ampla utilização em diversos tipos de aplicações. O terceiro são as dificuldades existentes na utilização dessa ferramenta, tanto no que diz respeito às longas durações na etapa de treinamento, quanto no esforço computacional requerido em algumas aplicações e na definição dos parâmetros de treinamento. Nesse sentido, o objetivo deste trabalho é possibilitar a utilização de redes do tipo MLP em problemas variados de RP com redução do tempo de treinamento e do custo computacional e com menor suscetibilidade a variações de parâmetros.

1.4 Estrutura do trabalho

No Capítulo 2 são apresentados fundamentos teóricos de RP e RNAs. O Capítulo 3 apresenta os fundamentos teóricos de RNAs do tipo MLP. No Capítulo 4 são abordados os fun-

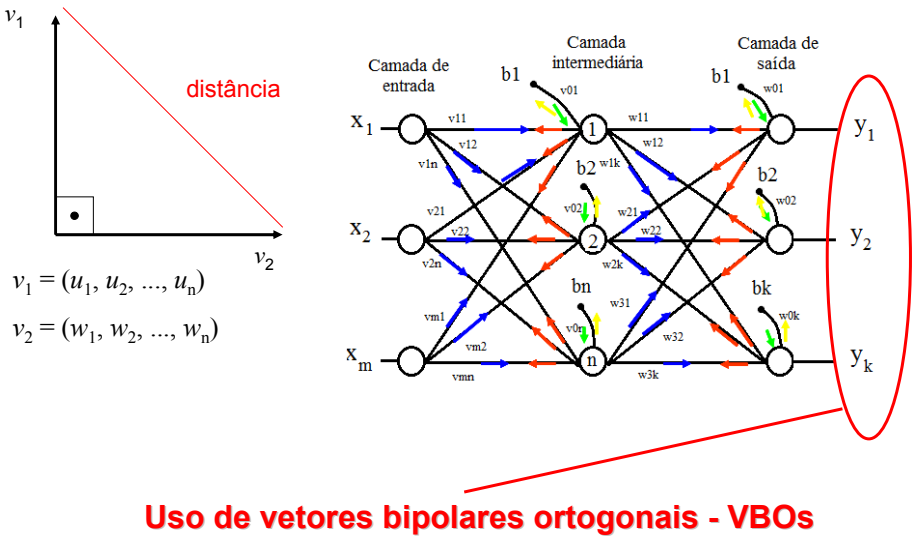


Figura 1.2: Figura ilustrativa da originalidade do trabalho - parte 2

damentos matemáticos envolvidos na pesquisa. Os métodos estatísticos utilizados no trabalho são abordados no Capítulo 5. No Capítulo 6 é realizada uma discussão matemática do uso de vetores-alvo em RNAs do tipo MLP. A discussão experimental sobre a redução da suscetibilidade ao erro de classificação de redes do tipo MLP com o uso de alvos ortogonais é apresentada no Capítulo 7. O Capítulo 8 aborda o comportamento do desempenho de redes do tipo MLP com o uso de alvos ortogonais. Os próximos passos propostos para o projeto de pesquisa são apresentados no Capítulo 9. O Capítulo 10 apresenta a conclusão deste trabalho.

Capítulo 2

Fundamentos teóricos de Reconhecimento de Padrões e Redes Neurais Artificiais

2.1 Fundamentos de reconhecimento de padrões

2.1.1 Conceitos de um sistema de reconhecimento de padrões

Entende-se por RPs o conjunto de técnicas capaz de separar objetos em conjuntos ou classes. Um padrão é a descrição quantitativa ou qualitativa de um objeto ou de outra entidade de interesse em uma imagem ou em um sinal (Gonzalez, 1992). Essa descrição pode ser feita por uma ou mais medidas que são denominadas atributos ou características do padrão. Um conjunto de padrões com características semelhantes é denominado classe (Gonzalez, 1992).

Uma técnica de reconhecimento de padrões tem por concepção a seleção de características de conjuntos de objetos e a separação dos objetos em suas devidas classes. De acordo com Duda et al. (2001), são técnicas que permitem uma representação mais simples de uma coleção de dados por meio das características que apresentam maior relevância, resultando na partição em classes. Essa representação é geralmente dada pela reunião das características em um vetor.

Contudo as técnicas de reconhecimento de padrões nem sempre são simples. Em

um dado problema, simples ou complexo, busca-se sempre chegar próximo de seu separador ideal. Em boa parte dos problemas de classificação, a determinação do separador desejado exige muito esforço, tanto na determinação dos parâmetros da técnica de RP utilizada, quanto no custo computacional. Na prática, a determinação de um separador desejado é quase sempre inviável. A Figura 2.1 representa um reconhecedor desejado capaz de separar amostras de duas classes com 100% de acerto. Alguns problemas toleram certo grau de erro. Dessa maneira, é possível encontrar um separador com um esforço viável que consiga separar os elementos das classes com um bom nível de acerto. A Figura 2.2 representa um reconhecedor próximo do desejado, capaz de separar duas classes com alguns erros de separação.

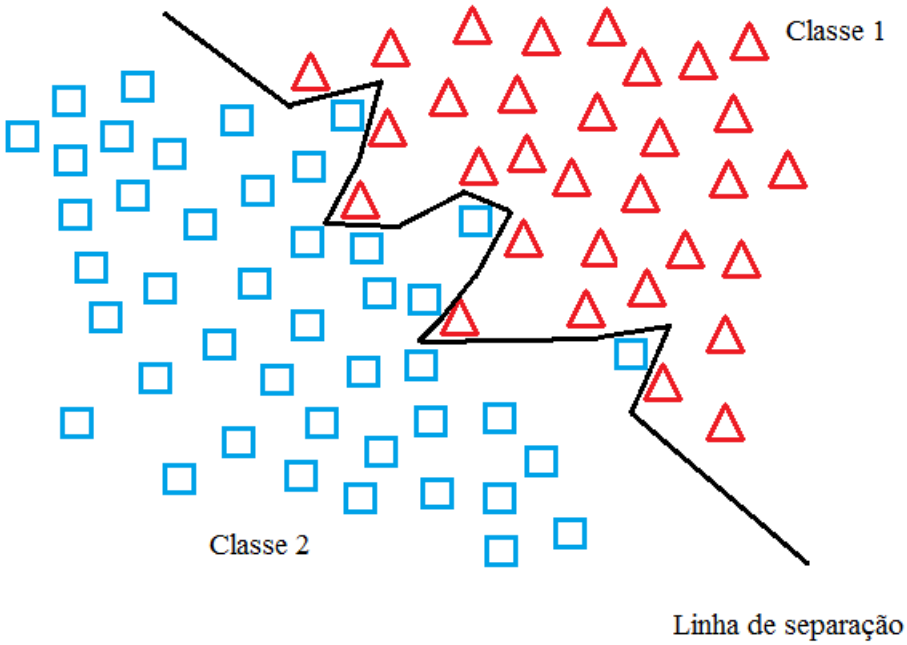


Figura 2.1: Exemplo de separador de classes desejado
Fonte: (Duda et al., 2001)

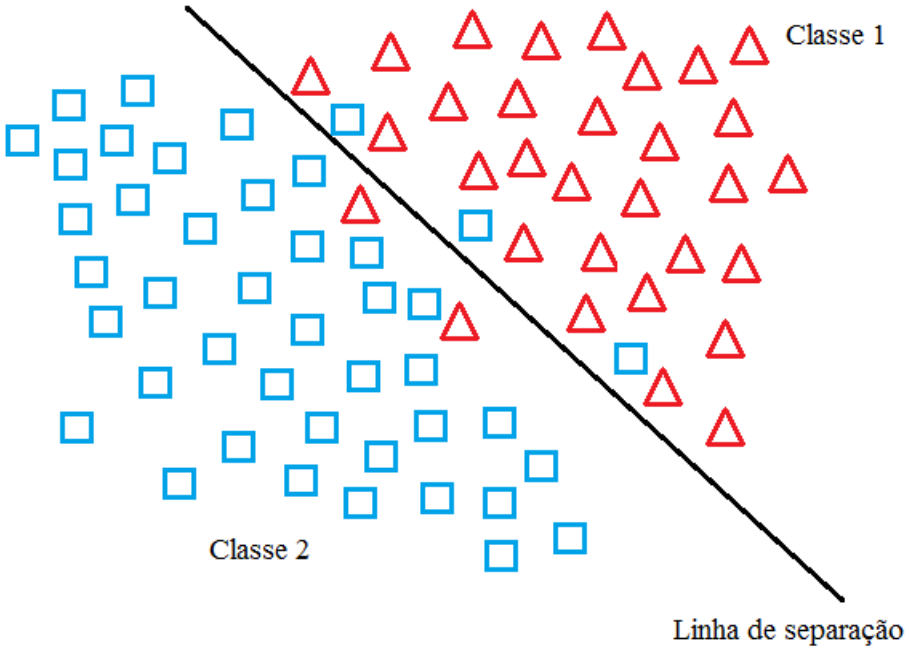


Figura 2.2: Exemplo de separador de classes próximo do desejado
Fonte: (Duda et al., 2001)

2.1.2 Etapas de um sistema de reconhecimento de padrões

De acordo com Duda et al. (2001), os sistemas de RPs seguem normalmente um conjunto de etapas, como é sugerido pela Figura 2.3.

- A primeira etapa consiste na aquisição das características dos padrões de treinamento. As técnicas de aquisição são inúmeras e sua escolha depende principalmente do tipo de aplicação que está sendo trabalhada.
- A segunda etapa consiste na segmentação que nada mais é do que a separação dentro das amostras das informações que têm relevância para o sistema das informações consideradas irrelevantes. Essa é uma das etapas mais complexas do sistema de RPs (Duda et al., 2001).
- Na etapa de extração de características, que é a terceira, de acordo com o modelo proposto, são formados os conjuntos de medidas ou valores que definirão as classes de objetos. Um bom processo de extração de características poderá facilitar imensamente a tarefa do classificador. Uma boa extração é aquela capaz de resumir ao

máximo as características relevantes e de fácil extração.

- A quarta etapa consiste na classificação dos padrões e pode ser simples ou complexa, dependendo da natureza dos dados. Grandes diferenças entre objetos de uma mesma classe, aliadas a uma pequena diferença entre classes diferentes, podem fazer com que essa etapa tenha um alto grau de dificuldade. Isso justifica a importância da execução correta das etapas anteriores. Note que é muito importante a escolha correta do tipo de classificador para o problema em questão (Duda et al., 2001).
- A quinta etapa corresponde ao pós-processamento.
- A sexta etapa, inexistente em alguns sistemas, corresponde à tomada de decisão a partir da tarefa de classificação.

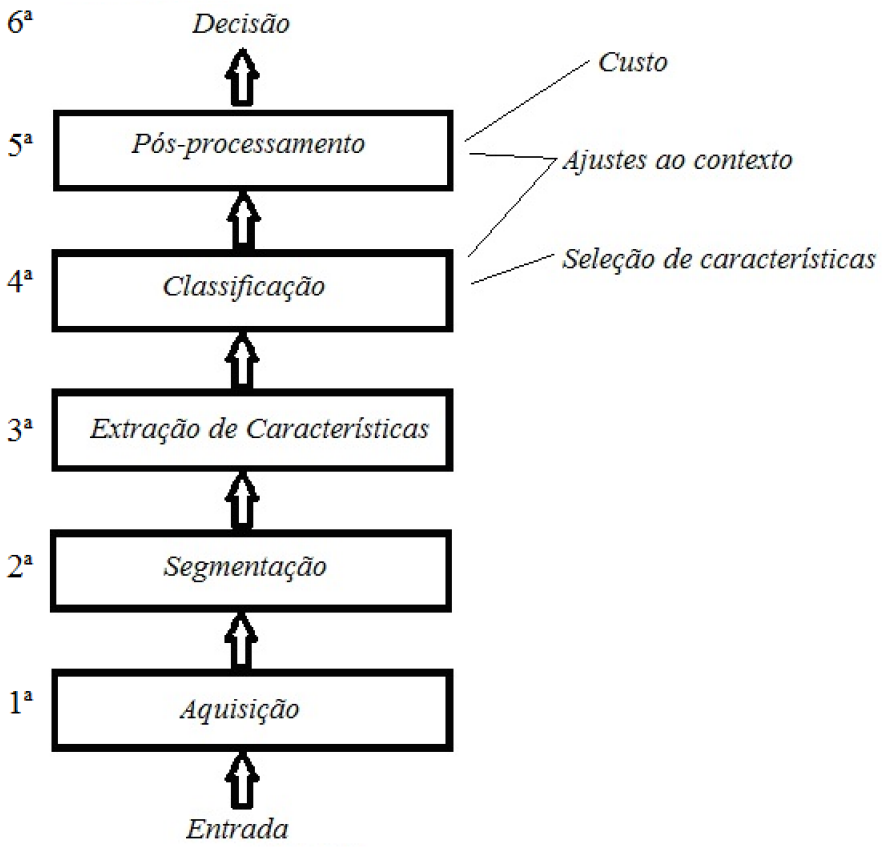


Figura 2.3: Ilustração das etapas normalmente envolvidas em um sistema de reconhecimento de padrões - Fonte: (Duda et al., 2001)

2.1.3 Técnicas para classificação de padrões

Existem dois grandes grupos de técnicas para classificação de padrões: técnicas com supervisão e técnicas sem supervisão. Quando as classes de um conjunto de dados já são conhecidas, a técnica de RPs deve apontar a saída esperada para cada padrão de treinamento. Assim, o conjunto de dados representa exemplos que farão o treinamento do sistema por experiência. No caso da classificação não supervisionada, as classes não estão bem definidas, e o próprio sistema agirá no sentido de separar essas classes, levantando as características mais evidentes.

Existe ainda uma etapa que poderia ser acrescentada ao modelo proposto na Figura 2.3 que consiste na validação do sistema. Essa etapa é importante para verificar se os resultados obtidos atendem à aplicação em questão. Na validação, os dados de teste são submetidos ao classificador a fim de se avaliar preliminarmente o desempenho do sistema, ainda que de forma super estimada. Caso a validação seja bem sucedida, pode-se passar para a etapa de classificação, em que o sistema será capaz de mapear objetos da mesma natureza nos dados de treinamento.

Dentre as técnicas de reconhecimento de padrões mais utilizadas, destacam-se a abordagem estatística (paramétrica e a não paramétrica) e a abordagem conexionista, que utiliza RNAs. No caso da abordagem estatística, o conjunto de dados de treinamento serve para realizar a estimação de parâmetros estatísticos de cada classe. Dessa maneira, cada classe terá uma distribuição específica, formando, assim, o classificador estatístico (Duda et al., 2001).

A abordagem estatística não paramétrica é subdividida em várias técnicas. Algumas delas se dão com a utilização de uma função de distância do objeto a ser mapeado em relação às classes disponíveis no espaço de características, direcionando padrões desconhecidos àquela classe que detém a menor distância com ele. As principais representantes das técnicas de distância são a distância euclidiana e a distância de Mahalanobis. A distância de Mahalanobis apresenta a vantagem de considerar a matriz de covariância de todas as classes além da média aritmética. Assim, problemas que contenham classes com diferentes variâncias serão melhor classificados com a distância de Mahalanobis. Dentro das técnicas não paramétricas, existem

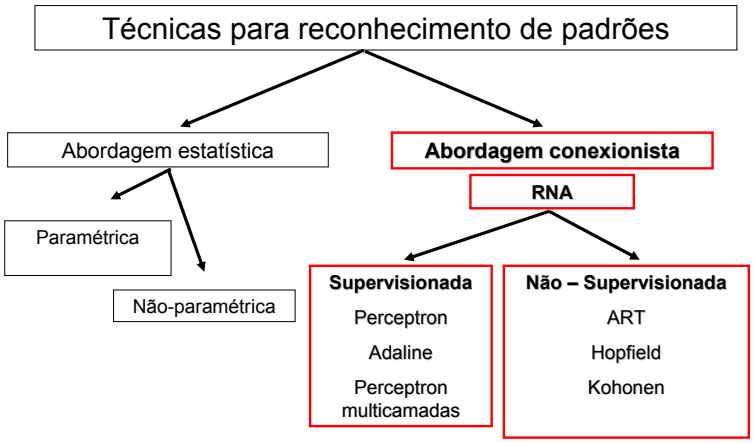


Figura 2.4: Ilustração das principais técnicas de reconhecimento de padrões

ainda as funções de Kernel, os K-vizinhos mais próximos e os histogramas (Duda et al., 2001).

Na abordagem estatística paramétrica, destaca-se o classificador de Bayes, que considera a probabilidade de um objeto desconhecido pertencer a uma determinada classe, o que, de acordo com Duda et al. (2001), é um tipo ótimo de classificador, pois minimiza a probabilidade média de erro na classificação. Existem também a função de discriminação e a regra Näive de Bayes.

No caso da abordagem conexionista, utilizam-se as RNAs. Trata-se de tipos de classificadores mais complexos, capazes de se adaptarem a qualquer tipo de distribuição de dados. São baseados no funcionamento das estruturas neurais inteligentes que aprendem por meio de exemplos.

Trabalhos destinados ao estudo do reconhecimento de padrões, tanto no que se refere à descrição, quanto à classificação, vêm ganhando grande destaque no campo da computação. Atualmente, as redes neurais artificiais têm se tornado uma técnica amplamente utilizada em razão de resultados bastante satisfatórios e promissores (Duda et al., 2001). A Figura 2.4 resume as principais técnicas de reconhecimento de padrões.

2.2 Técnicas de Redes Neurais Artificiais

De acordo com Silva et al. (2010), o primeiro registro do surgimento das RNAs se deu com uma publicação de um artigo de McCulloch e Pitts em 1943 (McCulloch & Pitts, 1943). Essa nova área da computação tinha por concepção o funcionamento dos neurônios biológicos. Segundo Silva et al. (2010), em 1949, Hebb apresentou o primeiro método de treinamento para RNAs. Entre 1957 e 1958, Frank Rosenblat desenvolveu o primeiro neurocomputador e, no período de 1958 a 1962, criou uma grande classe de RNAs denominada como Perceptrons (L. V. Fausett & Hall, 1994).

Em 1960, Widrow e Hoff desenvolveram um novo tipo de RNA denominado ADALINE (Adaptive Linear Element), que, posteriormente, recebeu aperfeiçoamentos correspondentes à associação de múltiplas redes Adaline, resultando no nome MADALINE (L. V. Fausett & Hall, 1994). Os resultados obtidos nessas pesquisas motivaram vários pesquisadores a estudarem as RNAs, até que, em 1969, Minsky e Papert demonstraram matematicamente as limitações das redes constituídas de uma única camada, como o Perceptron e o Adaline. No clássico livro *Perceptrons - An Introduction to Computational Geometry*, eles usam um simples problema de lógica denominado "ou exclusivo" para mostrarem que essas redes eram incapazes de resolvê-lo (L. V. Fausett & Hall, 1994).

O trabalho de Minsk e Papert causou grande impacto entre os pesquisadores da neurocomputação, fazendo com que o interesse pela área ficasse bastante reduzido. Isso, de certa forma, ocasionou a ausência de novos resultados para a área por um longo período (Silva et al., 2010). Nesse período de relativa turbulência das pesquisas envolvendo RNAs, foi implementada a rede ART (Adaptive Ressonance Theory). Num trabalho de Grossberg, em 1980, foi realizada a formulação de mapas auto-organizáveis de Kohonen em 1982 e a proposta de redes recorrentes de Hopfield em 1982. A partir do trabalho de Hopfield, a neurocomputação voltou a receber a atenção dos pesquisadores.

Contudo a teoria das RNAs conseguiu realmente se estabelecer a partir da publicação do livro de Rumelhart, Hinton e Williams, *Parallel Distributed Processing*, em 1986. Nele, os

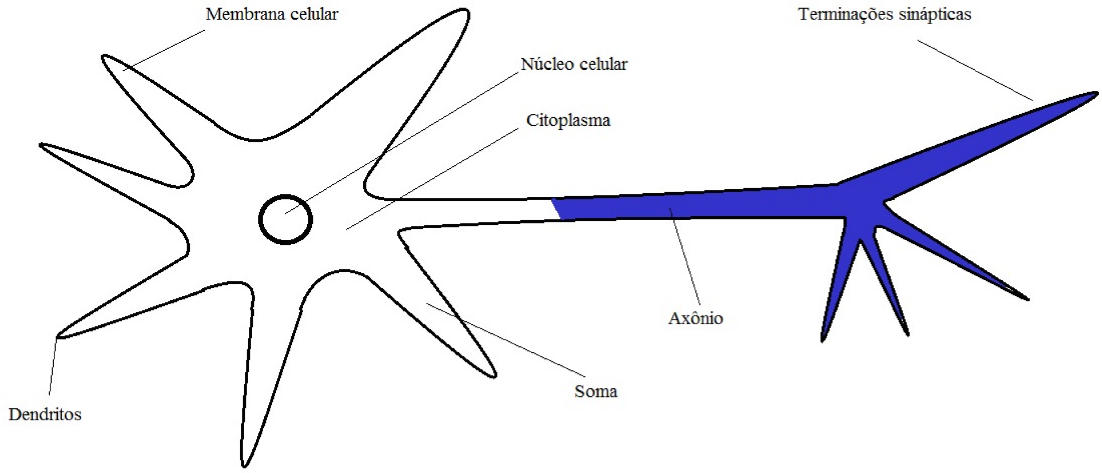


Figura 2.5: Ilustração de um neurônio biológico - Fonte: (Silva et al., 2010)

autores mostraram um algoritmo capaz de treinar redes com múltiplas camadas e que, por sua vez, eram capazes de resolver as limitações apresentadas por Minsk e Papert. Esses acontecimentos se deram no momento em que estavam sendo desenvolvidos computadores com maior capacidade de processamento (Silva et al., 2010).

2.2.1 Neurônio biológico

A Figura 2.5, montada com base na Figura 1.1 de Silva et al. (2010), ilustra um neurônio biológico, o qual é dividido em três partes principais: os dendritos, a soma ou corpo celular e o axônio.

Os dendritos captam sinais elétricos oriundos de outros neurônios ou mesmo do meio externo ao qual estão associados. O corpo celular processa a informação recebida, criando um potencial de ativação que, posteriormente, poderá ou não ser enviado ao axônio (Silva et al., 2010).

O axônio constitui-se de um único prolongamento que conduz impulsos elétricos para outros neurônios. No axônio existem ramificações denominadas como terminações sinápticas. O envio de informação das sinapses para os dendritos de outros neurônios ocorre por meio de substâncias neurotransmissoras, o que explica o fato de que entre neurônios distintos não existe ligação física. Essas substâncias neurotransmissoras, além de transmitir os impulsos elétricos,

realizam a ponderação da informação (L. V. Fausett & Hall, 1994).

O conjunto de bilhões dessas estruturas compõe a complexa estrutura denominada cérebro humano, que é capaz de realizar inúmeras tarefas com alto grau de dificuldade.

2.2.2 Neurônio artificial

Um neurônio artificial corresponde a um modelo bem simplificado do neurônio biológico. De modo semelhante ao que ocorre com os biológicos, os neurônios artificiais recebem a informação, processam-na de acordo com seu papel dentro da rede e enviam uma nova informação para outros neurônios ou para a saída do sistema (Silva et al., 2010). O modelo proposto por MucCulloch e Pitts tem em sua concepção de funcionamento o processamento paralelo da informação com alta conectividade. Esse tipo de modelo é ainda o mais utilizado nos modelos de RNAs. A Figura 2.6 representa um neurônio artificial e foi elaborada com base na Figura 1.5 proposta por Silva et al. (2010).

De forma análoga ao neurônio biológico, a informação chega ao neurônio com valores que são simbolizados pela variável x para uma das suas n entradas. Outra semelhança ocorre no fato de que o neurônio artificial também pondera a informação recebida por meio de valores denominados pesos sinápticos. Aqui esses valores estão simbolizados pela letra v para as n entradas do neurônio. Toda essa informação é reunida por meio de um somatório que processa junto a ela um valor denominado *bias* (L. V. Fausett & Hall, 1994). Após a reunião, essa informação recebe a ação da função de ativação, cuja existência também é análoga ao neurônio biológico. Finalmente, após a ação da ativação, o neurônio envia essa informação para outro neurônio ou para a saída.

2.2.3 Rede neural artificial

As abordagens clássicas da Inteligência Artificial trabalham com processamento sequencial. As redes neurais utilizam um modo de aprendizagem cujo processamento é distribuído e paralelo. Conforme o exposto anteriormente, sua metodologia de treinamento é inspirada no

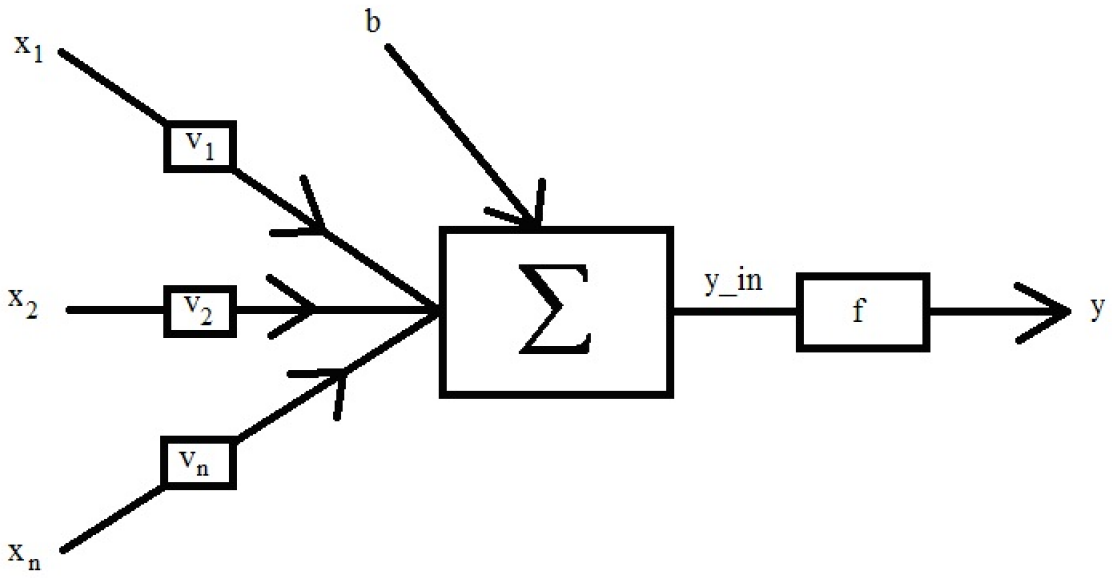


Figura 2.6: Ilustração de um neurônio artificial com base na Figura 1.5 de Silva et al. (2010) - Fonte: (Silva et al., 2010)

funcionamento dos neurônios biológicos, em que aprendizagem ocorre por meio de exemplos, fazendo com que a tentativa e o erro desencadeiem o processo de apropriação da habilidade de diferenciar padrões. RNAs realizam trabalho semelhante quando um grande número de neurônios envia sinais inibitórios ou excitatórios a outros neurônios da rede.

Basicamente, podem-se entender as RNAs como mecanismos capazes de receber o sinal de determinado padrão na sua entrada, analisá-lo e então informar sobre a classe a qual ele pertence. Isso é possível após o treinamento da rede, que pode acontecer de forma supervisionada ou não supervisionada em várias possibilidades de arquitetura e algoritmos de treinamento.

Há vários tipos de RNAs. Alguns possuem uma arquitetura mais, na qual há apenas duas camadas, sendo uma de entrada e outra de saída. Por outro lado, existem arquiteturas mais complexas, com a existência de três ou mais camadas. As camadas adicionais são conceituadas como intermediárias ou ocultas. Nessas redes, a camada de entrada tem a função do recebimento dos sinais advindos dos padrões de treinamento. A associação das características, bem como a separação das classes são feitas pelos neurônios da(s) camada(s) oculta(s). Fica a cargo da camada de saída a apresentação dos resultados finais da rede.

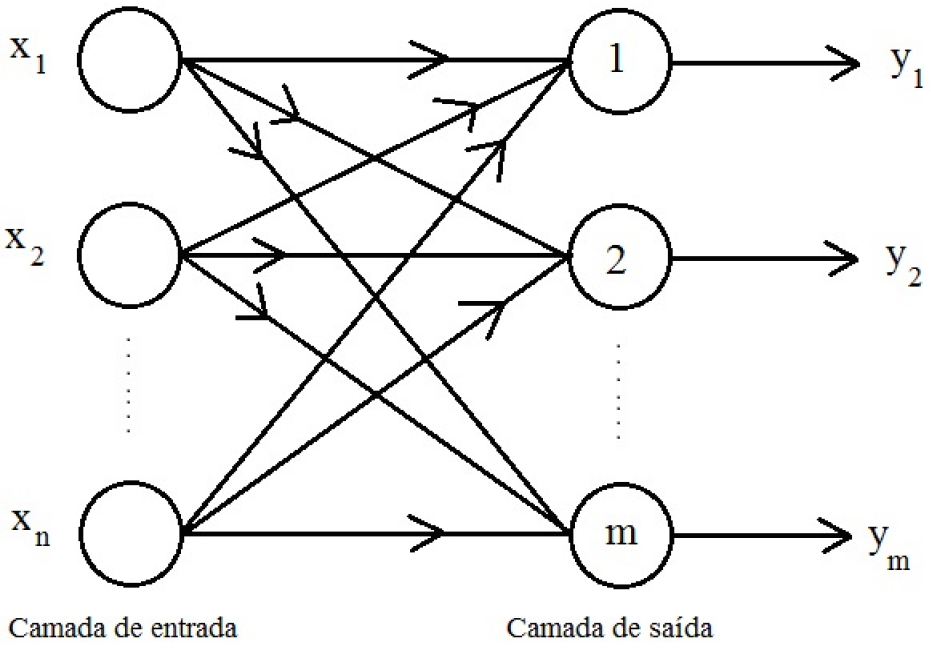


Figura 2.7: Arquitetura de redes de camadas simples - Fonte: (L. V. Fausett & Hall, 1994)

Nas RNAs mais simples, a camada de separação das classes é a própria camada de saída, como mostra a Figura 2.7. Esses tipos de redes são hábeis para problemas que têm classes linearmente separáveis. São representantes dessa abordagem as redes Perceptron e Adaline.

Nota-se um exemplo simples da função lógica “ou” na Figura 2.8. Considere-se que “1” seja o valor lógico para verdadeiro e que “-1” seja o valor lógico para falso. Essa operação é realizada sempre entre dois valores lógicos, que podem ser verdadeiro ou falso. O resultado dessa operação é sempre verdadeiro, exceto para o caso em que a operação “ou” seja realizada entre dois valores lógicos falsos, que, nesse caso, figuram como “-1”. Resultados verdadeiros estão representados em quadrado, e o único resultado falso está representado em um triângulo. Com uma reta, separam-se as duas classes, ou seja, os valores lógicos verdadeiros do valor lógico falso.

Entretanto, há problemas em que as classes não são linearmente separáveis. Dessa maneira, redes como a Perceptron e Adaline são incapazes de realizar a classificação. Isso foi mostrado por Minsk e Papert, já mencionado anteriormente. Considera-se a função lógica “ou exclusivo”, também denominada XOR. Diferentemente da operação lógica “ou” simples, essa

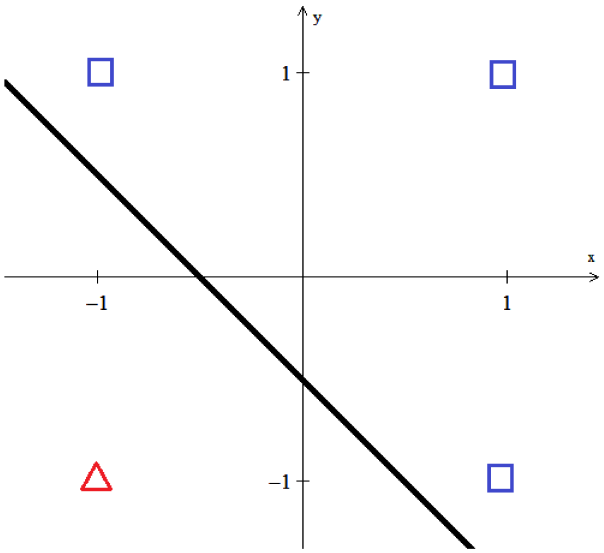


Figura 2.8: Representação da função lógica “ou” e a reta de separação das classes

função terá por resultado valores lógicos verdadeiros, quando os valores lógicos da operação forem diferentes. Quando os valores lógicos envolvidos pela operação forem iguais, o resultado será o valor lógico falso. Pode-se perceber, na Figura 2.9, que não é possível obter uma reta que separe a classe dos valores lógicos positivos da classe dos valores lógicos negativos.

Problemas dessa natureza requerem redes com mais de duas camadas, ou seja, redes que tenham pelo menos uma camada oculta. Os exemplos mais conhecidos de redes com múltiplas camadas são as redes Perceptron multicamadas, popularmente conhecidas como MLPs, e as redes de Hopfield, que usam o princípio da realimentação dos sinais da saída para a atualização dos pesos sinápticos. A Figura 2.10 ilustra a arquitetura desse tipo de rede com a indicação da propagação e retropropagação do sinal. A inclusão de uma ou mais camadas ocultas possibilita a separação não linear das classes. Isso resolve o problema do XOR e muitos outros de complexidade ainda maior. Como consequência, as redes com esse tipo de arquitetura tornaram-se as mais utilizadas pelos pesquisadores.

Em contrapartida, essas redes têm algoritmos bem mais complexos de serem implementados quando comparamos com as redes Perceptron e Adaline e exigem muito mais experiência e atenção do pesquisador para que produzam bons resultados.

Há também RNAs com uma estrutura denominada reticulada, cuja representante prin-

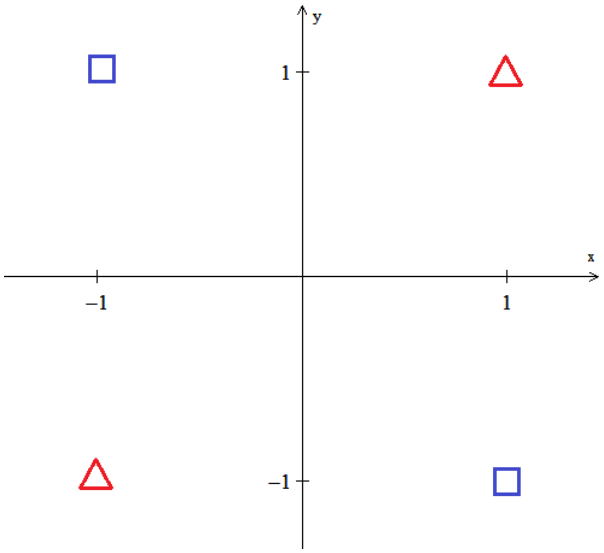


Figura 2.9: Representação da função lógica “ou exclusivo” (XOR)

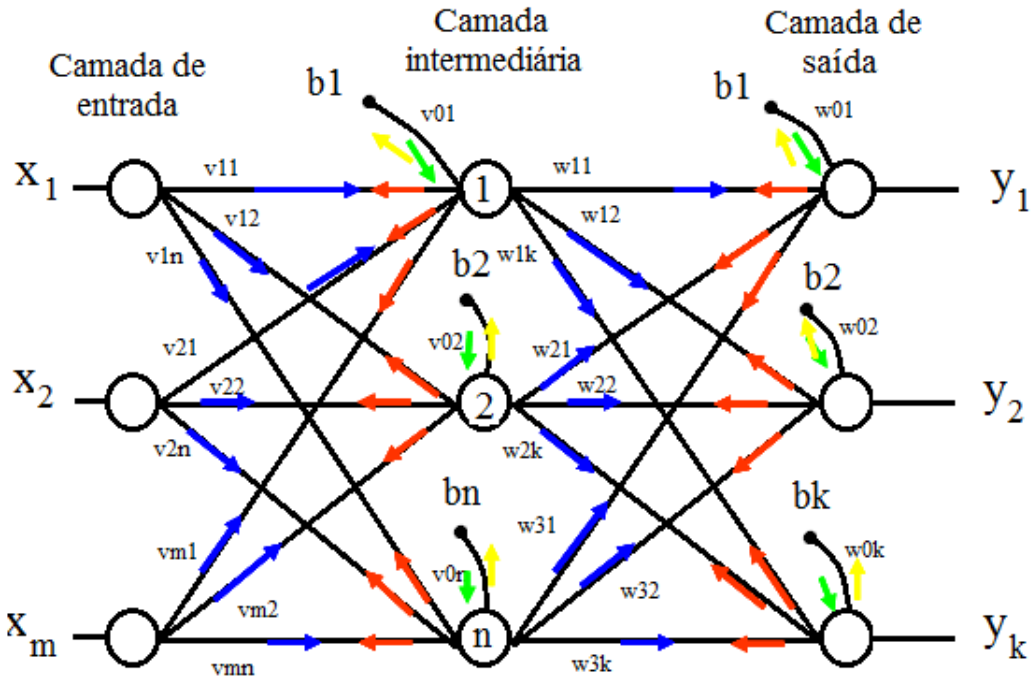


Figura 2.10: Arquitetura de redes multicamadas com realimentação - Fonte: (Silva et al., 2010)

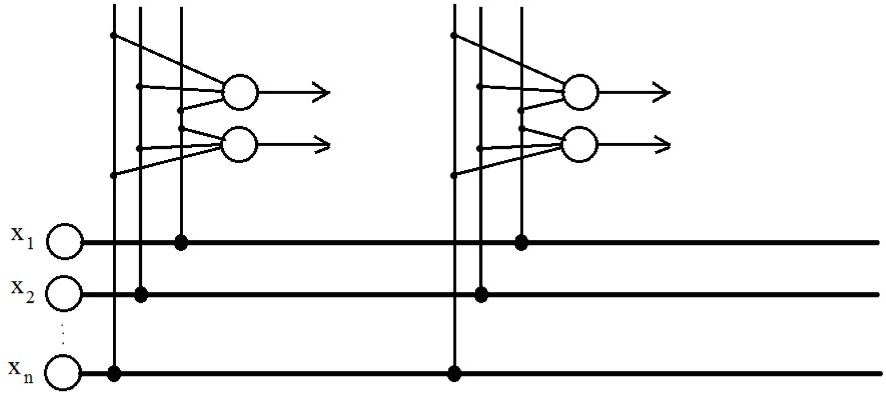


Figura 2.11: Arquitetura de rede em reticulado - Fonte: (Silva et al., 2010)

cipal é a rede de Kohonen (Silva et al., 2010). Nessa configuração, os neurônios ficam dispostos no espaço bidimensional, como ilustrado na Figura 2.11.

Além da arquitetura e dos tipos de RNAs existentes, um ponto de extrema importância é a forma de aprendizado das redes. Uma RNA pode ter seu treinamento executado de duas maneiras: supervisionada e não supervisionada, como citado anteriormente. No caso de redes supervisionadas, para cada padrão de treinamento associa-se uma saída esperada, ou um alvo a ser atingido. Por causa desse princípio, as saídas desejadas são denominadas vetores-alvo da rede.

Redes supervisionadas executam o treinamento de maneira a reduzir a diferença entre a saída obtida e a saída esperada para aquele padrão em cada passo da execução. Em outras palavras, diz-se, que a cada iteração, época ou ciclo, o treinamento procura reduzir o erro proveniente da comparação entre alvo e saída. Após atingir um erro tolerável, que dependerá de cada problema, o treinamento é encerrado. Perceptron, Adaline e Perceptron multicamadas são exemplos de redes supervisionadas.

Opondo-se ao supervisionado, no treinamento não-supervisionado não há saídas esperadas. A rede se auto-organiza, identificando durante o treinamento as similaridades entre as amostras. A partir dessas similaridades, subconjuntos são criados de maneira que os pesos sinápticos são ajustados com o intuito de propiciar a separação interna dos elementos de cada subconjunto. Representantes desse tipo de aprendizado são as redes de Hopfield e de Kohonen.

2.2.4 Funções de ativação

As funções de ativação são responsáveis por limitar o sinal da saída a um intervalo de interesse (Silva et al., 2010). Algumas funções têm imagem limitada ao intervalo real [0,1], enquanto outras ao intervalo real [-1,1]. Também são utilizadas funções com mais de uma sentença matemática quando o tipo de rede realiza a limiarização para o sinal de saída. O tipo de função de ativação depende do tipo de rede adotada. A Figura 2.12 representa o gráfico da função degrau da Equação 2.1. Essa função direciona saída "1" para valores líquidos maiores do que o limiar "d" estabelecido e saída "0" para outros valores líquidos.

$$f(x) = \begin{cases} 1 & \text{se} \quad x \geq d \\ 0 & \text{se} \quad x < d \end{cases} \tag{2.1}$$

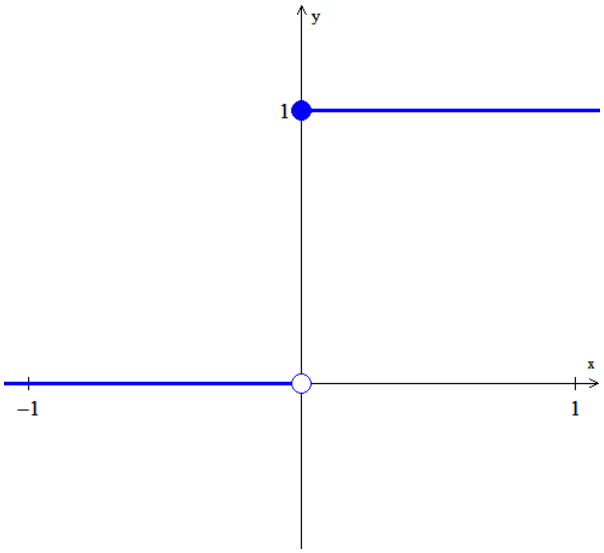


Figura 2.12: Função de ativação degrau - Fonte: (Silva et al., 2010)

Outra função definida por mais de uma sentença matemática é a função degrau bipolar, dada pela Equação 2.2 e representada pela Figura 2.13. Nessa função, valores líquidos menores do que o limiar "d" recebem saída "−1", e valores líquidos maiores ou iguais ao limiar recebem saída "1".

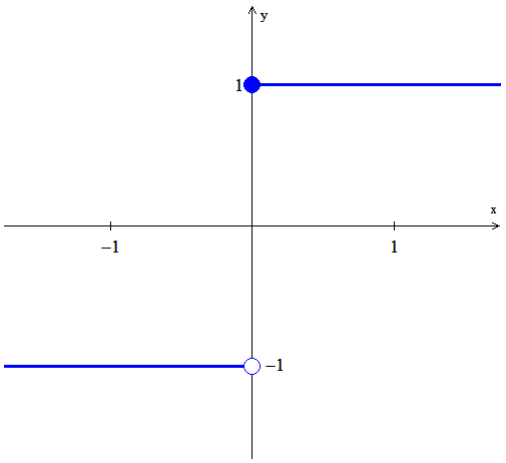


Figura 2.13: Função de ativação degrau bipolar - Fonte: (Silva et al., 2010)

$$f(x) = \begin{cases} 1 & \text{se} \quad x \geq d \\ -1 & \text{se} \quad x < d \end{cases} \tag{2.2}$$

Em alguns casos, é necessário utilizar outros intervalos de saída. A função rampa, dada pela equação 2.3, permite que, de acordo com a aplicação, se definam os limites do intervalo de saída. Além disso, essa função considera como saída da rede os próprios valores líquidos quando estão dentro dos limites estipulados. A Figura 2.14 representa o gráfico dessa função.

$$f(x) = \begin{cases} -a & \text{se} \quad x < a \\ x & \text{se} \quad -a \leq x \leq a \\ -a & \text{se} \quad x > a \end{cases} \tag{2.3}$$

As funções mostradas anteriormente não são diferenciáveis, ou seja, são deriváveis nos intervalos em que não ocorre o degrau ou a mudança de lei matemática, mas apresentam pontos de singularidade. As funções que serão mostradas a seguir são diferenciáveis em todo o intervalo real. A primeira função com essa característica é a logística binária, dada pela Equação 2.4. Essa função assume valores reais entre "0" e "1", e β interfere no nível de inclinação dessa função em relação ao ponto de inflexão, como mostra a Figura 2.15.

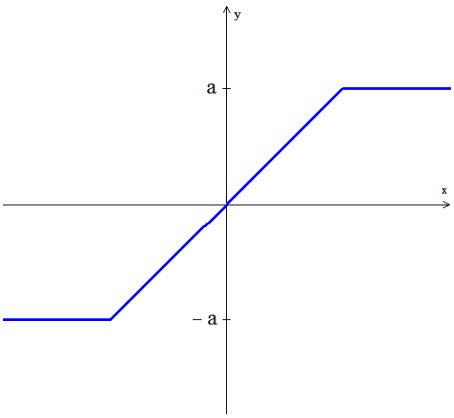


Figura 2.14: Função de ativação degrau rampa - Fonte: (Silva et al., 2010)

$$f(x) = \frac{1}{1 + e^{-\beta \cdot x}} \tag{2.4}$$

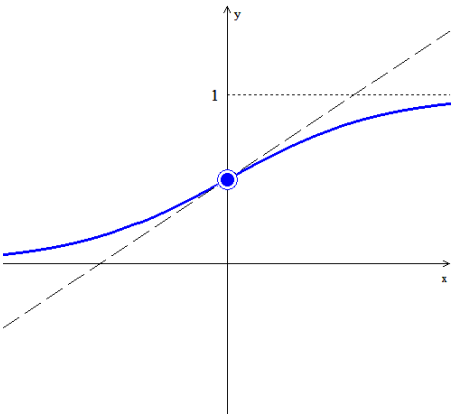


Figura 2.15: Função de ativação logística binária - Fonte: (Silva et al., 2010)

Há também a função logística bipolar dada pela Equação 2.5. A curvatura é semelhante à logística binária, porém essa função tem a imagem limitada no intervalo real [-1,1]. Seu gráfico está representado na Figura 2.16.

$$f(x) = \frac{2}{1 + e^{-\beta \cdot x}} - 1 \tag{2.5}$$

Outra função de ativação com imagem limitada pelo intervalo real [-1,1] é a função tangente hiperbólica dada pela Equação 2.6. Seu gráfico é mostrado na Figura 2.17.

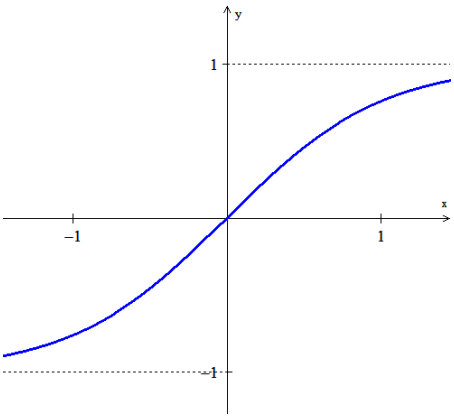


Figura 2.16: Função de ativação logística bipolar - Fonte: (Silva et al., 2010)

$$f(x) = \frac{1 - e^{-\beta \cdot x}}{1 + e^{-\beta \cdot x}} \tag{2.6}$$

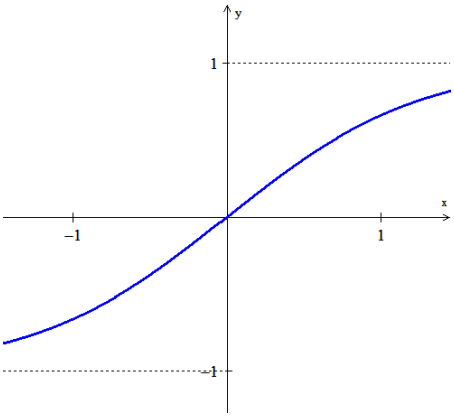


Figura 2.17: Função de ativação tangente hiperbólica bipolar - Fonte: (Silva et al., 2010)

Ainda temos a função de ativação gaussiana, dada pela equação 2.7, que, de acordo com Silva et al. (2010), produzirá resultados iguais para aqueles valores de potencial de ativação $\{x\}$ que fiquem a uma mesma distância da média que é o centro da distribuição. Essa curva depende do desvio-padrão da distribuição. A Figura 2.18 é uma ilustração gráfica para o caso em que $c = 1$. Geometricamente, nota-se que a curva é simétrica em relação à reta vertical que passa pela média.

$$f(x) = e^{-\frac{(x-c)^2}{2\sigma^2}} \tag{2.7}$$

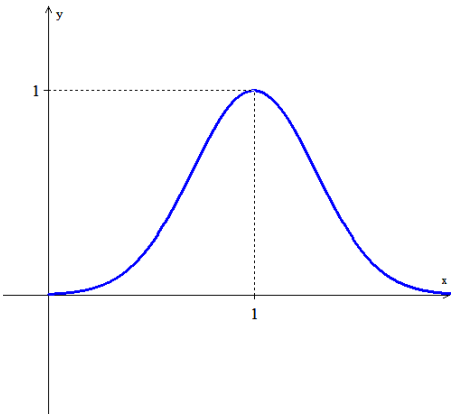


Figura 2.18: Função de ativação gaussiana - Fonte: (Silva et al., 2010)

Capítulo 3

Redes Neurais Artificiais do tipo Multilayer Perceptron

3.1 Arquitetura e características

As RNAs do tipo MLP são caracterizadas por apresentar uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída, todas elas compostas por um conjunto de nós sensoriais também conhecidos por neurônios (Haykin, 2008). A Figura 3.1 ilustra a arquitetura de uma rede do tipo MLP.

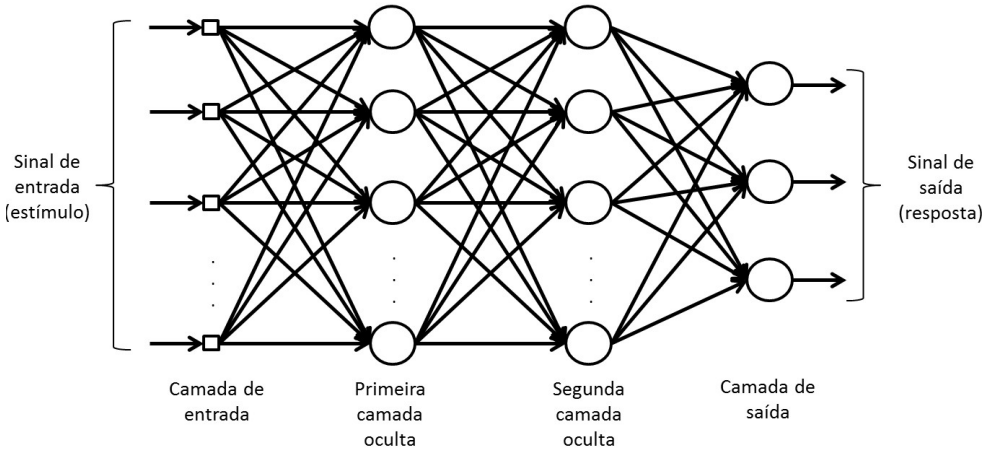


Figura 3.1: Arquitetura de uma rede do tipo MLP - Fonte: (Haykin, 2008)

Nesse tipo de RNA, o sinal de entrada propaga-se de camada em camada a partir da camada de entrada, chegando à camada de saída. A saída obtida pela rede é comparada a um alvo que também é conhecido como saída desejada. A diferença entre a saída gerada pela rede e a saída desejada é denominada erro. O sinal do erro é retro-propagado camada por camada, a partir da camada de saída, chegando à camada de entrada para atualizar os pesos sinápticos.

Segundo Haykin (Haykin, 2008), uma rede do tipo MLP tem três características que a distinguem dos demais tipos de rede:

1. Para cada neurônio da rede há uma função de ativação não linear, cuja curvatura é suave. Ao contrário do que ocorre com o Perceptron proposto por Rosenblatt, cuja função de ativação possui curvatura abrupta, nas redes do tipo MLP essa curvatura é diferenciável ao longo de todo o domínio. Um exemplo de função de ativação utilizada em redes do tipo MLP é a função logística mostrada na Equação 2.4. A não linearidade da função de ativação tem motivação biológica, uma vez que leva em consideração a fase refratária de neurônios reais (Haykin, 2008).
2. A rede contém uma ou mais camadas ocultas, que são diferentes da camada de entrada e de saída. Os neurônios dessas camadas ocultas são responsáveis pela capacidade de aprendizagem de problemas complexos.
3. Existe um alto grau de conectivade entre os neurônios. Isso significa que um neurônio de qualquer camada da rede está conectado a todos os neurônios da camada anterior. Uma simples mudança topológica, como a inclusão ou a exclusão de um neurônio em qualquer camada, implica mudança na população das conexões sinápticas ou de seus pesos.

Para ajustar os pesos sinápticos da rede a partir da função erro, são utilizadas equações de ajuste provenientes da minimização da função erro pelo método do gradiente descendente. Pelo fato de atuar nos dois sentidos (camada de entrada - camada de saída e camada de saída - camada de entrada), correspondendo a uma retropropagação, esse algoritmo recebe o nome de *backpropagation*.

3.2 Treinamento de redes Perceptron multicamadas – algoritmo *backpropagation*

O algoritmo de aprendizagem de uma rede MLP mais conhecido é denominado *backpropagation*, que, em uma tradução para o português, corresponde a retropropagação. Esse nome é justificado pelo fato de que o erro proveniente da diferença entre a saída estimada da rede com a saída esperada é usado no ajuste de todos os pesos sinápticos da rede. Cada padrão tem seu sinal propagado da entrada até a saída, e o sinal proveniente do erro é retropropagado da saída para a entrada.

Nesse algoritmo são definidas algumas variáveis, como a taxa de aprendizagem da rede "α", o número de neurônios da camada oculta e o critério de parada que pode ser, por exemplo, ao se atingir um certo número de ciclos ou pela determinação de um erro máximo admissível. A seguir, mostram-se os passos para a execução do algoritmo *backpropagation* de acordo com Fausset (1994):

3.2.1 Propagação de um padrão no algoritmo de treinamento

Passo 1: Inicialização dos pesos sinápticos, que devem ser valores aleatórios pequenos. Recomenda-se a utilização de valores entre -0,5 e 0,5 (L. V. Fausett & Hall, 1994).

Passo 2: Execução da propagação e retropropagação até que a condição de parada seja satisfeita.

Passo 3: Para cada padrão de treinamento, são executados os passos de 3 a 8.

Passo 4: Cada padrão de treinamento X tem seus valores propagados da camada de entrada para a camada oculta.

Passo 5: Na propagação de um padrão de treinamento, há uma influência dos pesos sinápticos zin_j calculados da seguinte maneira:

$$\overset{q}{zin}_j = \sum_{i=1}^{n} \left[\overset{q}{x}_i \cdot \overset{q}{v}_{ij} \right] + \overset{q}{v0}_j = \left[zin_1 \quad \overset{q}{zin_2} \quad \ldots \quad zin_j \right] \tag{3.1}$$

Em que:

- $\overset{q}{x_i}$ - é o valor de entrada para o *q-ésimo* padrão.
- $\overset{q}{v_{ij}}$ - é o peso sináptico dos neurônios que ligam a *i-ésima* entrada ao *j-ésimo* neurônio da camada oculta.
- $\overset{q}{v0_j}$ - é o peso sináptico dos *bias* da camada oculta.

Esses valores são submetidos a uma função de ativação para então serem enviados da camada oculta para a camada de saída.

$$\overset{q}{z_j} = f\left(\overset{q}{zin_j}\right) = \left[f(zin_1) \quad f(zin_2) \quad \ldots \quad f(zin_k)\right]^{q} = \left[z_1 \quad z_2 \quad \ldots \quad z_k\right] \tag{3.2}$$

Passo 6: Cada unidade de saída é obtida a partir da propagação dos valores de $\overset{q}{z_j}$ da camada oculta para a camada de saída, em que são aplicados os pesos sinápticos. Matematicamente, esse passo é representado por:

$$\overset{q}{yin_k} = \sum_{j=1}^{P}\left[\overset{q}{z_j} \cdot \overset{q}{w_{jk}}\right] + \overset{q}{w0_k} = \left[yin_1 \quad yin_2 \quad \ldots \quad yin_k\right]^{q} \tag{3.3}$$

Em que:

- $\overset{q}{w_{jk}}$ - é o peso sináptico que liga o *j-ésimo* neurônio da camada oculta à *k-ésima* saída.
- $\overset{q}{w0_k}$ - é o peso sináptico dos *bias* correspondentes às saídas da rede.

Os valores de *yin* são submetidos à função de ativação gerando os valores *y*, que representam a saída da rede de acordo com a Equação 3.4.

$$\overset{q}{y_k} = f\left(\overset{q}{yin_k}\right) = \left[f(yin_1) \quad f(yin_2) \quad \ldots \quad f(yin_k)\right]^{q} = \left[y_1 \quad y_2 \quad \ldots \quad y_k\right] \tag{3.4}$$

3.2.2 Retropropagação do erro

Passo 7: Cada padrão de entrada tem uma unidade de saída correspondente que é representada pelo vetor-alvo, visto que se trata de um treinamento supervisionado. O erro proveniente da diferença entre esses valores é usado para a atualização dos pesos. Na Equação (3.5), temos o cálculo do erro quadrático, e, nas Equações (3.6), (3.7) e (3.8), esse erro é utilizado nas matrizes de atualização de pesos entre a camada oculta e a camada de saída.

$$E = \frac{1}{2} \cdot \sum_{k} \left(\overset{q}{t}_k - \overset{q}{y}_k \right)^2 \tag{3.5}$$

Em que:

- $\overset{q}{t}_k$ - é a saída esperada para o *q-ésimo* padrão.

$$\overset{q}{\delta}_k = \left(\overset{q}{t}_k - \overset{q}{y}_k \right) \cdot f' \left(\overset{q}{yin}_k \right) \tag{3.6}$$

$$\Delta \overset{q}{w}_{jk} = \alpha \cdot \overset{q}{\delta}_k \cdot \overset{q}{z}_j \tag{3.7}$$

$$\Delta \overset{q}{w}0_k = \alpha \cdot \overset{q}{\delta}_k \tag{3.8}$$

Em que:

- α - é a taxa de aprendizagem da rede.

Passo 8: Cálculo das matrizes de atualização de pesos entre a camada de entrada e a camada oculta.

$$\overset{q}{\delta in}_j = \sum_{k=1}^{m} \left[\overset{q}{\delta}_k \cdot \overset{q}{w}_{jk} \right] \tag{3.9}$$

$$\overset{q}{\delta}_j = \overset{q}{\delta in}_j \cdot f' \left(\overset{q}{zin}_j \right) \tag{3.10}$$

$$\Delta v_{ij}^{q} = \alpha \cdot \delta_{j}^{q} \cdot x_{i}^{q} \tag{3.11}$$

$$\Delta v0_{j}^{q} = \alpha \cdot \delta_{j}^{q} \tag{3.12}$$

Passo 9: Atualização dos pesos e bias.

$$w_{jk}^{q+1} = w_{jk}^{q} + \Delta w_{jk}^{q} \tag{3.13}$$

$$v_{ij}^{q+1} = v_{ij}^{q} + \Delta v_{ij}^{q} \tag{3.14}$$

$$w0_{k}^{q+1} = w0_{k}^{q} + \Delta w0_{k}^{q} \tag{3.15}$$

$$v0_{j}^{q+1} = v0_{j}^{q} + \Delta v0_{j}^{q} \tag{3.16}$$

Passo 10: Verificação da condição de parada.

A propagação e retropropagação de todos os padrões completam um ciclo ou uma época. O algoritmo é executado até que a condição de parada seja satisfeita.

3.3 Principais mecanismos de melhoria de desempenho de redes Multilayer Perceptron

Segundo Haykin (Haykin, 2008), modelar uma RNA e aplicá-la a um problema qualquer é mais uma arte do que uma ciência. Isso significa que, muitas vezes, a escolha de parâmetros, tais como o valor da taxa de aprendizagem, o número de camadas ocultas, a quantidade de neurônios da camada oculta e o critério estabelecido para o término do treinamento, depende bastante da experiência particular de cada um.

Embora isso seja importante e traga consigo grande verdade, existem métodos que comprovadamente melhoram o desempenho do algoritmo *backpropagation*.

3.3.1 Maximização do conteúdo de informação

As amostras de treinamento são fundamentais para que a rede adquira informações sobre o problema e consiga classificar corretamente os padrões a ela apresentados. Por essa razão, a amostra de treinamento precisa representar de forma fidedigna o seu conjunto de forma que seu conteúdo de informação seja o maior possível para a tarefa considerada (LeCun, 1993). Para alcançar esse objetivo, duas heurísticas são indicadas na seleção das amostras de treinamento:

- A utilização de uma amostra que resulte no maior erro de treinamento.
- O uso de uma amostra que seja radicalmente diferente de todos as outras usadas anteriormente.

A adoção dessas heurísticas pode possibilitar a ampliação na busca do espaço de pesos.

Em problemas de reconhecimento de padrões, outra heurística que pode contribuir significativamente para que a rede tenha ganho de desempenho é a apresentação aleatória do conjunto de amostras de treinamento. Em vez de apresentar as amostras sempre na mesma ordem em todos os ciclos de treinamento, sugere-se que essa ordem seja embaralhada a cada ciclo.

3.3.2 Função de ativação

A função de ativação também exerce papel importante na velocidade de convergência da rede, bem como na capacidade de generalização. Funções de ativação do tipo sigmoide podem desencadear aceleração da convergência se forem antissimétricas ao invés de não simétricas. Uma função de ativação é antissimétrica, ou seja, é ímpar se

$$\varphi(-v) = -\varphi(v) \tag{3.17}$$

O gráfico de uma sigmóide (tangente hiperbólica) está representado na Figura 3.2. Uma função logística padrão conforme a Figura 3.3 não atende a essa condição (Haykin, 2008).

Uma função que atende a essa característica é a tangente hiperbólica mostrada na Equação (3.18). LeCun propôs que o valor de *a* seja igual a 1,7159 e que o valor de b seja igual a $\frac{2}{3}$ (LeCun, 1993).

$$\varphi(v) = atanh(bv) \tag{3.18}$$

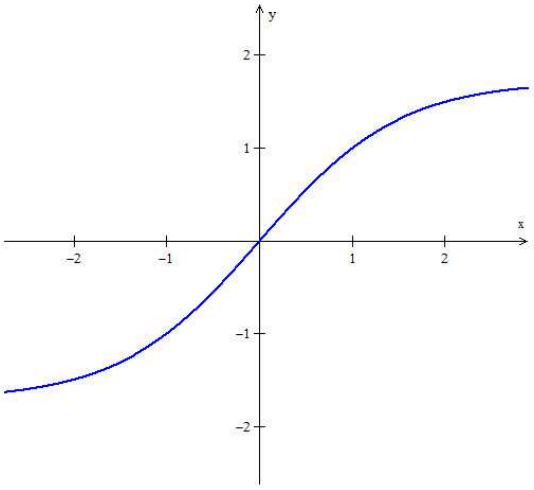


Figura 3.2: Função tangente hiperbólica ideal

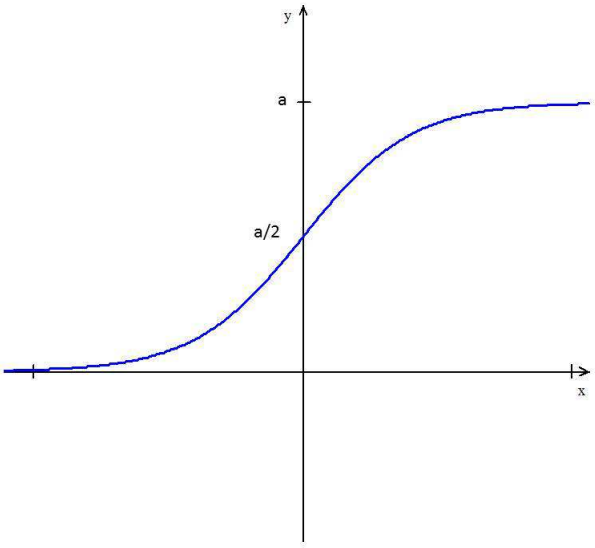


Figura 3.3: Função logística padrão

3.3.3 Normalização das entradas

De acordo com LeCun (LeCun, 1993), cada variável de entrada deve ser pré-processada no intuito de que a média calculada sobre todo o conjunto de treinamento esteja próximo de zero, ou que seja pequena em comparação ao desvio-padrão. Sem essa condição, o vetor peso de determinado neurônio pode ter dificuldades para convergir, tornando o treinamento lento. LeCun (LeCun, 1993) indica ainda duas medidas a serem incluídas na normalização dos dados de entrada:

- As variáveis de entrada pertencentes ao conjunto de treinamento devem ser não correlacionadas, o que pode ser verificado por meio da análise de componentes principais.
- Além de serem descorrelacionadas, as variáveis de entrada devem ser escolhidas de forma que suas covariâncias sejam aproximadamente iguais, a fim de que os diferentes pesos sinápticos da rede sejam treinados com velocidade aproximadamente igual.

3.3.4 Inicialização dos pesos sinápticos

Como exposto na introdução, a escolha dos pesos sinápticos iniciais é tema de diversas pesquisas. Essa escolha tem impacto extremamente importante no desempenho da rede. Junto à escolha dos pesos sinápticos, a escolha dos limiares de classificação também é importante.

Se a escolha dos pesos sinápticos for por valores grandes, os neurônios podem ser levados rapidamente à saturação. Nessa situação, os gradientes locais tornam-se demasiadamente pequenos, o que provoca lentidão no processo de treinamento (Haykin, 2008).

Por outro lado, se os valores forem muito pequenos, a superfície do erro pode se tornar muito plana em torno de sua origem. Como a origem é um ponto de sela, o algoritmo pode ficar preso em torno desse ponto, impedindo que ocorra efetivamente a aprendizagem requerida pelos problemas.

Em geral, há uma indicação de que a distribuição uniforme da qual os pesos sinápti-

cos são selecionados tenha média zero e variância igual ao recíproco do número de conexões sinápticas de um neurônio (Haykin, 2008).

3.3.5 Taxa de aprendizagem

A taxa de aprendizagem representa, em termos práticos, o tamanho do passo que a rede toma em busca do ajuste ideal dos pesos sinápticos. Passos muito pequenos podem fazer com que o treinamento fique lento. Passos muito longos podem conduzir o ajuste a direções incorretas em relação à direção ideal do treinamento, o que significa que o próximo passo irá requerer correção do erro anterior.

Além desse problema, a convergência pode passar por longos períodos de estagnação quando o gradiente cai em mínimos locais. Há indicações de procedimentos para se evitarem esses problemas. Uma delas é a de que a taxa de aprendizagem deve ser menor nas últimas camadas. Isso é indicado pelo fato de que as últimas camadas possuem gradientes locais maiores do que as camadas anteriores (Haykin, 2008).

Outra indicação é a de que neurônios com muitas entradas tenham taxa de aprendizagem menor (Haykin, 2008). LeCun (LeCun, 1993) sugere que, para um determinado neurônio, a taxa de aprendizagem seja inversamente proporcional à raiz quadrada das conexões sinápticas feitas com aquele neurônio.

Também há indicações de que a taxa de aprendizagem deva ser ajustada ao longo do treinamento. Jacobs (Jacobs, 1988) sugere quatro heurísticas para a taxa de aprendizagem.

1. Cada parâmetro ajustável da função de custo da rede deve ter seu parâmetro individual da taxa de aprendizagem. Isso significa que uma taxa de aprendizagem eficiente para determinado peso nem sempre é eficiente para outro peso. Cada região da superfície do erro pode se adaptar melhor a determinado valor de taxa de aprendizagem.
2. A taxa de aprendizagem deve poder variar de um ciclo para outro. Isso é especialmente importante porque a superfície do erro tem comportamento diferente em

ciclos diferentes do treinamento.

3. A taxa de aprendizagem deve ser aumentada sempre que a derivada da função custo em relação a um peso sináptico tiver o mesmo sinal algébrico para ciclos consecutivos do treinamento. A principal motivação para essa heurística, reside no fato de que, em porções mais planas da superfície do erro ao longo da dimensão de um peso particular, o sinal algébrico se mantenha igual em vários ciclos do treinamento. O aumento na taxa de aprendizagem pode reduzir o número de ciclos para atravessar essa porção plana.
4. A taxa de aprendizagem deve ser reduzida sempre quando a derivada da função custo em relação a um peso sináptico alternar de sinal por vários ciclos. Essa situação ocorre em regiões da superfície do erro com grande quantidade de picos e vales. Reduzindo-se o valor da taxa de aprendizagem, evita-se que o vetor gradiente mude de sinal por muitas vezes. Isso acelera a convergência do treinamento.

3.3.6 Parada antecipada do treinamento – *Early stopping*

Uma questão muito discutida é o momento ideal para concluir o treinamento de uma rede do tipo MLP. Sob condições normais, a curva do erro quadrático médio diminui ao longo da evolução do número de ciclos. Entende-se por ciclo a apresentação de todo o conjunto de amostras de treinamento durante a fase de aprendizagem da rede MLP. Normalmente, os padrões usados no treinamento possuem ruídos. Se a rede treinar muito, ela pode aprender com os ruídos dos padrões do conjunto de treinamento. Isso leva à perda de generalidade, fenômeno esse denominado *overfitting*.

Uma maneira de evitar esse problema é a técnica conhecida como *early stopping* (Haykin, 2008), que pode ser interpretada como técnica da parada antecipada do treinamento. Além do conjunto de treinamento, também é usado um conjunto de validação. Periodicamente, o conjunto de validação é submetido à classificação pela rede com os pesos sinápticos ajustados naquele ciclo de treinamento. O erro quadrático médio do conjunto de validação é calculado.

Se o erro começa a aumentar, a rede está começando a aprender com os ruídos do conjunto de treinamento, ou seja, está caminhando para uma condição de *overfitting*. Esse é o momento de se encerrar o treinamento. A Figura 3.4 ilustra genericamente a evolução da curva do erro para os conjuntos de treinamento e validação.

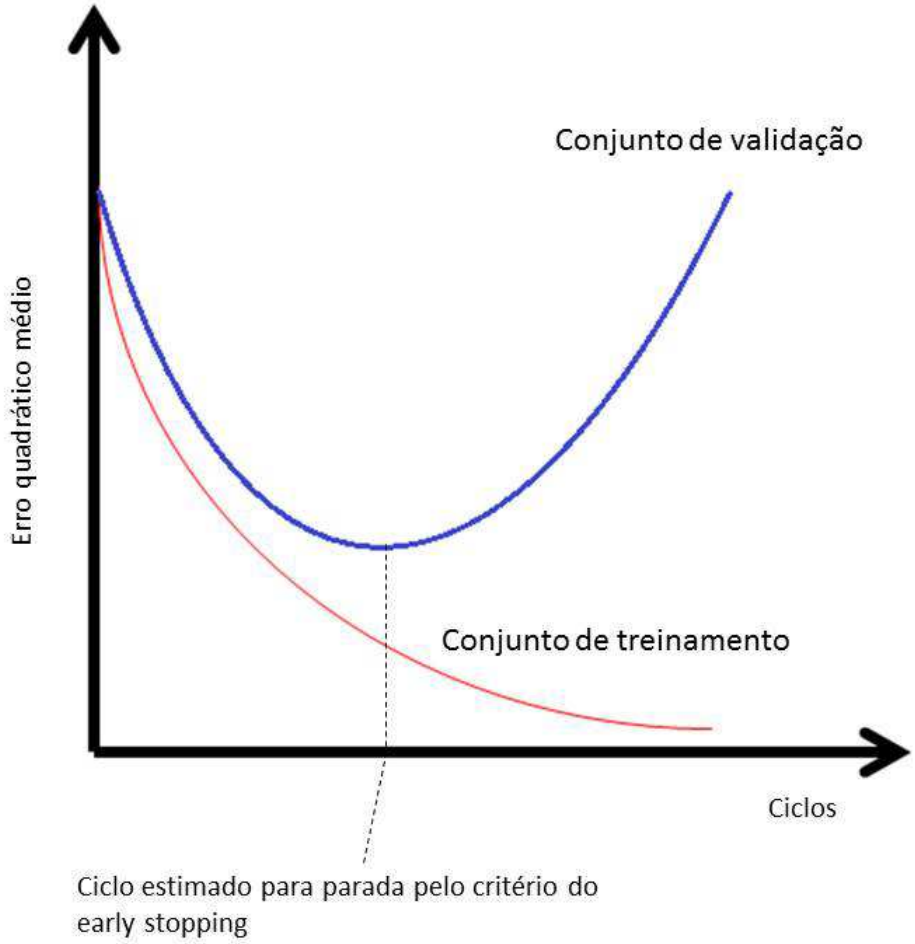


Figura 3.4: Ilustração da regra do *early stopping*

3.3.7 Termo Momentum

Durante o treinamento de uma RNA do tipo MLP, pode acontecer de a solução obtida com as matrizes de pesos atuais estar muito longe da solução final. Nesse caso, entre dois ciclos sucessivos, existe grande variação na direção oposta ao gradiente da função erro quadrático.

Por outro lado, em determinados momentos do treinamento, essa mesma variação

pode ser bem pequena, o que significa que a solução obtida com as matrizes de pesos atuais está próxima da solução final.

As duas situações indicam dois extremos no processo de treinamento. Na primeira, o passo que conduz a rede até a solução final precisa ser aumentado em direção ao mínimo da função erro. Na segunda, o passo de aprendizagem precisa ser pequeno, sob pena de se desviar o caminho da convergência para regiões distantes do mínimo da função erro.

Para melhorar o processo de convergência à solução final, aplica-se um termo na atualização de pesos denominado "Termo Momentum", o qual pondera a variação entre as matrizes de pesos sinápticos em ciclos sucessivos (Silva et al., 2010). Para qualquer matriz de pesos sinápticos V_{ij}, para uma constante real β, para a taxa de aprendizagem α e para o ciclo n, tem-se que a equação de ajuste de pesos é dada por:

$$V_{i,j}(n+1) = V_{i,j}(n+1) + \beta \cdot (V_{i,j}(n) - V_{i,j}(n-1)) + \alpha \cdot \delta_i \cdot Y_j \tag{3.19}$$

em que β é a taxa de ação do "Termo Momentum", $\beta \cdot (V_{i,j}(n) - V_{i,j}(n-1))$ é o próprio "Termo Momentum" e $\alpha \cdot \delta_i \cdot Y_j$ é o "Termo de Aprendizagem".

Considerando que o Termo Momentum é resultado da diferença entre duas matrizes de pesos consecutivas, se a diferença for significativa, a ação do Momentum também será significativa, acelerando o passo rumo à convergência para a solução final. Caso contrário, a interferência do Momentum torna-se pequena, o que faz com que a convergência seja conduzida praticamente apenas pelo Termo de Aprendizagem (Silva et al., 2010).

Estudos indicam os valores de $0 \leq \alpha \leq 0,9$ e $0,05 \leq \beta \leq 0,75$ para o treinamento de MLPs (Rumelhart, Hinton, & Williams, 1985).

Capítulo 4

Estatística utilizada no trabalho

Para ser possível inferir sobre as hipóteses levantadas pela pesquisa, é necessário utilizar de meios científicos que sejam capazes de validar conclusões acerca de resultados. Um dos caminhos é a demonstração matemática que elimina qualquer dúvida a respeito de alguma abordagem ou metodologia proposta. No entanto, muitas vezes a representação analítica ou geométrica de um problema pode ser desconhecida ou ainda não possível de ser viabilizada.

Outro caminho é o campo da inferência estatística quando se está diante de fenômenos aleatórios. Uma combinação de métodos probabilíticos, permite realizar inferências a partir de amostras de dados dentro de uma probabilidade de erro pequena. A inferência estatística consiste numa metodologia de experimentação presente no planejamento, análise e cálculo de estatísticas em testes de hipóteses. Garantido o controle de toda a experimentação, os métodos estatísticos permitem que afirmações sobre hipóteses sejam realizadas com grande segurança. Essa é uma abordagem alternativa em situações em que não é possível empregrar demonstrações matemáticas. Consequentemente, trata-se de uma abordagem amplamente utilizada em diversas áreas do meio científico.

A inferência estatística é composta por diversos testes. Cada teste é adequado aos diferentes tipos de situações. Assim, para cada problema, é necessário verificar qual a melhor metodologia de experimentação e qual o melhor teste de hipóteses. Considerando que redes MLP tem inicialização aleatória dos pesos sinápticos e que os pesos interferem na capacidade de

reconhecimento de padrões, faz-se necessário constatar se as hipóteses levantadas pelo trabalho são verdadeiras. Diante dessa necessidade, este trabalho utilizou de testes estatísticos para inferir sobre as hipóteses de melhor desempenho com a utilização de VBOs. Este capítulo apresenta os testes estatísticos utilizados na pesquisa.

4.1 Teste de Kolmogorov-Smirnov

Os métodos estatísticos dividem-se basicamente em duas famílias: a dos paramétricos e a dos não paramétricos. Os testes paramétricos dependem de alguns parâmetros que geralmente são a média e a variância, e pressupõe-se que os dados analisados correspondam a uma distribuição aproximadamente normal. Já os testes não paramétricos não dependem dos parâmetros nem do ajuste à normalidade pela distribuição dos dados analisados.

Logo, uma informação importante antes de decidir qual teste utilizar é se a distribuição dos dados a serem analisados se ajusta a uma distribuição normal. O teste de Kolmogorov Smirnov permite verificar se uma amostra de dados se ajusta a alguma distribuição teórica, por exemplo, a distribuição normal (Conover, 1999). Trata-se de um teste bastante utilizado e, nas análises estatísticas deste trabalho, ele foi empregado para decisão sobre a utilização de um teste paramétrico ou não paramétrico.

As hipóteses do Teste de Kolmogorov-Smirnov para verificação de ajuste à normalidade são:

- H_0: os dados seguem uma distribuição normal.
- H_1: os dados não seguem uma distribuição normal.

Para a realização do teste, os seguintes passos devem ser executados (Conover, 1999):

1. Ordenam-se os dados x_i em ordem crescente.
2. Para cada x_i dado é atribuído o valor $F_n(x_i)$, denominado valor empírico, que é o resultado da razão de i por n, em que n é o total de dados.
3. Para cada i, atribui-se o valor normal padronizado correspondente $F(x_i)$, denominado valor teórico, que é obtido pela Tabela de distribuição normal padrão e pela

fórmula representada pela equação 4.1:

$$Z_i = \frac{x_i - \overline{x}}{s} \tag{4.1}$$

em que s é o desvio-padrão da amostra de dados e $\overline{x}$ é a média dos dados.

4. Para cada i, calcula-se o módulo da diferença entre o valor teórico e o valor empírico $|F(x_i) - F_n(x_i)|$.
5. Para cada i, calcula-se o módulo da diferença entre o valor teórico e o valor empírico anterior $|F(x_i) - F_n(x_{i-1})|$.
6. Calcula-se a somatória das diferenças entre o valor teórico e o valor empírico $\Sigma|F(x_i) - F_n(x_i)|$, denominada D^+.
7. Calcula-se a somatória das diferenças entre o valor teórico e o valor empírico anterior $\Sigma|F(x_i) - F_n(x_{i-1})|$, denominada D^-.
8. Obtém-se o máximo entre D^+ e D^-, denominado D_n.
9. Para um nível de significância α e a quantidade de amostras n, compara-se o valor crítico V da Tabela com o valor de D_n. Se $D_n \leq V$, aceita-se H_0. Se $D_n > V$, rejeita-se H_0.

Para fins de ilustração, o cálculo de D_n pode ser realizado por meio da Tabela 4.1.

Tabela 4.1: Ilustração de Tabela de cálculo para o valor de D_n

x**(ordenado)**	$F_n(x_i)$	$F(x_i) = P\left(z_i \leq \frac{x_i - \overline{x}}{s}\right)$	$\lvert F(x_i) - F_n(x_i)\rvert$	$\left\lvert F(x_i) - F_n\left(x_{(i-1)}\right)\right\rvert$
x_1	$\frac{1}{n}$	$F(x_1) = P\left(z_1 \leq \frac{x_1 - \overline{x}}{s}\right)$	$\lvert F(x_1) - F_n(x_1)\rvert$	$\lvert F(x_1) - 0\rvert$
x_2	$\frac{2}{n}$	$F(x_2) = P\left(z_2 \leq \frac{x_2 - \overline{x}}{s}\right)$	$\lvert F(x_2) - F_n(x_2)\rvert$	$\lvert F(x_2) - 0\rvert$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_i	$\frac{i}{n}$	$F(x_i) = P\left(z_i \leq \frac{x_i - \overline{x}}{s}\right)$	$\lvert F(x_i) - F_n(x_i)\rvert$	$\left\lvert F(x_i) - F_n\left(x_{(i-1)}\right)\right\rvert$
Somatórios			$\Sigma\lvert F(x_i) - F_n(x_i)\rvert$	$\Sigma\left\lvert F(x_i) - F_n\left(x_{(i-1)}\right)\right\rvert$

4.2 Teste de Mann-Whitney

Quando a amostra de dados não se ajusta à distribuição de probabilidade normal, uma saída é a utilização de testes não paramétricos. E quando o problema consiste em comparar médias entre grupos independentes, há ainda a exigência de que as variâncias sejam iguais. O Teste de Mann-Whitney permite a comparação de igualdade de médias sem a exigência desses requisitos (Martins & Fonseca, 2006).

Para a realização do teste, deve-se proceder da seguinte forma:

1. Considera-se n_1 como o número de amostras do menor grupo e n_2 como o número de amostras do maior grupo.
2. Os dados dos dois grupos são reunidos e organizados em ordem crescente. O menor dado recebe o número 1 e todos os outros são ordenados até o último dado, que corresponde a $N = n_1 + n_2$.
3. Para as amostras iguais (empatadas), calcula-se a média entre seus postos, e cada uma recebe o posto médio.
4. Calcula-se R_1 = soma dos postos do grupo n_1 e R_2 = soma dos postos do grupo n_2.
5. Escolhe-se a menor soma entre R_1 e R_2.
6. Calculam-se as estatísticas:

$$\mu_1 = n_1 \cdot n_2 + \frac{n_1 (n_1 + 1)}{2} - R_1 \tag{4.2}$$

$$\mu_2 = n_1 \cdot n_2 + \frac{n_2 (n_2 + 1)}{2} - R_2 \tag{4.3}$$

7. Hipóteses do teste:

   H_0: não há diferença entre os grupos.

   H_1: há diferença entre os grupos.
8. Definição do nível α de significância.
9. Com o auxílio da Tabela de distribuição de probabilidades normal padronizada,

definem-se as regiões de aceitação e de rejeição de H_0 conforme ilustração da Figura 4.1.

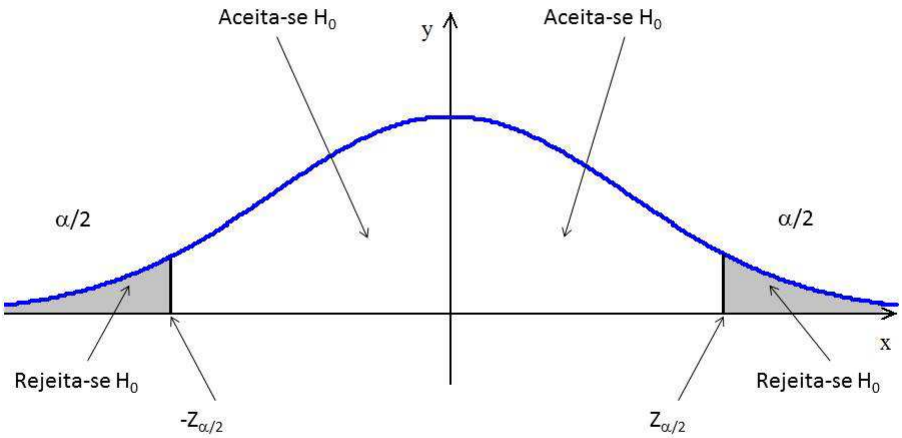


Figura 4.1: Esboço da região de aceitação e de rejeição para o Teste de Mann-Whitney

10. Cálculo do valor variável Z_{cal}:

$$Z_{cal} = \frac{\mu - \mu(u)}{\sigma(u)} \tag{4.4}$$

em que

$$\mu(u) = \frac{n_1 \cdot n_2}{2} \tag{4.5}$$

$$\sigma(u) = \sqrt{\frac{n_1 \cdot n_2 \left(n_1 + n_2 + 1\right)}{12}} \tag{4.6}$$

11. Conclusão:

Se $-Z\frac{\alpha}{2} \leq Z_{cal} \leq Z\frac{\alpha}{2}$, aceita-se H_0.

Se $-Z_{cal} > Z\frac{\alpha}{2}$ ou $Z_{cal} < -Z\frac{\alpha}{2}$, rejeita-se H_0, assumindo-se com risco α que os grupos

possuem diferença em relação às suas médias.

Capítulo 5

Conceitos matemáticos envolvidos na pesquisa

A proposta do trabalho perpassa conceitos matemáticos relacionados a área de Geometria Analítica e Álgebra Linear. Naturalmente a análise matemática acerca da hipótese faz parte do propósito do trabalho em mostrar o ganho de desempenho das MLPs treinadas com o uso de VBOs. Por essa razão, neste Capítulo são abordados alguns conceitos matemáticos que dão subsídio a discussões futuras sobre a proposta do trabalho. São os conceitos de produto interno, distância euclidiana, ângulo entre vetores e ortogonalidade.

5.1 Produto interno e distância euclidiana de vetores no espaço R^n

Considere-se que $\overrightarrow{V_i} = (v_1, v_2, ..., v_n)$ e $\overrightarrow{W_i} = (w_1, w_2, ..., w_n)$ sejam dois vetores do espaço R^n. Uma operação existente entre vetores pertencentes a espaços de qualquer dimensão é o produto interno. A Equação (5.1) representa o produto interno dos vetores $\overrightarrow{V_i}$ e $\overrightarrow{W_i}$.

$$\overrightarrow{V_i} \bullet \overrightarrow{W_i} = v_1 \cdot w_1 + v_2 \cdot w_2 + v_3 \cdot w_3 + ... + v_n \cdot w_n \tag{5.1}$$

Além do produto interno, também pode-se obter a distância euclidiana entre $\overrightarrow{V_i}$ e $\overrightarrow{W_i}$, independente da dimensão do espaço R^n. Essa distância é calculada por meio da Equação (5.2).

$$d_{V,W} = \sqrt{(w_1 - v_1)^2 + (w_2 - v_2)^2 + (w_3 - v_3)^2 + ... + (w_n - v_n)^2} \tag{5.2}$$

5.2 Ângulo e ortogonalidade entre vetores no espaço R^n

Considere-se ainda que os vetores $\overrightarrow{V_i}$ e $\overrightarrow{W_i}$ pertencem ao espaço R^n. O cosseno do ângulo entre esses vetores é calculado pela razão entre o seu produto interno e o produto entre os módulos dos vetores, o que é representado pela Equação (5.3).

$$cos(\theta) = \frac{\overrightarrow{V_i} \bullet \overrightarrow{W_i}}{\|\overrightarrow{V_i}\|\|\overrightarrow{W_i}\|} \tag{5.3}$$

O ângulo pode ser determinado por meio da função inversa do cosseno, que é representada pela Equação (5.4).

$$\theta = arccos\left(\frac{\overrightarrow{V_i} \bullet \overrightarrow{W_i}}{\|\overrightarrow{V_i}\|\|\overrightarrow{W_i}\|}\right) \tag{5.4}$$

A Figura 5.3 ilustra o ângulo entre dois vetores do espaço R^3.

A partir do cálculo do ângulo entre vetores do espaço R^n, de conceitos de trigonometria e de aritmética, podem-se destacar os seguintes fatos:

1. A função arco-cosseno tem sua imagem restrita ao intervalo $[0, \pi]$.
2. O produto dos módulos dos vetores dado por $\|\overrightarrow{V_i}\|\|\overrightarrow{W_i}\|$ é sempre positivo.
3. O cosseno é positivo no intervalo de $[0, \frac{\pi}{2}[$.
4. O cosseno é nulo para o arco $\frac{\pi}{2}$.
5. O cosseno é negativo no intervalo de $]\frac{\pi}{2}, \pi]$.

A partir dos fatos expostos anteriormente, pode-se afirmar que o ângulo entre os vetores $\overrightarrow{V_i}$ e $\overrightarrow{W_i}$ é agudo sempre que o produto interno for postivo; é reto sempre que o produto interno for igual a zero e é obtuso sempre que o produto interno for negativo.

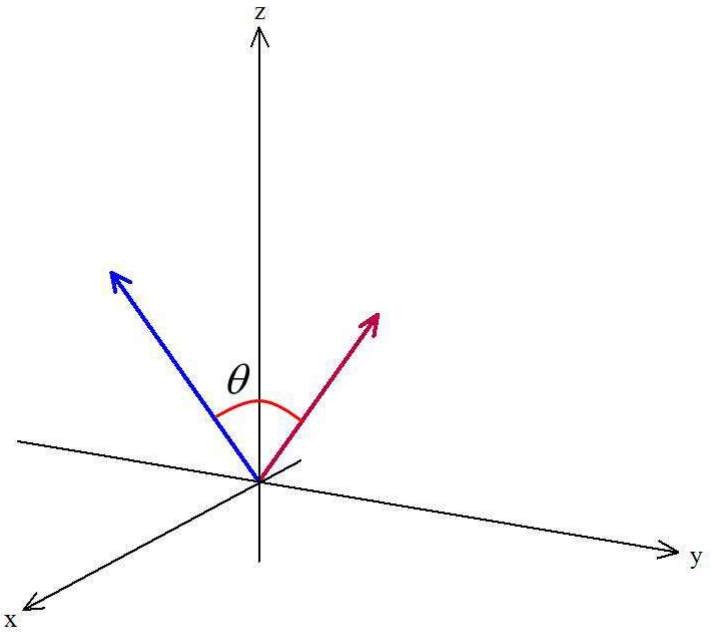


Figura 5.1: Ângulo entre vetores

Portanto dois vetores são ortogonais se o produto interno entre eles for igual a zero. A Figura 5.2 representa dois vetores com essa característica.

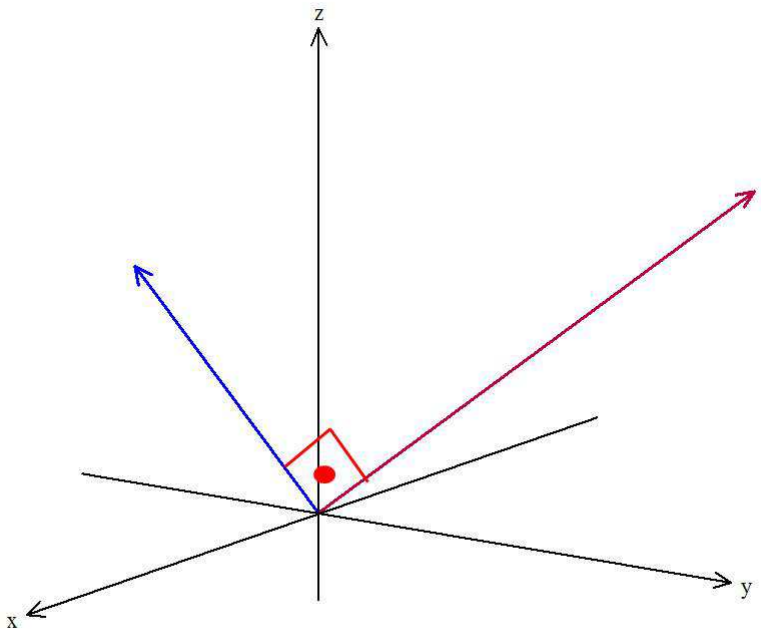


Figura 5.2: Ilustração de vetores ortogonais

Capítulo 6

Discussão matemática do uso de alvos ortogonais em redes Multilayer Perceptron

6.1 Definição de vetores-alvo

Convencionalmente, RNAs do tipo MLP utilizam dois tipos de vetores-alvo em problemas de reconhecimento de padrões. Os Vetores Binários (VBNs) e os Vetores Bipolares Convencionais (VBCs). A proposta deste trabalho é a utilização de vetores-alvo que possuem a característica de ser mutuamente ortogonais. São os Vetores Bipolares Ortogonais (VBOs). Esses vetores ortogonais têm por característica a dimensão sempre equivalente a uma potência de 2. Isso faz com que em aplicações onde o número de padrões a serem classificados é diferente de uma potência de 2, seja necessária a utilização de VBOs com dimensão maior do que a dos vetores convencionais. Para fins de comparação, este trabalho também utilizou vetores com a mesma característica dos vetores convencionais, tendo a mesma dimensão dos vetores ortogonais. Eles são denominados como Vetores não Ortogonais (VNOs). Para tornar essas informações mais claras, seguem as definições dos tipos de vetores-alvo:

- Vetores Binários (VBN) correspondendo à Equação (6.1): VBNs são vetores constituídos por n componentes. O valor n corresponde à quantidade de padrões a serem classificados pela RNA. Considerando uma matriz de VBNs, Cada linha i desta

matriz corresponde ao *i-ésimo* VBN contendo o componente "1" para $i = j$ e o componente "0" para os outros elementos.

$$\overrightarrow{V_{ij}} = \begin{cases} 1 & \text{for} \quad i = j \\ 0 & \text{for} \quad i \neq j \end{cases} \tag{6.1}$$

- Vetores Bipolares Convencionais (VBC): De forma semelhante ao que ocorre com VBNs, VBCs também são constituídos por n componentes e sua dimensão depende da quantidade de padrões a serem classificados pela RNA. A Equação (6.2) define uma matriz com n VBCs de dimensão n.

$$\overrightarrow{V_{ij}} = \begin{cases} 1 & \text{for} \quad i = j \\ -1 & \text{for} \quad i \neq j \end{cases} \tag{6.2}$$

- Vetores Bipolares Ortogonais (VBO): Estes vetores são mutuamente ortogonais. Para obter VBOs, é necessário utilizar um algoritmo que é apresentado na Seção 6.2. Por razões matemáticas, o tamanho do VBO é sempre uma potência de 2.
- Vetores Não Ortogonais (VNO): Estes vetores são uma extensão dos VBCs. Possuem a mesma característica dos vetores do tipo VBCs e a mesma dimensão dos VBOs. Eles foram utilizados neste trabalho apenas com o objetivo de propiciar uma comparação justa de vetores com a mesma característica dos VBCs, porém com a mesma dimensão dos VBOs. Assim, para obter VNOs, VBCs são complementados com o termo "−1" de modo a atingir o mesmo tamanho dos VBOs.

6.2 Algoritmo de geração de vetores bipolares ortogonais

O método de geração de vetores bipolares ortogonais é proposto por (L. V. Fausett & Hall, 1994). Nele é utilizado um vetor de sementes na geração de VBOs. Dá-se esse nome – vetor de sementes – pelo fato de que, a partir dele, será gerada uma sequência de vetores até a determinação final dos VBOs. O número de componentes de cada VBO depende do número de

vetores desejado e do número de componentes escolhido para o vetor de sementes. A fórmula dada pela Equação (6.3) leva a esse número de componentes de cada vetor.

$$n = 2^k m \tag{6.3}$$

Na Equação (6.3), 2^k é o número de vetores ortogonais entre si para $k > 0$ e m é o número de componentes em um vetor de sementes. É interessante observar que o número de vetores será sempre uma potência de base 2. Logicamente, o interessado constrói seu conjunto, observando esse detalhe, e, após a obtenção dos vetores, seleciona somente a quantidade de que necessitar.

Assim sendo, um conjunto com 2^k VBOs é construído com $2^k m$ componentes. O algoritmo é executado a partir dos seguintes passos:

Passo 1 - Iniciação de m e k – Os valores de m e k devem ser determinados de acordo com a necessidade da aplicação. O valor de m pode ser "1", e, à medida que o valor de m é aumentado, a quantidade de componentes dos VBOs cresce progressivamente.

Passo 2 - Iniciação do vetor de sementes – O vetor de sementes é obtido por meio da Equação (6.4). Vale ressaltar que o vetor de sementes pode ter um único componente.

$$V_m^0 = \overbrace{(1, 1, \ldots, 1)}^{m}{}^T \tag{6.4}$$

Passo 3 - Cálculo do número de componentes em um VBO – O cálculo do número de componentes do VBO é feito por meio da Equação (6.3).

Passo 4 - Construção de vetores – Usa-se a função de concatenação para a construção dos vetores.

Dados os vetores $U = (u_1, u_2, ..., u_n)$ e $W = (w_1, w_2, ..., w_n)$, a função de concatenação $fcc(U, W)$ é definida pela equação:

$$fcc(U, W) = (u_1, u_2, ..., u_n, w_1, w_2, ..., w_n) \tag{6.5}$$

A partir do vetor de sementes obtido no Passo 2, são construídos os vetores $V_{2m}^1 = fcc\,(V_m^0, V_m^0)$ e $V_{2m}^2 = fcc\,(V_m^0, -V_m^0)$. Pode-se perceber que o vetor de sementes é utilizado duas vezes, sendo que, na primeira, há a concatenação com o segundo argumento e a manutenção do sinal, e, na outra, o sinal do segundo argumento é trocado.

Passo 5 - Construção de vetores − A partir dos dois vetores obtidos no Passo 4, serão determinados quatro novos vetores com a utilização da função de concatenação. Os vetores construídos são $V_{4m}^1 = fcc\,(V_{2m}^1, V_{2m}^1)$, $V_{4m}^2 = fcc\,(V_{2m}^1, -V_{2m}^1)$, $V_{4m}^3 = fcc\,(V_{2m}^2, V_{2m}^2)$ e $V_{4m}^4 = fcc\,(V_{2m}^2, -V_{2m}^2)$. Os vetores determinados pelo passo anterior são usados duas vezes na função de concatenação, sendo que, na primeira vez, o sinal do segundo argumento é mantido e, na segunda, é trocado.

Passo 6 - Sequência da Concatenação − São realizadas concatenações até que se consigam 2^k vetores ortogonais com n componentes: $V_n^1, \ldots, V_n^{2^k}$.

Considere-se o exemplo em que se deseje construir 8 vetores ortogonais bipolares. Suponha-se que se decida usar um vetor de sementes com um componente. Assim, têm-se, de acordo com a Equação (6.6), 8 componentes em cada vetor.

$$n = 2^k m = 8 \cdot 1 = 8 \tag{6.6}$$

Esse vetor de sementes é dado então por $V_m^0 = (1)$. Passando ao passo 4, obtêm-se os vetores dados pelas Equações (6.7) e (6.8).

$$V_{2m}^1 = fcc\left(V_m^0, V_m^0\right) = (1, 1) \tag{6.7}$$

$$V_{2m}^1 = fcc\left(V_m^0, -V_m^0\right) = (1, -1) \tag{6.8}$$

No passo 5, obtém-se um novo conjunto de vetores dados pelas Equações (6.9), (6.10), (6.11) e (6.12).

$$V_{4m}^{1} = fcc\left(V_{2m}^{1}, V_{2m}^{1}\right) = (1,1,1,1) \tag{6.9}$$

$$V_{4m}^{2} = fcc\left(V_{2m}^{1}, -V_{2m}^{1}\right) = (1,1,-1,-1) \tag{6.10}$$

$$V_{4m}^{3} = fcc\left(V_{2m}^{2}, V_{2m}^{2}\right) = (1,-1,1,-1) \tag{6.11}$$

$$V_{4m}^{4} = fcc\left(V_{2m}^{2}, -V_{2m}^{2}\right) = (1,-1,-1,1) \tag{6.12}$$

Seguindo o algoritmo, em um possível Passo 6, obtêm-se os vetores dados pelas Equações 6.13, 6.14, 6.15, 6.16, 6.17, 6.18, 6.19 e 6.20.

$$V_{8m}^{1} = fcc\left(V_{4m}^{1}, V_{4m}^{1}\right) = (1,1,1,1,1,1,1,1) \tag{6.13}$$

$$V_{8m}^{2} = fcc\left(V_{4m}^{1}, -V_{4m}^{1}\right) = (1,1,1,1,-1,-1,-1,-1) \tag{6.14}$$

$$V_{8m}^{3} = fcc\left(V_{4m}^{2}, V_{4m}^{2}\right) = (1,1,-1,-1,1,1,-1,-1) \tag{6.15}$$

$$V_{8m}^{4} = fcc\left(V_{4m}^{2}, -V_{4m}^{2}\right) = (1,1,-1,-1,-1,-1,1,1) \tag{6.16}$$

$$V_{8m}^{5} = fcc\left(V_{4m}^{3}, V_{4m}^{3}\right) = (1,-1,1,-1,1,-1,1,-1) \tag{6.17}$$

$$V_{8m}^{6} = fcc\left(V_{4m}^{3}, -V_{4m}^{3}\right) = (1,-1,1,-1,-1,1,-1,1) \tag{6.18}$$

$$V_{8m}^{7} = fcc\left(V_{4m}^{4}, V_{4m}^{4}\right) = (1,-1,-1,1,1,-1,-1,1) \tag{6.19}$$

$$V_{8m}^{8} = fcc\left(V_{4m}^{4}, -V_{4m}^{4}\right) = (1,-1,-1,1,-1,1,1,-1) \tag{6.20}$$

Se for realizado o produto interno tomando cada par de vetores, verificar-se-á que serão iguais a zero. Dessa forma, são obtidos oito VBOs, e cada qual com oito componentes.

6.3 Observações sobre os vetores-alvo

O produto interno (dado pela Equação 5.1) entre dois VBCs, aumenta à medida que sua dimensão aumenta, diminuindo o ângulo (dado pela Equação 5.2) entre eles. Em outras palavras, se a dimensionalidade do espaço de saída é suficientemente elevada, a diferença entre dois VBCs torna-se cada vez menor.

Por outro lado, de acordo com o que foi discutido na seção 5.2, se o ângulo entre dois vetores é de 90 graus, o produto interno entre eles é nulo. Nesse caso, os vetores são ortogonais e a distância euclidiana entre eles é grande em comparação a vetores convencionais.

A distância euclidiana entre dois VBCs é igual a $2\sqrt{2}$ conforme cálculo mostrado pela Equação (6.21), independentemente da dimensão. No caso dos VBOs, a distância euclidiana entre dois vetores aumenta à medida que a dimensão desses vetores também aumenta, ou seja, quanto maior for a dimensão dos vetores do tipo VBO, maior será a distância euclidiana entre eles.

$$d_{U,W} = \sqrt{(-1-(-1))^2 + ... + (1-(-1))^2 + (-1-1)^2 + ... + (-1-(-1))^2} = \sqrt{2 \cdot 4} = 2\sqrt{2} \tag{6.21}$$

Pela característica dos VBOs, tem-se que, para vetores de dimensão n, metade das n diferenças da fórmula da distância se anularão mutuamente, e a outra metade é igual a $2^2 = 4$.

Por essa razão, a fórmula da Equação (5.2) pode ser reescrita na Equação (6.22).

$$d_{U,W} = \sqrt{\frac{n}{2} \cdot 4} = \sqrt{2n} \tag{6.22}$$

O gráfico da Figura 6.1 mostra a evolução da distância euclidiana de acordo com o aumento da dimensão entre os vetores-alvo. Com VBC e VNO, a distância permanece inalterada, valendo sempre $2\sqrt{2}$. Para VBOs, essa distância é sempre crescente.

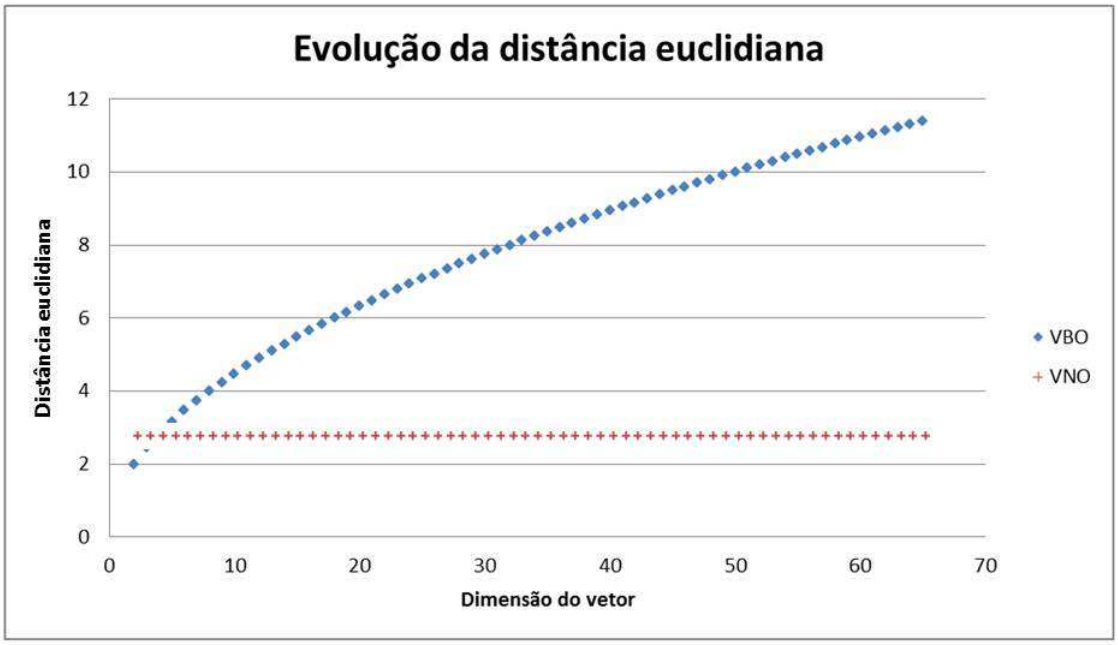


Figura 6.1: Distância euclidiana de vetores-alvo

Vetores-alvo do tipo VBNs são mutuamente ortogonais. Contudo a distância euclidiana entre cada par de VBNs é sempre igual a $2\sqrt{2}$.

No caso dos VNOs, eles podem ter o mesmo tamanho dos VBOs, mas seu produto interno é não nulo. A diferença dos VNOs em relação aos VBCs é somente a dimensão. O produto interno entre dois vetores do tipo VNO é também sempre igual a $2\sqrt{2}$. Se o tamanho dos VNOs é grande, então o produto interno correspondente entre eles é grande. Assumindo

que o produto interno pode representar uma métrica de similaridade entre VNOs, a semelhança aumenta quando seus tamanhos aumentam. Este trabalho parte da hipótese de que a alta similaridade entre os vetores-alvo pode causar baixo desempenho em MLPs.

6.4 O algoritmo *backpropagation* e os efeitos da distância euclidiana no desempenho da rede – discussão matemática

Redes neurais artificiais do tipo MLP utilizam o algoritmo de treinamento intitulado *backpropagation*. Esse algoritmo é deduzido por meio da função erro obtida pela diferença entre a saída encontrada e a saída desejada (L. Fausett, 1994). O algoritmo foi deduzido com o objetivo de se reduzir o valor do erro quadrático médio. Assim, a função erro é submetida a um algoritmo de gradiente descendente. Para uma discussão matemática apropriada, é necessário fornecer algumas fórmulas.

Considere x_i como o valor da i-ésima entrada, t_k como o valor da k-ésima saída desejada, z_j como o j-ésimo valor que chega à camada oculta e y_k como a k-ésima saída encontrada. Considere v_{ij} como o peso sináptico entre a i-ésima entrada e o j-ésimo neurônio da camada oculta, v_{0ij} como o peso sináptico do tipo bias do j-ésimo neurônio da camada oculta, w_{ij} como o peso sináptico entre o j-ésimo neurônio da camada oculta e o k-ésimo neurônio da camada de saída e w_{0ij} como o peso sináptico do tipo bias correspondente ao k-ésimo neurônio da camada de saída. As Equações (6.23) e (6.24) mostram o cálculo de z_j e as Equações (6.25) e (6.26) mostram o cálculo de y_k (L. Fausett, 1994).

$$zin_j = v_{0j} + \sum_i x_i v_{ij} \tag{6.23}$$

$$z_j = f\left(zin_j\right) \tag{6.24}$$

$$yin_k = w_{0k} + \sum_j z_j w_{jk} \tag{6.25}$$

$$y_k = f\left(yin_k\right) \tag{6.26}$$

A função erro é mostrada pela Equação (6.27), na qual t_k é a saída desejada e y_k é a saída encontrada. As Equações (6.28), (6.29), (6.30), (6.31) e (6.32) mostram o cálculo do gradiente descendente da função erro em relação aos pesos da camada de saída. A Equação (6.33) mostra a fórmula de atualização dos pesos da camada de saída.

$$E = \frac{1}{2} \sum_k \left[t_k - y_k\right]^2 \tag{6.27}$$

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \frac{1}{2} \sum_k \left[t_k - y_k\right]^2 \tag{6.28}$$

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \frac{1}{2} \left[t_k - f\left(yin_k\right)\right]^2 \tag{6.29}$$

$$\frac{\partial E}{\partial w_{jk}} = -\left[t_k - y_k\right] \frac{\partial}{\partial w_{jk}} f\left(yin_k\right) \tag{6.30}$$

$$\frac{\partial E}{\partial w_{jk}} = -\left[t_k - y_k\right] f'\left(yin_k\right) \frac{\partial}{\partial w_{jk}} \left(yin_k\right) \tag{6.31}$$

$$\frac{\partial E}{\partial w_{jk}} = -\left[t_k - y_k\right] f'\left(yin_k\right) z_j \tag{6.32}$$

$$\delta_k = \left[t_k - y_k\right] f'\left(yin_k\right) \tag{6.33}$$

O cálculo do gradiente descendente da função erro em relação aos pesos da camada oculta é mostrado pelas Equações (6.34), (6.35), (6.36), (6.37) e (6.38). A fórmula de atualiza-

ção dos pesos da camada oculta é mostrada pela Equação (6.39).

$$\frac{\partial E}{\partial v_{ij}} = -\sum_k [t_k - y_k] \frac{\partial}{\partial v_{ij}} y_k \tag{6.34}$$

$$\frac{\partial E}{\partial v_{ij}} = -\sum_k [t_k - y_k] f'(yin_k) \frac{\partial}{\partial v_{ij}} yin_k \tag{6.35}$$

$$\frac{\partial E}{\partial v_{ij}} = -\sum_k \delta_k \frac{\partial}{\partial v_{ij}} yin_k \tag{6.36}$$

$$\frac{\partial E}{\partial v_{ij}} = -\sum_k \delta_k w_{jk} \frac{\partial}{\partial v_{ij}} z_j \tag{6.37}$$

$$\frac{\partial E}{\partial v_{ij}} = -\sum_k \delta_k w_{jk} f'(zin_j) [x_i] \tag{6.38}$$

$$\delta_j = \sum_k \delta_k w_{jk} f'(zin_j) \tag{6.39}$$

É possível expressar matematicamente a função distância euclidiana em relação à função erro. Se a distância euclidiana diminui, o erro também diminui. Se a distância euclidiana aumenta, o erro também aumenta. A função distância euclidiana é mostrada pela Equação (6.40).

$$D = \left(\sum_k [t_k - y_k]^2 \right)^{\frac{1}{2}} \tag{6.40}$$

Assim, a função distância euclidiana é a raiz quadrada do dobro da função erro $D = \sqrt{2E}$. Aplicando o gradiente descendente para a função distância euclidana, obtém-se as fórmulas das Equações (6.41) e (6.42).

$$\delta_k = \frac{[t_k - y_k] f'(yin_k)}{\sqrt{\sum_k [t_k - y_k]^2}} \tag{6.41}$$

$$\delta_j = \sum_j \delta_k w_{jk} f'(zin_j) \tag{6.42}$$

Assim, as fórmulas obtidas pela aplicação do gradiente descendente para a função erro também estão relacionadas com as fórmulas obtidas pela aplicação do gradiente descendente para a função da distância euclidiana. Pode-se deduzir que, se os pontos do espaço de saída, intitulados "alvos" da rede MLP, estão mais distantes uns dos outros, as chances de uma saída obtida pela inserção de um determinado padrão se aproximar de alvos correspondentes a outros padrões são menores.

Os pontos do espaço de saída são equidistantes para qualquer tipo de vetor no espaço R^n. São considerados dois vetores-alvo distintos. Para fins de ilustração, considere-se também a representação desses pontos no plano. Ao longo do treinamento, cada saída obtida é projetada no espaço R^n. Nessa ilustração, cada saída é projetada no plano. Com a evolução dos ciclos, as saídas projetadas formam uma região de convergência em torno da saída desejada. A figura 6.2 ilustra essa discussão.

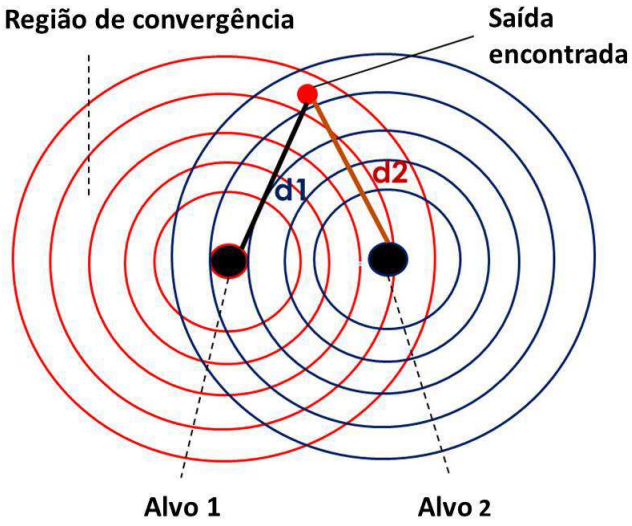


Figura 6.2: Ilustração de regiões de convergência

Nessa ilustração está clara a existência de uma grande quantidade de pontos pertencentes às duas regiões de convergência. Esses pontos dentro da intersecção das regiões de

convergência estão mais propensos a estar mais próximos de alvos correspondentes a outros padrões, ou seja, mais próximos de alvos incorretos. Isso faz com que a MLP classifique padrões incorretamente.

Contudo, se os alvos estão mais distantes uns dos outros, a intersecção entre as regiões de convergência é bem menor. Assim, a taxa de classificação é beneficiada, porque há menos saídas propensas a erros de classificação. A Figura (6.3) ilustra dois alvos dispostos a uma maior distância euclidiana.

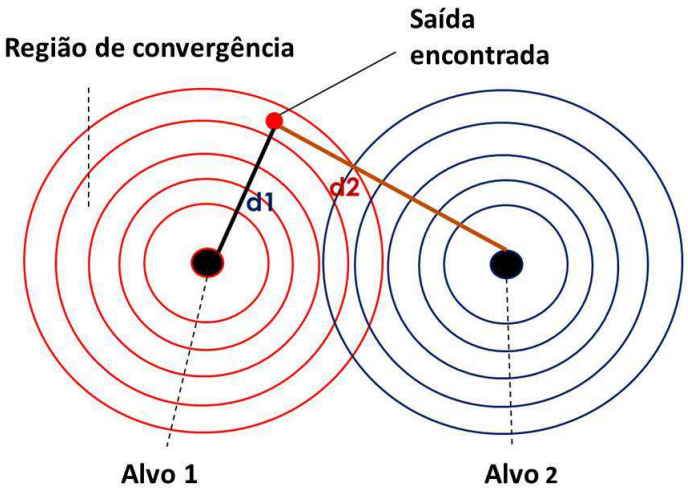


Figura 6.3: Ilustração de regiões de convergência com alvos distantes

Isso explica por que as redes do tipo MLP treinadas com VBOs têm melhor desempenho global. Também explica a superioridade de VBOs com poucos ciclos de treinamento. A característica dos VBOs de possuírem maior distância euclidiana reduz a interferência entre as regiões de convergência.

Capítulo 7

A redução da suscetibilidade ao erro de classificação de redes Multilayer Perceptron com o uso de alvos ortogonais

7.1 Dados experimentais

Os dados utilizados neste trabalho foram capturados por terceiros e disponibilizados por seus responsáveis. As seções seguintes explicam, de acordo com os mesmos, sobre o processo de captura e as adaptações a este trabalho.

7.1.1 Dígitos manuscritos

Foram realizados experimentos com dados de dígitos manuscritos obtidos por meio do repositório internacional conhecido como Semeion Handwritten Digit of Machine Learning Repository (Lichman, 2013). Esses padrões foram obtidos de um grupo de cerca de 80 pessoas que foram convidadas a escrever duas vezes os dígitos de 0 a 9. Na primeira requisição, as pessoas deveriam escrever os dígitos calmamente, primando pela perfeição da escrita. Na segunda requisição, as pessoas foram induzidas a escrever os dígitos rapidamente, sem se preocupar com a legibilidade (Lichman, 2013).

Cada figura foi escaneada em uma imagem contendo 256 pixels no formato de 16 linhas e 16 colunas. Cada imagem foi processada em uma escala de resolução de 256 níveis de cinza. Posteriormente, a matriz de pixels foi transformada em um vetor linha de 256 componentes, sendo que cada linha foi posicionada imediatamente à direita de sua linha superior na matriz. Para cada pixel correspondente ao fundo da imagem, foi atribuído o valor 0, e, para cada pixel correspondente à imagem, foi atribuído o valor 1, conforme informações descritas no repositório (Lichman, 2013). Nesse trabalho, os pixels correspondentes ao fundo da imagem receberam o valor "-1" no lugar de "0".

7.1.2 Íris humana

Também foram realizados experimentos com íris humanas obtidas da Chinese Academy of Sciences - Institute of Automation database denominada CASIA (Casia, 2010). O conjunto de dados contém íris de 108 indivíduos, em que para 71 deles há um conjunto completo de sete imagens. Por essa razão, foram utilizados os dados correspondentes a esses 71 indivíduos. Para cada teste, foram utilizadas quatro imagens para a fase de treinamento. De acordo com o repositório CASIA, essas imagens foram obtidas com o uso de luz infravermelha para obter as características de íris com contraste suficiente ao reconhecimento do padrão biométrico.

Os passos para o processamento das imagens da íris são descritos a seguir. O primeiro passo consiste na localização da região da íris na imagem, na qual é utilizada a transformação de Hough Circular. Em seguida, a região da íris, que tem formato de anel, é normalizada de modo a ser representada como uma matriz retangular. Finalmente, a extração das características da íris é realizada, que neste trabalho, deu-se por convolução da imagem normalizada com o chamado filtro de Log Gabor. A filtragem resulta em coeficientes complexos, cujas fases são quantizadas para um dos quatro quadrantes do plano complexo. Cada quadrante é referenciado por dois bits, e um modelo binário é criado (Daugman, 1993; Negin et al., 2000; Manzan, Yamanaka, & Nomura, 2011). Para cada imagem existem 8640 pixels dispostos em 18 círculos concêntricos,

cada um contendo 480 pixels.

Nesta pesquisa, os autores utilizaram somente as cinco primeiras circunferências a partir do centro do círculo, eliminando a interferência dos cílios e reduzindo o esforço computacional para a rede MLP (Manzan, Nomura, Yamanaka, Carneiro, & Veiga, 2012b). Assim, cada padrão de treinamento corresponde a um conjunto de $5 \times 480 = 2400$ pixels. Os pixels brancos foram representados por "-1", e os pixels pretos, por "1". Assim, os vetores de treinamento foram construídos para representar linhas simples contendo 2400 pixels que conectam os pontos do primeiro ao quinto círculo.

7.1.3 Signos da linguagem australiana de sinais

Finalmente, foram realizados experimentos com signos da linguagem de sinais australiana (Australian linguagem gestual). Foram capturadas 27 amostras de cada um dos 95 signos australianos usando-se rastreadores de posição de alta qualidade de indivíduos nativos (Kadous, 2002).

Os dados foram capturados usando-se uma configuração de dispositivos conforme descrito a seguir:

- Duas luvas de quinta dimensão (5DT), uma para a mão direita e outra para a mão esquerda;
- Dois rastreadores magnéticos de posição do tipo "Flock-de-Birds", um ligado a cada lado;
- Um cartão de série de quatro portas para lidar com quatro fontes de dados;
- Um PC (128MB RAM, Intel Pentium II 266MHz).

Os seguintes dados foram registrados para cada lado:

- Posição x: expressa a posição relativa em relação a um ponto ajustado ligeiramente abaixo do queixo (zero), em metros.
- Posição y: expressa a posição relativa em relação a um ponto ajustado ligeiramente abaixo do queixo (zero), em metros.

- Posição z: expressa a posição relativa em relação a um ponto ajustado ligeiramente abaixo do queixo (zero), em metros.
- Roll: expressa como um valor entre $-0,5$ e $0,5$, com 0 sendo palma para baixo. Positivo, significa que a palma gira no sentido horário a partir da perspectiva do pronunciador. Para adquirir diferentes graus, o valor é multiplicado por 180.
- Pitch: expressa como um valor entre $-0,5$ e $0,5$, com 0 sendo palma plana (horizontal). Positivo, significa que a palma está apontando para cima. Para adquirir diferentes graus, o valor é multiplicado por 180.
- Yaw: expressa um valor entre $-1,0$ e $1,0$, sendo 0 a palma para a frente a partir da perspectiva do pronunciador. Positivo, significa movimento no sentido horário a partir da perspectiva acima do pronunciador. Para adquirir diferentes graus, o valor é multiplicado por 180.
- Medida da curva do polegar entre 0 e 1. O valor 0 significa totalmente plana, e 1 significa totalmente dobrado. No entanto as medições na região da articulação do dedo não são muito precisas.
- Medida da curva do dedo indicador entre 0 e 1. O valor 0 indica o dedo totalmente plano, e 1 indica que o dedo está totalmente dobrado. No entanto as medições não são muito precisas.
- Medida da curva do dedo médio entre 0 e 1. O valor 0 indica o dedo totalmente plano, e o valor 1 indica o dedo totalmente dobrado. No entanto as medições não são muito precisas.
- Medida da curva do dedo anelar compreendida entre 0 e 1. O valor 0 indica o dedo totalmente plano, e o valor 1 indica o dedo totalmente dobrado. No entanto as medições não são muito precisas.
- Medida da curva do dedo mindinho compreendida entre 0 e 1. O valor 0 indica o dedo totalmente plano, e o valor 1 indica o dedo totalmente dobrado. No entanto as medições não são muito precisas.

7.2 Planejamento experimental e estatístico

Além da discussão matemática apresentada no Capítulo 6, foram realizados experimentos no reconhecimento de três tipos de dados reais, utilizando as redes do tipo MLP. Os dados correspondem a dígitos manuscritos, imagens de íris humana e linguagem de sinais australiana (signos australianos). O objetivo dos experimentos foi avaliar a distância euclidiana entre a saída obtida pela rede MLP proveniente da inserção de um determinado padrão e os alvos correspondentes aos demais padrões em tempo de treinamento. Em outras palavras, foi avaliada a distância euclidiana entre cada saída encontrada e os alvos incorretos.

Os experimentos foram realizados para a utilização de testes estatísticos. A geração dos pesos sinápticos iniciais foi aleatória. Considerando que os pesos iniciais interferem no treinamento da rede MLP, pode-se concluir que as diferenças entre saídas encontradas e alvos incorretos é um processo aleatório. Foram utilizadas sete redes no treinamento para cada tipo de dado. Cada uma foi treinada com um tipo e tamanho diferente de vetor-alvo. Os vetores utilizados nos experimentos com dígitos manuscritos são VBCs de tamanho 10, VNOs e VBOs de tamanhos 16, 32 e 64, conforme descrição apresentada na Seção 6.1. Nos experimentos com íris humana e signos australianos, foram utilizados VBCs de tamanho 71 e 95 respectivamente, VNOs e VBOs de tamanhos 128, 256 e 512.

Os experimentos foram realizados por 50 ciclos. Para cada modelo foram realizados 500 experimentos. Em cada experimento, os pesos sinápticos iniciais foram obtidos por geração aleatória com valores entre -0.2 e 0.2. A quantidade de ciclos e o critério de geração de pesos sinápticos foram obtidos empiricamente após a análise de diversos experimentos preliminares.

Nos experimentos com dígitos, foram utilizadas 45 amostras de cada dígito no treinamento. Portanto foram utilizadas 450 amostras para o treinamento da rede MLP. Foram utilizadas outras 45 amostras de cada dígito na fase de teste. Nos experimentos com íris humana foram utilizadas 2 imagens de cada um dos 71 indivíduos para treinamento e outras 2 amostras para teste. Portanto foram utilizadas 142 amostras de íris na fase de treinamento e outras 142 na fase de teste. Para os experimentos com signos australianos, foram utilizadas 9

amostras de cada um dos 95 tipos de sinais existentes para treinamento e outras 9 para a etapa de teste. Portanto foram utilizadas 855 amostras de signos de sinais australianos para cada etapa (treinamento e teste).

Os experimentos foram realizados com computadores e configurações de sistema operacional exatamente iguais. O programa de simulação foi criado no software Matlab(R) 2013.

O algoritmo utilizou a taxa de aprendizagem adaptativa e Termo Momentum. Para determinar o tamanho da camada oculta e a taxa de aprendizagem inicial, foi utilizado um algoritmo genético (AG). A taxa de acerto foi atribuída à função de aptidão. O AG foi executado com uma taxa de cruzamento de 90%, taxa de mutação de 10% e por 100 gerações. Foram utilizadas populações de 30 indivíduos, com elitismo de 2 indivíduos e seleção dos indivíduos aptos às operações de cruzamento por meio de torneio. Foram levados em consideração apenas os vetores VBCs e VNOs. A ideia é mostrar que, mesmo no melhor cenário de parâmetros dos vetores não ortogonais, os VBOs são superiores. A Tabela 7.1 mostra os parâmetros obtidos com a utilização do algoritmo genético para dígitos manuscritos. As Tabelas 7.2 e 7.3 mostram os parâmetros obtidos com a utilização de algoritmo genético, respectivamente, para íris humana e para signos australianos.

Tabela 7.1: Parâmetros obtidos com algoritmo genético - dígitos manuscritos

Dimensão do vetor-alvo	**Número de neurônios na camada oculta**	**Taxa de aprendizagem inicial**
10	95	0.0035
16	122	0.0025
32	152	0.0027
64	132	0.0034

Tabela 7.2: Parâmetros obtidos com algoritmo genético - íris humana

Dimensão do vetor-alvo	**Número de neurônios na camada oculta**	**Taxa de aprendizagem inicial**
71	425	0.03910
128	565	0.01345
256	513	0.03487
512	545	0.05312

Tabela 7.3: Parâmetros obtidos com algoritmo genético - signos australianos

Dimensão do vetor-alvo	Número de neurônios na camada oculta	Taxa de aprendizagem inicial
95	117	0.0047
128	131	0.0101
256	197	0.0091
512	389	0.0019

Considera-se como modelo cada rede MLP treinada com um tipo de vetor-alvo. A determinação da medida de comparação foi obtida a partir dos seguintes passos:

1. Para cada padrão propagado pela rede durante a etapa de treinamento, foram calculadas as distâncias euclidianas entre a saída obtida pela rede MLP e os alvos dos demais padrões (alvos incorretos). Considerando a existência de 10 padrões (10 dígitos), foram calculadas 9 distâncias euclidianas. No caso das íris, são 71 indivíduos, portanto 70 distâncias a alvos incorretos. Para signos australianos, são 94 distâncias, uma vez que são 95 tipos de signos.
2. Foi calculada a média das 9 distâncias euclidianas referentes aos dígitos, das 70 distâncias referentes às imagens das íris e das 94 distâncias referentes aos signos. Esse tipo de média foi denominado média do Tipo 1.
3. A média do Tipo 1 foi calculada para cada uma das 450 amostras de dígitos, 142 amostras de íris e 855 amostras de signos australianos.
4. Ao final de cada ciclo, foi calculada a média de todas as médias do Tipo 1, denominada média do Tipo 2. Considerando que cada treinamento foi realizado durante 50 ciclos, cada treinamento gerou 50 médias do Tipo 2.
5. Para cada tipo de vetor-alvo, foram realizados 500 treinamentos. Com esse procedimento, cada ciclo conta com 500 médias do Tipo 2.
6. Para cada ciclo, foi calculada a média das 500 médias do Tipo 2, denominada média do Tipo 3. Esses procedimentos foram realizados com modelos obtidos com a utilização dos vetores-alvo do tipo VBC10, VNO16, VNO32, VNO64, VBO16, VBO32 e VBO64. Nos experimentos com íris humana, os mesmos procedimentos

foram realizados com vetores-alvo do tipo VBC71, VNO128, VNO256, VNO512, VBO128, VB0256 E VBO512. Os procedimentos também foram realizados com vetores-alvo do tipo VBC95, VNO128, VNO256, VNO512, VBO128, VB0256 e VBO512 em experimentos com signos australianos.

7. Foram realizadas comparações estatísticas pelo teste de Mann-Whitney das médias do Tipo 3 obtidas com a utilização dos sete tipos de vetores-alvo em cada um dos 3 tipos de dados. As comparações foram feitas em relação aos ciclos 1, 5, 10, 15, 20, 25, 30, 35, 40, 45 e 50.

A figura 7.1 ilustra a determinação da média do Tipo 3. A média do Tipo 3 resume médias de distâncias entre saídas encontradas pela propagação de um padrão pela rede com os alvos correspondentes aos demais padrões. Portanto quanto maior for a média do Tipo 3, menor será a chance de ocorrer classificação incorreta do padrão por parte da rede do tipo MLP.

7.3 Resultados experimentais

A Tabela 7.4 mostra as médias do Tipo 3 para cada um dos tipos de vetores-alvo descritos na seção 7.2 referentes aos experimentos com dígitos manuscritos. As médias do Tipo 3 referem-se aos ciclos 1, 5, 10, 15, 20, 25, 30, 35, 40, 45 e 50. As Tabelas 7.5 e 7.6 mostram, respectivamente, as médias do Tipo 3 para experimentos com íris humana e signos australianos.

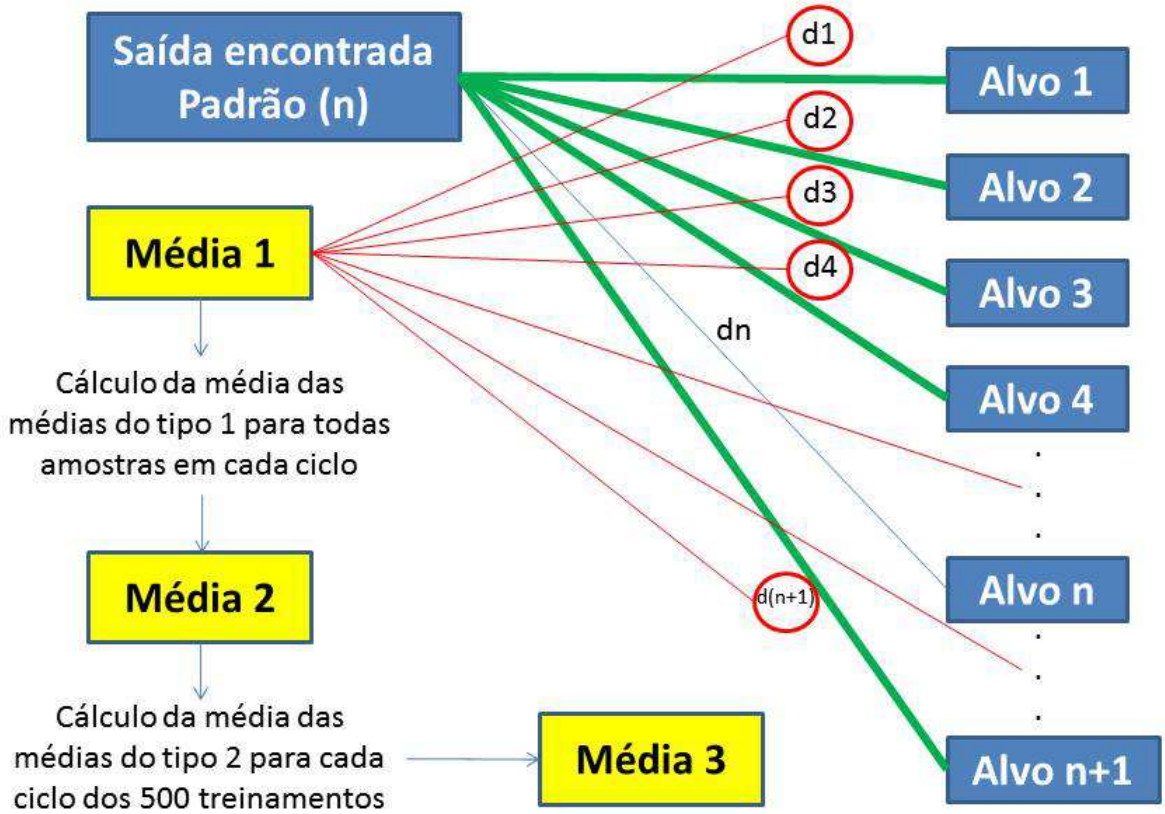


Figura 7.1: Esquema de determinação da Média do Tipo 3

Tabela 7.4: Média do Tipo 3 - dígitos manuscritos

Ciclo	VBC10	VNO16	VNO32	VNO64	VBO16	VBO32	VBO64
1	2.1284	2.1254	2.1213	2.1388	4.4179	6.1845	8.4209
5	2.3231	2.3087	2.2833	2.2816	4.7847	6.7127	9.2042
10	2.3780	2.3689	2.3464	2.3457	4.8643	6.8342	9.4251
15	2.4095	2.4025	2.3812	2.3812	4.8976	6.8864	9.5397
20	2.4293	2.4234	2.4043	2.4034	4.9175	6.9199	9.6157
25	2.4417	2.4378	2.4200	2.4186	4.9310	6.9424	9.6720
30	2.4493	2.4472	2.4296	2.4282	4.9404	6.9592	9.7141
35	2.4545	2.4528	2.4367	2.4351	4.9473	6.9722	9.7459
40	2.4582	2.4568	2.4410	2.4399	4.9528	6.9828	9.7707
45	2.4614	2.4597	2.4439	2.4434	4.9576	6.9914	9.7901
50	2.4641	2.4624	2.4463	2.4456	4.9616	6.9986	9.8062

Tabela 7.5: Média do Tipo 3 - íris humana

Ciclo	VBC71	VNO128	VNO256	VNO512	VBO128	VBO256	VBO512
1	2.0564	2.0717	2.1129	2.0569	15.4877	21.9734	31.1614
5	1.9634	1.9636	1.9662	1.9632	15.6670	22.1858	31.4142
10	1.9730	1.9717	1.9740	1.9732	15.6948	22.2188	31.4505
15	2.0007	1.9939	1.9988	2.0019	15.7090	22.2341	31.4660
20	2.0519	2.0352	2.0461	2.0526	15.7170	22.2419	31.4732
25	2.1171	2.0961	2.1156	2.1172	15.7218	22.2468	31.4774
30	2.1985	2.1692	2.2037	2.1977	15.7244	22.2495	31.4796
35	2.2801	2.2527	2.2973	2.2798	15.7260	22.2510	31.4811
40	2.3571	2.3329	2.3825	2.3567	15.7268	22.2520	31.4820
45	2.4237	2.4049	2.4524	2.4227	15.7276	22.2526	31.4826
50	2.4759	2.4604	2.5087	2.4747	15.7280	22.2530	31.4830

O gráfico da Figura 7.2 mostra as médias do Tipo 3 para todos os 50 ciclos de treinamento, levando-se em consideração a utilização dos sete tipos de vetores-alvo. Os gráficos para as médias do Tipo 3 referentes aos experimentos com íris humana e signos australianos são mostrados, respectivamente, pelas Figuras 7.3 e 7.4.

A Tabela 7.7 mostra os resultados do teste de normalidade de Kolmogorov-Smirnov (Conover, 1999) referentes às médias do Tipo 3 obtidas nos experimentos com dígitos manuscritos. A estatística desse teste é representada pela letra D. Nesse teste, todos os *valores-p* são menores do que $2.2E-16$. Portanto os dados não se ajustam à distribuição normal. O mesmo fenômeno ocorre para as médias do Tipo 3 obtidas nos experimentos com íris humana e signos

Tabela 7.6: Média do Tipo 3 - signos australianos

Epoch	VBC95	VNO128	VNO256	VNO512	VBO128	VBO256	VBO512
1	2.0567	2.0706	2.1131	2.1540	11.2275	15.9105	22.6315
5	1.9632	1.9636	1.9656	1.9705	11.5453	16.3551	23.3206
10	1.9732	1.9717	1.9730	1.9926	11.8777	16.8288	24.0614
15	2.0014	1.9942	1.9979	2.0694	12.1653	17.2241	24.6349
20	2.0513	2.0356	2.0465	2.1792	12.4041	17.5402	25.0642
25	2.1170	2.0951	2.1157	2.2844	12.6014	17.8237	25.4706
30	2.1977	2.1698	2.2038	2.3855	12.7627	18.0658	25.8321
35	2.2793	2.2536	2.2975	2.4729	12.8941	18.2685	26.1433
40	2.3580	2.3329	2.3820	2.5402	13.0028	18.4349	26.4156
45	2.4231	2.4035	2.4526	2.5874	13.0917	18.5775	26.6449
50	2.4758	2.4603	2.5084	2.6193	13.1670	18.6988	26.8439

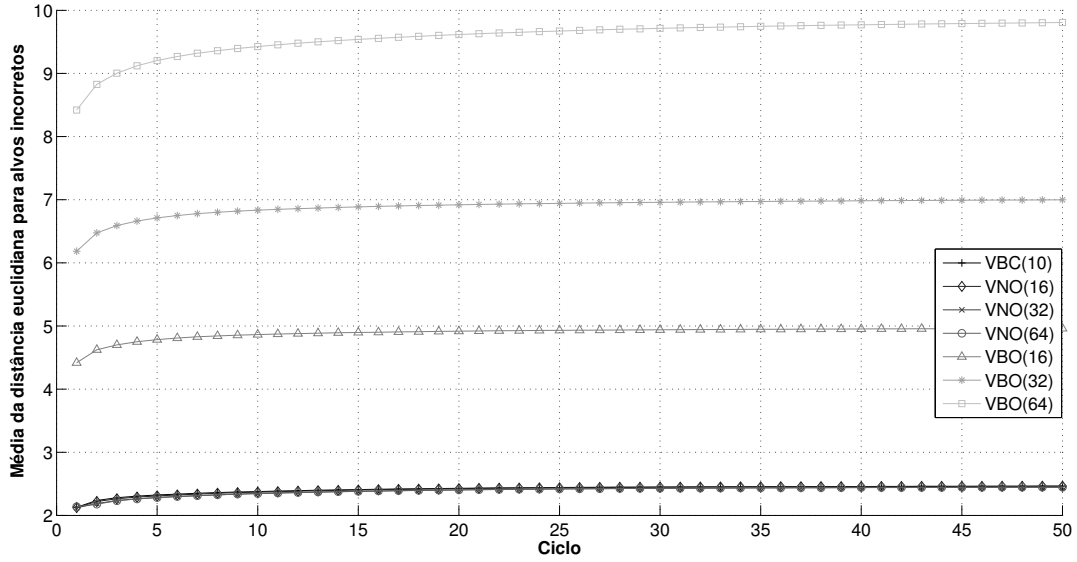


Figura 7.2: Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - dígitos manuscritos

australianos, como pode ser visto nas tabelas 7.8 e 7.9.

Por essa razão, o teste não paramétrico de Mann-Whitney foi utilizado para se compararem as médias do Tipo 3 (Conover, 1999) dos três tipos de conjuntos de dados.

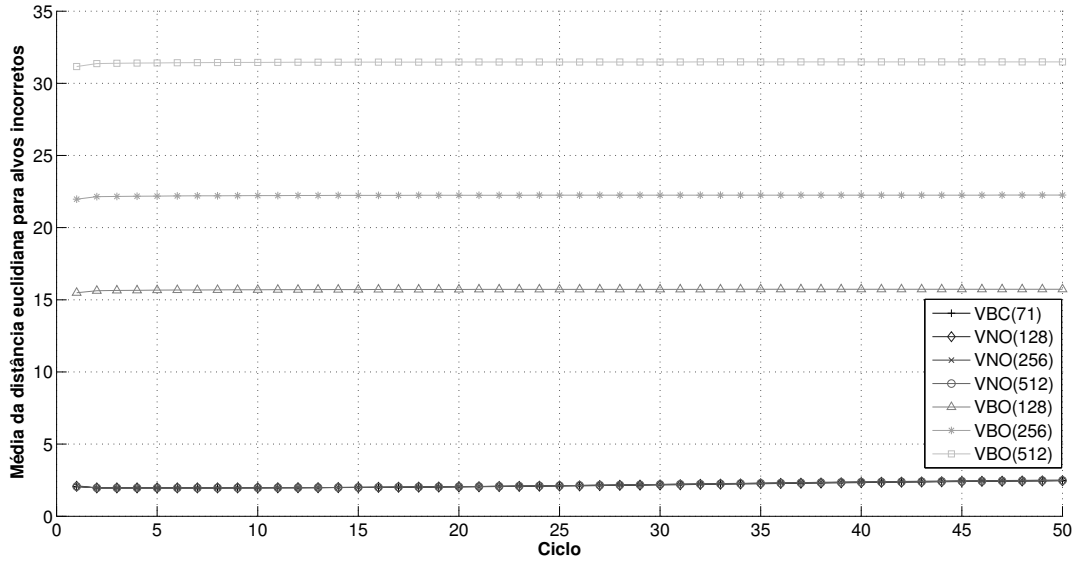


Figura 7.3: Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - íris humana

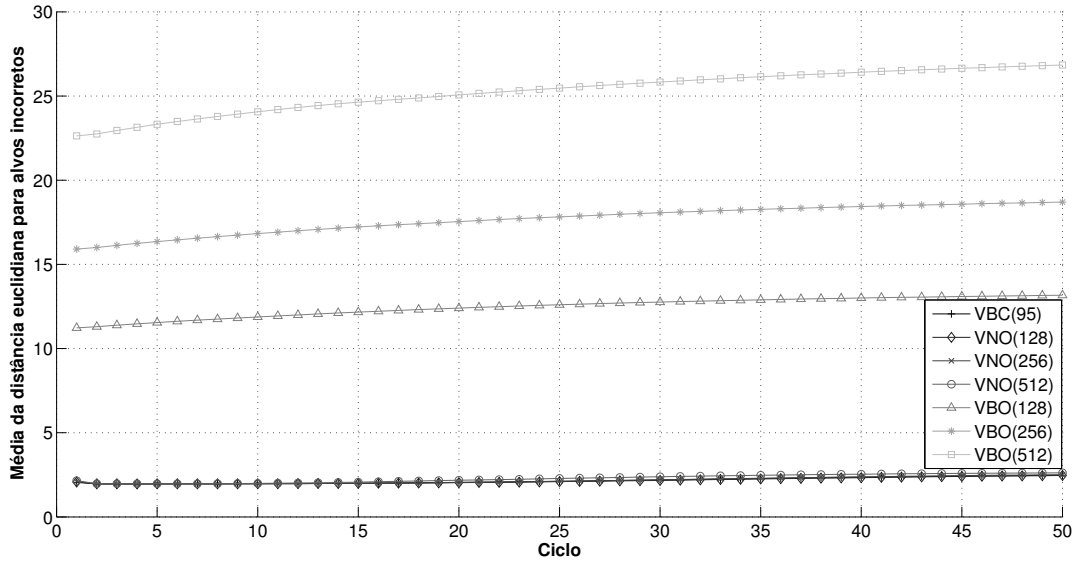


Figura 7.4: Comparação da média do Tipo 3 para todos os tipos de vetores-alvo - signos australianos

A Tabela 7.10 mostra os resultados do teste estatístico de Mann-Whitney de todas as comparações com o uso de diferentes vetores-alvo nos experimentos com dígitos manuscritos. A hipótese nula é que não existem diferenças significativas entre as médias, e a hipótese alternativa é que existem diferenças significativas entre as médias. O teste foi realizado com o

Tabela 7.7: Teste para normalidade de Kolmogorov-Smirnov - dígitos manuscritos

	Ciclos										
Ciclo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
D	0.34	0.33	0.34	0.34	0.34	0.34	0.34	0.34	0.34	0.34	0.34

Tabela 7.8: Teste para normalidade de Kolmogorov-Smirnov - íris humana

	Ciclos										
Ciclo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
D	0.10	0.12	0.13	0.13	0.13	0.13	0.12	0.12	0.12	0.12	0.12

software de análises estatísticas R (R Core Team, 2014). Para cada comparação é gerada uma estatística do teste representada como "valor-p". Entende-se por "valor-p" sob a hipótese nula (considerada como verdadeira) a probabilidade de obtenção de um valor igual ou mais extremo do que o valor obtido na amostra. Asteriscos indicam que não existem evidências para aceitar a hipótese nula ao nível de 1% de significância. De modo análogo, as Tabelas 7.11 e 7.12 mostram, respectivamente, os resultados do teste estatístico de Mann-Whitney referentes aos experimentos com íris humana e signos australianos.

A Tabela 7.13 mostra as comparações simultâneas das médias do Tipo 3 pelo teste estatístico de Mann-Whitney nos experimentos com dígitos manuscritos. Letras iguais indicam que não existem diferenças significativas ao nível de 1% pelo teste de Mann-Whitney. Letras diferentes indicam que existem diferenças significativas ao nível de 1% pelo teste de Mann-Whitney. A ordem alfabética das letras indica a superioridade das médias. Por exemplo, na comparação dos desempenhos obtidos com o uso de VBO64 e VBO32 para o ciclo 1, verificou-se a ocorrência de "valor-p" menor do que 0.01. A média obtida com o uso de VBO64 é maior

Tabela 7.9: Teste para normalidade de Kolmogorov-Smirnov - signos australianos

	Ciclos										
Ciclo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
D	0.35	0.36	0.36	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.35

Tabela 7.10: Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - dígitos manuscritos

	Ciclos										
Comparação	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBC10 x VNO16	0.48	0.29	*	0.01	0.01	*	*	*	*	*	0.01
VBC10 x VNO32	*	0.38	*	0.28	0.13	0.02	*	*	*	*	0.04
VBC10 x VNO64	0.02	*	0.31	0.74	0.92	0.42	*	0.14	0.12	0.18	0.39
VBC10 x VBO16	*	*	*	*	*	*	*	*	*	*	*
VBC10 x VBO32	*	*	*	*	*	*	*	*	*	*	*
VBC10 x VBO64	*	*	*	*	*	*	*	*	*	*	*
VNO16 x VNO32	0.13	0.04	0.53	0.76	0.84	0.55	0.23	0.14	0.17	0.23	0.42
VNO16 x VNO64	*	*	*	0.02	0.03	0.07	0.16	0.26	0.34	0.37	0.26
VNO16 x VBO32	*	*	*	*	*	*	*	*	*	*	*
VNO16 x VBO64	*	*	*	*	*	*	*	*	*	*	*
VNO32 x VNO64	*	0.08	0.03	0.08	0.04	0.03	0.01	*	0.02	0.03	0.05
VNO32 x VNO64	*	*	0.37	0.03	*	0.43	0.11	*	*	*	*
VNO32 x VBO16	*	*	*	*	*	*	*	*	*	*	*
VNO32 x VBO32	*	*	*	*	*	*	*	*	*	*	*
VNO32 x VBO64	*	*	*	*	*	*	*	*	*	*	*
VNO64 x VBO16	*	*	*	*	*	*	*	*	*	*	*
VNO64 x VBO32	*	*	*	*	*	*	*	*	*	*	*
VNO64 x VBO64	*	*	*	*	*	*	*	*	*	*	*
VBO16 x VBO32	*	*	*	*	*	*	*	*	*	*	*
VBO16 x VBO64	*	*	*	*	*	*	*	*	*	*	*
VBO32 x VBO64	*	*	*	*	*	*	*	*	*	*	*

que a média obtida com o uso de VBO16. Portanto a média obtida com o uso de VBO64 é classificada com a letra “a”, e a média obtida com o uso de “VBO16” é classificada com a letra “b”. As Tabelas 7.14 e 7.15, respectivamente, mostram as comparações simultâneas referentes aos experimentos com íris humana e signos australianos.

Tabela 7.11: Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - íris humana

	Ciclos										
Comparação	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBC71 x VNO128	*	0.81	0.26	*	*	*	*	*	*	*	*
VBC71 x VNO256	*	*	0.24	0.23	*	0.41	0.15	*	*	*	*
VBC71 x VNO512	0.75	0.85	0.78	0.34	0.74	0.97	0.86	0.95	0.84	0.76	0.55
VBC71 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VBC71 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VBC71 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VNO256	*	*	0.02	*	*	*	*	*	*	*	*
VNO128 x VNO512	*	0.66	0.15	*	*	*	*	*	*	*	*
VNO128 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VNO512	*	*	0.37	0.03	*	0.43	0.11	*	*	*	*
VNO256 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VBO128 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VBO128 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VBO256 x VBO512	*	*	*	*	*	*	*	*	*	*	*

Tabela 7.12: Teste estatístico de Mann-Whitney para comparação das médias do Tipo 3 - signos australianos

	Ciclos										
Comparação	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBC95 x VNO128	*	*	*	*	*	*	*	*	*	*	*
VBC95 x VNO256	*	*	0.35	*	0.02	0.49	0.08	*	*	*	*
VBC95 x VNO512	*	*	*	*	*	*	*	*	*	*	*
VBC95 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VBC95 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VBC95 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VNO256	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VNO512	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO128 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VNO512	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO256 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO128	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VNO512 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VBO128 x VBO256	*	*	*	*	*	*	*	*	*	*	*
VBO128 x VBO512	*	*	*	*	*	*	*	*	*	*	*
VBO256 x VBO512	*	*	*	*	*	*	*	*	*	*	*

Tabela 7.13: Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - dígitos manuscritos

	Ciclos										
Vetor-alvo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBO64	a	a	a	a	a	a	a	a	a	a	a
VBO32	b	b	b	b	b	b	b	b	b	b	b
VBO16	c	c	c	c	c	c	c	c	c	c	c
VBC10	de	d	d	d	d	d	d	d	d	d	d
VNO16	ef	de	d	d	d	e	e	e	d	d	d
VNO32	f	e	d	d	d	de	e	e	d	d	d
VNO64	d	ef	d	d	d	de	e	def	de	de	d

Tabela 7.14: Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - íris humana

	Ciclos										
Vetor-alvo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBO512	a	a	a	a	a	a	a	a	a	a	a
VBO256	b	b	b	b	b	b	b	b	b	b	b
VBO128	c	c	c	c	c	c	c	c	c	c	c
VBC71	f	e	d	d	d	d	d	ef	e	e	e
VNO128	e	e	d	e	f	e	e	g	f	f	f
VNO256	d	d	d	d	e	d	d	d	d	d	d
VNO512	f	e	d	d	d	d	d	f	e	e	e

Tabela 7.15: Comparação simultânea das médias do Tipo 3 por meio do teste estatístico - signos australianos

	Ciclos										
Vetor-alvo	**1**	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**
VBO512	a	a	a	a	a	a	a	a	a	a	a
VBO256	b	b	b	b	b	b	b	b	b	b	b
VBO128	c	c	c	c	c	c	c	c	c	c	c
VBC95	g	g	e	e	e	e	e	f	f	f	f
VNO128	f	f	f	g	f	f	f	g	g	g	g
VNO256	e	e	e	f	e	e	e	e	e	e	e
VNO512	d	d	d	d	d	d	d	d	d	d	d

7.4 Discussão

A discussão matemática e os resultados experimentais obtidos pela comparação das médias do Tipo 3 mostram correspondência entre a teoria e a prática. A maior distância entre os pontos-alvo do espaço de saída reduz as chances de um padrão ser classificado incorretamente

pela rede MLP. Por meio dos gráficos da Figuras 7.2, 7.3 e 7.4 e das Tabelas 7.13, 7.14 e 7.15, observa-se que o distanciamento entre os alvos faz com que as saídas obtidas pela rede fiquem mais distantes de pontos-alvo incorretos. Em outras palavras, o mapeamento de redes do tipo MLP treinadas com VBOs permite que a classificação seja melhor desde os primeiros ciclos de treinamento.

Os gráficos das Figuras 7.2, 7.3 e 7.4 mostram ainda outros detalhes importantes. As médias do Tipo 3 referentes aos experimentos com dígitos manuscritos para o uso de VBC10, VNO16, VNO32 e VNO64, que são todos vetores não ortogonais, possuem diferenças praticamente imperceptíveis no gráfico. O mesmo fenômeno pôde ser observado em relação aos vetores não ortogonais em experimentos com íris humana e signos australianos. Essas diferenças somente são detectadas por meio dos resultados de comparação do teste estatístico mostrados nas Tabelas 7.10, 7.11, 7.12, 7.13, 7.14 e 7.15. Portanto o aumento da dimensão de vetores-alvo do tipo VBCs, que resulta em vetores do tipo VNOs, não aumenta a distância entre as saídas encontradas pela rede e os alvos incorretos. De acordo com os resultados do teste estatístico mostrados nas Tabelas 7.13, 7.14 e 7.15, para alguns ciclos, houve até mesmo redução dessa distância. Já no caso dos VBOs, o aumento da dimensão implica aumento da distância entre as saídas obtidas e os alvos incorretos.

Esse comportamento foi previsto pela discussão matemática no sentido de que a função distância euclidiana entre a saída encontrada e a saída desejada foi expressa em função do erro quadrático médio. Havendo redução do erro, há também redução da distância euclidiana.

Embora seja possível encontrar diferenças significativas para a distância entre saída encontrada e alvos incorretos, pode-se observar que, mesmo após os primeiros ciclos de treinamento, essas diferenças são muito evidentes. Isso interfere de forma significativa na duração do treinamento. Redes treinadas com vetores-alvo do tipo VBO podem obter o desempenho desejado com um número de ciclos bastante inferior do que o necessário para treinamentos com a utilização de vetores-alvo convencionais. As contribuições são a de treinamentos mais rápidos e com menor esforço computacional.

Capítulo 8

O comportamento do desempenho de redes Multilayer Perceptron com o uso de alvos ortogonais

A partir da hipótese apresentada para este trabalho, neste Capítulo são mostrados resultados provenientes da análise do comportamento das redes do tipo MLP treinadas com VBOs, VBCs e VNOs. Para tanto, simulações foram realizadas com a utilização dos dígitos manuscritos, íris humanas e signos australianos já utilizados no capítulo anterior. Nesses experimentos, o desempenho da rede foi avaliado ao longo de várias etapas do treinamento. Além de analisar o desempenho geral, também foi analisado o desempenho nos primeiros ciclos de treinamento, antes do ponto de *overfitting* e depois do ponto de *overfitting*.

8.1 Vetores-alvo usados nos experimentos

Neste trabalho, foram classificados 10 dígitos manuscritos, portanto são 10 classes para serem classificadas. Então, VBCs de tamanho 10 são suficientes para o problema. Conforme visto no algoritmo de geração de VBOs, sua dimensão é sempre uma potência de 2. Para classificar os 10 dígitos, usam-se VBOs de dimensão 16. Também foram usados VNOs de

tamanho 16 para que a comparação entre os tipos de vetores fosse feita em igualdade de condições. Por isso, foram realizados experimentos com VBC10, VNO16 e VBO16. Os conjuntos de dados de íris humana correspondem a 71 pessoas, e os de signos australianos correspondem a 95 tipos de sinais. De forma análoga ao que foi delineado aos experimentos com dígitos manuscritos, nos experimentos com íris humana foram utilizados VBCs de tamanho 71 e VNOs e VBOs de tamanho 128. Nos experimentos com signos australianos, foram utilizados VBCs de tamanho 95 e VNOs e VBOs de tamanho 128.

8.2 Procedimento experimental

8.2.1 Parâmetros analisados

Os experimentos foram realizados por 50 ciclos, número este definido após sucessivas simulações de testes. Foram utilizados três conjuntos de dados: um para treinamento, um para validação e outro para teste. Cada conjunto contém 450 amostras de dígitos, e cada dígito possui 45 amostras em cada conjunto. Nos experimentos com íris humana, foram utilizadas 142 amostras, sendo 2 de cada um dos 71 indivíduos. Finalmente, foram utilizadas 9 amostras de cada um dos 95 tipos de sinais, totalizando 855 amostras para cada um dos conjuntos (treinamento, validação e teste) em experimentos com signos australianos. Todos os parâmetros de desempenho foram obtidos por meio dos conjuntos de teste. Em cada experimento, foram obtidos nove parâmetros:

- Parâmetro 1 - Desempenho máximo global: Esse é o desempenho máximo obtido em todos os ciclos de treinamento.
- Parâmetro 2 - Desempenho após o primeiro ciclo: Após o primeiro ciclo de treinamento, foi medido o desempenho da rede.
- Parâmetro 3 - Desempenho médio dos cinco primeiros ciclos: O desempenho dos cinco primeiros ciclos foi medido. Foi calculada a média desses desempenhos.
- Parâmetro 4 - Ponto de parada *overfitting*: Esse é o ciclo indicado para conclusão

do treinamento pela técnica do *earlystopping*. Esse parâmetro foi determinado em todas as simulações. No entanto, para análise de outros parâmetros, os treinamentos tiveram continuidade até o limite de 50 ciclos.

- Parâmetro 5 - Desempenho no ponto de parada: Esse é o desempenho da rede no ciclo indicado para parada do treinamento pela técnica do *earlystopping*.
- Parâmetro 6 - Desempenho máximo antes do ponto de parada: Antes do ciclo indicado para parada do treinamento pela técnica do *earlystopping*, foi determinado o desempenho máximo.
- Parâmetro 7 - Desempenho máximo depois do ponto de parada: Depois do ciclo indicado para parada do treinamento pela técnica do *earlystopping*, foi determinado o desempenho máximo.
- Parâmetro 8 - Desempenho médio antes do ponto de parada: Antes do ciclo indicado para parada do treinamento pela técnica do *earlystopping*, foi determinado o desempenho médio.
- Parâmetro 9 - Desempenho médio depois do ponto de parada: Depois do ciclo indicado para parada do treinamento pela técnica do *earlystopping*, foi determinado o desempenho médio.

A Figura 8.1 ilustra o arranjo dos parâmetros que foram escolhidos para avaliar o desempenho dos tipos de vetores-alvo em vários momentos do treinamento. Pelo fato de os VBOs terem a distância euclidiana maior, é esperado que eles ofereçam melhor desempenho nos ciclos iniciais de treinamento. Também é esperado melhor desempenho em MLPs treinadas com VBOs antes ou depois do momento indicado para término do treinamento pela técnica do *earlystopping*. Finalmente, é esperado que haja melhor desempenho em todos os momentos do treinamento.

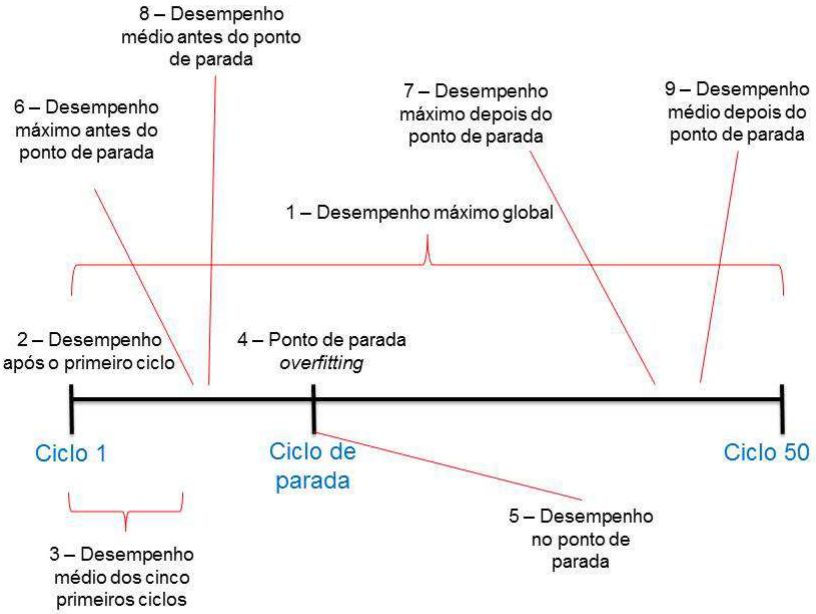


Figura 8.1: Ilustração dos parâmetros analisados

8.2.2 Topologia e taxa de aprendizagem inicial

O programa de simulação foi criado no software Matlab(R) 2013. A rede neural utiliza taxa de aprendizagem adaptativa (Duffner & Garcia, 2007) e o Termo Momentum. Para determinar o tamanho da camada oculta e a taxa de aprendizagem inicial, foi utilizado um algoritmo genético (AG). A taxa de acerto foi atribuída à função de aptidão. O AG foi executado com uma taxa de cruzamento de 90%, taxa de mutação de 10% e por 100 gerações. Utilizaram-se populações de 30 indivíduos, com elitismo de 2 indivíduos e seleção dos indivíduos aptos às operações de cruzamento por meio de torneio. Pelas mesmas razões dos experimentos mostrados no capítulo anterior, foram levados em consideração apenas os vetores VBCs e VNOs para determinação dos parâmetros de treinamento das MLPs. Os parâmetros ideais para VNOs foram replicados para os VBOs.

A Tabela 8.1 mostra os parâmetros de treinamento indicados pelo algoritmo genético.

Tabela 8.1: Parâmetros de treinamento da MLP indicados pelo algoritmo genético

Tipo de dado	Vetor-alvo	Nº de neurônios ocultos	Taxa de aprendizagem inicial
Dígitos	VBC10	95	0.0035
Dígitos	VNO16	122	0.0025
Dígitos	VBO16	122	0.0025
Íris humana	VBC71	187	0.0095
Íris humana	VNO128	231	0.0051
Íris humana	VBO128	231	0.0051
Signos australianos	VBC95	431	0.0013
Signos australianos	VNO128	631	0.0024
Signos australianos	VBO128	631	0.0024

8.2.3 Planejamento estatístico

Considera-se como modelo cada rede MLP treinada com um tipo de vetor-alvo. Para cada modelo foram realizados 500 experimentos, e, em cada um, os pesos sinápticos iniciais foram gerados aleatoriamente com valores entre -0.5 e 0.5. Esses experimentos foram realizados em computadores com configurações e sistema operacional exatamente iguais. Como os resultados não seguem uma distribuição normal, foi utilizado o teste não paramétrico de Mann-Whitney (Conover, 1999). Cada um dos parâmetros citados na subseção 8.2.1 foi analisado pelo teste estatístico.

8.3 Resultados experimentais

A Tabela 8.2 mostra o teste estatístico de Mann-Whitney quando comparadas as médias obtidas com o uso de VBC10, VNO16 e VBO16 nos experimentos com dígitos manuscritos. A hipótese nula é que não existem diferenças significativas entre as médias, e a hipótese alternativa é que existem diferenças significativas entre as médias. O teste foi realizado com o software de análises estatísticas R (R Core Team, 2014).

A Tabela 8.3 mostra as médias de desempenho obtidas com o uso de VBC10, VNO16 e VBO16 para os nove parâmetros analisados. A Tabela 8.4 mostra as comparações simultâneas dos parâmetros obtidos com o uso de vetores-alvo VBC10, VNO16 e VBO16 mostrados pela

Tabela 8.2.

Na Tabela 8.4, se o “valor-p” obtido pelo teste de Mann-Whitney é menor do que 0.05, as médias recebem letras diferentes. Se o “valor-p” obtido é maior do que 0.05, então as médias recebem a mesma letra. Letras iguais indicam que não existem diferenças significativas ao nível de 5% pelo teste de Mann-Whitney. Letras diferentes indicam que existem diferenças significativas ao nível de 5% pelo teste de Mann-Whitney.

Tabela 8.2: Teste estatístico de Mann-Whitney para a comparação de médias - dígitos manuscritos

Parâmetros analisados		**VBC10 X VNO16**	**VBC10 X VBO16**	**VNO16 X VBO16**
Desempenho máximo global	W	122796.50	61620.00	58245.50
	p	0.63	<2.2E-16∗	<2.2E-16∗
Desempenho após o primeiro ciclo	W	150391.00	57226.50	34144.50
	p	<2.696E-08∗	<2.2E-16∗	<2.2E-16∗
Desempenho médio dos cinco primeiros ciclos	W	144798.00	60306.00	40260.50
	p	1.45E-05∗	<2.2E-16∗	<2.2E-16∗
Ponto de parada *overfitting*	W	117270.00	107936.50	115407.50
	p	0.09	1.65E-04∗	0.03∗
Desempenho no ponto de parada	W	131358.00	64341.00	55688.00
	p	0.16	<2.2E-16∗	<2.2E-16∗
Desempenho máximo antes do ponto de parada	W	133838.50	64269.00	53982.50
	p	0.05	<2.2E-16∗	<2.2E-16∗
Desempenho máximo depois do ponto de parada	W	122922.00	61722.00	58431.00
	p	0.65	<2.2E-16∗	<2.2E-16∗
Desempenho médio antes do ponto de parada	W	141836.50	59735.00	42535.00
	p	2.27E-04∗	<2.2E-16∗	<2.2E-16∗
Desempenho médio depois do ponto de parada	W	125579.50	64563.00	60798.50
	p	0.90	<2.2E-16∗	<2.2E-16∗

Tabela 8.3: Médias de desempenho obtidas com o uso de VBC10, VNO16 e VBO16 - dígitos manuscritos

Parâmetros analisados	**VBC10**	**VNO16**	**VBO16**
Desempenho máximo global	79.49	79.34	84.39
Desempenho após o primeiro ciclo	56.13	50.79	69.59
Desempenho médio dos cinco primeiros ciclos	63.34	59.79	74.54
Ponto de parada *overfitting*	7.74	8.10	8.39
Desempenho no ponto de parada	70.38	69.03	79.02
Desempenho máximo antes do ponto de parada	71.10	69.62	79.72
Desempenho máximo depois do ponto de parada	79.47	79.32	84.37
Desempenho médio antes do ponto de parada	64.81	61.94	75.62
Desempenho médio depois do ponto de parada	76.91	76.52	82.45

Tabela 8.4: Comparação simultânea das médias - dígitos manuscritos

Parâmetros analisados	VBO16	VBC10	VNO16
Desempenho máximo global	a	b	b
Desempenho após o primeiro ciclo	a	b	c
Desempenho médio dos cinco primeiros ciclos	a	b	c
Ponto de parada *overfitting*	a	b	b
Desempenho no ponto de parada	a	b	b
Desempenho máximo antes do ponto de parada	a	b	b
Desempenho máximo depois do ponto de parada	a	b	b
Desempenho médio antes do ponto de parada	a	b	c
Desempenho médio depois do ponto de parada	a	b	b

De modo análogo, as Tabelas 8.5, 8.6 e 8.7 mostram os resultados dos experimentos com íris humana quando são comparados os vetores-alvo VBC71, VNO128 e VBO128. Finalmente, a comparação do teste de Mann-Whitney entre os vetores-alvo VBC95, VNO128 e VBO128 para signos australianos é mostrada pelas Tabelas 8.8, 8.9 e 8.10.

Tabela 8.5: Teste estatístico de Mann-Whitney para a comparação de médias - íris humana

Parâmetros analisados		VBC71 X VNO128	VBC71 X VBO128	VNO128 X VBO128
Desempenho máximo global	W	165091.00	343.50	25.50
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho após o primeiro ciclo	W	190606.00	0.00	0.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho médio dos cinco primeiros ciclos	W	244580.00	0.00	0.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Ponto de parada	W	123001.00	345.50	915.00
	p	0.03∗	<2.2E-16∗	<2.2E-16∗
Desempenho no ponto de parada	W	178786.50	1030.50	80.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho máximo antes do ponto de parada	W	176588.00	110.50	0.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho máximo depois do ponto de parada	W	164965.50	2539.50	555.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho médio antes do ponto de parada	W	248260.50	0.00	0.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗
Desempenho médio depois do ponto de parada	W	171250.00	1751.50	256.00
	p	<2.2E-16∗	<2.2E-16∗	<2.2E-16∗

As Figuras 8.2, 8.3, 8.4, 8.5, 8.6, 8.7, 8.8, 8.9 e 8.10 mostram, respectivamente, o gráfico de box-plot do desempenho máximo global, do desempenho após o primeiro ciclo, do desempenho médio dos cinco primeiros ciclos, do ciclo de parada pelo critério de *earlystopping*, do desempenho no ciclo de parada, do desempenho máximo antes do ponto de parada, do desempenho máximo depois do ponto de parada, do desempenho médio antes do ponto de parada e do desempenho médio depois do ponto de parada.

Tabela 8.6: Médias de desempenho obtidas com o uso de VBC71, VNO128 e VBO128 - íris humana

Parâmetros analisados	VBC71	VNO128	VBO128
Desempenho máximo global	83.88	82.72	90.44
Desempenho após o primeiro ciclo	7.01	5.16	77.13
Desempenho médio dos cinco primeiros ciclos	18.51	12.22	85.48
Ponto de parada	39.99	34.82	45.12
Desempenho no ponto de parada	83.00	81.33	89.20
Desempenho máximo antes do ponto de parada	82.95	81.36	90.44
Desempenho máximo depois do ponto de parada	83.87	82.70	89.19
Desempenho médio antes do ponto de parada	66.49	59.06	88.88
Desempenho médio depois do ponto de parada	83.48	82.13	89.18

Tabela 8.7: Comparação simultânea das médias - íris humana

Parâmetros analisados	VBO128	VBC71	VNO128
Desempenho máximo global	a	b	c
Desempenho após o primeiro ciclo	a	b	c
Desempenho médio dos cinco primeiros ciclos	a	b	c
Ponto de parada	a	c	b
Desempenho no ponto de parada	a	b	c
Desempenho máximo antes do ponto de parada	a	b	c
Desempenho máximo depois do ponto de parada	a	b	c
Desempenho médio antes do ponto de parada	a	b	c
Desempenho médio depois do ponto de parada	a	b	c

Tabela 8.8: Teste estatístico de Mann-Whitney para a comparação de médias - signos australianos

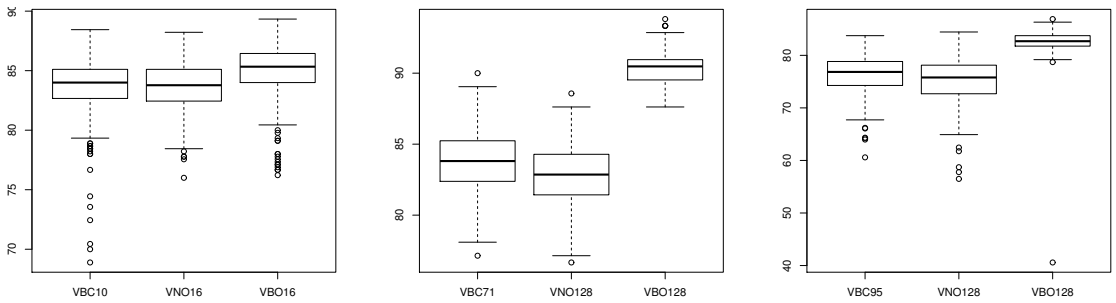
Parâmetros analisados		VBC95 X VNO128	VBC95 X VBO128	VNO128 X VBO128
Desempenho máximo global	W	145273.50	8022.00	6314.00
	p	9.011E-16⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho após o primeiro ciclo	W	134175.50	368.00	359.00
	p	0.04439⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho médio dos cinco primeiros ciclos	W	130119.50	171.50	147.50
	p	0.26⋆	<2.2E-16⋆	<2.2E-16⋆
Ponto de parada	W	127719.00	223353.50	217376.00
	p	0.55	<2.2E-16⋆	<2.2E-16⋆
Desempenho no ponto de parada	W	150599.50	801.00	617.50
	p	2.074E-08⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho máximo antes do ponto de parada	W	150815.50	712.50	681.50
	p	1.577E-08⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho máximo depois do ponto de parada	W	145277.00	8022.00	6314.00
	p	8.979E-06⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho médio antes do ponto de parada	W	156504.00	500.00	500.00
	p	5.251E-12⋆	<2.2E-16⋆	<2.2E-16⋆
Desempenho médio depois do ponto de parada	W	153597.00	597.00	556.00
	p	3.8E-10⋆	<2.2E-16⋆	<2.2E-16⋆

Tabela 8.9: Médias de desempenho obtidas com o uso de VBC95, VNO128 e VBO128 - signos australianos

Parâmetros analisados	**VBC95**	**VNO128**	**VBO128**
Desempenho máximo global	76.39	75.26	82.63
Desempenho após o primeiro ciclo	2.82	2.68	28.56
Desempenho médio dos cinco primeiros ciclos	2.88	2.79	47.06
Ponto de parada	26.66	26.58	20.18
Desempenho no ponto de parada	41.11	36.92	72.68
Desempenho máximo antes do ponto de parada	41.22	37.06	73.62
Desempenho máximo depois do ponto de parada	76.39	75.26	82.63
Desempenho médio antes do ponto de parada	15.95	14.04	62.36
Desempenho médio depois do ponto de parada	62.77	60.06	78.84

Tabela 8.10: Comparação simultânea das médias - signos australianos

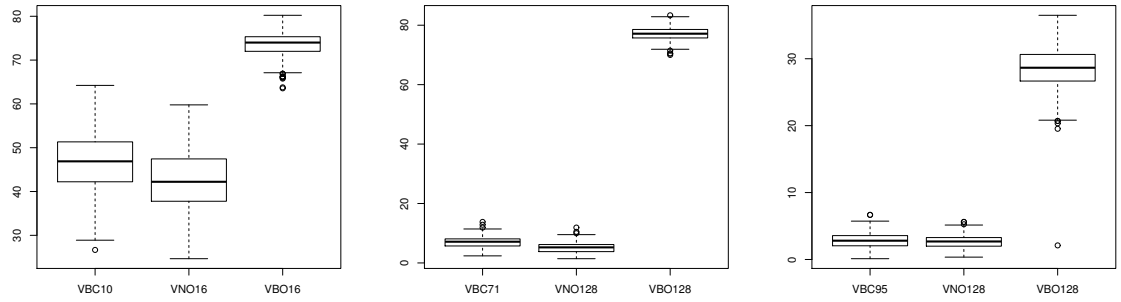
Parâmetros analisados	**VBO128**	**VBC95**	**VNO128**
Desempenho máximo global	a	b	c
Desempenho após o primeiro ciclo	a	b	c
Desempenho médio dos cinco primeiros ciclos	a	b	c
Ponto de parada	a	b	b
Desempenho no ponto de parada	a	b	c
Desempenho máximo antes do ponto de parada	a	b	c
Desempenho máximo depois do ponto de parada	a	b	c
Desempenho médio antes do ponto de parada	a	b	c
Desempenho médio depois do ponto de parada	a	b	c



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

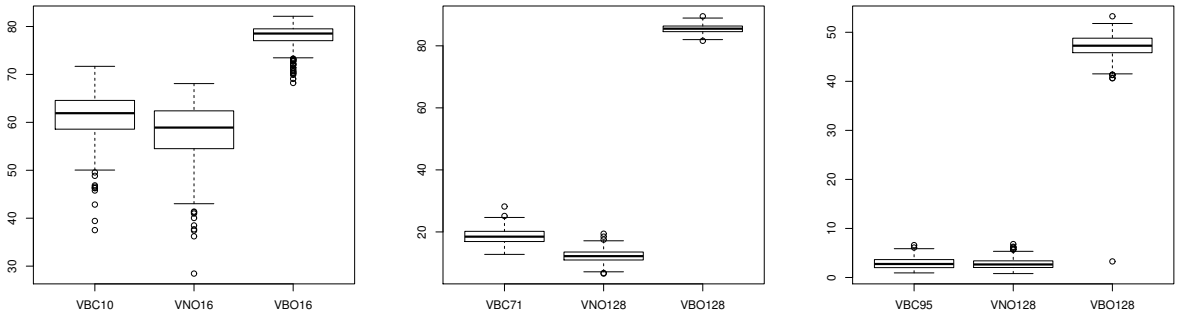
Figura 8.2: Desempenho máximo global

A Figura 8.11 mostra o desempenho médio obtido dos 500 experimentos em cada um dos cinquenta ciclos com o uso dos três tipos de vetores-alvo adotados. As linhas verticais mostram o ciclo médio indicado para interromper o treinamento pelo critério de *earlystopping* com o uso de cada um dos vetores-alvo.



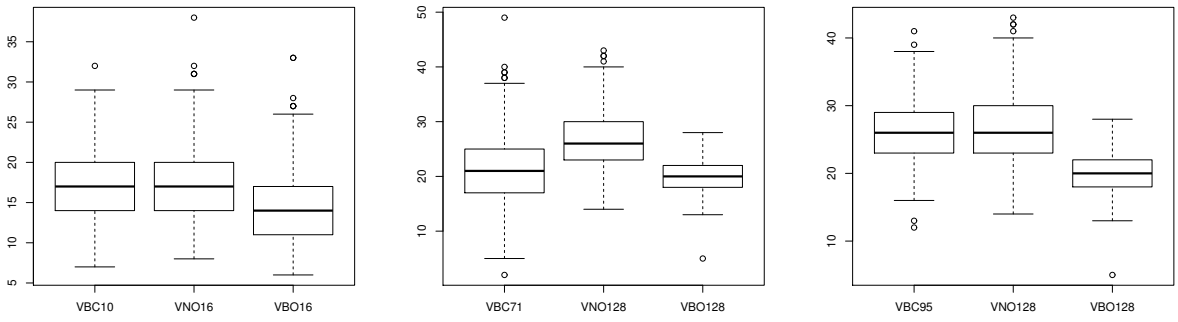
(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 8.3: Desempenho obtido após o primeiro ciclo



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

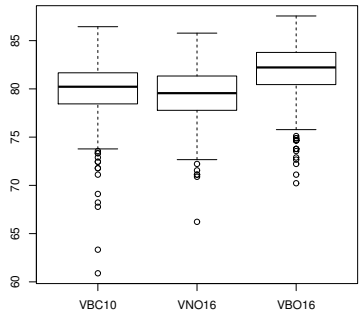
Figura 8.4: Desempenho médio obtido nos cinco primeiros ciclos



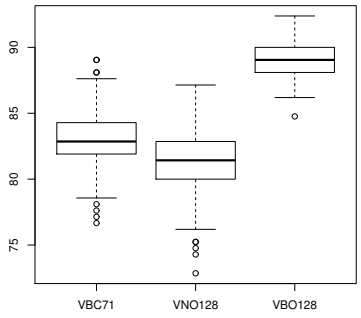
(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 8.5: Ponto de parada *overfitting*

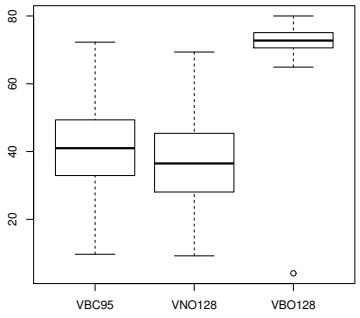
Os resultados das Tabelas 8.2, 8.5, 8.8, 8.4, 8.7 e 8.10 mostram que o uso de VBO16 para dígitos manuscritos e VBO128 para íris humana e signos australianos propiciou melhor desempenho máximo global.



(a) Dígitos manuscritos

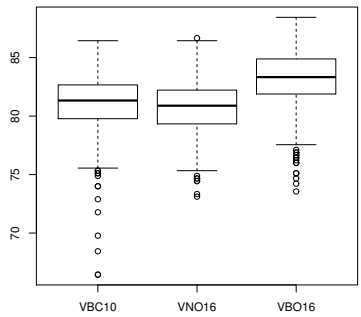


(b) Íris humana

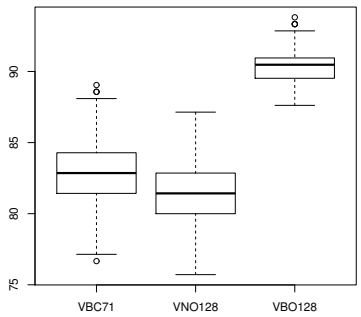


(c) Signos australianos

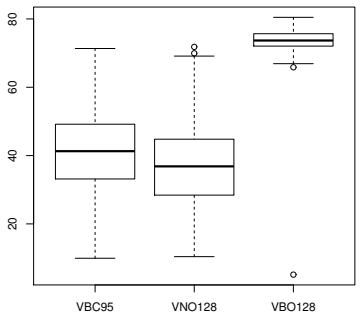
Figura 8.6: Desempenho obtido no ponto de parada



(a) Dígitos manuscritos

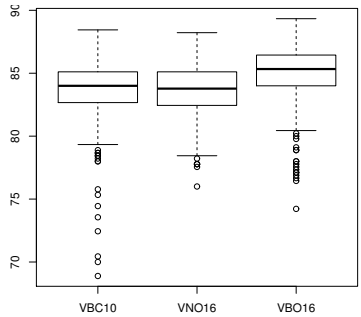


(b) Íris humana

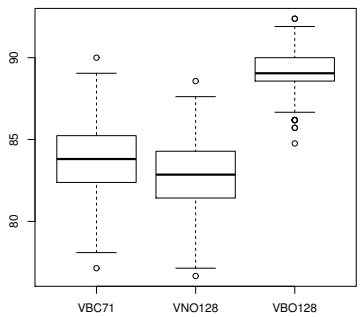


(c) Signos australianos

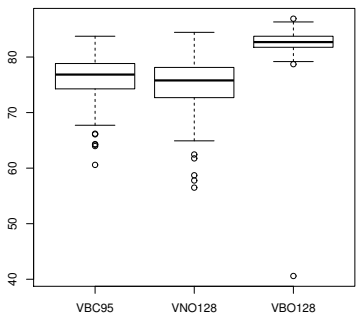
Figura 8.7: Desempenho máximo obtido antes do ponto de parada



(a) Dígitos manuscritos



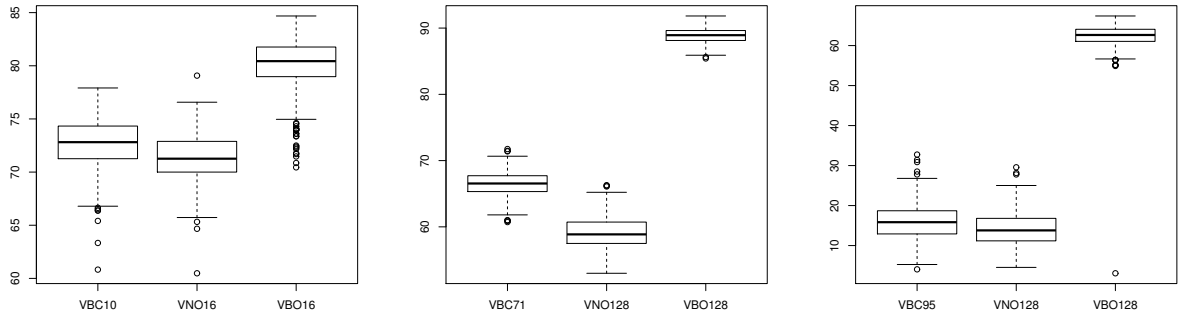
(b) Íris humana



(c) Signos australianos

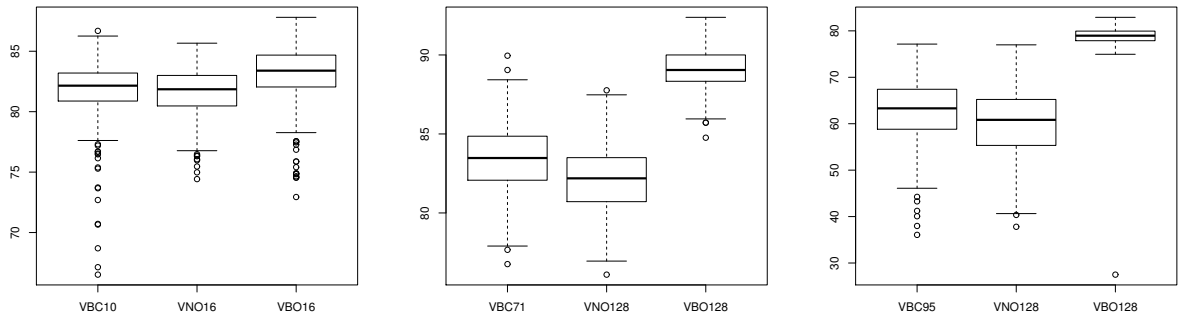
Figura 8.8: Desempenho máximo obtido depois do ponto de parada

O desempenho depois do primeiro ciclo também foi melhor com o uso dos VBOs para os três tipos de dados. Isso pode ser visto nas Tabelas 8.4, 8.7 e 8.10 e nos gráficos das Figuras 8.11, 8.12 e 8.13.



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 8.9: Desempenho médio obtido antes do ponto de parada



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 8.10: Desempenho médio obtido depois do ponto de parada

As Tabelas 8.4, 8.7 e 8.10 mostram que o desempenho médio dos cinco primeiros ciclos é melhor com o uso de VBOs. A Figura 8.4 confirma esse resultado para os três tipos de conjuntos de dados.

O ciclo indicado para término do treinamento pelo critério do *earlystopping* é ligeiramente menor com o uso de VBOs de acordo com a Figura 8.5.

Os resultados mostrados nas Tabelas 8.2, 8.5, 8.8, 8.4, 8.7 e 8.10 e nos gráficos da Figura 8.6 indicam que o uso de VBOs propiciam melhor desempenho no ciclo indicado para parada pelo critério do *earlystopping*.

O desempenho obtido com o uso de VBOs também é melhor com os parâmetros: Desempenho máximo antes do ponto de parada, Desempenho máximo depois do ponto de parada, Desempenho médio antes do ponto de parada e Desempenho médio depois do ponto de parada.

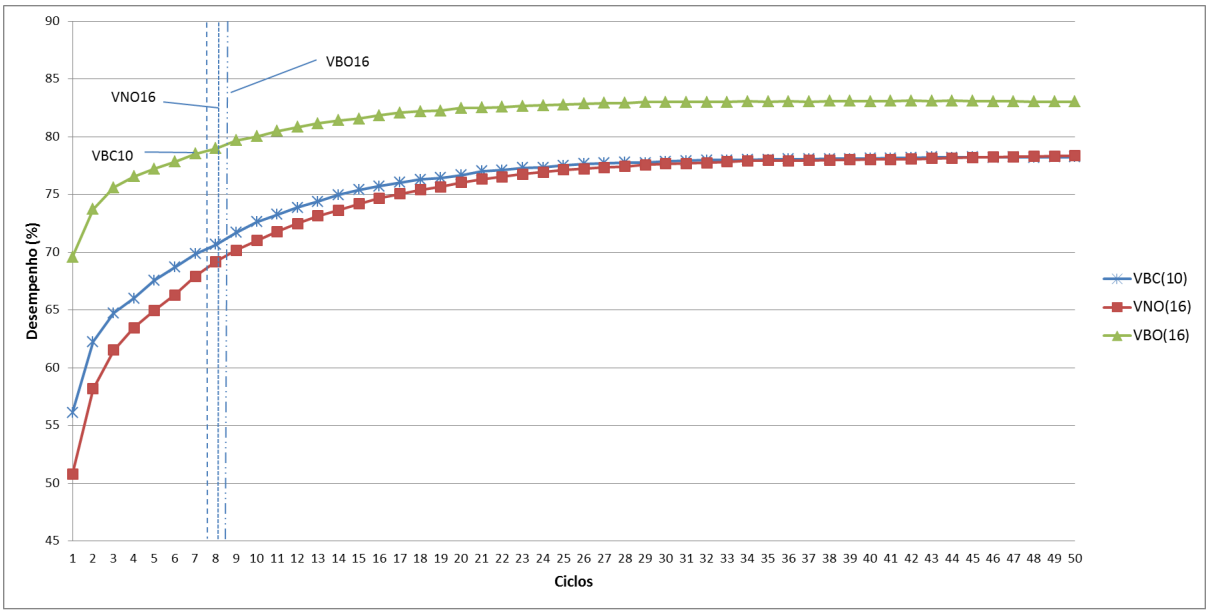


Figura 8.11: Média de desempenho em todos os ciclos - dígitos manuscritos

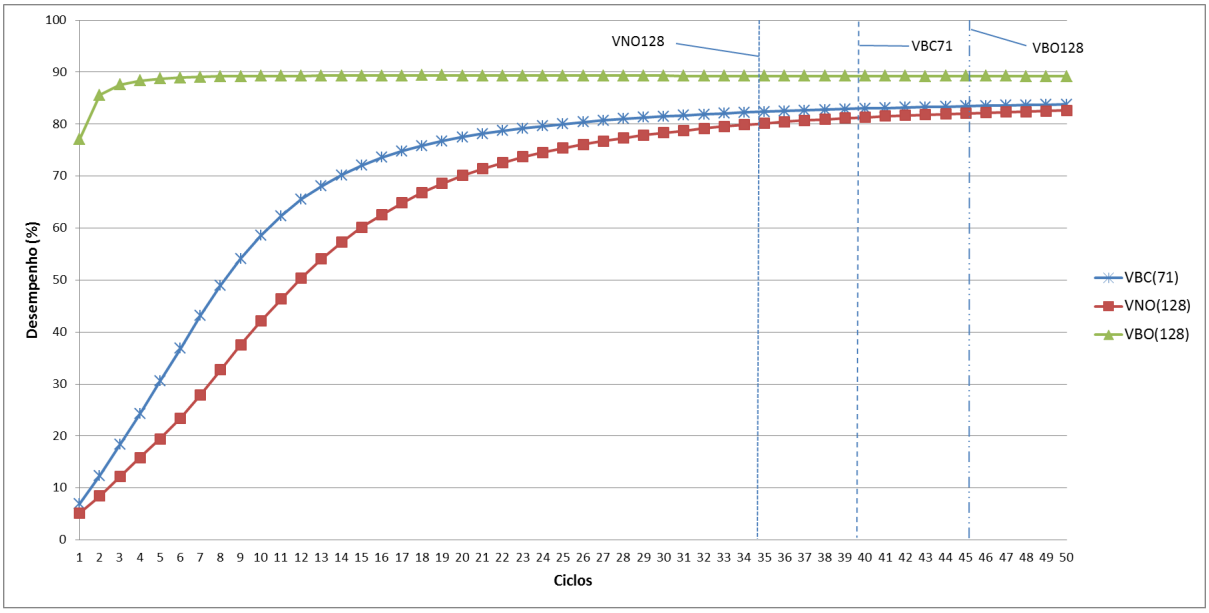


Figura 8.12: Média de desempenho em todos os ciclos - íris humana

Isso é mostrado nos gráficos das Figuras 8.11, 8.12 e 8.13 e nas Tabelas 8.4, 8.7 e 8.10 na aplicação dos três conjuntos de dados.

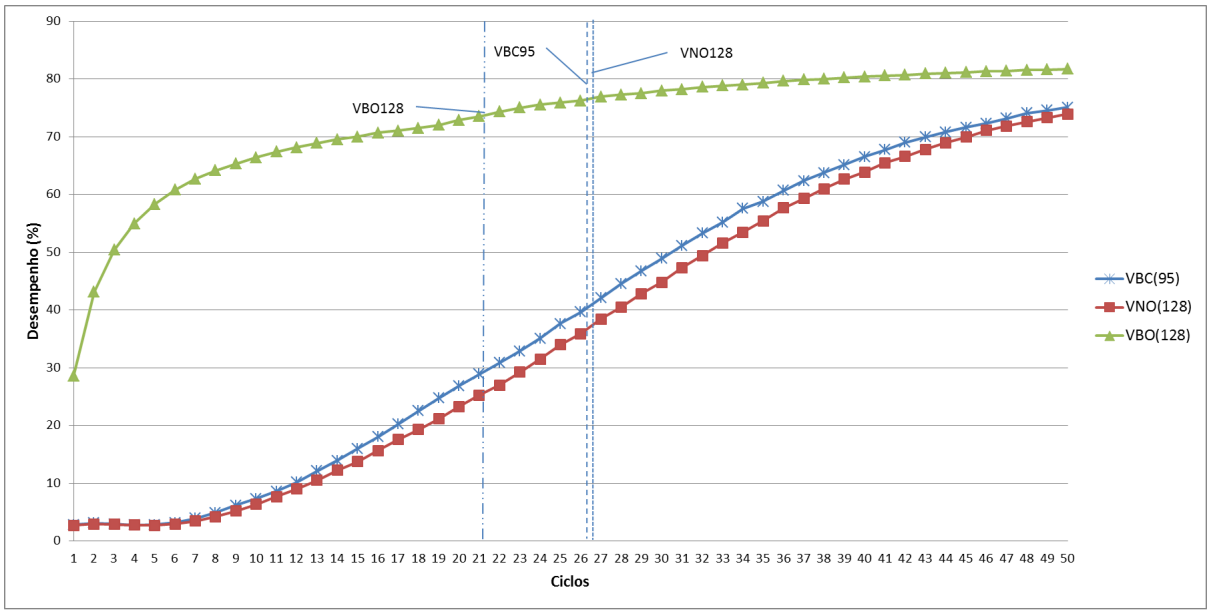


Figura 8.13: Média de desempenho em todos os ciclos - signos australianos

8.4 Discussão

O uso de VBOs como alvos de RNAs do tipo MLP propiciou taxas de acerto na ordem de 5% maiores do que as obtidas com o uso de vetores bipolares tradicionais no que se refere ao desempenho máximo global. Os resultados também mostram que o aumento de tamanho dos vetores bipolares convencionais não melhora o desempenho.

Também foi observado que o uso de VBOs propicia melhor desempenho com pouco treinamento. Após o primeiro ciclo de treinamento, a taxa de acerto obtida com o uso de VBOs é no mínimo 13.46% maior do que a obtida com o uso de VBCs para os três conjuntos de dados. Tal fenômeno se repete para os parâmetros: Média do desempenho dos cinco primeiros ciclos, Desempenho máximo antes do ponto de parada e Desempenho médio antes do ponto de parada. Isso é especialmente interessante porque permite o uso de RNAs do tipo MLP com pouco esforço computacional. Assim, pode-se resolver o inconveniente dos longos treinamentos desse tipo de rede.

Também pôde ser observado que, após o ponto de parada, o desempenho obtido usando-se VBOs foi melhor do que o desempenho obtido com VBCs e VNOs. Isso significa que, mesmo em condições de excesso de treinamento, o desempenho com o uso de VBOs

é melhor do que com a utilização de VBCs e VNOs.

Tudo isso indica que o uso de VBOs como vetores-alvo de MLPs oferece uma maior capacidade de generalização da rede. O resultado é um melhor desempenho na classificação de padrões.

Capítulo 9

A robustez de redes do tipo MLP com a utilização de VBOs

9.1 Robustez, a topologia e a taxa de aprendizagem inicial

Conforme visto na seção 3.3, a taxa de aprendizagem e a topologia exercem grande influência no desempenho de redes MLP em tarefas de RPs. Dessa maneira, esses parâmetros devem ser cuidadosamente escolhidos para que se possam obter desempenhos satisfatórios. Foram indicadas na mesma seção, sugestões de metodologias para a escolha desses parâmetros. Contudo sua aplicação pode não ser tão prática, principalmente para pessoas com pouca familiaridade com redes do tipo MLP. Por outro lado, essas metodologias de busca dos parâmetros ideais não garantem o alcance aos melhores parâmetros.

Durante os experimentos do trabalho descritos nos capítulos anteriores, foi observado um fenômeno que deu origem ao estudo desse capítulo. As redes MLP treinadas com VBOs mostraram-se menos suscetíveis às alterações nos parâmetros da taxa de aprendizagem e ao número de neurônios na camada intermediária. Diante dessas observações, achou-se por bem investigar o comportamento do desempenho das MLPs com a combinação de vários valores para taxa de aprendizagem e número de neurônios da camada intermediária.

9.2 Procedimento experimental

9.2.1 Planejamento experimental e estatístico

Os experimentos foram realizados com os mesmos conjuntos de dados utilizados nos experimentos descritos nos capítulos anteriores: dígitos manuscritos, íris humana e signos australianos. Para cada tipo de dado, foram realizados experimentos com 7 valores diferentes de taxa de apredizagem inicial e 12 valores diferentes de número de neurônios na camada oculta. Todas as combinações entre taxa de aprendizagem inicial e número de neurônios da camada intermediária foram submetidas ao uso dos três tipos de vetores-alvo característico de cada conjunto de dados: VBC10, VNO16 e VBO16 para dígitos manuscritos, VBC71, VNO128 e VBO128 para íris humana e VBC95, VNO128 e VBO128 para signos australianos. As figuras 9.1, 9.2 e 9.3 mostram as combinações entre vetores-alvo, taxas de aprendizagem iniciais e números de neurônios na camada intermediária para dígitos manuscritos, íris humanas e signos australianos.

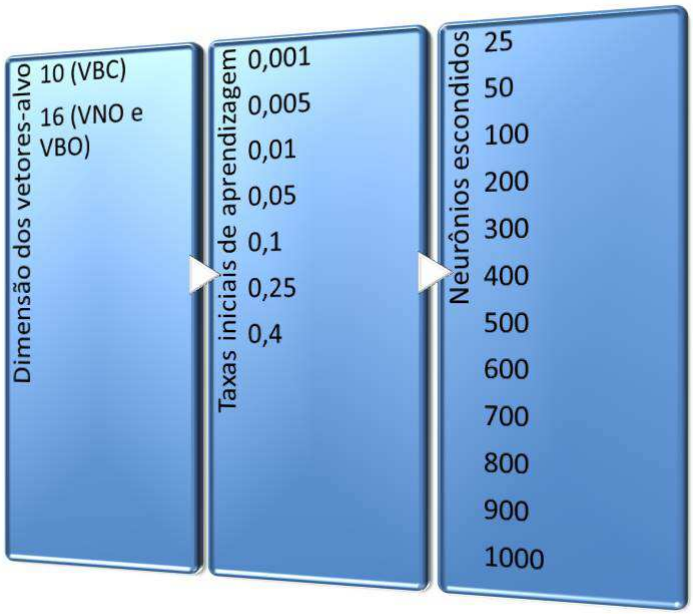


Figura 9.1: Combinações de parâmetros de treinamento para dígitos manuscritos

Os experimentos foram repetidos 100 vezes para cada combinação, com inicialização

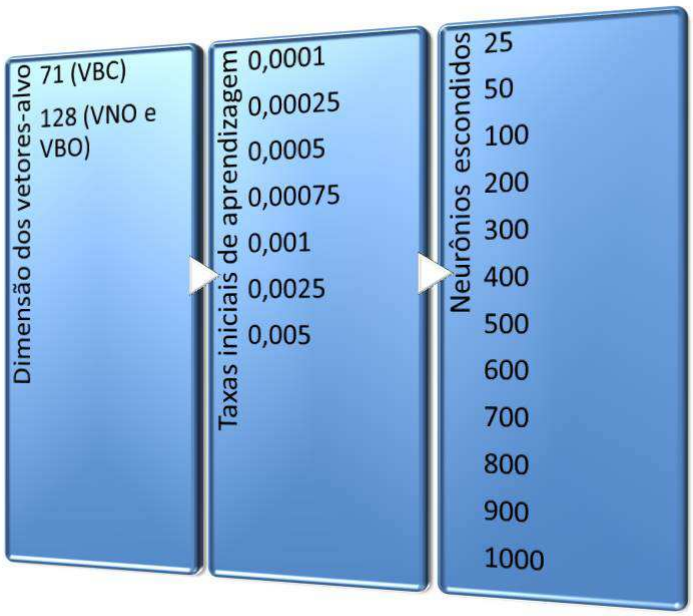


Figura 9.2: Combinações de parâmetros de treinamento para íris humana

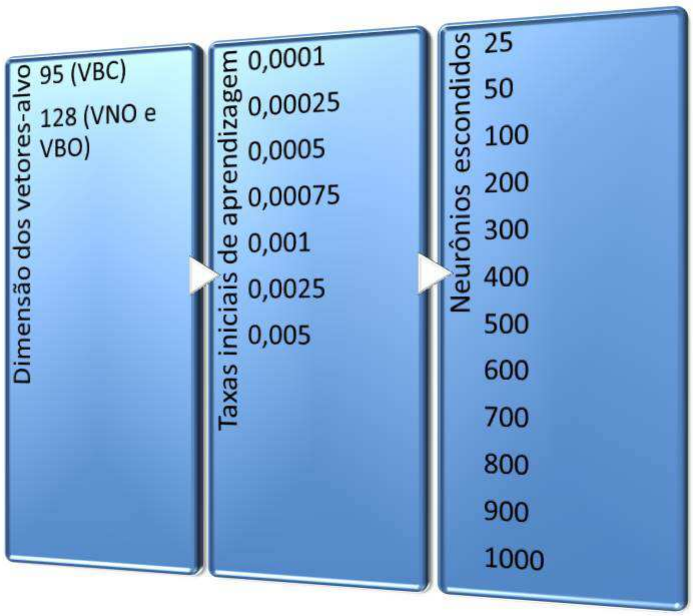


Figura 9.3: Combinações de parâmetros de treinamento para signos australianos

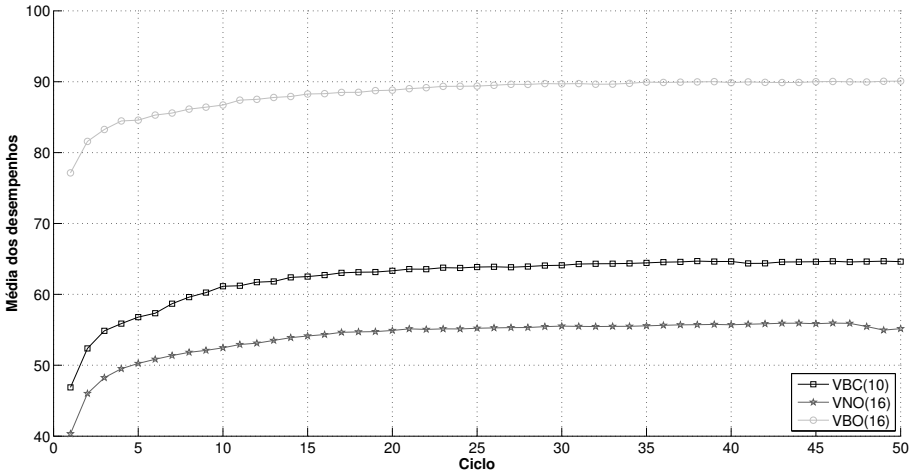
aleatória dos pesos. Os pesos sinápticos iniciais foram gerados aleatoriamente entre -0.5 e 0.5. Os valores de taxas de aprendizagem iniciais adotados para dígitos manuscritos foram: 0.001,

0.005, 0.01, 0.05, 0.1, 0.25 e 0.4. Os valores das taxas de aprendizagem iniciais adotados para íris humana e signos australianos foram: 0.0001, 0.00025, 0.0005, 0.00075, 0.001, 0.0025 e 0.005. Dessa maneira, cada tipo de vetor-alvo foi submetido a 84 combinações de taxa de aprendizagem inicial com número de neurônios na camada intermediária para cada tipo de dado. Considerando que foram realizados 100 experimentos para cada combinação, cada tipo de vetor-alvo foi submetido a 8400 experimentos. Portanto foram realizados 25200 experimentos para os três tipos de vetores-alvo: VBC, VNO e VBO. Finalmente, foram realizados 75600 experimentos contabilizando os três tipos de dados utilizados.

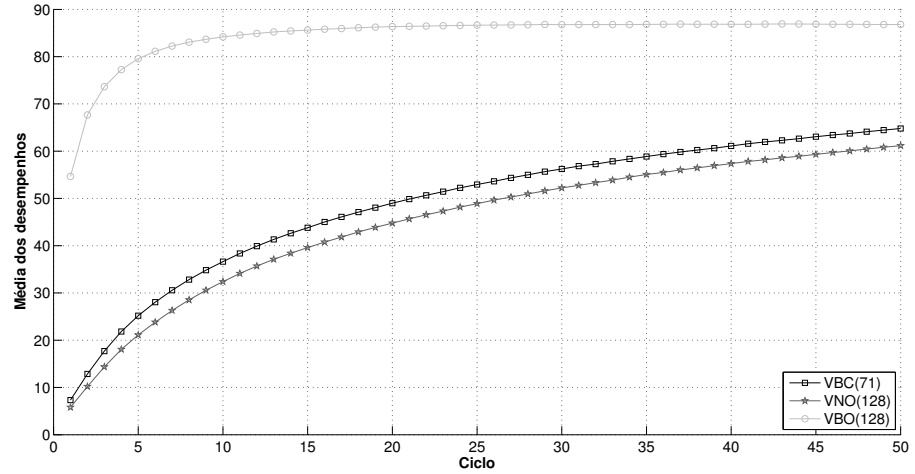
Cada treinamento teve duração de 50 ciclos. Para cada ciclo, um conjunto de teste distinto ao de treinamento foi submetido à rede, e a taxa de acerto (acurácia) foi coletada. Portanto cada experimento traz consigo 50 resultados de desempenho correspondentes a cada ciclo. Foram calculadas a média e o coeficiente de variação para os 25200 experimentos realizados para cada tipo de vetor-alvo. Também foram gerados gráficos do tipo box-plot para os ciclos 1, 10, 20, 30, 40 e 50. Das 84 combinações de parâmetros, também foi verificado em quantas delas houve desempenho superior a 70% nos ciclos 1, 10, 20, 30, 40 e 50.

Para a etapa de treinamento da MLP, 90 amostras de cada dígito foram usadas, atingindo um total de 900 amostras entre os 10 dígitos básicos. Para a etapa de treinamento com íris humana, 2 amostras de cada pessoa foram usadas. Assim, 142 amostras de íris foram usadas. Para sinais australianos, 9 amostras de cada signo foram usadas. Portanto foi usado um total de 855 amostras. Os conjuntos de teste de dígitos manuscritos, íris humana e signos australianos foram compostos respectivamente por 450, 213 e 855 amostras.

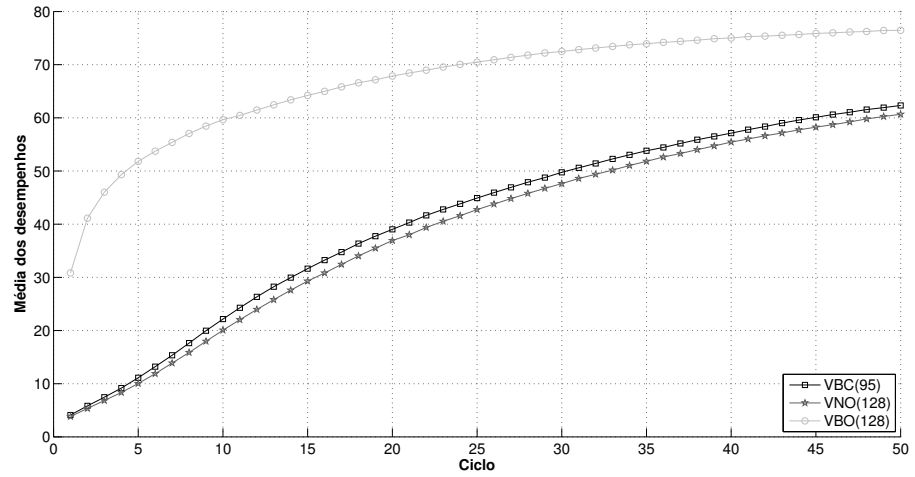
Os experimentos foram realizados com computadores utilizando exatamente as mesmas configurações de sistema operacional. Foi criado um programa de simulação utilizando o software Matlab®2013. O algoritmo utiliza taxa de aprendizagem adaptativa e o termo momentum.



(a) Dígitos manuscritos



(b) Íris humana



(c) Signos australianos

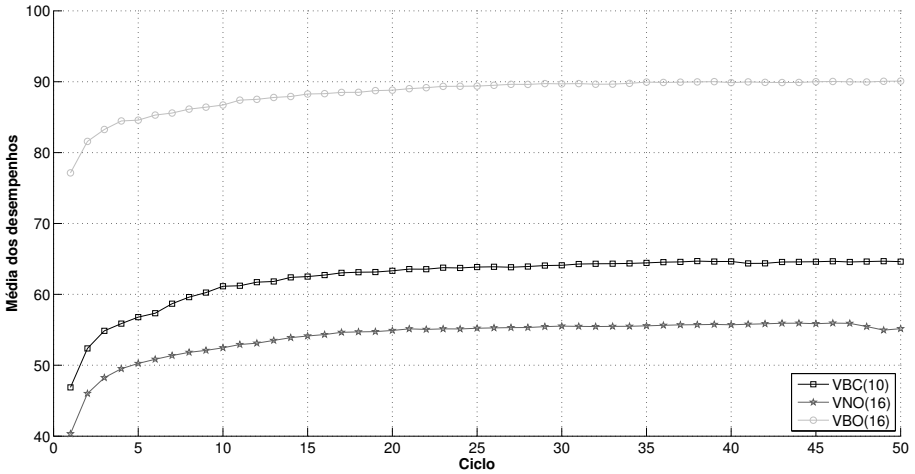
Figura 9.4: Média de desempenho para todos os ciclos e todas as combinações de parâmetros

9.3 Resultados experimentais

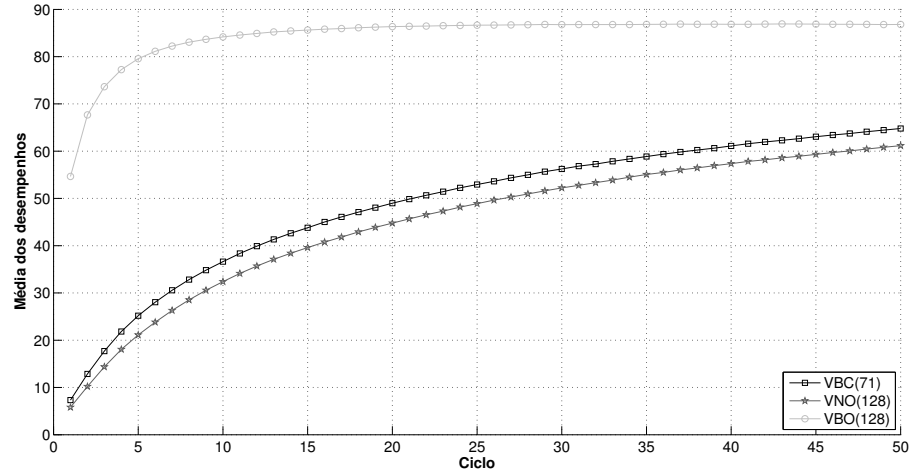
A Figura 9.5 mostra a média dos desempenhos (acurácia) coletados em todas as combinações de parâmetros para cada um dos 50 ciclos de treinamentos. Na parte "a" da referida figura, são mostrados os resultados referentes aos experimentos com dígitos manuscritos. Os resultados referentes aos experimentos com íris humana e signos australianos são mostrados respectivamente nas partes "b" e "c".

A Figura 9.6 mostra o coeficiente de variação dos desempenhos obtidos nos experimentos com dígitos manuscritos (a), íris humana (b) e signos australianos (c).

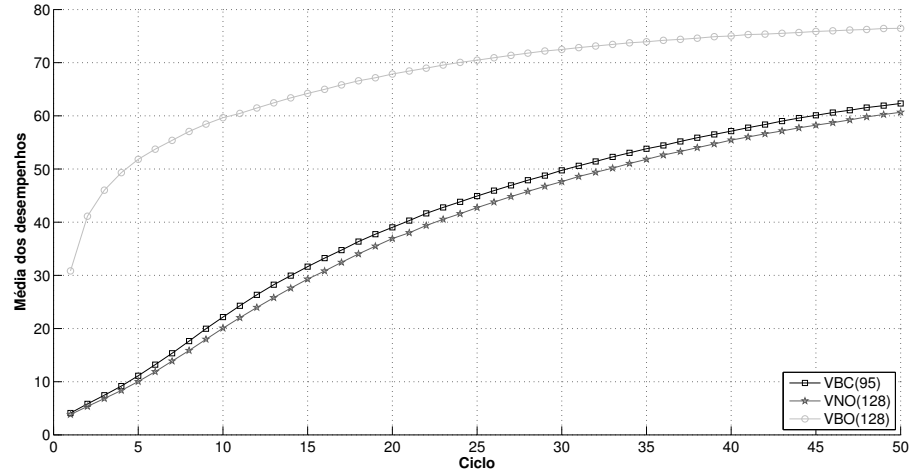
As Figuras 9.7, 9.8, 9.9, 9.10, 9.11 e 9.12 mostram os gráficos de box-plot referentes respectivamente aos ciclos 1, 10, 20, 30, 40 e 50. Os gráficos de box-plot referem-se aos desempenhos obtidos com todas as combinações de parâmetros. A parte "a" de cada box-plot refere-se aos experimentos realizados com dígitos manuscritos. As partes "b" e "c" referem-se aos experimentos realizados com íris humana e signos australianos.



(a) Dígitos manuscritos

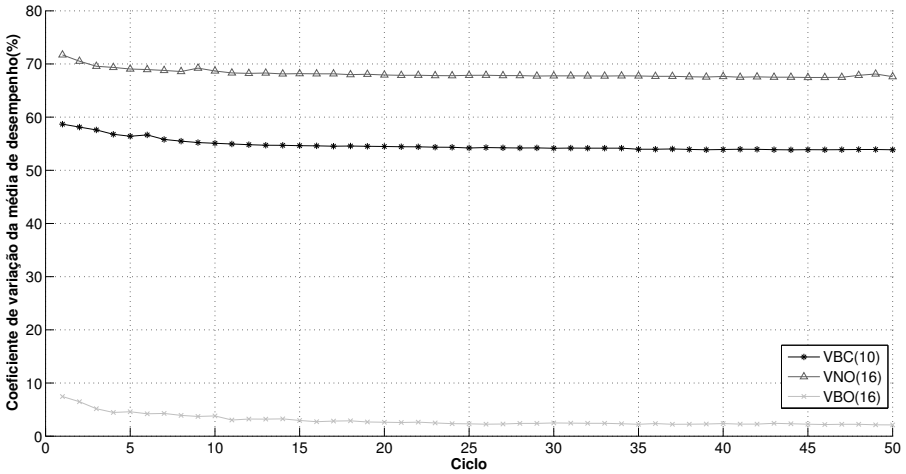


(b) Íris humana

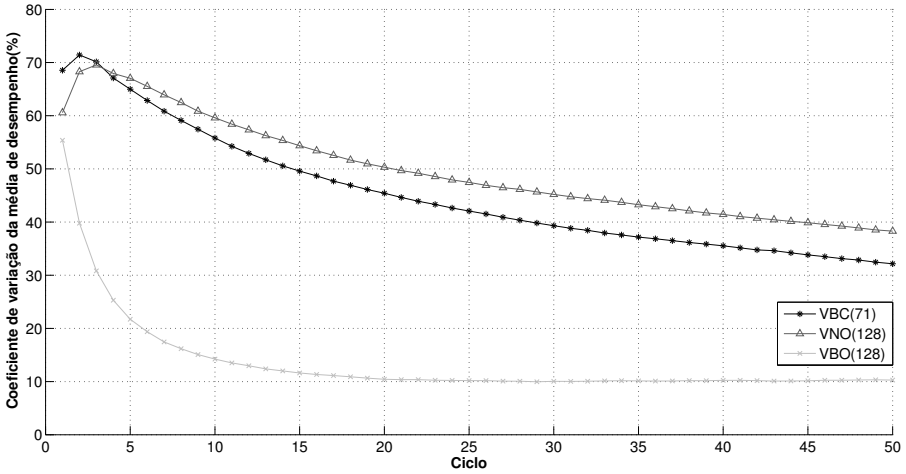


(c) Signos australianos

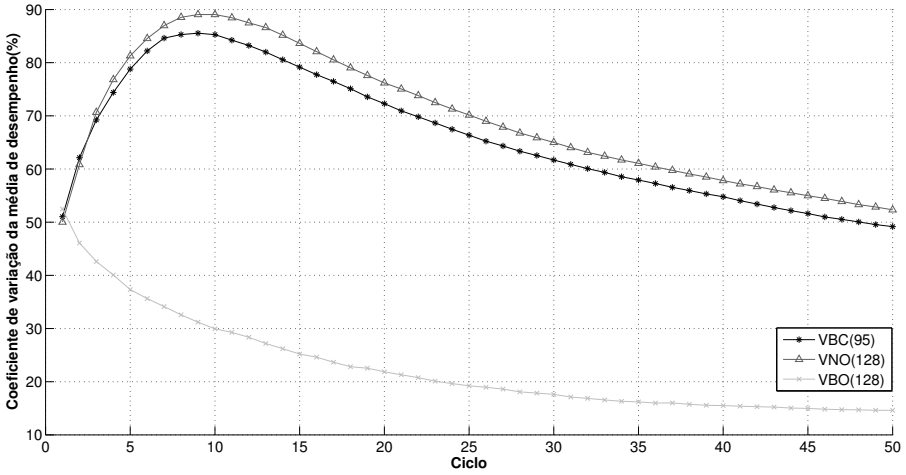
Figura 9.5: Média de desempenho para todos os ciclos e todas as combinações de parâmetros



(a) Dígitos manuscritos

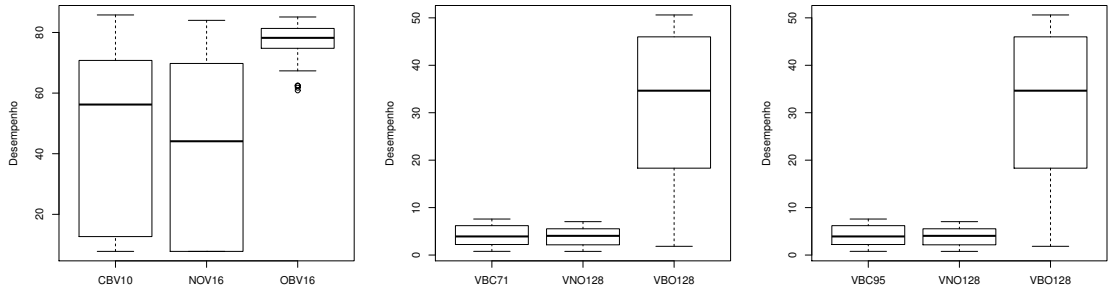


(b) Íris humana



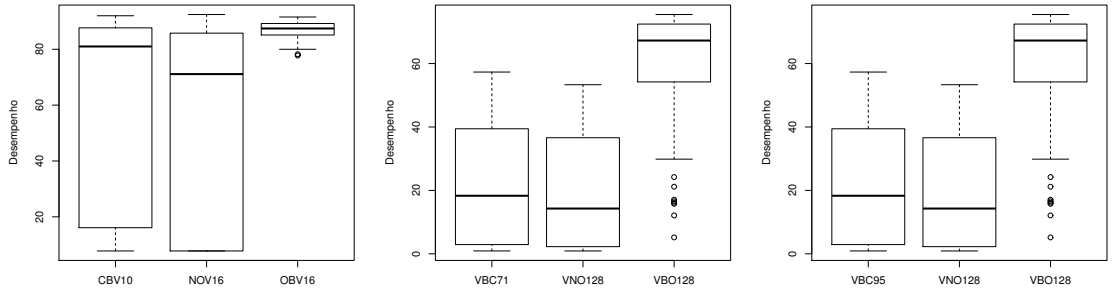
(c) Signos australianos

Figura 9.6: Coeficientes de variação dos desempenhos coletados durante os treinamentos



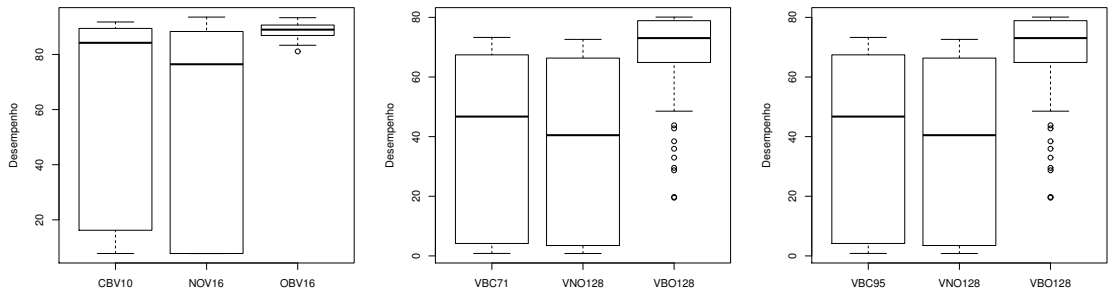
(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.7: Desempenho no ciclo 1



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.8: Desempenho no ciclo 10



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.9: Desempenho no ciclo 20

A Tabela 9.1 mostra a quantidade de experimentos com resultados superiores a 70% para cada tipo de vetor-alvo. Os resultados referem-se a todos os 84 experimentos formados pela combinação dos parâmetros para dígitos manuscritos. Os mesmos resultados referentes aos experimentos com íris humana e signos australianos são mostrados pelas Tabelas 9.2 e 9.3.

Tabela 9.1: Número de experimentos com desempenho superior a 70% - dígitos manuscritos

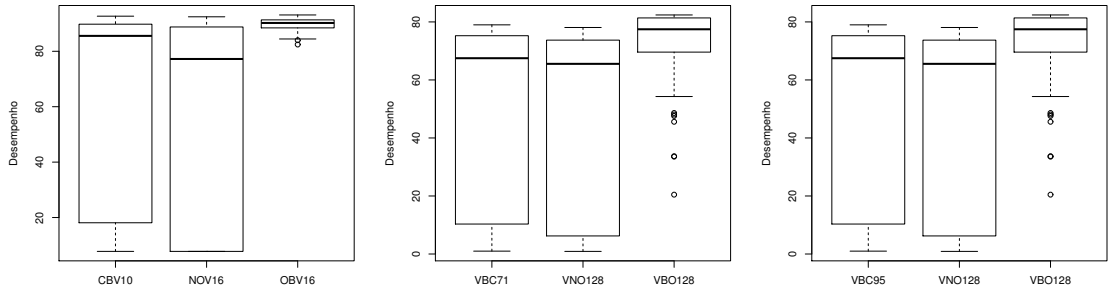
Vetores-alvo		**VBC10**	**VNO16**	**VBO16**
Ciclos	1	22	21	74
	10	52	43	84
	20	55	47	84
	30	55	49	84
	40	57	49	84
	50	57	47	84

Tabela 9.2: Número de experimentos com desempenho superior a 70% - íris humana

Vetores-alvo		**VBC71**	**VNO128**	**VBO128**
Ciclos	1	0	0	40
	10	0	0	74
	20	15	9	76
	30	27	22	77
	40	38	34	77
	50	49	44	76

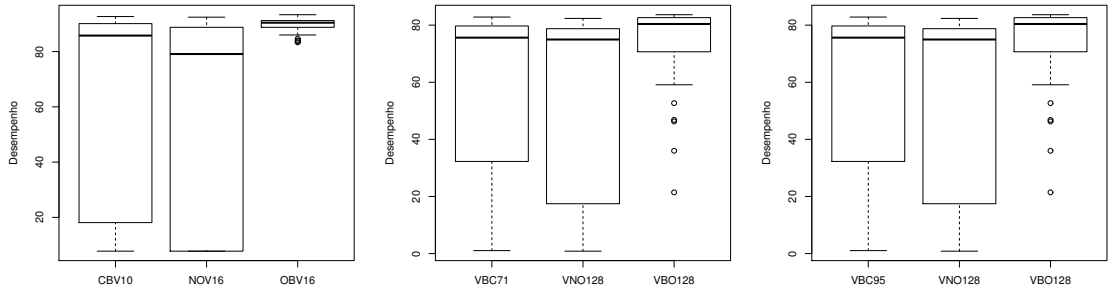
Tabela 9.3: Número de experimentos com desempenho superior a 70% - signos australianos

Vetores-alvo		**VBC95**	**VNO128**	**VBO128**
Ciclos	1	0	0	0
	10	0	0	32
	20	14	11	52
	30	39	37	61
	40	49	48	64
	50	60	59	68



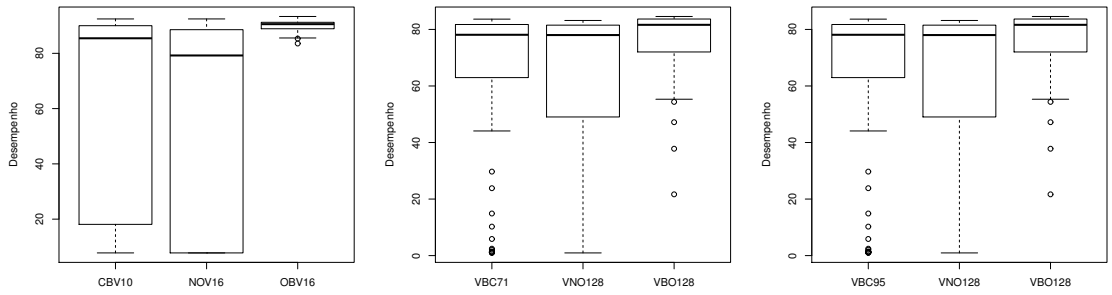
(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.10: Desempenho no ciclo 30



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.11: Desempenho no ciclo 40



(a) Dígitos manuscritos (b) Íris humana (c) Signos australianos

Figura 9.12: Desempenho no ciclo 50

9.4 Discussão

Os gráficos da Figura 9.5 mostram que a média do desempenho é superior com o uso de VBOs para os três tipos de dados. O gráfico da parte "a" na Figura 9.6 mostra que o coeficiente de variação é muito inferior em todos os ciclos quando são usados VBOs. Esse

gráfico refere-se aos dados de dígitos manuscritos. As partes “b” e “c” da Figura 9.6 mostram os coeficientes de variação referentes aos experimentos com íris humana e signos australianos. A trajetória de ambos os gráficos indica proximidade para os coeficientes de variação dos três tipos de vetores-alvo. Entretanto as linhas representativas dos coeficientes de variação dos VBOs reduzem-se drasticamente, enquanto que as linhas referentes aos VBCs e VNOs têm ligeira queda, mantendo-se bastante elevadas.

Os gráficos de box-plot da Figura 9.7 mostram melhor desempenho para VBOs após o primeiro ciclo de treinamento para os três tipos de conjuntos de dados. Entretanto a mesma Figura indica que a variabilidade do desempenho com VBOs é bem menor nos experimentos com dígitos manuscritos e maior nos experimentos com íris humana e signos australianos. A pouca variabilidade de VBCs e VNOs em experimentos com íris humana e signos australianos mostra que o desempenho é extremamente baixo para qualquer combinação de parâmetros. Por outro lado, a alta variabilidade obtida com a utilização de VBCs e VNOs mostra algum ganho de desempenho para algumas combinações de parâmetros. Ainda assim, essa maior variabilidade não garante aos vetores do tipo VBCs e VNOs desempenho próximo àquele obtido com VBOs.

Nos demais gráficos de box-plot, o desempenho se mantém superior nos resultados obtidos com o uso de VBOs. A variabilidade desse desempenho torna-se pequena com a utilização de VBOs e grande com a utilização de VBCs e VNOs. Isso mostra que o desempenho é pouco suscetível à alteração de parâmetros com o uso de VBOs, ao contrário do que ocorre com a utilização dos outros tipos de vetores-alvo.

As Tabelas 9.1, 9.2 e 9.3 mostram, respectivamente, a quantidade de experimentos cujas médias de desempenho foram superiores a 70% para dígitos manuscritos, íris humana e signos australianos. Os resultados confirmam o que já foi observado nos gráficos de desempenho, nos gráficos dos coeficientes de variação e nos box-plots. O desempenho é superior com o uso de VBOs e há regularidade desse fenômeno independentemente das alterações dos parâmetros.

9.5 Conclusão

A hipótese apresentada neste capítulo foi a de que o uso de VBOs como vetores-alvo em redes do tipo MLP propicia desempenho superior, comparando-se ao uso dos vetores convencionalmente utilizados dos tipos VBCs e VNOs. Além disso, tal hipótese também sugeriu que o desempenho das MLPs sofre menos influência das escolhas dos parâmetros de taxa de aprendizagem inicial e número de neurônios da camada intermediária quando são usados VBOs.

Os resultados referentes à média de desempenho superior e variabilidade reduzida confirmaram que o uso de VBOs torna a rede MLP mais robusta em tarefas de reconhecimento de padrões. O comportamento das curvas de média de desempenho e a variabilidade de desempenho mostram que as redes MLP são menos suscetíveis a alterações de parâmetros quando treinadas com VBOs. Isso significa que, enquanto é necessário escolher parâmetros ideais de treinamento na utilização de VBCs e VNOs, existe um intervalo muito mais amplo no espaço de busca de parâmetros que garante desempenhos elevados quando são utilizados VBOs.

Por essa razão, a utilização de VBOs em tarefas de reconhecimento de padrões é mais aconselhável. Para usuários de redes do tipo MLP com pouca experiência, haverá facilidade na obtenção de bom desempenho. Para usuários com vasta experiência em redes MLP, haverá ganho de desempenho. Os resultados também mostram que o alcance de desempenhos satisfatórios das MLPs treinadas com VBOs ocorre com poucos ciclos. Em alguns experimentos, as redes MLP atingem desempenho satisfatório em 3 ciclos quando são utilizados VBOs, ao passo que na abordagem convencional é necessário mais de 40 ciclos para o alcance de um desempenho satisfatório. Consequentemente, há uma redução significativa no esforço computacional e no tempo de treinamento.

Capítulo 10

Publicações do Trabalho

Os resultados deste trabalho possibilitaram a publicação dos seguintes artigos :

- Manzan, J. R. G.; Nomura, S. ; Yamanaka, K. . Orthogonal bipolar vectors as multilayer perceptron targets for biometric pattern recognition. In: 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2015, Zhangjiajie. 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). IEEE Conference. p. 1164. doi 10.1109/FSKD.2015.7382107.
- Manzan, J. R. G., Yamanaka, K., Peretta, I. S., Pinto, E. R., Oliveira, T. E. C., & Nomura, S. (2016). A mathematical discussion concerning the performance of multilayer perceptron-type artificial neural networks through use of orthogonal bipolar vectors. Computational and Applied Mathematics - Springer, 1-22. doi 10.1007/s40314-016-0377-x.

Outros trabalhos foram submetidos à duas revistas, e até a publicação da Tese, ainda não houve resposta a respeito da submissão.

Capítulo 11

Conclusão

A hipótese apresentada por este trabalho é a de que a utilização de novos vetores-alvo em RNAs do tipo MLP. Sua característica de ortogonalidade mútua faz com que eles estejam geometricamente mais distantes, quando sujeitos à distância euclidiana. Essa maior distância entre os pontos do espaço de saída oferece à rede maior capacidade de generalização, redução do esforço computacional e robustez.

Foi demonstrado matematicamente por meio do algoritmo *backpropagation*, que, ao longo do treinamento, a saída obtida pela rede para cada amostra de treinamento aproxima-se do seu alvo correspondente, uma vez que a distância euclidiana diminui quando o erro quadrático diminui. Logo, se os alvos estão mais distantes uns dos outros, haverá menor probabilidade de as saídas geradas pela rede em tempo de treinamento estarem mais próximas de alvos incorretos. Consequentemente, a tarefa de classificação dos padrões será facilitada.

Além da demonstração matemática, foram realizados experimentos no intuito de medir a distância euclidiana de cada saída obtida pela rede e os alvos não correspondentes a ela. Foi constatado estatisticamente que as redes treinadas com VBOs geram saídas cujas distâncias euclidianas em relação aos alvos incorretos é maior do que aquelas geradas por redes treinadas com vetores-alvo convencionais e não ortogonais. Isso significa que os resultados experimentais confirmam a demonstração matemática de que as redes treinadas com VBOs estão menos propensas a erros de classificação.

Também foi experimentalmente analisado o desempenho das redes MLP no reconhecimento de dígitos manuscritos, íris humana e signos australianos. Os experimentos também consistiram na comparação do desempenho em vários momentos do treinamento usando-se VBCs, VNOs e VBOs. O desempenho foi medido após o primeiro ciclo de treinamento, nos ciclos iniciais, no ponto indicado para parada do treinamento pelo critério *earlystopping*, antes do ponto de parada e depois do ponto de parada. Foi verificado que as redes treinadas com VBOs têm desempenho superior em todos esses momentos do treinamento. Foi verificado também que o aumento da dimensão dos vetores convencionais não melhora o desempenho, sendo que, em alguns casos, faz com que esse desempenho seja até inferior. A curva de desempenho também mostra que o uso dos VBOs torna a rede menos suscetível ao efeito do *overfitting*. Após o ponto indicado para interrupção do treinamento, a rede MLP continua tendo bom desempenho.

Destaca-se que, além do aumento do desempenho global do treinamento, o uso de VBOs como alvos de MLPs permite inclusive que a rede MLP alcance um nível satisfatório de desempenho com pouco treinamento. Isso é especialmente importante pelo fato de reduzir o esforço computacional, considerado um inconveniente na utilização de redes do tipo MLP em tarefas de reconhecimento de padrões.

Finalmente, foi mostrado experimentalmente que a variação da taxa de aprendizagem e do número de neurônios da camada intermediária causa grande interferência no desempenho de redes treinadas com VBCs e VNOs, fenômeno que não ocorre com redes treinadas com VBOs. Ao contrário, a utilização de VBOs como alvos de MLPs fez com que o desempenho sofra pouca interferência com a alteração dos parâmetros de treinamento. Esses resultados foram obtidos a partir da utilização de três tipos de dados: dígitos manuscritos, íris humana e signos australianos. Dessa forma, a utilização dos VBOs torna a rede MLP mais robusta em tarefas de reconhecimento de padrões, permitindo que seus usuários, com pouca ou vasta experiência com esse tipo de rede, tenham maior garantia de obtenção de bons resultados em suas aplicações.

Referências Bibliográficas

Ashwini, T. R., Devi, K. R., & Gangashetty, S. V. (2012). Multilayer feedforward neural network models for pattern recognition tasks in earthquake engineering. In *Advanced computing, networking and security* (pp. 154–162). Springer.

Birlutiu, A., d'Alche Buc, F., & Heskes, T. (2014). A bayesian framework for combining protein and network topology information for predicting protein-protein interactions. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, *PP*(99), 1-1. doi: 10.1109/TCBB.2014.2359441

Casia. (2010). *Human iris, database of 756. greyscale eye images.* Retrieved from `http://www.cbsr.ia.ac.cn/IrisDatabase.htm`

Castro, C. L., & Braga, A. P. (2013). Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. *Neural Networks and Learning Systems, IEEE Transactions on*, *24*(6), 888–899.

Cireşan, D. C., Meier, U., Gambardella, L. M., & Schmidhuber, J. (2012). Deep big multilayer perceptrons for digit recognition. In *Neural networks: Tricks of the trade* (pp. 581–598). Springer.

Conover, W. (1999). *Practical nonparametric statistics*. Wiley. Retrieved from `http://books.google.com.br/books?id=dYEpAQAAMAAJ`

Costa, M. A., Braga, A. P., & de Menezes, B. R. (2003). Improving neural networks generalization with new constructive and pruning methods. *Journal of Intelligent and Fuzzy Systems*, *13*(2), 75–83.

Danisman, T., Bilasco, I. M., Martinet, J., & Djeraba, C. (2013). Intelligent pixels of interest

selection with application to facial expression recognition using multilayer perceptron. *Signal Processing*, *93*(6), 1547–1556.

Daugman, J. G. (1993, November). High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. Pattern Anal. Mach. Intell.*, *15*(11), 1148–1161. Retrieved from http://dx.doi.org/10.1109/34.244676 doi: 10.1109/34.244676

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). Pattern classification. 2nd. *Edition. New York*.

Duffner, S., & Garcia, C. (2007). An online backpropagation algorithm with validation error-based adaptive learning rate. In *Artificial neural networks–icann 2007* (pp. 249–258). Springer.

Fausett, L. (1994). *Fundamentals of neural networks: architectures, algorithms, and applications*. Prentice-Hall, Inc.

Fausett, L. V., & Hall, P. (1994). *Fundamentals of neural networks: architectures, algorithms, and applications* (Vol. 40). Prentice-Hall Englewood Cliffs.

Gonzalez, R. C. (1992). Re woods, digital image processing. *Addison–Wesely Publishing Company*.

Hallinan, P. W. (1991). Recognizing human eyes. In *San diego,'91, san diego, ca* (pp. 214–226).

Hamedi, M., Salleh, S.-H., Astaraki, M., Noor, A. M., & Harris, A. R. A. (2014). Comparison of multilayer perceptron and radial basis function neural networks for emg-based facial gesture recognition. In *The 8th international conference on robotic, vision, signal processing & power applications* (pp. 285–294).

Haykin, S. (2008). *Redes neurais: Princípios e prática*. Bookman.

Healy, E. W., Yoho, S. E., Wang, Y., & Wang, D. (2013). An algorithm to improve speech recognition in noise for hearing-impaired listeners. *The Journal of the Acoustical Society of America*, *134*(4), 3029-3038. Retrieved from http://scitation.aip.org/content/asa/journal/jasa/134/4/10.1121/1.4820893 doi: http://dx.doi.org/10.1121/1.4820893

Huang, G., Huang, G.-B., Song, S., & You, K. (2015). Trends in extreme learning machines: A

review. *Neural Networks*, *61*(0), 32 - 48. Retrieved from http://www.sciencedirect.com/science/article/pii/S0893608014002214 doi: http://dx.doi.org/10.1016/j.neunet.2014.10.001

Hwang, J.-N., Choi, J. J., Oh, S., & Marks, R. (1991). Query-based learning applied to partially trained multilayer perceptrons. *Neural Networks, IEEE Transactions on*, *2*(1), 131–136.

Iosifidis, A., Tefas, A., & Pitas, I. (2015). Dropelm: Fast neural network regularization with dropout and dropconnect. *Neurocomputing*, *162*(0), 57 - 66. Retrieved from http://www.sciencedirect.com/science/article/pii/S0925231215004130 doi: http://dx.doi.org/10.1016/j.neucom.2015.04.006

Isa, N. A. M., & Mamat, W. M. F. W. (2011). Clustered-hybrid multilayer perceptron network for pattern recognition application. *Applied Soft Computing*, *11*(1), 1457–1466.

Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaptation. *Neural networks*, *1*(4), 295–307.

Jesús, J. R., Ortiz-Rodriguez, F., Mariaca-Gaspar, C. R., & Tovar, J. C. (2013). A method for online pattern recognition of abnormal eye movements. *Neural Computing and Applications*, *22*(3-4), 597–605.

Kadous, M. W. (2002). *Temporal classification: Extending the classification paradigm to multivariate time series* (Unpublished doctoral dissertation). The University of New South Wales.

Kim, D. (2005). Improving prediction performance of neural networks in pattern classification. *International Journal of Computer Mathematics*, *82*(4), 391–399.

Kruse, R., Borgelt, C., Klawonn, F., Moewes, C., Steinbrecher, M., & Held, P. (2013). *Computational intelligence: a methodological introduction*. Springer Science & Business Media.

Lawrence, S., Burns, I., Back, A., Tsoi, A. C., & Giles, C. L. (2012). Neural network classification and prior class probabilities. In *Neural networks: Tricks of the trade* (pp. 295–309). Springer.

LeCun, Y. (1993). Efficient learning and second order methods. In *Tutorial presented at neural*

information processing systems (Vol. 5, p. 49).

Lee, C.-H., Chang, F.-K., Kuo, C.-T., & Chang, H.-H. (2012). A hybrid of electromagnetism-like mechanism and back-propagation algorithms for recurrent neural fuzzy systems design. *International Journal of Systems Science*, *43*(2), 231–247.

Lee, C.-M., Yang, S.-S., & Ho, C.-L. (2006). Modified back-propagation algorithm applied to decision-feedback equalisation. *IEE Proceedings-Vision, Image and Signal Processing*, *153*(6), 805–809.

Lichman, M. (2013). *UCI machine learning repository.* Retrieved from `http://archive.ics.uci.edu/ml`

Manzan, J. R. G., Nomura, S., & Filho, J. B. D. (2014). Eeg signal classification by improved mlps with new target vectors. In *Proceedings on the international conference on artificial intelligence (icai)* (p. 1).

Manzan, J. R. G., Nomura, S., & Yamanaka, K. (2011a). A melhoria no desempenho de mlp com o uso de novos vetores alvo. In *10th brazilian congress on computational intelligence (cbic 2011).*

Manzan, J. R. G., Nomura, S., & Yamanaka, K. (2011b). A melhoria no desempenho de mlp com o uso de novos vetores alvo..

Manzan, J. R. G., Nomura, S., & Yamanaka, K. (2012). Mathematical evidence for target vector type influence on mlp learning improvement. In *Proceedings on the international conference on artificial intelligence (icai)* (p. 1).

Manzan, J. R. G., Nomura, S., & Yamanaka, K. (2015, Aug). Orthogonal bipolar vectors as multilayer perceptron targets for biometric pattern recognition. In *Fuzzy systems and knowledge discovery (fskd), 2015 12th international conference on* (p. 1164-1170). doi: 10.1109/FSKD.2015.7382107

Manzan, J. R. G., Nomura, S., Yamanaka, K., Carneiro, M. B. P., & Veiga, A. C. P. (2012a). Improving iris recognition through new target vectors in mlp artificial neural networks. In *Artificial neural networks in pattern recognition* (pp. 115–126). Springer.

Manzan, J. R. G., Nomura, S., Yamanaka, K., Carneiro, M. B. P., & Veiga, A. C. P. (2012b).

Improving iris recognition through new target vectors in mlp artificial neural networks. In N. Mana, F. Schwenker, & E. Trentin (Eds.), *Artificial neural networks in pattern recognition* (Vol. 7477, p. 115-126). Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-642-33212-8_11

Manzan, J. R. G., Yamanaka, K., & Nomura, S. (2011). Improvement in perfomance of mlp using new target vectors (in portuguese). *In: X Brazilian Congress on Computational Intelligence - Fortaleza*.

Manzan, J. R. G., Yamanaka, K., Peretta, I. S., Pinto, E. R., Oliveira, T. E. C., & Nomura, S. (2016). A mathematical discussion concerning the performance of multilayer perceptron-type artificial neural networks through use of orthogonal bipolar vectors. *Computational and Applied Mathematics*, 1–22. Retrieved from http://dx.doi.org/10.1007/s40314-016-0377-x doi: 10.1007/s40314-016-0377-x

Martins, G. A., & Fonseca, J. S. (2006). Curso de estatística. *Atlas, 6ª Edição*.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115–133.

Mikolov, T., Kombrink, S., Burget, L., Cernocky, J. H., & Khudanpur, S. (2011). Extensions of recurrent neural network language model. In *Acoustics, speech and signal processing (icassp), 2011 ieee international conference on* (pp. 5528–5531).

Muda, A. K., Choo, Y.-H., Abraham, A., & Srihari, S. N. (2014). *Computational intelligence in digital forensics: Forensic investigation and applications*. Springer.

Negin, M., Chmielewski, T. A., Salganicoff, M., Camus, T. A., Cahn, U. M. V. S., Venetianer, P. L., & Zhang, G. G. (2000, February). An iris biometric system for public and personal use. *Computer*, *33*(2), 70–75. Retrieved from http://dx.doi.org/10.1109/2.820042 doi: 10.1109/2.820042

Nomura, S., Manzan, J. R. G., & Yamanaka, K. (2011). Análise experimental de novos vetores alvo na melhoria do desempenho de mlp. *IX Conferência de estudos em engenharia elétrica (CBIC 2011)*.

Nomura, S., Yamanaka, K., Katai, O., Kawakami, H., & Shiose, T. (2005). Improved mlp

learning via orthogonal bipolar target vectors. *JACIII*, *9*(6), 580–589.

Nomura, S., Yamanaka, K., Katai, O., Kawwakami, H., & Shiose, T. (2004). A new approach to improving math performance of artificial neural networks (in portuguese).

R Core Team. (2014). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `http://www.R-project.org/`

Ranaee, V., & Ebrahimzadeh, A. (2013). Control chart pattern recognition using neural networks and efficient features: a comparative study. *Pattern Analysis and Applications*, *16*(3), 321–332.

Ruck, D. W., Rogers, S. K., Kabrisky, M., Oxley, M. E., & Suter, B. W. (1990). The multilayer perceptron as an approximation to a bayes optimal discriminant function. *Neural Networks, IEEE Transactions on*, *1*(4), 296–298.

Rudd, K., & Ferrari, S. (2015). A constrained integration (cint) approach to solving partial differential equations using artificial neural networks. *Neurocomputing*, *155*(0), 277 - 285. Retrieved from `http://www.sciencedirect.com/science/article/pii/S092523121401652X` doi: http://dx.doi.org/10.1016/j.neucom.2014.11.058

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation* (Tech. Rep.). DTIC Document.

Sahin, U., & Sahin, F. (2012). Pattern recognition with surface emg signal based wavelet transformation. In *Systems, man, and cybernetics (smc), 2012 ieee international conference on* (pp. 295–300).

Samal, A., Panda, J., & Das, N. (2015). Performance comparison of single-layer perceptron and flann-based structure for isolated digit recognition. In *Intelligent computing, communication and devices* (pp. 237–246). Springer.

Shimizu, Y., Yoshimoto, J., Toki, S., Takamura, M., Yoshimura, S., Okamoto, Y., ... Doya, K. (2015). Toward probabilistic diagnosis and understanding of depression based on functional mri data analysis with logistic group lasso. *PLOS-One*.

Silva, I. N., Spatti, D. H., & Flauzino, R. A. (2010). Redes neurais artificiais para engenharia e ciências aplicadas curso prático. *Artliber*.

Sivaram, G. S., & Hermansky, H. (2012). Sparse multilayer perceptron for phoneme recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, *20*(1), 23–29.

Wan, J., Wang, D., Hoi, S. C. H., Wu, P., Zhu, J., Zhang, Y., & Li, J. (2014). Deep learning for content-based image retrieval: A comprehensive study. In *Proceedings of the acm international conference on multimedia* (pp. 157–166). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2647868.2654948` doi: 10.1145/2647868.2654948

Wang, X., Chang, C.-C., & Du, F. (2002). Achieving a more robust neural network model for control of a mr damper by signal sensitivity analysis. *Neural Computing & Applications*, *10*(4), 330–338.

Xu, Y., Xu, D., Lin, S., Han, T. X., Cao, X., & Li, X. (2012, June). Detection of sudden pedestrian crossings for driving assistance systems. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *42*(3), 729-739. doi: 10.1109/TSMCB.2011.2175726

Zarei, J. (2012). Induction motors bearing fault detection using pattern recognition techniques. *Expert systems with Applications*, *39*(1), 68–73.