Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Programa de Pós Graduação em Engenharia de Eletricidade

Priscila Lima Rocha

Reconhecimento de Voz utilizando Seleção Dinâmica de Redes Neurais

São Luís – MA

2018

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Programa de Pós Graduação em Engenharia de Eletricidade

Priscila Lima Rocha

Reconhecimento de Voz utilizando Seleção Dinâmica de Redes Neurais

Dissertação submetida à Coordenação de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, Campus Bacanga como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Allan Kardec Duailibe Barros Filho
Coorientador: Washington Luís Santos Silva

São Luís – MA
2018

Rocha, Priscila Lima

Reconhecimento de voz utilizando seleção dinâmica de redes neurais /Priscila Lima Rocha. – 2018.

111f.: il.

Coorientador(a): Washington Luis Santos Silva.
Orientador(a): Allan Kardec Duailibe Barros Filho.
Dissertação (Mestrado) – Programa de Pós-graduação em Engenharia de Eletricidade/ccet, Universidade Federal do Maranhão, São Luís, 2018.

1. Mistura de Especialistas. 2. Reconhecimento Automático de Voz. 3. Redes Neurais. I. Barros Filho, Allan Kardec Duailibe. II. Silva, Washington Luis Santos. III. Título.

Priscila Lima Rocha

Reconhecimento de Voz utilizando Seleção Dinâmica de Redes Neurais

Dissertação submetida à Coordenação de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, Campus Bacanga como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Trabalho aprovado. São Luís – MA, 23 de fevereiro de 2018:

Prof. Dr. Allan Kardec Duailibe Barros Filho
Orientador

Prof. Dr. Washington Luís Santos Silva
Co-orientador

Prof. Dr. José Carlos Príncipe
Membro da Banca Examinadora

Prof. Dr. Francisco das Chagas de Souza
Membro da Banca Examinadora

São Luís – MA
2018

Este trabalho é dedicado à minha família e à todos que contribuíram direta ou indiretamente para a concretização do mesmo.

Agradecimentos

A Deus, em primeiro lugar, pois tudo aquilo que conquistei até aqui é devido a Sua vontade.

Ao meus pais, Vicente Rocha e Obede Lima, pelo amor incondicional, cuidado e amizade; por não medirem esforços para que eu pudesse dedicar-me integralmente aos estudos e por acreditarem nos meus sonhos. Amo vocês.

À minha irmã, Poliana, pelo carinho e companheirismo em todos os momentos. Pela paciência e compreensão nos meus momentos de estresses durante a faculdade e por me fazer rir com suas imitações e brincadeiras. Te amo minha Poly!

A todos quantos contribuíram direta e indiretamente para o desenvolvimento deste trabalho.

"Acredite que você pode, assim você já está no meio do caminho."
(Theodore Roosevelt)

Resumo

Este trabalho propõe uma arquitetura hierarquizada composta por um conjunto redes neurais especialistas baseada no método de comitês com seleção dinâmica de classificadores para aplicação em sistemas de reconhecimento de sinais de voz. A tarefa de reconhecimento de padrões proposta neste trabalho envolve um grupo de 30 comandos na língua portuguesa brasileira. Estes comandos são codificados por uma matriz temporal bidimensional, resultante da aplicação da Transforma Cosseno Discreta (TCD) nos coeficientes mel-cepstrais. Para evitar o problema de separabilidade dos padrões, eles são modificados através de uma transformação não linear para um espaço de alta dimensionalidade através de um conjunto de Funções de Base Radial Gaussiana (FBRG). A classificação é feita por meio do método de seleção dinâmica de classificadores, na qual as configurações Perceptron de Múltiplas Camadas (*Multilayer Perceptron* - MLP) e Aprendizado por Quantização Vetorial (*Learning Vector Quantization* - LVQ) são analisadas para constituir os múltiplos classificadores especializados nas subdivisões realizadas no total de classes a serem reconhecidas. Os desempenhos destas configurações são avaliados durante as fases de treinamento, validação e teste do sistema de reconhecimento de voz. Então, dado um novo padrão de teste, este é aplicado ao conjunto de FBRG, onde cada função está parametrizada com as características de centroide e variância das classes. Logo, aquela a FRBG que apresentar o maior valor de imagem para a função indica a que classe este padrão está localizado, direcionando assim, para a rede neural especialista que fornecerá o resultado final de classificação baseada na acurácia local. Ao final, verificou-se o desempenho das configurações de redes neurais escolhidas para a composição dos múltiplos classificadores. O resultado da comparação entre as configurações MLP e LVQ para o sistema proposto mostrou que a taxa de acurácia global utilizando padrões de dimensões 4, 9 e 16 no espaço de características original para as redes LVQ ficou em 87.52%, 88.39% e 89.6% , respectivamente. Já as redes MLP obtiveram uma taxa de acurácia global de 91.44%, 93.15% e 94.9%, respectivamente.

Palavras-chave: Redes Neurais, Reconhecimento Automático de Voz, Coeficientes Mel-Cepstrais, Modelos TCD, Perceptron de Múltiplas Camadas, Aprendizado por Quantização Vetorial, Função de Base Radial Gaussiana, Mistura de Especialistas.

Abstract

This work proposes a hierarchical architecture composed of a set of neural networks specialists based on the ensemble method with dynamic selection of classifiers for application in speech recognition systems. The task of pattern recognition proposed in this work involves a group of 30 commands in the Brazilian Portuguese language. These commands are coded by a two-dimensional temporal matrix, resulting from the application of the Discrete Cosine Transformation (DCT) in the mel-ceptral coefficients. To avoid the problem of separability of the patterns, they are modified through a nonlinear transformation to a high-dimensional space through a suitable set of Gaussian Radial Base Functions (GRBF). The classification is done through the dynamic classifier selection method, in which Multilayer Perceptron (MLP) and Vector Vector Quantization Learning (LVQ) configurations are analyzed to constitute the multiple classifiers specialized in the subdivisions made in the total of classes to be recognized. The performances these configurations are evaluated during the training, validation and testing phases of the voice recognition system. Then, given a new test pattern, this is applied to the GRBF set, where each function is parameterized with the centroid and variance characteristics of the classes. Therefore, the GRBF that present the highest image value for the function indicates to which class this pattern is located, thus directing, to the specialist neural network which will provide the final classification result based on the local accuracy. At the end, the performance of the neural network configurations chosen for the composition of the multiple classifiers was verified. The result of the comparison between MLP and LVQ configurations for the proposed system showed that the overall accuracy rate using patterns of dimensions 4, 9 and 16 in the original feature space for the LVQ networks was 87.52 %, 88.39 % and 89.6 %, respectively. The MLP networks obtained an overall accuracy rate of 91.44 %, 93.15 % and 94.9 %, respectively.

Keywords: Automatic Speech Recognition, Neural Network, DCT Models, Multilayer Perceptron, Learning Vector Quantization, Gaussian Radial Basis Function, Mixture of Experts.

Trabalhos Publicados pelo Autor

0.1 Capítulos de Livros

- Lima, Priscila; Barros, Allan ; Silva, Washington. *Neural Network Configurations Analysis for Identification of Speech Pattern with Low Order Parameters*. Studies in Computational Intelligence. 1ed.: Springer International Publishing, 2018, v. 751, p. 349-370.

- Lima, Priscila; Barros, Allan ; Silva, Washington. *Neural Network Configurations Analysis for Multilevel Speech Pattern Recognition System with Mixture of Experts*. Intelligent System. IN TECH d.o.o. Publishing, 2018, ISBN 978-953-51-5879-0

0.2 Artigos em Anais de Congressos

- ROCHA, PRISCILA LIMA; DUAILIBE FILHO, A. K. B. ; SILVA, W. L. S. . Análise de Configurações de Redes Neurais para Sistema de Reconhecimento de Padrões de Sinais de Voz utilizando Parâmetros TCD de Baixa Ordem. In: *XIII Simpósio Brasileiro de Automação Inteligente*, 2017, Porto Alegre.

- ROCHA, P. L.; SILVA, W. L. S. ; DUAILIBE FILHO, A. K. B. Neural Networks Applied to System of Speech Recognition based on DCT Parametric Models of Low Order. *In: Congresso Brasileiro de Automática - CBA*, 2016, Vitória.

- ROCHA, PRISCILA LIMA; SILVA, WASHINGTON LUIS SANTOS . Intelligent system of speech recognition using Neural Networks based on DCT parametric models of low order. In: *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, Vancouver.

- ROCHA, P. L.; SILVA, W.L.S. Recognition System of Numerical Comand of Speech Signal using Neural Networks based on DCT Parameters. In: *Intelligent Sstems Conference 2016 (IntelliSys 2016)*, 2016, Londres.

- ROCHA, PRISCILA LIMA; SILVA, WASHINGTON LUIS SANTOS . Artificial neural networks used for pattern recognition of speech signal based on DCT parametric models of low order. In: *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, 2016, Poitiers.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

MLP	Percetron de Multicamadas (*Multilayer Perceptron*)
LVQ	Quantização Vetorial por Aprendizagem (*Learning Vector Quantization*)
TCD	Transformada Cosseno Discreta
SRV	Sistemas de Reconhecimento de Voz
HMM	Modelos Ocultos de Markov (*Hidden Markov Models*)
RNA	Redes Neurais Artificiais
LPC	Codificação Preditiva Linear (*Linear Predictive Code*)
Mfcc	Coeficientes mel-cepstraias (*Mel-Frequency Cepstrum Coefficient*)
AI	Inteligência Artificial (*Artificial Intelligence*)
SOM	Mapas Auto-Organizáveis (*Self-Organization Maps*)
LVQ-1	Algoritmo de Treinamento Quantização Vetorial por Aprendizagem
LVQ-2	Algoritmo de Treinamento Quantização Vetorial por Aprendizagem
EPUSP	Escola Politécnica da Universidade de São Paulo
Inatel	Instituto Nacional de Telecomunicações
IFMA	Instituto Federal do Maranhão
EMQ	Erro Médio Quadrático
GD	Gradiente descendente
GDM	Gradiente descendente com constante de *momentum*
RP	Resilient Propagation
LM	Levenberg-Marquardt
FBRG	Função de Base Radial Gaussiana
GMM	Modelos de Misturas Gaussianas (*Gaussian Mixture Model*)

Lista de símbolos

$C_k(n,T)$	Matriz temporal bidimensional de ordem $k \times n$, calculados no segmento T
D_m^j	m-ésimo exemplo do modelo do dígito j
C_{kn}^{jm}	Matriz temporal bidimensional do m-ésimo exemplo da palavra j
C_N^{jm}	Vetor coluna gerado a partir de C_{kn}^{jm}
η	Taxa de aprendizagem
Ω_{NL}^{Tr}	Conjunto de treinamento
Ω_{NL}^{TM}	Conjunto de teste utilizando locutores masculinos
Ω_{NL}^{TF}	Conjunto de teste utilizando locutores femininos
Φ	Conjunto de funções de base radial gaussiana
χ	Espaço de características não linear de alta dimensionalidade
μ_j	Centroide da j classe
σ_j^2	Variância das j funções de base radial gaussianas
γ	Conjunto de classes de um problema multiclasse
$AL_{T,K}$	Acurácia Local da T-ésimo classificador para a região de competência do espaço de característica K

Sumário

1 Introdução

O reconhecimento ou classificação de padrões é uma das ações mais executadas pelos seres humanos. Esta é a maneira mais comum de como os indivíduos tomam determinada decisão ao observar características especificas que representam um determinado elemento ou objeto. Esta capacidade de reconhecimento de padrões pelos seres humanos envolve os sofisticados sistemas neural e cognitivo, que a partir do acúmulo de experiência sobre determinado meio, conseguem extrair as características relevantes que modelam determinada situação e armazenam essa informação para o uso quando houver uma necessidade, tornando o processo decisório muito mais veloz. Dessa forma, muitos pesquisadores trabalham no intuito de entender o mecanismo de reconhecimento de padrões biológico dos seres humanos para o desenvolvimento de algoritmos computacionais de aprendizado de máquinas cada vez mais robustos para utilização em aplicações práticas (DOUGHERTY, 2012; DUDA; HART; STORK, 2001).

Reconhecimento de padrões é uma área científica que tem por objetivo classificar os elementos de acordo com suas características, que formam um espaço multidimensional (espaço de características), em conjuntos distintos, que são denominados classes ou rótulos ou categorias para que posteriormente uma ação possa ser melhor desempenhada de acordo com cada categoria. Uma vez que é necessário exemplos dos padrões para obter os distintos conjuntos existentes, o processo de reconhecimento de padrões envolve uma análise estocástica para obtenção dos modelos, além da inserção ou não do conhecimento do especialista no domínio da aplicação, que pode caracterizar uma classificação supervisionada ou não supervisionada, respectivamente.

Dentre as diversas aplicações desta área, a tarefa de reconhecimento de sinais de voz é desafiante, já que os sinais obtidos no processo de produção da voz são altamente variáveis, devido à grande quantidade de atributos da voz humana, além das características próprias envolvidas na fala, como os ruídos de fundo e as propriedades de cada idioma. O desenvolvimento de sistemas baseados em reconhecimento de padrões de sinais de voz é uma das aplicações práticas de classificação de padrões. A voz é, de fato, o modo mais natural e expressivo de comunicação humana e, dessa forma, metodologias para a análise e reconhecimento do sinal de voz vêm sendo desenvolvidos influenciadas pelo conhecimento de como essa tarefa é solucionada pelos seres humanos (DOUGHERTY, 2012; DUDA; HART; STORK, 2001).

Atualmente, as aplicações de reconhecimento de sinais de voz abrangem os mais diversos domínios, a exemplo de: ferramentas de ditado em editores de texto, serviços de atendimento automático em centrais telefônicas, sistemas baseados em “mãos-livres”

(*hands-free*) em automóveis, acessibilidade de pessoas com deficiência motora, interface mobile via fala, aplicações de reservas de passagens em companhias aéreas, sistemas de segurança por identificação de voz, entre outros, que justificam a necessidade pelo desenvolvimento de sistemas cada vez mais robustos (HUSNJAK; PERAKOVIC; JOVOVIC, 2014; ŠPALE; SCHWEIZER, 2016; WENG et al., 2016).

A tarefa de reconhecimento de padrões envolve diferentes etapas e a execução eficiente de cada uma delas garante maior acurácia nos resultados. Dessa forma, as etapas necessárias para o desenvolvimento de um sistema de reconhecimento de padrões são:

1. Aquisição dos dados, pré-processamento e extração das características mais relevantes;
2. Representação dos dados;
3. Definição do classificador para tomada de decisão.

As técnicas de processamento digital de sinais e a codificação digital de sinais são as ferramentas que dão suporte a representação dos padrões. Os avanços nas metodologias de processamento digital de sinais de voz permitem o uso eficaz dos atributos do sinal para a utilização em reconhecimento da locução ou do locutor, dependendo da aplicação (CHADLI; BOUOUDEN; ZELINKA, 2016; BELLEGARDA; MONZ, 2016). Além da necessidade da extração de bons atributos para representar os padrões a serem reconhecidos, é importante também que estes padrões tenham uma quantidade reduzida de parâmetros. De fato, quanto mais informações forem adicionadas ao sistema, maior será a probabilidade de bons resultados na classificação. Porém, essa relação deve ser tomada com cautela, pelo fato de este incremento de dados aumentar a complexidade do sistema, o custo computacional e perda de generalização. Dessa forma, as técnicas de codificação digital de sinais contribuem significativamente para determinar o ponto de equilíbrio entre quantidade de parâmetros e custo computacional (PICONE, 1993b).

Após o processo de codificação do sinal de voz e obtenção dos padrões representativos pertencentes ao espaço amostral de identificação, a tarefa do reconhecimento pode ser realizada de forma eficiente utilizando algoritmos de classificação de padrões, conforme a terceira etapa citada. Estes algoritmos, também chamados de classificadores, desenvolvem modelos que generalizam cada categoria ou classe pertencente ao sistema. Isto é feito a partir de um conjunto de padrões, denominado conjunto de treinamento, onde cada padrão é identificado a classe que pertence através de um rótulo. Então, na fase de teste, o classificador é capaz de determinar em que categoria um novo padrão pertence (KAUTZ; ESKOFIER; PASLUOSTA, 2017).

Um ponto crucial para os classificadores é a determinação dos limites de decisão entre as classes, isto é, especificar o modelo que permita a identificação de um novo dado. Isto

se torna mais complexo a medida que o número de classes aumenta. Diversas abordagens de classificadores estão disponíveis na literatura, mas, quase sempre, estas abordagens são propostas para resolver problemas de classificação entre duas classes (). Entretanto, aplicações práticas exigem a discriminação entre múltiplas classes, o que demanda classificadores mais complexos que classificadores binários. A utilização de apenas uma única estrutura compacta para aprendizagem das características particulares de uma tarefa multiclasse pode elevar o custo computacional e a capacidade de generalização (SONG; JIANG; LIU, 2017).

Para contornar esta situação, o método de comitês de classificadores visa, a partir do princípio *dividir para conquistar*, fragmentar o espaço de características para que um conjunto de classificadores de topologia mais simples aprendam as especificidades de cada subespaço e ao final, o resultado da classificação será dado pelos resultados individuais ou pela escolha do resultado de um dos classificadores, segundo uma determinada regra. Dessa forma, eleva-se o resultado da tarefa multiclasse a partir de classificadores mais simples (KHERADPISHEH et al., 2014; SHIH; CHEN; WU, 2017).

Portanto, este trabalho propõe uma arquitetura hierarquizada composta por um conjunto redes neurais especialistas baseada no método de comitês com seleção dinâmica de classificadores para aplicação em sistemas de reconhecimento de sinais de voz. A tarefa de reconhecimento de padrões proposta neste trabalho envolve um grupo de 30 comandos na língua portuguesa brasileira. Estes comandos são codificados por uma matriz temporal bidimensional, resultante da aplicação da Transforma Cosseno Discreta (TCD) nos coeficientes mel-cepstrais. Para evitar o problema de separabilidade dos padrões, eles são modificados através de uma transformação não linear para um espaço de alta dimensionalidade através de um adequado conjunto de Funções de Base Radial Gaussiana (FBRG).

A classificação é feita por meio do método de seleção dinâmica de classificadores, na qual as configurações Perceptron de Múltiplas Camadas (*Multilayer Perceptron* - MLP) e Aprendizado por Quantização Vetorial (*Learning Vector Quantization* - LVQ) são analisadas para constituir os múltiplos classificadores especializados nas subdivisões realizadas no total de classes a serem reconhecidas. Os desempenhos destas configurações são avaliados durante as fases de treinamento, validação e teste do sistema de reconhecimento de voz (HAYKIN, 2009; SILVA; SPATTI; FLAUZINO, 2010).

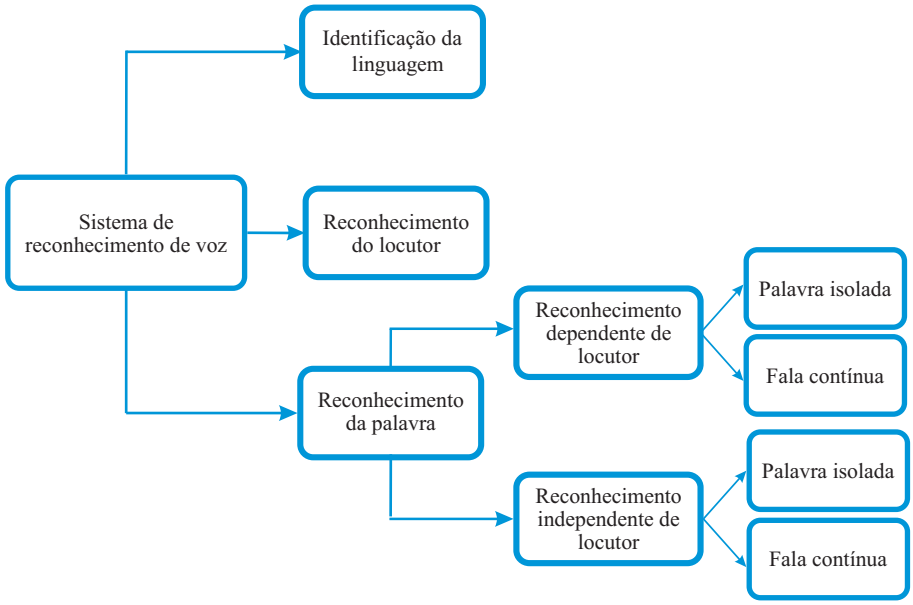
Então, dado um novo padrão de teste, este é aplicado ao conjunto de FBRG, onde cada função está parametrizada com as características de centroide e variância das classes. Logo, aquela a FRBG que apresentar o maior valor de imagem para o vetor de características de entrada indica a que classe este padrão está localizado, direcionando assim, para a rede neural especialista que fornecerá o resultado final de classificação baseada na acurácia local.

1.1 Sistemas de Reconhecimento de Voz – SRV

1.1.1 Tipos de Sistemas de Reconhecimento de Voz

Os sistemas de reconhecimento de voz definem-se por extrair características significativas do sinal de voz, obtendo-se assim, um padrão que represente este sinal e o classifique dentro de um espaço-alvo de classes definido no projeto de reconhecimento. Uma classe reúne padrões que possuem características similares. O objetivo do reconhecimento do sinal de voz permite que tais sistemas sejam relacionados de três formas: reconhecimento do locutor, identificação da linguagem e reconhecimento da palavra. Apresenta-se na Figura 1 um diagrama esquemático das ramificações que o sistema de reconhecimento de voz pode apresentar.

Figura 1 – Ramificações de um sistema de reconhecimento de voz.



Fonte: adaptado de Bresolin (2008, p. 2)

Assim, sistemas cujo foco é o reconhecimento do locutor visam distinguir entre diferentes indivíduos, a partir dos parâmetros extraídos do sinal de voz, a pessoa que pronunciou determinada palavra ou sentença. Já na identificação da linguagem, o objetivo do sistema de reconhecimento é determinar em que idioma tal palavra ou sentença é pronunciada. Finalmente, para o reconhecimento da palavra, o interesse é identificar qual palavra ou sentença foi pronunciada.

Tem-se, para o caso em que o sistema de reconhecimento de voz propõe-se distinguir a palavra ou sentença falada, a divisão em duas formas diferentes: o reconhecimento da palavra dependente do locutor e o reconhecimento da palavra independente do locu-

tor. Assim, no primeiro caso, tem-se o sistema treinado para identificar a palavra que foi falada por um indivíduo específico. Já no segundo caso, se reconhece a palavra de forma autônoma de quem pronuncia, ou seja, o sistema identifica a palavra ou sentença falada por pessoas diferentes daquelas utilizadas durante o treinamento.

Além da questão da dependência ou não do locutor, o reconhecimento da palavra pode ser realizado através de palavras isoladas ou da fala contínua. No primeiro caso, a locução entre uma palavra e outra deve ter um intervalo. Isto é feito para que se tenha uma distinção clara do início e fim da palavra, evitando o efeito da coarticulação que provoca alteração na forma de pronunciar os sons. Para o caso da fala contínua, o locutor fala de maneira natural, e, dessa forma, não se consegue distinguir o início e o fim de uma palavra, ocasionando a concatenação das mesmas (DELLER; HANSEN; PROAKIS, 2000; FURUI, 2000; RABINER; JUANG, 1993).

O reconhecimento da fala contínua é mais complexo devido não existir pausa entre uma palavra e outra, gerando um único som. Porém, sistemas que trabalham com esta forma de reconhecimento baseiam-se em unidades menores da palavra, como sílabas, fonemas, difones, trifones e etc (MORGAN; SCOFIELD, 2012; BRESOLIN, 2008).

1.1.2 Aprendizado Multiclasse e Teoria de decisão de Bayes

A teoria de decisão estatística ou teoria de decisão de Bayes é o fundamento clássico para definir matematicamente a tarefa de reconhecimento de padrões. Esta abordagem expressa a solução do problema em termos probabilísticos. Dessa forma, classificadores projetados a partir da regra de decisão de Bayes constituem-se em classificadores ótimos, na qual novas abordagens de classificação podem tomá-los como referencial para comparação dos resultados (KATAGIRI, 2000; FERREIRA, 2007; SILVA, 2015).

A regra de classificação baseada na Teoria de Bayes é melhor entendida quando analisada para tomar a decisão entre duas classes, porém, esta definição pode ser generalizada para a solução de uma tarefa multiclasse. Dessa forma, é possível calcular a probabilidade *a posteriori* da i-ésima classe γ_i ocorrer dado que um vetor de características de um dado padrão $\mathbf{x}$ é apresentado por meio da fórmula de Bayes (1.1):

$$\mathrm{P}(\gamma_i|\mathbf{x}) = \frac{p(\mathbf{x}|\gamma_i)\mathrm{P}(\gamma_i)}{p(\mathbf{x})} \tag{1.1}$$

onde:
$\mathrm{P}(\gamma_i|\mathbf{x})$ é a probabilidade *a posteriori* da classe γ_i;
$p(\mathbf{x}|\gamma_1)$ é a probabilidade condicional do padrão $\mathbf{x}$ dentro da classe γ_i;
$\mathrm{P}(\gamma_i)$ é a probabilidade *a priori* da classe γ_i;
$p(\mathbf{x})$ é a função densidade de probabilidade, dado por $\mathrm{p}(\mathbf{x}) = \sum_{k=1}^{i} p(\mathbf{x}|\gamma_k)\mathrm{P}(\gamma_k)$.

Observa-se pela fórmula de Bayes que a estimativa é relevante somente se as características do padrão desconhecido possuírem a mesma densidade de probabilidade condicional dos padrões que foram usados para estimar as verossimilhanças.

Nota-se que, se as probabilidades *a priori* são iguais, as probabilidades *a posteriori* são independentes das probabilidades *a priori*, de modo que a classificação depende, exclusivamente, das probabilidades das classes. Caso as verossimilhanças sejam iguais, isto é, se as características não têm qualquer poder discriminativo, a classificação depende somente das probabilidades *a priori*.

Considerando a classificação em mais de duas classes, ou seja, o objetivo é discriminar, entre uma das i classes do conjunto de classes $\gamma = \{\gamma_1, \gamma_2, \cdots, \gamma_i\}$, o vetor de características $\mathbf{x}$. Assim, o conjunto das probabilidades condicionais de cada classe obtidas pela fórmula de Bayes é (1.2):

$$\mathrm{P}(\gamma|\mathbf{x}) = \{\mathrm{P}(\gamma_1|\mathbf{x}), \mathrm{P}(\gamma_2|\mathbf{x}), \cdots, \mathrm{P}(\gamma_i|\mathbf{x})\} \tag{1.2}$$

Então, generalizando a regra de decisão de Bayes, o vetor de característica $\mathbf{x}$ será alocado na classe com maior probabilidade, dado por (1.3):

$$\widehat{\gamma} = \underset{\gamma_i \in \gamma}{\operatorname{argmax}} \left[\mathrm{P}(\gamma_i|\mathbf{x})\right] \tag{1.3}$$

Apesar do formalismo matemático Bayesiano, há uma grande dificuldade na aplicação prática do Teorema de Bayes devido à estimação das quantidades do lado direito da equação (1.1). A obtenção de uma boa estimativa das probabilidades a priori das classes $P(\gamma_i)$ é geralmente uma tarefa fácil, feita através da simples contagem de frequência de cada classe na amostra. Em contraste, a estimação das verossimilhanças $P(\mathbf{x}|\gamma_i)$ está sujeita a dificuldade conhecida como maldição da dimensionalidade que diz que o número de classes necessárias para uma estimativa confiável de verossimilhança cresce exponencialmente com a dimensão do vetor de características. Logo, quando representações de baixo nível dos padrões são utilizadas, o número de características pode ser muito grande.

Esta dificuldade aumenta quando o número de estimativas em um problema multiclasses deve ser definido simultaneamente com elevada acurácia, uma vez que os limites entre as distintas classes podem não ser bem definidas. Assim, novas metodologias são propostas para a obtenção de resultados mais robustos em tarefas de classificação multiclasses (BELLEGARDA; MONZ, 2016; DREYFUS, 2005; BISHOP, 1995).

1.2 Revisão Bibliográfica

Atualmente, o avanço dos sistemas de reconhecimento automático de voz direciona-se para a utilização das técnicas de aprendizagem profunda. Diversos trabalhos na literatura buscam melhorar tanto os aspectos da modelagem acústica do sinal de voz quanto do processo de classificação dos padrões. A utilização das técnicas de aprendizagem profunda em diversas áreas, inclusive em processamento de sinais de voz, foi motivada pelo aumento da capacidade computacional de processamento, permitindo que algoritmos cada vez mais complexos pudessem ser desenvolvidos utilizando um amplo banco de dados.

Hinton et al. (2012) apresentaram em seu trabalho um sistema híbrido de reconhecimento de voz que utiliza a técnica estatística clássica de modelagem do sinal de voz, os Modelos Ocultos de Markov (Hidden Markov Models - HMM) em conjunto com redes neurais profundas (*Deep Neural Network* - DNN). O artigo apresenta os resultados encontrados por três diferentes grupos de pesquisa que aplicaram esta abordagem em tarefas de reconhecimento de voz e compararam os resultados com a abordagem clássica nessa área: HMM associado ao Modelo de Misturas Gaussianas (*Gaussian Mixture Models*-GMM). O trabalho apresenta o processo de composição das camadas e de treinamento das redes neurais profundas propostas. O treinamento foi realizados utilizando o banco de dados TIMIT (MOHAMED et al., 2011; MOHAMED; DAHL; HINTON, 2012) e as melhores DNN's para a base TIMIT foram utilizadas em cinco diferentes bancos de voz amplo vocabulário. Observou-se que o sistema HMM-DNN apresentaram melhores resultados do que os sistemas HMM-DNN para os experimentos realizados.

Cai e Liu (2016) propõe uma melhoria no processo de treinamento das DNN's ao substituir os neurônios ocultos com função sigmoidal pelas funções de ativação *Maxout*, que evita o problema de branqueamento do gradiente. Então, os autores aplicam a função *Maxout*em duas populares estruturas de DNN para modelamento acústico, denominadas rede neural convolucional (*convolutional neural network*-CNN) e a rede neural recorrente de memória longa de curto-prazo (*long short-term memory recurrent neural network* - RNN LSTM). Diversas combinações dessas estruturas foram propostas e os experimentos foram realizados utilizando o banco de dados proveniente do Programa IARPA Babel que possui 6 linguagens diferentes. Em média, o banco de treinamento utilizado possui 87,9 horas de áudio. A abordagem realizada pelos autores também foi comparada com o método GMM-HMM em duas variantes. Os resultados obtidos mostraram-se superiores tanto em relação as estruturas de DNN sem a função de ativação *Maxout* quanto quando comparadas ao método GMM-HMM.

Li et al. (2015) comparam o desempenho de diferentes unidades de modelagem acústica em redes neurais profundas (DNNs) baseado em sistemas de reconhecimento de voz contínua com amplo vocabulário para a língua chinesa. São observados três unidades básicas de modelagem acústica: silabas, inicial/final e fonemas. O conjunto de treinamento

contém em torno de 30 horas de locução. Os autores também fazem uma comparação do resultados com o classificador GMM. Então, concluiu-se no trabalho, após todas as comparações, que o reconhecimento de voz na língua Chinesa utilizando DNN teve uma grande melhoria. Comparada com o melhor desempenho dos sistemas baseados em GMM, as DNN obtiveram uma diminuição de mais que 20% da taxa de erro por carácter.

Sainath et al. (2015) descreveram em seu trabalho a utilização de redes neurais convolucionais (*Convolution Neural Networks* - CNN) como uma abordagem mais eficiente do que as redes neurais profundas para sistemas de reconhecimento de voz. Neste artigo, os autores analisam vários aspectos, como a melhor arquitetura da CNN, as melhores de características do sinal de voz para constituir a entrada e a melhor forma de utilizar os elementos que constituem as camadas da rede convolucional (unidades lineares retificadas - ReLU e *dropout*). Após essa análise, os autores comparam o desempenho da CNN em relação a DNN e GMM/HMM em uma tarefa de reconhecimento de voz contínuo com amplo vocabulário com um banco de dados de 50 e 400 horas de notícias de transmissão em inglês (NTI) além de 300 horas de dados conversacionais de telefonia inglesa americana do *corpus Switchboard*(S). Assim, para todos os bancos testados, A arquitetura de CNN as propostas de locutor adaptado e *ReLU+dropout* permitiram uma melhora relativa de 12%–14% na taxa de erro por palavra acima de complexas DNN.

A empresa Google tem investido no aperfeiçoamento dos serviços que utilizam reconhecimento de comandos de voz. Durante a conferência do Google I/O 2015, foi revelado que a taxa de erros no sistema de reconhecimento de voz da empresa caiu de 23% para 8% em apenas um ano, de 2013 para 2014. O sucesso na tecnologia está no uso de Redes Neurais Profundas (*Deep Neural Networks*), que é um sistema interconectado e formado por camadas que envia quantidades imensas de dados para a inteligência artificial da empresa de forma parcelada. Assim, a máquina "aprende" determinada quantidade de informações a serem reconhecidas, e de acordo com as respostas obtidas, recebe a próxima carga para corrigir erros, expandir idiomas e aprimorar o que ela já adquiriu. As Redes Neurais Profundas da Google já possuem atualmente mais de 30 camadas e a objetivo é que a máquina compreenda o usuário e antecipe os próprios movimentos. Pesquisas no *Google Now*, reconhecimento de objetos no *YouTube* e até otimização de data centers são beneficiados com a melhoria no sistema de captura de comandos de voz (KLEINA, 2015).

Além dessas aplicações utilizando aprendizado profundo, encontram-se sistemas de reconhecimento de voz que utilizam bancos de dados menores e técnicas de classificação de padrões menos complexas e com menor exigência de processamento. Essas abordagens são interessantes para o uso em plataformas embarcadas.

Silva (2015) propôs uma metodologia inteligente para reconhecimento de voz. Neste trabalho, os autores utilizaram coeficientes mel–cepstrais e a transformada cosseno discreta para gerar uma matriz temporal bidimensional para cada padrão a ser reconhecido.

O reconhecimento é realizado através de um sistema de inferência fuzzy-Mamdani que é otimizado pelo algoritmo genético para maximizar o reconhecimento dos padrões com um número mínimo de parâmetros de codificação. Os experimentos são realizados com os 10 dígitos na língua portuguesa e os resultados são comparados com outras técnicas amplamente citadas na literatura.

Rajput e Verma (2014) apresentaram uma abordagem para reconhecimento da fala das letras do alfabeto inglês implementando uma rede neural auto-organizada. A extração das características do sinal de voz é feita com a codificação Preditiva Linear *(Linear Predictive Code – LPC)*. Para o sistema proposto, a rede Neural é treinada baseada no erro que é calculado da diferença entre as saídas desejadas e a saída atual. O valor do erro calculado é utilizado para atualizar os valores dos pesos.

Tang (2009) propôs em seu artigo um sistema híbrido HMM-RNA para o reconhecimento automático de voz. O híbrido HMM/RNA assume que a saída de uma RNA é enviada para o HMM para o reconhecimento automático de voz. A arquitetura assenta-se sobre uma interpretação probabilística das saídas da RNA. O modelo do HMM é primeiramente desenvolvido usando sinais de voz representados por dígitos compilados por Bellcore. Existem 9 estados no HMM. Para cada digito, 120 exemplos foram usados para o treinamento e 38 exemplos foram usados no teste. A rede neural foi projetada contendo 20 neurônios na camada escondida, 10 neurônios na cada de saída e 220 neurônios para a camada de entrada. Para a camada escondida foram realizados treinamentos com diferentes números de neurônios. O índice do erro médio quadrático *(mean square error-MSE)* por exemplar depois de 13000 interações com o algoritmo de treinamento *backpropagation* é usado para determinar o número de neurônios na camada oculta. O espaço de confusão é construído analisando os resultados do erro de reconhecimento do HMM e encontrando a segunda e a terceira melhor solução por meio da modificação do algoritmo de Viterbi. O hibrido HMM/RN tenta resolver esta confusão.

Seman, Bakar e Bakar (2010) apresentaram uma medida de desempenho do reconhecimento de voz de palavras isoladas da língua Malaya usando Redes Neurais de Multicamadas. O vocabulário a ser reconhecido compreende 25 palavras. A tarefa de segmentação da voz é executada através dos parâmetros baseado na energia e na medida da taxa de cruzamento por zero com modificações para localizar os melhores pontos de começo e final das palavras faladas. Os segmentos foram codificados utilizando os coeficientes mel-cepstrais. Três arquiteturas diferentes com duas camadas foram utilizadas, mantendo-se as mesmas entradas, atribuindo os mesmos alvos, mesmas funções de ativação, mesma estrutura da camada de saída, os mesmos parâmetros da rede e diferindo somente nas funções de aprendizado e os neurônios da camada escondida. A camada de saída tem 25 neurônios que correspondem as 25 palavras a serem reconhecidas. Experimentalmente foram selecionados 50, 100 e 150 neurônios na camada oculta para fazer

a alteração da arquitetura. A função de ativação da camada escondida escolhida foi a tangente hiperbólica e para a camada de saída a função usada foi a sigmoide logística. As redes foram treinadas com os algoritmos Gradiente Conjugado e de Levenberg-Marquardt.

1.3 Justificativa e Motivação

O processamento de sinais de voz tem por objetivo a transformação, de forma eficiente e precisa, do sinal acústico da voz para sua utilização em sistemas automáticos. O amplo desenvolvimento de pesquisas na área de processamento de voz demonstra o esforço para a melhoria de desempenho de sistemas de reconhecimento de voz para aplicações práticas.

A utilização de tais sistemas permite autonomia em áreas como telefonia, em que solicitações de serviços são direcionadas por meio de comandos de voz (CARDOSO et al., 2010); na automobilística, através do acionamento de dispositivos no interior dos automóveis (QIAN; LIU; JOHNSON, 2009; HUA; NG, 2010; LI et al., 2013); em sistemas de computação, por meio de programas utilitários em computadores, além da aplicação em robótica (KOO et al., 2014; BALAGANESH et al., 2010; ABDELHAMID; ABDULLA, 2013) e em automação residencial e hospitalar para a acessibilidade de pessoas com deficiências locomotoras e visuais (GNANASEKAR; JAYAVELU; NAGARAJAN, 2012; CUBUKCU et al., 2015; SINGH; YADAV, 2015; ALSHU'EILI; GUPTA; MUKHOPADHYAY, 2011).

Devido a grande aplicabilidade destes sistemas, diversas interfaces de programação (*Application Programming Interface* - API) foram implementadas no intuito de facilitar o desenvolvimento de sistemas de reconhecimento de voz para *softwares* e aplicações, tais como *Web speech*, *Java speech*, *Google cloud speech*, dentre outras (DEBATIN; HAENDCHEN; DAZZI, 2017).

Apesar de trabalharem com processamento de fala contínua ou um extenso vocabulário, estas interfaces possuem como desvantagem a não possibilidade de serem utilizadas sem internet, uma vez que o processamento acontecem em grandes servidores, daí o elevado índice de acurácia apresentado. Além disso, tem-se o fato que estes API's são proprietárias e sua utilização é dispendiosa, já que é dependente do número de requisições realizadas pelo API (DEBATIN; HAENDCHEN; DAZZI, 2017).

Dessa maneira, outras abordagens vêm sendo propostas por meio das utilização de redes neurais profundas. Estas redes tem alto poder de processamento e capacidade de generalização, podendo trabalhar com um grande banco de dados. Os resultados obtidos com esta abordagem apresentam soluções cada vez mais robustas para sistemas de reconhecimento de voz. Porém, a sua capacidade de reconhecimento só é relevante se um grande número de exemplos de treinamento for utilizado e o processador utilizado fica restrito a estações fixas de alto poder de processamento (BHOWMIK; CHOWDHURY;

MANDAL, 2018; ZHANG et al., 2018). A aquisição de um grande banco de dados de sinais de voz padronizado em uma dada língua é uma tarefa complicada e aplicações embarcadas são inviáveis com esta abordagem.

Então, observando aplicações voltadas para projetos embarcados, na qual o tamanho do vocabulário é restrito, não há a necessidade de muitos exemplos para o treinamento e podem ser utilizadas em um *hardware* DSP (Processador Digital de Sinais - *Digital Signal Processor*), as redes neurais convencionais apresentam-se como ótima ferramenta, conforme mostrado nos trabalhos (ROCHA; SILVA, 2016; **??**). Entretanto, observa-se a limitação destas redes neurais com a expansão do número de classes a ser reconhecida. Problemas com convergência do algoritmo no treinamento e baixo nível de acurácia das classes são encontrados.

Então, para contornar esta problemática foi proposto uma arquitetura que utiliza o poder discriminativo das redes neurais convencionais baseada no método de comitês, na qual a dificuldade encontrada por apenas uma estrutura de classificação ao aumento do número de classes do problema é superada pelo particionamento das classes entre um conjunto de redes neurais mais simples que são selecionadas para a classificação final por meio de funções de base radial gaussiana parametrizada com as características de cada classe do problema.

Assim, tem-se um sistema de reconhecimento de voz na língua portuguesa do Brasil que permite a expansão do vocabulário, sem a necessidade de um banco de dados extenso para garantir a generalização e que pode ser utilizado em sistemas embarcados.

1.4 Objetivos

1.4.1 Objetivo Geral

Desenvolvimento de um Sistema Automático de Reconhecimento de Comandos de Sinais de voz em Português.

1.4.2 Objetivos Específicos

1. Obter de banco de dados de sinais de voz composto de locutores do sexo feminino e do sexo masculino, na faixa etária de 18 a 50 anos.

2. Realizar o pré-processamento de amostras de sinal de voz compostas pelas locuções em português dos dígitos ‘0’, ‘1’, ‘2’, ‘3’, ‘4’, ‘5’, ‘6’, ‘7’, ‘8’, ‘9’ e de 20 comandos dados pelas palavras: “abaixo”, “abrir”, “acima”, “aumentar”, “ “desligar”, “diminuir”, “direita”, “esquerda”, “fechar”, “finalizar”, “iniciar”, “ligar”, “máximo”, “médio”, “mínimo”, “para trás”, “para frente”, “parar”, “repousar”, “salvar”.

3. Obter a matriz temporal bidimensional de baixa ordem por meio da aplicação da transformada cosseno discreta para a formação dos padrões de entrada do classificador;

4. Parametrizar do conjunto de funções de base radial gaussiana (FBRG) para transformação do espaço de característica dos padrões por meio do algoritmo k-means;

5. Especificar os elementos da topologia e algoritmos de aprendizado para o conjunto de Redes Neurais tanto na configuração MLP quanto LVQ;

6. Selecionar as Redes MLP e LVQ que apresentarem melhores desempenhos na fase de Treinamento e Validação para a realização de testes com novos padrões para verificação de acurácia local;

7. Verificar as regras de seleção dinâmica dos classificadores pelas FBRG e resultado final pelas redes neurais especialistas pré-definidas são válidas ao objetivo do sistema de reconhecimento proposto.

1.5 Organização do Trabalho

Para a compreensão gradual do tema abordado neste trabalho, o mesmo está estruturado da seguinte forma:

Capítulo 2: apresentam-se os principais aspectos relacionados ao sinal de voz, desde a sua natureza fisiológica até a fase de processamento e extração das características relevantes para o reconhecimento.

Capítulo 3: abordam-se as características das funções de base radial gaussianas, redes neurais, enfatizando as configurações de Rede Neural Multicamadas e a Quantização Vetorial por Aprendizagem e Método de Comitês que são os elementos-chaves da arquitetura hierarquizada proposta para o sistema de reconhecimento que será tratado neste trabalho.

Capítulo 4: descreve-se a metodologia utilizada para a obtenção dos resultados e apresenta-se a forma de aquisição do sinal de voz, o pré-processamento, codificação do sinal em coeficientes mel-cepstrais, geração da matriz bidimensional, mudança de dimensionalidade dos padrões de entrada por meio das FBRG, projeto dos conjuntos de redes especialistas MLP e LVQ.

Capítulo 5: demonstram-se os resultados obtidos durante o treinamento e validação das redes com as variações em seus elementos, análise de desempenho entre as configurações MLP e LVQ na arquitetura proposta, integração das redes especialistas com as FBRG e testes finais.

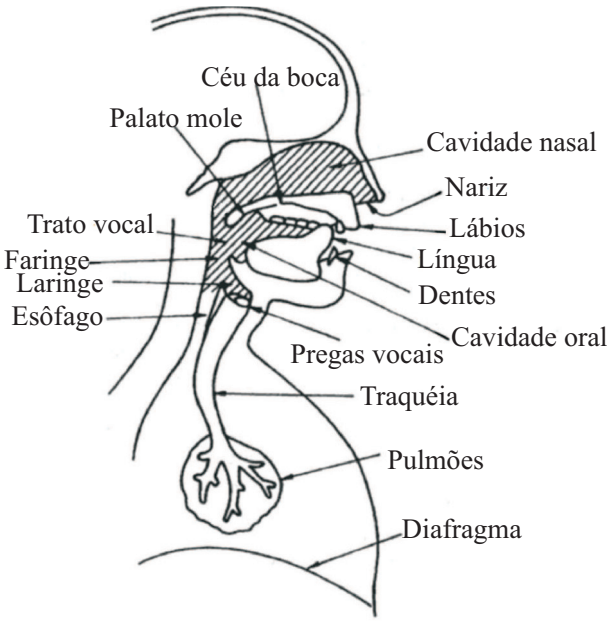
Capítulo 6: apontam-se as considerações finais e apresenta sugestões para trabalhos futuros.

2 Processamento do Sinal de Voz

2.1 Processo Fisiológico de Produção da Voz

O som da voz é uma onda de ar originária de ações complexas do corpo humano, suportadas por três unidades funcionais: geração de pressão de ar, regulação de vibração e controle de ressonadores. O aparelho fonador, ou seja, o conjunto de órgãos que fazem parte do processo de formação da voz no corpo humano é visualizado na Figura 2 (BENESTY; SONDHI; HUANG, 2007; FURUI, 2000).

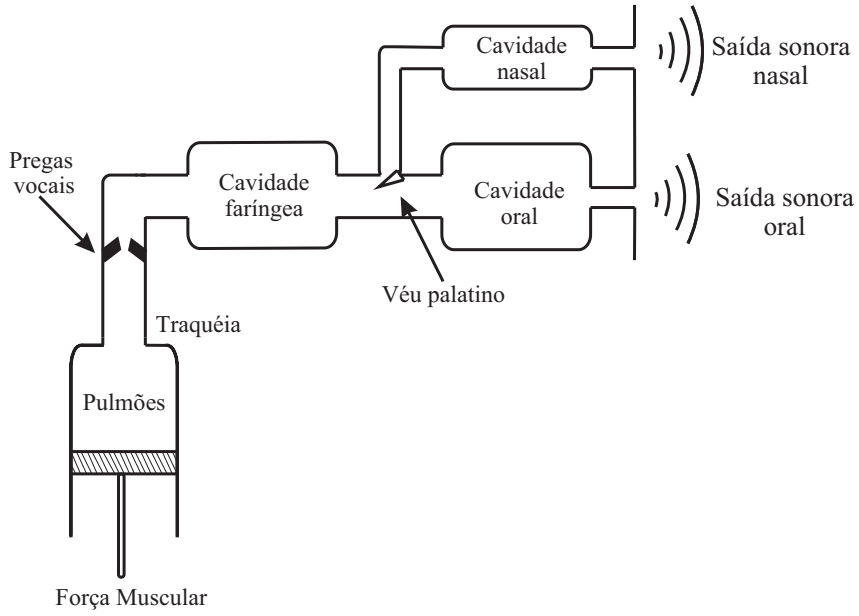
Figura 2 – Aparelho Fonador



Fonte: adaptado de Furui (2000, p. 9)

A produção da fala é normalmente inferida como uma operação de filtragem acústica, onde a anatomia humana, responsável pela produção da fala, é transformada em um modelo acústico de produção da voz. As três principais cavidades do sistema de produção da fala (cavidade faríngea, cavidade nasal e cavidade oral) compreendem o filtro acústico principal. Este filtro é excitado pelos órgãos abaixo dele e é alterado na sua saída principal pela impedância de radiação devido aos lábios. Os articuladores, a maioria dos quais estão associados ao seu próprio filtro, são usados para mudar as propriedades do sistema, a sua forma de excitação e sua saída a longo prazo. Na Figura 3 mostra-se o modelo acústico simplificado descrito (DELLER; HANSEN; PROAKIS, 2000; FLANAGAN, 2013).

Figura 3 – Diagrama de blocos da produção da voz humana



Fonte: adaptado de Deller, Hansen e Proakis (2000, p. 103)

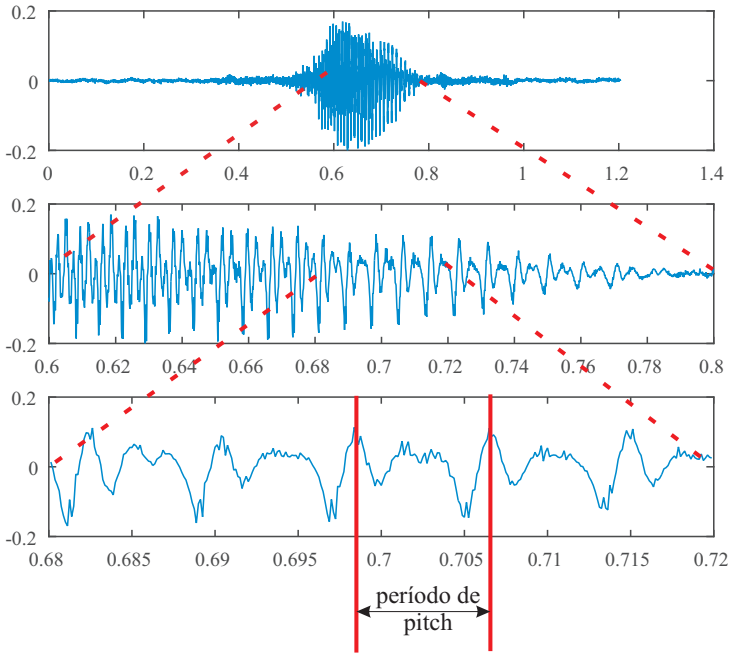
O processo natural de produção da fala inicia-se com a entrada de ar nos pulmões a partir do movimento dos músculos do diafragma. Quando o ar sai dos pulmões, o fluxo de ar passa através da traqueia e glote para a laringe. A glote ou o *gap* entre as pregas vocais direita e esquerda, que está normalmente aberta durante a respiração, torna-se estreita quando o locutor tenta produzir um som. Então, o fluxo de ar é periodicamente interrompido pela abertura e fechamento da glote de acordo com a interação entre o fluxo de ar e as pregas vocais (FURUI, 2000).

A passagem do fluxo de ar através das pregas vocais faz com que as mesmas vibrem durante um período de tempo para produzir o som, chamado de período fundamental e, consequentemente, tem-se a frequência de vibração fundamental F_0 ou *pitch*. Então, quando a pressão do ar originária dos pulmões é alta, o período de vibração das pregas vocais é pequeno, produzindo, assim, um som de alta frequência ou *pitch* alto. Já quando o fluxo de ar passa através das pregas vocais sob baixa pressão, o som produzido tem baixa frequência ou *pitch* baixo (FURUI, 2000; DELLER; HANSEN; PROAKIS, 2000; FURUI, 1991).

Os acentos e entonações resultam de variações da frequência fundamental. Um detalhe importante é que a frequência fundamental da voz é diferente para cada indivíduo. Isto se deve ao fato dos comprimentos das cordas vocais serem variáveis e ajustados pelo movimento relativo das cartilagens tireoide e cricóide, localizadas na laringe. Para falantes adultos, uma estimativa possível de F_0 para homens está entre 80-400 Hz e para as

mulheres, entre 120-800 Hz (BENESTY; SONDHI; HUANG, 2007). Observa-se na Figura 4 o pitch encontrado na pronúncia da palavra "'seis".

Figura 4 – Verificação de pitch na palavra "seis"



Estima-se, pela Figura 4, que a periodicidade existente na pronúncia da palavra "seis" tem a duração de $0,008$s. Dessa forma, $F_0 = 1/0,008 = 125Hz$. Sendo assim, este valor de F_0 indica que a palavra foi possivelmente falada por um locutor masculino.

A fonte sonora, constituída pela frequência fundamental e componentes harmônicas, é modificada pelo trato vocal para produzir os fonemas. Assim, pode-se classificar tanto os fonemas vocálicos quanto consonantais a partir destas modificações. Logo, tem-se a seguinte classificação (FURUI, 2000; RABINER; SCHAFER, 1978; FLANAGAN, 2013):

- Sons sonoros: são produzidas quando o tubo do trato vocal é excitado por pulso de ar sob alta pressão resultando na abertura e fechamento quase periódico do orifício glotal. Exemplo de sons sonoros: `/a/`, `/e/`, `/i/`, `/o/`, `/u/`, `/z/`, `/v/`, `/r/`, entre outros.
- Sons surdos: são produzidos pela criação de uma constrição em algum lugar do tubo do trato vocal e força o ar através da constrição, criando assim um fluxo de ar turbulento que atua como uma excitação de ruído aleatório no tubo do trato vocal. Exemplo: `/S/`- pronunciado como "sh", `/s/`.
- Fricativos Sonoros: Ocorre quando o trato vocal é parcialmente fechado causando um duplo fluxo turbulento para a constrição, ao mesmo tempo permitindo um duplo fluxo quase periódico para vibração das cordas vocais. Exemplos: `/z/`, `/v/`

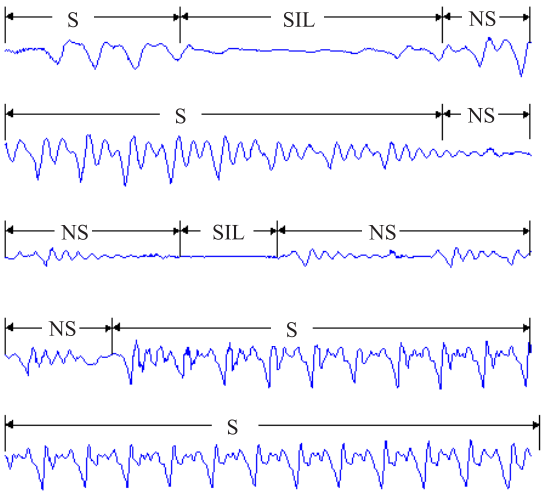
- Plosivos Sonoros: forma-se pelo fechamento momentâneo do fluxo de ar, permitindo que a pressão seja acumulada atrás da oclusão e então a pressão é abruptamente liberada. Exemplos: `/p/`, `/b/`, `/t/`, `/d/`, `/k/`, `/g/`.

2.2 Representação da forma de onda do sinal de voz no domínio do tempo

Conforme visto, a voz é definida por uma sequência de sons, produzidos pelo estado das cordas vocais, bem como as posições, formas e tamanhos dos vários articuladores que, alterados ao longo do tempo, refletem o som a ser produzido. Assim, a forma de onda do sinal de voz pode ser considerada estacionária quando analisada sob um curto intervalo de tempo; porém, para longos períodos de tempo, o sinal de voz apresenta mudanças que refletem os diferentes sons que estão sendo falados. Há várias formas de classificar os eventos que acontecem na voz. O mais usual e simples é aquele que se baseia no estado da fonte de produção da voz: as cordas vocais.

Assim, convenciona-se o uso de três estados representativos: silêncio (SIL), surdos (não-sonoras) (NS) e o sonoro (S). Estes três estados do sinal de voz, no domínio do tempo, podem ser observados na Figura 5.

Figura 5 – Estados do sinal de voz



Fonte: adaptado de Rabiner e Juang (1993, p. 18)

O estado de silêncio (SIL) não há produção de voz; já os sons surdos (não-sonoras) (NS) significam que as cordas vocais não estão vibrando, então a forma de onda da voz é aperiódica ou de natureza aleatória; por fim o sonoro (S) é aquele gerado quando as

cordas vocais são tensionadas e então vibram periodicamente pela passagem do fluxo de ar, resultando em uma forma de onda da voz quase periódica (RABINER; JUANG, 1993).

A segmentação da forma de onda da voz em três regiões bem definidas, devido aos três estados, não é exata. Normalmente, é difícil distinguir um som surdo fraco de um silêncio; ou um fraco som sonoro de um som surdo ou mesmo um silêncio (RABINER; JUANG, 1993).

Para os sons sonoros, a frequência fundamental (F_0) da voz é a menor componente harmônica que está relacionada à frequência natural de vibração das cordas vocais (BENESTY; SONDHI; HUANG, 2007). As vogais são consideradas sons sonoros produzidos pela excitação de uma forma essencialmente fixa do trato vocal, com pulsos de ar quase periódicos, causando a vibração das cordas vocais. Devido a isto, pode-se perceber na forma de onda de um som vocálico uma periodicidade bem definida.

O modo no qual a área da seção transversal varia ao longo do trato vocal determina as frequências ressonantes do trato, as chamadas frequências formantes, e assim, o som que é produzido (RABINER; JUANG, 1993). Geralmente, para a formação das vogais, são necessários três formantes, que são chamados de primeiro, segundo e terceiro formante (F_1, F_2, F_3) (FURUI, 2000).

As consoantes são classificadas como aqueles sons que são produzidos pelos rápidos movimentos de contração dos órgãos articulatórios, gerando sons bastante instáveis que evoluem ao longo do tempo. Para os sons fricativos, uma forte contração do trato vocal causa um ruído de fricção. Se as pregas vocais vibram ao mesmo tempo, as consoantes fricativas são então sonoras. Por outro lado, se as pregas vocais deixam o ar passar através delas sem produzir nenhum som, a fricativa é surda (MARIANI, 2013).

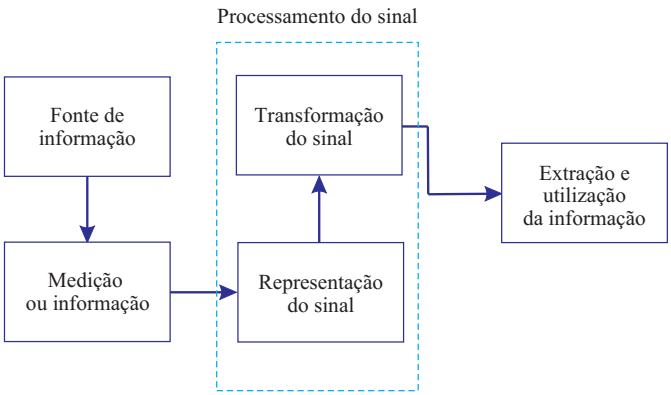
Os sons plosivos consonantais são obtidos pela obstrução completa do trato vocal, seguindo por uma fase de liberação. Se produzidos juntos com a vibração das cordas vocais, o plosivo é sonoro, caso contrário, ele é surdo (MARIANI, 2013).

Um som consonantal nasalizado é produzido se a cavidade nasal é aberta durante o fechamento da boca. As semivogais são consideradas sons consonantais sonoros, resultantes de um rápido movimento que brevemente passa através de uma posição de articulação de uma vogal (MARIANI, 2013).

2.3 Processamento do sinal de voz

O problema geral do processamento e manipulação da informação contida em um sinal é esquematizado no diagrama de blocos da Figura 6. Pode-se relacionar o diagrama de blocos da Figura 6 com a análise de sinais de voz, onde o locutor humano é a fonte da informação. Desta forma, a medição ou observação é dada sob a forma de onda acústica.

Figura 6 – Diagrama de blocos etapas de manipulação e processamento da informação



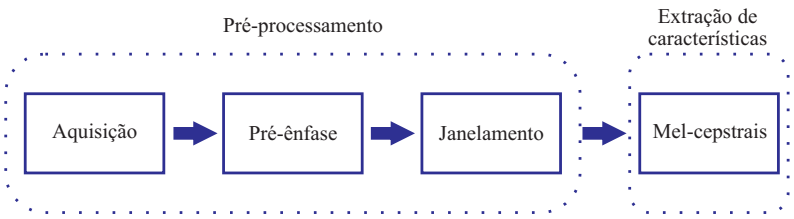
Fonte: adaptado de Rabiner e Schafer (1978, p. 3)

Após medir ou observar o sinal de voz, o processamento do mesmo envolve uma etapa que consiste em obter uma representação para este sinal a partir de um modelo e em seguida aplicar uma transformação com o objetivo de deixá-lo em uma forma mais conveniente. Por fim, tem-se a etapa de extração e utilização das características contidas no sinal de voz (RABINER; SCHAFER, 1978). Então, todas estas etapas podem ser realizadas por meio das técnicas de processamento digital de sinais, cujo avanço na área de processamento de sinais de voz possibilitou o desenvolvimento de sistemas mais robustos do que aqueles baseados em sistemas analógicos. Além disso, o desenvolvimento de hardwares digitais reforçaram as vantagens das técnicas de processamento digital sob os analógicos.

Assim, para aplicações em sistemas de reconhecimento de voz é necessário que sejam obtidos parâmetros do sinal de voz que representem o que está sendo pronunciado ou quem pronunciou, ou seja, o locutor. Dessa forma, a representação paramétrica baseia-se na caracterização do sinal como a saída de um modelo de produção da voz. Então, é feita uma representação digital da forma de onda, isto é, o sinal de voz é amostrado e quantizado e então é novamente processado para obter-se os parâmetros do modelo para a produção da voz. Desse modo, os parâmetros obtidos são convenientemente classificados como parâmetros de excitação ou como parâmetros de resposta do trato vocal (RABINER; SCHAFER, 1978).

Para obter-se a representação das características da voz, inicia-se o processo pela fase que se chama de pré-processamento do sinal de voz. Posterior ao pré-processamento, tem-se a fase de extração das características que representam o sinal de voz. A sequência de desenvolvimento destas fases e as subfases que cada uma compreende são representadas na Figura 7

Figura 7 – Fases para obter a representação do sinal de voz

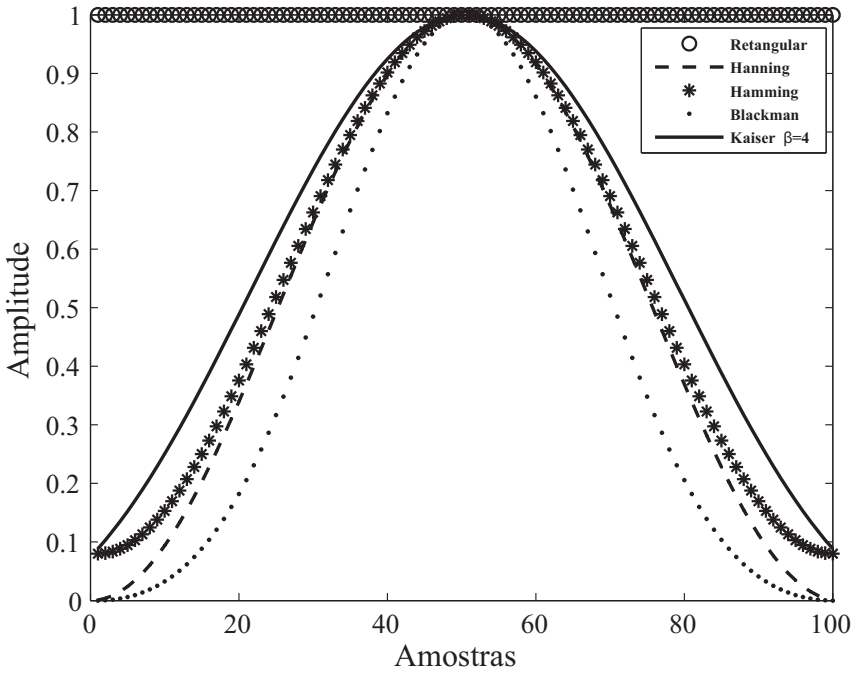


2.3.1 Pré-processamento o sinal de voz

2.3.1.1 Janelamento

Devido à variação naturalmente lenta do sinal de voz, é comum dividi-lo em segmentos, sobre as quais as propriedades da forma de onda da voz permanecem relativamente constantes (RABINER; JUANG, 1993). Assim, para obter-se a segmentação desejada do sinal original no domínio do tempo, faz-se a multiplicação por uma função chamada janela, que é uma sequência real e de tamanho finito. Algumas das janelas comumente usadas, como a janela retangular, a Hanning, Blackman, Kaiser e Hamming são as mostradas na Figura 8.

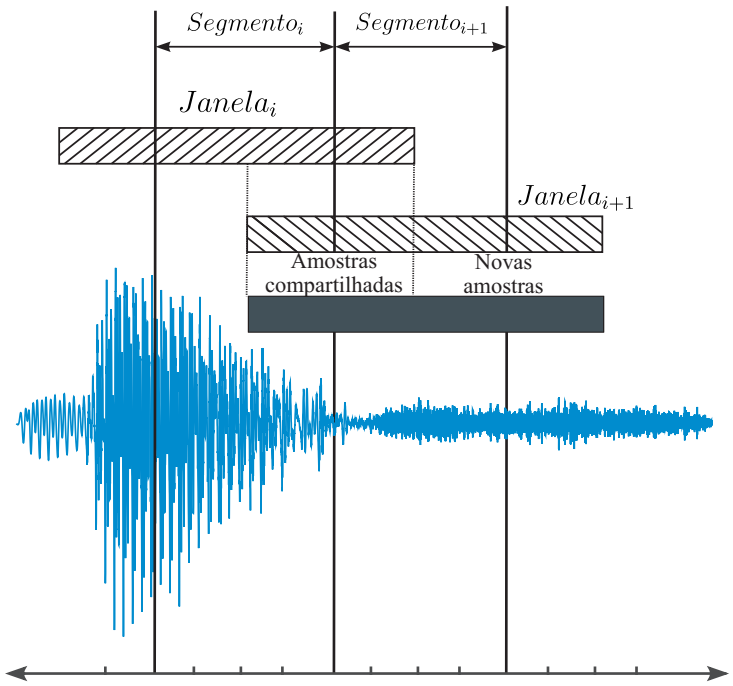
Figura 8 – Exemplos de janelas



Assume-se que a função janela é uma sequência causal de largura N. As janelas mais utilizadas são simétricas sobre o tempo $(N-1)/2$, que pode ser a metade entre dois pontos amostrados se N é par (DELLER; HANSEN; PROAKIS, 2000).

Qualquer que seja a janela utilizada, sua aplicação sobre o sinal é sempre acompanhada de sobreposições. As sobreposições entre as janelas têm como objetivo aumentar a correlação entre janelas sucessivas e evitar variações bruscas nas características extraídas entre janelas adjacentes (segmentos) e mudanças abruptas nas extremidades das janelas. Porém essa sobreposição aumenta o tempo de processamento (FURUI, 2000). A Figura 9 mostra a sobreposição de janelas no sinal de voz.

Figura 9 – Sobreposição de Janelas



Fonte: adaptado de Picone (1993a, p. 1220)

Em sistemas de reconhecimento de voz, a janela de Hamming é quase que exclusivamente utilizada (FURUI, 2000). Esta janela é um caso particular da janela de Hanning, que é definida pela equação geral dada pela equação (2.1):

$$w(n) = \frac{\alpha_w - (1 - \alpha_w)\cos\left(2\pi n/(N_s - 1)\right)}{\beta_w} \tag{2.1}$$

onde $0 < n < N - 1$ e N é o número de amostras que compõem o tamanho da janela. Para que a equação geral se torne uma janela de Hamming, é necessário que se faça α_w igual a 0.54. O valor para β_w é dado como uma constante de normalização de modo que a raiz do valor médio quadrático é unitário, definido pela equação (2.2) (Picone,1993):

$$\beta_w = \sqrt{\frac{1}{N_s}\sum_{n=0}^{N_s-1} w^2(n)} \tag{2.2}$$

A porcentagem de sobreposição entre janelas pode ser definida através da equação (2.3):

$$\%\text{Sobreposição} = \frac{(T_w - T_f)}{T_w} \times 100\% \tag{2.3}$$

onde T_w é a duração, em segundos, da janela e T_f é a duração do segmento. Se $T_w < T_f$, não se tem sobreposição (Picone,1993).

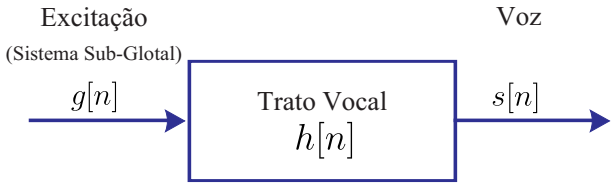
2.3.2 Extração das características do sinal de voz

2.3.2.1 Sistemas Homomórficos e Coeficientes Cepstrais

Sistemas homomórficos são uma classe de sistemas não-lineares que obedecem a um princípio generalizado de superposição. Desde a apresentação deste conceito por Oppenheim e Schafer (1968), as técnicas de processamento de sinal homomórficos têm sido de grande interesse na área de reconhecimento de voz.

O objetivo para o processamento homomórfico em análise de voz é resumido na Figura 10.

Figura 10 – Processamento homomórfico em análise de voz



Fonte: adaptado de Picone (1993a, p. 1225)

Assim, a partir da Figura 10, observa-se que os sistemas homomórficos oferecem uma metodologia bastante útil para o processamento de voz pelo fato de separar o sinal de excitação da função do trato vocal, uma vez que grande parte das abordagens para reconhecimento de voz baseiam-se nas características do modelo do trato vocal (PICONE, 1993a).

Desta forma, no modelo linear acústico de produção da voz, o espectro da voz, que pode ser obtido pela transformada de Fourier, consiste do sinal de excitação filtrado por um filtro linear variante no tempo representando a função de transferência do trato vocal. Logo, para separar estas duas componentes geralmente utiliza-se a convolução, que é descrita como a equação (2.4) (PICONE, 1993a).

$$s[n] = g[n] \otimes h[n] \tag{2.4}$$

onde $g[n]$ denota o sinal de excitação, $h[n]$ é a resposta impulsiva do trato vocal e $\otimes$ corresponde ao processo de convolução.

No domínio da frequência, esta representação do processo é dada pela equação (2.5).

$$S(f) = G(f) \cdot H(f) \tag{2.5}$$

Aplicando-se o logaritmo em ambos os lados da equação, tem-se a expressão dada pela equação (2.6):

$$\log\left(S(f)\right) = \log\left(G(f) \cdot V(f)\right) = \log\left(G(f)\right) + \log\left(H(f)\right) \tag{2.6}$$

Dessa forma, ao aplicar-se o logaritmo, a excitação e o filtro do trato vocal são superpostos e podem ser separados usando o processamento de sinal convencional, aplicando-se a transformada de Fourier inversa em $\log\left(S(f)\right)$. Define-se como cepstrum a transformada inversa de Fourier do logaritmo da magnitude espectral do sinal (RABINER; SCHAFER, 2007). A expressão para calcular o cepstrum é descrita pela equação (2.7).

$$c(n) = \frac{1}{N_s} \sum_{k=0}^{N_s-1} \log|S(k)| e^{j(2\pi/N_s)kn}, \quad 0 \leq n \leq N_s - 1 \tag{2.7}$$

onde, primeiro, se calcula o logaritmo da magnitude espectral $\log|S(k)|$ e posteriormente aplica-se a transformada inversa de Fourier (PICONE, 1993a).

O parâmetro para o cepstrum é chamado de *quefrency*, termo introduzido durante a elaboração do conceito de cepstrum, utilizado para descrever suas propriedades fundamentais. Dessa forma, baixas *quefrencies* correspondem as componentes do log da magnitude espectral que variam lentamente, enquanto que altas *quefrencies* correspondem as componentes do logaritmo da magnitude que variam rapidamente (BENESTY; SONDHI; HUANG, 2007).

A equação (2.7) apresenta o termo $c(0)$ que é o valor médio do log da magnitude do espectro, correspondendo, então, a uma medida de potência. Entretanto, verificou-se que este termo de potência absoluta não é confiável e entrou em desuso como coeficiente da sequencia cepstral (PICONE, 1993a).

A equação (2.7) pode ser convenientemente simplificada, observando que o espectro do log da magnitude é uma função real e simétrica, logo tem-se a expressão simplificada para a equação (2.7) dada na equação (2.8):

$$c(n) = \frac{2}{N_s} \sum_{k=0}^{N_s-1} S(I(k)) \cos\frac{2\pi}{N_s} kn \tag{2.8}$$

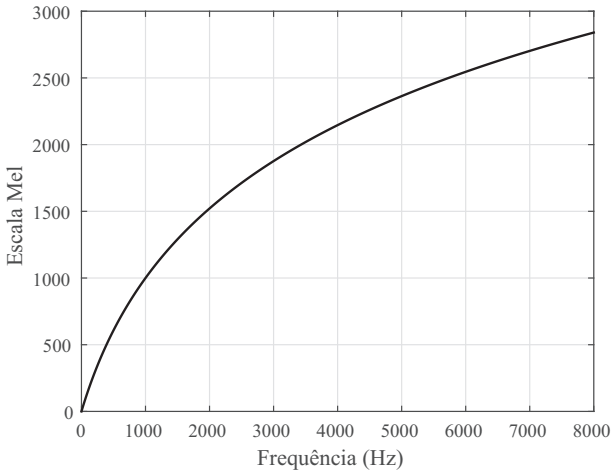
onde $c(n)$ é, normalmente, truncado para uma ordem muito menor que N (duração da janela em amostras). $I(k)$ é uma função de mapeamento que relaciona adequadamente o índice k com as amostras de $S(f)$.

2.3.2.2 Coeficientes Mel-Cepstrais

Existem outras variações da representação cepstral. Dentre ela, encontram-se os coeficientes mel-cepstrais (*mel frequency cepstral coefficients-mfcc*), formulada por Davis e Mermelstein (1980). Seus estudos basearam-se na psicoacústica, que mostra que a percepção das frequências de tons puros ou de sinais de voz não segue uma escala linear. Assim, uma escala que se aproxima desta percepção foi desenvolvida por Stevens (1940), a chamada escala mel.

O mel é a unidade de medida de um tom, isto é, de uma frequência única percebida pelo ouvinte. Como referência, definiu-se como 1000 mels a frequência de 1 kHz. Os outros valores da escala mel foram determinados a partir de experimentos, em que o ouvinte ajustava a frequência física de um tom até que a frequência percebida fosse igual a um múltiplo da frequência de referência. O gráfico da Figura 11 mostra a relação da escala mel com a frequência (STEVENS, 1940).

Figura 11 – Escala Mel



A escala mel é descrita conforme a equação (2.9) (PICONE, 1993a):

$$\text{frequência mel} = 2595 \log(1 + f/700.0) \tag{2.9}$$

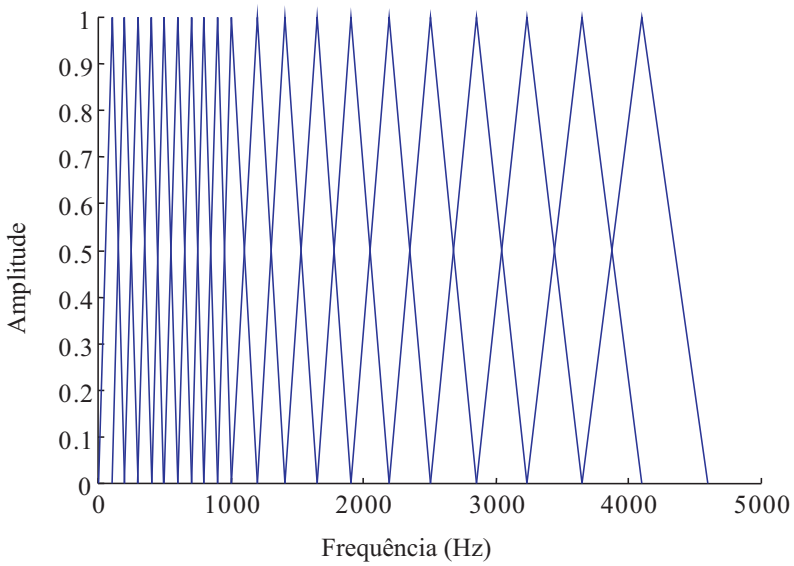
Outro conceito relacionado aos coeficientes mel-cepstrais é a banda crítica. Segundo os experimentos, a percepção de algumas frequências de sons complexos não é individualmente identificada dentro de certas bandas. Desta forma, quando um componente deste som deixa essa banda, chamada de banda crítica, este não pode ser identificado. O valor dessa banda crítica varia entre 10% a 20% da frequência central do som (JUANG; RABINER; WILPON, 1987; MENDOZA, 2009).

Logo, aliando-se o conceito da escala mel com a banda crítica, pode-se implementar um banco de filtros triangulares para calcular os *mfcc* dentro de uma faixa de frequência

de interesse (0 Hz – 4.6 kHz), em que cada filtro está centrado nas frequências da escala mel e igualmente espaçados na escala logarítmica. Normalmente, apenas as 20 primeiras amostras do banco de filtros são utilizadas (PICONE, 1993a; RABINER; SCHAFER, 2007).

Na Figura 12 mostra-se o banco de filtros utilizado para o cálculo dos coeficientes *mfcc*, desenvolvido com 20 filtros triangulares passa-banda, $f[i,k]$, onde $(0 \leq i \leq 20;\ 0 \leq k \leq 63)$ e sendo i o índice da banda e k um ponto em frequência na banda especificada sobre a faixa de frequência de interesse

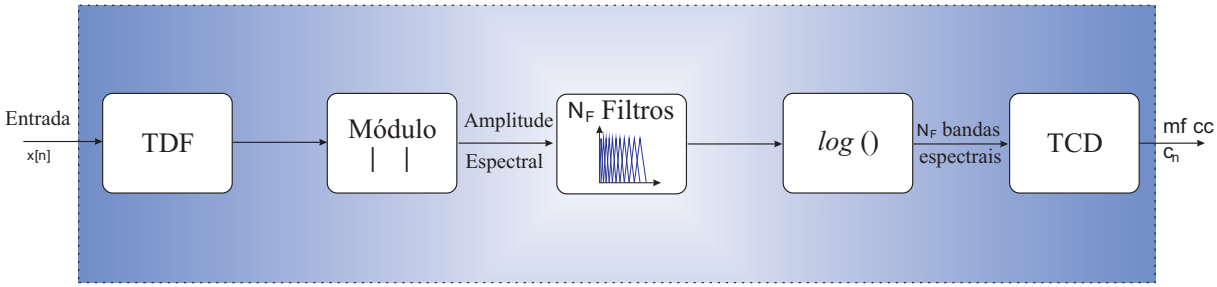
Figura 12 – Banco de filtros triangulares



Fonte: Silva (2015, p. 44)

Apresenta-se na Figura 13 um diagrama que representa as etapas necessárias para extrair os coeficientes mel-cepstrais:

Figura 13 – Diagrama de etapas para extração dos coeficientes *mfcc*



Os cálculos realizados para obtenção dos coeficientes *mfcc* são descritos através dos seguintes passos:

1. O sinal de voz, $s(t)$, passa pela etapa de aquisição, amostragem e pré-ênfase.

2. O sinal resultante, $s[n]$, é então dividido em M segmentos, tornando-se $s[m, n]$, representando a magnitude do n-ésimo ponto do m-ésimo segmento.

3. Para cada segmento é aplicado N janelas de Hamming, estimando-se, assim, o espectro $S[m, k]$ utilizando a Transformada de Fourier Discreta (TFD), dado pela equação (2.10):

$$S[m,k] = \sum_{n=0}^{N-1} s[m,n]e^{-j\frac{2\pi kn}{N}}, \quad 0 \leq k \leq N-1, \quad 0 \leq m \leq M-1 \tag{2.10}$$

   onde,
   $S[m, k]$: o espectro do k-ésimo ponto do m-ésimo segmento;
   N: a largura da janela utilizada;
   M: o número de segmentos do sinal de voz utilizado.

4. O módulo do espectro calculado é multiplicado pelos fatores de ponderação $f[i, k]$, e somam-se os produtos para todos os k's. Obtêm-se desta maneira a energia $E[m, i]$ para cada banda i de frequência do m-ésimo segmento, conforme equação (2.11):

$$E[m,i] = \sum_{k=0}^{N-1} |S[m,k]|f[i,k], \quad 0 \leq m \leq M-1, \quad 1 \leq i \leq 20 \tag{2.11}$$

   onde,
   i é o índice da banda do filtro;
   k é o índice do espectro;
   m é o número do segmento analisado;
   M é o número total de segmentos para análise.

5. Calcula-se então, a energia total na i-ésima banda como definido na equação (2.12):

$$E[i] = \sum_{m=0}^{M-1} |E[m,i]| \tag{2.12}$$

6. Por fim, os coeficientes mel-ceptrais, $mfcc[k]$ são então obtidos através da expressão dada por (2.13):

$$\text{mfcc}[k] = \sum_{i=1}^{N_F} E[i] \cos\left[\frac{i(k-0.5)\pi}{N_F}\right] \tag{2.13}$$

   sendo $k = 1, 2, \cdots, K$ o número de coeficientes mel-cepstrais, N_F o número de filtros utilizados e $E[i]$ é a saída log energia da i-ésima banda.

Portanto, após a obtenção dos parâmetros que caracterizam o sinal de voz através dos coeficientes mel-cepstrais, pode-se utilizá-los como entrada de um sistema reconhecedor cuja arquitetura esta baseada no método de comitês.

No próximo capítulo serão apresentados os elementos constituintes da arquitetura de classificação dos padrões, que utiliza funções de base radial gaussiana e um conjunto de redes neurais especialistas para formar um sistema de reconhecimento de voz de alto desempenho.

3 Sistemas Inteligentes baseados em Método de Comitês

3.1 Funções de Base Radial

As funções de base radial são importantes ferramentas na modelagem de tarefas de classificação e predição. Essas funções compreendem uma classe particular de funções que possuem uma resposta crescente ou decrescem monotonicamente com a distância a origem ou a um ponto central, tal que, $\phi(\mathbf{x}) = \phi(\|\mathbf{x}\|)$ ou $\phi(\mathbf{x}, \mathbf{c}) = \phi(\|\mathbf{x} - \mathbf{c}\|)$, respectivamente. Em geral, a norma utilizada em funções de base radial é a distância euclidiana, porém outras funções de distância podem ser empregadas (BUHMANN, 2003).

Matematicamente, uma função $\phi : \mathbb{R}^S \to \mathbb{R}$ é dita radial se existir uma função univariada, $\phi : [0, \infty) \to \mathbb{R}$ tal que $\phi(\mathbf{x}) = \varphi(r)$, onde $r = \|\mathbf{x} - \mathbf{c}\|$ e $\|\cdot\|$ é alguma norma em $\mathbb{R}^S$; geralmente, usa-se a norma Euclidiana (JAYASUMANA et al., 2015).

Existem alguns tipos de funções radiais, tais como:

- Multiquadráticas: $\phi(r) = (r^2 + \sigma^2)^{1/2}, c > 0$;
- Multiquadráticas Inversa: $\phi(r) = \frac{1}{(r^2+\sigma^2)^{1/2}}, c > 0$
- Função Gaussiana: $\phi(r) = e^{\frac{r^2}{2\sigma^2}}$

Dentre as funções radiais, a mais empregada é a função gaussiana (BISHOP, 1995; HAYKIN, 2009). Nesta função, o parâmetro $\mathbf{c}$ define o centro da gaussiana e σ^2 representa a variância dessa função, que caracteriza o alargamento da base da curva e indica o quão disperso está um vetor $\mathbf{x}$ em análise em relação ao centro $\mathbf{c}$, também chamado de campo receptivo. Para o caso do vetor $\mathbf{x}$ multivariado, a função de base radial gaussiana utiliza uma matriz de covariância Σ e a formulação para a função radial gaussiana multivariada é dada por (3.1):

$$\phi(\mathbf{x}) = exp\left\{-\frac{1}{2}\left(\mathbf{x} - \mathbf{c}^T\Sigma^{-1}\mathbf{x} - \mathbf{c}\right)\right\} \tag{3.1}$$

A matriz de covariância Σ pode ser definida de três maneiras, onde a escolha de uma delas determina como será a forma, o tamanho e a direção do campo receptivo (HAYKIN, 2009):

- $\mathbf{\Sigma} = \sigma^{\mathbf{2}}\mathbf{I}$, onde $\mathbf{I}$ é uma matriz identidade e σ^2 é uma variância comum. Neste caso, o campo receptivo da função é dado por uma hiperesfera com centro em c e raio σ;

- $\mathbf{\Sigma} = diag(\sigma_1^2, \sigma_1^2, \cdots, \sigma_n^2)$, onde σ_j^2 é a variância do j-ésimo elemento do vetor $\mathbf{x}$. O campo receptivo toma a forma de hiperelipse, cujos eixos individuais consistem com aqueles do espaço de entrada e com sua extensão ao longo do j-ésimo eixo sendo determinado por σ_j.

- $\mathbf{\Sigma}$ não é uma matriz diagonal. Por definição, Σ é uma matriz definida positiva. Então, pode-se fazer uma decomposição de $\mathbf{\Sigma}$ como: $\mathbf{Q}^T\mathbf{\Lambda}\mathbf{Q}$, onde $\mathbf{\Lambda}$ é uma matriz diagonal e $\mathbf{Q}$ é uma matriz ortonormal de rotação. A matriz $\mathbf{\Lambda}$ determina a forma e tamanho do campo receptivo enquanto que $\mathbf{Q}$ determina a direção.

Dessa forma, estes parâmetros precisam ser definidos para caracterizar uma função gaussiana. Estes parâmetros podem ser obtidos a partir dos dados que definem o problema a ser modelado (BUHMANN, 2003; GHOSH; NAG, 2001; DACHAPAK et al., 2004).

Então, ao verificar a resposta da função gaussiana, percebe-se que a resposta produzida pelo campo receptivo radial dessa função será mais significante quanto mais próximo um dado vetor $\mathbf{x}$ estiver do centro da gaussiana. Pelas características do modelo gaussiano, essa resposta também pode ser considerada como uma distribuição de probabilidade de um dado vetor $\mathbf{x}$.

As funções de base radial podem ser utilizadas para fazer o mapeamento não linear entre dois espaços de características. Assim, em problemas de classificação de padrões, por exemplo, dado um conjunto $\mathbf{\Omega}$ de m padrões, $\mathbf{\Omega} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$ de dimensão m_0, onde cada um desses vetores é atribuído a uma de duas classes, γ_1 e γ_2. Caso estes padrões não possam ser separados linearmente no espaço dimensional original, pode-se utilizar um conjunto de funções de bases radiais para fazer o mapeamento em um espaço que permita essa separação.

Então, para cada padrão $\mathbf{x}_m$ do conjunto Ω é definido um novo vetor Φ, onde cada elemento é dado pela resposta do conjunto de funções base radial $\phi_i(\mu_i, \sigma_i^2)|i = 1, 2, \ldots, m$, aplicado ao vetor $\mathbf{x}_m$, tendo-se assim $\Phi(\mathbf{x}_m) = [\phi_1(\mathbf{x}_m), \phi_2(\mathbf{x}_m), \cdots, \phi_m(\mathbf{x}_m)]^T$. Portanto, o vetor $\Phi(\mathbf{x}_m)$ mapeia os vetores de um espaço de entrada m_0 dimensional em um novo espaço de dimensão m_1.

Para a classificação de padrões complexos, o aumento do número de funções de base radial criará um espaço de alta dimensionalidade, o que o aumenta a probabilidade de separação linear desses dados neste novo espaço, tornando o problema de classificação bem mais simples.

Essa propriedade está apoiada no teorema de separabilidade de padrões de Cover (1965) que diz que um problema de classificação de padrões que está inserido num espaço de alta dimensão é mais provável ser linearmente separável do que em espaço de baixa dimensão (COVER, 1965; HAYKIN, 2009).

3.2 Introdução às Redes Neurais

As redes neurais artificiais (RNAs) são sistemas cuja estrutura computacional baseia-se na forma como o cérebro humano processa as informações do ambiente. Também conhecida como modelo conexionista ou processamento paralelo distribuído, as RNAs surgiram depois da apresentação do neurônio simplificado por McCulloch e Pitts em 1943 (MCCULLOCH; PITTS, 1943).

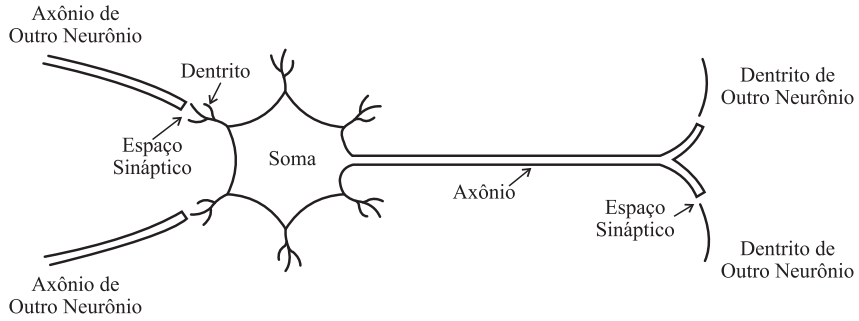
As RNAs são constituídas por sistemas paralelos distribuídos compostos por unidades de processamento simples (neurônios) que calculam determinadas funções matemáticas (normalmente não-lineares).

Estas unidades são dispostas em uma ou mais camadas e interligadas por um grande número de conexões, geralmente unidirecionais. Na maioria dos modelos, estas conexões estão associadas a pesos, que armazenam o conhecimento representado no modelo e ponderam a informação recebida na entrada de cada neurônio da rede (BRAGA, 2007).

Toda a estruturação das RNAs foi motivada na tentativa de reproduzir as funções das redes biológicas, buscando desenvolver seu comportamento básico e sua dinâmica (ANDERSON, 1995). Devido a esta analogia, as RNAs tem um potencial elevado em diversas áreas de aplicação.

Apresenta-se na Figura 14 um neurônio biológico genérico no qual as unidades de processamento artificiais das RNAs fazem referência.

Figura 14 – Neurônio biológico genérico



Fonte: adaptado de Fausett (1994, p. 6)

Logo, muitas das características de processamento dos elementos da RNA se assemelham às propriedades dos neurônios biológicos, a saber (FAUSETT, 1994):

- Os elementos de processamento recebem muitos sinais;
- Os sinais de entrada podem ser modificados pelo peso da sinapse receptora;

- Os elementos de processamento fazem uma soma ponderada das entradas;
- Em circunstâncias apropriadas (entradas suficientes), o neurônio transmite uma única saída;
- A saída de um neurônio em particular pode ir para muitos outros neurônios (as ramificações de axônios).

Outras características semelhantes das RNAs ao cérebro humano são quanto ao processo de aprendizagem, em que o conhecimento é adquirido a partir de seu ambiente e quanto às forças de conexão entre os neurônios ou pesos sinápticos, que armazenam o conhecimento adquirido (FAUSETT, 1994; HAYKIN, 2001). Apesar de não se ter conhecimento em sua totalidade do funcionamento das redes biológicas, estas similaridades entre os dois sistemas, permitem que as RNAs reproduzam com fidelidade várias funções somente encontrada nos seres humanos (BRAGA, 2007).

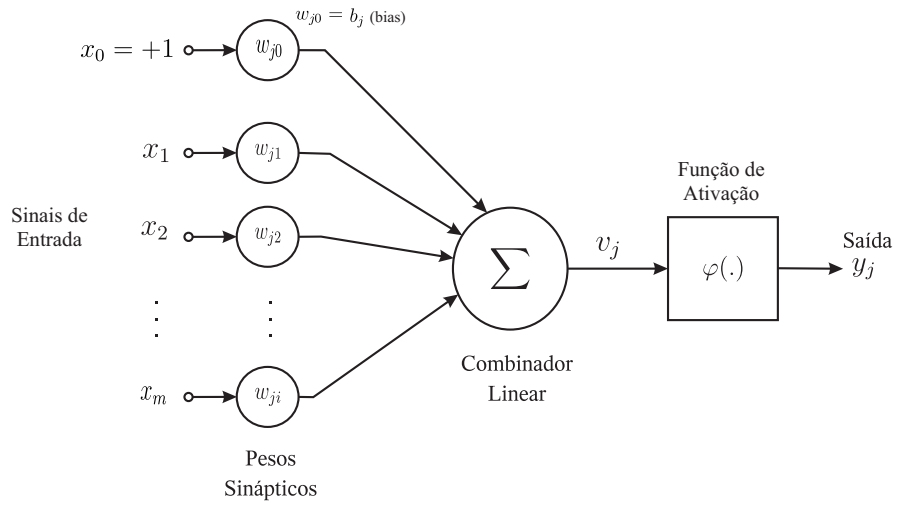
Dentre os atrativos para a utilização das RNAs em soluções de problemas, os principais são: sua capacidade de aprender através de exemplos apresentados a ela e a de generalizar a informação aprendida. Outras características que potencializam ainda mais seu uso podem ser listadas:

- Possibilidade de considerar o comportamento não-linear dos fenômenos físicos responsáveis pela geração dos dados de entrada;
- Necessidade de pouco conhecimento estatístico sobre o ambiente no qual a rede está inserida;
- Capacidade de aprendizagem, que é alcançada por meio de uma sessão de treinamento com exemplos entrada/saída que sejam representativos do ambiente ou somente pela categorização que a RNA faz internamente a partir das informações de entrada;
- Habilidade de aproximar qualquer mapeamento entrada/saída de natureza contínua;
- Adaptabilidade;
- Generalização;
- Conhecimento representado pela própria estrutura da RNA e pelo seu estado de ativação;
- Possibilidade de implementar em integração em escala muito ampla-VLSI(*Very-Large-Scale-Integration*).

3.2.1 Modelo do Neurônio Artificial

Representa-se na Figura 15 o modelo básico para a unidade de processamento que compõe a estrutura de uma rede neural.

Figura 15 – Modelo não-linear de um neurônio artificial com introdução do *bias*



Fonte: adaptado de HAYKIN (2001, p. 38)

Como pode ser visualizado no modelo da Figura 15, o neurônio artificial é composto por um conjunto de sinapses, onde cada uma possui um valor de peso específico. Assim, um sinal x_i na entrada da sinapse i conectada ao neurônio j é multiplicado pelo peso sináptico w_{ji}. O peso sináptico pode ter valor tanto positivo quanto negativo, o que não acontece com as sinapses do cérebro (HAYKIN, 2001).

Após os sinais de entrada serem ponderados pelos pesos das sinapses, um combinador linear faz o somatório destas entradas e adiciona a polarização externa ou *bias*. Esta soma ponderada é chamada de potencial de ativação v_j. O potencial de ativação é utilizado como parâmetro de uma função que pode ser linear ou não-linear, chamada de função de ativação $\varphi(.)$. Esta função limita dentro de um intervalo finito a amplitude da saída do neurônio, y_j. Tipicamente, a amplitude de saída do neurônio é normalizada, restringindo-se ao intervalo unitário fechado $[0, 1]$ ou $[-1, 1]$ (ROJAS; FELDMAN, 2013).

O efeito causado por esta polarização externa é aumentar ou diminuir o argumento da função de ativação, caso seja positivo ou negativo, respectivamente.

Em termos matemáticos, a saída do combinador linear e a saída do neurônio com a introdução do bias são dadas pela equação (3.2) e pela equação (3.3):

$$v_j = \sum_{i=1}^{m} w_{ji} x_i + bj \tag{3.2}$$

$$y_j = \varphi(v_j) \tag{3.3}$$

onde,

$x_1, x_2, ..., x_i$ são os sinais de entrada;
$w_{j1}, w_{j2}, ..., w_{ji}$ são os pesos sinápticos do neurônio j;
v_j é a saída do combinador linear devida aos sinais de entrada;
b_j é a polarização ou *bias*;
$\varphi(.)$ é a função de ativação;
y_j é o sinal de saída do neurônio.

A saída de neurônio é dada pela resposta de uma função ao potencial de ativação. Essa resposta não necessariamente apresenta os valores 0 ou 1 e está de acordo com o tipo de função de ativação. Dentre as funções de ativação mais utilizadas nos modelos de neurônios podem ser listadas:

1. **Função Linear ou função identidade:** produz resultados de saída idênticos aos valores do potencial de ativação v_j, cuja expressão matemática é dada pela equação (3.4):

$$\varphi(v_j) = v_j \tag{3.4}$$

2. **Função rampa simétrica:** os valores de resposta são iguais aos próprios valores dos potenciais de ativação quando estes estão definidos no intervalo $[-a, a]$, restringindo-se aos valores limites em caso contrário. A formulação matemática é dada por (3.5):

$$\varphi(v_j) = \begin{cases} a, & \text{se } v_j > a \\ v_j, & \text{se } -a \leq v_j \leq a \\ -a, & \text{se } v_j < a \end{cases} \tag{3.5}$$

3. **Função logística:** a saída da função logística assume sempre valores reais no intervalo $[0, 1]$, tendo como expressão matemática a equação (3.6):

$$\varphi(v_j) = \frac{1}{1 + e^{-\beta v_j}} \tag{3.6}$$

onde β é uma constante real associada ao nível de inclinação da função logística frente ao seu ponto de inflexão. A função logística é totalmente diferenciável em todo o seu domínio de definição.

4. **Função tangente hiperbólica:** a saída da função tangente hiperbólica assume sempre valores reais no intervalo $[-1, 1]$, tendo como expressão matemática a equa-

ção (3.7):

$$\varphi(v_j) = \frac{1 - e^{-\beta v_j}}{1 + e^{-\beta v_j}} \tag{3.7}$$

onde β é uma constante real associada ao nível de inclinação da função tangente hiperbólica frente ao seu ponto de inflexão.

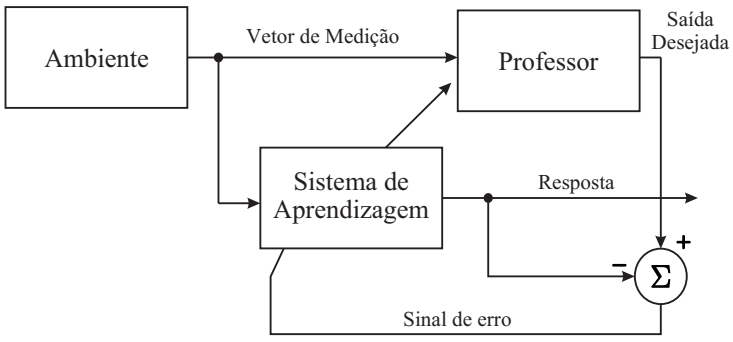
3.2.2 Formas de Aprendizado de uma RNA

Uma das características mais notáveis de uma RNA é a capacidade de aprender a partir de estímulos provenientes do ambiente em que está inserida e interpolar e extrapolar este aprendizado. No aprendizado conexionista, que é o caso das RNAs, não se procura obter regras como na abordagem simbólica da Inteligência Artificial (AI), mas sim determinar a intensidade de conexões entre os neurônios. O problema do aprendizado é justamente encontrar, através de um processo interativo, valores adequados das conexões entre os neurônios (parâmetros livres) que garantam o maior desempenho de aprendizado do processo do qual as informações apresentadas à rede estão inseridas. Portanto, como esta adequação dos parâmetros livres é feita de forma interativa, o aprendizado deve ocorrer pela rede de forma gradual (BRAGA, 2007).

Este processo de aprendizado interativo ocorre através de um algoritmo de aprendizagem composto por um conjunto de regras bem definidas e que pode atuar sob dois paradigmas principais: aprendizado supervisionado e aprendizado não supervisionado.

A metodologia de aprendizado supervisionado, a mais comum no treinamento das RNAs, as entradas correspondentes ao processo a ser aprendido são apresentadas à rede, juntamente com a saída desejada para as mesmas. Logo, o objetivo é ajustar os parâmetros da rede, a fim de encontrar uma ligação entre os pares de entrada e saída fornecidos (PRIDDY; KELLER, 2005). Ilustra-se na Figura 16 um diagrama de blocos que demonstra o processo de aprendizado supervisionado.

Figura 16 – Diagrama de blocos do processo de aprendizado supervisionado

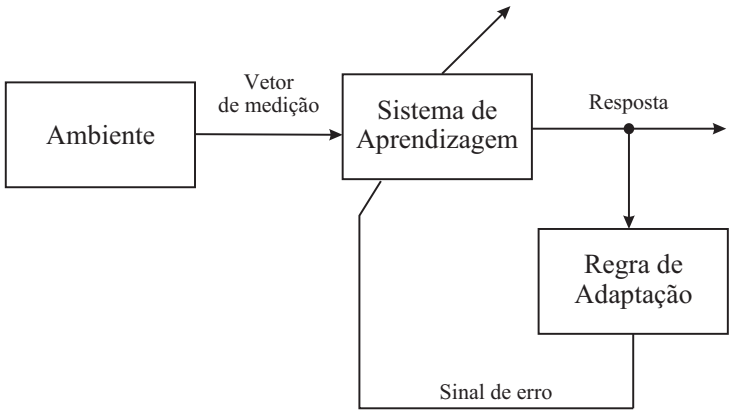


Fonte: Priddy e Keller (2005, p. 14)

Em relação ao aprendizado não-supervisionado ou auto-organizado, a rede neural não recebe informações referentes às saídas desejadas para o conjunto de dados de entrada do processo a ser aprendido (PRIDDY; KELLER, 2005). Dessa forma, a rede neural aprende somente pelas características regulares estatísticas presentes no conjunto de entradas do treinamento, desenvolvendo a habilidade de formar representações internas para codificar os dados de entrada, realizando agrupamentos que possuem características similares para os padrões de entrada (BRAGA, 2007; HAYKIN, 2009).

O diagrama de blocos apresentado na Figura 17 demonstra-se este método.

Figura 17 – Diagrama de blocos para aprendizado não-supervisionado



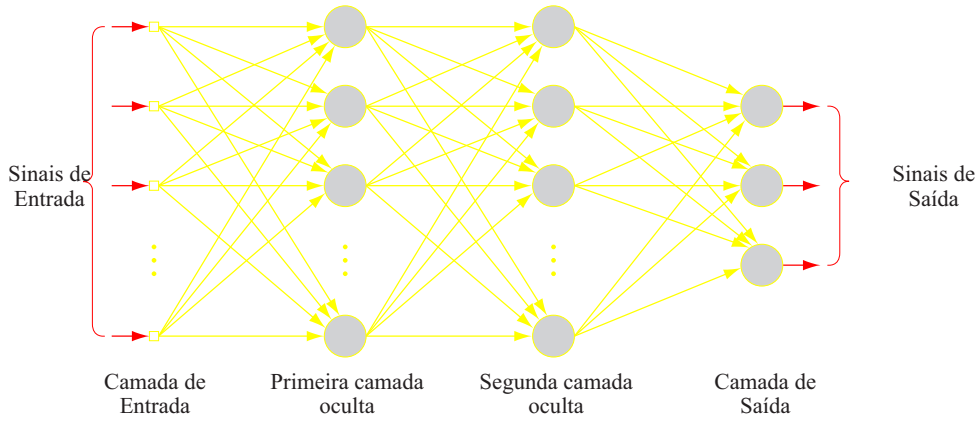
Fonte: Priddy e Keller (2005, p. 14)

3.2.3 Rede Perceptron de Múltiplas Camadas (*Multilayer Perceptron Network*-MLP)

As Redes Multilayer Perceptron são generalizações das Redes Perceptron, em que esta última utiliza somente uma camada de neurônios e encontra limitações para solucionar problemas que não são linearmente separáveis. Portanto, as Redes MLP, com o incremento de mais camadas de neurônios intermediárias, elimina a limitação das Redes Perceptron (MEHROTRA; MOHAN; RANKA, 1997; ANDERSON, 1995).

A aplicação com sucesso em problemas de reconhecimento de padrões, controle e processamento de sinais, fez com que a Rede MLP se tornasse uma rede de uso clássico em diversas áreas (MEHROTRA; MOHAN; RANKA, 1997; SILVA; SPATTI; FLAUZINO, 2010). Pode-se visualizar na Figura 18 que a arquitetura de uma Rede MLP é constituída por um conjunto de nós fontes, que recebem os dados de entrada, formando a camada de entrada; uma ou mais camadas intermediárias ou ocultas e uma camada de saída. Somente a camada de entrada não é constituída de neurônios, por isso, somente as camadas ocultas e de saída possuem capacidade computacional (BRAGA, 2007).

Figura 18 – Arquitetura de uma Rede MLP com duas camadas ocultas



Fonte: adaptado de Haykin (2009, p. 124)

Em um projeto de uma Rede MLP, o número de nós fonte é determinado pela dimensão dos padrões do conjunto de observação e o número de neurônios da camada de saída é dado pela dimensionalidade da resposta desejada. Deste modo, para um projeto completo é preciso determinar os seguintes itens (BRAGA, 2007; HAYKIN, 2001):

- O número de camadas escondidas;
- O número de neurônios em cada uma das camadas escondidas;
- Os pesos sinápticos que interconectam os neurônios das diferentes camadas da rede.

Não há regras determinadas para a especificação topológica dos dois primeiros itens de projeto citados. Isto se deve ao fato de que o número de camadas ocultas e o número de neurônios em cada camada dependem, entre outros fatores, do algoritmo de aprendizado utilizado, da forma como as matrizes de pesos são inicializadas, da complexidade do problema a ser mapeado, da disposição espacial das amostras e da qualidade do conjunto de treinamento disponível (SILVA; SPATTI; FLAUZINO, 2010).

O último item refere-se aos algoritmos de treinamento. Uma Rede MLP extrai as características dos dados de entrada a partir de um processo de aprendizagem supervisionado. Dessa forma, o algoritmo de aprendizagem mais conhecido para treinar Redes MLP é o algoritmo *backpropagation* ou algoritmo de retro-propagação do erro (RUMELHART; HINTON; WILLIAMS, 1986). Os outros algoritmos existentes são variações do *backpropagation*. A denominação *backpropagation* será utilizada ao longo deste trabalho para se referir ao algoritmo de aprendizagem da Rede MLP.

O algoritmo *backpropagation* é uma generalização da regra Delta e baseia-se na heurística do aprendizado pela correção do erro, em que o erro na saída da rede é retro-

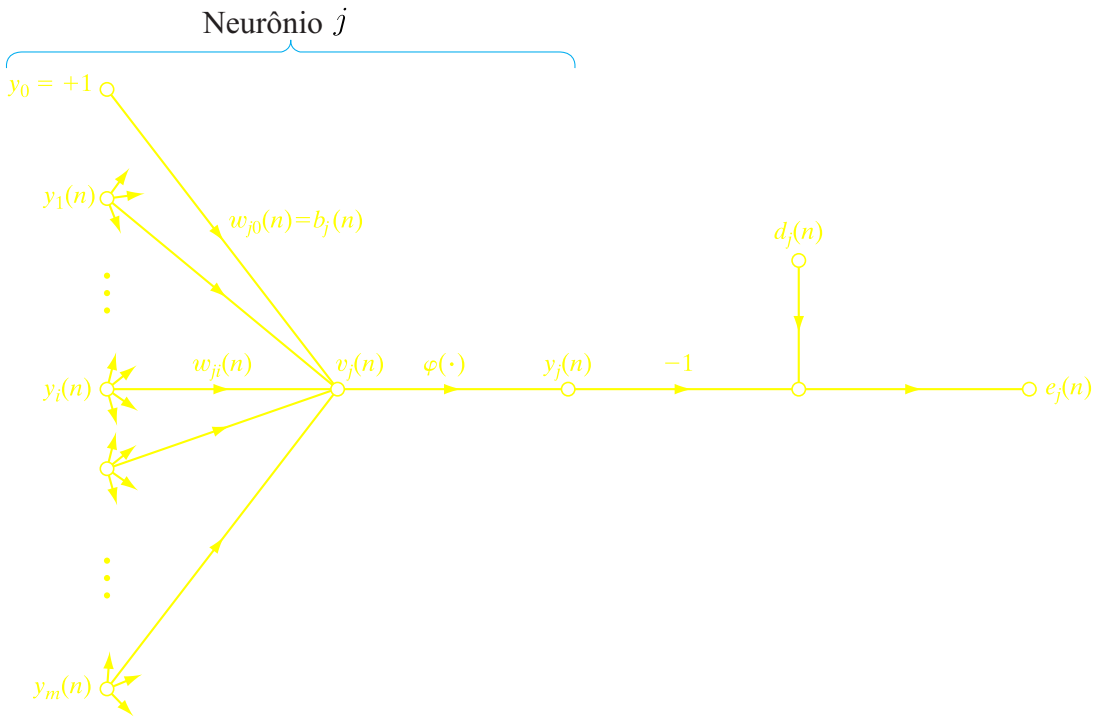
propagado para as camadas intermediárias da RNA (GURNEY, 2003). Dessa forma, o *backpropagation* constitui-se basicamente na aplicação sucessiva de duas fases bem definidas que serão explicadas na subseção seguinte.

3.2.3.1 Algoritmo de Retropropagação ou *BackPropagation*

A fase de propagação adiante é o passo inicial do algoritmo *backpropagation*. Nesta fase, as informações provenientes dos nós fonte propagam-se pela rede por meio dos sinais calculados em cada neurônio, que dependem da ponderação dos dados de entrada pelos pesos sinápticos, até a camada de saída, sem alterar os pesos sinápticos (RUMELHART; HINTON; WILLIAMS, 1986).

Dado o gráfico de fluxo de sinais representado na Figura 19, considera-se que o neurônio j é um neurônio pertencente à camada de saída e sua resposta é calculada através da equação (3.8):

Figura 19 – Grafo de fluxo de sinal no neurônio j



Fonte: adaptado de Haykin (2009, p. 129)

$$y_j(n) = \varphi(v_j(n)) \tag{3.8}$$

onde:

n é a n-ésima interação, passo de tempo;

y_j é a saída do neurônio j;

$\varphi(.)$ é a função de ativação do neurônio j;
$v_j(n)$ é o potencial de ativação do neurônio j.

O potencial de ativação $v_j(n)$ é dado pela expressão (3.9):

$$v_j(n) = \sum_{i=0}^{m} w_{ji}(n) y_i(n) \tag{3.9}$$

onde:
m é o número total de entradas aplicadas ao neurônio j;
$w_{ji}(n)$ é o peso sináptico que conecta a saída do neurônio i a entrada do neurônio j;
$y_i(n)$ é o sinal de saída do neurônio i.

Assim, com a resposta do neurônio de saída da rede, pode-se calcular o sinal de erro em relação ao valor da saída desejada, que será retropropagado camada a camada da rede. Logo, o sinal de erro é definido na equação (3.10):

$$e_j(n) = d_j(n) - y_j(n) \tag{3.10}$$

onde:
$e_j(n)$ é o sinal de erro na saída do neurônio j na n-ésima interação;
$d_j(n)$ é a resposta desejada para o neurônio j;

Para a camada de saída da rede é calculado o valor instantâneo $\mathscr{E}(n)$ da energia total do erro, somando-se o valor instantâneo da energia do erro de todos os neurônios da camada de saída. Logo, a expressão para o cálculo do erro total é dada por:

$$\mathscr{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \tag{3.11}$$

onde C inclui a quantidade de neurônios da camada de saída da rede. Definindo-se N como o número total de padrões a serem apresentados durante o treinamento, tem-se que a *energia média do erro quadrático* é determinada somando-se todos os $\mathscr{E}(n)$ de cada interação n e normalizando-se em relação ao tamanho do conjunto N, obtém-se a expressão (3.12):

$$\mathscr{E}_{\text{med}} = \frac{1}{N} \sum_{n=1}^{N} \mathscr{E}(n) \tag{3.12}$$

Constata-se que a energia instantânea do erro $\mathscr{E}(n)$, e consequentemente a energia média do erro $\mathscr{E}_{\text{med}}$, é uma função de todos os pesos sinápticos e *bias* da rede. Vale notar que o erro calculado para os neurônios de saída são os únicos que se pode obter diretamente pelo fato de serem os únicos neurônios acessíveis na rede. A função $\mathscr{E}_{\text{med}}$ caracteriza o processo de aprendizagem. O desempenho da rede passa pelo fato de ajustar

os parâmetros livres a fim de que o $\mathscr{E}_{\text{med}}$ seja mínimo. Logo, a função $\mathscr{E}_{\text{med}}$ é a função custo do sistema.

Portanto, durante o treinamento, os pesos são ajustados à medida que cada padrão é apresentado, sendo que ao término da apresentação do último elemento do conjunto de treinamento tem-se o que se chama de uma época. O ajuste é realizado com base no erro da saída para cada padrão individualmente. Ao final, faz-se uma média aritmética destas alterações individuais de peso para estimar a alteração real que resultaria caso os pesos fossem alterados baseados na minimização da função custo $\mathscr{E}_{\text{med}}$ sobre o conjunto de treinamento inteiro.

Baseado no algoritmo de mínimos quadrados (*Least Mean Square*-LMS), o algoritmo *backpropagation* faz a alteração no valor dos pesos a partir da aplicação da derivada parcial do erro total em relação aos pesos sinápticos do neurônio j. Deste modo, a aplicação do operador gradiente $\nabla E(W)$ em relação ao conjunto de pesos busca o valor ótimo que minimiza a função custo (WIDROW; HOFF, 1988). De acordo com a regra da cadeia do cálculo, chega-se a expressão para o gradiente:

$$\nabla E(W) = \frac{\partial \mathscr{E}(n)}{\partial w_{ji}(n)} = \frac{\partial \mathscr{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)} \tag{3.13}$$

O operador gradiente representa um fator de sensibilidade, que determina a direção de busca no espaço de pesos, para o peso sináptico w_{ji}. Então, deriva-se as expressões apresentadas nas equações anteriores (3.8), (3.9), (3.10), (3.11) em relação aos termos adequados, na qual a demonstração completa pode ser encontrada em (HAYKIN, 2009), obtêm-se a expressão para o operador gradiente, dado por 3.14:

$$\frac{\partial \mathscr{E}(n)}{\partial w_{ji}(n)} = -e_j(n)\varphi_j'(v_j(n))y_i(n) \tag{3.14}$$

O parâmetro de ajuste $\Delta w_{ji}(n)$ aplicada a w_{ji} é definido pela regra delta:

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathscr{E}(n)}{\partial w_{ji}(n)} \tag{3.15}$$

onde η é o parâmetro da taxa de aprendizagem do algoritmo *backpropagation*. O uso do sinal negativo na equação (3.15) indica que a variação dos pesos deve ser feita na direção oposta àquela do gradiente, a fim de buscar uma direção no espaço de pesos que reduza o valor de $\mathscr{E}(n)$. Logo, substituindo a equação (3.14) na equação (3.15), tem-se que:

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \tag{3.16}$$

onde o gradiente local $\delta_j(n)$ é definido por:

$$\begin{aligned} \delta_j(n) &= -\frac{\partial \mathscr{E}(n)}{\partial v_j(n)} \\ &= -\frac{\partial \mathscr{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \\ &= e_j(n)\varphi_j'(v_j(n)) \end{aligned} \tag{3.17}$$

Portanto, a expressão (3.17) implica que o gradiente local $\delta_j(n)$ para um dado neurônio de saída j é igual ao produto do erro associado a este neurônio pela derivada da função de ativação correspondente a este mesmo neurônio.

As equações de (3.8) a (3.17) apresentadas são calculadas de forma direta para o caso em que os neurônios pertencem à camada de saída, pois, como para estes neurônios tem-se uma resposta desejada, é fácil determinar o erro associado. Já para o caso dos neurônios ocultos, apesar de não se ter acesso direto aos mesmos, eles também são responsáveis pelos erros cometidos na rede. Entretanto, a dificuldade é como ponderar a influência de cada neurônio oculto no erro resultante.

Deste modo, para o caso de neurônios ocultos, o algoritmo *backpropagation* torna-se mais complexo, pelo fato de não se ter uma resposta desejada específica para estes neurônios. Então, o erro é determinado de forma recursiva em termos dos sinais de erro provenientes de todos os outros neurônios aos quais o neurônio oculto está diretamente conectado.

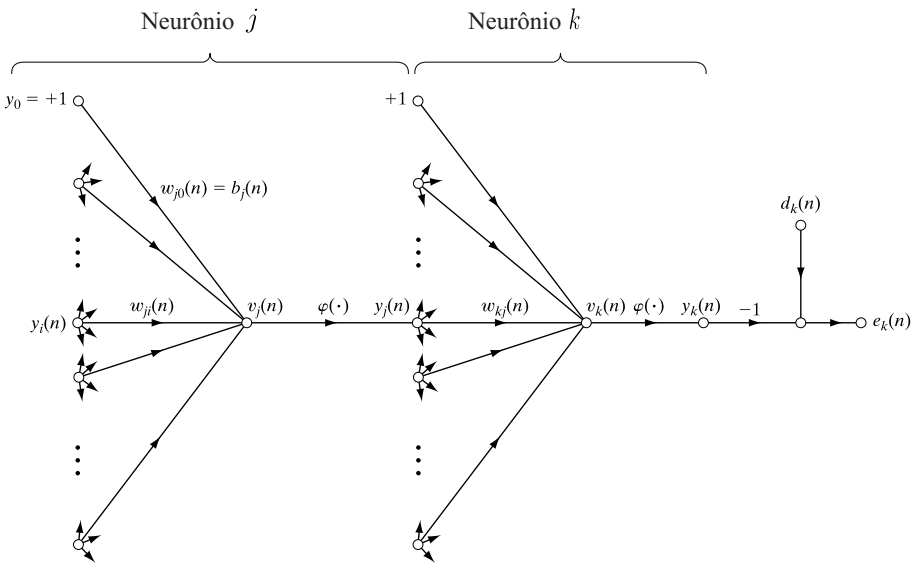
Exemplifica-se na Figura 20 o caso em que o neurônio j pertence a uma camada oculta da rede.

Para este caso, tem-se agora que o neurônio de saída é dado pelo índice k e antecedido pelo neurônio oculto j. Como definido pela equação (3.17), o gradiente local $\delta_j(n)$ para o caso do neurônio oculto j pode ser expresso por (3.18):

$$\begin{aligned} \delta_j(n) &= -\frac{\partial \mathscr{E}(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \\ &= -\frac{\partial \mathscr{E}(n)}{\partial y_j(n)} \varphi_j'(v_j(n)) \end{aligned} \tag{3.18}$$

Determina-se a derivada parcial do erro total em relação à saída do neurônio oculto j, partindo do erro obtido na saída do neurônio k. Conforme Figura 20, tem-se a equação (3.11), sendo que, para este caso, o erro instantâneo possui o índice k, já que o neurônio k é o neurônio de saída da rede. A partir da equação (3.11) modificada, o erro é diferenciado

Figura 20 – Grafo de fluxo de sinais o neurônio j pertencente a uma camada oculta



Fonte: adaptado de Haykin (2009, p. 132)

em relação ao sinal de saída do neurônio j, $y_j(n)$, obtendo a equação (3.19):

$$\frac{\partial \mathscr{E}(n)}{\partial y_j(n)} = \sum_k e_k \frac{\partial e_k(n)}{\partial y_j(n)} \tag{3.19}$$

Realizando as devidas demonstrações das derivadas parciais, conforme apresentado em Haykin (2009), obtém-se a fórmula do *backpropagation* para o gradiente local $\delta_j(n)$ de um neurônio oculto na equação (3.20):

$$\delta_j(n) = \varphi_j^{'}(v_j(n)) \sum_k \delta_k(n) w_{kj}(n) \tag{3.20}$$

Portanto, na equação (3.20), observa-se que o gradiente local $\delta_j(n)$ de um neurônio oculto depende de alguns fatores, como a derivada da sua função de ativação e o somatório do produto dos gradientes locais $\delta_k(n)$ dos neurônios localizados na camada imediatamente à sua direita pelos pesos sinápticos w_{kj} associados com a conexão das duas camadas.

Em resumo, o ajuste $\Delta w_{ji}(n)$ de acordo com as relações derivadas para o algoritmo *backpropagation* é dado por (3.21):

$$\begin{pmatrix} \text{Correção} \\ \text{de peso} \\ \Delta w_{ji}(n) \end{pmatrix} = \begin{pmatrix} \text{Parâmetro da} \\ \text{taxa de aprendizagem} \\ \eta \end{pmatrix} \cdot \begin{pmatrix} \text{Gradiente} \\ \text{local} \\ \delta_j(n) \end{pmatrix} \cdot \begin{pmatrix} \text{Sinal de entrada} \\ \text{do neurônio } j \\ y_i(n) \end{pmatrix} \tag{3.21}$$

onde o gradiente local $\delta_j(n)$ pode assumir dois casos:

1. Se o neurônio j for de uma camada de saída: $\delta_j(n)$ é obtido pelo produto da derivada $\varphi'(v_j(n))$ pelo sinal de erro $e_j(n)$, ambos sendo associados ao neurônio j, de acordo com a equação (3.17).

2. Se o neurônio j for de uma camada oculta: $\delta_j(n)$ é obtido pelo produto da derivada de sua função de ativação pela soma ponderada entre os gradientes locais dos neurônios da camada à direita e os pesos sinápticos que conectam as duas camadas, de acordo com a equação (3.20).

3.2.3.2 Otimizações para o algoritmo de treinamento *Backpropagation*

Como mencionado nas seções anteriores, o objetivo do algoritmo *backpropagation* é minimizar o erro obtido pela rede através do ajuste de pesos e limiares. A busca pelos valores de pesos e limiares que minimizem a função custo é realizada sobre uma superfície de erro e a combinação de pesos e limiares correspondem a um ponto nesta superfície.

Entretanto, não é garantido encontrar a combinação ótima de pesos e limiares que garantam a minimização da função custo em uma superfície complexa. O algoritmo do gradiente descendente pode demorar a convergir ou levar ao algoritmo para um mínimo local. Por isso, diversas variações do método *backpropagation* têm sido propostas com o objetivo de tornar o processo de convergência mais eficiente. Dentre estes métodos estão: gradiente descendente com *momentum*, método *resilient-propagation* e o método de *Levenberg-Marquardt* (KRÖSE; SMAGT, 1996).

3.2.3.2.1 Gradiente descendente com *momentum*

No algoritmo *backpropagation*, o aumento ou diminuição do valor da taxa de aprendizagem η utilizada na expressão de ajuste dos pesos permite com que as variações dos pesos, de uma interação para outra, sejam maiores ou menores, respectivamente. Isso faz com que a trajetória no espaço de pesos se relacione com essas variações, tornando-a mais suave ou abrupta.

É desejável que a trajetória no espaço de pesos seja a mais suave possível, porém isso pode tornar o processo de aprendizagem da rede lento. Ao aumentar-se a taxa de aprendizagem, devido a grandes modificações nos pesos sinápticos, a rede pode se tornar instável. Entretanto, a instabilidade pode ser resolvida com a adição de um termo chamado *momentum* (ROJAS; FELDMAN, 2013). Assim, a expressão para o ajuste dos pesos é dada por:

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n) \tag{3.22}$$

onde α é o valor do *momentum* que está compreendido entre $[0, 1]$; $\Delta w_{ji}(n-1)$ é a variação dos pesos sinápticos entre duas interações anteriores e sucessivas.

Observando a expressão (3.22), caso α seja zero, a expressão fica de acordo com o *backpropagation* convencional. Porém, para qualquer outro valor, ele já possui influência positiva no processo de convergência, pois adequa a velocidade de convergência a quão distante se está da solução final (VEELENTURF, 1995).

O uso do termo de *momentum* implica em acelerar a convergência da rede à razão de $\eta/(1-\alpha)$. Os valores compreendidos entre $(0.05 \leq \eta \leq 0.75)$ e $(0 \leq \alpha \leq 0.9)$ são normalmente recomendados para o treinamento de Redes MLP (SILVA; SPATTI; FLAUZINO, 2010).

3.2.3.2.2 Método *Resilient-Propagation*

Para as Rede MLP, as funções de ativação mais utilizadas são as funções não-lineares tangente hiperbólica ou a do tipo sigmoide logística. Estas funções podem ser levadas à saturação dependendo do valor do potencial de ativação ao qual são submetidas. Como o algoritmo *backpropagation* precisa da derivada da função de ativação para incrementar os valores dos pesos, nestes intervalos de saturação, o valor da derivada será bem pequeno, levando a uma demora no processo de convergência. A lentidão no processo de convergência do algoritmo se dá pelo fato do esforço computacional que será realizado para levar os valores da matriz de pesos da Rede MLP para as regiões de variação dinâmica da função de ativação. Devido a isto, o método *resilient propagation* leva em consideração a mudança de sinal do gradiente da função erro em vez da variação da magnitude do mesmo (ROJAS; FELDMAN, 2013).

Portanto, a taxa de aprendizado torna-se dinâmica, pois se não houver variação no sinal do gradiente entre duas interações sucessivas, significa que se pode incrementar a taxa de aprendizado devido ao fato de se estar longe do ponto de mínimo da função. Porém, se houver mudança de sinal, significa que o ponto de mínimo foi ultrapassado, devendo-se decrementar a taxa de aprendizagem para que o algoritmo convirja de forma suave para o mesmo (ROJAS; FELDMAN, 2013).

Em termos matemáticos, o valor de ajuste dos pesos é dado pela equação (3.23):

$$\Delta w_{ji}(n) = \begin{cases} -\Delta_{ji}, & \text{se } \frac{\partial E}{\partial w_{ji}} > 0 \\ +\Delta_{ji}, & \text{se } \frac{\partial E}{\partial w_{ji}} < 0 \\ 0, & \text{caso contrário} \end{cases} \tag{3.23}$$

O valor de atualização Δ_{ij} é definido através de um processo de adaptação que depende do sinal da derivada do erro com relação ao peso a ser ajustado, conforme equação (3.24):

$$\Delta_{ji}(t) = \begin{cases} \eta^{+}\Delta_{ji}(t-1), & \text{se } \frac{\partial E(t-1)}{\partial w_{ji}}\frac{\partial E(t)}{\partial w_{ji}} > 0 \\ \eta^{-}\Delta_{ji}(t-1), & \text{se } \frac{\partial E(t-1)}{\partial w_{ji}}\frac{\partial E(t)}{\partial w_{ji}} < 0 \\ \Delta_{ji}(t-1), & \text{caso contrário} \end{cases} \tag{3.24}$$

onde $0 < \eta^{-} < 1 < \eta^{+}$.

De acordo com a equação (3.24), a regra de adaptação dá-se quando a derivada parcial do erro em relação a um peso w_{ji} mantém o seu sinal, indicando que seu último ajuste reduziu o erro cometido e o valor de atualização Δ_{ij} é aumentado de fator η^{+}, acelerando a convergência. Já quando a derivada parcial muda de sinal, indicando que o seu último ajuste foi grande, o valor de atualização Δ_{ij} é reduzido pelo fator η^{-}, mudando a direção do ajuste (BRAGA, 2007).

3.2.3.2.3 Algoritmo de *Levenberg-Marquardt*

Assim como os outros métodos apresentados, o algoritmo de *Levenberg-Marquardt* também procura diminuir o esforço computacional empregado no processo de convergência do algoritmo *backpropagation*. Este algoritmo é um método gradiente de segunda ordem baseado em mínimos quadrados para modelos não-lineares que é incorporado ao algoritmo *backpropagation* a fim de potencializar a eficiência do processo de treinamento. O algoritmo de *Levenberg-Marquardt* diferencia-se do algoritmo *backpropagation* pelo fato de ser uma aproximação do método de Newton e não um método de descida do gradiente da função erro (ROJAS; FELDMAN, 2013).

Em termos matemáticos, o valor de ajuste dos pesos é dado pela equação (3.25):

$$\Delta w = -\left(\mathbf{J}^{T}\mathbf{J} + \mu\mathbf{I}\right)^{-1}\mathbf{J}^{T}\mathbf{e} \tag{3.25}$$

onde:
J uma matriz Jacobiana das derivadas parciais de primeira ordem;
I é uma matriz identidade;
μ é um parâmetro de controle;
e é um vetor de erros a ser minimizado.

3.2.4 Aspectos relacionados à escolha topológica da rede MLP

A escolha de uma Rede MLP que seja mais apropriada para fazer o mapeamento de um dado problema passa pela especificação da topologia da rede. Porém, esta especificação

não segue um conjunto de regras, pois depende de vários fatores, como os mencionados na seção 3.2.3.

Logo, é necessário estimar algumas estruturas topológicas candidatas a mapear o problema especificado e treiná-las com o conjunto de dados de entrada de modo que extraiam o comportamento que rege o processo a ser aprendido e generalize para novas informações apresentadas. Dessa forma, os elementos da topologia da rede escolhidos serão aqueles que melhor satisfizerem estas características de acordo com o critério adotado (HAYKIN, 2001).

3.2.4.1 Validação Cruzada

Uma ferramenta padrão da estatística conhecida como validação cruzada geralmente é utilizada na escolha da melhor estrutura de uma Rede MLP. Esta ferramenta propõe que o conjunto de dados disponível do problema seja dividido de forma aleatória em um conjunto de treinamento e teste. Então, o conjunto de treinamento deverá ser novamente dividido em dois grupos distintos (HAYKIN, 2001):

- *Subconjunto de estimação*, usado para selecionar o modelo;
- *Subconjunto de validação*, usado para validar o modelo.

O objetivo de subdividir-se o conjunto de treinamento é que após a determinação do modelo por meio do ajuste dos parâmetros livres, este modelo seja validado com dados que não foram utilizados no treinamento, avaliando-se assim o desempenho dos vários modelos candidatos e elegendo-se o que apresentou o melhor resultado. Uma escolha sensata para a divisão do conjunto de treinamento é que 80% dos dados sejam destinados ao subconjunto de estimação e os 20% restantes ao subconjunto de validação (KEARNS, 1997).

Então, a utilização da validação cruzada é importante quando se deseja projetar uma rede neural grande e que garanta uma boa generalização. A validação cruzada, portanto, é utilizada para determinar a rede MLP com o melhor número de neurônios ocultos e quando é o melhor momento para parar o treinamento.

3.2.4.2 Generalização

Um aspecto importante que as Redes MLP devem possuir é a capacidade de generalizar, ou seja, após o treinamento, a rede precisa apresentar um alto desempenho para dados que não foram apresentados a ela durante a fase de aprendizagem.

Uma rede neural generaliza de forma adequada quando produz um mapeamento da entrada-saída correto, mesmo quando a entrada for um pouco diferente dos exemplos

usados para treinar a rede. Porém, há casos em que a rede é treinada excessivamente e acaba memorizando os dados do treinamento, perdendo a habilidade de generalizar entre padrões de entrada-saída similares. Esta perda de generalização é conhecida como excesso de ajuste ou excesso de treinamento (*overfitting*) (GURNEY, 2003).

Cabe ressaltar que o aumento indiscriminado de neurônios assim como o aumento de camadas ocultas não garantem uma generalização apropriada quando são apresentados os dados de teste. Também deve-se atentar para o número reduzido de neurônios e camadas ocultas, pois essa quantidade pode ser insuficiente para extrair as informações necessárias para o aprendizado do processo em análise. Neste caso, tem-se o que se chama de subajuste (*underfitting*) e o erro quadrático para este caso, tanto na fase de treinamento quanto de teste é bem elevado (KRÖSE; SMAGT, 1996).

3.2.4.3 Inclusão de parada por antecipação (*Early Stopping*)

Para garantir uma boa generalização é difícil perceber o momento em que o treinamento deve ser encerrado, observando somente a curva de aprendizagem, já que o erro quadrático inicial na fase de treinamento é alto e vai decrescendo ao longo das épocas até o encontro de um mínimo local. Como dito anteriormente, a rede pode decorar os dados de treinamento se o mesmo não for encerrado no momento certo.

Dessa forma, um procedimento simples tem sido inserido na técnica de validação cruzada para efetuar a parada antecipada (*early stopping*) do treinamento visando a maior generalização da rede. Logo, uma dada topologia candidata é constantemente verificada através do uso do subconjunto de validação, de modo que o treinamento é finalizado quando o erro quadrático em relação aos dados de validação for maior que o erro em relação aos dados de estimação entre épocas sucessivas (SILVA; SPATTI; FLAUZINO, 2010).

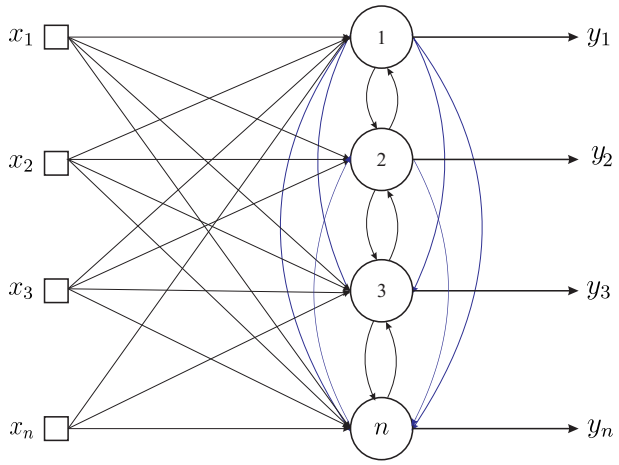
Portanto, a variação repentina do erro quadrático dos dados de validação indica que a rede está decorando as características do subconjunto de estimação, como por exemplo, os próprios ruídos de medição. Assim, o processo de especificação da topologia de redes neurais deve contemplar a capacidade de superar o *underfitting* e evitar o *overfitting*. O método da parada antecipada deve ser aplicado às topologias candidatas de forma individual, porém a seleção da melhor topologia é dada pela validação cruzada (GURNEY, 2003).

3.3 Rede Quantização Vetorial por Aprendizagem (*Learning Vector Quantization Network*-LVQ)

A Rede LVQ é uma das redes pioneiras em aplicações para classificação de padrões. Esta rede foi originalmente desenvolvida baseada na estrutura dos mapas de características auto organizáveis que simulam a representação fisiológica da memória. Contudo, seu princípio comportamental é simples e pode ser considerado como um treinamento adaptativo de vetores quantizadores, em que cada um representa basicamente um modelo de classe no espaço pré-definido de distâncias (FAUSETT, 1994; KATAGIRI, 2000).

A Rede LVQ é uma rede híbrida, uma vez que utiliza tanto o aprendizado não-supervisionado quanto o supervisionado para realizar classificações. Ela é composta de duas camadas, sendo a primeira, denominada de camada competitiva e a segunda, de camada linear. Então, a cada neurônio da primeira camada é designado a uma subclasse, mas, geralmente, vários neurônios são designados para representar uma mesma subclasse. Em seguida, cada subclasse é atribuída a um neurônio na segunda camada. Portanto, o número de neurônios na primeira camada deverá ser maior ou igual ao número de neurônios da segunda camada (HAGAN et al., 2014). Apresenta-se na Figura 21 a estrutura neural básica de uma rede competitiva.

Figura 21 – Estrutura neural básica de rede competitiva



Fonte: adaptado de SILVA, SPATTI e FLAUZINO (2010, p. 222)

Tal como acontece em uma rede competitiva, cada neurônio da primeira camada da Rede LVQ aprende um vetor protótipo $\{w^{(k)}\}$ que permite classificar uma região do espaço de entrada. Assim, dada uma determinada amostra de entrada, $\{x^{(k)}\}$, é realizado o cálculo da distância euclidiana entre $\{x^{(k)}\}$ e $\{w^{(k)}\}$ e aquele neurônio que possuir maior nível de proximidade com a amostra de entrada é declarado vencedor.

O cálculo da distância euclidiana é dado por (3.26):

$$\text{dist}_j^{(k)} = \sqrt{\sum_{i=1}^{n} (x_i^{(k)} - w_i^{(j)})^2}, \quad j = 1, \cdots, J \tag{3.26}$$

onde:
n é o número de parâmetros que compõe a amostra $\{x^{(k)}\}$;
J é o número de neurônios da camada competitiva;
$\text{dist}_j^{(k)}$ é a distância entre o vetor de entrada representando a k-ésima amostra $\{x^{(k)}\}$ em relação ao vetor de pesos do j-ésimo neurônio.

Portanto, o neurônio cujo vetor de peso esteja mais próximo ao vetor de entrada terá sua saída ativada, enquanto que os outros neurônios permanecerão inativos. Declarado o neurônio vencedor, é realizado o ajuste de seus pesos. Logo, se o neurônio vencedor $\{w^{(k)}\}$ estiver representando a classe atribuída a respectiva amostra $\{x^{(k)}\}$, então seus pesos são ajustados a fim de aproximá-lo ainda mais desta amostra. Caso contrário, o ajuste é realizado de forma a afastá-lo da amostra (FAUSETT, 1994; MEHROTRA; MOHAN; RANKA, 1997).

A regra de ajuste dos pesos é dado por (3.27):

$$\begin{aligned} &\text{Se } x^{(k)} \in C^{(j)}\text{: } w^{(j)} + \eta(x^{(k)} - w^{(j)}) \\ &\text{Se } x^{(k)} \notin C^{(j)}\text{: } w^{(j)} - \eta(x^{(k)} - w^{(j)}) \end{aligned} \tag{3.27}$$

O resultado final é dado pelo movimento dos pesos dos neurônios ocultos em direção ao vetor que representa a classe a qual a sua subclasse pertence e do afastamento dos vetores que estão atribuídos às outras classes (HAGAN et al., 2014). O algoritmo de treinamento apresentado é chamado de $LVQ-1$ e sua proposta é ajustar somente os pesos do neurônio vencedor, movendo-o em direção ao vetor de entrada quando a classificação é correta e afastando-o quando é realizada uma classificação incorreta. Outro algoritmo de aprendizagem proposto na literatura, o $LVQ-2$, além de fazer o ajuste no neurônio vencedor, também atualiza os pesos dos outros neurônios (HAGAN et al., 2014).

3.4 Método de Comitês

A solução de tarefas complexas pode ser dada por uma abordagem de aprendizado supervisionado que se baseia na ideia muito utilizada na engenharia: dividir uma tarefa complexa em partes menores para fazer a combinação das soluções individuais, também chamado de princípio dividir para conquistar (HAYKIN, 2009).

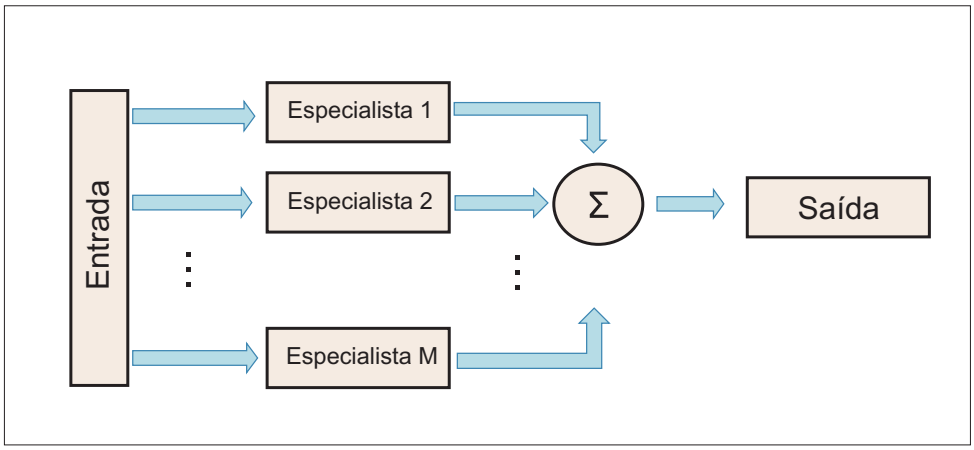
Baseado neste princípio, o método de comitês particiona um problema em subespaços, onde em cada subespaço é designado um algoritmo especialista simples para aprender

as características de cada partição. Dessa forma, as respostas individuais de cada especialista contribuem para a resposta final do problema, diminuindo assim, a complexidade do algoritmo de aprendizagem.

Assim, o método de comitês treina vários classificadores para resolver o mesmo problema. Também chamado de sistema de múltiplos classificadores, está abordagem utiliza classificadores com topologias e número de parâmetros ajustáveis mais simples do que se fosse utilizado uma única estrutura para a solução da mesma tarefa. Outra vantagem apresentada por este método é a diminuição no tempo de treinamento, uma vez que o treinamento de uma grande estrutura topológica provavelmente será maior do que treinar vários especialistas em paralelo. A simplicidade na estrutura dos especialistas evita também que aja um super ajuste dos dados, pois quando se tem um grande número de parâmetros livres a ser ajustado em relação ao tamanho do conjunto de treinamento, o risco de sobre ajuste aumenta (OKUN, 2008).

A arquitetura mais comum do método de comitês é visualidade na Figura 22.

Figura 22 – Arquitetura Geral do Método de Comitê



Fonte: adaptado de SILVA, SPATTI e FLAUZINO (2010, p. 222)

Como pode ser observado, existe um conjunto de classificadores que irão aprender as características dos dados de treinamento e são chamados de base de classificadores. Esta base pode ser formada por diversos algoritmos de aprendizagem, como as redes neurais. Normalmente, a base de classificadores é formada apenas por um tipo de classificador, mantendo a estrutura do comitê homogênea; porém, outras metodologias podem adotar distintos classificadores para formar a base, o que torna o comitê heterogêneo.

Uma característica importante do método de comitês é a capacidade de generalização. Essa metodologia transforma a base de classificadores, que é composta de estruturas simples, com baixo poder discriminatório, também chamada de classificadores fracos, em

uma estrutura capaz de apresentar alta acurácia de predição pela combinação de suas respostas, denominada classificador forte (HAYKIN, 2009).

Na abordagem de comitês, existem três variações que são utilizadas pelos pesquisadores: classificadores combinados, comitês de fracos aprendizes e misturas de especialistas. Os classificadores combinados utilizam um conjunto de classificadores de alto poder discriminatório e combinam os resultados dos mesmos através de regras para obter um elevado desempenho (KUNCHEVA, 2014).

Para o caso dos comitês de fracos aprendizes, a abordagem utiliza um conjunto de classificadores de baixo desempenho e projetam algoritmos para aumentar o desempenho global. O AdaBoost e o Bagging são exemplos de algoritmos utilizados em comitês de aprendizes fracos. A última variação do método de comitês, muito utilizada na área de redes neurais, a estratégia da mistura de especialistas utiliza conjuntos de modelos paramétricos simples que aprende subespaços da tarefa e através da definição de regras fornece uma solução geral (ZHOU, 2012).

Em tarefas de classificação de padrões, uma nova amostra pode ser classificada pelo método de comitês de duas formas: a primeira delas, faz a fusão das saídas dos classificadores, segundo um determinado procedimento, para obter a resposta final na etapa de classificação; já a segunda, apenas a resposta de um dos classificadores é tomada como a resposta final, de acordo com algum critério de seleção (ROKACH, 2010).

No próximo capítulo será apresentada a metodologia utilizada, bem como os aspectos relacionados a formação do conjunto de padrões dos comandos a serem reconhecidos, a parametrização das funções de base radial gaussianas, bem como o projeto da topologia para o conjunto de rede neurais especialistas nas configurações MLP e LVQ.

4 Metodologia

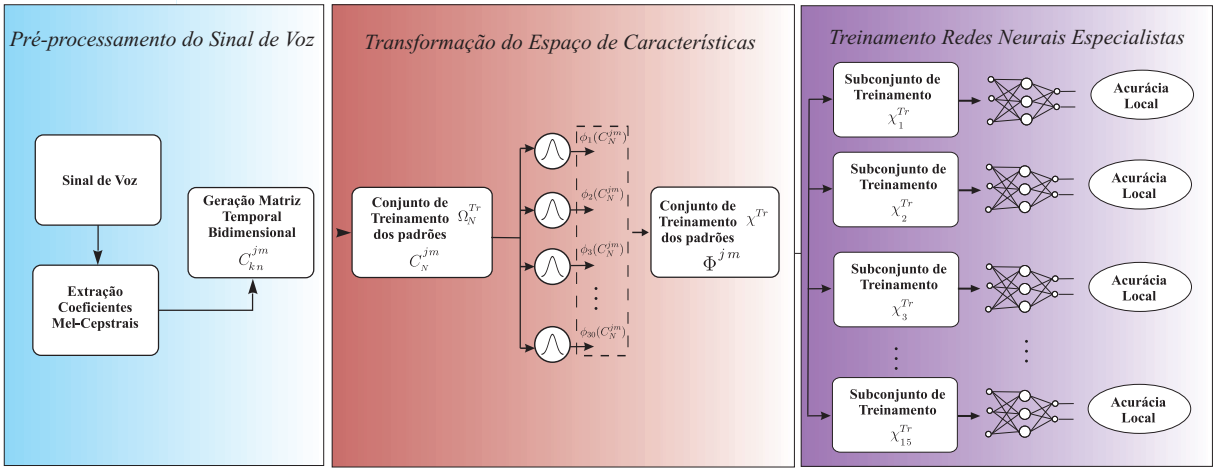
Diante da fundamentação teórica descrita, apresenta-se uma metodologia para a solução de tarefa multiclasse representada por um sistema de reconhecimento de padrões de sinais de voz dado pelas locuções na língua Portuguesa Brasileira dos seguintes comandos mostrados na Tabela 1:

Tabela 1 – Comandos utilizados no sistema de reconhecimento de voz

Comandos				
“Zero”	“Seis”	“Acima”	“Fechar”	“Mínimo”
“Um”	“Sete”	“Aumentar”	“Finalizar”	“Para Trás”
“Dois”	“Oito”	“Desligar”	“Iniciar”	“Para Frente”
“Três”	“Nove”	“Diminuir”	“Ligar”	“Parar”
“Quatro”	“Abaixo”	“Direita”	“Máximo”	“Repousar”
“Cinco”	“Abrir”	“Esquerda”	“Médio”	“Salvar”

Apresenta-se na Figura 23 e na Figura 24 o diagrama esquemático do sistema de reconhecimento de voz baseado na seleção dinâmica do conjunto de redes neurais especialistas a ser desenvolvido na fase de treinamento e na fase de teste, respectivamente.

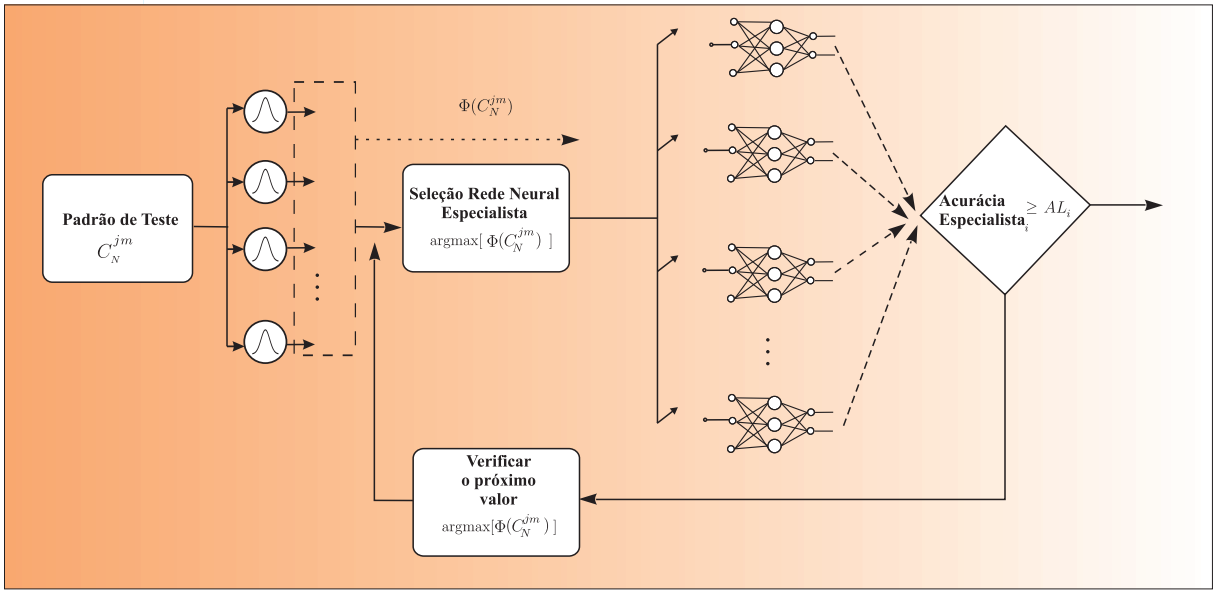
Figura 23 – Diagrama esquemático Sistema de Reconhecimento de Voz - Fase de Treinamento



Na etapa de codificação do sinais de voz, que utiliza os coeficientes mel-cepstrais e a Transformada Cosseno Discreta, fornece padrões obtidos na etapa de pré-processamento do sinal de voz por meio da geração de matrizes temporais bidimensionais, em que estas reproduzem as variações globais e locais no tempo do sinal de voz, assim como o envelope espectral (FISSORE; LAFACE; RAVERA, 1997).

Então, a quantidade de classes utilizada no sistema de reconhecimento de voz eleva a complexidade que o classificador deve possuir para fazer a discriminação das fronteiras de

Figura 24 – Diagrama esquemático Sistema de Reconhecimento de Voz - Fase de Teste



separação entres as classes que, muitas vezes, no espaço dimensional originário, encontram-se muito sobrepostas.

Assim, a partir de um conjunto $\Phi = \{\phi_1, \phi_2, \cdots, \phi_i\}$ de funções de base radial gaussianas (FBRG) adequadamente modeladas, onde $\phi_i, i = 1, 2, \cdots, 30$, tem-se o mapeamento do espaço de características primário para um novo espaço não linear de alta dimensionalidade $\Phi : \Omega \rightarrow \chi$, sendo $\Omega \subseteq \mathbb{R}^m$.

Esta mudança tem por objetivo aumentar a probabilidade de separação linear das categorias, conforme o teorema de Cover, facilitando assim, o processo de classificação. Para cada uma das classes j do problema, modelou-se uma função de base radial gaussiana por meio dos parâmetros centroide e variância $\phi_i(\mu_i, \sigma_i^2)$, extraído dos exemplos dos padrões do conjunto Ω que compõem as distintas classes.

Após a obtenção das 30 (trinta) funções de base radial gaussiana devidamente parametrizadas, cada um dos padrões obtidos através da matriz temporal bidimensional TCD foram mapeados para um espaço $\mathbb{R}^{30}$. Pelo fato das funções de base radial gaussianas estarem parametrizadas com as características de centro e variância de cada classe, neste espaço de alta dimensionalidade, espera-se que haja a clusterização adequada destes padrões.

Especificado o conjunto de treinamento, realizou-se o projeto e definição do classificador, dado por uma arquitetura hierarquizada que agrega a abordagem do método de comitês utilizando redes neurais artificiais com o conjunto de funções de base radial gaussiana. Nesta etapa é feita a análise de desempenho do resultado de classificação final quando se utiliza múltiplas classificadores dado por uma das seguintes configurações

amplamente utilizadas na literatura: as redes neurais MLP e LVQ.

A arquitetura hierarquizada proposta para o reconhecimento de padrões de sinais de voz fornece o resultado da classificação através da seleção dinâmica entre múltiplos classificadores, na qual um conjunto de redes neurais irão se especializar em cada uma das partições pré-definidas do espaço de características mapeado pelas FBRG, o que diminui a complexidade da topologia das configurações MLP e LVQ, o tempo de treinamento e aumenta capacidade de generalização.

Dessa forma, para o desenvolvimento da arquitetura de classificação dos padrões é necessário a execução de 3 etapas:

- treinamento/validação dos especialistas;
- acurácia local dos especialistas;
- seleção dinâmica no teste final.

Nesta primeira etapa, elementos de topologia e algoritmos de treinamento pré-determinados para as configurações de rede MLP e LVQ serão combinados para a geração das melhores características dos 15 especialistas, onde cada um será responsável pelo aprendizado das especificidades de duas classes. Assim, pode-se verificar tanto o comportamento das redes MLP quanto para as redes LVQ e selecionar as topologias dos especialistas que apresentaram maior acerto global e menor erro médio quadrático de validação.

Então, na etapa seguinte, as melhores topologias obtidas na fase de validação para cada especialista serão testadas individualmente para verificar o nível de generalização para as classes as quais foram designados. Dessa forma, é determinado o nível de acurácia local para cada especialista, informação que fará parte das regras definidas na etapa de seleção dinâmica no teste final. Dessa forma, para cada especialista, a topologia que obteve maior acurácia local irá compor o conjunto de múltiplos classificadores.

Por fim, a etapa de seleção dinâmica para o teste final consiste na definição de regras para indicar entre os especialistas aquele mais competente em fornecer a solução final da classificação. Assim, novos padrões gerados pela matriz temporal bidimensional TCD, diferentes daqueles utilizados na fase de treinamento, são aplicados nas 30 FBRG's, utilizadas tanto para fazer a mudança de espaço dos padrões de entrada quanto para fornecerem uma regra de seleção entre os múltiplos classificadores, uma vez que o maior valor de imagem ($y = argmax[\Phi((x))]$) encontrado entre as funções devido ao padrão de entrada irá direcionar para o especialista adequado para finalizar a classificação.

Logo, o resultado final de classificação dado pela rede neural especialista mais competente será comparado ao resultado de acurácia local obtido para este classificador

durante a segunda etapa. Caso o resultado esteja em um nível inferior, será observado no vetor de imagens Φ fornecidos pelas FBRG's qual é a segunda classe que o padrão de entrada tem maior valor. Logo, um novo especialista é selecionado e o resultado fornecido novamente é comparado ao resultado de acurácia local deste especialista. Este processo e repetido até que o resultado do especialista esteja de acordo com a acurácia local.

Portanto, diante dos procedimentos aplicados na elaboração deste trabalho, pode-se definir entre as configurações de redes neurais estudadas, aquela que melhor se adéqua como especialista ao sistema de reconhecimento de voz composto de múltiplos classificadores em um espaço de alta dimensionalidade, diante do problema de classificação de padrões de sinais de voz definidos por um número reduzido de parâmetros no espaço de entrada original.

Neste trabalho, o estudo de desempenho da rede neural LVQ como especialista fornece uma abordagem alternativa para o classificador, já que a configuração MLP aparece na literatura científica como a rede neural mais executada em problemas de reconhecimento de padrões.

4.1 Processamento do Sinal de Voz

4.1.1 Aquisição do sinal de voz

Os padrões de entrada do sistema reconhecedor proposto são provenientes de sinais de voz das pronúncias dos comandos apresentados na Tabela 1. Assim, três bancos de voz foram utilizados para compor as locuções dos comandos pré-estabelecidos para o reconhecimento. Então, as amostras de sinais de voz que representam os 10 dígitos são oriundas dos seguintes bancos de voz:

1. Banco de Voz do Laboratório de Processamento de Sinais-LPS, da Escola Politécnica da Universidade de São Paulo (EPUSP)[1]: banco de voz gravado em ambiente de laboratório, em sala acústica, com baixo nível de ruído, composto de vozes de cinco locutores do sexo masculino e cinco locutores do sexo feminino, todos na faixa etária de 18 a 30 anos de idade. Cada um dos locutores pronunciou os exemplos dos dígitos duas vezes, num total de duzentas locuções, com pausa entre as pronúncias de cada dígito;

2. Banco de voz do Instituto Nacional de Telecomunicações (Inatel) apresentado no trabalho (YNOGUTI; VIOLARO, 2008). Deste banco, foram tomadas pronúncias dos dígitos de cinco locutores do sexo masculino e cinco do sexo feminino, todos

[1] http://www.bv.fapesp.br/en/auxilios/58789/analysis-of-audio-and-speech-signals-for-reconstruction-and-recognition/

na faixa etária de 18 a 50 anos de idade. Cada um dos locutores pronunciou os exemplos dos dígitos uma vez, num total de cem locuções. Este banco é composto de exemplos de dígitos pronunciados de forma contínua, sem pausa entre as pronúncias dos dígitos;

3. Banco de voz gravado no Instituto Federal do Maranhão (IFMA): banco de voz gravado em ambiente sem controle de ruído, composto de vozes de doze locutores do sexo masculino e doze locutores do sexo feminino, todos na faixa etária de 18 a 50 anos de idade. Cada um dos locutores pronunciou os exemplos dos dígitos dez vezes, num total de duas mil e quatrocentas locuções, com pausa entre as pronúncias de cada dígito.

Para os demais comandos, um novo banco de voz foi gerado com locutores provenientes do IFMA e da Universidade Federal do Maranhão (UFMA). Como características, este banco foi gravado em ambiente sem controle de ruído, composto de vozes de 20 locutores do sexo masculino e 20 locutores do sexo feminino, todos na faixa etária de 18 a 60 anos de idade. Metade dos locutores femininos e dos locutores masculinos foram selecionados para compor as amostras de treinamento e cada um deles pronunciou cada comando 20 vezes, enquanto que a outra metade dos locutores masculinos e femininos fizeram parte do grupo para teste, onde cada locutor contribuiu com 10 exemplos de cada comando, totalizando 12 mil locuções. Estas palavras também foram gravadas com pausa entre as pronúncias.

Utilizou-se para as gravações dos sinais de voz uma frequência de amostragem $f_a = 22050$ Hz, com resolução de 16 bits. Portanto, mediante a formação do banco de dados, obteve-se as amostras necessárias à fase do pré-processamento do sinal de voz.

4.1.2 Pré-processamento do sinal de Voz

A fase do pré-processamento do sinal constituiu-se em fazer a segmentação e janelamento das amostras de sinal de voz do banco de dados construído. A segmentação das amostras dos sinais é necessária para limitar o intervalo de tempo no qual se considera válidos um determinado número de parâmetros para o cálculo realizado pela transformada de Fourier de curto prazo. Após definir o tamanho do segmento, a função janela é aplicada em cada um destes segmentos, porém é feita uma sobreposição entre janelas sucessivas devido a atenuação das amostras do sinal nas extremidades da janela.

No trabalho proposto, definiu-se no algoritmo de pré-processamento do sinal de voz desenvolvido, o janelamento dos segmentos através da função de Hamming, dado pela equação (2.1). A sobreposição entre as janelas foi de 50%, desta maneira, o resultado da equação (2.2) é igual a $\frac{1}{2}$. O tamanho da janela em amostras foi calculado por meio da multiplicação da duração da janela $T_\omega = 20$ ms pela frequência de amostragem f_a.

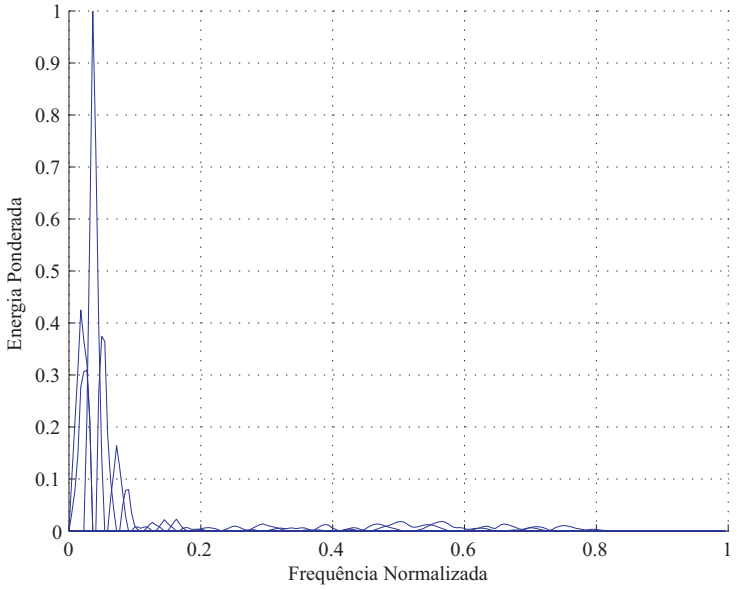
4.2 Extração dos coefientes mel-cepstrais do sinal de voz

Conforme apresentado na seção 2.3.2.2, calculam-se os elementos necessários para obter-se os coeficientes mel-cepstrais das amostras do sinal de voz. Desenvolveu-se também um banco de filtros espaçados na escala mel, abrangendo a faixa de 0 a 4600 Hz. O banco se encontra distribuído em 20 filtros, em que até a frequência limite para segmentação uniforme, dado por $F_u = 1$ kHz, os filtros encontram-se distribuídos em 10 intervalos uniformes, conforme apresentado na Figura 12.

Este banco de filtros é utilizado em cada segmento do sinal de voz, e a energia de cada banda de frequência é calculada. Porém, antes de se obter a energia das bandas de frequências do filtro, calculou-se o espectro de frequência do sinal de voz em análise através da transformada discreta de Fourier, dada pela equação (2.10). Então, realizou-se o cálculo da energia, dado pelo somatório do módulo do espectro de frequência do sinal de voz, ponderado pelas funções triangulares, que representam as bandas de frequência do banco de filtros, conforme equação (2.11).

Na Figura 25, ilustra-se a energia calculada para um segmento do sinal de voz por meio da aplicação do banco de filtros.

Figura 25 – Energia de um segmento de voz ponderada por um banco de 20 filtros triangulares



Fonte: Silva (2015, p. 45)

Por fim, os coeficientes mel-cepstrais foram obtidos utilizando-se a energia calculada para cada banda de frequência, conforme a equação (2.13).

4.3 Geração da matriz temporal bidimensional

Após a obtenção dos coeficientes mel-cepstrais do sinal de voz, foi realizada a codificação a partir da transformada cosseno discreta (TCD), que permite sintetizar as variações de longo prazo do envelope espectral do sinal de voz (FISSORE; LAFACE; RAVERA, 1997). O resultado desta codificação foi a geração de uma matriz temporal bidimensional TCD, obtida conforme equação (4.1):

$$C_k(n,T) = \frac{1}{N}\sum_{t=1}^{T} \text{mfcc}_k(t) \cos\left[\frac{(2t-1)n\pi}{2T}\right] \tag{4.1}$$

onde:
k, que varia de $1 \leq k \leq K$, é k-ésima linha componente do t-ésimo segmento da matriz.
K é o número de coeficientes mel-cepstrais;
n, que varia de $1 \leq n \leq N$ é a n-ésima coluna. n é a ordem da matriz TCD;
T é número de vetores de observação dos coeficientes mel-cepstrais no eixo do tempo;
$\text{mfcc}_k(t)$ representa os coeficientes mel-cepstrais.

Para cada amostra do sinal de voz foi formada uma matriz temporal bidimensional e os elementos constituintes da matriz foram obtidos por meio dos seguintes procedimentos:

1. Para cada um dos 30 comandos $\boldsymbol{P}$ a serem codificados são tomadas 20 locuções. Estes 20 exemplos de cada comando são devidamente segmentados em T segmentos ao logo do intervalo de tempo do sinal de voz. Assim, têm-se: D_0^0, D_1^0, D_2^0, ..., D_{20}^0, D_0^1, D_1^1, D_2^1, ..., D_{20}^1, ..., D_m^j, onde $j = 0, 1, 2, \cdots, 30$ representa o comando a ser codificado e $m = 0, 1, 2, \cdots, 20$ representa o exemplo tomado para cada comando;

2. Cada um dos segmentos de uma dada locução de um comando $\boldsymbol{P}$ produz uma quantidade K de coeficientes mel-cepstrais, alcançando assim, características significantes dentro de cada segmento ao longo do tempo;

3. Para os K coeficientes mel-cepstrais encontrados dentro de um segmento é calculada a TCD de ordem N. Dessa forma, tem-se c_1 do segmento t_1, c_1 do segmento $t_2, \cdots, c_1$ do segmento t_T, c_2 do segmento t_1, c_2 do segmento $t_2, \cdots$, c_2 do segmento t_T e assim por diante, obtendo-se os elementos $\{c_{11}, c_{12}, c_{13}, \cdots, c_{1N}\}$, $\{c_{21}, c_{21}, c_{23}, \cdots, c_{2N}\}$, $\{c_{K1}, c_{K2}, c_{K3}, \cdots, c_{KN}\}$, e a matriz na equação (4.1);

4. Assim, para cada locução do comando $\boldsymbol{P}$ tem-se uma matriz temporal bidimensional TCD C_{kn}^{jm}.

4.4 Mudança de Dimensionalidade do Espaço de Características dos Padrões

Cada exemplo m dos padrões j dos sinais de voz a serem reconhecidos pelo sistema proposto são gerados por meio de uma matriz temporal bidimensional TCD C_{kn}^{jm}, na qual é definida por $C_{kn}^{jm} = \{c_{il} \mid c_{il} \in \mathbb{R}, i = 1, \cdots, k \; e \; l = 1, \cdots, n\}, k = n$.

Considerando $\mathscr{V}(\mathbb{R})$ o espaço vetorial sob o corpo $\mathbb{R}$ das matrizes quadradas $\mathscr{M}(q,q) = \{A_{q\times q} = a_{il} \mid a_{il} \in \mathbb{R}, i = 1, \cdots, q \; e \; l = 1, \cdots, q\}$ que está munido das operações de adição $\mathscr{V} \times \mathscr{V} \to \mathscr{V}$ e multiplicação por escalar $\alpha \in \mathbb{R}, \alpha \times \mathscr{V} \to \mathscr{V}$ e satisfaz as propriedades de adição, multiplicação por escalar e distributividade.

Seja $\Lambda = C_{kn}^{11}, \cdots, C_{kn}^{1m}, C_{kn}^{21}, \cdots, C_{kn}^{2m}, \cdots, C_{kn}^{jm}$ o conjunto de matrizes quadradas dadas pelas matrizes TCD dos padrões j. Assim, $\Lambda \subset \mathscr{V}$ é um subespaço de $\mathscr{V}$, uma vez que $\alpha C_{kn}^{jm\prime} + \beta C_{kn}^{jm\prime\prime} \in \Lambda$ para todos $C_{kn}^{jm\prime}, C_{kn}^{jm\prime\prime} \in \Lambda$ e para todo $\alpha, \beta \in \mathbb{R}$.

Sejam:

$$\mathscr{C}_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}_{k\times n}, \mathscr{C}_2 = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}_{k\times n}, \cdots, \mathscr{C}_{kn} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}_{k\times n}$$

O conjunto $\lambda = \{\mathscr{C}_1, \mathscr{C}_2, \cdots, \mathscr{C}_{k\times n}\}$ é uma base canônica para geração do subespaço Λ, $\Lambda = G(\lambda)$. Dado que o número de matrizes que constitui a base geradora de Λ é finito, tem-se que Ω é chamado de *dimensão finita*. Logo, o número de elementos da base λ determina a dimensão de Λ, denotada por $\dim\Lambda$, sendo que $\dim\Lambda \leq \dim\mathscr{V}$. Então, $\dim\Lambda = k \times n$.

Tendo em vista a dimensão de Λ, pode-se fazer uma relação de equivalência ou isormorfismo de Λ com um subespaço dos $\mathbb{R}$. Assim, $\lambda = \{\mathscr{C}_1, \mathscr{C}_2, \cdots, \mathscr{C}_{k\times n}\}$ uma base de Λ sob $\mathbb{R}$ com $\dim\Lambda = k \times n$. Tem-se o isomorfismo $T : \Lambda \to \mathbb{R}^{k\times n}$ por $T(\mathscr{C}_{k\times n}) = e_{k\times n}$, onde $e_{k\times n}$ é $(k \times n)$-ésimo vetor de $(k \times n) - uplas$ da base canônica de $\mathbb{R}^{k\times n}$: $\upsilon = \{e_1 = (1, 0, \cdots, 0), e_2 = (0, 1, \cdots, 0), \cdots, e_{k\times n} = (0, 0, \cdots, 1)\}$.

Portanto, as matrizes temporais bidimensionais de Λ foram transformadas em vetores do $\mathbb{R}^{k\times n}$, denominados C_N^{jm}, onde $N = k \times n$, que preservam o alinhamento temporal dos coeficientes mel-cepstrais e são dados por:

$$C_N^{jm} = \left[c_{11}^{jm},\ c_{12}^{jm},\ \cdots,\ c_{1n}^{jm}, c_{21}^{jm},\ c_{22}^{jm},\ \cdots,\ c_{2n}^{jm},\ \cdots,\ c_{kn}^{jm}\right]', \quad \begin{array}{l} j = 1, 2, \cdots, 30 \\ m = 1, 2, \cdots, 20 \end{array}$$

Deste modo:

$$
\begin{aligned}
C_N^{11} &= [c_{11}^{11},\ c_{12}^{11},\ \cdots,\ c_{1n}^{11}, c_{21}^{11},\ c_{22}^{11},\ \cdots,\ c_{2n}^{11},\ \cdots,\ c_{kn}^{11}]' \\
C_N^{12} &= [c_{11}^{12},\ c_{12}^{12},\ \cdots,\ c_{1n}^{12}, c_{21}^{12},\ c_{22}^{12},\ \cdots,\ c_{2n}^{12},\ \cdots,\ c_{kn}^{12}]' \\
&\vdots \\
C_N^{120} &= [c_{11}^{120},\ c_{12}^{120},\ \cdots,\ c_{1n}^{120}, c_{21}^{120},\ c_{22}^{120},\ \cdots,\ c_{2n}^{120},\ \cdots,\ c_{kn}^{120}]' \\
C_N^{21} &= [c_{11}^{21},\ c_{12}^{21},\ \cdots,\ c_{1n}^{21}, c_{21}^{21},\ c_{22}^{21},\ \cdots,\ c_{2n}^{21},\ \cdots,\ c_{kn}^{21}]' \\
C_N^{22} &= [c_{11}^{22},\ c_{12}^{22},\ \cdots,\ c_{1n}^{22}, c_{21}^{22},\ c_{22}^{22},\ \cdots,\ c_{2n}^{22},\ \cdots,\ c_{kn}^{22}]' \\
&\vdots \\
C_N^{220} &= [c_{11}^{220},\ c_{12}^{220},\ \cdots,\ c_{1n}^{220}, c_{21}^{220},\ c_{22}^{220},\ \cdots,\ c_{2n}^{220},\ \cdots,\ c_{kn}^{220}]' \\
&\vdots \\
C_N^{3020} &= [c_{11}^{3020},\ c_{12}^{3020},\ \cdots,\ c_{1n}^{3020}, c_{21}^{3020},\ c_{22}^{3020},\ \cdots,\ c_{2n}^{3020},\ \cdots,\ c_{kn}^{3020}]'
\end{aligned}
$$

O subespaço vetorial Ω definido pelos vetores C_N^{jm} em $\mathbb{R}^N$ pode sofrer uma transformação para um novo espaço vetorial de maior dimensionalidade por meio de uma transformação $\Phi : \mathbb{R}^N \to \mathbb{R}^d$, onde $d > N$. A transformação aplicada é do tipo $\Phi(C_N^{jm}) = (\phi_1(C_N^{jm}), \phi_2(C_N^{jm}), \cdots, \phi_d(C_N^{jm}))$. O conjunto $\Phi = \{\phi_1, \phi_2, ..., \phi_d\}$ é formado por funções do tipo não-linear, o que confere a não linearidade na transformação.

Em tarefas de reconhecimento multiclasse, onde os padrões de distintas classes encontram-se sobrepostos no espaço dimensional original, a mudança de dimensionalidade dos padrões torna-se uma estratégia útil no processo de classificação. Segundo o Teorema de Cover, a probabilidade dos padrões serem separados linearmente aumenta quando a dimensionalidade do espaço é alta, permitindo assim, o uso de classificadores de estrutura simples (ZHOU, 2012).

4.5 Conjunto de Treinamento e Conjunto de Teste

O conjunto de treinamento Ω_{NL}^{Tr}, onde $N = 4, 9, 16$ representa o número de parâmetros dos padrões do conjunto; L, o número total de locuções e Tr indica que o conjunto é de treinamento foi composto da seguinte forma:

Composição dos exemplos de treinamento dos dígitos:

1. Banco EPUSP: Seleção de 6 locutores, na qual 3 são masculinos e 3 são femininos. Cada locutor pronunciou um exemplo de cada dígito, totalizando 60 dígitos pronunciados;

2. Banco INATEL: Seleção de 4 locutores, em que 2 são locutores masculinos e 2 são femininos. Cada locutor pronunciou um exemplo de cada dígito, num total de 40 locuções;

3. Banco IFMA: Seleção de 10 locutores, onde 5 locutores são masculinos e 5 locutores são femininos. Os locutores participaram com a pronuncia de um exemplo de cada dígito, totalizando 100 dígitos.

Composição dos exemplos de treinamento dos comandos de ordem:

1. Banco IFMA: Seleção de 20 locutores, onde 10 locutores são masculinos e 10 locutores são femininos. Cada um dos locutores contribuiu com a pronuncia de um exemplo de cada palavra, totalizando 400 locuções.

Assim, o conjunto de treinamento é composto de 600 locuções, em que se tem 20 exemplos de cada comando a ser reconhecido ($m = 20$). Logo, o conjunto de treinamento utilizado é do tipo balanceado, ou seja, todas as classes possuem o mesmo quantitativo de exemplos, o que evita o enviesamento do classificador.

O conjunto de treinamento foi particionado no subconjunto de estimação Ω_N^E, que possui 80% dos padrões do conjunto de treinamento, e no conjunto de validação Ω_N^V, representando os demais 20% do conjunto de treinamento $\Omega_{N600}^{Tr} = \Omega_N^E \cup \Omega_N^V$. Isto é feito para que durante o processo de treinamento da rede neural, as amostras pertencentes ao conjunto de validação sejam usadas para verificar o grau de generalização das redes neurais, uma vez que o algoritmo de treinamento é finalizado quando o erro de validação se eleva em relação ao erro de treinamento, o que significa que está ocorrendo *overfitting*.

A fase de teste é realizada utilizando um conjunto Ω_{NL}^T formado por 40 locutores, onde 20 locutores são masculinos (Ω_{NL}^{TM}) e 20 locutores são femininos (Ω_{NL}^{TF}). Todos os locutores pertencem ao banco de voz IFMA, porém são locutores que não participaram com pronúncias para o conjunto de treinamento. Dentre o total de locutores, metade deles contribuiu com 10 exemplos para cada dígito totalizando 1000 locuções masculinas e 1000 locuções femininas. A outra metade pronunciou 10 exemplos para cada comando de ordem, resultando em 2000 locuções femininas e 2000 locuções masculinas. Portanto, o conjunto de teste disponível para verificar a generalização do sistema de reconhecimento multinível possui 6000 amostras no total ($\Omega_{N6000}^T = \Omega_{N3000}^{TM} \cup \Omega_{N3000}^{TF}$).

4.6 Parametrização Funções de Base Radial Gaussiana

O reconhecimento de voz utilizando seleção dinâmica de redes neurais utiliza um conjunto de 30 funções de base radial gaussiana que possui duas finalidades no sistema proposto: a primeira, na fase de treinamento, é fazer o mapeamento do espaço de características original, dado pelos padrões C_N^{jm}, em um novo espaço não linear de alta dimensionalidade para facilitar a separabilidade dos padrões; a segunda, na fase de teste

é, além de fazer o mapeamento de uma nova amostra para o espaço de alta dimensionalidade, fornecer uma regra de seleção dos múltiplos classificadores.

A quantidade de funções de base radial gaussianas escolhidas está relacionada a quantidade de comandos a serem reconhecidos no problema. Desse modo, por meio das amostras pertencentes a cada comando j, determinou-se os parâmetros centroide μ_j e a matriz de covariância $\Sigma_j = \sigma_j^2 \mathrm{I}$. Para a obtenção apropriada dos 30 centroides das FBRG's, utilizou-se o método para esta finalidade denominado de *k-means*, cujo propósito é posicionar interativamente os centros de k-gaussinas em regiões onde os padrões de entrada tenderão a se agrupar (BISHOP, 1995; SILVA; SPATTI; FLAUZINO, 2010).

Assim, o conjunto de treinamento Ω_{NL}^{Tr} foi aplicado ao algoritmo *k-means*, onde k foi definido como 30, e os centroides e as matrizes de covariâncias para cada função foram obtidos, conforme os passos definidos no Algoritmo 1.

Algoritmo 1 *K-means* algoritmo

Entrada: Conjunto de treinamento Ω_{NL}^{Tr} contendo as L amostras dos padrões C_N^{jm}
Entrada: k = número de grupos
Saída: k vetores de média, isto é, os centroides μ_k dos k grupos θ
Saída: k variâncias σ_k^2

Início
  Escolha estimativas iniciais arbitrárias para os μ_k dos k grupos
  repita
    para todo $C_N^{jm} \in \Omega_{NL}^{Tr}$ **faça**
      Calcular as distâncias euclidianas entre $C_N^{jm(i)}$ e μ_k, considerando-se cada k centroide por vez
      Selecionar o centroide μ_k que contenha a menor distância com o intuito de agrupar o referida amostra junto ao centroide mais próximo
      Atribuir a amostra C_N^{jm} ao grupo θ_k
      para todo μ_k **faça**
        Ajustar μ_k de acordo com as amostras em θ_k:
        $\mu_k = \frac{1}{l} \sum_{C_N^{jm} \in \theta_k} C_N^{jm}$
                                  ▷ l é o número de amostras em θ_k
      fim para
    fim para
  até que não haja mais mudança nos k grupos entre as interações
  para todo μ_k **faça**
    Calcular a variância σ_k^2 de cada uma das funções de base radial gaussiana pelo critério da distância quadrática média:
    $\sigma_k^2 = \frac{1}{l} \sum_{C_N^{jm} \in \theta_k} \sum_{z=1}^{N} (C_N^{jm} - \mu_k)^2$
    $\Sigma_k = \sigma_k^2 \mathrm{I}$
  fim para
fim Início

Fonte: SILVA, SPATTI e FLAUZINO (2010, p. 105)

Portanto, ao final destes procedimentos, tem-se os parâmetros necessários para

o mapeamento de alta dimensionalidade não linear e o critério de pré-seleção das redes neurais especialistas através do conjunto de funções de base radial $\Phi = \{\phi_1, \phi_2, \cdots, \phi_{30}\}$ devidamente modeladas, na qual, quanto mais próxima estiver uma amostra de teste do centroide de uma dada função gaussiana, mais significativo será o valor da resposta do campo receptivo de cada classes.

4.7 Sistema de Múltiplos Classificadores - Seleção Dinâmica baseada em Acurácia Local (SD-AL)

Em um sistema de reconhecimento baseado no método de comitês, múltiplos classificadores são utilizados para dividir o espaço de características em regiões menores como forma de superar a dificuldade que um único classificador possui em delimitar os limites em uma tarefa multiclasse (GIACINTO; ROLI, 1999; DIDACI et al., 2005).

Assim, dado múltiplos classificadores, o resultado final de classificação é feita através da seleção dinâmica de um dos classificadores, especializados em determinada área local do espaço de características. Então, dado um vetor de características $\mathbf{x} \in \mathbb{R}^n$, o classificador especialista na região a qual $\mathbf{x}$ pertence é dado como o mais competente para rotular $\mathbf{x}$ (KUNCHEVA, 2000; BRITTO; SABOURIN; OLIVEIRA, 2014).

Então, seja $\mathscr{D} = \{D_1, D_2, \cdots, D_T\}$ o conjunto de classificadores e $\Gamma = \{\gamma_1, \gamma_2, \cdots, \gamma_i\}$ o conjunto de classes do problema. Seja K divisões do espaço de características $\mathbb{R}^n$ em regiões de competência, onde $K > 1$. Estas regiões de competência são denominadas por $R_1, R_2, \cdots, R_K$. Estas regiões não precisam ter uma forma especifica ou tamanho.

Durante a fase de treinamento dos múltiplos classificadores, cada classificador de $\mathscr{D} = \{D_1, D_2, \cdots, D_T\}$ é especificado para cada região R_K. Logo, o número de classificadores L não precisa ser igual ao número de regiões K e K pode ser menor ou igual ao número de classes γ_i. Ao final do treinamento, é definida *a priori* a acurácia local de cada especialista $AL_{T,K}$, ou seja, a generalização de cada especialista para suas respectivas regiões de competência.

Cada região R_K é caracterizada por uma função radial gaussiana ϕ_K com centroide μ_K e variância σ_K^2 que permite definir a distribuição de probabilidade de uma amostra de teste $\mathbf{x}^*$ para cada região R_K.

Na fase de teste, verifica-se a região R_K a qual uma amostra de teste $\mathbf{x}^*$ tem maior probabilidade de pertencer. Logo, o resultado $d_T(\mathbf{x}^*)$ do classificador D_T mais competente é comparado a $AL_{T,K}$ deste mesmo classificador. Assim, caso $d_T(\mathbf{x}^*) \geq AL_{T,K}$, a amostra $\mathbf{x}^*$ é associada a classe γ_i pertencente a região R_K com taxa de classificação $d_T(\mathbf{x}^*)$.

Se a condição $d_T(\mathbf{x}^*) \geq AL_{T,K}$ não for satisfeita, uma nova verificação no vetor de distribuição de probabilidades é realizada e um outro classificador do conjunto $\mathscr{D}$ é

selecionado para comparação da acurácia local (WOODS; KEGELMEYER; BOWYER, 1997).

4.8 Projeto das Redes Neurais

A segunda parte integrante do sistema de reconhecimento de voz são as redes neurais como especialistas. Para contornar os problemas relacionados a tarefa multiclasses, as 30 categorias do sistema de reconhecimento de voz foram divididas em subespaços de 2 classes.

Dessa forma, 15 estruturas de redes neurais foram definidas para aprender as características destes subespaços. Baseando-se no princípio de *dividir para conquistar* do método de comitês, esta abordagem de múltiplos especialistas simplifica a complexidade topológica que uma única rede neural apresentaria para solucionar a mesma tarefa, além das dificuldades de convergência do algoritmo e *overfitting* dos dados.

Dessa forma, visualiza-se na Tabela 2 a distribuição de cada uma das 30 classes entre os especialistas definidos, tanto para a configuração LVQ quanto MLP.

Tabela 2 – Divisão das Classes entre os Especialistas

Classe	**Especialista**	**Classe**	**Especialista**	**Classe**	**Especialista**
“Zero” “Um”	RN_Esp1	“Abaixo” “Abrir”	RN_Esp6	“ Iniciar” “Ligar”	RN_Esp11
“Dois” “Três”	RN_Esp2	“Acima” “Aumentar”	RN_Esp7	“Máximo” “Médio”	RN_Esp12
“Quatro” “Cinco”	RN_Esp3	“Desligar” “Diminuir”	RN_Esp8	“Mínimo” “‘Para Trás”	RN_Esp13
“Seis” “Sete”	RN_Esp4	“Direita” “Esquerda”	RN_Esp9	“Para Frente” “Parar”	RN_Esp14
“Oito” “Nove”	RN_Esp5	“Fechar” “Finalizar”	RN_Esp10	“Repousar “ Salvar”	RN_Esp15

Uma vez definida a quantidade de especialistas para a tarefa de reconhecimento dos padrões de sinais de voz, é necessário especificar a melhor estrutura para o aprendizado das características de cada subespaço do problema. Então, as configurações de redes neurais existentes na literatura possuem um conjunto de elementos de topologia variáveis que, escolhidos de forma apropriada, permitem que a rede neural apresente um erro mínimo de resposta em sua saída para o problema a ser solucionado (KATAGIRI, 2000; SILVA; SPATTI; FLAUZINO, 2010; HAYKIN, 2001).

Dessa forma, para as quinze redes neurais especialistas, tanto na configuração MLP quanto LVQ, foram pré-estabelecidos os mesmos elementos de topologias e algoritmos de treinamento a serem combinados durante a fase de treinamento para verificar quais

estruturas conseguiram extrair com maior desempenho as características dos subespaços as quais foram designadas.

A seguir são apresentados como as configurações LVQ e MLP foram especificadas na fase de treinamento.

4.8.1 Especialista LVQ

Para a estrutura da Rede Neural LVQ foi necessário definir a taxa de aprendizagem η e o número de neurônios n da camada competitiva. Logo, foram propostos o conjunto $\eta = 0.01, 0.1, 0.5, 0.9$ e o conjunto $n = 60, 90, 120, 150$ para a simular as diferentes topologias e permitir a escolha da melhor entre elas.

Os valores definidos no conjunto η são frequentemente utilizados na literatura especializada (HAYKIN, 2009; SILVA; SPATTI; FLAUZINO, 2010) e o conjunto n foi especificado levando-se em consideração que o número de neurônios na camada oculta deve ser maior que o número de entradas e maior que o número de saídas da rede neural.

Pelo fato dos vetores C_N^{jm} serem mapeados para um espaço $\mathbb{R}^{30}$, a entrada de cada um dos três especialistas LVQ será fixada em 30 nós fonte. Já a saída de cada especialista é dada pela quantidade de classes que integram cada subespaço especificados.

Devido ao particionamento do espaço de características total em quinze subespaços, a saída de cada especialista deve possuir 2 neurônios, ou seja, um neurônio para cada classe. Assim, definiu-se um conjunto de neurônios dado por múltiplos da quantidade de entradas da rede neural, iniciando com 60 neurônios como o menor número de neurônios na camada oculta no qual as simulações seriam realizadas.

O incremento de neurônios na camada oculta até o valor máximo de neurônios para o conjunto n, estipulado em 150, permite observar o comportamento da rede em relação ao aumento do número de neurônios na camada oculta.

Para cada conjunto de treinamento $(\Omega_{4L}^{Tr}, \Omega_{9L}^{Tr}, \Omega_{16L}^{Tr})$ do espaço de características original foram treinadas um total de 16 topologias da configuração LVQ. A simulação foi realizada mantendo-se o número de neurônios fixo enquanto se variava a taxa de aprendizagem. Dessa forma, obteve-se todas as possibilidades de topologia para os conjuntos η e n especificados.

O número de épocas adotado para as simulações foi de 1000. Verificou-se que a escolha deste valor foi adequada, uma vez que foi alcançada a convergência do algoritmo.

Uma verificação importante nestas simulações foi em relação a taxa de aprendizagem. Observou-se que o aumento da mesma levava à rápida convergência do algoritmo, porém instável, apresentando erro médio quadrático com resultados muito acima do erro de $10^{(-3)}$ estipulado no projeto. Assim, considerou-se apenas as topologias que tinham

como taxa de aprendizagem $\eta = 0.01$.

Desta maneira, resume-se na Tabela 3 os elementos da topologia e de treinamento para as simulações dos especialistas da Rede Neural LVQ.

Tabela 3 – Elementos Rede Neural LVQ

Nº de Neurônios	$n = \{60, 90, 120, 150\}$
Taxa de Aprendizagem	$\eta = 0.01$
Nº de Épocas	Epoch = 1000
Algoritmo de Treinamento	LVQ-1

Como observado na Tabela 3, as combinações possíveis estarão condicionadas à variação do número de neurônios da camada competitiva. Assim, pode ser observada a taxa de reconhecimento dos padrões de sinal de voz com o incremento dos elementos de processamento da rede neural. Após o treinamento das Redes LVQ, mediante todas as combinações dos elementos de topologia definidos, traçou-se o comportamento da Rede LVQ quanto à variação destes elementos.

4.8.2 Especialista MLP

A estrutura da Rede Neural MLP é definida por alguns elementos variáveis que, se escolhidos de forma adequada, permitem um bom desempenho da rede neural na solução do problema proposto. Na Tabela 4 apresenta-se estes elementos variáveis que são combinados em algumas simulações para definir a melhor topologia.

Tabela 4 – Elementos variáveis do Perceptron de Múltiplas Camadas

Elemento	**Símbolo**	**Intervalo Típico**
Nº de Camadas Ocultas	θ	$[1, \infty)$
Nº de Neurônios Ocultos	n	$[2, \infty)$
Taxa de aprendizagem	η	$[0, 1]$
Constante de Momento	α	$[0, 1]$

Logo, foram definidos os elementos variáveis para o treinamento das topologias. Mostra-se na Tabela 5 os intervalos escolhidos para as simulações.

Tabela 5 – Elementos variáveis do Perceptron de Múltiplas Camadas escolhidos

Elemento	**Símbolo**	**Intervalo Típico**
Nº de Camadas Ocultas	θ	1 e 2
Nº de Neurônios Ocultos	n	60,90,120,150
Taxa de aprendizagem	η	0.01,0.1,0.5,0.9
Constante de Momento	α	0.8

Além de definir os elementos da topologia da rede, utilizou-se neste trabalho quatro algoritmos de treinamento diferentes para a Rede MLP. Dessa forma, pode-se verificar o algoritmo que apresenta melhores resultados para o conjunto de padrões apresentados à rede. Os algoritmos de treinamento escolhidos foram:

- Gradiente descendente (GD);
- Gradiente descendente com *momentum* (GDM);
- *Resilient Propagation* (RP);
- *Levenberg-Marquardt* (LM).

Os números de camadas ocultas a serem simulados foram definidos pelo fato de que, para problemas de classificação de padrões, a utilização de até duas camadas é suficiente para esta aplicação (SILVA; SPATTI; FLAUZINO, 2010).

O conjunto η e o conjunto n foram definidos segundo os mesmos critérios da configuração LVQ.

Para as simulações envolvendo as Redes MLP de duas camadas ocultas, definiu-se que a segunda camada oculta n_{oc} apresenta 30 neurônios.

Especificou-se este valor levando-se em consideração que é um número menor do que todos aqueles pertencentes ao conjunto n e maior que o número de saídas da rede neural. Este valor é fixado para a combinação com todos os números de neurônios do conjunto n.

A função de ativação utilizada em todos os neurônios é a função tangente hiperbólica. Esta função foi escolhida pelo fato de ser uma função não-linear, o que permite o mapeamento de problemas complexos e apresenta um intervalo contínuo entre $[-1, 1]$, pois os parâmetros dos padrões de entrada da rede neural encontram-se normalizados neste intervalo.

As simulações foram realizadas buscando-se fazer todas as combinações entre os elementos variáveis, número de camadas ocultas e algoritmo de treinamento. Para cada um dos quatro algoritmos de treinamento, realizaram-se simulações com redes de uma camada oculta e redes de duas camadas ocultas.

Após a combinação entre os algoritmos de treinamento e o número de camadas, as redes tiveram os elementos variáveis η e n combinados.

Para as redes neurais com duas camadas, somente a primeira camada oculta tem o número de neurônios variável, sendo a segunda camada oculta fixada em 30 neurônios.

Foram realizados 100 treinamentos para cada combinação do algoritmo de treinamento “versus” número de camadas “versus” número de neurônios “versus” taxa de

aprendizagem. Cada um destes 100 treinamentos utilizou diferentes inicializações dos pesos, realizados de forma aleatória sobre uma distribuição uniforme entre os valores $[-0.01, 0.01]$.

Este intervalo de inicialização aleatória dos pesos justifica-se pelo fato de ser menor que o intervalo de valores que compreende os parâmetros dos padrões do conjunto de treinamento, evitando assim, a saturação da função de ativação utilizada pelos neurônios, impedindo a convergência da rede neural.

Observou-se o comportamento das redes neurais em relação ao tempo de treinamento e a capacidade de generalização, já que um conjunto adequado de pesos iniciais permite a diminuição no tempo de treinamento e uma alta probabilidade de atingir o mínimo global da função erro. Além disso, esse conjunto pode melhorar significativamente o desempenho na generalização (SOUSA, 2016)

As topologias simuladas são treinadas utilizando os conjuntos $(\Omega_{4L}^{Tr}, \Omega_{9L}^{Tr}, \Omega_{16L}^{Tr})$ e dessa forma, verifica-se a resposta da Rede MLP ao incremento do número de parâmetros dos padrões de sinal de voz apresentados no espaço de características original.

O número de épocas adotado para as simulações foi de 1000 da mesma forma como para as redes LVQ, apresentando-se adequada também no treinamento das redes MLP. Verificou-se também, pelo mesmo motivo que as redes neurais LVQ, que a taxa de aprendizagem $\eta = 0.01$ é mais adequada durante o treinamento das topologias. A constante de *momentum* é apenas utilizada pelo algoritmo de treinamento GDM.

Na Tabela 6 apresenta-se um resumo dos aspectos topológicos e algoritmos de treinamento utilizados neste trabalho para a escolha da topologia adequada para a Rede Neural MLP.

Tabela 6 – Elementos de treinamento das Redes Neurais MLP

Elemento	**Símbolo**	**Intervalo Definido**
Algoritmos de Treinamento	-	GD, GDM,RP,LM
Nº de Camadas Ocultas		1 e 2
Nº de Neurônios Ocultos 1 camada oculta	n	60,90,120,150
Nº de Neurônios Ocultos 2 camadas ocultas	n_oc	30
Taxa de aprendizagem	η	0.01
Constante de Momento	α	0.8

No capítulo seguinte serão apresentados os resultados obtidos durante a fase de treinamento e validação e durante a fase teste para o reconhecimento de voz utilizando as configurações MLP e LVQ, juntamente com uma análise destes resultados.

5 Resultados Experimentais

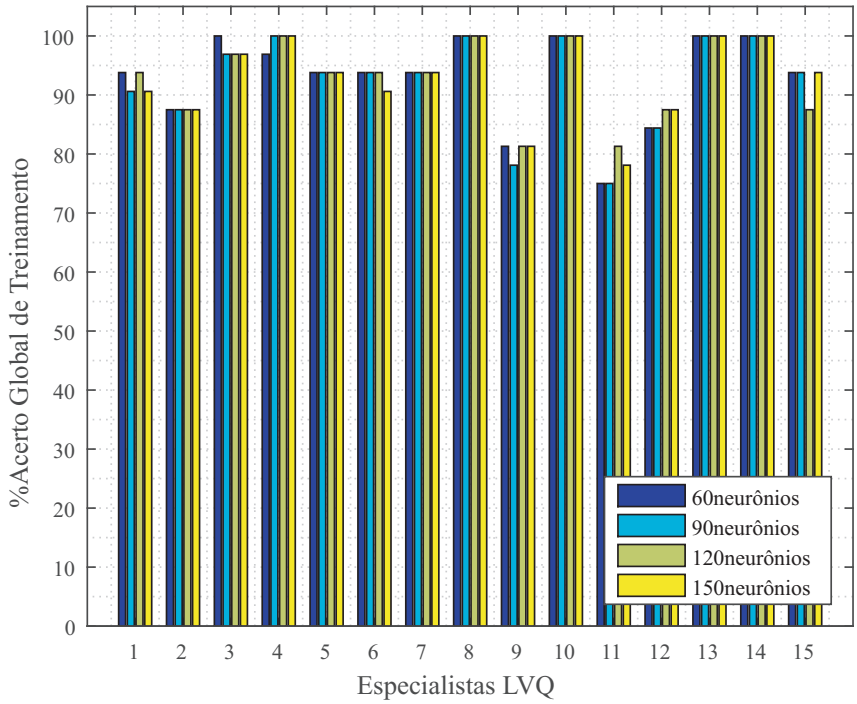
5.1 Resultados LVQ

5.1.1 Treinamento e Validação LVQ

5.1.1.1 1° Experimento: Rede Neural LVQ – 4 entradas

Apresenta-se na Figura 26 o resultado (em porcentagem) de acerto global dos comandos no treinamento e na Figura 27 o resultado de acerto global dos comandos durante a validação em relação ao conjunto n de neurônios simulados para os quinze especialistas. A média da porcentagem de reconhecimento dos dígitos durante o treinamento ficou em $92,88\%$ e para a fase de validação a média de acerto foi de 83.95%.

Figura 26 – LVQ C_4^{jm}:Resultado Global de Acerto de Treinamento



5.1.1.2 2° Experimento: Rede Neural LVQ – 9 entradas

Os resultados de acerto global dos comandos no treinamento e na validação em relação ao conjunto n de neurônios simulados para Ω_{NL}^{Tr}, $N = 9$ é visualizado, respectivamente, na Figura 28 e na Figura 29.

Figura 27 – LVQ C_4^{jm}:Resultado Global de Acerto de Validação

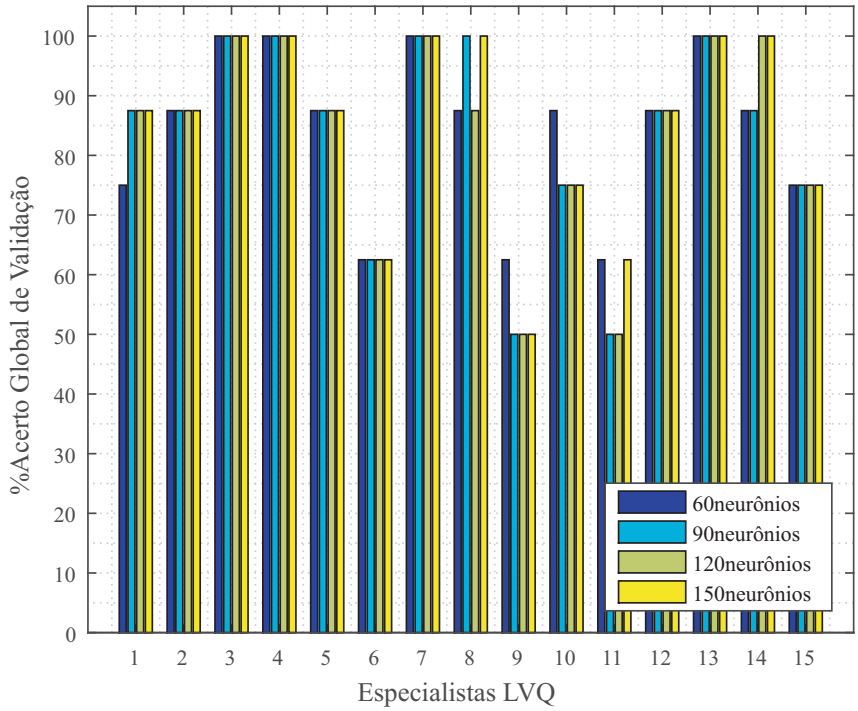


Figura 28 – LVQ C_9^{jm}:Resultado Global de Acerto de Treinamento

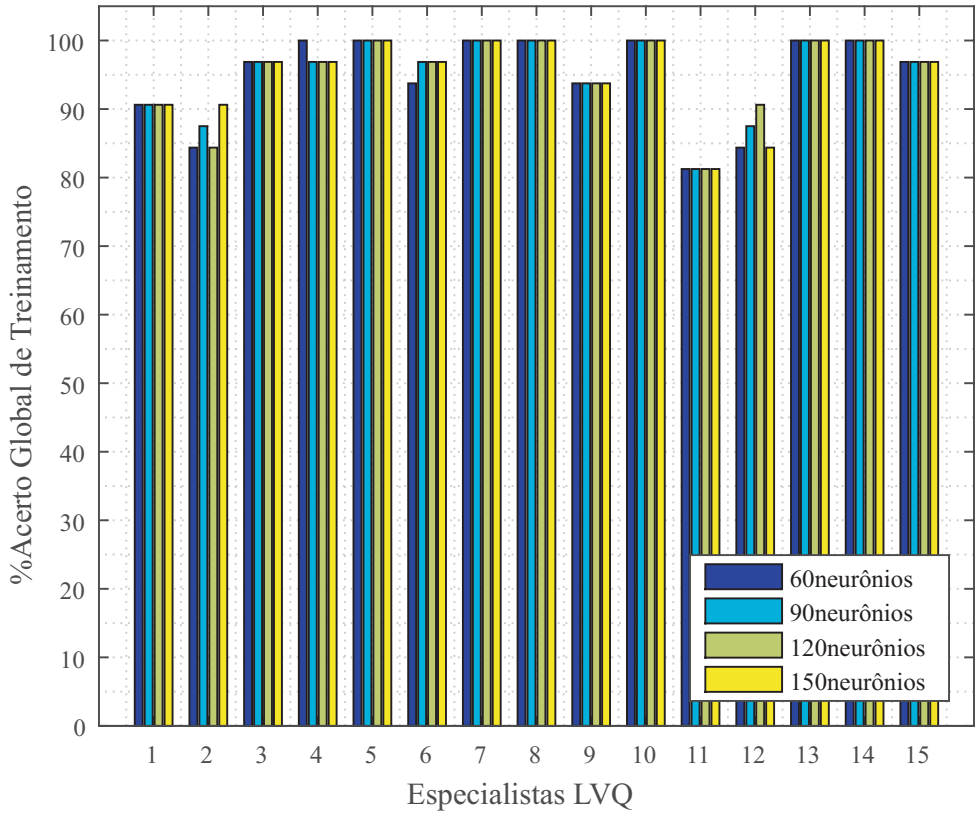
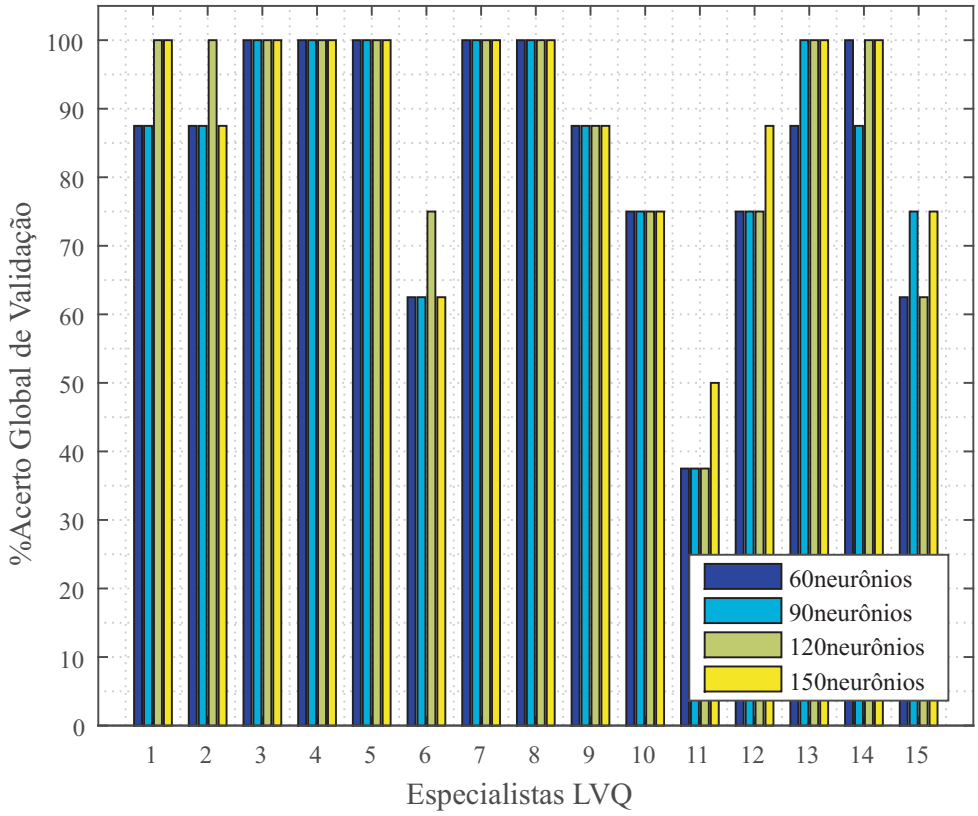


Figura 29 – LVQ C_9^{jm}:Resultado Global de Acerto de Validação



Para este experimento, o resultado de treinamento ficou com média de acerto de 95.10% e a validação com média de acerto de 86.25%. Observa-se na Figura 28 e Figura 29 que, ao aumentar o número de parâmetros dos padrões do conjunto de treinamento original, houve uma incremento na média dos acertos de reconhecimento dos comandos durante o treinamento e validação.

5.1.1.3 3° Experimento: Rede Neural LVQ – 16 entradas

No último experimento realizado, apresentou-se à entrada das redes neurais especialistas LVQ o conjunto de treinamento original Ω_{NL}^{Tr} com $N = 16$. Visualiza-se na Figura 30 e Figura 31, respectivamente, o resultado de acerto global dos comandos no treinamento e na validação em relação ao conjunto n de neurônios simulados. Observa-se que, ao utilizar os padrões C_{16}^{jm}, a média de acerto global no treinamento aumentou em relação aos dois experimentos anteriores, atingindo o valor de 97.5%. O resultado para o acerto médio de validação apresentou valor igual ao segundo experimento, ou seja, o valor médio foi de 91.45%.

Figura 30 – LVQ C_{16}^{jm}:Resultado Global de Acerto de Treinamento

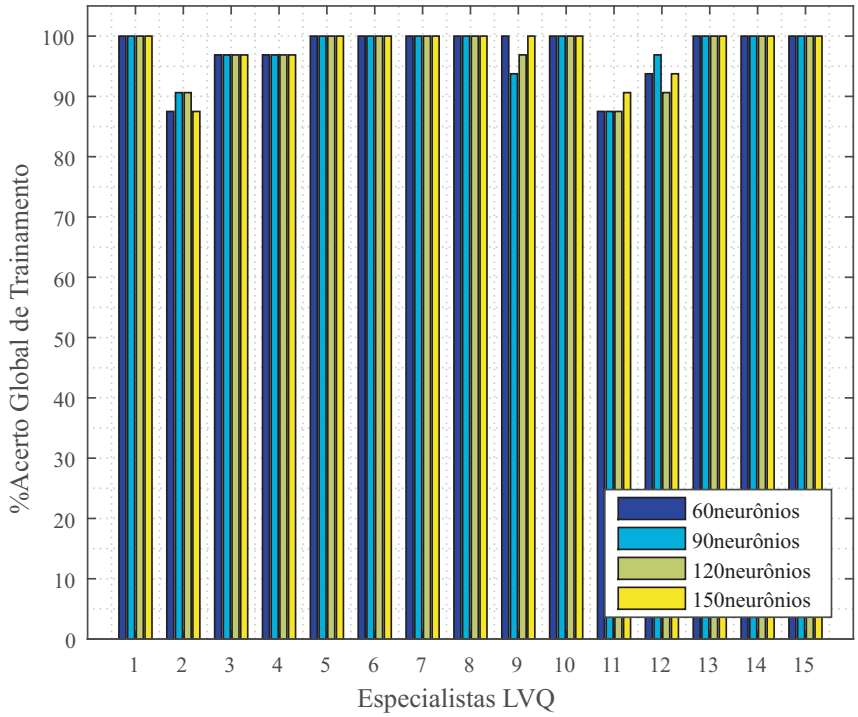
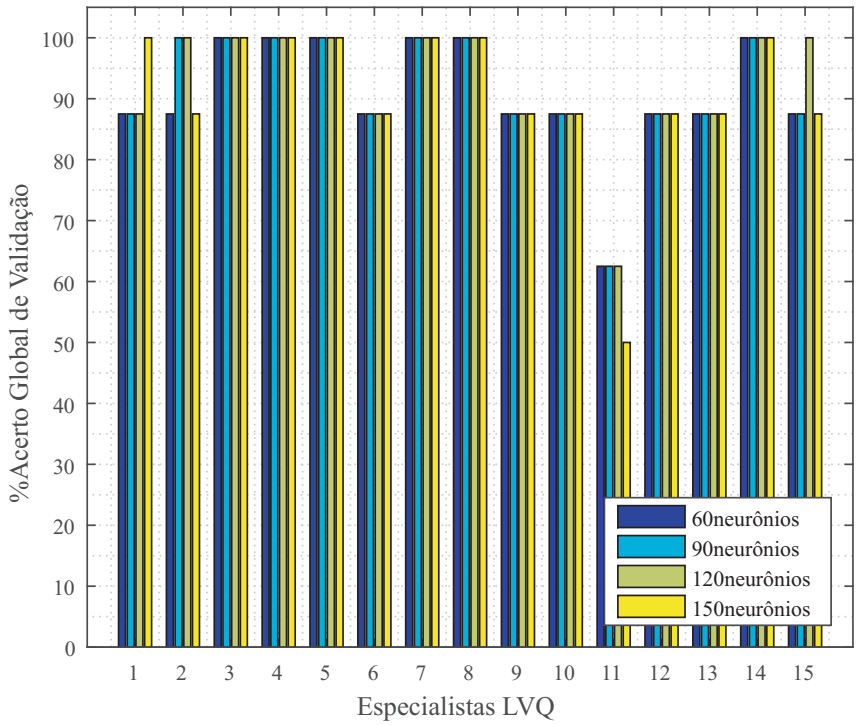


Figura 31 – LVQ C_{16}^{jm}:Resultado Global de Acerto de Validação



5.1.2 Acurácia Local Especialistas LVQ

Após a finalização da etapa de treinamento e validação, selecionou-se, dentre os resultados obtidos nas simulações, a topologia que apresentou os melhores resultados de reconhecimento dos padrões de sinais de voz para cada um dos quinze especialistas.

Diante disto, a partir dos resultados de treinamento e validação, foram testadas, para cada especialista, as topologias que apresentaram maior acerto global de validação. Dessa forma, para a aplicação dos testes, definiu-se a escolha das redes treinadas que apresentaram acerto global de validação acima de 80%.

Pode-se notar pelos resultados apresentados na seção 5.1.1 que poucos especialistas não conseguiram atingir o valor estipulado para o critério de teste. Porém, levou-se em conta também, além do critério do valor do acerto global de validação para a aplicação dos testes, a escolha de um topologia simples com o erro de validação aceitável.

Logo, pelos resultados de treinamento e validação, as redes neurais especialistas LVQ com 60 neurônios na camada competitiva foram escolhidas para a fase de teste individual.

A fase de teste individual tem o objetivo de verificar a capacidade de generalização das redes especialistas para as classes as quais foram treinadas. Com os resultados atingidos nesta etapa, definiu-se um nível de acurácia local para as saídas de cada especialista. A informação de acurácia local será parte integrante das regras de decisão do sistema de reconhecimento de voz baseado em seleção dinâmica das redes neurais especialistas.

Os quesitos estabelecidos para a escolha da melhor topologia foram aplicados para cada experimento realizado na fase de treinamento. Assim, os conjuntos de teste Ω_{N3000}^{TM} e Ω_{N3000}^{TF} com $N = 4, 9, 16$ foram aplicados às topologias nos três testes realizados.

Apresenta-se na Tabela 7 os resultados de classificação global e por classe dos testes individuais aplicados às topologias dos especialistas treinadas que apresentaram acerto global de validação acima de 80% e menor complexidade topológica (60 neurônios) para o conjunto de treinamento utilizando os padrões originais C_4^{jm}, C_9^{jm} e C_{16}^{jm}

5.2 Resultados MLP

5.2.1 Treinamento e Validação MLP

As combinações propostas na Tabela 6 foram simuladas também para a Rede Neural MLP com duas camadas ocultas com a finalidade de verificar a necessidade do incremento do número de camadas ocultas para extrair as características contidas nos padrões de entrada apresentados à rede.

Pelo fato da Rede MLP possuir mais elementos de topologia que podem ser com-

binados para chegar-se ao melhor desempenho, as simulações realizadas para as redes com duas camadas ocultas fornecem embasamento para os resultados apresentados pelas Redes Neurais MLP de uma camada oculta.

Tabela 7 – Teste Individual Especialistas LVQ com 60 neurônios

	C_4^{jm}			C_9^{jm}			C_{16}^{jm}		
RN Especialista	%Acurácia Global	% Acurácia Saída Classe 1	% Acerto Saída Classe 2	%Acurácia Global	% Acerto Saída Classe 1	% Acurácia Saída Classe 2	%Acurácia Global	% Acerto Saída Classe 1	% Acerto Saída Classe 2
1	82,1	85,3	78,9	95,8	93,2	98,4	90,3	85,3	95,3
2	79,5	88,9	70,0	91,6	93,7	89,5	86,1	94,7	77,4
3	99,2	98,9	99,5	99,5	100	98,9	100	100	100
4	91,1	83,2	98,9	94,5	94,2	94,7	94,2	93,2	95,3
5	91,8	91,6	92,1	95,5	95,3	95,8	97,1	94,2	100
6	88,9	85,8	92,1	87,4	93,2	81,6	82,4	87,9	76,88
7	93,2	88,9	97,4	94,2	90	98,4	99,5	100	98,9
8	97,1	96,3	97,9	97,1	95,3	98,9	99,7	99,5	100
9	77,9	70,0	85,8	72,4	71,6	73,2	93,9	96,3	91,6
10	82,9	75,3	90,5	85,8	93,7	77,9	92,4	92,1	92,6
11	75,0	90,0	60,0	73,4	86,8	60	72,1	65,8	78,4
12	82,1	76,3	87,9	73,7	54,2	93,2	73,4	69,5	77,4
13	99,2	98,4	100	99,5	98,9	100	99,7	99,5	100
14	97,1	98,9	95,3	97,6	97,9	97,4	95,8	96,8	94,7
15	75,8	72,1	79,5	67,9	60,5	75,3	71,3	62,1	80,5

Ao final de todas as simulações que combinam os elementos de topologia e algoritmos de treinamento e número de camadas ocultas, pode-se observar o comportamento das topologias propostas e definir o melhor resultado. Verificou-se durante as simulações que os algoritmos GD, GDM e LM não alcançaram bons resultados para o problema de reconhecimento de padrões com a codificação proposta, mostrando resultados globais de treinamento e validação inferior a 50%.

Além disso, as redes MLP treinadas com duas camadas ocultas não apresentaram resultados significativos em relação as redes treinadas com uma camada oculta, o que não justifica o aumento de complexidade da estrutura da rede. Por estes motivos, somente os resultados apresentados pelas redes treinadas com o algoritmo RP com uma camada oculta são apresentados.

5.2.1.1 1° Experimento: Rede Neural MLP – 4 entradas

Visualizam-se nas Figuras 32 e 33 os resultados da média de acerto global para o treinamento e validação para cada especialista, respectivamente, para as topologias treinadas com o algoritmo RP. A média dos acertos globais de treinamento e validação foram relacionados ao conjunto n de neurônios simulados. Como descrito anteriormente, foram simulados 100 treinamentos com diferentes inicializações de pesos para cada combinação dos elementos topológicos, de modo que possa ser analisada estatisticamente. Estes resultados foram obtidos com o treinamento das redes para uma camada oculta.

Figura 32 – MLP C_4^{jm}:Resultado Médio Global de Acerto de Treinamento

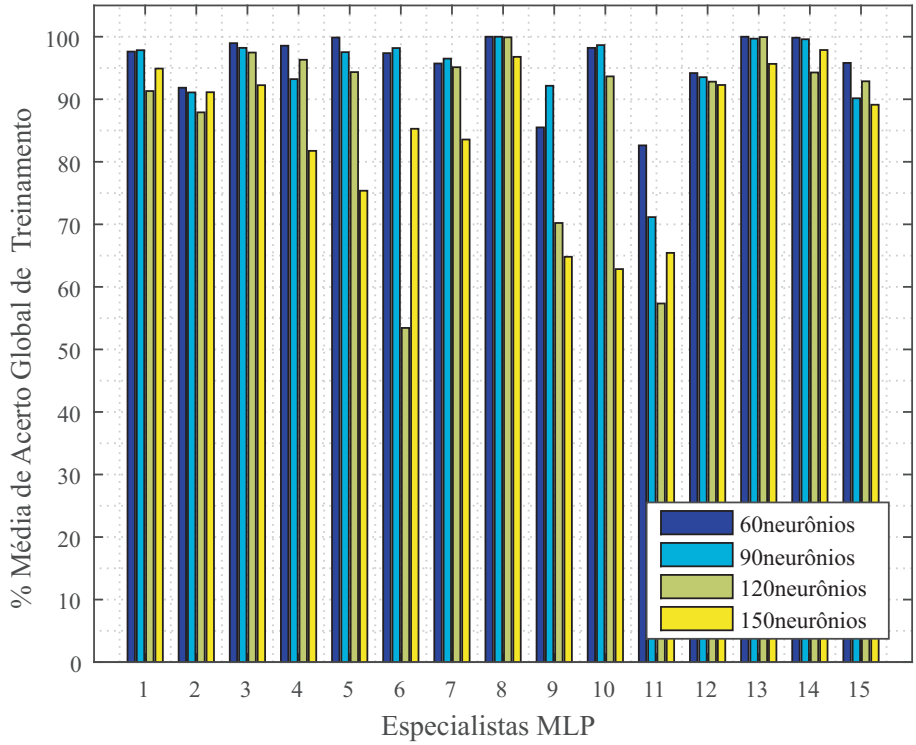
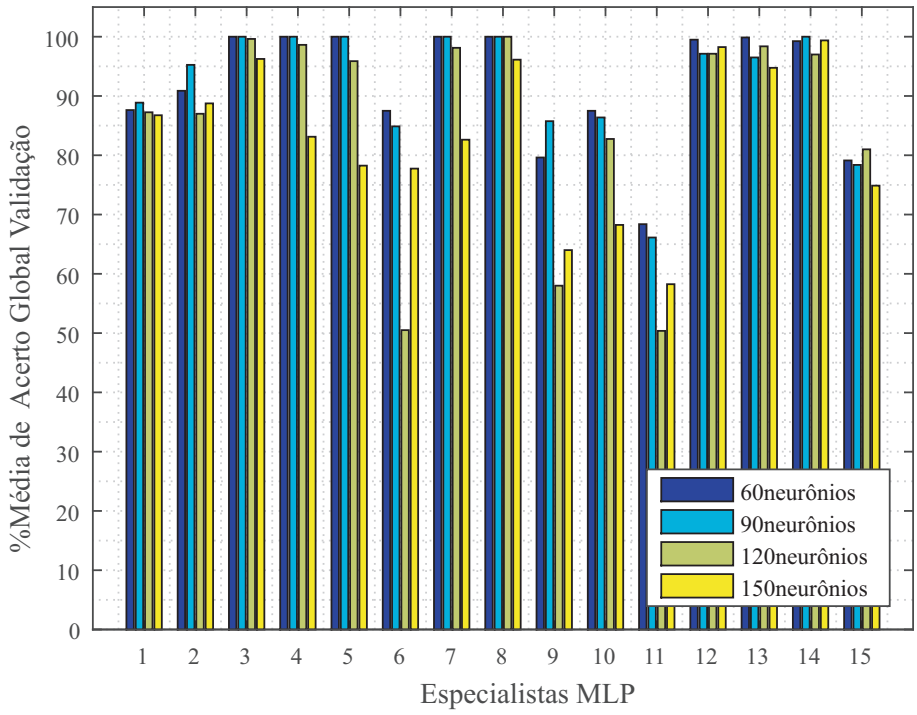


Figura 33 – MLP C_4^{jm}:Resultado Médio Global de Acerto de Validação

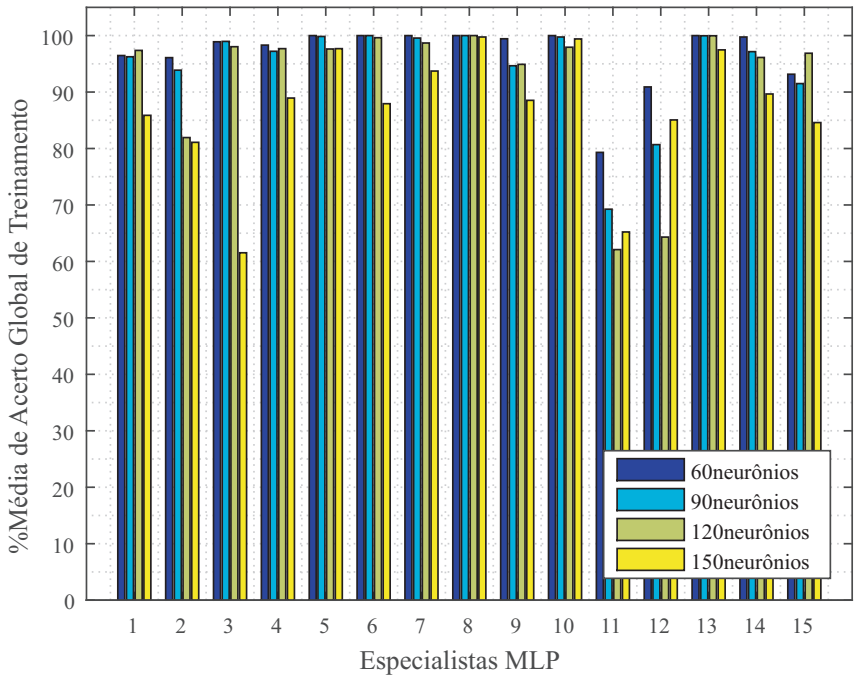


5.2.1.2 2º Experimento -Rede Neural MLP – 9 entradas

Para o segundo experimento para a configuração MLP para os especialistas, utilizou-se como conjunto de treinamento no espaço de baixa dimensionalidade Ω_{NL}^{Tr}, os vetores C_N^{jm} obtidos a partir da matriz temporal bidimensional TCD de ordem 3.

Apresentam-se nas Figura 34 e 35 os resultados médios de acerto global dos comandos no treinamento e validação das redes neurais de uma camada oculta utilizando o algoritmo RP, respectivamente.

Figura 34 – MLP C_9^{jm}:Resultado Médio Global de Acerto de Treinamento



5.2.1.3 3º Experimento -Rede Neural MLP – 16 entradas

No último experimento apresenta-se à entrada das Redes Neurais MLP especialistas o conjunto de treinamento de baixa dimensão Ω_{NL}^{Tr} cujos vetores C_N^{jm} foram gerados a partir da matriz temporal bidimensional TCD de ordem 4. Assim, como o segundo experimento, deseja-se observar o comportamento nos resultados apresentados pelas redes especialistas com o aumento dos parâmetros que constituem o padrão do sinal de voz de baixa dimensão. Assim, todas as topologias propostas foram treinadas com o conjunto de treinamento Ω_{NL}^{Tr}, $N = 16$. Os resultados médios de treinamento e validação para o algoritmo RP são demonstrados, respectivamente, nas Figuras 36 e 37.

Figura 35 – MLP C_9^{jm}:Resultado Médio Global de Acerto de Validação

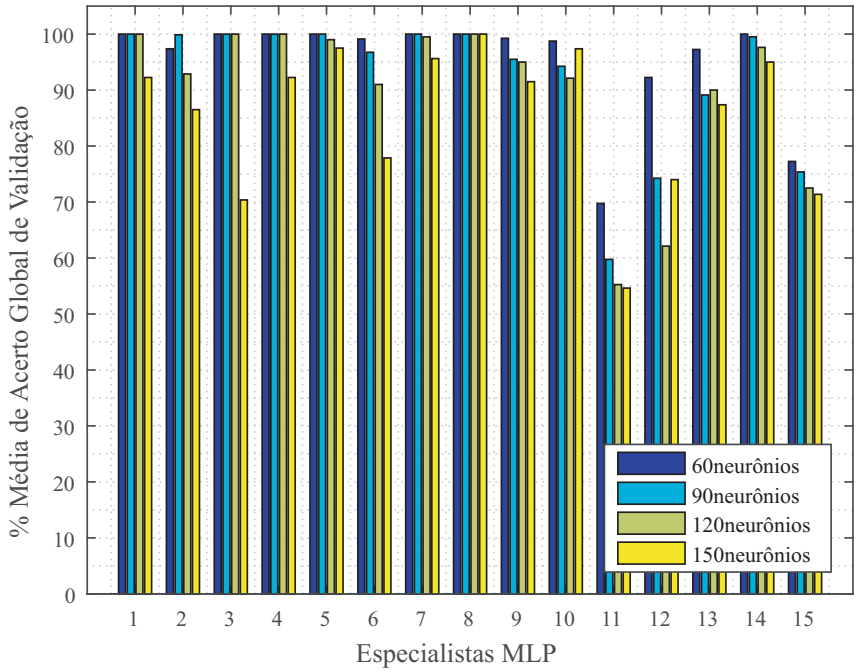


Figura 36 – MLP C_{16}^{jm}:Resultado Médio Global de Acerto de Treinamento

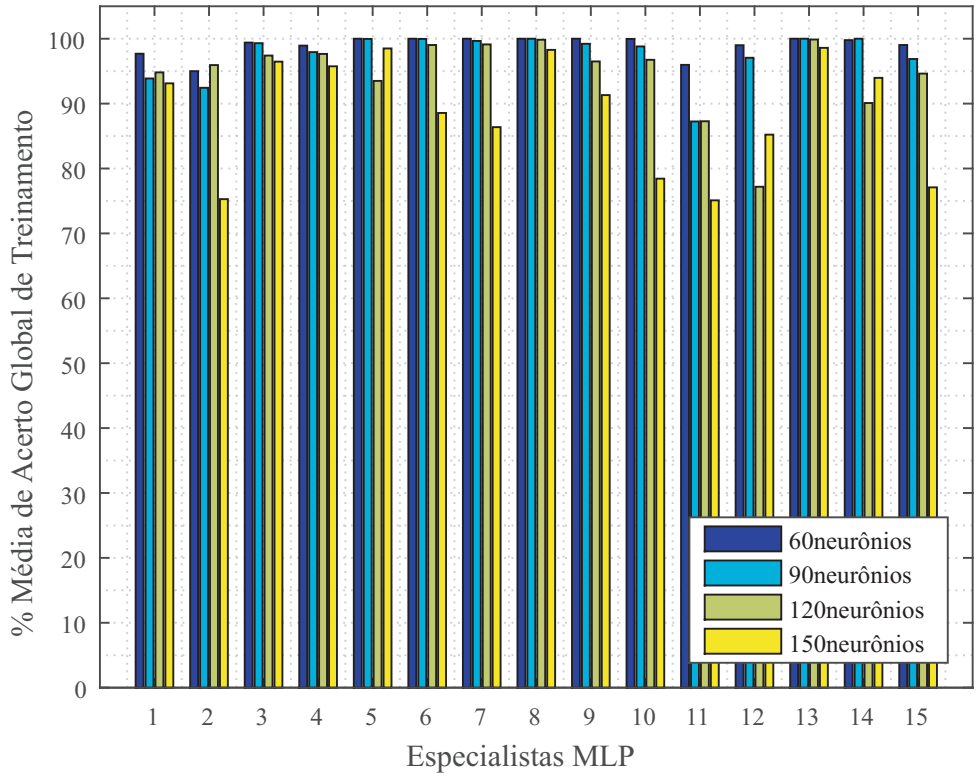
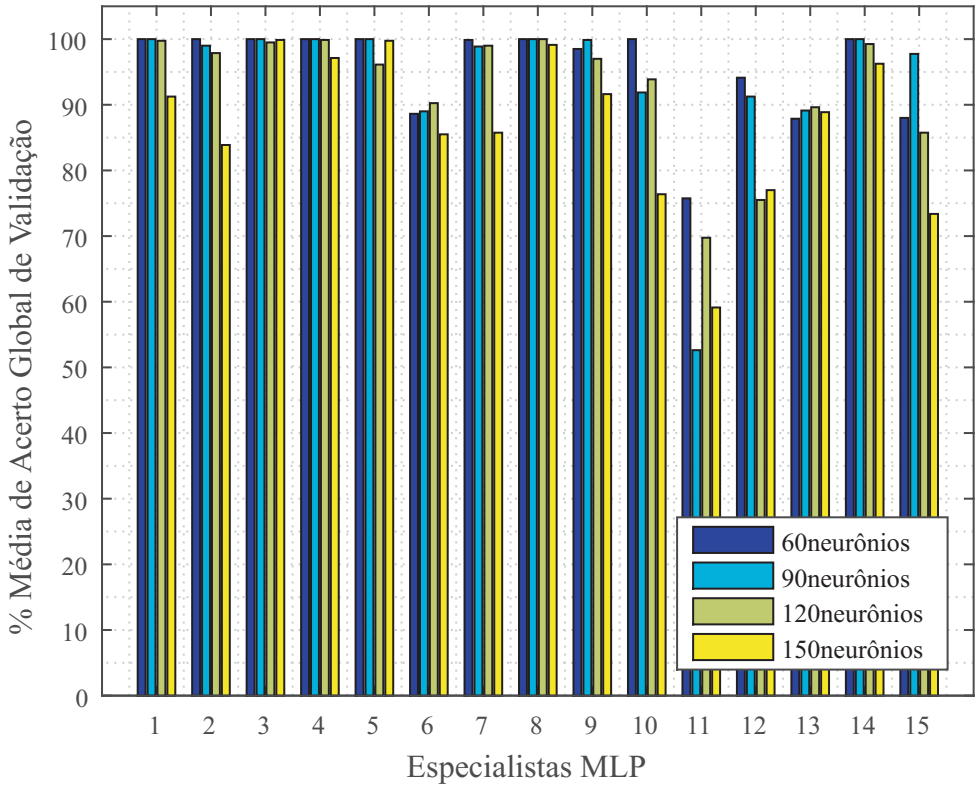


Figura 37 – MLP C_{16}^{jm}:Resultado Médio Global de Acerto de Validação



5.2.2 Acurácia Local Especialistas MLP

Assim como realizado para as Redes Neurais LVQ, após a finalização da etapa de treinamento e validação, selecionou-se, dentre os resultados obtidos nas simulações, as topologias das Redes MLP que apresentaram os melhores resultados de reconhecimento dos padrões de sinais de voz para cada um dos quinze especialistas.

Logo, a partir dos resultados observados no treinamento e validação, foram testadas apenas as topologias que apresentaram resultado de acerto global de validação acima de 80%. Pela avaliação dos resultados apresentados nos três experimentos anteriores, constatou-se que a topologia mais simples proposta para as simulações sempre apresentou resultados iguais ou superiores em relação as topologias mais complexas que também foram simuladas. Assim, o conjunto de topologias a serem levadas em consideração para aplicação dos testes em cada especialista se reduz aquelas topologias treinadas com uma camada oculta e 60 neurônios.

Dessa forma, entre as 100 simulações realizadas para a topologia com 60 neurônios para os 15 especialistas, aquela que apresentou maior acerto global de classificação no teste foi escolhida como a melhor topologia para o problema de reconhecimento de voz para cada especialista.

Logo, os resultados obtidos demonstram a capacidade de generalização das redes

especialistas para as classes as quais foram treinadas, além de determinar os níveis acurácia local para as saídas de cada especialista, como realizado para as redes especialistas LVQ.

Portanto, os conjuntos de teste de baixa ordem Ω_{N3000}^{TM} e Ω_{N3000}^{TF} com $N = 4, 9, 16$ foram mapeados pelo conjunto de FBR, para posterior aplicação às redes e os resultados obtidos para a acurácia global e por classe são apresentados.

Os melhores resultados (em porcentagem) encontrados nos testes executados para cada especialista, considerando as redes treinadas com uma camada oculta de 60 neurônios pelo algoritmo RP utilizando os padrões C_4^{jm}, C_9^{jm} e C_{16}^{jm} podem ser visualizados na Tabela 8.

Tabela 8 – Teste Individual Especialistas MLP com 60 neurônios

	C_4^{jm}			C_9^{jm}			C_{16}^{jm}		
RN Especialista	% Acurácia Global	% Acerto Saída Classe 1	% Acerto Saída Classe 2	% Acurácia Global	% Acerto Saída Classe 1	% Acerto Saída Classe 2	% Acurácia Global	% Acerto Saída Classe 1	% Acerto Saída Classe 2
1	97,6	95,8	99,5	98,4	97,4	99,5	98,7	97,9	99,5
2	80,5	83,2	77,9	87,4	93,7	81,1	86,3	95,8	76,8
3	100	100	100	100	100	100	100	100	100
4	98,7	97,9	99,5	96,3	93,2	99,5	98,7	97,9	99,5
5	96,8	93,7	100	97,1	94,2	100	97,4	94,7	100
6	93,7	87,9	99,5	94,7	95,8	93,7	90,3	87,4	93,2
7	96,9	93,8	100	96,8	98,4	95,3	98,4	98,4	98,4
8	97,6	95,3	100	98,2	96,3	100	97,4	94,7	100
9	81,8	72,1	91,6	90,8	87,4	94,2	98,7	99,5	97,9
10	90	81,6	98,4	93,7	88,4	98,9	92,1	84,2	100
11	76,1	73,2	78,9	85,5	86,3	84,7	90,5	88,9	92,1
12	85,8	76,3	95,3	81,6	82,1	81,1	98,7	98,9	98,4
13	100	100	100	99,7	99,5	100	99,7	99,5	100
14	98,7	97,9	99,5	98,9	97,9	100	99,2	98,4	100
15	77,4	74,2	80	78,2	75,8	80,5	77,6	72,1	83,2

5.3 Seleção Dinâmica das Redes Neurais Especialistas

Finalizada a etapa de projeto dos especialistas, dados pela análise das configurações LVQ e MLP e definido os níveis de acurácia local de classificação para cada saída dos especialistas, realizou-se a integração das funções de base radial modeladas com os parâmetros das classes definidas no sistema e as topologias especialistas com os melhores resultados de classificação.

Assim, apresenta-se na Tabela 9, Tabela 10 e Tabela 11, respectivamente, os resultados de pré-classificação dos padrões de teste do espaço de características original gerados pelas matrizes TCD de ordem 2, 3 e 4. Uma vez que a etapa de pré-seleção é a mesma tanto na arquitetura utilizando as redes especialistas na configuração MLP quanto LVQ, os resultados apresentados são os mesmos para ambas configurações.

Tabela 9 – Pré-classificação Padrões de Teste C_4^{jm}

Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista
“Zero”	“Zero”/ 67,0/RN_Esp1	“Abaixo"	“Dois” / 67,0 / RN_Esp2	“Iniciar”	“Iniciar”/ 56,5 / RN_Esp11
“Um”	“Um” / 45,0 / RN_Esp1	“Abrir”	“Abrir” / 78,5 / RN_Esp6	“Ligar”	“Ligar” / 44,5 /RN_Esp11
“Dois”	“Dois” / 71,5 / RN_Esp 2	“Acima”	“Acima” / 52,5 / RN_Esp7	“Máximo”	“Máximo” / 76,0 /RN_Esp12
“Três”	“Três” / 41,5 / RN_Esp 2	“Aumentar”	“Aumentar”/ 37,5 / RN_Esp7	“Médio”	“Médio” / 41,5 / RN_Esp12
“Quatro”	“Quatro”/ 34 / RN_Esp3	“Desligar”	“Desligar” /‘ 44,5 / RN_Esp8	“Mínimo”	“Mínimo”/ 19,0 / RN_Esp13
“Cinco”	“Cinco” / 78,0 / RN_Esp 3	“Diminuir”	“Sete” / 36,84 / RN_Esp4	“Para Trás”	“Sete”/ 22,0 / RN_Esp4
“Seis”	“Seis” / 72,0 / RN_Esp 4	“Direita”	“Mínimo” / 43,15 / RN_Esp 13	“Para Frente”	“Abrir”/ 33,68 / RN_Esp6
“Sete”	“Sete”/ 77,0 / RN_Esp 4	“Esquerda”	“Esquerda” / 33,5 / RN_Esp9	“Parar”	“Parar”/ 22,5 / RN_Esp 14
“Oito”	“Oito” / 54,5 / RN_Esp 5	“Fechar”	“Fechar” / 43,5 / RN_Esp10	“Repousar”	“Repousar”/ 45,5 / RN_Esp15
“Nove”	“Nove” / 39 / RN_Esp 5	“Finalizar”	“Finalizar” / 26,0 / RN_Esp10	“Salvar”	“Salvar”/ 67,5 / RN_Esp15

Tabela 10 – Pré-classificação Padrões de Teste C_9^{jm}

Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista
“Zero”	“Zero”/ 62,5/RN_Esp1	“Abaixo"	“Abaixo” / 40,5 / RN_Esp6	“Iniciar”	“Iniciar”/ 61,5 / RN_Esp11
“Um”	“Um” / 46,5 / RN_Esp1	“Abrir”	“Abrir” / 79,0 / RN_Esp6	“Ligar”	“Ligar” / 61,0 /RN_Esp11
“Dois”	“Dois” / 80,5 / RN_Esp 2	“Acima”	“Acima” / 63,0 / RN_Esp7	“Máximo”	“Máximo /89,0 /RN_Esp12
“Três”	“Três” / 46,5 / RN_Esp 2	“Aumentar”	“Aumentar”/ 75,5 / RN_Esp7	“Médio”	“Máximo” / 50,0 / RN_Esp12
“Quatro”	“Quatro”/ 52 / RN_Esp3	“Desligar”	“Desligar” /‘ 55,0 / RN_Esp8	“Mínimo”	“Mínimo”/ 27,5 / RN_Esp13
“Cinco”	“Cinco” / 81,5 / RN_Esp 3	“Diminuir”	“Sete” / 55,5 / RN_Esp4	“Para Trás”	“Para Trás”/ 21,0 / RN_Esp13
“Seis”	“Seis” / 69,0 / RN_Esp 4	“Direita”	“Direita” / 27,5 / RN_Esp 9	“Para Frente”	“Para Frente”/ 40,0 / RN_Esp14
“Sete”	“Sete”/ 81,5 / RN_Esp 4	“Esquerda”	“Máximo” / 33,5 / RN_Esp12	“Parar”	“Sete”/ 22,5 / RN_Esp 4
“Oito”	“Oito” / 73,0 / RN_Esp 5	“Fechar”	“Fechar” / 72,5 / RN_Esp10	“Repousar”	“Repousar”/ 44,0 / RN_Esp15
“Nove”	“Nove” / 71,0 / RN_Esp 5	“Finalizar”	“Finalizar” / 52,5 / RN_Esp10	“Salvar”	“Salvar”/ 79,5 / RN_Esp15

A partir dos resultados de pré-classificação demonstrados nas Tabelas 9, 10 e 11 observa-se que esta etapa seleciona, em grande maioria, os especialistas corretos para o segundo nível de classificação. Logo, a taxa de acerto na fase de pré-seleção pelas funções

Tabela 11 – Pré-classificação Padrões de Teste C_{16}^{jm}

Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista	Classe Testada	FBR/ %Max Prob/ RN Especialista
"Zero"	"Zero"/ 80,5/RN_Esp1	"Abaixo"	"Abaixo" / 57,0 / RN_Esp6	"Iniciar"	"Iniciar" / 71,5 / RN_Esp11
"Um"	"Um" / 54,5 / RN_Esp1	"Abrir"	"Abrir" / 78,0 / RN_Esp6	"Ligar"	"Ligar" / 48,5 /RN_Esp11
"Dois"	"Dois" / 87,0 / RN_Esp 2	"Acima"	"Acima" / 61,0 / RN_Esp7	"Máximo"	"Máximo" / 93,0 /RN_Esp12
"Três"	"Três" / 58,0 / RN_Esp 2	"Aumentar"	"Aumentar"/ 70,5 / RN_Esp7	"Médio"	"Máximo" / 40,5 / RN_Esp12
"Quatro"	"Quatro"/ 61,0 / RN_Esp3	"Desligar"	"Desligar" /' 47,0 / RN_Esp8	"Mínimo"	"Mínimo"/ 41,5 / RN_Esp13
"Cinco"	"Cinco" / 85,5 / RN_Esp 3	"Diminuir"	"Sete" / 67,0 / RN_Esp4	"Para Trás"	"Para Trás"/ 35,5 / RN_Esp13
"Seis"	"Seis" / 70,5 / RN_Esp 4	"Direita"	"Direita" / 32,0 / RN_Esp 9	"Para Frente"	"Para Frente"/ 48,0 / RN_Esp14
"Sete"	"Sete"/ 89,5 / RN_Esp 4	"Esquerda"	"Máximo / 56,0 / RN_Esp12	"Parar"	"Sete" / 33,5 / RN_Esp 4
"Oito"	"Oito" / 79,0 / RN_Esp 5	"Fechar"	"Fechar" / 83,0 / RN_Esp10	"Repousar"	"Repousar"/ 62,5 / RN_Esp15
"Nove"	"Nove" / 78,0 / RN_Esp 5	"Finalizar"	"Finalizar" / 68,5 / RN_Esp10	"Salvar"	"Salvar" / 85 / RN_Esp15

de base radial encontram-se, respectivamente, para os padrões de teste no espaço de baixa dimensionalidade C_4^{jm}, C_9^{jm} e C_{16}^{jm} de 83.33%, 86.33% e 86, 33%.

Para as classes que apresentaram erro na fase de pré-seleção, o algoritmo contornou este problema na fase de tomada de decisão a partir dos resultados obtidos nas saídas dos especialistas comparados aos valores de acurácia local.

Após a análise de desempenho entre as configurações MLP e LVQ para composição do conjunto de especialistas, observa-se na Figura 38, Figura 39 e Figura 40 a comparação entre ambas configurações utilizando os padrões C_4^{jm}, C_9^{jm} e C_{16}^{jm}, respectivamente, no teste final do sistema de reconhecimento de voz baseado em seleção dinâmica de redes neurais especialistas.

5.4 Análise dos resultados experimentais

Ao final da realização dos experimentos propostos e análises dos resultados obtidos relatados neste trabalho, apontam-se as seguintes considerações:

1. Por meio destes resultados, certificou-se a eficiência da parametrização bidimensional através dos coeficientes mel-cepstrais e da TCD no modelamento das variações locais e globais do sinal de voz, fornecendo assim, padrões adequados e reduzidos,

Figura 38 – C_4^{jm}: Comparação entre o teste final utilizando os especialistas MLP e LVQ

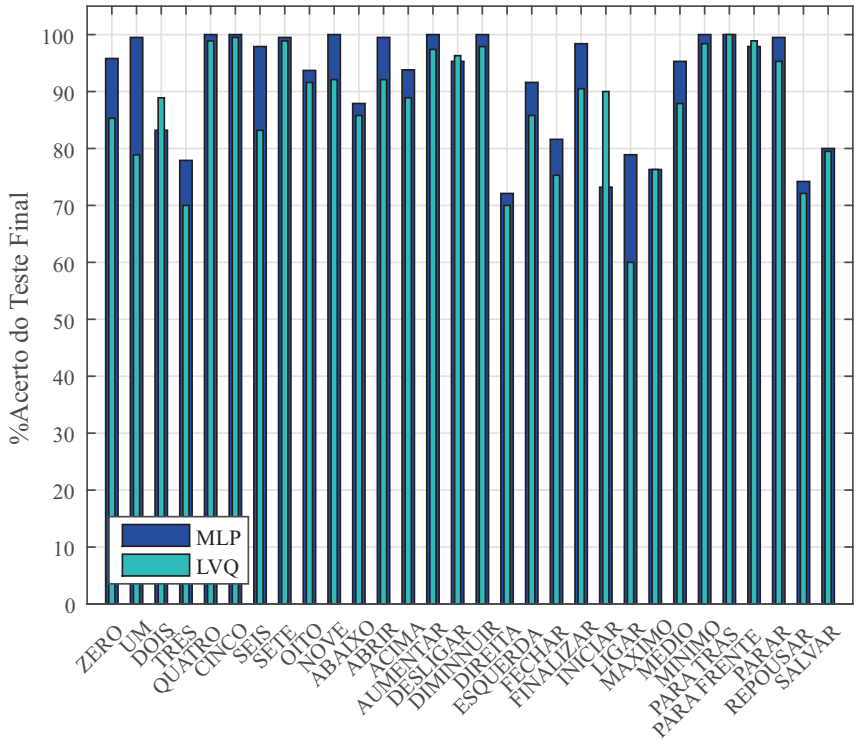


Figura 39 – C_9^{jm}: Comparação entre o teste final utilizando os especialistas MLP e LVQ

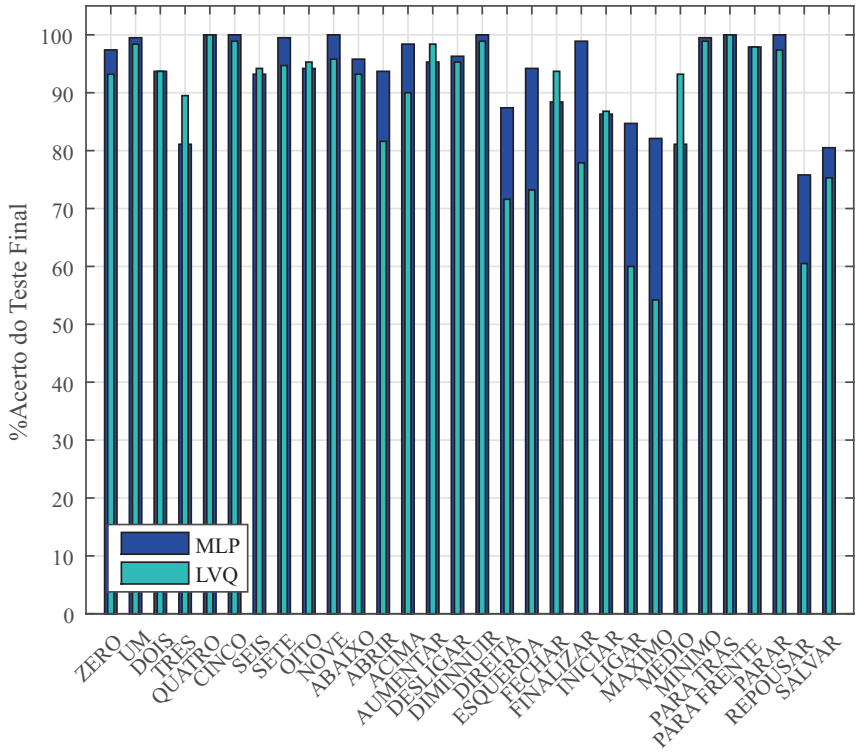
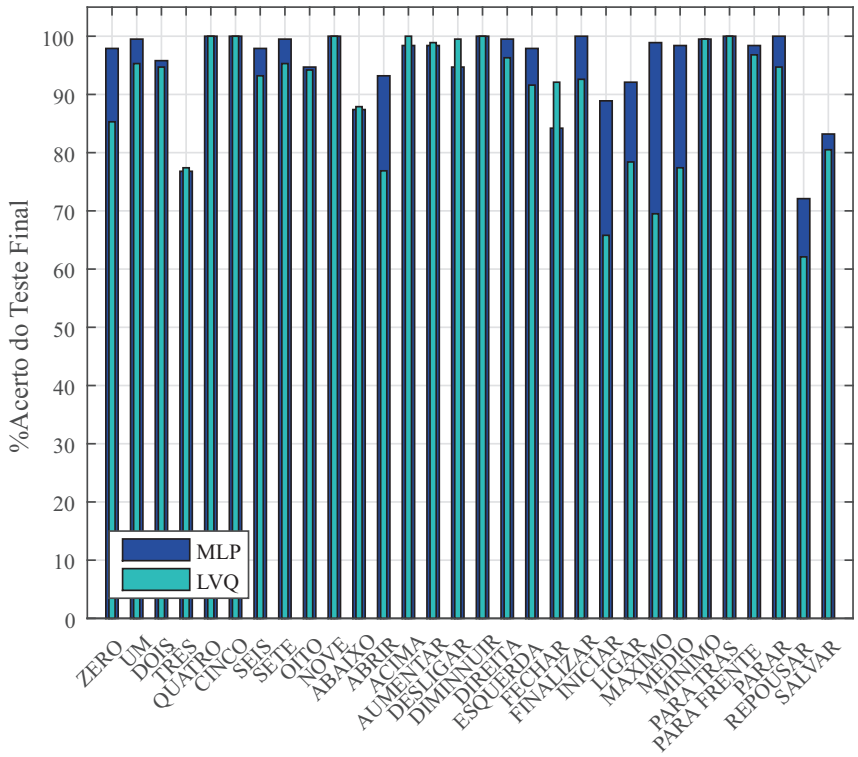


Figura 40 – C_{16}^{jm}: Comparação entre o teste final utilizando os especialistas MLP e LVQ



que condensam os elementos essenciais para o bom desempenho do classificador. Estes padrões constituíram o espaço de características de baixa dimensionalidade.

2. A utilização de um conjunto de funções de base radial gaussianas modeladas com as características de centroide e variância das 30 classes do sistema de reconhecimento permitiu a transformação adequada dos padrões TCD em um espaço de alta dimensionalidade, que possibilitou a superação do problema de separabilidade dos padrões, justificado pelo Teorema de Cover.

3. A divisão dos padrões no espaço de alta dimensionalidade em subespaços possibilitou a utilização de classificadores baseados em redes neurais mais simples, chamados de especialistas, que possuem uma estrutura topológica com um número de parâmetros ajustáveis mais simples do que se fosse utilizado uma única estrutura. O conjunto de 15 especialistas para o sistema de classificação permitiu a diminuição no tempo de treinamento e o sobreajuste dos dados pelas redes neurais.

4. A fase de treinamento e validação das redes especialistas MLP e LVQ nos três experimentos realizados demonstraram que não houve aumento significativo no acerto global com o incremento do número de neurônios na camada oculta. Para o caso das Redes MLP, constatou-se que o aumento no número de camadas ocultas não potencializou o resultado no reconhecimento, mostrando assim, que as redes conse-

guiram com um número reduzido de estruturas topológicas extrair as características específicas dos padrões sinais de voz mapeados pelas FBRG para um espaço de alta dimensionalidade apresentados.

5. Verificou-se a influência da inicialização dos pesos sobre os resultados alcançados pelas Redes MLP. A determinação de um conjunto adequado de pesos para inicializar o processo de treinamento, permite com que a rede convirja rapidamente e não seja direcionada a um mínimo local de maior valor entre os mínimos existentes na superfície de erro. Assim, é possível alcançar resultados satisfatórios em relação ao tempo de treinamento e também de generalização. Para cada combinação topológica proposta para as redes neurais especialistas MLP, realizou-se 100 treinamentos com distintos conjuntos de pesos iniciais com o intuito de encontrar o melhor resultado na superfície de erro.

6. Observou-se que, apesar dos padrões obtidos pelas matrizes temporais bidimensionais TCD constituírem uma representatividade adequadas das classes do sistema de reconhecimento, o incremento do número de parâmetros no espaço de baixa dimensionalidade não agregou informação significativa no espaço 30-dimensional, o que pode ser verificado pelos resultados de treinamento, validação e teste realizados.

7. A integração das funções de base radial com o conjunto de especialistas dados pelas configurações MLP e LVQ proporcionou um sistema de reconhecimento de padrões de sinais de voz simples e com elevada eficiência. A pré-seleção realizada pelas FBRG's possui uma taxa de erro baixa na seleção do especialista em relação ao número de classes do problema. A regra de decisão para a classificação na saída do sistema dado pela acurácia local estabelecido para cada classe do problema permitiu que os erros cometidos na etapa de pré-seleção fossem superados e a correta classificação fosse determinada.

8. Por meio de todos os testes realizados, constatou-se que a Rede LVQ pode ser utilizada satisfatoriamente em problemas de reconhecimento de padrões, especificamente a codificação do sinal de voz com um número reduzido de parâmetros proposto neste trabalho. Isto fica evidenciado pelo desempenho muito próximo da Rede MLP, cujo uso em diversas aplicações, dentre elas, classificação de padrões já é consagrada entre as pesquisas que abordam reconhecimento de voz. Assim, os resultados de acurácia global obtidos paras as redes especialistas LVQ utilizando os padrões originais C_4^{jm}, C_9^{jm} e C_{16}^{jm} foram, respectivamente, 87.52%, 88.39% e 89.6%. As redes MLP obtiveram um resultado superior, atingindo para os testes utilizando os três padrões, respectivamente, 91.44%, 93.15% e 94.9%.

6 Considerações Finais

Neste trabalho, propôs-se uma arquitetura hierarquizada utilizando a integração entre funções de base radial gaussianas e um conjunto de redes neurais especialistas como estratégia para o desenvolvimento do classificador em um sistema de reconhecimento de padrões de sinais de voz representados por um grupo de trinta comandos na língua portuguesa brasileira.

O desempenho dos sistemas de reconhecimento de voz depende tanto do classificador utilizado quanto do tipo de codificação realizada para a geração dos parâmetros que representam os modelos dos sinais de voz a serem classificados. Portanto, a metodologia desenvolvida supera a dificuldade encontrada inicialmente para o reconhecimento de múltiplas classes de palavras (acima de 10 classes). A associação da codificação eficiente do sinal de voz através da matriz temporal bidimensional TCD mapeada para um espaço não linear de alta dimensionalidade através de funções de base radial gaussianas (FBRG) e a hierarquia estabelecida pelas mesmas FBRG para seleção entre as redes neurais especialistas proporciona um sistema de reconhecimento de voz de alto desempenho.

Ao longo do desenvolvimento da pesquisa, os objetivos específicos traçados foram realizados de maneira satisfatória, validando assim, os resultados apresentados e atingindo o objetivo geral proposto neste trabalho.

6.1 Conclusões

Diante dos resultados apresentados, concluiu-se que a estratégia adotada para o desenvolvimento de um sistema de reconhecimento de voz utilizando a hierarquia estabelecida entre as funções de base radial, adequadamente modeladas com as características de cada classe envolvida no sistema, e o conjunto de redes neurais treinadas com específicas partes do espaço de características mostrou-se eficiente na discriminação das classes representadas pelas 30 palavras de comandos a serem reconhecidas.

Em face dos resultados apresentados, verifica-se que a parametrização do sinal de voz através da geração da matriz temporal bidimensional TCD provou ser eficiente na formação do conjunto de padrões de entrada. Estes padrões tiveram sua dimensionalidade modificada através de um conjunto de funções de base radial gaussianas parametrizadas com centroides e variâncias das classes envolvidas na tarefa multiclasse. Assim, neste novo espaço de alta dimensionalidade, os padrões são apresentados às redes neurais especialistas durante a fase de treinamento, validação e teste.

Observou-se que apesar do pequeno número de parâmetros que constituem o pa-

drão do sinal de voz, a matriz temporal bidimensional representa as variações de longo-prazo do envelope espectral das locuções a serem identificadas e estas características são reproduzidas no espaço multidimensional proposto.

A versatilidade do conjunto de FBRG proposta na estrutura do sistema de reconhecimento demonstra o potencial destas funções. Enfatiza-se que os parâmetros dos modelos da FBRG foram adequadamente determinados, uma vez que porcentagem de acerto na etapa de pré-seleção foi maior que 80%.

Através da análise de desempenho entre as duas configurações estudadas para compor o conjunto de redes neurais especialistas, responsáveis pela classificação final na etapa de teste, constatou-se que o incremento no número de nerônios das camadas das redes MLP e LVQ não apresentou melhora significante no acerto de validação global, na qual foi o critério utilizado para selecionar as melhores topologias para aplicação dos testes.

Baseando-se nos testes realizados, verificou-se que a rede LVQ pode ser usada de forma satisfatória em problemas de reconhecimento de padrões, especificamente para o sistema de reconhecimento de voz proposto neste trabalho. Isto é evidenciado pelo similar desempenho da rede MLP, que é largamente usada em classificação de padrões, conforme os resultados de acurácia global das classes obtidos pela LVQ: 87.52%, 88.39% e 89.6% respectivamente para os padrões C_4^{jm}, C_9^{jm} e C_{16}^{jm}. As redes MLP obtiveram os seguintes resultados: 91.44%, 93.15% e 94.9%, respectivamente para os mesmos padrões. .

Finalmente, destaca-se o desempenho na tarefa multiclasse de padrões de sinais de voz representado pela arquitetura composta por funções de base radial e redes neurais especializadas em distintas regiões do espaço de características. Esta abordagem permitiu que o sistema de reconhecimento de voz pudesse ser aplicado a um banco maior de palavras, uma vez que uma única estrutura de rede neural não foi capaz de gerar os modelos que representassem cada uma das classes pré-definidas. Logo, esta arquitetura possibilita o aumento do vocabulário a ser utilizado no sistema, proporcionando flexibilidade ao usuário.

6.2 Contribuições

Portanto, mediante as conclusões demonstradas, este trabalho apresenta como contribuições da metodologia desenvolvida os seguintes itens:

1. Integração entre Funções de Base Radial Gaussianas e um conjunto de redes neurais especialistas para compor um sistema de classificação hierarquizado para aplicação em sistemas de reconhecimento de voz. Cada função de base radial gaussiana tem seu centroide e variância associada a uma das 30 classes dadas pelas locuções dos

comandos. Então,quando um novo padrão é apresentado ao sistema, as FBRG fazem uma pré-seleção, indicando através da resposta dos campos receptivos das funções, a classe ao qual aquele padrão encontra-se mais próximo. Após isto, a rede neural especialista que foi treinada com a classe indicada fornece o resultado final de classificação.

2. Arquitetura hierarquizada e flexível ao aumento de vocabulário para sistemas de reconhecimento de padrões de sinais de voz na língua Portuguesa Brasileira com elevada acurácia para a maioria das classes a serem identificadas.

3. Simplificada estrutura que permite aplicações em sistemas de reconhecimento de voz embarcados.

6.3 Propostas Futuras

Portanto, devido ao potencial que a metodologia desenvolvida possui na área de reconhecimento de sinais de voz, como propostas futuras para o aperfeiçoamento do trabalho, têm-se as seguintes sugestões:

1. Utilizar técnicas de otimização bio-inspiradas, como o algoritmo genético, enxame de partículas, dentre outros, para obter melhores parâmetros das funções de base radial gaussianas e comparar com os resultados obtidos através do *k-means* para verificar se há um incremento no desempenho da pré seleção para aquelas funções que não direcionaram a princípio para a rede neural especialista correta.

2. Análise exploratória da complexidade do espaço de características para melhor definir as regiões de competência dos especialistas.

3. Composição heterogênea do conjunto de múltiplos classificadores utilizando as topologias de redes MLP e LVQ, verificando aquelas que apresentam os melhores resultados para cada classe.

4. Desenvolvimento em hardware com processador de sinais.

Referências

ABDELHAMID, A.; ABDULLA, W. Uml-based robotic speech recognition development: A case study. In: *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*. [S.l.: s.n.], 2013. p. 1–10. Citado na página 27.

ALSHU'EILI, H.; GUPTA, G. S.; MUKHOPADHYAY, S. Voice recognition based wireless home automation system. In: *Mechatronics (ICOM), 2011 4th International Conference On*. [S.l.: s.n.], 2011. p. 1–6. Citado na página 27.

ANDERSON, J. *An Introduction to Neural Networks*. [S.l.]: MIT Press, 1995. (A Bradford book). Citado 2 vezes nas páginas 47 e 52.

BALAGANESH, M. et al. Robotic arm showing writing skills by speech recognition. In: *Emerging Trends in Robotics and Communication Technologies (INTERACT), 2010 International Conference on*. [S.l.: s.n.], 2010. p. 12–15. Citado na página 27.

BELLEGARDA, J. R.; MONZ, C. State of the art in statistical methods for language and speech processing. *Computer Speech and Language*, 2016. v. 35, p. 163 – 184, 2016. Citado 2 vezes nas páginas 19 e 23.

BENESTY, J.; SONDHI, M.; HUANG, Y. *Springer Handbook of Speech Processing*. [S.l.]: Springer Berlin Heidelberg, 2007. (Springer Handbook of Speech Processing). Citado 4 vezes nas páginas 31, 33, 35 e 40.

BHOWMIK, T.; CHOWDHURY, A.; MANDAL, S. K. D. Deep neural network based place and manner of articulation detection and classification for bengali continuous speech. *Procedia Computer Science*, 2018. v. 125, p. 895 – 901, 2018. Citado na página 28.

BISHOP, C. *Neural Networks for Pattern Recognition*. [S.l.]: Clarendon Press, 1995. (Advanced Texts in Econometrics). Citado 3 vezes nas páginas 23, 45 e 78.

BRAGA, A. de P. *Redes neurais artificiais: teoria e aplicações*. [S.l.]: LTC Editora, 2007. Citado 6 vezes nas páginas 47, 48, 51, 52, 53 e 61.

BRESOLIN, A. de A. *Reconhecimento de voz através de unidades menores do que a palavra, utilizando Wavelet Packet e SVM, em uma nova estrutura hierárquica de decisão*. Tese (Doutorado) — Universidade Federal do Rio Grande do Norte, Natal, 12 2008. Citado 2 vezes nas páginas 21 e 22.

BRITTO, A. S.; SABOURIN, R.; OLIVEIRA, L. E. Dynamic selection of classifiers—a comprehensive review. *Pattern Recognition*, 2014. v. 47, n. 11, p. 3665 – 3680, 2014. Citado na página 79.

BUHMANN, M. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, 2003. (Cambridge Monographs on Applied and Computational Mathematics). ISBN 9781139435246. Disponível em: <https://books.google.com.br-/books?id=TRMf53opzlsC>. Citado 2 vezes nas páginas 45 e 46.

CAI, M.; LIU, J. Maxout neurons for deep convolutional and lstm neural networks in speech recognition. *Speech Communication*, 2016. v. 77, p. 53 – 64, 2016. Citado na página 24.

CARDOSO, S. A. et al. Sesame: sistema de reconhecimento de comandos de voz utilizando pds e rna. In: *Anais do XVIII Congresso Brasileiro de Automática*. [S.l.: s.n.], 2010. p. 1316–1323. Citado na página 27.

CHADLI, M.; BOUOUDEN, S.; ZELINKA, I. *Recent Advances in Electrical Engineering and Control Applications*. [S.l.]: Springer International Publishing, 2016. (Lecture Notes in Electrical Engineering). Citado na página 19.

COVER, T. M. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, 1965. EC-14, n. 3, p. 326–334, June 1965. ISSN 0367-7508. Citado na página 46.

CUBUKCU, A. et al. Development of a voice-controlled home automation using zigbee module. In: *Signal Processing and Communications Applications Conference (SIU), 2015 23th*. [S.l.: s.n.], 2015. p. 1801–1804. Citado na página 27.

DACHAPAK, C. et al. Orthogonal least squares for radial basis function network in reproducing kernel hilbert space. *IFAC Proceedings Volumes*, 2004. v. 37, n. 12, p. 847 – 852, 2004. Citado na página 46.

DAVIS, S.; MERMELSTEIN, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 1980. v. 28, n. 4, p. 357–366, Aug 1980. Citado na página 41.

DEBATIN, L.; HAENDCHEN, A.; DAZZI, R. L. S. O problema do reconhecimento de voz offline em dispositivos móveis: em busca de uma abordagem racional. In: *XXIII Simpósio Brasileiro de Sistemas Multimídia e Web: Workshops e Pôsteres*. [S.l.: s.n.], 2017. Citado na página 27.

DELLER, J.; HANSEN, J.; PROAKIS, J. *Discrete-Time Processing of Speech Signals*. [S.l.]: Wiley, 2000. (An IEEE Press classic reissue). Citado 4 vezes nas páginas 22, 31, 32 e 37.

DIDACI, L. et al. A study on the performances of dynamic classifier selection based on local accuracy estimation. *Pattern Recognition*, 2005. v. 38, n. 11, p. 2188 – 2191, 2005. Citado na página 79.

DOUGHERTY, G. *Pattern Recognition and Classification: An Introduction*. [S.l.]: Springer New York, 2012. (SpringerLink : Bücher). Citado na página 18.

DREYFUS, G. *Neural Networks: Methodology and Applications*. [S.l.]: Springer Berlin Heidelberg, 2005. Citado na página 23.

DUDA, R.; HART, P.; STORK, D. *Pattern classification*. [S.l.]: Wiley, 2001. (Pattern Classification and Scene Analysis: Pattern Classification). Citado na página 18.

FAUSETT, L. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. [S.l.]: Prentice-Hall, 1994. (Prentice-Hall international editions). Citado 4 vezes nas páginas 47, 48, 64 e 65.

FERREIRA, M. R. P. *Análise discriminante clássica e de núcleo:avaliações e algumas contribuições relativas aos métodos boosting e bootstrap*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, Recife, 2 2007. Citado na página 22.

FISSORE, L.; LAFACE, P.; RAVERA, F. Using word temporal structure in hmm speech recognition. In: *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*. [S.l.: s.n.], 1997. v. 2, p. 975–978. Citado 2 vezes nas páginas 68 e 74.

FLANAGAN, J. *Speech Analysis Synthesis and Perception*. [S.l.]: Springer Berlin Heidelberg, 2013. (Communication and Cybernetics). Citado 2 vezes nas páginas 31 e 33.

FURUI, S. *Advances in Speech Signal Processing*. [S.l.]: Taylor & Francis, 1991. (Electrical and Computer Engineering). Citado na página 32.

FURUI, S. *Digital Speech Processing: Synthesis, and Recognition, Second Edition,*. [S.l.]: Taylor & Francis, 2000. (Signal Processing and Communications). Citado 6 vezes nas páginas 22, 31, 32, 33, 35 e 38.

GHOSH, J.; NAG, A. An overview of radial basis function networks. In: _____. *Radial Basis Function Networks 2: New Advances in Design*. Heidelberg: Physica-Verlag HD, 2001. p. 1–36. Citado na página 46.

GIACINTO, G.; ROLI, F. Methods for dynamic classifier selection. In: *Proceedings 10th International Conference on Image Analysis and Processing*. [S.l.: s.n.], 1999. p. 659–664. Citado na página 79.

GNANASEKAR, A.; JAYAVELU, P.; NAGARAJAN, V. Speech recognition based wireless automation of home loads with fault identification for physically challenged. In: *Communications and Signal Processing (ICCSP), 2012 International Conference on*. [S.l.: s.n.], 2012. p. 128–132. Citado na página 27.

GURNEY, K. *An Introduction to Neural Networks*. [S.l.]: Taylor & Francis, 2003. Citado 2 vezes nas páginas 54 e 63.

HAGAN, M. et al. *Neural Network Design (2nd Edition)*. [S.l.]: Martin Hagan, 2014. Citado 2 vezes nas páginas 64 e 65.

HAYKIN, S. *Redes Neurais*. [S.l.]: BOOKMAN COMPANHIA ED, 2001. Citado 5 vezes nas páginas 48, 49, 53, 62 e 80.

HAYKIN, S. *Neural Networks and Learning Machines*. [S.l.]: Prentice Hall, 2009. (Neural networks and learning machines, 10). Citado 11 vezes nas páginas 20, 45, 46, 52, 53, 54, 56, 58, 65, 67 e 81.

HINTON, G. et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 2012. v. 29, n. 6, p. 82–97, Nov 2012. Citado na página 24.

HUA, Z.; NG, W. L. Speech recognition interface design for in-vehicle system. In: *Proceedings of the 2Nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. New York, NY, USA: ACM, 2010. (AutomotiveUI '10), p. 29–33. Citado na página 27.

HUSNJAK, S.; PERAKOVIC, D.; JOVOVIC, I. Possibilities of using speech recognition systems of smart terminal devices in traffic environment. *Procedia Engineering*, 2014. v. 69, p. 778 – 787, 2014. 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013. Citado na página 19.

JAYASUMANA, S. et al. Kernel methods on riemannian manifolds with gaussian rbf kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015. v. 37, n. 12, p. 2464–2477, Dec 2015. Citado na página 45.

JUANG, B.-H.; RABINER, L.; WILPON, J. On the use of bandpass liftering in speech recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 1987. v. 35, n. 7, p. 947–954, Jul 1987. Citado na página 41.

KATAGIRI, S. *Handbook of Neural Networks for Speech Processing*. [S.l.]: Artech House, 2000. (Artech House signal processing library). Citado 3 vezes nas páginas 22, 64 e 80.

KAUTZ, T.; ESKOFIER, B. M.; PASLUOSTA, C. F. Generic performance measure for multiclass-classifiers. *Pattern Recognition*, 2017. v. 68, p. 111 – 125, 2017. Citado na página 19.

KEARNS, M. A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split. *Neural Computation*, 1997. v. 9, n. 5, p. 1143–1161, Jul 1997. Citado na página 62.

KHERADPISHEH, S. R. et al. Mixture of feature specified experts. *Information Fusion*, 2014. v. 20, p. 242 – 251, 2014. Citado na página 20.

KLEINA, N. *Reconhecimento de voz da Google tem só 8% de erro e não para de melhorar*. 2015. Disponível em: <http://www.tecmundo.com.br/google-i-o-2015-/80678-reconhecimento-voz-google-tem-so-8-erro-nao-de-melhorar.htm>. Citado na página 25.

KOO, Y.-M. et al. An intelligent motion control of two wheel driving robot based voice recognition. In: *Control, Automation and Systems (ICCAS), 2014 14th International Conference on*. [S.l.: s.n.], 2014. p. 313–315. Citado na página 27.

KRÖSE, B.; SMAGT, P. van der. *An Introduction to Neural Networks*. [S.l.]: University of Amsterdam, 1996. Citado 2 vezes nas páginas 59 e 63.

KUNCHEVA, L. *Combining Pattern Classifiers: Methods and Algorithms*. [S.l.]: Wiley, 2014. Citado na página 67.

KUNCHEVA, L. I. Clustering-and-selection model for classifier combination. In: *Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on*. [S.l.: s.n.], 2000. v. 1, p. 185–188 vol.1. Citado na página 79.

LI, W. et al. Feature denoising using joint sparse representation for in-car speech recognition. *Signal Processing Letters, IEEE*, 2013. v. 20, n. 7, p. 681–684, July 2013. Citado na página 27.

LI, X. et al. A comparative study on selecting acoustic modeling units in deep neural networks based large vocabulary chinese speech recognition. *Neurocomputing*, 2015. v. 170, p. 251 – 256, 2015. Citado na página 24.

MARIANI, J. *Language and Speech Processing*. [S.l.]: Wiley, 2013. (ISTE). Citado na página 35.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 1943. v. 5, n. 4, p. 115–133, Dec 1943. Citado na página 47.

MEHROTRA, K.; MOHAN, C.; RANKA, S. *Elements of Artificial Neural Networks*. [S.l.]: MIT Press, 1997. (A Bradford book). Citado 2 vezes nas páginas 52 e 65.

MENDOZA, L. A. F. *Redes neurais e máquinas de vetores de suporte no reconhecimento de locutor usando coeficientes MFC e características do sinal glotal*. Dissertação (Mestrado) — Universidade Federal Fluminense, Niterói, 2009. Citado na página 41.

MOHAMED, A. r.; DAHL, G. E.; HINTON, G. Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012. v. 20, n. 1, p. 14–22, Jan 2012. Citado na página 24.

MOHAMED, A. r. et al. Deep belief networks using discriminative features for phone recognition. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.: s.n.], 2011. p. 5060–5063. Citado na página 24.

MORGAN, D.; SCOFIELD, C. *Neural Networks and Speech Processing*. [S.l.]: Springer US, 2012. (The Springer International Series in Engineering and Computer Science). Citado na página 22.

OKUN, O. *Supervised and Unsupervised Ensemble Methods and their Applications*. [S.l.]: Springer Berlin Heidelberg, 2008. (Studies in Computational Intelligence). ISBN 9783540789802. Citado na página 66.

OPPENHEIM, A.; SCHAFER, R. Homomorphic analysis of speech. *Audio and Electroacoustics, IEEE Transactions on*, 1968. v. 16, n. 2, p. 221–226, Jun 1968. Citado na página 39.

ŠPALE, J.; SCHWEIZER, C. Speech control of measurement devices. *IFAC-PapersOnLine*, 2016. v. 49, n. 25, p. 13 – 18, 2016. 14th IFAC Conference on Programmable Devices and Embedded Systems PDES 2016. Citado na página 19.

PICONE, J. Signal modeling techniques in speech recognition. *Proceedings of the IEEE*, 1993. v. 81, n. 9, p. 1215–1247, Sep 1993. ISSN 0018-9219. Citado 5 vezes nas páginas 38, 39, 40, 41 e 42.

PICONE, J. W. Signal modeling techniques in speech recognition. *Proceedings of the IEEE*, 1993. v. 81, n. 9, p. 1215–1247, Sep 1993. Citado na página 19.

PRIDDY, K.; KELLER, P. *Artificial Neural Networks: An Introduction*. [S.l.]: Society of Photo Optical, 2005. (Tutorial Text Series). Citado 2 vezes nas páginas 51 e 52.

QIAN, Y.; LIU, J.; JOHNSON, M. Efficient embedded speech recognition for very large vocabulary mandarin car-navigation systems. *Consumer Electronics, IEEE Transactions on*, 2009. v. 55, n. 3, p. 1496–1500, August 2009. Citado na página 27.

RABINER, L.; JUANG, B. *Fundamentals of Speech Recognition*. [S.l.]: PTR Prentice Hall, 1993. (Prentice-Hall Signal Processing Series: Advanced monographs). Citado 4 vezes nas páginas 22, 34, 35 e 37.

RABINER, L.; SCHAFER, R. *Digital Processing of Speech Signals*. [S.l.]: Prentice-Hall, 1978. (Prentice-Hall signal processing series). Citado 2 vezes nas páginas 33 e 36.

RABINER, L.; SCHAFER, R. *Introduction to Digital Speech Processing*. [S.l.]: Now Publishers, 2007. (Foundations and Trends in Technology). Citado 2 vezes nas páginas 40 e 42.

RAJPUT, N.; VERMA, S. Back propagation feed forward neural network approach for speech recognition. In: *Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2014 3rd International Conference on*. [S.l.: s.n.], 2014. p. 1–6. Citado na página 26.

ROCHA, P. L.; SILVA, W. L. S. Artificial neural networks used for pattern recognition of speech signal based on dct parametric models of low order. In: *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*. [S.l.: s.n.], 2016. p. 46–51. Citado na página 28.

ROJAS, R.; FELDMAN, J. *Neural Networks: A Systematic Introduction*. [S.l.]: Springer Berlin Heidelberg, 2013. Citado 4 vezes nas páginas 49, 59, 60 e 61.

ROKACH, L. *Pattern Classification Using Ensemble Methods*. [S.l.]: World Scientific Publishing Company Pte Limited, 2010. (Series in machine perception and artificial intelligence). Citado na página 67.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *Nature*, 1986. v. 323, n. 9, p. 533–536, Oct 1986. Citado 2 vezes nas páginas 53 e 54.

SAINATH, T. N. et al. Deep convolutional neural networks for large-scale speech tasks. *Neural Networks*, 2015. v. 64, p. 39 – 48, 2015. Citado na página 25.

SEMAN, N.; BAKAR, Z.; BAKAR, N. Measuring the performance of isolated spoken malay speech recognition using multi-layer neural networks. In: *Science and Social Research (CSSR), 2010 International Conference on*. [S.l.: s.n.], 2010. p. 182–186. Citado na página 26.

SHIH, P. Y.; CHEN, C. P.; WU, C. H. Speech emotion recognition with ensemble learning methods. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.: s.n.], 2017. p. 2756–2760. Citado na página 20.

SILVA, I. D.; SPATTI, D.; FLAUZINO, R. *Redes Neurais Artificiais para Engenharia e Ciências Aplicadas- Curso Prático*. [S.l.]: ARTLIBER, 2010. Citado 11 vezes nas páginas 20, 52, 53, 60, 63, 64, 66, 78, 80, 81 e 83.

SILVA, W. L. S. *Sistema de inferência genético-nebuloso para reconhecimento de voz: uma abordagem em modelos preditivos de baixa ordem utilizando a transformada cosseno discreta*. Tese (Doutorado) — Universidade Federal do Maranhão, São Luís, 3 2015. Citado 4 vezes nas páginas 22, 25, 42 e 73.

SINGH, T.; YADAV, N. Voice recognition based advance patient's room automation. *IJRET: International Journal of Research in Engineering and Technology*, 2015. v. 4, p. 308 – 310, 2015. ISSN 2319-1163. Citado na página 27.

SONG, Q.; JIANG, H.; LIU, J. Feature selection based on fda and f-score for multi-class classification. *Expert Systems with Applications*, 2017. v. 81, p. 22 – 27, 2017. Citado na página 20.

SOUSA, C. A. R. de. An overview on weight initialization methods for feedforward neural networks. In: *2016 International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2016. p. 52–59. Citado na página 84.

STEVENS, J. V. S. S. The relation of pitch to frequency: A revised scale. *The American Journal of Psychology*, 1940. University of Illinois Press, v. 53, n. 3, p. 329–353, 1940. Citado na página 41.

TANG, X. Hybrid hidden markov model and artificial neural network for automatic speech recognition. In: *Circuits, Communications and Systems, 2009. PACCS '09. Pacific-Asia Conference on*. [S.l.: s.n.], 2009. p. 682–685. Citado na página 26.

VEELENTURF, L. *Analysis and Applications of Artificial Neural Networks*. [S.l.]: Prentice Hall, 1995. Citado na página 60.

WENG, F. et al. Conversational in-vehicle dialog systems: The past, present, and future. *IEEE Signal Processing Magazine*, 2016. v. 33, n. 6, p. 49–60, Nov 2016. Citado na página 19.

WIDROW, B.; HOFF, M. E. Neurocomputing: Foundations of research. In: ANDERSON, J. A.; ROSENFELD, E. (Ed.). Cambridge, MA, USA: MIT Press, 1988. cap. Adaptive Switching Circuits, p. 123–134. ISBN 0-262-01097-6. Citado na página 56.

WOODS, K.; KEGELMEYER, W. P.; BOWYER, K. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997. v. 19, n. 4, p. 405–410, Apr 1997. Citado na página 80.

YNOGUTI, C. A.; VIOLARO, F. A brazilian portuguese speech database. In: *XXVI Simpósio Brasileiro de Telecomunicações*. [S.l.: s.n.], 2008. Citado na página 71.

ZHANG, Q. et al. A survey on deep learning for big data. *Information Fusion*, 2018. v. 42, p. 146 – 157, 2018. Citado na página 28.

ZHOU, Z. *Ensemble Methods: Foundations and Algorithms*. [S.l.]: CRC Press, 2012. Citado 2 vezes nas páginas 67 e 76.