

ALEX BRANDÃO ROSSOW

**CLASSIFICAÇÃO DE ESTÁGIOS DO SONO
PELA ANÁLISE DO SINAL DE EEG**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Evandro Ottoni Teatini Salles.

Co-orientador: Prof. Dr. Klaus Fabian Côco.

VITÓRIA
2010

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Central da Universidade Federal do Espírito Santo, ES, Brasil)

R838c Rossow, Alex Brandão, 1982-
Classificação de estágios do sono pela análise do sinal de EEG / Alex Brandão Rossow. – 2010.
99 f. : il.

Orientador: Evandro Ottoni Teatini Salles.
Co-Orientador: Klaus Fabian Côco.
Dissertação (Mestrado em Engenharia Elétrica) –
Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Processamento de sinais. 2. Reconhecimento de padrões.
3. Eletroencefalografia. I. Salles, Evandro Ottoni Teatini. II. Côco, Klaus Fabian, 1971-. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

ALEX BRANDÃO ROSSOW

**CLASSIFICAÇÃO DE ESTÁGIOS DO SONO
PELA ANÁLISE DO SINAL DE EEG**

Dissertação submetida ao programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para a obtenção do Grau de Mestre em Engenharia Elétrica.

Aprovada em 07 de dezembro de 2010.

COMISSÃO EXAMINADORA



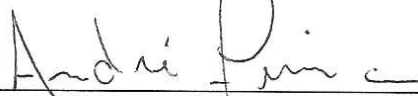
Prof. Dr. Evandro Ottoni Teatini Salles
Universidade Federal do Espírito Santo
Orientador



Prof. Dr. Klaus Fabian Côco
Universidade Federal do Espírito Santo
Co-orientador



Prof. Dr. Vicente Idalberto Becerra Sablón
Centro Universitário Salesiano de São Paulo - Campus São José



Prof. Dr. André Ferreira
Universidade Federal do Espírito Santo

Aos meus pais, Irineu e Tereza e à minha irmã, Angelica

Agradecimentos

A todos que contribuíram com a realização deste trabalho.

Ao Fundo de Apoio à Ciência e Tecnologia do Município de Vitória - FACITEC pelo suporte financeiro.

Sumário

1	Introdução	16
1.1	Motivação	16
1.2	Objetivos	19
1.3	A Metodologia Proposta	19
1.3.1	Abordagem Baseada na Transformada <i>Wavelet Packet</i>	20
1.3.2	Abordagem Baseada no Filtro de Kalman com HMM	23
1.4	Trabalhos Correlatos	23
1.5	Estrutura desta Dissertação	25
2	O Eletroencefalograma e Estágios do Sono	27
2.1	O Sinal de Eletroencefalograma	27
2.1.1	Atividade Neuronal Sincronizada	29
2.1.2	O Sistema de Derivação 10-20	29
2.2	Evolução dos Estágios do Sono	30
2.3	Sinais Utilizados na Classificação de Estágios do Sono	31
2.4	Características dos Sinais Registrados em Cada Estágio	32
3	Fundamentação Teórica	37
3.1	Análise Frequencial do Sinal	37

3.1.1	A Transformada de Fourier	38
3.1.2	A Transformada <i>Wavelet</i> Contínua	39
3.1.3	Transformada <i>Wavelet</i> por Banco de Filtros	40
3.2	Espaços Vetoriais Gerados por Transformadas <i>Wavelet</i> e <i>Wavelet Packet</i> . .	43
3.2.1	Espaços Vetoriais Gerados por Transformada <i>Wavelet</i>	43
3.2.2	Espaços Vetoriais Gerados por Transformada <i>Wavelet Packet</i>	44
3.3	Escolha da Base mais Discriminante	45
3.4	Medida Discriminante e Algoritmo de Escolha dos Nós	46
3.4.1	Normalização da Variância do Sinal Projetado em Cada Nó	46
3.4.2	Medida de Divergência de Kullback-Leibler	47
3.4.3	Atribuição de Medidas Discriminantes aos Nós da <i>Wavelet Packet</i> .	48
3.4.4	Algoritmo de Escolha dos Nós	48
3.5	Classificadores	49
3.5.1	<i>Perceptron</i> de Múltiplas Camadas	49
3.5.2	O classificador KNN (K Nearest Neighbors)	52
3.5.3	O classificador SVM (Support Vector Machine)	53
3.6	Filtro de Kalman Discreto	56
3.7	O modelo ARMA (<i>Autoregressive Moving Average</i>)	58
3.8	HMM (<i>Hidden Markov Models</i>)	61
3.9	Métricas de Desempenho	64
3.9.1	Métricas para Classificadores de Duas Classes	64
3.9.2	Curva de ROC (<i>Receiver Operating Characteristic</i>)	66
3.9.3	Coeficiente Kappa de Cohen κ	68
3.9.4	Exatidão para várias classes	70
3.9.5	Medidas discriminantes utilizadas neste trabalho	70

4	Testes Resultados	72
4.1	Formação dos Conjuntos de Treino e de Teste	72
4.2	Testes com <i>Wavelet Packet</i>	73
4.2.1	Escolha da Função <i>Wavelet</i> e do Número de Nós	73
4.2.2	Configuração da Rede <i>Perceptron</i> de Múltiplas Camadas	76
4.2.3	Definição do Valor de k para o Classificador KNN	79
4.2.4	Configuração dos Parâmetros do Classificador SVM	80
4.2.5	Resultados <i>Wavelet Packet</i>	80
4.3	Testes HMM	87
4.3.1	Resultados HMM	88
4.4	Discussão dos resultados	90
5	Conclusões e Trabalhos Futuros	93

Lista de Tabelas

1.1	Relatório de Acidentes X Causa, nas rodovias federais em 2007 (DPRF, 2010).	17
3.1	Matriz de confusão genérica para duas classes.	64
3.2	Matriz de probabilidade de rotulação.	69
3.3	Matriz de confusão genérica de uma classificação com m classes.	70
4.1	Composição dos conjuntos de treino e de teste.	73
4.2	Teste de desempenho para família <i>wavelet</i> Daubechies. Média do kappa de Cohen para 10 ciclos de treinamento e teste.	74
4.3	Teste de desempenho para família <i>wavelet</i> Symlets. Média do kappa de Cohen para 10 ciclos de treinamento e teste.	75
4.4	Teste de desempenho para família <i>wavelet</i> Coiflets. Média do kappa de Cohen para 10 ciclos de treinamento e teste.	75
4.5	Teste de desempenho para família <i>wavelet</i> "Discrete" Meyer. Média do kappa de Cohen para 10 ciclos de treinamento e teste.	75
4.6	Matriz de confusão para um ciclo de treinamento e teste com a rede <i>perceptron</i>	84
4.7	Matriz de confusão para um ciclo de treinamento e teste com KNN.	84
4.8	Matriz de confusão para um ciclo de treino e teste com SVM.	84
4.9	kappa de Cohen da classificação com HMM.	89
4.10	Taxa de acerto da classificação com HMM.	89
4.11	Matriz de confusão para o modelo ARMA(15,1) na 2 ^a execução. Classificação do conjunto de treino.	89

4.12 Matriz de confusão para o modelo ARMA(15,1) na 2ª execução. Classificação do conjunto de teste.	90
--	----

Lista de Figuras

1.1	Hipnograma (Parisi et al., 2005).	18
1.2	Abordagem utilizando a transformada <i>wavelet packet</i>	22
1.3	Análise de cruzamento pelo zero (Gudmundsson et al., 2005).	24
1.4	Decomposição por <i>wavelet packet</i> . Os subespaços sombreados são utilizados para extração de características(Fraivan et al., 2009).	25
2.1	Sistema 10-20 (Malmivuo e Plonsey, 1995).	30
2.2	Derivações indicadas no sistemas de R&K	32
2.3	Sinais característicos do estado de vigília, com olhos fechados	33
2.4	Sinais característicos do estado de vigília, com olhos abertos	33
2.5	Sinais característicos do estágio 1 em crianças	34
2.6	Sinais característicos do estágio 1	34
2.7	Sinais característicos do estágio 2	34
2.8	Sinais característicos do estágio 3	35
2.9	Sinais característicos do estágio 4	35
2.10	Sinais característicos do estágio REM	36
3.1	Plano tempo-frequência para a STFT (Gargour et al., 2009).	39
3.2	Plano tempo-frequência para a <i>wavelet</i> com a discretização dos parâmetros <i>a</i> e <i>b</i> (Gargour et al., 2009).	40

3.3	Banco de filtros de análise (parte superior) e síntese (parte inferior) (Diniz et al., 2004).	41
3.4	Banco de filtros de análise (parte superior) e síntese (parte inferior) após a aplicação das identidades nobres (Diniz et al., 2004).	42
3.5	Implementação da transformada wavelet por banco de filtros. É apresentada a resposta ao impulso dos canais de análise passa-altas (Diniz et al., 2004).	42
3.6	Decomposição de $\Omega_{0,0}$ em subespaços mutualmente ortogonais usando a transformada <i>wavelet</i> (com $J = 3$). G e H representam o filtros passa-altas e passa-baixas seguidos da etapa de subamostragem.	44
3.7	Decomposição de $\Omega_{0,0}$ em subespaços mutualmente ortogonais usando a transformada <i>wavelet packet</i> (com $J = 3$).	45
3.8	Estrutura de um classificador perceptron com uma camada oculta	50
3.9	Exemplos de hiperplanos de separação entre classes de um classificador <i>perceptron</i> (Bishop, 2006).	51
3.10	Representação da margem. Função dos vetores de suporte, indicados pelos círculos, na formação da superfície de decisão (Bishop, 2006).	54
3.11	Modelo ARMA(10,1) de um sinal de EEG encontrado pelo filtro de Kalman.	60
3.12	Exemplo de um limiar de decisão para curva ROC.	67
3.13	Exemplo de um limiar de decisão deslocado para curva ROC.	67
3.14	Exemplos de curvas ROC.	68
4.1	Estrutura da rede <i>perceptron</i> utilizada.	76
4.2	Função de ativação η	77
4.3	Erro médio de classificação durante o treinamento do <i>perceptron</i> com o algoritmo de retropropagação do erro.	78
4.4	Valor médio do kappa de Cohen, com desvio padrão, (em 50 ciclos de treinamento e teste) para diferentes número de neurônios da camada oculta da rede <i>perceptron</i>	78
4.5	Valor médio do kappa de Cohen, com desvio padrão, (em 50 ciclos de treinamento e teste) para diferentes valores de k com o classificador KNN.	79

4.6	Valores médios do kappa de Cohen com desvio padrão (para 100 ciclos de treinamento e teste) dos três classificadores testados.	81
4.7	Histogramas do kappa de Cohen (para 100 ciclos de treinamento e teste) dos três classificadores testados.	82
4.8	Valores médios das taxas de acerto com desvio padrão (para 100 ciclos de treinamento e teste) dos três classificadores testados.	83
4.9	Histogramas das taxas de acerto (para 100 ciclos de treinamento e teste) dos três classificadores testados.	83
4.10	Curvas ROC de um ciclo de treino e teste com a rede <i>perceptron</i>	85
4.11	Curvas ROC de um ciclo de treino e teste com SVM.	85
4.12	Valores médios da taxa de acerto com os desvios padrão (para 100 ciclos de treino e teste) com os 20 nós menos discriminantes e com os 20 nós mais discriminantes, utilizando SVM.	86
4.13	Histograma da taxa de acerto (para 100 ciclos de treino e teste) com os 20 nós menos discriminantes e com os 20 nós mais discriminantes, utilizando SVM.	86
4.14	Treinamento dos modelos HMM. Um modelo $\bar{\lambda}(A, B, \pi)$ atribuído a cada classe.	87
4.15	Classificação de um período do sono com HMM, $i \in \{1 \dots 6\}$	88

Resumo

Este trabalho trata do problema da classificação automática de estágios do sono pela análise de um canal do sinal de eletroencefalograma (EEG). O estudo foi desenvolvido com a utilização do banco de dados MIT-BIH *Polysomnographic Database*, disponibilizado pela PhysioNet. Os principais aspectos envolvidos neste trabalho são extração de características de um sinal e classificação de padrões. Neste sentido, duas técnicas adaptativas de caracterização de sinais são testadas: a primeira baseando-se na transformada *wavelet packet* e a segunda empregando filtro de Kalman. Para o primeiro caso, avaliou-se três técnicas de classificação: Máquina de Vetores de Suporte, K Vizinhos mais Próximos e Rede Neural *Perceptron* Multicamadas. Já para o segundo caso, selecionou-se os Modelos Ocultos de Markov a fim de se realizar a classificação. Para cada situação, o resultado da classificação é comparado com o resultado da classificação feita por um especialista, empregando-se métricas comumente utilizadas nas áreas de análise automática de EEG e Reconhecimento de Padrões. Os resultados obtidos, apresentados e discutidos, encorajam o emprego de sistemas automáticos na classificação de estágios do sono.

Abstract

This paper addresses the problem of automatic classification of sleep stages by analyzing a signal channel electroencephalogram (EEG). The study was conducted using the MIT-BIH Polysomnographic Database, available from Physionet. The main aspects of this work are feature extraction and classification of signal patterns. In this sense, two adaptive techniques, commonly for characterization of signals, are tested: the first one based on wavelet packet and the second one using the Kalman filter. For the first case it was evaluated three classification techniques: Support Vector Machine, K-Nearest Neighbors and Multilayer Perceptron Neural Network. For the second case, it was selected the Hidden Markov Models in order to perform the classification. For each situation, the classification results are compared with the ratings given by a specialist, using metrics commonly used in the areas of automatic analysis of EEG and Pattern Recognition. The results, here presented and discussed, reinforcing the use of automatic classification of sleep stages.

Capítulo 1

Introdução

1.1 Motivação

A qualidade de vida das pessoas está relacionada a diversos fatores, como a prática de esporte, a alimentação, o trabalho, entre outros. Dentre esses fatores, a qualidade do sono possui grande importância. Distúrbios do sono podem ocasionar diversos problemas, como por exemplo, sonolência excessiva durante o período de atividade ou falta de descanso adequado. Tais problemas podem gerar dificuldades para a realização das atividades cotidianas.

Em atividades que envolvam algum tipo de risco, as consequências da má qualidade de sono podem ser graves. Trabalhadores que operam máquinas perigosas podem ocasionar acidentes graves pela sonolência ou pela falta de atenção geradas pela falta de qualidade do sono.

Um dos setores onde os problemas relacionados ao sono são mais evidentes é o de transporte. No Brasil, a matriz de transporte é predominantemente rodoviária, com esta modalidade correspondendo a cerca de 96,2% da matriz de transporte de passageiros e a 61,8% da matriz de transporte de cargas (CNT, 2006).

Os acidentes de trânsito em nosso país são responsáveis por um grande número de mortes e sequelas permanentes. Um bom indicador desses números é a quantidade de pagamentos realizados pelo seguro obrigatório de Danos Pessoais causados por Veículos Automotores de via Terrestre (DPVAT). Em 2008, foram pagos 57.116 sinistros de morte, 89.474 de invalidez permanente e 125.413 de despesas com assistência médica (DPVAT, 2010).

Dentre as causas de acidentes, uma que apresenta importância significativa é o sono ao dirigir. A Tabela 1.1 apresenta as causas dos acidentes ocorridos nas rodovias federais no ano de 2007 (DPRF, 2010).

Tabela 1.1: Relatório de Acidentes X Causa, nas rodovias federais em 2007 (DPRF, 2010).

Causa	Acidentes	Mortos
Outras	46422	3479
Falta de atenção	42508	1626
Não guardar distância de segurança	8145	71
Velocidade incompatível	6027	540
Defeito mecânico em veículo	5407	122
Animais na Pista	4207	102
Desobediência à sinalização	4062	289
Dormindo	3367	223
Ingestão de álcool	3133	188
Ultrapassagem indevida	2843	517
Defeito na via	2346	71
TOTAL	128467	7228

Como se observa, o número de acidentes causados por motoristas que dormiram ao volante supera o número de acidentes por causas conhecidas como ingestão de álcool ou defeito na via.

Os motoristas profissionais, que dirigem por muitas horas seguidas ou tem o ciclo de sono alterado pela jornada de trabalho, estão muito sujeitos a sofrer acidentes devido à sonolência ao volante. Em estudo realizado no Brasil com 400 motoristas de ônibus interestaduais, 16% admitiram já ter cochilado ao volante (Mello et al., 2000). Em pesquisa realizada nos Estados Unidos com 593 caminhoneiros de viagens longas, 47,1% dos entrevistados disseram já ter dormido ao volante (McCartt et al., 2000).

Existem estudos que relacionam o trabalho noturno e os distúrbios do sono com o aumento no risco de ocorrência de acidentes de trabalho, devido à sonolência excessiva ou déficit de atenção (Pires et al., 2009) (Bittencourt et al., 2005).

Uma pesquisa realizada com 2.110 entrevistados acima dos 16 anos em 150 cidades do Brasil, mostrou que 63% das pessoas ouvidas apresenta algum tipo de reclamação com relação ao sono. Esse estudo também destaca a relação dos distúrbios do sono com queda de qualidade de vida, da produtividade e da capacidade intelectual assim como com a ocorrência de doenças crônicas (Bittencourt et al., 2009).

A evolução normal do sono em adultos apresenta um padrão bem definido e conhecido. Portanto, o seu estudo poder ser útil para se identificar alguma anomalia associada a alterações desse padrão.

O diagnóstico de distúrbios do sono é feito por meio da análise da evolução do sono. O estudo da evolução do sono é feito por meio da polissonografia, onde são registrados diferentes sinais biológicos do paciente durante o sono. Um dos principais aspectos estudados na polissonografia é a evolução dos estágios do sono. Para esse estudo, o tempo de sono é dividido em períodos, normalmente de 20 ou 30 segundos, aos quais são atribuídos um estado do sono.

O padrão para classificação criado por Allan Rechtschaffen e Anthony Kales em 1968 (Carskadon e Rechtschaffen, 2000), R&K, é o mais difundido atualmente. Este padrão se baseia na análise de três sinais, o eletroencefalograma (EEG), o eletrooculograma (EOG) e o eletromiograma (EMG). Normalmente, utiliza-se a classificação em seis estágios do sono possíveis, estado de vigília relaxado (*wakefulness*), estados com movimentos não rápidos dos olhos (*non-rapid eye movement*, NREM) de 1 a 4 e estágio com movimentos rápidos dos olhos (*rapid eye movement*, REM). A representação da evolução dos estágios do sono se chama hipnograma. A Figura 1.1 apresenta um hipnograma de algumas horas de sono.

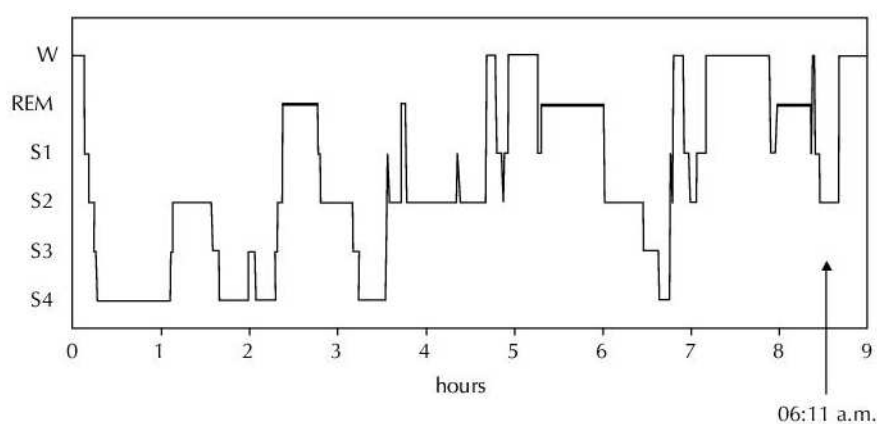


Figura 1.1: Hipnograma (Parisi et al., 2005).

Tradicionalmente, a classificação dos sinais registrados é realizada por um especialista. O desenvolvimento de ferramentas que gerem o hipnograma de forma automática pode reduzir grande parte dos custos e do tempo gasto na análise dos sinais registrados durante o sono, tornando estudos e diagnósticos envolvendo distúrbios do sono mais acessíveis.

Ferramentas de identificação de estágios de consciência também podem ajudar no desenvolvimento de equipamentos que monitorem trabalhadores, cujas atividades envolvam riscos relacionados com o sono, durante sua jornada de trabalho.

1.2 Objetivos

Neste trabalho pretende-se investigar o problema da classificação automática dos estágios do sono, tomando-se como observações os sinais de eletroencefalograma (EEG). Devido à natureza não estacionária do sinal de EEG, selecionou-se duas técnicas adaptativas para sua caracterização: empregando-se os coeficientes de uma transformada *wavelet packet* e empregando-se coeficientes auto-regressivos ajustados por um filtro de Kalman. A primeira abordagem gera um vetor de características não alinhado temporalmente, vetor este classificado por três técnicas: Máquina de Vetores de Suporte, K Vizinhos mais Próximos e Rede Neural *Perceptron* Multicamadas. Já para a segunda abordagem, no intuito de se aproveitar seu caráter estocástico adaptativo, selecionou-se os Modelos Ocultos de Markov a fim de se realizar a classificação.

O resultado da classificação automática deve ser comparado com o resultado da classificação feita pelo especialista empregando-se métricas comumente utilizadas nas áreas de análise automática de EEG e Reconhecimento de Padrões. Finalmente, cabe ressaltar que esse trabalho visa à classificação utilizando os seis estados possíveis do polissonograma sem a fusão de nenhum dos estágios possíveis em uma classe nova.

1.3 A Metodologia Proposta

O presente trabalho faz o uso do banco de dados *MIT-BIH Polysomnographic Database*, disponibilizado pela PhysioNet (Goldberger et al., 2000). Foram utilizados os registros com o sinal de EEG tomado na derivação C4-A1¹, que são: slp01a, slp01b, slp32, slp37 e slp41. Cada um desses registros corresponde a um paciente. Esses sinais são divididos em períodos de 30 segundos, sendo cada período rotulado, por um especialista, como pertencente a um determinado estágio do sono.

O conjunto de observações temporais do sinal de EEG o caracteriza como uma série temporal. A dificuldade maior é saber se o processo estocástico gerado por esta série é de natureza estacionária ou não estacionária. É possível assumi-lo como tal desde que o período de análise seja curto (Tarvainen et al., 2004). É neste sentido que se insere a proposta desta dissertação. Uma das técnicas mais amplamente utilizadas para a análise de sinais não estacionários é a transformada localizada, usando-se o método de janelamento (Diniz et al., 2004), onde localmente avalia-se o sinal. No presente caso, para primeira abordagem, optou-se pela transformada *wavelet* (Mallat, 1998)(Diniz et al., 2004) pelos seus reconhecidos

¹O sinal de EEG, assim como suas derivações são apresentados no capítulo 2

resultados em análise multiescalar de sinais. Assim sendo, a transformada *wavelet* representa uma base de decomposição do sinal, em um novo espaço, onde é possível realizar de maneira mais eficiente a discriminação das classes. E, no intuito de se maximizar a separação entre as classes, neste trabalho empregou-se a técnica de busca da melhor base que identifica as classes a partir de um algoritmo de poda da decomposição conhecida como *wavelet packet*. Ou seja, a base *wavelet packet* desenvolvida, para o banco de dados empregado, gera um espaço onde espera-se que as classes estejam bem separadas. Entretanto, como será visto no capítulo 4, tal fato não garante por si só uma separação perfeita, devido à complexidade dos dados. Mesmo assim, a técnica proporciona resultados aceitáveis. Adotou-se a estrutura *wavelet packet* com 9 camadas de decomposição. Para a etapa de classificação, testamos os classificadores SVM, KNN e uma rede neural tipo *perceptron* multicamadas.

A abordagem acima não trata os dados de maneira estocástica, como uma série temporal. Sua ideia principal é que, localmente, as características extraídas são de um grupo de observações estacionárias, o que pode gerar uma pequena variação intra-classes e assim contribuir na classificação. Entretanto, uma outra abordagem, agora estocástica, associa um modelo espectral a um conjunto de dados. Aqui não se supõe estacionariedade a priori. Sendo assim, empregou-se o filtro de Kalman para continuamente gerar modelos espectrais variantes no tempo. Nossa suposição inicial é de que cada um dos estágios do sono terá uma dinâmica própria e a sequência obtida pelos coeficientes dos modelos ajustados pelo filtro de Kalman gera um processo estocástico que representa bem o referido estágio. Para modelar cada sequência referente a um estágio do sono, optamos pelas Cadeias Ocultas de Markov (HMM) por apresentarem bom desempenho na modelagem de processos estocásticos. Como os modelos HMM, compostos por uma certa quantidade de estados, necessitam associar a cada estado um sentido físico, o que não é possível no nosso caso (diferentemente para o problema de reconhecimento de palavras onde associa-se um estado a um fonema da palavra), optamos pela utilização do modelo K-Means Segmental, tópico a ser abordado no capítulo 3.

Nas duas abordagens os sinais de EEG do banco de dados são divididos em um conjunto de treino e um conjunto de teste, onde cada elemento desses conjuntos corresponde a um período de 30 segundos, rotulado, do EEG.

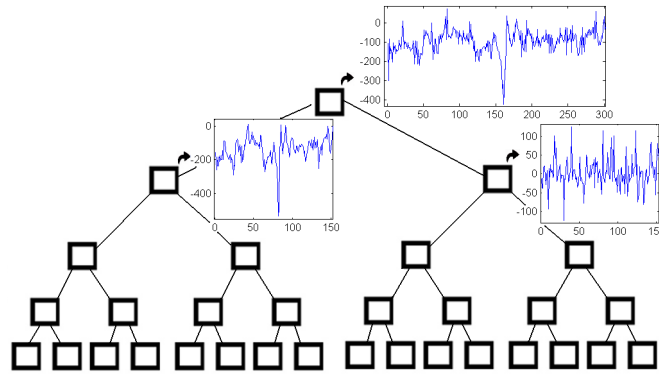
1.3.1 Abordagem Baseada na Transformada *Wavelet Packet*

A primeira abordagem se baseia na decomposição do sinal por meio de uma transformada *wavelet packet* (*wavelet packet transform*, WPT).

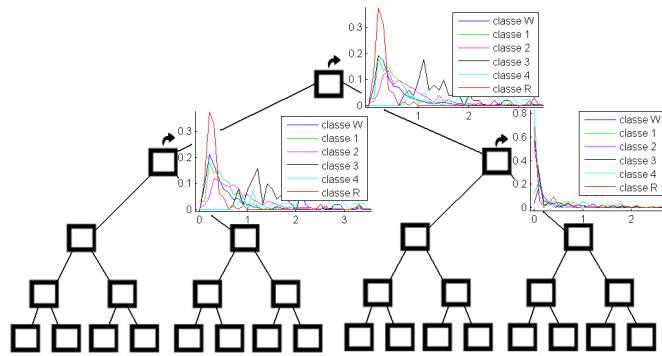
Utilizando o conjunto de teste, busca-se encontrar a base com maior capacidade de discriminar

minação entre as classes, utilizando a distância de Kullback-Leibler. Em seguida, a potência de cada período de 30 segundos do sinal projetado nos nós escolhidos é utilizada para compor o vetor de característica que é usado como entrada dos classificadores.

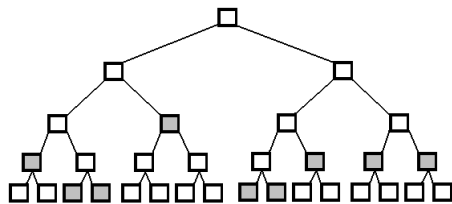
Utilizam-se os classificadores Perceptron de Múltiplas Camadas, k vizinhos mais próximos (*k nearest neighbors*, kNN) e Máquinas de Vetor de Suporte (*Support Vector Machine*, SVM). Os testes com os três classificadores são realizados em condições semelhantes para que os resultados possam ser comparados. Essa primeira abordagem é apresentada na Figura 1.2.



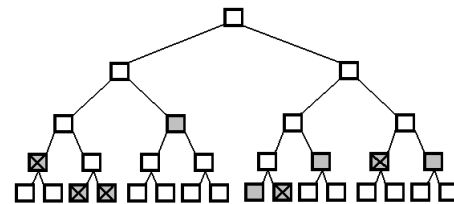
(a) Decomposição dos sinais pela *wavelet packet*.



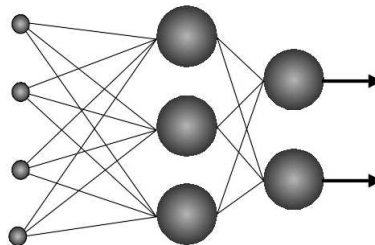
(b) Cálculo dos histogramas de potência dos coeficientes normalizados.



(c) Escolha automática da base mais discriminante pela distância de Kullback-Leibler (formada pelos nós sombreados).



(d) Escolha dos nós mais discriminantes (marcados com um x) da base escolhida para realizar a classificação.



(e) Execução da classificação com os classificadores *perceptron*, KNN e SVM. Utilizando os nós escolhidos para compor os vetores de características.

Figura 1.2: Abordagem utilizando a transformada *wavelet packet*.

1.3.2 Abordagem Baseada no Filtro de Kalman com HMM

Nessa abordagem, o filtro de Kalman é utilizado para gerar um modelo ARMA para os sinais de EEG. Os coeficientes AR são, então, utilizados para a realização da classificação com HMM. Parte do banco de dados é tomado para compor os vetores de treinamento de cada classe. A outra parte do banco de dados é tomado para realizar a classificação. No treinamento, são formados seis conjuntos, um para cada classe. Então, para cada classe, é gerado um modelo HMM. Na classificação, é analisada a probabilidade da amostra a ser classificada pertencer a cada uma das classes. A classe a qual a amostra apresentar maior probabilidade de pertencer é atribuída à amostra.

1.4 Trabalhos Correlatos

Existem alguns trabalhos realizados que buscam a classificação de estágios do sono através de diferentes sinais. Cada um desses trabalhos possuem características específicas, como os sinais utilizados, o número de estados utilizados, a fusão de estados e a utilização de bancos de dados específicos. Aqui são apresentados alguns trabalhos com suas principais características.

Em (Gudmundsson et al., 2005) foi obtida uma taxa de acerto de 81% utilizando um canal de EEG tomado na derivação C3-A2. A classificação foi feita entre quatro classes, os estágios N-REM 1 e 2 foram combinados em uma classe assim como os estágios N-REM 3 e 4. Esse resultado foi obtido com a utilização do classificador SVM. Como entrada do classificador utilizou-se o histograma de características baseado em medidas da forma de onda. Essa técnica utiliza um histograma de duas dimensões com as distribuições de frequência e amplitude do sinal de EEG para cada período do sono a ser classificado. Cada histograma de frequência e amplitude é construído pela análise dos cruzamentos sucessivos do sinal pelo nível definido com valor zero, como mostrado na Figura 1.3. O tempo entre dois cruzamentos de zero sucessivos é denominado T_i e a amplitude, pico a pico, como A_i . Os intervalos do histograma são atualizados para cada par (T_i, A_i) . O eixo de frequência do histograma é dividido nos seguintes intervalos, 0,5 - 2 Hz, 2 - 4 Hz, 4 - 5 Hz, 5 - 7 Hz, 7 - 10 Hz, 10 - 13 Hz, 13 - 15 Hz, 15 - 20 Hz, 30 - 40 Hz. O eixo de amplitude é dividido em cinco intervalos: abaixo de 5 μV , 5 - 30 μV , 30 - 75 μV , 75 - 100 μV e 100 - 400 μV .

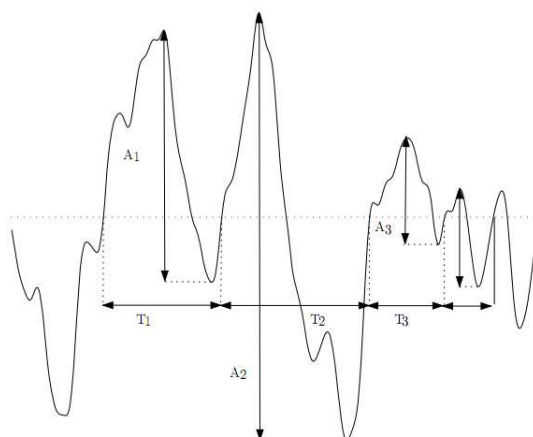


Figura 1.3: Análise de cruzamento pelo zero (Gudmundsson et al., 2005).

Em (Berthomier et al., 2007) obteve-se a taxa de acerto de 82,9%, utilizando um canal do EEG tomado na derivação Cz-Pz. Nesse trabalho a classificação foi feita considerando-se 5 classes possíveis, os estágios N-REM 3 e 4 foram agrupados em uma única classe. Uma característica marcante desse trabalho é que o banco de dados foi classificado manualmente por dois especialistas e os períodos onde houve discordância foram descartados, o que elimina os períodos mais difíceis de se classificar. Esse trabalho utiliza diferentes técnicas de extração de características. Dependendo da informação que se deseja estimar, utiliza-se modelagem auto regressiva, transformada de Fourier, ou medidas instantâneas de frequências (*instantaneous frequency measurement*, IFM), para a extração de informação espectral e temporal, assim como a detecção de microestruturas do sono (*spindles*, complexos K, surgimento de ondas alpha). A classificação é feita com a utilização de lógica difusa (lógica *fuzzy*).

Em (Fraiwan et al., 2009) a taxa de acerto obtida foi de 75%, com a classificação feita pela análise de um canal do EEG, tomado na derivação Pz-Oz e considerando os 6 estágios do sono possíveis do polissonograma. Esse trabalho decompõe o sinal de EEG através da transformada *wavelet packet*. A decomposição é feita até o nível quatro utilizando a função wavelet Daubechies de ordem 20. A decomposição é apresentada na Figura 1.4, onde os nós utilizados para extração de características são apresentados sombreados. As características extraídas em cada nó foram: média, desvio padrão, potência, curtose, máximo e mínimo. No total são gerados 54 parâmetros. Um processo de seleção toma 20 parâmetros para a classificação. Utilizou-se um classificador baseado em árvores de regressão.

L(0,0)								Nível 0
L(1,0)				L(1,1)				Nível 1
L(2,0)		L(2,1)		L(2,2)		L(2,3)		Nível 2
L(3,0)	L(3,1)	L(3,2)	L(3,3)	L(3,4)	L(3,5)	L(3,6)	L(3,7)	Nível 3
							Nível 4

Figura 1.4: Decomposição por wavelet packet. Os subespaços sombreados são utilizados para extração de características (Fraivan et al., 2009).

Em (Zhovna e Shallom, 2008) obteve-se a taxa de acerto de 93,2%, com os dados de treino e com a utilização de um sistema de EEG multi-canal de cinco eletrodos. Neste caso, realizou-se a classificação com 4 classes possíveis, onde os estágios N-REM 3 e 4 foram agrupados em uma classe e o estágio de vigília não foi considerado. Esse trabalho gera um modelo Auto-Regressivo Multi-canal (*Multichannel Auto Regressive*, MAR). Os parâmetros são submetidos à quantização vetorial (*Vector Quantization*, VQ) e então, é criado um histograma para cada classe com os *codbooks* resultantes. A classificação é feita pela comparação dos histogramas das amostras a serem classificadas com os histogramas dos dados de treino, pela medida de divergência de Kullback-Leibler.

Em (Anderer et al., 2007) obteve-se uma taxa de acerto de 78,3%, com o kappa de Cohen² (κ) igual a 0,71, com a utilização de um canal de EEG, dois canais de EOG e um canal de EMG. Esse trabalho utilizou um banco de dados que foi classificado por dois especialistas de diferentes centros de estudo do sono e revisado por um especialista de consenso. Esse trabalho descreve o estudo de desenvolvimento do sistema de classificação automático *Somnolyzer 24x7*, desenvolvido pelo grupo de estudos do sono *The Siesta Group* (Siesta, 2010). De todos os trabalhos analisados esse é o único que apresenta um sistema que já é explorado comercialmente. Não são apresentados detalhes de técnicas de processamento e classificação. Esse trabalho informa que se busca realizar uma classificação de forma semelhante àquela feita de forma visual pelo especialista. E para tal, são utilizados mais de 20 métodos diferentes para a identificação de padrões relacionados aos estágios do sono

1.5 Estrutura desta Dissertação

Este trabalho está organizado da seguinte forma: no capítulo 2 é apresentada a teoria sobre a classificação de estágios do sono e o principal padrão utilizado para essa classificação. São mostrados também os sinais biológicos utilizados na classificação assim como as características que enquadram esses sinais em cada um dos estágios possíveis.

²A medida do kappa de Cohen é explicada na seção 3.9.3.

No capítulo 3 são apresentadas as técnicas abordadas neste trabalho. Na seção 3.1 são apresentadas algumas técnicas de análise frequencial do sinal como a transformada de Fourier, a transformada de Fourier janelada e as transformadas *wavelet* e *wavelet packet*. Na seção 3.2 é mostrado o estudo das transformadas *wavelet* e *wavelet packet* como espaços vetoriais. A seção 3.3 mostra o processo de escolha da base que aumente a capacidade de discriminação entre as classes correspondentes aos estágios do sono. A seção 3.4 apresenta a medida e o algoritmo utilizados na escolha dos nós da estrutura de decomposição da *wavelet packet*, que devem fornecer as informações para a classificação dos estágios do sono. A seção 3.5 apresenta os classificadores utilizados para a classificação dos sinais decompostos pela transformada *wavelet packet*. A seção 3.6 apresenta o filtro de Kalman discreto. A seção 3.7 mostra como pode-se utilizar o filtro de Kalman para criar um modelo ARMA de um sinal de EEG. A seção 3.8 apresenta os Modelos Ocultos de Markov e sua utilização para classificação. A seção 3.9 apresenta um estudo sobre métricas para a análise do desempenho de sistemas de classificação.

No capítulo 4 são apresentados os resultados dos testes realizados. Na seção 4.2 são apresentados os testes e resultados da abordagem baseada na transformada *wavelet packet*. Na seção 4.3 são apresentados os testes e resultados da abordagem baseada nos Modelos Ocultos de Markov. Na seção 4.4 são discutidos os resultados. E no capítulo 5 são apresentadas as conclusões e as sugestões para trabalhos futuros.

Capítulo 2

O Eletroencefalograma e Estágios do Sono

Neste capítulo são apresentadas características do sinal de eletroencefalograma e o conceito sobre estágios do sono. É mostrado como se dá, normalmente, a transição entre estágios e quais as características de cada estágio.

2.1 O Sinal de Eletroencefalograma

O primeiro registro do campo elétrico do cérebro humano foi feito pelo psiquiatra alemão Hans Berger em 1924. Ele deu ao seu registro o nome de eletroencefalograma (EEG). A amplitude do EEG está na faixa de $100 \mu\text{V}$ quando medido sobre o couro cabeludo, e por volta de 1-2 mV quando medido sobre a superfície do cérebro. A banda de frequência ocupada pelo sinal vai de menos de 1 Hz até aproximadamente 50 Hz (Malmivuo e Plonsey, 1995).

O eletroencefalograma pode trazer consigo o registro de atividade não cerebral. Tais sinais que prejudicam o EEG, recebem o nome de artefatos. Os artefatos podem ser classificados em fisiológicos ou extra-fisiológicos. Artefatos *fisiológicos* são aqueles que se originam no corpo em fontes que não o cérebro e artefatos *extra-fisiológicos* são originados em fontes externas ao corpo. A seguir são apresentados alguns artefatos fisiológicos e extra-fisiológicos (Benbadis, 2006).

Artefatos fisiológicos

Atividade muscular (eletromiograma): os potenciais gerados pela atividade muscular

são, provavelmente, os artefatos mais comuns, sendo presenciados em praticamente todo EEG registrado na prática clínica. Mastigação e aperto da mandíbula são comuns.

Movimentação da língua: semelhante ao globo ocular, a língua funciona como um dipolo, com a ponta negativa em relação à base. Dessa forma, a movimentação da língua pode gerar artefatos no registro do EEG.

Atividade cardíaca (eletrocardiograma, ECG): a atividade elétrica do coração também pode ser capturada pelos eletrodos que registram o EEG. Esse tipo de artefato é facilmente identificável pela coincidência com o registro do eletrocardiograma.

Pulso: artefatos de pulso podem ocorrer quando o eletrodo de registro do EEG é posicionado sobre um vaso pulsante. Existe uma relação direta entre as ondas geradas pela pulsação e o eletrocardiograma. As ondas geradas pela pulsação aparecem com atraso por volta de 200-300 ms em relação ao ECG.

Movimentação dos olhos: o globo ocular atua como um dipolo, com polo positivo na parte anterior (córnea) e polo negativo na parte posterior (retina). Quando o globo ocular gira em torno do seu eixo, ele gera um sinal alternado de alta amplitude que é detectável por qualquer eletrodo próximo aos olhos.

Artefatos extra-fisiológicos

Artefatos gerados pelos eletrodos: o artefato de eletrodo mais comum é o desprendimento de um eletrodo, que pode ser identificado por um transiente abrupto nos registros que envolvem somente um determinado eletrodo. Outra possibilidade é a mudança de impedância do eletrodo, neste caso o efeito no sinal é mais suave.

Corrente alternada (60 Hz): o correto aterramento do paciente pode resolver esse tipo de artefato, porém se alguma falha de aterramento ocorrer, pode surgir um artefato com a mesma frequência da rede elétrica. Esse artefato é facilmente identificável por sua frequência bem definida.

Movimentação no ambiente: a movimentação de outras pessoas ao redor do paciente pode gerar artefatos, usualmente, de origem capacitiva ou eletrostática. A interferência de aparelhos hospitalares e equipamentos cirúrgicos também pode gerar artefatos no sinal de EEG.

Estímulo luminoso: o estímulo luminoso pode gerar artefatos fisiológicos, como a resposta do cérebro aos estímulos e pode gerar artefato extra-fisiológico pelo efeito de célula foto-elétrica dos eletrodos. Esse último artefato pode ser eliminado pela blindagem do eletrodo com relação à luz.

2.1.1 Atividade Neuronal Sincronizada

O sinal de EEG captado por um determinado eletrodo registra a atividade de um conjunto de neurônios de uma determinada região do cérebro. Quando os neurônios de uma região específica são ativados ao mesmo tempo, o campo elétrico produzido pelo grupo de neurônios possui amplitude maior que aquele produzido quando os neurônios não estão em atividade sincronizada. Essa diferença pode ser captada através do EEG (Benington, 2007).

Existem eventos, principalmente estímulos sensoriais, que podem ocasionar mudança na atividade de populações neuronais. Essas mudanças podem ser captadas pelo EEG e são chamadas *Potenciais Relacionados a Eventos (Event-Related Potentials, ERPs)*. Quando o evento causa uma diminuição na sincronia de uma população de neurônios, observa-se uma diminuição da potência do sinal em determinadas bandas de frequência, esse fenômeno é chamado de *Dessincronização Relacionada a Eventos (Event-Related Desynchronization, ERD)*. Quando o evento causa um aumento na sincronia de uma determinada população de neurônios observa-se um aumento da potência do sinal em determinadas bandas de frequência, esse fenômeno é chamado de *Sincronização Relacionada a Eventos (Event - Related Synchronization, ERS)* (Pfurtscheller e da Silva, 1999).

2.1.2 O Sistema de Derivação 10-20

O sistema 10-20, internacionalmente padronizado, é o sistema de derivações normalmente utilizado no registro do EEG. Aqui esse sistema merece destaque por ser o sistema recomendado pelo padrão de classificação de estágios do sono de Rechtschaffen e Kales. Nesse sistema, 19 eletrodos são posicionados na superfície do couro cabeludo como mostrado nas Figuras 2.1a e 2.1b. As posições são definidas como segue: os pontos de referência são o *násio*, que é a depressão acima do nariz, no nível dos olhos; e o *ínion*, que é a protuberância óssea do crânio no centro da parte posterior da cabeça. A partir desses pontos, o perímetro do crânio é medido em planos transversais e medianos. As localizações dos eletrodos são determinadas pela divisão desses perímetros em intervalos de 10% e 20%. Dois eletrodos são posicionados nos lóbulos das orelhas (Malmivuo e Plonsey, 1995).

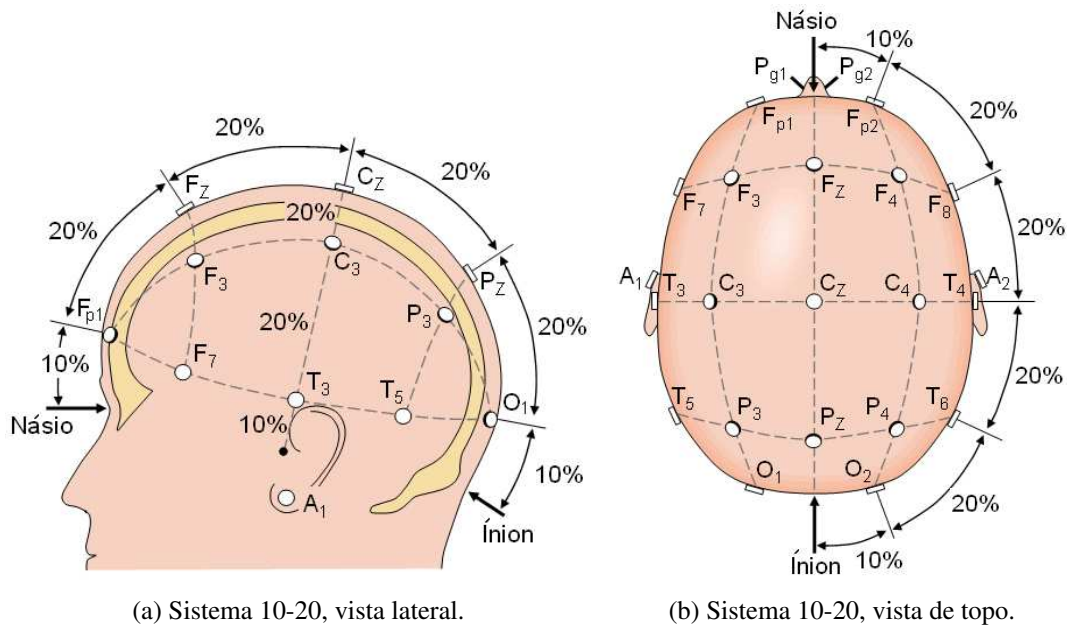


Figura 2.1: Sistema 10-20 (Malmivuo e Plonsey, 1995).

2.2 Evolução dos Estágios do Sono

A estrutura normal do sono de um humano adulto compreende dois estados; estado sem a presença de movimentos rápidos dos olhos (*non-rapid eyes movement*, NREM) e estado com movimentos rápidos dos olhos (*rapid eye movement*, REM). Esses estados apresentam características bem definidas.

O sono NREM apresenta o eletroencefalograma (EEG) com algumas características de atividade cortical síncrona (como *sleep spindles*, complexos K e ondas lentas, da ordem de 2 ciclos por segundo) associado com pouca força muscular e atividade psicológica mínima. É dividido em quatro estágios, numerados de 1 a 4, indicando o sono mais leve até o mais profundo (Carskadon e Dement, 2005).

O sono REM se apresenta no EEG com características que indicam atividade cortical dessincronizada (ondas teta associadas com ondas alfa), ausência de força muscular e o surgimento de sonhos é típico nesse estágio.

O padrão do sono de humanos adultos em uma noite apresenta algumas características definidas: O sono se inicia como NREM estágio 1 e progride para os estágios NREM mais profundos (estágios 2, 3, e 4). O primeiro episódio REM ocorre aproximadamente entre 80 e 100 minutos após o início do sono e dura entre 1 e 5 minutos. Esse ciclo de estágios

NREM-REM se repete durante a noite, sendo que o tempo em que se fica nos estágios 3 e 4 vai diminuindo até desaparecer sendo ocupado pelo estágio 2. O tempo de duração do estágio REM aumenta a cada ciclo. O tempo médio do primeiro ciclo de sono NREM-REM é aproximadamente de 70 a 100 minutos e o tempo médio dos ciclos posteriores é aproximadamente de 90 a 120 minutos (Carskadon e Dement, 2005).

O estudo da evolução do sono é feito por meio do polissonograma, onde o tempo de sono é dividido em períodos aos quais são atribuídos um estado de sono. O padrão para classificação criado por Allan Rechtschaffen e Anthony Kales em 1968 (Carskadon e Rechtschaffen, 2000), R&K, é o mais difundido atualmente. Este padrão se baseia na análise de três sinais, o eletroencefalograma (EEG), o eletrooculograma (EOG) e o eletromiograma (EMG). Outros sinais podem ser registrados dependendo do interesse do estudo.

Como a evolução do sono em um adulto possui um padrão bem definido, pode-se analisar o polissonograma na busca de alterações que possam indicar alguma doença. Em (de Azevedo Abreu et al., 2009) é dito que a polissonografia é o padrão-ouro na identificação da síndrome da apneia e hipopneia obstrutiva do sono.

2.3 Sinais Utilizados na Classificação de Estágios do Sono

No sistema de classificação de Rechtschaffen e Kales é recomendada a utilização do sistema internacional de posicionamento de eletrodos 10-20 (Carskadon e Rechtschaffen, 2000) com o objetivo de evitar o mau posicionamento dos eletrodos pela utilização de outro sistema. Os pontos de referência normalmente são os pontos pré-auriculares direito e esquerdo, respectivamente representados por A2 e A1. É adotada a convenção "negativo para cima" no registro do EEG, o que significa que um sinal de polaridade negativa é mostrado como um deslocamento para cima do gráfico traçado. A medida de amplitude usada na classificação dos estágios do sono é tomada pico a pico. Existem quatro grupos de frequências (ritmos) utilizados para caracterizar um sinal de EEG, apresentados a seguir.

1. Ritmo Alfa: de 8 a 13 Hz;
2. Ritmo Beta: mais de 13 Hz;
3. Ritmo Delta: menos de 4 Hz;
4. Ritmo Teta: de 4 a 7 Hz.

No sistema de classificação de R&K são utilizados, além do EEG, os sinais de EOG e EMG. O sinal de EEG é captado na derivação C3/A2 ou C4/A1. Pode-se utilizar a gravação do EEG da região occipital (O1/A2 ou O2/A1). O EOG é captado a partir de eletrodos posicionados nos cantos externos de ambos os olhos, os eletrodos devem estar levemente desalinhados na horizontal. As posições desses eletrodos são conhecidas como ROC e LOC, *Right Outer Canthus* e *Left Outer Canthus*. Como referência para o registro do EOG utilizam-se os pontos A1 ou A2. Para o registro do eletromiograma utilizam-se eletrodos posicionados abaixo do queixo. As derivações para captura dos sinais recomendados no sistema de R&K são apresentadas na Figura 2.2.

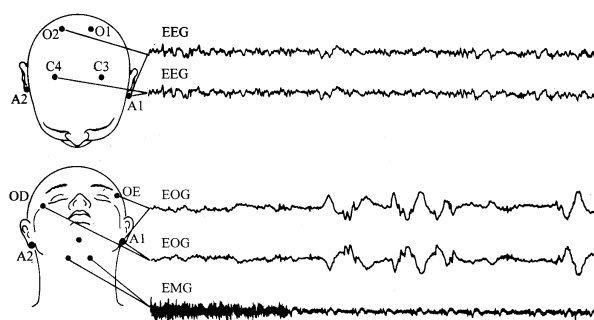


Figura 2.2: Derivações indicadas no sistema de R&K, (Guiot, 1996).

2.4 Características dos Sinais Registrados em Cada Estágio

O período de análise para a atribuição de um estado é normalmente de 20 a 30 segundos. Segundo o sistema de Rechtschaffen e Kales, os sinais de EEG, EOG e EMG em cada estágio apresentam as seguintes características (Carskadon e Rechtschaffen, 2000):

- **Vigília relaxada (*wakefulness*):** com os olhos fechados o EEG apresenta ritmo alfa (8-13 Hz) proeminente no registro occipital, essa onda se atenua com o aumento da atenção, os registros desse estado são mostrados na Figura 2.3. Com os olhos abertos, o EEG apresenta baixa amplitude e é composto de frequências variadas. O movimento dos olhos é voluntário. O EOG apresenta movimentos rápidos dos olhos e piscadas quando os olhos estão abertos, e não registra movimentos quando os olhos estão fechados. Movimentos giratórios lentos podem surgir em segundos ou minutos antes de se entrar no estágio 1. O EMG mostra atividade muscular de nível relativamente alto. Em indivíduos muito relaxados o EMG normalmente não se distingue daquele

observado no sono NREM. A Figura 2.4 mostra os registros de um homem em vigília relaxada com os olhos abertos.

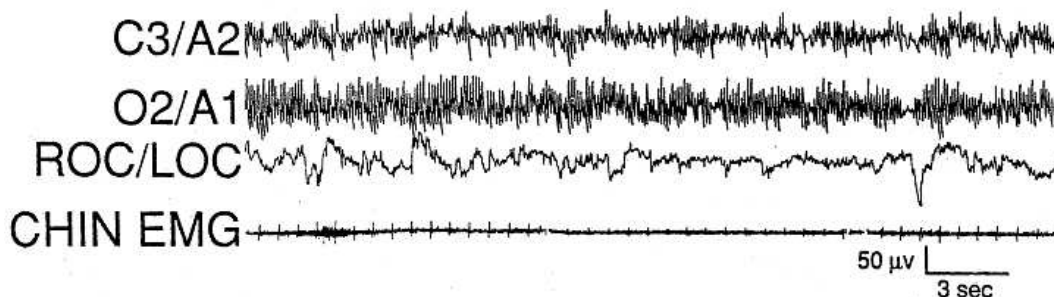


Figura 2.3: EEG de um homem acordado com olhos fechados, observa-se a presença de atividade alfa, principalmente no registo O2/A1 (Carskadon e Rechtschaffen, 2000).



Figura 2.4: EEG de um homem acordado com olhos abertos, observa-se a atenuação da atividade alfa. Sinal de baixa amplitude formado por diferentes frequências. (Carskadon e Rechtschaffen, 2000).

- **Estágio 1:** o EEG apresenta baixa amplitude de frequências variadas com as ondas teta (3-7 Hz) apresentando amplitude mais elevada. Presença de ondas de vértice pontiagudo. Surgimento de atividade teta com grande amplitude em crianças, mostrado na Figura 2.5. A força muscular é mantida nos estágios NREM, o que pode ser observado por uma atividade de baixa amplitude do EMG. Frequentemente um aumento de amplitude do sinal captado pelo EMG é o único indicador de ocorrência de uma transição para o estágio 1 dentro de um período de sono REM. A Figura 2.6 apresenta algumas características do estágio 1.

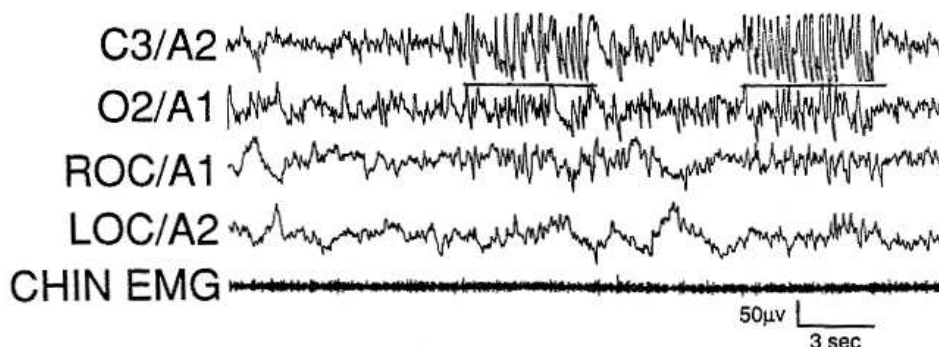


Figura 2.5: Aqui se observa a atividade teta (sublinhada) com grande amplitude, comum no início do estágio 1 em crianças. Esse sinal é de um jovem de 14 anos do sexo masculino (Carskadon e Rechtschaffen, 2000).

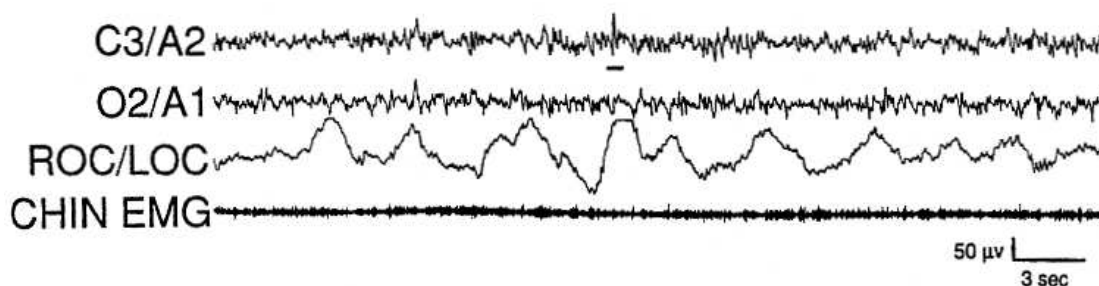


Figura 2.6: A presença de ondas com vértice pontiagudo é comum no início do estágio 1. Aqui observa-se a presença de um desses vértices, sublinhado, no registro C3/A2 (Carskadon e Rechtschaffen, 2000).

- **Estágio 2:** EEG com frequências variadas com baixa amplitude. Surgimento de “*sleep spindles*”, que é uma onda de frequência elevada (12-14 Hz) com duração maior que 0,5 segundo. Presença do complexo K, que é uma forma de onda característica do estágio 2. O complexo K consiste de um pico negativo bem definido imediatamente seguido por uma componente positiva. O tempo total de duração do complexo K deve exceder a 0,5 segundo. Essas características são mostradas na Figura 2.7

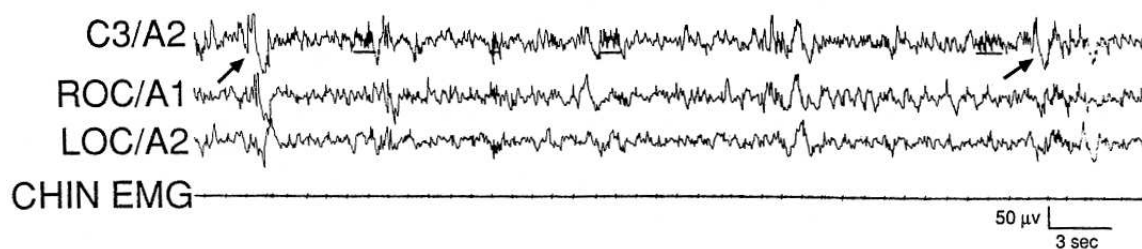


Figura 2.7: EEG característico do estágio 2. Aqui se observa a presença de “*sleep spindle*” (sublinhado) e de complexos K (apontados pelas setas) (Carskadon e Rechtschaffen, 2000).

- **Estágio 3:** EEG com presença de sinal de alta amplitude ($> 75\mu\text{V}$) e baixa frequência ($\leq 2\text{ Hz}$) entre 20 e 50% do período a ser classificado. A Figura 2.8 mostra os sinais característicos do estágio 3.

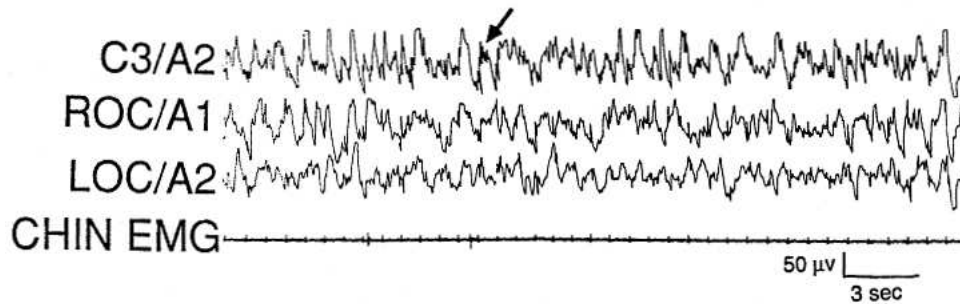


Figura 2.8: Estágio 3, presença de ondas de alta amplitude ($> 75\mu\text{V}$) e baixa frequência ($\leq 2\text{ Hz}$) por um período maior que 20% e menor que 50% do tempo de análise. “*Sleep spindles*” podem ocorrer no estágio 3, aqui temos a seta apontando um caso (Carskadon e Rechtschaffen, 2000).

- **Estágio 4:** EEG com presença do sinal de alta amplitude e baixa frequência em mais de 50 % do período a ser classificado, como se pode ver na Figura 2.9.

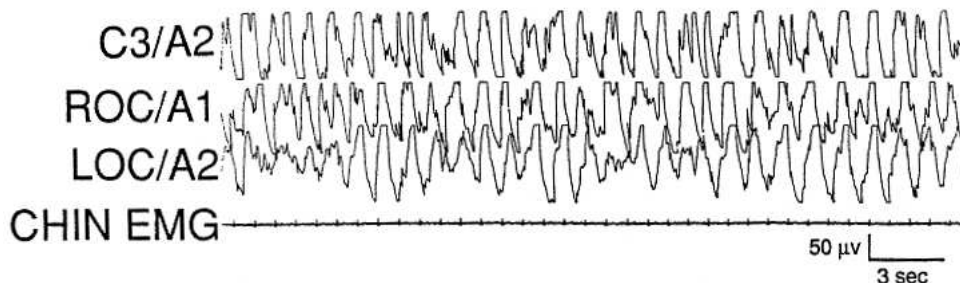


Figura 2.9: Estágio 4, predominância de ondas de alta amplitude e baixa frequência em mais de 50 % do período a ser classificado (Carskadon e Rechtschaffen, 2000).

- **REM:** A detecção do estágio REM depende da ocorrência de características específicas nas três medidas registradas. O EEG apresenta frequências variadas com baixa amplitude, ondas em forma dente de serra, presença de ondas teta e sinal alfa com baixa amplitude. O EOG apresenta intervalos com ocorrência de movimentos rápidos do olhos. O EMG apresenta sua menor amplitude indicando uma supressão da força muscular. Podem ocorrer contrações musculares ocasionais, principalmente próximo a períodos com movimentos rápidos do olhos. As principais características do estágio REM são mostradas na Figura 2.10

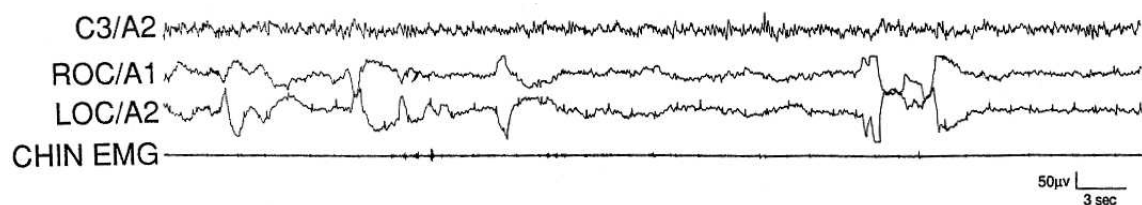


Figura 2.10: EEG característico do estágio REM, sinal de baixa amplitude composto por frequências variadas. O EOG mostra a ocorrência de movimentos rápidos dos olhos. Os registros dos sinais ROC e LOC apresentam fases opostas. O EMG apresenta a supressão da atividade muscular. Observa-se o registro de contrações musculares pelo EMG no primeiro período de movimentos rápidos dos olhos (Carskadon e Rechtschaffen, 2000).

Capítulo 3

Fundamentação Teórica

Esse capítulo apresenta as técnicas empregadas neste trabalho. Começando pela análise frequencial do sinal, onde são apresentadas a Transformada de Fourier, a Transformada de Fourier janelada e as Transformadas *Wavelet* e *Wavelet Packet*. É apresentada a implementação das Transformadas *Wavelet* e *Wavelet Packet* a partir de bancos de filtros. É apresentado o processo de escolha dos nós da *wavelet packet* para formar um espaço vetorial no qual a projeção dos sinais de EEG tenha uma boa capacidade de discriminação entre os estágios do sono. Traz-se uma breve introdução sobre os classificadores *perceptron* de múltiplas camadas, KNN e SVM. É apresentado o filtro de Kalman discreto e como esse filtro pode ser usado na criação de um modelo ARMA para o sinal de EEG. São apresentados o Modelos Ocultos de Markov e sua aplicação na classificação de séries temporais estocásticas. Por fim são mostradas algumas métricas de desempenho úteis para problemas de classificação.

3.1 Análise Frequencial do Sinal

Essa seção apresenta algumas técnicas de análise frequencial de sinais. Começando pela Transformada de Fourier. Em seguida é mostrada a Transformada de Fourier janelada, que fornece informações sobre a composição espectral de um sinal em instantes de tempo específicos. Após isso, faz-se uma breve introdução à transformada *wavelet*. Por fim, é apresentada a implementação da transformada *wavelet* através de bancos de filtros discretos.

3.1.1 A Transformada de Fourier

A transformada de Fourier de um sinal $x(t)$, consiste na sua projeção na base constituída pelo conjunto de funções complexas $\{e^{j\Omega t}\}$, conforme se observa abaixo:

$$FT\{x(t)\} = X(\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t} dt, \quad \Omega = 2\pi F, \quad (3.1)$$

onde $j = \sqrt{-1}$, F é a frequência em Hz e Ω é a frequência angular em rad/s.

A transformada de Fourier nos dá a representação global do sinal no domínio da frequência. Ela não fornece informação sobre a variação espectral do sinal ao longo do tempo. Para contornar esse problema foi criada a Transformada de Fourier janelada (*Short-time Fourier Transform*, STFT), que consiste na aplicação da transformada em partes do sinal delimitadas por uma janela de tamanho fixo que se desloca ao longo do tempo, conforme a equação a seguir:

$$STFT_{g(\Omega,b)}\{x(t)\} = X_g(\Omega, b) = \int_{-\infty}^{\infty} x(t)g(t-b)e^{-j\Omega t} dt, \quad (3.2)$$

onde $g(t)$ é a janela de tamanho fixo. O parâmetro b permite que a janela seja posicionada possibilitando a aplicação da Transformada de Fourier em uma parte específica do sinal. Definindo

$$\Psi_{\Omega,b}(t) = g(t-b)e^{j\Omega t} \quad (3.3)$$

a STFT pode ser expressa como:

$$STFT_{g(\Omega,b)}\{x(t)\} = X_g(\Omega, b) = \int_{-\infty}^{\infty} x(t)[\Psi_{\Omega,b}^*(t)]dt, \quad (3.4)$$

onde $\Psi_{\Omega,b}^*(t)$ é o conjugado complexo de $\Psi_{\Omega,b}(t)$.

Sendo Δ_t a resolução no tempo e Δ_Ω a resolução na frequência, dadas pela janela, temos a inequação de Heisenberg

$$\Delta_t \Delta_\Omega \geq \frac{1}{2} \quad (3.5)$$

limitando a resolução no plano tempo-frequência. A melhor resolução é obtida, isso é a equação é satisfeita, quando a janela $g(t)$ é uma função Gaussiana dada por

$$g_\sigma(t) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{t^2}{2\sigma^2}}, \quad (\Delta_t = \sigma, \Delta_\Omega = \frac{1}{2\sigma}). \quad (3.6)$$

Com essa função janela a STFT recebe o nome de transformada de Gabor (Gargour et al., 2009).

Uma característica da STFT é que, escolhido o tamanho da janela, todas as componentes tempo-frequência são extraídas do sinal com a mesma resolução de tempo e frequência, o que pode ser visualizado na Figura 3.1.

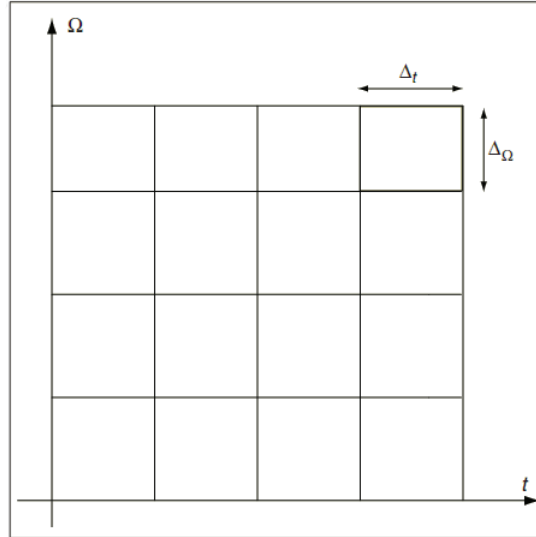


Figura 3.1: Plano tempo-frequência para a STFT (Gargour et al., 2009).

3.1.2 A Transformada *Wavelet* Contínua

Para analisar um sinal com estruturas de tamanho muito diferente, é necessário usar estruturas tempo-frequência (chamadas átomos) com diferentes suportes de tempo. A Transformada *wavelet* decompõe os sinais sobre *wavelets* dilatadas e transladadas. Uma *wavelet* é uma função, do tempo, $\psi \in L^2(\mathbb{R})$ com média zero, (Mallat, 1998):

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (3.7)$$

A *wavelet* é normalizada $\|\psi\| = 1$, e centralizada na vizinhança de $t = 0$. A família de átomos tempo-frequência é obtida pelo dimensionamento de ψ por a e pela sua translação por b , com $a \in \mathbb{R}^+$ e $b \in \mathbb{R}$, conforme

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right). \quad (3.8)$$

Esses átomos continuam normalizados: $\|\Psi_{a,b}\| = 1$. A transformada *wavelet* contínua de um sinal $x(t)$ no tempo b e escala a é:

$$X_{\Psi}(a,b) = \langle x(t), \Psi_{a,b}(t) \rangle = \int_{-\infty}^{\infty} x(t) \frac{1}{\sqrt{a}} \psi^*\left(\frac{t-b}{a}\right) dt \quad (3.9)$$

onde o asterisco (*) denota o conjugado complexo da função.

O fato dos parâmetros a e b serem contínuos gera um alto nível de redundância, pois não seria possível ter átomos tempo-frequência sem sobreposição. Para reduzir essa redundância ambos parâmetros podem ser amostrados. Gerando átomos tempo-frequência que se tangem mas não se sobrepõem. É possível realizar a amostragem como se segue:

$$a = 2^m, b = n2^m, m \in \mathbb{Z}, n \in \mathbb{Z}. \quad (3.10)$$

Fazendo isso obtemos uma versão da *wavelet* com deslocamento e dimensionamento discretos, segundo

$$\Psi_{m,n}(t) = 2^{-\frac{m}{2}} \psi(2^{-m}t - n), m \in \mathbb{Z}, n \in \mathbb{Z}. \quad (3.11)$$

Essa versão da transformada *wavelet* gera um ladrilhamento do plano tempo-frequência como mostrado na Figura 3.2.

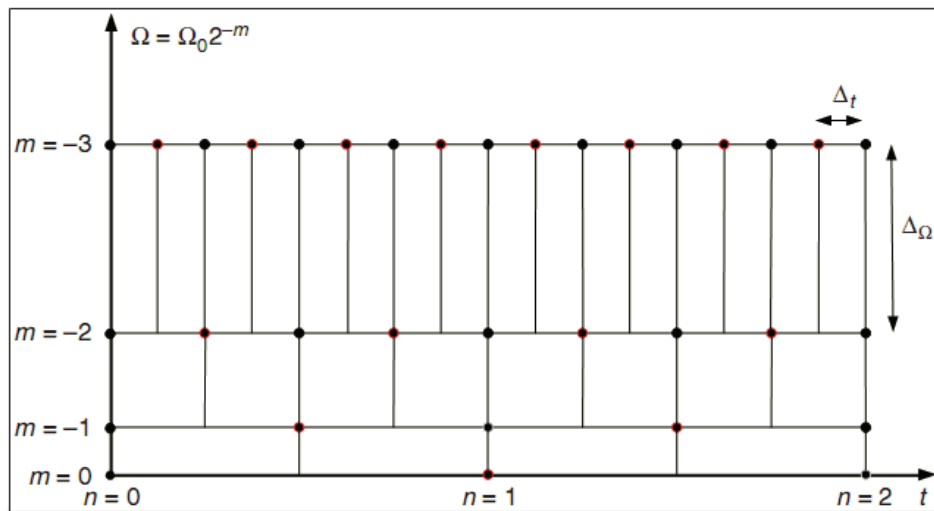


Figura 3.2: Plano tempo-frequência para a *wavelet* com a discretização dos parâmetros a e b (Gargour et al., 2009).

3.1.3 Transformada *Wavelet* por Banco de Filtros

A transformada *wavelet* pode ser implementada por banco de filtros para sinais discretos. A Figura 3.3 apresenta uma estrutura de banco de filtros, de análise e síntese, que pode ser usada para implementar a transformada *wavelet* para sinais discretos, onde $H_0(z)$ é um filtro passa-baixas e $H_1(z)$ é um filtro passa-altas. As saídas dos canais passa-baixas após o $(S + 1)$ -ésimo estágio de decomposição são $x_{S,n}$, e as saídas dos canais passa-altas são $c_{S,n}$, com $S \geq 1$.

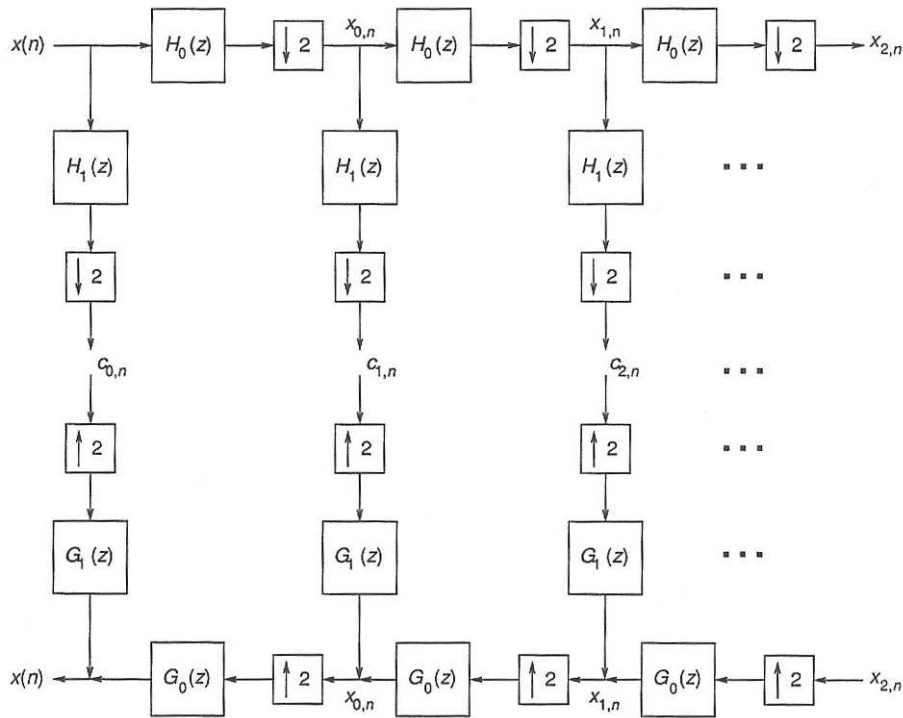


Figura 3.3: Banco de filtros de análise (parte superior) e síntese (parte inferior) (Diniz et al., 2004).

Aplicando as identidades nobres (Diniz et al., 2004) à essa estrutura chegamos ao esquema apresentado na Figura 3.4.

Nessa estrutura, após $(S + 1)$ estágios e antes da decimação por um fator $2^{(S+1)}$, as transformadas z dos canais de análise passa-baixas e passa-altas, $H_{pb}^{(S)}(z)$ e $H_{pa}^{(S)}(z)$, são:

$$H_{pb}^{(S)}(z) = \frac{X_S(z)}{X(z)} = \prod_{k=0}^S H_0(z^{2^k}) \quad (3.12)$$

$$H_{pa}^{(S)}(z) = \frac{C_S(z)}{X(z)} = H_1(z^{2^S}) H_{pb}^{(S-1)}(z) \quad (3.13)$$

Na Figura 3.5 pode-se ver que a escolha adequada dos filtros $H_0(z)$ e $H_1(z)$ faz com que as envoltórias da resposta ao impulso dos filtros da equação 3.13 tenham o mesmo formato. Ou seja, podem ser representadas por expansões e contrações de uma única função $\psi(t)$, que é a função *wavelet* (Diniz et al., 2004). O mesmo acontece com a envoltória da resposta ao impulso do canal de análise passa-baixas $H_{pb}^{(s)}(z)$, que em cada nível s pode ser representada pela expansão ou pela contração de uma função $\phi(t)$, chamada função de escalamento (Diniz et al., 2004).

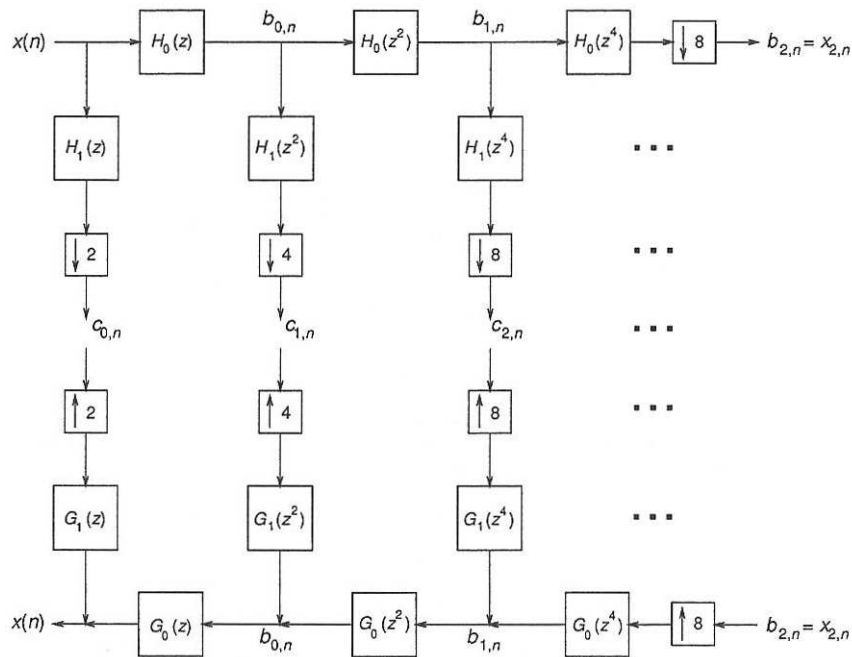


Figura 3.4: Banco de filtros de análise (parte superior) e síntese (parte inferior) após a aplicação das identidades nobres (Diniz et al., 2004).

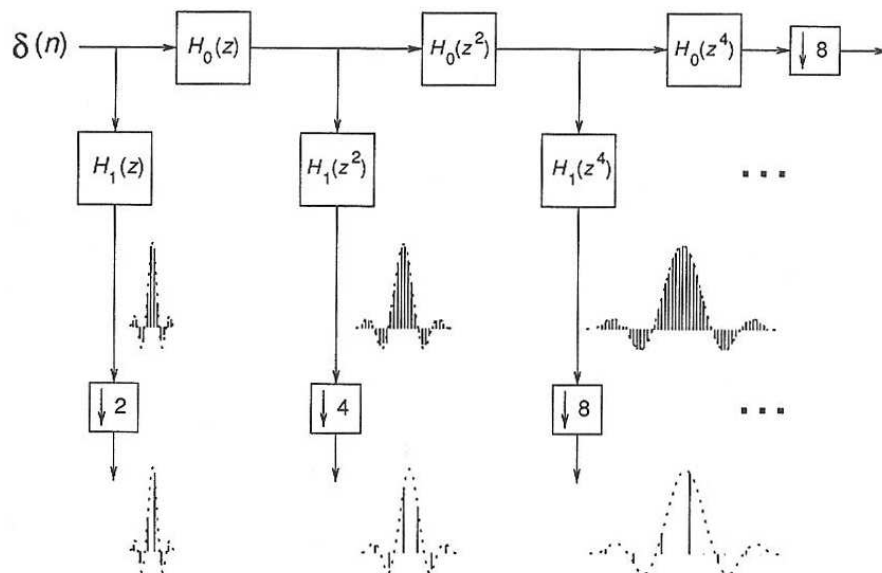


Figura 3.5: Implementação da transformada wavelet por banco de filtros. É apresentada a resposta ao impulso dos canais de análise passa-altas (Diniz et al., 2004).

3.2 Espaços Vetoriais Gerados por Transformadas *Wavelet* e *Wavelet Packet*

Aqui é mostrado como a aplicação das transformadas *wavelet* e *wavelet packet* correspondem à projeção de um sinal em um espaço vetorial.

Vamos considerar para a nossa análise que a dimensão de um sinal discreto será $n = 2^{n_0}$. Então temos o sinal $x = (x_k)_{k=0}^{n-1}$.

3.2.1 Espaços Vetoriais Gerados por Transformada *Wavelet*

Na Transformada *Wavelet* o sinal é, primeiramente, decomposto em uma banda de baixa frequência e uma banda de alta frequência através da convolução com um par de filtros espelhados em quadratura (*quadrature mirror filter*, QMF) de dimensão L , sendo um filtro passa-baixas $h_{k=0}^{L-1}$ e um filtro passa-altas $g_{k=0}^{L-1}$, seguida por uma etapa de subamostragem. Vamos considerar H e G como a convolução com os filtros passa-baixas e passa-altas, respectivamente, seguida da subamostragem. A aplicação do operador H gera um subespaço chamado **aproximação** e a aplicação do operador G gera um subespaço chamado **detalhe**.

Na primeira etapa, um sinal $x = \{x_k\}_{k=0}^{n-1}$ é dividido em duas subseqüências Hx e Gx de dimensão $n/2$. A saída do filtro passa-baixas subamostrada (aproximação) pode passar pelo mesmo processo, obtendo-se duas novas subseqüências H^2x e GHx , de dimensão $n/4$. A função de transferência do filtro resultante de sucessivas camadas de filtragem e subamostragem pode ser obtida com a aplicação das identidades nobres (Diniz et al., 2004). Repetindo-se esse processo J vezes obtém-se uma estrutura como a mostrada na Figura 3.6.

Seja $\Omega_{0,0}$ o espaço vetorial \mathbb{R}^n . A aplicação dos operadores de projeção H e G a $\Omega_{0,0}$ geram, respectivamente, os subespaços mutuamente ortogonais $\Omega_{1,0}$ e $\Omega_{1,1}$. Em geral o j -ésimo passo de decomposição pode ser escrito como mostrado na equação 3.14

$$\Omega_{j,0} = \Omega_{j+1,0} \oplus \Omega_{j+1,1}, \quad (3.14)$$

onde o símbolo \oplus refere-se à concatenação de espaços ortogonais.

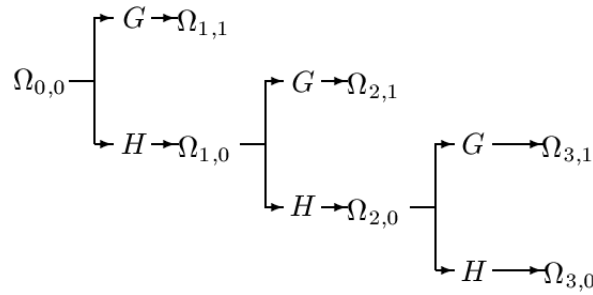


Figura 3.6: Decomposição de $\Omega_{0,0}$ em subespaços mutuamente ortogonais usando a transformada *wavelet* (com $J = 3$). G e H representam o filtros passa-altas e passa-baixas seguidos da etapa de subamostragem.

3.2.2 Espaços Vetoriais Gerados por Transformada *Wavelet Packet*

Como foi visto na seção 3.2.1, na transformada *wavelet* apenas os espaços de aproximação são decompostos. Diferentemente disso, na *wavelet packet* tanto as saídas dos canais passa-altas (detalhes) quanto as saídas dos canais passa-baixas (aproximações) são decompostas, gerando a estrutura mostrada na Figura 3.7.

O primeiro nível de composição gera Hx e Gx . O segundo nível gera quatro subsequências, H^2x, GHx, HGx, G^2x . Repetindo-se esse processo J vezes acabamos tendo $J \cdot n$ coeficientes de expansão (em cada nível o número de coeficientes se mantém igual à dimensão do sinal). A realização desse processo, de forma iterativa, gera subespaços de \mathbb{R}^n em uma estrutura de árvore binária onde os nós da árvore representam subespaços de frequências diferentes. A raiz da árvore é representada por $\Omega_{0,0}$ (espaço do sinal original). Um nó $\Omega_{j,k}$ é dividido em dois subespaços ortogonais $\Omega_{j+1,2k}$ e $\Omega_{j+1,2k+1}$ pelos operadores H e G respectivamente, o que pode ser representado pela equação 3.15:

$$\Omega_{j,k} = \Omega_{j+1,2k} \oplus \Omega_{j+1,2k+1}, \quad (3.15)$$

para $j = 0, 1, \dots, J$ $k = 0, \dots, 2^j - 1$

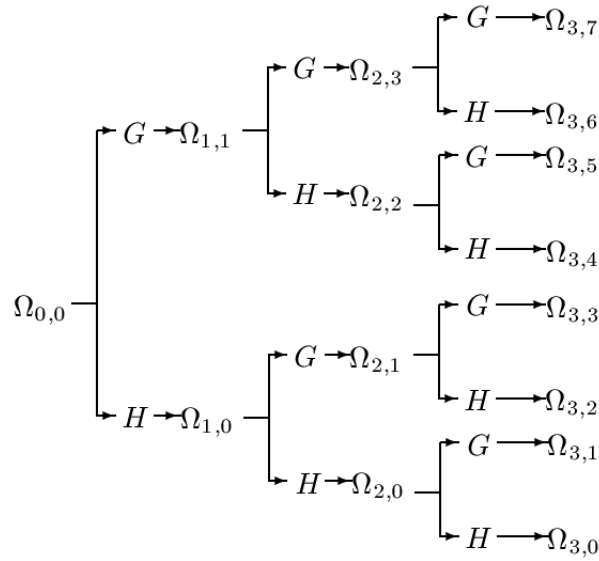


Figura 3.7: Decomposição de $\Omega_{0,0}$ em subespaços mutuamente ortogonais usando a transformada *wavelet packet* (com $J = 3$).

Para se formar um espaço de \mathbb{R}^n a partir da árvore de decomposição da *wavelet packet* os nós que formarão a base devem ser escolhidos de forma que, ao se percorrer qualquer ramo a partir da raiz até a camada de decomposição J , um e apenas um nó seja escolhido no caminho.

Cada subespaço $\Omega_{j,k}$ é composto por 2^{n_0-j} vetores base $\{w_{j,k,l}\}_{l=0}^{2^{n_0-j}-1}$. O vetor $w_{j,k,l}$ é aproximadamente centralizado em $2^j l$ (Saito, 1994). Denotando o conjunto de vetores base de $\Omega_{j,k}$ por $B_{j,k}$, considerando os vetores base $w_{j,k,l}$ vetores coluna, pode-se adotar a representação matricial a seguir

$$B_{j,k} = (w_{j,k,0}, \dots, w_{j,k,2^{n_0-j}-1})^T. \quad (3.16)$$

A projeção do sinal x (vetor coluna) sobre cada um desses vetores base (produto interno) gera um coeficiente $w_{j,k,l}^T \cdot x$. Portanto, cada nó da árvore é composto por 2^{n_0-j} coeficientes. A projeção de x sobre um subespaço $\Omega_{j,k}$ pode ser representada por $B_{j,k} \cdot x$ (Saito, 1994).

3.3 Escolha da Base mais Discriminante

Depois de escolhida a função *wavelet* a ser usada (escolha dos filtros que compõem o banco) a estrutura de árvore apresentada na Figura 3.7 pode ser vista como um dicionário de bases. Ou seja, a escolha dos nós, seguindo o critério apresentado na sessão anterior, forma uma base sobre a qual o sinal é projetado.

A busca pela melhor base consiste em escolher os nós (subespaços) de forma a construir uma base que maximize ou minimize algum parâmetro de interesse. Por exemplo, para compressão de dados pode-se buscar uma base que minimize a entropia dos coeficientes, ou seja, uma base onde a energia se concentre em poucos coeficientes (Saito, 1994).

No caso deste trabalho busca-se uma base onde a projeção do sinal de EEG tenha a máxima capacidade de discriminação entre as classes (estágios do sono). Vale a pena ressaltar que para se representar completamente um sinal de dimensão n , permitindo a sua recuperação, deve-se escolher uma base de dimensão \mathbb{R}^n . Como neste trabalho o interesse é a classificação dos dados e não seu armazenamento, nem todos os nós escolhidos para formar a base de projeção serão tomados para a classificação.

A limitação do número de nós a ser utilizado se deve ao fato de que a potência do sinal projetado em cada nó será tomada como parâmetro de entrada dos classificadores. Organizando-se os nós da base escolhida em ordem decrescente de capacidade de discriminação e tomando os primeiros nós para a classificação chega-se num ponto onde a adição de mais nós piora o desempenho dos classificadores.

Outra possibilidade gerada pela não necessidade de recuperação do sinal é a de se aplicar transformações sobre os subespaços (nós da árvore) de forma a aumentar o desempenho do algoritmo de escolha de base e dos classificadores. Tais transformações resultariam em um sinal distorcido se fosse feita a tentativa de recuperação do sinal original.

3.4 Medida Discriminante e Algoritmo de Escolha dos Nós

Aqui são descritas as etapas para se obter a base mais discriminante após a decomposição do sinal pela transformada *wavelet packet*.

3.4.1 Normalização da Variância do Sinal Projetado em Cada Nó

Para o cálculo da medida discriminante de cada nó avalia-se a potência do sinal projetado em cada nó da *wavelet packet*. Em seguida, constrói-se um histograma de potência de cada nó para cada uma das seis classes.

Recapitulando o que foi apresentado na seção 3.2.2. Cada nó da transformada *wavelet packet* representa um subespaço $\Omega_{j,k}$ cuja base é dada pelo conjunto de vetores base $B_{j,k}$. A projeção de um sinal x em $\Omega_{j,k}$ pode ser representada por $B_{j,k}x$.

A projeção do sinal em cada nó, representa uma dada faixa de frequência. Dessa forma, as componentes de frequência de maior amplitude do sinal farão com que os nós correspondentes tenham maior potência. Essa característica dificulta a construção dos histogramas, visto que a faixa de valores de potência varia muito de um nó para o outro, além de não ser conveniente para a utilização da potência do sinal projetado nos nós como características para classificação.

A fim de se evitar esse problema, foi utilizado um fator de normalização de variância ρ para cada nó. Para o cálculo de ρ , são escolhidas, aleatoriamente, n amostras de cada uma das seis classes. Para cada nó da *wavelet packet* obtém-se um valor $\rho_{j,k}$ concatenando-se a projeção de todas as amostras escolhidas e calculando-se o inverso do desvio padrão do sinal gerado por essa concatenação, segundo apresentado na equação seguinte

$$\rho_{j,k} = \frac{1}{\sqrt{\text{VAR}(\text{CONCAT}_{i=1}^{6n}(B_{j,k}x_i))}}, \quad (3.17)$$

onde: VAR representa variância,

CONCAT representa concatenação,

n é o número de amostras escolhidas de cada classe (nesse trabalho adotou-se $n = 20$),

x_i representa as amostras selecionadas aleatoriamente.

Dessa forma, a projeção do sinal x em um dado nó $\Omega_{j,k}$, com variância normalizada, é dada por

$$x_{proj} = \rho_{j,k} B_{j,k} x. \quad (3.18)$$

3.4.2 Medida de Divergência de Kullback-Leibler

Como visto na seção anterior, a escolha da melhor base consiste em selecionar os nós de uma árvore de decomposição *wavelet packet*, que é o dicionário de bases, de forma a se maximizar ou minimizar um parâmetro desejado. Neste trabalho, o objetivo é encontrar uma base que maximize a capacidade de discriminação entre as classes (estágios do sono). Para cada nó será atribuído uma medida de capacidade de discriminação entre as classes, \mathcal{D} , de forma que os nós possam ser comparados.

Como medida discriminante foi utilizada a versão simétrica da medida de divergência de Kullback-Leibler (Kullback e Leibler, 1951).

Seja H_1 e H_2 histogramas, de d pontos, de duas classes distintas c_1 e c_2 . A versão não

simétrica da divergência de K-L é dada por:

$$I(c_1, c_2) = \sum_{i=1}^d H_1(i) \log \frac{\max(H_1(i), \delta)}{\max(H_2(i), \delta)}, \quad (3.19)$$

onde: $\max(a, b)$ retorna o maior valor entre a e b ,

δ é um número pequeno que tem por objetivo evitar a divisão por zero ou o logaritmo de zero. Foi utilizado $\delta = 0,001$.

A versão simétrica da divergência de K-L, J , é dada por:

$$J(c_1, c_2) = I(c_1, c_2) + I(c_2, c_1). \quad (3.20)$$

3.4.3 Atribuição de Medidas Discriminantes aos Nós da *Wavelet Packet*

O primeiro passo para a atribuição de uma medida discriminante para um nó $\Omega_{j,k}$ da *wavelet packet* é o cálculo da potência do sinal projetado normalizado neste nó, dado pela equação 3.18. Esse cálculo é realizado para todas as amostras de uma classe c_i , obtendo-se, então, o histograma de potência da classe (H_i). Com os histogramas das seis classes, calcula-se a medida discriminante $\mathcal{D}_{j,k}$ do nó segundo

$$\mathcal{D}_{j,k} = \sum_{m=1}^5 \sum_{n=m+1}^6 J(c_m, c_n), \quad (3.21)$$

onde as classes c_1, \dots, c_6 representam as classes referentes aos seis estágios do sono.

A equação 3.21 representa o somatório das medidas de divergência J de todas as combinações possíveis das seis classes tomadas de duas em duas. Esse cálculo é realizado para todos os nós da *wavelet packet* para que esses possam ser comparados no processo de escolha de quais nós serão usados para a classificação.

3.4.4 Algoritmo de Escolha dos Nós

O processo de escolha dos nós da *wavelet packet* que serão utilizados para a extração das características a serem utilizadas na classificação pode ser resumido nos seguintes passos:

Passo 0 Especificar os filtros que serão utilizados para executar a *wavelet packet* (escolha do dicionário de bases ortonormais).

Passo 1 Executar a decomposição das amostras rotuladas em J camadas, calculando-se o valor da medida discriminante de todos os nós.

Passo 2 Atribuir para cada nó $\Omega_{j,k}$ um sinalizador $s_{j,k}$ que indica se esse nó faz ou não parte da base discriminante escolhida, $s_{j,k} = 1$ indica que o nó $\Omega_{j,k}$ faz parte da base escolhida, $s_{j,k} = 0$ indica que o nó $\Omega_{j,k}$ não faz parte da base escolhida. Inicializar os sinalizadores dos nós da última camada com 1 e os demais com 0. Atribuir para cada nó $\Omega_{j,k}$ uma medida discriminante $\Delta_{j,k}$ que é atualizada durante o processo de escolha da base. Inicializar Δ para todos os nós com os valores das medidas discriminantes calculadas pela equação 3.21 ($\Delta_{j,k} = \mathcal{D}_{j,k}$).

Passo 3 Executar a escolha dos nós $\Omega_{j,k}$ para $j = J - 1, \dots, 0$ $k = 0, \dots, 2^j - 1$ seguindo as seguintes regras:

se $\Delta_{j,k} \geq \Delta_{j+1,2k}$ e $\Delta_{j,k} \geq \Delta_{j+1,2k+1}$

então $s_{j,k} = 1$,

e atribui o valor zero a todos os sinalizadores dos nós pertencentes aos ramos da árvore que se originam em $\Omega_{j,k}$,

senão $\Delta_{j,k} = \max(\Delta_{j+1,2k}, \Delta_{j+1,2k+1})$.

Nesse processo são selecionados, de forma automática, p nós que formam uma base de projeção do sinal que maximiza a capacidade de discriminação entre as classes.

Passo 4 Escolher os q nós ($q < p$) com maior valor de Δ para serem utilizados como características de classificação.

3.5 Classificadores

Essa seção apresenta uma breve introdução aos três classificadores que foram utilizados com a técnica de decomposição do sinal pela transformada *wavelet packet*. Os classificadores utilizados foram o *Perceptron* de Múltiplas Camadas, o classificador de K vizinhos mais próximos (K *Nearest Neighbors*, KNN) e o classificador de máquina de vetor de suporte (*Support Vector Machine*, SVM).

3.5.1 *Perceptron* de Múltiplas Camadas

O classificador *Perceptron* de Múltiplas Camadas (Bishop, 2006) com uma camada oculta pode ser representado pela rede mostrada na Figura 3.8. Nessa rede, o vetor de entrada \mathbf{x} possui dimensão D , a primeira camada da rede \mathbf{z} possui dimensão M e a saída \mathbf{y} possui dimensão K .

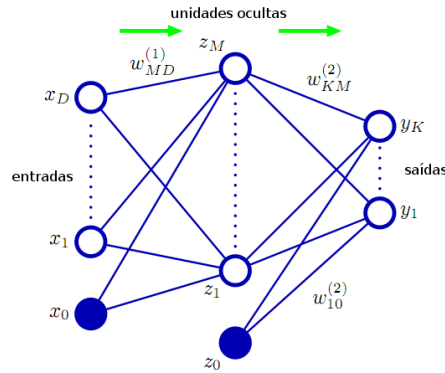


Figura 3.8: Estrutura de um classificador perceptron com uma camada oculta (Bishop, 2006).

Na primeira camada da rede obtém-se M combinações lineares das variáveis de entrada $\{x_1, \dots, x_D\}$

$$a_j = \sum_{i=1}^D w_{ji}^{(1)} x_i + w_{j0}^{(1)} \quad (3.22)$$

onde (1) indica que se trata da primeira camada da rede, j é índice dos elementos da primeira camada da rede e i é o índice dos elementos no vetor de entrada. Considera-se $w_{ji}^{(1)}$ como pesos e $w_{j0}^{(1)}$ como parâmetro de polarização, a_j é o resultado da combinação linear dos parâmetros de entrada.

Aplicando uma função de ativação não linear $h(\cdot)$ a cada uma das combinações lineares obtidas, obtém-se os elementos da primeira camada.

$$z_j = h(a_j) \quad (3.23)$$

Realizando processo semelhante obtém-se K combinações lineares dos elementos da primeira camada da rede:

$$a_k = \sum_{j=1}^M w_{kj}^{(2)} z_j + w_{k0}^{(2)} \quad (3.24)$$

onde (2) indica que se trata da segunda camada da rede, k é índice dos elementos da camada de saída da rede. Considera-se $w_{kj}^{(2)}$ como pesos e $w_{k0}^{(2)}$ como parâmetro de polarização, a_k é o resultado da combinação linear dos parâmetros da primeira camada da rede.

Aplicando-se a função de ativação $\eta(\cdot)$ a cada uma das saídas a_k obtém-se as saídas da rede, conforme a equação abaixo

$$y_k = \eta(a_k). \quad (3.25)$$

Fixando-se os valores de x_0 e z_0 iguais a 1, o valor das saídas é função dos vetores de entrada \mathbf{x} e de pesos \mathbf{w} . A valor de cada uma das saídas, para um vetor de entrada \mathbf{x} é dado pela equação a seguir:

$$y_k(\mathbf{x}, \mathbf{w}) = \eta \left(\sum_{j=1}^M w_{kj}^{(2)} h \left(\sum_{i=1}^D w_{ji}^{(1)} x_i + w_{j0}^{(1)} \right) + w_{k0}^{(2)} \right). \quad (3.26)$$

As funções $\eta(\cdot)$ e $h(\cdot)$ são funções de ativação. Normalmente funções sigmóides (função logística ou função tangente hiperbólica).

O número de saídas da rede corresponde ao número de classes possíveis. Para um dado vetor de entrada \mathbf{x} , a saída que apresentar maior valor é a classe atribuída pelo classificador à esse vetor de entrada.

Para um vetor de pesos \mathbf{w} fixo, os possíveis vetores de entrada que faz com que duas saídas possuam o mesmo valor formam hiperplanos. Esses hiperplanos correspondem às fronteiras de transição entre as classes.

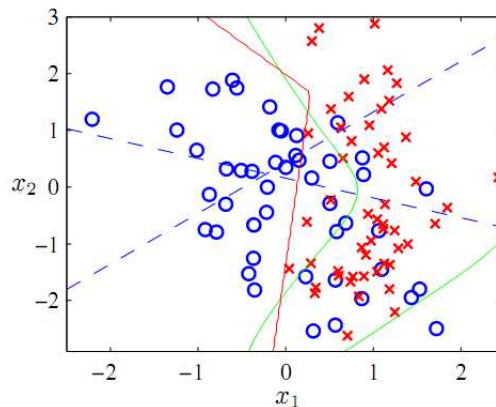


Figura 3.9: Exemplos de hiperplanos de separação entre classes de um classificador *perceptron* (Bishop, 2006).

O processo de treino consiste em ajustar os pesos de forma a diminuir o valor do erro de classificação. Um algoritmo de treino comumente utilizado é o algoritmo de retropropagação do erro. Dada a n -ésima entrada $\mathbf{x}(n) = (x_1(n), \dots, x_D(n))$ cuja saída desejada é dada por $(d_1(n), \dots, d_K(n))$ com a saída obtida $\mathbf{y}(n) = (y_1(n), \dots, y_K(n))$. O sinal de erro na saída do neurônio j , na iteração n (i.e., a apresentação do n -ésimo exemplo de treino), é definido por:

$$e_j(n) = d_j(n) - y_j(n), \quad (3.27)$$

O valor instantâneo da energia do erro para cada neurônio j é definido como $\frac{1}{2}e_j^2(n)$. O valor

instantâneo do erro $\mathcal{E}(n)$ é obtido somando-se os $\frac{1}{2}e_j^2(n)$ de todos os neurônios

$$\mathcal{E}(n) = \frac{1}{2} \sum_{j=1}^K e_j^2(n). \quad (3.28)$$

O algoritmo de retropropagação do erro altera os pesos \mathbf{w} de forma a reduzir o valor de $\mathcal{E}(n)$. Ele faz isso através de um processo de descida de gradiente de $\mathcal{E}(n)$ em relação aos pesos \mathbf{w} . O processo de atualização é realizado com as amostras de treino disponíveis até que uma condição de parada seja atingida. Define-se **época** a apresentação de todas as amostras de treino utilizadas. O treino da rede por retropropagação pode ser realizados através de dois métodos diferentes, *modo sequencial* ou *modo por lote*.

1. *Modo sequencial*: nesse modo de operação os pesos são atualizados a cada amostra do conjunto de treino apresentada à rede.
2. *Modo por lote*: no modo por lote a atualização dos pesos é feita após a apresentação de todas as amostras que constituem uma época.

Nesse trabalho, utilizou-se o treino sequencial com o embaralhamento das amostras ao final de cada época. O processo detalhado de implementação do algoritmo de retropropagação pode ser encontrado em (Haykin, 2001).

3.5.2 O classificador KNN (K Nearest Neighbors)

O classificador KNN (Bishop, 2006) pode ser explicado com a aplicação do teorema de Bayes. Considerando um conjunto de dados com um total de N pontos, cada classe C_k possui N_k pontos, logo $\sum_k N_k = N$. Para classificar um ponto x toma-se uma hiper-esfera centrada em \mathbf{x} contendo exatamente K pontos, independente da classe. Supondo que essa hiper-esfera possui volume V e contém K_k pontos da classe C_k . A equação 3.29 fornece uma estimativa da densidade de probabilidade associada a cada classe,

$$p(\mathbf{x}|C_k) = \frac{K_k}{N_k V}. \quad (3.29)$$

A probabilidade não condicional é dado por

$$p(\mathbf{x}) = \frac{K}{NV}, \quad (3.30)$$

e a probabilidade a priori da classe é dada por

$$p(C_k) = \frac{N_k}{N}. \quad (3.31)$$

Combinando as equações 3.29, 3.30 e 3.31 e usando o teorema de Bayes obtém-se a probabilidade a posteriori das classes, segundo:

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})} = \frac{K_k}{K}. \quad (3.32)$$

Para minimizar a probabilidade de erro de classificação deve-se atribuir a classe com maior probabilidade posteriori, ou seja, identifica-se os K pontos, do conjunto de treino, mais próximos do ponto a ser classificado e atribui-se a ele a classe que possui maior número de representantes no grupo escolhido.

3.5.3 O classificador SVM (Support Vector Machine)

O classificador SVM tem como objetivo realizar a separação entre duas classes de forma que a margem entre a superfície de separação e as amostras mais próximas a essa superfície seja maximizada. Diferente do que acontece com a rede perceptron treinada por retro propagação do erro, onde os vetores de peso são influenciados por todas as amostras de treinamento, no SVM a superfície de decisão é definida apenas por alguns elementos do conjunto de treino. Esses elementos recebem o nome de **vetores de suporte**.

Considerando a função de discriminante para o caso de classificação com duas classes dada por

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b, \quad (3.33)$$

onde a função ϕ representa uma função de transformação do espaço de características, \mathbf{w} corresponde à uma transformação linear e b corresponde ao parâmetro de polarização. O conjunto de treinamento compreende N vetores de entrada $\mathbf{x}_1, \dots, \mathbf{x}_N$, com rótulos correspondentes t_1, \dots, t_N , onde $t_n \in \{-1, 1\}$, e novos pontos \mathbf{x} são classificados de acordo com o sinal de $y(\mathbf{x})$.

Considerando que os conjunto de dados é linearmente separável no espaço de características, então existe pelo menos uma escolha dos parâmetros \mathbf{w} e b tal que a equação 3.33 satisfaça $y(\mathbf{x}_n) > 0$ para os pontos que possuam $t_n = +1$ e $y(\mathbf{x}_n) < 0$ para os pontos que possuem $t_n = -1$ tal que $t_n y(\mathbf{x}) > 0$ para todos os pontos de treinamento.

No SVM a superfície de decisão é escolhida de forma a maximizar a margem entre essa superfície e o ponto do conjunto de treino mais próximo. A Figura 3.10 apresenta o conceito da margem e mostra como os vetores de suporte atuam na formação da superfície de decisão.

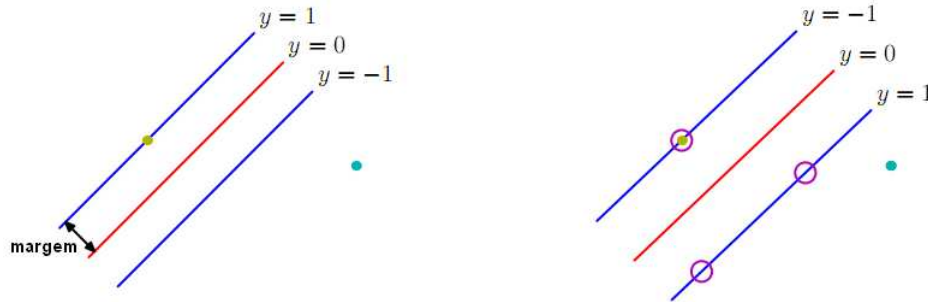


Figura 3.10: Representação da margem. Função dos vetores de suporte, indicados pelos círculos, na formação da superfície de decisão (Bishop, 2006).

A distância de um ponto \mathbf{x} até um hiperplano definido por $y(\mathbf{x}) = 0$ (equação 3.33) é dada por $|y(\mathbf{x})|/\|\mathbf{w}\|$, onde $\|\cdot\|$ representa a norma Euclidiana. Como estamos interessados nas soluções classificadas corretamente, tal que $t_n y(\mathbf{x}_n) > 0$ para todo n , então a distância do ponto \mathbf{x}_n até a superfície de decisão é dada por

$$\frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n (\mathbf{w}^T \phi(\mathbf{x}_n) + b)}{\|\mathbf{w}\|}. \quad (3.34)$$

Deseja-se maximizar a distância perpendicular entre a fronteira de decisão e o ponto \mathbf{x}_n , do conjunto de treinamento, mais próximo através da otimização de \mathbf{w} e b . Então a solução que maximiza a margem é encontrada pela solução de

$$\arg \max_{\mathbf{w}, b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n [t_n (\mathbf{w}^T \phi(\mathbf{x}_n) + b)] \right\}. \quad (3.35)$$

Reescalando, por um fator k , $\mathbf{w} \rightarrow k\mathbf{w}$ e $b \rightarrow kb$, a distância de qualquer ponto \mathbf{x}_n até a superfície de decisão, dada por $t_n y(\mathbf{x}_n)/\|\mathbf{w}\|$ não é alterada. Assim pode-se ajustar

$$t_n (\mathbf{w}^T \phi(\mathbf{x}_n) + b) = 1 \quad (3.36)$$

para o ponto mais próximo da superfície.

Assim, todos os pontos vão satisfazer a restrição

$$t_n (\mathbf{w}^T \phi(\mathbf{x}_n) + b) \geq 1, \quad n = 1, \dots, N. \quad (3.37)$$

Com essa restrição o problema descrito por 3.35 pode ser simplificado na maximização de $\|\mathbf{w}\|^{-1}$, o que é equivalente a minimizar $\|\mathbf{w}\|^2$. Ou seja, deve-se resolver o problema de otimização

$$\arg \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.38)$$

sujeito à restrição dada por 3.37. O fator $1/2$ é incluído por conveniência para o momento de derivar a função Lagrangeana com relação a \mathbf{w} .

Para resolver o problema de otimização descrito por 3.38 sujeito às restrições dadas por 3.37 utiliza-se os multiplicadores de Lagrange $a_n \geq 0$, com um multiplicador para cada restrição em 3.37, o que dá a função Lagrangeana a seguir:

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{n=1}^N a_n \{t_n (\mathbf{w}^T \phi(\mathbf{x}_n) + b) - 1\}, \quad (3.39)$$

onde $\mathbf{a} = (a_1, \dots, a_N)^T$. O sinal menos na frente do multiplicador de Lagrange se deve ao fato de estar sendo feita uma minimização com relação a \mathbf{w} e b e uma maximização com relação a \mathbf{a} . Fazendo as derivadas de $L(\mathbf{w}, b, \mathbf{a})$ com relação a \mathbf{w} e b iguais a zero, obtém-se as seguintes duas condições:

$$\mathbf{w} = \sum_{n=1}^N a_n t_n \phi(\mathbf{x}_n), \quad (3.40)$$

$$0 = \sum_{n=1}^N a_n t_n. \quad (3.41)$$

Eliminando \mathbf{w} and b de $L(\mathbf{w}, b, \mathbf{a})$ usando essas condições obtém-se a *representação dual* do problema de maximização da margem, dada por

$$\tilde{L}(a) = \sum_{n=1}^N a_n - \frac{1}{2} \sum_{n=1}^N \sum_{m=1}^N a_n a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m), \quad (3.42)$$

sujeita às restrições

$$a_n \geq 0, \quad n = 1, \dots, N, \quad (3.43)$$

$$\sum_{n=1}^N a_n t_n = 0. \quad (3.44)$$

Aqui a função kernel é definida por $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$

Para classificar uma nova amostra usando o modelo treinado, avalia-se o sinal de $y(\mathbf{x})$ definido pela equação 3.33. Esse valor pode ser expresso em função dos parâmetros a_n e da função kernel pela substituição de \mathbf{w} usando a equação 3.40. O que resulta em

$$y(\mathbf{x}) = \sum_{n=1}^N a_n t_n k(\mathbf{x}, \mathbf{x}_n) + b. \quad (3.45)$$

Esse tipo de otimização deve obedecer às condições de Karush-Kuhn-Tucker (KKT), que nesse caso requer que as três propriedades a seguir sejam válidas (Bishop, 2006)

$$a_n \geq 0, \quad (3.46)$$

$$t_n y(\mathbf{x}_n) - 1 \geq 0, \quad (3.47)$$

$$a_n \{t_n y(\mathbf{x}_n) - 1\} = 0. \quad (3.48)$$

Então, para todos os pontos do conjunto de treino, deve-se ter $a_n = 0$ ou $t_n y(\mathbf{x}_n) = 1$. Os pontos para os quais $a_n = 0$ não aparecem no somatório em 3.45 e portanto não apresentam função na classificação de novos pontos.

Depois de encontrado o vetor \mathbf{a} , pode-se encontrar o valor do parâmetro de polarização b considerando que qualquer um dos vetores de suporte deve satisfazer $t_n y(\mathbf{x}_n) = 1$. Por 3.45 tem-se

$$t_n \left(\sum_{m \in \mathcal{S}} a_m t_m k(\mathbf{x}_n, \mathbf{x}_m) + b \right) = 1 \quad (3.49)$$

onde \mathcal{S} representa o conjunto de índices dos vetores de suporte. Embora o valor de b possa ser encontrado resolvendo essa equação para apenas um vetor de suporte, uma solução mais confiável é encontrada calculando-se o valor de b para todos os vetores de suporte e posteriormente dividindo o valor pelo número total de vetores de suporte $N_{\mathcal{S}}$. Multiplicando a equação 3.49 por t_n e usando o fato de que $t_n^2 = 1$ e calculando a média obtém-se

$$b = \frac{1}{N_{\mathcal{S}}} \sum_{n \in \mathcal{S}} \left(t_n - \sum_{m \in \mathcal{S}} a_m t_m k(\mathbf{x}_n, \mathbf{x}_m) \right). \quad (3.50)$$

3.6 Filtro de Kalman Discreto

O filtro de Kalman permite a estimação de um processo $x(n)$, não estacionário, a partir de observações, mesmo que ruidosas, de um segundo processo $y(n)$, que tenha sua relação com $x(n)$ conhecida. O processo $x(n)$ é representado por um vetor de estados $\mathbf{x}(n)$ cuja evolução é dada por 3.51. O processo $y(n)$ consiste de um vetor de observações $\mathbf{y}(n)$, a relação entre o vetor de estados que representa o processo estimado e as observações é dada pela equação 3.54. A seguir são apresentadas as principais variáveis e equações relacionadas ao filtro de Kalman.

Seja um processo representado por um vetor de estados de dimensão c (Hayes, 1996) que evolui de acordo com a equação

$$\mathbf{x}(n) = \mathbf{A}(n-1)\mathbf{x}(n-1) + \mathbf{w}(n), \quad (3.51)$$

onde $\mathbf{A}(n-1)$ é uma matriz de transição de estados variante no tempo de dimensão $c \times c$ e $\mathbf{w}(n)$ é um vetor que representa um ruído branco com variância σ_w^2 . A matriz de covariância $\mathbf{Q}_w(n)$ do ruído $\mathbf{w}(n)$ é dada por

$$E \{ \mathbf{w}(n) \mathbf{w}^H(k) \} = \mathbf{Q}_w(n) = \begin{cases} \sigma_w^2 & k = n \\ 0 & k \neq n, \end{cases} \quad (3.52)$$

onde $E\{\cdot\}$ representa o cálculo do valor esperado,

\mathbf{w}^H representa o vetor hermitiano de \mathbf{w} , que é o conjugado complexo de \mathbf{w} transposto, então

$$\mathbf{Q}_w(n) = \sigma_w^2 \mathbf{I}_{(c \times c)}, \quad (3.53)$$

onde \mathbf{I} é uma matriz identidade.

Seja $\mathbf{y}(n)$ um vetor de observações de dimensão d formado de acordo com a equação

$$\mathbf{y}(n) = \mathbf{C}(n) \mathbf{x}(n) + \mathbf{v}(n), \quad (3.54)$$

onde $\mathbf{C}(n)$ é uma matriz variante no tempo de dimensão $d \times c$ e $\mathbf{v}(n)$ é um vetor que representa um ruído branco, com variância σ_v^2 , e é estatisticamente independente de $\mathbf{w}(n)$. A matriz de covariância $\mathbf{Q}_v(n)$ do ruído branco $\mathbf{v}(n)$ é dada por

$$E \{ \mathbf{v}(n) \mathbf{v}^H(k) \} = \mathbf{Q}_v(n) = \begin{cases} \sigma_v^2 & k = n \\ 0 & k \neq n, \end{cases} \quad (3.55)$$

então

$$\mathbf{Q}_v(n) = \sigma_v^2 \mathbf{I}_{(d \times d)}. \quad (3.56)$$

Os estados não são acessíveis, as observações são a única informação disponível do processo. Se $\mathbf{A}(n)$, $\mathbf{C}(n)$, $\mathbf{Q}_w(n)$, e $\mathbf{Q}_v(n)$ são conhecidos, o filtro de Kalman discreto pode ser usado para estimar os estados do processo. Seja $\hat{\mathbf{x}}(n|n)$ a melhor estimativa linear dadas as observações $\mathbf{y}(i)$ para $i = 1, 2, \dots, n$, e seja $\hat{\mathbf{x}}(n|n-1)$ a melhor estimativa linear dadas as observações até o tempo $n-1$. Os erros de estimação de estado correspondentes são

$$\begin{aligned} \mathbf{e}(n|n) &= \mathbf{x}(n) - \hat{\mathbf{x}}(n|n) \\ \mathbf{e}(n|n-1) &= \mathbf{x}(n) - \hat{\mathbf{x}}(n|n-1), \end{aligned} \quad (3.57)$$

e $\mathbf{P}(n|n)$ e $\mathbf{P}(n|n-1)$ são as matrizes de covariância do erro,

$$\begin{aligned} \mathbf{P}(n|n) &= E \{ \mathbf{e}(n|n) \mathbf{e}^H(n|n) \} \\ \mathbf{P}(n|n-1) &= E \{ \mathbf{e}(n|n-1) \mathbf{e}^H(n|n-1) \}. \end{aligned} \quad (3.58)$$

O filtro de Kalman discreto estima os estados do processo recursivamente de acordo com as equações de 3.59 a 3.63.

$$\hat{\mathbf{x}}(n|n-1) = \mathbf{A}(n-1)\hat{\mathbf{x}}(n-1|n-1) \quad (3.59)$$

$$\mathbf{P}(n|n-1) = \mathbf{A}(n-1)\mathbf{P}(n-1|n-1)\mathbf{A}^H(n-1) + \mathbf{Q}_w(n) \quad (3.60)$$

$$\mathbf{K}(n) = \mathbf{P}(n|n-1)\mathbf{C}^H(n) [\mathbf{C}(n)\mathbf{P}(n|n-1)\mathbf{C}^H(n) + \mathbf{Q}_v(n)]^{-1} \quad (3.61)$$

$$\hat{\mathbf{x}}(n|n) = \hat{\mathbf{x}}(n|n-1) + \mathbf{K}(n)[\mathbf{y}(n) - \mathbf{C}(n)\hat{\mathbf{x}}(n|n-1)] \quad (3.62)$$

$$\mathbf{P}(n|n) = [\mathbf{I} - \mathbf{K}(n)\mathbf{C}(n)]\mathbf{P}(n|n-1), \quad (3.63)$$

onde $\mathbf{K}(n)$ é o ganho de Kalman.

Os vetores $\hat{\mathbf{x}}(0|0)$ e $\mathbf{P}(0|0)$ são iniciados com valores aleatórios.

3.7 O modelo ARMA (*Autoregressive Moving Average*)

Um sinal discreto y pode ser representado no instante n por um modelo ARMA(p, q) (*autoregressive moving average*), com ordem AR igual a p e ordem MA igual a q , pela equação 3.64

$$y(n) = -\sum_{j=1}^p a(n)^{(j)}y(n-j) + \sum_{k=1}^q b(n)^{(k)}e(n-k) + e(n), \quad (3.64)$$

onde $a(n)^{(j)}$ e $b(n)^{(k)}$ são os coeficientes das partes AR e MA no instante n e $e(n)$ é um ruído branco (Tarvainen et al., 2004).

Fazendo o vetor de estados igual a

$$\mathbf{x}(n) = \left(-a(n)^{(1)}, \dots, -a(n)^{(p)}, b(n)^{(1)}, \dots, b(n)^{(q)} \right)^T, \quad (3.65)$$

e

$$\mathbf{C}(n) = (y(n-1), \dots, y(n-p), e(n-1), \dots, e(n-q)), \quad (3.66)$$

é possível escrever a equação 3.64 como

$$y(n) = \mathbf{C}(n)\mathbf{x}(n) + e(n), \quad (3.67)$$

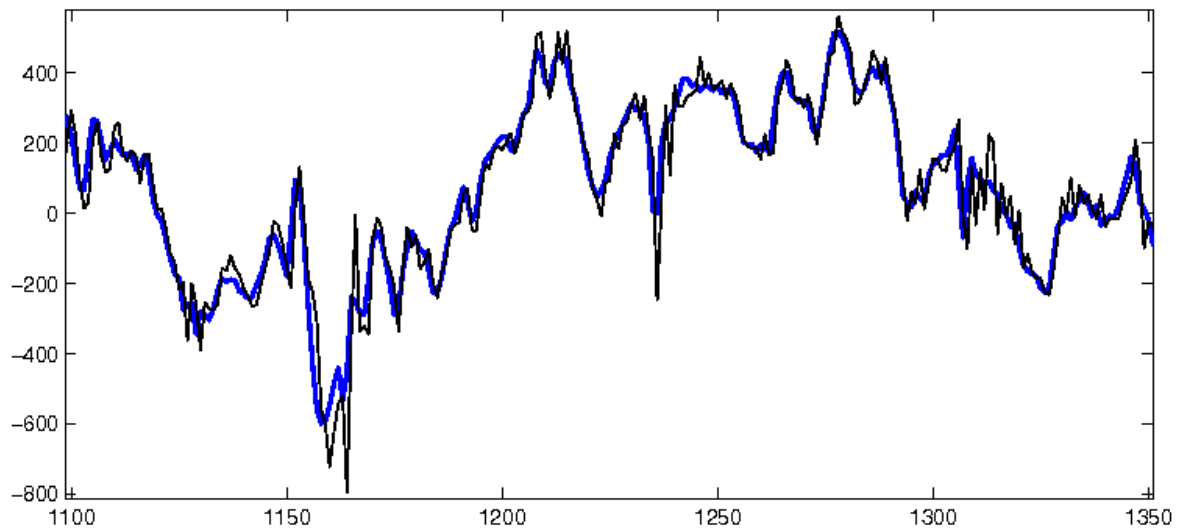
onde $e(n)$ se torna o erro de estimação entre o sinal $y(n)$ e o modelo representado por $\mathbf{C}(n)$ e $\mathbf{x}(n)$. O erro $e(n)$ deve ser calculado a cada instante n .

Para usar o filtro de Kalman discreto para estimar o vetor de estados, a equação de estados deve ser definida como

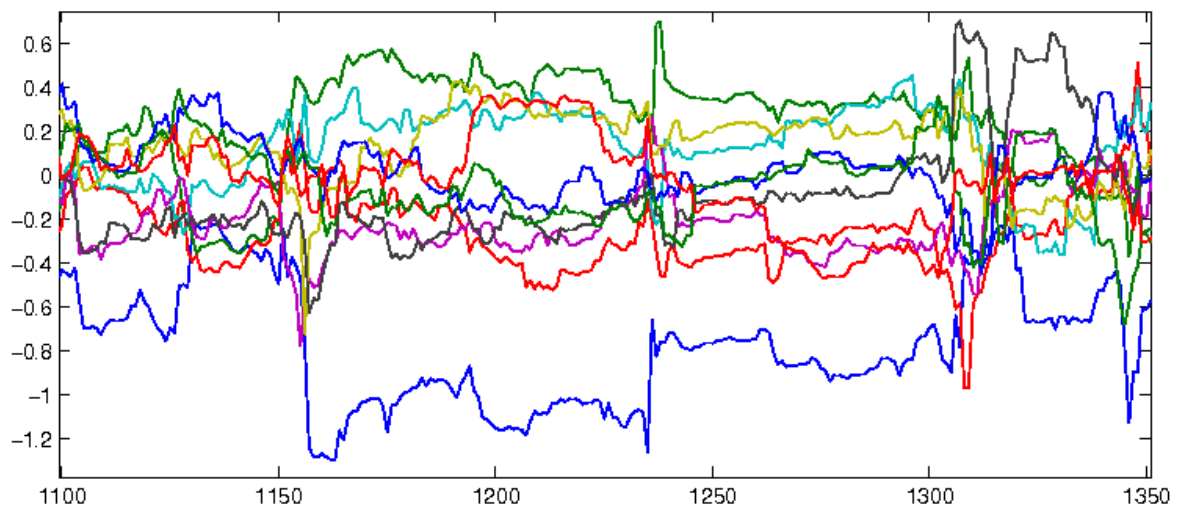
$$\mathbf{x}(n+1) = \mathbf{I}\mathbf{x}(n) + \mathbf{w}(n). \quad (3.68)$$

Comparando (3.68) com (3.51) é possível perceber que a matriz de identidade \mathbf{I} é tomada como matriz de transição de estados \mathbf{A} .

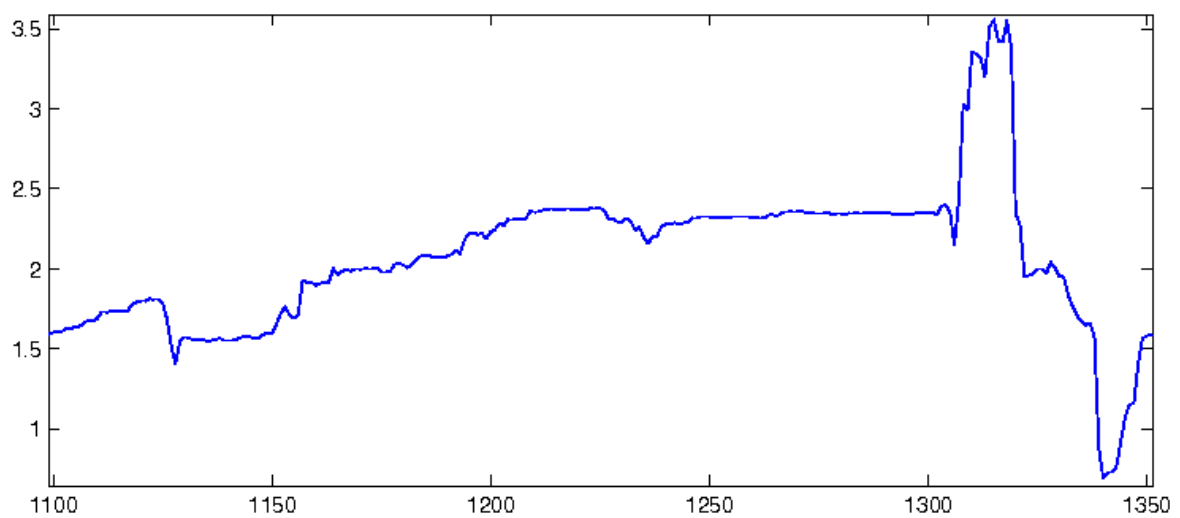
A Figura 3.11a mostra um trecho de 3 segundos (isolado do período de 30 segundos para facilitar a visualização) de um sinal de EEG e o sinal correspondente gerado por um modelo ARMA. As Figuras 3.11b e 3.11c mostram os coeficientes AR e MA respectivamente.



(a) EEG original (linha mais grossa) e o resultado do seu modelo ARMA (linha mais fina).



(b) Coeficientes AR.



(c) Coeficientes MA.

Figura 3.11: Modelo ARMA(10,1) de um sinal de EEG encontrado pelo filtro de Kalman.

3.8 HMM (*Hidden Markov Models*)

Uma das técnicas utilizada neste trabalho é a modelagem do EEG em cada um dos estágios do sono por uma abordagem estatística conhecida como Cadeias Ocultas de Markov. Em uma cadeia de Markov tradicional um sinal estocástico é modelado pelas transições entre estados. Nesse caso, as observações, ao longo do tempo, são associadas à sequência de estados. Porém, essa associação é muito restritiva pela relação estrita entre estados e observações.

Os modelos ocultos de Markov (*Hidden Markov Model*, HMM) constituem uma técnica muito mais eficiente pela suposição de um segundo grau de aleatoriedade. Agora, as observações não estão diretamente conectada à sequência de estados (processo oculto). As observações estão conectadas a um segundo processo que representa as funções de probabilidade de um estado (variável do processo), o que é mais eficiente do que conectar uma observação a um estado específico. Então, a partir de uma sequência de observações O constituída de T vetores O_t , $O = \{O_1, O_2, \dots, O_t, \dots, O_T\}$, $t \in \{1, \dots, T\}$, é possível estimar todos os parâmetros do modelo HMM: a matriz de transição de estados, a lei de probabilidade de emissão de símbolos e a probabilidade inicial de cada estado.

Neste trabalho, a identificação dos estágios do sono se dá pela forma como o modelo HMM representa a sequência de observações dos vetores de características. Nesse caso, os vetores de características são formados pelos coeficientes AR do modelo ARMA do sinal de EEG gerado pelo filtro de Kalman. Alguns aspectos do HMM são apresentados a seguir. Para mais detalhes, consulte (Rabiner, 1989).

O modelo HMM tradicional (Rabiner, 1989) é dado por:

- Uma sequência O (período de 30 segundos de EEG) constituída de T vetores de observação \mathbf{o}_t (vetores linha)

$$O = \begin{bmatrix} \mathbf{o}_1 \\ \vdots \\ \mathbf{o}_t \\ \vdots \\ \mathbf{o}_T \end{bmatrix} \quad (3.69)$$

- Um conjunto de estados q_j , $j \in \{1, \dots, N\}$. Onde j representa o índice dos estados. Em um instante específico $t \in \{1, \dots, T\}$ o modelo HMM se encontra em um estado q_j
- A probabilidade inicial dos estados π

$$\pi = [P(q_1|t=1), P(q_2|t=1), \dots, P(q_N|t=1)]^T. \quad (3.70)$$

- A matriz de transição de estados A de dimensão $N \times N$, onde cada elemento é representado por a_{jl} , $j, l \in \{1, \dots, N\}$ e representa a probabilidade de transição do estado j para o estado l .
- Um conjunto de probabilidade de uma variável observada ser emitida por cada estado: $B = \{P(o_t|q_j)\}$, $t \in \{1, \dots, T\}$ e $j \in \{1, \dots, N\}$. Quanto o modelo HMM é contínuo, $P(o_t|q_j)$ é, geralmente, dado por um modelo de misturas.

Um modelo HMM é, normalmente, representado por um parâmetro $\lambda(A, B, \pi)$. Antes da construção do modelo HMM deve-se definir o número de estados N assim como a sua topologia de interconexão, o que define a estrutura da matriz A . Neste trabalho é utilizada a topologia ergódica, onde é permitida a transição entre todos os estados, o que resulta em uma matriz A completa.

Para encontrar $P(O|\lambda)$ deve-se resolver a equação 3.71

$$P(O|\lambda) = \sum_{\{Q_S\}} P(O, Q_S|\lambda), \quad (3.71)$$

onde $\{Q_S\}$ representa o conjunto de todas as sequências de estados possíveis. A etapa de treinamento, que é baseada na técnica de EM (*expectation-maximization*), é dada por

$$\begin{aligned} \arg \max_{\lambda_i} P(O_i|\omega_i, \lambda_i), \quad \text{o que é equivalente a} \\ \arg \max_{\lambda_i} \sum_{\{Q_S\}} P(O_i, Q_S|\omega_i, \lambda_i), \end{aligned} \quad (3.72)$$

onde ω_i é o i -ésimo estágio do sono.

Na etapa de classificação atribui-se uma classe para cada sequência de observações O . A classificação com M classes, pode ser representada por:

$$i^* = \arg \max_i P(O|\omega_i, \lambda_i, Q_i^*), \quad \forall i \in \{1, \dots, M\}, \quad (3.73)$$

onde i representa os índices das classes de estágio do sono e Q^* é a melhor sequência de estados. Isso significa que para a classificação deve-se encontrar a melhor sequência de estados para o modelo HMM de cada classe.

Em alguns casos é possível atribuir significados físicos aos estados, associando a cada um (ou a um grupo de estados) um significado particular que se baseia em algum conhecimento prévio sobre o sistema sendo modelado. Porém, neste trabalho, não é possível atribuir significado físico aos estados, então outra abordagem deve ser usada. Além disso, o processo de treinamento mostrado pela equação (3.72) requer, de acordo com (Juang e Rabiner,

1990), que todas as possíveis sequências de estados sejam consideradas para o cálculo da probabilidade, o que implica em dificuldades computacionais.

Uma abordagem possível para o treinamento do HMM é conhecido como K-means Segmental. Essa abordagem é baseada na equação (3.71) mudando o forma de calcular $P(O_i|\omega_i, \lambda_i)$, o que é aproximado aqui pela probabilidade dos estados otimizados $P^{max}(O_i|\omega_i, \lambda_i)$, como mostrado na equação abaixo

$$P^{max}(O_i|\omega_i, \lambda_i) = \max_{Q_S} P(O_i, Q_S|\omega_i, \lambda_i). \quad (3.74)$$

Equação (3.74) significa que o cálculo de probabilidade de (3.72) é substituída pelo operador “max” (máximo), em vez da soma sobre todas as sequências de estados possíveis. Então, a sequencia de estados que melhor representa a sequênciade observações O é tomada para estimar o modelo HMM. O treinamento do HMM é dado por:

$$\begin{aligned} \arg \max_{\lambda_i} P(O_i|\omega_i, \lambda_i), \quad \text{o que é equivalente a} \quad (3.75) \\ \arg \max_{\lambda_i} \{ \max_{Q_S} P(O_i, Q_S|\omega_i, \lambda_i) \} \end{aligned}$$

Para o treinamento, a estimativa da melhor sequência Q_i^* e a estimativa dos melhores parâmetros do modelo $\tilde{\lambda}_i$ são realizadas alternadamente. O algoritmo final pode ser escrito como mostrado na equação 3.76

$$\tilde{\lambda}_i = \arg \max_{\lambda_i} \{ \max_{Q_S} P(O_i, Q_S|\omega_i, \lambda_i) \}. \quad (3.76)$$

O processo é iniciado a partir de um modelo inicial λ_i e em iterações subsequentes o modelo é atualizado pela equação (3.76) até que uma condição de parada seja atingida.

Neste estudo cada vetor de observação \mathbf{o}_t é composto pelos coeficientes AR $a(t)^{(j)}$ do modelo ARMA do sinal de EEG na amostra t . Considerando um modelo ARMA(p,q), o vetor de observações é dado por

$$\mathbf{o}_t = [a(t)^{(1)}, \dots, a(t)^{(p)}], \quad (3.77)$$

então a sequência O , dada por (3.69), gerada por uma época de 30 segundos do EEG composto por T amostras é dada por

$$O = \begin{bmatrix} a(1)^{(1)}, \dots, a(1)^{(p)} \\ \vdots \\ a(t)^{(1)}, \dots, a(t)^{(p)} \\ \vdots \\ a(T)^{(1)}, \dots, a(T)^{(p)} \end{bmatrix} \quad (3.78)$$

Para uma classe ω_i , que possui um conjunto de treino composto por S amostras (períodos de 30 segundos de EEG), a sequência de treinamento, $O_{\text{treinamento}\omega_i}$, é construída pela concatenação das sequências de observações dadas pela equação (3.78) para cada conjunto de treinamento, como mostrado a seguir

$$O_{\text{treinamento}\omega_i} = \begin{bmatrix} O_{\omega_i}^{(1)} \\ \vdots \\ O_{\omega_i}^{(s)} \\ \vdots \\ O_{\omega_i}^{(S)} \end{bmatrix} \quad (3.79)$$

onde $s \in \{1, \dots, S\}$.

3.9 Métricas de Desempenho

Um dos principais pontos a se levar em conta em problemas que envolvem a classificação de padrões é a escolha de uma métrica para se avaliar o desempenho do sistema. A escolha de uma métrica deve ser feita de forma a representar adequadamente a qualidade do classificador, sendo minimamente influenciada por características como a diferença entre o número de elementos de cada classe.

3.9.1 Métricas para Classificadores de Duas Classes

Considerando o sistema de classificação de duas classes apresentado na Tabela 3.1. Uma classe com a presença de uma característica (positivo) e uma classe com a ausência dessa característica (negativo).

Tabela 3.1: Matriz de confusão genérica para duas classes.

		Classe atribuída		
		positivo	negativo	
Classe real	positivo	VP verdadeiro positivo	FN falso negativo	Sensibilidade = VP/(VP+FN)
	negativo	FP falso positivo	VN verdadeiro negativo	Especificidade = VN/(FP+VN)
		PPV = VP/(VP+FP)	NPV = VN/(FN+VN)	

Abaixo são apresentadas algumas medidas de desempenho comumente utilizadas

Sensibilidade: também conhecida como *recall* ou **taxa de verdadeiro positivo** (*true positive rate*), mede a capacidade do classificador indicar a presença da característica entre os elementos que a possuem.

$$\text{sensibilidade} = \frac{\text{acertos positivos}}{\text{total de positivos}} = \frac{VP}{VP + FN}. \quad (3.80)$$

Especificidade: mede a capacidade do sistema identificar a ausência da característica entre os elementos que realmente não a possuem.

$$\text{especificidade} = \frac{\text{acertos negativos}}{\text{total de negativos}} = \frac{VN}{VN + FP}. \quad (3.81)$$

Exatidão (Accuracy): também conhecida como **taxa de acerto**, esse valor mostra a proporção de classificações corretas. Esse valor é altamente influenciado pelo desbalanceamento entre as classes do conjunto de dados, podendo fornecer informações erradas sobre o desempenho do classificador.

$$\text{exatidão} = \frac{\text{Total de acertos}}{\text{Total de elementos avaliados}} = \frac{VP + VN}{VP + VN + FP + FN}. \quad (3.82)$$

Valor Preditivo Positivo (Positive Predictive Value - PPV): também conhecido como *precision*, indica a proporção dos elementos classificados como possuidores da característica que realmente a possuem.

$$\text{PPV} = \frac{\text{acertos positivos}}{\text{total de predições positivas}} = \frac{VP}{VP + FP}. \quad (3.83)$$

Valor Preditivo Negativo (Negative Predictive Value - NPV) indica a proporção dos verdadeiros negativos em relação a todas as classificações negativas.

$$\text{NPV} = \frac{\text{acertos negativos}}{\text{total de predições negativas}} = \frac{VN}{VN + FN}. \quad (3.84)$$

Taxa de falsos positivos (false positive rate) : também conhecido como **taxa de falsos alarmes**, informa a proporção das amostras sem a característica classificadas como possuidoras dessa.

$$\text{taxa de falsos positivos} = \frac{\text{falsos positivos}}{\text{total de negativos}} = \frac{FP}{VN + FN} = 1 - \text{especificidade}. \quad (3.85)$$

Índice de Youden (J): os valores de sensibilidade e especificidade indicam tendências opostas de um classificador. Classificadores que tendem a gerar muitos valores positivos tendem a gerar muitos valores de verdadeiros positivos, o que aumenta a sensibilidade. Porém também geram muitos falsos positivos, o que diminui a especificidade. Esse contrabalanço também ocorre quando há a tendência de se gerar muitos valores negativos. O índice de Youden, representado por J , é uma grandeza que leva em conta os valores de sensibilidade e de especificidade, portanto pode indicar o grau de equilíbrio do classificador em relação às duas classes. Essa grandeza varia entre -1 e 1. Quanto maior o valor melhor o desempenho. Valores abaixo de 0 indicam eficácia não superior à classificação ao acaso.

$$J = \text{sensibilidade} + \text{especificidade} - 1. \quad (3.86)$$

3.9.2 Curva de ROC (*Receiver Operating Characteristic*)

A curva ROC é um gráfico para a visualização do desempenho de classificadores. Essa curva mostra a relação de ganhos e perdas entre a taxa de verdadeiros positivos e a taxa de falsos alarmes. A curva ROC é muito utilizada em sistemas que envolvem a escolha entre duas classes possíveis, sendo muito utilizada para a avaliação de sistemas médicos de diagnóstico (Fawcett, 2006), (da Silva Braga, 2000), considerando um sistema que deve analisar a presença de uma característica (classificação positiva, classe ω_1) ou a ausência dessa característica (classificação negativa, classe ω_2). Uma possibilidade de classificação para esse sistema é, a partir do vetor de características de uma amostra, fornecer um valor x . A escolha da classe a ser atribuída à amostra é dada por

$$\text{classe atribuída} = \arg \max_{\omega_i} P(x|\omega_i). \quad (3.87)$$

Essa regra de decisão pode ser representada pelo limiar $x = c$ acima do qual se classifica a amostra como pertencente à classe ω_1 . Esse limiar de decisão é mostrado na Figura 3.12. Como as funções de densidade de probabilidade (FDP) das duas classes se interceptam existe a possibilidade do classificador cometer erros. A área sombreada I corresponde à probabilidade do classificador cometer um erro de falso positivo. Já a área sombreada II corresponde à probabilidade de um erro de falso negativo.

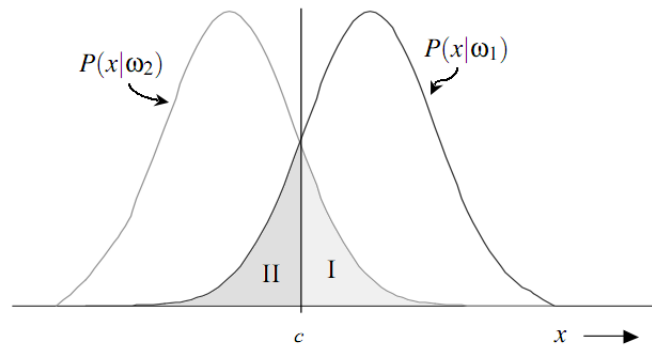


Figura 3.12: Exemplo de um limiar de decisão para curva ROC.

A curva ROC analisa a capacidade de separação das duas classes realizada pelo classificador. Isso é feito pela escolha de vários limiares de decisão e observando para cada limiar o desempenho de classificação. Um exemplo da escolha de outro limiar de decisão é mostrado na Figura 3.13. Nessa figura pode-se ver como as probabilidades de falsos positivos e falsos negativos são alteradas pela mudança do limiar de decisão.

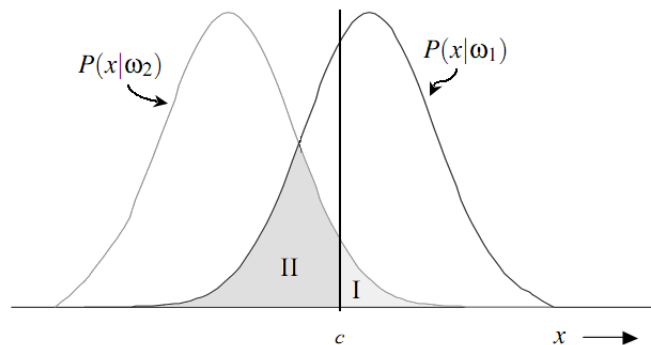
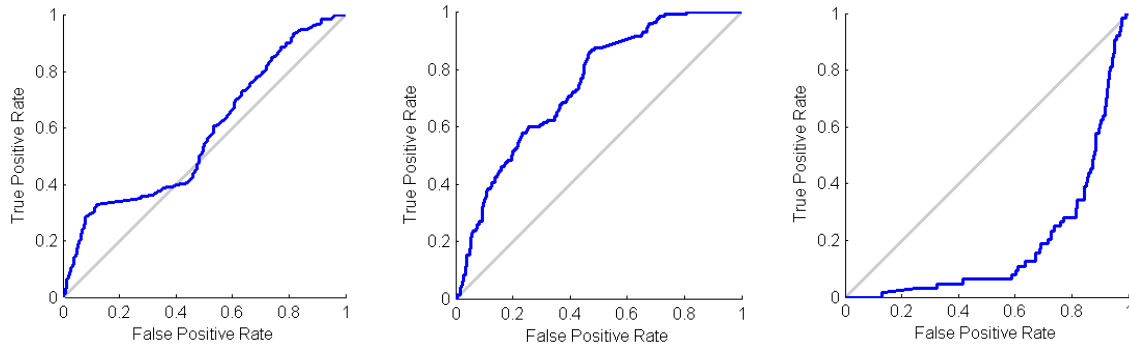


Figura 3.13: Exemplo de um limiar de decisão deslocado para curva ROC.

A avaliação do desempenho do classificador é feita pela curva que mostra a taxa de verdadeiros positivos (sensibilidade) no eixo “y” e a taxa de falsos positivos (1-especificidade) no eixo “x”. A curva é traçada plotando esses valores para diferentes limiares de decisão escolhidos.

Normalmente é desenhada a linha $y = x$ para auxiliar a análise do gráfico, essa linha corresponde à classificação aleatória, ou seja, o classificador possui o mesmo desempenho que a atribuição de classe ao acaso. Um exemplo de curva de um classificador com desempenho próximo à classificação aleatória é apresentado na Figura 3.14a. Um classificador que apresente alguma capacidade de classificação superior à classificação aleatória apresenta a curva



(a) Desempenho de classificação próximo ao aleatório.

(b) Desempenho de classificação superior ao aleatório.

(c) Desempenho de classificação inferior ao aleatório.

Figura 3.14: Exemplos de curvas ROC.

no triângulo superior, acima da reta $y = x$, como mostrado na Figura 3.14b. Um classificador ideal apresenta a uma curva constituída por uma reta do ponto (0,0) até o ponto (0,1) e outra do ponto (0,1) ao ponto (1,1). Normalmente a curva ROC não passa pelo triângulo inferior. Um classificador com a curva no triângulo inferior, como apresentado na Figura 3.14c, possui informação útil à classificação, porém, utiliza essa informação de forma errada. Um parâmetro comumente utilizado para avaliar o desempenho de um classificador é a área sob a curva ROC, (*Area Under the ROC curve* - AUC). A área sob a curva está entre 0 e 1. Quanto maior a área sob a curva melhor o desempenho do classificador. Um classificador com desempenho aleatório possui área 0,5, portanto é de se esperar que um classificador não tenha área inferior a 0.5. Um classificador ideal possui a área sob a curva igual a 1.

3.9.3 Coeficiente Kappa de Cohen κ

O coeficiente Kappa de Cohen, κ , foi criado como medida de concordância entre observações de comportamentos psicológicos. A ideia original do coeficiente kappa de Cohen é medir o grau de concordância, ou discordância, entre duas pessoas observando o mesmo fenômeno (Mercy, 2010) (Fernández et al., 2010) (Ben-David, 2007).

Supondo que dois sujeitos devessem classificar objetos nas categorias 1 ou 2. A Tabela 3.2 apresenta as probabilidades p de rotulação pelos dois sujeitos.

Tabela 3.2: Matriz de probabilidade de rotulação.

		sujeito A		
		1	2	total
sujeito B	1	p_{11}	p_{12}	p_{B1}
	2	p_{21}	p_{22}	p_{B2}
	total	p_{A1}	p_{A2}	1

Para calcular o valor de Kappa deve-se primeiramente calcular o nível de concordância observado, dado por:

$$p_0 = p_{11} + p_{22}. \quad (3.88)$$

Esse valor deve ser comparado ao valor de concordância que se deveria esperar se os dois sujeitos respondessem de forma totalmente independente, dado por

$$p_e = p_{A1}p_{B1} + p_{A2}p_{B2}. \quad (3.89)$$

O valor de Kappa é, então, definido como

$$\kappa = \frac{p_0 - p_e}{1 - p_e}. \quad (3.90)$$

O valor de Kappa é sempre menor que 1. Kappa igual a 1 representa a concordância perfeita. Valores negativos de Kappa dificilmente são obtidos, esses valores correspondem a uma concordância pior do que aquela obtida ao acaso.

O exemplo dado foi com duas classes (duas categorias), porém o procedimento apresentado para se calcular o valor de Kappa pode ser estendido para se trabalhar com mais classes. Também pode-se trabalhar com o número de classificações realizadas no lugar de se trabalhar com as probabilidades de classificação.

Vamos considerar a matriz de confusão apresentada na Tabela 3.3, que representa um sistema de classificação com m classes.

Nesse caso o coeficiente Kappa de Cohen é dado por

$$\kappa = \frac{T \sum_{i=1}^m h_{ii} - \sum_{i=1}^m T_{ri}T_{ci}}{T^2 - \sum_{i=1}^m T_{ri}T_{ci}} \quad (3.91)$$

Tabela 3.3: Matriz de confusão genérica de uma classificação com m classes.

		Classe predita				total
		C1	C2	...	Cm	
Classe correta	C1	h_{11}	h_{12}	...	h_{1m}	T_{r1}
	C2	h_{21}	h_{22}		h_{2m}	T_{r2}
	⋮	⋮		⋱		⋮
	Cm	h_{m1}	h_{m2}		h_{mm}	T_{rm}
	total	T_{c1}	T_{c2}	...	T_{cm}	T

3.9.4 Exatidão para várias classes

A medida de exatidão (*accuracy*), ou taxa de acerto, representada pela equação 3.82, pode ser aplicada a várias classes. Considerando a matriz de confusão de um classificador com m classes apresentado na Tabela 3.3, o valor da exatidão dessa classificação é dada por

$$exatidão = \frac{\sum_{i=1}^m h_{ii}}{T}. \quad (3.92)$$

Um grave problema apresentado pela medida de exatidão é a sua vulnerabilidade ao número de amostras disponíveis em cada classes. Em (Ben-David, 2007) é discutida a influência do acaso na medida da exatidão e como coeficiente Kappa de Cohen lida com esse problema. Portanto, o kappa de Cohen é uma medida de desempenho mais adequada.

3.9.5 Medidas discriminantes utilizadas neste trabalho

Como esse trabalho busca a classificação de dados rotulados, deve-se utilizar alguma medida de desempenho para sua avaliação e, posteriormente, possíveis comparações com outros trabalhos. Também, no meio do processo, deve-se comparar o desempenho de diferentes classificadores a fim de se determinar uma arquitetura mais eficiente. Como, por exemplo, para se definir o número de neurônios da camada oculta de um classificador *perceptron*.

Como medida de desempenho principal será utilizado o coeficiente Kappa de Cohen, por se julgar uma das medidas mais robustas disponíveis para classificadores que trabalham com várias classes. O coeficiente Kappa de Cohen será utilizado para realizar as comparações de desempenho necessárias. Também será apresentado nos testes finais de desempenho, depois que o sistema estiver sido finalizado, os valores de exatidão (*accuracy*) para facilitar a comparação com outros trabalhos. Nos exemplos de classificação que apresentarem a matriz de

confusão serão apresentados os valores de sensibilidade e especificidade de cada classe. Para esses exemplos, nos casos da classificação com SVM e *perceptron*, também serão mostradas as curvas ROC. As curvas ROC serão traçadas para cada classe separadamente. Considerando as possibilidades da amostra pertencer à classe analisada no momento, ou pertencer à uma das outras classes.

Existem técnicas que apresentam a área sob a curva ROC para avaliação de classificadores multi-classes, como apresentado em (Hand e Till, 2001). Porém, como essas técnicas ainda não estão amplamente difundidas, optou-se por utilizar o coeficiente Kappa de Cohen como a medida de desempenho principal deste trabalho.

Capítulo 4

Testes Resultados

Nesta seção são apresentados, primeiramente, os teste realizados com *wavelet packet* e, posteriormente, os teste realizados com a aplicação de HMM.

Antes da apresentação dos testes com as duas técnicas é apresentada a forma de composição dos conjuntos de treino e de teste. Para possibilitar a comparação, os testes das duas técnicas adotam o mesmo sistema de formação dos conjuntos de treino e de teste.

Os algoritmos deste trabalho foram desenvolvidos no MatlabTM.

4.1 Formação dos Conjuntos de Treino e de Teste

O banco de dados utilizado possui 2020 períodos de 30 segundos de sinais de EEG classificados como pertencente a um estágio do sono, distribuído da seguinte forma: 810 períodos classificados como vigília relaxada (*wakefulness*), classe W; 285 períodos classificados como estágio 1, classe 1; 610 períodos classificados como estágio 2, classe 2; 103 períodos classificados como estágio 3, classe 3; 83 períodos classificados como estágio 4, classe 4; 128 períodos classificados como estágio REM, classe R e uma amostra não classificada como pertencente a um estágio do sono. Esses sinais são registrados com uma taxa de amostragem de 250 amostras por segundo.

O banco de dados é dividido em dois grupos, cada um com metade dos elementos (períodos de 30 segundos) de cada classe. O primeiro grupo é tomado como **conjunto de treino**, todos os elementos que compõem o conjunto de treino são utilizados para a escolha da melhor base da *wavelet packet*. Do conjunto de treino também são tomadas, aleatoriamente, 41 elementos de cada classe para compor o **conjunto de treino dos classificadores**, usado para

treinar os classificadores KNN, SVM, *perceptron* de multicamadas e para treinar os modelos HMM. O outro grupo foi tomado totalmente para compor o **conjunto de teste**, como mostrado na Tabela 4.1. Esse processo garante que os conjuntos de treino e de teste não possuam elementos em comum, ou seja, os elementos utilizados para o teste de desempenho não são utilizados para a escolha da base *wavelet packet*, para o treino dos classificadores ou para o treino dos modelos HMM. A cada ciclo de treino e teste esse processo de divisão dos conjuntos é repetido.

Tabela 4.1: Composição dos conjuntos de treino e de teste.

	Conj. de treino (Conj. de treino dos classificadores)	Conj. de teste	Total
classe W	405 (41)	405	810
classe 1	142 (41)	143	285
classe 2	305 (41)	305	610
classe 3	51 (41)	52	103
classe 4	41 (41)	42	83
classe R	64 (41)	64	128

4.2 Testes com *Wavelet Packet*

4.2.1 Escolha da Função *Wavelet* e do Número de Nós

Quando se trabalha com a transformada *wavelet* deve-se escolher a função *wavelet*, ψ , a ser utilizada. Cada função ψ possui uma função de escalamento ϕ associada. Neste trabalho a decomposição do sinal foi realizada pela transformada *wavelet packet* com 9 camadas.

Outro fator que deve ser definido para os testes é o número de nós utilizados para compor os vetores de características. Como apresentado na seção 3.4.4, o algoritmo de escolha dos nós seleciona, automaticamente, p nós para compor a base mais discriminante. Porém deve-se escolher os q nós mais discriminantes para compor os vetores de características ($q < p$). Isso se deve ao fato de a adição de um número exagerado de nós diminuir o desempenho dos classificadores.

Para a escolha da função *wavelet* e do número de nós a serem utilizados usou-se como parâmetro o desempenho do classificador SVM, isso porque foi o classificador que apresentou melhor desempenho em testes prévios.

Foram testadas as seguintes famílias *wavelets*: Daubechies, Symlets, Coiflets e Discrete Meyer. O número q de nós utilizados na classificação foram: 5, 10, 15, 20, 25, 30, 35, 40. As

Tabelas 4.2, 4.3, 4.4 e 4.5 apresentam os desempenhos para as funções ψ testadas em cada uma das famílias. A notação do nome da função *wavelet* apresentada é aquela adotada pelo comando “wfilters” do MatlabTM. O valor apresentado em cada célula corresponde à média do kappa de Cohen, para 10 ciclos de treinamento e teste, com os parâmetros avaliados. Em cada ciclo são selecionados novos conjuntos de treinamento e de teste de acordo com a seção 4.1. O maior valor da média do kappa de Cohen para cada família *wavelet* é apresentado em negrito.

Tabela 4.2: Teste de desempenho para família *wavelet* Daubechies. Média do kappa de Cohen para 10 ciclos de treinamento e teste.

		Número de nós utilizados							
		5	10	15	20	25	30	35	40
ψ	db1	0.2515	0.4058	0.4280	0.4265	0.4281	0.4275	0.4310	0.4299
	db2	0.3437	0.4493	0.4516	0.4451	0.4458	0.4439	0.4437	0.4399
	db3	0.3224	0.4596	0.4494	0.4461	0.4432	0.4411	0.4385	0.4350
	db4	0.3447	0.4444	0.4662	0.4593	0.4511	0.4539	0.4480	0.4442
	db5	0.3754	0.4560	0.4535	0.4462	0.4496	0.4444	0.4419	0.4355
	db6	0.3907	0.4395	0.4414	0.4509	0.4504	0.4495	0.4478	0.4502
	db7	0.3980	0.4516	0.4714	0.4731	0.4730	0.4689	0.4620	0.4600
	db8	0.3903	0.4339	0.4353	0.4310	0.4324	0.4275	0.4127	0.4287
	db9	0.3924	0.4497	0.4569	0.4580	0.4556	0.4559	0.4572	0.4544
	db10	0.3897	0.4499	0.4573	0.4567	0.4642	0.4599	0.4566	0.4511
	db11	0.3310	0.4311	0.4556	0.4539	0.4504	0.4481	0.4475	0.4344
	db12	0.3907	0.4122	0.4400	0.4378	0.4359	0.4368	0.4312	0.4301
	db13	0.3795	0.4397	0.4655	0.4582	0.4535	0.4451	0.4394	0.4380
	db14	0.3920	0.4375	0.4388	0.4532	0.4454	0.4446	0.4446	0.4251
	db15	0.3935	0.4580	0.4675	0.4665	0.4607	0.4567	0.4545	0.4560

Tabela 4.3: Teste de desempenho para família *wavelet* Symlets. Média do kappa de Cohen para 10 ciclos de treinamento e teste.

		Número de nós utilizados							
		5	10	15	20	25	30	35	40
⊃	sym2	0.2898	0.4049	0.4559	0.4566	0.4529	0.4516	0.4502	0.4513
	sym3	0.3214	0.4380	0.4617	0.4521	0.4491	0.4477	0.4481	0.4443
	sym4	0.3649	0.4451	0.4703	0.4709	0.4624	0.4659	0.4595	0.4575
	sym5	0.3479	0.4331	0.4581	0.4587	0.4555	0.4582	0.4558	0.4480
	sym6	0.3582	0.4224	0.4655	0.4670	0.4630	0.4601	0.4557	0.4547
	sym7	0.3698	0.4424	0.4567	0.4535	0.4447	0.4419	0.4396	0.4386
	sym8	0.3747	0.4139	0.4361	0.4541	0.4495	0.4483	0.4463	0.4436
	sym9	0.3791	0.4422	0.4572	0.4653	0.4672	0.4646	0.4618	0.4570
	sym10	0.3707	0.4280	0.4582	0.4574	0.4575	0.4620	0.4594	0.4560
	sym11	0.3981	0.4380	0.4637	0.4581	0.4577	0.4545	0.4551	0.4542
	sym12	0.3875	0.4311	0.4648	0.4788	0.4814	0.4702	0.4691	0.4627
	sym13	0.3856	0.4715	0.4954	0.5026	0.4949	0.4877	0.4811	0.4815
	sym14	0.3689	0.4445	0.4558	0.4775	0.4728	0.4713	0.4677	0.4609
	sym15	0.3897	0.4545	0.4753	0.4700	0.4712	0.4694	0.4683	0.4648

Tabela 4.4: Teste de desempenho para família *wavelet* Coiflets. Média do kappa de Cohen para 10 ciclos de treinamento e teste.

		Número de nós utilizados							
		5	10	15	20	25	30	35	40
⊃	coif1	0.3273	0.4182	0.4521	0.4450	0.4502	0.4413	0.4374	0.4345
	coif2	0.3564	0.4494	0.4550	0.4601	0.4586	0.4512	0.4502	0.4518
	coif3	0.4054	0.4318	0.4524	0.4549	0.4499	0.4443	0.4411	0.4343
	coif4	0.3902	0.4314	0.4641	0.4717	0.4700	0.4664	0.4643	0.4604
	coif5	0.3815	0.4296	0.4618	0.4690	0.4675	0.4669	0.4655	0.4647

Tabela 4.5: Teste de desempenho para família *wavelet* "Discrete" Meyer. Média do kappa de Cohen para 10 ciclos de treinamento e teste.

		Número de nós utilizados							
		5	10	15	20	25	30	35	40
⊃	dmey	0.3668	0.4245	0.4702	0.4763	0.4690	0.4668	0.4661	0.4584

Observa-se que o melhor desempenho foi conseguido com a função *wavelet* da família Symlets, sym13, com a utilização dos 20 nós mais discriminantes para a composição dos

vetores de característica. Portanto, daqui em diante todos os testes com *wavelet packet* utilizarão como função ψ a *wavelet sym13* e tomarão os 20 nós mais discriminantes para compor os vetores de características.

4.2.2 Configuração da Rede *Perceptron* de Múltiplas Camadas

Nesse trabalho adotou-se um rede perceptron com uma camada oculta. Como descrito na seção 4.2.1 foram tomados os 20 nós mais discriminantes para compor os vetores de características $\mathbf{x} = (x_1 \dots x_{20})$. Como existem seis classes possíveis a rede apresenta saída de dimensão seis, $\mathbf{y} = (y_1 \dots y_6)$. A estrutura utilizada com M neurônios na camada oculta é representada na Figura 4.1.

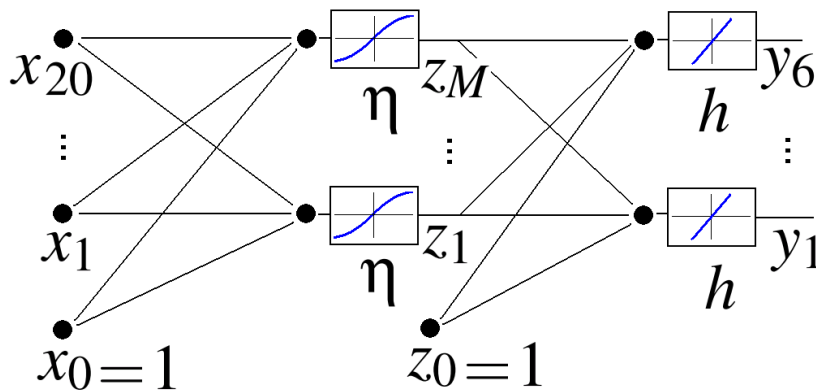


Figura 4.1: Estrutura da rede *perceptron* utilizada.

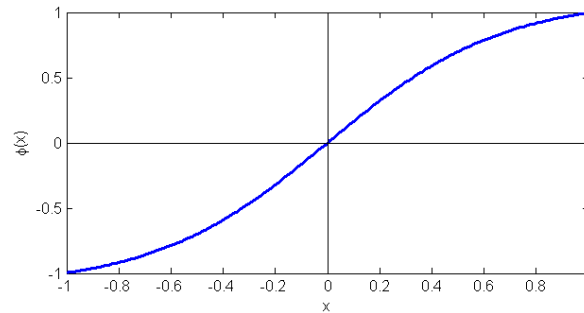
Como função de ativação da camada de saída adotou-se a função linear h dada pela equação 4.1. Na camada oculta adotou-se uma função η , dada pela equação 4.2.

$$h(x) = x; \quad (4.1)$$

$$\eta(x) = a * \tanh(b * x), \quad (4.2)$$

onde : \tanh é a função tangente hiperbólica
 a é um fator de controle de amplitude.
 b controla a velocidade de subida de função.

Por meio de testes manuais adotou-se os valores $a = 1.1$ e $b = 1.5$. Com esses valores a função η apresenta a forma mostrada na Figura 4.2.

Figura 4.2: Função de ativação η .

Diferentes combinações de funções de ativação foram testadas, com as funções tangente hiperbólica, logística e linear. Essa configuração com a função tangente hiperbólica alterada pelos fatores a e b na camada oculta e com a função linear na camada de saída foi a que ofereceu melhor desempenho.

A rede foi implementada no MatlabTM, utilizando a técnica de retropropagação do erro (*backpropagation*) para realizar o treinamento. O algoritmo de retropropagação do erro implementado se baseia em (Haykin, 2001).

Adotou-se o treinamento sequencial, com o embaralhamento das amostras do conjunto de treino ao final de cada época. O treinamento da rede consistiu na execução de 100 épocas. Antes da execução da primeira época, os pesos dos neurônios são inicializados aleatoriamente. Em cada época é calculado o valor médio de $\mathcal{E}(n)$, dado pela equação 3.28, de todas as amostras do conjunto de treino (erro médio). Quando o valor do erro médio não sofre melhora por duas épocas seguidas os pesos dos neurônios são novamente inicializados aleatoriamente.

O melhor conjunto de pesos encontrado, aquele que forneceu o menor valor médio de $\mathcal{E}(n)$, é mantido armazenado durante o processo e é tomado como resultado do treinamento.

A Figura 4.3 mostra a evolução do erro médio ao longo das 100 épocas que compõem o treinamento onde os picos correspondem aos pontos de reinicialização dos pesos dos neurônios. Observa-se nessa figura o erro médio diminuindo pela ação do algoritmo de retropropagação do erro.

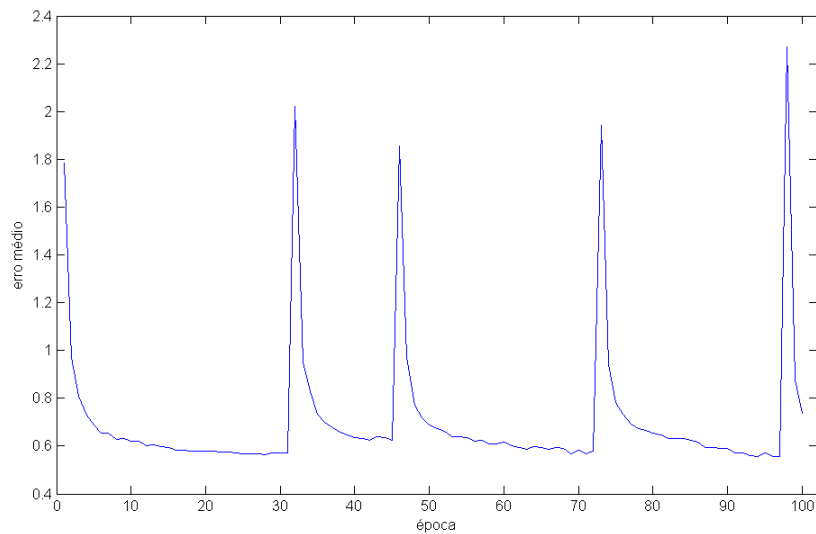


Figura 4.3: Erro médio de classificação durante o treinamento do *perceptron* com o algoritmo de retropropagação do erro.

O número M de neurônios da camada oculta foi definido por meio de teste. Para cada valor de M testado executou-se 50 ciclos de treinamento e teste, separando os conjuntos de treinamento e de teste a cada ciclo. A média dos valores do kappa de Cohen obtidos, juntamente com os desvios padrão, são apresentados na Figura 4.4.

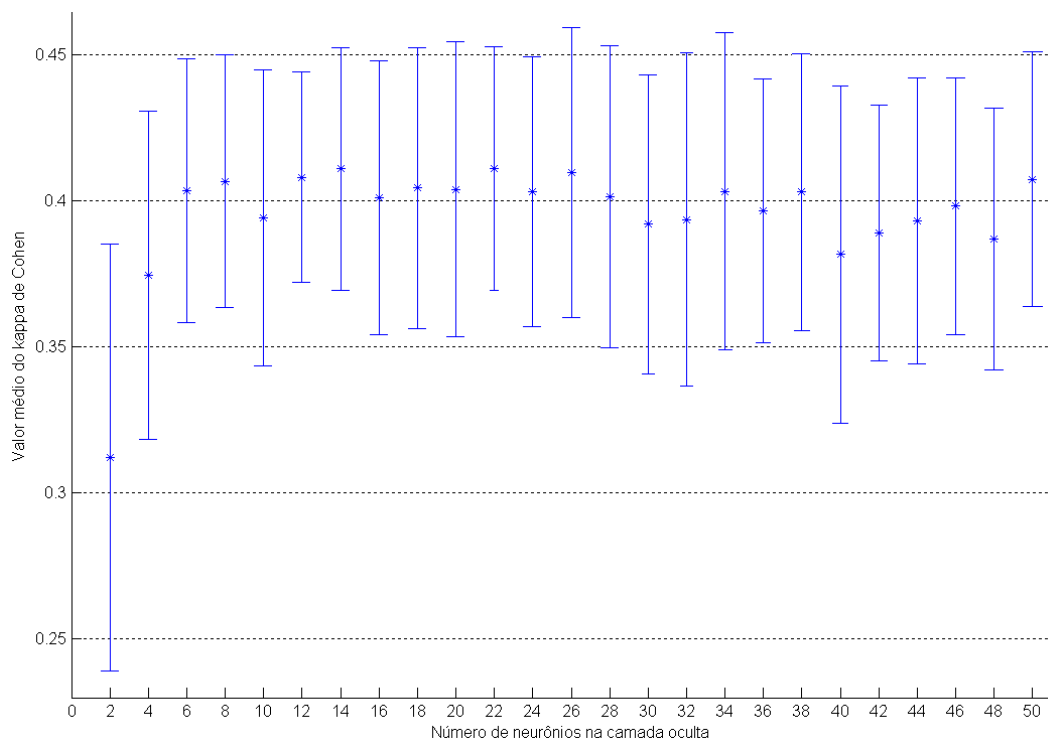


Figura 4.4: Valor médio do kappa de Cohen, com desvio padrão, (em 50 ciclos de treinamento e teste) para diferentes número de neurônios da camada oculta da rede *perceptron*.

Nesse teste o melhor desempenho foi obtido com $M = 14$ e $M = 22$, em ambos os casos obteve-se ($\kappa = 0,4109$). Como o teste com $M = 14$ forneceu um desvio padrão levemente menor, e também por formar uma estrutura mais simples com desempenho semelhante, adotou-se esse valor. Portanto, adotou-se a estrutura da rede *perceptron* com 14 neurônios na camada oculta.

4.2.3 Definição do Valor de k para o Classificador KNN

Como visto na seção 3.5.2, a classificação de uma amostra pelo classificador KNN se baseia na análise de vizinhança composta por k amostras rotuladas. Como parâmetro para esse classificador deve-se informar o valor de k . Esse teste buscou estimar um valor adequado para k . Para isso se executou 50 ciclos de treinamento e teste, com escolha dos conjuntos de treinamento e de teste a cada ciclo. A média dos valores do kappa de Cohen obtidos, juntamente com os desvios padrão, são apresentados na Figura 4.5.

Nesse teste o melhor desempenho de classificação foi obtido com os valores de k variando de 6 a 12, com κ assumindo valores entre 0,4262 e 0,4275. Escolheu-se $k = 6$, por ser a configuração, dentre as anteriormente citadas, que apresentou menor desvio padrão do valor de κ . Portanto, esse será o valor adotado nos testes.

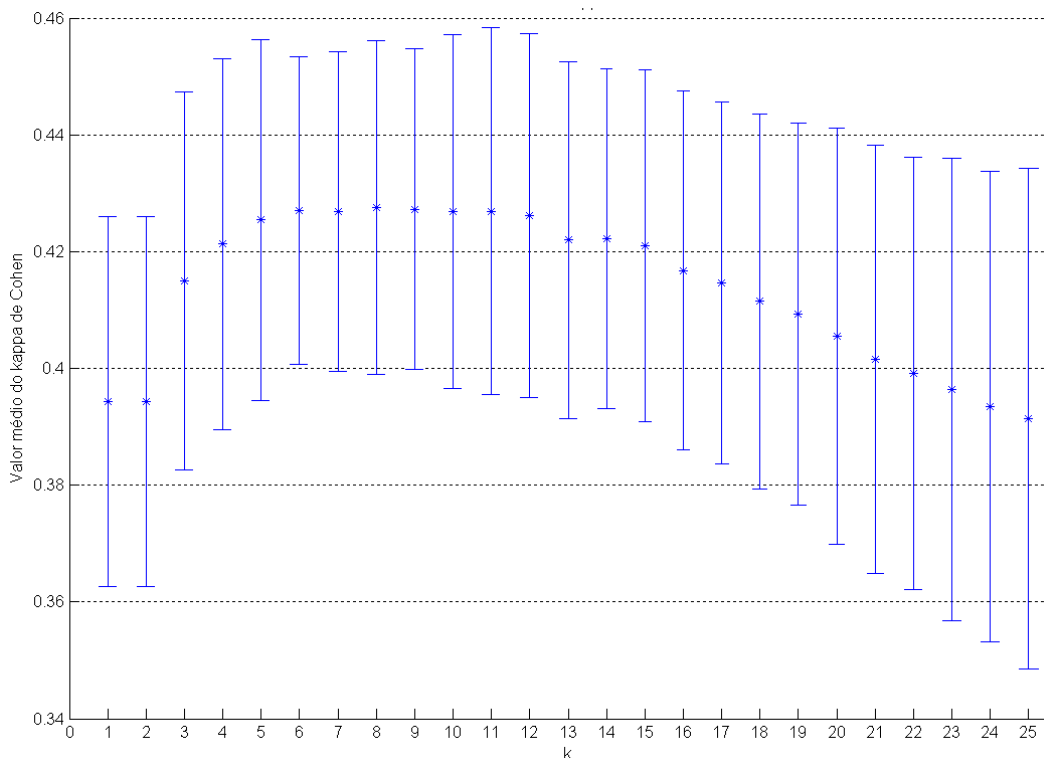


Figura 4.5: Valor médio do kappa de Cohen, com desvio padrão, (em 50 ciclos de treinamento e teste) para diferentes valores de k com o classificador KNN.

4.2.4 Configuração dos Parâmetros do Classificador SVM

Para a classificação com SVM, utilizou-se a biblioteca libSVM (Chang e Lin, 2001). Para a utilização dessa biblioteca deve-se fazer algumas escolhas e configurar alguns parâmetros. As configurações apresentadas aqui foram realizadas manualmente de forma a melhorar o desempenho do classificador. A seguir é apresentada uma breve descrição de cada parâmetro utilizado com o valor escolhido.

-s tipo de SVM : esse parâmetro foi configurado como 0, definindo a escolha do C-SVM.

-t tipo do kernel : configurado como 2, o que representa a utilização de uma função de base radial como kernel k apresentado a seguir

$$k(u, v) = e^{(-\text{gamma}|u-v|^2)}; \quad (4.3)$$

-g gamma : parâmetro gamma da função kernel, mostrada na equação 4.3, configurado com o valor $g = 1,0$;

-c custo : define o valor do custo para amostras que não são corretamente classificadas durante o treinamento, configurado com o valor $c = 1,5$;

-b estimativa de probabilidade : esse parâmetro deve assumir os valores 0 ou 1, somente configurando-o como 1 se conseguiu a classificação.

4.2.5 Resultados *Wavelet Packet*

Para os testes foi utilizada a função *wavelet* da família Symlets sym13, tomando os 20 nós mais discriminantes para realizar a classificação, essas escolhas foram feitas por meio do teste de sensibilidade descrito na seção 4.2.1. Os testes de classificação foram realizados com os classificadores *perceptron* de múltiplas camadas, KNN, e SVM com as configurações definidas nas seções 4.2.2, 4.2.3 e 4.2.4.

Nos testes foram executados 100 ciclos de treinamento e teste, onde cada ciclo corresponde à separação dos conjuntos de treino e de teste, como descrito na seção 4.1, seguida pela escolha da base mais discriminante e pela seleção dos nós da base encontrada para a classificação, e pelo treinamento do classificador com o conjunto de treino e pela classificação dos dados do conjunto de teste. Nesse teste foram utilizados os 20 nós mais discriminantes para realizar a classificação. A cada ciclo os 3 classificadores são testados com os mesmos conjuntos de treino e de teste, de forma que a diferença de desempenho entre eles não seja devido à diferença dos dados.

A média dos valores do kappa de Cohen obtidos nos 100 ciclos de treinamento e teste são apresentados na Figura 4.6, juntamente com os desvios padrão, para os três classificadores testados. Os histogramas com os valores do kappa de Cohen obtidos para cada um dos classificadores são apresentados na Figura 4.7

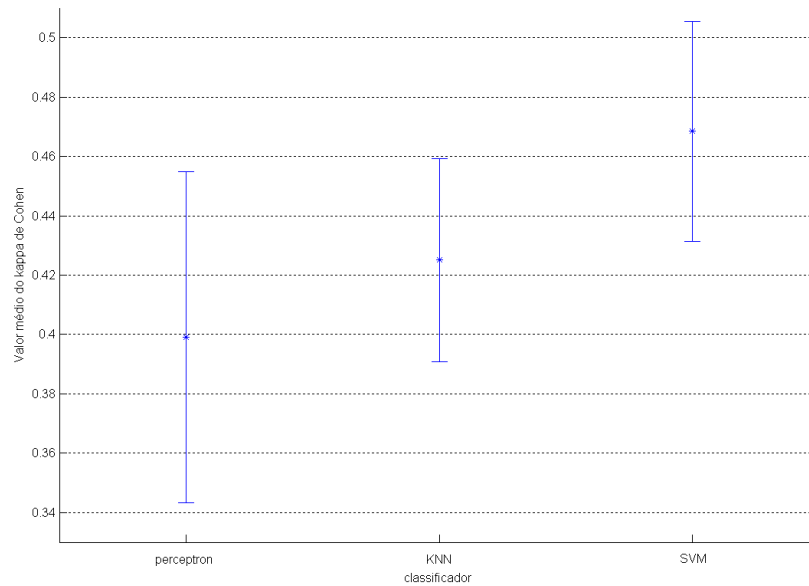


Figura 4.6: Valores médios do kappa de Cohen com desvio padrão (para 100 ciclos de treinamento e teste) dos três classificadores testados.

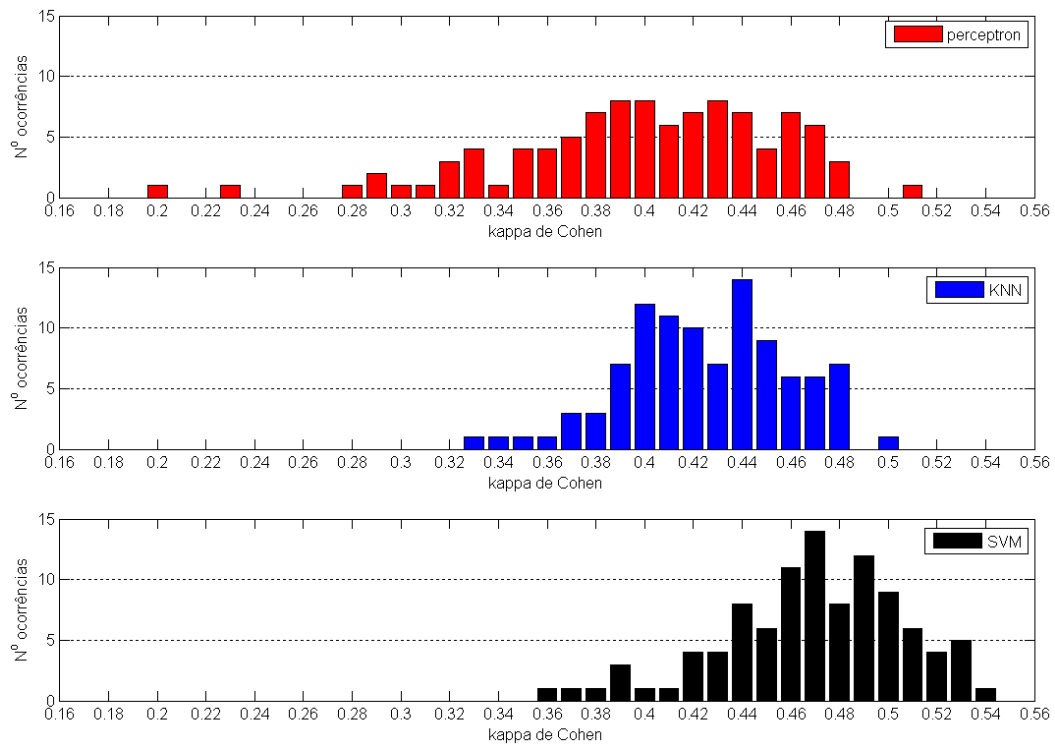


Figura 4.7: Histogramas do kappa de Cohen (para 100 ciclos de treinamento e teste) dos três classificadores testados.

As médias e os desvios padrão das taxas de acerto obtidas nos 100 ciclos de treinamento e teste para os três classificadores testados são apresentado na Figura 4.8. Os histogramas das taxas de acerto de cada classificador são apresentados na Figura 4.9.

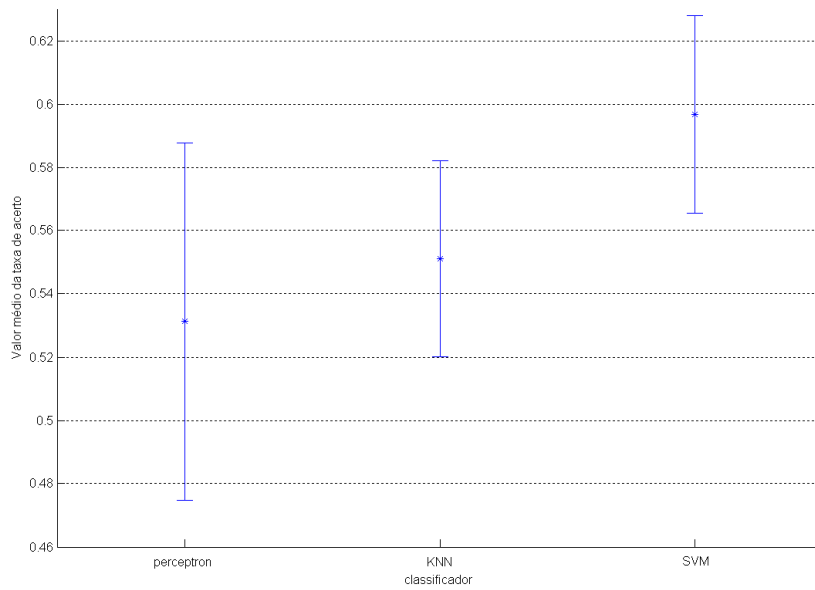


Figura 4.8: Valores médios das taxas de acerto com desvio padrão (para 100 ciclos de treinamento e teste) dos três classificadores testados.

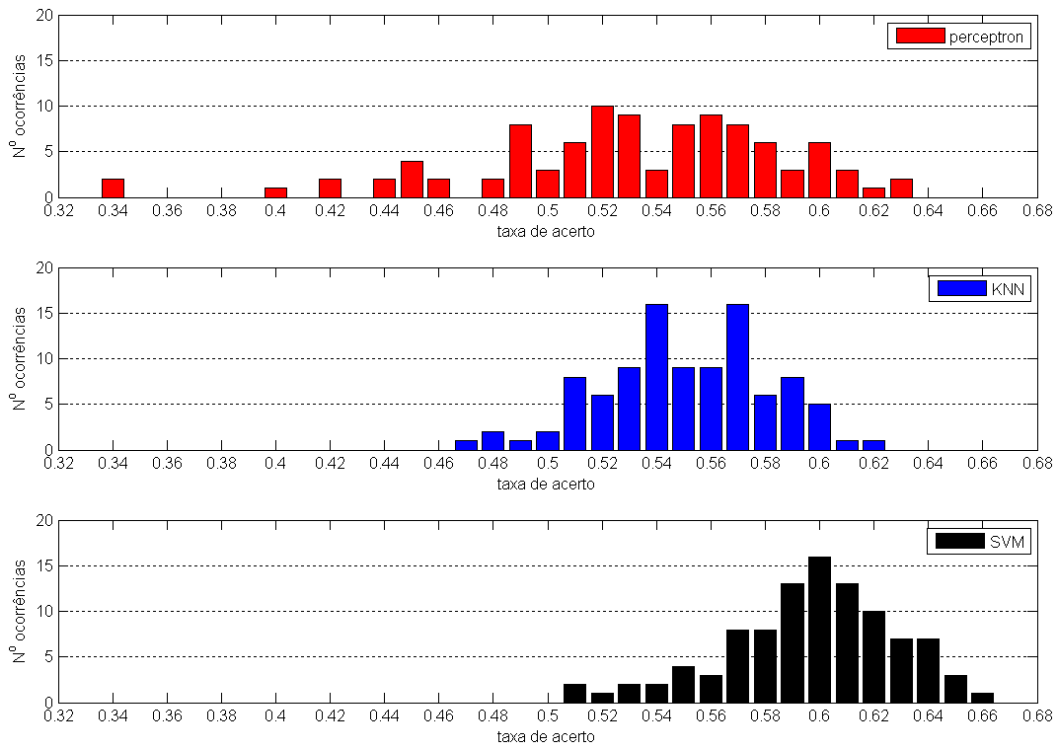


Figura 4.9: Histogramas das taxas de acerto (para 100 ciclos de treinamento e teste) dos três classificadores testados.

As Tabelas 4.6, 4.7 e 4.8 apresentam as matrizes de confusão para um ciclo de treinamento e teste com os classificadores *perceptron*, KNN e SVM, respectivamente. As Figuras

4.10 e 4.11 apresentam as curvas ROC geradas nos mesmos ciclos de treinamento e teste que geraram as Tabelas 4.6 (*perceptron*) e 4.8 (SVM).

Tabela 4.6: Matriz de confusão para um ciclo de treinamento e teste com a rede *perceptron*.

		clas. com <i>perceptron</i>								
		W	1	2	3	4	R	total	sensibilidade	especificidade
clas. pelo especialista	W	326	11	19	7	6	36	405	80,49%	89,93%
	1	33	1	47	1	1	60	143	0,70%	98,27%
	2	23	4	126	62	33	57	305	41,31%	89,66%
	3	1	0	5	25	21	0	52	48,08%	91,55%
	4	1	0	0	11	30	0	42	71,43%	93,70%
	R	3	0	2	0	0	59	64	92,19%	83,84%
	total	387	16	199	106	91	212	1011	taxa de acerto = 56,08%, $\kappa = 0,4241$	

Tabela 4.7: Matriz de confusão para um ciclo de treinamento e teste com KNN.

		clas. com KNN								
		W	1	2	3	4	R	total	sensibilidade	especificidade
clas. pelo especialista	W	337	47	3	0	1	17	405	83,21%	91,42%
	1	29	69	5	0	0	40	143	48,25%	85,36%
	2	21	73	89	54	29	39	305	29,18%	97,03%
	3	0	0	8	29	15	0	52	55,77%	93,53%
	4	1	0	2	8	31	0	42	73,81%	95,36%
	R	1	7	3	0	0	53	64	82,81%	89,86%
	total	389	196	110	91	76	149	1011	taxa de acerto = 60,14%, $\kappa = 0,4813$	

Tabela 4.8: Matriz de confusão para um ciclo de treino e teste com SVM.

		clas. com SVM								
		W	1	2	3	4	R	total	sensibilidade	especificidade
clas. pelo especialista	W	355	30	3	0	0	17	405	87,65%	88,28%
	1	33	58	10	1	0	41	143	40,56%	90,32%
	2	32	47	107	67	10	42	305	35,08%	96,88%
	3	1	0	8	34	9	0	52	65,38%	91,66%
	4	1	0	0	12	29	0	42	69,05%	98,04%
	R	4	7	1	0	0	52	64	81,25%	89,44%
	total	426	142	129	114	48	152	1011	taxa de acerto = 62,81%, $\kappa = 0,5078$	

Como um dos aspectos principais da abordagem por *wavelet packet* é a escolha dos nós mais discriminantes para a realização da classificação, realizou-se um teste com objetivo de analisar a diferença do desempenho de classificação utilizando os nós menos discriminantes

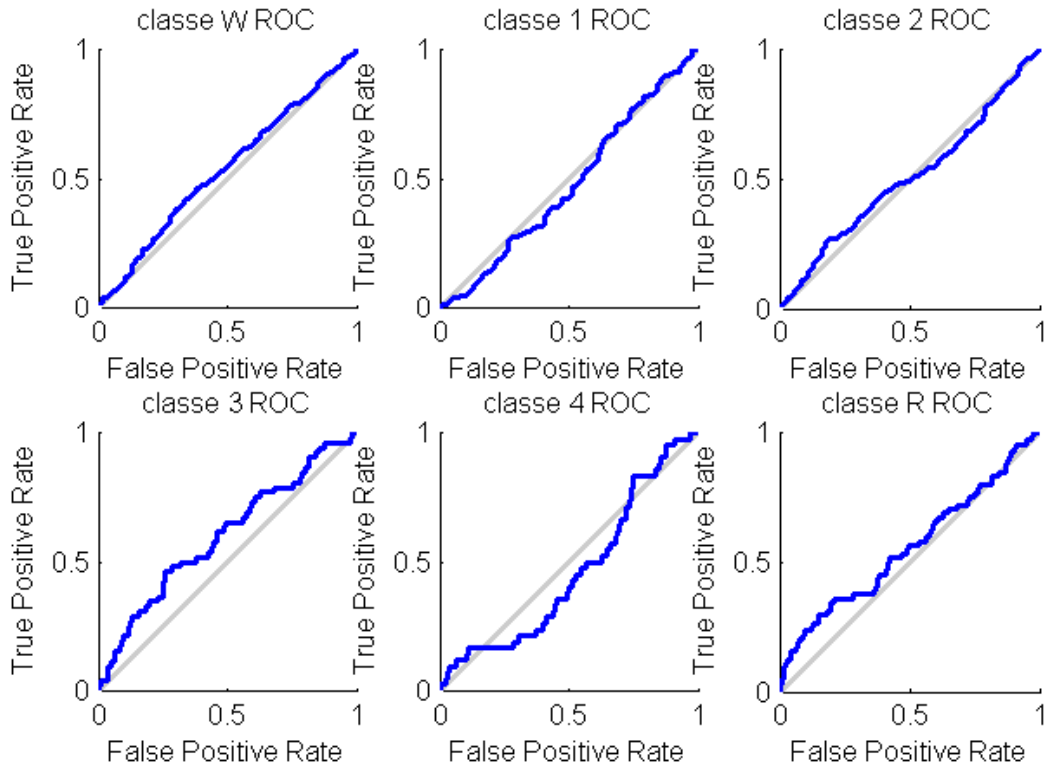


Figura 4.10: Curvas ROC de um ciclo de treino e teste com a rede *perceptron*.

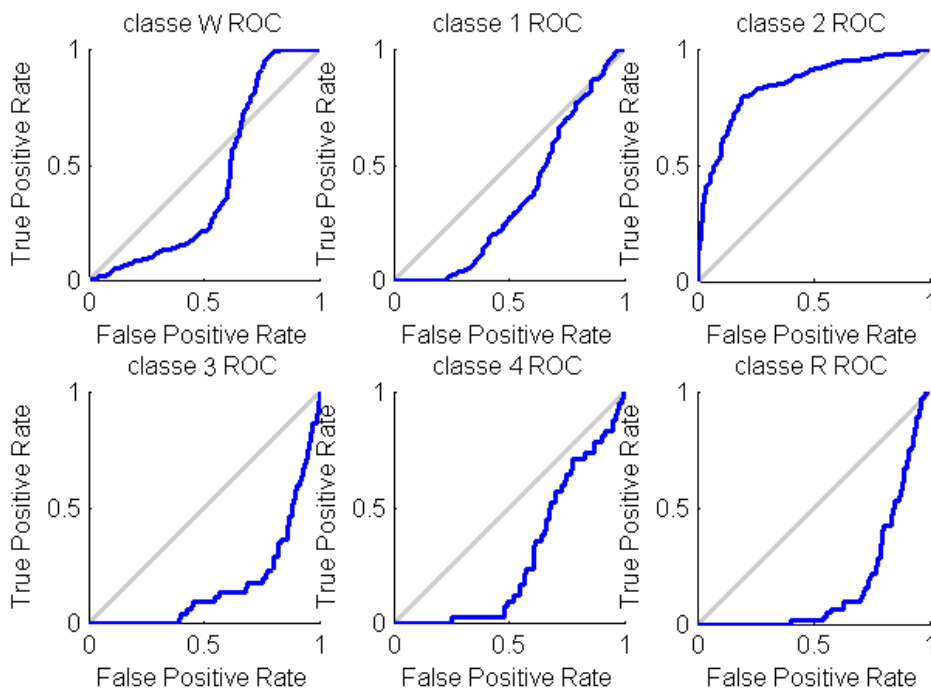


Figura 4.11: Curvas ROC de um ciclo de treino e teste com SVM.

da base escolhida e os nós mais discriminantes da mesma base. Nesse teste realizou-se 100 ciclos de treino e teste, utilizando o classificador SVM para realizar a classificação. A cada ciclo a classificação foi realizada tomando os 20 nós menos discriminantes e os 20 nós mais discriminantes. As médias da taxa de acerto com o desvio padrão são apresentados na Figura 4.12. Os histogramas das taxas de acerto de cada caso são apresentados na Figura 4.13.

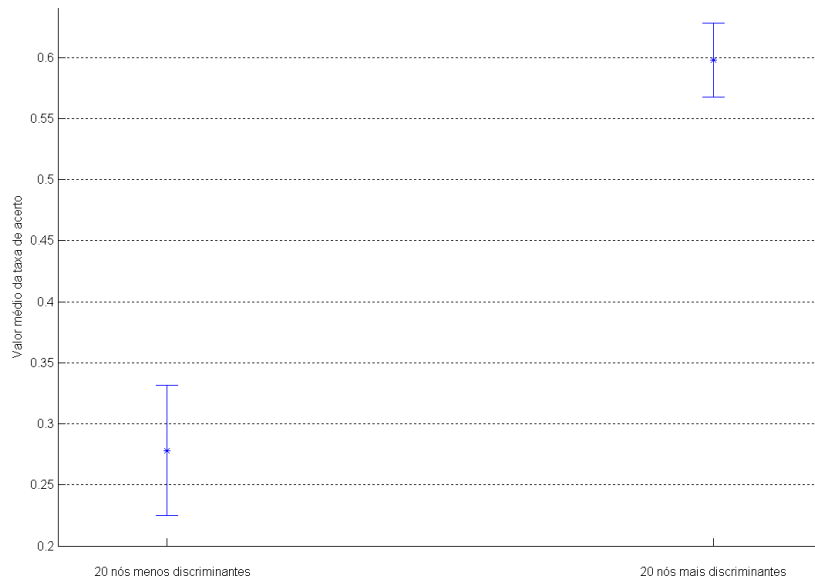


Figura 4.12: Valores médios da taxa de acerto com os desvios padrão (para 100 ciclos de treino e teste) com os 20 nós menos discriminantes e com os 20 nós mais discriminantes, utilizando SVM.

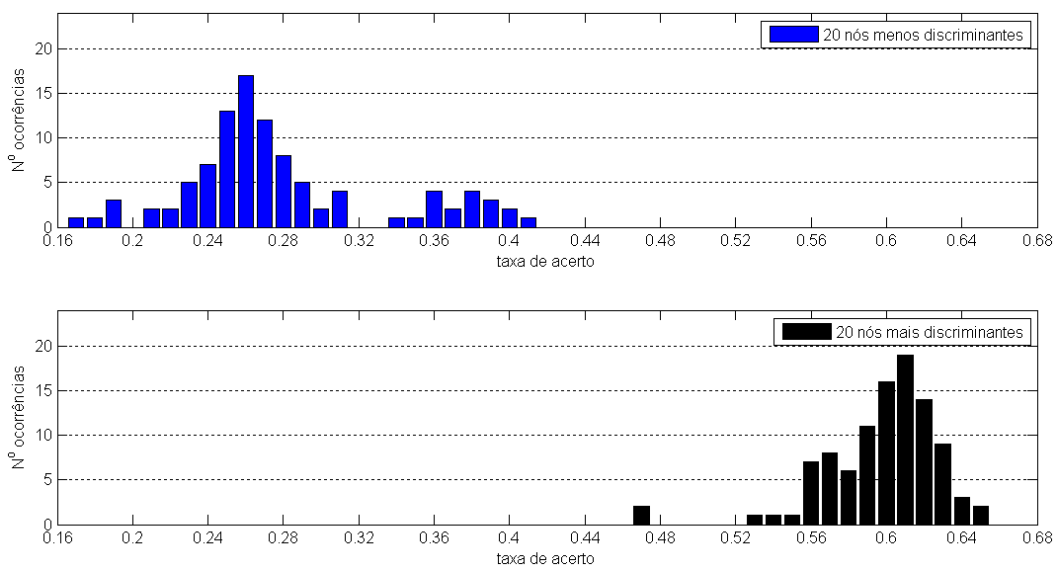


Figura 4.13: Histograma da taxa de acerto (para 100 ciclos de treino e teste) com os 20 nós menos discriminantes e com os 20 nós mais discriminantes, utilizando SVM.

4.3 Testes HMM

Para reduzir a taxa de amostragem os sinais originais foram filtrados por um filtro passa-baixas com frequência de corte de 40Hz e sub-amostrados por 3, o que reduz a taxa de amostragem para 83,33 amostras por segundo. Essa redução da taxa de amostragem foi necessária para reduzir o tempo de treinamento dos modelos HMM. Um ciclo de treinamento com os sinais já sub-amostrados ficou com duração em torno de uma hora. Portanto, os testes com os sinais originais ficariam muito dispendiosos de tempo.

A variância do ruído branco $\mathbf{w}(n)$ foi ajustado em $\sigma_w^2 = 0.0003$. E a variância do ruído branco $\mathbf{v}(n)$ foi ajustado em $\sigma_v^2 = 18.75$. Esses valores foram encontrados por meio de um algoritmo genético de forma a melhorar o desempenho do filtro de Kalman. Esses parâmetros afetam a qualidade da representação do sinal pelo modelo ARMA.

Após a subamostragem, os coeficientes ARMA dos sinais de EEG são calculados pelo filtro de Kalman. O modelo ARMA possui a mesma taxa de amostragem do sinal modelado. Portanto, cada época de 30 segundos de EEG do banco de dados gera um modelo ARMA próprio com mesma duração.

Representando os estágios do sono W, 1, 2, 3, 4 e R por $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$ e ω_6 , o treinamento do HMM pode ser representado pela Figura 4.14. Cada classe gera um modelo HMM, o qual é treinado usando o conjunto de treinamento de cada classe $O_{\text{treinamento}\omega_i}$, $i \in \{1, \dots, 6\}$, dado pela equação 3.79. Por meio de experimentos, o número de estados N do modelo HMM foi fixado em 4.

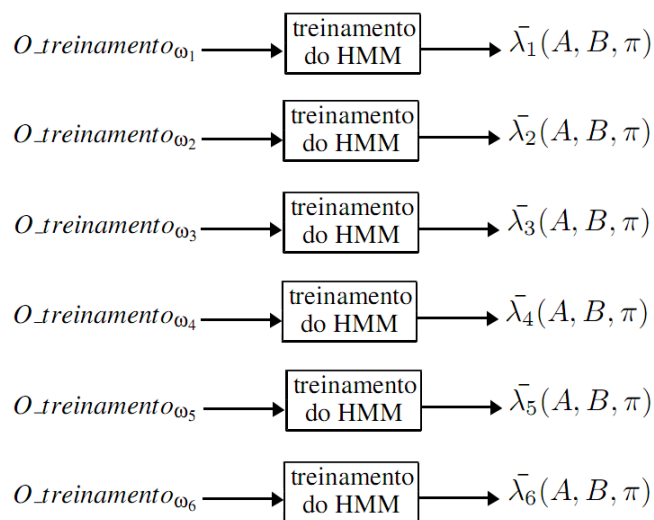


Figura 4.14: Treinamento dos modelos HMM. Um modelo $\bar{\lambda}(A, B, \pi)$ atribuído a cada classe.

A classificação de uma amostra de 30 segundos do EEG como um estágio do sono é

representada na Figura 4.15. A saída do classificador i^* é o índice da classe atribuída à amostra sendo classificada.

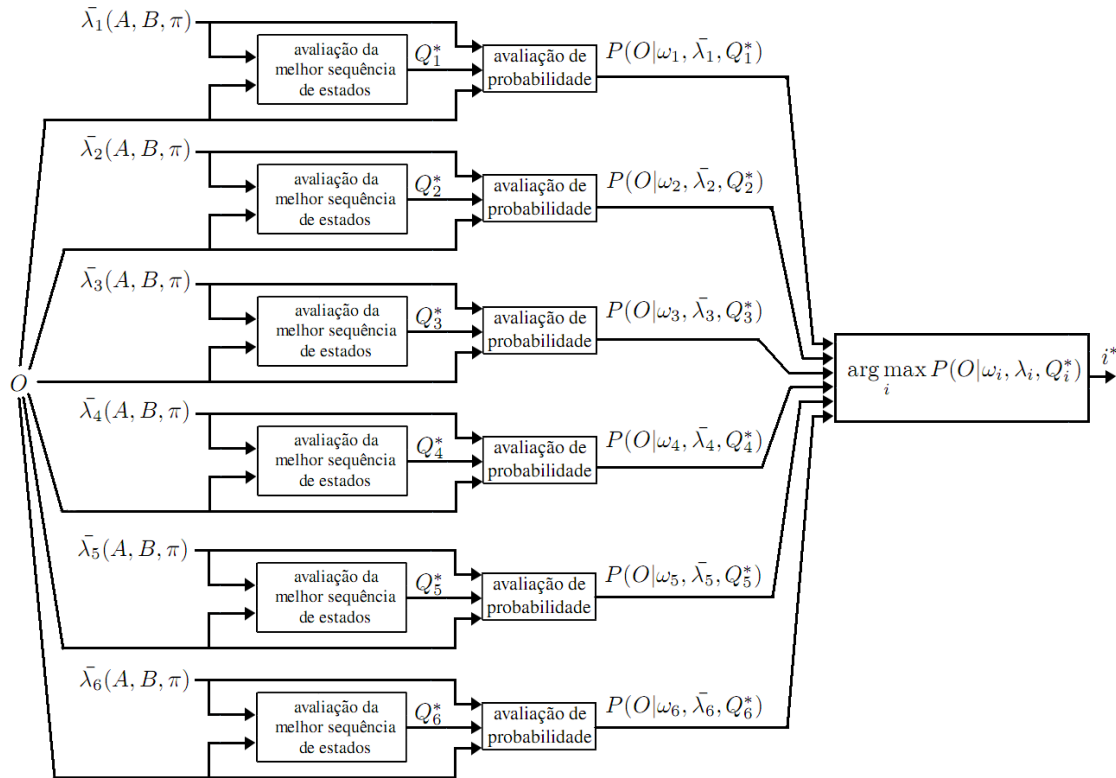


Figura 4.15: Classificação de um período do sono com HMM, $i \in \{1 \dots 6\}$.

4.3.1 Resultados HMM

Diferentes ordens para o modelo $\text{ARMA}(p, q)$ foram testados. Foi fixado $q = 1$ e selecionados os seguintes valores para p : 5, 10, 15 e 20. Para os testes, os conjuntos de treinamento e teste foram tomados como descrito na seção 4.1. Mantendo esses conjuntos fixos, os modelos HMM foram treinados para modelos ARMA diferentes, e foi feita a classificação com os dados do conjunto de treinamento e de teste. Esse processo foi repetido três vezes, os valores do kappa de Cohen obtidos são apresentados na Tabela 4.9, as taxas de acerto são apresentadas na Tabela 4.10.

As Tabelas 4.11 e 4.12 mostram as matrizes de confusão para a segunda execução da classificação, com $p = 15$, para os conjuntos de treino e de teste, respectivamente.

Tabela 4.9: kappa de Cohen da classificação com HMM.

Execução	Conjunto	kappa de Cohen (por ordem AR)			
		5	10	15	20
1 ^a	treino	0,5512	0,6195	0,6976	0,7951
	teste	0,4776	0,4454	0,4557	0,4394
2 ^a	treino	0,5024	0,5756	0,7220	0,7707
	teste	0,4141	0,4763	0,4816	0,4754
3 ^a	treino	0,5561	0,7171	0,7707	0,8098
	teste	0,4605	0,4732	0,4721	0,4848
média	treino	0,5366	0,6374	0,7301	0,7919
	teste	0,4507	0,4650	0,4698	0,4665

Tabela 4.10: Taxa de acerto da classificação com HMM.

Execução	Conjunto	Taxa de acerto (por ordem AR)			
		5	10	15	20
1 ^a	treino	62,60%	68,29%	74,80%	82,93%
	teste	59,64%	57,37%	58,56%	57,17%
2 ^a	treino	58,54%	64,63%	76,83%	80,89%
	teste	53,91%	60,04%	60,14%	59,74%
3 ^a	treino	63,01%	76,42%	80,89%	84,15%
	teste	58,65%	59,94%	59,84%	60,93%
média	treino	61,38%	69,78%	77,51%	82,66%
	teste	57,40%	59,12%	59,51%	59,28%

Tabela 4.11: Matriz de confusão para o modelo ARMA(15,1) na 2^a execução. Classificação do conjunto de treino.

		classificação pelo HMM						total	sensibilidade	especificidade
		W	1	2	3	4	R			
clas. pelo especialista	W	38	0	0	0	0	3	41	92,68%	96,10%
	1	4	32	4	0	0	1	41	78,05%	95,12%
	2	2	6	28	4	0	1	41	68,29%	95,61%
	3	1	2	2	23	13	0	41	56,10%	94,15%
	4	0	0	1	8	32	0	41	78,05%	93,65%
	R	1	2	2	0	0	36	41	87,80%	97,56%
	total	46	42	37	35	45	41	246	taxa de acerto = 76,83%, $\kappa = 0,7220$	

Tabela 4.12: Matriz de confusão para o modelo ARMA(15,1) na 2ª execução. Classificação do conjunto de teste.

		classificação pelo HMM						total	sensibilidade	especificidade
		W	1	2	3	4	R			
clas. pelo especialista	W	328	45	12	1	0	19	405	80,99%	95,38%
	1	15	78	16	4	0	30	143	54,55%	85,48%
	2	12	72	100	73	11	37	305	32,79%	92,63%
	3	0	1	11	23	16	1	52	44,23%	91,35%
	4	0	0	3	5	34	0	42	80,95%	97,21%
	R	1	8	10	0	0	45	64	70,31%	90,81%
	total	356	204	152	106	61	132	1011	taxa de acerto = 60,14%, $\kappa = 0,4816$	

4.4 Discussão dos resultados

Nos testes que utilizaram a transformada *wavelet packet* para a extração de características, o classificador SVM apresentou o melhor desempenho com uma taxa de acerto médio de 59,67% ($\kappa = 0,4685$), seguido pelo KNN com 55,11% ($\kappa = 0,4251$) e com o perceptron de múltiplas camadas apresentando o pior resultado com uma média de acerto de 53,13% ($\kappa = 0,3991$). Esses valores de taxa de acerto, juntamente com os desvios padrão, das taxas de acerto são apresentados na Figura 4.8. Os valores de kappa de Cohen obtidos, com os respectivos desvios padrão, são apresentados na Figura 4.6

O bom desempenho do classificador KNN comparado com os outros classificadores está de acordo com os resultados obtidos por Gudmundsson(Gudmundsson et al., 2005), onde o KNN obteve desempenho muito próximo ao SVM. Esse último trabalho citado também destaca que o KNN é conhecido por obter bom desempenho em muitos problemas práticos.

O melhor desempenho obtido com o SVM é explicado por suas características implícitas. A busca pela maximização da margem entre a fronteira de decisão e as amostras mais próximas a essa fronteira, a influência exclusiva dos vetores de suporte na classificação e o sistema de penalidades são características que fazem do SVM um classificador bastante robusto.

O perceptron de múltiplas camadas, apesar de possuir a capacidade de gerar superfícies discriminantes complexas, apresenta características de treinamento que dificultam muito a obtenção da fronteira de decisão ideal. O algoritmo de retropropagação do erro, com parâmetros bem configurados, é eficiente na busca do ponto de menor erro. Porém, esse algoritmo é dependente da condição inicial dos pesos. Uma má escolha dos pesos iniciais faz com que o algoritmo convirja para um mínimo local da função de erro. Aí se faz necessário

a utilização de heurísticas de treinamento. A mais simples é a execução do algoritmo de retropropagação do erro várias vezes, iniciando-se os pesos dos neurônios em cada uma delas. Esse procedimento, porém, faz com que o treinamento consuma muito tempo e não garante que uma boa solução seja encontrada.

Os histogramas com o kappa de Cohen e com a taxa de acerto para os 100 ciclos de treinamento e teste, apresentados nas Figuras 4.7 e 4.9, mostram o número de ocorrências de cada valor de κ e de taxa de acerto. Pelos histogramas se observa que os classificadores KNN e SVM apresentam menor variação de desempenho. Já o classificador perceptron de múltiplas camadas apresenta uma grande variação no valor de κ e da taxa de acerto. Observa-se que algumas vezes o perceptron possui desempenho semelhante ao do KNN e do SVM, o que indica que ele possui a capacidade de realizar uma boa classificação. Porém, muitas vezes, não se atinge uma boa solução, o que faz com que o perceptron tenha um desempenho inferior na média.

As Figuras 4.12 e 4.13 apresentam o desempenho de classificação do SVM utilizando os 20 nós menos discriminantes e os 20 nós mais discriminantes da base encontrada. Nesse teste confirmou-se a importância de se escolher os nós com maior capacidade discriminante, e a eficiência da distância de Kullback-Leibler como medida discriminante.

Os resultados obtidos com o HMM, apresentados na Tabela 4.10, mostram que a classificação com HMM, utilizando os coeficientes AR do sinal, apresenta desempenho semelhante ao do SVM. A média de acerto utilizando o modelo ARMA(15,1) foi de 59,51% (valor médio do $\kappa = 0,4698$).

As matrizes de confusão dos classificadores SVM e KNN apresentadas nas Tabelas 4.8 e 4.7 e da classificação com HMM apresentada na Tabela 4.12, que foram os classificadores que apresentaram melhor resultado, apresentam características semelhantes.

Em primeiro lugar se destaca o fato da maioria dos erros ocorrerem entre estágios vizinhos. O que está de acordo com o fato das características do sinal de EEG sofrerem alterações de forma gradual do estágio W até o estágio 4. É de se esperar que existam períodos de 30 segundos classificados que correspondam a uma transição entre dois estágios, o que justifica o erro entre estágios vizinhos.

Outra característica marcante é o número de erros envolvendo o estágio REM e as classes 1 e 2. Isso se deve ao fato das características do EEG entre esses três estágios apresentarem frequências parecidas. A diferenciação do estágio REM dos estágios 1 e 2 pelo especialista é auxiliada pelo EOG, que apresenta a movimentação rápida dos olhos, e pelo EMG, que apresenta uma ausência de força muscular durante os episódios de sono REM. Como esse trabalho utiliza apenas o sinal de EEG, essa diferenciação é dificultada.

A análise da matriz de confusão do classificador *perceptron* de múltiplas camadas apresenta uma característica gerada pela convergência do algoritmo de treino para uma solução que não é a melhor possível. Há uma forte tendência de classificar as amostras como pertencentes a uma determinada classe em detrimento de outras. Na matriz de confusão mostrada na Tabela 4.6, por exemplo, observa-se que quase todas as amostras pertencentes ao estágio 1 foram classificadas como pertencentes aos estágios W, 2 ou REM.

Por fim, a análise das curvas ROC de uma classificação com os classificadores *perceptron* e SVM apresentados nas Figuras 4.10 e 4.11 apresentam o indicativo de classificação para algumas classes pior que o de uma classificação aleatória (curva ROC abaixo da reta diagonal apresentada junto ao gráfico). Porém, analisando-se as matrizes de confusão apresentadas nas Tabelas 4.6 e 4.8 que correspondem às mesmas classificações apresentadas nas curvas ROC, observa-se que a taxa de acerto para a maiorias das classes está acima daquela que se obteria com uma classificação aleatória (16,67% para seis classes), a única exceção é a classe 1 com o classificador *perceptron*. Esse fato indica que as curvas ROC, sem adaptações, não são adequadas para a avaliação dos sistemas de classificação multiclasse.

Capítulo 5

Conclusões e Trabalhos Futuros

Esse trabalho propôs a classificação automática de estágios do sono pela análise de um canal do eletroencefalograma.

Utilizou-se a técnica de decomposição do sinal pela transformada *wavelet packet*, com a escolha dos nós mais discriminantes para a classificação pela distância de Kullback-Leibler e posteriormente a classificação dos sinais pela potência projetada nos nós mais discriminantes. Nessa técnica o aspecto principal é a utilização dos dados de treino não só para treinar os classificadores, mas também para informar quais são as características mais relevantes para a discriminação entre classes, essa é a função realizada pela escolha dos nós mais discriminantes da transformada *wavelet packet*. Nessa abordagem foram utilizados os classificadores SVM, KNN e *perceptron* de múltiplas camadas.

Utilizou-se também a técnica de modelagem do sinal por meio de um modelo ARMA, onde os coeficientes foram calculados por um filtro de Kalman e posteriormente os coeficientes AR dos sinais modelados foram utilizados para se realizar a classificação pela técnica de Modelo Oculto de Markov (Hidden Markov model, HMM).

Levando-se em consideração o grau de dificuldade do problema proposto, as taxas de acerto obtidas com a transformada *wavelet packet* mais o classificador SVM e as obtidas com o filtro de Kalman mais o HMM, que estiveram, em média, entre 59% e 60%, podem ser consideradas satisfatórias.

A dificuldade de classificação de estágios do sono é confirmada por Anderer (Anderer et al., 2007), que apresenta um estudo onde a taxa de concordância entre dois especialistas é de 76,9%. Esse valor indica que mesmo especialistas que utilizam outros sinais além do EEG para fazer a classificação podem divergir na sua escolha de rótulo.

A gravidade do problema de rotulagem pode ser vista pelos trabalhos de Berthomier (Berthomier et al., 2007) e (Anderer et al., 2007) que utilizam bancos de dados que passaram pela classificação de mais de um especialista, descartando as amostras que receberam rótulos diferentes ou atribuindo os rótulos por um especialista de consenso depois de duas classificações anteriores. A análise da taxa de acerto obtida nesse trabalho deve levar em conta que os dados do banco MIT-BIH, utilizados aqui, foram classificados por apenas um especialista e não houve descarte de amostras.

Como trabalho futuro pode-se incorporar os sinais de EOG e EMG na tentativa de diminuir o número de erros envolvendo o estágio REM. Também pode-se buscar a utilização de novos bancos de dados e a aplicação de alguma técnica de normalização dos sinais para que o sistema seja compatível com sinais provenientes de diferentes bancos de dados.

Referências Bibliográficas

- Anderer, P., Gruber, G., Parapatics, S., e Dorffner, G. (2007). Automatic sleep classification according to rechtschaffen and kales. *Proceedings of the 29th Annual International Conference of the IEEE EMBS*, páginas 3994–3997.
- Ben-David, A. (2007). A lot of randomness is hiding in accuracy. *Engineering Applications of Artificial Intelligence*, 20(7):875–885.
- Benbadis, S. R. (2006). Introduction to sleep electroencephalography. Em Lee-Chiong, T., editor, *Sleep: A Comprehensive Handbook*, páginas 989–1004. John Wiley & Sons.
- Benington, J. H. (2007). Fundamentals of electroencephalography and other biopotentials. Em Nic Butkov, T. L.-C., editor, *Fundamentals of sleep technology*, páginas 253–258. Lippincott Williams & Wilkins.
- Berthomier, C., Drouot, X., Herman-Stoïca, M., Berthomier, P., Prado, J., Bokar-Thire, D., Benoit, O., Mattout, J., e d’Ortho, M.-P. (2007). Automatic analysis of single-channel sleep eeg: Validation in healthy individuals. *Sleep*, 30(11):1587–1595.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Bittencourt, L. R. A., Santos-Silva, R., Taddei, J. A., Andersen, M. L., de Mello, M. T., e Tuik, S. (2009). Sleep complaints in the adult brazilian population: A national survey based on screening questions. *Journal of Clinical Sleep Medicine*, 5(5):459–463.
- Bittencourt, L. R. A., Silva, R. S., Santos, R. F., Pires, M. L. N., e de Mello, M. T. (2005). Sonolência excessiva. *Revista Brasileira de Psiquiatria*, 27(1):16–21.
- Carskadon, M. A. e Dement, W. C. (2005). Normal human sleep: an overview. Em Kryger, M. H., Dement, W. C., e Roth, T., editors, *Principles and practice of sleep medicine*, páginas 13–23. Saunders, 4 edição.
- Carskadon, M. A. e Rechtschaffen, A. (2000). Monitoring and staging human sleep. Em Kryger, M. H., Dement, W. C., e Roth, T., editors, *Principles and practice of sleep medicine*, páginas 1197–1215. Saunders, 3 edição.

- Chang, C.-C. e Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- CNT (2006). *Atlas do Transporte*. Confederação Nacional do Transporte, CNT.
- da Silva Braga, A. C. (2000). *Curvas ROC: aspectos funcionais e aplicações*. doctoralthesis, Universidade do Minho.
- de Azevedo Abreu, G., de Oliveira, L. C. L., da Rocha Nogueira, A., e Bloch, K. V. (2009). Quadro clínico: reconhecimento do paciente com apneia obstrutiva do sono. *Revista Brasileira de Hipertensão*, 16(3):164–168.
- Diniz, P. S. R., da Silva, E. A. B., e Netto, S. L. (2004). *Processamento Digital de Sinais: Projeto e Análise de Sistemas*. Bookman.
- DPRF (2010). *Departamento de Polícia Rodoviária Federal, Estatísticas*. <http://www.dprf.gov.br/PortalInternet/estatistica.faces> acessado em 23 de maio de 2010.
- DPVAT (2010). *Seguro DPVAT, Informações*. <http://www.dpvatseguro.com.br/conheca/informacoes.asp> acessado em 22 de maio de 2010.
- Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874.
- Fernández, A., García, S., Luengo, J., Bernadó-Mansilla, E., e Herrera, F. (2010). Genetics-based machine learning for rule induction: Taxonomy, experimental study and state of the art. *IEEE Transactions on Evolutionary Computation*.
- Fraiwan, L. A., Khaswaneh, N. Y., e Lweesy, K. Y. (2009). Automatic sleep stage scoring with wavelet packets based on single eeg recording. *World Academy of Science, Engineering and Technology*, 54:485–488.
- Gargour, C., Gabrea, M., Ramachandran, V., e Lina, J.-M. (2009). A short introduction to wavelets and their applications. *Circuits and Systems Magazine*, 9(2):57–68.
- Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., e Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220. *Circulation Electronic Pages*: <http://circ.ahajournals.org/cgi/content/full/101/23/e215>.
- Gudmundsson, S., Runarsson, T. P., e Sigurdsson, S. (2005). Automatic sleep staging using support vector machines with posterior probability estimates. *CIMCA-IAWTIC'05*.

- Guiot, M. (1996). Polissonografia. Em Reimão, R., editor, *Sono : Estudo Abrangente*. Atheneu, 2 edição.
- Hand, D. J. e Till, R. J. (2001). A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine Learning*, 45(2):171–186.
- Hayes, M. H. (1996). *Statistical Digital Signal Processing and Modeling*. Wiley.
- Haykin, S. (2001). *Redes Neurais: Princípios e prática*. Bookman, 2 edição.
- Juang, B.-H. e Rabiner, L. R. (1990). The segmental k-means algorithm for estimating parameters of hidden markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(9):1639–1641.
- Kullback, S. e Leibler, R. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.
- Mallat, S. (1998). *A wavelet tour of signal processing*. Academic Press.
- Malmivuo, J. A. e Plonsey, R. (1995). *Bioelectromagnetism, Principles and Applications of Bioelectric and Biomagnetic Fields*. Oxford University Press.
- McCartt, A. T., Rohrbaugh, J. W., Hammer, M. C., e Fuller, S. Z. (2000). Factors associated with falling asleep at the wheel among long-distance truck drivers. *Accident Analysis and Prevention*, (32):493–504.
- Mello, M., Santana, M., Souza, L., Oliveira, P., Ventura, M., Stampi, C., e Tufik, S. (2000). Sleep patterns and sleep-related complaints of brazilian interstate bus drivers. *Brazilian Journal of Medical and Biological Research*, (33):71–77.
- Mercy, C. (2010). *What is a Kappa coefficient? (Cohen's Kappa)*. Children's Mercy Hospitals and Clinics <http://www.childrens-mercy.org/stats/definitions/kappa.htm> acessado em 14 de outubro de 2010.
- Parisi, P., Ferri, R., Pagani, J., Cecili, M., Montemitro, E., e Villa, M. P. (2005). Ictal video-polysomnography and eeg spectral analysis in a child with severe panayiotopoulos syndrome. *Epileptic Disord*, 7(4):333–339.
- Pfurtscheller, G. e da Silva, F. L. (1999). Event-related eeg/meg synchronization and desynchronization: Basic principles. *Clinical Neurophysiology*, 110:1842–1857.
- Pires, M., Teixeira, C., Esteves, A., Bittencourt, L., Silva, R., Santos, R., Tuik, S., e Mello, M. (2009). Sleep, ageing and night work. *Brazilian Journal of Medical and Biological Research*, (42):839–843.

- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.
- Saito, N. (1994). *Local Feature Extraction and Its Application Using Library Bases*. PhD dissertation, Faculty of the Graduate School of Yale University. This is a full PHDTHESIS entry.
- Siesta (2010). *The Siesta Group*. <http://www.thesiestagroup.com> acessado em 23 de outubro de 2010.
- Tarvainen, M. P., Hiltunen, J. K., Ranta-aho, P. O., e Karjalainen, P. A. (2004). Estimation of nonstationary eeg with kalman smoother approach: An application to event-related synchronization (ers). *IEEE Transactions on Biomedical Engineering*, 51(3):516–524.
- Zhovna, I. e Shallom, I. D. (2008). Automatic detection and classification of sleep stages by multichannel eeg signal modeling. *30th Annual International IEEE EMBS Conference*, páginas 2665–2668.