



DETECÇÃO DE NOVIDADE PARA SISTEMAS DE SONAR PASSIVO

Natanael Nunes de Moura Junior

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientador: José Manoel de Seixas

Rio de Janeiro
Março de 2018

DETECÇÃO DE NOVIDADE PARA SISTEMAS DE SONAR PASSIVO

Natanael Nunes de Moura Junior

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. José Manoel de Seixas, D.Sc.

Prof. Mariane Rembold Petraglia, Ph.D.

Prof. João Baptista de Oliveira e Souza Filho, D.Sc.

Prof. Carlos Eduardo Parente Ribeiro, D.Sc.

Prof. João Marcos Travassos Romano, Ph.D.

Prof. Antônio de Pádua Braga, Ph.D.

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2018

Moura Junior, Natanael Nunes de

Detecção de Novidade para Sistemas de Sonar Passivo/Natanael Nunes de Moura Junior. – Rio de Janeiro: UFRJ/COPPE, 2018.

XXIV, 117 p.: il.; 29,7cm.

Orientador: José Manoel de Seixas

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2018.

Referências Bibliográficas: p. 90 – 107.

1. Processamento de Sinais. 2. Sistemas de Sonar Passivo. 3. Inteligência Computacional. 4. Aprendizado de Máquina. I. Seixas, José Manoel de. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

*Ainda que eu falasse as línguas
dos homens e dos anjos, e não
tivesse amor, seria como o metal
que soa ou como o sino que tine.*

*E ainda que tivesse o dom de
profecia, e conhecesse todos os
mistérios e toda a ciência, e
ainda que tivesse toda a fé, de
maneira tal que transportasse os
montes, e não tivesse amor,
nada seria.*

1 Coríntios 13:1 e 2

Agradecimentos

Primeiramente, eu tenho que agradecer a Deus. Sem Deus, eu não seria nada, não teria razão de viver e muito menos a alegria necessária para chegar onde cheguei. Mesmo com a minha infidelidade, minha incredulidade e minha imperfeição, Deus olhou para mim e viu um valor inestimável, por isso, eu agradeço e tento honrar o Seu nome. Deus é meu refúgio e minha fortaleza, socorro presente nos dias maus e, acredite, a inspiração para cada um dos gráficos, códigos e ideias aqui apresentadas

Aos meus pais, Natanael e Neidimar, pela força nessa caminhada. Minha mãe que me mostrou o sentido real de amor e carinho, com a sua paciência que não pode ser estimada ou classificada. Ela me indicou o caminho, me abraçou com carinho e me deu seu colo quando era preciso. Meu pai que me mostrou que a perseverança que ele possui não é apenas em se manter de pé, mas também em me aconselhar e me motivar. Além deles, quero agradecer a minha irmã e meu cunhado que me deram as duas mais lindas princesas do mundo: Manuela e Mariana, que foram uma válvula de escape durante o período de desenvolvimento deste trabalho.

Além dos meus pais e da minha família, eu também tenho que agradecer a Gabriella Mattar, que foi meu porto seguro, minha amiga e minha companheira durante o processo de desenvolvimento da minha tese e, espero eu, que seja pelo resto da vida. Sempre que precisei de tempo para fazer o meu trabalho, ela me respeitou e me ajudou a ter motivação para seguir em frente. Por isso, muito obrigado Gaby.

Ao meu orientador, José Manoel de Seixas, eu devo agradecer pelos papos que me tiraram o estresse e as discussões que me abriram os meus olhos para análises que nunca teria imaginado. Posso agradecer a ele também a inspiração em ser um professor para ajudar ao próximo e, através disso, melhorar, crescer e buscar mais conhecimento, mesmo que, as vezes, eu não queira ou tente invariavelmente parar.

Aos meus amigos, eu preciso agradecer por tudo, desde as brincadeiras até os momentos em que eu desabafava. Eles foram meus suportes firmes nessa caminhada. Ao longo do período desta tese, os bravos soldados do DEL da turma 2005/02 de engenharia eletrônica e de computação da UFRJ, não se furtaram em me dar apoio e momentos felizes para continuar. Em especial aos amigos Daniel Dias, Gustavo Campos, Gustavo Moraes, Luiz Eduardo Balabram Filho e Rodrigo Pereira.

As meus colegas de laboratório, eu agradeço pelas ideias, discussões produtivas e pelos códigos compartilhados. Por muitas vezes, as dúvidas sobre conceitos importantes desta tese foram esclarecidas por amigos que dividiam a mesma bancada de trabalho que eu. Não poderia deixar de mencionar o Werner Freund, Rodrigo Coura Torres, Phillip Gaspar, Luiz Frias, Breno Arosa, Hellen Lima, Vinícius Mello, Gustavo Goltz e Fabrício Bozzi.

Aos meu alunos, eu gostaria de agradecer por me inspirarem e gostaria de pedir desculpas pelas minhas imperfeições como professor, mas deixo a promessa que melhorarei a cada dia. Cada aluno que tive me ajudou a entender que durante o processo de ensino, quem mais ganha é o professor. Eu tive a sorte de trabalhar durante o doutorado no departamento de engenharia eletrônica e de computação da UFRJ, local onde me formei e que tem alunos ótimos e, que eu tenho certeza que serão ótimos engenheiros.

Aos meus professores do Programa de Engenharia Elétrica da COPPE/UFRJ e do Departamento de Engenharia Eletrônica e de Computação da Escola Politécnica da UFRJ, eu agradeço as oportunidades recebidas e a força para nunca desistir e isso fez com que um garoto da roça se tornasse um doutor em engenharia. Especialmente, eu cito os professores Luiz Pereira Calôba, Jomar Gozzi, Luiz Wagner Pereira Biscainho, Paulo Sergio Ramirez Diniz, Sergio Lima Netto.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

DETECÇÃO DE NOVIDADE PARA SISTEMAS DE SONAR PASSIVO

Natanael Nunes de Moura Junior

Março/2018

Orientador: José Manoel de Seixas

Programa: Engenharia Elétrica

O som é uma onda mecânica que se propaga por grandes distâncias nos oceanos e, por essa razão, pode ser utilizado para a detecção e classificação de contatos em meios submarinos, tarefas básicas de um sistema sonar. O desenvolvimento de tais sistemas está diretamente ligado à defesa de um país com dimensões continentais, como o Brasil. Recentemente, a Marinha do Brasil definiu como prioridade estratégica a área de acústica submarina. Sistemas de sonar passivo podem ser instalados para monitorar a costa brasileira de maneira furtiva e eficiente. Ademais, estes são utilizados em submarinos militares para diferentes aplicações. Como neste ambiente de operação, cada navio possui uma assinatura acústica única, e navios cujos dados não foram adquiridos podem ser observados, faz-se necessário o desenvolvimento de um detector de novidade operando em conjunto com os classificadores de contatos implementados em sistemas da Marinha do Brasil. Como os classificadores operam competindo por recursos computacionais com os detectores de novidade, estes podem impactar na eficiência de classificação. A quantidade de classes, neste ambiente, é muito grande e, devido a isso, índices de desempenho específicos foram criados para avaliar a eficiência dos modelos desenvolvidos. Além disso, diferentes extratores de informação foram desenvolvidos para acessar informações relevantes dos navios em questão, dentre eles podem ser citados PCD, kPCA, NLPCA e SAE. O desenvolvimento deste modelo de detecção foi baseado no ambiente de operação da Marinha do Brasil e, como este pode ter suas condições operativas alteradas ao longo do tempo, um sistema de monitoramento da estacionaridade baseado em estatística de ordem superior foi proposto. Tanto o detector de novidade quanto o sistema de monitoramento de estacionaridade foram desenvolvidos com dados experimentais disponibilizados pela Marinha do Brasil.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

NOVELTY DETECTION FOR PASSIVE SONAR SYSTEMS

Natanael Nunes de Moura Junior

March/2018

Advisor: José Manoel de Seixas

Department: Electrical Engineering

Sound is a mechanical wave that propagates over great distances in the oceans and it can, therefore, be used for vessel detection and classification in underwater environments, which are basic sonar system tasks. The development of such systems is directly linked to the country defense, especially, in countries with continental dimensions, such as Brazil. Recently, the Brazilian Navy defined underwater acoustics as a strategic priority area. Passive sonar systems can be installed to monitor the Brazilian coast in a stealthy and efficient way. In addition, these are used in military submarines for different applications. As in this operating environment, each ship has a unique acoustic signature, and ships whose data have not been acquired can be observed, it is necessary to develop a novelty detector operating in conjunction with the contact classifiers implemented in Brazilian Navy systems. Because classification systems operate competing for computing resources with novelty detectors, they can impact in classification efficiency. The number of classes in this environment is very large, and because of this, specific performance indices were created to evaluate the developed model efficiency. In addition, different data compressors were developed to access relevant ship information of, among them can be cited PCD, kPCA, NLPCA and SAE. The novelty detection development was based on the operating environment of the Brazilian Navy and since it can have its operating conditions changed over time, a stationarity monitoring system based on higher order statistics was proposed. Both the novelty detector and the stationarity monitoring system were developed with experimental data provided by the Brazilian Navy.

Sumário

Lista de Figuras	xi
Lista de Tabelas	xx
Lista de Símbolos	xxi
Lista de Abreviaturas	xxii
1 Introdução	1
1.1 Motivação	4
1.2 Objetivo do Trabalho	6
1.3 Contribuições do Trabalho	7
1.4 Organização do Texto	7
2 Sistema Sonar	9
2.1 Equação Sonar	11
2.1.1 Equação Sonar para Sistemas de Sonar Ativo	12
2.1.2 Equação Sonar para Sistemas de Sonar Passivo	13
2.2 Subsistemas de um Sistema de Sonar Passivo	14
2.2.1 Conformador de Feixes ou <i>Beamforming</i>	15
2.3 Análises de Marcação no Tempo	16
2.4 Análises de Marcação em Frequência	17
2.5 Tracking	18
2.6 Classificadores	19
3 Detecção de Novidade	20
3.1 SVM para Detecção de Novidade: SVM de uma única classe	29
3.2 Redes neurais para a detecção de novidade	35
3.3 Análise de Estacionaridade	38
3.3.1 Testes de Raiz Unitária	40
3.3.2 Teste de Divergência de Kullback-Leibler	41

4	Método	43
4.1	Processamento de Sinais	44
4.2	Pré-processamento dos dados	46
4.3	Detectores de Novidade	49
4.3.1	One-Class SVM	49
4.3.2	Redes Neurais	50
4.3.3	Figuras de Mérito utilizadas	50
4.4	Análise de Estacionaridade	52
5	Resultados Experimentais	53
5.1	Dados Utilizados	53
5.1.1	Processamento de Sinais	54
5.2	Pré-processamentos	56
5.3	Detecção de Novidade	62
6	Conclusões e Trabalhos Futuros	86
6.1	Trabalhos Futuros	88
	Referências Bibliográficas	90
A	Produção Científica	108
B	Aprendizado de Máquina	111
B.1	Redes Neurais Artificiais	111
B.2	Support Vector Machines	114

Lista de Figuras

2.1	Bolhas sendo formadas pelo processo de cavitação, figura extraída de [40]	11
2.2	Sistema de Sonar Ativo	12
2.3	Sistema de Sonar Passivo	14
2.4	Diagrama em blocos de operação de um sistema de sonar passivo. . .	15
2.5	Gráfico de Marcação vs Tempo	16
2.6	Diagrama em blocos da Análise DEMON	17
2.7	Diagrama em blocos da Análise LOFAR	18
2.8	Gráfico de Frequência vs Tempo	18
3.1	Um exemplo de detecção de novidade baseada em domínio	24
3.2	Um exemplo de como o nível de exigência para um modelo treinado pode ser utilizado para detectar novidade	25
3.3	<i>One-Class SVM</i> : variação nos valores de ν	33
3.4	<i>One-Class SVM</i> : variação na largura de banda do kernel	34
3.5	Modelo baseado em rede neural MLP com um limiar associado a saída.	36
3.6	Modelo baseado em rede neural ART com raio de similaridade.	37
4.1	Método Proposto para Detecção de Novidade	43
4.2	LOFARgrama para diferentes quantidade de pontos em cada uma das janelas de processamento	44
4.3	LOFARgrama para diferentes índices de decimação	45
4.4	Janela utilizada para cálculo da média local do algoritmo TPSW	46
5.1	Diagrama em blocos da Análise LOFAR com parâmetros específicos	54
5.2	Eficiência de classificação do treinamento neural para classificação e percentual de rejeição ao modelo estacionário para diferentes tamanhos de janela de aquisição de dados	55

5.3	Distribuições dos sinais no domínio do tempo obtidos para janela definida para obtenção da menor probabilidade de perda de estacionaridade no sentido amplo. A linha em azul (esquerda superior) representa as distribuições obtidos para eventos que pertencem a Classe A. Já o <i>plot</i> em vermelho (direita superior) representa as distribuições para eventos pertencentes a Classe B. A linha em verde (esquerda inferior) apresenta as distribuições para eventos pertencentes a Classe C e, por fim, o gráfico em amarelo (direita inferior) representa as distribuições obtidas para eventos pertencentes a Classe D. As distribuições não foram normalizadas, sendo que os seus valores representam os sinais sonoros obtidos pelo hidrofone durante o período de aquisição de dados para cada uma das classes	57
5.4	Resultados obtidos com a divergência KL para janelas consecutivas de sinais, no sentido direto e no sentido reverso, quando a mesma é aplicada no domínio do tempo. Na parte esquerda superior, os valores da divergência KL para eventos que pertencem a Classe A são apresentados. Já a parte direita superior, representa os valores de divergência KL para eventos pertencentes a Classe B. A parte esquerda inferior apresenta os valores de divergência KL para eventos pertencentes a Classe C e, por fim, os gráficos apresentados na parte direita inferior representam os valores para a divergência KL obtidos para eventos pertencentes a Classe D. As distribuições foram normalizadas (área embaixo da curva igual a um) para ser a estimativa da PDF dos eventos.	58
5.5	Resultados para a energia acumulada e erro de reconstrução dos componentes principais. Cada gráfico representa o resultado para uma classe sendo tratada como novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, a quantidade de energia para a faixa de componentes que acumulam 90% da soma de energia total.	59

- 5.6 Resultados para a eficiência de classificação variando com o número de componentes principais por kernel extraídos, o kernel utilizado foio gaussiano e a largura de banda do kernel foi escolhida para minimizar o erro de representação dos dados. Cada gráfico representa o resultado para uma classe sendo tratada como novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, em cada um dos gráficos, a quantidade de componentes onde houve a estabilização do índice SP, ou seja, não há um aumento nos valores de SP quando há adição de outro componente. 60
- 5.7 Resultados para o erro de reconstrução obtidos com a extração de componentes principais não-lineares, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. O erro de reconstrução foi avaliado para a topologia de componentes principais não-lineares definida no apêndice B. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, em cada um dos gráficos, a quantidade de componentes onde houve a estabilização do índice SP, ou seja, não há um aumento nos valores de SP quando há adição de outro componente. 61
- 5.8 Variação dos dois índices de desempenho pela quantidade de componentes principais de discriminação extraídos por deflação. A cada componente, menos informação resta no conjunto de variáveis de entrada. Além da variação no número de componentes, cada uma das classes disponíveis foi tratada como classe novidade. 63
- 5.9 Variação dos dois índices de desempenho pela quantidade de componentes principais de discriminação extraídos cooperativamente. A cada componente, menos informação resta no conjunto de variáveis de entrada. Além da variação no número de componentes, cada uma das classes disponíveis foi tratada como classe novidade. 64

- 5.10 Resultado para detecção de novidade utilizando redes neurais MLP com um limiar de excitação na camada de saída e dados pré-processados com componentes principais de discriminação extraídos por cooperação entre componentes. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. 65
- 5.11 Resultado para detecção de novidade utilizando redes neurais MLP com um limiar de excitação na camada de saída e dados pré-processados com componentes principais de discriminação extraídos por deflação. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. . . . 66

5.12	Resultado para Detecção de Novidade baseada em modelos <i>one-class</i> SVM sem a aplicação de técnicas de pré-processamento. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (<i>plots</i> coloridos), a acurácia de classificação, o índice SP e o <i>trigger</i> , todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.	68
5.13	Resultado para Detecção de Novidade baseada em modelos <i>one-class</i> SVM alimentados com dados projetados nos componentes principais (baseados nos auto-vetores da matriz de autocorrelação). Os resultados expostos representam a eficiência de cada uma das classes conhecidas (<i>plots</i> coloridos), a acurácia de classificação, o índice SP e o <i>trigger</i> , todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.	69

- 5.14 Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais não-lineares (extraídos com base em uma rede neural auto-associativa). Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. 70
- 5.15 Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais não-lineares (extraídos com a projeção no espaço de *kernel*), ou kPCA. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. . 72

- 5.16 Resultado da Detecção de Novidade para PCD por cooperação como pré-processamento. Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais de discriminação extraídos cooperativamente. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. 73
- 5.17 Resultado da Detecção de Novidade para PCD por cooperação como pré-processamento. Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais de discriminação extraídos por deflação. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. 74

- 5.18 Comparação dos resultados obtidos com diferentes pré-processamentos para detecção de novidade e *trigger*. Comparação entre os resultados para diferentes pré-processamentos aplicados durante o desenvolvimento da tese. Os resultados expostos representam os valores de *trigger* (*plots* tracejados) e de detecção de novidade (*plots* com linhas sólidas) para cada um dos pré-processamentos, sendo que os valores de *trigger* foram obtidos para as classes conhecidas e os valores de detecção novidade foram obtidos para cada uma das classes disponíveis sendo tratadas como classe novidade, uma por vez. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. 75
- 5.19 Comparação dos resultados obtidos com diferentes pré-processamentos para detecção de novidade e índice SP. Comparação entre os resultados para diferentes pré-processamentos aplicados durante o desenvolvimento da tese. Os resultados expostos representam os valores do índice SP (*plots* tracejados) e de detecção de novidade (*plots* com linhas sólidas) para cada um dos pré-processamentos, sendo que os valores do índice SP foram obtidos para as classes conhecidas e os valores de detecção novidade foram obtidos para cada uma das classes disponíveis sendo tratadas como classe novidade, uma por vez. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados. 76

5.20	Resultados de classificação e reconstrução para uma rede neural auto-associativa com duas camadas, uma de codificação e outra de decodificação. Os resultados expostos representam a eficiência de classificação (índice SP, <i>plots</i> em azul) e a dissimilaridade de representação (divergência KL, <i>plots</i> em vermelho) para cada uma das classes conhecidas, variando-se a quantidade de neurônios na camada de representação. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.	78
5.21	Comparação dos resultados de detecção de novidade baseada em <i>stacked auto-encoders</i> com uma e duas camadas	81
5.22	Comparação dos resultados de detecção de novidade baseada em <i>stacked auto-encoders</i> com três e quatro camadas	82
5.23	Comparação dos resultados de detecção de novidade baseada em <i>stacked auto-encoders</i> com cinco e seis camadas	83
5.24	Comparação dos resultados de detecção de novidade baseada em <i>stacked auto-encoders</i> com sete e oito camadas	84
B.1	Rede Neural Artificial	112
B.2	Exemplo de um classificador linear de margem máxima	115
B.3	Exemplo de um classificador linear de margem máxima com uma variável de relaxamento	116

Lista de Tabelas

5.1	Referência para leitura das tabelas de resultados	80
5.2	Comparação entre os resultados obtidos com diferentes pré-processamentos, com a referência de leitura da Tabela 5.1	80
5.3	Comparação entre os resultados obtidos com diferentes quantidade de camadas de <i>Stacked AutoEncoders</i> , com a referência de leitura Tabela 5.1.	85
5.4	Comparação entre os resultados obtidos com <i>Deep Learning</i> e diferentes pré-processamentos, com a referência de leitura Tabela 5.1. . .	85

Lista de Símbolos

Eff	Eficiência para uma classe conhecida, p. 50
R	vector que representa o raio da hiper-esfera para o treinamento de SVM, p. 32
Ω	Grau da decimação, p. 18
α	taxa de aprendizado de uma rede neural, p. 111
γ	largura de banda do kernel gaussiano, p. 31
ν	parâmetro de controle da regularização na técnica de SVM, p. 30
ω	matriz que representa os pesos de uma rede neural, p. 111
ρ	parâmetro que determina a menor distância entre um ponto e superfície de decisão de SVM, normalizada pela normal do vector que define a norma, p. 30
ξ_i	i-ésima variável de relaxamento de SVM, p. 30
w	matriz que define a margem da SVM, p. 30
x_0	vector que representa o centro da hiper-esfera, p. 32

Lista de Abreviaturas

ADF	<i>augmented Dickey–Fuller test</i> ou teste de Dickey–Fuller aumentado, p. 41
AG	Ganho do Array ou <i>Array Gain</i> , p. 13
ART	Teoria da Ressonância Adaptativa ou <i>Adaptive Resonance Theory</i> , p. 17
AR	modelo auto-regressivo ou <i>Autoregressive model</i> , p. 40
BSS	Separação Cega de Fontes ou <i>Blind Source Separation</i> , p. 17
CNN	<i>Convolutional Neural Networks</i> ou Redes Neurais Convolucionais, p. 113
DBN	<i>Deep belief networks</i> , p. 113
DEMON	<i>DEtection MOdulation on Noise</i> , p. 3
DOA	Direção de Aproximação ou <i>Direction of Arrival</i> , p. 3
HOS	<i>High Order Statistics</i> ou Estatística de Ordem Superior, p. 6
ICA	Análise de Componentes Independentes ou <i>Independent Component Analysis</i> , p. 17
KDE	<i>kernel density estimation</i> ou estimação de densidade por kernel, p. 42
KL	divergência de Kullback-Leibler, p. 41
LOFAR	<i>LOw Frequency Analysis and Recording</i> , p. 3
LSTM	<i>Long Short-Term Memory</i> , p. 113
MB	<i>Marinha do Brasil</i> ou <i>Marinha Brasileira</i> , p. 1
MFC	<i>mel-frequency cepstrum</i> , p. 19

MG	Marinha de Guerra, p. 9
MLP	<i>Multi-Layer Perceptron</i> ou Perceptron multicamadas, p. 6
MM	Marinha Mercante, p. 9
NLPCA	<i>Non-linear Principal Component Analysis</i> ou Análise de Componentes Principais não-lineares, p. 6
NL	Nível de Ruído ou <i>Noise Level</i> , p. 13
PCA	<i>Principal Component Analysis</i> ou Análise de Componentes Principais, p. 6
PCD	<i>Principal Components of Discrimination</i> ou Análise de Componentes Principais de Discriminação, p. 6
PDF	<i>Função Densidade de Probabilidade</i> , p. 7
PP	<i>Phillips–Perron test</i> ou teste de Phillips–Perron, p. 41
PROSUB	Programa de Desenvolvimento de Submarinos, p. 2
PSO	<i>Particle Swarm Optimization</i> ou Otimização com Enxame de Partículas, p. 27
RNN	<i>Recurrent Neural Network</i> ou Redes Neurais Recursivas, p. 113
SAE	<i>Stacked Auto-Encoders</i> , p. 113
SISGAAz	Sistema de Gerenciamento da Amazônia Azul, p. 2
SL	Nível da Fonte ou <i>Source Level</i> , p. 12
SNR	Razão Sinal-Ruído ou <i>Signal to Noise Ratio</i> , p. 11
SONAR	<i>SOund Navigation and Ranging</i> , p. 3
STFT	<i>short-time fourier transform</i> ou transformada de Fourier de tempo curto, p. 54
SVM	<i>Support Vector Machines</i> ou Máquinas de Vector Suporte, p. 6
SV	Vector Suporte ou <i>Support Vector</i> , p. 30
StSS	<i>Strict Sense Stationarity</i> ou Estacionaridade no Sentido Estrito, p. 39

TL	Perda de Transmissão ou <i>Transmission Loss</i> , p. 12
TS	<i>Força do Alvo - Target Strength</i> , p. 13
WSS	<i>Wide Sense Stationarity</i> ou Estacionaridade no Sentido Amplo, p. 39
ZEE	Zona Econômica Exclusiva, p. 1
kPCA	<i>Kernel Principal Component Analysis</i> ou Análise de Componentes Principais por Kernel, p. 6

Capítulo 1

Introdução

O Brasil é o maior país da América do Sul, o terceiro maior país das Américas e o quinto maior país do mundo em área territorial, que contempla cerca de $8.515.767.049 \text{ km}^2$ de extensão [1, 2]. Como o país possui um vasto litoral, com cerca de 7.491 km , a sua zona econômica exclusiva (ZEE)¹ abrange cerca de $3.646.514 \text{ km}^2$, incluindo a plataforma continental, os arquipélagos de Fernando de Noronha, de São Pedro e São Paulo e de Trindade e Martim Vaz. Não obstante o tamanho da ZEE, o Brasil pleiteou, junto à Comissão das Nações Unidas sobre os Limites da Plataforma Continental, a expansão de sua plataforma em $900 \cdot 10^3 \text{ km}^2$ [1]. Como esta possui extensão territorial semelhante a da floresta amazônica, a ZEE é conhecida como Amazônia Azul.

Como se trata de uma ZEE, o Brasil pode explorar as diversas riquezas contidas na Amazônia Azul de maneira exclusiva, dentre as quais podem ser citadas: a pesca; a biodiversidade da fauna marítima, os minerais metálicos e outros recursos minerais do subsolo marinho, como o petróleo, encontrado na Bacia de Campos e no pré-sal (Bacia de Campos, Bacia de Santos e Bacia do Espírito Santo - a prospecção nestas áreas já corresponde a dois milhões de barris de petróleo por dia, ou seja, 90% da atual produção brasileira); aproveitamento de energia maremotriz e energia eólica em alto-mar ou *off-shore*.

A Marinha do Brasil (MB) tem por missão “preparar e empregar o poder naval, a fim de contribuir para a defesa da Pátria” e se propõe ser uma força que “estará permanentemente pronta para atuar no mar e em águas interiores, de forma singular ou conjunta, de modo a atender aos propósitos instituídos na sua missão”. Sendo assim, a proteção da Amazônia Azul é, atualmente, uma das prioridades da MB

¹De acordo com a Convenção das Nações Unidas sobre o Direito do Mar, os países costeiros têm direito a declarar uma ZEE de espaço marítimo para além das suas águas territoriais, na qual têm prerrogativas na utilização dos recursos, tanto vivos como não-vivos, e responsabilidade na sua gestão ambiental. A ZEE é delimitada, em princípio, por uma linha situada a 200 milhas marítimas da costa, mas pode ter uma extensão maior, de acordo com a da plataforma continental. A ZEE separa as águas nacionais das águas internacionais [3].

[4, 5].

Para isso, a MB conta com diversos equipamentos de patrulhamento e segurança [6], dentre eles: um porta-aviões, fragatas Classe Niterói modernizadas recentemente, fragatas Classe Greenhalgh, corvetas, navios-tanque, navios de desembarque-doca, navios de desembarque de carros de combate, navio de transporte de tropas, submarinos, navio-escola, navio-veleiro e navio de socorro submarino. A esquadra brasileira é dividida em 5 sub-categorias: Equipamentos de Esquadra, Equipamentos Distritais, Equipamentos de Pesquisa, Equipamentos de Instrução e Equipamentos Avisos. Os navios que compõem a sub-categoria esquadra, por exemplo, somam 35 equipamentos, sendo que destes, 12 possuem integralmente tecnologia nacional ($\approx 34\%$), 13 ($\approx 38\%$) são fruto de colaboração com outros país (transferência de tecnologia), e 10 têm tecnologia totalmente estrangeira ($\approx 28\%$). Posto isso, os equipamentos de esquadra brasileiros possuem pouca tecnologia nacional, o que reduz a soberania nacional da armada.

Para aumentar o percentual de tecnologia nacional em sua esquadra e visando a proteção da costa brasileira e do pré-sal, a MB lançou alguns projetos, dentre os quais podem ser citados o sistema de Gerenciamento da Amazônia Azul (SISGAAz)[7] e o Programa de Desenvolvimento de Submarinos (PROSUB) [8]. O primeiro visa desenvolver um sistema de monitoramento, vigilância e gerenciamento da Amazônia Azul, enquanto o segundo produzirá o primeiro submarino de propulsão nuclear do Brasil e outros quatro submarinos convencionais, através de transferência de tecnologia com a França. Como o submarino militar utiliza sistemas de sonar para navegação, localização de ameaças e outras tarefas específicas, este equipamento depende muito da qualidade de seus sistemas de sonar. Assim sendo, esta tecnologia é de fundamental importância para a MB, o que pode ser observado com a definição da área de acústica submarina como prioritária.

Uma das tarefas mais fundamentais de qualquer esquadra é a detecção e classificação de navios, quer sejam navios inimigos, quer sejam navios de sua própria esquadra ou aliados [9]. Para a detecção e classificação de navios de superfície, métodos baseados em ondas eletromagnéticas de alta-frequência, como as utilizadas em sistemas de radar, ou ainda, métodos baseados na detecção do calor irradiado pelo navio, com ondas de infravermelho, podem ser aplicados[10]. Estes métodos, embora eficazes, possuem as suas limitações no mar, tais como a distância de detecção e a perda de furtividade. No caso de um navio que não opera na superfície, como um submarino, a utilização destas técnicas se torna ineficaz, pois as ondas eletromagnéticas de alta-frequência são atenuadas rapidamente no meio oceânico. Uma das soluções possíveis neste caso são as chamadas ondas eletromagnéticas de frequência extremamente baixa. Estas ondas são utilizadas para a comunicação de submarinos, quando os mesmos se encontram submersos [11]. Tais sinais eletro-

magnéticos se propagam por uma maior distância, embora necessitem de antenas muito grandes e operem a uma baixa taxa de transmissão.

O som é uma onda mecânica que se propaga eficientemente no mar e, assim sendo, pode ser utilizado para a detecção e classificação de sinais [12]. Para isso, são utilizados os sistemas de *Sound Navigation and Ranging*, ou simplesmente Sonar [13]. O sonar é o conjunto de técnicas que utiliza as propriedades do som se propagando na água do mar para a navegação, comunicação e detecção de outros navios. Existem dois tipos principais de sistemas de Sonar: o passivo e o ativo. O primeiro se beneficia da emissão acústica dos navios de interesse para a detecção e classificação dos navios, enquanto que o segundo emite um sinal e analisa seu eco para detectar possíveis alvos [9].

No caso dos submarinos, que são embarcações que primam pela furtividade, a utilização do sonar ativo deve ser feita com enorme parcimônia. Devido a isso, a utilização de sonares passivos é de extrema necessidade. Sinais processados por estes sistemas são conhecidamente complexos e, por muitas vezes, a tomada de decisão baseada nestes deve ser auxiliada por sistemas de processamento digital de sinais e de aprendizado de máquina, que operam em um volume expressivo de dimensões de entrada. Tendo em vista a complexidade e a sofisticação das técnicas utilizadas, o custo computacional para desenvolvimento e operação de tais sistemas pode ser elevado.

Um sistema de sonar passivo tem como principais objetivos [14]: a estimação da direção de aproximação ou DOA (*Direction of Arrival*), a detecção de contatos, sua identificação bem como seu acompanhamento. A estimação da DOA é feita utilizando algoritmos de conformação de feixes[15]. Uma vez que o feixe é formado, ou seja houve detecção de um sinal de interesse em uma determinada direção, o sinal pode ser submetido a uma extração de características visando a sua classificação. Atualmente, a Marinha do Brasil utiliza duas análises para a extração de características de tais sinais, a saber: a análise LOFAR (*LOw Frequency Analysis and Recording*) e a análise DEMON (*DEtection MOdulation on Noise*) [16].

As características extraídas por ambas as análises podem ser utilizadas para a classificação dos contatos que as originaram. Para tanto, os sistemas de sonar passivo, geralmente, apoiam-se em modelos de aprendizado de máquina para realizar a classificação. O aprendizado de máquina [17] é um sub-campo da inteligência artificial dedicado ao desenvolvimento de algoritmos e técnicas que permitam o treinamento de modelos de classificação, estimação, predição ou agrupamento de dados[18]. Como, no caso de sistemas de sonar passivo, cada um dos navios possui uma assinatura acústica única, detectores de novidade representam uma parte importante do sistema de classificação, uma vez que seria impraticável a aquisição de todas as possíveis classes de navios presentes em todas as esquadras militares das

marinhas ao redor do mundo. Ademais, novos navios podem ser desenvolvidos por outras marinhas e, como estes nunca entraram em contato com os sistemas de classificação da MB, tais sistemas devem ser desenvolvidos com a esta previsão. Assim, além do sistema de classificação já implementado pela MB, o desenvolvimento de um sistema de detecção de novidade que opere em conjunto com este é desejável.

Além de serem provenientes de um cenário de operação complexo, os sinais de sonar passivo, comumente, variam com o tempo devido a variações nas condições de mar e operativas[16, 19]. Devido a este fato, há modificações das características estatísticas do sinal ao longo das janelas de aquisição, o que leva à uma perda de estacionaridade [20]. A estacionaridade é uma propriedade fundamental para a análise e processamento de sinais. Segundo [20], o sentido estrito de estacionaridade implica na ausência de variação na estrutura geradora do sinal observado ao longo do tempo. Neste caso, os parâmetros extraídos de um trecho do sinal poderiam ser aplicados em outro trecho. Como o processo de classificação se baseia no treinamento de modelos, este pode ser afetado pela perda de estacionaridade. Assim sendo, um monitoramento deste característica deve ser feito a fim de se realizar a atualização dos parâmetros extraídos, à medida que se detecte a perda da mesma.

1.1 Motivação

Para garantir a soberania brasileira no mar, a MB investe na expansão da força naval e no desenvolvimento da indústria de defesa. Parte essencial desse investimento é o Programa de Desenvolvimento de Submarinos (PROSUB). Nascido com um acordo de transferência de tecnologia entre Brasil e França, em 2008, o programa viabilizará a produção de quatro submarinos convencionais e a fabricação do primeiro submarino brasileiro com propulsão nuclear.

O PROSUB visa o desenvolvimento de tecnologia nuclear brasileira para a propulsão de submarinhos – ponto destacado na Estratégia Nacional de Defesa [21]. A concretização do programa fortalece, ainda, setores industriais nacionais de importância estratégica para o desenvolvimento econômico do país. Priorizando a aquisição de componentes fabricados no Brasil para os submarinos, o PROSUB é um forte incentivo ao nosso parque industrial.

Além dos cinco submarinos, o PROSUB também estabelece a construção de um complexo de infraestrutura naval, que engloba o Estaleiro, a Base Naval (EBN) e a Unidade de Fabricação de Estruturas Metálicas (UFEM), em Itaguaí, RJ, realizada através de uma parceria público-privada [21].

O submarino militar é considerado o braço letal de uma esquadra. Valendo-se do seu silêncio e operando em águas profundas, tem a sua detecção dificultada; sendo, geralmente, utilizado para ataques específicos e a vigilância de grandes áreas. Por

esta razão, o submarino é empregado como uma das principais armas de dissuasão nos mares durante um conflito, e, assim, desperta grande interesse da MB.

O principal sistema de detecção de contatos de um submarino militar é o sonar passivo, que é composto, basicamente, por vários hidrofones combinados em um mesmo arranjo, chamado de *array*, e por um sistema de processamento de sinais, que atua sobre os sinais adquiridos. Cada hidrofone do *array* adquire sinais sonoros provenientes de diversas direções e as combinações desses sinais sonoros geram feixes direcionais de sinais. Estes feixes são chamados de marcações e são utilizados para a classificação dos contatos. Em um cenário tático de operação, vários contatos podem estar sobrepostos em uma única marcação. Sendo assim, faz-se necessária uma separação dos sinais da marcação que apresentem sobreposição.

Como todas as marinhas possuem esquadras e estas possuem diversos navios, cada um com uma assinatura acústica particular, a classificação de classes oriundas das leituras do sistema de sonar passivo deve ainda contar com um sistema de apoio à decisão. Este sistema deve conter classificadores robustos às possíveis variações dos sinais de entrada, uma vez que cada navio pode operar em diversas condições de máquina e, possuir detectores de novidade: como o sistema de apoio à decisão em questão pode se deparar com uma classe não considerada à época do seu desenvolvimento, seja pelo desenvolvimento de um novo navio por uma marinha inimiga ou por um navio que foi mantido em sigilo, este deve dispor de um detector de novidade.

Os dados provenientes de sistemas de sonar passivo basicamente são ruídos de interesse (emitidos por navios-alvos) corrompidos por outros ruídos de natureza diversa e, devido a isso, a utilização de métodos de pré-processamento pode ser de grande valia para acessar a informação neles contida de maneira mais eficiente. Atualmente, técnicas de pré-processamento de dados se mostraram mais eficientes em aplicações de aprendizados de máquina, quando comparadas com modelos treinados sem esta etapa[22, 23]. Técnicas de *deep learning* podem ser empregadas como estratégia de pré-processamento para sistemas de classificação complexos [24], motivando sua aplicação neste trabalho.

Como os modelos treinados durante o período de desenvolvimento da tese são extratores de características alimentados com dados de entrada que variam com o tempo, o sistema de apoio à decisão deve ainda contar com uma etapa de análise da perda de estacionaridade dos dados de entrada. Caso estes sinais sofram variações consideráveis, pode-se observar a degradação excessiva da estacionaridade e, assim, uma nova extração de características deve ser realizada.

1.2 Objetivo do Trabalho

Este trabalho foi desenvolvido em condições similares às de operação de classificação do sonar passivo da MB. Assim sendo, sinais experimentais disponibilizados pela MB são utilizados para o desenvolvimento de detectores de novidade e avaliação do impacto de tal detecção no processo de classificação.

Duas técnicas de detecção de novidade: máquinas de vector suporte (SVM [25]) e redes neurais artificiais (redes multilayer-perceptron - MLP [26] e redes neurais profundas, treinadas com métodos de *Deep Learning* [27]) foram testadas durante o desenvolvimento deste trabalho. A técnica SVM vem obtendo destaque na área de encapsulamento de dados por ser uma técnica que pode ser aplicada diretamente para tal fim. Redes neurais artificiais foram utilizadas anteriormente em pesquisas em sonar passivo, principalmente para classificação e, por isso, foram também aplicadas para detecção de novidade. Por fim, técnicas de *Deep Learning* chamaram a atenção da comunidade científica nos últimos anos e, com isso, houve uma motivação para a sua aplicação em detectores de novidade no ambiente de sonar passivo.

Como os dados utilizados possuem uma alta dimensionalidade, alguns métodos de pré-processamento são avaliados para a detecção de novidade. Os métodos de pré-processamento foram aplicados visando a representação dos dados de entrada em um espaço que favoreça a detecção de novidade e, como algumas técnicas de compactação foram aplicadas com sucesso em sinais de sonar passivo, a motivação para a sua aplicação está presente. A técnica SVM se baseia em uma transformação por kernel para acessar informações de estatística de ordem superior (HOS) e, assim sendo, a análise de componentes principais por kernel (kPCA), que utiliza uma transformação similar, foi utilizada como pré-processamento dos dados que alimentam detectores de novidade. Assim como a kPCA acessa informações em um espaço de dados transformados, a aplicação da análise de componentes principais não-lineares (NLPCA) se mostra uma outra alternativa natural, uma vez que esta realiza uma transformação não-linear dos dados e, com os dados transformados, extrai os componentes principais. Além dos métodos anteriormente citados, a análise de componentes principais linear (PCA) foi considerada para a compactação de dados. Outro método de pré-processamento utilizado foi a análise de componentes principais de discriminação (PCD), que visa obter as direções que maximizam o potencial discriminatório dos dados de entrada. Esta técnica foi aplicada anteriormente com sucesso na detecção de novidade com dados de sonar passivo.

Como dados de sonar passivo, usualmente, envolvem muitas classes, a avaliação de eficiência do modelo classe a classe se torna inviável. Assim sendo, índices de desempenho tradicionais, como a acurácia de eficiência, não podem ser aplicados.

Um outro objetivo que pode ser destacado nesta tese é a análise de estacionari-

dade. Como a perda de estacionaridade pode impactar diretamente na eficiência de classificação, um método de monitoramento da estacionaridade ao longo do processo de extração de características deve ser desenvolvido.

Um processo estacionário pode ser visto, em seu sentido estrito, como um processo estocástico cuja função densidade de probabilidade (PDF) conjunta não muda ao longo de uma variação de tempo [28, 29]. Assim sendo, todos os momentos da PDF, tais como média, variância, *kurtosis*, dentre outros, não apresentam variações ao longo do tempo. Como a obtenção da estimativa da PDF de um processo pode ser muito custosa computacionalmente, geralmente, a estacionaridade no sentido amplo é observada e esta é obtida pela média e função de auto-correlação do processo, ou seja, os momentos de ordem mais alta não são considerados.

1.3 Contribuições do Trabalho

Este trabalho visa contribuir com o desenvolvimento de técnicas que possam ser aplicadas a sinais provenientes de sistemas de sonar passivo pertencentes a MB. Além de contribuir com a soberania nacional, materializando o domínio desta tecnologia por parte das forças armadas brasileiras, esta tese tem como produto um detector de novidade que possa ser utilizado em conjunto com o sistema de classificação de contatos a ser implementado e embarcado nos submarinos da MB.

Posto isso, esta tese visa, de forma experimental, analisar: qual é o melhor algoritmo de compactação de dados no âmbito da detecção de novidade aplicada a sinais de sonar passivo (dentre os modelos expostos anteriormente); comparar entre diferentes técnicas de detecção de novidade propondo uma nova figura de mérito para auxiliar o projeto dos modelos de detectores de novidade; e por fim, avaliar o impacto da estacionaridade nos algoritmos de treinamento e avaliar a estacionaridade de sinais de sonar passivo, propondo-se um novo teste de estacionaridade.

1.4 Organização do Texto

O texto é organizado da seguinte maneira. No Capítulo 2, tem-se a introdução de um sistema de sonar e suas características. Além disso, ainda neste capítulo, será realizada uma breve revisão bibliográfica relacionada ao desenvolvimento de sistemas de sonar em linhas gerais. As técnicas consagradas de aprendizado de máquina relacionadas a detecção de novidade são apresentados no Capítulo 3. No Capítulo 4, o método aplicado para a obtenção dos resultados experimentais será detalhadamente apresentado. Já o Capítulo 5 trará a apresentação dos resultados experimentais obtidos durante o período de desenvolvimento e, por fim, o Capítulo 6 trará as conclusões e os trabalhos futuros e, este é seguido pelo capítulo de re-

ferências bibliográficas. Para encerrar o texto, dois apêndices, um para os trabalhos publicados durante o período de desenvolvimento da tese e outro descrevendo os detalhes das técnicas de aprendizado de máquina.

Capítulo 2

Sistema Sonar

Este capítulo tem por objetivo a exposição do contexto envolvido no trabalho. A Marinha do Brasil e suas estruturas serão brevemente descritas, bem como será realizada uma breve abordagem sobre sistemas de sonar, contemplando sonares ativos e passivos. Posteriormente, o processamento dos sinais adquiridos por sistemas de sonar passivo será descrito.

A Marinha é o conjunto das organizações e dos meios (pessoal, equipamentos, infraestruturas e outros recursos) dedicados às atividades marítimas, sobre tempo de guerra ou paz. A MB se subdivide em: Marinha de Guerra (MG) e Marinha Mercante (MM) - a última subdivida ainda em Marinha de Comércio, Marinha de Pesca, Marinha de Recreio.

A Marinha de Guerra é uma das três forças armadas brasileiras (compartilhando essa classificação com o Exército e a Aeronáutica). A MG é especializada na condução da guerra naval e anfíbia. Conforme o país, a MG pode ser designada, alternativamente, por termos como Armada, Marinha Militar ou Força Naval.

A organização, a tática e os meios empregados na guerra naval foram evoluindo ao longo do tempo, acompanhando a evolução militar, náutica e tecnológica. Durante a primeira guerra mundial, o poder marítimo atingiu sua máxima importância e se acredita que o poder marítimo inglês tenha decidido a guerra a favor dos aliados [30].

O objetivo estratégico ofensivo da MB é o da projeção de forças em áreas fora da costa do seu país, no sentido de controlar as linhas de navegação, transportar tropas ou atacar navios, portos e instalações costeiras inimigas. Já o objetivo estratégico defensivo da MB é dissuadir ou neutralizar a projeção de forças navais por parte de um inimigo. Para tanto, somente o fato de um país possuir uma frota de navios militares organizados em esquadras de maneira eficiente, pode evitar um ataque por forças inimigas ao seu território por vias aquáticas.

Uma das principais forças de uma esquadra é o submarino militar. Este equipamento é conhecido por sua efetividade em cenários de batalha devido à sua fur-

tividade e capacidade de camuflagem. O submarino militar, geralmente, opera em grandes profundidades um cenários reais de combate, o mesmo deve se orientar por instrumentos que possuam elevada acurácia. Dentre esses instrumentos, temos o sonar, que pode ser dividido em duas classes: ativo, usualmente empregado por navios de superfície anti-submarinos para a detecção de submarinos, e passivo, geralmente utilizado por submarinhos para a detecção de contatos sem a exposição da sua localização. Com os avanços tecnológicos na construção de navios de guerra, cada vez mais a emissão acústica dos mesmos tem sido reduzida.

O ruído gerado por um navio possui algumas fontes principais, a saber: ruído de máquinas, ruído de hélice, e ruído hidrodinâmico [31–33]. O ruído de máquinas é, majoritariamente, composto por ruídos gerados pelo sistema de propulsão do navio e por ruídos gerados por máquinas auxiliares do navio (como geradores, bombas, equipamentos de ar condicionados e outros). Os sistemas de propulsão atualmente utilizados são: propulsão à diesel [34], nuclear [35], hélice [36], à gás, solar [37] e biodiesel [38, 39].

O ruído do hélice, geralmente, pode ser descrito como sendo a soma do ruído de cavitação, gerado durante o movimento do hélice, e pela ressonância da vibração do casco do navio. O processo de cavitação se dá quando o hélice gira na água e, devido a isso, criam-se regiões de alta pressão (na parte anterior da pá do hélice) e regiões de baixa pressão (na parte posterior da pá do hélice). Se estas regiões sofrem uma quantidade de pressão suficiente para romper a resistência física da água, ocorre a formação de bolhas de ar (ou cavidades), como pode ser observado na Figura 2.1. Quando as bolhas formadas no processo são atingidas pelo fluxo de água ou pelo próprio hélice, as mesmas colapsam e emitem um pulso de som na água. O espectro do ruído gerado é, em média, dominado por altas frequências.

Além dessa característica espectral de altas frequências, o ruído de cavitação também é modulado em amplitude devido às chamadas batidas do hélice. À medida que o ruído de cavitação é produzido, são observados aumentos periódicos de amplitude ocorrendo na frequência de rotação do eixo propulsor, gerando, assim, ondas com picos espectrais próximos à frequência de rotação do eixo. A intensidade do ruído varia de acordo com o número de pás do hélice, de maneira que o ruído de cavitação é modulado. Esse processo de modulação auxilia na classificação dos sinais produzidos pelos navios de interesse que sejam adquiridos pelo sistema sonar, uma vez que cada navio tem, definido em projeto, um número específico de hélices, e estes um número específico de pás.

Já o ruído hidrodinâmico é gerado pela passagem do fluxo irregular de líquido pela embarcação, quando a mesma se encontra em movimento. Como a região contato com a água pode variar de acordo com o movimento das ondas e a porção da embarcação que está imersa, a intensidade de ruído irradiado varia de forma



Figura 2.1: Bolhas sendo formadas pelo processo de cavitação, figura extraída de [40]

irregular. Esse ruído pode ainda entrar em processo de ressonância com partes da embarcação, e isso pode excitar partes específicas da mesma, e gerar fenômenos que podem facilitar a identificação de alguns tipos de navio, como, por exemplo, o hélice cantante [31, 41]. Esse fenômeno se dá em função do processo de ressonância das pás do hélice excitadas pelo ruído hidrodinâmico e tem um som agudo semelhante ao de um canto.

2.1 Equação Sonar

A equação que modela a propagação dos ruídos no ambiente marinho é chamada de equação sonar [9]. A equação sonar é uma maneira sistemática de estimar as amplitudes de sinal e de interferências que podem ser observadas nos transdutores do sistema sonar e assim pode-se obter a razão sinal-ruído (SNR) de tal sistema. A SNR determina se um sistema sonar será capaz de detectar um sinal de interesse na presença de ruído de fundo, pois, caso o sinal de interesse não tenha amplitude suficiente para se diferenciar do ruído, a SNR apresentará um valor baixo e a detecção de tal sinal será ineficiente.

A equação sonar leva em consideração o nível da fonte emissora de sinal, o espalhamento do som, absorção sonora, as perdas de reflexão, o nível de ruído ambiente e, as características do receptor. Além destes parâmetros, a equação sonar pode ser desenvolvida para dois cenários: o de um sistema de sonar ativo ou o de um sistema passivo.

2.1.1 Equação Sonar para Sistemas de Sonar Ativo

Para sonares ativos, tais como localizadores de peixes, ecossondas, sonares de varredura lateral e sonares militares, há a transmissão de um pulso sonoro para a análise do eco recebido. O modelo de sonar ativo pode ser visto na Figura 2.2.

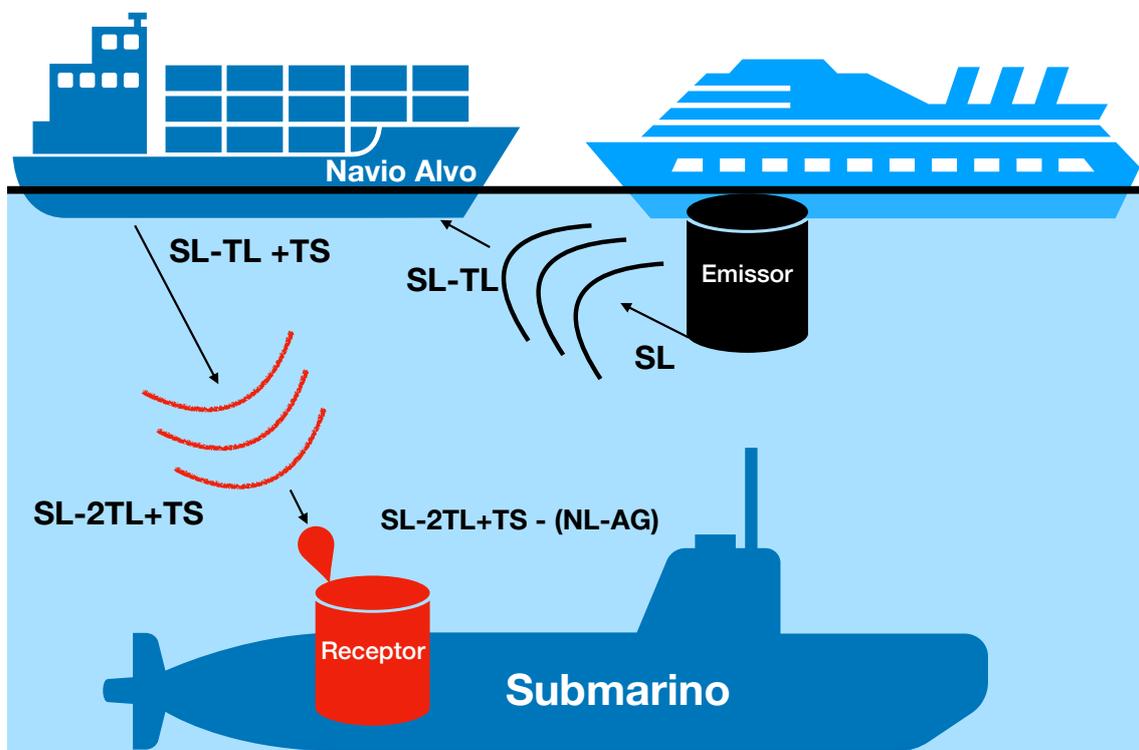


Figura 2.2: Sistema de Sonar Ativo

Nesta figura, uma onda é emitida por um sistema de emissão (cilindro preto), localizado em um navio, e quando atinge o navio alvo, a mesma é refletida e se propaga até ser adquirida por um receptor, posicionado em um submarino. Vale ressaltar que os sensores que captam o sinal possuem uma seleção de diretividade, representada na figura pela estrutura acoplada ao receptor.

A equação sonar para sonares ativos deve levar em consideração a potência da onda, ou soma de ondas, emitida pelo emissor (SL), a atenuação da emissão causada pelo pelo meio (TL), a potência de reflexão do pulso emitido, após o navio de

interesse (TS), o ruído de fundo (NL) e as características dos sensores de recepção (AG). O ruído de fundo, neste caso, é a soma de todas as fontes sonoras que não são de interesse. Os termos da equação de sonar são medidos em decibéis, pois representam ganhos, atenuações ou potências e, portanto adicionadas para formar a equação sonar para sistemas de sonares ativos (Equação 2.1).

$$SNR_{Ativo} = SL - 2TL + TS - (NL - AG) \quad (2.1)$$

O sonar ativo transmite um sinal com uma potência de fonte SL . O som é atenuado à medida que se propaga em direção ao alvo, devido à perda de transmissão TL . A intensidade do som para o alvo é dada por $SL - TL$. Apenas esta parte do som que atinge o alvo pode ser refletida de volta para a plataforma do sonar ativo. A intensidade do eco a um metro do alvo é dada por TS . Ou seja, a intensidade do eco retornado é dada por $(SL - TL) + TS - TL$. Se o nível de ruído no receptor é NL , a relação sinal-ruído no receptor pode ser calculada como $SL - 2TL + TS - NL$.

Geralmente, os receptores, como os utilizados em sistemas sonar, são constituídos por diversos transdutores mais simples que captam sinais em todas as direções. Para o processamento desses sinais, geralmente, a soma dos sinais provenientes de todos os transdutores é feita levando-se em consideração a diferença de fase observada na frente de onda plana adquirida. Para tanto, variam-se todas as possíveis DOA e observa-se em qual obtém-se o maior valor de energia com diferença de fase em questão. Assim sendo, pode-se fazer a análise dos sinais recebidos em uma direção específica. Todo este processo pode ser representado como a diretividade do arranjo de sensores.

2.1.2 Equação Sonar para Sistemas de Sonar Passivo

Sistemas de sonar passivo adquirem sons gerados por diversas fontes sonoras e não emitem sons usando a análise de ecos refletidos de um determinado contato. Assim sendo, a equação sonar deve levar em consideração a amplitude do ruído da fonte (SL), o espalhamento do som emitido pela fonte e consequente atenuação do som ao se propagar no meio (TL), o ruído de fundo no receptor (NL) e as características do receptor (AG). A equação sonar, neste caso, não levará em consideração a intensidade do eco (TS), uma vez que não temos a transmissão de um pulso, como no sonar ativo.

Na Figura 2.3 temos uma representação da operação de um sistema de sonar passivo. Como se pode notar, neste caso, o sistema de emissão sonora não existe, ou seja, não há exposição da posição do submarino e, conseqüentemente, a detecção e processamento do sinal emitido se tornam mais desafiadores. Aqui, a emissão do sinal de interesse é dada pela propulsão/movimento do navio alvo e, todos os outros

sinais neste cenário são considerados ruído de fundo.

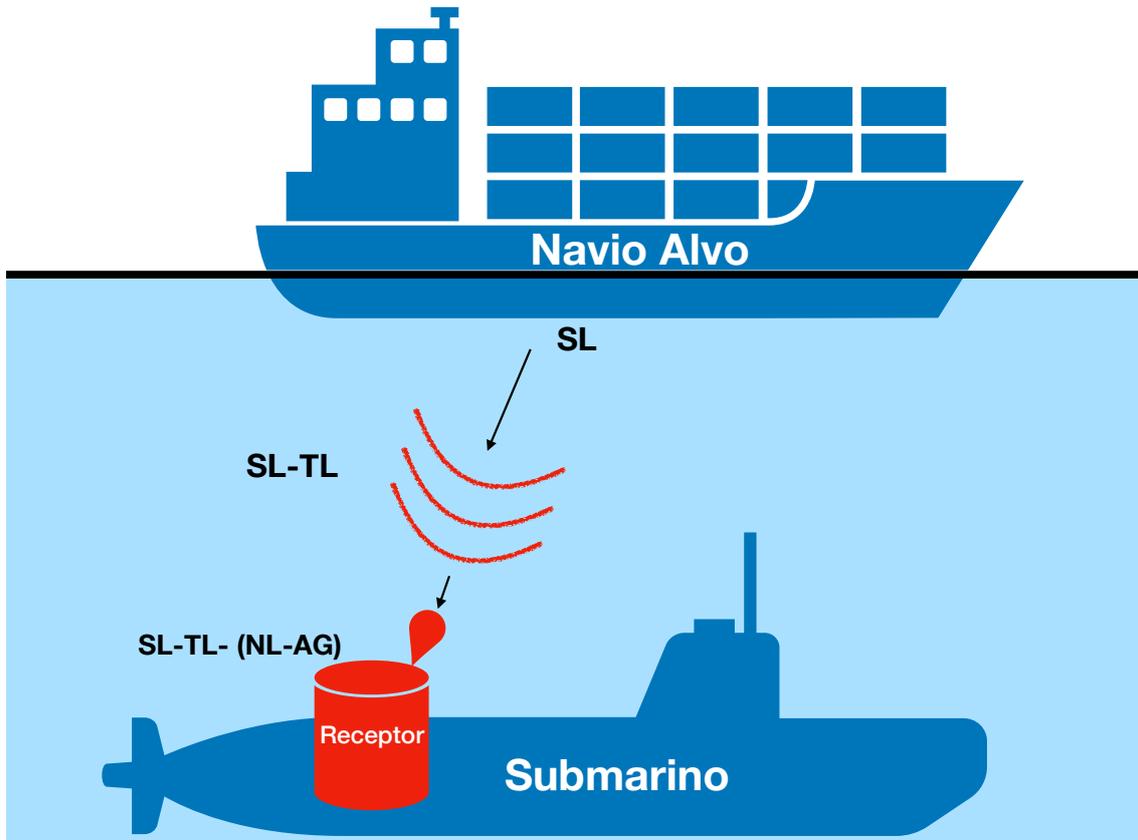


Figura 2.3: Sistema de Sonar Passivo

A fonte sonora, quer se trate de uma baleia, vulcão, submarino ou qualquer outro tipo de plataforma irradiadora de ruído sonoro, emite um sinal com um nível de fonte SL . A intensidade do som é reduzida pela perda de transmissão TL , uma vez que o som viaja a partir da fonte para o receptor. A intensidade do sinal no receptor é dada por $SL - TL$. A relação sinal ruído é dada por $SL - TL - NL$, no caso de um receptor composto por um único transdutor. Como para sistemas de sonar passivo também podemos ter receptores compostos por diversos transdutores ou sensores, temos que a relação sinal ruído, levando em consideração as características espaciais do receptor (AG), é dada pela Equação 2.2

$$SNR_{Passivo} = SL - TL - (NL - AG) \quad (2.2)$$

2.2 Subistemas de um Sistema de Sonar Passivo

Como mostrado na Equação 2.2, a razão sinal-ruído de um sistema de sonar passivo tende a ser menor do que a de um sistema de sonar ativo (Equação 2.1), pois temos a eliminação da emissão do pulso de referência, que é conhecido. Devido a isso,

o processo de detecção de um contato se torna mais desafiador no primeiro. Para tornar o processo de detecção mais eficiente, diversos subsistemas são acoplados ao sistema de sonar passivo, cada um com características e funcionalidades específicas.

O sistema de sonar passivo processa sinais seguindo o diagrama em blocos da Figura 2.4. Os n sinais, representados por s_1, s_2, \dots, s_n , são adquiridos por n transdutores, chamados de hidrofones, que compõem o sistema de sonar passivo. Estes sinais são, geralmente, processados por um conformador de feixes. Após os sinais serem conformados, estes podem ser observados nos domínios do tempo e da frequência. Os dois domínios podem ser utilizados para detecção dos sinais, e, uma vez que os sinais foram detectados, o seu acompanhamento pode ser realizado. Com os contatos sendo acompanhados, o processo de classificação é iniciado.

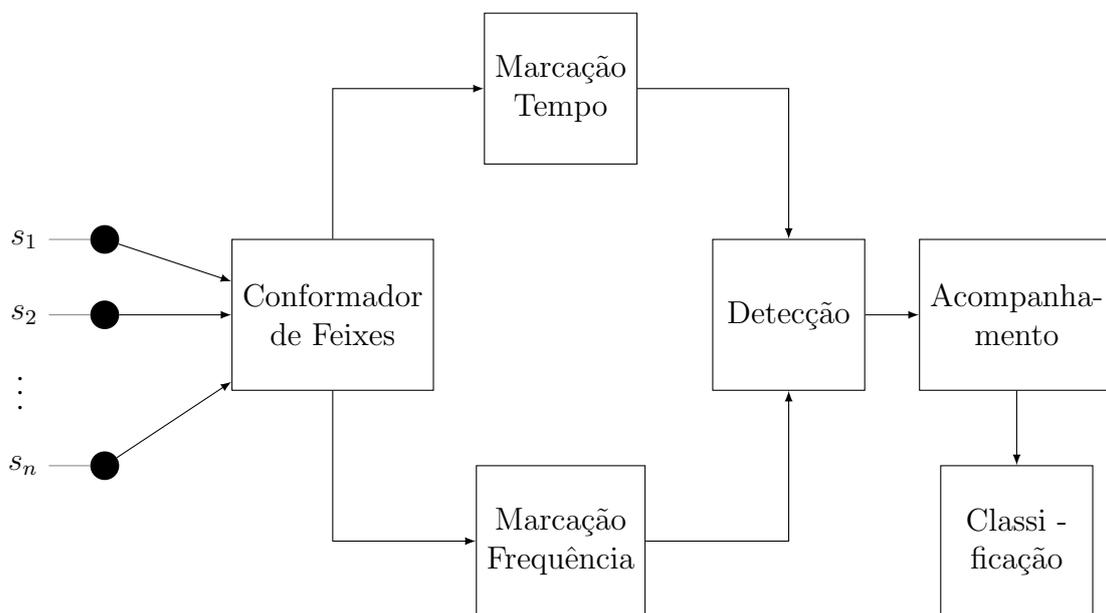


Figura 2.4: Diagrama em blocos de operação de um sistema de sonar passivo.

2.2.1 Conformador de Feixes ou *Beamforming*

O processo de conformação de feixes pode ser visto como sendo uma filtragem espacial dos sinais provenientes dos transdutores para a obtenção das DOA. A literatura discorre sobre diversas técnicas para a estimação das direção de aproximação [42–50]. Basicamente, essas técnicas consistem em fazer a composição dos sinais de maneira a cancelar ou adicionar atrasos de tempo decorrentes das diferenças de tempo da chegada da frente de onda em cada um dos sensores.

Para cada geometria do conjunto de hidrofones, chamado de *array* de hidrofones, um tipo de compensação deve ser empregado. As técnicas mais simples de *beamforming* consistem na aplicação de coeficientes de filtros previamente definidos de

acordo com a geometria do *array*. Técnicas mais complexas de conformação de feixe visam a otimização da resposta espacial do *array*, com diferentes funções objetivos, seja para a redução dos lóbulos laterais na resposta do conformador [48], seja para a minimização da variância em frequência da resposta do conformador [49]. Recentemente, um estudo sobre a quantidade de transdutores a ser utilizada para a formação de feixes e novos métodos de formação de feixes baseados em deconvolução do ruído envolvido na leitura de cada um dos hidrofones visando maximização da razão sinal-ruído foi desenvolvido em [50].

2.3 Análises de Marcação no Tempo

Uma vez que os sinais provenientes dos transdutores estão conformados formando um conjunto de feixes, pode se iniciar as primeiras análises para a detecção dos contatos, que geraram os sinais adquiridos. A partir deste ponto, é criado um mapa que representa a energia de cada DOA, ao longo do tempo. Geralmente, esse mapa é preenchido ao longo do tempo, formando um gráfico em cascata ou *waterfall display* e que é chamado de gráfico Marcação versus Tempo. Na Figura 2.5, um exemplo para este tipo de gráfico é mostrado, na qual pode ser observada a presença de algumas estrias mais escuras. Estas estrias são pontos de maior concentração de energia nas direções em questão ao longo do tempo. Além disso, na parte superior deste gráfico, tem-se também a energia média (ao longo do tempo) e a instantânea (no último instante de tempo) para cada uma das direções.

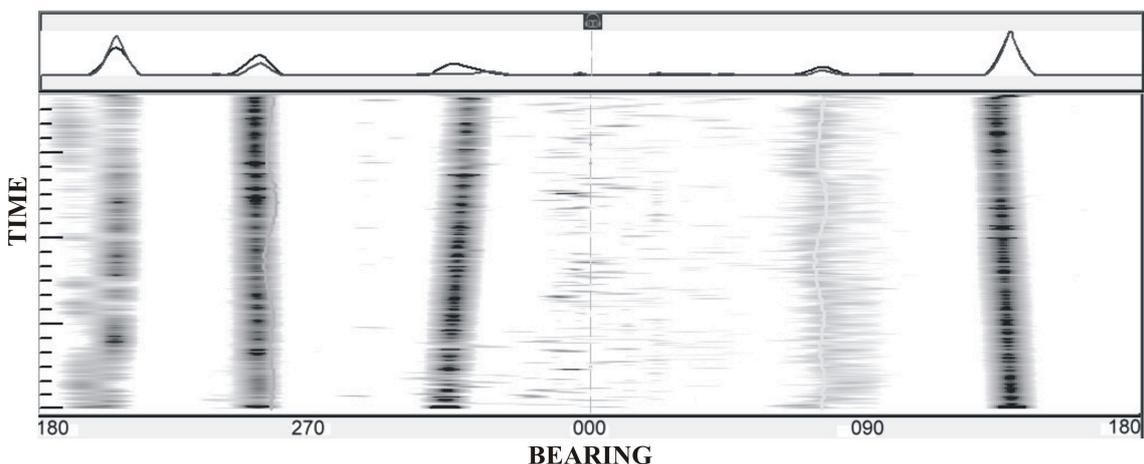


Figura 2.5: Gráfico de Marcação vs Tempo

A análise da energia depositada em uma dada direção pode ser utilizada como um método de detecção de contatos. Se a energia em uma dada direção se destacar frente ao ruído de fundo, esta direção pode conter um navio de interesse, que será posteriormente acompanhado e classificado.

Uma das análises que podem ser aplicadas nesta etapa é a análise DEMON (Figura 2.6). Esta análise tem por objetivo extrair características da propulsão do navio detectado. Em um cenário tático, não é incomum que mais de um navio possa ser observado na mesma DOA. Assim, algumas técnicas podem ser aplicadas com o objetivo de se obter uma melhor separação de sinais de interesse neste nível de análise. Um dos exemplos que pode ser citado é a aplicação de técnicas de Separação Cega de Fontes (BSS), como por exemplo a análise de componentes independentes (ICA) que visa a independência entre fontes extraídas de uma mesma direção. Esta técnica foi aplicada eficientemente a sinais de sonar passivo processado com análise DEMON em [51].

Um outro trabalho que foi desenvolvido neste nível de processamento foi a análise de estacionaridade utilizando redes neurais com treinamento não-supervisionado [52]. Neste trabalho, redes neurais baseadas na teoria de ressonância adaptativa (ART) foram aplicadas à sinais de sonar passivo. Um das características das redes ART é a capacidade de identificar padrões que se distanciem dos padrões conhecidos no espaço de características, e, geralmente, a distância em questão é a distância euclidiana. Para o trabalho desenvolvido em [52], caso algum dos padrões observados fosse considerado fora dos padrões conhecidos do modelo, ou seja, com uma distância euclidiana maior do que um limiar definido, a hipótese da presença de estacionaridade é descartada.

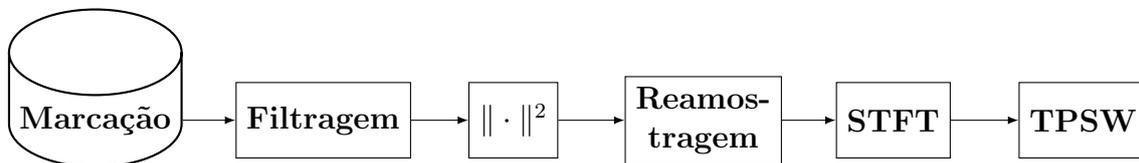


Figura 2.6: Diagrama em blocos da Análise DEMON

2.4 Análises de Marcação em Frequência

O sinal conformado pode ainda ser analisado em outro domínio, o domínio das frequências. A análise da marcação no domínio das frequências tem por objetivo encontrar tons frequenciais que auxiliem as tarefas detecção e classificação de contatos. Neste nível, um outro mapa é analisado, o chamado mapa de Frequência vs Tempo, que mostra a energia em cada uma das frequências ao longo do tempo para uma DOA. Depois de detectar um contato na análise da marcação no tempo, a detecção pode ser confirmada pelo comportamento espectral de direções selecionadas após a conformação de feixes.

Neste nível de análise, a análise LOFAR, que tem o seu diagrama em blocos

mostrado na Figura 2.7, pode ser destacada [31] e um exemplo do seu resultado com $\Omega = 3$ pode ser visto na Figura 2.8. Aplicando essa análise, foram desenvolvidos diversos trabalhos científicos em colaboração com a Marinha do Brasil [31, 53–57]

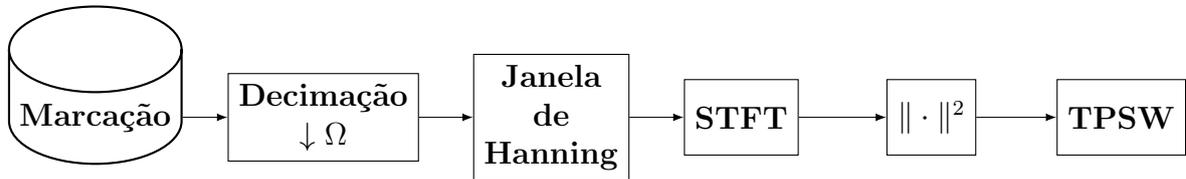


Figura 2.7: Diagrama em blocos da Análise LOFAR

LOFARgram for ClassA with Decimation Ratio: 3 and 1024 FFT Points

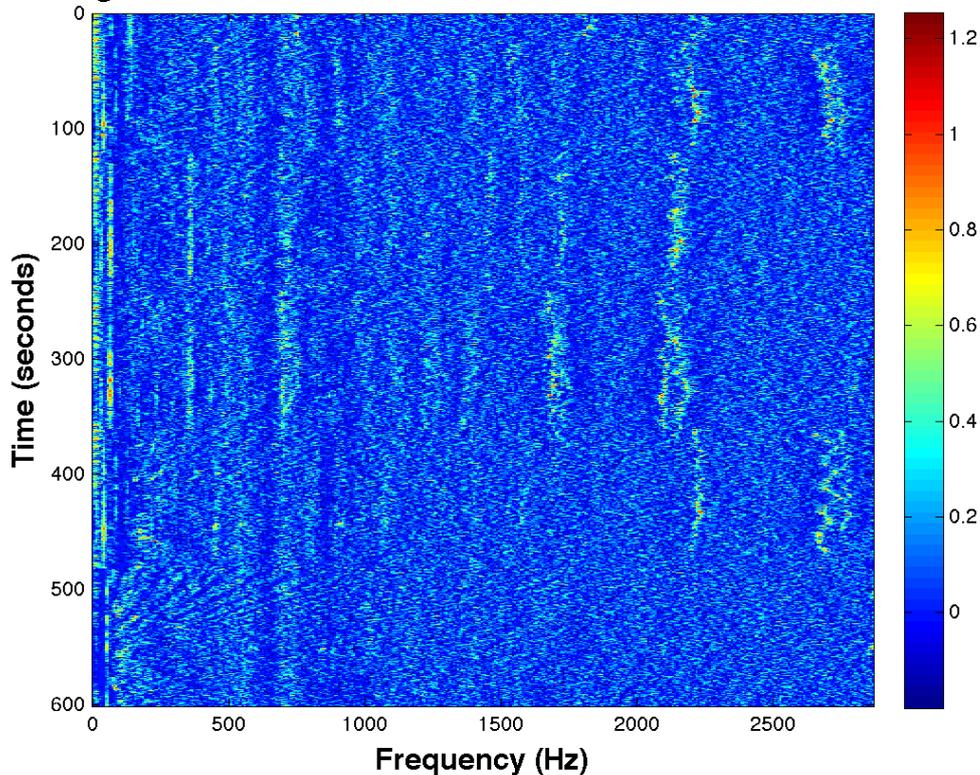


Figura 2.8: Gráfico de Frequência vs Tempo

2.5 Tracking

Depois de acessar as informações das diferentes direções tanto no domínio do tempo quanto no domínio da frequência, a detecção de um contato pode ser feita. Após o processo de detecção, que, geralmente, considera um limiar determinado em conformidade com a equação sonar (Equação 2.2), o processo de acompanhamento do sistema de sonar passivo é ativado.

O processo de acompanhamento, ou *tracking*, recebe as informações dos subsistemas anteriores e faz uma classificação simples para identificar para qual direção o contato em questão está se dirigindo. De posse dessa classificação, o sistema de *tracking* pode acompanhar cada um dos contatos classificados ao longo do tempo.

Diversas técnicas de acompanhamento em sistemas de sonar passivo podem ser citadas [58–63]. Entre os artigos citados anteriormente, pode-se destacar [58], que propõe a utilização de um sistema de acompanhamento bayesiano baseado na função de verossimilhança, alimentado com informações ao nível de marcação no tempo. Já em [59] é proposto, mais uma vez, um sistema de acompanhamento bayesiano, mas que acessa informações ao nível do conformador de feixes.

2.6 Classificadores

Uma vez que o sinal de um contato foi conformado, detectado e acompanhado, esse contato deve ser classificado [31]. Com acesso total às informações sobre o contato, o processo de classificação pode se dar de diversas formas.

Em [31] é proposto um classificador baseado em modelos de redes neurais artificiais alimentados com a informação das marcações em frequência (após a realização da análise LOFAR). Já em [53], a análise do impacto de uma compactação de dados em sinais de sonar passivo processados com a análise LOFAR é desenvolvida. Em [54, 55], técnicas de classificação baseadas em classificadores especialistas, do tipo classe-não-classe, foram aplicadas para sinais de sonar passivo. Uma outra forma de visualização de dados amplamente utilizada é a transformada Wavelet [64, 65], aplicada à classificação de sinais de sonar passivo em [57].

Além das técnicas descritas anteriormente, destacam-se também os artigos [66–68] que apresentam outras implementações para a classificação em sinais de sonar passivo. Em [66], a informação utilizada para o treinamento do classificador foi extraída utilizando os chamados *mel-frequency cepstrum* (MFC). Os MFC são uma representação do espectro de potência em uma curta janela de aquisição de dados. Para a sua extração, devemos fazer uma transformação cossenoidal do logaritmo do espectro de potência na escala mel (*mel scale*). Essa representação tem sido utilizada com sucesso para aplicações em áudio, como reconhecimento de fala, extração de características sonoras e classificação de falantes [69–71]. Em [68], técnicas de *Deep Learning* são introduzidas para a classificação em um ambiente submarino com resultados promissores.

Capítulo 3

Detecção de Novidade

Este capítulo visa a descrição de métodos de detecção de novidade, quer no ambiente de sonar passivo, quer em outros ambientes de pesquisa.

O reconhecimento de padrões tem suas origens na engenharia, enquanto o aprendizado de máquina cresceu a partir da ciência da computação [72]. Ambas as atividades podem ser vistas como duas faces de uma mesma moeda e, em conjunto, elas passaram por um desenvolvimento substancial nas últimas décadas, devido a dois fatores: primeiramente, o poder computacional aumentou sensivelmente nos últimos anos; em segundo lugar, o aumento na complexidade das tarefas a serem executadas automaticamente, que faz com que modelos mais simples, como modelos lineares, não obtenham bons resultados.

A área de reconhecimento de padrões tem uma história longa e bem-sucedida e é, por muitas vezes, considerada fundamental para o desenvolvimento de outras áreas de conhecimento. Por exemplo, as extensas observações de padrões astronômicos por Tycho Brahe no século XVI permitiram a Johannes Kepler descobrir as leis empíricas do movimento planetário que, por sua vez, proporcionaram um trampolim para o desenvolvimento da mecânica clássica [73]. O campo do reconhecimento de padrões está relacionado com a descoberta automática de características nos dados através do uso de algoritmos computacionais, bem como o uso dessas características para a tomada de decisões.

Basicamente, o reconhecimento de padrões pode ser resumido como a estimação de uma função $f(\mathbf{x})$, que recebe um vector de variáveis \mathbf{x} , chamado vector de entradas e , mapeia estas entradas em um valor y , como pode ser visto na Equação 3.1. Esta função pode ter como objetivo a estimação do valor uma variável ou a separação de classes.

$$y = f(\mathbf{x}) \tag{3.1}$$

Diversas técnicas são utilizadas para o treinamento de classificadores [72]. Neste

trabalho, utilizaremos: redes neurais artificiais [26], redes neurais com técnicas de *deep learning* [27] e máquinas de vector suporte [25]. Cada técnica tem como objetivo gerar uma função $f(\mathbf{x})$ que mapeia a entrada \mathbf{x} no alvo do treinamento y (como descrito na equação 3.1), embora cada uma das técnicas possua suas nuances e características próprias que serão abordadas a seguir.

Tipicamente, técnicas de classificação tentam resolver problemas com duas ou mais classes [72]. Os problemas de classificação com mais de uma classe, geralmente, podem ser decompostos em problemas de classificação de duas classes, que são considerados mais simples [74]. Tendo em vista esse fato, o problema de duas classes pode ser considerado a tarefa básica de classificação [75]. Em alguns casos mais específicos, o sistema de classificação pode se deparar com classes não disponíveis durante o seu desenvolvimento, assim sendo, uma etapa do sistema de classificação deve ser desenvolvida para detectar novas classes [76].

A detecção de novidade pode ser definida de diferentes formas. Alguns autores fundem a detecção de novidade com a detecção de *outliers* ou detecção de anomalias em uma mesma tarefa [77], enquanto outros desmembram cada uma destas técnicas por seus principais objetivos [78]. A detecção de novidade pode ser definida como sendo a detecção de classes que não estavam presentes durante o processo de desenvolvimento do sistema de classificação. Já a detecção de anomalias visa classificar dados que têm um comportamento que foge do comportamento médio dos eventos pertencentes à classe, onde o comportamento médio pode ser definido como o comportamento mais provável de uma dada classe. Na essência, ambas as aplicações podem ser interpretadas da mesma maneira: têm-se dois conjuntos de eventos, um com dados de treinamento e outro com dados de teste, que podem conter eventos desconhecidos ou anômalos. O primeiro definirá o padrão de conhecimento, ou seja, o comportamento normal das classes conhecidas; enquanto os outros dados vão ser analisados e, caso se diferenciem dos primeiros, de acordo com alguma figura de mérito, estes serão classificados como não-pertencentes ao padrão de conhecimento, quer por pertencerem a uma nova classe (detecção de novidade), quer por não pertencer ao padrão de conhecimento definido pelo conjunto de treinamento (detecção de anomalias).

A detecção de novidade, por vezes, deve ser aplicada devido à própria natureza da aplicação. Em aplicações onde o custo de aquisição de eventos de novas classes pode ser proibitivo, a detecção de novidade pode ser aplicada como um etapa do processamento dos dados, fazendo assim com que dados de novas classes sejam adquiridos de maneira automática. Outras aplicações em que se tem uma escassez de exemplos de outras classes motivar a aplicação de tal técnica. Por exemplo, em um sistema de monitoramento de um motor de um automóvel, duas classes podem ser modeladas, essencialmente. Uma que representa o motor em pleno funcionamento,

ou em seu estado normal, e outra classe que representa o motor operando com uma falha, ou em um estado anormal. Se o estado anormal leva o motor à sua destruição, obter eventos desta classe custaria o valor de um motor. Assim, no exemplo dado, a frequência de ocorrência da classe “motor em anomalia” é muito baixa e o custo de obtenção de um de seus exemplos é igual ao custo de fabricação de um motor novo, enquanto que exemplos de um motor operando em seu estado normal custariam bem menos, neste caso. Desta maneira, pode-se utilizar um detector de novidade para monitorar o motor, e quando este entrar em um estado de operação anormal, um alarme pode ser acionado e os parâmetros da operação podem ser registados automaticamente.

Não são raras as aplicações na literatura onde se encontram esse tipo de cenário e os exemplos a seguir podem ser citados: detecção de células que podem formar câncer de mama [79] (células normais em exames de mamografia são extremamente mais abundantes do que células de formação de câncer); detecção de fraudes em sistemas de cartões de crédito e telefones celulares [80–82] (o comportamento de usuários fraudulentos costuma ser raro) e detecção de neologismos em textos americanos [83], entre outros.

Existem diversas formas de se atacar o problema de detecção de novidade. De maneira geral, a mais utilizada é a chamada classificação de uma única classe (*one-class classification*) [84–86]. A classificação de uma única classe consiste em modelar os dados conhecidos como pertencentes a uma classe, neste caso, chamada de classe normal ou positiva. Neste caso, os dados conhecidos, mesmo que pertencentes a diferentes classes, são apresentados para o sistema de classificação como membros de uma única classe, a classe conhecida. Geralmente, assume-se que a classe conhecida é muito bem amostrada, enquanto outras classes estão severamente sub-amostradas [87, 88].

De maneira geral, os padrões conhecidos são apresentados aos modelos durante o seu treinamento, e estes são treinados para reconhecer os mesmos. Depois do processo de treinamento, o processo de análise consistirá na apresentação de dados desconhecidos para o modelo, e a estimação de figuras de mérito específicas. De aumentar a eficiência na detecção de novidade é aumentar o nível de exigência do modelo para a classificação de classes conhecidas. Quando aumenta-se este nível de exigência, mais eventos desconhecidos serão corretamente classificados. Em compensação, devido ao aumento da exigência de classificação de classes conhecidas, mais eventos conhecidos poderão ser classificados como pertencentes às classes desconhecidas, e assim, o impacto do processo de detecção se torna evidente. Devido a isto, existe um compromisso entre a eficiência de classificação e a detecção de novidade em sistemas de apoio à tomada de decisão que compartilham os mesmos recursos computacionais. A detecção de novidade pode ser realizada com diferentes

técnicas, cada qual com suas características. Assim sendo, o ato de aumento o nível de exigência pode ser mapeado como um hiper parâmetro a ser estimado.

As técnicas de detecção de novidade podem ser classificadas em cinco categorias gerais [84]: (i) probabilística, (ii) baseadas em distância, (iii) baseadas na reconstrução, (iv) baseadas em domínio e (v) baseadas em teoria da informação.

A abordagem (i) considera métodos probabilísticos que, muitas vezes, envolvem a estimativa da função densidade de probabilidade da classe normal. Esses métodos pressupõem que áreas com baixa densidade de eventos no conjunto de treinamento indicam áreas com baixa probabilidade de padrões conhecidos ocorrerem [77, 89–91].

A abordagem (ii) inclui os conceitos de vizinhança mais próxima e análise de agrupamentos que são comumente utilizados em problemas de classificação. O pressuposto aqui é que os dados normais têm uma maior probabilidade de se encontrarem agrupados, enquanto os dados da classe novidade não possuem uma estrutura de agrupamento e se encontram espalhados [92–95].

A abordagem (iii) envolve o treinamento de um modelo de reconstrução de eventos que captura a estrutura do dado de entrada. Quando os dados da classe desconhecida são mapeados pelo modelo treinado, o erro entre a saída do modelo e a entrada se torna um *score* de novidade e, no caso de dados desconhecidos, tende a ter um valor elevado. Redes neurais artificiais podem ser utilizadas nessa abordagem através da estrutura de *auto-encoders* [96–99].

A abordagem (iv) usa métodos baseados em técnicas que determinam um domínio para caracterizar os dados de treinamento. Durante o treinamento, um limite que envolve os dados normais é determinado visando, geralmente, que o hipervolume desse limite seja o menor possível. Uma vez de posse desse limite, os dados que forem mapeados fora desse limite são classificados como dados de novidade [100, 101].

A abordagem (v) utiliza informações de estatística estatística de ordem superior, extraídas dos dados de treinamento, e em conjunto com medidas de teoria da informação, tais como a entropia ou a discrepância de Kolmogorov, para analisar se um dado é ou não pertencente à classe normal [102–104]. O conceito principal nessa abordagem é que se um conjunto de dados possui uma função densidade de probabilidade muito diferente da função densidade de probabilidade do conjunto de dados de treinamento, provavelmente os dados não pertencem a nenhuma das classes conhecidas, dentro de uma tolerância. Se os dados do conjunto de teste apresentam distribuições diferentes das distribuições estimadas no conjunto de teste, a hipótese de que os dados pertençam à classes conhecidas pode ser descartada.

O enfoque deste trabalho são técnicas baseadas em domínio. Os métodos baseados em domínio criam um limite com base na estrutura do conjunto de dados de treinamento. Esses métodos são, tipicamente, insensíveis à amostragem e a densi-

dade específicas da classe normal, porque descrevem o limite ou o domínio da classe normal e não estimam a PDF da classe, que geralmente é um processo mais complexo e custoso computacionalmente [100].

Um exemplo da aplicação de detecção de novidade baseada em domínio pode ser visto na Figura 3.1. Os exemplos dos dados disponíveis durante o processo de treinamento são representados, em duas de suas dimensões, por círculos azuis, enquanto os quadrados vermelhos representam dados de classes desconhecidas. Além disso, dois domínios são mostrados: o verde, que possui uma área menor e captura todos os eventos conhecidos; e o laranja, que possui uma área maior e não captura todos os eventos disponíveis durante o treinamento.

Neste exemplo, o nível de exigência de classificação de classes conhecidas pode ser modelado como inversamente proporcional a área do domínio. Assim sendo, a exigência para que um evento conhecido seja classificado como conhecido para o domínio em laranja é muito pequena. Pela estrutura do domínio em verde, pode-se observar que a probabilidade de um exemplo pertencente à classe conhecida, que possua características ligeiramente diferentes dos eventos apresentados durante o processo de treinamento, pode ser facilmente classificado como pertencente à classe novidade. Ou seja, neste caso, a perda de generalização do modelo leva a uma maior eficiência de novidade.

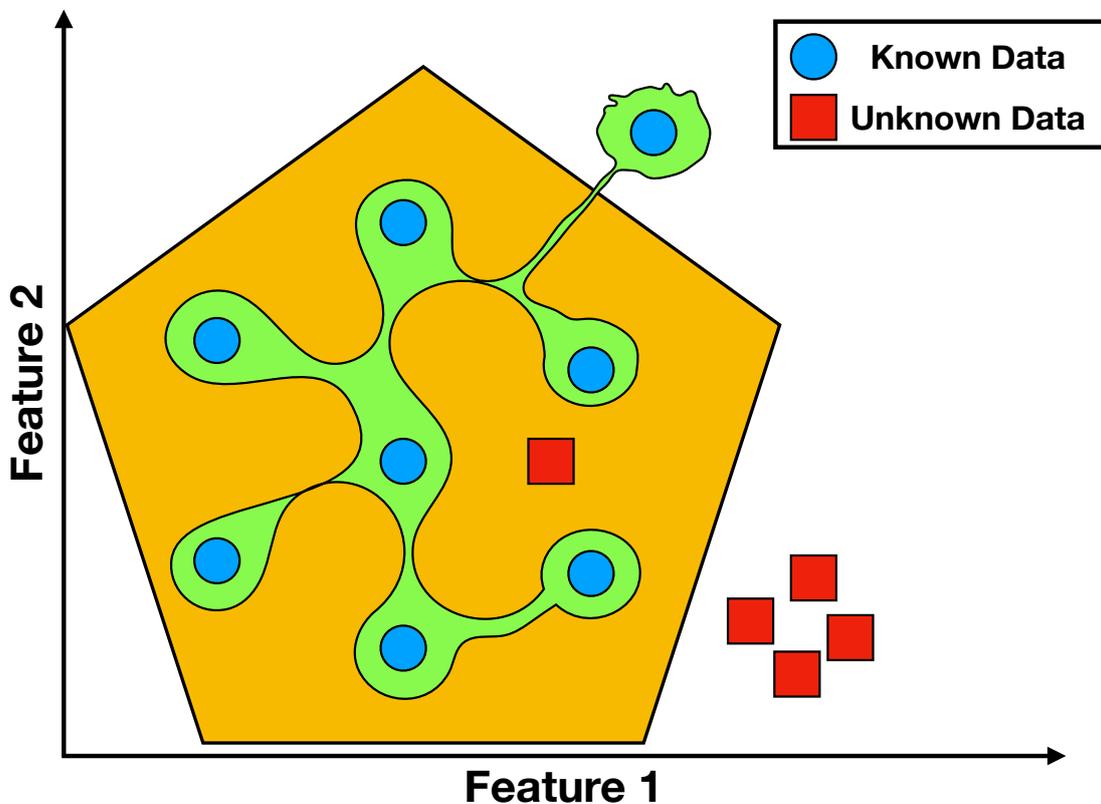


Figura 3.1: Um exemplo de detecção de novidade baseada em domínio

Por muitas vezes, um modelo pouco complexo, como visto no domínio laranja apresentado na Figura 3.1, pode apresentar uma capacidade de generalização maior do que modelos mais complexos (domínio verde da mesma figura). Este fato pode resultar em modelos com uma maior eficiência de classificação para dados pertencentes às classes conhecidas. Por outro lado, domínios menos complexos podem capturar dados desconhecidos e classificá-los como dados conhecidos. Deste modo, para a detecção de novidade, um domínio com menor área deve ser empregado.

Como exemplo, a Figura 3.2 pode ser analisada. Nesta figura, as eficiências na detecção de novidade de quatro modelos treinados podem ser observadas e, cada uma das curvas, representa a eficiência na detecção de novidade para um modelo diferente. No modelo representado pela curva preta, com um baixo nível de exigência, a eficiência na detecção de novidade obtida apresenta seu valor de máximo. Este comportamento, para determinadas aplicações, pode ser considerado o ideal.

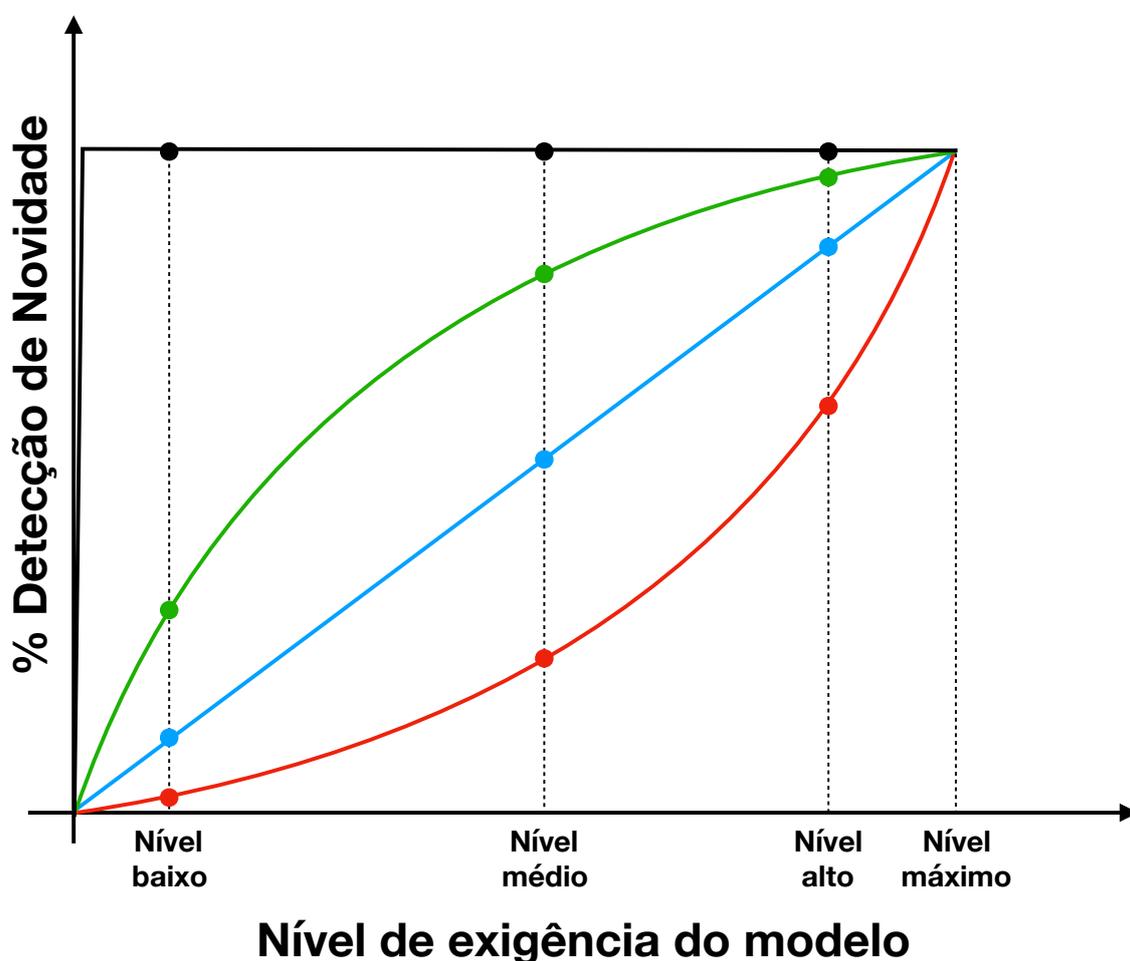


Figura 3.2: Um exemplo de como o nível de exigência para um modelo treinado pode ser utilizado para detectar novidade

A curva em vermelho ilustra um modelo com resultado diferente do modelo ideal. Neste caso, para valores de exigência próximos do máximo, a curva em vermelho

apresenta valores de detecção de novidade próximos das outras 3 curvas apresentadas. Este modelo de detecção apresenta valores menores do que os demais para níveis, sendo considerado assim, um modelo mais distante do ideal. Caso este modelo estivesse operando em conjunto com um modelo de classificação, os impactos na eficiência de classificação seriam maiores para a obtenção de um mesmo nível de detecção de novidade. Além dos modelos analisados anteriores, a curva em azul também foi apresentada. Neste modelo teórico de detector de novidade, apresenta valores de detecção de novidade maiores do que o modelo representado em vermelho, sendo assim, este pode ser considerado melhor para a detecção de novos eventos, enquanto que o modelo vermelho pode ser considerado pior. Por último, o modelo teórico a ser observado é o modelo teórico representado pela curva em verde. Este modelo ilustra um modelo de detecção de novas classes que é mais eficiente do que os modelos em vermelho e azul. Este resultado pode ser observado pelo valores de detecção obtidos em cada um dos níveis de exigência propostos. O nível de exigência do modelo pode ser exemplificado de diferentes maneiras. No caso de modelos baseados em domínio, o nível de exigência pode ser definido como o hiper-volume definido pelo modelo.

Além disso, para aplicações mais sensíveis a erros, apresenta-se outro problema: a ponderação do erro. Voltando ao exemplo do monitoramento do motor de um automóvel, o erro em se classificar o estado de operação deve ser ponderado. Neste caso, um falso alarme de mau funcionamento custa menos do que a explosão do motor, ou seja, naturalmente, o erro em classificar um dado conhecido como desconhecido custa muito menos do que classificar um exemplo desconhecido como conhecido e não soar o alarme. No exemplo da Figura 3.1, o domínio laranja deixa de classificar corretamente um dos dados conhecidos, devido a sua simplicidade e poder de generalização e, além disso, classifica erroneamente um dos exemplos de novidade como dado conhecido, diferente do domínio em verde, que classifica corretamente todos os exemplos das classes conhecidas com uma área menor. Este domínio pode apresentar um processo de sobre-especialização do treinamento, ou *overfit*.

Para se realizar a detecção de novidade baseada em domínio, algumas técnicas podem ser destacadas. Uma das técnicas mais populares aplicadas são as SVM de uma única classe ou *one-class SVM* [25, 101]. Esta técnica foi aplicada com sucesso nas mais diversas áreas, tais como: detecção de eventos anormais através de áudio em sistemas de segurança [105], detecção de fraudes em assinaturas [106], detecção de intrusos em comunicações industriais [107], detecção de falhas em máquinas [108], detecção de *malwares* [109] e predição de falhas em *softwares*.

Em [105], a técnica de *one-class SVM* foi aplicada em dados provenientes de sistemas de segurança. Os dados deste trabalho eram informações de vídeo e áudio de um sistema de segurança voltado para o monitoramento de multidões. O foco

aqui era reconhecer um padrão de anomalia no sinal de áudio e detectar um possível padrão anormal ao sistema. Neste caso, os coeficientes cepstrais foram extraídos do sinal de áudio e alimentaram um modelo de detecção de anomalias.

Um outro trabalho que pode ser citado é [106], onde a técnica foi aplicada para a detecção de fraudes em assinaturas. Para a detecção de fraudes, idealmente, duas classes podem ser modeladas: uma com a assinatura verdadeira e outra com a assinatura falsa. Obviamente, as assinaturas falsas são mais custosas, uma vez que podem estar envolvidas em processos de investigação ou outros trâmites legais. Assim sendo, o banco de dados utilizado neste trabalho dispunha apenas de dados com assinaturas verdadeiras, ou seja, um problema de classificação em duas classes se tornou um problema de uma única classe. Para atacar este problema, uma transformada derivada da transformada Wavelet para imagens, chamada de transformada Curvelet [110], foi utilizada em conjunto com *one-class* SVM.

A técnica de *one-class* SVM também já foi aplicada a dados de comunicação industrial em [107], que são considerados sensíveis devido a sua confidencialidade. Na aplicação proposta em [107], a detecção de novidade se resume a detecção de uma possível tentativa de invasão na comunicação. Claramente, a classe normal possui uma quantidade significativamente maior do que a classe anormal, e, assim sendo, o problema foi atacado como um problema de uma única classe. Como dados de invasão em comunicação industrial são muito raros, os dados sem intrusos foram utilizados para o desenvolvimento dos modelos. Um ponto que vale ser observado aqui é que o treinamento dos modelos foi realizado com base na técnica de otimização intitulada enxame de partículas (PSO).

Um dos casos clássicos de aplicação de *one-class* SVM pode ser encontrado em [108]. Neste trabalho, dados adquiridos por sensores posicionados em máquinas industriais com motores que provocam vibrações, foram para o desenvolvimento de detectores de anomalias, sendo que estes objetivavam a detecção de pontos de operação que poderiam danificar os equipamentos estudados devido a amplitude da vibração. Os dados utilizados como entrada continham informações temporais e frequenciais e o trabalho comparou duas implementações de SVM, a implementação clássica de SVM para duas classes e a técnica *one-class* SVM, bem como uma outra técnica padrão da área. Os melhores resultados foram obtidos com *one-class* SVM.

Malwares são pequenos *softwares* programados para roubar informações e dados, e um problema frequente para usuários comuns da internet [111]. Em [109], a proposta de aplicação de *one-class* SVM para a detecção de *malwares*. Os dados explorados foram de um banco de dados disponibilizado pela *Microsoft* para um desafio em reconhecimento de padrões. Um fato interessante sobre o trabalho apresentado em [109] é o processo de treinamento, que foi desenvolvido com um número diferente de informações de entrada do que o disponibilizado inicialmente para o

desafio. Estas informações extras foram obtidas depois de um processamento de linguagem natural, e chamadas de informações privilegiadas. Este novo conjunto de entradas foi selecionados para uma melhor convergência do processo de treinamento bem como em uma tentativa de aumentar a eficiência do modelo em comparação com modelos aplicados durante o desafio da *Microsoft*.

Sistemas de predição de falhas em software vêm sendo empregados extensamente [112]. Por este motivo, têm chamado atenção da comunidade científica devido a sua complexidade, e por se tratar de uma aplicação desafiadora. Programas de computador desenvolvidos por seres podem apresentar uma grande quantidade de falhas, que se originam de diferentes maneiras. Como um *software* é programado para funcionar corretamente, a presença de falhas durante a sua operação é bem mais rara do que a operação plena da aplicação. Assim sendo, mais uma vez, um problema de uma única classe pode ser explorado nesta situação. Em particular, um software desenvolvido para testes em predição de falhas foi utilizado para gerar pares de entrada e saída com e sem a presença de falhas para o treinamento de modelos baseados em SVM [113]. Com isso, foram gerados 6 bancos de dados extremamente desbalanceados e os resultados indicaram que *one-class* SVM obteve um desempenho satisfatório em cada um deles.

Outra aplicação interessante para SVM em detecção de novidade é apresentada em [70]. Mais uma vez utilizando coeficientes cepstrais, o objetivo deste trabalho era classificar se o falante encontra-se sobre situação de estresse. O modelo aplicado neste sistema simples: todos os falantes presentes no conjunto de treinamento encontram-se em situação de relaxamento. Assim, como foram extraídos os coeficientes cepstrais, qualquer classe de fala que se diferenciasse da classe de relaxamento poderia ser classificada como uma de quatro classes de estresse possíveis, sendo esta classificação realizada por um modelo multi-classes SVM, cuja acurácia de classificação para o conjunto de teste foi superior a 98%

Tendo em vista os trabalhos anteriormente citados, mostra-se que a detecção de novidade utilizando SVM encontra-se como uma área de pesquisa ativa e costuma conduzir a bons resultados em diferentes áreas de pesquisa. Até o presente momento, duas publicações foram encontradas onde há aplicação da técnica de *one-class* SVM à sinais de sonar passivo, ambas desenvolvidas pelo autor dessa tese . Em uma delas, a aplicação de SVM em dados de sonar passivo compactados com a análise de componentes principais foi abordada [114] e na outra, os mesmos dados foram comprimidos com a análise de componentes principais por kernel e utilizados para alimentar um modelo baseado em *one-class* SVM. A referência ao último trabalho ainda não foi indexada pois o mesmo se encontra em processo de revisão na revista *Neurocomputing* da Elsevier.

Além dos trabalhos apresentados anteriormente, uma outra técnica deve ser des-

tacada para este problema: as redes neurais artificiais [26]. Foram encontrados trabalhos sobre redes neurais sendo aplicadas à detecção de novidade, embora os mesmos sejam mais focados em técnicas baseadas em reconstrução, como [96, 98, 99]. Em [98] redes neurais são aplicadas como *auto-encoders* para a detecção de eventos que se diferem do conjunto de treinamento em dados de invasão de redes de computadores [115] e dados de cancer de mama [116].

Uma aplicação de redes neurais para detecção de novidade que diferencia totalmente das anteriores pode ser encontrada em [97]. Neste trabalho, redes neurais artificiais são utilizadas como uma técnica de domínio e este trabalho será descrito posteriormente ainda neste capítulo.

3.1 SVM para Detecção de Novidade: SVM de uma única classe

Segundo a abordagem descrita em [25], a detecção de novidade pode ser vista como um processo de encapsulamento dos dados disponíveis durante o processo de treinamento. Assim, durante a operação do modelo, eventos que se diferenciarem, segundo uma medida de dissimilaridade previamente estabelecida, são classificados como eventos de novidade. A classificação na classe de novidade é então determinada pela localização dos dados com relação a região definida pelo processo de treinamento.

Tal como acontece com SVM de duas classes, a técnica de *one-class SVM* mapeia os dados em um espaço de alta-dimensão, potencialmente infinita, para obterem um maior potencial discriminatório dos dados. Neste novo espaço, busca-se maximizar a separação de tais dados da origem do mesmo. Em outras palavras, através da minimização do hiper-volume associado à região de separação, pode-se obter um hiper-plano que representa o melhor compromisso relativo à separação dos dados. Neste caso, alguns dos dados disponíveis durante o treinamento, a saber os que estão mais próximos do limite de decisão, suportam este limite, e são chamados de vetores de suporte. Todos os outros dados do conjunto de treinamento (aqueles que não são vetores de suporte) não são considerados ao se definir o limite de decisão.

Existem duas formas básicas de treinamento da técnica de *one-class SVM*. O método proposto em [101] basicamente separa todos os pontos de treinamento da origem depois do mapeamento resultante do truque de *kernel* (Equação B.2), e maximiza a distância do hiperplano à origem. Com isso, uma função não-linear é treinada, que assume o valor -1 para regiões no espaço de características onde não houveram eventos de treinamento e o valor $+1$ para regiões no espaço onde houveram eventos de treinamento, e esta função é associada a uma região de separação com o menor hiper-volume possível. De posse desta função binária, pode-se realizar

a detecção de novidade mapeando os dados no espaço de características e se os mesmos forem mapeados dentro da região com saída +1, eles serão classificados como pertencentes a classe normal. Caso algum ponto mapeado se encontre na região associada à saída -1, será classificado como novidade.

A função objetivo empregada durante o processo de treinamento é quadrática e pode ser vista na Equação 3.2.

$$\min_{w, \xi_i, \rho} \frac{1}{2} \|w\|^2 + \frac{1}{\nu n} \sum_{i=1} \xi_i - \rho$$

sujeita a:

$$\begin{aligned} (w \cdot \varphi(x_i)) &\geq \rho - \xi_i && \text{para todo } i = 1, \dots, n \\ \xi_i &\geq 0 && \text{para todo } i = 1, \dots, n \end{aligned}$$

(3.2)

O processo de minimização se dá em $\|w\|^2$, que é a norma do vector normal a margem margem de classificação, isto faz com que, o hiper-plano treinado possua a máxima distância possível para a margem, ou próximo tão próximo da máxima distância quanto se deseje. Além disso, a variável ν controla a suavidade da superfície de separação e a quantidade de vectores suporte (SV) que são selecionados durante o treinamento para agirem como variáveis de relaxamento (*slack variables*). Nestes pontos de relaxamento, a margem se adaptar à presença do SV, fazendo com que, independente da posição da margem, esta seja suportada por um ponto específico. Em outras palavras, este ponto é um ponto de exceção no processo de classificação.

Caso o hiper-parâmetro ν , que controla a probabilidade destes pontos de exceção, tenda a zero, poucos SV são selecionados e a superfície de decisão tende a se afastar do valor mínimo com relação ao hiper-volume. Quando o valor de ν tende à 1, a quantidade de SV selecionados tende à quantidade de pontos no conjunto de treinamento, o que faz com que a superfície de decisão tenha menor hiper-volume, em comparação às superfícies geradas com valores menores de ν . Em compensação, a probabilidade de memorização dos dados de treinamento aumenta, pois a superfície de decisão praticamente envolve cada ponto do conjunto de treinamento. Além dos parâmetros anteriormente discutidos, na Equação 3.2, temos o peso de cada um das variáveis de relaxamento ξ_i e o parâmetro ρ que determina a distância entre o ponto mais próximo da superfície de decisão e a mesma, com a distância normalizada por $\|w\|^2$.

Como a função objetivo para a técnica SVM é uma função convexa, isto é, possui somente um mínimo global, o processo de treinamento implementa um processo de otimização convexa [117]. Assim sendo, o processo de treinamento de um modelo

baseado em máquinas de vector suporte se torna um processo de otimização convexa se a função kernel que transformou os dados não for a distância euclidiana [118].

Utilizando os polinômios de Lagrange para fazer o processo de minimização, temos que depois do processo de treinamento, a função de decisão se torna a Equação 3.3, onde x' é um evento a ser classificado. Ou seja, se a subtração da projeção do ponto e a distância mínima ao plano de decisão for menor do que 0, o evento não está contido na região definida pelo treinamento e será, então, classificado como não pertencente a classe normal.

$$f(x') = \text{sgn}(w \cdot \varphi(x') - \rho) \quad (3.3)$$

Uma ilustração do comportamento da técnica de *one-class* SVM para um conjunto de dados sintéticos pode ser visto nas Figuras 3.3 e 3.4. Para a síntese dos dados utilizados neste exemplo, a classe considerou uma distribuição gaussiana com média zero e variância unitária, para a qual foram geradas duas nuvens de pontos, centradas em $(-2, -2)$ e em $(+2, +2)$, respectivamente. O treinamento foi realizado utilizando o kernel gaussiano assumindo alguns valores particulares de ν e de γ , o último correspondendo à largura de banda do kernel gaussiano. Além disso, os dados da classe desconhecida foram sintetizados a partir de uma distribuição uniforme, variando de -4 a 4 , nas duas dimensões. Na Figura 3.3 são apresentados os resultados para o parâmetro ν assumindo os valores $0, 1, 0,5$ e $0,9$, respectivamente, com o parâmetro γ fixado em $0,1$. Já para Figura 3.4 são apresentados os resultados foram obtidos com o parâmetro ν fixado em $0,1$ enquanto o parâmetro γ variou nos valores de $0, 1, 0,5$ e $0,9$, respectivamente.

Nas sub-figuras, que compõem ambas as figuras, existem três tipos diferentes de dados: os dados que foram utilizados durante o período de treinamento, dados que foram utilizados durante o período de teste e dados da classe de novidade. Os dados de treinamento são representados por pontos brancos enquanto os dados de teste são representados por pontos roxos. Além disso, os pontos associados à classe de novidade são representados pelos círculos amarelos.

Para realizar a detecção de novidade, foi treinada uma máquina de vector suporte para fazer o encapsulamento dos dados da classe conhecida. A região colorida na cor rosa, definida pela margem em vermelho, representa o espaço onde eventos da classe conhecida se encontram, enquanto que as regiões coloridas na cor verde são posições onde eventos de novidade se encontram próximos a margem de separação. Quando mais escuro a cor verde do região, mais próximo da margem os eventos de novidade associados a região em questão se encontram da margem. Além disso, os pontos que suportam a margem treinadas foram destacados sendo envolvidos com

círculos vermelhos.

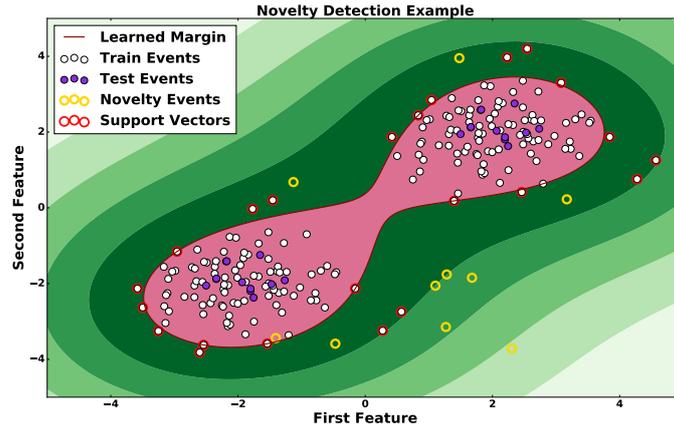
Como pode ser visto na Figura 3.3, quando aumentamos o valor do parâmetro ν , aumentamos a quantidade de SV selecionados, aumentamos a quantidade de variáveis de relaxamento e diminuimos a probabilidade de detecção das classes conhecidas, pois, uma vez que temos a maior probabilidade de memorização dos dados conhecidos, possivelmente, uma maior quantidade de dados conhecidos pertencentes ao conjunto de teste serão classificados como pertencentes a classe novidade. Além disso, vemos que a região vermelha (região que encapsula dados conhecidos) se torna cada vez menor, o que comprova o processo de *overfit*.

Quando observamos a Figura 3.4, é possível perceber as influências da largura de banda do kernel gaussiano aos resultados de detecção de novidade. Quando o valor de γ aumenta, a largura de banda do kernel é reduzida e, conseqüentemente, apresenta-se uma região de encapsulamento menor.

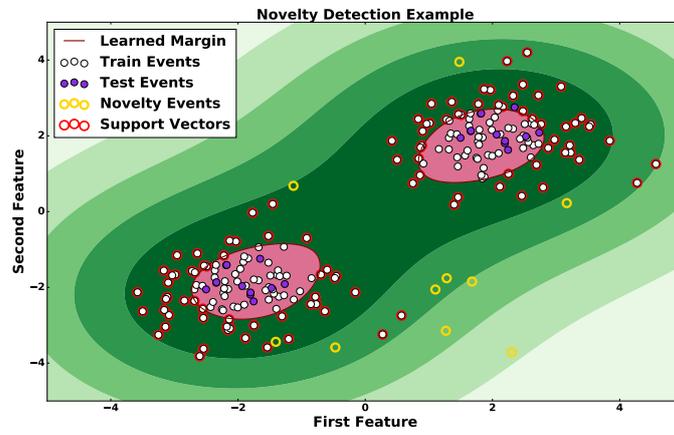
O método proposto em [100] tem uma abordagem diferente, pois o plano de decisão, neste caso, é substituído por hiper-esferas com raios iniciais pre-definidos, de maneira similar a uma rede neural do tipo ART [119]. A ideia é capturar todos os dados dentro de uma hiper-esfera no espaço de atributos, e reduzir o seu hiper-volume para a minimização da influência de *outliers*.

Como resultado do treinamento, uma hiper-esfera treinada para envolver os dados pode ser caracterizada por estar centrada no ponto \mathbf{x}_0 e possuir raio $R > 0$, que foi minimizado pelo algoritmo. Por definição, os pontos x_0 são os dados que suportam as esferas que definem o hiper-volume de encapsulamento de dados, ou seja os SV desta implementação. Em [101], foi proposta uma regularização análoga a utilizada no algoritmo *one-class SVM*, em que se permite que uma quantidade limitada de pontos se encontre fora da região associada as classes conhecidas. Assim, o processo de minimização pode ser visto na Equação 3.4.

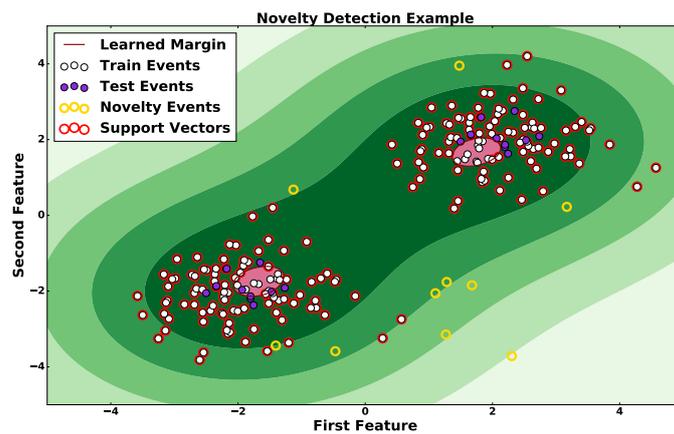
$$\begin{aligned}
 & \min_{R, x_0} R^2 + C \sum_{i=1} \xi_i \\
 & \text{sujeito a:} \\
 & \|x_i - x_0\| \leq R^2 + \xi_i \quad \text{para todo } i = 1, \dots, n \\
 & \xi_i \geq 0 \quad \text{para todo } i = 1, \dots, n
 \end{aligned} \tag{3.4}$$



(a) Exemplo de One-Class SVM ($\nu = 0,1$ e $\gamma = 0,1$)

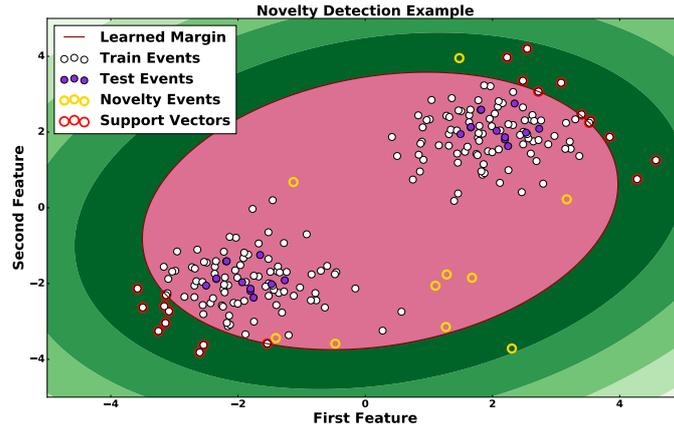


(b) Exemplo de One-Class SVM ($\nu = 0,5$ e $\gamma = 0,1$)

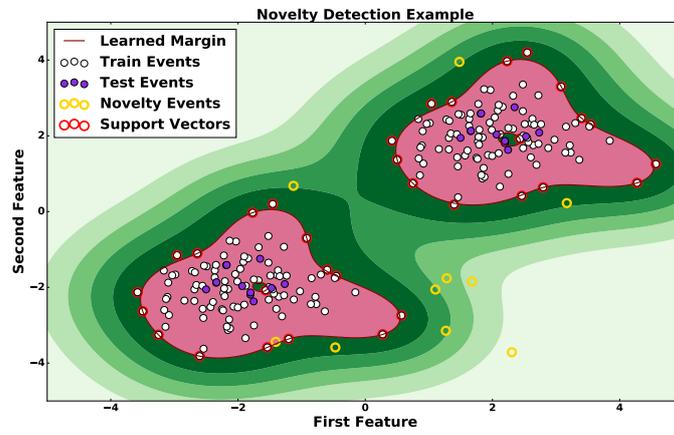


(c) Exemplo de One-Class SVM ($\nu = 0,9$ e $\gamma = 0,1$)

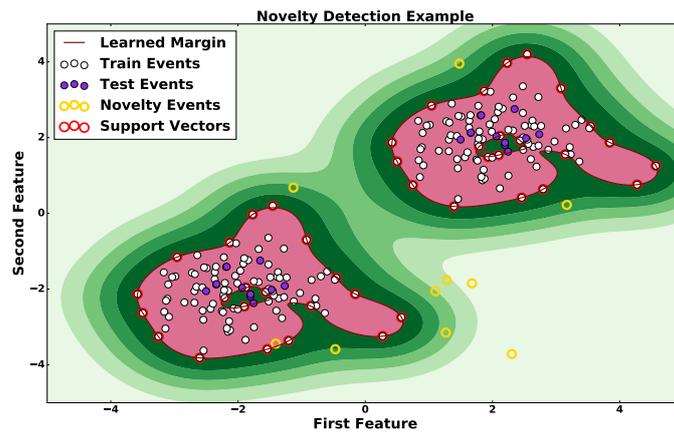
Figura 3.3: *One-Class SVM*: variação nos valores de ν



(a) Exemplo de One-Class SVM ($\nu = 0,1$ e $\gamma = 0,01$)



(b) Exemplo de One-Class SVM ($\nu = 0,1$ e $\gamma = 0,5$)



(c) Exemplo de One-Class SVM ($\nu = 0,1$ e $\gamma = 0,9$)

Figura 3.4: *One-Class SVM*: variação na largura de banda do kernel

3.2 Redes neurais para a detecção de novidade

Redes neurais artificiais têm sido utilizadas para a detecção de novidade segundo diferentes abordagens [85]. Em [120], são propostos dois métodos baseados em redes MLP. No primeiro, a saída do neurônio mais excitado na camada de saída é analisada e, caso não ultrapassasse um limiar de decisão arbitrário, o exemplo em questão é associado à classe de novidade. No segundo método, a distância entre o padrão obtido na saída e todos os alvos é medida e, se a menor distância obtida for maior do que um determinado valor, o dado de entrada é classificado como novidade. Os resultados obtidos em [120] indicam que os dois métodos obtiveram resultados similares para dados de imagem disponibilizados pela Força Aérea Americana.

Em [98] redes neurais auto-associativas, foram aplicadas ao problema de detecção de novidade, considerando uma reconstrução não-linear dos dados de entrada com o menor erro possível, visando acessar características contidas nos dados. Após uma compressão não linear, feita por três camadas ocultas [76], este modelo de MLP pode detectar novidade pelo erro de reconstrução, medido da entrada para a saída do modelo para um determinado exemplo. Caso este erro apresente valores muito altos, este padrão não foi visto pelo modelo durante o processo de treinamento e, assim, o exemplo associado a tal padrão será classificado como exemplo da classe novidade.

Este método de detecção foi aplicado em diferentes trabalhos [97, 121, 122]. Em [121], redes auto-associativas foram comparadas com técnicas baseadas em funções kernel para modelos alimentados com dados de equipamentos industriais (como plantas elétricas e motores) contaminados com falhas não disponíveis durante o treinamento. Além de serem aplicadas para extração de características, as redes auto-associativas também foram eficientes na compactação de dados em questão.

Já em [122], redes neurais apenas com duas classes como alvo, as chamadas redes neurais binárias, foram aplicadas à detecção de classes desconhecidas utilizando dois métodos de treinamento: não-supervisionado e supervisionado. No primeiro, os dados conhecidos foram compactados por meio de uma rede neural auto-associativa; enquanto no segundo, os dados conhecidos são rotulados como pertencentes a classe positiva e o processo de treinamento é realizado para minimizar o erro de mapeamento dos dados conhecidos para esta classe, sem exemplos da classe negativa sendo apresentados na etapa de treinamento. Para 3 diferentes bancos de dados (dados de monitoramento da caixa de marcha de helicópteros, dados de imagens do DNA humano e monitoramento sonoro do leito do mar), as redes neurais com treinamento não-supervisionado alcançaram resultados melhores do que suas contra-partes de treinamento supervisionado.

Em [97], tal qual em [120], foi definido um limiar (*threshold*), que é aplicado a

todas as saídas da rede neural. Os dados de sonar passivo da MB foram utilizados neste trabalho e, após um processo de compactação, alimentaram um modelo neural que foi treinado com saídas definidas através da técnica dos alvos maximamente esparsos, ou seja, com uma dimensão para cada classe totalizando, neste caso, 3 dimensões (uma dimensão para cada uma das classes disponíveis durante a etapa de treinamento). A técnica de compactação utilizada neste trabalho foi a PCD extraída por deflação, e que será apresentada detalhadamente no Apêndice B.

Como, cada dimensão do vector de saída está associada a uma classe conhecida, a detecção de novidade se dá, para um determinado evento, se nenhuma das saídas do modelo apresentou um valor maior do que o valor estabelecido para o limiar de classificação. O limiar de saída, neste caso, pode ser visto como o nível de exigência de classificação e, na Figura 3.5, apresenta-se um modelo baseado em redes neurais MLP é mostrado e, como pode ser visto, um limiar de classificação foi associado a camada de saída. Quanto maior o valor do estabelecido para o limiar de classificação, maior a exigência de excitação apresentada na camada de saída para classificação. Para se estabelecer um valor para o limiar de classificação a ser utilizado, no caso do método proposto em [97], uma das classes conhecidas foi utilizada como classe desconhecida e, portanto, não foi apresentada durante a fase de treinamento do modelo. Após o treinamento, os eventos desta classe foram apresentados ao modelo e a eficiência de detecção foi estimada. O valor de limiar foi obtido para permitir um valor de eficiência de detecção de novidade baixo, para não impactar negativamente na eficiência de classificação do modelo.

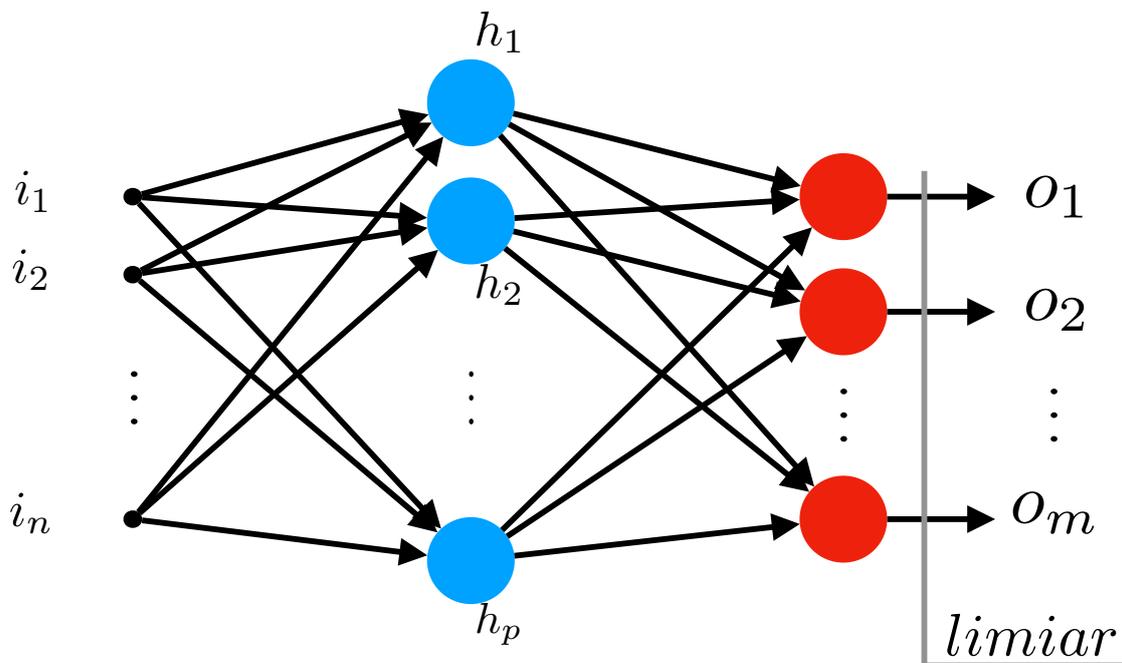


Figura 3.5: Modelo baseado em rede neural MLP com um limiar associado a saída.

Além disso, também em [97], houve a proposta de aplicação de um outro modelo de detector de novidade, este baseado em redes ART. Esse tipo de rede neural é utilizado para aprendizado não-supervisionado, e a sua topologia pode variar de acordo com os dados de entrada. Na Figura 3.6 encontra-se uma representação do modelo de um rede ART e, a rede opera avaliando a similaridade, no caso de [97], a função de similaridade foi a distância euclideana. Na figura, a função de similaridade foi estimada para cada uma das variáveis u_i , onde i é um padrão ativo da rede. Quando o valor de similaridade obtido é maior do que o parâmetro u_0 , conhecido como neurônio de similaridade máxima, outro padrão é criado no valor do neurônio. Quando o valor de similaridade máxima não é atingido, então, o neurônio que apresentou o menor valor de distância euclideana é atualizado. Ao final do treinamento, neurônios que foram ativados por poucos eventos são descartados.

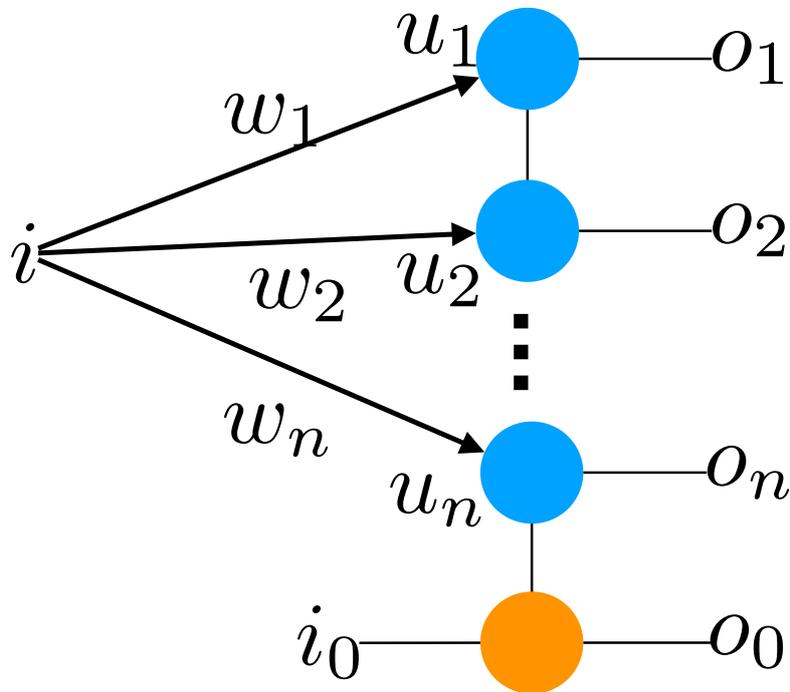


Figura 3.6: Modelo baseado em rede neural ART com raio de similaridade.

A detecção de novidade por modelos baseados em redes ART foi feito utilizando a informação sobre a criação de um novo padrão. Ou seja, se um novo padrão fosse criado, o evento era classificado como novidade e o padrão criado era salvo para uma posterior análise. No caso proposto em [97], a rede ART foi utilizada para classificação, com os padrões conhecidos sendo mantidos ativos, sem descarte ao final do treinamento.

3.3 Análise de Estacionaridade

Um processo estocástico ou aleatório é uma entidade matemática que é, normalmente, definida como uma coleção de variáveis aleatórias que se desenvolvem em um determinado período de tempo [123]. Historicamente, um processo estocástico pode modelar diversos fenômenos físicos como por exemplo: o crescimento de uma população bacteriana, valores nominais de uma ação na bolsa de valores ou valores de bins de frequência de um navio se deslocando no mar [124].

Estudos de fenômenos físicos, por sua vez, inspiraram o desenvolvimento de novos modelos de processos estocásticos que podem ser amplamente utilizados pela comunidade científica. Exemplos de tais processos incluem o processo de Wiener, que foi utilizado por Louis Bachelier para modelar as mudanças nas cotações de ações na Bolsa de Paris [125]; e o processo de Poisson, utilizado por A. K. Erlang para estudar o número de telefonemas que ocorrem em um determinado período de tempo [126].

Baseado nas suas propriedades, os processos estocásticos podem ser divididos em algumas categorias, que incluem *random walks*, *martingales*, processos de Markov (*Markov processes*), processos de Lévy (*Lévy processes*), processos gaussianos (*Gaussian processes*), campos aleatórios (*random fields*), entre outros [127].

Uma das propriedades mais importantes de um processo estocástico é a estacionaridade [20]. A estacionaridade pode ser observada quando a função densidade de probabilidade conjunta que gerou os dados do processo estocástico não muda ao longo do tempo de observação [123].

Dados que variam ao longo do tempo sempre foram um desafio ao processamento e ao tratamento dos mesmos, pois, como a estacionaridade não pode ser afirmada, a extração de características se torna muito mais desafiadora. Em [128], um estudo sobre os limites das atuais técnicas de aprendizado de máquina em um ambiente não-estacionário foi publicado. O conceito de deslocamento de informação foi amplamente abordado em [129], enquanto o conceito de mudanças em funções geratrizes de sinais foi coberto por [130].

Em um sistema convencional de reconhecimento de padrões, o foco é, tipicamente, a extração de características que possam ser úteis no processo de tomada de decisão. Para modelos que não apresentam realimentação de entradas, como redes neurais *feedforward*, a saída é obtida supondo que as entradas não possuem relação temporal. Assim sendo, se a informação contida nos dados varia ao longo do tempo, tais modelos podem apresentar uma redução em sua eficiência de classificação. Além disso, o desenvolvimento de modelos baseados em técnicas de aprendizado de máquina, frequentemente, pressupõe que os dados utilizados representam toda a estatística do problema e, na maioria das vezes, tal suposição pode levar a

um modelo treinado de maneira equivocada. No processo de desenvolvimento de detectores de novidade para sonar passivo, todos estes problemas foram observados e, como a degradação da estacionaridade é um problema conhecido para dados provenientes de sistemas de sonar passivo, a análise de estacionaridade deve ser desenvolvida.

Como, frequentemente, a estacionaridade é uma suposição básica em muitos procedimentos de extração de características, os dados não-estacionários são, muitas vezes, processados para se tornarem estacionários. Geralmente, os sistemas que processam estes dados utilizam janelas curtas de tempo para que, ao se reduzir o tempo de observação, a quantidade de amostras também é reduzida e assim, e obtenha-se um processo aleatório estacionário dentro de cada uma das janelas de processamento, conhecida como estacionaridade intra-janela.

Um processo estacionário pode ser definido como um processo onde a função geratriz dos dados não muda ao longo do tempo. Por consequência, as características extraídas são geradas pela mesma função densidade de probabilidade. Assim sendo, uma vez que o processo estocástico é estacionário, os modelos treinados podem ser mantidos durante a etapa de operação. Caso o processo seja tipicamente não-estacionário, a estacionaridade deve ser monitorada para que assim que ocorra uma degradação significativa da mesma, a extração de características seja refeita neste novo cenário (novas condições de operação).

Uma das causas mais comuns da degradação da estacionaridade é a tendência da média, que pode ser devida à presença de uma raiz unitária ou uma tendência determinística, a ser removida através de um extrator de tendência em séries temporais [131].

A definição formal de estacionaridade pode ser vista na Equação 3.5, onde $F_{\mathbf{X}}$ é a função geratriz dos dados, t é um instante de tempo e τ é um intervalo de tempo que pode assumir qualquer valor. A definição de estacionaridade também é conhecida como estacionaridade no sentido estrito (*Strict Sense Stationarity* ou StSS).

$$F_{\mathbf{X}}(t) = F_{\mathbf{X}}(t + \tau) \quad (3.5)$$

A avaliação da estacionaridade pela sua definição é muito custosa e, devido a isso, foi criada uma formulação mais simples: a estacionaridade no sentido amplo (*Wide Sense Stationarity* ou WSS). Para um processo ser considerado WSS, este deve obedecer a Equação 3.6. Como descrito na Equação 3.6, para um processo ser classificado como WSS, o valor esperado da distribuição, representado por μ_x , não varia ao longo do tempo. Além do valor esperado, a função de covariância da distribuição, representada por C_{XX} , não depende do tempo e sim da diferença entre

os tempos $\tau = t_1 - t_2$. Esse fato tem impacto direto na função de autocorrelação R_{XX} , que também se torna dependente somente de um intervalo de tempo no caso da estacionaridade WSS, como é mostrado na Equação 3.6.

$$\begin{aligned} \mathbb{E}[X] = \mu_x(t) &= \mu_x(t + \tau), & \text{para todo } \tau, \\ C_{XX}(\tau) &= C_{XX}(t_1 - t_2), & R_{XX}(\tau) = R_{XX}(t_1 - t_2) \end{aligned} \tag{3.6}$$

Para se avaliar se uma série temporal apresenta estacionaridade, tanto no sentido amplo, quanto no sentido estrito, atualmente, existem alguns testes de estacionaridade na literatura. Geralmente, estes testes avaliam a presença de estacionaridade WSS, acessando momentos baixos da distribuição, como a média e a variância. Sendo assim, para variáveis aleatórias que são caracterizadas por momentos de ordens mais altas podem apresentar estacionaridade WSS e não apresentarem estacionaridade no sentido estrito. Assim sendo, outras propostas de testes de estacionaridade devem ser capazes de acessar HOS.

3.3.1 Testes de Raiz Unitária

Testes de raiz unitária verificam se uma variável de um processo estocástico apresenta estacionaridade ou não através da presença da chamada raiz unitária. A hipótese nula desse teste é, geralmente, definida como a ausência de estacionaridade e presença da raiz unitária; a hipótese alternativa é a presença da estacionaridade e ausência da raiz unitária.

Um processo estocástico pode ser representado na forma da Equação 3.7, onde $y(t)$ é uma amostra no tempo t , $D(t)$ representa o seu componente determinístico, que pode se apresentar como uma tendência ou comportamento sazonal (frequentemente, retirado através técnicas de processamento de séries temporais); $z(t)$ representa a parte estocástica da série, e $\varepsilon(t)$ é um erro estocástico da modelagem do processo.

$$y(t) = D(t) + z(t) + \varepsilon(t) \tag{3.7}$$

Com a extração da parte determinística $D(t)$, o modelo se torna $y(t) = z(t) + \varepsilon(t)$ e pode-se aplicar o chamado modelo auto-regressivo (AR ou *Autoregressive model*). Esse modelo representa um processo aleatório como uma composição linear dos valores anteriores do processo e um termo estocástico, que pode ser visto como o

erro de modelagem. Assim sendo, temos que a representação da parte estocástica do processo pode ser modelada no modelo AR de ordem p conforme a Equação 3.8. Onde ι_i correspondem aos parâmetros do modelo. O modelo AR pode interpretado como a saída de um filtro IIR, só polos, de ordem p que possui como entrada ruído-branco.

$$y(t) = \sum_{i=1}^p \iota_i y(t-i) + \varepsilon(t) \quad (3.8)$$

De posse desse modelo, alguns testes de estacionaridade foram desenvolvidos e, entre eles, podem ser citados: ADF, Phillips–Perron e KPSS todos partindo da mesma premissa: se as amostras anteriores do processo forem suficientes para modelar a amostra atual do modelo dentro de uma margem de confiança, então todas as amostras podem ter sido geradas pela mesma função geratriz, logo a hipótese da não-estacionaridade não pode ser rejeitada. Se, por meio do modelo AR, nenhum conjunto de parâmetros possa ser ajustado para as dadas amostras, então estas provavelmente não foram geradas pela mesma função e, conseqüentemente, rejeita-se tal hipótese.

O teste ADF ou Teste de Dickey–Fuller aumentado (*augmented Dickey–Fuller test*) utiliza um modelo AR de primeira ordem combinado com um teste de hipótese baseado na distribuição t de student paramétrica para avaliar a hipótese de estacionaridade. O teste Phillips–Perron (PP) também utiliza um modelo AR de primeira ordem, embora o teste de hipótese utilizado não seja a distribuição t de Student mas a distribuição não-paramétrica.

3.3.2 Teste de Divergência de Kullback-Leibler

A divergência de Kullback-Leibler (divergência KL) é uma medida, e não uma métrica, da dissimilaridade entre duas distribuições de probabilidade no espaço de funções de densidade de probabilidade [132]. A divergência KL é um caso particular de uma família de divergências chamadas divergências f (*f-divergences*) e também pode ser classificada como uma das chamadas divergências de Bregman (*Bregman divergences*). A definição da divergência KL pode ser vista na Equação 3.9 e como pode ser visto, $D_{KL}(p_x||p_y)$ pode ser diferente de $D_{KL}(p_y||p_x)$, o que faz com que a divergência não possa ser classificada como uma métrica.

$$D_{KL}(p_x||p_y) = \sum_{i=1}^N p_x(i) \log \left(\frac{p_x(i)}{p_y(i)} \right) \quad (3.9)$$

Para se estabelecer o valor da divergência KL, deve-se fazer a estimação da função densidade de probabilidade. A estimação de funções de densidade de probabilidade pode ser feita através de diversas técnicas, dentre elas a estimação de densidade por kernel, ou, KDE (kernel density estimation), que é uma estimativa não-paramétrica.

Suponha que uma variável aleatória X se desenvolva em n amostras, tal que $X = \{x_1, x_2, \dots, x_n\}$. Se os dados da variável aleatória foram gerados por uma função geratriz f , então a função estimada \hat{f} pode ser obtida pela Equação 3.10, onde K é um kernel arbitrário disponíveis e h é um parâmetro de suavização.

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (3.10)$$

Durante a aquisição dos dados, os mesmos desenvolvem uma função de densidade de probabilidade que pode ser estimada utilizando KDE. No caso de uma janela de aquisição obter uma PDF muito diferente de outras janelas de aquisição, a estacionaridade da janela em questão pode ter sido perdida. O teste estacionaridade de divergência KL se baseia no fato de que a estacionaridade deve ser manter entre as janelas de aquisição, caso isso não ocorra, a extração de características deve ser refeita. Assim sendo, a divergência entre janelas deve ser monitorada e, se o valor obtido ultrapassar um dado limiar, o teste rejeita a hipótese de estacionaridade.

Capítulo 4

Método

O capítulo a seguir descreve os métodos utilizados durante o trabalho. O método utilizado neste trabalho foi inspirado no trabalho exposto em [97].

Os experimentos foram conduzidos seguindo a Figura 4.1. Primeiramente, os sinais provenientes de uma marcação (depois da conformação de feixes e detecção do contato) são processados através da análise lofar. Os dados, já no domínio das frequências, podem ser pré-processados por um extrator de características (PCA, kPCA, NLPCA e PCD) ou não e assim alimentam modelos voltados para a classificação e a detecção de novidade.

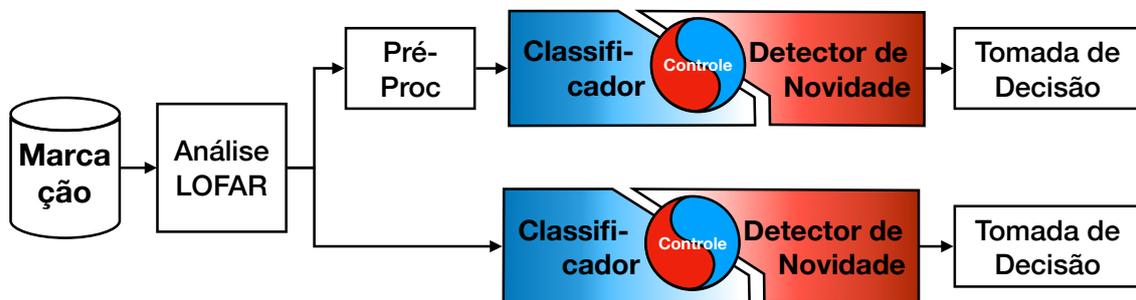


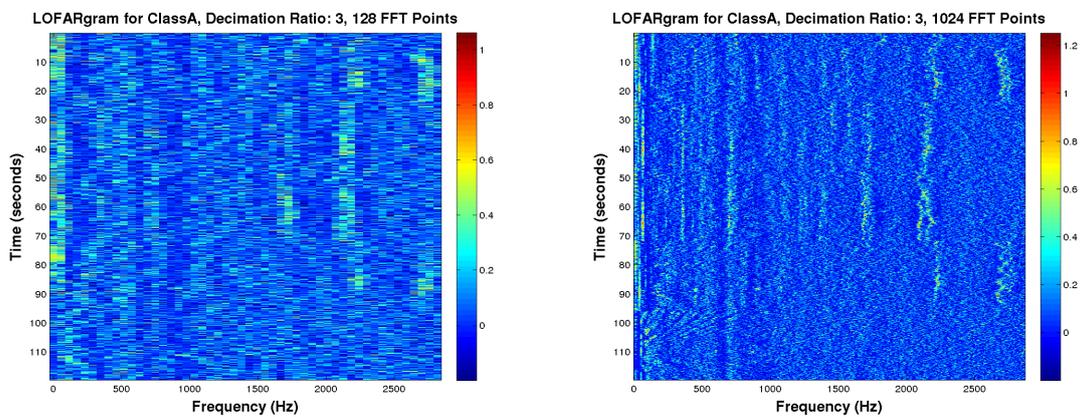
Figura 4.1: Método Proposto para Detecção de Novidade

Como pode ser visto na Figura 4.1, o sistema proposto pode operar tanto para a tarefa de detecção de novidade, quanto para a tarefa de classificação. Caso o operador do sistema tenha necessidade, pode haver a transição gradual entre detecção de novidade e classificação. Durante a transição, o sistema opera em uma composição entre as tarefas. Esta transição é controlada por uma variável do sistema que indica qual a operação será priorizada, ou seja, o percentual de cada uma das tarefas na composição. Este percentual pode ser totalmente composto de classificação ou de detecção de novidade.

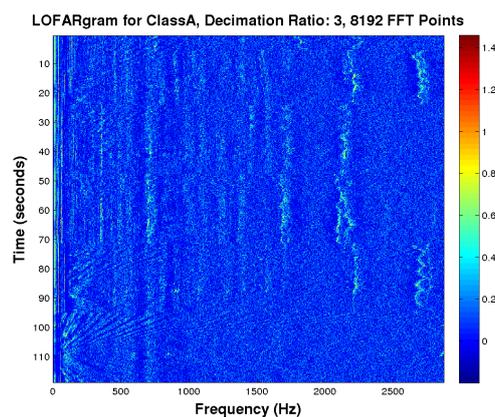
4.1 Processamento de Sinais

Como descrito anteriormente, a análise LOFAR foi o método de processamento utilizado para acessar as informações contidas nos sinais de áudio disponibilizados pela MB. Os parâmetros necessários para a aplicação da análise LOFAR são: o tamanho da janela de processamento, o índice de decimação e os parâmetros da estimação do ruído de fundo (TPSW).

O tamanho da janela de processamento tem um impacto direto sobre a estacionaridade dos sinais processados. Uma das estratégias para atacar o problema da não-estacionaridade é reduzir o tamanho da janela de processamento. Quanto maior é a largura de cada janela, maior a probabilidade de perda de estacionaridade. Na Figura 4.2 são ilustrados os efeitos do uso de diferentes larguras de janelas de processamento a análise LOFAR.



(a) 128 pontos em cada janela de processamento (b) 1024 pontos em cada janela de processamento



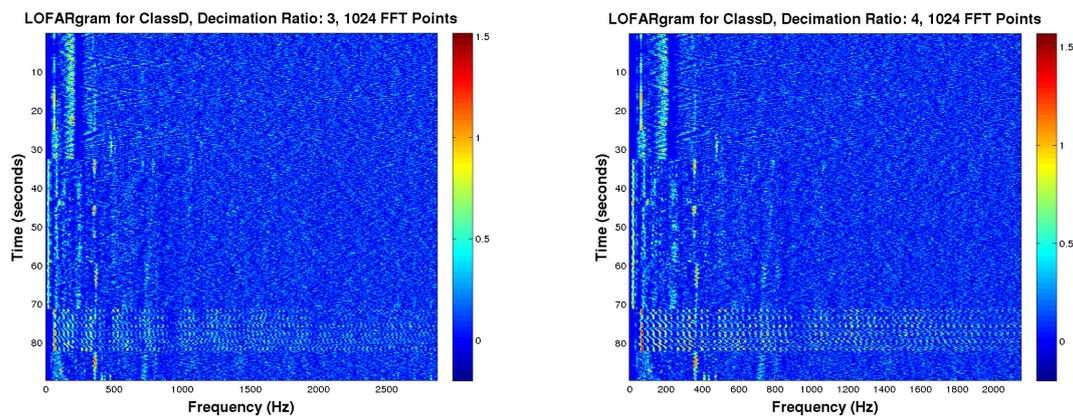
(c) 8192 pontos em cada janela de processamento

Figura 4.2: LOFARgrama para diferentes quantidade de pontos em cada uma das janelas de processamento

Como pode ser visto nestas figuras, a quantidade de pontos processados é de

fundamental importância. Na Figura 4.2a, algumas estrias, tanto para baixas quanto para altas-frequências, se espalham por mais de um valor de frequência discreta. Este efeito fica mais claro quando comparamos com as Figuras 4.2a e 4.2b, onde as estrias previamente mencionadas se mostram melhor definidas na última. O efeito oposto também pode ser observado quando comparamos as Figuras 4.2b e 4.2c. Aqui, observa-se uma grande quantidade de valores de frequência que não possuem energia significativa quando comparados aos seus pares mais energéticos. Este efeito pode gerar dimensões que não são ativadas durante o processamento dos sinais, gerando assim dados de entrada que não são relevantes para os modelos. Quanto maior a quantidade de pontos processados, maior é quantidade de dimensões de entrada dos modelos.

Além do tamanho da janela de processamento, outro parâmetro da análise lofar que pode variar é o índice de decimação. A decimação foi projetada para fazer a seleção da banda de interesse dos sistemas de sonar passivo. Sinais de sonar passivo têm sua informação de interesse concentrada em frequências mais baixas [97], portanto, uma decimação de ordem baixa pode ser utilizada na análise LOFAR de maneira efetiva.



(a) Índice de Decimação de 3

(b) Índice de Decimação de 4

Figura 4.3: LOFARgrama para diferentes índices de decimação

Na Figura 4.3 são mostrados dois exemplos de valores de decimação que são utilizados pela MB. Numa comparação entre as Figuras 4.3a e 4.3b, pode-se observar que a diferença entre os espectros obtidos é relativamente pequena, o que justifica a utilização de qualquer um dos valores para classificação.

Por fim, depois da extração do módulo do sinal, o algoritmo TPSW é aplicado para a redução da amplitude do ruído de fundo e tem sido utilizado amplamente nos sistemas da MB [31, 51, 52, 97, 114]. Este algoritmo estima o ruído de fundo aplicando uma janela específica a cada um dos *bins* encontrados no espectro estimado. A janela utilizada neste trabalho está exemplificada na Figura 4.4 (com largura de

fenda central de 5 *bins* e largura de fenda lateral de 17 *bins*). Os parâmetros utilizados neste trabalho seguiram os aplicados em trabalhos anteriores e também por sistemas atuais da MB.

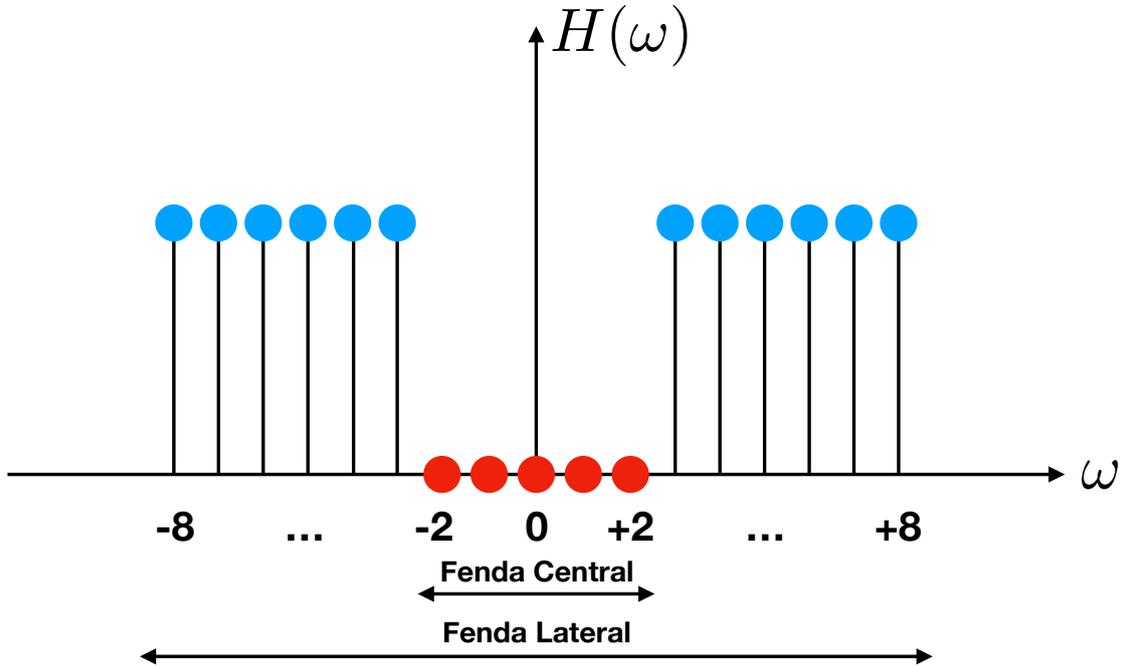


Figura 4.4: Janela utilizada para cálculo da média local do algoritmo TPSW

4.2 Pré-processamento dos dados

Como dados de sonar passivo foram processados pela análise LOFAR e, após este processamento, encontram-se no domínio das frequências, geralmente, se faz uma compactação para a redução da dimensionalidade dos mesmos. Uma vez que os dados estejam compactados, estes podem ser usados para alimentar modelos de aprendizado de máquina. Alguns tipos de pré-processamento visam diminuir a dimensionalidade dos dados sem perda de informação. Para isso, a compressão de dados é, na maioria dos casos, feita acessando HOS.

Dentre os métodos de compactação utilizados durante o período de desenvolvimento da tese, PCA, kPCA, NLPCA, PCD podem ser citados. No caso do pré-processamento com PCA, os únicos parâmetros envolvidos são o número de componentes extraídos e a forma de extração dos componentes. O primeiro foi determinado através do percentual de energia retido em cada um dos componentes, e o segundo, assumiu uma extração paralela de componentes.

Já no caso do pré-processamento baseado em kPCA (detalhado no apêndice B), diferentes funções de kernel podem ser utilizadas para o mapeamento dos dados. As

funções kernel visam mapear (implicitamente) cada um dos pontos em um espaço de alta dimensão, gerado a partir da dissimilaridade entre cada um dos exemplos. Neste trabalho, a função kernel utilizada foi o kernel gaussiano (Equação 4.1)

$$K(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}} \quad (4.1)$$

Esta função foi escolhida pois, primeiramente é a mais utilizada entre as funções kernel, e em segundo lugar, por possui algumas aproximações que tornam o seu emprego menos custoso computacionalmente. Como SVM e outros modelos similares, que se baseam em funções kernel para estimarem a dissimilaridade entre os dados de treinamento, não escalam bem para grandes bancos de dados ou entradas com grandes dimensões, empregou-se o método de Nyström [133] para aproximar a composição da matriz de Gram K , o que reduz sensivelmente a complexidade do mapeamento do kernel. Para se obter o valor mais adequado para largura de banda do kernel, um processo de busca exaustiva (*greedy search*) foi implementado.

Como o kernel gaussiano foi utilizado para diferentes processos, a sua largura de banda pode ser definida para um dos métodos e o seu valor pode ser compartilhado para os demais. Assim sendo, o método escolhido para a escolha da largura de kernel foi a estimação da função densidade de probabilidade dos dados conhecidos. O processo consiste em estimar a PDF das classes conhecidas em um conjunto de treinamento (previamente definido através dos sorteios realizados pelo método de validação cruzada) de tal maneira que se maximize a função de verossimilhança avaliada considerando os dados de treinamento (extração de parâmetros) e validação. Caso a função de verossimilhança estime que os dados de teste não obtiveram um bom resultados para os parâmetros estimados no conjunto de treinamento, a largura de kernel é atualiza e o processo se repete até que o valor do parâmetro maximize a função de verossimilhança, ou se aproxime do ponto de máximo. Como critério de parada da busca exaustiva, o limite de 0,01 foi imposto.

Outro parâmetro importante para a utilização da kPCA como pré-processamento é a quantidade de componentes reconstruídas depois da compactação. O kPCA concentra-se principalmente no mapeamento dos dados através da sua dissimilaridade (função kernel), ou seja, eventos da mesma classe (menor dissimilaridade) são projetados em pontos mais próximos no espaço transformado. Como os dados projetados têm uma dimensionalidade consideravelmente alta, ou potencialmente infinita, as dimensões de baixa energia podem ser interpretadas como dimensões de ruído (dimensões sem dados de sinal) e o conceito de pré-imagem surge do fato de que, se os componentes de ruído forem removidos, os dados reconstruídos da kPCA têm menos ruído do que dados originais, ou seja, os dados reconstruídos são uma aproximação dos dados antes da corrupção de ruído [134]. Assim sendo, dimensões

que apresentaram uma baixa energia no espaço transformado pela função kernel foram retiradas depois da extração dos componentes principais por kernel.

Além da redução de dimensões pela técnica de pré-imagem, a quantidade de componentes que foram utilizadas para o treinamento dos modelos foi estimada com uma técnica indireta. Os componentes foram inseridos um a um em um modelo neural de classificação para a estimação da eficiência de classificação. Primeiramente, foi extraído um componente principal por kernel, após a sua extração, uma rede neural artificial foi treinada para a classificação das classes conhecidas e a sua eficiência de classificação foi medida através de figuras de mérito de classificação. Após esta etapa, o segundo componente foi adicionado ao vector de entradas e outro processo de treinamento foi inicializado, e este processo se estendeu por todos os componentes que não foram retirados pelo método da pré-imagem. À medida que a quantidade de componentes da kPCA que foram introduzidos ao modelo neural de classificação aumenta, maior a quantidade de informação que o classificador tem acesso. A partir do momento em que a adição de um componente não impacta positivamente na eficiência de classificação, número de componentes ótimos é obtido.

Na extração da NLPCA, o número de neurônios nas camadas de codificação e decodificação tem papel fundamental. Como a rede neural auto-associativa possui muitos parâmetros devido a sua estrutura, o treinamento do mesmo pode ser muito custoso. Para a aplicação deste pré-processamento, que acessa informações não-lineares dos dados como a kPCA, foi implementado um processo experimental de estimação da quantidade de neurônios nas camadas de codificação e decodificação, que neste caso terão o mesmo número de neurônios. O processo se inicia com um único neurônio nas camadas em questão e 10 componentes sendo extraídos, o modelo é treinado e estimado o erro de reconstrução associado a 1 neurônio fazendo a codificação dos dados de entrada. A partir daí, outro neurônio é adicionado na camada de codificação (e em sua contra-parte) e o erro associado é estimado. Os neurônios vão sendo adicionados até que ocorra a estabilização na redução no erro de reconstrução do modelo e, neste tem-se a melhor topologia para a codificação e decodificação.

Além da topologia das camadas intermediárias do modelo de extração, o número de componentes extraídos deve ser estimado. Uma vez que a camada de codificação e sua contra-parte estão definidas, o mesmo processo é repetido variando-se o número de componentes a serem extraídos.

No caso das PCD, em ambas as extrações, outro processo experimental foi implementado, desta vez visando maximizar a capacidade de classificação dos dados conhecidos por partes dos componentes extraídos. Para cada uma das extrações, tanto por deflação quanto cooperativamente, o número de componentes extraídos foi aumentado gradativamente buscando se obter o ponto onde a máxima eficiência

de classificação foi alcançada. Neste ponto, não há incremento na eficiência de classificação caso um outro componente seja extraído.

4.3 Detectores de Novidade

Com o intuito de acessar possíveis flutuações estatísticas no conjunto de treinamento, cada uma das classes disponíveis no banco de dados será tratada como classe de novidade por vez. Ou seja, uma das classes será selecionada como novidade e todas as outras classes serão apresentadas para os modelos de extração de características, treinamento dos classificadores e dos detectores de novidade.

Neste trabalho foram utilizados dois tipos de detectores de novidade, ambos baseados em domínio. Nesta sessão, os parâmetros utilizados para o treinamento e operação de cada um deles serão detalhados, bem como as figuras de mérito utilizadas para a avaliação de cada um dos modelos de aprendizado de máquina.

4.3.1 One-Class SVM

Os detectores de novidade baseados em SVM foram treinados com o kernel gaussiano, que foi escolhido pelos mesmos motivos já discutidos anteriormente para kPCA. Além disso, o valor do parâmetro de largura de banda do kernel foi o mesmo utilizado para a pré-processamento kPCA, uma vez que os dados utilizados foram os mesmos.

Para o processo de treinamento, foram desenvolvidos um detector de novidade e um classificador para cada uma das classes conhecidas. Todos os modelos foram desenvolvidos com base nas SVM de uma classe, sendo que os modelos de classificação foram treinados para encapsular os dados de uma classe conhecida contra todas as outras classes conhecidas. Assim, os classificadores de classes conhecidas podem ser denominados classe-especialistas. Já no caso do detector de novidade,

Ambos os modelos, de classificação e de detecção de novidade foram treinados com diferentes valores do parâmetro ν . Este parâmetro controla a generalização do modelo treinado, bem como a quantidade de vetores suporte. Se o valor de ν for próximo de zero, os classificadores e detectores de novidade treinados serão mais generalistas e a quantidade de vetores suporte será baixa. Caso contrário, com ν tendendo a um, os modelos serão menos generalistas gerando uma hiper-superfície de decisão com mais pontos a suportando. O ponto de operação do sistema de classificação/detecção de novidade será definido através dos valores do parâmetro ν .

4.3.2 Redes Neurais

Para redes neurais, um processo inspirado em [97] foi implementado. Depois do treinamento, que visava a maximização da eficiência de classificação das classes conhecidas, um limiar de classificação foi utilizado. Este limiar é aplicado aos neurônios da camada de saída como descrito anteriormente.

Como os *Stacked AutoEncoders* podem ser vistos como redes neurais artificiais com muitas camadas escondidas, o mesmo modelo de limiar de detecção de novidade pode ser aplicado a estes. Além disso, dados compactados por camadas intermediárias dos *Stacked AutoEncoders* serão usados para treinar e operar detectores de novidade baseados em SVM.

4.3.3 Figuras de Mérito utilizadas

Diversas medidas de desempenho são utilizadas para avaliar a eficiência de métodos de classificação. Para a detecção de novidade, as medidas clássicas de classificação devem sofrer algumas modificações para serem aplicadas. A eficiência das técnicas de detecção de novidade podem ser encaradas como medidas de classificação para duas classes, uma vez que podemos visualizar o problema de detecção de novidade como sendo um classificador simples de classe vs. não-classe (onde a classe é a classe normal e a não-classe é a classe novidade ou anormal).

Assim, a eficiência na detecção da classe desconhecida deve ser avaliada. Essa eficiência nada mais é do que a quantidade de eventos da classe desconhecida que foram classificados corretamente:

$$Eff_{NoveltyDetection} = 100\% \frac{\# \text{Dados de novidade classificados como novidade}}{\# \text{Total de Dados de novidade}} \quad (4.2)$$

Temos também a eficiência das classes conhecidas (equação 4.3). Quando classificadores são treinados sem o viés de detecção de novidade, temos a máxima eficiência de classificação nas classes conhecidas. Uma vez que a detecção de novidade é inserida e analisada, pode-se deteriorar a classificação de classes conhecidas, tendo em vista um aumento na eficiência da detecção da classe desconhecida. Assim sendo, deve-se analisar a classificação das classes conhecidas e o impacto do aumento da eficiência da detecção de novidade na classificação das classes conhecidas.

$$Eff_{Known\ Class_i} = 100\% \frac{\# \text{Dados da classe i classificados como classe i}}{\# \text{Total de Dados de da classe i}} \quad (4.3)$$

Outra figura de mérito que pode ser avaliada é o índice SP [135]. Esse índice foi criado para avaliar de maneira equilibrada a eficiência de classificadores, sendo bastante útil quando a aplicação envolve mais de duas classes. Como pode ser visto na equação 4.4, se, pelo menos, uma das classes obtiver uma baixa eficiência, o índice SP tende a colapsar. Se todas as classes obtiverem eficiências altas, o índice SP assumirá um valor próximo de 1. No caso da detecção de novidade, o índice SP será avaliado para as classes conhecidas.

$$SP = \sqrt[2]{\left(\frac{\sum_{i=1}^{N_{class}} Eff_i}{N_{class}}\right) \cdot \left(N_{class} \sqrt{\prod_{i=1}^{N_{class}} Eff_i}\right)} \quad (4.4)$$

Uma outra figura de mérito que pode ser definida é a acurácia das classes conhecidas, que é definida pela porcentagem de eventos conhecidos que foram corretamente classificados. Essa medida de desempenho é amplamente utilizada na avaliação de classificadores e visa avaliar a eficiência total de classificação das classes conhecidas pelo classificador.

Como a detecção de novidade pode ser vista como uma classificação classe vs não-classe, pode-se avaliar a sensibilidade do detector de novidade a eventos conhecidos. Uma das principais contribuições desta tese é a definição desta medida de desempenho. Como, geralmente, a tarefa de detecção de novidade se dá um ambiente com muitas classes, o impacto de um detector de novas classes, operando em conjunto com um sistema de classificação, pode ser observado de maneiras diferentes para cada uma das classes conhecidas. Assim sendo, uma nova medida de desempenho foi desenvolvido para simplificar o processo de desenvolvimento dos modelos treinados.

Esta medida tem por objetivo avaliar o impacto da inserção de um detector de novidade no ambiente de classificação. Como outras medidas, específicas de classificação foram avaliadas, esta figura de mérito visa apenas estimar o impacto da leitura e processamento de novas classes ao ambiente. Assim sendo, a figura de mérito *trigger* pode ser definida como o percentual de eventos conhecidos que foram classificados como conhecidos. Assim sendo, se um evento de uma classe conhecida i for classificado como pertencente a uma classe conhecida j , tal que $i \neq j$, o valor do *trigger* irá sofrer uma variação positiva, pois o evento foi classificado como um evento conhecido.

Ou seja, através da definição desta medida, o desenvolvimento de um detector de novidade, que antes poderia ser uma tarefa que envolvia diversas eficiências de classificação ou medidas de desempenho diversas, como a acurácia de classificação e o índice SP, agora, se resume a medida do *trigger*, que resume a eficiência de classificação de classes conhecidas em um único valor.

4.4 Análise de Estacionaridade

A análise de estacionaridade foi utilizada para acessar informações estatísticas que possam ser úteis no processo de classificação. Primeiramente, testes de raiz unitária foram utilizados para a tomada de decisão do tamanho ideal da janela de processamento da análise lofar. Os testes de raiz unitária por acessarem informações de baixa ordem podem ser melhorados com testes que acessem estatística de ordem superior, como por exemplo a divergência KL (descrita anteriormente).

Os testes de raiz unitária apresentados serão aplicados a dados com diferentes tamanho de janela e a rejeição a hipótese de estacionaridade será usada como resultado do teste. O grau de confiança do teste será descartado uma vez que se trata de um estudo preliminar de como a estacionaridade impacta na classificação.

O teste com a divergência KL foi para estimar a perda de estacionaridade entre janelas contíguas, ou seja, caso o valor de divergência obtido entre duas janelas de processamento seja acima de um limiar, isto pode indicar perda de estacionaridade.

As amostras, neste caso, serão divididas em conjunto de desenvolvimento e teste. Do conjunto de desenvolvimento serão extraídas informações como o limiar para a tomada a rejeição da hipótese de estacionaridade para a divergência KL.

Ambos os testes de estacionaridade são aplicados no domínio do tempo e podem ser utilizados também como uma forma de detectar exemplos de novas classes, ou exemplos que sejam gerados por uma PDF diferente do que foi apresentado durante o processo de treinamento. Assim sendo, com estes testes, a detecção de novidade probabilística pode ser realizada.

Uma outra aplicação da análise de estacionaridade é a seleção de eventos de treinamento. A extração de características, idealmente, deve ser refeita ou atualizada sempre que estacionaridade é perdida no conjunto de entrada. Em um ambiente de operação, o custo de retreinamento deve ser levado em consideração, mas em um ambiente de desenvolvimento *offline*, o processo de extração de características pode ser realizado de maneira eficiente. Assim sendo, a análise de estacionaridade pode ser utilizada como seletor de eventos para o conjunto de treinamento, teste e validação.

Capítulo 5

Resultados Experimentais

Neste capítulo, serão expostos os resultados experimentais obtidos durante o desenvolvimento do trabalho. Primeiramente serão expostas as características dos dados experimentais disponibilizados pela MB e utilizados para a obtenção dos resultados. Posteriormente, a seleção de parâmetros para a extração dos pré-processadores será discutida. A seguir, os resultados obtidos tendo como base a detecção de novidade com diferentes classificadores alimentados com diferentes pré-processamentos são apresentados. Uma vez que o desenvolvimento dos detectores de novidade for concluído, aos algoritmos de *deep learning* serão comparados com os melhores modelos de detecção de novidade e, por fim, uma discussão dos resultados será feita ao final do capítulo.

5.1 Dados Utilizados

Os dados deste trabalho correspondem a ruídos irradiados por diferentes navios que foram adquiridos pela MB. O conjunto de dados provem de 4 classes de navios que foram adquiridas na raia acústica de Arraial do Cabo. Os experimentos na raia acústica consistem em navios correndo em um ambiente controlado pela MB com 1.500 metros de comprimento.

Geralmente, as gravações são feitas com um único hidrofone, que é acoplado a um sistema de aquisição que possui frequência de amostragem de 22.050 Hz e 8 bits de resolução. A gravação do experimento é feita quando o navio se encontra a 1.000 metros antes da posição do hidrofone e termina quando o navio se encontra a 500 metros depois da posição do hidrofone. A profundidade do hidrofone é de 45 metros.

Cada uma das classes foi adquirida em diferentes condições de mar e de máquina. As classes B e D totalizaram 10 corridas diferentes em cada uma, enquanto as classes A e C têm 4 e 8 corridas, respectivamente. Os dados de cada uma das corridas foram disponibilizados pela MB em arquivos de áudio separados.

5.1.1 Processamento de Sinais

O processamento dos sinais obtidos segue o método descrito na seção 2.4 do capítulo 2. A análise LOFAR (Figura 5.1) foi aplicada com o conjunto de parâmetros utilizados pela MB anteriormente, isto é, decimação por 3 (fazendo com que o espectro de análise se reduza de 11.025 Hz para 3.675 Hz), seguido de um filtro passa-baixas digital de ordem 8, que foi desenvolvido com frequência de corte em 2.870 Hz e atenuação de 60 dB para a frequência de 4.480 Hz, fato que evita o *aliasing* dos sinais [31]. Uma transformada de Fourier de tempo curto (STFT) foi aplicada para a extração dos espectros. Com o sinal foi filtrado, alguns bins de frequência discreta estimados pela transformada de Fourier não são preenchidos com sinal, devido a isso e a estudos anteriores, 80% da quantidade total de bins são utilizados para a análise, totalizando 400 bins de frequência discreta. Uma vez obtidos os espectros, a plotagem da potência em cada um dos bins de frequência discreta ao longo do tempo em formato de *waterfall display* pode ser realizada e a esse gráfico dá-se o nome de Lofargrama.

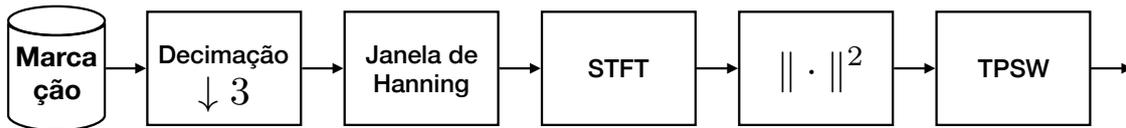


Figura 5.1: Diagrama em blocos da Análise LOFAR com parâmetros específicos

Para se definir a quantidade de pontos na janela de processamento, foi realizado um processo de treinamento supervisionado baseado em redes neurais artificiais. Para uma determinada quantidade de pontos processados em cada uma das janelas de processamento, os dados de saída da análise LOFAR alimentaram o treinamento de um modelo neural de classificação. As redes neurais de classificação foram treinadas com 10 neurônios na camada escondida e com 3 neurônios na camada de saída (um neurônio para cada classe conhecida). Os resultados obtidos neste treinamento estão expostos na Figura 5.2. Além dos resultados de classificação, aqui obtidos estimando o valor do índice SP, também são mostrados os resultados para cada um dos testes de raiz unitária aplicados, a saber: ADF e PP.

Os resultados expostos na Figura 5.2 foram avaliados com a quantidade de pontos por janela variando em potências de 2 com os seguintes valores: 128, 256, 512, 1.024, 2.048, 4.096 e 8.192. A figura de mérito utilizada para a avaliação do treinamento foi o índice SP, que teve o seu valor monitorado para o conjunto de validação (conjunto que não é exposto ao modelo durante o processo de treinamento) e além do índice SP, os testes de estacionaridade ADF e PP foram medidos em cada uma das janelas.

O treinamento foi realizado utilizando 10 *folds* com 50 inicializações e para cada *fold*, apenas o modelo que obteve o melhor índice SP foi mantido. Como pode ser

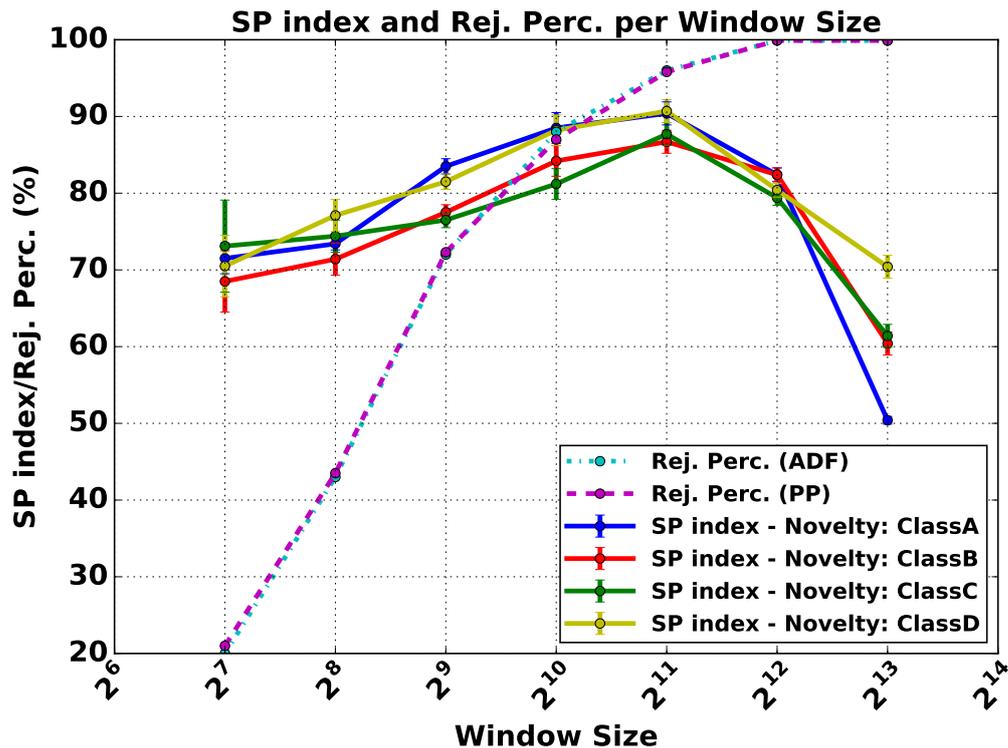


Figura 5.2: Eficiência de classificação do treinamento neural para classificação e percentual de rejeição ao modelo estacionário para diferentes tamanhos de janela de aquisição de dados

visto na figura, para janelas com menos pontos, a probabilidade de rejeição obtida em ambos os testes de estacionaridade foi relativamente pequena. Não obstante, a eficiência de classificação para as classes conhecidas aumenta para modelos treinados com janelas maiores, embora esse comportamento mude quando as janelas de processamento têm mais de 2.048 pontos. A perda de estacionaridade pode explicar a perda de eficiência para janelas maiores, pois os parâmetros extraídos foram treinados com dados gerados por uma função geratriz diferente dos dados de validação, o que faz com que a eficiência no conjunto de validação diminua.

Considerando os resultados obtidos, a maximização da eficiência de classificação ocorre para janelas com 2^{11} e 2^{10} pontos, considerando todas as classes como novidade e a barra de erro obtida com a validação cruzada. Como as janelas de 2^{10} pontos são utilizadas por sistemas de MB e possuem menor potencial de rejeição nos teste de raiz unitária, este foi o valor escolhido para o treinamento dos modelos.

Uma vez que o tamanho ótimo para a janela de processamento foi escolhido, a cadeia completa de processamento pode ser vista abaixo (Figura 5.1).

Uma análise detalhada do Lofargrama, em conjunto com técnicas de identificação do sinal de áudio, são as principais ferramentas para classificação de sinais em ambiente de sonar passivo. As informações extraídas de um Lofargrama podem variar de

uma forma extensa, por exemplo: pode-se extrair do Lofargrama se um navio possui engrenagem redutora entre o seu motor e o hélice. Um outro exemplo: o Lofargrama pode proporcionar informações sobre quais equipamentos elétricos estão ligados no navio e qual seu consumo instantâneo.

Para avaliar se o processo de aquisição de sinais com a janela proposta é estacionário, foi proposto um novo teste baseado na divergência KL. Neste teste, duas janelas consecutivas de sinal são avaliadas e caso o valor da divergência obtido seja maior do que um limiar, a hipótese de estacionaridade é rejeitada. Na Figuras 5.3 são apresentadas as distribuições das janelas de sinais processadas ao longo do tempo para cada uma das classes disponíveis para o treinamento. Como pode ser observado, as distribuições obtidas são diferentes entre si, até mesmo em janelas da mesma classe. Como as distribuições são diferentes, pode-se inferir que a função geratriz de cada uma delas é diferente também. Este fato comprova a perda de estacionaridade e, com isso, faz-se necessário o retreinamento de cada um dos modelos de classificação.

A Figura 5.4 apresenta os resultados da divergência KL para duas janelas consecutivas de sinais processados para cada uma das classes disponíveis. Nesta figura, as linhas verticais pontilhadas representam o limite em cada uma das corridas das classes disponíveis. Como a divergência não apresenta comportamento simétrico, os dois sentidos de avaliação foram utilizados. O sentido direto representa o resultado para a divergência apresentada com a janela anterior sendo p_x e a janela atual sendo p_y , enquanto que o sentido reverso representa o resultado da divergência com a janela anterior sendo p_y e a janela atual sendo p_x . Uma proposta de limiar de decisão foi a média (linhas pontilhadas azul e vermelha) e de posse deste limiar, os resultados apontam para uma maior chance de perda de estacionaridade no final das corridas adquiridas, ou seja, quando o navio encontra-se mais perto do hidrofone que fez a aquisição. Isto pode ser explicado pelo comportamento aleatório da explosão das bolhas de ar que se formam com a passagem do navio. Esse comportamento, em grandes distâncias não rege o comportamento do sinal adquirido, embora a pequenas distâncias, esse ruído seja dominante.

5.2 Pré-processamentos

Todos os resultados obtidos para os métodos de pré-processamento foram avaliados com a validação cruzada. Como este método pressupõe que cada um dos conjuntos é independente de todos os outros, pode-se aproximar a incerteza associada ao processo de estimação como sendo a soma das variâncias de cada um dos conjuntos envolvidos na validação cruzada. Neste trabalho, foram utilizados 10 *folds* para a extração de características.

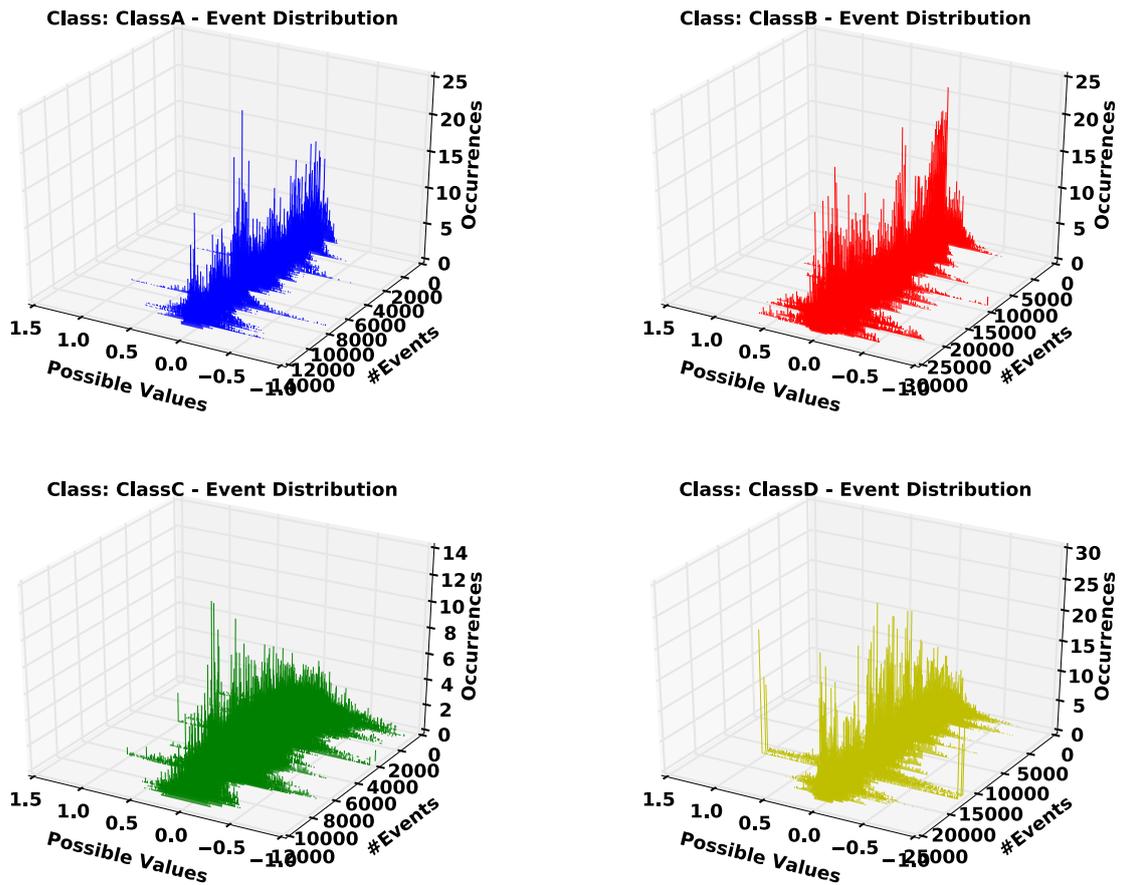


Figura 5.3: Distribuições dos sinais no domínio do tempo obtidos para janela definida para obtenção da menor probabilidade de perda de estacionaridade no sentido amplo. A linha em azul (esquerda superior) representa as distribuições obtidos para eventos que pertencem a Classe A. Já o *plot* em vermelho (direita superior) representa as distribuições para eventos pertencentes a Classe B. A linha em verde (esquerda inferior) apresenta as distribuições para eventos pertencentes a Classe C e, por fim, o gráfico em amarelo (direita inferior) representa as distribuições obtidas para eventos pertencentes a Classe D. As distribuições não foram normalizadas, sendo que os seus valores representam os sinais sonoros obtidos pelo hidrofone durante o período de aquisição de dados para cada uma das classes

Para o caso do método de pré-processamento PCA, a Figura 5.5 mostra a energia percentual acumulada a cada um dos componentes é apresentada. Como a análise de componentes principais gera componentes ordenados pela quantidade de energia, vemos que o componente de maior energia da base PCA gerada com as classes B, C e D como conhecidas (classe A como novidade), obteve 1,8% da energia total e o segundo componente de maior energia possui 1,4% da energia total e esse comportamento pode ser visto em cada uma das outras 3 classes sendo tratada como novidade. Aqui pode ser visto que para acumular 90% da energia com a classe A como novidade, são necessários aproximadamente 290 componentes (com ponto em vermelho) e esse comportamento se repete para as outras classes como novidade.

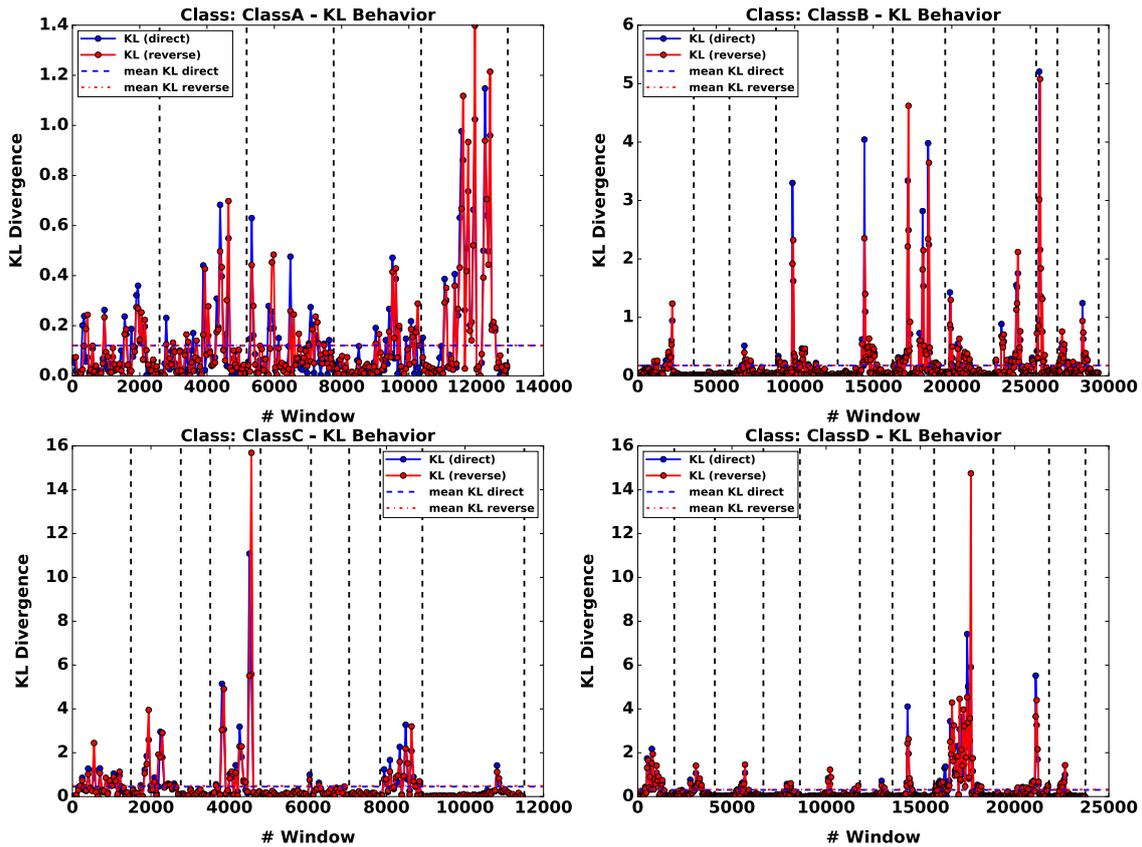


Figura 5.4: Resultados obtidos com a divergência KL para janelas consecutivas de sinais, no sentido direto e no sentido reverso, quando a mesma é aplicada no domínio do tempo. Na parte esquerda superior, os valores da divergência KL para eventos que pertencem a Classe A são apresentados. Já a parte direita superior, representa os valores de divergência KL para eventos pertencentes a Classe B. A parte esquerda inferior apresenta os valores de divergência KL para eventos pertencentes a Classe C e, por fim, os gráficos apresentados na parte direita inferior representam os valores para a divergência KL obtidos para eventos pertencentes a Classe D. As distribuições foram normalizadas (área embaixo da curva igual a um) para ser a estimativa da PDF dos eventos.

No geral, a variabilidade é pequena com a classe que é tratada como novidade.

A Figura 5.6 apresenta a variação das eficiências de classificação pela variação do número de componentes principais por kernel extraídos para diferentes classes sendo tratadas como novidades. Como uma das classes disponíveis durante o treinamento foi tomada como novidade, são apresentados 4 *plots* diferentes. Um fato que vale a pena ser observado é que a classe C, apresentada em verde, sempre obteve uma eficiência de classificação menor do que as outras classes conhecidas e, quando a mesma foi tratada como novidade, a eficiência de classificação da classe B (vermelha) apresentou uma diminuição, mostrando assim uma relação não direta entre essas duas classes.

A eficiência de classificação tem um comportamento como esperado na teoria, e,

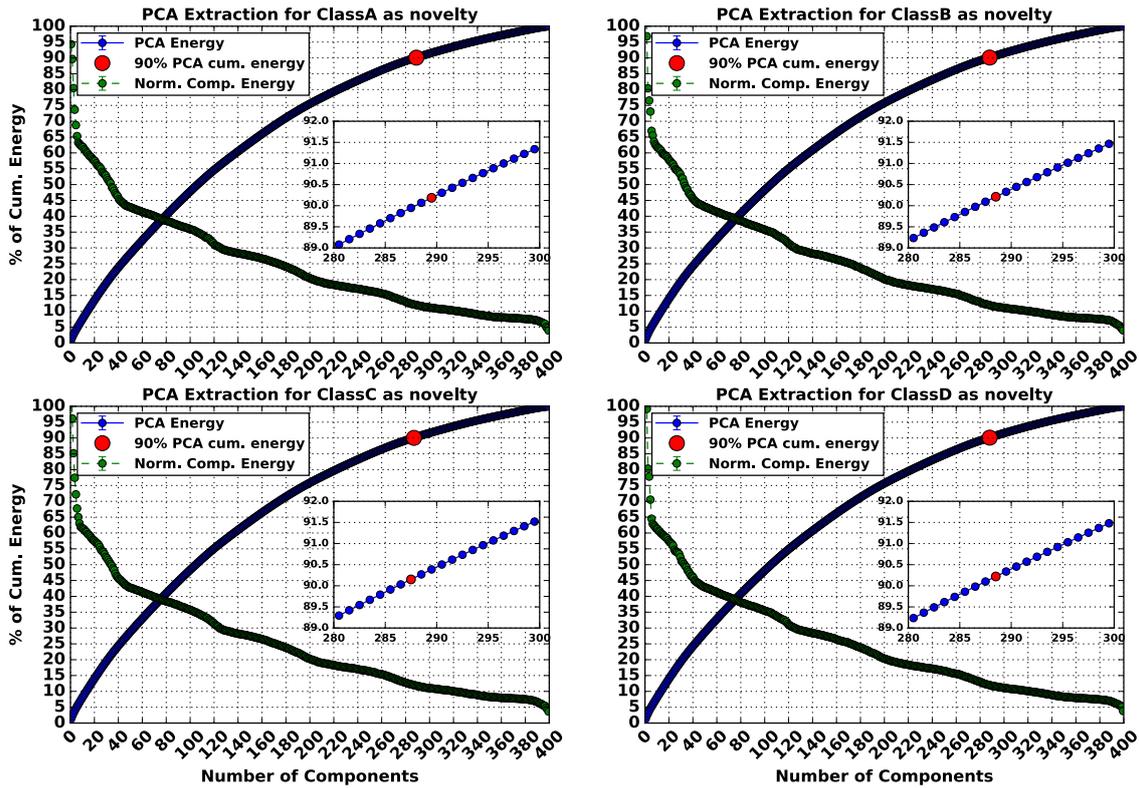


Figura 5.5: Resultados para a energia acumulada e erro de reconstrução dos componentes principais. Cada gráfico representa o resultado para uma classe sendo tratada como novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, a quantidade de energia para a faixa de componentes que acumulam 90% da soma de energia total.

apresenta um crescimento a medida que o número de componentes aumenta. Além disso, na Figura 5.6, mostra-se que, se as eficiências de classificação das classes conhecidas, bem como os valores para o índice SP, se estabilizam quando o número de componentes alcança em torno de 35 componentes, dentro da variação obtida pelo processo de validação cruzada e da variabilidade da classe de novidade.

O treinamento da NLPCA se baseia em uma estrutura específica de rede neural que visa a representação dos dados com altos níveis de abstração, isto é, a cada camada, o modelo compacta os dados de entrada e faz com que o erro de reconstrução seja mínimo para a representação dos dados. Para isso, o treinamento do extrator utilizou a regularização L_2 (norma dois, com parâmetro 0,25) visando evitar que o mesmo apresentasse o processo de *overtraining*. Como o modelo de extração possui uma grande quantidade de parâmetros, a probabilidade do mesmo apresentar um processo de *overtraining* é maior. Para a estimativa de quantos neurônios são ne-

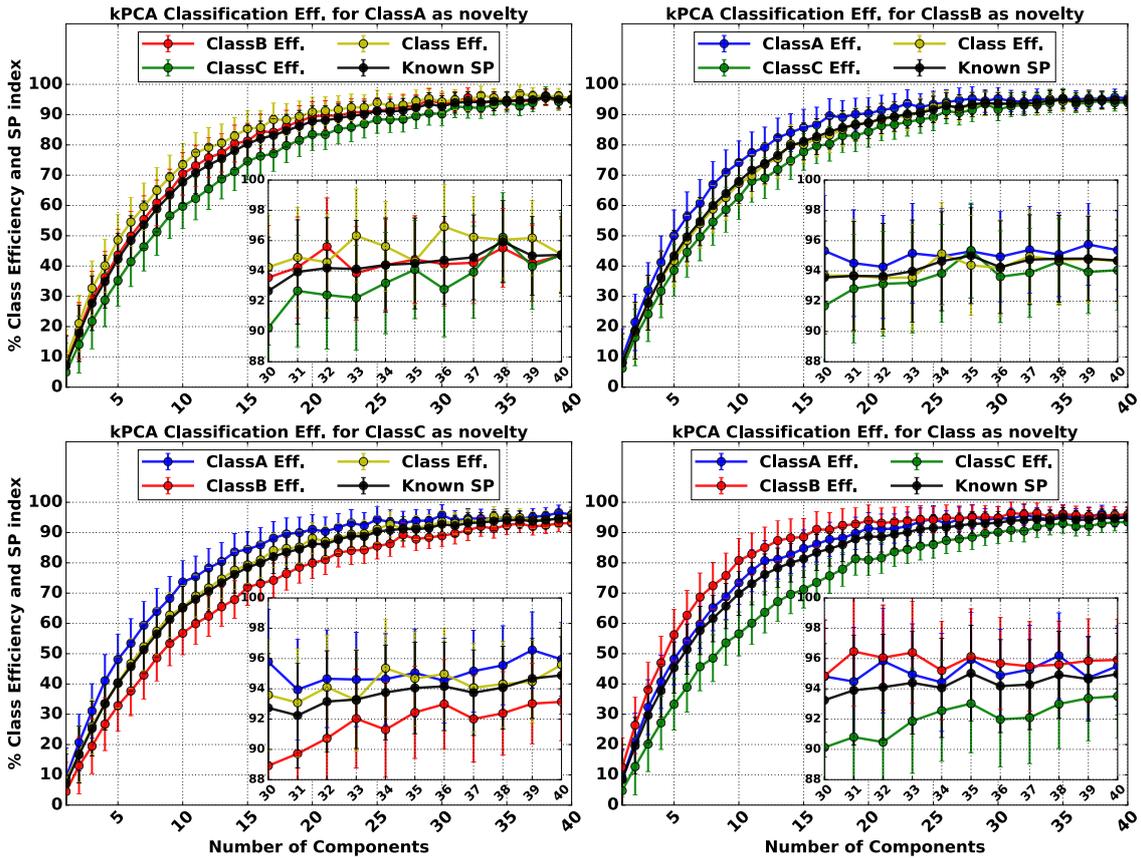


Figura 5.6: Resultados para a eficiência de classificação variando com o número de componentes principais por kernel extraídos, o kernel utilizado foi gaussiano e a largura de banda do kernel foi escolhida para minimizar o erro de representação dos dados. Cada gráfico representa o resultado para uma classe sendo tratada como novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, em cada um dos gráficos, a quantidade de componentes onde houve a estabilização do índice SP, ou seja, não há um aumento nos valores de SP quando há adição de outro componente.

cessários na camada de codificação, foi utilizada uma análise do erro de reconstrução, que, também, pode ser encontrada em algumas referências em *deep learning*.

O erro de reconstrução é o erro médio quadrático medido entre a entrada e a saída na configuração de *auto-encoder*. Nesta configuração, a rede neural visa a reconstrução dos dados de entrada através das funções de ativação não-lineares dos neurônios que compõem a camada escondida. Assim sendo, à medida que há a adição de neurônios à camada escondida, maior poder de reconstrução dos dados é conferido ao modelo, que tem à tendência a reduzir seu erro de reconstrução. Na Figura 5.7, vemos que o erro de reconstrução em função da quantidade de neurônios da camada de codificação (*auto-encoder*). Através dessa figura, fica clara uma tendência a

redução do erro à medida que a quantidade de neurônios aumenta, embora não fique definida qual a quantidade exata de neurônios que faz com o erro se estabilize.

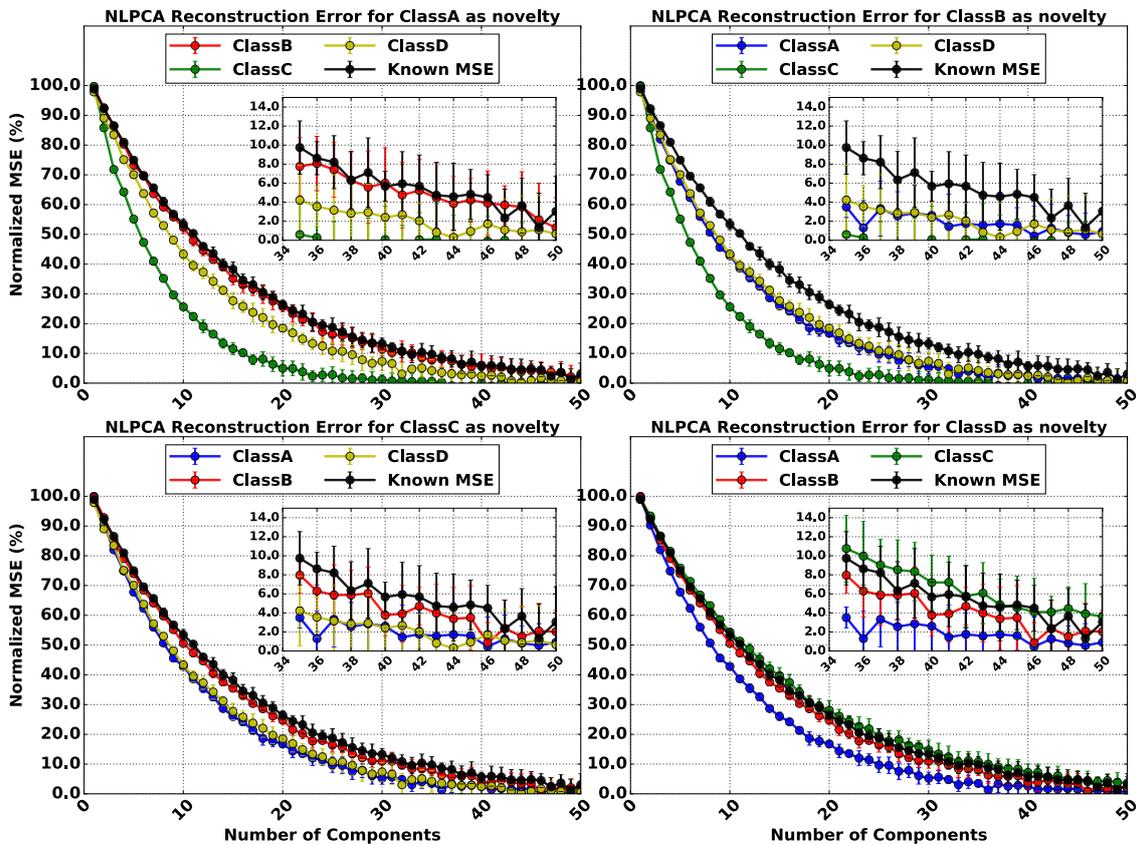


Figura 5.7: Resultados para o erro de reconstrução obtidos com a extração de componentes principais não-lineares, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. O erro de reconstrução foi avaliado para a topologia de componentes principais não-lineares definida no apêndice B. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe, em cada um dos gráficos, a quantidade de componentes onde houve a estabilização do índice SP, ou seja, não há um aumento nos valores de SP quando há adição de outro componente.

Para se obter a quantidade ideal de neurônios na camada de codificação, foi utilizada uma outra análise, que consiste na avaliação do erro de classificação. Depois dos dados serem mapeados pela primeira camada do *auto-encoder*, os mesmos são aplicados a um classificador para um treinamento visando a minimização do erro de classificação.

Com isso, a Figura 5.7 foi obtida, onde podemos ver que, à medida que se aumenta a quantidade de neurônios na camada de codificação, menor fica o erro de reconstrução, até que a estabilidade é obtida para, aproximadamente, 30 neurônios. Outro ponto que pode ser observado aqui é o erro de reconstrução para a Classe B.

Este erro, em específico, obteve valores menores do que os outros error de classes conhecidas, exceto na ausência da Classe D, quando a mesma foi tratada como classe novidade.

Os componentes principais de discriminação extraídos por deflação utilizaram o método descrito detalhadamente no apêndice B. Na teoria, a medida que a extração de cada componente é feita, menos informação relevante para a classificação resta nos dados residuais, ou seja, a extração pode ser monitorada pelo erro de classificação, à medida que cada componente é extraído, durante o processo de treinamento. Além disso, é esperado que quanto mais componentes sejam extraídos, menores variações na eficiência dos classificadores sejam observadas.

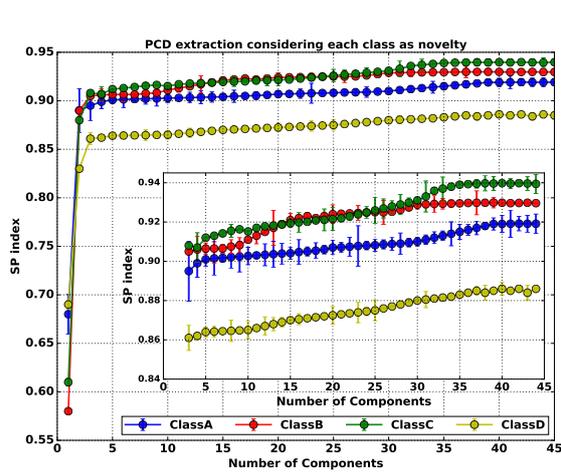
Nas Figuras 5.8a e 5.8b são apresentados as variações do índice SP das classes conhecidas e de acurácia de classificação pela variação de componentes extraídos, para os índices de desempenho. Cada um dos gráficos coloridos representa uma das classes conhecidas sendo tratada como novidade. Como visto anteriormente, quando as classes B e C são tratadas como novidade, tem-se uma melhora expressiva tanto nos valores de acurácia quanto nos valores obtidos pelo índice SP. Além disso, nesta extração, quando a classe D foi tratada como novidade, houve uma alteração negativa nos valores eficiência de classificação (tanto para acurácia quanto para índice SP).

Como na Figura 5.8, as Figuras 5.9a e 5.9b mostram os resultados da extração para índice SP e a acurácia. Um ponto a ser destacado aqui é que a extração cooperativa de componentes obteve um resultado pior para uma menor quantidade de componentes extraídos. Como a extração de cooperativa depende dos componentes anteriores de uma maneira mais forte do que no caso da deflação, este modelo se mostra mais sensível, no que tange a eficiência de classificação, a um menor número de componentes extraídos.

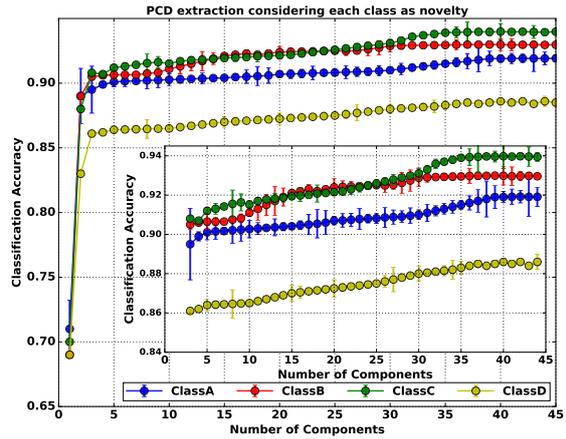
Outro ponto que deve ser observado é que, para alguns componentes, a barra de erro estimada apresentou valores maiores do que outros. Este fato pode ser explicado por uma variação dos pesos iniciais de cada um dos modelos, que mesmo com as 100 inicializações, podem ter ficado presos em um mínimo local durante o processo de treinamento. O resultado mostrado aqui permite concluir que o número ótimo de componentes a serem extraídos é de 40 componentes principais de discriminação extraídos em ambos os casos.

5.3 Detecção de Novidade

O processo de detecção de novidade foi feito utilizando o método exposto no Capítulo 3. Uma vez extraídos os parâmetros dos pré-processamentos, estes foram treinados e o treinamento dos classificadores foi feito com base nos conjuntos de treinamento,



(a) Variação do Índice SP pela variação no número de componentes de discriminação extraídos por deflação. Cada linha colorida no gráfico representa o resultado para uma classe sendo tratada como novidade. A linha em azul representa o resultado obtido para a Classe A sendo tratada como classe novidade. Já a linha em vermelho representa o resultado para a Classe B como classe novidade. A linha em verde representa o resultado para a Classe C como novidade e, por fim, a linha amarela representa a o resultado obtido para a Classe D tratada como novidade. No detalhe, os pontos mais relevantes dos gráficos anteriores são apresentados.

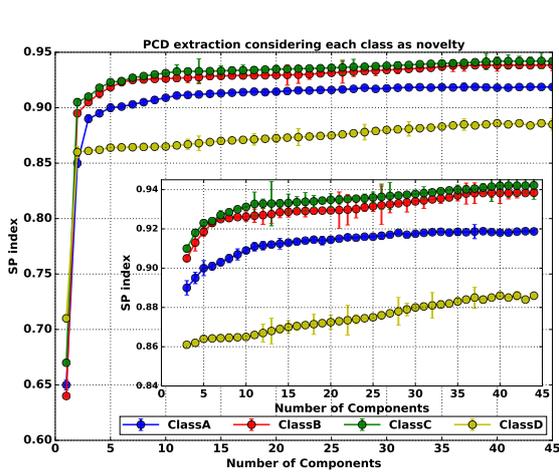


(b) Variação do Acurácia de Classificação pela variação do número de componentes de discriminação extraídos por deflação. Cada linha colorida no gráfico representa o resultado para uma classe sendo tratada como novidade. A linha em azul representa o resultado obtido para a Classe A sendo tratada como classe novidade. Já a linha em vermelho representa o resultado para a Classe B como classe novidade. A linha em verde representa o resultado para a Classe C como novidade e, por fim, a linha amarela representa a o resultado obtido para a Classe D tratada como novidade. No detalhe, os pontos mais relevantes dos gráficos anteriores são apresentados.

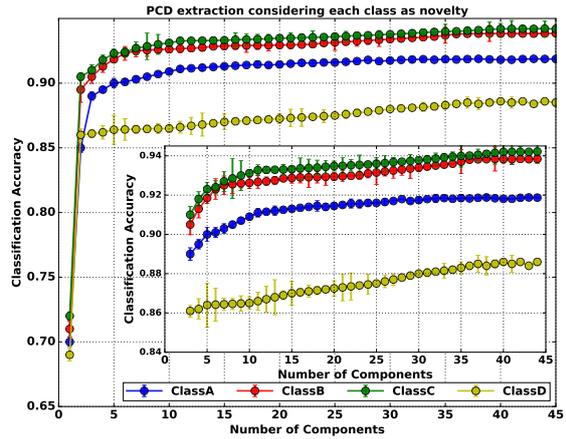
Figura 5.8: Variação dos dois índices de desempenho pela quantidade de componentes principais de discriminação extraídos por deflação. A cada componente, menos informação resta no conjunto de variáveis de entrada. Além da variação no número de componentes, cada uma das classes disponíveis foi tratada como classe novidade.

teste e validação utilizados para a extração de características.

Como descrito anteriormente, a saída de uma rede neural pode ser utilizada para fazer o encapsulamento dos dados conhecidos com a aplicação de um limiar. Usando como referência [97], um treinamento neural foi desenvolvido para a detecção de novidade utilizando como pré-processamento os componentes principais de discriminação extraídos por deflação e extraídos por cooperação. Nas Figuras 5.10 e 5.11 os resultados são mostrados para o treinamento realizado com dados projetados no espaço gerado pela projeção nos componentes principais de discriminação extraído por cooperação e por deflação, respectivamente. Como pode ser visto nas figuras, à medida que aumenta-se o valor do limiar de decisão, a eficiência de classificação das classes conhecidas diminui, pois eventos que excitaram pouco os neurônios de saída passam a ser classificados como eventos pertencentes à classe novidade, até que; no



(a) Variação do Índice SP pela variação no número de componentes de discriminação extraídos por cooperação. Cada linha colorida no gráfico representa o resultado para uma classe sendo tratada como novidade. A linha em azul representa o resultado obtido para a Classe A sendo tratada como classe novidade. Já a linha em vermelho representa o resultado para a Classe B como classe novidade. A linha em verde representa o resultado para a Classe C como novidade e, por fim, a linha amarela representa a o resultado obtido para a Classe D tratada como novidade. No detalhe, os pontos mais relevantes dos gráficos anteriores são apresentados.



(b) Variação do Acurácia de Classificação pela variação do número de componentes de discriminação extraídos por cooperação. Cada linha colorida no gráfico representa o resultado para uma classe sendo tratada como novidade. A linha em azul representa o resultado obtido para a Classe A sendo tratada como classe novidade. Já a linha em vermelho representa o resultado para a Classe B como classe novidade. A linha em verde representa o resultado para a Classe C como novidade e, por fim, a linha amarela representa a o resultado obtido para a Classe D tratada como novidade. No detalhe, os pontos mais relevantes dos gráficos anteriores são apresentados.

Figura 5.9: Variação dos dois índices de desempenho pela quantidade de componentes principais de discriminação extraídos cooperativamente. A cada componente, menos informação resta no conjunto de variáveis de entrada. Além da variação no número de componentes, cada uma das classes disponíveis foi tratada como classe novidade.

limite, quando o valor do limiar se torna 1,0, todos os eventos são classificados como novidade, o que faz com que a detecção de novidade seja completa e a classificação de classes conhecidas seja inexistente.

Os neurônios de saída da rede neural treinada para a detecção de novidade possuem como função de ativação a tangente hiperbólica; assim sendo, a faixa de variação do limiar utilizado para a detecção de novidade varia de $-1,0$ a $+1,0$. Na Figura 5.10 pode ser observado que, para valores de limiar de decisão menores do que $-0,75$, a detecção de novidade sofre poucas alterações, bem como as eficiências de classificação de classes conhecidas; ou seja, pode-se inferir que não houve saídas classificadas corretamente com valores menores do que $-0,75$. Assim sendo, vemos que os gráficos de eficiência de classificação apresentam valores um pouco abaixo de

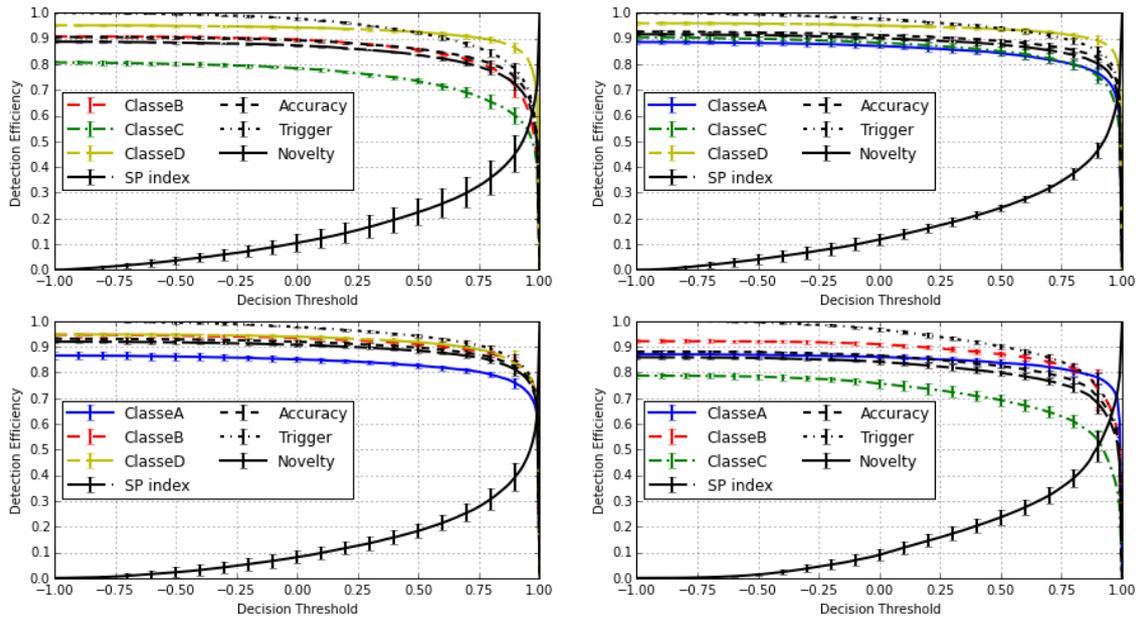


Figura 5.10: Resultado para detecção de novidade utilizando redes neurais MLP com um limiar de excitação na camada de saída e dados pré-processados com componentes principais de discriminação extraídos por cooperação entre componentes. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados.

100%. Ou seja, mesmo com o mínimo valor para o limiar, que, neste caso, pode ser visto como o ato de desligar a detecção de novidade, ainda há alguma confusão entre as classes conhecidas.

Um fato que deve ser considerado é o posicionamento das classes no espaço de entradas da rede neural. Por exemplo, quando uma análise dos resultados é feita tomando como referência a Classe D, pode-se obter uma informação que, no primeiro momento, não está expressa no resultado apresentado na Figura 5.10 de maneira direta. Quando a Classe D é treinada como uma classe conhecida, a sua eficiência de classificação é mantida, praticamente, inalterada na presença de outras classes, o que pode indicar que eventos da Classe D se encontram afastados de eventos de outras classes no espaço de entradas da rede neural. O mesmo não ocorre com as outras classes e isso pode ser visto tomando-se a Classe A, como exemplo.

Outro fato que deve ser levado em consideração, neste caso, é que diferentes classes podem ocupar o mesmo local no espaço de características, o que pode tornar a sua classificação desafiadora. Caso a Classe C seja treinada como classe conhecida,

ocorre uma variação na sua eficiência de classificação se os eventos desta classe forem classificados por classificadores treinados com diferentes conjuntos de classes conhecidas. Quando a Classe C é treinada como classe conhecida junto com eventos das Classes B e D (Figura 5.10 na parte esquerda superior), a sua eficiência se aproxima do valor 90% quando o limiar está abaixo de $-0,75$. No caso de um treinamento realizado com as classes A, B e C como conhecidas (Figura 5.10 na parte direita inferior), a sua eficiência sofre uma leve melhora. Ou seja, na presença da classe D, a classe C sofre uma redução nos valores de sua classificação e isso pode ser visto como as duas classes compartilhando o mesmo espaço no domínio das entradas.

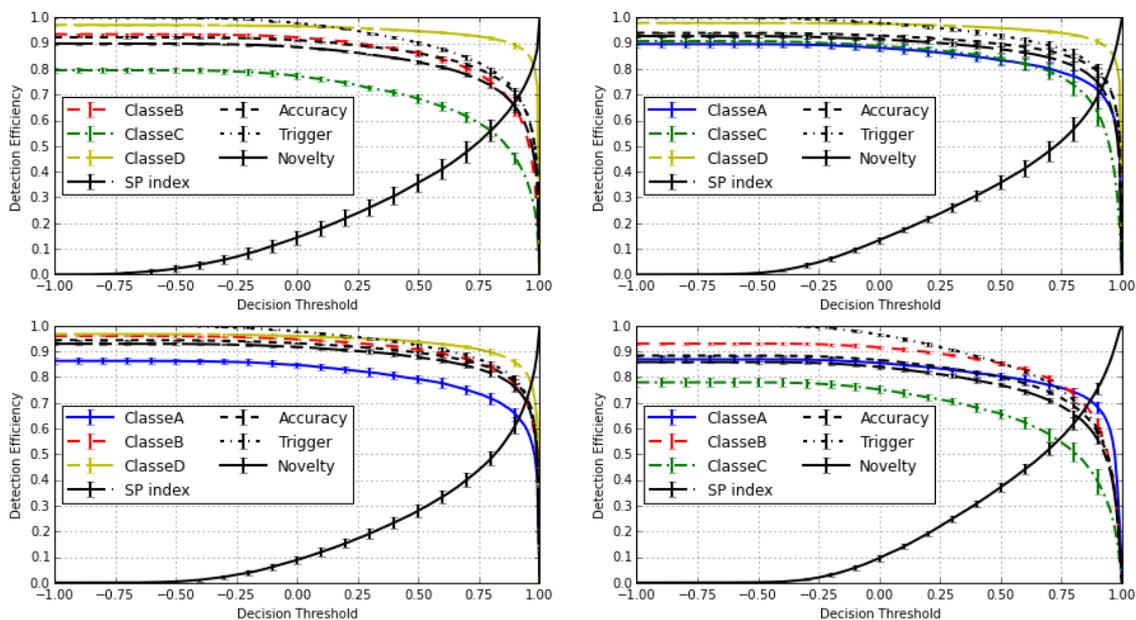


Figura 5.11: Resultado para detecção de novidade utilizando redes neurais MLP com um limiar de excitação na camada de saída e dados pré-processados com componentes principais de discriminação extraídos por deflação. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados.

A Figura 5.11 mostra que os resultados obtidos com modelos alimentados com dados projetados nos componentes principais de discriminação extraídas por deflação apresentaram valores de saída mais próximos à saturação da função de ativação tangente hiperbólica: para que ocorra uma redução na eficiência de classificação, de maneira geral, os valores do limiar tendem a ser um pouco maiores neste caso,

quando comparados ao caso de redes neurais treinadas com dados projetados nos componentes principais de discriminação extraídos de maneira cooperativa. Embora, de maneira geral, o comportamento obtido seja o mesmo.

A realização da detecção de novidade a partir deste ponto foi feita com base na técnica de *One-Class SVM*, utilizada da seguinte maneira: para cada classe conhecida foi treinado um classificador, chamado de encapsulador de classe conhecida, e para todos os eventos conhecidos foi treinado um encapsulador geral, chamado de detector de novidade. Como são 4 classes que compõem o conjunto de dados, cada uma das classes foi omitida durante o processo de treinamento, restando assim 3 classes conhecidas.

O kernel utilizado foi o gaussiano, com parâmetro γ igual a 0, 1. Além disso, uma variação dos valores do parâmetro ν foi feita. Como dito anteriormente, quando o valor de ν é muito pequeno, mais plástico é o classificador, menor a probabilidade de *overtraining* e maior é a sua eficiência para a classificação de classes conhecidas. Quanto maior o valor de ν , menos plástico é o classificador, maior a probabilidade de *overtraining* e menor a sua eficiência para a classificação de classes conhecidas.

No que tange a detecção de novidade, quanto menos plástico for o classificador, maior a eficiência em detectar eventos da classe desconhecida. Apenas eventos muito próximos do perfil de eventos conhecidos serão classificados como pertencentes a classe conhecida. Todos os demais eventos serão classificados como novidade.

A Figura 5.12 mostra os resultados obtidos sem o pré-processamento dos dados, ou seja, a saída da análise LOFAR foi apresentada diretamente para os modelos de SVM. Aqui, temos a apresentação dos resultados para cada uma das classes disponíveis no banco de dados fazendo o papel de classe novidade. O resultado esperado teoricamente foi observado; ou seja, modelos com valores de ν menores apresentaram maior eficiência de classificação, enquanto que modelos com ν maiores apresentaram maior eficiência na detecção de novidade. Nesta figura, as linhas coloridas representam a eficiência de cada uma das classes conhecidas (aqui representada por C- i , onde i é o identificador da classe, por motivos de organização) enquanto que a linha preta sólida mostra o resultado para a detecção de novidade. Além disso, são mostrados também os gráficos de acurácia de classes conhecidas (preto pontilhado com linhas) e o *trigger* das classes conhecidas. Como pode ser observado, houve uma variação relativamente grande nos resultados obtidos, uma vez que todos os gráficos apresentam uma barra de erro de pelo menos 5%. Tais incertezas também foram observadas nos resultados de redes neurais artificiais alimentadas com dados compactados por ambos os métodos de extração de componentes principais de discriminação.

Além disso, como foi observado anteriormente, na presença da classe B, a eficiência da classe C tem uma leve redução. Outro ponto que vale ser destacado é o ponto de subida do gráfico de detecção de novidade (que ocorreu em torno de

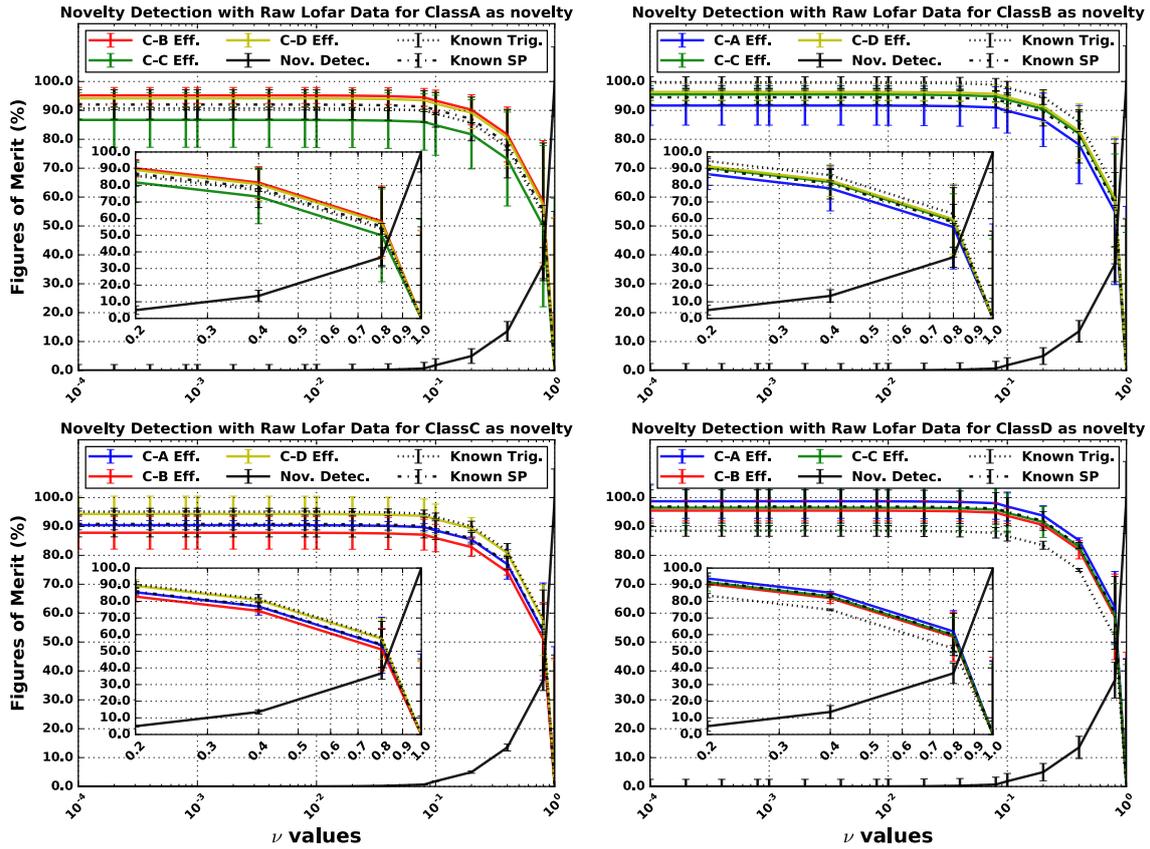


Figura 5.12: Resultado para Detecção de Novidade baseada em modelos *one-class* SVM sem a aplicação de técnicas de pré-processamento. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

$\nu = 1,0 \cdot 10^{-2}$), este valor pode ser utilizado como uma figura de mérito indireta para a comparação de diversos detectores. Além disso, os valores do índice SP obtidos para modelos voltados para a classificação (valores pequenos de ν) foram ligeiramente maiores do que os valores obtidos com as redes neurais anteriormente apresentadas, chegando a alcançar 98,0% para classe D como novidade (Figura 5.12).

Na Figura 5.13, os resultados de detecção de novidade para modelos alimentados por dados compactados com PCA (percentual de compactação de 72,5%). Nota-se uma redução na área embaixo da curva de eficiências de classificação quando comparadas com a Figura 5.12. Observando-se os valores do índice SP (curva pontilhada

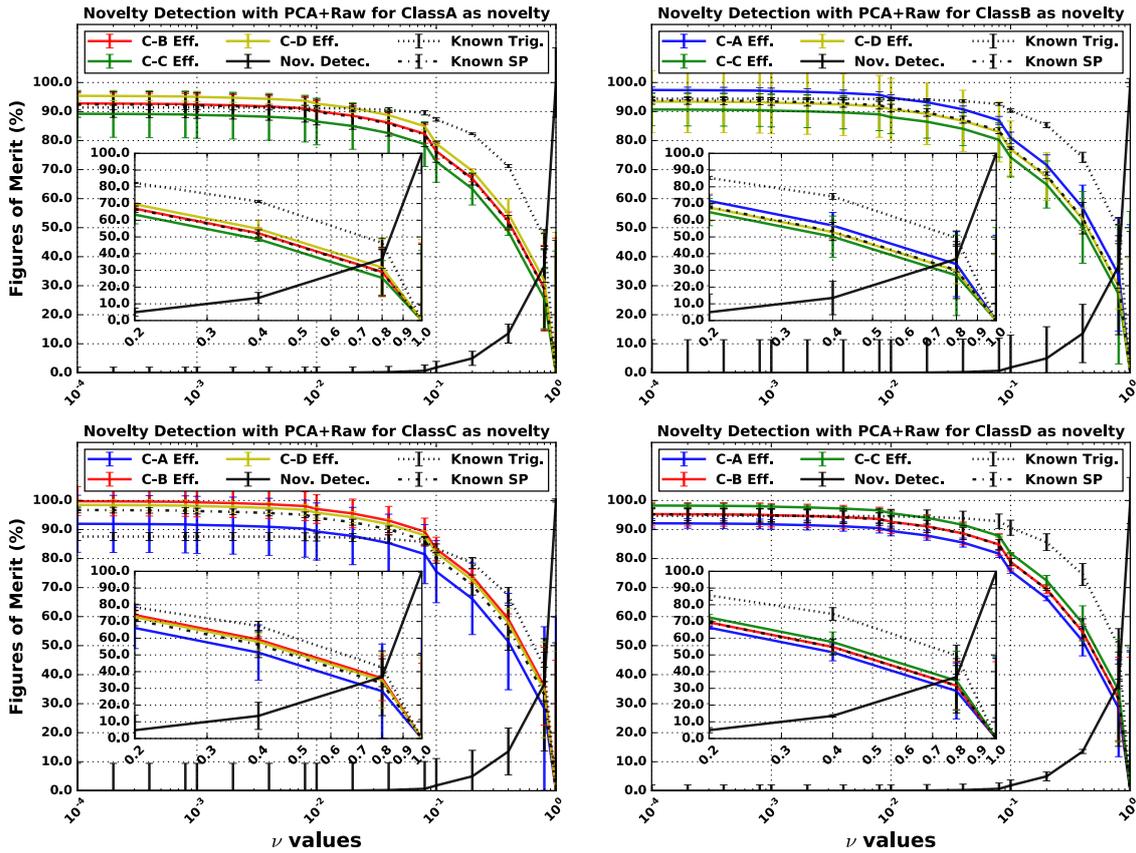


Figura 5.13: Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais (baseados nos auto-vetores da matriz de autocorrelação). Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

com linhas pretas), vê-se que os mesmos sofreram uma redução para valores de $\nu > 10^{-1}$. Este fato indica uma redução eficiências de classificação para as classes conhecidas para valores menos de ν . Outra observação que pode ser feita é que o valor de ν que permite que eficiência na detecção de novidade apresente um valor acima de zero, e este se dá em torno de $\nu = 1,0 \cdot 10^{-2}$ (mesmo valor apresentado para dados sem compactação). Por estes resultados, conclui-se que, com a compactação por PCA, os dados de sonar passivo apresentaram um poder discriminatório menor do que com os dados sem compactação alguma e os dados das classes não apresentadas durante o período de treinamento apresentaram a mesma capacidade de discriminação (uma vez que o valor, significativamente, acima de zero se deu, apro-

ximadamente, no mesmo ponto).

Outro dado que pode ser retirado da figura em questão é a variância dos resultados obtidos. De um modo geral, os resultados apresentaram uma maior variância quando comparados com os resultados obtidos sem a compactação de dados. Uma variável que se destacou neste cenário foi o *trigger*, uma vez que este índice de desempenho não apresentou uma grande variação (barra de erro) neste etapa de desenvolvimento dos modelos.

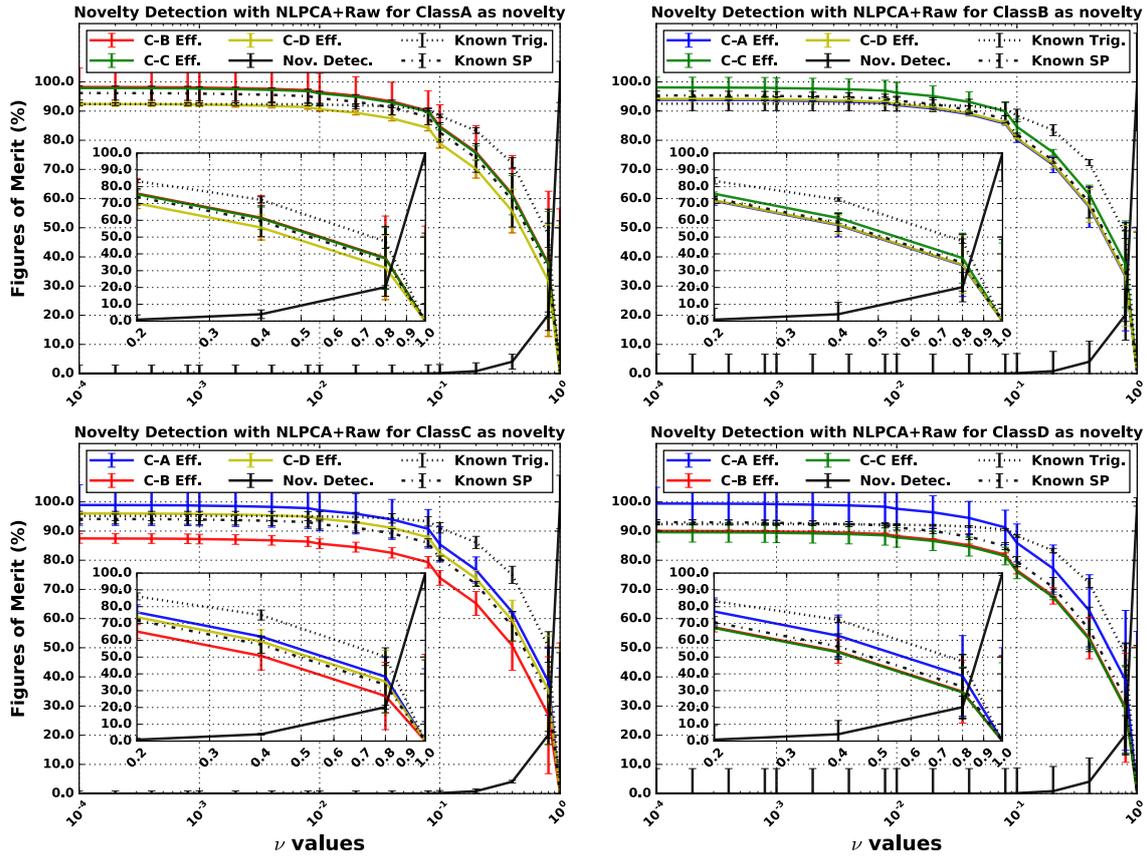


Figura 5.14: Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais não-lineares (extraídos com base em uma rede neural auto-associativa). Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

Os resultados obtidos com a compactação NLPCA estão expostos na Figura 5.14. Nesta figura, o mesmo comportamento apresentado nos resultados obtidos com outros modelos de compactação de dados foi observado. Aqui, de uma maneira

mais impactante do que no caso da compactação por PCA. No caso da Figura 5.14, os valores de *trigger* não se diferenciaram dos valores de eficiência das classes conhecidas. Este fato pode ser explicado por um mapeamento dos dados conhecidos em uma única região do espaço não-linear, criando assim, uma região bem definida para o encapsulamento dos dados conhecidos. Assim sendo, os dados de classes conhecidas que sejam adquiridos pelo sistema de sonar passivo, serão encapsulados pelos modelos treinados como dados conhecidos, aumentando assim o valor obtido do índice de desempenho *trigger*.

Outro fato que pode ser observado é a redução na eficiência de classificação para valores de ν pequenos, que podem ser chamados de ν de classificação. A presença de um valor, significativamente, maior do que zero para de detecção de novidade se dá em $\nu > 3 \cdot 10^{-1}$, ou seja, um valor maior do que os valores apresentados para classificadores alimentados com dados brutos e dados compactados com PCA. Por outro lado, a área embaixo da curva do índice SP, de maneira geral, se mostrou maior do que a apresentada para classificadores treinados com dados compactados por PCA, mas menor do que a mesma obtida por classificadores treinados sem compactação.

Os classificadores treinados com dados compactados por kPCA têm os seus resultados apresentados na Figura 5.15. Nesta figura, vale a pena notar que a área embaixo da curva dos valores do índice SP é consideravelmente maior do que as mesmas áreas obtidas anteriormente (Figuras 5.12, 5.13 e 5.14). A eficiência de classificação do sistema ajustado para classificação ($\nu \approx 0$) se manteve na mesma ordem de grandeza dos resultados obtidos com modelos alimentados com dados não compactados.

No que tange a presença de um valor significativo para detecção de novidade, é possível observar que o valor de ν que permite tal fato é ligeiramente menor do que os anteriores, isto é, com uma menor degradação da eficiência de classificação, para estes modelos, é possível se obter um valor maior de detecção de novidade, quando comparados com os modelos treinados anteriormente. Cabe ainda ressaltar que para a detecção de novidade, a variância dos resultados obtidos é melhor do que suas contra-partes obtidas anteriormente.

A variância dos resultados obtidos neste caso, indica uma maior sensibilidade do pré-processamento aos dados de sistemas de sonar passivo que foram utilizados para a extração dos parâmetros. Vale ressaltar que se nenhuma das dimensões geradas pelas kPCA possui energia estatística próxima de zero, tem-se uma reconstrução por pré-imagem.

Na Figura 5.10 são apresentados os resultados para os modelos alimentados com dados compactados por PCD extraídas de forma cooperativa. Inicialmente, observa-se que um valor significativamente maior do que zero para a detecção de novidade

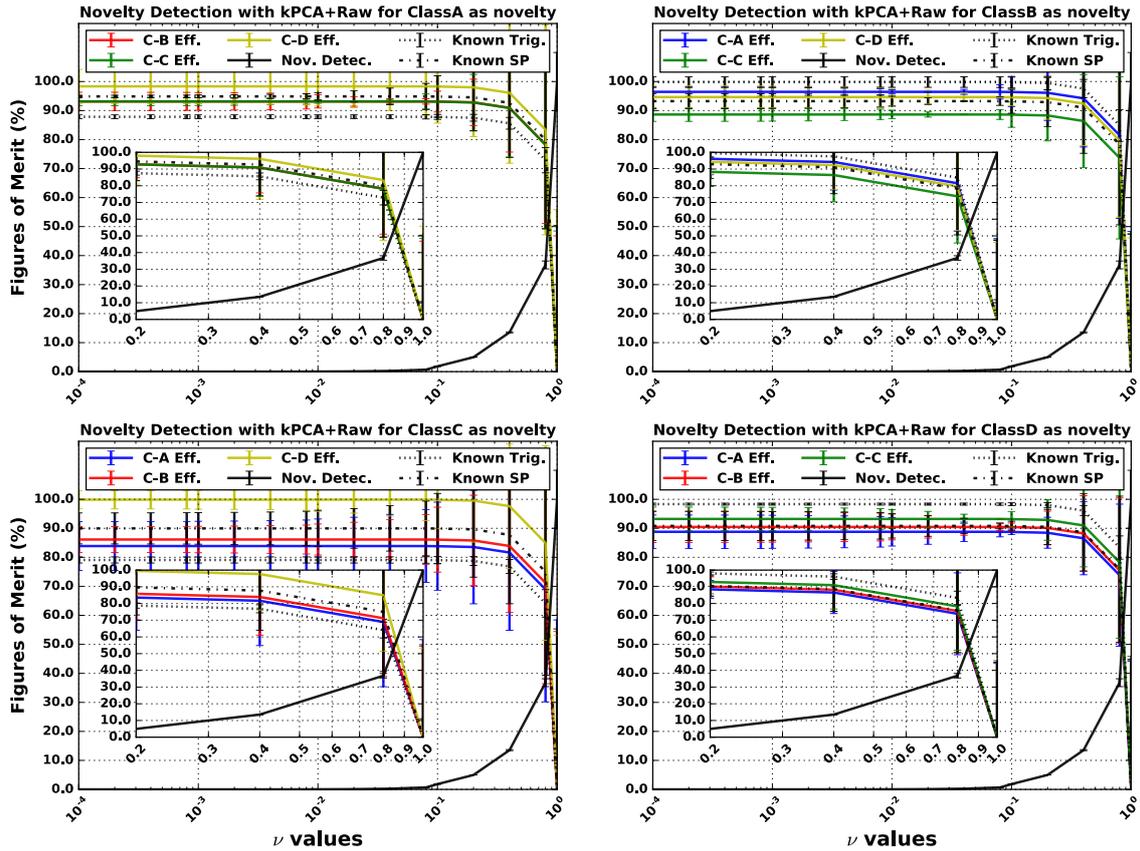


Figura 5.15: Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais não-lineares (extraídos com a projeção no espaço de *kernel*), ou kPCA. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

se dá para $\nu > 1 \cdot 10^{-3}$, ou seja, o menor valor de ν obtido até o presente momento, quando se compara, o mesmo ponto, para os outros métodos de compactação

No caso da classificação, pode-se observar que a área embaixo da curva do índice SP supera as áreas obtidas anteriormente, até mesmo a área obtida com os classificadores treinados com a compactação kPCA, que apresentou os melhores resultados para área embaixo da curva. De maneira geral, o resultado de classificação apresentou um menor valor de variância quando comparado com os outros resultados obtidos. Em compensação, a barra de erro apresentada no gráfico de eficiência de detecção de novidade tem valores relativamente pequenos.

Pode-se verificar, nestes resultados, a ocorrência do mesmo efeito de interferência

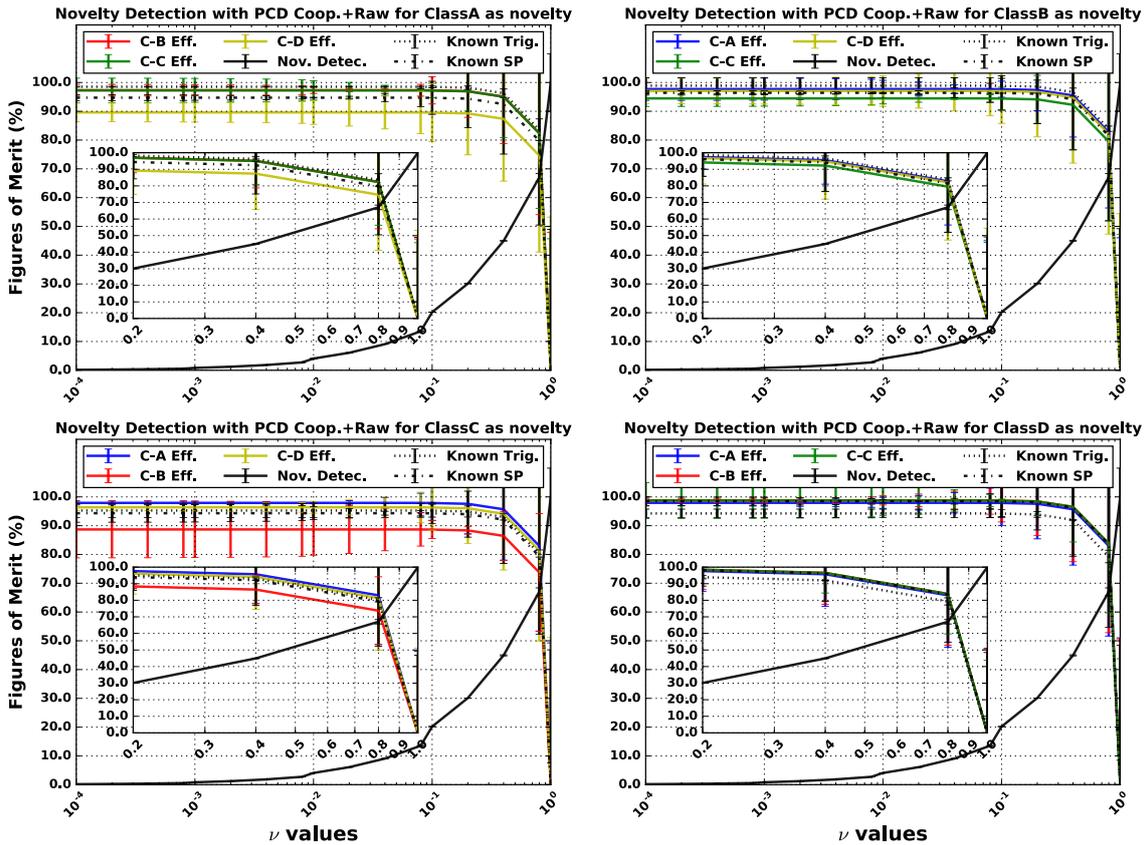


Figura 5.16: Resultado da Detecção de Novidade para PCD por cooperação como pré-processamento. Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais de discriminação extraídos cooperativamente. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

na classificação das classes B e C. Quando uma destas classes não está presente, a outra sobre uma alteração significativa em sua eficiência, o que pode evidenciar uma correlação não-linear entre os eventos pertencentes as classes em questão.

No caso de componentes de PCD extraídos por deflação, os resultados são apresentados na Figura 5.17. As áreas embaixo das curvas do índice SP para cada uma das classes fazendo as vezes de classes de novidade são da mesma ordem de grandeza dos resultados obtidos com modelos alimentados por dados projetados nos componentes principais de discriminação extraídos cooperativamente.

Além disso, a presença de um valor acima de zero para detecção de novidade se

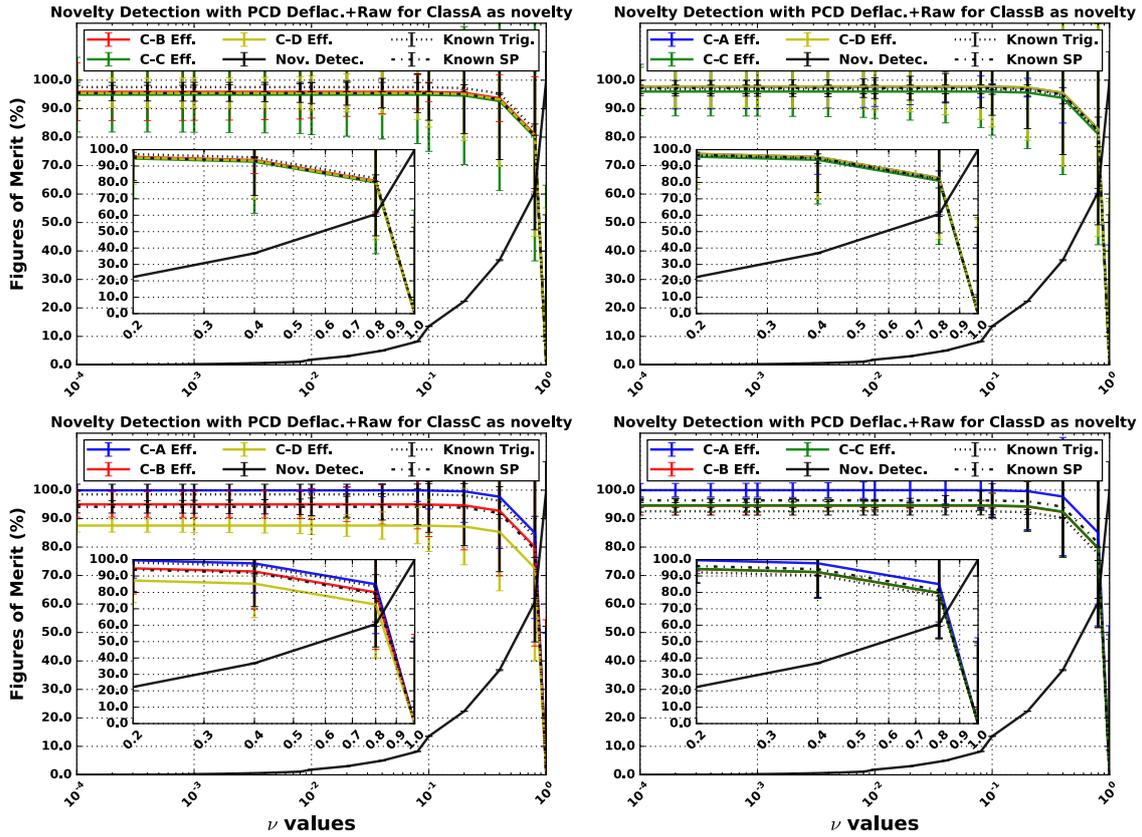


Figura 5.17: Resultado da Detecção de Novidade para PCD por cooperação como pré-processamento. Resultado para Detecção de Novidade baseada em modelos *one-class* SVM alimentados com dados projetados nos componentes principais de discriminação extraídos por deflação. Os resultados expostos representam a eficiência de cada uma das classes conhecidas (*plots* coloridos), a acurácia de classificação, o índice SP e o *trigger*, todos estes obtidos para as classes conhecidas, onde cada um dos gráficos foi obtido com uma classe sendo tratada como classe novidade. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

dá em valores de ν relativamente baixos ($\nu = 10^2$). E a variância dos resultados encontra-se na mesma ordem de grandeza das obtidas por modelos alimentados com dados projetados nos componentes principais de discriminação extraídos pelo método cooperativo.

Uma das maneiras de se realizar a comparação entre os métodos é ilustrado nas Figuras 5.18 e 5.19. Na Figura 5.18, os gráficos com os valores de *trigger* e da eficiência de detecção de novidade para todos os pré-processamentos testados são comparados em uma única figura, variando-se os valores do parâmetro ν . Já na Figura 5.19, o mesmo resultado, mas desta vez para valores do índice SP, ao invés

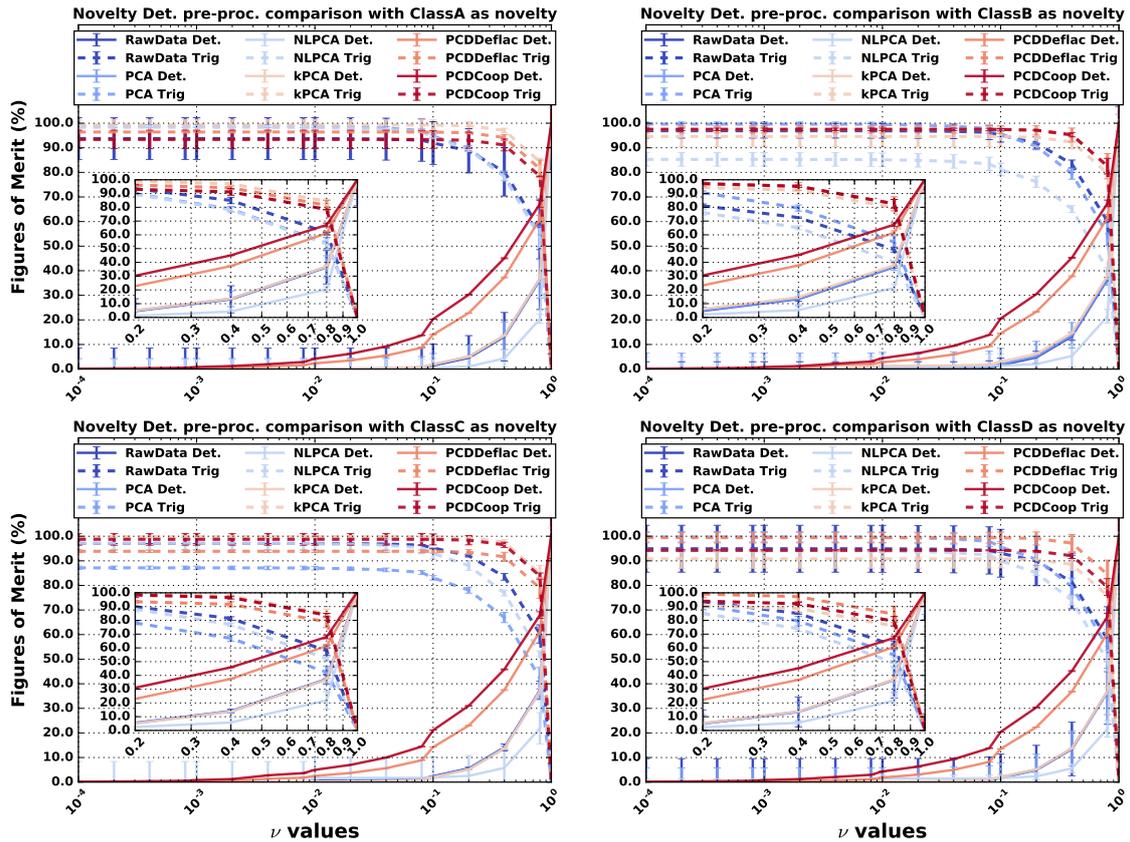


Figura 5.18: Comparação dos resultados obtidos com diferentes pré-processamentos para detecção de novidade e *trigger*. Comparação entre os resultados para diferentes pré-processamentos aplicados durante o desenvolvimento da tese. Os resultados expostos representam os valores de *trigger* (*plots* tracejados) e de detecção de novidade (*plots* com linhas sólidas) para cada um dos pré-processamentos, sendo que os valores de *trigger* foram obtidos para as classes conhecidas e os valores de detecção novidade foram obtidos para cada uma das classes disponíveis sendo tratadas como classe novidade, uma por vez. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

de valores de *trigger*, é apresentado.

Na Figura 5.18, observa-se que, modelos que foram treinados com dados pré-processados com NLPCA obtiveram os piores resultados na figura de mérito *trigger*. Pode-se notar também, que ambas as extrações de componentes principais de discriminação obtiveram valores de *trigger* elevados, e seus gráficos apresentaram valores maiores de *trigger* para valores de ν mais elevados, superando até mesmo os resultados obtidos com dados sem compactação, o que indica uma redução de dimensões que atrapalhavam a classificação das classes conhecidas. Analisando esta figura, pode-se dizer que os outros métodos de pré-processamento apresentaram valores

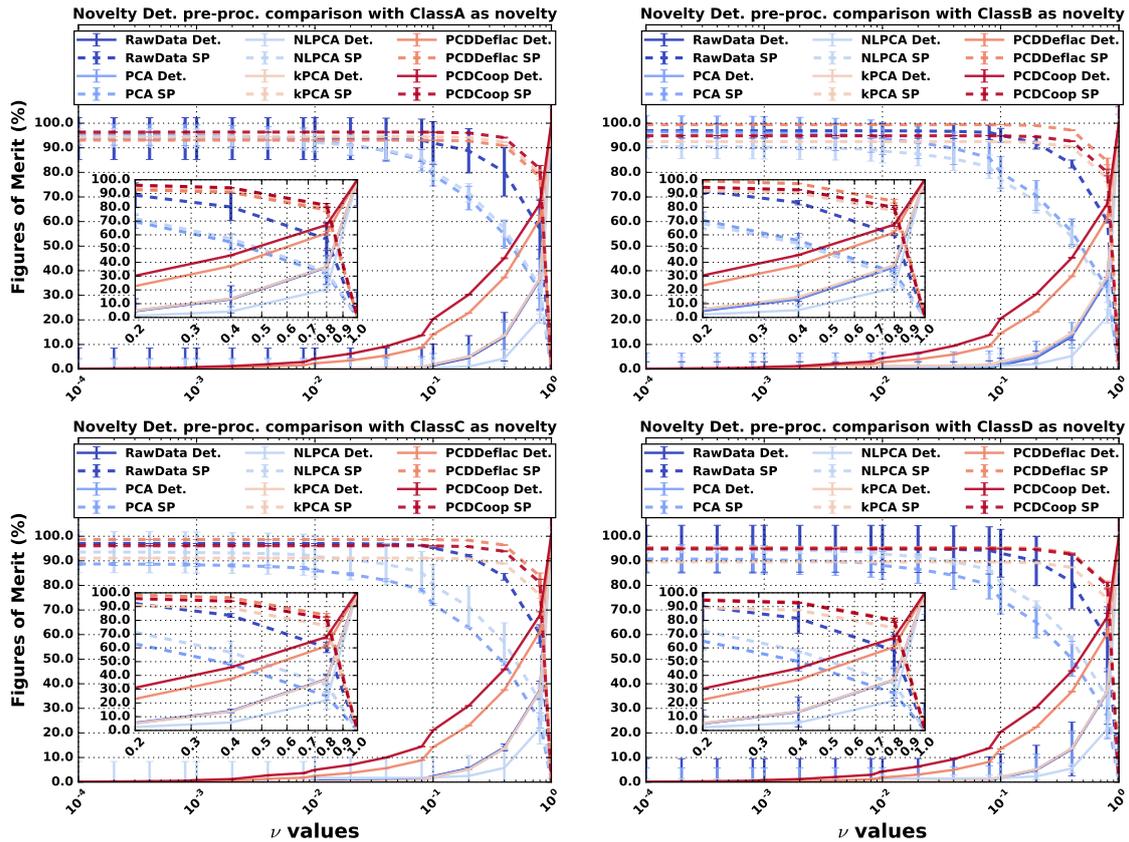


Figura 5.19: Comparação dos resultados obtidos com diferentes pré-processamentos para detecção de novidade e índice SP. Comparação entre os resultados para diferentes pré-processamentos aplicados durante o desenvolvimento da tese. Os resultados expostos representam os valores do índice SP (*plots* tracejados) e de detecção de novidade (*plots* com linhas sólidas) para cada um dos pré-processamentos, sendo que os valores do índice SP foram obtidos para as classes conhecidas e os valores de detecção novidade foram obtidos para cada uma das classes disponíveis sendo tratadas como classe novidade, uma por vez. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

similares de *trigger*. Além disso, ainda nesta figura, os valores para detecção de novidade apresentados para modelos alimentados com dados projetados nos primeiros componentes principais de discriminação podem ser analisados de acordo com a área embaixo da curva dos mesmos, e assim, uma maneira quantitativa de avaliação e desenvolvimento de modelos pode ser definida. Através deste índice de desempenho, o melhor método de compactação para detecção de novidade foram os componentes principais de discriminação por deflação.

A Figura 5.19 compara os valores de SP e detecção para diferentes valores de ν e pode-se observar o mesmo comportamento obtido para o *trigger*, exceto pela

equivalência entre PCA e NLPCA em seus resultados.

Os resultados obtidos com os *Stacked AutoEncoders* estão apresentados nas figuras a seguir. Na Figura 5.20, são apresentados os resultados para a primeira camada de um modelo neural inicializado com os pesos extraídos de dois *auto-encoder* empilhados com topologia 400 – 350 – 300 – 350 – 400 para a Classe A como classe novidade. Esta figura foi gerada a partir da variação do número de neurônios na camada escondida do modelo e, estimando-se a eficiência de classificação (aqui representada pelo índice SP, eixo y esquerdo) e a eficiência de reconstrução, aqui obtida pela divergência KL medida entre os dados de entradas e os dados de saída, separados entre classe conhecidas e classe novidade. Ou seja, a primeira camada de *auto-encoder* foi inicializada para fazer a compactação dos dados originais de 400 para 350 dimensões. Posteriormente, este mapeamento foi utilizado como entrada da segunda camada de compactação, desta vez reduzindo de 350 para 300 dimensões. Como pode ser observado, para poucos neurônios na primeira camada de codificação, existe uma diferença na reconstrução (observa-se a diferença de valores entre dados conhecidos e desconhecidos na divergência KL). Além disso, a máxima eficiência de classificação foi obtida para 350 neurônios na camada escondida do modelo.

De posse deste modelo de compactação, foram realizados os treinamentos de modelos de aprendizado profundo baseados nos *auto-encoders*. Os modelos foram desenvolvidos utilizando a seguinte regra: a cada nova camada adicionada, uma compactação seria realizada e a dimensão de saída seria 50 neurônios menor do que a dimensão de entrada (tomando como base o resultado da Figura 5.20). Assim sendo, foram geradas as Figuras 5.21, 5.22, 5.23 e 5.24.

Uma vez realizado o treinamento de cada uma camadas para a reconstrução dos dados, um outro treinamento, desta vez para classificação das classes conhecidas é realizado e, por fim, é aplicado o método do limiar para a detecção de novidade. Para cada uma das figuras abaixo, os modelos foram treinados com uma quantidade diferente de camadas, à medida que o número de camadas aumenta, o número de neurônios pertencentes a estas, diminui, realizando assim a compactação dos dados de entrada em dimensões cada vez menores e com informações mais transformadas.

Os resultados obtidos para estes treinamentos mostram que as eficiências para as classes conhecidas encontram-se um pouco abaixo daquelas obtidas pelo modelo neural MLP (Figuras 5.10 e 5.11) e das obtidas com os modelos baseados em SVM e alimentados com dados pré-processados ou não compactados (Figura ??), exceto para o caso de PCA e NLPCA. Um ponto a ser observado é que o valor do *trigger* obtido foi consideravelmente elevado, o que pode ser justificado pelo mapeamento do modelo treinado. Se este mapeamento levou os dados conhecidos para uma mesma região, a capacidade discriminatória do modelo final treinado pode ser reduzida (eficiências individuais das classes conhecidas apresentando valores menores), e ainda

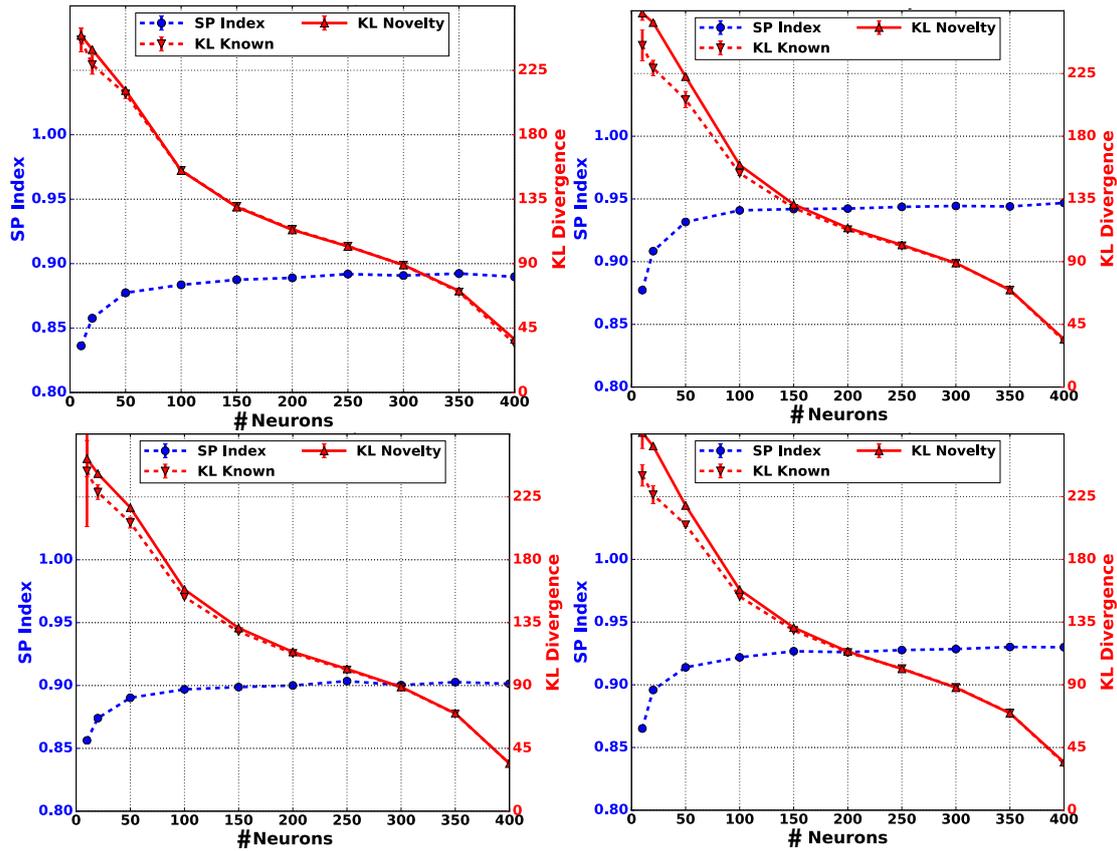


Figura 5.20: Resultados de classificação e reconstrução para uma rede neural auto-associativa com duas camadas, uma de codificação e outra de decodificação. Os resultados expostos representam a eficiência de classificação (índice SP, *plots* em azul) e a dissimilaridade de representação (divergência KL, *plots* em vermelho) para cada uma das classes conhecidas, variando-se a quantidade de neurônios na camada de representação. Na parte esquerda superior, os resultados obtidos para a classe A como novidade são apresentados. Na parte direita superior, os resultados para a classe B como novidade. Na parte esquerda inferior, encontram-se os resultados para a classe C como novidade, enquanto que na parte direita inferior, os resultados para a classe D sendo tratada como novidade são apresentados. No detalhe de cada um dos gráficos, pontos de interesse são detalhados.

assim, o *trigger* apresentará um valor elevado. Além disso, para camadas mais profundas, o valor de *trigger* obtido se reduziu para valores de limiar menores do que para camadas mais rasas.

Observado-se a curva de detecção de novidade, nota-se que seu comportamento é um pouco diferente do apresentado nos modelos anteriores. A inclinação das curvas de novidade apresentadas se mostra maior do que suas contra-partes obtidas com MLP e SVM. Além de apresentar um valor acima de zero para valores relativamente menores de limiar de detecção do que os apresentados em outros modelos. Para camadas mais profundas (Figuras 5.24), as curvas de detecção de novidade se mostraram muito mais íngremes. Estes fatos permitem concluir que modelos com mais camadas parecem ser mais sensíveis a dados de novidade do que modelos com menos

camadas (NLPCA e MLP) embora esta detecção custe muito para a detecção das classes conhecidas (valores *trigger* sofrendo reduções para limiares menores).

As Tabelas 5.2, 5.3 e 5.4 (com a Tabela 5.1 sendo a referência de leitura) mostram um resumo dos resultados obtidos, incluindo detectores de novidade baseados em *Stacked AutoEncoders* e os melhores resultados obtidos por todas as topologias aplicadas durante o período de desenvolvimento da tese, respectivamente. As cores das tabelas foram inseridas para a comparação facilitada pelo leitor. As células de cores verdes obtiveram resultados melhores do que os resultados obtidos pela referência (células em branco). As células em amarelo representam os modelos com resultados próximos aos resultados das células em branco. Por fim, os modelos que alcançaram resultados piores do que os da referência tiveram as suas células coloridas em vermelho.

Na Tabela 5.2, os resultados comparativos de diferentes modelos de detecção de novidade baseados na técnica de *one-class SVM* são apresentados. Como pode ser observado, os melhores resultados foram obtidos com modelos alimentados por dados compactados com componentes principais de discriminação, tanto extraídos por deflação quando pelo método de deflação. A exceção se dá quando são observados os valores para 0%-novidade, onde os modelos alimentados com kPCA atingiram uma melhor eficiência no que tange a medida de performance *trigger*. Com relação aos piores resultados, estes foram atingidos com máquinas de vector suporte que possuíam como entrada de dados compactados com PCA e NLPCA. Isto pode ser explicado pois ambos os pré-processamentos têm por objetivo a reconstrução dos dados. Enquanto PCD e kPCA buscam a classificação e a reconstrução dos dados no mesmo espaço de alta dimensão onde SVM é aplicada, respectivamente.

Já na Tabela 5.3, apresenta os resultados para diferentes quantidades de camadas de redes auto-associativas empilhadas. Como a análise de componentes principais não-lineares pode ser visto como uma rede auto-associativa com 3 camadas escondidas, este modelo de pré-processamento foi adicionado para comparação. Analisado o mais baixo nível de exigência (0%-novidade), os resultados obtidos com redes neurais, que foram inicializadas com os pesos obtidos para representação dos dados, foram melhores do que os resultados da referência (sem compactação) e do que os resultados obtidos com NLPCA. Este resultados não se mantém para um nível intermediário de exigência por parte dos detectores de novidade. Já para um nível alto de exigência de classificação, os detectores de novidade alimentados com dados compactados com 8 camadas de *Stacked AutoEncoders* obtiveram os melhores resultados.

Encerrando a etapa de comparações, os dados compactados com os SAE alimentaram um detector de novidade baseado em *one-class SVM* e estes resultados foram comparados na Tabela 5.4. Os melhores resultados para detecção de novidade com

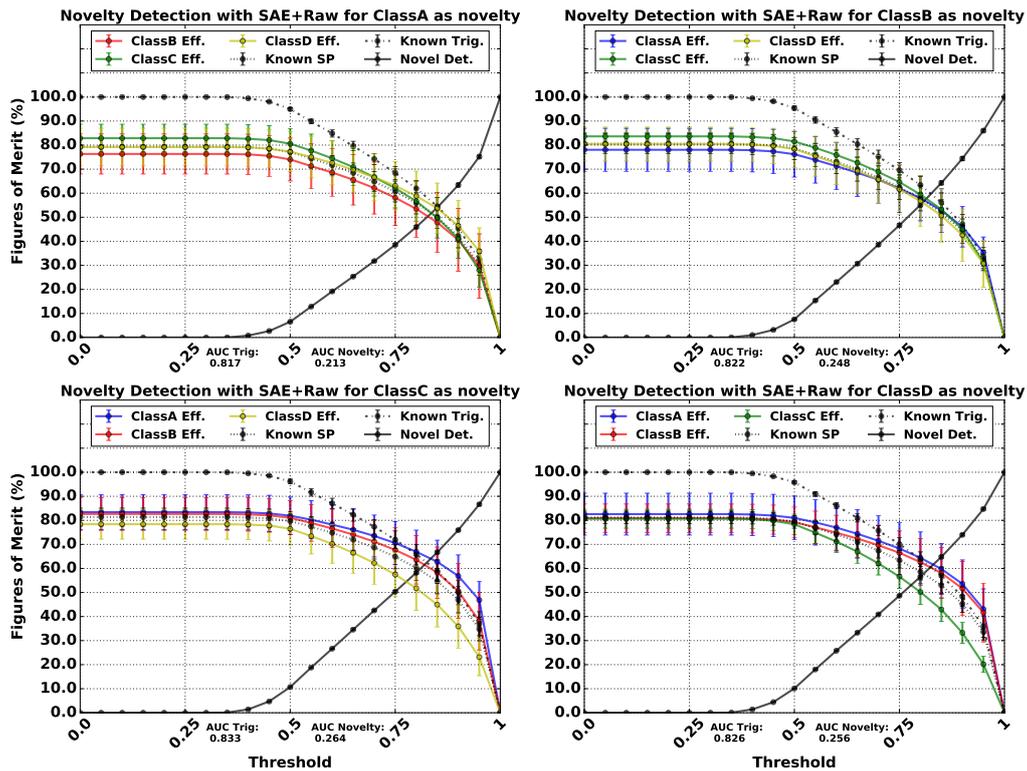
Tabela 5.1: Referência para leitura das tabelas de resultados

	Nível de Exigência	
Descrição do modelo utilizado para extração do resultado	Classe A	Classe B
	Classe C	Classe D

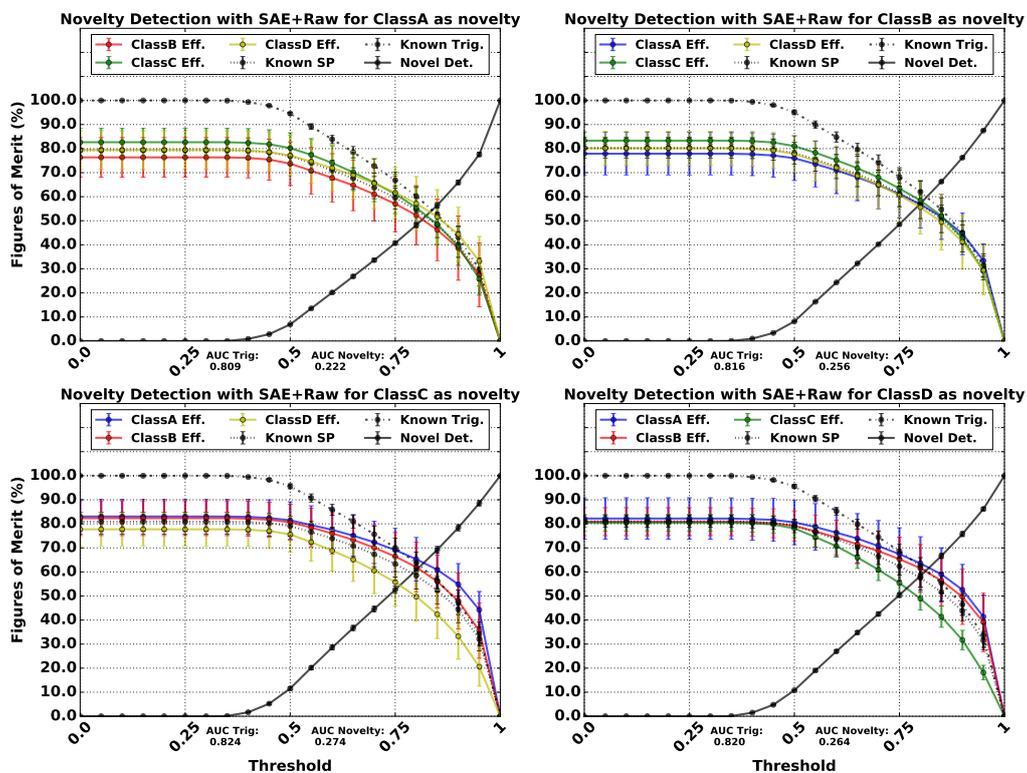
Tabela 5.2: Comparação entre os resultados obtidos com diferentes pré-processamentos, com a referência de leitura da Tabela 5.1

Pre-proc	<i>Trigger</i>					
	0%-Novidade		50%-Novidade		75%-Novidade	
Dados crus	95,2%+-7,9%	98,8%+-1,0%	45,5%+-8,9%	43,5%+-5,5%	22,5%+-0,5%	20,3%+-1,5%
	97,5%+-0,5%	95,5%+-11,5%	41,5%+-4,5%	58,5%+-5,6%	18,8%+-1,0%	12,3%+-1,4%
PCA	92,5%+-2,5%	85,1%+-5,5%	32,5%+-0,5%	24,5%+-0,5%	12,5%+-0,5%	11,0%+-0,2%
	88,0%+-1,5%	94,5%+-1,5%	38,5%+-1,5%	36,5%+-4,0%	28,5%+-0,5%	24,2%+-0,4%
NLPCA	98,2%+-2,5%	85,2%+-5,0%	33,2%+-0,5%	20,3%+-0,2%	15,3%+-0,1%	9,8%+-0,1%
	95,8%+-4,5%	95,0%+-3,0%	35,6%+-0,0%	28,4%+-1,0%	11,5%+-0,0%	12,5%+-0,1%
kPCA	99,1%+-1,5%	95,6%+-4,5%	63,5%+-2,2%	61,2%+-0,4%	26,2%+-0,1%	28,2%+-0,1%
	99,5%+-0,2%	91,9%+-5,0%	65,3%+-1,3%	59,3%+-1,1%	27,8%+-0,1%	29,3%+-0,1%
PCDi	93,9%+-2,0%	98,5%+-1,3%	85,4%+-3,0%	89,6%+-2,4%	57,4%+-0,1%	60,1%+-0,0%
	98,7%+-3,3%	94,1%+-2,6%	84,8%+-2,2%	90,1%+-1,4%	53,5%+-0,2%	50,3%+-0,0%
PCDc	94,1%+-4,5%	97,3%+-3,2%	89,9%+-3,3%	92,3%+-2,7%	59,3%+-0,1%	61,3%+-0,1%
	99,1%+-2,2%	94,3%+-0,5%	95,4%+-2,9%	90,1%+-0,6%	62,4%+-0,0%	58,9%+-0,2%

um nível baixo de exigência foram obtidos para máquinas de vector suporte alimentadas com dados compactados por 8 camadas de redes auto-associativas. Comparados com os resultados obtidos com dados que não foram pré-processados, redes neurais MLP alimentadas com dados processados por PCD, obtiveram valores de *trigger* mais baixos. *One-class SVM* alimentadas com dados processados com PCD, extraídos com método cooperativo, alcançaram os melhores resultados, tanto para níveis intermediários de exigência (50%-novidade), quanto para níveis mais altos de exigência (78%-novidade). Uma observação relevante é a maioria das técnicas obteve resultados melhores do que os da referência, ou seja dados sem compactação, o que motiva a aplicação de pré-processamentos para a extração de informações relevantes para a detecção de novidade em sinais de sonar passivo.

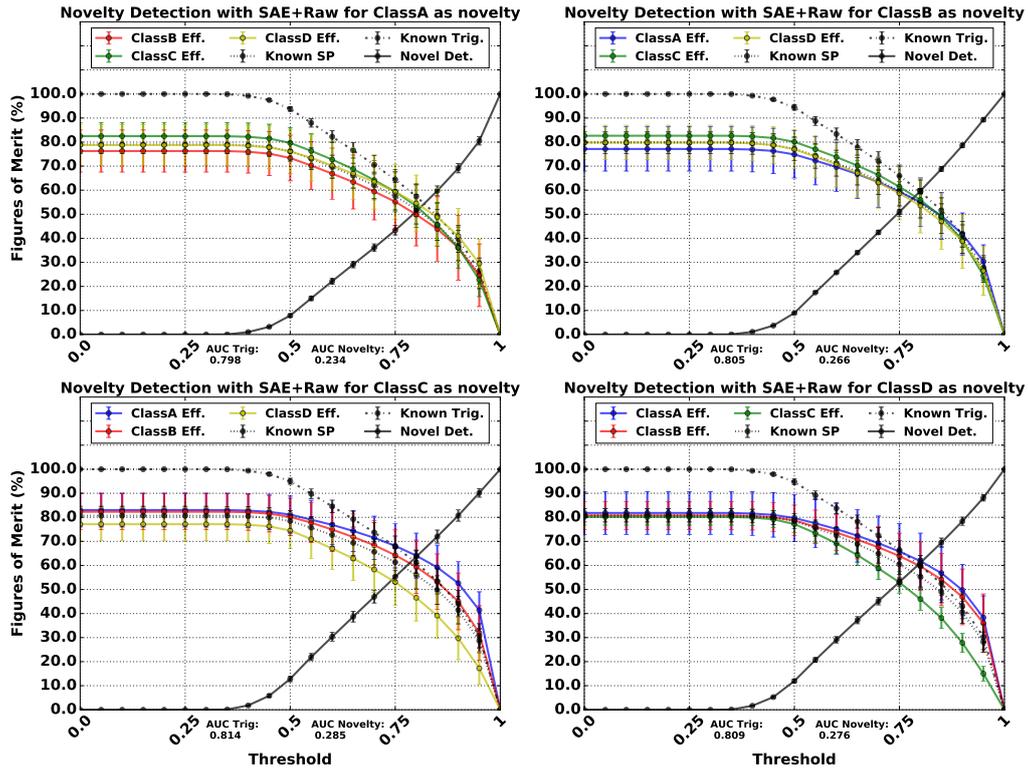


(a) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 1 camada. Topologia de inicialização dos pesos: 400 – 350 – 400

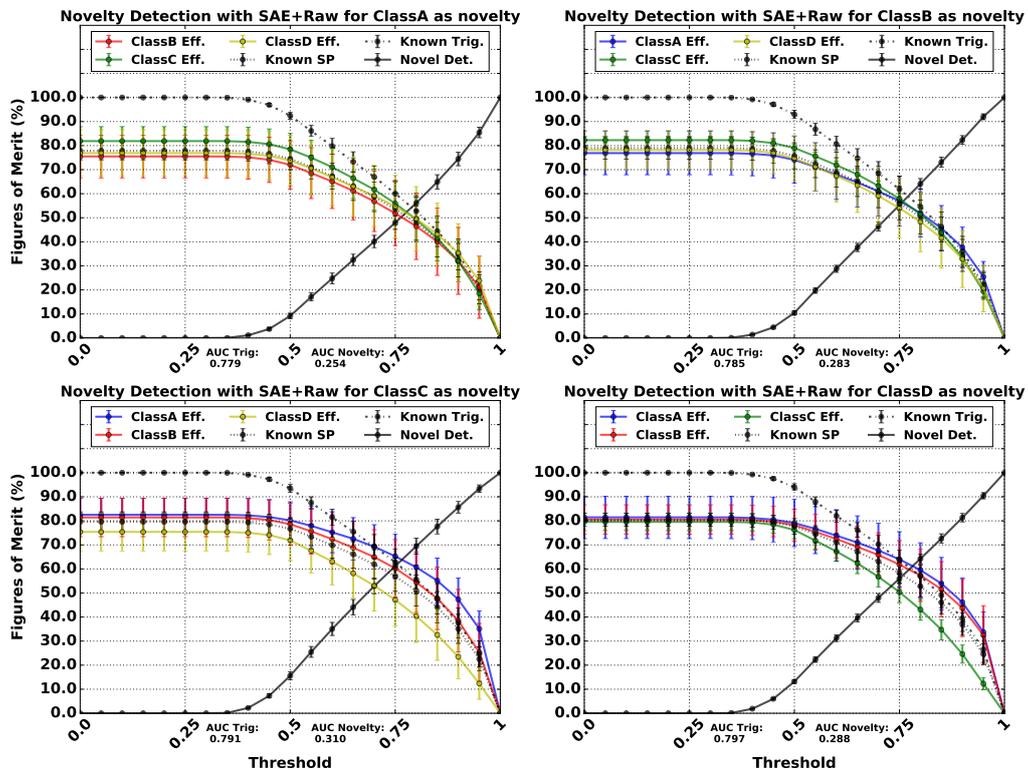


(b) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 2 camadas. Topologia de inicialização dos pesos: 400 – 350 – 300 – 350 – 400

Figura 5.21: Comparação dos resultados de detecção de novidade baseada em *stacked auto-encoders* com uma e duas camadas

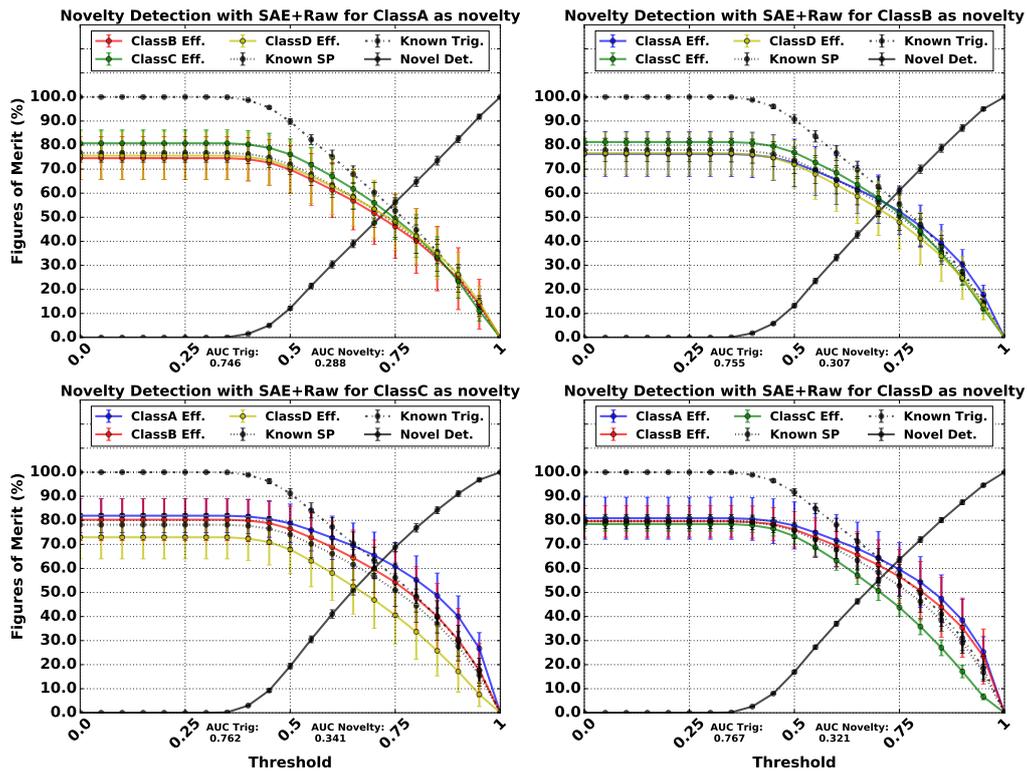


(a) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 3 camadas. Topologia de inicialização dos pesos: 400 – 350 – 300 – 250 – 300 – 350 – 400

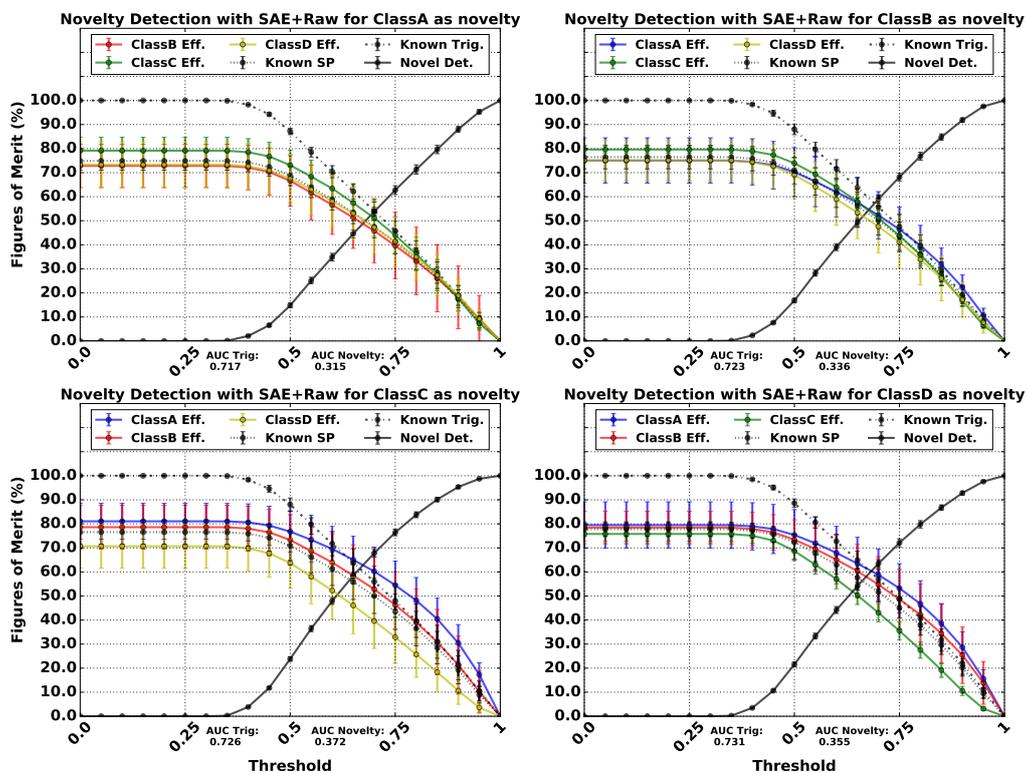


(b) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 4 camadas. Topologia de inicialização dos pesos: 400 – 350 – 300 – 250 – 200 – 250 – 300 – 350 – 400

Figura 5.22: Comparação dos resultados de detecção de novidade baseada em *stacked auto-encoders* com três e quatro camadas

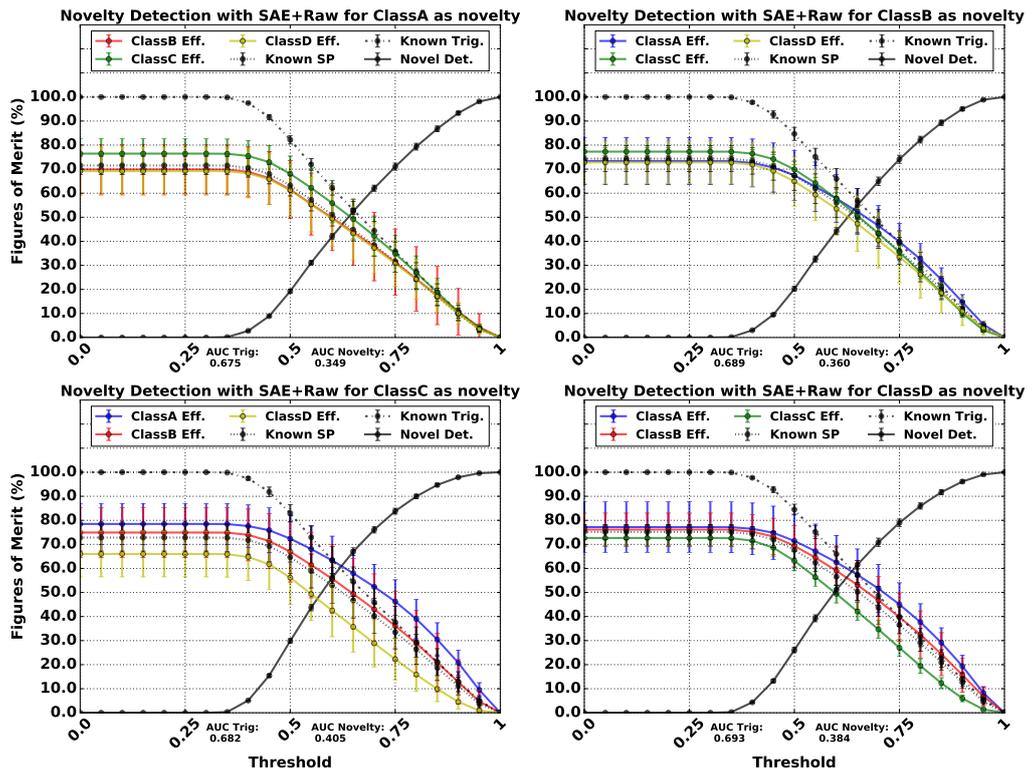


(a) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 5 camadas. Topologia de inicialização dos pesos: 400–350–300–250–200–150–200–250–300–350–400

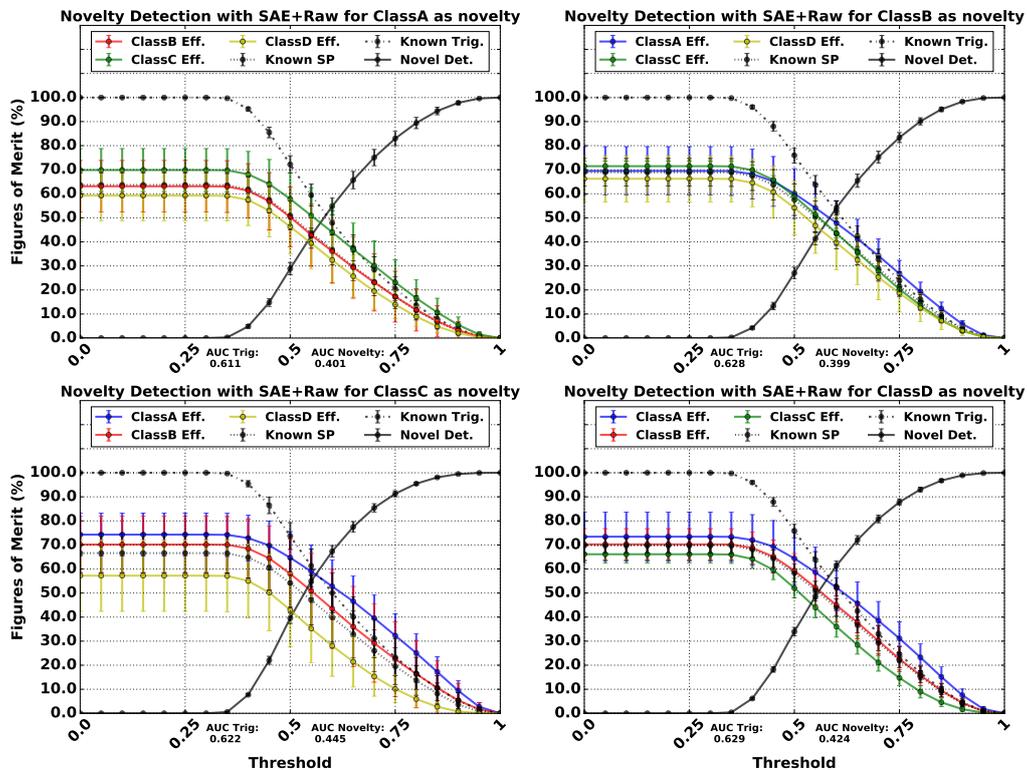


(b) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 6 camadas. Topologia de inicialização dos pesos: 400–350–300–250–200–150–100–150–200–250–300–350–400

Figura 5.23: Comparação dos resultados de detecção de novidade baseada em *stacked auto-encoders* com cinco e seis camadas



(a) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 7 camadas. Topologia de inicialização dos pesos:400 – 350 – 300 – 250 – 200 – 150 – 100 – 50 – 100 – 150 – 200 – 250 – 300 – 350 – 400



(b) Resultados para detecção de novidade baseada em um modelo neural inicializado com base em um *stacked auto-encoder* com 8 camadas. Topologia de inicialização dos pesos:400 – 350 – 300 – 250 – 200 – 150 – 100 – 50 – 25 – 50 – 100 – 150 – 200 – 250 – 300 – 350 – 400

Figura 5.24: Comparação dos resultados de detecção de novidade baseada em *stacked auto-encoders* com sete e oito camadas

Tabela 5.3: Comparação entre os resultados obtidos com diferentes quantidade de camadas de *Stacked AutoEncoders*, com a referência de leitura Tabela 5.1.

		<i>Trigger</i>					
Pre-proc	0%-Novidade		50%-Novidade		75%-Novidade		
Dados crus	95,2%+-7,9%	94,8%+-3,0%	45,5%+-8,9%	53,5%+-5,5%	22,5%+-0,5%	20,3%+-1,5%	
	97,5%+-0,5%	95,5%+-11,5%	61,5%+-4,5%	58,5%+-5,6%	18,8%+-1,0%	12,3%+-1,4%	
NLPCA	98,2%+-2,5%	85,2%+-5,0%	33,2%+-0,5%	20,3%+-0,2%	15,3%+-0,1%	9,8%+-0,1%	
	95,8%+-4,5%	95,0%+-3,0%	35,6%+-0,0%	28,4%+-1,0%	11,5%+-0,0%	12,5%+-0,1%	
1-SAE	99,9%+-0,0%	99,8%+-0,2%	44,8%+-1,3%	45,8%+-1,8%	17,7%+-0,2%	19,7%+-0,1%	
	99,8%+-0,3%	99,9%+-0,0%	47,3%+-2,3%	46,1%+-2,4%	16,7%+-0,1%	18,6%+-0,2%	
5-SAE	99,7%+-0,2%	99,8%+-0,2%	44,8%+-4,2%	45,8%+-3,3%	17,7%+-0,1%	19,7%+-0,1%	
	99,9%+-0,1%	99,7%+-0,1%	47,3%+-2,8%	46,1%+-3,8%	16,7%+-0,1%	18,6%+-0,0%	
8-SAE	99,8%+-0,1%	99,9%+-0,2%	50,1%+-3,0%	48,9%+-2,5%	28,9%+-0,1%	30,9%+-0,0%	
	99,8%+-0,3%	99,8%+-0,0%	52,4%+-3,4%	49,6%+-1,6%	24,5%+-0,2%	20,8%+-0,1%	

Tabela 5.4: Comparação entre os resultados obtidos com *Deep Learning* e diferentes pré-processamentos, com a referência de leitura Tabela 5.1.

		<i>Trigger</i>					
Método	0%-Novidade		50%-Novidade		75%-Novidade		
PCDc +MLP	90,0%+-1,0%	91,0%+-1,0%	69,0%+-1,0%	82,0%+-1,0%	50,0%+-1,0%	48,0%+-1,0%	
	92,0%+-1,0%	88,0%+-1,0%	77,0%+-1,0%	70,0%+-1,0%	49,0%+-1,0%	60,0%+-1,0%	
PCDi +MLP	92,0%+-1,0%	93,0%+-1,0%	80,0%+-1,0%	82,0%+-1,0%	48,0%+-1,0%	43,0%+-1,0%	
	94,0%+-1,0%	89,0%+-1,0%	83,0%+-1,0%	72,0%+-1,0%	44,0%+-1,0%	64,0%+-1,0%	
Dados crus+SVM	95,2%+-7,9%	94,8%+-3,0%	45,5%+-8,9%	53,5%+-5,5%	22,5%+-0,5%	20,3%+-1,5%	
	97,5%+-0,5%	95,5%+-11,5%	61,5%+-4,5%	58,5%+-5,6%	18,8%+-1,0%	12,3%+-1,4%	
kPCA +SVM	99,1%+-1,5%	95,6%+-4,5%	63,5%+-2,2%	61,2%+-0,4%	26,2%+-0,1%	28,2%+-0,1%	
	99,5%+-0,2%	91,9%+-5,0%	65,3%+-1,3%	59,3%+-1,1%	27,8%+-0,1%	29,3%+-0,1%	
PCDi +SVM	93,9%+-2,0%	98,5%+-1,3%	85,4%+-3,0%	89,6%+-2,4%	57,4%+-0,1%	60,1%+-0,0%	
	98,7%+-3,3%	94,1%+-2,6%	84,8%+-2,2%	90,1%+-1,4%	53,5%+-0,2%	50,3%+-0,0%	
PCDc +SVM	94,1%+-4,5%	97,3%+-3,2%	89,9%+-3,3%	92,3%+-2,7%	59,3%+-0,1%	61,3%+-0,1%	
	99,1%+-2,2%	94,3%+-0,5%	95,4%+-2,9%	90,1%+-0,6%	62,4%+-0,0%	58,9%+-0,2%	
8-SAE +SVM	99,8%+-0,1%	99,9%+-0,2%	50,1%+-3,0%	48,9%+-2,5%	28,9%+-0,1%	30,9%+-0,0%	
	99,8%+-0,3%	99,8%+-0,0%	52,4%+-3,4%	49,6%+-1,6%	24,5%+-0,2%	20,8%+-0,1%	

Capítulo 6

Conclusões e Trabalhos Futuros

O monitoramento e detecção de eventos que podem pertencer a uma classe desconhecida é de suma importância para um sistema on-line de classificação no ambiente de sonar passivo. Marinhas de outros países podem possuir navios que nunca entraram em contato com a Marinha do Brasil. Assim sendo, um detector de novidade que não tenha impacto significativo na eficiência de classificação se mostra extremamente importante, embora exista um compromisso entre o aumento da detecção de novidade e a eficiência de classificação.

A Marinha do Brasil tem priorizado, recentemente, a acústica submarina e tem investido em projetos que visam desenvolver tecnologia nacional. Tecnologia militar é bastante custosa para o seu desenvolvimento e, geralmente, a tecnologia de ponta não se encontra a venda por um simples motivo: nenhum país vai armar um potencial inimigo com sua melhor tecnologia. Ou seja, o desenvolvimento de tecnologia nacional se mostra de extrema importância.

Esta tese se desenvolveu em duas frentes: a análise de estacionaridade e a detecção de novidade. A primeira linha de trabalho busca explorar os efeitos da perda de estacionaridade para diferentes janelas temporais de processamento e propor um novo teste de estacionaridade em si. A segunda visa mostrar os resultados da detecção de novidade inserida em um processo de classificação de navios em um sistema de sonar passivo. Indo além, esta linha de trabalho buscou mostrar o impacto de um processo de detecção de novidade que possa operar ao mesmo tempo compartilhando recursos com um processo de classificação.

A análise de estacionaridade foi aplicada aos dados para garantir que os dados possuíssem a mesma função geratriz e, para se comprovar a presença de estacionaridade, dois testes foram aplicados: o ADF e o PP. À medida que o tamanho da janela de aquisição de dados aumenta, os testes têm a maior probabilidade de rejeitar a hipótese da janela ser estacionária. Os resultados obtidos pelas análises de estacionaridade iniciais apontam para uma redução no tamanho da janela de aquisição, mas se a janela se reduzir muito, a quantidade de informação que é trazida pela

mesma, pode ser insuficiente para a extração de características e treinamento de classificadores. Assim sendo, um treinamento neural para diferentes tamanho de janelas foi realizado para a análise do tamanho ideal de janelas de aquisição. Os valores obtidos como sendo ideais para maximizar o poder discriminatório dos dados foram 1.024 e 2.048.

Além disso, foi aplicado um novo teste baseado na divergência KL. Este novo teste foi sensível a janelas que pertenciam a parte final das corridas dos navios. A parte final da corrida é a parte onde o navio se aproxima mais do sensor e, por isso, ocorre uma maior variação da intensidade sonora que está sendo capitada. Assim sendo, a probabilidade de que duas janelas consecutivas possuam a mesma função geratriz diminui consideravelmente. Ou seja, o teste se mostrou sensível a um comportamento que pode tornar o processo de classificação mais desafiador.

A detecção de novidade que foi apresentada neste trabalho se baseou em duas técnicas de classificação básicas: redes neurais e máquinas de vector suporte. À primeira, foi aplicada uma modificação para aumentar o poder de classificação e para a operação da segunda, um método de treinamento foi utilizado visando especificamente a detecção de novidade e o encapsulamento de dados.

Além disso, para avaliar o impacto de diferentes mapeamentos nos dados, foram propostos treinamento de máquinas de vector suporte voltadas para o encapsulamento de dados processados através de diferentes pré-processamentos como PCA, kPCA, NLPCA, PCD cooperativa e PCD por deflação. Para os dados processados pela PCA, os resultados mostraram que, aproximadamente, 270 dos primeiros componentes retêm 90% da energia estatística dos dados, pode-se também observar que o resultado para a detecção de novidade apresentou um comportamento muito próximo a um comportamento diretamente proporcional ao valor de ν (que é a variável que controla a ocorrência e a intensidade de *overtraining*). Outro comportamento que pode ser observado é que as eficiências de classificação foram muito próximas entre si, mesmo quando houve uma variação da classe considerada novidade.

Para dados processados com kPCA, uma análise da variação na eficiência de classificação que é feita à medida que são adicionados componentes foi utilizada para a determinação da quantidade ideal de componentes a serem extraídos e, a partir dessa análise, foram definidos 30 componentes. No que tange a detecção de novidade, foi observado um comportamento não-linear entre os valores de ν e a detecção de novidade. Para definir a quantidade de NLPCA a serem extraídos, a mesma análise aplicada a kPCA foi utilizada, obtendo 40 componentes para obter o máximo de eficiência de classificação e os resultados obtidos para detecção foram basicamente da mesma ordem de grandeza dos obtidos com kPCA.

Técnicas de pré-processamento que visam a reconstrução dos dados de entrada,

como PCA, kPCA e NLPCA, projetam os dados em um espaço que pode dificultar o processo de classificação. Por esse motivo, um processo de pré-processamento que visa a classificação pode facilitar a tomada de decisão, esse é o princípio de operação da PCD e dois tipos de PCD foram aplicados aos dados. A quantidade de componentes foi estimada visando maximizar a eficiência de classificação e foi definida como 40 PCD para os dois tipos de PCD. No âmbito da detecção de novidade, as PCD apresentaram um resultado interessante para o *trigger*, aumentando o valor desta medida de desempenho. Este fato se deve ao mapeamento dos dados conhecidos no espaço gerado pela PCD que melhora o poder discriminatório do detector de novidade, aumentando, assim, a eficiência em classificar dados conhecidos, embora esse mapeamento, de maneira geral, não tenha aumentado a eficiência de classificação dos classificadores de classes conhecidas.

Outro resultado que foi obtido durante o desenvolvimento do trabalho foi a detecção de novidade baseada em uma das técnicas mais consagradas de *deep learning*, os chamados *Stacked Auto-Encoders*. Estes se baseiam no mapeamento não-linear dos dados de entradas através de camadas de reconstrução. Os resultados obtidos foram comparáveis aos obtidos com detectores baseados em SVM e um resultado interessante é que a medida que a quantidade de camadas aumentava, maior era a eficiência na detecção de novidade, em compensação, para modelos com mais camadas houve uma perda na eficiência de classificação.

6.1 Trabalhos Futuros

O mapeamento dos dados em diferentes espaços pode ajudar a detecção de novidade, como foi observado no caso das PCD. Devido a isso, uma análise do poder de mapeamento dos *Stacked Auto-Encoders* deve ser desenvolvida. Além disso, como as técnicas de *Deep Learning* pressupõem pouco ou nenhum conhecimento especialista dos dados de entrada, então as mesmas não fazem uso de pré-processamento nos dados de entrada e geralmente, processa os dados dentro de sua estrutura profunda. Uma análise que pode ser realizada é uma comparação entre o resultado para a detecção de novidade de uma técnica clássica, como SVM ou rede neural, somada a um pré-processamento que visa classificação, como PCD e o resultado, também para a detecção de novidade, de uma técnica de aprendizado profundo.

A informação de sistemas de sonar passivo, naturalmente, possui uma estrutura temporal que varia ao longo do tempo. Atualmente, em *Deep Learning*, duas técnicas visam acessar informações temporais, a CNN e a RNN. A primeira foi criada para o processamento de imagens e pode ser utilizada para o processamento dos sinais provenientes de sinais de sonar passivo através da concatenação das informações de uma quantidade de janelas consecutivas. A segunda, acessa a informação temporal

dentro de sua estrutura e pode ser utilizada para processar os dados dos áudios criando um espaço que é chamado de contexto e, assim, fazer a detecção de novidade. A aplicação destas técnicas de *Deep Learning* pode vir a trazer um ganho substancial no que diz respeito à detecção de novidade em sistemas de sonar passivo.

No contexto da análise de estacionariedade, uma medida deve ser definida para o teste baseado na divergência KL e os testes baseados na extração de características devem ser explorados mais profundamente.

Referências Bibliográficas

- [1] DE OLIVEIRA, N. “Área territorial do Brasil aumenta em 890 Km^2 após atualização do IBGE”. Abril 2013. Disponível em: <http://www.ebc.com.br/noticias/brasil/2013/01/area-territorial-do-brasil-teve-incremento-de-001-em-atualizacao-divulgada>.
- [2] DE GEOGRAFIA E ESTATÍSTICA, I. B. “Área Territorial Brasileira”. Julho 2016. Disponível em: https://ww2.ibge.gov.br/home/geociencias/cartografia/default_territ_area.shtm.
- [3] DOS DEPUTADOS, C. “Legislação Informatizada - DECRETO Nº 99.165, DE 12 DE MARÇO DE 1990 - Publicação Original”. Março 1990. Disponível em: <http://www2.camara.leg.br/legin/fed/decret/1990/decreto-99165-12-marco-1990-328535-publicacaooriginal-1-pe.html>.
- [4] DO BRASIL, M. “Marinha do Brasil comemora Dia Nacional da Amazônia Azul”. Disponível em: <http://www.naval.com.br/blog/2016/11/16/marinha-do-brasil-comemora-dia-nacional-da-amazonia-azul/>.
- [5] CAVALCANTE, F. “Brasil foi alvo de mesma manobra naval dos EUA que irritou China”. Disponível em: <http://www1.folha.uol.com.br/mundo/2015/10/1700336-brasil-foi-alvo-de-ato-naval-semelhante-ao-que-os-eua-realizaram-na-china.shtml>.
- [6] DO BRASIL, M. “Esquadra Brasileira”. Setembro 2017. Disponível em: https://www.marinha.mil.br/sites/default/files/silhueta_internet.pdf.
- [7] DA DEFESA, M. “Projeto SisGAAz”.
- [8] DA DEFESA, M. “Programa de Desenvolvimento de Submarinos: submarino com propulsão nuclear, vamos construí-lo”.

- [9] URICK, R. *Principles of Underwater Sound for Engineers*. McGraw-Hill, 1983.
- [10] YUSOF, M. A. B., KABIR, S. “An Overview of Sonar and Electromagnetic Waves for Underwater Communication”, *IETE Technical Review*, v. 29, n. 4, pp. 307–317, 2012. doi: 10.4103/0256-4602.101312.
- [11] XU, L., XU, T. *Digital Underwater Acoustic Communications*. Academic Press, 2016.
- [12] AINSLIE, M. *Principles of Sonar Performance Modelling*. Springer-Verlag Berlin Heidelberg, 2010.
- [13] BURDIC, W. S. *Underwater Acoustic System Analysis*. Peninsula Pub, 2003.
- [14] NIELSEN, R. O. *Sonar Signal Processing*. Norwood, MA, USA, Artech House, Inc., 1991. ISBN: 0890064539.
- [15] BLOUIN, S. “Simultaneous Design of Underwater Acoustic Sensor and Communication Networks”. In: *Proceedings of the International Conference on Underwater Networks & Systems, WUWNET’17*, pp. 16:1–16:5, New York, NY, USA, 2017. ACM. ISBN: 978-1-4503-5561-2. doi: 10.1145/3148675.3148676.
- [16] LI, Q. *Digital Sonar Design in Underwater Acoustics*. Springer-Verlag Berlin Heidelberg, 2012.
- [17] MOHRI, M., ROSTAMIZADEH, A., TALWALKAR, A. *Foundations of Machine Learning*. The MIT Press, 2012. ISBN: 026201825X, 9780262018258.
- [18] WITTEN, I. H., FRANK, E. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems)*. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc., 2005. ISBN: 0120884070.
- [19] SAYED-MOUCHAWEH, M., LUGHOFER, E. *Learning in Non-Stationary Environments: Methods and Applications*. Springer-Verlag New York, 2012.
- [20] PEEBLES, P. *Probability, Random Variables, and Random Signal Principles*. McGraw-Hill Science/Engineering/Math, 2000.
- [21] DA DEFESA, M. “Estratégia Nacional de Defesa”. .
- [22] LAWRENCE, N. D. “A Unifying Probabilistic Perspective for Spectral Dimensionality Reduction: Insights and New Models”, *J. Mach. Learn. Res.*, v. 13, n. 1, pp. 1609–1638, maio 2012. ISSN: 1532-4435.

- [23] IZENMAN, A. J. “Introduction to manifold learning”, *Wiley Interdisciplinary Reviews: Computational Statistics*, v. 4, n. 5, pp. 439–446, 2012. ISSN: 1939-0068. doi: 10.1002/wics.1222.
- [24] BROSCHE, T., TAM, R., INITIATIVE, A. D. N., et al. “Manifold learning of brain MRIs by deep learning”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2013.
- [25] SCHOLKOPF, B., SMOLA, A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA, MIT Press, 2001. ISBN: 0262194759.
- [26] HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. 2nd ed. Upper Saddle River, NJ, USA, Prentice Hall PTR, 1998. ISBN: 0132733501.
- [27] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep Learning*. MIT Press, 2016.
- [28] SHUMWAY, R. H., STOFFER, D. S. *Time Series Analysis and Its Applications (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., 2005.
- [29] PEEBLES, P. Z. *Probability, Random Variables and Random Signal Principles*. McGraw-Hill Education (India) Pvt Limited, 2002.
- [30] SONDHAUS, L. *The Great War at Sea: A Naval History of the First World War*. Cambridge University Press, 2014.
- [31] FILHO, W. S. *Classificação do ruído irradiado por navios usando redes neurais*. Tese de Doutorado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2001.
- [32] BUBLIC, I., SKELIN, D., VUKADIN, P. “Measurement of vessel underwater noise signature”. In: *2008 50th International Symposium ELMAR*, 2008.
- [33] LIANG, C. “Simulating vessel’s underwater radiated noise in spectrum based on actual measurement data”. In: *2016 IEEE/OES China Ocean Acoustics (COA)*, 2016.
- [34] XIROS, N. *Robust Control of Diesel Ship Propulsion*. Springer-Verlag London, 2002.
- [35] MAERLI, M. B. “Components of Naval Nuclear Fuel Transparency”. NATO-EAPC Fellowship Report, 2001.

- [36] FAGIANO, L., MILANESE, M., RAZZA, V., et al. “Control of power kites for naval propulsion”. In: *Proceedings of the 2010 American Control Conference*, 2010.
- [37] YUFANG, Z., PENG, Q., HONG, Y. “Fuel free ship, design for next generation”. In: *2013 Eighth International Conference and Exhibition on Ecological Vehicles and Renewable Energies (EVER)*, 2013.
- [38] RAHMAN, A., KARIM, M. M. “Overview of Alternative Fuels and Their Drivers to Reduce Emissions in the Shipping Industry”. In: *International Conference on Mechanical And Industrial Engineering, ICMAIE'2015*, 2015.
- [39] BABU, M., SUBRAMANIAN, K. *Alternative Transportation Fuels: Utilisation in Combustion Engines*. CRC Press, 2013. ISBN: 9781439872826. Disponível em: <<https://books.google.com.br/books?id=5K7MBQAAQBAJ>>.
- [40] WIKIPEDIA. “Cavitation”. 2016 - Acessado em Maio de 2016. Disponível em: <<https://en.wikipedia.org/wiki/Cavitation>>.
- [41] DOS SANTOS, F. L. *Auxílio à classificação sonar passiva por tons característicos*. Tese de Mestrado, Programa de Engenharia Naval e Oceânica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2005.
- [42] STERGIOPOULOS, S. *Advanced Signal Processing Handbook: Theory and Implementation for Radar, Sonar, and Medical Imaging Real-Time Systems*. 1st ed. Boca Raton, FL, USA, CRC Press, Inc., 2000. ISBN: 0849336910.
- [43] STEELE, A. K., BYRNE, C. L., RILEY, J. L., et al. “Performance comparison of high resolution bearing estimation algorithms using simulated and sea test data”, *IEEE Journal of Oceanic Engineering*, 1993.
- [44] HERBORDT, W., KELLERMANN, W. “Efficient frequency-domain realization of robust generalized, sidelobe cancellers”. In: *2001 IEEE Fourth Workshop on Multimedia Signal Processing (Cat. No.01TH8564)*, 2001.
- [45] JABLON, N. “Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections”, *IEEE Transactions on Antennas and Propagation*, 1986.

- [46] BUCKLEY, K., GRIFFITHS, L. “An adaptive generalized sidelobe canceller with derivative constraints”, *IEEE Transactions on Antennas and Propagation*, 1986.
- [47] LI, J., STOICA, P. *Robust Adaptive Beamforming*. Wiley Series in Telecommunications and Signal Processing. Wiley, 2005. ISBN: 9780471733461. Disponível em: <https://books.google.com.br/books?id=a98fGhHNo_kC>.
- [48] FELZKY, M. A. *Uma contribuição às técnicas de localização de fontes sonoras através de um sistema sonar passivo utilizando filtros fracionários*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2007.
- [49] DE SÁ, F. P. G. *Conformação de Feixes em Sonar Passivo Baseada em Invariância na Frequência*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2016.
- [50] DE ABREU BOZZI, F. *Conformação de Feixe em Sonar Passivo para um Arranjo Cilíndrico de Hidrofonos*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2016.
- [51] DE MOURA, N. N. *Deteção e Classificação de Sinais de Sonar Passivo Usando Métodos de Separação Cega de Fontes*. Tese de Doutorado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2013.
- [52] RAMOS, R. R. *Redes Neurais para Monitoração de Estacionariedade de Sinais de Sonar Passivo*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2011.
- [53] DE OLIVEIRA E SOUZA FILHO, J. B. *Análise de Componentes Principais em Sistemas de Sonar*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2002.
- [54] DE OLIVEIRA E SOUZA FILHO, J. B. *Classificação Neural de Sinais de Sonar Passivo*. Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, 2007.

- [55] FILHO, J. B. O. S., DE SEIXAS, J. M. “Class-modular multi-layer perceptron networks for supporting passive sonar signal classification”, *IET Radar, Sonar Navigation*, 2016.
- [56] FILHO, W. S., SEIXAS, J. M. D., MOURA, N. N. D. “Preprocessing passive sonar signals for neural classification”, *IET Radar, Sonar Navigation*, 2011.
- [57] SEIXAS, J. M., DAMAZIO, D. O., DINIZ, P. S. R., et al. “Wavelet transform as a preprocessing method for neural classification of passive sonar signals”, *ICECS 2001. 8th IEEE International Conference on Electronics, Circuits and Systems (Cat. No.01EX483)*, 2001.
- [58] PIRKL, R. J., YOCOM, B. A., AUGHENBAUGH, J. M. “Bayesian broadband passive sonar tracking”, *2015 18th International Conference on Information Fusion (Fusion)*, 2015.
- [59] PIRKL, R. J., AUGHENBAUGH, J. M. “Bayesian passive sonar tracking with conventional beamformer-level data”. In: *2015 18th International Conference on Information Fusion (Fusion)*, 2015.
- [60] YOCOM, B. A., COUR, B. R. L., YUDICHAK, T. W. “A Bayesian Approach to Passive Sonar Detection and Tracking in the Presence of Interferers”, *IEEE Journal of Oceanic Engineering*, 2011.
- [61] YOCOM, B. A., AUGHENBAUGH, J. M., COUR, B. R. L. “Range-sensitive Bayesian passive sonar tracking”. In: *2010 13th International Conference on Information Fusion*, 2010.
- [62] JOSEPH, V. O., AMBAT, S. K., GEORGE, J. “Design and implementation of a novel real time target tracking scheme for passive SONARs using MVDR beam forming and Kalman filtering”. In: *2009 International Symposium on Ocean Electronics (SYMPOL 2009)*, 2009.
- [63] YOCOM, B. A., YUDICHAK, T. W., COUR, B. R. L. “Bayesian passive sonar tracking in the presence of known interferers”. In: *2009 12th International Conference on Information Fusion*, 2009.
- [64] PERCIVAL, D. B., WALDEN, A. T. *Wavelet Methods for Time Series Analysis (Cambridge Series in Statistical and Probabilistic Mathematics)*. Cambridge University Press, 2006.

- [65] PRESS, W. H., TEUKOLSKY, S. A., VETTERLING, W. T., et al. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.
- [66] DAS, A., KUMAR, A., BAHL, R. “Marine vessel classification based on passive sonar data: the cepstrum-based approach”, *IET Radar, Sonar Navigation*, 2013.
- [67] CASSELMAN, F. L., FREEMAN, D. F., KERRIGAN, D. A., et al. “A Neural Network-based Passive Sonar Detection And Classification Design With A Low False Alarm Rate”, *OCEANS 91 Proceedings*, 1991.
- [68] KAMAL, S., MOHAMMED, S. K., PILLAI, P. R. S., et al. “Deep learning architectures for underwater target recognition”, *2013 Ocean Electronics (SYMPOL)*, 2013.
- [69] CHAZAN, D., HOORY, R., COHEN, G., et al. “Speech reconstruction from mel frequency cepstral coefficients and pitch frequency”. In: *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, 2000.
- [70] ZHOU, X., GARCIA-ROMERO, D., DURAISWAMI, R., et al. “Linear versus mel frequency cepstral coefficients for speaker recognition”. In: *2011 IEEE Workshop on Automatic Speech Recognition Understanding*, 2011.
- [71] TYAGI, V., MCCOWAN, I., MISRA, H., et al. “Mel-cepstrum modulation spectrum (MCMS) features for robust ASR”. In: *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No.03EX721)*, 2003.
- [72] BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA, Springer-Verlag New York, Inc., 2006. ISBN: 0387310738.
- [73] FERGUSON, K. *THE NOBLEMAN AND HIS HOUSEDOG. Tycho Brahe and Johannes Kepler: The Strange Partnership that revolutionised Science*. Review, 2002.
- [74] BARBER, D. *Bayesian Reasoning and Machine Learning*. New York, NY, USA, Cambridge University Press, 2012. ISBN: 0521518148, 9780521518147.
- [75] *Encyclopedia of Machine Learning*. Springer US, 2010.

- [76] PIMENTEL, M. A., CLIFTON, D. A., CLIFTON, L., et al. “A review of novelty detection”, *Signal Processing*, v. 99, pp. 215 – 249, 2014. ISSN: 0165-1684. doi: <http://dx.doi.org/10.1016/j.sigpro.2013.12.026>. Disponível em: <http://www.sciencedirect.com/science/article/pii/S016516841300515X>.
- [77] HODGE, V., AUSTIN, J. “A Survey of Outlier Detection Methodologies”, *Artif. Intell. Rev.*, v. 22, n. 2, pp. 85–126, out. 2004. ISSN: 0269-2821. doi: 10.1023/B:AIRE.0000045502.10941.a9. Disponível em: <http://dx.doi.org/10.1023/B:AIRE.0000045502.10941.a9>.
- [78] SCHOLKOPF, B., PLATT, J. C., SHAWE-TAYLOR, J. C., et al. “Estimating the Support of a High-Dimensional Distribution”, *Neural Comput.*, v. 13, n. 7, pp. 1443–1471, jul. 2001. ISSN: 0899-7667.
- [79] TARASSENKO, L., HAYTON, P., CERNEAZ, N., et al. “Novelty detection for the identification of masses in mammograms”. In: *1995 Fourth International Conference on Artificial Neural Networks*, 1995.
- [80] PATCHA, A., PARK, J.-M. “An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends”, *Comput. Netw.*, v. 51, n. 12, pp. 3448–3470, ago. 2007. ISSN: 1389-1286. doi: 10.1016/j.comnet.2007.02.001. Disponível em: <http://dx.doi.org/10.1016/j.comnet.2007.02.001>.
- [81] DOMINGUES, R., BUONORA, F., SENESI, R., et al. “An Application of Unsupervised Fraud Detection to Passenger Name Records”. In: *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W)*, 2016.
- [82] GARCÍA-TEODORO, P., DÍAZ-VERDEJO, J., MACIÁ-FERNÁNDEZ, G., et al. “Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges”, *Comput. Secur.*, v. 28, n. 1-2, pp. 18–28, fev. 2009. ISSN: 0167-4048. doi: 10.1016/j.cose.2008.08.003. Disponível em: <http://dx.doi.org/10.1016/j.cose.2008.08.003>.
- [83] BASU, S., BILENKO, M., MOONEY, R. J. “A Probabilistic Framework for Semi-supervised Clustering”. In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pp. 59–68, New York, NY, USA, 2004. ACM. ISBN: 1-58113-888-1. doi: 10.1145/1014052.1014062. Disponível em: <http://doi.acm.org/10.1145/1014052.1014062>.

- [84] MARKOU, M., SINGH, S. “Novelty Detection: A Review&Mdash;Part 1: Statistical Approaches”, *Signal Process.*, v. 83, n. 12, pp. 2481–2497, dez. 2003. ISSN: 0165-1684. doi: 10.1016/j.sigpro.2003.07.018. Disponível em: <<http://dx.doi.org/10.1016/j.sigpro.2003.07.018>>.
- [85] MARKOU, M., SINGH, S. “Novelty Detection: A Review&Mdash;Part 2: Neural Network Based Approaches”, *Signal Process.*, v. 83, n. 12, pp. 2499–2521, dez. 2003. ISSN: 0165-1684. doi: 10.1016/j.sigpro.2003.07.019. Disponível em: <<http://dx.doi.org/10.1016/j.sigpro.2003.07.019>>.
- [86] FILZMOSER, P., MARONNA, R., WERNER, M. “Outlier Identification in High Dimensions”, *Comput. Stat. Data Anal.*, v. 52, n. 3, pp. 1694–1711, jan. 2008. ISSN: 0167-9473. doi: 10.1016/j.csda.2007.05.018. Disponível em: <<http://dx.doi.org/10.1016/j.csda.2007.05.018>>.
- [87] HE, H., GARCIA, E. A. “Learning from Imbalanced Data”, *IEEE Transactions on Knowledge and Data Engineering*, v. 21, n. 9, pp. 1263–1284, September 2009.
- [88] LEE, H.-J., CHO, S. “The Novelty Detection Approach for Different Degrees of Class Imbalance”. In: King, I., Wang, J., Chan, L.-W., et al. (Eds.), *Neural Information Processing: 13th International Conference, ICONIP 2006, Hong Kong, China, October 3-6, 2006. Proceedings, Part II*, pp. 21–30, Berlin, Heidelberg, Springer Berlin Heidelberg, 2006. ISBN: 978-3-540-46482-2. doi: 10.1007/11893257_3. Disponível em: <http://dx.doi.org/10.1007/11893257_3>.
- [89] BARNETT, V., LEWIS, T. *Outliers in Statistical Data*. Wiley, 1994.
- [90] WHITACRE, J. M., PHAM, T. Q., SARKER, R. A. “Use of Statistical Outlier Detection Method in Adaptive Evolutionary Algorithms”. In: *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, GECCO '06*, pp. 1345–1352, New York, NY, USA, 2006. ACM. ISBN: 1-59593-186-4. doi: 10.1145/1143997.1144205. Disponível em: <<http://doi.acm.org/10.1145/1143997.1144205>>.
- [91] SOLBERG, H. E., LAHTI, A. “Detection of Outliers in Reference Distributions: Performance of Horn’s Algorithm”, *Clinical Chemistry*, v. 51, n. 12, pp. 2326–2332, 2005. ISSN: 0009-9147. doi: 10.1373/clinchem.2005.058339. Disponível em: <<http://clinchem.aaccjnls.org/content/51/12/2326>>.

- [92] HAUTAMAKI, V., KARKKAINEN, I., FRANTI, P. “Outlier detection using k-nearest neighbour graph”. In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2004.
- [93] CHAWLA, S., SUN, P. “SLOM: a new measure for local spatial outliers”, *Knowledge and Information Systems*, v. 9, n. 4, pp. 412–429, 2006. ISSN: 0219-3116. doi: 10.1007/s10115-005-0200-2. Disponível em: <<http://dx.doi.org/10.1007/s10115-005-0200-2>>.
- [94] ZHANG, J., WANG, H. “Detecting outlying subspaces for high-dimensional data: the new task, algorithms, and performance”, *Knowledge and Information Systems*, v. 10, n. 3, pp. 333–355, 2006. ISSN: 0219-3116. doi: 10.1007/s10115-006-0020-z. Disponível em: <<http://dx.doi.org/10.1007/s10115-006-0020-z>>.
- [95] POKRAJAC, D., LAZAREVIC, A., LATECKI, L. J. “Incremental Local Outlier Detection for Data Streams.” In: *CIDM*, 2007.
- [96] CROOK, P. A., MARSLAND, S., HAYES, G., et al. “A tale of two filters-online novelty detection”. In: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, 2002.
- [97] SOARES-FILHO, W., SEIXAS, J. M., CALÔBA, L. P. “Enlarging neural class detection capacity in passive sonar systems”. In: *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, v. 3, pp. III–105–III–108 vol.3, 2002.
- [98] HAWKINS, S., HE, H., WILLIAMS, G., et al. “Outlier Detection Using Replicator Neural Networks”. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (Eds.), *Data Warehousing and Knowledge Discovery: 4th International Conference, DaWaK 2002 Aix-en-Provence, France, September 4–6, 2002 Proceedings*, pp. 170–180, Berlin, Heidelberg, Springer Berlin Heidelberg, 2002. ISBN: 978-3-540-46145-6. doi: 10.1007/3-540-46145-0_17. Disponível em: <http://dx.doi.org/10.1007/3-540-46145-0_17>.
- [99] THOMPSON, B. B., MARKS, R. J., CHOI, J. J., et al. “Implicit learning in autoencoder novelty assessment”. In: *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, 2002.
- [100] TAX, D. M., DUIN, R. P. “Support Vector Data Description”, *Machine Learning*, v. 54, n. 1, pp. 45–66, 2004. ISSN: 1573-0565. doi: 10.1023/B:MACH.0000008084.60811.49. Disponível em: <<http://dx.doi.org/10.1023/B:MACH.0000008084.60811.49>>.

- [101] SCHÖLKOPF, B., WILLIAMSON, R., SMOLA, A., et al. “Support Vector Method for Novelty Detection”. In: *Proceedings of the 12th International Conference on Neural Information Processing Systems, NIPS’99*, pp. 582–588, Cambridge, MA, USA, 1999. MIT Press. Disponível em: <<http://dl.acm.org/citation.cfm?id=3009657.3009740>>.
- [102] HE, Z., DENG, S., XU, X. “An Optimization Model for Outlier Detection in Categorical Data”. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (Eds.), *Advances in Intelligent Computing: International Conference on Intelligent Computing, ICIC 2005, Hefei, China, August 23-26, 2005, Proceedings, Part I*, pp. 400–409, Berlin, Heidelberg, Springer Berlin Heidelberg, 2005. ISBN: 978-3-540-31902-3. doi: 10.1007/11538059_42. Disponível em: <http://dx.doi.org/10.1007/11538059_42>.
- [103] ANDO, S. “Clustering Needles in a Haystack: An Information Theoretic Analysis of Minority and Outlier Detection”. In: *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, 2007.
- [104] FILIPPONE, M., SANGUINETTI, G. “Information Theoretic Novelty Detection”, *Pattern Recogn.*, v. 43, n. 3, pp. 805–814, mar. 2010. ISSN: 0031-3203. doi: 10.1016/j.patcog.2009.07.002. Disponível em: <<http://dx.doi.org/10.1016/j.patcog.2009.07.002>>.
- [105] LECOMTE, S., LENGELLÉ, R., RICHARD, C., et al. “Abnormal events detection using unsupervised One-Class SVM - Application to audio surveillance and evaluation”. In: *2011 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 124–129, August 2011.
- [106] GUERBAI, Y., CHIBANI, Y., HADJADJI, B. “Writer-independent Handwritten Signature Verification based on One-Class SVM classifier”. In: *2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 327–331, July 2014.
- [107] SHANG, W., LI, L., WAN, M., et al. “Industrial communication intrusion detection algorithm based on improved one-class SVM”. In: *2015 World Congress on Industrial Control Systems Security (WCICSS)*, pp. 21–25, December 2015.
- [108] AMIR, R. B., GUL, S. T., KHAN, A. Q. “A comparative analysis of classical and one class SVM classifiers for machine fault detection using vibration signals”. In: *2016 International Conference on Emerging Technologies (ICET)*, pp. 1–6, October 2016.

- [109] BURNAEV, E., SMOLYAKOV, D. “One-Class SVM with Privileged Information and Its Application to Malware Detection”. In: *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pp. 273–280, December 2016.
- [110] CANDÈS, E., DONOHO, D. *Curvelets: A Surprisingly Effective Nonadaptive Representation of Objects with Edges*. Relatório técnico, 1999.
- [111] MIAO, Q., LIU, J., CAO, Y., et al. “Malware detection using bilayer behavior abstraction and improved one-class support vector machines”, *International Journal of Information Security*, v. 15, n. 4, pp. 361–379, 2016.
- [112] CATAL, C. “Review: Software Fault Prediction: A Literature Review and Current Trends”, *Expert Syst. Appl.*, v. 38, n. 4, pp. 4626–4636, abr. 2011.
- [113] CHEN, L., FANG, B., SHANG, Z. “Software fault prediction based on one-class SVM”. In: *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, v. 2, pp. 1003–1008, July 2016.
- [114] DE MOURA JUNIOR, N. N., DE SEIXAS, J. M. “Novelty detection in passive SONAR systems using support vector machines”. In: *2015 Latin America Congress on Computational Intelligence (LA-CCI)*, 2015.
- [115] STOLFO, J., FAN, W., LEE, W., et al. “Cost-based modeling and evaluation for data mining with application to fraud and intrusion detection”, *Results from the JAM Project by Salvatore*, pp. 1–15, 2000.
- [116] WOLBERG, W. H., STREET, W. N., MANGASARIAN, O. “Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates”, *Cancer letters*, v. 77, n. 2-3, pp. 163–171, 1994.
- [117] KIM, S. J., ZYMNIS, A., MAGNANI, A., et al. “Learning the kernel via convex optimization”. In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008.
- [118] PELCKMANS, K., SUYKENS, J. A. K., MOOR, B. D. “Regularization constants in LS-SVMs: a fast estimate via convex optimization”. In: *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, 2004.
- [119] TIAN, D., LIU, Y., SHI, J. “DYNAMIC CLUSTERING ALGORITHM BASED ON ADAPTIVE RESONANCE THEORY”. In: LIU, G., TAN,

- V., HAN, X. (Eds.), *Computational Methods*, pp. 1239–1248, Dordrecht, Springer Netherlands, 2006. ISBN: 978-1-4020-3953-9. doi: 10.1007/978-1-4020-3953-9_35. Disponível em: <http://dx.doi.org/10.1007/978-1-4020-3953-9_35>.
- [120] AUGUSTEIJN, M. F., FOLKERT, B. A. “Neural network classification and novelty detection”, *International Journal of Remote Sensing*, v. 23, n. 14, pp. 2891–2902, 2002. doi: 10.1080/01431160110055804.
- [121] DIAZ, I., HOLLMEN, J. “Residual generation and visualization for understanding novel process conditions”. In: *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, pp. 2070–2075, 2002.
- [122] JAPKOWICZ, N. “Supervised Versus Unsupervised Binary-Learning by Feedforward Neural Networks”, *Machine Learning*, v. 42, n. 1, pp. 97–122, January 2001. ISSN: 1573-0565. doi: 10.1023/A:1007660820062.
- [123] BREIPOHL, A. M. *Probabilistic Systems Analysis: An Introduction to Probabilistic Models, Decisions, and Applications of Random Processes*. Wiley, 1990.
- [124] STEELE, J. M. *Stochastic Calculus and Financial Applications*. Springer-Verlag New York, 2001.
- [125] JARROW, R., PROTTER, P. “A short history of stochastic integration and mathematical finance: the early years, 1880–1970”. v. Volume 45, Institute of Mathematical Statistics, 2004.
- [126] STIRZAKER, D. “Advice to hedgehogs, or, constants can vary”, *The Mathematical Gazette*, v. 84, n. 500, pp. 197–210, 007 2000. doi: 10.2307/3621649. Disponível em: <<https://www.cambridge.org/core/article/div-class-title-advice-to-hedgehogs-or-constants-can-vary-div/1D63475577200FAD00403086193EAE0E>>.
- [127] YATES, R. D., GOODMAN, D. J. *Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers*. Wiley, 2004.
- [128] DITZLER, G., ROVERI, M., ALIPPI, C., et al. “Learning in Nonstationary Environments: A Survey”, *IEEE Computational Intelligence Magazine*, v. 10, n. 4, pp. 12–25, November 2015.

- [129] GAMA, J., ZLIOBAITE, I., BIFET, A., et al. “A Survey on Concept Drift Adaptation”, *ACM Comput. Surv.*, v. 46, n. 4, pp. 44:1–44:37, mar. 2014. ISSN: 0360-0300. doi: 10.1145/2523813. Disponível em: <<http://doi.acm.org/10.1145/2523813>>.
- [130] WIDYANTORO, D. H., IOERGER, T. R., YEN, J. “An Adaptive Algorithm for Learning Changes in User Interests”. In: *Proceedings of the Eighth International Conference on Information and Knowledge Management, CIKM '99*, pp. 405–412, New York, NY, USA, 1999. ACM. ISBN: 1-58113-146-1. doi: 10.1145/319950.323230. Disponível em: <<http://doi.acm.org/10.1145/319950.323230>>.
- [131] SHUMWAY, R. H., STOFFER, D. S. *Time Series Analysis and Its Applications*. Springer-Verlag New York, 2011.
- [132] KULLBACK, S., LEIBLER, R. A. “On Information and Sufficiency”, *Ann. Math. Statist.*, 1951.
- [133] WILLIAMS, C. K. I., SEEGER, M. “Using the Nyström Method to Speed Up Kernel Machines”. In: Leen, T. K., Dietterich, T. G., Tresp, V. (Eds.), *Advances in Neural Information Processing Systems 13*, MIT Press, pp. 682–688, 2001. Disponível em: <<http://papers.nips.cc/paper/1866-using-the-nystrom-method-to-speed-up-kernel-machines.pdf>>.
- [134] SCHÖLKOPF, B., MIKA, S., SMOLA, A. J., et al. “Kernel PCA Pattern Reconstruction via Approximate Pre-Images”. cap. 10 in *ICANN'98. Perspectives in Neural Computing*, Springer, London, 1998.
- [135] DOS ANJOS, A., TORRES, R., SEIXAS, J., et al. “Neural triggering system operating on high resolution calorimetry information”, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, v. 559, n. 1, pp. 134 – 138, 2006. ISSN: 0168-9002. doi: <http://dx.doi.org/10.1016/j.nima.2005.11.132>. Disponível em: <[//www.sciencedirect.com/science/article/pii/S0168900205022552](http://www.sciencedirect.com/science/article/pii/S0168900205022552)>. Proceedings of the X International Workshop on Advanced Computing and Analysis Techniques in Physics Research-A-CAT 05X International Workshop on Advanced Computing and Analysis Techniques.
- [136] RUMELHART, D. E., HINTON, G. E., WILLIAMS, R. J. “Learning representations by back-propagating errors”, *Nature*, v. 323, n. 6088, pp. 533–536, 10 1986. Disponível em: <<http://dx.doi.org/10.1038/323533a0>>.

- [137] CYBENKO, G. “Approximation by superpositions of a sigmoidal function”, *Mathematics of Control, Signals and Systems*, v. 2, n. 4, pp. 303–314, 1989. doi: 10.1007/BF02551274. Disponível em: <<http://dx.doi.org/10.1007/BF02551274>>.
- [138] SCHMIDHUBER, J. “Deep Learning in Neural Networks: An Overview”, *ArXiv e-prints*, 2014.
- [139] DENG, L., YU, D. *Deep Learning: Methods and Applications*. Foundations and trends in signal processing. Now Publishers, 2014. ISBN: 9781601988140. Disponível em: <<https://books.google.com.br/books?id=46qNoAEACAAJ>>.
- [140] LECUN, Y., BENGIO, Y., HINTON, G. “Deep learning”, *Nature*, v. 521, n. 7553, pp. 436–444, 05 2015. Disponível em: <<http://dx.doi.org/10.1038/nature14539>>.
- [141] LECUN, Y., BOTTOU, L., BENGIO, Y., et al. “Gradient-based learning applied to document recognition”, *Proceedings of the IEEE*, 1998.
- [142] LAWRENCE, S., GILES, C. L., TSOI, A. C., et al. “Face recognition: a convolutional neural-network approach”, *IEEE Transactions on Neural Networks*, 1997.
- [143] XIE, J., LU, H., NAN, D., et al. “Sparse Deep Belief Net for Handwritten Digits Classification”. In: *Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence: Part I, AICI’10*, pp. 71–78, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN: 3-642-16529-X, 978-3-642-16529-0. Disponível em: <<http://dl.acm.org/citation.cfm?id=1927730.1927743>>.
- [144] LU, S., LIU, H., LI, C. “Manifold Regularized Stacked Autoencoder for Feature Learning”. In: *2015 IEEE International Conference on Systems, Man, and Cybernetics*, 2015.
- [145] WEI, T. H., CHEN, J. C. “Video Super-Resolution via Convolution Neural Network”. In: *2016 3rd International Conference on Green Technology and Sustainable Development (GTSD)*, 2016.
- [146] MATSUGU, M., MORI, K., MITARI, Y., et al. “Subject Independent Facial Expression Recognition with Robust Face Detection Using a Convolutional Neural Network”, *Neural Netw.*, v. 16, n. 5-6, pp. 555–559, jun. 2003. ISSN: 0893-6080. doi: 10.1016/S0893-6080(03)00115-1. Disponível em: <[http://dx.doi.org/10.1016/S0893-6080\(03\)00115-1](http://dx.doi.org/10.1016/S0893-6080(03)00115-1)>.

- [147] OORD, A. V. D., DIELEMAN, S., SCHRAUWEN, B. “Deep Content-based Music Recommendation”. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS’13*, pp. 2643–2651, USA, 2013. Curran Associates Inc. Disponível em: <<http://dl.acm.org/citation.cfm?id=2999792.2999907>>.
- [148] HOCHREITER, S., SCHMIDHUBER, J. “Long Short-Term Memory”, *Neural Computation*, v. 9, n. 8, pp. 1735–1780, 2017/01/12 1997. doi: 10.1162/neco.1997.9.8.1735. Disponível em: <<http://dx.doi.org/10.1162/neco.1997.9.8.1735>>.
- [149] LI, L.-S., GAN, S.-J., YIN, X.-D. “Feedback recurrent neural network-based embedded vector and its application in topic model”, *EURASIP Journal on Embedded Systems*, v. 2017, n. 1, pp. 5, 2016. ISSN: 1687-3963. doi: 10.1186/s13639-016-0038-6. Disponível em: <<http://dx.doi.org/10.1186/s13639-016-0038-6>>.
- [150] GREFF, K., SRIVASTAVA, R. K., KOUTNÍK, J., et al. “LSTM: A Search Space Odyssey”, *ArXiv e-prints*, 2015.
- [151] HINTON, G. E., OSINDERO, S., TEH, Y.-W. “A Fast Learning Algorithm for Deep Belief Nets”, *Neural Comput.*, v. 18, n. 7, pp. 1527–1554, jul. 2006. ISSN: 0899-7667. doi: 10.1162/neco.2006.18.7.1527. Disponível em: <<http://dx.doi.org/10.1162/neco.2006.18.7.1527>>.
- [152] GOLOVKO, V., KROSHCHANKA, A., TREADWELL, D. “The Nature of Unsupervised Learning in Deep Neural Networks: A New Understanding and Novel Approach”, *Opt. Mem. Neural Netw.*, v. 25, n. 3, pp. 127–141, jul. 2016. ISSN: 1060-992X. doi: 10.3103/S1060992X16030073. Disponível em: <<http://dx.doi.org/10.3103/S1060992X16030073>>.
- [153] MONTUFAR, G., AY, N. “Refinements of Universal Approximation Results for Deep Belief Networks and Restricted Boltzmann Machines”, *Neural Comput.*, v. 23, n. 5, pp. 1306–1319, maio 2011. ISSN: 0899-7667. doi: 10.1162/NECO_a_00113. Disponível em: <http://dx.doi.org/10.1162/NECO_a_00113>.
- [154] VINCENT, P., LAROCHELLE, H., LAJOIE, I., et al. “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion”, *J. Mach. Learn. Res.*, v. 11, pp. 3371–3408, dez. 2010. ISSN: 1532-4435. Disponível em: <<http://dl.acm.org/citation.cfm?id=1756006.1953039>>.

- [155] XU, Q., ZHANG, C., ZHANG, L., et al. “The Learning Effect of Different Hidden Layers Stacked Autoencoder”. In: *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2016.
- [156] XU, Q., ZHANG, C., ZHANG, L., et al. “The Learning Effect of Different Hidden Layers Stacked Autoencoder”. In: *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2016.
- [157] LIANG, J., LIU, R. “Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network”. In: *2015 8th International Congress on Image and Signal Processing (CISP)*, 2015.
- [158] CORTES, C., VAPNIK, V. “Support-Vector Networks”, *Machine Learning*, v. 20, n. 3, pp. 273–297, 1995. ISSN: 1573-0565. doi: 10.1023/A:1022627411411. Disponível em: <<http://dx.doi.org/10.1023/A:1022627411411>>.
- [159] LECUN, Y. “THE MNIST DATABASE of handwritten digits” . .
- [160] DECOSTE, D., SCHÖLKOPF, B. “Training invariant support vector machines”, *Machine learning*, v. 46, n. 1-3, pp. 161–190, 2002.
- [161] MITCHELL, T. M. *Machine Learning*. 1 ed. New York, NY, USA, McGraw-Hill, Inc., 1997. ISBN: 0070428077, 9780070428072.
- [162] VAPNIK, V., LERNER, A. “Pattern Recognition using Generalized Portrait Method”, *Automation and Remote Control*, v. 24, 1963.
- [163] ROSENBLATT, F. *The Perceptron: A Perceiving and Recognizing Automaton (Project PARA)*. Relatório técnico, Cornell Aeronautical Laboratory, 1957.
- [164] BOSER, B. E., GUYON, I. M., VAPNIK, V. N. “A Training Algorithm for Optimal Margin Classifiers”. In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92*, pp. 144–152, New York, NY, USA, 1992. ACM. ISBN: 0-89791-497-X. doi: 10.1145/130385.130401. Disponível em: <<http://doi.acm.org/10.1145/130385.130401>>.
- [165] ZIMEK, A., SCHUBERT, E., KRIEGEL, H.-P. “A Survey on Unsupervised Outlier Detection in High-dimensional Numerical Data”, *Stat. Anal. Data Min.*, v. 5, n. 5, pp. 363–387, out. 2012. ISSN: 1932-1864. doi: 10.

1002/sam.11161. Disponível em: <<http://dx.doi.org/10.1002/sam.11161>>.

- [166] CHEN, S., MONTGOMERY, J., BOLUFÉ-RÖHLER, A. “Some Measurements on the Effects of the Curse of Dimensionality”. In: *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation, GECCO Comp '14*, pp. 1447–1448, New York, NY, USA, 2014. ACM. ISBN: 978-1-4503-2881-4. doi: 10.1145/2598394.2602271. Disponível em: <<http://doi.acm.org/10.1145/2598394.2602271>>.
- [167] NANAS, N., VAVALIS, M., DE ROECK, A. “A Network-based Model for High-dimensional Information Filtering”. In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '10*, pp. 202–209, New York, NY, USA, 2010. ACM. ISBN: 978-1-4503-0153-4. doi: 10.1145/1835449.1835485. Disponível em: <<http://doi.acm.org/10.1145/1835449.1835485>>.

Apêndice A

Produção Científica

Este apêndice visa mostrar a produção de artigos científicos durante o período de desenvolvimento da tese. A lista abaixo está definido de acordo com o veículo de publicação

- Trabalhos em periódicos

- Trabalhos Submetidos

1. MOURA JUNIOR, N. N.; T. C. Xavier ; N. N. Moura ; J. M. Seixas. A Kernel-based Novelty Detection for Passive Sonar Signal Classification. *NEUROCOMPUTING*, 2018. Abstract: In naval warfare operations, several techniques have been developed for passive sonar signal detection and classification. Sonar systems operate under very noisy conditions and, in modern warfare scenario, it might be necessary to classify ships whose signal signatures were not available for the classifier training process. Kernel-based algorithms efficiently access high-order statistics and, thus, have successfully applied in different preprocessing chains envisaging classification. This is the case of kernel PCA (kPCA), which is a Principal Component Analysis (PCA) extension. For general novelty detection, kernels have also been used thru Support Vector Machine (SVM) techniques. The one-class SVM is quite suitable, as it detects events that were generated from the same generating function estimated along the training process. This paper develops a one-class SVM model, which is fed from kPCA transformed data for detecting novelties in passive sonar systems. For comparison purposes, PCA and its other non-linear extension (NLPCA) are both considered in the preprocessing step. Results are from experimental data and novelty detection fluctuations are assessed by considering a 4-class classification in which one class at a time is taken as the novelty. Comparison with another

work, which is based on neural networks and used the same data set is also addressed.

– Trabalhos em Submissão

1. N. N. Moura ; MOURA JUNIOR, N. N. ; J. M. Seixas. Independent Component Analysis for Reducing Cross-interference in Passive Sonar Signals. 2018.
2. T. C. Xavier ; MOURA JUNIOR, N. N.; N. N. Moura ; J. M. Seixas. A Fusion of Neural Network Models for Identifying Novelty in Passive Sonar Signal Classification. 2018.

• Trabalhos em Conferências Internacionais

– Trabalhos Submetidos

1. MOURA JUNIOR, N. N.; J. M. Seixas . Novelty Detection in Passive Sonar Systems using a Kernel Approach. 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery. Guilin, China, 2017.
2. MOURA JUNIOR, N. N.; J. M. Seixas . Novelty Detection in Passive SONAR Systems using Support Vector Machines. 2nd LA-CCI (Latin American) Congress on Computational Intelligence, Curitiba, Brazil, 2015

– Trabalhos em Submissão

1. V. S. Mello ; MOURA JUNIOR, N. N. ; J. M. Seixas . Novelty Detection in Passive Sonar Systems using Stacked AutoEncoders, IEEE World Congress on Computational Intelligence, Rio de Janeiro, Brazil, 2018

• Trabalhos em Conferências Nacionais

– Trabalhos Submetidos

1. MOURA JUNIOR, N. N.; V. S. Mello ; L. A. Cerqueira ; J. M. Seixas . Classificação de sinais de Sonar Passivo utilizando Stacked AutoEncoders. Congresso Brasileiro de Inteligência Computacional. Niterói, Brasil, 2017.
2. MOURA JUNIOR, N. N.; J. M. Seixas . Avanços na detecção de novidade em sinais de sonar passivo utilizando máquinas de vector suporte. XII Encontro em Tecnologia Acústica Submarina, Rio de Janeiro, Brasil, 2016.

3. L. A. Cerqueira ; MOURA JUNIOR, N. N. ; J. M. Seixas . Aplicação de redes neurais profundas à sinais de sonar passivo. XII Encontro em Tecnologia Acústica Submarina, Rio de Janeiro, Brasil, 2016.
4. P. O. D. Proença ; P. H. Sabino ; MOURA JUNIOR, N. N. ; J. M. Seixas . Componentes Principais não lineares para a identificação neural de novas classes de navios. In: XII Encontro de Tecnologia em Acústica Submarina, 2016, Rio de Janeiro. XII Encontro de Tecnologia em Acústica Submarina, Rio de Janeiro, Brasil, 2016.
5. MOURA JUNIOR, N. N.; J. M. Seixas . Avanços na análise de estacionaridade em sinais de sonar passivo. XII Encontro em Tecnologia Acústica Submarina, Rio de Janeiro, Brasil, 2016.
6. FERNANDES, N. F. ; MOURA JUNIOR, N. N. ; J. M. Seixas . Componentes Principais de Discriminação para a identificação de novas classes de navios. In: XII Encontro de Tecnologia em Acústica Submarina, 2016, Rio de Janeiro. XII Encontro de Tecnologia em Acústica Submarina, Rio de Janeiro, Brasil, 2016.
7. Fernandes, N. F. ; Moura Junior, N. N. ; J. M. Seixas. Redes neurais para classificação de sinais de sonar passivo. Reunião anual da FeSBE, São Paulo, Brasil, 2015.
8. MOURA JUNIOR, N. N.; J. M. Seixas . Análise de Estacionaridade em Sinais de Sonar Passivo. XI Encontro em Tecnologia Acústica Submarina, Rio de Janeiro, Brasil, 2014.

Apêndice B

Aprendizado de Máquina

Este apêndice tem por objetivo detalhar de maneira direta alguns dos modelos de aprendizado de máquina utilizados durante a tese.

B.1 Redes Neurais Artificiais

Redes Neurais Artificiais são uma família de técnicas de aprendizado de máquina que se baseia em um modelo computacional inspirado na estrutura neural de seres vivos que possuem cognição [26]. Em uma rede neural biológica, o conhecimento se armazena nas sinapses que conectam os neurônios, que são as unidades básicas de processamento. A cada excitação, a informação navega pela rede sináptica como impulsos elétricos conduzidos pelos neurônios e que tem a sua transmissão facilitada pelos chamados neurotransmissores. À medida que a quantidade de neurotransmissores depositada em uma sinapse aumenta, mais fácil é a transmissão do sinal pela mesma. À medida que a quantidade de neurotransmissores depositada em uma sinapse diminui, a transmissão de um sinal pela mesma pode ser dificultada.

Uma rede neural artificial é composta por diversas unidades básicas de processamento conectadas. Cada unidade básica de processamento possui poder de processamento relativamente simples e é conectada as demais como pode ser visto na figura B.1. A quantidade de neurotransmissores depositada pode ser modelada pelo valor do peso da sinapse artificial.

O treinamento desse modelo consiste na minimização de uma função custo definida pelo usuário. Geralmente, as funções custo são funções de dissimilaridade entre o alvo e a saída do modelo neural. Essa dissimilaridade pode ser vista como um erro entre a saída e o alvo. Para a minimização deste erro, o algoritmo de retropropagação do erro foi proposto em [136] e pode ser visto na equação B.1. Nesta equação, ω representa os pesos de uma camada de uma rede neural artificial, sendo que um dos valores dos pesos pode ser visto como ω_{ij} , t representa uma época de treinamento, α representa a taxa de aprendizado do modelo e $F_{custo}(\omega)$ é a função

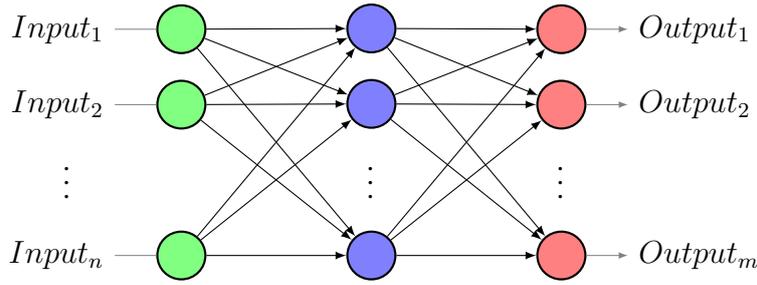


Figura B.1: Rede Neural Artificial

custo do modelo ou a função de dissimilaridade entre o alvo e a saída. A medida que o treinamento ocorre os pesos são atualizados para que a saída se torne menos dissimilar ao alvo até que algum dos critérios de parada do treinamento seja obtido.

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \left(\frac{\partial F_{custo}(\omega)}{\partial \omega_{ij}} \right) \quad (\text{B.1})$$

Os modelos neurais alcançaram grande popularidade nos anos 90 devido ao teorema da aproximação universal (*Universal Approximation Theorem*) [137], que diz que qualquer função contínua pode ser aproximada por um somatório de funções não-lineares ponderadas. O teorema da aproximação universal somado ao treinamento com retropropagação do erro, permitiu que redes neurais artificiais fossem amplamente difundidas e pudessem estimar funções complexas como funções geratriz de dados com distribuições complexas e, assim, modelos neurais puderam ser utilizados para a solução do problemas complexos.

Deep Learning

Deep Learning (também conhecido como aprendizagem profunda ou aprendizagem hierárquica) pode ser visto como uma derivação de redes neurais no ramo de aprendizado de máquina. Técnicas de deep learning são baseadas em um conjunto de algoritmos que tentam modelar altos níveis de abstrações extraídas dos dados [27, 138–140]. Como nos modelos neurais, os modelos são baseados em estruturas biológicas, mas ao contrário dos anteriores, as estruturas que inspiraram os modelos de deep learning foram estruturas especializadas, como por exemplo: neurônios que se conectam às células da retina ou neurônios que se conectam às células do ouvido humano. A ideia básica é: a cada camada, a informação transmitida possui um nível de abstração maior e com isso, a tomada de decisão pode se dar em um nível de abstração mais alto.

Algumas estruturas de aprendizado de máquina podem ser classificadas como técnicas de deep learning. Para que uma estrutura seja considerada de deep learning, a mesma deve atender a alguns critérios:

- ser estruturada em uma cascata de muitas camadas de unidades de processamento não-lineares para extração de características e de transformação. Cada camada sucessiva usa a saída da camada anterior como entrada. As estruturas podem possuir aprendizado supervisionado ou não supervisionado.
- ser baseado no aprendizado em múltiplos níveis de características e/ou representações dos dados. Os diferentes níveis se estruturam hierarquicamente formando assim a chamada representação hierárquica
- o aprendizado da estrutura deve tirar vantagem dos diversos níveis que compõem o modelo.

Dentre as estruturas mais difundidas de deep learning que podem ser citadas temos: CNN (*Convolutional Neural Networks* ou Redes Neurais Convolucionais) [141, 142], LSTM (*Long Short-Term Memory*) [ref], DBN (*Deep Belief Networks*) [143] e SAE (*Stacked Auto-Encoders*) [144].

CNN são redes neurais artificiais que se organizam de maneira inspirada em neurônios do córtex visual animal [141, 142, 145]. Neurônios do córtex visual respondem a estímulos em uma região restrita no campo visual. Cada região é, usualmente, chamada de campo receptivo. Campos receptivos de diferentes neurônios se sobrepõem para a formação do campo visual completo. A resposta de cada neurônio a estímulos dentro do seu campo receptivo pode ser aproximada à operação de convolução. Esse fato inspira as CNN [146] e utilizam diversas camadas intermediárias para minimizar processos auxiliares de pré-processamento [147].

LSTM é uma das estruturas mais difundidas das chamadas redes neurais recursivas (RNN) [148–150]. As estruturas de redes neurais recursivas aplicam recursivamente o mesmo conjunto de pesos a um conjunto de dados. Assim sendo, estas redes neurais conseguem acessar informações contidas nos dados de maneira sequencial. As RNN têm sido bem-sucedidas em aplicações com processamento de linguagem natural e classificação estruturada de dados.

DBN são modelos neurais compostos por múltiplas camadas de variáveis latentes (“unidades ocultas”), com conexões entre as camadas, mas não entre unidades pertencentes a uma única camada [143, 151–153].

SAE são estruturas motivadas pelo conceito de uma representação em altos níveis de abstração [144, 154–157]. Ou seja, quanto mais camadas de processamento de dados, mais informações de alto nível os dados podem oferecer. Por exemplo, para um classificador, uma boa representação dos dados de entrada pode maximizar o potencial discriminatório dos mesmos e elevar a eficiência final do treinamento.

Para se obter essa representação, uma sequência de camadas não-lineares forma a estrutura de um SAE. O treinamento é desenvolvido para a minimização do erro

de reconstrução de cada uma das camadas. Ou seja, a cada camada os parâmetros são extraídos e repete-se o processo para todas as camadas.

B.2 Support Vector Machines

As SVM são modelos de aprendizado de máquina que foram largamente utilizados durante as décadas de 1990 e 2000 [25, 158–160]. Até o surgimento das técnicas de deep learning, treinamentos de SVM e suas associações com técnicas de *ensemble* dominavam o cenário de aprendizado de máquina [161].

A ideia das máquinas de vector suporte é, inicialmente, muito simples e consiste em encontrar um plano de decisão que separe duas classes de maneira a maximizar a distância entre os mais próximos de cada uma das classes e plano de separação. Para fazer isso, um treinamento supervisionado é desenvolvido, onde cada um dos eventos possui a classe a qual pertence.

Suponha que eventos pertencentes a um banco de dados formam duas classes distintas. Sendo assim, o objetivo de uma máquina de vector suporte é definir um plano de decisão que classifique corretamente os eventos no conjunto de treinamento e que possa ser generalizado para um conjunto de testes previamente definido. Inicialmente, para SVM linear, o plano de decisão é linear.

Geralmente, temos uma quantidade de dados que permite definir vários planos de decisão lineares que poderiam separar os dados em duas classes de maneira eficiente. Em [162], Vapnik e Lerner propõem um método de definição de planos de decisão lineares que maximizam a distância do plano ao ponto dos dados mais próximos. Se tal plano de decisão existe, o mesmo é conhecido como plano de margem máxima e o classificador linear que ele define é chamado de classificador linear de margem máxima, como pode ser visto na figura B.2.

Como pode ser visto na figura B.2, o plano de decisão é definido como sendo $w \cdot x - b = 0$. Assim sendo, qualquer ponto x' que se encontrar acima desse plano será classificado como sendo da classe “+1” e se o ponto x' se encontrar abaixo desse plano será classificado como da classe “-1”. Além disso, para maximizar o potencial discriminatório do classificador, o algoritmo proposto define outros dois planos auxiliares e que são definidos pelas equações $w \cdot x - b = -1$ e $w \cdot x - b = +1$ e definem os pontos de cada uma das classes que estão mais próximos do plano de decisão. Esses pontos são chamados de vetores suporte e são representados por dois círculos na figura e a distância mínima permitida para os planos secundários, que no exemplo foi definida como 1, pode ser definida durante o treinamento através de um parâmetro de treinamento chamado de ρ .

No caso de classes linearmente separáveis, o algoritmo proposto em [162] funciona perfeitamente, assim como o perceptron idealizado por Rosenblatt em [163]. As duas

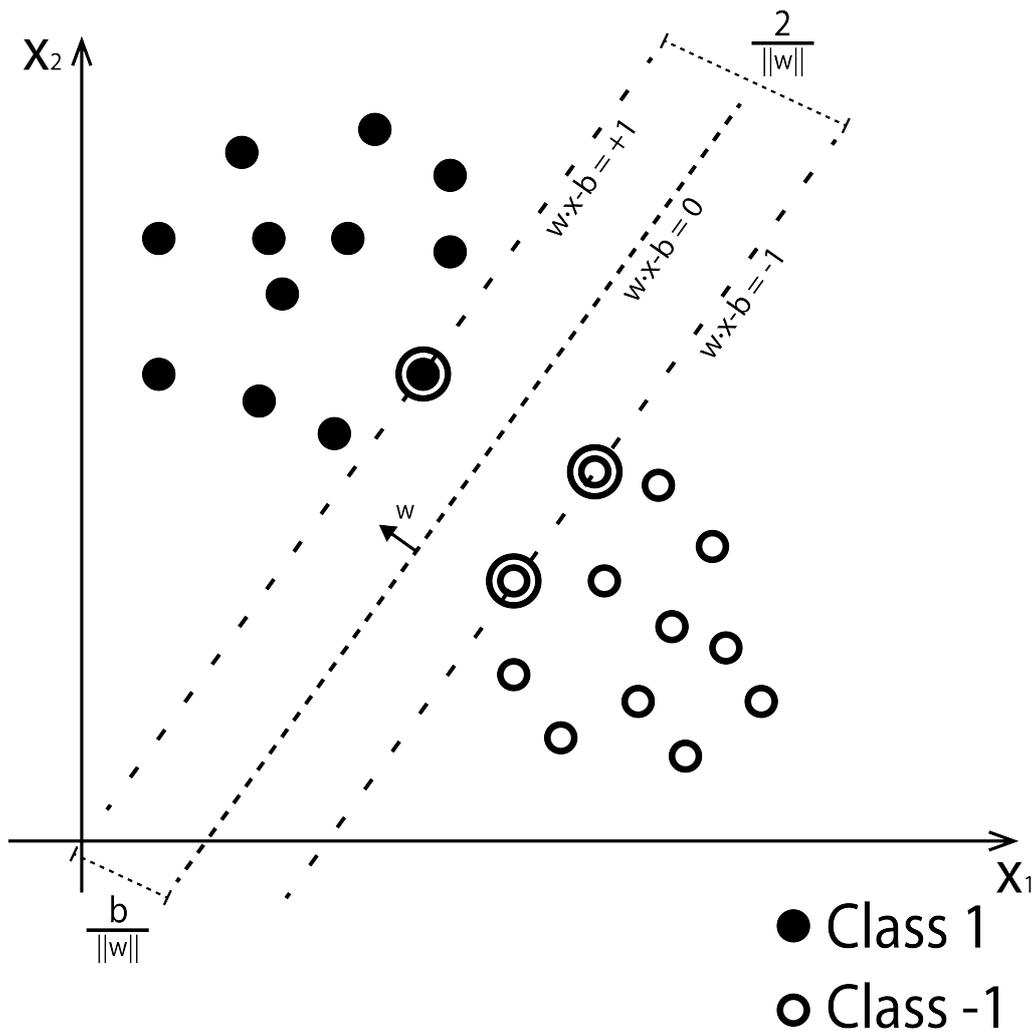


Figura B.2: Exemplo de um classificador linear de margem máxima

técnicas apresentam falhas na definição de uma regra de classificação para classes não-linearmente separáveis. Para isso, algumas técnicas foram propostas, tanto para SVM, quanto para redes neurais. No caso de redes neurais, a MLP foi implementada com o algoritmo de treinamento *backpropagation* e outros que permitiram que as redes pudessem fazer classificações mais complexas. No caso de SVM, inicialmente, a proposta foi de adicionar pontos de relaxamento na margem de decisão.

Ou seja, o plano de decisão, que anteriormente seria linear, se torna não-linear em alguns pontos. A essa margem que permite alguns pontos de relaxamento, foi dado o nome de *soft-margin*. Na figura B.3 é mostrada uma *soft-margin* com uma variável de relaxamento. A margem de classificação se afasta do ponto ótimo de minimização para permitir minimizar o erro de classificação. No caso extremo, o treinamento do classificador pode gerar tantas variáveis de relaxamento quanto o número de pontos, assim sendo, o classificador se torna muito especializado e perde generalização. Para evitar o processo de *overtraining*, geralmente, a criação de pontos de relaxamento é monitorada durante o treinamento e são utilizadas diversas regularizações durante

o processo.

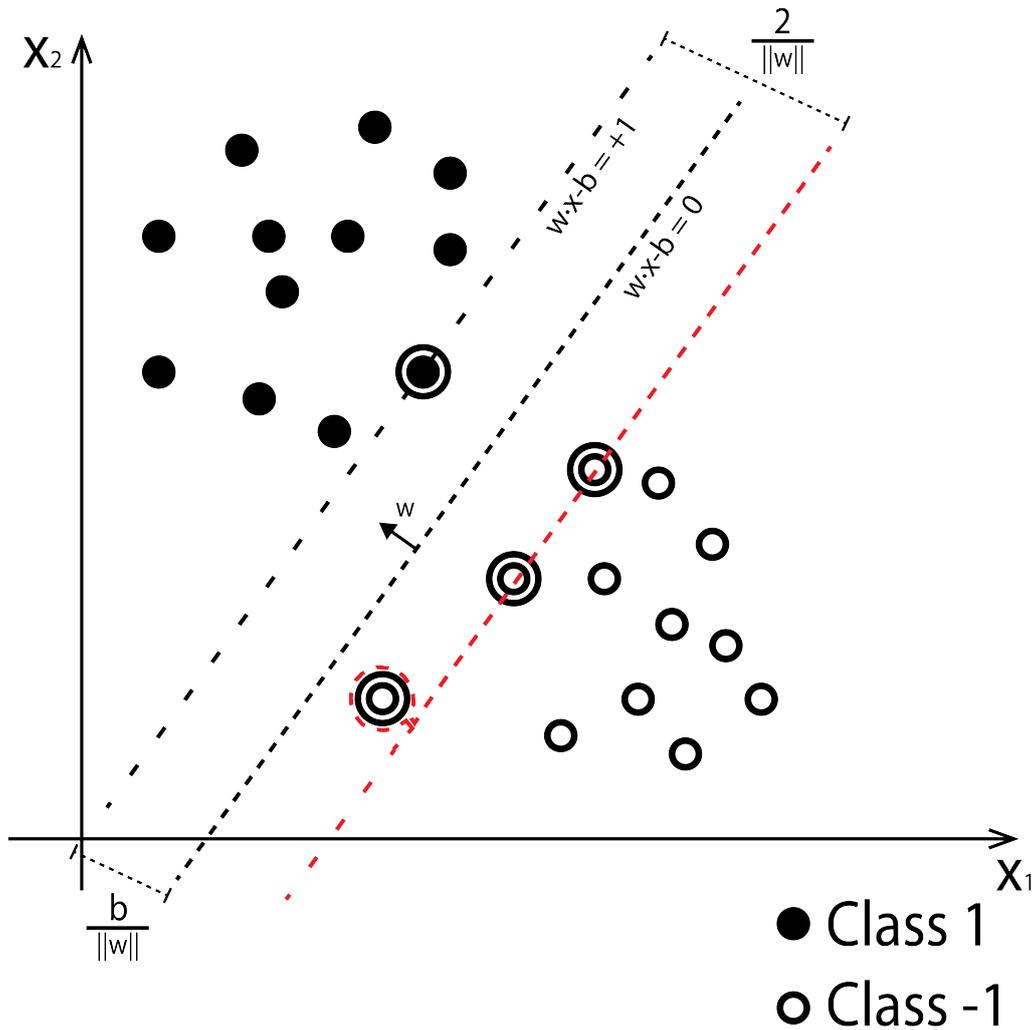


Figura B.3: Exemplo de um classificador linear de margem máxima com uma variável de relaxamento

Mesmo com a modificação de um classificador de margem máxima para um classificador *soft-margin*, alguns problemas de classificação exigiam ainda mais complexidade dos classificadores [25]. Então, em [164], foi proposta uma maneira de adaptar o algoritmo original de margem máxima para a criação de classificadores não-lineares utilizando o chamado truque de *kernel* ou *kernel trick* (equação B.2).

$$k(x, y) = \langle \varphi(x), \varphi(y) \rangle \quad (\text{B.2})$$

Por vezes, a tomada de decisão no espaço de dados pode gerar classificadores não-lineares muito complexos e que não obtenham um nível de eficiência satisfatório, pois os dados que podem se desenvolver em PDF tem um potencial discriminatório baixo. O truque de *kernel* é uma ferramenta que tem sido bastante utilizada para evitar este fato.

O processo de tomada de decisão não é feito no espaço de dados, mas sim em um mapeamento dos dados em um espaço gerado pelos kernels, chamado de espaço de características (*feature space*), onde um número de dimensões maior ou, potencialmente, infinito é obtido. Com esse número maior de dimensões, as distribuições dos dados tendem a obter um potencial discriminatório maior e a tomada de decisão pode ser feita com um classificador mais simples.

A expansão no número de dimensões pode gerar a chamada maldição da dimensionalidade [165–167]. Esse é um problema que afeta consideravelmente as redes neurais e outras técnicas de aprendizado de máquina mas não afeta SVM pois o algoritmo de treinamento de SVM é um algoritmo convexo. Ou seja, sempre obtém o ponto ótimo de minimização a cada treinamento com os dados em questão.